Robust, Gain-Scheduled Control of Wind Turbines

Østergaard, Kasper Zinck

Publication date:
2008

Document Version
Publisher's PDF, also known as Version of record

Robust, Gain-Scheduled Control of Wind Turbines

Ph.D. Thesis

Kasper Zinck Østergaard

Automation and Control
Department of Electronic Systems
Aalborg University
Fredrik Bajers Vej 7C, 9220 Aalborg East, Denmark

Load, Aerodynamics and Control
Vestas Wind Systems A/S
Alsvej 21, 8900 Randers, Denmark

ISBN: 978-87-90664-33-6
December 2008

This thesis was typeset using LaTeX2e

Printing: Vester Kopi, Denmark

Preface and Acknowledgements

This thesis is submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy at Automation and Control, Department of Electronic Systems, Aalborg University. The research has been carried out from October 2004 to November 2007 in a co-operation between Vestas Wind Systems A/S and Aalborg University in a framework denoted industrial ph.d. This means that the project has been funded partly by Vestas Wind Systems A/S and partly by the Danish Ministry of Science, Technology and Innovation.

I am very grateful to my supervisors Professor Jakob Stoustrup from Aalborg University and ph.d. Per Brath from Vestas Wind Systems A/S. Through many enlightening discussions on theoretical as well as practical issues they have not only shared their extensive knowledge but also provided the motivation and support throughout the study.

Also I would like to express my gratitude to Vestas Wind Systems A/S and the Danish Ministry of Science, Technology and Development. Without their financial support, it would not have been possible to perform the study resulting in this thesis.

A special thanks to Professor Carsten Scherer from Delft Center for Systems and Control, The Netherlands, for many interesting discussions during my five month stay in 2006.

Furthermore I would like to thank my colleagues at both Aalborg University and Vestas Wind Systems A/S for many interesting discussions throughout the study.

Finally and most importantly I would like to thank my wife, Jeanette, for all her patience and support. Without this support from home, the great results in the thesis could not have been possible.

Abstract

The control of wind turbines is a very challenging task because it requires multi-objective control that needs to take into account effects like structural fatigue, power production/quality, and slip in the generator. In this thesis, a number of linear parameter varying (LPV) design methods are investigated for use with medium to large scale systems and a method has been found to provide a numerically reliable computation of the controller.

The thesis begins with an introduction to the control of wind turbines and provides an overview of the different methods that have been applied. This introduction is followed by a general introduction to LPV systems and the base for the available design methods.

The application of the LPV framework to the control of nonlinear systems requires a method for describing the system by a LPV model. For doing this there are two main directions: Jacobian-based linearisation or quasi-LPV models through a substitution of nonlinearities by parameters. In the latter method the scheduling parameter is a function of the state vector which can introduce conservatism. This issue is investigated in the thesis leading to a proposed procedure with focus on frozen parameter dynamics.

The numerics in the design algorithms is one of the key issues for getting the methods working for medium to large scale systems. For the associated optimisation problem, a grid-based method has been found promising and it has been found that the choice of state space realisation plays a very important role and a simulation based method has been developed for choosing a proper realisation for controller design. For closed loop analysis a Gramian-based method has been found promising.

The construction of the LPV controller can be very challenging from a numerical point of view. Several algorithms have been investigated and a method based on a classical result from $\mathcal{H}_\infty$ control has been found superior from a numerical point of view. This method does not directly produce a parameterised controller and another method (which can calculate a parameterised controller) has been modified to enhance numerical performance. The numerical performance of this modified controller is still not as strong as the first method, which makes the choice of construction algorithm a trade-off between numerical performance and calculating a parameterised controller.

A graphical design tool has been developed on the basis of the numerical findings for conditioning of algorithms for the design of LPV controllers. This tool provides an easy user interface which makes it possible for non-experts to design LPV controllers without worrying about matrix inequalities, construction algorithms, etc. The designer only needs to enter the model, weighting functions, tolerated rate of variation and basis functions for the storage (Lyapunov) functions. Then the controller can be designed automatically by pressing a few buttons.

Large scale design problems can be very difficult to solve from a numerical point of view. For these systems the tool provides a number of numerical tuning handles with

which the numerical performance can be adjusted. The use of these tuning handles requires a deeper understanding in the synthesis LMIs and construction algorithm because the handles are implemented through upper/lower bounds on the matrix inequalities and design variables.

The design tool has been applied to the design of LPV controllers for wind turbines. It has been demonstrated that the switching between a partial load controller and a full load controller is not necessary because a single LPV controller can handle both operating conditions. In fact the fatigue loads can be reduced in several structural components throughout the whole range of operating conditions by using this approach in comparison to classical means.

When designing gain scheduled controller the rate of variation of the scheduling variable can be a critical issue. The classical gain scheduling method assumes no parameter variations whereas many LPV methods assume arbitrary fast parameter variations. In this thesis it is demonstrated that for the application of control of wind turbines a great performance improvement can be obtained by taking a bound on the rate of variation into account.

Synopsis

Vindmølleregulering er meget omfattende eftersom der er mange modstridende krav såsom strukturel udmattelse, el-produktion, el-kvalitet og variation i generator hastighed fra synkron hastighed. I denne afhandling vil forskellige lineær parameter varierende (LPV) design blive undersøgt for mellem og stor-skala systemer. Ud fra denne undersøgelse er der fundet en metode som giver numerisk pålidelig beregning af regulatoren.

Den første del af afhandlingen beskriver hvorfor vi regulerer vindmøller og giver et overblik over tidligere anvendte metoder. Denne introduktion er fulgt af en introduktion til de fundamentale dele af LPV regulering.

For at kunne anvende LPV paradigmet til regulering af ulineære systemer er det nødvendigt at kunne beskrive det ulineære system gennem LPV modeller. Dette gøres gennem en af de to følgende principper: Jacobi linearisering eller kvasi-LPV modeller hvor ulineariterne udskiftes med parametre. For det sidstnævnte princip er skeduleringsvariablen en funktion af tilstandsvariablen hvilket kan gøre designet konservativt. Denne konservatisme er undersøgt i afhandlingen, hvilket har ført til en fremgangsmåde som har fokus på modellens dynamik ved frosne parametre.

Design algoritmens numerik er et af de vigtigste fokusområder for at få LPV metoderne til at fungere for mellem og stor-skala systemer. Vedrørende det associerede optimeringsproblem er det blevet konkluderet fra et numerisk synspunkt at den bedste metode bygger på et princip hvor optimeringsproblemet løses i et fintmasket net. Derudover har undersøgelser i afhandlingen vist at valget tilstands realisation spiller en vigtig rolle og en simulerings-baseret metode er blevet udviklet til at vælge en god realisation. For lukket-sløjfe analyse har en tilsvarende undersøgelse vist at en Gram-metode til valget af realisation ser ud til at fungere.

Det kan være en særdeles udfordrende, fra en numerisk synsvinkel, at konstruere regulatoren. Flere algoritmer har derfor været undersøgt og det er blevet konkluderet at den bedste metode er baseret på et klassisk resultat fra $\mathcal{H}_\infty$. Denne metode kan ikke direkte beregne en parameteriseret regulator og en anden metode (som er i stand til at beregne en parameteriseret regulator) er blevet modificeret til at have en forbedret numerik. Numerikken i denne modificerede algoritme er dog stadig ikke så god som i den førstnævnte algoritme, hvilket betyder at valget af algoritme bliver en afvejning mellem de numeriske krav og beregning af en parameteriseret regulator.

Et grafisk design-værktøj er blevet udviklet og er baseret på de numeriske resultater i den valgte design algoritme. Dette værktøj giver en simpel brugergrænseflade hvilket gør det muligt at designe LPV regulatorer uden at have detailkendskab til metoden. Dvs. designeren behøver ikke bekymre sig om matrix uligheder, algoritmer, m.m. Det eneste designeren behøver at gøre er at specificere model, vægtningsfunktioner, accepteret pa-

rametervariation, og basisfunktioner for lagrings (Lyapunov) funktioner. Herefter bliver regulatoren automatisk beregnet efter tryk på et par taster.

Regulator design for stor-skala systemer kan være temmelig kompliceret ud fra en numerisk synsvinkel. For denne type systemer giver værktøjet et antal numeriske tuningshåndtag hvormed den numeriske ydeevne kan bliver justeret. Det kræver dog en mere detaljeret indsigt i algoritmerne for at benytte disse håndtag effektivt eftersom håndtagene direkte påvirker øvre/nedre begrænsninger for matrix ulighederne og design variablerne

Design værktøjet er blevet anvendt til at designe LPV regulatorer til vindmøller. Det er blevet demonstreret at det ikke længere er nødvendigt at skifte mellem to forskellige regulatorer for henholdsvis delvis last og fuld last. I stedet kan en LPV regulator designes til at dække hele arbejdsområdet og i sammenligning med en klassisk regulator kan denne regulator endda give en reduktion i strukturelle belastninger for hele arbejdsområdet.

Når vi designer spekulerede regulatorer kan hastigheden af parametervariationen bliver afgørende. Klassiske metoder antager at parameteren ikke kan variere i tid mens mange LPV metoder antager vilkårligt hurtige variationer. I denne afhandling er det demonstreret at for regulering af vindmøller kan man opnå en betydelig forbedring i ydeevne ved at tage denne hastighed af parametervariation i betragtning.

Contents

Nomenclature

Highlighting of variables

Syntax highlighting is used to clarify what is design variables and scheduling variables.

RED variables indicate design variables

BLUE variables indicate scheduling variables

Wind turbine specific signals

ρ	Air density
A_r	Rotor swept area
R	Radius of rotor swept area
v	Effective wind speed
c_P	Aerodynamic efficiency at current operating condition
ω_g	Rotational speed of high speed shaft (generator side)
ω_r	Rotational speed of slow speed shaft (rotor side)
P	active power production
β	(Collective) Pitch position of blades, i.e. rotation around longitudinal axis
Q_a	Torque on main shaft from aerodynamics
Q_g	Generator reaction torque

Symbols specific to the analysis and design method

P	Storage function for closed loop system
X, Y	Storage functions used for controller synthesis
$\partial P, \partial X, \partial Y$	Representation used to handle derivative of storage function, see page 33 for its definition
γ	Upper bound on worst case energy amplification
λ	Represents time derivative of scheduling variable independently on specific trajectories

Dynamic models

s	Represents Laplace variable for LTI systems
χ	State vector of closed loop system
x	State vector of open loop system
x_c	State vector of dynamic controller
w	Vector of performance inputs
z	Vector of performance outputs
u	Vector of control inputs
y	Vector of control outputs
δ	Vector of scheduling variables
Δ	Matrix valued scheduling function used in linear fractional representations
$A, B, B_p, C, C_p, D, E, F$	System matrices for open loop system
$\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D}$	System matrices for closed loop system
A_c, B_c, C_c, D_c	Matrices describing controller dynamics

Other symbols

$\dot{}$	Operator for time derivative, e.g. $\dot{x}$ is the time derivative of x
$\hat{}$	Indicates estimated variable, e.g. $\hat{x}$ is an estimate of x
$\bar{}$	Indicates equilibrium values used for linearisation, e.g. $\bar{x}$ is an equilibrium point for x.
$\tilde{}$	Indicates deviations from the equilibrium, e.g. $\tilde{x}$ is the deviation of x from the equilibrium point $barx$.
$\bullet$	This symbol represents components of a large matrix that will not be used in future calculations. The symbol is included for notational simplicity.
$\star$	This symbol represents values that are induced by symmetry and is included to simplify notation. Consider for instance symmetric matrices A and C and a non-symmetric matrix B. Then $$\begin{bmatrix} A+B+(\star) & \star \\ C & 0 \end{bmatrix} = \begin{bmatrix} A+B+B^T & C^T \\ C & 0 \end{bmatrix}$$
<empty>	Empty fields in matrices are included for notational simplicity to indicate zero terms. An example of this is given below $$\begin{bmatrix} A & \\ & B \end{bmatrix} = \begin{bmatrix} A & 0 \\ 0 & B \end{bmatrix}$$
$\prec, \succ$	This symbol indicates comparison of matrices in terms of definiteness, e.g. $A \prec B$ means that $A - B$ is negative definite.

Abbreviations

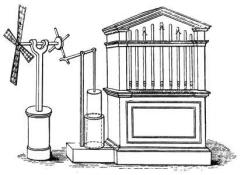
avg.	Average
dam.	Damage
drt.	Drive train
gen.	Generator
GUI	Graphical user interface
LFT	Linear fractional transformation
LMI	Linear matrix inequality
LPV	Linear parameter varying
LQG	Linear quadratic Gaussian
LTI	Linear time invariant
NL	Nonlinear
PID	Proportional integral derivative
QMI	Quadratic matrix inequality
RFC	Rain flow count
spd.	speed
std.	Standard deviation

Chapter 1

Introduction

Contents

1.1. Background



Wind organ. [56]
Renaissance reconstruction.
© Ballantine Books.

Since ancient times the wind has been utilised as a power source to ease labour. One way was to mount sails on ships in order to travel further and with less effort. However, this was not the only use of wind as power source in the ancient world. Heron of Alexandria* described an organ which was powered by the wind. Apparently this might indicate that windmills were used in ancient time, but the Romans never exploited windmills. Probably because there was a rich amount of streams in the most of the empire, encouraging the use of water mills instead. [56]

It is said that the windmill was invented in China more than 2000 years ago, however the earliest actual documentation is from 1219 A.D. In Persia, the use of wind energy for grain grinding was eventually taken up around the 9th century. Unlike the windmill described by Heron of Alexandria these windmills used a vertical shaft with milling stones on top of this shaft. [56]

The idea of windmills arrived in England in 1137 and around this time the design of the windmills changed from having the sails rotating on a vertical shaft to horizontal axis windmills with the driving force caused by lift instead of drag. The principle is actually very much similar to what was described by Heron of Alexandra, only in larger scale and with a different application. [56]

These first European windmills were denoted postmills because they were constructed with the main body of the windmill resting on a post enabling the windmill to be rotated into the wind field. As the windmills became larger in size the weight increased substantially and it became difficult to turn the windmill into the wind. The design was therefore changed so that only part of the windmill needed to be turned in order to get the rotor into the wind field. This new type of wind mill was typically denoted a tower mill and the main part of the mill was stationary and with either wooden, thatched or masonry walls. On the top of it a cap was placed onto which the rotor was mounted, and only the cap with the rotor needed to be moved in order to rotate the windmill into the wind.

Post mill.

Thatched tower mill.

In several centuries the structural parts of the windmills were refined in order to create larger and larger facilities for the two main purposes: milling grain or pumping water. Eventually there was a need for easier control of the windmills in the sense of adapting the sails to the changing wind. In 1772 a Scottish engineer, Andrew Meikle, invented a new type of sail made from a series of shutters which could be opened or closed by a system of levers. Later in 1807 William Cubbit improved the design so that the windmill did not have to stop in order to adjust the sails. [127]

Because the windmill needed to be manned for the milling

*Heron of Alexandria was a Mathematician, Physicist and Engineer who lived in the first century AD.

process there was no need for refining the control further. In the 19th century steam engines took over a lot of the work from the large windmills and the use of these windmills declined. The exploration of wind power changed however to a totally new direction.

In 1888 Charles F. Brush combined a post mill with a DC generator into what is known as the first large scale wind turbine producing electricity. The wind turbine had a rotor with a diameter of 17 meters and was able to produce 12 kW power. Despite its relative success of operation in 20 years it showed that low-speed, high solidity rotors were not ideal. In comparison a modern, fast-rotating wind turbine with the same rotor diameter would produce approximately 70-100 kW. [37, 41]

The Smith Putnam wind turbine. [91]

The Danish physicist and meteorologist Poul la Cour developed in 1891 a fast rotating wind turbine with primitive airfoil shapes. One of his students, Johannes Juul, built in the early 1950's the 200 kW Gedser Wind Turbine which is seen as one of the major milestones in the history of wind turbines. In the period from 1935-70 there was in fact a lot of research going on in United States, Denmark, France, Germany and Great Britain and one of the major results were the Gedser Wind Turbine. The most incredible result was however the Smith Putnam wind turbine erected in Vermont, 1941. This milestone was a 1.25 MW wind turbine with a rotor diameter of 53 meters which was built by famous scientists and engineers like Theodore von Karman and Jacob Pieter den Hartog. Although the rotor only lasted for some hundred hours of intermittent operation over several years, it showed a potential for large wind turbines†. In general it can be said that the period from 1935-70 showed the theoretical potential in large-scale wind turbines even though the actual practical applicable wind turbine was not created. [37, 41, 26]

The next major step happened because of the major oil crisis in 1973. This crisis started a rapid developing market (especially in California) and the rapid growth caused a lot of poor quality wind turbines in the first generation. These wind turbines resulted in a poor image of the whole wind energy business and the market decreased extensively from the late 1980s. Instead the markets in Europe started growing. Especially in Germany the market grew from the early 1990s and Denmark and Spain followed the growth subsequently.

A modern wind turbine.

This growing market first in California and later in Europe yielded larger and larger wind turbines ranging from a capacity of around 20-60 kW machines in the early 1980s to commercial prototypes of 5 MW wind turbines in 2004. Besides this the global installed capacity has increased with an annual growth of approximately 30% from 1992 to 2003. [26]

Along with the large growth in both wind turbine size and market size the power generation costs have reduced by approx-

† The blades were made of steel and broke off near the hub, apparently because of metal fatigue.

imately 80% over the last 20 years[26]. Today by combining the individual wind turbines into wind farms, wind power production facilities are today comparable in size to conventional facilities and also the costs are close to being competitive to conventional energy production.

1.2. Control of wind turbines

The classical windmills were operated by a miller whose family would live in bottom floors of the windmill. If the mean wind speed should change, the miller could then adapt the area of the sails so that a reasonable torque was applied to the grinding stone. For the wind turbines producing electricity it would be very costly to have an operator situated at each wind turbine and further the control task has become much harder because of the need to dampen structural and electrical oscillations. Instead the wind turbine will automatically adjust the input torque and determine when to connect to and disconnect from the grid.

The main issue when designing wind turbines is to trade-off the annual power production with the lifetime and cost of the machine. The power that is captured from the wind field can be described by a nonlinear function

$$P = \frac{1}{2} \cdot \rho \cdot A_r \cdot v^3 \cdot c_P \tag{1.1}$$

where ρ is the air density, A_r is the rotor swept area, v is the effective wind speed, and c_P is the aerodynamic efficiency of the rotor. If we assume that the maximum efficiency of the rotor is independent on the effective wind speed, the captured power can grow with the cube of the wind speed. Power electronics in the multi-megawatt scale are very expensive and an upper limit for the power production is defined to make a trade-off between the annual power production and the cost of the wind turbine. This upper limit is denoted the rating of the wind turbine and takes into account that the lower and mid wind speeds are more likely than the high wind speeds. The principle is illustrated in Figure 1.1 in which Betz limit[‡] is used as the aerodynamic efficiency, c_P, for a wind turbine with a diameter of 90 meters.

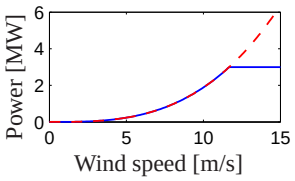


Figure 1.1: Illustration of the available (dashed red line) and captured power (solid blue line) from the wind.

From a power production point of view this means that a wind turbine has two operational modes. In partial load the target is to capture as much kinetic energy from the

[‡]The Betz limit of $\frac{16}{27}$ is the maximum possible efficiency of a horizontal axis wind turbine. Determined by the German physicist Albert Betz in 1919.

wind as possible and in full load the target is to keep the power production as close as possible to the rating.

The first range of commercial horizontal axis wind turbines were denoted passive stall wind turbines. These are typically operating with a squirrel cage induction generator which means that the generator speed can vary only very little from the grid frequency. Further the blades are mounted without any way of actively changing their orientation which means that there is no active way to control the energy into the wind turbine. This means that the only way to ensure that the power production will be below the solid line in Figure 1.1 is to design the airfoils in a proper manner.

To perform such an airfoil design a crucial observation is that the angle-of-attack changes with wind speed because the wind speed and direction seen from the blade segment depends on both the wind speed into the rotor swept area, and the angular speed of the rotor. Then the design of the airfoil can be made so that the blades enters stall for large wind speeds thereby limiting the captured energy below the threshold for the given wind speed.

This is the traditional design of wind turbines where the focus has been on the structural and aerodynamic design and not on active control of them. The main advantage of this concept is that it has a low complexity in terms of the number of considered components, and there is no risk in controller hardware/software breaking down. As a downside it is very difficult to get the power curve close to optimum over the whole region of operational wind speeds (typically from 4 m/s to 25 m/s). Besides this there is no way actively to control the loads introduced to the wind turbine.

To increase the competitiveness with conventional power sources active control was introduced to a new generation of wind turbines denoted Active-Stall wind turbines[§]. The extension from passive stall wind turbines to active stall wind turbines is essentially the introduction of pitch[¶] actuators which means that it is now possible to actively change the angle of attack of the blades.

With this introduction of active control of the wind turbines the power curve was improved significantly for full load operation. In partial load operation the improvement was however not as large because the target in this region is to maximise power production, i.e. not to limit it. The aerodynamic efficiency can be characterised by a concave function of pitch angle and tip speed ratio[‖]. This means that in order to maximise energy production it is necessary to control not only the pitch angle but also the rotor speed.

In the latest generation of wind turbines power electronics have been included with which it is possible to vary the generator speed and therefore also the rotor speed. Then with active control of not only the blade orientation but also the tip speed ratio the power production in partial load operation can be increased.

In full load operation the power production must be limited from above as mentioned previously in the section. This is done by pitching the blades away from the value yielding maximum efficiency. When pitching in the negative direction this will result in an increased angle of attack eventually leading to a separation in the flow, denoted stall. Oppositely when pitching to positive degrees the angle attack is decreased and the flow remains laminar but with a decreased efficiency. This strategy is denoted pitching

[§] In other literature the Active Stall concept might be called Assisted Stall or CombiStall

[¶] Rotating the blades around their longitude axis is denoted pitching.

[‖] The tip speed ratio is defined as the ratio between the translational speed of the blade tip and the effective wind speed.

to feather and the two concepts are illustrated in Figure 1.2 from which it can further be observed that the efficiency is a concave function having a single operating point with maximum efficiency.

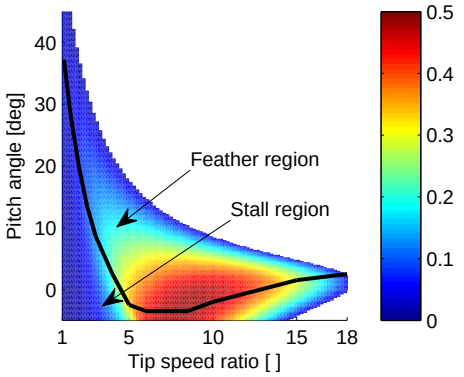


Figure 1.2: Illustration of aerodynamic efficiency. It is shown how the efficiency is a concave function of pitch angle and tip speed ratio. Further it is indicated where the wind turbine will operate in respectively stall and feather.

It has been decided in full load to pitch the blades into feather instead of into stall, which has the main advantage that the acoustic noise and fatigue loads in the blades are reduced. Further the non-separated flow is much better understood when compared with separated flow caused by stall. The main disadvantage is that it requires a larger pitch system because the necessary distance the pitch actuator needs to travel is larger when pitching to feather as opposed to pitching to stall. As an example the operating area for active stall control can be pitch angles of approximately 0° to -5° while for pitch control the operating area is around 0° to 30°.

The main trend for commercial wind turbines is today to use this latest generation of wind turbines, and especially passive stall wind turbines are not used much anymore. In the remainder of this thesis we will focus on this generation of wind turbines and we will denote pitching to feather as pitching if it is not stated otherwise in the specific context.

When designing wind turbines the power output is not the only design parameter. The main design parameter is the outcome per kWh averaged over the lifetime of the wind turbine. This means that figures such as the cost of maintenance, production and development are included. The controller design must therefore take both power production and loads on individual components into account. Besides this the design must be reusable in next generation of wind turbines in order to keep the development costs low. There are also environmental requirements such as acoustic noise emission and power quality that need to be taken into account.

1.2.1. State of the art for control of wind turbines

The control of wind turbines has achieved an increased attention in the past few decades and as a result a number of survey papers have been conducted on the topic [71, 72, 25, 9, 19, 11].

A classical structure for the control of wind turbines is illustrated in Figure 1.3. The main loop in this structure is the "power and speed controller" which has the main purpose to track a specified power and generator speed reference from the "reference generation". Typically also a steady state optimal pitch reference is supplied to avoid stall operation.

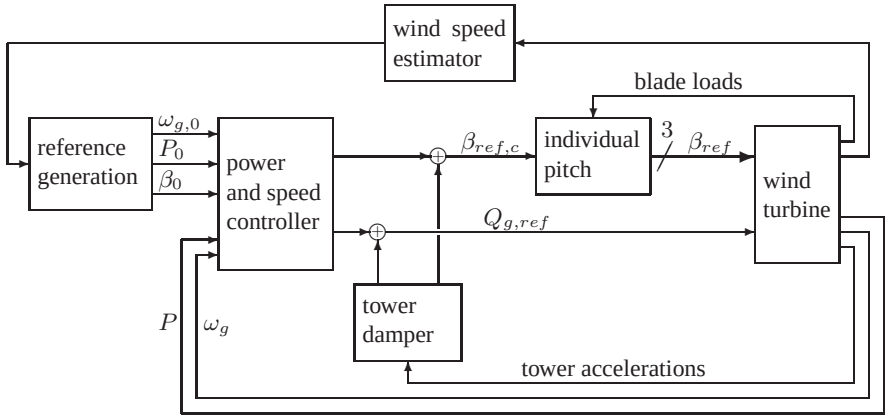


Figure 1.3: Typical structure for a control system for wind turbines. The abbreviations are interpreted the following way: ω_g is generator speed, P is active power, β is pitch angle, Q_g is generator reaction torque. A subindex, 0, means a steady state optimal value, ref means a dynamic reference, and $\beta_{ref,c}$ means a collective pitch reference.

The classical approach to the design of the "power and speed controller" is to design one controller for partial load and another controller for full load operation. In partial load operation the controller is typically a PI controller to track the generator speed reference, $\omega_{g,0}$ with the generator torque, $Q_{g,ref}$ as control signal – the pitch angle, $\beta_{ref,c}$ is kept constant at the optimal value, β_0. In full load operation another PI controller is used to track the generator speed reference with pitch reference as control signal – the generator torque is in full load kept at the nominal torque, $P_0/\omega_{g,0}$. A systematic approach for tuning the PI components can be found in [126, 51, 101] and a discussion of the setup is presented in [124].

For larger wind turbines it does not suffice to track the specified generator speed and power. Modern wind turbines are lightly damped structures which will start to oscillate if no active control is performed to overcome this. One of the main issues is that the tower will start oscillating fore-aft and sideways. However without much performance loss (in terms of power quality and pitch movement) this oscillation can be alleviated by a feedback term on the tower acceleration in each direction [74, 42]. A similar approach is usually applied to limit structural oscillations in the drive-train. In this case the generator speed is fed back to the generator reaction torque with a band pass filter on the drive train eigen-frequency[36] – this feedback is not illustrated in Figure 1.3.

Recently a lot of attention has been addressed at individual pitch control [117, 122, 132, 18, 65, 20, 48]. The purpose here is mainly to take into account that the wind field is unevenly distributed over the rotor swept area, e.g. due to wind shear, yaw error or wake from other wind turbines. The principle is then usually to superpose the collective pitch reference, $\beta_{ref,c}$, from the "power and speed controller" by a cyclic function in

order to remove oscillations induced by the rotation of the rotor.

For the "reference generation" the three reference variables, $\omega_{g,0}$, P_0, and β_0 are determined by solving a set of static equations. These equations determine the trajectory of nominal operating conditions which must be determined a-priori to the dynamic controller design. For the implementation of such a scheme for calculating the references it is necessary to have a variable available from which each operating condition can uniquely be determined. The effective wind speed is a natural choice because with this it requires only one variable to determine each operating condition for both partial load and full load. The downside is that this variable is not online measurable with adequate precision and it must therefore be estimated. This is usually done by solving the dynamic equations for the drive train to determine the aerodynamic torque which is then used to calculate the effective wind speed by an inversion of the aerodynamic model [81, 106, 38].

With an estimate of the effective wind speed available an intuitive extension of the control scheme is to include a feed-forward term from this estimate to the two control signals of the "power and speed controller" [85, 126, 62, 123]. This way we get a quicker response to variations in the large signal content of the effective wind speed – the estimators currently available are not fast and accurate enough for reacting to the small signal content.

The observant reader would have noticed that the controller structure presented in Figure 1.3 can be very challenging from a design point of view because it requires controllers that operate in parallel. In the figure the "power and speed controller" is illustrated to operate in parallel with the "tower damper", but the same will be the case for feed forward terms, drive train damper, etc. One way to solve this is to decouple the different loops and an example of this is presented in [70] where the speed controller is decoupled from the drive train damper. Another way is by the use of multi-variable controllers which can handle several inputs and outputs.

For these multi-variable controllers the starting point is a linear and time invariant (LTI) system which means that an LTI controller is determined for a model obtained by linearisation at a given operating condition. A direct approach for the controller design of LTI systems is by using pole placement algorithms as illustrated in [131] for the case of disturbance accommodating control of wind turbines.

It can be difficult to decide where to place the closed loop poles and for doing this we typically use optimisation algorithms. For stochastic systems a usual approach is to minimise the root-mean-square (RMS) value of a specified output given that the the input is unit intensity white noise. We denote such a problem formulation $\mathcal{H}_2$ control (or LQG control for a particular structure) and there are several applications of this method for the control of wind turbines [99, 87, 50, 118, 84, 30, 49, 90]

As an alternative approach we can consider the energy gain through a specified performance channel also denoted as $\mathcal{H}_\infty$ control. This methodology has been proven particular useful for guaranteeing closed loop performance in the case of models with associated uncertain elements. This method has been applied in several variations [29, 14, 16, 15, 61, 102, 13, 103].

It is clear that since both the $\mathcal{H}_2$ control and the $\mathcal{H}_\infty$ control are LTI design methods we need a way to apply it to the whole range of operating conditions and not only a single operating condition. One way of doing this is by using predictive control in which the optimisation problem is solved online. This essentially means that we can update the

linearised wind turbine model with one for the current operating condition as presented in [88, 33, 28, 105, 53, 21].

This approach of using predictive controllers has, however, the main drawback that it requires solving the optimisation problem online – which can be a very heavy task for industrial micro controllers. An alternative is the use of adaptive control in which specific model parameters are estimated and subsequently used to update the controller parameters. This approach has been used not only for the adaptation to changes along the nominal trajectory of operating conditions [115, 138, 23] but also for adaptation to varying aerodynamic coefficients in order to maximise energy capture in partial load operation [63, 59, 58, 80].

A similar but more straight forward approach is the use of gain-scheduling. Instead of determining the controller variables online from the estimated parameters, the controller is specified offline as a function of the operating condition. Then it only remains to choose the controller online as a function of the current operating condition. Classically the gain scheduling is obtained by varying only a few gains in the controller as in [73, 64, 66, 67, 125, 120]. Alternatively the gain scheduling can be obtained by connecting LTI controllers from for example $\mathcal{H}_2$ or $\mathcal{H}_\infty$ design. The interconnection can be done in many ways with examples like switching, interpolation of the state space matrices, and interpolation of the output of parallel controllers as presented in [64, 57, 32].

All the methods presented above to extend the LTI design methods to nonlinear approaches have a significant drawback. They do not take into account that the operating condition can change in time. In fact they assume that variations in operating conditions happen so slowly that they have no effect on the wind turbine dynamics. It is not clear what effect this assumption has on the operation of wind turbines. In Figure 1.4 a measurement of the wind speed is presented. This measurement is performed at Høvsøre**, Denmark, by a meteorology mast at an altitude of 80 meters above ground and it can be observed that most of the time the large signal variations are slow. On the other hand we observe occasional rapid variations in mean wind speed similar to the one around 210-220 seconds in the plot. As a result of this observation we find it important to understand what impact fast variations in operating conditions have on the closed loop performance.

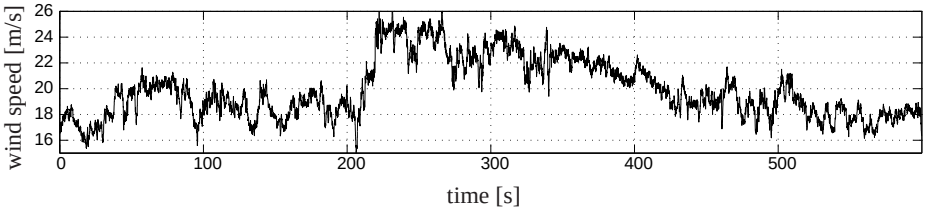


Figure 1.4: Measurement of point wind speeds. Note that the fast fluctuations are due to local phenomena which will be filtered out by the spatial averaging of the rotor.

A systematic method to take into account that operating point can change in time is a method denoted linear parameter varying (LPV) control. In this methodology the nonlinearity in the model is characterised by a set of parameters updating the state space

** Several wind turbine manufacturers use the Høvsøre site for prototype tests.

formulation. With this model characterisation a controller that depends on these parameters can be formulated by solving a convex optimisation problem involving a number of linear matrix inequalities (LMIs). Compared with the above described methods this approach is new but it has already achieved a significant attention for the control of wind turbines [12, 92, 76, 78, 75].

1.3. Objectives

The main focus of this project is to investigate a new method for the nominal[††] control of wind turbines with which it is possible to increase the life time availability and reduce the time to market for new controllers.

In Section 1.3.1 we elaborate on these main objectives and present the objectives of the project from a commercial point of view. Then in Section 1.3.2 we discuss the method that will be applied to meet the commercial objectives. The shortcomings in the available methods will be illustrated followed by a presentation of the scientific objectives required in order to fulfil the commercial objectives with the chosen methodology.

1.3.1. Commercial objectives

The main objective when designing controllers for the nominal operation of wind turbines is to minimise the induced fatigue loads while maximising the annual energy production. Further it is very important that critical variables are kept within their limits to avoid stopping the wind turbine in standard weather conditions, e.g. because of generator over-speed or overheating of electrical components. Based on these broad objectives a number of specific objectives are presented in the following with which the cost-effectiveness of wind turbines can be improved.

Reduction of loads with power curve kept constant

In order to make wind energy more competitive with conventional energy sources a key issue is to make them more cost effective. From a control point of view an important issue to do this is to reduce the fatigue loads – increase the life time of critical components – while keeping or even increasing the annual power production.

One controller for the entire operating region

Because of different requirements for the operation in partial load and full load, two different controllers are traditionally developed for these operational modes. The transition between these two operational modes is often done by simple switching which can introduce structural loads because of a rapid change in the behaviour of the controller. Further there is an increased risk for over-speeds in the region close to the switching because of the different requirements for tracking performance in the two regions.

To reduce the risk of over-speeds and remove the loads introduced by the switching we aim at a method with which the switching can be avoided. This means it must

[††] With "nominal" we mean that we deal only with nominal operation, i.e. no faults or failures are taken into account.

be possible with the method to design one controller for the entire range of operating conditions.

Robustness towards under-modelling

The physical components for construction of the wind turbine are manufactured by a number of sub-suppliers – typically with more than one supplier for each component. This together with variations in the manufacturing process leads to variations in the components and in the end this means that there are discrepancies between models used for the design and the dynamics of the physical wind turbine. Further we typically apply model simplifications in order to get a model order of realistic size. In the end this means that the model is associated with an uncertainty which it should be possible to take into account.

Simplified tuning process

Today the controller design is usually done by manually updating the variables of PID components for the speed controller, tower damping, drive train damping, etc. The update is then followed by an analysis of the controller performance (typically through simulation studies) and the process is iterated until a satisfactory level of performance is obtained.

This approach for the design of dynamic controllers becomes very complicated for large systems with conflicting requirements, e.g. the trade-off between tower movement and tracking of generator speed reference. The objective is then in broad terms that the method must provide tuning handles that relate to the physical requirements.

1.3.2. Scientific objectives

The design of LTI controllers is well-understood for many different applications including wind turbines which means that many tools are available for the design of LTI controllers. Unfortunately the wind turbine is a nonlinear system for which it is not possible to get satisfactory performance with LTI controllers when considered larger regions of operating conditions. As presented in Section 1.2.1 an intuitive extension is the use of gain-scheduling where we design LTI controllers for different operating conditions and afterwards interconnect them in a clever way. Further with the LPV methodology we get a systematic method for designing gain-scheduled controllers where it is possible take into account that the operating condition varies in time. This essentially means that the method provides controllers that are similar to the LTI controllers at each operating condition and that the performance level is guaranteed for a specified rate of variation of the scheduling parameter. Because of these strong advantages it has been decided to focus on the design of LPV controllers for wind turbines.

Investigation of numerical properties of LPV design method

The LPV methodology is still new (initially proposed in 1994 by prof. Andrew Packard). This means that there are a number of different algorithms available dependent on the type of parameter dependency (affine, polynomial, rational, etc.) and the underlying assumption of rate of variation on the scheduling variable.

It appears that the different algorithms have very different properties from a numerical point of view. For the application to the control of wind turbines it is very important to determine a method which provides good numerical stability in the algorithm in order to get a method with practical applicability.

Understand influence on rate-of-variation of scheduling parameter on obtainable performance level

The classical gain-scheduled control methods assume that the scheduling parameter can not change in time and some of the LPV methods assume that the scheduling parameter can change arbitrarily fast. With other LPV methods it is possible to specify an upper bound on the rate-of-variation, but these methods are more computationally demanding. Because of this it is important to get an understanding of the impact of the rate-of-variation on the obtainable performance in order to decide which approach is most relevant for the control of wind turbines.

Application of theory of linear parameter varying systems

The application of LPV methods for control of wind turbines is a relatively new field. A few investigations have been done for the control of wind turbines, however mostly with the most simplistic approach which is to assume affine parameter dependency. Such a parameter dependency is not deemed appropriate for the control of wind turbines when taking into account that the requirements for performance should also be scheduled. It is therefore necessary to investigate the applicability of the LPV methods for higher order parameter dependent wind turbine models.

Adaption of modern estimation theory to wind turbines

An intuitive choice for the scheduling function is the effective wind speed which is not online measurable with adequate precision. It is therefore required to establish a method by which the variable can be online estimated with adequate precision.

1.4. Thesis outline

The main contributions of the thesis are given in Chapter 3 which has three main focus areas: modelling of LPV systems, making LPV tools applicable to practical applications, and applying the LPV method to control of wind turbines. To be able to keep the focus on the contributions in Chapter 3 a general introduction to LPV systems and control is given in Chapter 2. Then conclusions and perspectives are given in Chapter 4.

The main purpose of Chapter 3 is to give an overview of the investigations performed throughout the study. For further details it is suggested to consult the papers and technical reports conducted throughout the study. These are provided at the end of the thesis and an overview is given below.

Paper A: Estimation of effective wind speed

The effective wind speed is often used as a variable in the controller design, e.g. for reference generation, feed forward, and as gain-scheduling variable. Measurement of

this variable with adequate precision is not possible and in this paper a new approach to the estimation of effective wind speed is provided together with a comparison with a classical result.

Paper B: Gain-scheduling LQ control of wind turbines

In this paper a classical approach to gain scheduling is investigated. A state estimator for the wind turbine model is designed together with a gain-scheduled LQ state-feedback controller. On the basis of simulation studies it is concluded that gain-scheduling is applicable to the control of wind turbines. This means that LPV control should be able to provide a good result on both local as well as global scale.

Paper C: LPV control of wind turbines for PLC and FLC

A usual problem for the implementation of wind turbine controllers is that there is a need for switching between a partial load controller (PLC) and a full load controller (FLC). In this paper a gain-scheduled approach is applied in the framework of linear parameter varying (LPV) systems. With this approach a smooth transition between partial load operation and full load operation can be observed, and simulations results show that fatigue load in structural components can be reduced when comparing with a classical control strategy.

Paper D: Rate bounded LPV control of a wind turbine in full load

In gain-scheduling control a critical variable is the gain-scheduling parameter. In classical gain-scheduling this parameter is assumed constant whereas some LPV methods assume that it can vary arbitrary fast. In this paper it is found that the obtainable performance level is indeed dependent on the assumptions on parameter rate of variation. A controller has been designed with rate bounded parameter variations to give local performance close to local LTI controllers. By the applied design method this performance level is then guaranteed globally.

Report E: Quasi-LPV control of wind turbines using LFTs

A direct way for performing LPV control of nonlinear systems is through a transformation from the nonlinear dynamic equations to a quasi-LPV formulation. This formulation takes the form of a LPV but with exact matching of the nonlinear dynamics through the parameter variation. This approach might at first appear very advantageous, but in this technical report it is pointed out that the designer must be very careful with the choice of representation in order to get satisfactory results.

Report F: LPV control of wind turbines using LFTs

The control of wind turbines involves not only nonlinear dynamic models but also performance requirements that vary with operating condition. This means that a weighted LPV system capturing the nonlinear dynamics and varying requirements cannot be described by affine parameter dependency. As an alternative approach this report investigates an LPV method for rational parameter dependency and it is concluded that this

method suffers from a number of numerical difficulties that must be handled in order to get a practical design method.

Report G: Manual for LPV design tool

The proposed design method in this thesis is based on gridding the parameter space. A tool has been developed to simplify the process of designing LPV controllers for systems with a single parameter dependency. This report provides a manual for the tool.

Chapter 2

Methods for the Control of Linear Parameter Varying Systems

Contents

This chapter is included to provide an overview of the available methods for designing LPV controllers and thereby give a starting point for the contributions of this thesis.

In Section 2.1 the main concept of LPV control is presented and an overview of the most common design algorithms is given. This general methods requires solving an optimisation problem with an infinite number of LMIs and in Section 2.2 it is presented how to reduce this problem to a finite dimensional optimisation problem.

2.1. Control of linear parameter varying systems

A classical approach to the control of nonlinear systems is to use gain scheduling in which LTI methods are used to design controllers at each operating conditions. The gain scheduled controller is then obtained from the designed LTI controllers by interpolation or switching and an overview of available approaches is given in [69, 104].

This approach of designing a number of LTI controllers to satisfy local requirements has the advantage that there are many design methods available which are well understood from both a theoretical point of view and a practical point of view. The main disadvantage of this approach is that the variations in operating condition is not taken into account, i.e. it is assumed that the operating condition changes so slowly that it has no impact on the closed loop performance as discussed in [114]. LPV control is a methodology that in many ways resemble the classical gain-scheduling, but there is a main difference in that the entire controller is designed in one shot and the methodology can take the rate of variation of the scheduling variables into account.

An LPV system is in [113] defined as a specific class of nonlinear systems that can be described as

$$\begin{bmatrix} \dot{\xi}(t) \\ z(t) \end{bmatrix} = \begin{bmatrix} \mathcal{A}[\delta(t)] & \mathcal{B}[\delta(t)] \\ \mathcal{C}[\delta(t)] & \mathcal{D}[\delta(t)] \end{bmatrix} \begin{bmatrix} \xi(t) \\ w(t) \end{bmatrix} \tag{2.1}$$

where ξ is the state vector, w and z are inputs and outputs, and δ is an exogenous scheduling parameter. This representation means that for each frozen parameter ($\dot{\delta}(t) = 0$) the system is LTI, and with a time-varying parameter the system dynamics will change depending on the parameter variations. In LPV analysis and control there are no a-priori assumptions about the trajectory of $\delta(t)$, however the possible parameter values and associated rate of variations must be contained in a specified set. An example of such a set of possible parameter values and rates of variation is presented in Figure 2.1 for a 2-D parameter dependency.

Remark 1. In this context it should be noted the parameter is assumed exogenous which means that it is sufficient only to include assumptions about a set possible parameter values and a set of parameter rates of variation. The method can also be applied for systems where the parameter is a function of state vector (denoted quasi-LPV systems). In this case the LPV analysis and design can be conservative because the number of possible parameter trajectories is limited in the quasi-LPV formulation when comparing with the LPV formulation. ◊

Within the last decade tools have been developed for analysing performance and designing controllers using two different methods for measuring performance. Mainly the focus has been on a generalisation of $\mathcal{H}_\infty$ control which is particularly useful for

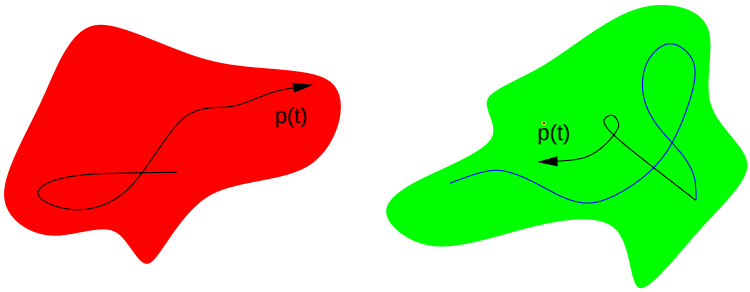


Figure 2.1: Example of illustration of a parameter range for a 2-D parameter dependency. The red figure shows the possible parameter values and the green figure shows the possible rates of variation. The two curves illustrate one possible parameter trajectory.

minimising closed loop oscillations [94, 10, 3, 4, 2, 110, 134, 140]. An LTI interpretation of this approach is that a frequency dependent upper bound is specified on the frequency response. The alternative method is a generalisation of $\mathcal{H}_2$ control for which an LTI interpretation is that it is measures the variance of the performance output given a Gaussian unity variance input [108, 31, 35, 137, 34].

For the control of wind turbines each of the two approaches are advantageous depending on the performance criterion considered. The main disturbance, the wind speed, is most accurately described by a stochastic process which indicates that the generalised $\mathcal{H}_2$ methodology is best suited for the tracking problem of generator speed and power references. On the other hand the structural oscillations are highly related to the frequency response of the closed loop system which indicates that the generalised $\mathcal{H}_\infty$ approach is best suited for minimising structural oscillations. Further the $\mathcal{H}_\infty$ approach is well-suited for handling model uncertainty and is the most usual approach for designing robust controllers.

It is possible to combine the two methods in a multi-channel approach as presented in [107, 111, 5], however for the controller design this approach is potentially conservative and significantly more demanding from a numerical point of view. It has has therefore been decided to focus only on single channel controller design in the generalised $\mathcal{H}_\infty$ approach.

The considered performance specification originates from the performance measure of dissipative systems described in [129, 130]. This formulation considers a very general class of nonlinear systems which in this thesis will be referred to in a simplified form

$$\dot{x} = f(x, w) \tag{2.2}$$
$$z = g(x, w) \tag{2.3}$$

with x as the state vector, w and z are inputs and outputs, and f and g are deterministic functions. A system of the form (2.2) is then said to be dissipative with a supply rate $s(t) = s(w(t), z(t))$ if there is a nonnegative storage function $p(x(t))$ to satisfy

$$p(x(t)) + \int_t^{t+T} s(t)\, dt \geq p(x(t+T)) \quad , \quad \text{for all } T \geq 0$$

for every possible trajectory of the system. This performance specification is a generalisation of Lyapunov theory in the sense that if we let $s(t) = 0$ we get a characterisation

of stability of the system with $p(x(t))$ as the Lyapunov function. This means that with a negative supply rate, the system is assured stable and $p(t)$ can be used as a Lyapunov function. Because of this the storage function, $p(t)$, is often denoted the Lyapunov function.

For the control of LPV systems of the form (2.1) a quadratic supply rate

$$s(t) = -\begin{bmatrix} w(t) \\ z(t) \end{bmatrix} \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} w(t) \\ z(t) \end{bmatrix} \tag{2.4}$$

will be used and we search for a quadratic storage function, $p(t) = x(t)^T P(\delta(t))\, x(t)$. This storage function is differentiable which means that the performance measure simplifies to $\frac{d}{dt}p(x(t)) \leq s(t)$ which can be written explicitly as

$$\begin{bmatrix} x(t) \\ \dot{x}(t) \end{bmatrix} \begin{bmatrix} \dot{P}(\delta(t)) & P(\delta(t)) \\ P(\delta(t)) & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ \dot{x}(t) \end{bmatrix} + \begin{bmatrix} w(t) \\ z(t) \end{bmatrix} \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} w(t) \\ z(t) \end{bmatrix} \leq 0 \tag{2.5}$$

for all time instances along every possible trajectory of the system. Now recall that $p(t)$ must be non-negative which is equivalent to $P(\delta(t)) \succeq 0$ for all time instances along every trajectory of the scheduling parameter. In LPV control this constraint is typically tightened to $P(\delta(t)) \succ 0$ because this requires the search for exponentially stable solutions.

2.1.1. Analysis and synthesis with general parameter dependency

In this thesis the performance measure in focus is the $\mathcal{H}_\infty$ norm generalised to LPV systems. This performance measure is denoted the induced $\mathcal{L}_2/\mathcal{L}_2$ norm which essentially measures the energy gain from the input, w, to the output, z, for all possible trajectories of the system. The system in (2.1) has induced $\mathcal{L}_2/\mathcal{L}_2$ gain lower than γ if

$$\int_{t=0}^{\infty} z(t)^T z(t)\, dt < \gamma^2 \int_{t=0}^{\infty} w(t)^T w(t)\, dt$$

for all non-zero inputs w with finite energy (i.e. for all $w \in \mathcal{L}_2 \backslash \{0\}$) under the assumption that the system is initially at rest.

By applying Parseval's theorem the time domain specification can for LTI systems be translated into the frequency domain specification

$$\sup_{\omega} \bar{\sigma}(G(j\omega)) = ||G||_{\mathcal{H}_\infty} < \gamma \quad \Leftrightarrow \quad G(j\omega)^* G(j\omega) \prec \gamma^2 I \quad , \quad \forall\, \omega \in \mathbb{R} \cup \{\infty\}$$

where $G(s)$ represents the closed loop dynamics from w to z. A solution to the LTI design problem has been developed through a set of Riccati equations [40]. The analysis and synthesis formulation will here be based on an LMI formulation which is highly related but more general [54, 45, 141] and which has been possible due to the advances in convex optimisation algorithms [89, 47].

It can be shown that this frequency domain specification is equivalent to the quadratic performance specification in (2.5) with $Q = -\gamma^2 I$, $S = 0$, $R = I$, by applying the Kalman-Yakubovich-Popov (KYP) lemma [60, 139, 98, 100]. Then by inserting the

system equations into (2.5) an induced $\mathcal{L}_2/\mathcal{L}_2$ gain less than γ is guaranteed if there exist a matrix function $P(\delta(t))$ which is positive definite for all possible parameter values and

$$\begin{bmatrix}\star\\ \star\end{bmatrix}^T \underbrace{\left[\begin{array}{c|cc} \star & \dot{P}(\delta(t)) & P(\delta(t)) \\ \star & P(\delta(t)) & 0 \\ \hline \star & & -\gamma^2 I \quad 0 \\ \star & & 0 \quad I \end{array}\right]^T \left[\begin{array}{cc|cc} I & 0 & & \\ \mathcal{A}(\delta(t)) & \mathcal{B}(\delta(t)) & & \\ \hline 0 & I & & \\ \mathcal{C}(\delta(t)) & \mathcal{D}(\delta(t)) & & \end{array}\right]}_{M(\delta(t))} \begin{bmatrix} x(t) \\ w(t) \end{bmatrix} \prec 0 \tag{2.6}$$

for all possible trajectories of w, x, and δ. This is equivalent to requiring $\mathrm{M}(\delta(t)) \prec 0$ for all possible parameter trajectories. To make the performance analysis computationally tractable it is necessary to make (2.6) independent of specific trajectories. To do so we define a function

$$\partial P(p,q) = \sum_{i=0}^{m} q_i \frac{\partial}{\partial p_i} P(p)$$

for which it can be observed that $\dot{P}(\delta) = \partial P(\delta, \dot{\delta})$. This means that if we plug in $\partial P(\delta, \lambda)$ instead of $\dot{P}(\delta)$ and solve the inequality for all possible values of δ and λ in respectively the set of parameter values and the set of rates of variation it is ensured that also (2.6) is satisfied.

Then on an assumption that every possible combination of parameter value and rate of variation will be experienced by some trajectory, the performance specification is given in Theorem 2 through a set of LMIs that are independent of time.

Theorem 2. *Let an LPV system, Σ, be given by (2.1) with all possible parameter trajectories contained in Δ and all possible parameter rates of variation contained in Λ. Then Σ is exponentially stable and has an induced $\mathcal{L}_2/\mathcal{L}_2$ gain less than γ if there exist a symmetric matrix function, $P(\delta)$, for which*

$$P(\delta) \succ 0 \tag{2.7a}$$

$$\left[\begin{array}{cc|cc} I & 0 & & \\ \mathcal{A}(\delta) & \mathcal{B}(\delta) & & \\ \hline 0 & I & & \\ \mathcal{C}(\delta) & \mathcal{D}(\delta) & & \end{array}\right]^T \left[\begin{array}{c|cc} \partial P(\delta,\lambda) & P(\delta) & \\ P(\delta) & 0 & \\ \hline & & -\gamma^2 I \quad 0 \\ & & 0 \quad I \end{array}\right] \left[\begin{array}{cc|cc} I & 0 & & \\ \mathcal{A}(\delta) & \mathcal{B}(\delta) & & \\ \hline 0 & I & & \\ \mathcal{C}(\delta) & \mathcal{D}(\delta) & & \end{array}\right] \prec 0 \tag{2.7b}$$

for all $(\delta, \lambda) \in \Delta \times \Lambda$.

Remark 3. Note that the parameter value δ and rate of variation λ are included in the formulation as two independent variables. Doing this is based on the assumption that all combinations of parameter values and rates of variation are encountered along at least one trajectory for the system. ◊

From Theorem 2 it is now possible to measure the closed loop performance of an LPV system by solving a set of linear matrix inequalities (LMIs). Regarding the design of LPV dynamic output feedback controllers the synthesis is based on the analysis

problem in Theorem 2 with the interconnection of the open loop system with the controller inserted as the closed loop matrices. In this context an LPV open loop system is considered of the form

$$\begin{bmatrix} \dot{x}(t) \\ z(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} A(\delta(t)) & B_p(\delta(t)) & B(\delta(t)) \\ C_p(\delta(t)) & D(\delta(t)) & E(\delta(t)) \\ C(\delta(t)) & F(\delta(t)) & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ w(t) \\ u(t) \end{bmatrix} \tag{2.8}$$

with x as the state vector, w and z as the inputs and outputs of the performance channel, and u and y as the inputs and outputs of the control channel. The objective is then to find a controller of the form

$$\begin{bmatrix} \dot{x}_c(t) \\ u(t) \end{bmatrix} = \begin{bmatrix} A_c(\delta(t), \dot{\delta}(t)) & B_c(\delta(t), \dot{\delta}(t)) \\ C_c(\delta(t), \dot{\delta}(t)) & D_c(\delta(t), \dot{\delta}(t)) \end{bmatrix} \begin{bmatrix} x_c(t) \\ y(t) \end{bmatrix} \tag{2.9}$$

with x_c as the controller state so that the closed loop interconnection will satisfy a performance specification γ according to Theorem 2. Along standard lines for the interconnection of systems the closed loop system can be described by the parameter dependent system matrices in (2.10) from which it is observed that the closed loop system matrices are affine in the controller variables.

$$\begin{bmatrix} \mathcal{A}(\delta(t), \dot{\delta}(t)) & \mathcal{B}(\delta, \dot{\delta}(t)) \\ \mathcal{C}(\delta(t), \dot{\delta}(t)) & \mathcal{D}(\delta, \dot{\delta}(t)) \end{bmatrix} = \left[\begin{array}{cc|c} A(\delta(t)) & 0 & B_p(\delta(t)) \\ 0 & 0 & 0 \\ \hline C_p(\delta(t)) & 0 & D(\delta(t)) \end{array}\right] + \tag{2.10}$$

$$+ \left[\begin{array}{cc} 0 & B(\delta(t)) \\ I & 0 \\ \hline 0 & E(\delta(t)) \end{array}\right] \begin{bmatrix} A_c(\delta(t), \dot{\delta}(t)) & B_c(\delta(t), \dot{\delta}(t)) \\ C_c(\delta(t), \dot{\delta}(t)) & D_c(\delta(t), \dot{\delta}(t)) \end{bmatrix} \left[\begin{array}{cc|c} 0 & I & 0 \\ C(\delta(t)) & 0 & F(\delta(t)) \end{array}\right]$$

When inserting this into (2.7b) it can be observed that the matrix inequality is quadratic in the controller variables due to the term

$$\begin{bmatrix} \mathcal{C}(\delta, \lambda) & \mathcal{D}(\delta, \lambda) \end{bmatrix}^T \begin{bmatrix} \mathcal{C}(\delta, \lambda) & \mathcal{D}(\delta, \lambda) \end{bmatrix}$$

This is not critical and can be resolved by a Schur complement. What is critical on the other hand is that the matrix inequality is bilinear in $P(\delta)$ and the controller variables as a consequence of the term

$$P(\delta) \begin{bmatrix} \mathcal{A}(\delta, \lambda) & \mathcal{B}(\delta, \lambda) \end{bmatrix}$$

Essentially this bi-linearity means that determining a controller to satisfy the performance level is a non-convex optimisation problem in this set of variables. Fortunately the formulation can be transformed into another formulation which is convex in its variables. One method to convexify the design problem is to use a set of projections to eliminate the controller variables from the formulation as illustrated in [45, 52]. The resulting synthesis formulation is given in Theorem 4 from which it can be determined if a controller exists to satisfy the performance specification.

The LMI (2.11b) is obtained by a projection of (2.7b) onto a domain of what is not directly measurable and the LMI (2.11c) is obtained by a projection of the dual version of (2.7b) onto what is not directly affected by control. The two variables $X(\delta)$ and $Y(\delta)$ are related to $P(\delta)$ as

$$P(\delta) = \begin{bmatrix} X(\delta) & \bullet \\ \bullet & \bullet \end{bmatrix} \quad , \quad P(\delta)^{-1} = \begin{bmatrix} Y(\delta) & \bullet \\ \bullet & \bullet \end{bmatrix}$$

where $\bullet$ represents fields that will not be used in the derivation.

Theorem 4. *There exists a controller on the form (2.9) for an LPV open loop system described by (2.8) if there exist symmetric matrix functions $X(\delta)$ and $Y(\delta)$ for which the following set of LMIs are satisfied for all possible parameter values, δ, and rates of of variation, λ.*

$$\begin{bmatrix} Y(\delta) & I \\ I & X(\delta) \end{bmatrix} \succ 0 \tag{2.11a}$$

$$\Psi^T \left[\begin{array}{c} \star \\ \star \\ \hline \star \\ \star \end{array}\right]^T \left[\begin{array}{cc|cc} \partial X(\delta,\lambda) & X(\delta) & & \\ X(\delta) & 0 & & \\ \hline & & -\gamma^2 I & 0 \\ & & 0 & I \end{array}\right] \left[\begin{array}{cc} I & 0 \\ A(\delta) & B_p(\delta) \\ \hline 0 & I \\ C_p(\delta) & D(\delta) \end{array}\right] \Psi \prec 0 \tag{2.11b}$$

$$\Phi^T \left[\begin{array}{c} \star \\ \star \\ \hline \star \\ \star \end{array}\right]^T \left[\begin{array}{cc|cc} 0 & Y(\delta) & & \\ Y(\delta) & \partial Y(\delta,\lambda) & & \\ \hline & & -\frac{1}{\gamma}^2 I & 0 \\ & & 0 & I \end{array}\right] \left[\begin{array}{cc} -A(\delta)^T & -C_p(\delta)^T \\ I & 0 \\ \hline -B_p(\delta)^T & -D(\delta)^T \\ 0 & I \end{array}\right] \Phi \succ 0 \tag{2.11c}$$

where Ψ is a basis for the null space of $\begin{bmatrix} C(\delta) & F(\delta) \end{bmatrix}$ and Φ forms a basis for the null space of $\begin{bmatrix} B(\delta)^T & E(\delta)^T \end{bmatrix}$.

It should be noted that Theorem 4 only provides a direct method to determine if a controller exists or not. It is however a well-known result that $P(\delta)$ can be determined from $X(\delta)$ and $Y(\delta)$ [45, 4]. This means that $P(\delta)$ can be considered an available variable in (2.7b) and the matrix inequality is therefore now a quadratic matrix inequality in the controller variables. As mentioned above such a quadratic form can be transformed into an LMI by a Schur complement and the controller can then be reconstructed from solving this LMI.

An alternative approach to transform (2.7) for synthesis problems into a convex optimisation problem is to perform a congruence transformation followed by a change of variables [111]. Again the storage function $P(\delta)$ is partitioned as

$$P(\delta) = \begin{bmatrix} X(\delta) & U(\delta) \\ U(\delta)^T & \bullet \end{bmatrix} \quad , \quad P(\delta)^{-1} = \begin{bmatrix} Y(\delta) & V(\delta) \\ V(\delta)^T & \bullet \end{bmatrix} \tag{2.12}$$

and by performing a congruence transformation of (2.7b) with

$$\mathcal{Y}(\delta) = \begin{bmatrix} Y(\delta) & I \\ V(\delta)^T & 0 \end{bmatrix} \tag{2.13}$$

which results in

$$
\left[\begin{array}{c}\star\\ \star\end{array}\right]^T
\left[\begin{array}{c}\star\\ \star\\ \hline \star\\ \star\end{array}\right]^T
\left[\begin{array}{cc|cc} \partial P(\delta,\lambda) & P(\delta) & & \\ P(\delta) & 0 & & \\ \hline & & -\gamma^2 I & 0 \\ & & 0 & I \end{array}\right]
\left[\begin{array}{cc} I & 0 \\ \mathcal{A}(\delta) & \mathcal{B}(\delta) \\ \hline 0 & I \\ \mathcal{C}(\delta) & \mathcal{D}(\delta) \end{array}\right]
\left[\begin{array}{cc} \mathcal{Y}(\delta) & 0 \\ 0 & I \end{array}\right] \prec 0
$$

$$
\Updownarrow
$$

$$
\left[\begin{array}{c}\star\\ \star\\ \star\\ \hline \star\\ \star\end{array}\right]^T
\left[\begin{array}{ccc|cc} 0 & \frac{1}{2}I & 0 & & \\ \frac{1}{2}I & 0 & I & & \\ 0 & I & 0 & & \\ \hline & & & -\gamma^2 I & 0 \\ & & & 0 & I \end{array}\right]
\left[\begin{array}{cc} \mathcal{Y}(\delta)^T \partial P(\delta,\lambda)\mathcal{Y}(\delta) & 0 \\ I & 0 \\ \mathcal{Y}(\delta)^T P(\delta)\mathcal{A}(\delta)\mathcal{Y}(\delta) & \mathcal{Y}(\delta)^T P(\delta)\mathcal{B}(\delta) \\ \hline 0 & I \\ \mathcal{C}(\delta)\mathcal{Y}(\delta) & \mathcal{D}(\delta) \end{array}\right] \prec 0 \tag{2.14}
$$

It is at this point not clear how this transformation is going to help in getting a synthesis formulation in terms of LMIs. To clarify this it is first of all necessary to observe that

$$
\mathcal{Y}(\delta)^T \partial P(\delta,\lambda)\mathcal{Y}(\delta) = \begin{bmatrix} -\partial Y(\delta,\lambda) & -\partial Y(\delta,\lambda)\, X(\delta) - \partial V(\delta,\lambda)\, U(\delta)^T \\ \partial X(\delta,\lambda)\, Y(\delta) - \partial U(\delta,\lambda)\, V(\delta)^T & \partial X(\delta,\lambda) \end{bmatrix}
$$

and further that

$$
\mathcal{Y}(\delta)^T P(\delta) = \begin{bmatrix} Y(\delta) & V(\delta) \\ I & 0 \end{bmatrix} \begin{bmatrix} X(\delta) & U(\delta) \\ U(\delta)^T & \bullet \end{bmatrix} = \begin{bmatrix} I & 0 \\ X(\delta) & U(\delta) \end{bmatrix}
$$

and on the basis of these two observations a variable substitution can be performed with

$$
\begin{bmatrix} K(\delta,\lambda) & L(\delta,\lambda) \\ M(\delta,\lambda) & N(\delta,\lambda) \end{bmatrix} = \begin{bmatrix} U(\delta) & X(\delta)B(\delta) \\ 0 & I \end{bmatrix} \begin{bmatrix} A_c(\delta,\lambda) & B_c(\delta,\lambda) \\ C_c(\delta,\lambda) & D_c(\delta,\lambda) \end{bmatrix} \begin{bmatrix} V(\delta)^T & 0 \\ C(\delta)Y(\delta) & I \end{bmatrix} + \begin{bmatrix} X(\delta)A(\delta)Y(\delta) + \frac{1}{2}\partial X(\delta,\lambda)\, Y(\delta) + \frac{1}{2}\partial U(\delta,\lambda)\, V(\delta)^T & 0 \\ 0 & 0 \end{bmatrix} \tag{2.15}
$$

and with the abbreviations

$$
\begin{aligned}
\mathbf{Z}(\delta,\lambda) &= \begin{bmatrix} -\partial Y(\delta,\lambda) & 0 \\ 0 & \partial X(\delta,\lambda) \end{bmatrix} \\
\mathbf{A}(\delta,\lambda) &= \begin{bmatrix} A(\delta)M(\delta,\lambda) + B(\delta)M(\delta,\lambda) & A(\delta) + B(\delta)N(\delta,\lambda)C(\delta) \\ K(\delta,\lambda) & X(\delta)A(\delta) + L(\delta,\lambda)C(\delta) \end{bmatrix} \\
\mathbf{B}(\delta,\lambda) &= \begin{bmatrix} B_p(\delta) + B(\delta)N(\delta,\lambda)F(\delta) \\ X(\delta)B_p(\delta) + L(\delta,\lambda)F(\delta) \end{bmatrix} \\
\mathbf{C}(\delta,\lambda) &= \begin{bmatrix} C_p(\delta)Y(\delta) & C_p(\delta) + E(\delta)N(\delta,\lambda)C(\delta) \end{bmatrix} \\
\mathbf{D}(\delta,\lambda) &= D_p + E(\delta)N(\delta,\lambda)F(\delta)
\end{aligned}
$$

the matrix inequality (2.14) can be transformed into an LMI in the new set of variable which leads to the alternative synthesis specification in Theorem 5.

Theorem 5. *There exist a controller on the form (2.9) for an LPV system described by (2.8) to satisfy a performance level γ if there exist variables X, Y, K, L, M, and N so that (2.16) is satisfied for all δ and λ in the range of parameter values and range of parameter variation rates. Further the controller can be constructed by inversion of the variable substitution in (2.15).*

$$\begin{bmatrix} Y(\delta) & I \\ I & X(\delta) \end{bmatrix} \succ 0 \tag{2.16a}$$

$$\left[\begin{array}{cc} \mathbf{Z}(\delta,\lambda) & 0 \\ I & 0 \\ \hline \mathbf{A}(\delta,\lambda) & \mathbf{B}(\delta,\lambda) \\ \hline 0 & I \\ \mathbf{C}(\delta,\lambda) & \mathbf{D}(\delta,\lambda) \end{array}\right]^T \left[\begin{array}{ccc|cc} 0 & \frac{1}{2}I & 0 & & \\ \frac{1}{2}I & 0 & I & & \\ 0 & I & 0 & & \\ \hline & & & -\gamma^2 I & 0 \\ & & & 0 & I \end{array}\right] \left[\begin{array}{cc} \mathbf{Z}(\delta,\lambda) & 0 \\ I & 0 \\ \hline \mathbf{A}(\delta,\lambda) & \mathbf{B}(\delta,\lambda) \\ \hline 0 & I \\ \mathbf{C}(\delta,\lambda) & \mathbf{D}(\delta,\lambda) \end{array}\right] \prec 0 \tag{2.16b}$$

2.2. From infinite to finite dimensional analysis and synthesis formulations

The algorithms for assessing closed loop performance and designing controllers in Theorem 2, 4, and 5 requires the solution of a set of LMIs for all possible combinations of parameter values and parameter rates of variation. This means that it is required to solve an infinite number of LMIs for both the analysis and the synthesis problem. For practical implementation this is not possible and a remedy must be determined.

Concerning the parameter rate of variation it can be observed that λ enters affinely in the parameter dependent LMIs. This means that under the assumption that the region of parameter rates of variation is polytopic, it suffices to test the vertices of this polytope.

If no assumption is imposed on the structure of the parameter dependency there are two different approaches. The direct approach is to choose a basis function for $P(\delta)$ and grid the parameter range as suggested by [136] and with an illustrative example in [8]. By gridding the parameter space the set of LMIs are only tested for a selected number of operating conditions which means that no guarantee is given for the entire operating region – only the selected points. The assumption in this approach is that the inter-grid behaviour can be investigated by using different grid sizes and checking the convergence.

An alternative approach is to use probabilistic approaches to solve the LMIs as proposed by [119, 24, 93, 43]. Instead of solving the set of parameter dependent LMIs in one optimisation problem, a randomised iterative algorithm is used which in a finite number of steps will converge to a solution for all possible parameter values. This approach has the major advantage that the solution is guaranteed for the original problem whereas the grid-based method only provides a guarantee for a subset, i.e. the selected operating conditions. The disadvantage of the probabilistic algorithm is that it requires a very large number of iterations and even though each step is fast when comparing with the grid-based method the number of steps is very large in order to provide the guarantee.

If it is possible to impose a specific structure of the parameter dependency and the parameter range can be described by a polytope, the infinite number of LMIs can be

verified by testing only the vertices of this polytope. This approach is in many cases based on a few (possible conservative) restrictions on the LMI problem discussed in [6].

2.2.1. Affine parameter dependency

The most simple case of parameter dependency is where the parameter enters affinely in the system matrices, i.e.

$$\begin{bmatrix} \mathcal{A}(\delta) & \mathcal{B}(\delta) \\ \mathcal{C}(\delta) & \mathcal{D}(\delta) \end{bmatrix} = \begin{bmatrix} \mathcal{A}_0 & \mathcal{B}_0 \\ \mathcal{C}_0 & \mathcal{D}_0 \end{bmatrix} + \sum_{i=1}^{m} \delta_i \begin{bmatrix} \mathcal{A}_i & \mathcal{B}_i \\ \mathcal{C}_i & \mathcal{D}_i \end{bmatrix}$$

For the analysis and/or design it is decided to search for a storage function of similar form, i.e.

$$P(\delta) = P_0 + \sum_{i=1}^{m} \delta_i P_i$$

Then by inserting this special case into the analysis (2.7b) it can be observed that the LMI depends quadratically on the parameters. Under the assumption that the range of parameter values is polytopic it is required only to test the vertices if the following extra constraint is enforced. [46, 1, 116]

$$\begin{bmatrix} \mathcal{A}_i^T P_i + P_i \mathcal{A}_i & P_i \mathcal{B}_i \\ \mathcal{B}_i P_i & 0 \end{bmatrix} \geq 0 \quad , \quad \text{for} \quad i = 1, \ldots, m$$

This extra restriction can introduce conservatism and the gap should be investigated by an optimistic approach, e.g. gridding. Note that in the special case of arbitrary fast variations we have that $P_i = 0$ for $i \neq 0$ and the approach is in this special case not conservative. For the controller synthesis the approach is analogous and will therefore not be discussed here. Concerning the control of wind turbines there are several examples of the application of this approach [92, 12, 82, 76, 128, 78, 75].

The approach has further been extended to piecewise affine parameter dependency in [79] by imposing a similar restriction. An example of this approach applied to the control of wind turbines is recently presented in [77].

It should be noted that the choice of an affine parameter dependency in the storage function $P(\delta)$ can be restrictive and recently the results have been generalised to polynomial storage functions to provide less conservatism [27, 86].

2.2.2. Rational parameter dependency

Concerning a more general parameter dependency the most focus has been on rational parameter dependency because of its relation to linear fractional transformations (LFTs). It should on the other hand also be noted that results have been obtained for polynomial parameter dependency with either extensions of the affine parameter dependency approach or through the use of the sum-of-squares (SOS) approach [7, 55, 135].

A starting point for the LPV design with rational parameter dependency was to use an LFT of the LPV system, i.e. to separate the dynamics into an LTI component and a component only containing the scheduling parameters. An scheduling block identical

to that of the open loop system would then be used as a scheduling function of the controller. Then the LTI based robust control method with structured uncertainty would be used to design the LTI part of the controller [112, 95, 97, 94, 3].

This approach has the same disadvantage as the classical gain scheduling approach that it does not take the rate-of-variation into account. An alternative approach based on the S-procedure is given in [109, 110] for arbitrary fast parameter rates and in [133] for rate bounded parameter variations. The analysis and synthesis is essentially split into a set of LMIs for the parameter dependency which shall be solved at each vertex of a polytope covering the parameter range. The other part is a set of LMIs to be solved for the LTI part of the formulation as presented in Theorem 6 for the performance analysis.

Theorem 6. *Consider an LPV system described by (2.17) with the parameter block $\Delta(t)$ having trajectories in a bounded set $\mathbf{\Delta}$.*

$$\begin{bmatrix} \dot{\xi}(t) \\ z_u(t) \\ z_p(t) \end{bmatrix} = \begin{bmatrix} \mathcal{A} & \mathcal{B}_u & \mathcal{B}_p \\ \mathcal{C}_u & \mathcal{D}_{uu} & \mathcal{D}_{up} \\ \mathcal{C}_p & \mathcal{D}_{pu} & \mathcal{D}_{pp} \end{bmatrix} \begin{bmatrix} \xi(t) \\ w_u(t) \\ w_p(t) \end{bmatrix} \quad , \quad w_u(t) = \Delta(t)\, z_u(t) \tag{2.17}$$

This system is exponentially stable and has an induced $\mathcal{L}_2/\mathcal{L}_2$ gain less than gamma if there exists a positive definite storage function P, symmetric multipliers Q and R and a multiplier S for which

$$\begin{bmatrix} \Delta \\ I \end{bmatrix} \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} \Delta \\ I \end{bmatrix} \succ 0 \quad , \quad \textit{for all} \quad \Delta \in \mathbf{\Delta}$$

$$\begin{bmatrix} I & 0 & 0 \\ \mathcal{A} & \mathcal{B}_u & \mathcal{B}_p \\ 0 & I & 0 \\ \mathcal{C}_u & \mathcal{D}_{uu} & \mathcal{D}_{up} \\ 0 & 0 & I \\ \mathcal{C}_p & \mathcal{D}_{pu} & \mathcal{D}_{pp} \end{bmatrix}^T \begin{bmatrix} 0 & P & & & & \\ P & 0 & & & & \\ & & Q & S & & \\ & & S^T & R & & \\ & & & & -\gamma^2 & 0 \\ & & & & 0 & I \end{bmatrix} \begin{bmatrix} I & 0 & 0 \\ \mathcal{A} & \mathcal{B}_u & \mathcal{B}_p \\ 0 & I & 0 \\ \mathcal{C}_u & \mathcal{D}_{uu} & \mathcal{D}_{up} \\ 0 & 0 & I \\ \mathcal{C}_p & \mathcal{D}_{pu} & \mathcal{D}_{pp} \end{bmatrix} \prec 0$$

Further if the analysis is restricted by $Q \prec 0$ it suffices to test only the vertices of a polytope containing $\mathbf{\Delta}$.

The assumption of arbitrary fast parameter rates can be conservative and in [134] the method has recently been generalised to rate bounded parameter variations. An alternative approach through duality and conjugate storage functions has been proposed in [39].

Recently also an alternative approach has been proposed in which the rational parameter dependent system is transformed to a descriptor system with affine parameter dependency [22].

2.3. Summary

To conclude, an LMI formulation for the performance analysis of LPV systems has been described. The presented algorithm requires solving an LMI for each possible parameter value in a pre-specified region of parameter values.

Further two different approaches for controller design have been presented. The main difference between the two algorithms is that the method in Theorem 4 is an efficient algorithm to establish if a controller exists due to the low number of variables for the optimisation problem. The method in Theorem 5 has a significantly higher number of variables and is therefore more demanding from a computational point of view – as well as regarding numerical conditioning. On the other hand the construction of the controller in this method is relatively simple requiring only a few computations whereas the method in Theorem 4 requires the solution of an additional set of LMIs – which though can be solved analytically.

This solution of the analysis and design problem requires an optimisation problem constrained by infinitely many LMIs which is not practical. In the case of general parameter dependency the available methods are computationally expensive and the approximative gridding approach seems as the best choice. In the special cases of affine or rational parameter dependency there are available methods for transforming the analysis and design problems into finite dimensional optimisation problems. These methods can introduce conservatism which should be investigated, e.g. by calculating a lower bound on γ through gridding.

Chapter 3

Linear Parameter Varying Control of Wind Turbines

Contents

This chapter contains the main contributions of the thesis. The following sections will describe the main ideas and conclusions on a number of different subjects. This means that for the technical details the reader should consult the attached papers and technical reports.

In Section 3.1 a method will be presented for the modelling of LPV systems followed by an investigation of the applicability of LPV methods to quasi-LPV models in Section 3.2.

For large scale systems the numerics play an important role in controller design with LPV models and a number of numerical issues are discussed and resolved in Section 3.3. This discussion is followed by a presentation of a novel tool in Section 3.4 for automating the synthesis and construction process.

For the control of wind turbines it is in Section 3.5 presented how it is possible to avoid the need for switching between partial load controllers and full load controllers. It has been observed that the rate of variation should be bounded from above in order to minimise conservatism and a full load controller has been designed in Section 3.6 to conform with such parameter rate bounds.

3.1. Modelling of LPV systems

The modelling of dynamics systems in an LPV framework can be a challenge because the nonlinearities entering the dynamics are not necessarily available in an algebraic form, e.g. only available through look-up tables. Further, it might not necessarily be clear how to choose the transformation from nonlinear models to an LPV formulation or even how to choose the scheduling variables. In this section a modularised approach to the modelling of LPV systems is presented.

3.1.1. Proposed modelling approach

The main idea is based on an observation that many physical systems can be represented by the interconnection of static nonlinearities with dynamic LTI models. An example of such an interconnection of sub-models is presented in Figure 3.1 in which three static nonlinearities interact with four LTI components. Some of these components will typically be known a-priori through first principles whereas other components need to be identified. Then a modularised approach can provide a good overview of which signals can be considered available for system identification of a subsystem.

It is assumed that each component can be determined by first principles or through a sequential procedure identifying components where inputs and outputs are available, potentially through already known subsystems. When the static nonlinearities have been determined they must be rewritten in an LPV form to obtain an LPV representation of the plant.

There are two ways of transforming the static nonlinearities into an LPV form. An appealing way is to use the quasi-LPV approach which uses an equivalent transformation and the alternative way is to use Jacobian-based linearisation along a trajectory of operating conditions. Which method to use depends on the character of the nonlinearity and in Section 3.2 the two methods will be discussed in the case of LPV controller design. The modelling in this section is independent on which of the two methods are used and it will be assumed that an LPV form has been determined.

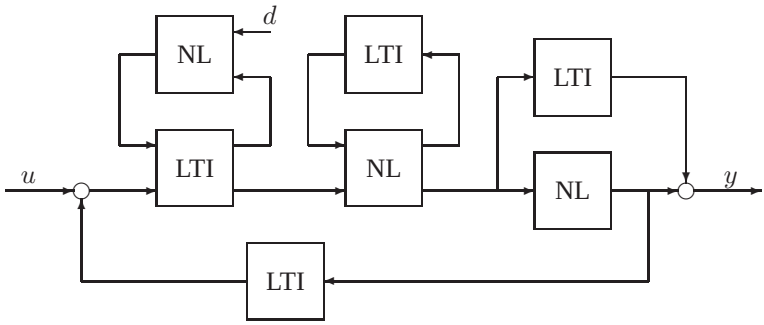


Figure 3.1: Block diagram of a nonlinear model presented as the interconnection of static nonlinearities with dynamic LTI models. All signals can be vector valued

It is assumed that the nonlinearity is determined as a rational function which means that the LPV model of it can be represented by the interconnection of a constant matrix with a matrix function linear in the scheduling variables. As an example consider the LPV model

$$y = \frac{\delta^2 + 2\delta + 3}{\delta + 4}\, u$$

This model is simply rewritten as

$$y = \frac{1}{4}\left(\delta^2 + 2\delta + 3\right) u - \frac{1}{4}\,\delta\, y$$

which can be described by interconnection of

$$\left[\begin{array}{c} z_1 \\ z_2 \\ \hline y \end{array}\right] = \left[\begin{array}{cc|c} -\frac{1}{4} & 1 & \frac{5}{4} \\ 0 & 0 & 1 \\ \hline \frac{1}{4} & 0 & \frac{3}{4} \end{array}\right] \left[\begin{array}{c} w_1 \\ w_2 \\ \hline u \end{array}\right] \quad \text{with} \quad \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = \underbrace{\begin{bmatrix} \delta & 0 \\ 0 & \delta \end{bmatrix}}_{\Delta(\delta)} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}$$

denoted a linear fractional transformation (LFT).

The advantage of the method is that it is usually simpler to model and verify smaller components than the whole plant. Then since the interconnection of several LFTs will result in a single LFT the method only requires very little work after the individual components have been identified. The proposed method has been applied to the modelling of wind turbines in Report E and F for respectively a quasi-LPV formulation and the linearisation based approach.

In Section 3.3.2 it will be found that the state space representation should be based on an experimental method. In this method the state variables are scaled so that all states have similar variance for the considered operating condition. This scaling is simple to do for each LTI component separately and it is therefore suggested to do this scaling before the components are interconnected.

To summarise, the following procedure has been proposed for the modelling of nonlinear systems in the LPV framework

1. Identify model structure (Separate static nonlinearities from LTI dynamics)

2. Choose scheduling variables

3. Determine range of scheduling variables (experimentally vs. desired behaviour)

4. Identify components
   LTI components: First principles vs. system identification
   Static nonlinearities: First principles vs. line fit

5. Scale states of LTI components by experiments. Scale system inputs/outputs similarly

6. Make rational approximation of each nonlinearity and describe them by LFTs.

7. Interconnect components by standard LFT operations

3.1.2. Determination of operating condition

To obtain a high numerical performance it is necessary that the rational form of the nonlinearity has as low order as possible and in particular that the block, $\Delta(\delta)$, has a minimal size. There is no general recipe for any of these two issues, but it is clear that the choice of operating range plays an important role. Further, the choice of operating range is in general very important for the obtainable performance level.

Two different methods will be proposed for determining the range of operating conditions depending on if the scheduling parameter is endogenous or exogenous. For many control applications there exist already controllers with a performance that will be comparable to the LPV controller to be designed – at least locally. In this case it is suggested to perform experimental/simulation studies similar to what is done in Report E in order to determine the correlation between scheduling variables. And ultimately such a study will result in for example a polytope containing the observed scheduling variables.

This approach is particularly interesting for quasi-LPV formulations where the scheduling variable depends on the dynamics of the system. In this case it can be difficult to determine the range by analytical means. However it should be noted that if the statistical properties of the input is known it might be possible to propagate these analytically, e.g. if the input is Gaussian the variance of the process can be determined.

In this context it is very important to stress that the above mentioned investigations are based on an expected behaviour of the closed loop system and it should always be investigated a-posteriori if the assumptions are met by the designed controller.

For “true” LPV systems where the scheduling variable is in fact exogenous the above approach is usually not interesting because the parameters in many cases will be uncorrelated. Alternatively the range of operating conditions can be described through desired behaviour. If for instance the scheduling variables are references and disturbances the operating range can be determined through the algorithm for reference generation. An example of this approach can be found in Report F where the nominal trajectory of operating conditions is determined.

To illustrate the two different methods let us consider describing the operating condition for wind speed and pitch angle. A simulation has been performed with a classical controller and the correlation is plotted in Figure 3.2 as the blue dots. This means that the blue points in the figure mark experienced operating conditions which should be

included in the design. A polytope with four vertices is in this example considered appropriate and the location of the vertices have been fitted to minimise the size of the polytope while having all experienced operating conditions inside.

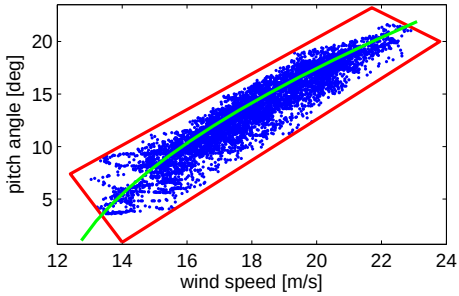


Figure 3.2: Illustration of determination of operating region. Blue dots show simulated data. Red polytope bounds simulated data. Green line shows nominal trajectory.

In the alternative approach we wish to describe the target trajectory on the basis of desired behaviour in contrast to observed behaviour. First observe that the described region is full load which means that the active power reference, P_{ref} and generator speed reference, $\omega_{g,ref}$, is constant. From the steady state point of view this means that the only variables in (1.1) on page 18 are wind speed and pitch reference. Then by using that we wish the wind turbine to operate in feather as opposed to stall, a unique mapping between pitch angle and wind speed can be obtained. This relation is shown as the green line in Figure 3.2 and the designer can then use this relation as an alternative representation for the operating region – the "width" of the curve can potentially be specified by the expected tracking performance.

3.1.3. Summary

A modularised method has been proposed for the modelling of LPV systems with rational parameter dependency. The main idea is to split the determination of the model into smaller components that are easier to identify. In particular it is suggested to separate nonlinear static components from dynamic components. The nonlinearities should be represented in a linear fractional way which makes the system interconnections very simple.

Further, two methods have been proposed for the determination of the operating region. For quasi-LPV systems with endogenous scheduling variables it is suggested to use an experimental method for the identification of the operating region. On the other hand for LPV systems with exogenous parameter dependency it is recommended to use an analytic method to determine for example a nominal trajectory of operating conditions. This can be done by analysing the reference generation.

3.2. Formulation for feasible control of quasi-LPV systems

The LPV design method is often applied to the design of controllers for quasi-LPV systems, and in this section the applicability of the LPV design framework is investigated for this special class of nonlinear systems. Traditionally, quasi-LPV systems are defined as LPV systems for which the scheduling parameter is not an exogenous function but instead a function of the state vector [96]. In this formulation a slightly more general formulation is investigated in which the scheduling parameter is dependent on not only the state vector but also the input vector. Further, in contrast to what is presented in [68] we consider quasi-LPV models that are obtained by a transformation of the nonlinear dynamic equations and not through linearisation.

3.2.1. Quasi-LPV model definition

We consider a nonlinear dynamic model of the form

$$\begin{aligned}\dot{x}(t) &= f[x(t), w(t), u(t)]\\ z(t) &= g[x(t), w(t), u(t)]\\ y(t) &= h[x(t), w(t), u(t)]\end{aligned}$$

where x is the state vector, w and z are inputs and outputs of a performance channel, and u and y are inputs and outputs of a control channel. The target is then to determine a transformation into a quasi-LPV model of the form

$$\begin{aligned}\dot{x}(t) &= A(\delta)\,x(t) + B_1(\delta)\,w(t) + B_2(\delta)\,u(t)\\ z(t) &= C_1(\delta)\,x(t) + D_{11}(\delta)\,w(t) + D_{12}(\delta)\,u(t)\\ y(t) &= C_2(\delta)\,x(t) + D_{21}(\delta)\,w(t) + D_{22}(\delta)\,u(t)\end{aligned}$$

where the scheduling parameter, $\delta[x(t), w(t), u(t)]$, is a function of both state vector and inputs.

3.2.2. Control problem for quasi-LPV systems

The considered control formulation is presented in Chapter 2 and the main idea is to find a matrix function $P(\delta)$ to satisfy the set of LMIs in Theorem 4 on page 35 or Theorem 5 on page 37 for each possible parameter value and rate of variation.

It is clear that if the LMIs must be satisfied for all possible parameter rates of variation it must also be satisfied for no parameter rate of variation. This means that the performance level of an LPV controller is bounded by the performance that can be obtained locally by LTI controllers.

For LPV systems with exogenous parameters this is intuitively clear and does not pose restrictions, however for the case of quasi-LPV systems this can be very restrictive. The reason for this conservatism is that quasi-LPV formulations are in general not unique and the different formulations will lead to frozen parameter dynamics that depend heavily on the choice of formulation.

In Report E a study has been performed for the LPV control of wind turbines through quasi-LPV models. This study concludes that it is crucial to choose the quasi-LPV representation to have two conditions satisfied:

- Good performance level for frozen parameters
- Embed as little dynamics as possible in scheduling variable

To illustrate this a simple example is given in the following.

3.2.3. Example

Consider a nonlinear system described by

$$\dot{x} = -x + w\,x + u\,x + u\,w$$
$$z = y = x$$

which can be described by the quasi-LPV formulation

$$\begin{bmatrix} \dot{x} \\ z \\ y \end{bmatrix} = \begin{bmatrix} -1 + k_1\,w + k_2\,u & (1-k_1)\,x + k_3 u & (1-k_2)\,x + (1-k_3)w \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ w \\ u \end{bmatrix}$$

where k_1, k_2 and k_3 are design variables which choose the specific quasi-LPV representation of the nonlinear model. The target is then to design an LPV controller, $u = K\,y$ to minimise the induced $\mathcal{L}_2/\mathcal{L}_2$ gain from w to z.

From the point of view of frozen parameter dynamics a few interesting observations can be made. If for instance we select $k_1 = k_2 = k_3 = 1$ the frozen parameter dynamics are not controllable and the frozen parameter closed loop dynamics will be unstable where $w \geq 1$ for any controller.

This is clearly conservative from a nonlinear control point of view since a controller gain

$$u = K\,x = \frac{-w}{x+w}\,x$$

will give a closed loop system, $\dot{x} = -x$, which is stable. Also the linearised dynamics

$$\dot{\tilde{x}} = (-1 + \bar{w})x + (\bar{x} + \bar{u})w + (\bar{x} + \bar{w})u$$

can be stabilised by state feedback for all operating conditions where $\bar{x} \neq -\bar{w}$ and $\bar{w} \geq 1$ which is a sufficiently larger region of stabilisable operating conditions when compared with the quasi-LPV formulation.

If we on the other hand choose $k_1 = k_2 = 0$ the open loop system is stable for all frozen parameters or k_2 and k_3 can be chosen so that the control will affect the plant for all frozen parameters. Another interesting observation is that if $k_1 = 1$ and $k_3 = 0$ the disturbance input will have no impact on the open loop dynamics from the frozen parameter point of view. This essentially means that the problem formulation would not make sense from a frozen parameter point view.

3.2.4. Summary

When using LPV methods for quasi-LPV models a key requirement is to choose the formulation to obtain good performance level with frozen parameters. Further, the representation should be chosen to embed a minimum amount of dynamics by the dynamics of the scheduling variable. Otherwise the performance might degrade when increasing tolerated parameter rate of variation.

Unfortunately the issue is very complicated and no general statement can be made towards which representation to choose. It is suggested to determine a general formulation that encaptures all possible quasi-LPV models. This is done by introducing a number of design variables with which the quasi-LPV representations can be chosen.

The control engineer shall then seek to understand open loop stability, controllability and observability and how the choice of representation affects these properties. Often all of these properties can vary with the choice of representation leading to very different synthesis results for the frozen parameters.

3.3. Numerical conditioning of LPV design algorithms

In Chapter 2 an overview was given for the analysis and synthesis of LPV controllers. In particular for the synthesis it was shown that for general parameter dependency there are two methods which are applicable. One has a low number of variables in the optimisation problem but the associated controller construction is quite complicated. The other method has a significantly higher number of design variables for the optimisation problem whereas the controller construction is much simpler. In the following, different algorithms for the controller construction will be investigated from a numerical point of view and it will be investigated how the state space realisation influences obtainable performance.

3.3.1. Conditioning of construction algorithm

For real-life applications with large state space models it is important to have a fast and reliable design algorithm, because the controller design is often a tuning process with several iterations to find for example the desired weighting functions. Because of this it has been decided to focus on the first presented method in which the controller variables are eliminated from the synthesis LMIs. The focus of this paper is then to determine a numerically reliable method for construction of the dynamic controller.

To understand the numerical difficulties of controller construction in the LPV control formulation, the construction algorithm is summarised in the following. For numerical simplicity the algorithms are first investigated for parameter independent storage functions X and Y. From the optimisation problem in Theorem 4 the variables X and Y have been determined to satisfy a performance level, γ. The target is then on the basis of these variables to determine a set of controller variables, $A_c(\delta)$, $B_c(\delta)$, $C_c(\delta)$, and $D_c(\delta)$ to satisfy the performance specification.

This is done by solving the set of LMIs in (3.1) for P, where the closed system matrices $\mathcal{A}(\delta)$, $\mathcal{B}(\delta)$, $\mathcal{C}(\delta)$, and $\mathcal{D}(\delta)$ are affine functions of the controller variables as expressed in (2.10) on page 34. The empty fields in (3.1) represent zero terms and is included to simplify notation.

$$\left[\begin{array}{cc} I & 0 \\ \mathcal{A}(\delta) & \mathcal{B}(\delta) \\ \hline 0 & I \\ \mathcal{C}(\delta) & \mathcal{D}(\delta) \end{array}\right]^T \left[\begin{array}{cc|cc} 0 & P & & \\ P & 0 & & \\ \hline & & -\gamma^2 I & 0 \\ & & 0 & I \end{array}\right] \left[\begin{array}{cc} I & 0 \\ \mathcal{A}(\delta) & \mathcal{B}(\delta) \\ \hline 0 & I \\ \mathcal{C}(\delta) & \mathcal{D}(\delta) \end{array}\right] \prec 0 \tag{3.1}$$

From the synthesis equations the variable P is not directly available, but the two variables X and Y are strongly related to it as expressed in (2.12) on page 35. This means that P can be calculated from solving

$$\begin{bmatrix} Y & V \\ I & 0 \end{bmatrix} P = \begin{bmatrix} I & 0 \\ X & U \end{bmatrix} \quad , \quad UV^T = I - XY \tag{3.2}$$

Then with P available the controller construction from (3.1) is a quadratic matrix inequality (QMI) which can be solved analytically in several ways [54, 45, 52]. From a computational point of view these methods are advantageous, and it has been the author's experience that the following approach for solving the QMI proposed by [121] is advantageous.

The matrix inequality (3.1) for closed loop analysis can be rewritten in a synthesis framework as

$$\left[\begin{array}{c} I \\ L(\delta)^T K(\delta) R(\delta) + M(\delta) \end{array}\right]^T \left[\begin{array}{cc|cc} 0 & 0 & P & 0 \\ 0 & -\gamma I & 0 & 0 \\ \hline P & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\gamma} \end{array}\right] \left[\begin{array}{c} I \\ L(\delta)^T K(\delta) R(\delta) + M(\delta) \end{array}\right] \prec 0 \tag{3.3}$$

with

$$L(\delta) = \begin{bmatrix} 0 & B(\delta) \\ I & 0 \\ 0 & E(\delta) \end{bmatrix} \quad , \quad R(\delta) = \begin{bmatrix} 0 & I & 0 \\ C(\delta) & 0 & F(\delta) \end{bmatrix}$$

$$K(\delta) = \begin{bmatrix} A_c(\delta) & B_c(\delta) \\ C_c(\delta) & D_c(\delta) \end{bmatrix} \quad , \quad M(\delta) = \begin{bmatrix} A(\delta) & 0 & B_p(\delta) \\ 0 & 0 & 0 \\ C_p(\delta) & 0 & D(\delta) \end{bmatrix}$$

and the idea is then to find a controller, $K(\delta)$, and a storage function P, to satisfy the performance level, γ.

If we define the following set of matrices

$$\Lambda(\delta) = \begin{bmatrix} \Lambda_{11}(\delta) & \Lambda_{12}(\delta) \\ \Lambda_{21}(\delta) & \Lambda_{22}(\delta) \end{bmatrix} = \Phi(\delta)^T M(\delta) \Psi(\delta)$$

$$\Upsilon(\delta) = \begin{bmatrix} I & 0 \\ 0 & 0 \\ 0 & I \\ \Lambda_{21}(\delta) & 0 \end{bmatrix}, \quad \Omega(\delta) = \begin{bmatrix} 0 \\ I \\ \Lambda_{12}(\delta) \\ \Lambda_{22}(\delta) \end{bmatrix}$$

$$\Pi(\delta) = \begin{bmatrix} \Psi(\delta) & 0 \\ 0 & \Phi(\delta)^{-T} \end{bmatrix}^T \left[\begin{array}{cc|cc} 0 & 0 & P & 0 \\ 0 & -\gamma I & 0 & 0 \\ \hline P & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\gamma} \end{array}\right] \begin{bmatrix} \Psi(\delta) & 0 \\ 0 & \Phi(\delta)^{-T} \end{bmatrix}$$

with $L(\delta)\,\Phi(\delta) = \begin{bmatrix} L_1(\delta) & 0 \end{bmatrix}$ and $R(\delta)\,\Psi(\delta) = \begin{bmatrix} R_1(\delta) & 0 \end{bmatrix}$ in which $L_1(\delta)$ and $R_1(\delta)$ are full rank for all parameter values. Then $K(\delta)$ can be chosen as

$$K(\delta) = L_1(\delta)^{-T} \left(\Theta_2(\delta)\Theta_1(\delta)^{-1} - \Lambda_{11}(\delta)\right) R_1(\delta)^{-1}$$

where $\Theta_1(\delta)$ and $\Theta_2(\delta)$ are chosen to satisfy the QMI

$$\begin{bmatrix} \Theta_1 \\ \Theta_2 \end{bmatrix}^T \left(\Upsilon^T \Pi \Upsilon - \Upsilon^T \Pi \Omega \left(\Omega^T \Pi \Omega\right)^{-1} \Omega^T \Pi \Upsilon\right) \begin{bmatrix} \Theta_1 \\ \Theta_2 \end{bmatrix} \prec 0$$

for all parameter values which in theory is always possible, e.g. from an eigenvalue decomposition. Note that the dependency on the scheduling parameter, δ, has been left out in the above inequality for notational simplicity. The proposal in [121] is then to choose $\Psi(\delta)$ in a specific way to make $\Omega(\delta)^T \Pi(\delta) \Omega(\delta)$ well-conditioned and thereby make the inversion possible from a numerical point of view.

Still the QMI can be ill-conditioned if for example there are large entries in the system matrices or P, or if γ is far from unity. It is expected that issues related to the entries in the system matrices and performance level can be resolved by a proper choice of state space realisation and performance specification. This means that the conditioning of the algorithm for controller construction is determined mainly from the conditioning of the extended storage function, P.

Unfortunately the constructed variable P is often very ill-conditioned, e.g. as a consequence of the full order controller being close to non-minimal. One option is to apply methods for reduced order design, but since this approach introduces a non-convex constraint in the optimisation problem it has not been considered in this thesis. It has therefore been chosen to focus on how to enhance numerical performance of the construction of full order controllers.

As a consequence the main cause of the ill-conditioning of the algorithm is that $\Pi(\delta)$ is ill-conditioned. From the investigations in Paper C it has then been experienced that conditioning of the construction algorithm can be improved by rearranging (3.3) so that the inner and outer terms have similar conditioning, i.e. improving conditioning of the inner term while worsening the conditioning of the outer terms. This means that the objective is to find a transformation of the QMI to minimise the maximum condition number of the inner and outer term in the QMI.

From (3.2) it is observed that P can be formulated as

$$P = {P_1}^{-1} P_2 = {P_2}^T {P_1}^{-T}$$

and the numerical performance can be enhanced by moving either ${P_1}^{-1}$ or P_2 from the inner term to the outer term. Alternatively a congruence transformation has been applied with

$$\begin{bmatrix} P_1 & 0 \\ 0 & I \end{bmatrix} \quad \text{or} \quad \begin{bmatrix} {P_2}^{-1} & 0 \\ 0 & I \end{bmatrix}$$

which will eliminate respectively P_1 or P_2 from the inner terms. The numerical study in Paper C indicates that the best numerical performance among the four presented approaches is obtained by moving P_1 or P_2 from the inner to the outer LMI and with

$$\begin{aligned} P_1 = \begin{bmatrix} N & T \\ M^{-1} & 0 \end{bmatrix} \quad , \quad P_2 = \begin{bmatrix} N^{-1} & 0 \\ M & S \end{bmatrix} \\ M^T M = X \quad , \quad N^T N = Y \\ S T^T = \left(N M^T \right)^{-1} - M N^T \end{aligned}$$

It appears that the approach with a congruence transformation of the QMI can make the QMI as a whole worse conditioned whereas the other method does not change the QMI. On the other hand some interesting observations can be made regarding the approach with a congruence transformation with P_1. When applying this congruence transformation to the QMI we end up with the formulation in (2.14) on page 36. Now with X, Y, U, and V available the variable substitution used for Theorem 5 is no longer necessary. A classical result in LTI $\mathcal{H}_\infty$ synthesis in [44] reveals that with a simple transformation of this QMI the three controller variables $B_c(\delta)$, $C_c(\delta)$, and $D_c(\delta)$ can be calculated from individual and smaller sized LMIs. Then with $A_c(\delta)$ chosen to zero out off-diagonal terms between the LMIs (which is always possible) the controller is constructed to satisfy the performance criterion.

A careful investigation reveals that the method is applicable to LPV systems, even with parameter dependent storage functions. This is discussed in [2] for a class of LPV systems and in Paper D the method is presented in more detail for a more general formulation of LPV system with an example regarding the control of wind turbines.

A comparison of the numerical performance in the methods applied in Paper C and Paper D indicates that the latter method is advantageous from a numerical point of view. In order to get good numerical performance it is still necessary to perform conditioning similar to what is described above for the alternative algorithm. In this case the choices for factorisation are a bit different with the basic ideas presented in [44].

The conclusion of this investigation is that a generalisation of a classical construction algorithm from $\mathcal{H}_\infty$ control provides superior numerical performance compared with the alternative algorithm. The downside of this method is that it must be solved for each possible parameter in the set. This essentially means that either the controller must be constructed online as a function of the current parameter value or alternatively the controller must be constructed offline for a finite number of parameter values. There are advantages and disadvantages of both approaches. By constructing the controller online it is in theory ensured that the controller is constructed correctly to meet the

performance specification, but if the numerics fail in this online computation it can have fatal consequences. With the offline computation of the controller it is known before implementation how the controller will behave, however the performance level is not guaranteed in between grid points.

This discussion of offline versus online computation of the controller is not as relevant for the first described construction algorithm because the ideas can be directly translated to the construction algorithm in [110]. With this algorithm it is possible to calculate a parameterised controller directly which means that the controller and scheduling function can be calculated offline to satisfy all parameter values in the set. It should be noted that the algorithm requires an assumption of arbitrary fast parameter variations which can be very restrictive.

It is therefore not possible to provide a recommendation towards one algorithm that is superior for any LPV design problem. The general recommendation is to use the LFT algorithm in [110] if two requirements are satisfied. First of all it must not be restrictive to assume that the scheduling variables can vary arbitrarily fast and furthermore the problem size must be small enough so that the construction algorithm is numerically stable. In the more general cases it is recommended to calculate the controller offline in a number of grid points similar to what is done in Paper D. Then the parameterised controller can in many applications be determined from investigations of frequency responses at each grid point.

3.3.2. Influence of state space realisation on numerical performance

Concerning the numerical issues related to the LPV methods it should be noted that the choice of state space realisation can have a great influence on the obtainable performance.

To get an indication of this numerical issue an investigation has been performed with a controller which is known to perform well for the particular problem. The considered control problem is the tracking of generator speed while minimising drive train oscillations and control activity in pitch actuator. The considered controller is a gain scheduled PI controller tracking generator speed by applying a pitch reference. For dampening drive train oscillations, an LTI band pass filter from generator speed to active power used. This controller structure has shown good performance in real-life applications both from a local point of view and for the whole full load operation.

Weighting functions have been designed to trade-off tracking of generator speed with power fluctuations, and damage in drive train and pitch system. Further, the weights have been scaled to give a performance level, $\gamma = 1$, over the whole operating region in consideration which is the interval 14-24 m/s. An LFT formulation of the gain scheduled controller is then connected to an LFT representation of a dynamic model of the wind turbine for use in LPV performance analysis.

The closed loop analysis is performed by solving the set of LMIs in Theorem 6 for three different realisations. The first choice is to use the model as is, i.e. no balancing. The second choice is to use a numerical algorithm to balance the control and observability Gramians (done with `balreal`), and the last method is to use a diagonal similarity transformation to make the system matrices have similar row and column norm (done with `ssbal`). Note that it is important when balancing LPV models to keep the global coordinate system which means that the realisation is only balanced exactly

for one operating condition. It is then expected that by applying the same coordinate transformation to the other operating conditions these models will be close to balanced. When performing the analysis it is expected that for a very small considered operating region the performance level should be close to the LTI norm, i.e. close to unity. Then for larger ranges of operating conditions the guaranteed performance level will degrade, i.e. the gamma value will be larger.

The analysis is therefore performed for various sizes of the operating condition to understand the degradation and the result is illustrated in Figure 3.3. The leftmost illustration shows the scaling of the operating condition, e.g. for a scaling of 0.5 the considered wind speed range is 16-21 m/s. The rightmost figures show the guaranteed performance as a function of scaling for the three different realisations at it can be observed that the three realisations result in significantly different performance levels. In particular observe that with no balancing the obtainable performance level is almost constant at $\gamma = 34$ which does not fit with the expectation of a performance close to unity for very small scalings. The rightmost plot with a diagonal similarity transformation shows a result in which the optimisation problem appears very difficult to solve – leading to large variations in guaranteed performance.

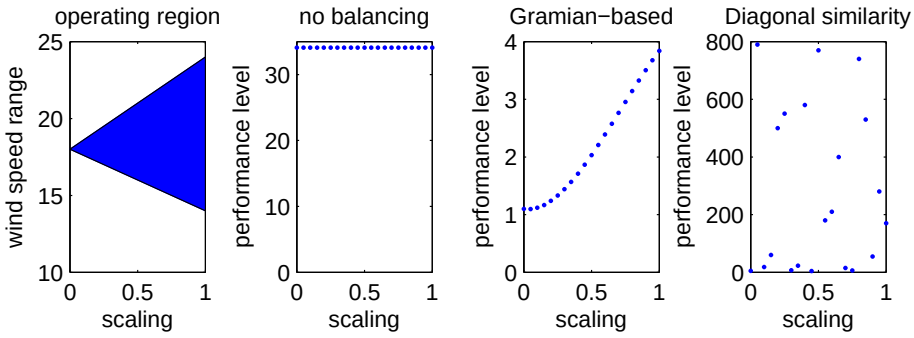


Figure 3.3: Guaranteed performance level for different realisations. Observe how the pattern changes significantly with choice of realisation.

On the basis of the study in Figure 3.3 it can be observed that it is indeed possible to analyse the performance bound of an LPV system using the method in Theorem 6, however the method seems very sensitive to the choice of realisation. The conclusion from this is that when using the method there must be a focus on the choice of realisation, however it can on the basis of this study alone not be concluded that the Gramian-based balancing is superior to the other two in general.

Regarding controller synthesis a difference in obtainable performance level has been observed depending on the choice of design method. Two different design methods have been applied to the same plant model with identical weighting functions. In Report F the LFT approach has been applied and in Paper D the grid based method has been applied. To understand the numerics, various state space realisations have been applied for both design techniques and it is observed that with the LFT method the performance level varies in the interval 33-38 and with the grid method with constant storage function the performance level varies between 1.9 and 850 depending on state space realisation.

Two main points can be concluded from this. First of all it can be noted that the performance level of the LFT approach does not vary much with the choice of state space realisation whereas the grid based method varies significantly. This was not expected a-priori because the optimisation problem in the LFT method has a significantly larger number of variables. On the basis of this it is expected that the performance level of the LFT method is limited by some other factor like non-minimality in the choice of scheduling function. It is difficult for many applications to investigate minimality for LPV systems because of the coupling between frozen parameter dynamics and the dynamics of the scheduling parameter. As a consequence of this it is important to investigate further the impact of non-minimality on the optimisation problem for controller synthesis.

Another observation is the large difference in the best obtained performance for the LFT method and the grid-based method. A reason for this can be that the grid-based method is too optimistic, however this is not expected to be the main cause for two reasons. First of all it can be observed that the performance level only decreases by 1% from a grid size of 4 points to 25 points and further by gridding in the LFT method there is only an insignificant difference in performance. Finally, simulation results with the constructed controller with the grid-based method indicate a performance level which is similar to what can be calculated with the grid-based method.

It is on the basis of the above observations concluded that the designer must be aware that different state space realisations can lead to very different results for both analysis and synthesis. Further, the obtainable performance level for controller synthesis can vary much between design methods which are identical in theory. This indicates that the numerics play a crucial role for the use of the LPV analysis and design methods for practical applications and further research is necessary in order to determine what drives the numerical performance and how to make the optimisation problem reliable from a numerical point of view.

3.3.3. Summary

Two methods have been proposed for the construction of LPV controllers. One is based on the construction of extended storage functions and solving a quadratic matrix inequality and the other is based on computing the controller matrices from different LMIs. The first method should be used for small problems where the numerics are not too difficult and where it is not restrictive to assume arbitrary fast parameter variations. The latter can be applied to more general problems, but in contrast to the first method it does not provide a parameterised controller which means that interpolation is required.

It has also been investigated how the choice of realisation affects the obtainable performance level. From this study it has been experienced that the choice of realisation can have a great influence on the convergence of the optimisation problem. This means that it can be very important to choose an appropriate realisation. Unfortunately no general statement can be made towards the choice of realisation, but it appears that Gramian-based balancing is good for closed loop analysis. For controller synthesis a good method seems to be balancing the amplitude of inputs/outputs through simulation studies.

3.4. Design tool for LPV control

The design formulation can appear very complicated for engineers that are not familiar with LMIs in control. When designing controllers for practical applications, many engineers do not want to spend time on fine tuning the algorithms to give a reliable computation of the controller. Instead the main focus is to update tuning handles related to the performance specification and let a design tool do the actual computation of the controller.

3.4.1. Presentation of design tool

A design tool has been developed* to reflect many of the needs for the design of LPV controllers from a practical point of view. The main idea in the tool is to hide as much as possible the technical details which has resulted in a graphical user interface as illustrated in Figure 3.4.

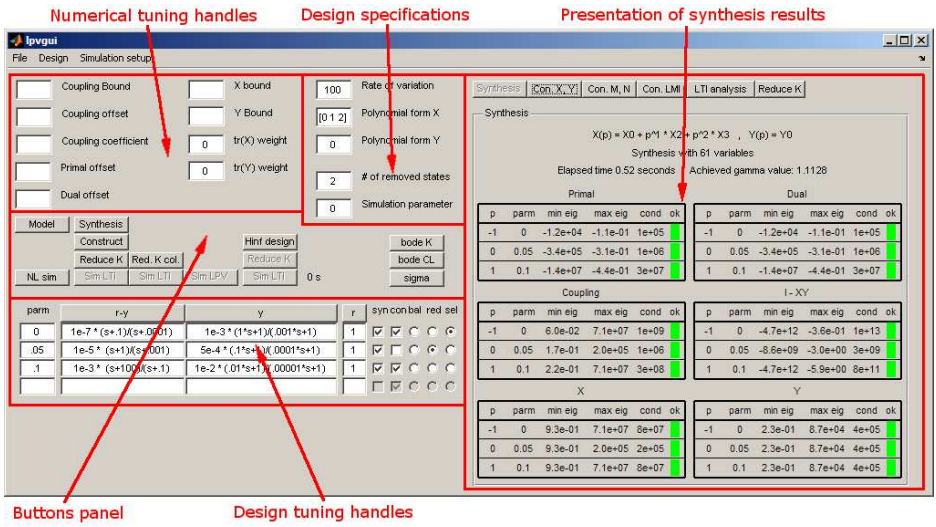


Figure 3.4: Illustration of main window of developed graphical used interface.

For many practical control applications it can be difficult to translate the physical performance specification into an induced $\mathcal{L}_2/\mathcal{L}_2$ gain through a performance channel. As a consequence the design weights are often used as tuning handles that are updated on the basis of simulation/experimental studies.

A large effort has been put into the design tool to reflect this issue. First of all a panel has been included in which it is simple to update the performance weight for each performance input/output – note that the performance weights are specified in symbolic notation which gives a nice overview. Further, check-boxes have been included to make it simple to test different combinations of design points.

*The design tool is not public domain.

Within each iteration of tuning the performance weights should be simple to analyse the resulting controller. As argued above this is often done through simulation studies and to make the controller design a unified process in one tool, a link to a set of Simulink models can be made and simulation studies can then be made directly from the tool by pressing a button. It is well-known that the $\mathcal{H}_\infty$ approach often leads to fast modes that are not implementable in practice, and to deal with this a method has been included to remove these fast modes.

From a technical point of view a bound on the rate of variation must be specified and on the basis of this choice the tool will automatically switch between three different algorithms: one for zero rate of variation, another for bounded parameter variations, and a third for arbitrary fast parameter variations. The first two algorithms are based on parameter dependent storage functions for which a set of basis functions need to be specified manually – only polynomial parameter dependency is supported. This way it is easy to test obtainable performance as a function of the size of polynomial expansion.

The design algorithms implemented in the tool show good numerical performance for small and medium sized control problems. Controller design for large scale systems can be very challenging from a numerical point of view and to deal with this critical class systems a number of numerical tuning handles have been implemented. Essentially the tool provides means for bounding specific design variables and/or LMIs which can enhance numerical performance of the construction algorithm.

These numerical tuning handles are meant for advanced users who have detailed knowledge about the implemented algorithm. To guide such users, information about the numerical results of the algorithm are presented in a number of windows to the right of Figure 3.4. The illustrated window shows an example of the results of the synthesis LMIs from where it can be observed that all LMIs are satisfied. Depending on the choice of optimiser the LMIs can be only approximatively satisfied. Then for example if the primal LMIs have positive eigenvalues, an offset can be included to enforce negative definiteness.

3.4.2. Example

An example is provided to give a basic idea of the applicability of the design tool. It considers the design of an LPV tracking controller for a double integrator and the target is to have an aggressive response for large tracking errors and a relaxed response for smaller errors. A block diagram of the considered control problem is presented in Figure 3.5 and the performance criterion will be the trade-off between sensitivity and complementary sensitivity with the trade-off varying with tracking error.

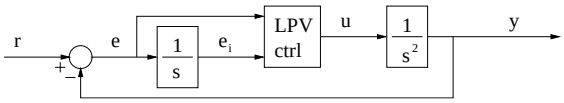


Figure 3.5: Block diagram of interconnection of LTI open loop plant with LPV controller.

The performance weights are illustrated in the bottom left of Figure 3.4 for three values of the scheduling function: $|e| = 0$, $|e| = 0.05$, and $|e| = 1$. These performance

functions have a structure given by

$$W_{out}(s,\delta) = \begin{bmatrix} K_1(\delta)\frac{s+b_1(\delta)}{s+a_1(\delta)} & 0 \\ 0 & K_2(\delta)\frac{s+b_2(\delta)}{s+a_2(\delta)} \end{bmatrix}$$

for each operating condition with the gains and time constants being gain-scheduled. A rate-of-variation of 100 has been chosen and a polynomial expansion of the storage functions have been chosen as

$$X(\delta) = X_0 + \delta\, X_1 + \delta^2\, X_2 \quad , \quad Y(\delta) = Y_0$$

On the basis of this formulation, the weighted model can be created by pushing `Model` and the controller is created by pushing `Synthesis` followed by `Construct`. The designed controller behaviour is demonstrated by a step response of the LPV controller in a comparison with local LTI controllers for each grid point. A simulation result is presented in Figure 3.6 from which it can be observed that the LPV controller responds aggressively for large tracking errors whereas it has a relaxed behaviour for small tracking errors.

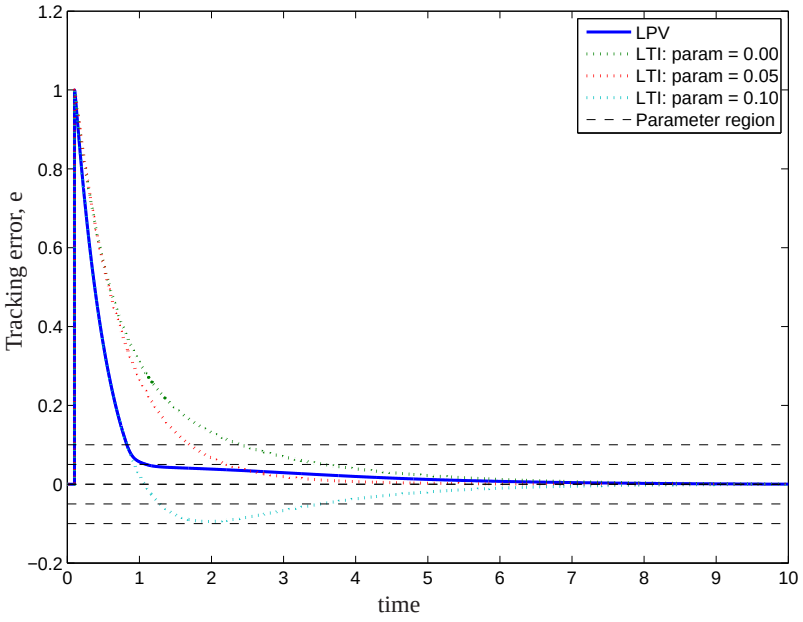


Figure 3.6: Illustration of tracking error for a step response with designed LPV controller and frozen parameter controllers. It can be observed how the LPV controllers resemble the aggressive LTI controller for larger errors whereas it is relaxed and resemble the slow LTI controller for small tracking errors.

3.4.3. Summary

A design tool has been developed for the design of LPV controllers using the grid-based method. The user interface provides a simple interface in which the user can input weighting functions for each considered operating condition together with the desired bound on rate of variation and basis function for the storage function. Then the optimal performance level and the controller can be constructed simply by pushing three buttons. For more advanced users it is further possible to limit bounds on critical variables in order to make the numerical conditioning better for the algorithm used in controller construction.

The applicability of the tool is illustrated by an example which demonstrates the simplicity in its use. The tool has further been verified for the larger problems considered in Paper C and D.

3.5. A unified wind turbine controller for the entire operating region

This section is intended to illustrate one of the main advantages of the LPV design framework for the control of wind turbines. Through gain scheduling it is possible to design one controller for the entire operating region of wind turbines which will be illustrated on the basis of the investigation performed in Paper C.

3.5.1. Preliminaries

For the range of operating conditions it has been decided to focus only on nominal operation which means that we do not take special operating modes into account, e.g. derating of power or generator speed. Because of this, the range of operating conditions can be determined from one exogenous variable, the wind speed, and in Figure 3.7 some of the main variables are illustrated as a function of the wind speed.

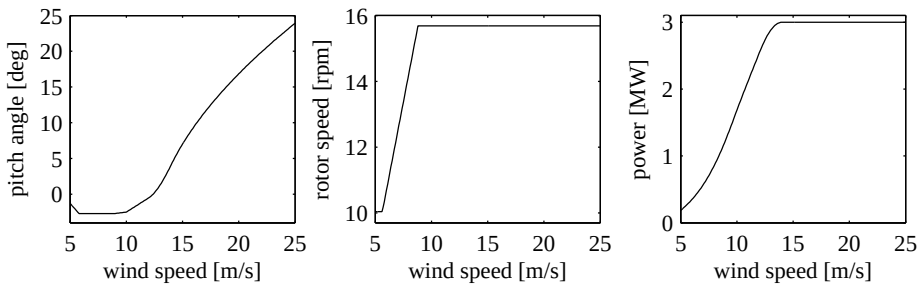


Figure 3.7: Illustration of main variables for operating region of wind turbines.

It is important to note that by choosing the wind speed as scheduling variable the scheduling variable is not directly measurable. There are anemometers mounted on most wind turbines, but the quality of these measurements is not good enough for use

in a high performance controller design. The reason is partly that the measurement is disturbed by the rotor and partly because the measurement is a point wind measurement and not the measurement of the spatial mean of the free wind†. To handle this a wind speed estimator has been developed in Paper A which shows an improved performance in comparison with a typical approach.

The main nonlinearity in the considered model origins from the aerodynamics and is a smooth function of wind speed, pitch angle, and rotor speed. It has therefore been decided to use the linearisation-based method in Section 3.1, and to handle the constant terms from the linearisation the controller structure presented in Figure 3.8 has been applied. The symbols should be interpreted as v is wind speed, ω_g is generator speed, ω_e is tracking error, β_{ref} is pitch reference, $Q_{g,ref}$ is generator torque reference, P_{ref} is power reference, y is tower top movement (fore-aft), and Q_{sh} is torque between low speed shaft and high speed shaft. A dot over a symbol ($\dot{}$) means first derivative, and a hat ($\hat{}$) means an estimate.

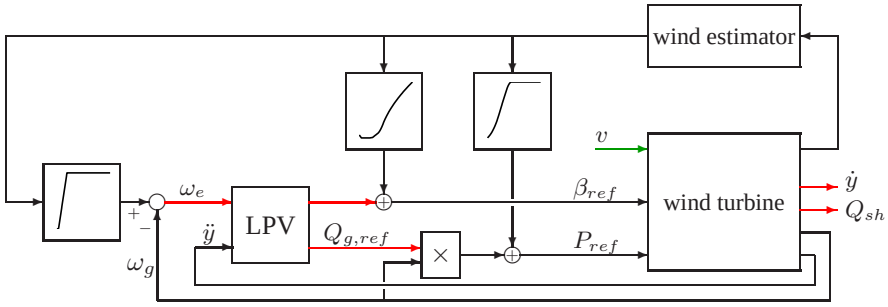


Figure 3.8: Implementation structure for an LPV controller based on linearisation along a trajectory of equilibrium points. Green marks input of performance channel and red marks output of performance channel. The block diagram is simplified and does not illustrate integrators for asymptotic tracking.

3.5.2. LPV controller design

The main performance objective is to trade-off the tracking of the reference trajectory with noise on power and fatigue loads in tower, drive train, and pitch system. The fatigue loads are typically measured by a simulation based method called rain-flow count [83]. For the tower and drive train this can be approximated fairly well by choosing $\dot{y}$ and Q_{sh} as output signals because they have a peak at respectively the tower eigen-frequency and the drive train eigen-frequency. The fatigue in the pitch system can be minimised by a frequency dependent weight on the pitch reference punishing high frequency components. Lastly the noise on active power can be minimised by minimising the generator torque because the generator speed will change slowly in comparison. By choosing the performance outputs this way the performance channel will automatically have a peak where we want emphasis in the design formulation which means that low order weights can be used. In fact this choice of performance outputs means that an LPV controller

† The free wind is defined as the wind if the wind turbines was not there.

can be designed by minimising the induced $\mathcal{L}_2/\mathcal{L}_2$ gain from w to z with

$$z = \begin{bmatrix} K_{\dot{y}}(\hat{v}) & 0 & 0 & 0 & 0 \\ 0 & K_{Q_{sh}}(\hat{v}) & 0 & 0 & 0 \\ 0 & 0 & K_{\omega}(\hat{v}) & 0 & 0 \\ 0 & 0 & 0 & \frac{K_{\beta}(\hat{v})}{(Ts+1)^3} & 0 \\ 0 & 0 & 0 & 0 & K_{Q_g}(\hat{v}) \end{bmatrix} \begin{bmatrix} \dot{y} \\ Q_{sh} \\ \omega_e \\ \beta_{ref} \\ Q_{g,ref} \end{bmatrix}, \quad w = v$$

where $K(\hat{v})$ is a nonlinear function of $\hat{v}$, i.e. the output weight is gain scheduled with wind speed to make the trade-off between tracking and control effort depend on operating condition. Note that the only output signal with a frequency dependent weight is the pitch angle. The main reason for having gain scheduled weighting functions is that the performance requirements for partial load is very different from the specifications for full load. In Paper C this issue is discussed in detail and the main conclusion is that generator torque should be used as the main control signal in partial load whereas the pitch should be used mainly for full load operation. Also the requirements regarding tracking and fatigue load reduction changes with mean wind speed because the amplitude of the disturbance (wind speed) varies in with mean wind speed. Note that the physical requirements are described in absolute numbers and not by an amplification.

The gains $K(\hat{v})$ and the time constant T have been determined through an iterative procedure where the resulting controller has been evaluated through simulations to measure actual fatigue and deviations from reference trajectory.

The resulting controller has been compared to a classical design through simulation studies and in Table 3.1 a summary of a simulation is presented. It can be observed that the tower movement is slightly increased when comparing with the classical controller. In this wind speed range tower oscillations are not very high in absolute numbers and the slight increase is therefore not expected to have a significant impact on the life time of the tower. A more interesting observation is that drive train loads are reduced and that the travel of the pitch system is almost halved without reducing the power production.

Table 3.1: Comparison of an LPV controller and a classical controller at low wind speeds. Damage is normalised according to level of classical controller. The increased tower damage during low wind speeds is not critical because the absolute level is small in this operating range.

	Tower damage	Drive train damage	Pitch travel	avg. Power
LPV	1.13	0.80	$13 \cdot 10^9$ deg	918 kW
Classic	1.00	1.00	$23 \cdot 10^9$ deg	906 kW

A similar result is presented in Table 3.2 for the mid wind speed range where the generator speed is kept close to rated speed but where rated power has not been reached. In this range a significant reduction can be observed in the tower fatigue damage together with a slight reduction in drive train fatigue and pitch travel. Further, this is obtained without increasing the variations in tracking of generator speed reference or reducing the power production.

For full load operation a summary is presented in Table 3.3 from which it can be observed that drive train oscillations are significantly reduced with a slight reduction in pitch travel and fatigue damage on tower. Also the generator speed variations and power fluctuations are reduced significantly.

Table 3.2: Comparison of an LPV controller and a classical controller at medium wind speeds. Damage is normalised according to level of classical controller.

	Tower dam.	Drive train dam.	Pitch travel	Speed peak-peak	avg. power
LPV	0.52	0.90	$288 \cdot 10^9$ deg	218 rpm	2419 kW
Classic	1.00	1.00	$310 \cdot 10^9$ deg	221 rpm	2416 kW

Table 3.3: Comparison of an LPV controller and a classical controller at high wind speeds. Damage is normalised according to level of classical controller.

	Tower dam.	Drive train dam.	Pitch travel	Speed peak-peak	Power std.
LPV	0.81	0.57	$603 \cdot 10^9$ deg	188 rpm	17.3 MW
Classic	1.00	1.00	$558 \cdot 10^9$ deg	293 rpm	23.2 MW

3.5.3. Conclusions

In conclusion it can be observed that local simulations show that the LPV controller increases the performance for all three ranges of operating conditions when comparing with a controller designed by classical methods. Most importantly it should be noted that pitch travel is decreased in low wind speeds, tower oscillations are decreased in mid wind speeds and drive train loads and generator speed variations are reduced at high wind speeds.

Concerning the behaviour for operating conditions varying between the three operating conditions a simulation is illustrated in Figure 3.9 which traverses all three operating conditions. From the graphs it can be seen that for example the transition between partial load and full load is smooth as a consequence of the gain scheduling as opposed to switching, i.e. no transients are observed in the simulation.

3.6. Influence of parameter rate bounds on gain scheduled control of wind turbines

The design performed in Section 3.5 was done with a method assuming arbitrary fast parameter variations. During this design it was experienced that the underlying assumption of arbitrary fast parameter variations is restrictive for this particular application and this section is provided to investigate if the use of finite rate bounds will enhance closed loop performance.

3.6.1. Preliminaries

A first observation in the design presented in the previous section was that the performance levels differ significantly between local LTI controller design and the LPV design over the range of operating conditions. This large difference is either because the problem is intrinsically difficult when taking parameter variations into account, or caused by conservatism introduced by the underlying assumptions or numerics. In the following it is investigated how the assumption of arbitrary fast parameter variations influences the obtainable performance level.

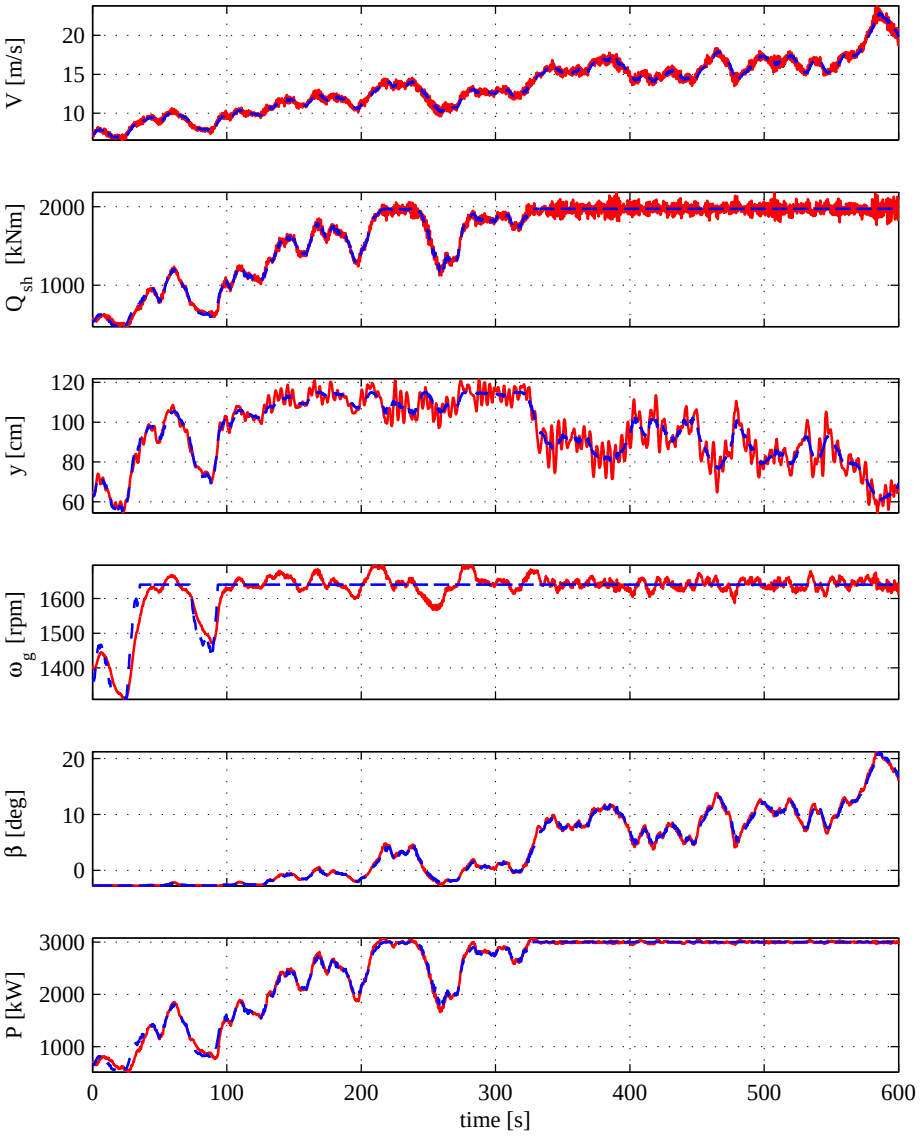


Figure 3.9: Example of simulation with stochastic wind input in a ramp form. Dashed black: reference trajectory, full grey: simulated response. It can be observed how the closed loop tracks the reference trajectory smoothly with only small fluctuations on power in full load operation.

In Figure 3.10 a set of local LTI controllers are illustrated in red for the tracking of a generator speed reference. In the same figure the LPV controller is sampled at the same operating conditions and it can be observed that there is only an insignificant difference between these local samples of the LPV controller whereas the LTI controllers differ significantly. This large coupling between the local controllers with the LPV method is the reason for the local performance degradation and is caused by the assumption of constant storage functions.

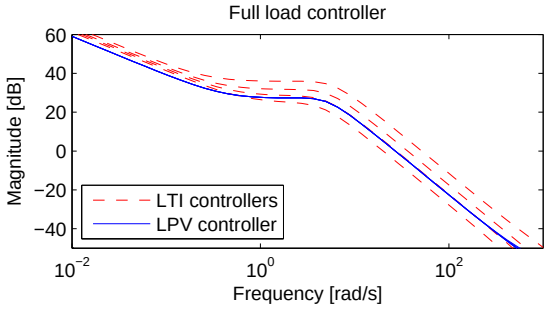


Figure 3.10: Illustration of difference between optimal controllers designed by LTI methods and an LPV method with arbitrary fast parameter variations.

It has been observed that it is possible to separate the local controllers obtained by the LPV method by exaggerating the performance weights. Essentially this means that it is possible to obtain LPV controllers which satisfy the requirements from a simulations point of view. The downside is that weighting functions do no longer make sense from an LTI point of view which makes it even harder to translate physical requirements into weighting functions. Further, with the large difference between LTI performance and LPV performance the well known LTI tools for analysing the closed loop performance no longer applied.

To investigate this difference, a gain scheduled controller was designed in Paper B using the classical method of interpolation of LTI controllers in equivalent coordinate systems. This investigation showed good performance from a simulations point of view and this indicates that the assumption of arbitrary fast parameter variations might be conservative.

3.6.2. LPV design with rate bounds

As a consequence of the preliminary studies described above it can be concluded that the gap between the LTI and LPV designs should be investigated further. One way to do this is using rate bounded parameter variations for the controller design. To simplify the investigation the design will only focus on full load control and the tracking of generator speed and active power while minimising the drive train loads. The details of the investigation is presented in Paper D and in the following an introduction will be given to the main idea and the obtained results.

When comparing with classical approaches it can be concluded from the study in Paper C that no significant benefit could be obtained regarding drive train damage at the drive train eigen-frequency. This conclusion was based on simulation studies and the

reason is mainly that the pitch actuator can not be used around these high frequencies because it would wear out the actuator too quickly. It has therefore been chosen to use a classical approach for the drive train damper similar to what is done in [17]. The main idea is to feed the generator speed to the generator torque through a band pass filter around the drive train eigen-frequency. When the drive train starts oscillating at the lightly damped eigen-frequency, the generator torque will be updated to counter the oscillations.

With this drive train damper the controller structure becomes as illustrated in Figure 3.11 and the objective for the LPV controller simplifies to a trade-off between pitch travel and tracking of the generator speed. Further, the drive train model can be described by a first order system as a consequence of the included drive train damper.

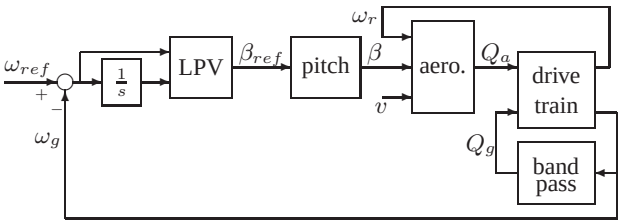


Figure 3.11: Controller structure for full load control when considering trade-off between tracking of generator speed and pitch activity while minimising drive train oscillations.

The LPV controller can then be designed for the high wind speed range with a trajectory of nominal operating conditions as in Figure 3.7, for wind speeds above $15\ m/s$, i.e. rotor/generator speed and active power at rated values. The model is obtained as in Section 3.1 with the linearisation based method and the design tool in Section 3.4 has been used to synthesise and construct a controller. For the performance specification effective wind speed has been chosen as performance input, and control effort (pitch movement) and integral of tracking error have been chosen as performance outputs. A gain scheduled frequency independent weight has been applied to the tracking error and a gain scheduled dynamic weight has been applied for the control effort as shown in (3.4). The symbol $\hat{v}$ represents the scheduling variable, effective wind speed, which can be obtained from the estimator described in Paper A.

$$w(s) = v(s) \quad , \quad z(s,\hat{v}) = \begin{bmatrix} K_\omega(\hat{v}) \cdot \frac{1}{s} \cdot \Big(\omega_{ref}(s) - \omega_g(s)\Big) \\ K_\beta(\hat{v}) \cdot \Big(\frac{T(\hat{v})s+1}{\epsilon s+1}\Big)^3 \cdot \beta_{ref}(s) \end{bmatrix} \tag{3.4}$$

The three functions K_ω, K_β, and T have been determined through an iterative procedure using local LTI designs along the nominal trajectory. This ensures that a performance level close to one will give a closed loop performance similar to what can be obtained locally for LTI controllers.

When designing controllers for rate bounded parameter variations it is necessary to use parameter dependent storage functions and in this context it is important to choose the right basis function for the storage function. To evaluate the choice of performance function a design has been performed for zero rate of variation and it can be observed that the performance degradation when comparing with LTI designs vary between 50% for constant storage functions to 0.2% for $X(\hat{v}) = X_0$ and $Y(\hat{v}) = Y_0 + \hat{v}Y_1 + \hat{v}^2Y_2$.

The latter has been selected as basis for the storage function and controllers have been synthesised for various rates of variation to give a relation between rate of variation and obtainable performance level illustrated in Figure 3.12. It can be observed from the figure that the reduction in performance level is less than 6% for rates of variation below $1\ m/s^2$ whereas the performance quickly degrades for larger rates of variation until the worst case of approximately 50% reduction is reached.

From physical experiments it is known that the rate of variation of the effective wind speed is bounded well below $100\ m/s^2$ which means it is conservative not to take the rate bound into consideration in the design. It is suggested to use a limit around $1\ m/s^2$ because this value gives a good local performance and it is rare that the rate of variation will exceed this value. In special operating conditions like in the case of large gusts the rate bound can be exceeded, but these events happen rarely and it is very unlikely that the event will happen before the transient of the previous event has died out. It is therefore estimated that it is better to focus on good nominal performance for a bounded rate of variation and then deal with the transient response of the special events individually.

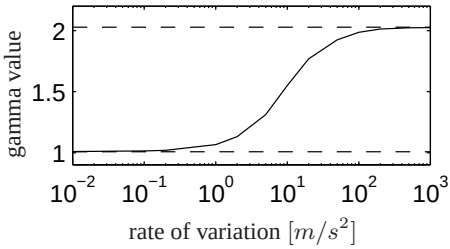


Figure 3.12: Guaranteed $\mathcal{L}_2$ gain as a function of tolerated parameter rate of variation. Dashed lines indicate indicate lower and upper bounds given by respectively zero and arbitrary fast rate of variation.

In Figure 3.13 a comparison by simulation of the LPV controller with an LTI controller is given. The LTI controller has been selected for the high wind speed range and it can be observed that the two controllers perform similarly for this range of operating conditions. At the end of the time series it can be observed that the LTI controller starts oscillating whereas the LPV controller behaves nicely. The reason for this oscillation is that the operating condition is too far from the point for which the LTI controller has been designed.

3.6.3. Conclusions

From the analysis, it can be concluded that the LPV controller is indeed superior to the LTI controllers from the global perspective. From the local perspective the LPV controller has been compared through simulations to LTI controllers designed for each grid point. A summary of this comparison is presented in Table 3.4 from which it can be observed that the local performance of the LPV controller is only slightly worse than what can be obtained locally by LTI controllers. Because of this the controller design has been concluded to be successful.

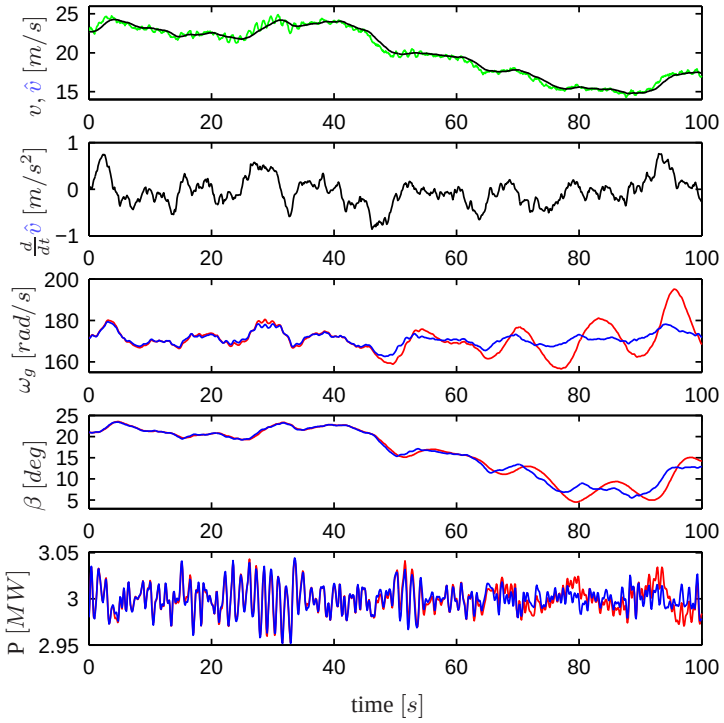


Figure 3.13: Simulation results with LPV controller with rate of variation up to 1 m/s together with LTI controller. Green: Effective wind speed. Black: Scheduling variable. Blue: Closed loop with LPV controller. Red: Closed loop with LTI controller. Observe difference in damping of oscillations for low and high wind speeds.

Table 3.4: Selected performance outputs of the LPV controller measured relative to the LTI controllers.

mean wind	RFC. drt.	gen. spd.	pitch	P std.
15 m/s	101 %	104 %	96 %	99 %
18 m/s	97 %	92 %	101 %	100 %
21 m/s	97 %	94 %	100 %	100 %
25 m/s	97 %	90 %	104 %	100 %

Chapter 4

Conclusions and Perspectives

Contents

This thesis has demonstrated how gain scheduled controllers can be designed for a practical application using a systematic design method denoted linear parameter varying (LPV) control. The numerics play an important role when using the methods in practice and a number of numerical issues have been pointed out with directions for the solution of a selection of these. In Section 4.1 the contributions of the thesis will be summarised and in Section 4.2 it will be discussed to what extend the objectives have been met. Then in Section 4.3 a recommendation for future directions is given.

4.1. Contributions

The main contributions of the thesis are listed in the following:

Modelling of LPV systems:

Modelling of nonlinear systems can be very complicated because all information is not a-priori available and parameters or nonlinear functions need to be identified. A modularised procedure has been proposed for the modelling of nonlinear systems using LPV models. Further an experimental method and a method based on a desired trajectory of operating conditions have proposed for the determination of the region of parameter values. The methods have been demonstrated by the application to modelling of a wind turbine.

Formulation for feasible control of quasi-LPV systems:

It is shown that the performance level of LPV controllers is bounded by the performance level obtainable locally by LTI controllers. For quasi-LPV formulations this is particularly important because the choice of frozen parameter dynamics is not unique. A procedure has been formulated in which a generalised quasi-LPV formulation should be found to cover all quasi-LPV formulations. Then the frozen parameter properties of the different quasi-LPV models should be analysed before LPV synthesis to determine an appropriate representation.

Numerical conditioning of design algorithm:

Several algorithms have been investigated for the design of LPV controllers and it has been concluded that an adapted version of the classical grid-based method provides the best numerical performance. Concerning the controller construction, the numerical performance of several adapted algorithms have been analysed and a numerical solution has been found which is based on a classical result from $\mathcal{H}_\infty$ control.

Design tool for LPV control:

The findings regarding numerical conditioning of the design algorithms have been implemented in a new design tool. This tool has made it possible for non-experts in the field to use the systematic approach to design gain-scheduled controllers. For very demanding problems it is further possible for advanced users to tweak the numerical algorithms to enhance numerical performance.

A unified wind turbine controller for entire operating region:

The control of wind turbines usually requires the design of one controller for partial load operation and another controller for full load operation. It has been shown that with the LPV framework this is not necessary and a controller has been designed to cover the full range of operating conditions. The effective wind speed is used as scheduling variable, and since this variable is not measurable a method has been provided for online estimation of it. Through simulation studies it has been concluded that not only does the LPV controller show a smooth transition from partial load to full load. The fatigue loads can be reduced for all nominal operating conditions when compared with a classical controller – without affecting the power quality/production.

Influence of parameter rate bounds on control of wind turbines:

Through the design of a gain-scheduled LQ controller it has been shown that classical gain-scheduled controllers can provide good properties for the control of wind turbines in normal operation. It has been shown that the guaranteed performance of an LPV controller varies significantly as a function of assumed worst case rate of variation. An LPV controller has been designed for a bound of $1\ m/s^2$ and to guarantee a performance that is only $10\ \%$ worse than what can be obtained locally by LTI controllers. Simulation results support this and demonstrates that the performance of the LTI controllers deteriorates quickly when moving away from the design point.

4.2. Reflection on initial objectives

The main focus of the study has been to determine a systematic method with which it is possible to improve the life time of the main components of wind turbines without sacrificing on the power production. In Section 1.3 a number of commercial objectives were set up, and scientific objectives were defined to be able to modify available tools so that the commercial objectives can be met. In the following it will first be discussed to what extent the scientific objectives were met and then a similar discussion is presented for the commercial objectives.

4.2.1. Scientific objectives

The scientific objectives are mainly related to the development of tools with which it is possible to design LPV controllers in a numerically reliable way. With numerically reliable is meant that the control engineer should be able to focus on performance specifications and tuning the controller and not modifications of design algorithms.

Investigation of numerical properties of LPV design methods

A number of results have been obtained regarding this issue. First of all it has been identified that a grid-based algorithm is superior to an LFT-based algorithm from a numerical point of view. For the controller construction it has been identified that a generalisation of the algorithm from LTI $\mathcal{H}_\infty$ control is superior to an algorithm with construction of extended storage functions – even with a set of proposed modifications of the latter algorithm. Finally it has been identified that the choice of state space realisation has a large

impact on numerical performance of the algorithms. A recommendation has been made regarding the choice of realisation but more research is necessary in order to give final conclusions about this issue.

Understanding the influence of rate-of-variation of the scheduling parameter on obtainable performance level

It was discovered that the design algorithm can result in controllers that are very similar along the trajectory of operating conditions. This essentially means that the controller is designed to be robust against the parameter variation and not gain-scheduled to handle it. The main cause of this is that the storage function is independent on the scheduling parameter and it was experienced that with rate bounded parameter variations the controllers would spread out more resulting in a significant improvement in performance level. This indicates that the assumption of arbitrary fast parameter variations can be very conservative and rated bounded variations should be taken into account.

Application of the theory of linear parameter varying systems

The design algorithm has indeed been applied to the design of LPV controllers using a higher order parameter dependency. The applied method can handle any parameter dependency as long as the parameter is exogenous and resides in an a-priori known set.

Adaption of modern estimation theory to wind turbines

A modularised state estimator has been proposed in Paper B consisting of two unscented Kalman filters and an LTI Kalman filter interconnected with a PI component for the estimation of the aerodynamic torque. Further a novel approach has been proposed for the calculation of effective wind speed from aerodynamic torque, rotor speed and pitch angle.

4.2.2. Commercial objectives

The obtained scientific results can then be applied to the specific application and in the following it will be discussed to what extent the commercial objectives are met.

Reduction of loads with power curve kept constant

The results from the presented controller designs show significant potential for the LPV design method in the application of fatigue reduction on wind turbines. The controller designed for both partial load and full load operating shows a reduction in wear in pitch system, tower damage and drive train damage without reduction in power level/quality. Similar results are obtained in the full load controller with bounded parameter rate of variation.

A unified controller for the entire operating region

A single LPV controller has been designed for both partial load operation and full load operation to obtain a smooth transition between the two operating modes. Simulation studies show smooth behaviour over the entire operating region.

Simplified tuning process

In the performance specification it is possible to weight each component individually, e.g. generator speed variations, tower oscillations and pitch variations. This means that the tuning handles are closely related to the physical requirements when comparing with just tuning the gains of the interconnection of PID controllers. Further a design tool has been developed so that the designer can focus on the application and not the technical details of the algorithm.

Robustness towards under-modelling

Since the applied LPV method is a generalisation of $\mathcal{H}_\infty$ control it offers a direct way of including model uncertainty both in terms of parametric uncertainty and neglected dynamics. At this point no uncertainty modelling has been performed due to time limitations and the robustness has therefore not been included in the controller design.

4.2.3. Summary on reflections

On the basis of the above discussion of the level at which the scientific objectives have been met it is concluded that the project is a success from a scientific point of view. Further from a commercial point of view a great potential has been demonstrated for the method and the project is concluded to be successful also from an industrial point of view.

4.3. Perspectives

Some of the main work in this thesis has been to work towards numerically reliable algorithms for the design of LPV controllers. These algorithms are still not matured completely and need further research in a number of areas. Further for the application of LPV control of wind turbines also a number of suggestions for future directions can be given.

Optimal state space realisation for LPV control

It has been experienced that the choice of state space realisation has a large influence on the numerical performance of both controller synthesis and controller construction. Further research is required to understand which realisation to choose in order to get numerically reliable controller design.

Reliable computation of parameterised LPV controller

There is a large difference in numerical performance between the presented methods for parameterised controller computation and the controller computation for a particular parameter value. It would be a great advantage to have a unified numerically reliable algorithm for the construction of a parameterised controller.

Investigate LPV controller on real wind turbine

To get commercial value of the project it is required to get the controllers running on real wind turbines which most likely will require an additional number of iterations. Before getting to this, a detailed simulation study should be be performed in which the model size is increased slightly and detailed investigations of the interconnection between the parameter estimator and the LPV controller should also be performed.

Extensions to derating control, scheduling on fatigue level

A great advantage of gain scheduling is that the controller can be adapted to changing operating conditions. This means that different wind turbines experiencing different load levels can have their operational mode reflect the experienced fatigue loads.

Bibliography

[1] P. Apkarian and R. J. Adams. Advanced gain-scheduling techniques for uncertain systems. In *American Control Conference*, pages 3331–3335, 1997. http://dx.doi.org/10.1109/87.654874.

[2] P. Apkarian and R. J. Adams. Advanced gain-scheduling techniques for uncertain systems. *Trans. on Control Systems Technology*, 6(1):21–32, 1998. http://dx.doi.org/10.1109/87.654874.

[3] P. Apkarian and P. Gahinet. A convex characterization of gain-scheduled $\mathcal{H}_\infty$ controllers. *Trans. on Automatic Control*, 40(5):853–864, 1995. http://dx.doi.org/10.1109/9.384219.

[4] P. Apkarian, P. Gahinet, and G. Becker. Self-scheduled $\mathcal{H}_\infty$ control of linear parameter-varying systems – a design example. *Automatica*, 31(9):1251–1261, 1995. http://dx.doi.org/10.1016/0005-1098(95)00038-X.

[5] P. Apkarian, P. C. Pellanda, and H. D. Tuan. Mixed $\mathcal{H}_2/\mathcal{H}_\infty$ multi-channel linear parameter-varying control in discrete time. *Systems and Control Letters*, 41:333–346, 2000. http://dx.doi.org/10.1016/S0167-6911(00)00076-1.

[6] P. Apkarian and H. D. Tuan. Parameterized LMIs in control theory. *SIAM J. of Control and Optimization*, 38(4):1241–1264, 2000.

[7] T. Azuma, R. Watanabe, K. Uchida, and M. Fujita. A new LMI approach to analysis of linear systems depending on scheduling parameter in polynomial forms. *Theoretische Arbeit*, 48(4):199–204, 2000. http://dx.doi.org/10.1524/auto.2000.48.4.199.

[8] G. J. Balas. Linear, parameter-varying control and its applications to a turbofan engine. *Int. J. of Robust and Nonlinear Control*, 12:763–796, 2002. http://dx.doi.org/10.1002/rnc.704.

[9] M. J. Balas, A. D. Wright, M. M. Hand, and K. A. Stol. Dynamics and control of horizontal axis wind turbines. In *American Control Conference*, pages 3781–3793, 2003. http://dx.doi.org/10.1109/ACC.2003.1240423.

[10] G. Becker and A. Packard. Robust performance of linear parametrically varying systems using parametrically-dependent linear feedback. *Systems and Control Letters*, 23:205–215, 1994. http://dx.doi.org/10.1016/0167-6911(94)90006-X.

[11] F. D. Bianchi, H. De Battista, and R. J. Mantz. *Wind Turbine Control Systems*. Springer, 2007.

[12] F. D. Bianchi, R. J. Mantz, and C. F. Christiansen. Control of variable-speed wind turbines by LPV gain scheduling. *Wind Energy*, 7:1–8, 2004. http://dx.doi.org/10.1002/we.103.

[13] F. D. Bianchi, R. J. Mantz, and C. F. Christiansen. Power regulation in pitch-controlled variable-speed WECS above rated wind speed. *Renewable Energy*, 29:1911–1922, 2004. http://dx.doi.org/10.1016/j.renene.2004.02.009.

[14] P. M. M. Bongers. Robust control using coprime factorizations application to a flexible wind turbine. In *Conf. on Decision and Control*, pages 2436–2441, 1992.

[15] P. M. M. Bongers. *Modeling and Identification of Flexible Wind Turbines and a Factorizational Approach to Robust Control*. PhD thesis, Delft University of Technology, 1994.

[16] P. M. M. Bongers, G. E. van Baars, and S. Dijkstra. Load reduction in a wind energy conversion system using an $\mathcal{H}_\infty$ control. In *Conf. on Control Applications*, pages 965–970, 1993.

[17] E. A. Bossanyi. The design of closed loop controllers for wind turbines. *Wind Energy*, 3:149–163, 2000. http://dx.doi.org/10.1002/we.34.

[18] E. A. Bossanyi. Individual blade pitch control for load reduction. *Wind Energy*, 6:119–128, 2003. http://dx.doi.org/10.1002/we.76.

[19] E. A. Bossanyi. Wind turbine control for load reduction. *Wind Energy*, 6:229–244, 2003. http://dx.doi.org/10.1002/we.95.

[20] E. A. Bossanyi. Further load reduction with individual pitch control. *Wind Energy*, 8:481–485, 2005. http://dx.doi.org/10.1002/we.166.

[21] C. L. Bottasso, A. Croce, and B. Savini. Performance comparison of control schemes for variable-speed wind turbines. *Journal of Physics: Conference Series*, 75, 2007. http://dx.doi.org/10.1088/1742-6596/1/012079.

[22] A. Bouali, P. Chevrel, and M. Yagoubi. About gain scheduled state feedback controllers for rational lpv systems. In *Int. Conf. on Control, Automation, Robotics and Vision*, pages 1–6, 2006. http://dx.doi.org/10.1109/ICARCV.2006.345380.

[23] B. Boukhezzar and H. Siguerdidjana. Nonlinear control of variable speed wind turbines without wind speed measurement. In *Conf. on Decision and Control*, pages 3456–3461, 2005.

[24] G. C. Calafiore, F. Dabbene, and R. Tempo. Randomized algorithms for probabilistic robustness with real and complex structured uncertainty. *Trans. on Automatic Control*, 45(12):2218–2235, 2000. http://dx.doi.org/10.1109/9.895560.

[25] P. W. Carlin, E. B. Muljadi, and A. S. Laxson. The history and state of the art of variable-speed wind turbine technology. *Wind Energy*, 6:129–159, 2003. http://dx.doi.org/10.1002/we.77.

[26] Hugo Chandler, editor. *Wind Energy, the Facts – An Analysis of Wind Energy in the EU-25*, volume 1. EWEA publishing, 2004.

[27] G. Chesi, A. Garulli, Alberto Tesi, and A. Vicino. Robust stability of time-varying polytopic systems via parameter-dependent homogenous lyapunov functions. *Automatica*, 43:309–316, 2007. http://dx.doi.org/10.1016/j.automatica.2006.08.024.

[28] S. B. Christensen and N. P. Ottsen. Wind turbine control – a model predictive control approach. Master's thesis, Aalborg University, 2004.

[29] B. Connor, S. N. Iyer, W. E. Leithead, and M. J Grimble. Control of a horizontal axis wind turbine using $\mathcal{H}_\infty$ control. In *Conf. on Control Applications*, pages 117–122, 1992. http://dx.doi.org/10.1109/CCA.1992.269889.

[30] B. Connor, W. E. Leithead, and M. J. Grimble. LQG control of a constant speed horizontal axis wind turbine. In *Conf. on Control Applications*, pages 251–252, 1994. http://dx.doi.org/10.1109/CCA.1994.381190.

[31] C. Courties and J. Bernussou. LPV control by dynamic output feedback. In *American Control Conference*, pages 2267–2271, 1999. http://dx.doi.org/10.1109/ACC.1999.786412.

[32] N. A. Cutululis, E. Ceanga, A. D. Hansen, and P. Sørensen. Robust multi-model control of an autonomous wind power system. *Wind Energy*, 9:399–419, 2006. http://dx.doi.org/10.1002/we.194.

[33] A. Dadone, L. Dambrosio, and B. Fortunato. One step ahead adaptive control technique for wind systems. *Energy Conversion and Management*, 1998. http://dx.doi.org/10.1016/S0140-6701(98)80423-2.

[34] C. E. de Souza and A. Trofino. Gain-scheduled $\mathcal{H}_2$ controller synthesis for linear parameter varying systems via parameter dependent lyapunov functions. *Int. J. of Robust and Nonlinear Control*, 16:243–257, 2006. http://dx.doi.org/10.1002/rnc.1040.

[35] C. E. de Souza, A. Trofino, and J. de Oliveira. Parametric lyapunov function approach to $\mathcal{H}_2$ analysis and control of linear parameter-dependent systems. In *Conf. on Control Theory and Applications*, pages 501–508, 2003. http://dx.doi.org/10.1049/ip-cta:20030709.

[36] A. Dixit and S. Suryanarayanan. Towards pitch-scheduled drive train damping in variable-speed, horizontal-axis large wind turbines. In *Conf. on Decision and Control*, pages 1295–1300, 2005.

[37] Darrell M. Dodge. Illustrated history of wind power development, oct 2007. http://telosnet.com/wind/.

[38] L. Dodson, K. Busawon, and M. Jovanovic. Estimation of the power coefficient in a wind conversion system. In *Conf. on Decision and Control*, pages 3450–3455, 2005.

[39] K. Dong and F. Wu. Robust and gain-scheduling control of LFT systems through duality and conjugate lyapunov functions. *Int. J. of Control*, 80(4):555–568, 2007. http://dx.doi.org/10.1080/00207170601080213.

[40] J. C. Doyle, K. Glover, P. P. Khargonekar, and B. A. Francis. State-space solutions to standard $\mathcal{H}_2$ and $\mathcal{H}_\infty$ control problems. *Trans. on Automatic Control*, 34(8):831–847, 1989.

[41] Danish Wind Industry Association DWIA, oct 2007. http://www.windpower.org.

[42] T. Fischer, M. Kühn, and P. Passon. Load mitigation of aerodynamically and hydrodynamically induced loads of offshore wind turbines. In *European Wind Energy Conference*, 2007.

[43] Y. Fujisaki, F. Dabbene, and R. Tempo. Probabilistic design of LPV control systems. *Automatica*, 39:1323–1337, 2003. http://dx.doi.org/10.1016/S0005-1098(03)00108-0.

[44] P. Gahinet. Explicit controller formulas for LMI-based $\mathcal{H}_\infty$ synthesis. *Automatica*, 32(7):1007–1014, 1996. http://dx.doi.org/10.1016/0005-1098(96)00033-7.

[45] P. Gahinet and P. Apkarian. A linear matrix inequality approach to $\mathcal{H}_\infty$ control. *Int. J. of Robust and Nonlinear Control*, 4:421–448, 1994. http://dx.doi.org/10.1002/rnc.4590040403.

[46] P. Gahinet, P. Apkarian, and M. Chilali. Affine parameter-dependent lyapunov functions and real parametric uncertainty. *Trans. on Automatic Control*, 41(3):436–442, 1996. http://dx.doi.org/10.1109/9.486646.

[47] P. Gahinet, A. Nemirovski, A. J. Laub, and M. Chilali. The LMI control toolbox. In *Conf. on Decision and Control*, pages 2038–2041, 1994. http://dx.doi.org/10.1109/CDC.1994.411440.

[48] M. Geyler and P. Caselitz. Individual blade pitch control design for load reduction on large wind turbines. In *European Wind Energy Conference*, 2007.

[49] M. J. Grimble. Two and a half degree of freedom LQG controller and application to wind turbines. *Trans. on Automatic Control*, 39(1):122–127, 1992. http://dx.doi.org/10.1109/9.273347.

[50] M. M. Hand and M. J. Balas. Load mitigation control design for a wind turbine operating in the path of vortices. In *The Science of Making Torque From Wind*, 2004.

[51] M. Maureen Hand and M. J. Balas. Systematic controller design methodology for variable-speed wind turbines. Technical report, NREL, 2002.

[52] A. Helmerssonn. IQC synthesis based on inertia constraints. In *IFAC World Congress*, 1999.

[53] L. C. Henriksen. Model predictive control of a wind turbine. Master's thesis, Technical University of Denmark, 2007.

[54] T. Iwasaki and R. E. Skelton. All controllers for the general $\mathcal{H}_\infty$ control problem: LMI existence conditions and state space formulas. *Automatica*, 30(8):1307–1317, 1994. http://dx.doi.org/10.1016/0005-1098(94)90110-4.

[55] T. Iwasaki, P. Tsiotras, and X. Zhang. State-feedback controller synthesis for parameter-dependent LTI systems. In *American Control Conference*, pages 593–597, 2005. http://dx.doi.org/10.1109/ACC.2005.1470021.

[56] Peter James and Nick Thorpe. *Ancient Inventions*. Ballantine Books, New York, 1994. ISBN: 0-345-36476-7.

[57] M. Jelavic, N. Peric, I. Petrovic, S. Car, and M. Madercic. Design of a wind turbine pitch controller for loads and fatigue reduction. In *European Wind Energy Conference*, 2007.

[58] K. E. Johnson. Adaptive torque control of variable speed wind turbines. Technical report, NREL, 2004.

[59] K. E. Johnson, L. J. Fingersh, M. J. Balas, and L. Y. Pao. Methods for increasing region 2 power capture on a variable speed HAWT. In *ASME Wind Energy Symposium*, 2004.

[60] R. E. Kalman. Lyapunov functions for the problem of lur'e in automatic control. In *Proc. Nat. Acad. Sci.*, volume 49, pages 201–205, 1963.

[61] T. Knudsen, P. Andersen, and S. Tøffner-Clausen. Comparing PI and robust pitch controllers on a 400 kW wind turbine by full scale tests. In *European Wind Energy Conference*, 1997.

[62] N. Kodama, T. Matsuzaka, and N. Inomata. Power variation control of a wind turbine generator using probalistic optimal control, including feed-forward control from wind speed. *Wind Engineering*, 24(1):13–23, 2000.

[63] M. Komatsu, H. Miyamoto, H. Ohmori, and A. Sano. Output maximization control of wind turbine based on extremum control strategy. In *American Control Conference*, pages 1739–1740, 2001. http://dx.doi.org/10.1109/ACC.2001.945982.

[64] I. Kraan and P. M. M. Bongers. Control of a wind turbine using several linear robust controllers. In *Conf. on Decision and Control*, pages 1928–1929, 1993. http://dx.doi.org/10.1109/CDC.1993.325530.

[65] T. J. Larsen, H. A. Madsen, and K. Thomsen. Active load reduction using individual pitch, based on local blade flow measurements. *Wind Energy*, 8:67–80, 2005. http://dx.doi.org/10.1002/we.141.

[66] D. J. Leith and W. E. Leithead. Application of nonlinear control to a HAWT. In *Conf. on Control Applications*, pages 245–250, 1994. http://dx.doi.org/10.1109/CCA.1994.381191.

[67] D. J. Leith and W. E. Leithead. Appropriate realization of gain-scheduled controllers with application to wind turbine regulation. *Int. J. of Control*, 65(2):223–248, 1996. http://dx.doi.org/10.1080/00207179608921695.

[68] D. J. Leith and W. E. Leithead. On formulating nonlinear dynamics in LPV form. In *Conf. on Decision and Control*, pages 3526–3527, 2000. http://dx.doi.org/10.1109/CDC.2000.912250.

[69] D. J. Leith and W. E. Leithead. Survey of gain-scheduling analysis and design. *Int. J. of Control*, 73(11):1001–1025, 2000. http://dx.doi.org/10.1080/002071700411304.

[70] W. E. Leithead and B. Connor. Control of a variable speed wind turbine with induction generator. In *Int. Conf. on Control*, pages 1215–1220, 1994. http://dx.doi.org/10.1049/cp:19940310.

[71] W. E. Leithead and B. Connor. Control of variable speed wind turbines: design task. *Int. J. of Control*, 73(13):1189–1212, 2000. http://dx.doi.org/10.1080/002071700417849.

[72] W. E. Leithead and B. Connor. Control of variable speed wind turbines: dynamic models. *Int. J. of Control*, 73:1173–1188, 2000. http://dx.doi.org/10.1080/002071700417830.

[73] W. E. Leithead, S. A. de la Salle, D. Reardon, and M. J. Grimble. Wind turbine modelling and control. In *Int. Conf. on Control*, pages 1–6, 1991.

[74] W. E. Leithead and S. D. Ruiz. Controller design for the cancellation of the tower fore-aft mode in a wind turbine. In *Conf. on Decision and Control*, pages 1276–1281, 2005.

[75] F. Lescher, H. Camblong, O. Curea, and R. Briand. LPV control of wind turbines for fatigue loads reduction using intelligent micro sensors. In *American Control Conference*, pages 6061–6066, 2007. http://dx.doi.org/10.1109/ACC.2007.4282790.

[76] F. Lescher, J. Y. Zhao, and P. Borne. Robust gain scheduling controller for pitch regulated variable speed wind turbine. *Studies in Informatics and Control*, 14(4):299–315, 2005.

[77] F. Lescher, J. Y. Zhao, and P. Borne. Switching LPV controllers for a variable speed pitch regulated wind turbine. *Int. J. Computers, Communications & Control*, 1(4):73–84, 2006.

[78] F. Lescher, J. Y. Zhao, and A. Martinez. Multiobjective $\mathcal{H}_2/\mathcal{H}_\infty$ control of a pitch regulated wind turbine for mechanical load reduction. In *Int. Conf. on Renewable Energies and Power Quality*, 2006.

[79] S. Lim. *Analysis and control of linear parameter-varying systems*. PhD thesis, Stanford University, 1998.

[80] X. Ma. *Adaptive Extremum Control and Wind Turbine Control*. PhD thesis, Technical University of Denmark, 1997.

[81] X. Ma, N. K. Poulsen, and H. Bindner. Estimation of wind speed in connection to a wind turbine. Technical report, Technical University of Denmark, 1995.

[82] R. J. Mantz, F. D. Bianchi, and C. F. Christiansen. Gain scheduling control of variable-speed wind energy conversion systems using quasi-LPV models. *Control Engineering Practice*, 13:247–255, 2005. http://dx.doi.org/10.1016/j.conengprac.2004.03.006.

[83] M. Matsuiski and T. Endo. Fatigue of metals subjected to varying stress. *Japan Soc. Mech. Engrg.*, 1969.

[84] T. Matsuzaka, N. Kodama, and N. Inomata. Control strategy of a wind generator to reduce power variations using probalistic optimal control, combining feed forward control from measured wind speed with feed back control. In *European Wind Energy Conference*, pages 1178–1181, 2001.

[85] T. Matsuzaka and K. Tuchiya. Power fluctuation stabilization of a wind generator by using feedforward control. In *European Wind Energy Conference*, pages 898–901, 1996.

[86] V. F. Montagner, R. C. L. F. Oliveira, and P. L. D. Peres. Design of $\mathcal{H}_\infty$ gain-scheduled controllers for linear time-varying systems by means of polynomial lyapunov functions. In *Conf. on Decision and Control*, pages 5839–5844, 2006. http://dx.doi.org/10.1109/CDC.2006.376707.

[87] I. Munteanu, N. A. Cutululis, A. I. Bratcu, and E. Ceanga. Optimization of variable speed wind power systems based on a LQG approach. *Control Engineering Practice*, 13(7):903–912, 2004. http://dx.doi.org/10.1016/j.conengprac.2004.10.013.

[88] N. Nanayakkara, M. Nakamura, and H. Hatazaki. Predictive control of wind turbines in small power systems at high turbulent wind speeds. *Control Engineering Practice*, 5(8):1063–1069, 1997. http://dx.doi.org/10.1016/S0967-0661(97)00097-X.

[89] Y. Nesterov and A. Nemirovskii. Interior-point polynomial algorithms in convex programming. *SIAM Review*, 36(4):682–683, 1994. http://dx.doi.org/10.1137/1036175.

[90] P. Novak, T. Ekelund, I. Jovik, and B. Schmidtbauer. Modeling and control of variable-speed wind-turbine drive-system dynamics. *Control Systems Magazine*, 15(4):28–38, 1995. http://dx.doi.org/10.1109/37.408463.

[91] NREL. Photographic information exchange, oct 2007. http://www.nrel.gov/data/pix/searchpix.html.

[92] K. Ohtsubo and H. Kajiwara. LPV technique for rotational speed control of wind turbines using measured wind speed. *Techno Ocean*, pages 1847–1853, 2004. http://dx.doi.org/10.1109/OCEANS.2004.1406423.

[93] Y. Oishi and H. Kimura. Randomized algorithms to solve parameter-dependent linear matrix inequalities and their computational complexity. In *Conf. on Decision and Control*, pages 2025–2030, 2001. http://dx.doi.org/10.1109/.2001.981208.

[94] A. Packard. Gain scheduling via linear fractional transformations. *Systems and Control Letters*, 1994. http://dx.doi.org/10.1016/0167-6911(94)90102-3.

[95] A. Packard and J. C. Doyle. The complex structured singular value. *Automatica*, 29(1):71–109, 1993. http://dx.doi.org/10.1016/0005-1098(93)90175-S.

[96] A. Packard and M. Kantner. Gain scheduling the LPV way. In *Conf. on Decision and Control*, pages 3938–3941, 1996. http://dx.doi.org/10.1109/CDC.1996.577296.

[97] K. Poolla and A. Tikku. Robust performance against time-varying structured pertubations. *Trans. on Automatic Control*, 40(9):1589–1602, 1995. http://dx.doi.org/10.1109/9.412628.

[98] V. M. Popov. Absolute stability of nonlinear systems of automatic control. *Automation and Remote Control*, 22:857–875, 1961.

[99] N. K. Poulsen, T. J. Larsen, and M. H. Hansen. Comparison between a PI and LQ-regulation for a 2 MW wind turbine. Technical report, Risø National Laboratory, 2005.

[100] A. Rantzer. On the kalman-yakubovich-popov lemma. *Systems and Control Letters*, 28:7–10, 1996. http://dx.doi.org/10.1016/0167-6911(95)00063-1.

[101] M. Rasila. Torque- and speed control of a pitch regulated wind turbine. Technical report, Chalmers University of Technology, 2003.

[102] R. Rocha and L. S. M. Filho. A multivariable $\mathcal{H}_\infty$ control for wind energy conversion system. In *Conf. on Control Applications*, pages 206–211, 2003.

[103] R. Rocha, L. S. M. Filho, and M. V. Bortolus. Optimal multivariable control for wind energy conversion system – a comparison between $\mathcal{H}_2$ and $\mathcal{H}_\infty$ controllers. In *Conf. on Decision and Control*, pages 7906–7911, 2005.

[104] W. J. Rugh and J. S. Shamma. Research on gain scheduling. *Automatica*, 36:1401–1425, 2000. http://dx.doi.org/10.1016/S0005-1098(00)00058-3.

[105] R. Sakamoto, T. Senjyu, T. Kinjo, N. Urasaki, T. Funabashi, H. Fujita, and H. Sekine. Output power leveling of wind turbine generator for all operating regions by pitch angle control. In *Power Engineering Society General Meeting*, pages 45–52, 2005. http://dx.doi.org/10.1109/PES.2005.1489543.

[106] D. Sbarbaro and R. Pena. A non-linear wind velocity observer for a small wind energy system. In *Conf. on Decision and Control*, pages 3086–3087, 2000.

[107] C. W. Scherer. From single-channel LMI analysis to multi-channel mixed lmi synthesis: a general procedure. *Selected Topics in IMC*, 8:1–8, 1995.

[108] C. W. Scherer. Robust generalized $\mathcal{H}_2$ control for uncertain and LPV systems with general scalings. In *Conf. on Decision and Control*, pages 3970–3975, 1996. http://dx.doi.org/10.1109/CDC.1996.577324.

[109] C. W. Scherer. *Robust Mixed Control and LPV Control with Full Block Scalings*. SIAM, 2000.

[110] C. W. Scherer. LPV control and full block multipliers. *Automatica*, 37:361–375, 2001. http://dx.doi.org/10.1016/S0005-1098(00)00176-X.

[111] C. W. Scherer, P. Gahinet, and M. Chilali. Multiobjective output-feedback control via LMI optimization. *Trans. on Automatic Control*, 42(7):896–911, 1997. http://dx.doi.org/10.1109/9.599969.

[112] J. S. Shamma. Robustness analysis for time-varying systems. In *Conf. on Decision and Control*, pages 3163–3168, 1992.

[113] J. S. Shamma and M. Athans. Analysis of gain scheduled control for nonlinear plants. *Trans. on Automatic Control*, 35(8):898–907, 1990. http://dx.doi.org/10.1109/9.58498.

[114] J. S. Shamma and M. Athans. Gain scheduling – potential hazards and possible remedies. *Control Systems Magazine*, 12(3):101–107, 1992. http://dx.doi.org/10.1109/37.165527.

[115] Y. D. Song, B. Dhinakaran, and X. Y. Bao. Variable speed control of wind turbines using nonlinear and adaptive algorithms. *Wind Engineering and Industrial Aerodynamics*, 85:293–308, 2000. http://dx.doi.org/10.1016/S0167-6105(99)00131-2.

[116] A. G. Sparks. Analysis of affinely parameter-varying systems using parameter dependent lyapunov functions. In *Conf. on Decision and Control*, pages 990–991, 1997. http://dx.doi.org/10.1109/CDC.1997.657573.

[117] K. A. Stol. *Dynamics Modeling and Periodic Control of Horizontal-Axis Wind Turbines*. PhD thesis, University of Colorado, 2001.

[118] K. A. Stol. Disturbance tracking and load control of wind turbines in variable-speed operation. In *ASME Wind Energy Symposium*, 2003.

[119] R. Tempo, E. W. Bai, and F. Dabbene. Probabilistic robustness analysis: Explicit bounds for theminimum number of samples. *Systems and Control Letters*, 30:237–242, 1997. http://dx.doi.org/10.1016/S0167-6911(97)00005-4.

[120] S. C. Thomsen. Nonlinear control of a wind turbine. Master's thesis, Technical University of Denmark, 2006.

[121] K. Trangbæk. *Linear Parameter Varying Control of Induction Motors*. PhD thesis, Aalborg University, 2001.

[122] D. Trudnowski and D. LeMieux. Independent pitch control using rotor position feedback for windshear and gravity fatigue reduction in a wind turbine. In *American Control Conference*, pages 4335–4340, 2002. http://dx.doi.org/10.1109/ACC.2002.1025328.

[123] E. L. van der Hooft and T. G. van Engelen. Feed forward control of estimated wind speed. Technical report, ECN, 2003.

[124] T. G. van Engelen. Control design based on aero-hydro-servo-elastic linear models from TURBO. In *Adaptive Systems for Signal Processing, Communications, and Control Symposium*, 2007.

[125] T. G. van Engelen, E. L. van der Hooft, and P. Schaak. Development of wind turbine control algorithms for industrial use. In *European Wind Energy Conference*, 2003.

[126] H. Vihriälä, P. Ridanpää, R. Perälä, and L. Söderlund. Control of a variable speed wind turbine with feedforward of aerodynamic torque. In *European Wind Energy Conference*, pages 881–884, 1999.

[127] John Vince and Sarah Keen. Windmills and how they work, oct 2007. http://www.garfnet.org.uk/new_mill/autumn96/jv_mill.htm.

[128] C. Wang and G. Weiss. Self-scheduled LPV control of a wind driven doubly-fed induction generator. In *Conf. on Decision and Control*, pages 1246–1251, 2006. http://dx.doi.org/10.1109/CDC.2006.377474.

[129] J. C. Willems. Dissipative dynamical systems, part 1: General theory. *Archive for Rational Mechanics and Analysis*, 45:321–351, 1972.

[130] J. C. Willems. Dissipative dynamical systems, part 2: Linear systems with quadratic supply rates. *Archive for Rational Mechanics and Analysis*, 45:351–393, 1972.

[131] A. D. Wright and M. J. Balas. Design of state-space-based control algorithms for wind turbine speed regulation. In *ASME Wind Energy Symposium*, 2002.

[132] A. D. Wright and M. J. Balas. Design of controls to attenuate loads in the controls advanced research turbine. In *ASME Wind Energy Symposium*, 2004. http://dx.doi.org/10.1115/1.1792654.

[133] F. Wu. A generalized LPV system analysis and control synthesis framework. *Int. J. of Control*, 74(7):745–759, 2001. http://dx.doi.org/10.1080/00207170010031495.

[134] F. Wu and K. Dong. Gain-scheduling control of LFT systems using parameter-dependent lyapunov functions. *Automatica*, 42:39–50, 2006. http://dx.doi.org/10.1016/j.automatica.2005.08.020.

[135] F. Wu and S. Prajna. SOS-based solution approach to polynomial LPV system analysis and synthesis problems. *Int. J. of Control*, 78(8):600–611, 2005. http://dx.doi.org/10.1080/00207170500114865.

[136] F. Wu, X. H. Yang, A. Packard, and G. Becker. Induced l2-norm control for LPV system with bounded parameter variation rates. In *American Control Conference*, pages 2379–2383, 1995.

[137] W. Xie. $\mathcal{H}_2$ gain scheduled state feedback for LPV system with new LMI formulation. In *Conf. on Control Theory and Applications*, pages 693–697, 2005. http://dx.doi.org/10.1049/ip-cta:20050052.

[138] Z. Xin-fang, X. Da-ping, and L. Yi-bing. Adaptive optimal fuzzy control for variable speed fixed pitch wind turbines. In *World Congress on Intelligent Control and Automation*, pages 2481–2485, 2004. http://dx.doi.org/10.1109/WCICA.2004.1342041.

[139] V. A. Yakubovich. Solution of certain matrix inequalities in the stability theory of nonlinear control systems. *Soviet Math. Dokl.*, 3:620–623, 1962.

[140] P. Yan and H. Özbay. On switching $\mathcal{H}_\infty$ controllers for a class of linear parameter varying systems. *Systems and Control Letters*, 56:504–511, 2007. http://dx.doi.org/10.1016/j.sysconle.2007.02.002.

[141] K. Zhou, P. P. Khargonekar, J. Stoustrup, and H. Niemann. Robust performance of systems with structured uncertainties in state space. *Automatica*, 31(2):249–255, 1995.

Paper A

Estimation of Effective Wind Speed

Contents

Conference paper presented at:
The 2nd Conference on The Science of Making Torque From Wind, August 2007.

Estimation of effective wind speed

K Z Østergaard[1], **P Brath**[1] **and J Stoustrup**[2]
[1] Turbine Control and Operation, Vestas Wind Systems
[2] Automation and Control, Dept. Electronic Systems, Aalborg University
E-mail: kazos@vestas.com

Abstract. The wind speed has a huge impact on the dynamic response of wind turbine. Because of this, many control algorithms use a measure of the wind speed to increase performance, e.g. by gain scheduling and feed forward.
Unfortunately, no accurate measurement of the effective wind speed is online available from direct measurements, which means that it must be estimated in order to make such control methods applicable in practice.
In this paper a new method is presented for the estimation of the effective wind speed. First, the rotor speed and aerodynamic torque are estimated by a combined state and input observer. These two variables combined with the measured pitch angle is then used to calculate the effective wind speed by an inversion of a static aerodynamic model.

1. Introduction

With the increasing competition in the wind energy market it is becoming very important to have control algorithms with which the structural fatigue is minimised without compromising the energy production. In contrast to many other control problems, the dynamics of wind turbines are driven by a disturbance, namely the wind speed. This means that the wind not only excites oscillations in various structural components but is also one of the main variables to select the operating condition of wind turbines – together with different control strategies like the rating of generator speed and power production.

One of the ways to handle the variations in operating conditions is the use of gain scheduling or adaptive control [1, 2, 3, 4, 5]. In these control methods, the controller variables are updated online on the basis of scheduling variables that are measured or constructed from measured variables. In these control methods it is very important that all operating conditions can be determined uniquely from the scheduling variables. This means that in the case of wind turbines, all variables determining the aerodynamics must be used, e.g. wind speed, rotor speed and pitch angle. In practice the number of variables is usually reduced by assuming a certain operating trajectory. Typically the pitch angle is chosen in full load operation and the generator speed is used in partial load operation as in [1, 6, 7]. Alternatively the controller is scheduled on wind speed [8, 9, 3], which has the advantage that the same scheduling variable can be used over the entire operating envelope. However, the wind speed is not directly available and must therefore be estimated. Further, if combined with pitch angle and generator speed it is also possible to schedule for operating conditions outside the nominal trajectory, which can be advantageous in the context of derating strategies, extreme weather conditions, fault situations, etc.

Another important reason for considering efficient estimation of the wind speed is the use of feed forward control. The variations in wind speed not only changes the dy-

namic response of the wind turbine but also the steady state values of important signals like shaft torque, tower thrust etc. To compensate for this the references for the controller is generated to give appropriate steady state values and to give a fast response, a nonlinear feed forward term is usually included from wind speed or aerodynamic torque to the relevant control signals [10, 11, 7, 12]

Both examples above indicate that there is a need for precise estimates of the wind speed in order to get a good performance in the overall control loop. It is assumed that the estimated wind speed will be used to schedule a controller with the main purpose of tracking generator speed references, tower thrust-wise movement, etc. In this context, the main drivers of the variations in set point and dynamics are the torque, Q_a, on the main shaft and the thrust, F_t, on the tower. These two variables can be described by static functions of rotor speed, ω_r, pitch position, β, and effective wind speed, v, as in (1) with the effective wind speed being defined as the spatial average of the wind field over the rotor plane with the wind stream being unaffected by the wind turbine, i.e. as if the wind turbines was not there [13].

$$Q_a = \frac{1}{2}\,\rho\,\pi\,R^2\,\frac{v^3}{\omega_r}\,c_P(\beta,\lambda) \tag{1a}$$

$$F_t = \frac{1}{2}\,\rho\,\pi\,R^2\,v^2\,c_T(\beta,\lambda) \tag{1b}$$

The constants, ρ, and R, describe respectively the air density and rotor radius and λ is the tip speed ratio defined as $\lambda = \frac{\omega_r\,R}{v}$. The purpose is then to estimate the effective wind speed, v, and it has been chosen to use (1a) together with a dynamic model of the drive train in the observer design.

In the literature many different algorithms have been investigated. The most simple algorithm assume that there is a static relation between electrical power production and the effective wind speed [8, 14, 15]. This assumption means that for example the energy stored in the speed-up of the rotor is neglected – which is a very crude assumption. In [15] it is concluded that using dynamic models significantly improves the observer performance, and it is therefore estimated that the use of static relations does not give satisfactory performance.

As a solution to the above mentioned issues, most papers in the literature propose a method which utilises a simple drive train model as in (2) with Q_g as the generator reaction torque and Q_{loss} being a loss term describing for example friction. [10, 16, 7, 11]

$$J\,\dot{\omega} = Q_a - Q_g - Q_{loss} \tag{2}$$

This model assumes that the drive train is infinitely stiff which means that drive train oscillations are neglected and that the lag between rotor acceleration and generator acceleration in case of gusts is also neglected. The first issue can most likely be handled by a notch filter at the drive train eigen-frequency, whereas the second issue will need further investigations in order to understand its significance. The observer algorithm is simply to calculate Q_a from measurement Q_g, differentiated measurement ω_g, and modelled loss term, Q_{loss}. In practice this method is very sensitive to measurement noise as indicated in [11]. It is therefore necessary to low pass filter either ω_g or the estimated output as in [16]. This approach imposes a very particular structure of the observer in order to reject measurement noise. It is well-known that a low pass filter will introduce a

time delay in the estimated quantity and the particular observer structure can potentially lead to a poor trade-off between noise rejection and time delay. It is therefore as in [11] estimated that a better performance can be achieved by using dynamic observers.

Dynamic observers for estimating the effective wind speed has not been as intensively investigated as the above mentioned methods. The main trend in the design of dynamic estimators has been either to design a linear Kalman filter for estimating the aerodynamic torque and then calculate the effective wind speed using (1a) [11]. The alternative is to combine a linear model of the drive train with the nonlinear aerodynamic model and use nonlinear algorithms to estimate the wind speed directly – either by online linearisation (extended Kalman filter) [15] or by using more dedicated algorithms [17].

The main advantage of all three algorithms is that they are all dynamic observers, which means that the filtering is designed via the cost function for the design algorithm. However, they all have the disadvantage that none of them are suited directly for input estimation – only state estimation. To counter this, a model of the input is created with the unknown input as a state variable. In the most simple form, Q_a, can be assumed to vary very slowly compared with the observer bandwidth. Then combined with the simple drive train model (2), the augmented model becomes

$$\begin{bmatrix} \dot{\omega} \\ \dot{Q}_a \end{bmatrix} = \begin{bmatrix} 0 & \frac{1}{J} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \omega \\ Q_a \end{bmatrix} + \begin{bmatrix} -\frac{1}{J} \\ 0 \end{bmatrix} Q_g + \begin{bmatrix} -\frac{1}{J} \\ 0 \end{bmatrix} Q_{loss} \tag{3}$$

The dynamic observer is then constructed by combining the augmented model with an update term, $L \cdot (\omega - \hat{\omega})$, that updates the state vector based on the estimation error in measured output. This means that the observer is of the form (4) with the accent $\hat{}$ denoting estimated variables

$$\begin{bmatrix} \dot{\hat{\omega}} \\ \dot{\hat{Q}}_a \end{bmatrix} = \begin{bmatrix} 0 & \frac{1}{J} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{\omega} \\ \hat{Q}_a \end{bmatrix} + \begin{bmatrix} -\frac{1}{J} \\ 0 \end{bmatrix} Q_g + \begin{bmatrix} -\frac{1}{J} \\ 0 \end{bmatrix} Q_{loss} + \begin{bmatrix} L_1 \\ L_2 \end{bmatrix} (\omega - \hat{\omega}) \tag{4}$$

It is clear that there are two major issues in this way of estimating the unknown input. First of all it has to be chosen, which model to use for the aerodynamic torque. In the example, the most simple form, $\dot{Q}_a = 0$ was chosen, but it can be extended to models that reflect the expected spectrum of Q_a. However, the difficulty in this part is to determine what spectrum to use, because wind turbines can encounter very different wind spectra depending on their respective location, e.g. plains, mountain areas, offshore, etc. A possible approach could be a self-tuning procedure, which would slowly identify location specific parameters and use these to adapt a wind spectrum model. This, however, would involve a comprehensive collection of representative data and is outside the scope of this paper.

The other issue is the trade-off between state estimation and input estimation. When there is an estimation error in the measured variable, $\omega - \hat{\omega}$, it must be identified how much this error shall affect the update of the state vector, $\hat{\omega}$, and how much the unknown input shall be updated. This is essentially the trade-off between the sizes of L_1 and L_2. If L_1 becomes too large compared to L_2, Q_a is not updated sufficiently leading to small estimation error in the state vector, but high estimation error in the unknown input, Q_a. On the other hand, if L_1 becomes too small compared to L_2, the state vector is not updated correctly. Then the estimation error, $\omega - \hat{\omega}$ will increase (not necessarily

to instability) and give poor estimates in both state vector and Q_a. Besides the balance between L_1 and L_2, their size must also be balanced between time propagation and time update. In the Kalman filter approach this is done in an optimisation function which minimises the mean square error of the state (and unknown input) estimate weighted by constant scalings.

To summarise: The simplified method of using steady state equations is very easy to design, but does not give sufficient estimation quality. The method that uses differentiation of generator speed is in its direct form also very simple to design, but amplifies to a large extend the measurement error and drive train oscillations. This problem can be handled by proper filtering which, however, introduces a time delay in the process and complexity in the design. Finally, the observer based estimator has the major advantage that filtering is included in the algorithm. The disadvantage is on the other hand that the complexity in the algorithm is increased – especially regarding the choice of model for the unknown input and the weighting between state estimation and input estimation.

This paper presents a method that is quite similar to the observer based method presented above. The major difference is that instead of augmenting the state model as in (3), the state and input estimation problem is split into two separate problems. A dynamic observer based on the Kalman filtering approach is designed for the state estimation and an input observer based on ideas from tracking controllers is designed for estimation of the aerodynamic torque. In this setting it is expected that the tracking performance will be significantly better than the steady state estimation method. Further it is expected that the trade-off between noise rejection and time delay is improved when compared with the method using differentiation of measured generator speed. Finally it is expected to have similar performance to other dynamic observers in the literature. However by splitting the observer problem into a state estimator and an input estimator, the design problem is simplified as it will be illustrated and the choice of wind model is transformed into the choice of observer structure for the input estimator – from experience in tracking controllers, this problem is efficiently solved by a PID structure.

In Section 2 we present a method for the design of an observer to estimate the angular velocity of the rotor and the aerodynamic torque acting on the low speed shaft. Then in Section 3, these two variables together with measured pitch position are used to calculate the effective wind speed by inversion of the aerodynamic model. Finally in Section 4 the conclusions are given.

2. Estimation of rotor speed and aerodynamic torque

In this section we take advantage of methods from the field of state estimation and combine them with ideas from tracking controllers to obtain what is known as disturbance estimators [18]. In the following it is assumed that the drive train to a sufficient level of accuracy can be described by two inertias interconnected by a spring and damper and with viscous friction on each inertia. The external forces to this 2-DOF system is then the aerodynamic torque, Q_a, on the slow speed shaft and generator reaction torque, Q_g, on the high speed shaft. This results in the system of equations in (5) which for simplicity in the notation will be referred to via the general state space form (6) with $x = \begin{bmatrix} \omega_r & \omega_g & \theta_\Delta \end{bmatrix}^T$ as the state vector.

$$
\begin{aligned}
J_r\,\dot{\omega}_r &= Q_a - B_r\,\omega_r - \mu\,(\omega_r - \omega_g) - K\,\theta_\Delta && \text{(5a)}\\
J_g\,\dot{\omega}_g &= -Q_g - B_g\,\omega_g + \mu\,(\omega_r - \omega_g) + K\,\theta_\Delta && \text{(5b)}\\
\dot{\theta}_\Delta &= \omega_r - \omega_g && \text{(5c)}
\end{aligned}
$$

$$
\begin{aligned}
\dot{x} &= A\,x + B_a\,Q_a + B_g\,Q_g && \text{(6a)}\\
\omega_r &= C_r\,x && \text{(6b)}\\
\omega_g &= C_g\,x && \text{(6c)}
\end{aligned}
$$

2.1. Dynamic observer design.
For the observer design we assume that the generator speed, ω_g, and generator torque, Q_g, is available through measurements. Let us for a moment assume also that the aerodynamic torque is available through measurements. Then we are left with only the state estimation problem. The state estimator is designed by propagating the input signals, Q_a and Q_g, through (5). The state vector is furthermore updated by a scaling, L, of the error in estimated output as described in (7)

$$
\begin{aligned}
\dot{\hat{x}} &= A\,\hat{x} + B_a\,Q_a + B_g\,Q_g + L\,(\omega_g - \hat{\omega}_g) && \text{(7a)}\\
\begin{bmatrix} \hat{\omega}_r \\ \hat{\omega}_g \end{bmatrix} &= \begin{bmatrix} C_r \\ C_g \end{bmatrix}\hat{x} && \text{(7b)}
\end{aligned}
$$

The observer gain, L, can be designed using a number of different methods. It has been chosen to use the Kalman filtering approach, which is a method that minimises the expected value of the square of the estimation error: $E[(x-\hat{x})^2]$.

In practice, the aerodynamic torque, Q_a, is not measurable, which means that we need to extend the observer described by (7) with a term to estimate Q_a. In the literature this issue is handled by augmenting the dynamic model by a model of the unknown input to estimate as in (3) – typically with the 2-DOF drive train model instead of the 1-DOF model in the example.

In contrast to the methods in the literature, it has been chosen to split the observer design into two observers operating in a cascaded coupled setup. The inner part is a Kalman filter designed along standard lines on the basis of (7), i.e. under the assumption that Q_a is available. The outer loop is then setup as a tracking configuration with ω_g as the tracking variable and $\hat{Q}_a$ as the "control signal". The "controller" has been chosen to be of the PI structure in order to have an integral term taking care of the asymptotic tracking and a direct gain handling the faster variations. The complete observer structure is then as shown in Figure 1.

This approach has some resemblance to the observer with the augmented wind model because the PI controller can be considered a known wind model with which the model is augmented. The integral term corresponds in this context to the typical wind model, $\dot{Q}_a = 0$, but with the proportional term a quicker response is ensured. In this context it should be noted that by increasing the proportional gain it corresponds to an increase in L in the input direction. This corresponds to increasing the bandwidth of the outer loop to something close to the bandwidth of the inner loop which can potentially lead

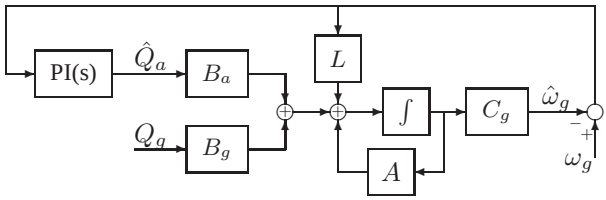


Figure 1: Block diagram of observer structure.

to instability. Because of this issue, the stability of the interconnection must always be checked a posteriori to the observer design.

The major advantage of this method for observer design is that the split into two interconnected observers lead to a two-step design method. First a state estimator is designed to have a sufficient bandwidth and noise rejection as if the aerodynamic torque is available. Afterwards the input observer can be designed by investigating different structures – with the constraint that there is a sufficient gap in bandwidth between the inner and outer loop.

In the case of estimation of the aerodynamic torque, a PI controller is a natural choice for estimation of the effective wind speed. Process knowledge can also be used to improve the estimation process. It is well-known that the generator speed might suffer from oscillations at the drive train eigen-frequency without having noticeable oscillations in the aerodynamic torque. Because of this it might be advantageous to filter the signal to the PI observer at the drive train eigen-frequency without filtering the signal for the state estimator. In this way the state estimator still estimates the rotor speed correctly, and the noise on the estimate of aerodynamic torque is reduced. Also gain-scheduling of the PI observer can be introduced in order to take into account that there might be different requirements to bandwidth in high wind speeds as opposed to low wind speeds.

2.2. Simulation results

The performance of the observer designed in Section 2.1 is tested against an estimator designed on the basis of solving the differential equation (2) by differentiating the generator speed measurement. In this estimator, the loss term is described by viscous friction, i.e. $Q_{loss} = B_r \cdot \omega$. Further the rotor rotational speed, ω is assumed equivalent to the generator speed with the drive train eigen-frequency filtered out. The differentiation will amplify the measurement noise and a first order low pass filter will be used to smoothen out the estimation. This leads to the estimator structure in Figure 2 with T as the tuning parameter for the trade-off between time delay in the estimation and noise rejection.

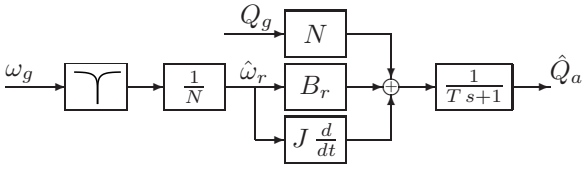


Figure 2: Block diagram of differentiation based estimator.

For the verification, both estimators have been simulated with identical controller parameters and wind conditions (according to IEC 1A), and in Figure 3 a comparison of the performance of the two estimators is given. It can be observed that the estimation of aerodynamic torque is improved slightly and when comparing standard deviations it can be concluded that there is an improvement of approximately 18% (standard deviation is respectively 72 kNm and 88 kNm for the two algorithms). For the case of estimation of rotor speed, the dynamic observer shows significantly improved performance.

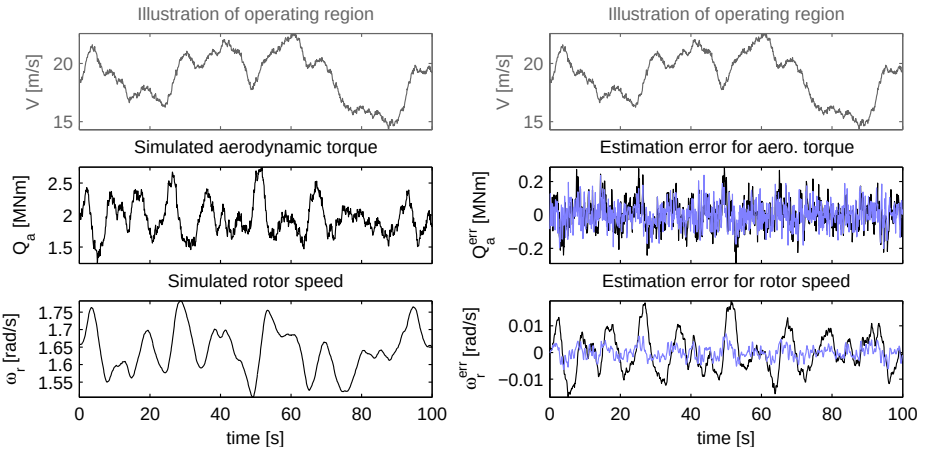


Figure 3: Left: Simulation results of variables to estimate. Right: estimation error for selected signals (Light blue: observer based estimate, and black: differentiation based estimate).

If we zoom in on a time interval with a large change in aerodynamic torque as in Figure 4, it can be observed that the differentiation based method suffers from a larger time delay in the estimation which is caused by the low pass filtering of the estimate. To counter this, the time constant in the filter can be decreased which has the side-effect that the high frequency noise will be increased and result in an even worse overall performance.

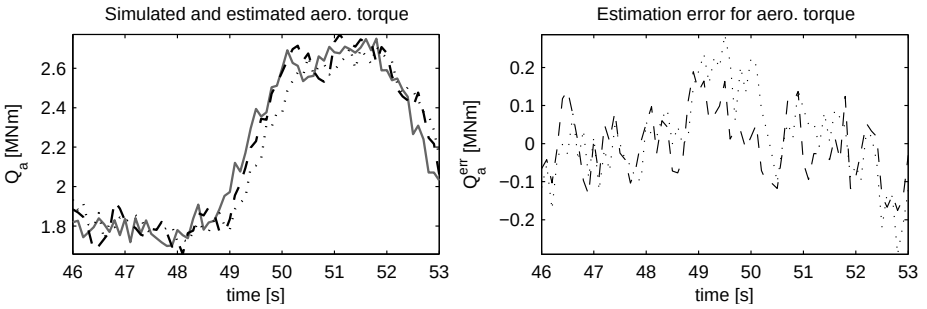


Figure 4: Detailed simulation of estimation of aerodynamic torque. Solid lines: simulation values, dashed lines: observer based estimate, and dotted: differentiation based estimate.

3. Calculation of wind speed

In the previous section a dynamic observer was presented for estimating the rotor speed and aerodynamic torque. These two variables together with measured pitch angle will be used in this section to calculate the effective wind speed by using (1a). First (1a) is rewritten as in (8) and under the assumption that the air density is known, all variables on the left hand side of (8b) will be online available. The right hand side will be a function of λ alone, because β is online available.

$$Q_a \omega_r = \frac{1}{2} \rho \pi R^2 \frac{R^3 \omega_r^3}{\lambda^3} c_P(\beta, \lambda) \quad \Leftrightarrow \tag{8a}$$

$$\frac{2\, Q_a}{\rho \pi R^5 \omega_r^2} = \frac{c_P(\beta, \lambda)}{\lambda^3} \tag{8b}$$

In the following, $c_P(\beta, \lambda)$, for a particular choice of β will be denoted $c_{P,\beta}(\lambda)$. The effective wind speed is then calculated by first solving (8b) for λ and then calculating the effective wind speed as $v = \frac{\omega_r R}{\lambda}$.

In order to be able to solve (8b) for λ we first need to understand the monotonicity properties of $\lambda^{-3} \cdot c_{P,\beta}(\lambda)$. λ^{-3} is clearly a monotonously decreasing function, but $c_{P,\beta}(\lambda)$ is concave which means that two different tip speed ratio will lead to the same power coefficient, c_P: one for the stall region and one for the pitch region. When multiplying these two factors the result is monotonous for some values of β, whereas it is non-monotonous for other values – determined by the region where the positive slope of $c_{P,\beta}(\lambda)$ is steeper than the negative slope of λ^{-3}. This issue is illustrated in Figure 5 from which it can be seen that the function is invertible for large pitch angles whereas it is not invertible for small pitch angles.

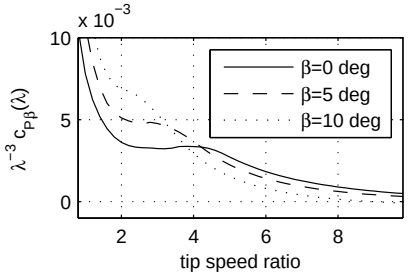


Figure 5: Illustration of non-monotonicity of $\lambda^{-3} \cdot c_{P,\beta}(\lambda)$.

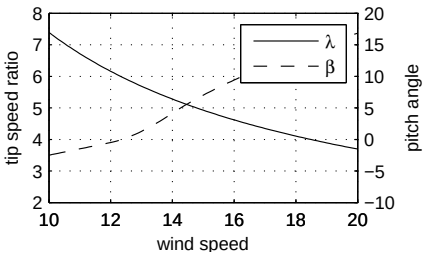


Figure 6: Nominal operating range.

Because the right hand side of (8b) is not invertible for specific choices of the pitch angle, knowledge about operation of the wind turbine is used to calculate the most likely λ that would solve the equation. From the above discussion, the monotonicity can only occur in the stall region because the slope of $c_{P,\beta}(\lambda)$ must be positive. This means that the issue is unlikely to occur during nominal operation because the algorithm is designed for pitch controlled wind turbines not operating in stalled operation. Gusts, or fault situations might on the other hand lead to short periods of time operating in the stalled region. In this case the largest tip speed ratio satisfying the equation is used, which is based on the assumption that it is more likely that the wind turbine is operating in slight stall than in deep stall.

If we investigate a bit further in which operating conditions the problem of monotonicity might happen, a plot of the nominal tip speed ratio and pitch angle is given in Figure 6. From the illustration it can be seen that the region 10-14 m/s is the most critical operating region for having numerically stable inversion of (8b), because both tip speed ratio and pitch angle are relatively small. Below 10 m/s the issue is still relevant, but the nominal tip speed ratio will be larger relative to the local maximum of $\lambda^{-3} \cdot c_{P,\beta}(\lambda)$, and in order to reduce λ to a critical size the gust must therefore be large. For larger mean wind speeds the nominal pitch angle will be larger which means that the function will be monotonously decreasing during nominal operation.

For the signals presented in Figure 3, the procedure described above has been applied to calculate the wind speed estimate shown in Figure 7. From the plot on the right hand side, it can be observed that wind speed estimate is slightly improved by using the dynamic observer as base when comparing with the differentiation based method. And when comparing the standard deviation of the estimation errors it can be seen that the estimate is improved by approximately 15% (standard deviation is respectively $0.20\ m/s$ and $0.23\ m/s$ for the two methods). This improvement in standard deviation between the two methods is similar to that of the estimation of Q_a, which indicates that the significantly improved estimation of ω_r does not increase the performance much in terms of estimation of the effective wind speed.

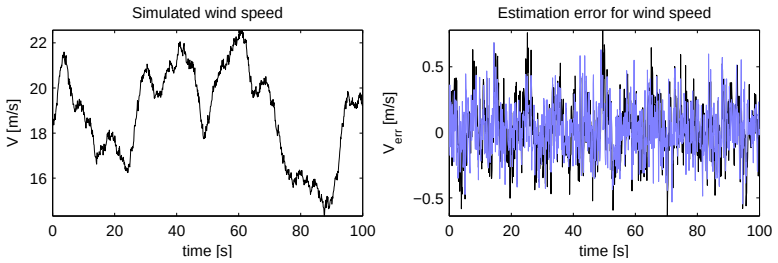


Figure 7: Left: simulated wind speed. Right: estimation error (Light blue: observer based estimate, and black: differentiation based estimate).

4. Conclusions

This paper has presented a method for estimation of the effective wind speed. The observer consists of two components: A state and input observer for the estimation of rotor speed and aerodynamic torque and a calculation of the effective wind speed by inversion of the monotonous part of a static model of the aerodynamics.

The state and input observer showed a significant improvement in performance, when comparing with methods that solve the estimation problem by solving the differential equation using differentiation.

The calculation of the effective wind speed has shown to be numerically stable during nominal operation. Further investigations are necessary, though to understand how the algorithm will perform in the case of large and fast increases in the mean wind speed – especially in the region around rated generator speed.

It is expected from this improvement in quality of the estimate of effective wind speed that control algorithms that at present time use this variable will benefit from using

this algorithm. Those algorithms will typically be controller that are gain-scheduled on wind speed or which use wind speed in a feed forward setting.

In order to achieve an even higher precision in the estimation it might be required to design the observer as one single component, because it is not clear how the estimation error in Q_a and ω_r transforms into estimation error in effective wind speed. In that case it is necessary to take the aerodynamic model into account, which means that the presented method needs to be extended to nonlinear methods, e.g. by using the unscented Kalman filter for the state estimation. Also the PI observer might need to be modified to a nonlinear observer. This will make the design problem harder in practice but will potentially give a better performance.

BIBLIOGRAPHY

[1] D.J. Leith and William E. Leithead. Appropriate realization of gain-scheduled controllers with application to wind turbine regulation. *Int. Journal of Control*, 65(2):223–248, 1996.

[2] Xin Ma. *Adaptive Extremum Control and Wind Turbine Control*. PhD thesis, Technical University of Denmark, 1997.

[3] F.D. Bianchi, R.J. Mantz, and C.F. Christiansen. Control of variable-speed wind turbines by LPV gains scheduling. *Wind Energy*, 7:1–8, 2004.

[4] Niels K. Poulsen, Torben Juul Larsen, and Morten H. Hansen. Comparison between a PI and LQ-regulation for a 2 MW wind turbine. Technical report, Risø National Laboratory, 2005.

[5] Nicolas Antonio Cutululis, Emil Ceanga, Anca Daniela Hansen, and Poul Sørensen. Robust multi-model control of an autonomous wind power system. *Wind Energy*, 9:399–419, 2006.

[6] Fabien Lescher, Jing Yun Zhao, and Pierre Borne. Robust gain scheduling controller for pitch regulated variable speed wind turbine. *Studies in Informatics and Control*, 14(4):299–315, 2005.

[7] T.G. Engelen, E. L. van der Hooft, and P. Schaak. Development of wind turbine control algorithms for industrial use. In *European Wind Energy Conference*, 2003.

[8] Shibashis Bhowmik and René Spée. Wind speed estimation based on variable speed wind power generation. In *Conf. of Industrial Electronics Society*, pages 596–601, 1998.

[9] Kazuhisa Ohtsubo and Hiroyuki Kajiwara. Lpv technique for rotational speed control of wind turbines using measured wind speed. *Techno Ocean*, pages 1847–1853, 2004.

[10] E. L. van der Hooft and T.G. Engelen. Estimated wind speed feed forward control for wind turbine operation optimisation. In *European Wind Energy Conference*, 2004.

[11] Harri Vihriälä, P. Ridanpää, R. Perälä, and L. Söderlund. Control of a variable speed wind turbine with feedforward of aerodynamic torque. In *European Wind Energy Conference*, pages 881–884, 1999.

[12] Tomoyuki Matsuzaka and Keiichi Tuchiya. Power fluctuation stabilization of a wind generator by using feedforward control. In *European wind energy conference (EWEC)*, pages 898–901, 1996.

[13] Tony Burton, David Sharpe, Nick Jenkins, and Ervin A. Bossanyi. *Wind Energy Handbook*. Wiley, 2001.

[14] Torbjörn Thiringer and Andreas Petersson. Control of a variable-speed pitch-regulated wind turbine. Technical report, Chalmers University of Technology, 2005.

[15] Xin Ma, Niels K. Poulsen, and H. Bindner. Estimation of wind speed in connection to a wind turbine. Technical report, Technical University of Denmark, 1995.

[16] Naruhito Kodama, Tomoyuki Matsuzaka, and Noboru Inomata. Power variation control of a wind turbine generator using probalistic optimal control, including feed-forward control from wind speed. *Wind Engineering*, 24(1):13–23, 2000.

[17] Daniel Sbarbaro and Ruben Pena. A non-linear wind velocity observer for a small wind energy system. In *Conf. on Decision and Control*, pages 3086–3087, 2000.

[18] Jung-Han Kim and Jun-Ho Oh. Disturbance estimation using sliding mode for discrete kalman filter. In *Conf. on Decision and Control*, pages 1918–1919, 1998.

Paper B

Gain-scheduled Linear Quadratic Control of Wind Turbines Operating at High Wind Speed

Contents

Conference paper presented at:
The 16th Conference on Control Applications, October 2007.
As part of Multi-conference on Systems and Control.

Gain-scheduled Linear Quadratic Control of Wind Turbines Operating at High Wind Speed

Kasper Zinck Østergaard
Vestas Wind Systems
Loads, Aerodynamics and Control
kazos@vestas.com

Per Brath
Vestas Wind Systems
Loads, Aerodynamics and Control

Jakob Stoustrup
Aalborg University
Dept. of Electronic Systems
Automation and Control

Abstract— This paper addresses state estimation and linear quadratic (LQ) control of variable speed variable pitch wind turbines. On the basis of a nonlinear model of a wind turbine, a set of operating conditions is identified and a LQ controller is designed for each operating point. The controller gains are then interpolated linearly to get a control law for the entire operating envelope.

The states and the gain-scheduling variable are not online available and an observer is designed. This is done in a modular approach in which a linear estimator is used to estimate the non-measured state variables and the unknown input, aerodynamic torque. From the estimated aerodynamic torque and rotor speed and measured pitch angle the scheduling variable effective wind speed) is calculated by inverting the aerodynamic model.

Simulation results are given that display good performance of the observers and comparisons with a controller designed by classical methods display the potential of the method.

I. INTRODUCTION

Within the past decade the cost-effectiveness of wind turbines has increased significantly. This is primarily achieved by reducing the amount of material for a fixed wind turbine size. This reduction of mass in the structural components causes each component to be less robust towards fatigue loads. The problem can be countered by the introduction of active control, which was done quite some years ago.

In this paper we consider wind turbines for which it is possible to rotate the blades along the longitudinal axis (denoted pitch) and hereby controlling the energy input. Furthermore we consider a wind turbine with a doubly-fed induction generator with which it is possible to control the reaction torque from the generator to make the rotational speed vary approximately $\pm 30\%$ from the synchronous speed. This way we can better control loads in the transmission system and also obtain a higher energy output at low wind speeds. In the following we will denote such a wind turbine: a variable speed, variable pitch wind turbine.

This introduction of two control variables (pitch angle and generator torque) has led to many investigations in the design of control algorithms that give the best trade-off between variations in the power and fatigue loads. Also the LQ control technique has been applied to the control of wind turbines. [2, 6, 8, 13]

These publications address the design of a static state feedback controller for a linearised plant model at a selected wind speed, and most of them also address the problem of estimating the states using Kalman filters or similar. Also the problem of interpolating the controllers has been addressed in [13] in which the gain scheduling variable is estimated from steady state equations.

The papers in the literature on LQG control of wind turbines are in general split into three different categories: Some present a detailed controller design for a model

linearised at a single operating point. The two other categories focus on the nonlinear model but take very different approaches: either the academic approach with focus on modern control techniques or the practical approach which focuses on getting simple algorithms working in practise. This means that focus is in most papers either on the application of modern control techniques with the relation to physical requirement missing to some degree. The opposite approach with the large focus on physical interpretation often misses the possible advantages of modern control techniques.

This paper is an attempt to close the gap between the papers focusing on modern control techniques and the focus on requirements that are met in todays operation of wind turbines. A gain-scheduled LQ controller is designed with performance weight similar to that of [8] – with the addition of a weight on the shaft torque to limit fatigue loads in the transmission system. For the state estimation it should be noted that the disturbance (wind speed) has a large impact on the wind turbine dynamics and that this disturbance is not measured. Because of this it has been chosen to use the principles of disturbance estimation [4] in the observer design. Furthermore the wind speed will be used as the gain scheduling variable and must therefore be online available.

II. WIND TURBINE MODEL

In this control formulation we are interested in maintaining the generator speed within its limits, minimising the power fluctuations around its nominal value, and keeping the fatigue loads in the transmission below a certain level. The controller design will be based on a two degree of freedom model of the transmission, a static model of the aerodynamics and actuator dynamics.

A. Aerodynamics

The main input to the wind turbine is the wind, which through the aerodynamic lift and drag effects the main shaft by a driving torque. The angle of attack of the wind onto the blades can be assumed dependent on only the pitch orientation of the blades and the ratio between the speed of the blade tip and the wind speed (denoted tip-speed-ratio). We assume for simplicity that all blades are pitched to the same orientation and that the lift/drag on the blades directly affect the driving torque, Q_a, on the main shaft – i.e. the structural dynamics of the blades are incorporated into the parameters of the model of the transmission system. With this assumption we can describe the aerodynamics as a static nonlinear mapping of the collective pitch orientation, β, the angular velocity of the rotor, ω_r, and the effective wind speed*, v, as shown in (1) where ρ is the air density and R is the rotor radius. The function c_P describes the aerodynamic efficiency of the rotor design and is described by a nonlinear mapping of the pitch angle and tip speed ratio, λ, as illustrated in Fig. 1

$$Q_a = \frac{1}{2}\,\rho\,\pi\,R^2\,\frac{v^3}{\omega_r}\,c_P(\beta,\lambda)\;, \qquad \lambda = \frac{R\,\omega_r}{v} \tag{1}$$

B. Transmission system

The aerodynamic torque, Q_a, from (1) is input to the transmission system on the low speed side. The transmission system is modelled as two inertias interconnected by a

*The effective wind speed (also denoted the free wind speed) is an abstract term that describes the spatial average of the wind field at the rotor position with the wind stream not being affected by the wind turbine.

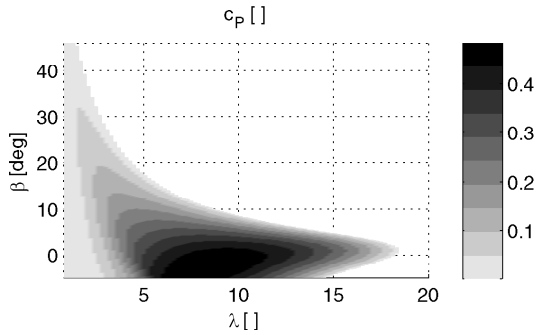


Fig. 1: Illustration of aerodynamic coefficient, c_P.

spring/damper and a gearing. The rotor inertia and stiffness is included in the parameters of the low speed side. On the high speed side the generator is mounted giving opportunity to control the reaction torque from the generator. Friction, stiffness, etc. is assumed linear leading to a linear model of the transmission system with aerodynamic torque, Q_a, and generator torque, Q_g, as inputs. The outputs are angular speed on the low speed side, ω_r, on the high speed side, ω_g, and the torsion between the two inertias, Q_{sh}. From these observations we can setup a dynamic model of the transmission system of the form (2) with x_t being the state vector.

$$\dot{x}_t = A_t \cdot x_t + B_{t,r} \cdot Q_a + B_{t,g} \cdot Q_g \tag{2a}$$
$$Q_{sh} = C_{t,Q} \cdot x_t \tag{2b}$$
$$\omega_r = C_{t,r} \cdot x_t \tag{2c}$$
$$\omega_g = C_{t,g} \cdot x_t \tag{2d}$$

C. *Pitch system*

To deal with the low frequency variations in wind speed we can alter the pitch orientation of the blades causing the aerodynamic torque to be manipulated. The actuator is highly nonlinear and a cascade coupled solution has been chosen to handle the nonlinearity.

In Fig. 2 the loop containing both the model of the pitch actuator and its associated controller is shown. The actuator is a hydraulic actuator with the transfer function from control voltage to pitch rate being modelled as a combination of a static nonlinear gain, $G(u)$, a time delay and a low pass filter. To counteract the nonlinear gain, $G(u)$, a gain-scheduled proportional controller, $K(e)$, has been designed to track a pitch reference, β_{ref}.

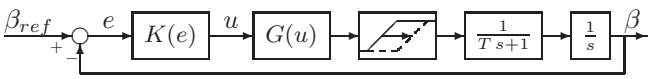


Fig. 2: Inner pitch control loop

From the perspective of the outer loop, the proportional controller, $K(e)$, has a linearising effect on the nonlinear pitch gain when the pitch error, e, is in the region of nominal operation. For larger errors the nonlinearity has still quite some effect which

means that we cannot use it in extreme operating conditions that lead to extraordinary pitch activity. This can happen in extreme weather conditions or in the case of faults in the wind turbine leading to very high pitch activity. However, for nominal operation we conclude that the linear model is appropriate.

D. Generator and converter system

On the high speed shaft we can change the reaction torque from the generator by changing the ratio between power in the rotor and stator. This control action enables the possibility to make a trade-off between the variations in active power production and variations in rotational speed.

The generator and converter dynamics is modelled as a constant gain because it only contains very high frequency components. Then the active power, P_e, can be expressed as the sum of the power in the rotor, P_r and stator, P_s. Furthermore the power in the rotor is proportional to the stator power and the slip, $s = \frac{1}{\omega_{NET}}(\omega_g \cdot pp - \omega_{NET})$ – with pp being the number of pole pairs in the generator. This means that the electric power can be expressed as in (3).

$$\begin{aligned} P_e(t) &= P_s(t) + P_r(t) = P_s(t) \cdot (1 + s(t)) \\ &= P_s(t) \cdot \left(1 + \frac{\omega_g(t) \cdot pp - \omega_{NET}}{\omega_{NET}}\right) \\ &= \frac{pp}{\omega_{NET}} \cdot P_s(t) \cdot \omega_g(t) \end{aligned} \tag{3}$$

The loss in the generator and converter is assumed proportional to the active power and independent of the operating condition. This means that the reaction torque from the generator, Q_g, can be expressed as in (4) with η being the generator/converter efficiency.

$$Q_g(t) = \frac{P_e(t)}{\eta \cdot \omega_g(t)} = \frac{pp}{\eta \cdot \omega_{NET}} \cdot P_s(t) \tag{4}$$

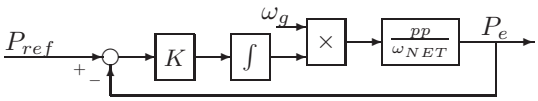
The main objective of the controller design for the generator loop is to ensure a proper power quality and to produce the desired power level. The controller includes high frequency components as well as integral action on the tracking of the power reference. The high frequency components can be disregarded when seen from the outer loop which leads to the loop illustrated in Fig. 3a with K being a linear control gain. When designing the outer loop, it is more relevant to have the formulation in terms of the reaction torque which can be achieved by using (4). Then we get a generator closed loop model as in Fig. 3b.

E. Interconnection

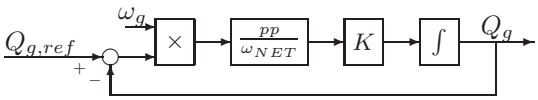
To summarise, the model of the wind turbine consists of four components: A static nonlinear function describing the aerodynamics, a third order LTI model of the transmission system, and a gain-scheduled first order model of the two actuators. The interconnection of these components is illustrated in Fig. 4.

III. CONTROLLER DESIGN

The controller design has two main objectives. First of all it must keep the structural/electrical loads within the design specifications. The structural loads are in this formulation measured by the shaft torsion, Q_{sh}, i.e. minimising Q_{sh} will minimise the



(a) Block diagram for control of power fluctuations.



(b) Block diagram for load reduction.

Fig. 3: Block diagram of generator loop.

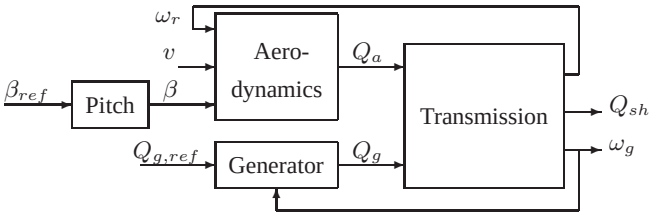


Fig. 4: Block diagram of wind turbine model.

structural loads in the transmission system. The electrical loads are mainly given by the slip in the generator, and by limiting the variations from a statically calculated generator speed reference the electrical loads in the generator/converter will be limited. The other main objective is the power quality and with the electrical power being proportional to the generator reaction torque, Q_g. The power quality is in this context measured in the variations in reaction torque from a statically given set-point. From these objectives the control formulation is given as a tracking problem of generator speed and torque references and the minimisation of the shaft torsion.

The model described in Section II is highly nonlinear, mostly because of the coupling through the aerodynamics. Also the actuator loops are nonlinear, but in the high wind speed region a linearised model of these loops is deemed appropriate.

It has been chosen to design the controller as a gain-scheduled static state feedback with the effective wind speed as the gain-scheduling variable. Along a selected trajectory of operating conditions the nonlinear model is linearised and an LQ controller is designed to trade off the three objectives described above. The trajectory of operating conditions is determined from the following observations.

A. Target trajectory

In the high wind speed region the generator speed must be maintained close to a specific rated value in order to keep the electrical loads low. Furthermore the power production must be close to the rated power production in order to maximise the production. The rated rotor speed and aerodynamic torque can then be calculated from the DC response of the linear transmission system model combined with the rated values for the generator speed and generator torque – which is easily calculated from the speed, power and generator efficiency. Then there are only two variables left in (1) and with the assumption that the wind turbine is not operating in the stall region, there is a one-to-one mapping from mean wind speed to mean pitch angle as illustrated in Fig. 5 – in order to obtain

rated speed and power.

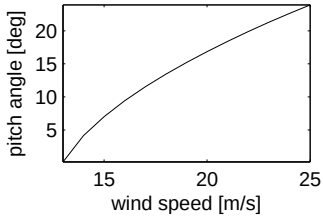


Fig. 5: Steady state pitch angles at rated rotor speed and power.

B. LQ design

With the operating points determined, a set of controllers can now be determined using LQ control. In order to have good DC tracking performance for both tracking problems an integrator on each tracking error is included in the setup. This then gives the control setup in Fig. 6 and the controller gain, K, is calculated at each operating point to minimise the cost J_c, in (5). In the implementation the intermediate controller gains are then interpolated linearly from the discrete number of controller gains.

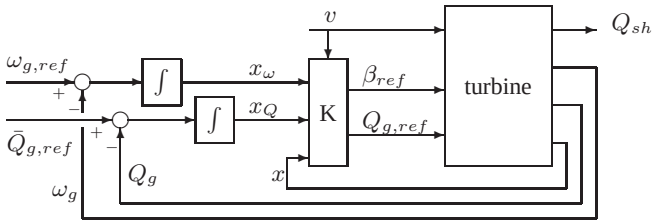


Fig. 6: Block diagram of controller formulation.

$$
\begin{aligned}
J_c &= \int_{t=0}^{\infty} z(t)^T\, Q\, z(t) + u(t)^T\, R\, u(t)\, dt \qquad (5)\\
z &= \begin{bmatrix} Q_{sh} & x_\omega & x_Q \end{bmatrix}^T\\
u &= \begin{bmatrix} \beta_{ref} & Q_{g,ref} \end{bmatrix}^T
\end{aligned}
$$

Remark 7. It should be noted that with the approach of interpolating controller gains, we can give no guarantees in terms of stability and performance for the intermediate operating conditions. In practise, however, the method has shown applicability for several application areas.

One way to overcome the problem is to apply LPV techniques for the gain scheduling. In these methods, the model is scheduled upon a time-varying parameter for which the values are a-priori unknown but measurable online. See e.g. [11, 7, 9, 5, 10, 1]. ♢

IV. OBSERVER DESIGN

There is no high quality measurement of the rotor speed and only a very rough measurement of the wind speed available. It is therefore necessary to have a good state estimator to get the controllers implementable. Further we need an estimate of the wind speed to get the scheduling variable. A detailed presentation of the observer design is given in [12] and we give only an overview of the parts that directly apply to the current controller design. Preliminary practical experiments indicate that the measurements of the actuator outputs are given with a precision that allows us to focus only on the estimation of the dynamics in the transmission system. This means that we assume that the pitch angle, generator speed and generator torque are online available. We split the observer design into two main components: a dynamic observer for the state and input estimation of the transmission system, and a calculation of the wind speed from the estimated aerodynamic torque.

A. *Observer for transmission system*

From linear analysis it is clear that the transmission model is observable from the measured output, ω_g. Also one of the inputs, Q_g, is available online – and for simplicity in the algorithm it has been chosen to separate the observer design for the transmission system from the nonlinear aerodynamic model. This means that we must also estimate the aerodynamic input, Q_a. One approach is to augment the transmission model with a state representing the unknown input and use a Kalman filter to estimate the augmented state vector. Alternatively it has been chosen to use a method where the observer design is split into a standard state estimation problem combined with an input observer. The main idea is illustrated in Fig. 7: A Kalman gain, L, is designed as if the unknown input was available with process noise reflecting the expected variance in the estimation of Q_a. Then an observer is designed in parallel with the observer gain, L, to estimate the "disturbance", Q_a.

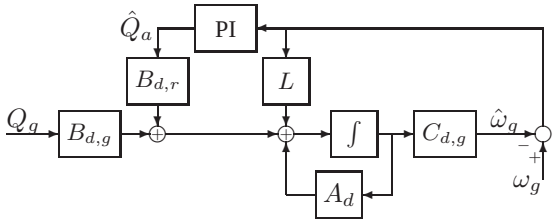


Fig. 7: Block diagram of observer for transmission system.

It has in this case been chosen to use the PI controller in the disturbance estimation for its simplicity in the tuning process and because it achieves asymptotic tracking. The method can though easily be extended to other controller structures.

B. *Calculation of wind speed*

In the above sections, a set of observers were designed to estimate the state vector in the wind turbine model. In this section the gain-scheduling variable, v, will be calculated from the output of these observers. Then (1) can be rewritten as (6) with C_ω being a

constant for each rotor speed and $c_{P,\beta}$ being only a function of λ at a given pitch angle.

$$Q_a = \frac{1}{2}\,\rho\,\pi\,R^2\,\frac{v^3}{\omega_r}\,c_P(\beta,\lambda) = C_\omega \cdot \frac{c_{P,\beta}(\lambda)}{\lambda^3} \tag{6}$$

When the wind turbine is not operating in the stall region, $c_{P,\beta}$ is a decreasing function and thus, Q_a is a decreasing function in λ and therefore invertible. If, however, the wind turbine is in stall operation, $c_{P,\beta}$ is increasing and Q_a is not monotonous and thereby not invertible. From the physical interpretation it can be observed that the phenomenon happens only during stall operation[12], and because of this it is deemed that we will not encounter this problem in nominal operation. Therefore it has been decided to use only the monotonous part of the function in the calculation of the tip speed ratio. When the tip speed ratio has been calculated, the effective wind speed is simply calculated from $v = \frac{R\,\omega_r}{\lambda}$.

V. SIMULATION RESULTS

The controller and observer design has been validated through simulations with stochastic wind input with wind speeds in the high wind speed region. In this section some of the simulation results are illustrated with wind input reflecting the "A" turbulence described in the IEC norm [3]. It has been chosen to simulate with a mean wind speed of 18 m/s which results in a turbulence intensity of 17 % in order to get wind speeds in most of the high wind speed region.

The simulation model used in the validation of the controller and observer design is of higher order compared to the design model. The second order nonlinear model of the pitch system is used, and also high frequency components of the generator model are included. Besides this, tower fore-aft and sideways dynamics are included in the structural dynamics.

Simulation results for the observer for the transmission system combined with the wind speed calculation are given in Fig. 8. The left column displays several variables and their estimated values and the right column displays the estimation errors. The true values are illustrated in black and the estimated values in blue and dashed.

It can be observed that the estimation of the rotor/generator speed is very good. At first it seems that the estimation error in the aerodynamic torque, Q_a, and shaft torque, Q_{sh}, is slightly above what can be expected, but a more thorough investigation shows that most of the estimation error is caused by a small time delay in the estimation and that the torque can change rapidly. An example of this is shown in Fig. 9 and from this kind of investigations it has been concluded that the estimation error is appropriate. Further, a standard deviation in the wind speed estimation of $0.2 - 0.3m/s$ is deemed small in the context of using it as gain scheduling variable. For the evaluation of the performance of the designed controller, the performance of the newly designed controller has been tested against a controller designed using classical principles that has been validated to satisfy the design requirements on Vestas wind turbines.

The classical controller is essentially a PID controller for tracking of the generator speed with the pitch orientation as control signal. The dynamic component on the power reference is a feed forward term on the generator speed band pass filtered around the transmission eigen-frequency. The controller is illustrated in (7) below.

$$\beta_{ref} = PID(s) \cdot (\omega_{ref} - \omega_g) \tag{7a}$$

$$P_{ref} = \bar{P}_{ref} + BP(s) \cdot \omega_g \tag{7b}$$

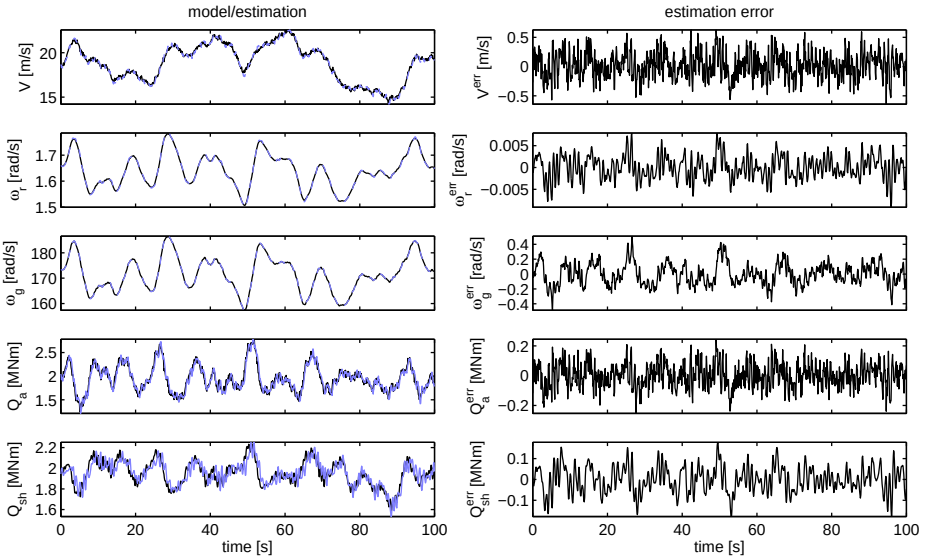


Fig. 8: Simulation of wind observer at high wind speeds. Blue, dashed lines: model, black lines: estimation

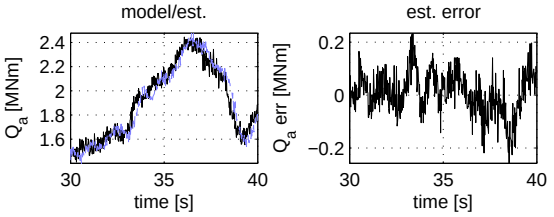


Fig. 9: Simulation of aerodynamics torque. Blue, dashed line: model, black line: estimation

For the simulations the two different closed loops have been operated on identical wind turbine simulation models and with identical wind input. Simulation results are illustrated in Fig. 10 with the newly designed controller in blue and the classical controller in black.

From the figure it can be observed that the LQG controller is superior to the classical controller in the sense of fatigue loads because it has significantly smaller variations in generator speed for similar shaft torque. Furthermore this performance is obtained for similar control effort in the pitch system and less effort in the generator torque.

From the graph of electrical power it can be observed that the LQG controller has slow variations around the nominal power level of $3\ MW$. The classical controller has in contrast much smaller variations except from a few small time intervals in which the fluctuations exceed the level of the LQG controller. From this is is concluded that the classical controller has better performance when observing the power quality.

The difference between the two controllers is caused partly by differences in the

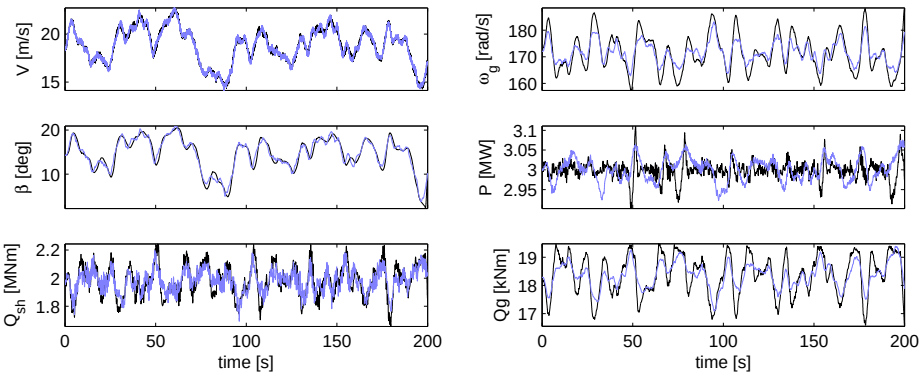


Fig. 10: Comparison of LQG (blue) and classical (black) controller.

tuning process. It should, however, be noted that there is one significant difference between the two controllers in that the LQG controller does not contain the power as a variable. The reason is that power is not suitable as a variable for linear controller design because of the nonlinear coupling between torque and power. This means that the power can only enter in the performance criterion and not in the controlled channel making it more sensitive to under-modelling. In the classical controller this is not an issue because it is tuned on the basis of the nonlinear model making it possible to include the power as a variable in the controlled channel.

VI. CONCLUSIONS

This paper addressed the problem of designing a gain-scheduled linear quadratic controller combined with a state and disturbance estimation algorithm. The controller was designed by linearising the nonlinear plant model along a trajectory of operating points scheduled on the effective wind speed. The observer was designed by a modular approach in which a linear observer was designed for state and disturbance estimation in the transmission model and the wind speed was calculated from inversion of the static aerodynamic model.

The simulation results showed good performance of the observers and the comparison of the resulting closed loop system with another controller designed using classical methods showed good performance in terms of fatigue loads. This good performance came with the cost of slow variations on the active power.

REFERENCES

[1] Carlos E. de Souza and Alexandre Trofino. Gain-scheduled $\mathcal{H}_2$ controller synthesis for linear parameter varying systems via parameter dependent lyapunov functions. *Int. Journal of Robust And Nonlinear Control*, 16:243–257, 2006.

[2] Michael J Grimble. Two and a half degree of freedom LQG controller and application to wind turbines. *IEEE Trans. on Automatic Control*, 39(1):122–127, 1992.

[3] IEC. *IEC 61400-1*. International Standard, 2005.

[4] Jung-Han Kim and Jun-Ho Oh. Disturbance estimation using sliding mode for discrete kalman filter. In *IEEE Conf. on Decision and Control*, pages 1918–1919, 1998.

[5] D.J. Leith and William E. Leithead. Survey of gain-scheduling analysis and design. *Int. Journal of Control*, 73(11):1001–1025, 2000.

[6] Iulian Munteanu, Nicolas Antonio Cutululis, Antoneta Iuliana Bratcu, and Emil Ceanga. Optimization of variable speed wind power systems based on a LQG approach. *Control Engineering Practice*, 13(7):903–912, 2004.

[7] Andy Packard. Gain scheduling via linear fractional transformations. *Systems and Control Letters*, 1994.

[8] Niels K. Poulsen, Torben Juul Larsen, and Morten H. Hansen. Comparison between a PI and LQ-regulation for a 2 MW wind turbine. Technical report, Risø National Laboratory, 2005.

[9] Wilson J. Rugh and Jeff S. Shamma. Research on gain scheduling. *Automatica*, 36:1401–1425, 2000.

[10] Carsten W. Scherer. LPV control and full block multipliers. *Automatica*, 37:361–375, 2001.

[11] Jeff S. Shamma and Michael Athans. Analysis of gain scheduled control for nonlinear plants. *Trans. on Automatic Control*, 35(8):898–907, 1990.

[12] Kasper Zinck Østergaard, Per Brath, and Jakob Stoustrup. Estimation of effective wind speed. In *EWEA, The Science of Making Torque from Wind*, Journal of Physics: Conference Series, 2007. Accepted for publication.

[13] Karl A. Stol. *Dynamics Modeling and Periodic Control of Horizontal-Axis Wind Turbines*. PhD thesis, University of Colorado, 2001.

Paper C

Linear Parameter Varying Control of Wind Turbines Covering Both Partial Load and Full Load Conditions

Contents

Journal paper submitted to:
International Journal of Robust and Nonlinear Control.

Linear parameter varying control of wind turbines covering both partial load and full load conditions

Kasper Zinck Østergaard[1*], Jakob Stoustrup[2], Per Brath[1]

[1] *Turbine Control and Operation R&D, Vestas Wind Systems A/S, Alsvej 21, 8900 Randers, Denmark*
[2] *Automation and Control, Department of Electronic Systems, Aalborg University, Fredrik Bajers Vej 7C, 9220 Aalborg Ø, Denmark*

SUMMARY

This paper considers the design of linear parameter varying controllers for wind turbines in order to obtain a multi variable control law that covers the entire nominal trajectory of operating conditions.

The paper first presents a controller structure for selecting a proper operating condition as a function of estimated wind speed. The dynamic control law is based on linear parameter varying controller synthesis with general parameter dependency by gridding the parameter space.

The controller construction can for medium to large scale systems be difficult from a numerical point of view, because the involved matrix operations tend to be ill-conditioned. The paper proposes a controller construction algorithm together with various remedies for improving the numerical conditioning the algorithm.

The proposed algorithm is applied to the design of a linear parameter varying controller for wind turbines, and a comparison is made with a controller designed using classical techniques to conclude that an improvement in performance is obtained for the entire operating envelope.

KEY WORDS: linear parameter varying control; control of wind turbines; gain scheduling; numerical conditioning

1. Introduction

In the wind energy industry there has been a large focus on increasing the capacity of wind turbines in order to reduce the installation costs when seen relative to the power production during the lifetime of the wind turbine. This has resulted in a rapid growth in rotor size and electrical power production as illustrated in Figure 1. In the period from 1980 to 2003, the largest wind turbines size has grown from approximately 50 kW to 5000 kW, which is more than a 20 % increase per year for more than 20 years. Similar an average increase in rotor size of almost 10 % has been seen in the same time period [1].

This dramatic increase in wind turbine size and capacity has made it ever more challenging to design wind turbines, because many of the structural and electrical components are not scalable, i.e. the costs introduced by scaling the components grow at a higher rate than the benefits from increased production. For the structural components, this means that the individual components must be made lighter without compromising their durability. Basically this can be done in two ways: Classically in the industry, new

*Correspondence to: Kasper Zinck Østergaard, Turbine Control and Operation R&D, Vestas Wind Systems A/S, Alsvej 21, 8900 Randers

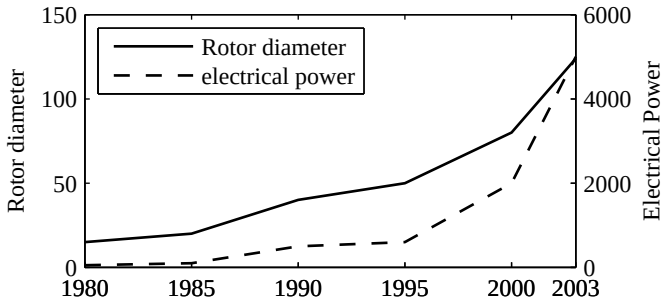


Figure 1: Illustration of growth in wind turbine size.[1]

materials and shapes have been investigated in order to increase the strength and stiffness of various components. Alternatively the fatigue in the components can be reduced by introducing active control.

Several approaches have been investigated for the control of wind turbines. This paper will focus on controller design for pitch regulated variable speed wind turbines and for these wind turbines, the main control loop is set up as a tracking configuration for following a target trajectory for the generator rotational speed – this controller will be denoted "speed controller". In [2] it is discussed in detail how to choose a satisfactory target trajectory along with other design issues related to the speed control loop.

In the industry, the speed controller is typically designed as a PID controller as in [3, 4, 5], because the tuning of these controller structures on the basis of simulations is well understood. When using only the PID controller in a feedback controller setup, the controller is reacting slowly to for example wind gusts leading to stepwise changes in the aerodynamic torque. The reason for this slow response is that the rotor blades can be seen as a large inertia that only slowly changes its rotational speed. To circumvent this problem and get a quicker controller response to these kind of wind speed changes a feed forward term is included from estimated wind speed or aerodynamic torque [6, 7, 8].

As an alternative to the PID controllers a number of other SISO approaches have been studied ranging from memory based methods [9] similar to the PID controller to more specialised nonlinear techniques [10]. Also well-known methods like internal model control [11], controllers based on fuzzy logic [12], and pole placement algorithms [13] have been investigated for the design task of tracking a generator speed reference.

With the increased wind turbine size, it is not enough to consider only the problem of tracking the selected trajectory of operating conditions. Some of the methods presented above take some of the issues into account and in [14] an overview is given for the most relevant issues related to structural loads (tower vibrations, drive train oscillations, and blade vibrations) along with design guidelines for the speed controller. The dampening of these structural oscillations has achieved increased attention within the past few years, with a number of papers concerning individual pitch control to minimise blade loads [15, 16, 17, 18, 19, 20]. Also an increased attention to specific control actions for drive train oscillations and tower oscillations has been presented [21, 22, 23].

With the approach presented above it means that a number of controllers are necessary to be running in parallel in order to meet design requirements: having small deviations from the desired target trajectory together with a reduced amount of struc-

tural fatigue in the major components. By having the controllers operating in parallel, the design task is made even more challenging because it is necessary to understand how the various controllers affect the requirements of the other loops. In [24] it is illustrated how this can be handled by decoupling in the case of two objectives: minimising the tracking error and reducing the drive train oscillations.

When having several objectives the strategy of decoupling the controllers becomes harder and the focus is therefore turned towards optimisation based algorithms which synthesise a multi-variable controller that minimises a combination of the objectives within a specific performance function. The first choice for such a design method is the linear quadratic Gaussian (LQG) design as in [25, 26, 27, 28, 29, 30, 31, 32, 33], because it is known to minimise a weighted combination of standard deviations in the outputs measuring performance under the assumption of a Gaussian disturbance. As an alternative to weighting the standard deviation given Gaussian input, the worst case energy amplification can be minimised by using the $\mathcal{H}_\infty$ design approach as done in [34, 35, 36, 37, 38].

The methods above have the advantage that a lot of engineering experience has been gained in how to use the algorithms in practice and as a consequence, many practical design tools are available. The disadvantage is however, that the methods are based on an assumption of linear models and having equivalent specifications for all operating conditions.

The methods above have shown that linear methods provide good closed loop results when observing local behaviour. A natural choice for controller design covering the entire operating envelope is therefore to design linear controllers at chosen operating conditions and then interconnect them in an appropriate way in order to get a control formulation for the entire operating region. This approach is denoted gain scheduling and in [39, 40, 41] this is done by interpolating the outputs of a set of local controllers (either by linear interpolation or by switching). Alternatively, parameters of the controller are updated according to pre-specified function of a measured/estimated variable [42, 43, 44].

A systematic way of designing such parameter dependent controllers is within the framework of linear parameter varying (LPV) systems. Here, the model will be represented by a linear model at all operating conditions and a controller with similar parameter dependency is synthesised to guarantee a certain performance specification for all possible parameter values within a specified set. A major difference to the previous methods is that the method gives a way also to take the rate-of-variation of the parameter into account. In classical gain scheduling approach, such as [41, 43] an underlying assumption is that the parameter will only change slowly compared to the system dynamics.

In [45, 46, 47] the LPV design procedure is applied for the special case of affine parameter dependency. However, affine parameter dependency is a very strict requirement for designing controllers for wind turbines in the entire operating condition. Mainly because the performance criteria are very different in partial load when compared to full load control, but also because the nonlinear aerodynamics need to be approximated by a second order function of the chosen parameters. In [45, 46] this is handled by designing different LPV controllers for the below and above rated wind speeds and then switching between them.

The gain-scheduled design methods in the literature consider one controller for par-

tial load and another controller for full load operation (Except for [47] in which a simplistic parameter dependency is assumed). This means that a method for bumpless transfer between two very different controllers needs to be implemented to make the control law work in practice. In this paper we consider an alternative approach in which a gain-scheduled controller is designed for both partial load and full load operation for a 3 MW wind turbine with a rotor diameter of 90 m. Further we consider not only the tracking controller, but also the interconnection structure for generation of the reference trajectory.

As performance criteria the design will focus on tracking a generator speed reference together with minimisation of fatigue damage in vital components, e.g. drive train and tower. Also control effort is taken into account to avoid especially the pitch system to wear out and the power fluctuations to be too high.

The aerodynamics is highly nonlinear and causes the dynamics to change significantly over the operating region and because the controller must cover two very different operating regions, the performance weights will depend heavily on the scheduling parameter. This means that the weighted plant model will have a complicated parameter dependency, and the available design tool for LPV design with affine parameter dependency is deemed inappropriate. Instead it has been chosen to focus on a grid based method as in [48]. In this method the design problem is handled by solving a set of linear matrix inequalities (LMIs) at a number of design points followed by the construction of a controller from the design variables. The controller construction is very sensitive to numerical issues and a number of remedies are presented for conditioning the construction procedure.

The paper is structured as follows. In Section 2 we discuss the closed loop objectives that will be considered in the design process and in Section 3 the considered wind turbine model is presented. A controller structure is given in Section 4 for tracking the desired target trajectory followed by a presentation of the performance channel for the LPV design in Section 5. Then in Section 6 the LPV controller controller design method is presented followed by a discussion of numerical remedies for controller construction in Section 7. In Section 8 closed loop simulation results are presented and conclusions are given in Section 9.

2. Control objectives

The main objective for control of wind turbines is to maximise the trade-off between the annual energy production and the cost of the wind turbine in terms of construction and maintenance costs. This has resulted in two operational modes for pitch regulated variable speed wind turbines

Partial load operation is the mode in which there is not enough kinetic energy in the wind to achieve nominal electrical power production. In this mode the primary objective is to control the pitch and rotor speed to achieve the maximum aerodynamic efficiency of the wind turbine.

Full load operation is the mode in which the kinetic energy in the wind field has exceeded the nominal electrical power production and conversion losses. In this mode the generator speed should be kept close to the nominal speed and the pitch angle should be controlled to achieve nominal electrical power production. Further it is important to reduce the fluctuations on the power, known as flicker.

In both operating regions it is important to minimise the fatigue loads in critical structural components, and in this paper the following structural loads will be considered:

Fatigue damage in drive train will be measured by the torque between an inertia representing the high speed shaft and an inertia representing the slow speed shaft (including blades).

Fatigue damage in tower fore aft movement will be measured by the position of the tower top.

Wear in pitch system will be measured by the travelled distance of the pitch system.

Also it is crucial that the generator speed does not exceed the maximum generator speed in order not to overheat the electrical components. When combining these conditions, the steady state trajectory of the wind turbine can be described as a function of wind speed as in Figure 2. The requirements regarding optimisation of power production and deviations from nominal generator speed then amounts to tracking this steady state trajectory.

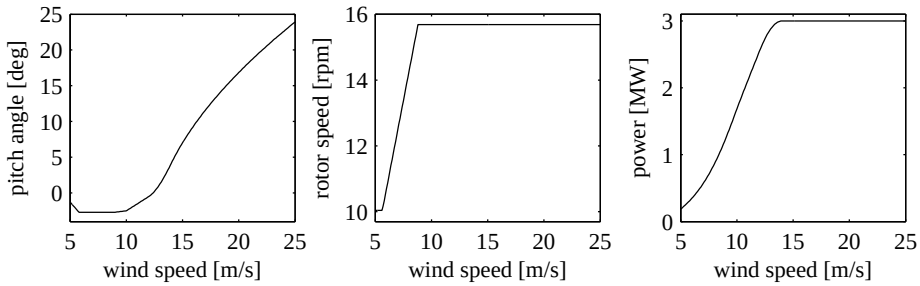


Figure 2: Illustration of operating region for wind turbine.

The other performance criteria can be seen as detuning of the tracking controller to limit structural oscillations and high amounts of control effort. For the pitch activity it is well-known that during high wind speeds a high amount of activity is required to track the target trajectory and minimise tower loads. During low wind speeds, the pitch angle should be kept close to the value yielding maximum aerodynamic performance. This means that there needs to be high emphasis on the wear in the pitch system during low wind speeds whereas higher variations can be tolerated during high wind speeds.

In contrast the electrical power should have only small fluctuations during high wind speeds whereas higher fluctuations can be accepted in partial load operation in which the electrical power should follow the available power closely.

This leads to the following different requirements for the control law during partial load operation and full operation:

Partial load operation

- Track trajectory of optimal steady state operating conditions
- Minimise pitch activity
- Minimise drive train and tower fore-aft loads

Full load operation

- Track trajectory of optimal steady state operating conditions
- Minimise electrical power fluctuations
- Minimise drive train and tower fore-aft loads

As discussed above the requirements are different for low wind speeds and high wind speeds. Not only in terms of use of the control signals, but also for fatigue loads and tracking performance because the energy in the wind increases with wind speed. Since the wind speed will vary between low wind speeds and high wind speeds the performance criteria will change significantly during operation. In the sequel we shall demonstrate that a systematic approach to obtain the above objectives can be achieved by virtue of an LPV controller design. This way we obtain a smooth transition between controllers satisfying the design requirements along the design trajectory. Further it shall be noted that nonlinearities caused for example by the aerodynamics can easily be incorporated into the LPV design framework.

3. Wind turbine model

Based on the control objectives in the previous section, a nonlinear model of suitable complexity will be constructed. This model is then linearised along the desired trajectory of operating conditions to obtain a model that is scheduled on wind speed.

It has been evaluated that five components are necessary for a proper controller design according to the presented requirements: Drive train, tower, aerodynamics, pitch system, and generator and converter system. These five components will be presented in the following sections.

3.1. Drive train

The drive train is modelled by two inertias interconnected by a spring and damper. In this design, the dynamics of the blades will not be included and the blade stiffness and inertia is therefore lumped into the slow speed shaft. Friction is included in terms of linear friction coefficients on each inertia. This leads to the following formulation of the drive train model:

$$\begin{aligned} J_r\,\dot{\omega}_r &= Q_a - B_r\,\omega_r - N\mu\,(N\omega_r - \omega_g) - N\,K\,\theta_\Delta \\ J_g\,\dot{\omega}_g &= -Q_g - B_g\,\omega_g + \mu\,(N\omega_r - \omega_g) + K\,\theta_\Delta \\ \dot{\theta}_\Delta &= N\omega_r - \omega_g \\ Q_{sh} &= \mu(N\omega_r - \omega_g) + K\,\theta_\Delta \end{aligned}$$

where: ω_r and ω_g is the rotational speed of respectively the slow and high speed shaft, Q_a and Q_g are the input torques on each shaft, J_r and J_g are the two moments of inertia, B_r and B_g are the friction coefficients on the two shafts, N is the gearing ratio, μ and K are the stiffness and damping coefficients of the interconnection of the two inertias, and Q_{sh} is the strain in the shaft used when measuring fatigue damage. This model is easily

written in the standard state space form (1) with $x = [\omega_r\ \omega_g\ \theta_\Delta]^T$, $u = [Q_a\ Q_g]^T$, and $y = [\omega_r\ \omega_g\ Q_{sh}]^T$.

$$\dot{x} = Ax + Bu \tag{1a}$$
$$y = Cx + Du \tag{1b}$$

From Figure 2 it can be concluded that the aerodynamic torque, Q_a varies in the range from a few kNm to more than $1\ MNm$. Also the other variables vary significantly in size over the operating region which can not only cause problems in the implementation/simulation phase, but also cause the design algorithm to suffer from numerical difficulties. It has therefore been chosen to perform a suitable coordinate transformation and scaling of inputs/outputs. This has been done by performing a number of closed loop simulations with a PID controller that has been shown to satisfy acceptable performance. The states, inputs, and outputs are then scaled so that the standard deviation of all of these variables is close to unity.

3.2. Tower

The tower will be modelled by a mass spring system as in (2) with y as the displacement of the tower top, m is the equivalent mass, B_t is the structural damping of the tower, K_t is tower stiffness, and F_t is the thrust on the tower. Again a suitable scaling of states, inputs and outputs is performed.

$$m\ddot{y} = F_t - B_t\dot{y} - K_t y \tag{2}$$

3.3. Aerodynamics

The aerodynamics will be approximated by static functions of spatial average of wind speed, rotor speed, and pitch angle according to [49]

$$Q_a = \frac{1}{2}\,\rho\,\pi\,R^2\,\frac{v^3}{\omega_r}\,c_P(\lambda,\beta) \tag{3a}$$
$$F_t = \frac{1}{2}\,\rho\,\pi\,R^2\,v^2\,c_T(\lambda,\beta) \tag{3b}$$
$$\lambda = \frac{\omega_r R}{v} \tag{3c}$$

where ρ is the air density, R is the rotor radius, and λ is denoted the tip speed ratio. The two coefficients c_P and c_T represent respectively the aerodynamic efficiency and the thrust coefficient, which we illustrate in Figure 3.

For the linearised model, the partial derivatives of the two nonlinear functions are evaluated along the desired trajectory to obtain a description of the form (4) with the tilde indicating deviations from the design equilibrium point, EQ.

$$\tilde{Q}_a = \left.\frac{\partial Q_a}{\partial v}\right|_{EQ}\cdot\tilde{v} + \left.\frac{\partial Q_a}{\partial \beta}\right|_{EQ}\cdot\tilde{\beta} + \left.\frac{\partial Q_a}{\partial \omega_r}\right|_{EQ}\cdot\tilde{\omega}_r \tag{4a}$$
$$\tilde{F}_t = \left.\frac{\partial F_t}{\partial v}\right|_{EQ}\cdot\tilde{v} + \left.\frac{\partial F_t}{\partial \beta}\right|_{EQ}\cdot\tilde{\beta} + \left.\frac{\partial F_t}{\partial \omega_r}\right|_{EQ}\cdot\tilde{\omega}_r \tag{4b}$$

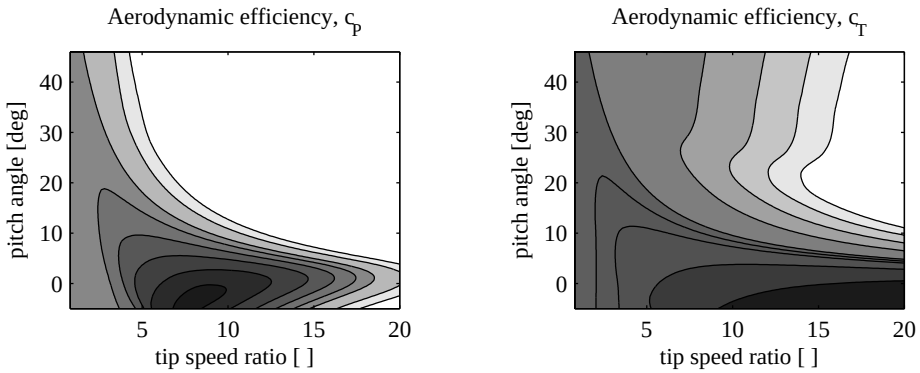


Figure 3: Illustration of aerodynamic efficiency and thrust coefficient. Light colours indicate small coefficients and dark colours indicate large coefficients.

3.4. Pitch system

The pitch system is a complicated and highly nonlinear hydraulic actuator that on the basis of a given control voltage moves the piston and thereby rotates the blade at a certain rotational rate. To simplify the LPV controller design, an inner control loop has been designed to handle the nonlinear effects. Then the closed loop pitch system can be described by a second order linear model from pitch reference to pitch position.

3.5. Generator and converter system

For the other actuator, the generator and converter system, high frequency requirements in terms of minimisation of power fluctuations apply. This is better handled in an inner loop which has been designed to meet these requirements. For the LPV controller design the generator and converter system can then be approximated by a first order system with the associated time constant being scheduled on generator speed as in (5)[†]. Again the states, inputs and outputs are scaled appropriately and the nonlinear model is not linearised because it is already on a very simple LPV form, i.e. affine dependency.

$$Q_g = \frac{T_0\,\omega_g}{s + T_0\,\omega_g}\,Q_{g,ref} \tag{5a}$$

$$P_e = \eta\,Q_g\,\omega_g \tag{5b}$$

3.6. Interconnection

With the above described components, the wind turbine model can be obtained by standard interconnection of the blocks described in the above sections. The only nonlinear components described above are the aerodynamics and generator system. With the assumption that the wind turbine is operating on the nominal trajectory specified in Figure 2, the equilibrium values for pitch angle and rotor/generator speed can be described uniquely by the wind speed. This means that the wind turbine model can be described

[†] The authors are aware of the abuse of notation since ω_g varies in time.

as an LPV model scheduled on only wind speed as in (6).

$$\dot{x} = A(v)\,x + B_p(v)\,v + B(v)\begin{bmatrix} Q_{g,ref} \\ \beta_{ref} \end{bmatrix} \tag{6a}$$

$$z = \begin{bmatrix} Q_{sh} \\ y \\ \omega_g \\ \beta \end{bmatrix} = C_p(v)\,x \tag{6b}$$

$$y = \begin{bmatrix} \omega_g \\ \ddot{y} \end{bmatrix} = C(v)\,x + F(v)\,v \tag{6c}$$

4. Controller structure

Before discussing the specific choice of controller structure, it is important to understand the desired behaviour of the control system in the different operating modes. In partial load operation an important objective is to maximise the electrical power production. This means that the pitch angle and rotor speed should be controlled in a way so that the aerodynamic coefficient, c_P, is maximised. By observing the leftmost illustration on Figure 3 this means the pitch angle should be kept constant close to -2 deg and the rotor speed should be controlled proportional to the effective wind speed to get a desired tip speed ratio. In practice this will be done by tracking a generator speed reference via an update of the generator torque.

In full load operation, the generator speed and power needs to be kept close to constant nominal values. This means that the generator torque should be varied as little as possible and the pitch position as the remaining control signal is controlled to keep the aerodynamic power constant thereby keeping the electric power constant.

The main part of the controller structure is then a setup for tracking a specified generator speed reference by mainly using generator torque in partial load operation and pitch angle in full load operation. On top of this main structure, there will be components using both control signals in order to minimise the oscillations in drive train and tower fore aft movement. For the minimisation of tower oscillations a measurement of tower acceleration is included.

When designing linear controllers for a nonlinear system, the system is typically linearised by using a first order Taylor series of the nonlinear differential equation. In this formulation the desired equilibrium is transformed into the origin in the linearised set of variables. Then when implementing the controller in practice, the control signals associated with the equilibrium in the design must be added to the output of the controller. Similarly the equilibrium of the measurements should be subtracted from the online measurements. In this controller formulation, the LPV model will be obtained using linearisation along a chosen trajectory of operating conditions. This means that the same issue therefore applies to the implementation of the controller which is illustrated in Figure 4. In the block diagram the three blocks having inputs from estimated wind speed, $\hat{v}$, should be interpreted as nonlinear static functions of an estimate of the effective wind speed – derived from the equilibrium conditions described in Figure 2. The wind speed estimate can for this purpose be calculated using a dynamic observer as in [50].

From Figure 4 it can be seen that this way of handling the issue of equilibrium points being different from the origin will lead to feed-forward terms. For classical controllers

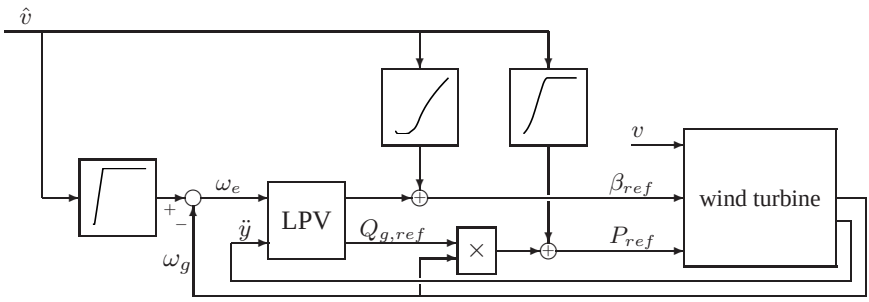


Figure 4: Implementation structure for an LPV controller based on linearisation along a trajectory of equilibrium points.

such feed-forward terms have proven to be beneficial as discussed in the introduction. There is, however a risk that the feed-forward term might degrade overall performance if it operates in the same frequency region as the LPV controller. In this case it should be included in the LPV design using a 2-DOF controller design. It is expected that the feed-forward terms will not interfere with the LPV controller because of the successful results obtained for classical controllers. The LPV design presented in this paper will therefore not include feed-forward terms, because this would increase the dimensionality of the design problem and thereby make numerical difficulties more likely.

It is not expected that the open loop calculation of feed-forward terms will result in exact tracking of the target trajectory e.g. as a consequence of modelling uncertainty. To counter this, feedback terms with slow integral action have been included. For partial load control, it has been decided that only a term for the power reference should be included as

$$P_0 = \bar{P}(\hat{v}) + \frac{I_{plc}}{s} \cdot (\bar{\omega}_g(\hat{v}) - \omega_g)$$

with $\bar{P}(\hat{v})$ and $\bar{\omega}_g(\hat{v})$ the open loop calculation of respectively equilibrium power and generator speed from the estimate of wind speed. ω_g is measured generator speed and I_{plc} is the integration time. In full load operation asymptotic tracking of nominal power is achieved by integral action of the error in tracking the power reference

$$P_0 = \bar{P}(\hat{v}) + \frac{I_{flc,P}}{s} \cdot \big(\bar{P}(\hat{v}) - P\big)$$

with P as measured power. Also asymptotic tracking of generator speed is in full load operation obtained by an integral term with the pitch angle as the control signal:

$$\beta_0 = \bar{\beta}(\hat{v}) + \frac{I_{flc,\beta}}{s} \cdot (\bar{\omega}_g(\hat{v}) - \omega_g)$$

There is a significant difference between the structure for integral action in partial load and full load operation. To make the control law implementable, the output of the integrators will be gain scheduled on wind speed in a similar way to what is done in [39]. This is done to ensure bump-less transfer between the two operating modes. Further the integrators will be reset when the scheduling gain is zero. This leads to the calculation of the two "feed-forward" terms illustrated in Figure 5.

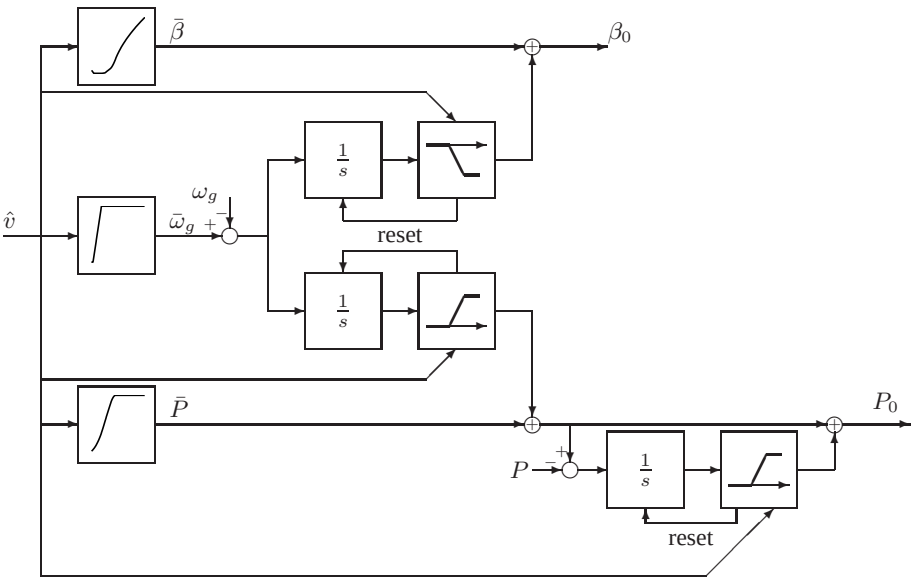


Figure 5: Calculation of "feed-forward" terms with included integral action.

Remark 8. *When designing the integration time for the three integrators it should be ensured that the bandwidth of the outer tracking controller is sufficiently slow not to interfere with the LPV controller.*

5. Selection of performance channels and associated weights

In the applied formulation of LPV control, the performance is measured in terms of energy amplification from a number of selected inputs to a selection of outputs – very similar to $\mathcal{H}_\infty$ control. However the objectives in Section 2 are given in physical terms, such as to minimise the fatigue damage in the tower, to limit the maximum generator speed, etc. As performance input, the wind speed is chosen and to transform the objectives into something that can be measured by energy amplification, the following performance outputs have been chosen:

Tower top velocity in fore-aft direction, $\dot{y}$: The tower is lightly damped close to its eigen-frequency and it is therefore most important to actively dampen oscillations around this frequency. In the linearised model, the tower top velocity has a peak at the tower eigen-frequency for all nominal operating modes. Thus by introducing a frequency independent scaling of this performance output, the dampening of the oscillations around the tower eigen-frequency can be included in the performance function for controller design.

Torsion torque in drive train, Q_{sh}: For the drive train the issue is very similar, because it is lightly damped around its eigen-frequency. The shaft torque, Q_{sh}, has a peak at this eigen-frequency which means that a frequency independent scaling can be used – with the same argumentation as for the tower. If a high emphasis is to be put on the drive train oscillations it might be advantageous to use the difference in rotational speed between the two inertias ($N\omega_r - \omega_g$), because this

will decrease the weight on low frequency and thus increasing the weight on the eigen-frequency.

Tracking error for generator speed, ω_e: The main target of using this performance output is to minimise the variations in generator speed from the design trajectory caused by variations in wind speed. The main content of the wind speed is in lower frequencies which means that putting an emphasis on low frequency components will be satisfactory. Because the low frequency components are dominant in the transfer function from wind speed to generator speed (for all the linearised models) this means that again a frequency independent scaling is applicable. However, the scaling should be gain scheduled by effective wind speed to take into account that the issue of over-speeds is most likely to occur around rated generator speed, i.e. a larger weight for medium wind speeds than on high and low wind speeds should be used.

Pitch activity, β: It is very undesirable to have high frequency content in the controlled pitch angle because this would cause a high amount of wear in blade bearings and in the pitch hydraulics. As a consequence it is important to limit the activity in partial load control because the pitch has very limited authority in this region if the power production is to be maximised. Because of this it has been decided to include a scaling in form of a high pass filter to put make the high frequency pitch activity more expensive than the low frequency activity. Further the scaling is gain scheduled with wind speed to allow for high activity in full load operation and low activity in partial load operation.

Variations in generator torque, Q_g: In the linearised model for controller design, the electrical power is not directly accessible as a linear combination of states and inputs. However, in full load operation the variations in generator torque relates in a direct way to power fluctuations, because the steady state generator speed be constant and variations are limited by the tracking controller. Therefore the generator torque will be weighted by a large frequency independent scaling at high wind speeds whereas a smaller scaling is used for low wind speeds to allow for sufficient tracking of generator speed reference.

With the selected performance inputs, outputs and weighting functions, the open loop model for the LPV design can now be formulated with the inputs and outputs defined as

$$z = \begin{bmatrix} W_{\dot{y}}(\hat{v}) & 0 & 0 & 0 & 0 \\ 0 & W_{Q_{sh}}(\hat{v}) & 0 & 0 & 0 \\ 0 & 0 & W_{\omega}(\hat{v}) & 0 & 0 \\ 0 & 0 & 0 & \frac{W_{\beta}(\hat{v})}{(Ts+1)^3} & 0 \\ 0 & 0 & 0 & 0 & W_{Q_g}(\hat{v}) \end{bmatrix} \begin{bmatrix} \dot{y} \\ Q_{sh} \\ \omega_e \\ \beta \\ Q_g \end{bmatrix}$$

$$w = v \quad , \quad y = \begin{bmatrix} \omega_e \\ \ddot{y} \end{bmatrix} \quad , \quad u = \begin{bmatrix} \beta_{ref} \\ Q_{g,ref} \end{bmatrix}$$

It is expected that the control law will result in a high amount of $3P$[‡] content. The issue of $3P$ oscillations is expected to be handled better by an individual pitch controller

[‡]The $3P$ frequency is three times the rotational speed of the rotor.

interconnected with the LPV controller. To decouple these two controllers, a notch filter will be applied to the control outputs at the $3P$ frequency. Because the $3P$ frequency is close to the drive train eigen-frequency, an oscillator is also applied to the control signal, $Q_{g,ref}$, by using the ideas from internal model control.

6. Linear parameter varying control

With the overall controller structure presented in Section 4, the control objectives given in Section 2, and the choice of performance channels and weights decided in Section 5, the preparations for the dynamic LPV controller design have been done. This leads to the design of an LPV controller for the control of wind turbines. This section will present the base of the algorithm for controller synthesis and construction.

6.1. Synthesis of LPV controllers

In [51] an analysis is provided for LPV closed loop systems under the assumption of slowly varying parameters and in [52, 53] the general framework for designing LPV controllers with arbitrary rate of variation was presented in an LMI formulation. In the following, the design procedure is summarised and it is discussed how to handle numerical difficulties that might occur for higher order systems.

During this section, an abstract representation will be used for the model derived in Section 3 combined with the performance weights. We consider a weighted open loop model of the form (7a) with associated controller (7b). The closed loop interconnection can then be expressed as in (7c) with the short hand notation in (7d).

$$\begin{bmatrix} \dot{x}(t) \\ z(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} A(\delta(t)) & B_p(\delta(t)) & B(\delta(t)) \\ C_p(\delta(t)) & D(\delta(t)) & E(\delta(t)) \\ C(\delta(t)) & F(\delta(t)) & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ w(t) \\ u(t) \end{bmatrix} \tag{7a}$$

$$\begin{bmatrix} \dot{x}_c(t) \\ u(t) \end{bmatrix} = \begin{bmatrix} A_c(\delta(t)) & B_c(\delta(t)) \\ C_c(\delta(t)) & D_c(\delta(t)) \end{bmatrix} \begin{bmatrix} x_c(t) \\ y(t) \end{bmatrix} \tag{7b}$$

$$\begin{aligned} \begin{bmatrix} \dot{x}(t) \\ \dot{x}_c(t) \\ z(t) \end{bmatrix} = \Bigg(& \begin{bmatrix} A(\delta(t)) & 0 & B_p(\delta(t)) \\ 0 & 0 & 0 \\ C_p(\delta(t)) & 0 & D(\delta(t)) \end{bmatrix} + \\ & + \begin{bmatrix} 0 & B(\delta(t)) \\ I & 0 \\ 0 & E(\delta(t)) \end{bmatrix} \begin{bmatrix} A_c(\delta(t)) & B_c(\delta(t)) \\ C_c(\delta(t)) & D_c(\delta(t)) \end{bmatrix} \begin{bmatrix} 0 & I & 0 \\ C(\delta(t)) & 0 & F(\delta(t)) \end{bmatrix} \Bigg) \begin{bmatrix} x(t) \\ x_c(t) \\ w(t) \end{bmatrix} \end{aligned} \tag{7c}$$

$$\begin{bmatrix} \dot{x}_{cl}(t) \\ z(t) \end{bmatrix} = \begin{bmatrix} A_{cl}(\delta(t)) & B_{cl}(\delta(t)) \\ C_{cl}(\delta(t)) & D_{cl}(\delta(t)) \end{bmatrix} \begin{bmatrix} x_{cl}(t) \\ w(t) \end{bmatrix} \tag{7d}$$

In the above equations, $x(t)$ and $x_c(t)$ denoting the state vector of respectively the open loop system and controller, $y(t)$ is the measured variables and $u(t)$ is the control signal. The input from the performance channel is denoted $w(t)$ and its associated output is $z(t)$. It can be noted that all system and controller matrices depend on a time-varying parameter, $delta(t)$, which is assumed online available (not necessarily a measured variable – it could also be the output from an estimator[§]) and therefore also used to schedule

[§]It should be noted that using an estimate for the scheduling parameter will invalidate the assumption that it is exogenous. In practice this has not been experienced to pose a problem.

the controller. In this paper, the performance criterion will be measured by the induced $\mathcal{L}_2$ gain given in Definition 9.

Definition 9. *Let Σ denote a closed loop system described by (7c), and Δ be the set of possible parameter values. Then Σ is said to satisfy the induced $\mathcal{L}_2$ gain, $\gamma > 0$, if*

$$\exists\, \epsilon \in \mathbb{R} : \int_0^\infty z(t)^T z(t)\, dt \leq \left(\gamma^2 - \epsilon^2\right) \int_0^\infty w(t)^T w(t)\, dt$$

for all possible trajectories $(\xi(.), z(.), w(.))$ satisfying $w(.) \in \mathcal{L}_2$, $x(0) = x_c(0) = 0$, and $\delta(.) \in \Delta$.

Via the Kalman-Yakubovich-Popov Lemma [54, 55, 56] it has been shown that a sufficient condition for satisfying the performance specification is given by the LMI, in (8) with $\prec$ and $\succ$ interpreted as generalised inequalities in the convex cone of semidefinite matrices, i.e. $X \prec Y$ is interpreted as $X - Y$ being negative definite.

$$\text{for all } \delta \in \Delta \text{ there exist } X_{cl} \succ 0 :$$

$$\begin{bmatrix} I & 0 \\ 0 & I \\ A_{cl}(\delta) & B_{cl}(\delta) \\ C_{cl}(\delta) & D_{cl}(\delta) \end{bmatrix}^T \begin{bmatrix} 0 & 0 & X_{cl} & 0 \\ 0 & -\gamma I & 0 & 0 \\ X_{cl} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\gamma} I \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & I \\ A_{cl}(\delta) & B_{cl}(\delta) \\ C_{cl}(\delta) & D_{cl}(\delta) \end{bmatrix} \prec 0 \tag{8}$$

The step from closed loop analysis to controller synthesis is done by describing the closed loop system matrices as functions of known open loop matrices and unknown controller matrices as in (7c) and (7d), and then inserting the expression in (8). This results in a nonlinear matrix inequality in the new set of variables, X_{cl}, $A_c(\delta)$, $B_c(\delta)$, and $C_c(\delta)$ which cannot directly be solved using convex optimisation.

It has been shown in [57, 58] that the nonlinear matrix inequality can be transformed into two LMIs by eliminating the controller variables from the original matrix inequality. The conditions for controller synthesis is given in Theorem 10 in which the only unknown variables are X and Y.

Theorem 10. *Consider an open loop system described by (7a) with associated parameter, $\delta(t) \in \Delta$ allowed to vary arbitrarily fast. Then there is a stabilising controller of the form (7b) that satisfies a closed loop performance level, γ, if there exist symmetric matrices X and Y satisfying*

$$\begin{bmatrix} X & I \\ I & Y \end{bmatrix} \succ 0 \tag{9a}$$

$$\begin{bmatrix} C(\delta)_\perp \\ F(\delta)_\perp \end{bmatrix}^T \begin{bmatrix} I & 0 \\ 0 & I \\ A(\delta) & B_p(\delta) \\ C_p(\delta) & D(\delta) \end{bmatrix}^T \begin{bmatrix} 0 & 0 & X & 0 \\ 0 & -\gamma I & 0 & 0 \\ X & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\gamma} I \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & I \\ A(\delta) & B_p(\delta) \\ C_p(\delta) & D(\delta) \end{bmatrix} \begin{bmatrix} C(\delta)_\perp \\ F(\delta)_\perp \end{bmatrix} \prec 0 \tag{9b}$$

$$\begin{bmatrix} \star \\ \star \end{bmatrix}^T \begin{bmatrix} \star \\ \star \\ \star \\ \star \end{bmatrix}^T \begin{bmatrix} 0 & 0 & Y & 0 \\ 0 & -\frac{1}{\gamma} I & 0 & 0 \\ Y & 0 & 0 & 0 \\ 0 & 0 & 0 & \gamma I \end{bmatrix} \begin{bmatrix} -A(\delta)^T & -C_p(\delta)^T \\ -B_p(\delta)^T & -D(\delta)^T \\ I & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} B(\delta)_\perp^T & 0 \\ E(\delta)_\perp^T & I \end{bmatrix} \succ 0 \tag{9c}$$

for all possible parameter values, $\delta \in \Delta$, with arbitrary rate of variation.

The matrix inequalities in Theorem 10 are clearly linear in the new variables X and Y. In the context of controller synthesis it is typically desirable to determine a controller that minimises γ and not only one that satisfies a pre-specified performance specification. This can be done by bisection or alternatively by using the well-known Schur lemma, whereby the two matrix inequalities are transformed into matrix inequalities that are also linear in γ. Then γ can be used directly in the convex optimisation problem.

In the above performance specification it is required to solve infinitely many LMIs – one for each possible parameter – and the problem can therefore not be solved in finite time. In the special case of affine parameter dependency and with the parameters varying within a convex polytope, the problem reduces to checking only the vertices [59]. As discussed in the introduction, affine parameter dependency is not expected to give a satisfactory performance for the application in mind. Alternatively a method has been developed for designing controllers for the case of rational parameter dependency [60, 61]. This approach is much more appealing, but suffers from numerical issues in the construction of controllers from the synthesis variables. One issue is that the numerics is highly dependent on the choice (and size) of a linear fractional representation which makes it a very demanding task to design controllers for which the performance function is scheduled and the scheduling might change between iterations in the design process. Because of these reasons it has been chosen to focus on an approximative method (using a grid) presented in [62] but still with the assumption of arbitrarily fast parameter variations.

The density of the grid is to be determined from a trade-off between having a lot of grid points causing heavy computational time and a few grid points not catching the nonlinear behaviour to a sufficient degree. To understand how close the grid points should be, observe first the following: Let

$$M_1(x) = M_{1,0} + \sum_{i=1}^{N} x_i \cdot M_{1,i} \prec 0 \quad , \quad M_2(x) = M_{2,0} + \sum_{i=1}^{N} x_i \cdot M_{2,i} \prec 0$$

Then with $M_{3,i} = \alpha M_{1,i} + (1-\alpha) M_{2,i}$ with $0 \leq \alpha \leq 1$ we have that

$$\begin{aligned} M_3(x) &= M_{3,0} + \sum_{i=1}^{N} x_i \cdot M_{3,i} \prec 0 \\ &= \alpha \underbrace{\left(M_{1,0} + \sum_{i=1}^{N} x_i \cdot M_{1,i} \right)}_{M_1(x)} + (1-\alpha) \underbrace{\left(M_{2,0} + \sum_{i=1}^{N} x_i \cdot M_{2,i} \right)}_{M_2(x)} \prec 0 \end{aligned}$$

which means that if we can assume affine parameter dependency of the weighted open loop in the interval between two grid points, the stability and performance will be given for the intermediate parameter values. For the specific application of LPV control for wind turbines, it has been chosen to select two grid points in each of the following three operating modes: partial load operation with variable generator speed, partial load operation with nominal generator speed, and full load operation. This provides an acceptable trade-off between having a few number of grid points and not violating the assumption of piecewise affine parameter dependency too much.

6.2. Controller construction

By solving the optimisation problem for controller synthesis in Theorem 10, we get an achievable performance level, γ, and the matrices X and Y for the quadratic storage/Lyapunov function used to measure the performance and stability of the closed loop interconnection.

Note on the other hand that the controller variables A_c, B_c, C_c, and D_c are eliminated in the step from analysis to synthesis which means that they are not directly available from solving the synthesis LMIs. However, observe that under the assumption that X_{cl} and γ are known variables, the analysis inequality (8) is a quadratic matrix inequality (QMI) in the remaining controller variables. This QMI has a particular structure of a known constant inner term and the outer terms are structured as the identity over a term which is affine in an unstructured matrix (the controller variables). This particular structure for QMIs is linearised in [57, 58] by eliminating the variables in the outer terms. Further the proofs are constructive which means that even though the controller variables are not directly available from the synthesis LMIs, they can be constructed afterwards by following the steps given in the proofs. The result is presented in Lemma 11 and a construction procedure based on the constructive proof in [60] is given in Algorithm 12.

Lemma 11 (Elimination Lemma [57, 58])**.** *Assume there is a matrix $M \in \mathbb{R}^{n\times m}$ and a symmetric matrix $P \in \mathbb{R}^{(n+m)\times(n+m)}$ with no zero eigenvalues*

$$\begin{bmatrix} I \\ L^T K R + M \end{bmatrix}^T P \begin{bmatrix} I \\ L^T K R + M \end{bmatrix} \prec 0 \tag{10}$$

in the unstructured unknown K has a solution if and only if

$$R_\perp^T \begin{bmatrix} I \\ M \end{bmatrix}^T P \begin{bmatrix} I \\ M \end{bmatrix} R_\perp \prec 0 \tag{11a}$$

$$L_\perp^T \begin{bmatrix} -M^T \\ I \end{bmatrix}^T P^{-1} \begin{bmatrix} -M^T \\ I \end{bmatrix} L_\perp \succ 0 \tag{11b}$$

Algorithm 12

Step 1 Choose non-singular matrices Φ and Ψ such that $L\Phi = [L_1\ 0]$ and $R\Psi = [R_1\ 0]$ such that $L_1 \in \mathbb{R}^{n_L\times m_L}$ and $R_1 \in \mathbb{R}^{n_R\times m_R}$ have full column rank

Step 2 Let

$$\Lambda = \begin{bmatrix} \Lambda_{11} & \Lambda_{12} \\ \Lambda_{21} & \Lambda_{22} \end{bmatrix} \triangleq \Phi^T M \Psi, \quad \Upsilon \triangleq \begin{bmatrix} I & 0 \\ 0 & 0 \\ 0 & I \\ \Lambda_{21} & 0 \end{bmatrix}, \quad \Omega \triangleq \begin{bmatrix} 0 \\ I \\ \Lambda_{12} \\ \Lambda_{22} \end{bmatrix},$$

$$\Pi \triangleq \begin{bmatrix} \Psi & 0 \\ 0 & \Phi^{-T} \end{bmatrix}^T P \begin{bmatrix} \Psi & 0 \\ 0 & \Phi^{-T} \end{bmatrix}$$

Step 3 Find $\Theta_1 \in \mathbb{R}^{m_L\times m_R}$ and Θ_2 such that

$$\begin{bmatrix} \Theta_1 \\ \Theta_2 \end{bmatrix}^T \left(\Upsilon^T \Pi \Upsilon - \Upsilon^T \Pi \Omega \left(\Omega^T \Pi \Omega \right)^{-1} \Omega^T \Pi \Upsilon \right) \begin{bmatrix} \Theta_1 \\ \Theta_2 \end{bmatrix} \prec 0 \tag{12}$$

e.g. by an eigenvalue decomposition.

Step 4 Then $K = L_1^{-T}(\Theta_2\Theta_1^{-1} - \Lambda_{11})R_1^{-1}$

Returning to Theorem 10, it can now be seen that (8) with control variables inserted is equivalent to (9b) and (9c) by plugging in

$$X_{cl} = \begin{bmatrix} X & U \\ U^T & \hat{X} \end{bmatrix} \quad , \quad X_{cl}^{-1} = \begin{bmatrix} Y & V \\ V^T & \hat{Y} \end{bmatrix}$$

$$L = \begin{bmatrix} 0 & B(\delta) \\ I & 0 \\ 0 & E(\delta) \end{bmatrix}^T \quad , \quad R = \begin{bmatrix} 0 & I & 0 \\ C(\delta) & 0 & F(\delta) \end{bmatrix} \quad , \quad K = \begin{bmatrix} A_c(\delta) & B_c(\delta) \\ C_c(\delta) & D_c(\delta) \end{bmatrix} \tag{13}$$

$$M = \begin{bmatrix} A(\delta) & 0 & B_p(\delta) \\ 0 & 0 & 0 \\ C_p(\delta) & 0 & D(\delta) \end{bmatrix} \quad , \quad P = \begin{bmatrix} 0 & 0 & X_{cl} & 0 \\ 0 & -\gamma I & 0 & 0 \\ X_{cl} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\gamma} I \end{bmatrix}$$

Now we are only left with showing that $X_{cl} \succ 0$ is equivalent to (9a) with the particular choice of X_{cl}. This is a standard result and is done by using a congruent transformation.

$$X_{cl} = \begin{bmatrix} X & U \\ U^T & \hat{X} \end{bmatrix} \succ 0 \quad , \quad X_{cl}^{-1} = \begin{bmatrix} Y & V \\ V^T & \hat{Y} \end{bmatrix}$$

$$\Updownarrow$$

$$\begin{bmatrix} Y & I \\ V^T & 0 \end{bmatrix}^T \begin{bmatrix} X & U \\ U^T & \hat{X} \end{bmatrix} \begin{bmatrix} Y & I \\ V^T & 0 \end{bmatrix} = \begin{bmatrix} I & 0 \\ X & U \end{bmatrix} \begin{bmatrix} Y & I \\ V^T & 0 \end{bmatrix} = \begin{bmatrix} Y & I \\ I & X \end{bmatrix} \succ 0$$

Also because the controller construction will depend upon X_{cl}, it is required to show how X_{cl} can be reconstructed from X and Y. In [63] it is suggested to calculate it as follows

$$X_{cl} = \begin{bmatrix} Y & V \\ I & 0 \end{bmatrix}^{-1} \begin{bmatrix} I & 0 \\ X & U \end{bmatrix} \quad , \quad UV^T = I - XY \tag{14}$$

7. Numerical conditioning of LPV controller design

To summarise we have seen that the LPV controller design problem can be solved by determining X and Y satisfying (9) for which γ is minimised. Then X_{cl} can be constructed using (14) and the controller matrices in K can then be determined by algorithm 12 with variables as in (13). In practice the solution is unfortunately not that simple because numerical issues might make the controller construction impossible if no extra measures are taken. This section will present methods for conditioning the controller synthesis and construction to make it possible to design LPV controllers for practical applications.

7.1. Choice of realisation

In the synthesis problem for the control of wind turbines with performance specification as in Section 5 there are 183 variables. This number of variables is rather small for the size of the open loop model, because the performance channels were chosen in a way to give low order weights. Still when seen from the point of view of semi-definite programming, the number of variables is high which means that the sensitivity to some of the variables can be poor. Because of this, the optimisation problem might in numerical practice be non-convex and since the optimisation assumes convexity the algorithm

might lead to only a local minimum in γ. From a number of linear control methods based on optimisation algorithms it is well-known that the choice of realisation has a crucial significance on the achieved performance level.

It is important to pay attention to how the synthesis problem responds to different realisations and it has therefore been chosen to investigate two different methods for obtaining a realisation that is appropriate for LPV controller design. The first method is a simulation based method in which the realisation is trimmed so that the inputs, outputs, and state variables all have a standard deviation close to unity. This is done by applying a wind speed input and associated references to a closed loop simulation with a PID controller as presented in Section 3. The alternative method is based on Gramian-based balancing. A representative operating condition is determined for which an LTI Gramian-based balancing is done along standard lines. Then the transformation used to balance this realisation is used at the other operating conditions to keep the operating conditions in the same coordinate system. For these two methods used to form the realisation the first method showed superior performance with the Gramian-based method having a performance level ranging from 333 times worse to the optimisation problem not even being feasible – depending on the choice of equilibrium used as the basis for the state transformation.

7.2. Inversion of $\Omega^T \Pi \Omega$

It can be shown that (11a) is equivalent to requiring $\Omega^T \Pi \Omega \prec 0$. This means that $\Omega^T \Pi \Omega$ is always non-singular, but in practice it might become very ill-conditioned which can be seen from the the particular setup for control of wind turbines. In this application the primal LMI has a condition number in the order of 10^{10} and it is concluded that the inversion of a matrix this ill-conditioned might affect the numerical conditioning of the algorithm. In [64] a remedy is suggested by an appropriate scaling of Ψ:

Choose $\tilde{\Psi}$ such that $R\tilde{\Psi} = [\tilde{R}_1 \; 0]$ as in Algorithm 12, e.g. by a singular value decomposition of R. Then let

$$J = \begin{bmatrix} J_{11} & J_{12} \\ J_{21} & J_{22} \end{bmatrix} = \tilde{\Psi}^T \begin{bmatrix} I \\ M \end{bmatrix}^T P \begin{bmatrix} I \\ M \end{bmatrix} \tilde{\Psi} \quad , \quad \Psi = \tilde{\Psi} \begin{bmatrix} I & 0 \\ 0 & Q \end{bmatrix} \quad , \quad Q^T Q = -J_{22}^{-1}$$

We can observe that Ψ satisfies the condition $R\Psi = [R_1 \; 0]$ but is now scaled to give better conditioning of $\Omega^T \Pi \Omega$ as can be seen from the following observation.

$$\begin{aligned} \Omega^T \Pi \Omega &= \begin{bmatrix} 0 \\ I \\ \Phi^T M \tilde{\Psi} \begin{bmatrix} 0 \\ Q \end{bmatrix} \end{bmatrix}^T \Pi \begin{bmatrix} 0 \\ I \\ \Phi^T M \tilde{\Psi} \begin{bmatrix} 0 \\ Q \end{bmatrix} \end{bmatrix} \\ &= \begin{bmatrix} \tilde{\Psi} \begin{bmatrix} 0 \\ Q \end{bmatrix} \\ M \tilde{\Psi} \begin{bmatrix} 0 \\ Q \end{bmatrix} \end{bmatrix}^T P \begin{bmatrix} \tilde{\Psi} \begin{bmatrix} 0 \\ Q \end{bmatrix} \\ M \tilde{\Psi} \begin{bmatrix} 0 \\ Q \end{bmatrix} \end{bmatrix} = \begin{bmatrix} 0 \\ Q \end{bmatrix}^T J_{22} \begin{bmatrix} 0 \\ Q \end{bmatrix} = -I \end{aligned}$$

We are of course still left with finding a suitable Q which is not easier from a numerical point of view, but the point is that even with an approximate solution $\Omega^T \Pi \Omega$ is made better conditioned.

7.3. Conditioning of variables

The main matrix inequality used to determine the controller variables is (12) which contains a number of products between the matrices, Υ, Π, and Ω. If one of more of these variables are ill-conditioned, numerical errors might make small negative eigenvalues of (12) shift to positive eigenvalues, which in the end means that it is impossible to determine a Θ_1 and Θ_2 of appropriate dimension that renders the matrix inequality satisfied.

The conditioning of Υ and Ω is giving mainly by the norm of M and Ψ, because they are bounded from below by 1. If a proper realisation is chosen (e.g. according to the discussion above and by scaling of the inputs and outputs), the norm of M will not have any significant impact on numerical stability of the algorithm. The norm of Ψ will be given mainly by the square root of J_{22} which is expected to have a norm in the same scale as P. This means that with a proper choice of realisation and scaling of inputs and outputs, the numerical issues in controller construction with the proposed algorithm is determined by the conditioning of Π and thereby P.

For reasonable choices of performance level (γ close to unity), the conditioning of P is given by the conditioning of X_{cl}. X_{cl} is unfortunately often close to singular when the optimisation problem approaches optimum. If we consider (10) in the special case for controller construction with symmetric $X_{cl} = X_1^{-1}X_2$, we obtain (15).

$$\left[\begin{array}{c} I \\ \hline L^TKR+M \end{array}\right]^T \left[\begin{array}{cc|cc} 0 & 0 & X_1^{-1}X_2 & 0 \\ 0 & -\gamma I & 0 & 0 \\ \hline X_2^TX_1^{-T} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\gamma}I \end{array}\right] \left[\begin{array}{c} I \\ \hline L^TKR+M \end{array}\right] \prec 0 \tag{15}$$

Then we can equivalently reformulate the LMI as

$$\left[\begin{array}{c} \star \\ \hline \star \end{array}\right]^T \left[\begin{array}{cc|cc} 0 & 0 & X_1^{-1} & 0 \\ 0 & -\gamma I & 0 & 0 \\ \hline X_1^{-T} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\gamma}I \end{array}\right] \left[\begin{array}{c} I \\ \hline \begin{bmatrix} X_2 & 0 \\ 0 & I \end{bmatrix}\left(L^TKR+M\right) \end{array}\right] \prec 0 \tag{16}$$

or equivalently by using that X_{cl} is symmetric

$$\left[\begin{array}{c} \star \\ \hline \star \end{array}\right]^T \left[\begin{array}{cc|cc} 0 & 0 & X_2^T & 0 \\ 0 & -\gamma I & 0 & 0 \\ \hline X_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\gamma}I \end{array}\right] \left[\begin{array}{c} I \\ \hline \begin{bmatrix} X_1^{-T} & 0 \\ 0 & I \end{bmatrix}\left(L^TKR+M\right) \end{array}\right] \prec 0 \tag{17}$$

Also by using a congruent transformation with $\begin{bmatrix} X_1 & 0 \\ 0 & I \end{bmatrix}$ or $\begin{bmatrix} X_2^{-1} & 0 \\ 0 & I \end{bmatrix}$ the LMI can equivalently be formulated as

$$\left[\begin{array}{c} \star \\ \hline \star \end{array}\right]^T \left[\begin{array}{cc|cc} 0 & 0 & X_2^T & 0 \\ 0 & -\gamma I & 0 & 0 \\ \hline X_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\gamma}I \end{array}\right] \left[\begin{array}{c} I \\ \hline \left(L^TKR+M\right)\begin{bmatrix} X_1^T & 0 \\ 0 & I \end{bmatrix} \end{array}\right] \prec 0 \tag{18}$$

or

$$\left[\begin{array}{c}\star\\\hline\star\end{array}\right]^T \left[\begin{array}{cc|cc} 0 & 0 & X_1^{-1} & 0 \\ 0 & -\gamma I & 0 & 0 \\ \hline X_1^{-T} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\gamma} I \end{array}\right] \left[\begin{array}{c} I \\ \hline \left(L^T K R + M\right) \begin{bmatrix} X_2^{-1} & 0 \\ 0 & I \end{bmatrix} \end{array}\right] \prec 0 \qquad (19)$$

For the choice of X_1 and X_2 consider again (14). In this formulation, X_1 and X_2 may be very differently conditioned, e.g. because of different conditioning of X, Y, and $I - XY$. In order to make the algorithm better suited for numerical computations it is suggested to use that X and Y are symmetric and positive definite. This means that they can be expressed as $X = M^T M$ and $Y = N^T N$ by for example a Cholesky factorisation. Then the following rearrangement of (14) can be made

$$\begin{aligned} X_{cl} &= \begin{bmatrix} N^T N & V \\ I & 0 \end{bmatrix}^{-1} \begin{bmatrix} I & 0 \\ M^T M & U \end{bmatrix} = \begin{bmatrix} N & N^{-T} V \\ I & 0 \end{bmatrix}^{-1} \begin{bmatrix} N^{-T} & 0 \\ 0 & M^T \end{bmatrix} \begin{bmatrix} I & 0 \\ M & M^{-T} U \end{bmatrix} \\ &= \begin{bmatrix} N & N^{-T} V \\ M^{-T} & 0 \end{bmatrix}^{-1} \begin{bmatrix} N^{-T} & 0 \\ M & M^{-T} U \end{bmatrix} \end{aligned} \qquad (20)$$

If we denote $S = M^{-T} U$ and $T = N^{-T} V$ we can exploit the explicit expression for U and V in the following way

$$S T^T = M^{-T} U V^T N^{-1} = M^{-T} (I - XY) N^{-1} = M^{-T} N^{-1} - M N^T \qquad (21)$$

This means that we have three different ways of calculating X_1 and X_2

1. Original method from [63]

$$X_1 = \begin{bmatrix} Y & V \\ I & 0 \end{bmatrix} \quad , \quad X_2 = \begin{bmatrix} I & 0 \\ X & U \end{bmatrix} \quad , \quad U V^T = I - XY \qquad (22)$$

2. Use of $X = M^T M$ and $Y = N^T N$

$$X_1 = \begin{bmatrix} N & N^{-T} V \\ M^{-T} & 0 \end{bmatrix} \quad , \quad X_2 = \begin{bmatrix} N^{-T} & 0 \\ M & M^{-T} U \end{bmatrix} \qquad (23)$$

3. Exploit structure of $M^{-T} U V N^{-1}$

$$X_1 = \begin{bmatrix} N & T \\ M^{-T} & 0 \end{bmatrix} \quad , \quad X_2 = \begin{bmatrix} N^{-T} & 0 \\ M & S \end{bmatrix} \quad , \quad S T^T = M^{-T} N^{-1} - M N^T \qquad (24)$$

For the application, these different methods have been investigated together with the different rearrangements of the LMI given in (16-19). A summary of the results for the construction of X_1 and X_2 is given in Table 1 from which it can be seen that the original method in (22) results in a big difference in conditioning of X_1 and X_2. A significant improvement can be seen in using (23) in which the conditioning of X_1 and X_2 is equally distributed and with the change of variables in (24), the conditioning of all three variables is improved.

Table 1: Conditioning of X_1, X_2, and X_{cl} using different representations.

method from eq.	cond(X_1)	cond(X_2)	cond(X_{cl})
(22)	$3 \cdot 10^5$	$6 \cdot 10^{14}$	$6 \cdot 10^{17}$
(23)	$8 \cdot 10^8$	$8 \cdot 10^8$	$6 \cdot 10^{17}$
(24)	$1 \cdot 10^7$	$1 \cdot 10^7$	$2 \cdot 10^{14}$

From Table 1 it can be seen that there is potentially a large advantage in using X_1 and X_2 instead of X_{cl} in the construction QMI – especially when using (23) and (24). The three different methods for constructing X_1 and X_2 have been investigated in the context of constructing the controller variables using the four variants of the construction QMI (16-19). For all four QMIs there is a significant improvement in using (23) and (24) over (22), whereas there is no noticeable difference between using (23) and (24). It is expected that even though there was no significant difference between the two latter methods, (24) might show numerical performance improvements for other applications.

Regarding the four variants of the QMI it is concluded that all four methods show a significant improvement over using X_{cl} directly, but that (16) and (17) show a significant improvement over using (18) and (19). It is concluded that the reason for (18) and (19) having worse performance is that by performing a congruent transformation of the QMI with X_1 or X_2^{-1} the QMI is essentially changed. This can result in the QMI being worse conditioned and potentially indefinite which mean that the construction cannot be solved.

7.4. Bounding synthesis LMIs and variables

The remedies presented in Section 7.1 through 7.3 involve modifications to the construction algorithm to render the controller construction possible from a numerical points of view. In practical applications the controller construction might still fail because of numerical issues.

For a typical application, X and Y will become large when reaching optimum and $I - XY$ will be close to singular. This means that the construction of X_1 and X_2 of full rank is difficult because U or V (or equivalently S and T) will be ill-conditioned. In the particular application this is also the case with $I - XY$ having a condition number in the order of 10^{13}. To handle this issue a slack variable β has been included in the coupling condition (9a) as in (25) to separate the eigenvalues of X from the eigenvalues of Y^{-1}. With a Schur complement of (25) and a reordering of the terms we can see that $I - XY \prec -(\beta^2 - 1)I$ which means that by increasing β we can make $I - XY$ better conditioned.

$$\begin{bmatrix} Y & \beta I \\ \beta I & X \end{bmatrix} \succ 0 \tag{25}$$

As mentioned, X and Y might also become very large in norm which increases the norm of X_1 and X_2 and in many cases making them worse conditioned. To avoid this an upper bound is included for the two variables.

Another issue is that Θ_1 in (12) can be singular and therefore not invertible. Because the inequalities are strict it is suggested to perturb Θ_1 to make it non-singular without violating the matrix inequality. In practice it might not be possible to perturb Θ_1 enough to make it invertible from a numerical point of view. In this case it can be necessary to

modify the constraints in the original LMI (10) in order to make it possible to construct, K, i.e. it might be necessary to change the QMI which essentially is done by bounding critical variables or LMIs.

It should be noted that these introduced coefficients and bounds comes with the cost that the synthesis problem in Theorem 10 most likely will not satisfy the performance criterion, γ, with the introduced bounds. This means that it might be necessary to go for a controller satisfying a slightly poorer performance level. For the particular application this means a decrease in performance from $\gamma = 1$ to $\gamma = 1.33$.

7.5. Design algorithm

In summary we can design a controller to satisfy the performance criterion in Definition 9 by following Algorithm 13.

Algorithm 13

Step 1 Balance the open loop system, e.g. by making the standard deviation of all states have size close to unity. Also scale inputs and outputs to have sizes close to unity.

Step 2 Determine X and Y by solving the set of LMIs in Theorem 10.

Step 3 Determine a representation for X_1 and X_2 according to (22-24) and choose one of the reformulations of the QMI for construction (16-19)

Step 4 Calculate the controller, K, from Algorithm 12 with the suggested modifications.

Step 5 If controller construction fails, bound critical variables/LMIs and reiterate from Step 2.

8. Simulation results

The proposed design algorithm has been applied to the control of wind turbines. The gain and time constant in the performance weights have been chosen by an iterative procedure by first designing an LTI controller at the operating points to satisfy the desired performance specifications after which the LPV controller is designed. The performance for especially the fatigue damage is difficult to evaluate directly from the energy gain as discussed in [45]. The damage rate is instead evaluated from rain-flow count which is a simulation based method. Also the requirement of limiting the maximum generator speed error is better evaluated by simulation studies because it will depend on the expected spectrum and amplitude of the effective wind speed.

In the design procedure it was experienced that it was necessary to exaggerate the gain of the performance weights to get appropriate performance from a simulations point of view. Especially the weight on pitch in low wind speeds, tracking error of generator speed in mid wind speeds and generator torque in high wind speeds needed to be modified significantly from the LTI design to the LPV design to get similar controllers. Otherwise the LPV controller in low wind speed would resemble the high wind speed controller too much. This indicates that the method of allowing arbitrarily fast parameter variations is restrictive for the particular application. In the following it will on the other hand be shown that the LPV controller designed using the exaggerated performance weights show satisfactory performance from simulations point of view, and the LPV controller design is therefore concluded to be successful.

The resulting LPV controller has been implemented together with a nonlinear model of the wind turbine. It has then been simulated with a stochastic wind input with turbulence according to the IEC 1A standard [65]. The simulation results have been compared to a controller similar to a commercial controller, designed using classical techniques, i.e. PID tracking control of generator speed reference with control signal as respectively electrical power and pitch angle for the two operational modes. This PID controller is then combined with feed-forward terms and feedback loops to mitigate drive train oscillations, tower fatigue, etc.

A simulation result for partial load operation is shown in Figure 6 from which it can be seen that the LPV controller follows the reference trajectory very well in order to optimise electric power production. Special attention should be made to the two bottom graphs on the figure displaying pitch angle and active power from which it can be seen that the power production follows the reference trajectory well without much pitch activity. In Table 2 a comparison of some of the important criteria for partial load is given. Most importantly it can be seen that the pitch activity is reduced by 43% without reducing power production. Also the drive train loads are reduced by 20%. The downside is that the tower damage is increased by 13%, but this is not critical because in the low wind speeds, the absolute level in tower damage is low during low wind speeds.

Table 2: Comparison of an LPV controller and a classical controller at low wind speeds. Damage is normalised according to level of classical controller.

	Tower damage	Drive train damage	Pitch travel	avg. Power
LPV	1.13	0.80	$13 \cdot 10^9$ deg	918 kW
Classic	1.00	1.00	$23 \cdot 10^9$ deg	906 kW

In Figure 7 another simulation result is given for partial load during higher wind speeds where the generator speed reference is saturated from above. In this figure it should be noted that the generator speed variations are bounded by plus/minus 100 rpm without affecting the power production significantly. A comparison to a simulation with the classical controller is shown in Table 3 from which the most important observation is that tower loads are reduced significantly (almost halved) without increasing the pitch activity or variation in generator speed or decreasing the average power production. Further the drive train loads are reduced by 10%.

Table 3: Comparison of an LPV controller and a classical controller at medium wind speeds. Damage is normalised according to level of classical controller.

	Tower dam.	Drive train dam.	Pitch travel	Speed peak-peak	avg. power
LPV	0.52	0.90	$288 \cdot 10^9$ deg	218 rpm	2419 kW
Classic	1.00	1.00	$310 \cdot 10^9$ deg	221 rpm	2416 kW

In full load operation improvements can be observed when comparing to the classical controller. A result from a simulation in full load is given in Figure 8 from which it can be seen that the reference trajectory is tracked well with small power fluctuations and the generator speed bounded by less than plus/minus 100 rpm. From a comparison to the classical controller given in Table 4 it can be observed that the tower loads, drive

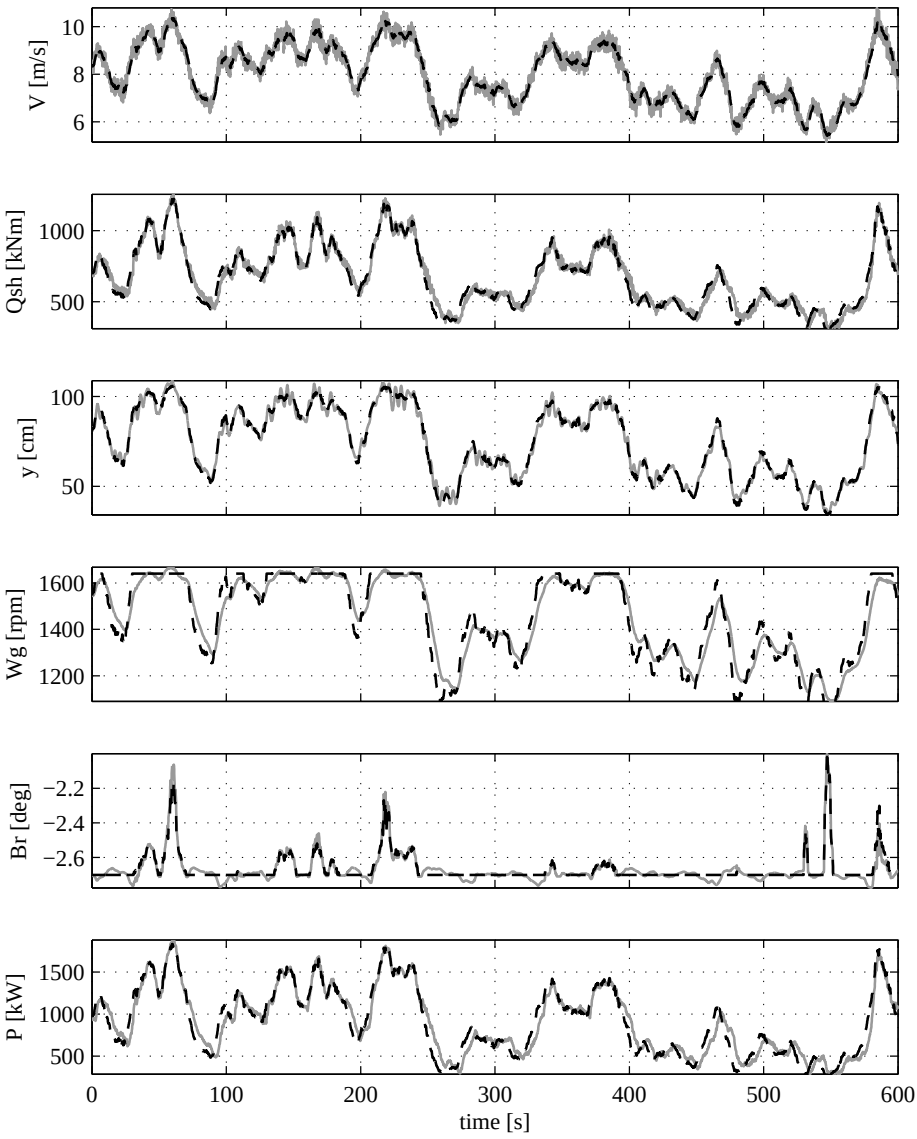


Figure 6: Simulation results for low wind speeds. Dashed black lines: reference trajectory from estimated wind speed. Full grey lines: Response with LPV controller. Good tracking performance can be observed for power and generator speed together with a low pitch activity.

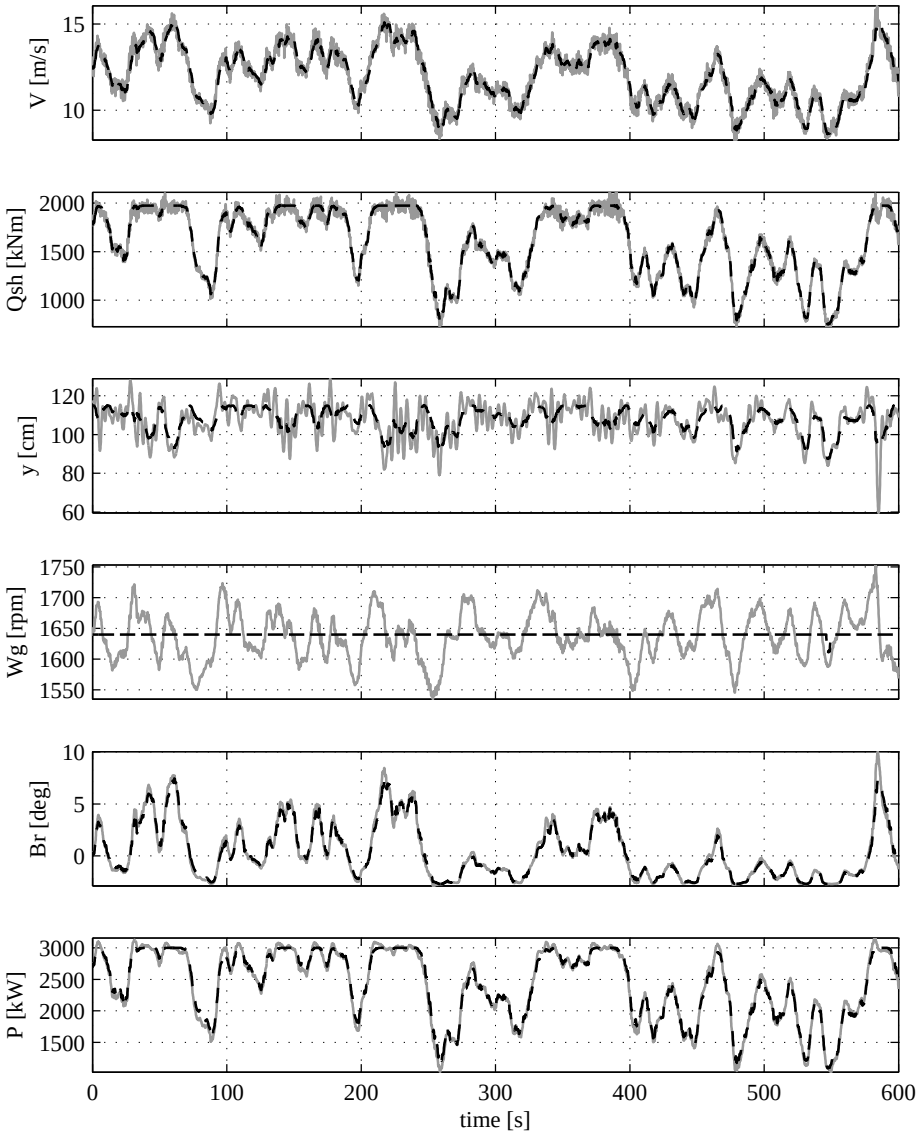


Figure 7: Simulation results for medium wind speeds. Dashed black lines: reference trajectory from estimated wind speed. Full grey lines: Response with LPV controller. Good tracking performance of power together with limitation of generator speed within nominal speed ± 100 rpm.

train loads and variation in generator speed are all decreased when comparing to the classical controller. Further the standard deviation in tracking error of power reference is reduced, but the pitch activity is increased by 8%.

Table 4: Comparison of an LPV controller and a classical controller at high wind speeds. Damage is normalised according to level of classical controller.

	Tower dam.	Drive train dam.	Pitch travel	Speed peak-peak	std. Power
LPV	0.81	0.57	$603 \cdot 10^9$ deg	188 rpm	17.3 MW
Classic	1.00	1.00	$558 \cdot 10^9$ deg	293 rpm	23.2 MW

Finally the scheduling in the LPV controller is tested by performing a simulation with a ramp like wind speed to cover the transition between the operating conditions. Such a simulation is given in Figure 9 from which it can be seen that the controller provides a smooth transfer between all three operating modes: partial load control with variable generator speed reference, partial load control with fixed generator speed reference, and full load control.

To summarise the three scenarios it can be concluded that the pitch activity is decreased significantly during low wind speeds and tower and drive train loads are reduced significantly in higher wind speeds. Also the variations in generator speed are reduced in higher wind speeds – especially in full load operation with a reduction of 36%.

The performance increase comes with the cost of higher tower loads during low wind speeds (deemed insignificant) and slightly increased pitch activity in full load operation. From a design of LTI controllers at each of the three regions it can be observed that these two issues can be handled better on a local scale. This indicates that the assumption of arbitrary fast parameter variations is restrictive and that a better performance can be obtained by relaxing this constraint to rate bounded parameter variations. This might, however, complicate the controller construction further because X_{cl} then will be parameter dependent and the conditioning might vary significantly over wind speed. Further there is no general method for identifying a basis function for the dependency of X_{cl} on the parameter.

9. Conclusions/discussion

This paper has presented a systematic method for designing a single control law to cover both partial load operation and full load operation. The proposed controller is based on the LPV design method which can be interpreted as a gain scheduling that provides a smooth transition between different LTI controllers for a target trajectory to take into account model nonlinearities and different design requirements along the trajectory.

The LPV design method suffers from numerical issues that makes computation of the controller difficult for medium to large scale systems. To make the controller design with general parameter dependency possible for the specific application, the paper has presented and discussed several issues related to the numerical computation of LPV controllers. The impact of a different representation of the design variable, X_{cl}, was investigated using three different methods and it was concluded that an equal spread in conditioning for the rational representation $X_{cl} = X_1^{-1} X_2$ is important for numerical stability in the algorithm.

The proposed method for obtaining a numerically stable design algorithm has been used for the design of an LPV controller for control of wind turbines in both partial

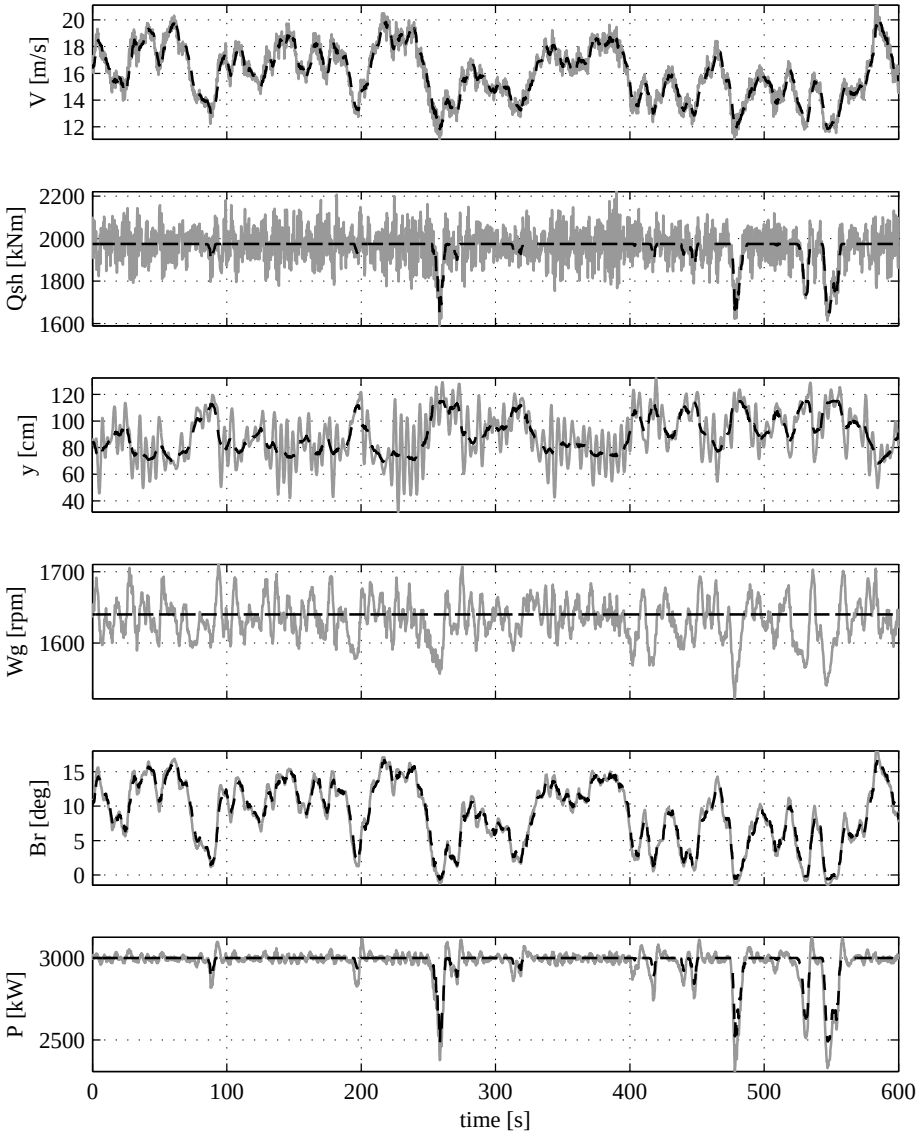


Figure 8: Simulation results for high wind speeds. Dashed black lines: reference trajectory from estimated wind speed. Full grey lines: Response with LPV controller. Small power fluctuations can be observed together with limitation of generator speed within nominal speed ± 100 rpm. Power decreases at e.g. 260s are caused by the wind speed dropping below approx. 13 m/s.

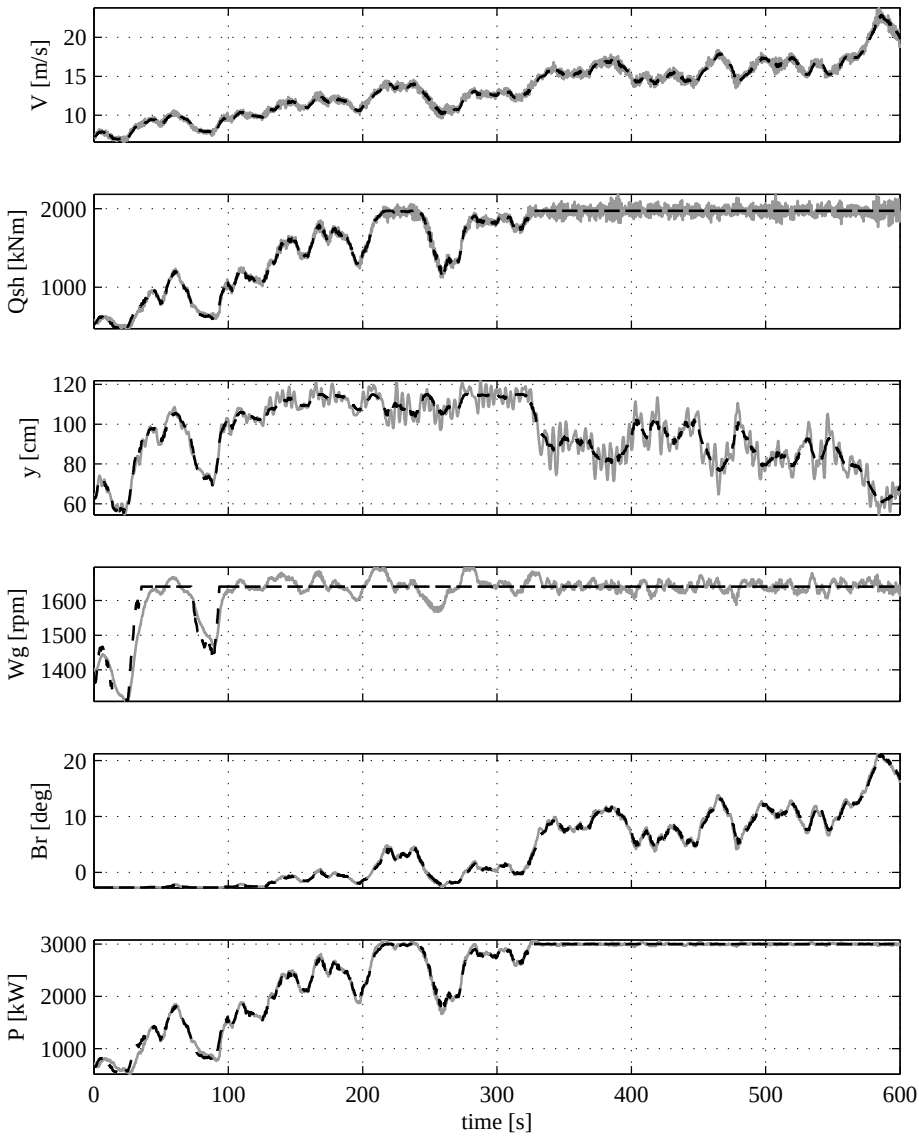


Figure 9: Simulation results for whole wind speed range. Dashed black lines: reference trajectory from estimated wind speed. Full grey lines: Response with LPV controller. A smooth transition between partial load operation and full load operation is observed.

load and full load operation. Using simulation studies, the proposed controller has been compared to a controller designed using classical techniques and it has been concluded that the LPV controller achieves significantly better performance. Most importantly, a decrease in pitch activity is observed for low wind speeds and tower and drive train loads are reduced in higher wind speeds. This performance increase is obtained without affecting the produced power, power fluctuations, and generator speed variations. However, the tower loads are observed to increase during low wind speeds and pitch activity is increased slightly during high wind speeds. Overall it is still concluded that these two cases of decrease in performance level do not have a significant influence when comparing to the above mentioned increases and the design is therefore concluded to be successful.

The proposed controller has not yet been implemented on a real wind turbine. Before this can be done it should be investigated how the control law affects the structural components not included in the design model, e.g. blade dynamics and tower sideways movement. Also it should be investigated how the control law behaves in the case of rapid wind speed variations – especially in the context of the interconnection between the feed forward term and the LPV controller.

Finally it has been experienced that there is a large difference between the combination of weights that are appropriate for designing local $\mathcal{H}_\infty$ controllers and the weights necessary for appropriate simulation results in the LPV framework. For the weights for the LPV control it was necessary to exaggerate the weights to get the desired performance from a simulation point of view. This indicates that the assumption of arbitrarily fast parameter variations is conservative and it is therefore suggested to do similar investigations for rate-bounded parameter variations. The design algorithm in this case is in theory very similar, but the numerics are expected to be more difficult to handle, because conditioning of the design variables can vary more over the operating trajectory.

As a concluding remark it should be noted that model uncertainty is not handled directly in the design formulation, but the performance channels considering tower and drive train oscillations can be considered as a detuning of the tracking controller at the two respective eigen-frequencies. Also the channels from disturbance (wind speed) to control signals can be considered similar to the control sensitivity usually used in robust controller techniques. Robustness towards parametric uncertainty can be covered fairly well by sampling the parameter space.

ACKNOWLEDGEMENTS

The authors would like to thank Professor C. Scherer from Delft University of Technology, The Netherlands, for helpful discussions about the numerics related to controller construction for LPV controllers.

REFERENCES

[1] P Gardner, A Garrad, P Jamieson, H Snodin, G Nichols, and A Tindal. *Wind energy – the facts – volume 1*. Technical report, Garrad Hassan for European Wind Energy Association, 2002.

[2] WE Leithead and B Connor. Control of variable speed wind turbines: design task. *Int. J. of Control*, **73**(13):1189–1212, 2000. `http://dx.doi.org/10.1080/002071700417849`.

[3] MM Hand and MJ Balas. *Systematic controller design methodology for variable-speed wind turbines.* Technical report, NREL, 2002.

[4] M Rasila. *Torque- and speed control of a pitch regulated wind turbine.* Technical report, Chalmers University of Technology, 2003.

[5] TG van Engelen. Control design based on aero-hydro-servo-elastic linear models from TURBO. In *Adaptive Systems for Signal Processing, Communications, and Control Symposium*, 2007.

[6] T Matsuzaka and K Tuchiya. Power fluctuation stabilization of a wind generator by using feedforward control. In *European Wind Energy Conference*, pages 898–901, 1996.

[7] EL van der Hooft and TG van Engelen. *Feed forward control of estimated wind speed.* Technical report, ECN, 2003.

[8] H Vihriälä, P Ridanpää, R Perälä, and L Söderlund. Control of a variable speed wind turbine with feedforward of aerodynamic torque. In *European Wind Energy Conference*, pages 881–884, 1999.

[9] YD Song. Control of wind turbines using memory-based method. In *American Control Conference*, pages 1715–1719, 1998. http://dx.doi.org/10.1109/ACC.1998.707299.

[10] YD Song and B Dhinakaran. Nonlinear variable speed control of wind turbines. In *Conf. on Control Applications*, pages 814–819, 1999. http://dx.doi.org/10.1109/CCA.1999.8077710.

[11] T Thiringer and A Petersson. *Control of a variable-speed pitch-regulated wind turbine.* Technical report, Chalmers University of Technology, 2005.

[12] H Vihriälä. *Control of Variable Speed Wind Turbine*. PhD thesis, Tampere University of Technology, 1998.

[13] AD Wright and MJ Balas. Design of state-space-based control algorithms for wind turbine speed regulation. In *ASME Wind Energy Symposium*, 2002.

[14] EA Bossanyi. Wind turbine control for load reduction. *Wind Energy*, **6**:229–244, 2003. http://dx.doi.org/10.1002/we.95.

[15] EA Bossanyi. Individual blade pitch control for load reduction. *Wind Energy*, **6**:119–128, 2003. http://dx.doi.org/10.1002/we.76.

[16] EA Bossanyi. Further load reduction with individual pitch control. *Wind Energy*, **8**:481–485, 2005. http://dx.doi.org/10.1002/we.166.

[17] M Geyler and P Caselitz. Individual blade pitch control design for load reduction on large wind turbines. In *European Wind Energy Conference*, 2007.

[18] TJ Larsen, HA Madsen, and K Thomsen. Active load reduction using individual pitch, based on local blade flow measurements. *Wind Energy*, **8**:67–80, 2005. http://dx.doi.org/10.1002/we.141.

[19] D Trudnowski and D LeMieux. Independent pitch control using rotor position feedback for wind-shear and gravity fatigue reduction in a wind turbine. In *American Control Conference*, pages 4335–4340, 2002. http://dx.doi.org/10.1109/ACC.2002.1025328.

[20] AD Wright and MJ Balas. Design of controls to attenuate loads in the controls advanced research turbine. In *ASME Wind Energy Symposium*, 2004.

[21] A Dixit and S Suryanarayanan. Towards pitch-scheduled drive train damping in variable-speed, horizontal-axis large wind turbines. In *Conf. on Decision and Control*, pages 1295–1300, 2005.

[22] T Fischer, M Kühn, and P Passon. Load mitigation of aerodynamically and hydrodynamically induced loads of offshore wind turbines. In *European Wind Energy Conference*, 2007.

[23] WE Leithead and SD Ruiz. Controller design for the cancellation of the tower fore-aft mode in a wind turbine. In *Conf. on Decision and Control*, pages 1276–1281, 2005.

[24] WE Leithead and B Connor. Control of a variable speed wind turbine with induction generator. In *Int. Conf. on Control*, pages 1215–1220, 1994.

[25] B Connor, WE Leithead, and MJ Grimble. LQG control of a constant speed horizontal axis wind turbine. In *Conf. on Control Applications*, pages 251–252, 1994. http://dx.doi.org/10.1109/CCA.1994.381190.

[26] MJ Grimble. Two and a half degree of freedom LQG controller and application to wind turbines. *Trans. on Automatic Control*, **39**(1):122–127, 1992. http://dx.doi.org/10.1109/9.273347.

[27] MM Hand and MJ Balas. Load mitigation control design for a wind turbine operating in the path of vortices. In *The Science of Making Torque From Wind*, 2004.

[28] N Kodama, T Matsuzaka, and N Inomata. Power variation control of a wind turbine generator using probalistic optimal control, including feed-forward control from wind speed. *Wind Engineering*, **24**(1):13–23, 2000.

[29] T Matsuzaka, N Kodama, and N Inomata. Control strategy of a wind generator to reduce power variations using probalistic optimal control, combining feed forward control from measured wind speed with feed back control. In *European Wind Energy Conference*, pages 1178–1181, 2001.

[30] I Munteanu, NA Cutululis, AI Bratcu, and E Ceanga. Optimization of variable speed wind power systems based on a LQG approach. *Control Engineering Practice*, **13**(7):903–912, 2004. http://dx.doi.org/10.1016/j.conengprac.2004.10.013.

[31] P Novak, T Ekelund, I Jovik, and B Schmidtbauer. Modeling and control of variable-speed wind-turbine drive-system dynamics. *Control Systems Magazine*, **15**(4):28–38, 1995. http://dx.doi.org/10.1109/37.408463.

[32] NK Poulsen, TJ Larsen, and MH Hansen. *Comparison between a PI and LQ-regulation for a 2 MW wind turbine.* Technical report, Risø National Laboratory, 2005.

[33] KA Stol. Disturbance tracking and load control of wind turbines in variable-speed operation. In *ASME Wind Energy Symposium*, 2003.

[34] FD Bianchi, RJ Mantz, and CF Christiansen. Power regulation in pitch-controlled variable-speed WECS above rated wind speed. *Renewable Energy*, **29**:1911–1922, 2004. http://dx.doi.org/10.1016/j.renene.2004.02.009.

[35] B Connor, SN Iyer, WE Leithead, and MJ Grimble. Control of a horizontal axis wind turbine using $\mathcal{H}_\infty$ control. In *Conf. on Control Applications*, pages 117–122, 1992. http://dx.doi.org/10.1109/CCA.1992.269889.

[36] T Knudsen, P Andersen, and S Tøffner-Clausen. Comparing PI and robust pitch controllers on a 400 kW wind turbine by full scale tests. In *European Wind Energy Conference*, 1997.

[37] R Rocha and LSM Filho. A multivariable $\mathcal{H}_\infty$ control for wind energy conversion system. In *Conf. on Control Applications*, pages 206–211, 2003.

[38] R Rocha, LSM Filho, and MV Bortolus. Optimal multivariable control for wind energy conversion system - a comparison between $\mathcal{H}_2$ and $\mathcal{H}_\infty$ controllers. In *Conf. on Decision and Control*, pages 7906–7911, 2005.

[39] NA Cutululis, E Ceanga, AD Hansen, and P Sørensen. Robust multi-model control of an autonomous wind power system. *Wind Energy*, **9**:399–419, 2006. http://dx.doi.org/10.1002/we.194.

[40] M Jelavić, N Perić, I Petrović, S Car, and M Maderčić. Design of a wind turbine pitch controller for loads and fatigue reduction. In *European Wind Energy Conference*, 2007.

[41] I Kraan and PMM Bongers. Control of a wind turbine using several linear robust controllers. In *Conf. on Decision and Control*, pages 1928–1929, 1993. http://dx.doi.org/10.1109/CDC.1993.325530.

[42] DJ Leith and WE Leithead. Application of nonlinear control to a HAWT. In *Conf. on Control Applications*, pages 245–250, 1994. http://dx.doi.org/10.1109/CCA.1994.381191.

[43] DJ Leith and WE Leithead. Appropriate realization of gain-scheduled controllers with application to wind turbine regulation. *Int. J. of Control*, **65**(2):223–248, 1996. http://dx.doi.org/10.1080/00207179608921695.

[44] TG van Engelen, EL van der Hooft, and P Schaak. Development of wind turbine control algorithms for industrial use. In *European Wind Energy Conference*, 2003.

[45] F Lescher, H Camblong, O Curea, and R Briand. LPV control of wind turbines for fatigue loads reduction using intelligent micro sensors. In *American Control Conference*, pages 6061–6066, 2007.

[46] F Lescher, JY Zhao, and P Borne. Robust gain scheduling controller for pitch regulated variable speed wind turbine. *Studies in Informatics and Control*, **14**(4):299–315, 2005.

[47] RJ Mantz, FD Bianchi, and CF Christiansen. Gain scheduling control of variable-speed wind energy conversion systems using quasi-LPV models. *Control Engineering Practice*, **13**:247–255, 2005. http://dx.doi.org/10.1016/j.conengprac.2004.03.0060.

[48] GJ Balas. Linear, parameter-varying control and its applications to a turbofan engine. *Int. J. of Robust and Nonlinear Control*, **12**:763–796, 2002. http://dx.doi.org/10.1002/rnc.704.

[49] T Burton, D Sharpe, N Jenkins, and EA Bossanyi. *Wind Energy Handbook.* Wiley, 2001. http://dx.doi.org/10.1002/0470846062.

[50] KZ Østeraard, P Brath, and J Stoustrup. Estimation of effective wind speed. In *The Science of Making Torque from Wind*. EAWE, 2007.

[51] JS Shamma and M Athans. Analysis of gain scheduled control for nonlinear plants. *Trans. on Automatic Control*, **35**(8):898–907, 1990. http://dx.doi.org/10.1109/9.58498.

[52] G Becker and A Packard. Robust performance of linear parametrically varying systems using parametrically-dependent linear feedback. *Systems and Control Letters*, **23**:205–215, 1994. http://dx.doi.org/10.1016/0167-6911(94)90006-X.

[53] P Gahinet and P Apkarian. A linear matrix inequality approach to $\mathcal{H}_\infty$ control. *Int. J. of Robust and Nonlinear Control*, **4**:421–448, 1994. http://dx.doi.org/10.1002/rnc.4590040403.

[54] RE Kalman. Lyapunov functions for the problem of lur'e in automatic control. *Proc. Nat. Acad. Sci.*, **49**(2):201–205, 1963.

[55] VM Popov. Absolute stability of nonlinear systems of automatic control. *Automation and Remote Control*, **22**:857–875, 1961.

[56] VA Yakubovich. Solution of certain matrix inequalities in the stability theory of nonlinear control systems. *Soviet Math. Dokl.*, **3**:620–623, 1962.

[57] A Helmersson. IQC synthesis based on inertia constraints. In *IFAC World Congress*, 1999.

[58] CW Scherer. *Recent Advances on LMI Methods in Control*, chapter Robust Mixed Control and LPV Control with Full Block Scalings. SIAM, 2000.

[59] P Apkarian, P Gahinet, and G Becker. Self-scheduled $\mathcal{H}_\infty$ control of linear parameter-varying systems - a design example. *Automatica*, **31**(9):1251–1261, 1995. http://dx.doi.org/10.1016/0005-1098(95)00038-X.

[60] CW Scherer. LPV control and full block multipliers. *Automatica*, 37:361–375, 2001. http://dx.doi.org/10.1016/S0005-1098(00)00176-X.

[61] T Iwasaki and G Shibata. LPV system analysis via quadratic separator for uncertain implicit systems. *Trans. on Automatic Control*, **46**(8):1195–1208, 2001. 10.1109/9.940924.

[62] F Wu, XH Yang, A Packard, and G Becker. Induced $\mathcal{L}_2$-norm control for LPV system with bounded parameter variation rates. In *American Control Conference*, pages 2379–2383, 1995.

[63] P Gahinet. Explicit controller formulas for LMI-based $\mathcal{H}_\infty$ synthesis. *Automatica*, **32**(7):1007–1014, 1996. http://dx.doi.org/10.1016/0005-1098(96)00033-7.

[64] K Trangbæk. *Linear Parameter Varying Control of Induction Motors*. PhD thesis, Aalborg University, 2001.

[65] IEC. *IEC 61400-1*. Int. Standard, 2005.

Paper D

Rate Bounded LPV Control of a Wind Turbine in Full Load

Contents

Conference paper submitted to:
The 17th IFAC World Congress, July 2008.

Rate bounded linear parameter varying control of a wind turbine in full load operation

Kasper Zinck Østergaard * **Per Brath** * **Jakob Stoustrup** **

* *Turbine Control and Operation R&D, Vestas Wind Systems.*
** *Automation and Control, Department of Electronic Systems, Aalborg University*

Abstract: This paper considers the control of wind turbines using the LPV design technique. The controller design is done using a combination of the method with elimination of controller variables and the method using a congruent transformation followed by a change of variables.
An investigation is performed to understand the gap between zero rate of variation and arbitrary fast rate of variation for the selected scheduling variable. In particular it is analysed for which rate of variation, the local performance level starts to deteriorate from the performance level that can be obtained locally by LTI controllers.
A rate of variation is selected which is expected only to be exceeded outside the normal operating conditions. For this rate of variation a controller has been designed and simulations show a performance level over the operating region which is very similar to what can be obtained by LTI designs for the specific operating condition.

Keywords: gain scheduling; linear parameter varying systems; Modelling, operation and control of power systems; Output feedback control; LMIs; Industrial applications of optimal control

1. INTRODUCTION

Several gain scheduled controller design approaches have been investigated for the control of wind turbines. Most approaches either neglect the rate of variation of the scheduling parameter as in [4, 8, 15] or alternatively controllers have been designed to allow for arbitrary fast parameter variations as in [9, 11, 13]. The nominal operating condition is essentially determined by the average wind speed together with operational settings such as rating of active power and generator speed.

With the assumption of zero parameter variations it is possible to get a high level of performance locally for all operating conditions. The disadvantage is that if the assumption of very slow parameter variations is violated it is unknown how the controller will perform, potentially leading to a decrease in performance level and perhaps closed loop instability.

The other extreme, allowing for arbitrary fast parameter variations has the advantage that the performance level is guaranteed for all possible rates of variation. The disadvantage is that the assumption might impose strict requirements on the controller making the local performance poor.

This paper will deal with the controller design with the scheduling parameter limited to a rate of variation between the two extreme values to give an understanding of the gap

in performance level between slow and fast parameter variations. In [10] a controller design with rate bounded parameter variations is done for a piecewise affine model of a wind turbine using the multi-convexity property for this special case as described in [6]. In this paper an alternative approach is taken by gridding the parameter space. The advantage in this approach is that it does not suffer from the potential conservative restrictions associated with using multi-convexity. It should on the other hand be noted that by the gridding method, no guarantee is given for the parameter values in between the grid points. This is not expected to cause a problem and can be examined by testing the synthesis linear matrix inequalities (LMIs) in a denser grid.

The LPV controller will be designed to have a level of performance locally at each operating condition that is similar to what can be obtained by an LTI controller designed for the particular operating condition. At the same time the controller must maintain this level of performance even with parameter variations in a specified interval. The design method will be based on a combination of two methods in order to obtain a convex optimisation problem with low complexity and also a numerically stable algorithm for construction of the controller.

In Section 2 the considered control problem will be presented and a controller structure is selected. Then in Section 3 the LPV controller design algorithm is presented followed by a discussion of practical considerations in Section 4. In Section 5 the controller is then designed and simulation results are presented in Section 6 followed by the conclusion in Section 7.

The notation used in the paper is as follows: For real symmetric matrices, M, $M \prec 0$ is interpreted as M being negative definite, i.e. all eigenvalues are negative. In large matrix expressions the symbol $\star$ will denote terms that are induced by symmetry. Let X and Y be symmetric matrices and M and N be non-symmetric matrices then:

$$\begin{bmatrix} X + M + (\star) & \star \\ N & Y \end{bmatrix} := \begin{bmatrix} X + M + M^T & N^T \\ N & Y \end{bmatrix}$$

Also a short hand for functional dependency will be applied when necessary for notational simplicity. A function $f(a(t), b(t), \ldots)$ will be abbreviated as $f^{a,b,\cdots}$.

2. CONSIDERED CONTROL PROBLEM

The aim is to design a full load controller that limits the drive train oscillations while tracking nominal generator speed and active power. We consider a 3 MW, tree-bladed wind turbine with a rotor diameter of 90 m and a doubly-fed induction generator. The wind turbine has three pitch actuators making it possible to change the angle of attack of the blades individually, but in this paper only collective pitch is considered because the objective is regarding drive train oscillations and tracking of speed and power references. Another actuator is the generator reaction torque which can be altered by changing the current in the rotor of the generator.

In full load operation the active power should be kept close to the rated value of 3 MW with a low amount of fluctuations in order not to introduce electrical noise onto the grid. The generator speed must also be kept in the neighbourhood of the rated speed, because the the generator and converter system can overheat if the generator speed exceeds the tolerated level. Regarding oscillations, the drive train is lightly damped around 10 rad/s which means that small disturbances at the drive train eigen-frequency will lead

to large oscillations. These oscillations lead to increased fatigue damage in the drive train and in order to make the wind turbine profitable means for minimising the drive train oscillations are necessary for the control of modern wind turbines. Finally the heart of the pitch system is a hydraulic actuator that can only be used to deal with the slow disturbances caused by changes in wind speed. A high frequency component in the hydraulic actuator will result in a large wear in the mechanics making it very expensive.

It has been decided to use the generator torque to dampen the drive train oscillations and the pitch system to track the generator speed reference. The reason for this split is that the drive train oscillations occur at a relatively high frequency which will lead to a high pitch activity (and thereby wear) if it is dealt with by the pitch system. The pitch system is on the other hand necessary for controlling the kinetic energy captured by the wind turbine. It has been chosen use only pitch angle for the speed control because it is more important to limit power fluctuations than error in tracking the generator speed as long as the limits are not exceeded – and the pitch system will deal with the wind induced variations.

This split into two control algorithms calls for a two-step design algorithm. The drive train damper is designed by a classical strategy in which a band pass filter containing the drive train eigen frequency is fed back from generator speed to generator torque. The speed controller is then designed as a tracking controller with integral action. A gain-scheduled, linear parameter varying (LPV) controller is chosen for the speed controller in order to handle the nonlinear aerodynamics and further to take into account that more control effort is accepted at lower wind speeds because to tracking is harder at these frequencies. The interconnection of the wind turbine model with the controller is then as illustrated in Fig. 1 with the following signal definitions: pitch angle (β), generator speed (ω_g), rotor speed (ω_r), aerodynamic torque (Q_a), and generator torque (Q_g).

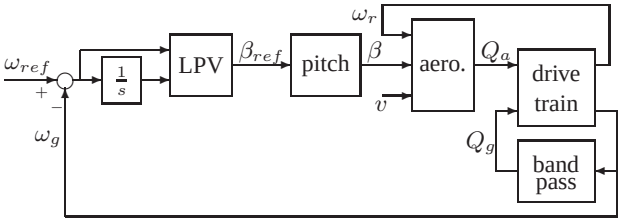


Figure 1: Block diagram of controller structure.

The drive train damper has been chosen along classical lines as a band pass filter of the generator speed fed back to the generator reaction torque as in (1) with $\Delta\omega$ a small number determining the width of the filter. The gain, K, is chosen to give a satisfactory trade-off between damping of oscillations and control effort (noise on the power production) as indicated by Fig. 2.

$$\frac{Q_g(s)}{\omega_g(s)} = \frac{K \cdot s}{(s + \omega_0 - \Delta\omega)(s + \omega_0 + \Delta\omega)} \tag{1}$$

When designing the LPV controller, the interconnection of the drive train with the damper can now be considered as a first order low pass filter from aerodynamic torque to generator speed and with the rotor speed proportional to the generator speed. The LPV

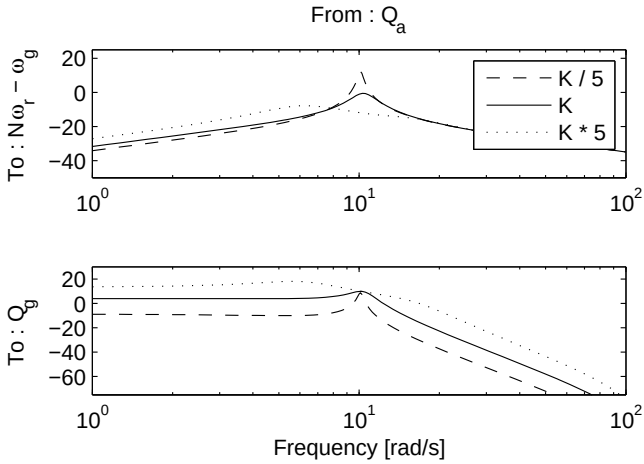


Figure 2: Magnitude plot of interconnection of drive train with drive train damper.

controller can now be designed to trade off the tracking of generator speed with control effort (wear on pitch actuator).

3. LINEAR PARAMETER VARYING CONTROL

This section will describe the approach used in this paper to design an LPV controller for the control of wind turbines. The closed loop performance level γ will be measured by the energy gain (induced $\mathcal{L}_2$ gain) from a specified performance input, $w(t)$, to a chosen performance output, $z(t)$, i.e. measured by $||z(t)||_2 < \gamma||w(t)||_2$ for all nonzero inputs with finite energy.

For notational simplicity we will describe the weighted open loop system by (2) and the objective is then to design a controller of the form (3) to satisfy an energy gain γ for the closed loop interconnection (4) with $x_{cl} = [x^T x_c^T]^T$.

$$\dot{x}(t) = a(v_s(t))x(t) + b_p(v_s(t))w(t) + b(v_s(t))u(t) \tag{2a}$$
$$z(t) = c_p(v_s(t))x(t) + d(v_s(t))w(t) + e(v_s(t))u(t) \tag{2b}$$
$$y(t) = c(v_s(t))x(t) + f(v_s(t))w(t) \tag{2c}$$

$$\dot{x}_c(t) = a_c(v_s(t), \dot{v}_s(t))x_c(t) + b_c(v_s(t), \dot{v}_s(t))y(t) \tag{3a}$$
$$u(t) = c_c(v_s(t), \dot{v}_s(t))x_c(t) + d_c(v_s(t), \dot{v}_s(t))y(t) \tag{3b}$$

$$\dot{x}_{cl}(t) = A_{cl}(v_s(t), \dot{v}_s(t))x_{cl}(t) + B_{cl}(v_s(t), \dot{v}_s(t))w(t) \tag{4a}$$
$$z(t) = C_{cl}(v_s(t), \dot{v}_s(t))x_{cl}(t) + D_{cl}(v_s(t), \dot{v}_s(t))y(t) \tag{4b}$$

From dissipativity arguments it is known that the closed loop system is exponentially stable and achieves an energy gain γ if there exist a symmetric, $X_{cl}(v_s(t))$, for which

the following two requirements hold: $X_{cl}(v_s(t))$ is positive definite for all possible parameter values, v_s, in the interval from 15 m/s to 25 m/s, and the inequality (5) is satisfied for all possible trajectories of the plant and all possible parameter values in the interval.

$$\frac{d}{dt}x_{cl}(t)^T X_{cl}(v_s(t))x_{cl}(t) + z(t)^T z(t) < \gamma^2 w(t)^T w(t) \tag{5}$$

The inequality (5) can be formulated as an LMI which means that determining the energy gain of a closed loop system can be formulated by a convex optimisation problem. In the case of controller synthesis we plug in the open loop system and controller variables in the analysis formulation, but it turns out to be nonlinear in X_{cl} and the controller variables. In [2] it was shown that the controller variables can be eliminated from the nonlinear matrix inequality. By a partitioning of X_{cl} according to (6) we can formulate the controller synthesis as determining two symmetric matrix functions $X(v_s(t))$ and $Y(v_s(t))$ such that (7) is satisfied for all parameter values in the interval of expected wind speeds and associated rates of variation.

$$X_{cl}^v = \begin{bmatrix} X^v & M^v \\ M^{vT} & \hat{X}^v \end{bmatrix} \quad , \quad X^{v-1}_{cl} = \begin{bmatrix} Y^v & N^v \\ N^{vT} & \hat{Y}^v \end{bmatrix} \tag{6}$$

$$\begin{bmatrix} Y^v & I \\ I & X^v \end{bmatrix} \succ 0 \tag{7a}$$

$$\begin{bmatrix} \star \\ \star \\ \star \end{bmatrix}^T \begin{bmatrix} \dot{X}^{v,\dot{v}} + X^v a^v + (\star) & X^v b_p^v & \star \\ \star & -\gamma I & \star \\ c_p^v & d^v & -\gamma I \end{bmatrix} \begin{bmatrix} c_\perp^v & 0 \\ f_\perp^v & 0 \\ 0 & I \end{bmatrix} \prec 0 \tag{7b}$$

$$\begin{bmatrix} \star \\ \star \\ \star \end{bmatrix}^T \begin{bmatrix} -\dot{Y}^{v,\dot{v}} + a^v Y^v + (\star) & b_p^v & \star \\ \star & -\gamma I & \star \\ c_p^v Y^v & d^v & -\gamma I \end{bmatrix} \begin{bmatrix} b_\perp^{vT} & 0 \\ e_\perp^{vT} & 0 \\ 0 & I \end{bmatrix} \prec 0 \tag{7c}$$

Alternatively by a congruent transformation similar to what is done in [14] and [3] we can get an alternative formulation for the synthesis as in (8) with the variables defined as in (9), (10), and (11). This matrix inequality is still nonlinear, however with a suitable variable substitution it can be turned into an LMI.

$$\left[\begin{array}{cc|cc} Q_{11} & \star & L_{11} & L_{12} \\ Q_{21} & Q_{22} & L_{21} & L_{22} \\ \hline \multicolumn{2}{c|}{\star} & \multicolumn{2}{c}{\Delta} \end{array}\right] \prec 0 \tag{8}$$

$$\begin{aligned} Q_{11} =& -\dot{Y}^{v,\dot{v}} + a^v Y^v + b^v {d_c}^{v,\dot{v}} c^v Y^v + \\ & + b^v {c_c}^{v,\dot{v}} N^{vT} + (\star) \end{aligned} \tag{9a}$$

$$\begin{aligned} Q_{22} =& \dot{X}^{v,\dot{v}} + X^v a^v + X^v b^v {d_c}^{v,\dot{v}} c^v + \\ & + M^v {b_c}^{v,\dot{v}} c^v + (\star) \end{aligned} \tag{9b}$$

$$\begin{aligned} Q_{12} =& \dot{X}^{v,\dot{v}} Y + \dot{M}^{v,\dot{v}} N^{vT} + X^v a^v Y^v + \\ & + X^v b^v {c_c}^{v,\dot{v}} N^{vT} + M^v {b_c}^{v,\dot{v}} c^v Y^v + \\ & + M^v {a_c}^{v,\dot{v}} N^{vT} + (a^v + b^v {d_c}^{v,\dot{v}} c^v)^T \end{aligned} \tag{9c}$$

$$L_{11} = b_p^v + b^v {d_c}^{v,\dot{v}} f^v \tag{10a}$$
$$L_{22} = (c_p^v + e^v {d_c}^{v,\dot{v}} c^v)^T \tag{10b}$$
$$L_{21} = X^v b_p^v + X^v b^v {d_c}^{v,\dot{v}} f^v + M^v {b_c}^{v,\dot{v}} f^v \tag{10c}$$
$$L_{12} = (c_p^v Y^v + e^v {d_c}^{v,\dot{v}} c^v Y^v + e^v {c_c}^{v,\dot{v}} N^{vT})^T \tag{10d}$$

$$\Delta = \begin{bmatrix} -\gamma I & (d^v + e^v {d_c}^{v,\dot{v}} f^v)^T \\ d^v + e^v {d_c}^{v,\dot{v}} f^v & -\gamma I \end{bmatrix} \tag{11}$$

In this paper we choose an alternative approach. Instead of applying a variable substitution, we assume that $X(v_s(t))$ and $Y(v_s(t))$ are known from solving (7). Then we can calculate $M(v_s(t))$ and $N(v_s(t))$ from the relation (12) which means that the matrix inequality (8) is an LMI in the variables, $a(v_s(t), \dot{v}_s(t))$, $b(v_s(t)$, $c(v_s(t))$, and $d(v_s(t))$.

$$M(v_s(t))N(v_s(t))^T = I - X(v_s(t))Y(v_s(t)) \tag{12}$$

If we assume that (7) is satisfied then we know from the elimination lemma that there is a $d_c(v_s(t))$ such that $\Delta \prec 0$. It is therefore possible to perform a Schur complement of (8) to arrive at (13). If we then assumed that we have determined a $d_c(v_s(t))$ to satisfy $\Delta < 0$ it can be observed that the upper left block of (13) is only dependent on $c_c(v_s(t))$ and the lower right block only depends upon $b_c(v_s(t))$. In [5] it is shown for LTI systems that if (7) is satisfied it is always possible to find b_c and c_c to make to diagonal blocks negative definite and the off-diagonal blocks zero by choosing a_c properly.

$$\begin{bmatrix} Q_{11} & Q_{12} \\ Q_{12}^T & Q_{22} \end{bmatrix} - \begin{bmatrix} L_{11} & L_{12} \\ L_{21} & L_{22} \end{bmatrix} \Delta^{-1} \begin{bmatrix} L_{11} & L_{12} \\ L_{21} & L_{22} \end{bmatrix}^T \prec 0 \tag{13}$$

The same procedure can essentially be applied for LPV systems as argued in [1]. This means that $b_c(v_s(t))$ and $c_c(v_s(t))$ can be determined independently to satisfy (14) and (15), and $a_c(v_s(t), \dot{v}_s(t))$ can be calculated by solving (16).

$$\begin{bmatrix} \dot{X}^{v,\dot{v}} + X^v(a^v + b^v {d_c}^{v,\dot{v}} c^v) + \star & \star & \star \\ (b_p^v + b^v {d_c}^{v,\dot{v}} f^v)^T X^v & -\gamma I & \star \\ (c_p^v + e^v {d_c}^{v,\dot{v}} c^v) & c_p^v + e^v {d_c}^{v,\dot{v}} c^v & -\gamma I \end{bmatrix} + \\ + \begin{bmatrix} M^v \\ 0 \\ 0 \end{bmatrix} {b_c}^v \begin{bmatrix} c^v & f^v & 0 \end{bmatrix} + \star \prec 0 \tag{14}$$

$$\begin{bmatrix} -\dot{Y}^{v,\dot{v}} + (a^v + b^v {d_c}^{v,\dot{v}} c^v) Y^v + \star & \star & \star \\ (b_p^v + b^v {d_c}^{v,\dot{v}} f^v)^T & -\gamma I & \star \\ (c_p^v + e^v {d_c}^{v,\dot{v}} c^v) Y^v & c_p^v + e^v {d_c}^{v,\dot{v}} c^v & -\gamma I \end{bmatrix} + \\ + \begin{bmatrix} b^v \\ e^v \\ 0 \end{bmatrix} {c_c}^v \begin{bmatrix} N^T & 0 & 0 \end{bmatrix} + \star \prec 0 \tag{15}$$

Further,

$$
\begin{aligned}
-M^v {a_c}^{v,\dot{v}} N^T = {} & \dot{X}^{v,\dot{v}} Y^v + \dot{M}^{v,\dot{v}} N^{vT} + \\
& + X^v a^v Y^v + (a^v + b^v d_c^v c^v)^T + \\
& + X^v b^v d_c^v c^v Y^v + X^v b^v {c_c}^{v,\dot{v}} N^{vT} + M^v {b_c}^{v,\dot{v}} c^v Y^v + \\
& + \begin{bmatrix} (X^v b_p^v + \tilde{b}_c^v f^v)^T \\ c_p^v + e^v d_c^v c^v \end{bmatrix}^T \Delta^{-1} \begin{bmatrix} (b_p^v + b^v d_c^v f^v)^T \\ c_p^v Y^v + e^v \tilde{c}_c^v \end{bmatrix}
\end{aligned} \tag{16}
$$

4. PRACTICAL CONSIDERATIONS

In order to design an LPV controller using the approach presented in the previous section it requires solving (7) for infinitely many combinations of parameter values and rates of variation. To handle this we first of all assume that the maximum wind speed acceleration (parameter rate of variation) is similar for all wind speeds in the operating region. This means that the number of synthesis LMIs is reduced to five for each wind speed in the operating region. Note that if this assumption is too restrictive, it can be relaxed by scheduling the worst case acceleration on wind speed.

Still the controller design requires solving infinitely many LMIs because (7) must be solved for all parameter values in the operating region. This is solved by an approximative approach in which $X(v_s(t))$ and $Y(v_s(t))$ are described by basis functions and the synthesis LMIs are then sampled at a finite number of operating conditions.

For the control of wind turbines the effective wind speed is not measurable with adequate precision and it must be estimated as discussed in the introduction. With available methods for estimating the effective wind speed it is not possible to obtain an estimate of the acceleration of the wind field with adequate precision. The controller variables must therefore be made independent on $\dot{v}_s(t)$.

First of all we can observe that the calculation of $d_c(v_s(t))$ is independent on the derivative term. Further, because we use only the bounds of $\dot{X}(v_s(t), \dot{v}_s(t))$ and $\dot{Y}(v_s(t), \dot{v}_s(t))$ in the calculation of $b_c(v_s(t))$ a $c_c(v_s(t))$, these two variables are also independent on $\dot{v}_s(t)$. This means that $a_c(v_s(t), \dot{v}_s(t))$ is the only controller variable that depends on the time derivative of the effective wind speed.

Note that the construction of $M(v_s(t))$ and $N(v_s(t))$ according to (12) can always be made so that one of the variables is independent of $v_s(t)$. This means that if we require $X(v_s(t))$ to be independent on $v_s(t)$ (i.e. $\dot{X}(v_s(t), \dot{v}_s(t) = 0$) we can make $a_c(v_s(t), \dot{v}_s(t))$ independent of $\dot{v}_s(t)$ by choosing $M(v_s(t))$ constant. Furthermore from the properties of the partitioning of $X_{cl}(v_s(t)$ we have that

$$
\dot{X}^{v,\dot{v}} Y^v + \dot{M}^{v,\dot{v}} N^{vT} = -(X^v \dot{Y}^{v,\dot{v}} + M^v \dot{N}^{v,\dot{v}T})
$$

which means that by restricting the controller design to either $X(v_s(t))$ or $Y(v_s(t))$ being constant we can make $a_c(v_s(t), \dot{v}_s(t))$ independent on $\dot{v}_s(t)$ by choosing respectively $M(v_s(t))$ and $N(v_s(t)$ to be constant.

5. LPV CONTROL OF WIND TURBINES

In this section an LPV speed controller will be designed for the high wind speed region and throughout the design it is assumed that the power and speed rating is well-known.

This means that the trajectory of equilibria for the LPV controller design can be determined uniquely from the effective wind speed which can be estimated as in [12]. We will consider a controller design for the interval of wind speeds from 15 m/s to 25 m/s.

The open loop for the controller design is determined by interconnecting the components in Fig. 1 in which the pitch system is described as a second order model from pitch reference to pitch angle and the interconnection of drive train with the damper is approximated by a first order model. The aerodynamics are assumed static nonlinear functions that are linearised along the trajectory of equilibria.

For the controller design we wish to reduce the sensitivity on wind speed variations in the tracking of generator speed while keeping the pitch activity low. A performance function in terms of the energy gain can then be setup by choosing

$$\tilde{w}(t) = v_s(t) \text{ and } \tilde{z}(t) = \begin{bmatrix} \int_0^t \omega_{ref}(\tau) - \omega_g(\tau) d\tau \\ \beta_{ref}(t) \end{bmatrix}.$$

The performance inputs and outputs are then obtained by scaling $\tilde{w}(t)$ and $\tilde{z}(t)$ appropriately over frequency to to give a reasonable trade-off between tracking performance and wear in the pitch system. To make high frequency components in the pitch reference more "costly" than low frequency components, a high pass filter is included in the weight for the pitch reference. This means that the weighted performance inputs and outputs can be described as

$$\begin{aligned} w(t) &= v_s(t) \\ z(t) &= \begin{bmatrix} W_\omega(v_s) \int_0^t \omega_{ref}(\tau) - \omega_g(\tau) d\tau \\ W_\beta(v_s) \frac{T(v_s)s+1}{\epsilon s+1} \beta_{ref}(t) \end{bmatrix} \end{aligned}$$

with $W_\omega(v_s(t))$, $W_\omega(v_s(t))$ being scalings that are gain scheduled on wind speed and where $T(v_s(t))$ is the time constant in the high pass filter which is also gain scheduled on wind speed. The parameter values for the weights have been chosen in an iterative procedure and are illustrated in Fig. 3. In this figure it can be seen that the actuator is most expensive at high wind speeds and that the focus on tracking performance is highest in the mid wind speed range, because this range of wind speeds is the region where it is most difficult to maintain the generator speed in the tolerated range. To make the synthesis procedure applicable to practical computation of controllers and to simplify the tuning of the weights it has been decided to limit the synthesis to only four grid points: 15, 18, 21, and 25 m/s.

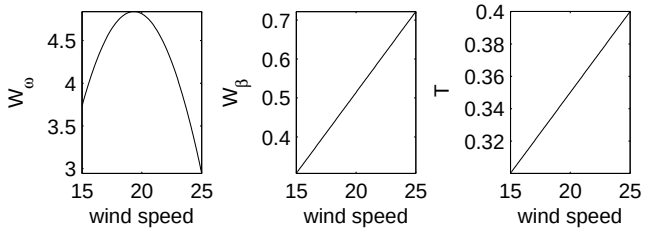


Figure 3: Illustration scheduled variables for the performance weights.

For the construction of suitable scalings, $X(v_s(t))$ and $Y(v_s(t))$, polynomial scalings have been investigated, i.e.

$$X(v_s(t)) = \sum_i X_i v_s(t)^i \quad \text{and} \quad Y(v_s(t)) = Y_0$$

or

$$X(v_s(t)) = X_0 \quad \text{and} \quad Y(v_s(t)) = \sum_i Y_i v_s(t)^i \quad .$$

Then to identify the size of polynomial expansion and which of the two variables that should be parameter dependent, a comparison is made with LTI synthesis at the chosen grid points. LPV controllers have been designed with $\dot{v}_s(t) = 0$ for three different choices of polynomial expansion:

LPV Y1: $X(v_s(t)) = X_0, Y(v_s(t)) = Y_0 + v_s(t)Y_1$.

LPV Y2: $X(v_s(t)) = X_0, Y(v_s(t)) = Y_0 + v_s(t)Y_1 + v_s(t)^2 Y_2$.

LPV X: $X(v_s(t)) = X_0 + v_s(t)X_1 + v_s(t)^2 X_2, Y(v_s(t)) = Y_0$.

The $\mathcal{H}_\infty$ norm of the weighted closed loop has then been calculated for the LTI controller and each of the three LPV controllers at the design points with a comparison given in Table 1. From this comparison it can be concluded that for the particular application it is advantageous to use $Y(v_s(t))$ as the parameter dependent variable and to use a second order approximation.

Table 1: Comparison of $\mathcal{H}_\infty$ synthesis and the closed loop with three different LPV controllers with zero rate of variation.

parm.	H_∞	LPV Y1	LPV Y2	LPV X
15 m/s	0.9998	1.1467	1.0031	1.5456
18 m/s	1.0012	1.2193	1.0016	1.2332
21 m/s	0.9991	1.2125	1.0035	1.1329
25 m/s	1.0012	1.0282	1.0052	1.8139

With the choice of weights and basis functions for $X(v_s(t))$ and $Y(v_s(t))$ in place it is now possible to design the controller with rate bounded parameter variations. Such a design has been done for a number of possible values of rate variation and in Fig. 4 the performance level is illustrated is a function of rate of variation. From the figure it can be seen that the performance level remains almost unchanged until a rate of variation of $0.1\ m/s^2$ where it starts decreasing slightly. Then in the interval from $1\ m/s^2$ to $100\ m/s^2$ it decreases rapidly until it is close to the upper limit (approximately 50% reduction in performance level) given by synthesis with arbitrary fast parameter variations.

From Fig. 4 it can be seen that it is quite inexpensive from a local performance point of view to use $1\ m/s^2$ as the upper limit on parameter rate of variation, which means that the local performance level is decrease by no more than 10% when comparing with LTI controllers for the specific operating point. Furthermore it is expected that the the gain scheduling variable only will have faster rate of variation in extreme operating conditions which will be handled by dedicated control algorithms. It has therefore been decided to focus on the LPV controller design with a parameter rate of variation of $1\ m/s^2$.

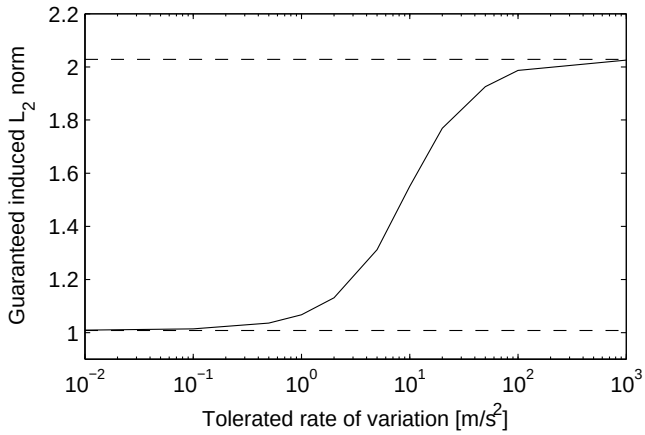


Figure 4: Guaranteed $\mathcal{L}_2$ gain as a function of tolerated parameter rate of variation. Dashed lines indicate indicate lower and upper bounds given by respectively zero and arbitrary fast rate of variation.

6. SIMULATION RESULTS

The chosen controller has been tested in a simulation environment including tower fore-aft and sideways movement, a third order drive train, and nonlinear pitch and generator system models. As wind input it has been decided to use wind specifications according to the IEC IA standard [7] with 10 minute mean wind speeds in the interval from 15 to 25 m/s. A snapshot of a simulation result is given in Fig. 5 from which it can be seen that the generator speed is kept within $\pm 10\ rad/s$ with only low frequency pitch activity and further the noise on a active power is limited to $\pm 50\ kW$.

The controller performance has been tested against LTI controllers designed at each grid point. A comparison by simulation is shown in Table 2 in which the first column represent the damage on the drive train, measured by the rain-flow count (RFC) algorithm. The second column represents the generator speed interval and the third column shows the pitch activity. The fourth and last column illustrates the noise on active power in terms of the standard deviation. The values are shown for the LPV controller relative to the LTI controllers for the particular mean wind speed. From the table it can be observed that the LPV controller is slightly more aggressive in higher wind speeds when comparing with the LTI controller. The reason for this is that with the tolerated rate of variation included in the design, the controllers will be slightly similar over the operating condition. Still it is concluded that the variation from the local design is small enough to conclude that the controller design is successful. Also note that the decrease in performance level for pitch activity comes with an increase in performance on the generator speed tracking and vice versa.

7. CONCLUSIONS

In this paper a LPV controller controller has been designed for the control of wind turbines in full load operation. The design method combines the benefits of two algorithms in the literature. First the two scheduled functions X and Y are determined to give an

Table 2: Selected performance outputs of the LPV controller measured relative to the LTI controllers.

mean wind	RFC. drt.	gen. spd.	pitch	P std.
15 m/s	101 %	104 %	96 %	99 %
18 m/s	97 %	92 %	101 %	100 %
21 m/s	97 %	94 %	100 %	100 %
25 m/s	97 %	90 %	104 %	100 %

optimal performance level γ. This is done on the basis of a method eliminating the controller variables which has an advantage in terms of computational complexity in the associated convex optimisation problem. Then the controller variables are determined by solving a set of LMIs without the need for a reconstruction of the "storage" function, X_{cl}, for the closed loop. This is done by relating the result of the optimisation problem with a method that does not eliminate the controller variables and therefore has an advantage in the construction of the controller.

The controller synthesis shows that the local performance of an LTI controller can approximately be obtained with LPV controller design for the entire operating region with a rate of variation up to $0.1\ m/s^2$. It has been estimated that $1\ m/s^2$ is a suitable worst case for the tolerated rate of variation. For this case choice the performance level is locally decreased by no more than 10% for all operating conditions when comparing with LTI controllers designed for each operating condition.

The selected LPV controller has been simulated on a higher order simulation model and a comparison has been made to a set of local LTI controllers. From this comparison it can be seen that the performance level of the LPV controller and LTI controllers are very similar for each investigated operating conditions.

REFERENCES

[1] P. Apkarian and R.J. Adams. Advanced gain-scheduling techniques for uncertain systems. *Trans. on Control Systems Technology*, 6(1):21–32, 1998.

[2] P. Apkarian and P. Gahinet. A convex characterization of gain-scheduled $\mathcal{H}_\infty$ controllers. *Trans. on Automatic Control*, 40(5):853–864, 1995. doi: 10.1109/9.384219.

[3] M. Chilali and P. Gahinet. $\mathcal{H}_\infty$ design with pole placement contraints. An LMI approach. *Trans. on Automatic Control*, 41(3):358–367, 1996.

[4] N.A. Cutululis, E. Ceanga, A.D. Hansen, and P. Sørensen. Robust multi-model control of an autonomous wind power system. *Wind Energy*, 9:399–419, 2006. doi:10.1002/we.194.

[5] P. Gahinet. Explicit controller formulas for lmi-based $\mathcal{H}_\infty$ synthesis. *Automatica*, 32(7):1007–1014, 1996. doi: 10.1016/0005-1098(96)00033-7.

[6] P. Gahinet, P. Apkarian, and M. Chilali. Affine parameter-dependent lyapunov functions and real parametric uncertaint. *Trans. on Automatic Control*, 41(3):436–442, 1996. doi: 10.1109/9.486646.

[7] IEC. *International Standard, Wind turbines part 1, design requirments*. International Electrotechnical Commission, 2005. IEC 61400-1.

[8] D.J. Leith and W.E. Leithead. Application of nonlinear control to a HAWT. In *Conf. on Control Applications*, pages 245–250, 1994. doi:10.1109/CCA.1994.381191.

[9] F. Lescher, J.Y. Zhao, and P. Borne. Robust gain scheduling controller for pitch regulated variable speed wind turbine. *Studies in Informatics and Control*, 14(4):299–315, 2005.

[10] F. Lescher, J.Y. Zhao, and P. Borne. Switching LPV controllers for a varaible speed pitch regulated wind turbine. *Int. J. Computers, Communications & Control*, 1(4):73–84, 2006.

[11] R.J. Mantz, F.D. Bianchi, and C.F. Christiansen. Gain scheduling control of variable-speed wind energy conversion systems using quasi-LPV models. *Control Engineering Practice*, 13:247–255, 2005. doi:10.1016/j.conengprac.2004.03.006.

[12] K.Z. Østergaard, P. Brath, and J. Stoustrup. Estimation of effective wind speed. In *The Science of Making Torque from Wind*. EAWE, 2007.

[13] K.Z. Østergaard, J. Stoustrup, and P. Brath. Linear parameter varying control of wind turbines covering both partial load and full load conditions. *Submittet to Int. J. Robust and Nonlinear Control*, 2007.

[14] C.W. Scherer. *Mixed $\mathcal{H}_2/\mathcal{H}_\infty$ Control*. Delft University of Technology, 1995.

[15] T.G. van Engelen, E.L. van der Hooft, and P. Schaak. Development of wind turbine control algorithms for industrial use. In *European Wind Energy Conference*, 2003.

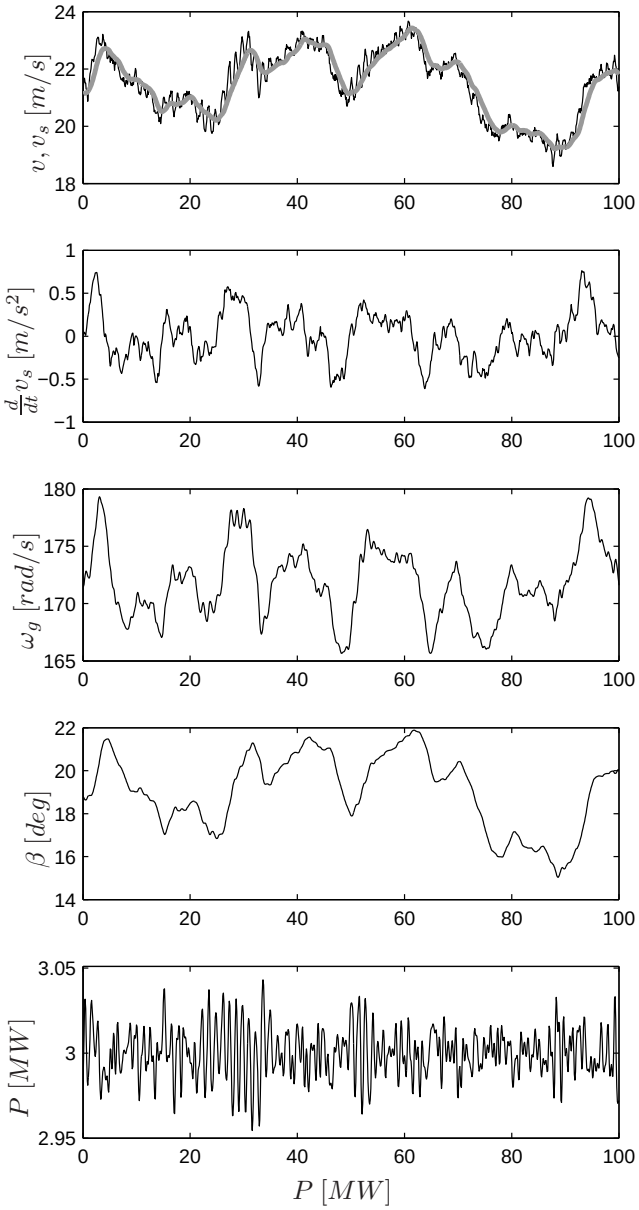


Figure 5: Simulation results with LPV controller with rate of variation up to 0.1 m/s. Black lines: simulation variables. Gray line: Scheduling variable.

Report E

Quasi-LPV Controller Design for Wind Turbines Using Linear Fractional Transformations

Contents

Technical report.

Quasi-LPV controller design for wind turbines using linear fractional transformations

Kasper Zinck Østergaard
Turbine Control and Operation
Loads, Aerodynamics and Control
Vestas Wind Systems A/S

Jakob Stoustrup
Automation and Control
Department of Electronic Systems
Aalborg University

Per Brath
Turbine Control and Operation
Loads, Aerodynamics and Control
Vestas Wind Systems A/S

Abstract. Linear parameter varying (LPV) control is a methodology with which it is possible to design a nonlinear controller for a specific class of nonlinear systems. If the considered model is in the class of quasi-LPV models in which the scheduling parameter is dependent on system variables, the methodology has however certain limitations.
In this report it is investigated how to design an LPV controller for a wind turbine operating in the high wind speed range on the basis of a quasi-LPV formulation of the weighted open loop model. From the findings in the report it is concluded that performance level for frozen parameters is crucial for the obtainable results with the LPV method. It is therefore concluded that a minimum requirement is that the controllability and observability conditions are similar for the linearised dynamics and the quasi-LPV model with frozen parameters.
Essentially a condition for obtaining good results with the methodology it is required to have frozen parameter dynamics which are similar to what is obtained by linearising the dynamics at the desired operating condition.

1. Introduction

This report considers the design of a gain scheduled controller for full load operation. Classical gain scheduling has the disadvantage that it assumes that the operating condition changes very slow compared to the dynamics of the wind turbine[3, 2]. It is expected that this assumption is violated even in nominal operation because even smaller wind gusts can make the operating point change quickly.

As an alternative approach we consider the gain scheduling approach denoted linear parameter varying (LPV) control which can take into account how fast the operating point can change. In particular we apply an algorithm designed for rational parameter dependency in which arbitrary fast parameter variation is assumed.

In Section 2 it is described how to use simulation studies to describe the parameter region for wind turbines operating in full load. Further a dynamic model is developed to include drive train dynamics, nonlinear actuators, and nonlinear aerodynamics that are approximated by a polynomial static function.

Then in Section 3 it is investigated how to design an LPV controller on the basis of the methodology in [4, 5]. It is emphasised that the quasi-LPV characterisation of the nonlinear model has a very important influence on the performance level that can be obtained by the design method. Finally in Section 4 the conclusions and recommendations are given.

The notation in the report is as follows: The symbol $\prec$ ($\succ$) is used in linear matrix inequalities (LMIs) to denote negative (positive) definiteness of the left hand side, e.g. $A \prec B$ means $A - B \prec 0$. Further we use red coloured symbols to indicate design variables, e.g. a design variable X will be denoted $\mathcal{X}$. Also the LPV formulations colours are used to indicate the scheduling parameter, e.g. a scheduling parameter p will be denoted $\mathcal{p}$.

2. Model

In the introduction it was presented that focus will be on the design of a controller for full load operation. More specifically we will focus on minimising the oscillations in the drive train while minimising the generator speed variations and control actuation. For this problem formulation a dynamic model is used with the components illustrated in Figure 1. In the following subsections each component in the model is presented and they are interconnected in a representation denoted upper linear fractional transformation (LFT). However, first it is necessary to describe the operating region that will be assumed throughout the design.

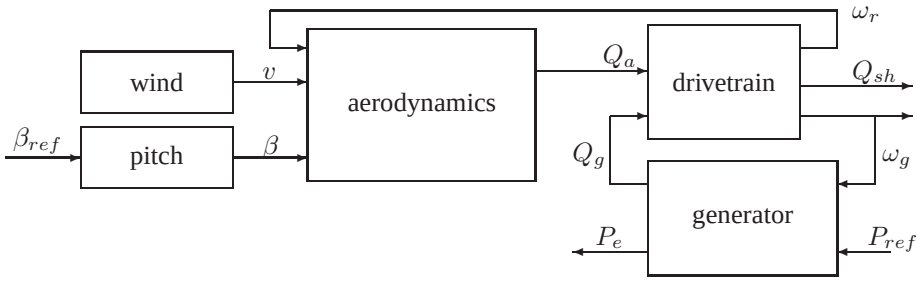


Figure 1: Block diagram of model interconnection.

2.1. Operating region

In Section 2.6 concerning the aerodynamics model it will be clear that the nonlinearities caused by the aerodynamics can be described by the wind speed, v, pitch angle, β, and rotor speed, ω_r. Further in Section 2.4 it is shown that the generator model is nonlinear and that the nonlinearity can be characterised by the generator speed, ω_g. Throughout this report it is assumed that these four variables can uniquely determine all considered operating conditions for the wind turbine. In this section the range of these four variables will be investigated through simulations with an existing controller that has performance similar to what is expected by the designed controller. The simulations are performed with stochastic wind input with turbulence intensity according to the IEC A specification [1] – an example of such a simulation is presented in Figure 2.

In Figure 3 different relations between the mentioned variables are illustrated. In the topmost plot, the relation between the rotor speed and generator speed is illustrated. From this plot it can be seen that there is a strong relation between the rotor speed and the generator speed. This relation will be characterised by a 2-D polytope as shown by the red figure. This polytope can be characterised by the set of equations

$$14.0 \leq \omega_r \leq 17.3 \quad , \quad (\omega_r - 0.2) \cdot N \leq \omega_g \leq (\omega_r + 0.2) \cdot N \tag{1}$$

where:

ω_r is the rotor speed [rpm]

ω_g is the generator speed [rpm]

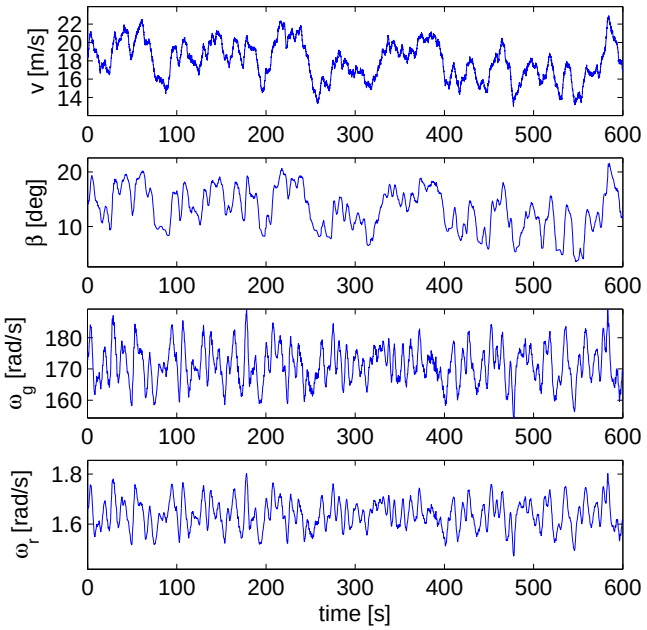


Figure 2: Time response of variables used to determine operating condition. A strong correlation can be observed between wind speed and pitch angle and between generator speed and rotor speed.

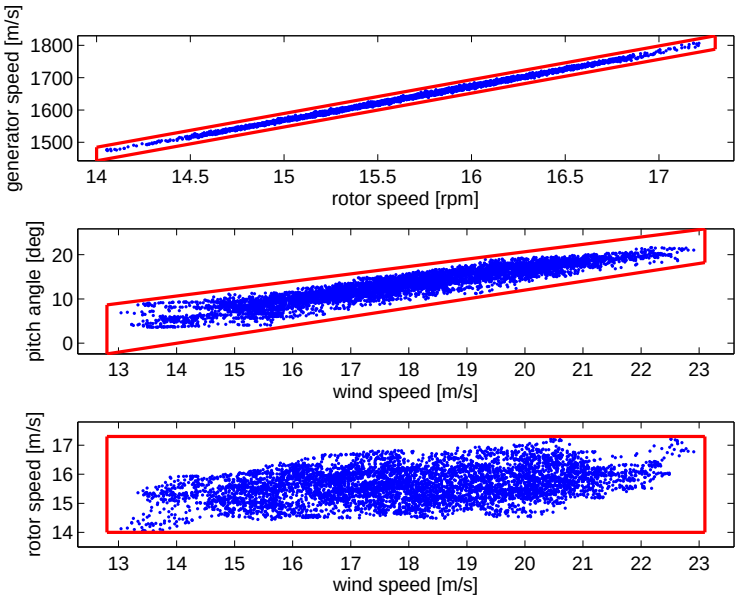


Figure 3: Simulation plot for operating region. Strong correlation between rotor speed and generator speed and between wind speed and pitch angle described by polytopic regions. Wind speed and rotor speed assumed independent and relation is therefore described by a box.

N is the gearing ratio between the low speed shaft and the high speed shaft.

In the middle illustration of Figure 3, the relation between the wind speed and the pitch angle is shown. From this graph it can be seen that the pitch angle as expected increases with wind speed according to the relation

$$12.8 \leq v \leq 23.1 \quad , \quad 2 \cdot v - 28 \leq \beta \leq \frac{5}{3} \cdot v - \frac{38}{3} \tag{2}$$

where

v is the effective wind speed [m/s]

β is the pitch angle [deg]

By studying the bottom graph in Figure 3 together with Figure 4, no strong correlation between rotor speed and wind speed can be concluded. It has therefore been decided to assume that they are independent which means that the four variables determining the operating condition can be described by (1) and (2). It could be argued that the operating region can be reduced by rotating the "box" slightly counter-clockwise. From a more detailed inspection it can be observed that the marks in the bottom left and to right corner of the box are sparse compared with the centre marks. Further the controller to be designed should result in a mean value close to the rated rotor speed (around 16 rpm), i.e. good tracking of the reference. As a consequence it has been decided to describe the two variables as being uncorrelated.

2.2. Wind model

The wind model is achieved by system identification with an ARMA model. The mean of the time series in the topmost graph in Figure 2 is removed from the time series and standard LTI methods for system identification of the ARMA model are used. It turns out that the spectrum can be approximated appropriately in this region by a second order LTI model of the form:

$$v(s) = \frac{b_1 s + b_2}{s^2 + a_1 s + a_2}\,\hat{\chi}(s) + \bar{v} = G_w(s)\hat{\chi}(s) + \bar{v} = G_w(s)(\hat{\chi}(s) + G_w(0)^{-1}\bar{v})$$

where:

$v(.)$ is the wind speed experienced by the rotor swept area.

$\hat{\chi}(.)$ is zero mean Gaussian white noise with unity standard deviation.

$\bar{v}$ is the mean wind speed which is assumed constant.

2.3. Pitch model

For the pitch system a simplistic nonlinear second order model is assumed from control voltage, u, to pitch angle, β. This model lumps the hydraulic, mechanical and electrical dynamics of the pitch system into a model of the form (3) which includes the main characteristics when seen from the mechanical point of view. The model includes a static

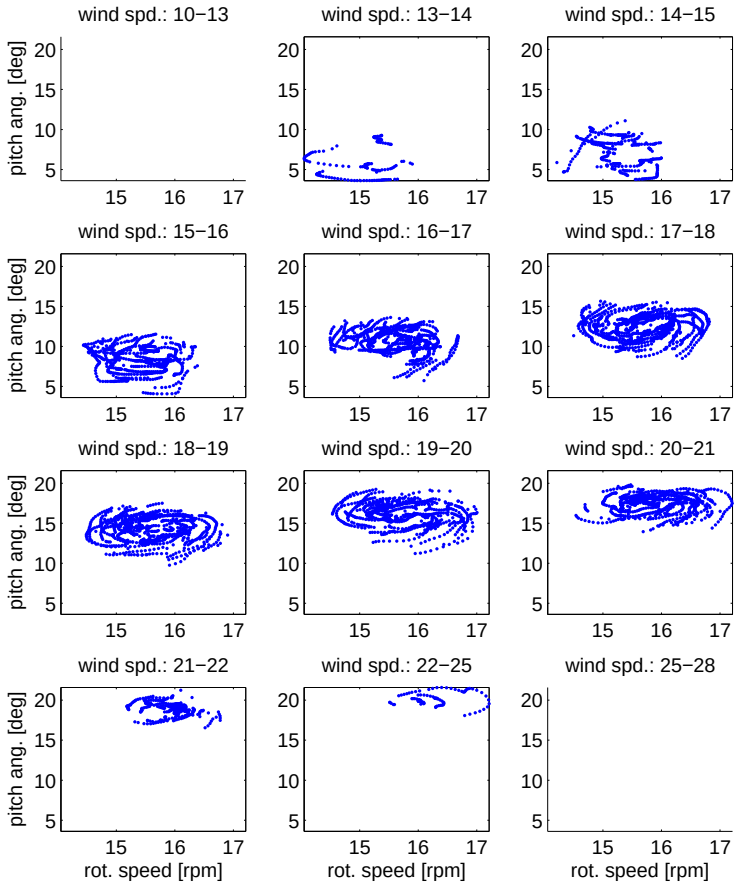


Figure 4: Relation between generator speed and pitch angle binned into categories of different wind speeds. Most recorded data is in the wind speed range 15-21 m/s and analysis based on the two middle rows. Observe that the range of rotor speed is similar for all six plots which indicates that rotor speed and wind speed are independent. Note that the top left and bottom right plot are empty due as a consequence of having no data points in the respective wind speed regions.

nonlinearity f which combines the nonlinear valve characteristics with the mapping from piston movement to blade rotation. Furthermore the model has two parameters: T_β which is the time constant of the system and T_{delay} which is a combination of computational delays and a hydraulic time delay.

$$T_\beta \ddot{\beta}(t) + \dot{\beta}(t) = f\big(u(t - T_{delay})\big) \tag{3}$$

In the leftmost illustration of Figure 5 the nonlinearity in the pitch system is illustrated. Most of this nonlinearity is taken care of by a nonlinear P-controller with the open loop shown in the illustration in the middle. The open loop gain of this controller interconnected with the nonlinear part of the model is then illustrated in the right most illustration. With a pitch error of ± 2.5 deg the linearisation shown in Figure 5 is assumed adequate and the closed loop is then as shown in (4) with the slope of the red

dashed line in Figure 5 denoted K_β.

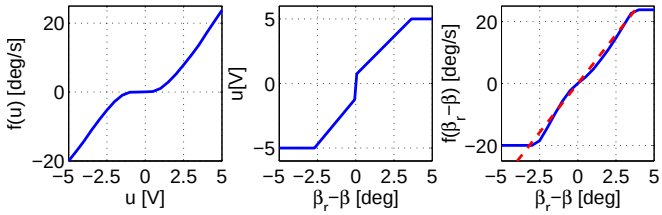


Figure 5: Linearisation of nonlinear part of pitch system. Left: Nonlinear gain, f, in pitch model. Middle: Nonlinear controller gain, P. Right: Open loop interconnection of model gain with controller gain, $P \cdot f$. Observe that the controller linearises the model nonlinearity for small tracking errors with open loop gain, $P \cdot f \approx K_\beta$.

$$\beta = \frac{K_\beta}{s(T_\beta s + 1) + K_\beta} \cdot \beta_r \tag{4}$$

2.4. Generator model

The generator and converter model is modelled to be a gain that is scheduled on the current generator speed in the form

$$P_e(t) = \frac{\omega_g(t)\, pp}{\omega_{NET}} P_s(t)$$

where

$P_e(t)$ is the active power produced by the wind turbine

$P_s(t)$ is the stator power

$\omega_g(t)$ is the generator speed

ω_{NET} is the net (grid) frequency

pp is the number of pole pairs in the generator

The inner controller is modelled as an integrator with an associated gain as

$$P_s(t) = K \int_0^t \left(P_{ref}(s) - P_e(s)\right) d\tau \quad .$$

And by considering the relation between generator torque and active power as

$$\eta\, \omega_g(t)\, Q_g(t) = P_e(t)$$

with η being the generator efficiency we can obtain an LPV model of the generator and converter system as in (5), where ω_g is the scheduling variable.

$$\begin{bmatrix} \dot{x}_g \\ Q_g \\ P_e \end{bmatrix} = \begin{bmatrix} -\frac{\omega_g}{T} & \frac{1}{T} \\ \frac{1}{\eta} & 0 \\ \omega_g & 0 \end{bmatrix} \begin{bmatrix} x_g \\ P_{ref} \end{bmatrix} \quad , \quad T = \frac{\omega_{NET}}{pp\, K} \tag{5}$$

This state space representation can be described by an LFT in the following way: Let $w = \omega_g\, z$ and $z = x_g$, then the model can be represented as the interconnection of an LTI system with ω_g as shown in (6).

$$\begin{bmatrix} z_g \\ \dot{x}_g \\ Q_g \\ P_e \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ -\frac{1}{T} & 0 & \frac{1}{T} \\ 0 & \frac{1}{\eta} & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} w_g \\ x_g \\ P_{ref} \end{bmatrix} \quad , \qquad w_g = \omega_g z_g \tag{6}$$

2.5. Drive train model

The drive train will be described by a two degree-of-freedom model as illustrated in Figure 6. The model consists of a low speed shaft in which the parameters of the main shaft and blades are lumped into one component. The other main component is the high speed shaft which includes also the generator inertia and flexibility. Further the dynamics of the gear box is lumped into the two shafts. For the dynamics, the two inertias are interconnected by a spring and damper, and friction is included on both components.

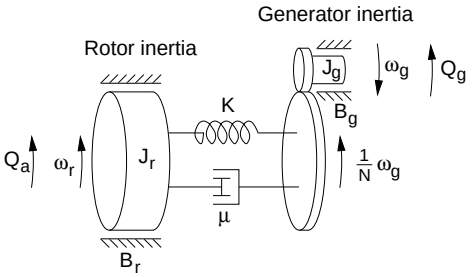


Figure 6: Model of drive train as the interconnection of two inertias by a spring, damper, and gearing ration.

When interconnecting the components shown in Figure 6 the drive train model can be described by the LTI differential equations in (7) and the damage in the shaft will be measured by the torque between the two inertias given by (8).

$$J_r \cdot \dot{\omega}_r(t) = Q_a(t) - B_r \cdot \omega_r(t) - \mu \cdot \left(\omega_r(t) - \frac{1}{N} \cdot \omega_g(t)\right) - K \cdot \theta_\Delta(t) \tag{7a}$$

$$J_g \cdot \dot{\omega}_g(t) = -Q_g(t) - B_g \cdot \omega_g(t) + \frac{\mu}{N} \cdot \left(\omega_r(t) - \frac{1}{N} \cdot \omega_g(t)\right) + \frac{K}{N} \cdot \theta_\Delta(t) \tag{7b}$$

$$\dot{\theta}_\Delta(t) = \omega_r(t) - \frac{1}{N} \cdot \omega_g(t) \tag{7c}$$

$$Q_{sh} = \mu \cdot \left(\omega_r(t) - \frac{1}{N} \cdot \omega_g(t)\right) + K \cdot \theta_\Delta(t) \tag{8}$$

2.6. Aerodynamics model

In this report only collective pitch will be considered which means that it suffices to consider the spatial average of the wind field over the rotor swept area. Then the aerodynamics can be described by the following nonlinear, static function

$$Q_a(v, \omega_r) = \frac{1}{2} \cdot \rho \cdot A_r \cdot \frac{v^3}{\omega_r} \cdot c_P(\beta, \lambda) \quad , \quad \lambda = \frac{R \cdot \omega_r}{v} \tag{9}$$

where $c_p(\beta, \lambda)$ is a nonlinear function describing the efficiency of the rotor in the given set point. From Section 2.1 the operating region for the controller design was determined for the variables, ω_r, v, and β. The tip speed ratio, λ is used in determining the aerodynamics, and for the description of $c_P(\beta, \lambda)$ as a rational function of the two variables, the relation between the tip speed ratio and pitch angle is determined.

In Figure 7 the operating region is illustrated with the blue figure showing the region for the minimum rotor speed while the green figure shows the region for the maximum rotor speed. Because of the relation between λ, ω_r and v (given in (9)), the region for intermediate values of ω_r can be determined by linear interpolation which yields the red dashed figure in Figure 7. This dashed figure then illustrates the operating region for the two variables according to the described operating region in Section 2.1. The two horizontal red lines illustrate the bounds for which there is measurements of the pitch angle.

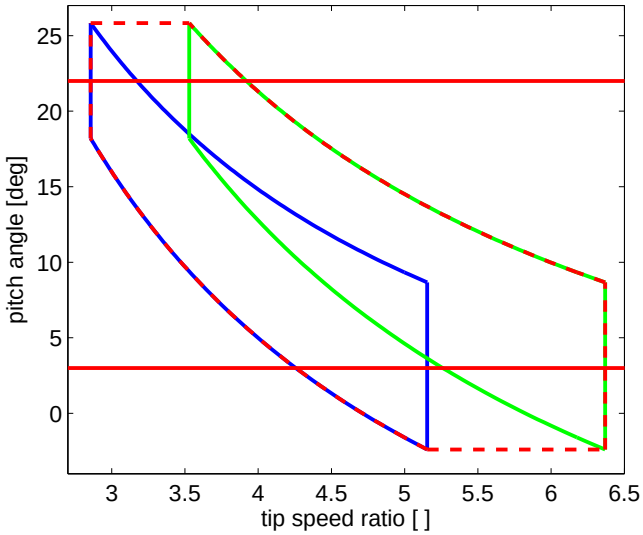


Figure 7: Operating region for tip speed ratio and pitch angle. Blue (green) figure marks operating region for minimum (maximum) rotor speed. The red dashed figure illustrates the complete operating region by combination of the two extremes. The solid red lines indicate the limit within which measurements of pitch angle have been recorded.

For simplicity in the description of the polytope these two bounds have not been included, but in the approximation of the aerodynamics the inclusion of these two bounds greatly simplifies the approximation of the aerodynamics. It has therefore been chosen to include these two bounds when doing the approximation of the aerodynamics.

Note 14. The inclusion of the two extra bounds in the approximation of the aerodynamics means that outside these two bounds the stability and performance guarantees by the analysis and synthesis cannot be trusted. A better approximation in these regions will be necessary for giving these guarantees.

When restricting the pitch angle and the tip speed ratio to be inside this operating region, $c_P(\beta, \lambda)$ is approximated by a polynomial function in the two variables. The result of this approximation is shown in Figure 8. An indication of the approximation error can be seen from (10) when restriction the approximation to be valid within the two bounds.

$$\frac{\max(err) - \min(err)}{\max(c_P) - \min(c_P)} = 4.9\% \tag{10}$$

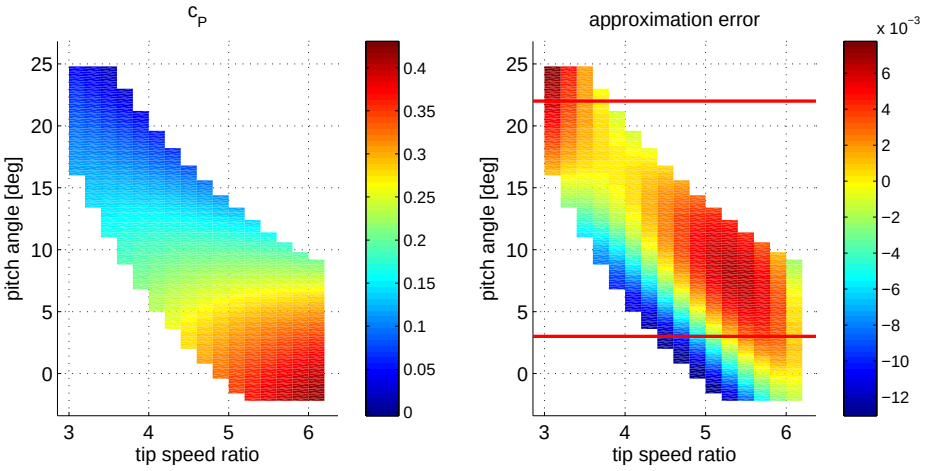


Figure 8: Approximation of c_P-lookup table. The left plot illustrates $c_p(\beta, \lambda)$ and the right plot illustrates the approximation error by the polynomial fit. Note that the approximation error is within ± 0.01 in the region where measurements of pitch have been observed (marked by the horizontal red lines).

The approximation is of the form (11) which when inserted in (9) gives the static polynomial model of the aerodynamics in (13) – with q_i being model constants. For the purpose of deriving an LPV model for the wind turbine it is necessary to transform (13) into an input-output LPV form as done in (14). In this representation it has been decided to introduce scalings, k, in order to be able to tune the realisation into one that is appropriate for the numerical computations used in the controller design.

$$\begin{aligned} c_P(\beta, \lambda) \approx & c_1\,\lambda + c_2\,\lambda^2 + c_3\,\lambda^3 + c_4\,\beta\,\lambda + c_5\,\beta\,\lambda^2 + c_6\,\beta\,\lambda^3 \\ & + c_7\,\beta^2\,\lambda + c_8\,\beta^2\,\lambda^2 + c_9\,\beta^3\,\lambda \end{aligned} \tag{11}$$

(12)

$$Q_a(\beta, v, \omega_r) = \frac{1}{2}\,\rho\,A_r\,\frac{v^3}{\omega_r}\,c_P(\beta, \frac{R\,\omega_r}{v})$$
$$\approx q_1\,v^2 + q_2\,v\,\omega_r + q_3\,{\omega_r}^2 + q_4\,\beta\,v^2 + q_5\,\beta\,v\,\omega_r + q_6\,\beta\,{\omega_r}^2$$
$$+ q_7\,\beta^2\,v^2 + q_8\,\beta^2\,v\,\omega_r + q_9\,\beta^3\,v^2 \tag{13}$$

$$Q_a(\beta, v, \omega_r) \approx \left(k_4^\beta\,q_4\,v^2 + k_5^\beta\,q_5\,v\,\omega_r + k_7^\beta\,q_7\,\beta\,v^2 + k_8^\beta\,q_8\,\beta\,v\,\omega_r + q_9\,\beta^2\,v^2\right)\,\beta$$
$$+ \left(q_1\,v + k_2^v\,q_2\,\omega_r + k_4^v\,q_4\,\beta\,v + k_5^v\,q_5\,\beta\,\omega_r + k_7^v\,q_7\,\beta^2\,v\right)\,v$$
$$+ \left(k_2^\omega\,q_2\,v + q_3\,\omega_r + k_5^\omega\,q_5\,\beta\,v + q_6\,\beta\,\omega_r + k_8^\omega\,q_8\,\beta^2\,v\right)\,\omega_r \tag{14}$$
$$\text{where: } k_i^\beta + k_i^v + k_i^\omega = 1$$

Note 15. The scalings k_6^β, k_8^v, and k_9^v have been forced to zero, because the inclusion of these would lead to a parameter block of size larger than 6×6.

Linear fractional transformation of aerodynamics model

On the basis of the factorisation of the LPV model in (14), a linear fractional transformation (LFT) is performed as shown in the following.

Note 16. It should be noted that neither the LPV model nor the LFT is unique and for technical reasons different LFTs might give different results for analysis and synthesis in terms of the performance that can be achieved.

First we parameterise (14) as

$$Q_a = \left(\omega_r v\left(k_5^\beta\,q_5 + k_8^\beta\,q_8\beta\right) + v^2\left(k_4^\beta\,q_4 + \beta(k_7^\beta\,q_7 + q_9\,\beta)\right)\right)\beta$$
$$+ \left(\omega_r\left(k_2^v\,q_2 + k_5^v\,q_5\beta\right) + v\left(q_1 + \beta(k_4^v\,q_4 + k_7^v\,q_7\,\beta)\right)\right)v$$
$$+ \left(\omega_r(q_3 + q_6\,\beta) + v\left(k_2^\omega\,q_2 + \beta(k_5^\omega\,q_5 + k_8^\omega\,q_8\,\beta)\right)\right)\omega_r$$

from which we can choose ω_r and v as the first block of parameters. This means that the aerodynamics model can be described by the interconnection

$$\begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = \begin{bmatrix} \omega_r & 0 \\ 0 & v \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}$$
$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} v(k_5^\beta\,q_5 + k_8^\beta\,q_8\beta) & k_2^v\,q_2 + k_5^v\,q_5\,\beta \\ v\left(k_4^\beta\,q_4 + \beta(k_7^\beta\,q_7 + q_9\,\beta)\right) & q_1 + \beta(k_4^v\,q_4 + k_7^v\,q_7\,\beta) \\ q_3 + q_6\,\beta & \\ k_2^\omega\,q_2 + \beta(k_5^\omega\,q_5 + k_8^\omega\,q_8\,\beta) & \end{bmatrix} \begin{bmatrix} \beta \\ v \\ \omega_r \end{bmatrix}$$

Then the aerodynamic torque is determined by $Q_a = w_1 + w_2$. Still there are several parameters in z_1 and z_2 we need to pull out in order to get a LFT model of the

aerodynamics. Let $w_3 = v z_3$ and $z_3 = \beta$. Then

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} k_5^\beta\, q_5 + k_8^\beta\, q_8 \beta \\ k_4^\beta\, q_4 + \beta(k_7^\beta\, q_7 + q_9\, \beta) \end{bmatrix} w_3 + \begin{bmatrix} 0 & k_2^v\, q_2 + (k_5^v\, q_5)\, \beta & q_3 + q_6\, \beta \\ 0 & q_1 + \beta(k_4^v\, q_4 + k_7^v\, q_7\, \beta) & k_2^\omega\, q_2 + \beta(k_5^\omega\, q_5 + k_8^\omega\, q_8\, \beta) \end{bmatrix} \begin{bmatrix} \beta \\ v \\ \omega_r \end{bmatrix}$$

Now let

$$\begin{bmatrix} w_4 \\ w_5 \end{bmatrix} = \begin{bmatrix} \beta & 0 \\ 0 & \beta \end{bmatrix} \begin{bmatrix} z_4 \\ z_5 \end{bmatrix}$$

$$\begin{bmatrix} z_4 \\ z_5 \end{bmatrix} = \begin{bmatrix} k_8^\beta\, q_8 \\ k_7^\beta\, q_7 + q_9\, \beta \end{bmatrix} w_3 + \begin{bmatrix} 0 & k_5^v\, q_5 & q_6 \\ 0 & k_4^v\, q_4 + k_7^v\, q_7\, \beta & k_5^\omega\, q_5 + k_8^\omega\, q_8\, \beta \end{bmatrix} \begin{bmatrix} \beta \\ v \\ \omega_r \end{bmatrix}$$

and

$$w_6 = \beta\, z_6 \quad , \qquad z_6 = q_9\, w_3 + \begin{bmatrix} 0 & k_7^v\, q_7 & k_8^\omega\, q_8 \end{bmatrix} \begin{bmatrix} \beta \\ v \\ \omega_r \end{bmatrix}$$

Then the aerodynamic model can be described by the interconnection of the two components

$$\begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \\ w_5 \\ w_6 \end{bmatrix} = \begin{bmatrix} \omega_r & & & & & \\ & v & & & & \\ & & v & & & \\ & & & \beta & & \\ & & & & \beta & \\ & & & & & \beta \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \\ z_6 \end{bmatrix}$$

$$\left[\begin{array}{c} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \\ z_6 \\ \hline Q_a \end{array}\right] = \left[\begin{array}{ccccccc|ccc} 0 & 0 & k_5^\beta q_5 & 1 & 0 & 0 & 0 & k_2^v q_2 & q_3 \\ 0 & 0 & k_4^\beta q_4 & 0 & 1 & 0 & 0 & q_1 & k_2^\omega q_2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & k_8^\beta q_8 & 0 & 0 & 0 & 0 & k_5^v q_5 & q_6 \\ 0 & 0 & k_7^\beta q_7 & 0 & 0 & 1 & 0 & k_4^v q_4 & k_5^\omega q_5 \\ 0 & 0 & q_9 & 0 & 0 & 0 & 0 & k_7^v q_7 & k_8^\omega q_8 \\ \hline 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right] \left[\begin{array}{c} w_1 \\ w_2 \\ w_3 \\ w_4 \\ w_5 \\ w_6 \\ \hline \beta \\ v \\ \omega_r \end{array}\right]$$

2.7. Interconnection of sub-models

The interconnection of the components described in the above sections can be done by standard LFT techniques like in [6] to get

$$\left[\begin{array}{c} \dot{x} \\ \hline z_u \\ \hline Q_{sh} \\ \omega_r \\ \omega_g \\ v \\ \beta \end{array}\right] = \left[\begin{array}{c|c|c} A & B_u & B_p \\ \hline C_u & D_u & D_{up} \\ \hline C_p & D_{pu} & D_p \end{array}\right] \left[\begin{array}{c} x \\ \hline w_u \\ \hline \chi \\ \beta_{ref} \\ P_{ref} \end{array}\right] \quad , \quad w_u = \Delta z_u$$

with

$$\Delta = \begin{bmatrix} \omega_g & & & & & & \\ & \omega_r & & & & & \\ & & v & & & & \\ & & & v & & & \\ & & & & \beta & & \\ & & & & & \beta & \\ & & & & & & \beta \end{bmatrix}$$

3. Controller design

With the model set up in Section 2, the controller can now be designed to track a generator speed reference and power reference with minimal control effort and drive train oscillations. To obtain this, it has been decided to use a tracking setup as indicated by Figure 9.

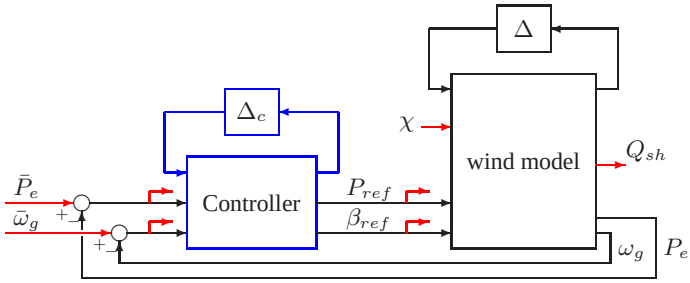


Figure 9: Block diagram of closed loop interconnection. Blue lines mark controller and red lines mark performance inputs and outputs.

The block diagram illustrates to the right the interconnection of the LTI component of the model with the parameter block, Δ. The target is then to design a controller consisting of an LTI part and a scheduling function, Δ_c, such that the performance specification is satisfied. For the design the performance inputs and performance outputs

will be weighted according to

$$W_{in} = \begin{bmatrix} K_{i1} & & \\ & K_{i2}\frac{s+\omega_{i2}}{s+\epsilon} & \\ & & K_{i3}\frac{s+\omega_{i3}}{s+\epsilon} \end{bmatrix}$$

$$W_{out} = \begin{bmatrix} K_{o1}\frac{s}{s+\omega_{o1}} & & & & \\ & K_{o2}\frac{s+\omega_{o2}}{s+\epsilon} & & & \\ & & K_{o3}\frac{s+\omega_{o3}}{s+\epsilon} & & \\ & & & K_{o4}\frac{s}{s+\omega_{o4}} & \\ & & & & K_{o5}\frac{s}{s+\omega_{o5}} \end{bmatrix}$$

The design problem is solved on the basis of the results presented in [4, 5]. The procedure is summarised in Theorem 17 on the facing page and is based on a weighted open loop model described by (15) with w_p and z_p as the input and outputs of the performance channel, and u and y as the inputs and outputs of the control channel. With the parameter region described by a convex polytope it is then possible to test if an LPV controller exists by solving a set of LMIs at the vertices of this polytope.

$$\begin{bmatrix} \dot{x} \\ z_u \\ z_p \\ y \end{bmatrix} = \begin{bmatrix} A & B_u & B_p & B \\ C_u & D_{uu} & D_{up} & E_u \\ C_p & D_{pu} & D_{pp} & E_p \\ C & F_u & F_p & 0 \end{bmatrix} \begin{bmatrix} x \\ w_u \\ w_p \\ u \end{bmatrix} \quad , \quad w_u = \Delta z_u \tag{15}$$

When designing dynamic controllers to be robust against a specified uncertainty a usual practice is to start with a design of a nominal controller – with the uncertainty block, Δ, set to zero. Then the uncertainty is graduately increased to the desired amount of uncertainty in order to keep track of the amount of tolerated uncertainty if the design process should fail for the desired level of uncertainty.

A similar approach is taken for the design of the LPV controller. For numerical reasons the nominal performance level is scaled to $\gamma_{nom} = 0.001$. Then the obtainable performance level is determined by the approach in Theorem 17 for a graduately increased scaling of the parameter region – with 0 as the nominal controller and 1 as the desired range of parameter values. In Figure 10 the result of this study is presented from which it can be seen that it is not possible to obtain even stability for a scaling of 20 % and larger.

It is clear that this result is caused by a limitation in the design process, because real world applications show that it is indeed possible to stabilise wind turbines with controllers having simple structure. Also the simulations presented in Figure 2 illustrate this. In order to understand the issue we need a further analysis which is presented in the following.

First of all we recall that the requirements in Theorem 17 are equivalent to requiring (16) satisfied with $A(p)$, $B_p(p)$, ..., $D_p(p)$ being the open loop system matrices evaluated at each frozen parameter in the polytope of parameter values. This means that a minimal requirement is that the performance specification is satisfied for each frozen parameter value in the set of tolerated parameter values. This is a substantial difference when comparing with other nonlinear control techniques because it means that the parameter variations can never increase the possible performance level in the applied

Theorem 17 ([4, 5]). *Let the weighted open loop be given by (15) and assume that the set of parameters vary in a convex polytope with arbitrary fast parameter variations. Let p_i denote the parameter vector in the n vertices of this polytope and $\Delta(p_i)$ is then the uncertainty block each each vertex. Then there is a controller for which the induces $\mathcal{L}_2$ norm, γ, is satisfied if there is a set of symmetric variables: $Q, R, \tilde{Q}, \tilde{R}, X, Y$ and unstructured variables: S, $\tilde{S}$ for which the following set of LMIs is satisfied*

$$Q \prec 0 \quad , \quad \begin{bmatrix} \Delta(p_i) \\ I \end{bmatrix}^T \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} \Delta(p_i) \\ I \end{bmatrix} \succ 0 \quad , i = 1, \ldots, n$$

$$\tilde{R} \succ 0 \quad , \quad \begin{bmatrix} I \\ -\Delta(p_i)^T \end{bmatrix}^T \begin{bmatrix} \tilde{Q} & \tilde{S} \\ \tilde{S}^T & \tilde{R} \end{bmatrix} \begin{bmatrix} I \\ -\Delta(p_i)^T \end{bmatrix} \prec 0 \quad , i = 1, \ldots, n$$

$$\left[\begin{array}{c|c} \Psi & 0 \\ \hline 0 & I \end{array}\right]^T \left[\begin{array}{ccc|c} XA + A^TX + C_u^TRC_u & XB_u + C_u^TS^T + C_u^TRD_u & XB_p + C_u^TSD_{up} & C_p^T \\ B_u^TX + SC_u + D_u^TRC_u & Q + SD_u + D_u^TS^T + D_u^TRD_u & SD_{up} + D_u^TRD_{up} & D_{pu}^T \\ B_p^TX + D_{up}^TRC_u & D_{up}^TS^T + D_{up}^TRD_u & -\gamma I + D_{up}^TRD_{up} & D_p^T \\ \hline C_p & D_{pu} & D_p & -\gamma I \end{array}\right] \left[\begin{array}{c|c} \Psi & 0 \\ \hline 0 & I \end{array}\right] \prec 0$$

$$\left[\begin{array}{c|c} \Phi & 0 \\ \hline 0 & I \end{array}\right]^T \left[\begin{array}{ccc|c} AY + YA^T - B_u\tilde{Q}B_u^T & YC_u^T + B_u\tilde{S} - B_u\tilde{Q}D_u^T & YC_p^T - B_u\tilde{Q}D_{pu}^T & B_p \\ C_uY + \tilde{S}^TB_u^T - D_u\tilde{Q}B_u^T & -\tilde{R} + D_u\tilde{S} + \tilde{S}^TD_u^T - D_u\tilde{Q}D_u^T & \tilde{S}^TD_{pu}^T - D_u\tilde{Q}D_{pu}^T & D_{up} \\ C_pY - D_{pu}\tilde{Q}B_u^T & D_{pu}\tilde{S} - D_{pu}\tilde{Q}D_u^T & -\gamma I - D_{pu}\tilde{Q}D_{pu}^T & D_p \\ \hline B_p^T & D_{up}^T & D_p^T & -\gamma I \end{array}\right] \left[\begin{array}{c|c} \Phi & 0 \\ \hline 0 & I \end{array}\right] \prec 0$$

where Ψ is a basis for the kernel of $\begin{bmatrix} C & F_u & F_p \end{bmatrix}$ and Φ is a basis for the kernel of $\begin{bmatrix} B^T & E_u^T & E_p^T \end{bmatrix}$.

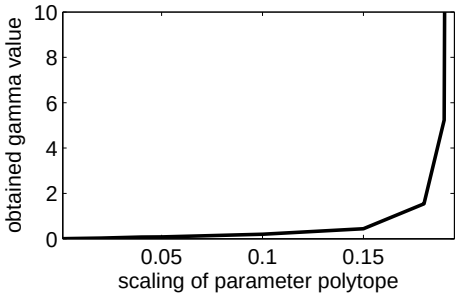


Figure 10: Performance level as function of scaling of parameter region. Note that the synthesis fails for scalings larger than 20% of the design region.

methodology. This is a critical issue when using LPV methods for quasi-LPV models because the method assumes that the parameter can vary in any way within the specified bounds. For quasi-LPV systems the parameter vector depends on the state vector and is therefore restricted further in possible variation which is not taken into account by the method.

$$\begin{bmatrix} Y & I \\ I & X \end{bmatrix} \succ 0 \tag{16a}$$

$$\begin{bmatrix} \Psi(p) & 0 \\ 0 & I \end{bmatrix}^T \begin{bmatrix} XA(p) + A(p)X & XB_p(p) & C_p(p)^T \\ B_p(p)^T X & -\gamma I & D_p(p)^T \\ C_p(p) & D_p(p) & -\gamma I \end{bmatrix} \begin{bmatrix} \Psi(p) & 0 \\ 0 & I \end{bmatrix} \prec 0 \tag{16b}$$

$$\begin{bmatrix} \Phi(p) & 0 \\ 0 & I \end{bmatrix}^T \begin{bmatrix} A(p)Y + YA(p) & YC_p(p)^T & B_p(p) \\ C_p(p)Y & -\gamma I & D_p(p) \\ B_p(p)^T & D_p(p)^T & -\gamma I \end{bmatrix} \begin{bmatrix} \Phi(p) & 0 \\ 0 & I \end{bmatrix} \prec 0 \tag{16c}$$

Because of this limitation it is crucial that the quasi-LPV model is determined in a way with which the desired performance level can be determined for all frozen parameter values. With these limitations in mind it will now be analysed why it is not possible to obtain stability for the parameter region with the decided realisation.

Investigations similar to what was presented in Figure 10 has been performed with several different choices of nominal operating condition in order to narrow down the critical region of parameter values. After this smaller region has been determined, the region is meshed by a fine grid and a frozen parameter (LTI) synthesis is performed for each grid point. The result of such a study is presented in Figure 11 from which it can be observed that a high level of performance can be obtained for all frozen parameter values except along a line through the operating region for which no stabilising controller exists.

To understand further why it is not possible to stabilise the plant for all frozen parameters we can study the quasi-LPV formulation of the aerodynamics. Recalling the parameterisation of the aerodynamic torque in (14) we have a model of the aerodynamics which can be described by

$$Q_a = Q_a^\beta(v, \beta, \omega_r) \cdot \beta + Q_a^v(v, \beta, \omega_r) \cdot v + Q_a^\omega(v, \beta, \omega_r) \cdot \omega_r$$

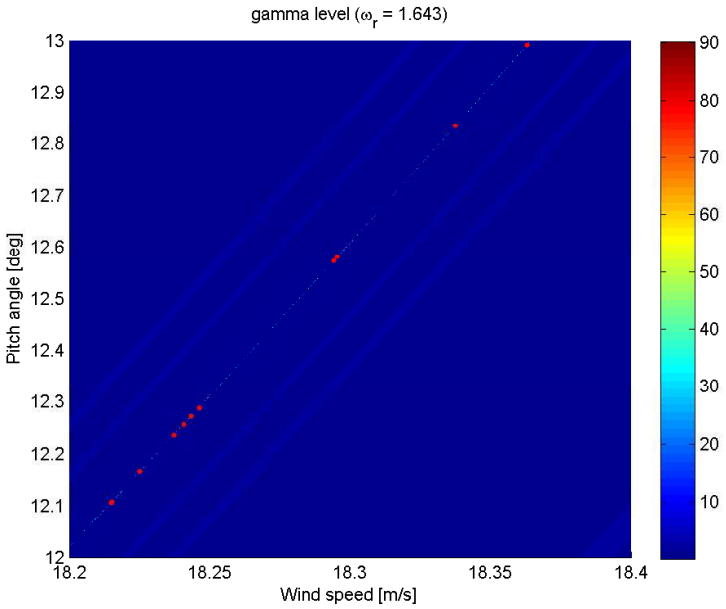


Figure 11: Performance analysis for frozen parameters with rotor and generator speed parameters fixed at the rated speed. Red dots indicate the grid points where no stabilising controller existed for the frozen parameters. Note that the parameters where synthesis failed occurs at a line through the domain.

which is equivalent to the nonlinear aerodynamics model, i.e. no linearisation is performed. The three coefficients, Q_a^β, Q_a^v, and Q_a^ω are illustrated in Figure 12 for the considered parameter values from which it can be observed that Q_a^β is zero along the line of operating conditions where the design failed whereas Q_a^v and Q_a^ω remain nonzero.

It essentially means that from the point of view of frozen parameters, the pitch has no effect on the power into the wind turbine. This is highly critical in the high wind speed region in which the pitch is the main control signal for tracking the specified generator speed reference. For the frozen parameter synthesis it is therefore very important that Q_a^β is non-zero for all operating conditions in the considered parameter region.

The parameter values for which Q_a^β is zero can from studying (14) be controlled by choosing k_4^β, k_5^β, k_7^β, and k_8^β so that

$$\begin{bmatrix} q_4 v^2 & q_5 \omega_r v & q_7 \beta v^2 & q_8 \omega_r \beta v \end{bmatrix} \begin{bmatrix} k_4^\beta \\ k_5^\beta \\ k_7^\beta \\ k_8^\beta \end{bmatrix} + q_9 \beta^2 v^2 \neq 0$$

for all parameter values in the polytope. This can for example be done by choosing all four variables to be zero. However by this choice we still obtain open loop dynamics with frozen parameters which significantly differ from what will be obtained by linearising the model at the considered operating condition.

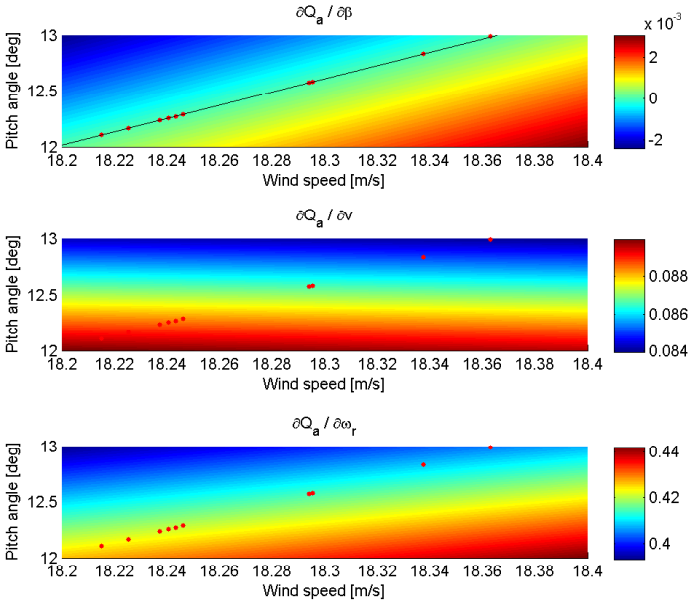


Figure 12: Analysis of aerodynamic torque for frozen parameters with rotor and generator speed parameters fixed at the rated speed. The red marks indicate where the frozen parameter synthesis fails, and the black line shows where $\frac{\partial Q_a}{\partial \beta} = 0$. Observe that the synthesis failed when $\frac{\partial Q_a}{\partial \beta} = 0$.

Now recall the previous discussion about the performance level with frozen parameters being the best possible level obtainable by LPV control. It is therefore of great importance that the parameterisation of Q_a^β, Q_a^v, and Q_a^ω is chosen so that the quasi-LPV model with frozen parameters to a large extent resembles models that can be obtained by linearisation at each operating condition. Otherwise it might be difficult to obtain a performance level similar to what can be obtained locally by LTI controllers.

By using partial derivatives of Q_a in (13) with respect to β, v, and ω_r it is then evident that k_i^β, k_i^v, and k_i^ω must be chosen to satisfy

$$
\begin{aligned}
&k_4^\beta\, q_4\, v^2 + k_5^\beta\, q_5\, v\, \omega_r + k_7^\beta\, q_7\, \beta\, v^2 + k_8^\beta\, q_8\, \beta\, v\, \omega_r + q_9\, \beta^2\, v^2 = \\
&\qquad q_4 v^2 + q_5 v \omega_r + q_6 {\omega_r}^2 + 2q_7 \beta v^2 + 2q_8 \beta v \omega_r + 3q_9 \beta^2 v^2 \\
&q_1\, v + k_2^v\, q_2\, \omega_r + k_4^v\, q_4\, \beta\, v + k_5^v\, q_5\, \beta\, \omega_r + k_7^v\, q_7\, \beta^2\, v = \\
&\qquad 2q_1 v + q_2 \omega_r + 2q_4 \beta v + q_5 \beta \omega_r + 2q_7 \beta^2 v + q_8 \beta^2 \omega_r + q_8 \beta^2 \omega_r + 2q_9 \beta^3 v \\
&k_2^\omega\, q_2\, v + q_3\, \omega_r + k_5^\omega\, q_5\, \beta\, v + q_6\, \beta\, \omega_r + k_8^\omega\, q_8\, \beta^2\, v = \\
&\qquad q_2 v + 2q_3 \omega_r + q_5 \beta v + 2q_6 \beta \omega_r + q_8 \beta^2 v \\
&1 = k_2^v + k_2^\omega = k_4^\beta + k_4^v = k_5^\beta + k_5^v + k_5^\omega = k_7^\beta + k_7^v = k_8^\beta + k_8^\omega
\end{aligned}
$$

for all possible values of v, β, and ω_r in the considered operating region. It is not expected that the above set of equations has a solution, however the gap between the linearised dynamics and the quasi-LPV formulation with frozen parameters can be minimised by the use of optimisation algorithms.

4. Conclusions

In this report it has been described how to determine the parameter range for the scheduling variables of an LPV model. Also a description of the considered quasi-LPV model is presented and it is shown how to derive an LFT of the model.

Concerning the design of an LPV controller on the basis of this quasi-LPV model, it has been concluded that the controllability and observability conditions of the frozen parameter dynamics are essential for the performance level that can be obtained by the methodology. This basically means that it is not sufficient that the nonlinear dynamics are controllable and observable but also that the frozen parameter dynamics need to have this property.

The performance level of the LPV controller is limited by the level obtainable with frozen parameter dynamics. It is therefore suggested to use a quasi-LPV formulation with which the frozen parameter dynamics resemble as much as possible the dynamics obtainable by linearising the nonlinear dynamics. This can be done by using the tuning variables k_i^β, k_i^v, and k_i^ω to minimise the gap between the two representations over the range of parameter values.

For a realistic size of the parameter region it is not expected to have a sufficient level of resemblance between the frozen parameter dynamics and the linearised dynamics. This makes it difficult to design controllers with a performance level comparable to what can be obtained by LTI controllers at each operating condition. It is therefore suggested to consider an LPV formulation based on linearised dynamics along a trajectory of operating conditions instead of the quasi-LPV formulation.

REFERENCES

[1] IEC. *IEC 61400-1*. Int. Standard, 2005.

[2] D. J. Leith and W. E. Leithead. Survey of gain-scheduling analysis and design. *Int. J. of Control*, 73(11):1001–1025, 2000. http://dx.doi.org/10.1080/002071700411304.

[3] W. J. Rugh and J. S. Shamma. Research on gain scheduling. *Automatica*, 36:1401–1425, 2000. http://dx.doi.org/10.1016/S0005-1098(00)00058-3.

[4] C. W. Scherer. *Robust Mixed Control and LPV Control with Full Block Scalings*. SIAM, 2000.

[5] C. W. Scherer. LPV control and full block multipliers. *Automatica*, 37:361–375, 2001. http://dx.doi.org/10.1016/S0005-1098(00)00176-X.

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

Report F

Linear Parameter Varying Controller Design for Wind Turbines Using Linear Fractional Transformations

Contents

Technical report.

Linear parameter varying controller design for wind turbines using linear fractional transformations

Kasper Zinck Østergaard
Turbine Control and Operation
Loads, Aerodynamics and Control
Vestas Wind Systems A/S

Per Brath
Turbine Control and Operation
Loads, Aerodynamics and Control
Vestas Wind Systems A/S

Jakob Stoustrup
Automation and Control
Department of Electronic Systems
Aalborg University

Abstract. The control of wind turbines is a challenging task which for most loops require the designer to consider nonlinear couplings caused by for example the aerodynamics.
In particular for the speed control of wind turbines in full load operation it is required to keep the generator speed variations within a given range which is specified in absolute terms. This essentially means that a different control effort is required for lower wind speeds (close to rated power) when comparing with high wind speeds. As a result not only the wind turbine model is nonlinear, but also the performance weights must be scheduled on wind speed in order to satisfy the generator speed bounds with as little control effort as possible.
Gain-scheduling has previously shown good results for control of wind turbines and it has therefore been decided to focus on linear parameter varying control which is a systematic way of designing gain-scheduled controllers.
When combining the nonlinearities of the model with the gain-scheduled performance weights a model with polynomial parameter dependency is obtained. For this kind of parameter dependency the linear fractional transformation approach for controller design appears appropriate.
The main part of the report then investigates the applicability of an available algorithm for designing linear parameter varying controllers on the basis of a linear fractional representation. It is concluded that the method is not at the moment matured enough – from a numerical point of view – for the considered application. Several issues are pointed out which need further research in order to make the algorithm numerically stable and give a satisfactory performance level.

1. Introduction

It is well-known that the aerodynamics of a wind turbine is non-linear and that LTI controllers are insufficient for getting satisfactory closed loop performance. Traditionally the controllers have been obtained by designing a set of LTI controllers along a nominal trajectory of operating conditions and then afterwards interconnecting the controllers to obtain what is denoted a gain-scheduled controller. This approach has the underlying risk that the design is based on LTI investigations do not not take into account that the operating conditions will vary in time. In the worst case this means that the closed loop can be unstable in real life applications because the operating conditions in fact do vary in time. The design of linear parameter varying (LPV) controllers is a systematic design method for gain-scheduled controllers which does take into account the variations of the operating condition.

Several approaches have been undertaken for the design of LPV controllers for the operation of wind turbines [3, 7, 10, 6]. For these applications a simple assumption of affine parameter dependency has been assumed which is realistic when considering only the nonlinearities in the aerodynamics for full load operation. However, to get a reasonable performance it is necessary to vary the trade-off over the trajectory of operating conditions between tracking a generator speed reference and pitch activity. This is

done to make the generator speed variations have a similar scale in absolute terms. The performance weights must therefore also be gain scheduled and the resulting weighted open loop model has a parameter dependency that can not with reasonable precision be described by affine parameter dependency on a convex domain.

As a consequence of this complicated parameter dependency a more general design method has been applied for the controller design. This method assumes a rational parameter dependency on a convex domain which can easily enclose the parameter dependency. The design method is on the other hand not as matured (from a numerical point of view) as the previously applied methods and in the report a number of issues are pointed out for future investigations in order to get a suitable method for LPV controller design for wind turbines.

In Section 2 the considered model is described followed by the main part of the report in Section 3 about the controller design. Finally the conclusions and recommendations for future work is given in Section 4.

2. Model

The purpose of the investigation is to understand the applicability of an LFT based algorithm for designing LPV controllers. It has therefore been decided to focus on a very simplistic wind turbine model. The design model will be an interconnection of a pitch system with a drive train model through a static aerodynamic model. In the following, the nominal trajectory of operating conditions are determined and then each of the three model components are identified.

2.1. Operating region

With the wind turbine operating in full load we have that the nominal rotor speed, generator speed, and active power takes constant, known values denoted rated speed and power. This means that operating conditions can be described by identifying the relation between effective wind speed and pitch angle. The loading on the drive train and electrical components is similar for all operating conditions in full load which means that a reasonable assumption is that the mechanical and electrical losses are equivalent along the trajectory of operating conditions.

This assumption together with a constant rated electrical power and rotor speed means that the power extracted from the kinetic energy in the wind remains constant along the trajectory of operating conditions. Because of this the variables determining the operating condition can be studied by considering only the aerodynamic model

$$P_{aero} = \frac{1}{2}\,\rho\,\pi\,R^2\,v^3\,c_P(\lambda,\beta) \quad , \quad \lambda = \frac{\omega_r\,R}{v} \tag{1}$$

where ρ is the air density, R is the rotor radius, v is the effective wind speed, ω_r, is the rotor rotational speed, and β is the pitch angle. By rearranging (1) we can calculate the nominal aerodynamic efficiency, c_P, to give nominal power production with nominal rotor speed. Such a nominal value of c_P is illustrated in Figure 1 as a function of effective wind speed.

When comparing this nominal value for c_P with its functional dependency on pitch angle and tip speed ratio a static relation between nominal pitch angle and effective wind speed as as illustrated in Figure 2.

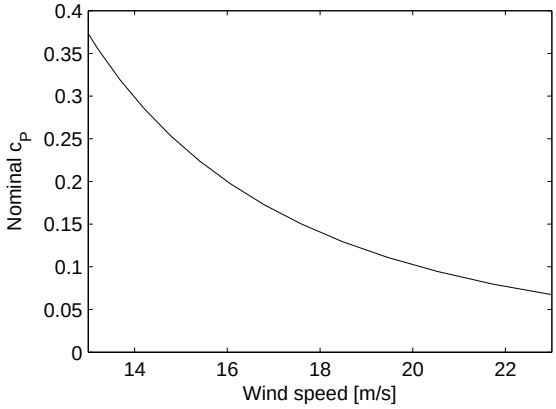


Figure 1: Nominal aerodynamic efficiency as function of effective wind speed.

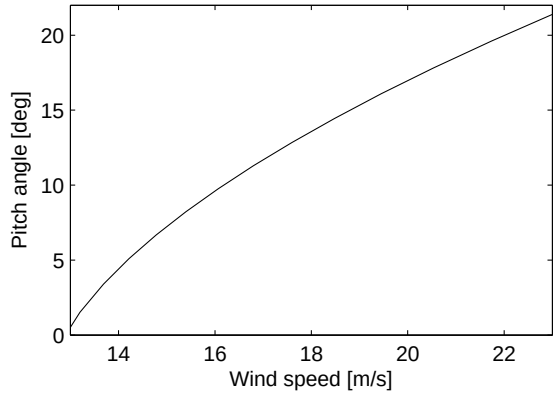


Figure 2: Nominal relation between pitch angle and effective wind speed in full load operation.

2.2. Aerodynamics

The goal of this section is to linearise the aerodynamic model (1) along the trajectory of operating conditions determined by Figure 2 in order to get a model in an input-output form that is rational in the scheduling parameter, v, so that a linear fractional transformation (LFT) can be performed. For interconnection with the drive train model the aerodynamic model will be described as a static model from wind speed, v, pitch angle, β, and rotor speed, ω_r to aerodynamic torque, Q_a according to (2).

$$Q_a \approx Q_a(\bar{v}) + \frac{\partial Q_a}{\partial v}\Big|_{\bar{v}} (v - \bar{v}) + \frac{\partial Q_a}{\partial \beta}\Big|_{\bar{v}} (\beta - \bar{\beta}(\bar{v})) + \frac{\partial Q_a}{\partial \omega_r}\Big|_{\bar{v}} (\omega_r - \bar{\omega}_r) \qquad (2)$$

From a numerical point of view it is advantageous to calculate the partial derivatives of c_P and use them in the calculation of the partial derivatives of Q_a because the values of c_P are only known in table form with a limited grid density. To be able to do this, the partial derivatives have been rewritten in terms of the partial derivatives of c_P as shown in (3).

$$\frac{\partial Q_a}{\partial v} = \frac{1}{2}\,\rho\,\pi\,R^2\,\frac{1}{\omega_r}\left(3\,v^2\,c_P + v^3\,\frac{\partial c_P}{\partial \lambda}\,\frac{\partial \lambda}{\partial v}\right)$$
$$= \frac{1}{2}\,\rho\,\pi\,R^2\,\frac{1}{\omega_r}\left(3\,v^2\,c_P - v^3\,\frac{\partial c_P}{\partial \lambda}\,\frac{R\,\omega_r}{v^2}\right)$$
$$= \frac{1}{2}\,\rho\,\pi\,R^3\,v\left(3\,\frac{1}{\lambda}\,c_P - \frac{\partial c_P}{\partial \lambda}\right) \tag{3a}$$

$$\frac{\partial Q_a}{\partial v} = \frac{1}{2}\,\rho\,\pi\,R^2\,v^3\left(-\frac{1}{\omega_r^2}c_P + \frac{1}{\omega_r}\,\frac{\partial c_P}{\partial \lambda}\,\frac{\partial \lambda}{\partial \omega_r}\right) = \frac{1}{2}\,\rho\,\pi\,R^2\,\frac{v^3}{\omega_r}\left(\frac{\partial c_P}{\partial \lambda}\,\frac{R}{v} - \frac{1}{\omega_r}\,c_P\right)$$
$$= \frac{1}{2}\,\rho\,\pi\,R^3\,\frac{v^2}{\omega_r}\left(\frac{\partial c_P}{\partial \lambda} - \frac{1}{\lambda}\,c_P\right) \tag{3b}$$

$$\frac{\partial Q_a}{\partial v} = \frac{1}{2}\,\rho\,\pi\,R^2\frac{v^3}{\omega_r}\,\frac{\partial c_P}{\partial \beta} \tag{3c}$$

The partial derivatives of c_P are illustrated in Figure 3 along the nominal trajectory, and they are then used to calculate the partial derivatives of Q_a shown by the black lines in Figure 4. These three functions for the partial derivatives are approximated by a quadratic function by using a least squares fit, i.e.

$$\left.\frac{\partial Q_a}{\partial v}\right|_{\bar{v}} \approx a_2^{Qv}\,\bar{v}^2 + a_1^{Qv}\,\bar{v} + a_0^{Qv} \tag{4a}$$
$$\left.\frac{\partial Q_a}{\partial \omega_r}\right|_{\bar{v}} \approx a_2^{Q\omega}\,\bar{v}^2 + a_1^{Q\omega}\,\bar{v} + a_0^{Q\omega} \tag{4b}$$
$$\left.\frac{\partial Q_a}{\partial \beta}\right|_{\bar{v}} \approx a_2^{Q\beta}\,\bar{v}^2 + a_1^{Q\beta}\,\bar{v} + a_0^{Q\beta} \tag{4c}$$

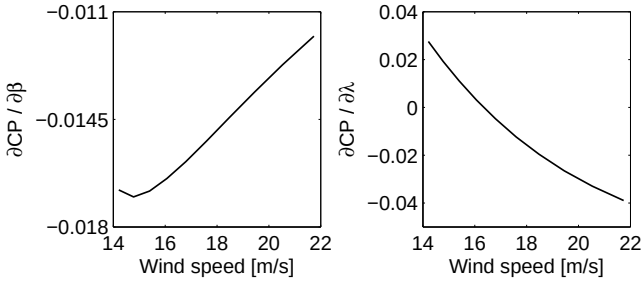


Figure 3: Partial derivatives of aerodynamic efficiency with respect to pitch angle and tip speed ratio.

Then the aerodynamic torque can be described by a quadratic function in the effective wind speed

$$Q_a \approx \left(\begin{bmatrix} a_2^{Qv} & a_2^{Q\omega} & a_2^{Q\beta} \end{bmatrix} v^2 + \begin{bmatrix} a_1^{Qv} & a_1^{Q\omega} & a_1^{Q\beta} \end{bmatrix} v + \begin{bmatrix} a_0^{Qv} & a_0^{Q\omega} & a_0^{Q\beta} \end{bmatrix}\right) \begin{bmatrix} v \\ \omega_r \\ \beta \end{bmatrix}$$

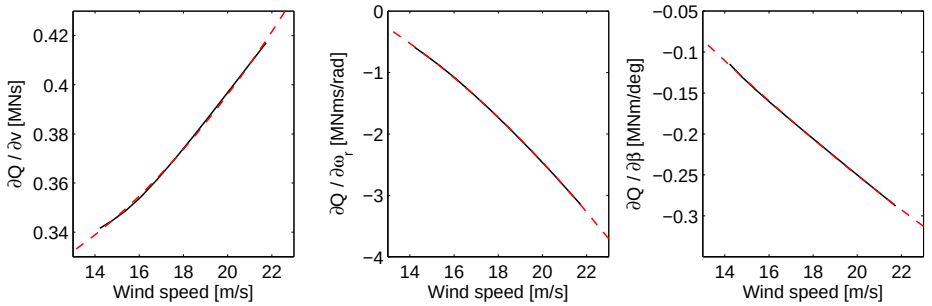


Figure 4: Partial derivatives of aerodynamic torque along nominal trajectory of operating conditions. Black lines show actual value. Red, dashed lines show second order approximation.

which can be described by an LFT by the following observations: Choose $w_1 = v\, z_2$ and $z_1 = \begin{bmatrix} a_2^{Qv} & a_2^{Q\omega} & a_2^{Q\beta} \end{bmatrix}$. Then

$$Q_a \approx \left(\left(w_1 + \begin{bmatrix} a_1^{Qv} & a_1^{Q\omega} & a_1^{Q\beta} \end{bmatrix} \right) v + \begin{bmatrix} a_0^{Qv} & a_0^{Q\omega} & a_0^{Q\beta} \end{bmatrix} \right) \begin{bmatrix} v \\ \omega_r \\ \beta \end{bmatrix}$$

Now by choosing $w_1 = v$ and $z_1 = w_1 + \begin{bmatrix} a_1^{Qv} & a_1^{Q\omega} & a_1^{Q\beta} \end{bmatrix}$ we have the following LFT of the model

$$\begin{bmatrix} z_1 \\ z_1 \\ Q_a \end{bmatrix} = \begin{bmatrix} 0 & 0 & a_2^{Qv} & a_2^{Q\omega} & a_2^{Q\beta} \\ 0 & 1 & a_1^{Qv} & a_1^{Q\omega} & a_1^{Q\beta} \\ 1 & 0 & a_0^{Qv} & a_0^{Q\omega} & a_0^{Q\beta} \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ v \\ \omega_r \\ \beta \end{bmatrix}, \quad \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = \begin{bmatrix} v & 0 \\ 0 & v \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \tag{5}$$

2.3. *Drive train*

The drive train dynamics are typically described by two inertias interconnected by a spring and damper. In this study it is assumed that the oscillations in the drive train are minimised by another control loop using the generator reaction torque. With this assumption the drive train model can be simplified to a first order system of the form

$$J\, \dot{\omega}_g(t) = \frac{1}{N} Q_a(t) - Q_g(t) - B_r\, \omega_g(t)$$

where $\omega_g(t)$ is the generator speed, $Q_a(t)$ is the aerodynamic torque, $Q_g(t)$ is the generator reaction torque, J is the inertia of the rotating part of the wind turbine, N is the gearing ratio, and B_r is a linear friction constant.

In this design formulation we do not consider the generator reaction torque which means that $Q_g(t)$ can be considered a constant determined by the nominal generator speed and power production.

2.4. *Pitch system*

For the pitch system a simplistic nonlinear second order model is assumed from control voltage, u, to pitch angle, β. This model lumps the hydraulic, mechanical and electrical

dynamics of the pitch system into a model of the form (6) which includes the main characteristics when seen from the mechanical point of view. The model includes a static nonlinearity f which combines the nonlinear valve characteristics with the mapping from piston movement to blade rotation. Furthermore the model has two parameters: T_β which is the time constant of the system, and T_{delay} which is a combination of computational delays and a hydraulic time delay.

$$T_\beta \, \ddot{\beta}(t) + \dot{\beta}(t) = f\big(u(t - T_{delay})\big) \tag{6}$$

In the leftmost illustration of Figure 5 the nonlinearity in the pitch system is illustrated. Most of this nonlinearity is taken care of by a nonlinear static controller with the open loop shown in the illustration in the middle. The open loop gain of this controller interconnected with the nonlinear part of the model is then illustrated in the right most illustration. With a pitch error of ± 2.5 deg the linearisation shown in Figure 5 is assumed adequate and the closed loop is then as shown in (7) with the slope of the red dashed line in Figure 5 denoted K_β.

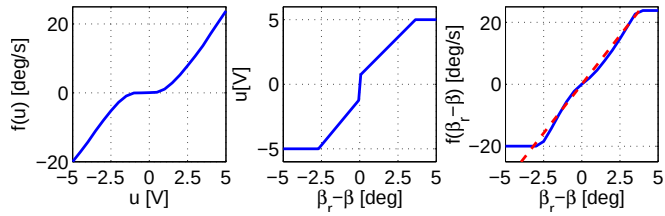


Figure 5: Linearisation of nonlinear part of pitch system. Left: Nonlinear gain, f, in pitch model. Middle: Nonlinear controller gain, P. Right: Open loop interconnection of model gain with controller gain, $P \cdot f$. Observe that controller linearises model nonlinearity for small tracking errors with open loop gain, $P \cdot f \approx K_\beta$.

$$\beta = \frac{K_\beta}{s(T_\beta s + 1) + K_\beta} \cdot \beta_r \tag{7}$$

3. Controller design

The purpose of this controller design is to get an understanding of a design algorithm for obtaining linear parameter varying controller that can guarantee a performance level measured by the induced $\mathcal{L}_2$ norm. For this application we investigate the design of a speed controller for full load operation. For simplicity, no structural oscillations (for example drive train torsion and tower movement) will be considered, i.e. only the trade-off between control effort in the pitch system and the tracking performance.

This control formulation is graphically illustrated in Figure 6 from which it can be seen that the target is to design a LPV controller with tracking error (and its integral) as inputs and pitch reference as output. The performance is then measured by the energy gain (induced $\mathcal{L}_2$ norm) from the wind input to integral of tracking error and pitch reference – both weighted by gain-scheduled weighting functions W_ω and W_β. Note also that the aerodynamics is described by an LPV model.

The weighting functions have been determined through a LTI investigation along the trajectory of operating conditions. For the tracking of the generator speed it is important

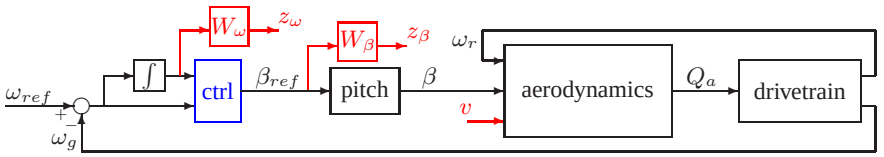


Figure 6: Block diagram of considered control problem. Blue indicates the LPV controller to be designed and red indicates the considered performance channels and performance weights.

to have good performance at low frequencies. At higher frequencies the performance is not as important and should have less weight in order to obtain robustness and reduce control effort. This indicates that the weighting function should be a low pass filter to put high emphasis on low frequency components and less emphasis on high frequency components. From the LTI investigations it has been observed that a desired performance level can be obtained with a frequency independent scaling of the form:

$$W_\omega(v) = K_2^\omega\, v^2 + K_1^\omega\, v + K_0^\omega$$

For the control effort it is important to punish high frequency movement because this kind of movement will quickly wear out the pitch system. Also by reducing the high frequency control signal it is less likely to excite under-modelled high frequency dynamics. To obtain this it has been decided to use a third order high pass filter of the form*

$$W_\beta(s, v) = K^\beta(v) \cdot \left(\frac{T^\beta(v)\, s + 1}{\epsilon\, s + 1} \right)^3$$

where

$$K_\beta(v) = K^\beta\, v + K_0^\omega \quad , \quad T_\beta(v) = T^\beta\, v + T_0^\omega \quad ,$$

ϵ is a small positive number, s is the Laplace variable and v is the scheduling variable (effective wind speed). The three gain scheduled parameters for the two weights are illustrated in Figure 7.

3.1. Synthesis with frozen parameters

To assess the performance level of the closed loop with the LPV controller it has been decided to use LTI controllers at a number of analysis points along the trajectory of nominal operating conditions. In Table 1 a result from such a LTI investigation is presented and it can be observed that the performance weights have been chosen to make the performance level similar for all operating conditions. By doing this it is expected that the LPV controller design will put equal emphasis on all operating conditions.

3.2. LPV synthesis

The design of the LPV controller has been done by using an algorithm (and design tool) developed by prof. Carsten Scherer [8, 9] – the tool has been slightly modified to take into account the gain scheduled performance weights. The applied procedure

*The authors are aware of the abuse of notation.

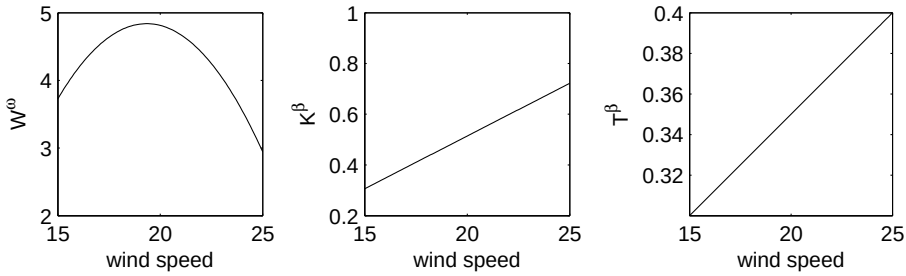


Figure 7: Parameters for performance weights W_ω and W_β.

wind speed	15	17	19	21	23	25
performance level	0.987	1.008	1.025	1.028	1.016	0.990

Table 1: LTI synthesis result at selected number of operating conditions.

for designing an LPV controller is a two-step algorithm which in the first step finds a performance level and associated multipliers. The second step is then to construct the controller on the basis of the desired performance level and multipliers. In this section the essentials of the first step in the algorithm is described and the obtainable performance level is given.

For the controller synthesis we consider a weighted open loop model of the form

$$\begin{bmatrix} \dot{x} \\ z_p \\ y \end{bmatrix} = \begin{bmatrix} A(v) & B_p(v) & B(v) \\ C_p(v) & D_p(v) & E(v) \\ C(v) & F(v) & 0 \end{bmatrix} \begin{bmatrix} x \\ w_p \\ u \end{bmatrix} \tag{8}$$

with x as the state vector, w_p and z_p the performance inputs and outputs, and u and y the control inputs and outputs. The open loop matrices are assumed to have a rational parameter dependency which means that it can be represented by

$$\begin{bmatrix} \dot{x} \\ z_u \\ z_p \\ y \end{bmatrix} = \begin{bmatrix} A & B_u & B_p & B \\ C_u & D_u & D_{up} & E_u \\ C_p & D_{pu} & D_p & E_p \\ C & F_u & F_p & 0 \end{bmatrix} \begin{bmatrix} x \\ w_u \\ w_p \\ u \end{bmatrix} \quad , \quad w_u = \Delta(v)\, z_u \tag{9}$$

where $\Delta(v)$ is a linear function of δ_v.

The closed loop performance will be measured by the energy gain (induced $\mathcal{L}_2$ norm). This means that the target of this first step is to identify a performance level γ so that

$$\int_0^\infty z_p(t)^T z_p(t)\, dt < \gamma^2 \int_0^\infty w_p(t)^T w_p(t)\, dt \tag{10}$$

for all non-zero performance inputs, $w(t)$, with finite energy into the closed loop interconnection of (8) with the designed controller. The controller synthesis in the applied

algorithm is based on a method for analysing the performance level for closed loop interconnection. If we assume that a linear parameter varying controller has been determined with rational parameter dependency, then we have a closed loop interconnection of the form

$$\begin{bmatrix} \dot{x} \\ z_p \end{bmatrix} = \begin{bmatrix} \mathcal{A}(v) & \mathcal{B}(v) \\ \mathcal{C}(v) & \mathcal{D}(v) \end{bmatrix} \begin{bmatrix} x \\ w_p \end{bmatrix} \tag{11}$$

which can be described by the LFT

$$\begin{bmatrix} \dot{x} \\ z_u \\ z_p \end{bmatrix} = \begin{bmatrix} \mathcal{A} & \mathcal{B}_u & \mathcal{B}_p \\ \mathcal{C}_u & \mathcal{D}_u & \mathcal{D}_{up} \\ \mathcal{C}_p & \mathcal{D}_{pu} & \mathcal{D}_p \end{bmatrix} \begin{bmatrix} x \\ w_u \\ w_p \end{bmatrix} \quad , \quad w_u = \Delta(v)\, z_u \tag{12}$$

In [2] it was then shown that the performance level, γ, measured by (10) can be determined by determining a positive definite matrix function $\mathcal{X}(v)$ for which a linear matrix inequality (LMI) is satisfied

$$\begin{bmatrix} I & 0 \\ \mathcal{A}(v) & \mathcal{B}(v) \\ 0 & I \\ \mathcal{C}(v) & \mathcal{D}(v) \end{bmatrix}^T \begin{bmatrix} \dot{\mathcal{X}}(v) & \mathcal{X}(v) & & \\ \mathcal{X}(v) & 0 & & \\ & & -\gamma I & 0 \\ & & 0 & \frac{1}{\gamma} I \end{bmatrix} \begin{bmatrix} I & 0 \\ \mathcal{A}(v) & \mathcal{B}(v) \\ 0 & I \\ \mathcal{C}(v) & \mathcal{D}(v) \end{bmatrix} \prec 0 \tag{13}$$

for each possible parameter and parameter rate of variation in the range of possible operating conditions. This requires that we solve an infinite number of LMIs which is not practically possible. In [8, 9] it was then shown that if we allow for arbitrary fast parameter variations and (which might be conservative) enclose the range of parameter values by a convex polytope the analysis problem can be described by a finite number of LMIs. We must then find positive definite $\mathcal{X}$ and a symmetric multiplier

$$\mathcal{P} = \begin{bmatrix} \mathcal{Q} & \mathcal{S} \\ \mathcal{S}^T & \mathcal{R} \end{bmatrix}$$

for which

$$\begin{bmatrix} \Delta(v) \\ I \end{bmatrix}^T \begin{bmatrix} \mathcal{Q} & \mathcal{S} \\ \mathcal{S}^T & \mathcal{R} \end{bmatrix} \begin{bmatrix} \Delta(v) \\ I \end{bmatrix} \succ 0 \tag{14a}$$

for all parameter values at the vertices of the polytope and

$$\begin{bmatrix} I & 0 & 0 \\ \mathcal{A} & \mathcal{B}_u & \mathcal{B}_p \\ 0 & I & 0 \\ \mathcal{C}_u & \mathcal{D}_u & \mathcal{D}_{up} \\ 0 & 0 & I \\ \mathcal{C}_p & \mathcal{D}_{pu} & D_p \end{bmatrix}^T \begin{bmatrix} 0 & \mathcal{X} & & & & \\ \mathcal{X} & 0 & & & & \\ & & \mathcal{Q} & \mathcal{S} & & \\ & & \mathcal{S}^T & \mathcal{R} & & \\ & & & & -\gamma I & 0 \\ & & & & 0 & \frac{1}{\gamma} I \end{bmatrix} \begin{bmatrix} I & 0 & 0 \\ \mathcal{A} & \mathcal{B}_u & \mathcal{B}_p \\ 0 & I & 0 \\ \mathcal{C}_u & \mathcal{D}_u & \mathcal{D}_{up} \\ 0 & 0 & I \\ \mathcal{C}_p & \mathcal{D}_{pu} & D_p \end{bmatrix} \prec 0 \tag{14b}$$

for the closed loop system given in (12).

Now, what was presented in the above concerned the performance analysis for a closed loop synthesis when given the controller. The main focus is on the other hand

to design a controller to satisfy a performance level, γ, of the closed loop. This is essentially done by inserting the controller variables into the analysis LMIs which results in a set of nonlinear matrix inequalities in the controller variables, $\mathcal{X}$ and P for the LFT approach. The controller variables can however easily be removed from the synthesis LMIs as shown in [2, 8, 9] for the two different algorithms. For the LFT approach this results in the synthesis problem described in Theorem 18.

Theorem 18 (Proof provided in [9])**.** *There is a LPV controller in LFT form for the open loop system (9) which satisfies a performance level γ according to (10) if there exist variables: $X = X^T$, $Y = Y^T$, $Q = Q^T$, S, $R = R^T$, $\tilde{Q} = \tilde{Q}^T$, $\tilde{S}$, $\tilde{R} = \tilde{R}^T$ for which*

$$Q \prec 0 \quad , \quad \begin{bmatrix} \Delta(v) \\ I \end{bmatrix}^T \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \begin{bmatrix} \Delta(v) \\ I \end{bmatrix} \succ 0$$

$$\tilde{R} \succ 0 \quad , \quad \begin{bmatrix} I \\ -\Delta(v)^T \end{bmatrix}^T \begin{bmatrix} \tilde{Q} & \tilde{S} \\ \tilde{S}^T & \tilde{R} \end{bmatrix} \begin{bmatrix} I \\ -\Delta(v)^T \end{bmatrix} \prec 0$$

for all parameter values in the range of operating conditions and

$$\begin{bmatrix} Y & I \\ I & X \end{bmatrix} \succ 0$$

$$\Phi^T \begin{bmatrix} \star \\ \star \\ \star \\ \star \\ \star \\ \star \end{bmatrix}^T \begin{bmatrix} 0 & X & & & & & \\ X & 0 & & & & & \\ & & Q & S & & & \\ & & S^T & R & & & \\ & & & & -\gamma I & 0 & \\ & & & & 0 & \frac{1}{\gamma} I & \end{bmatrix} \begin{bmatrix} I & 0 & 0 \\ A & B_u & B_p \\ 0 & I & 0 \\ C_u & D_u & D_{up} \\ 0 & 0 & I \\ C_p & D_{pu} & D_p \end{bmatrix} \Phi \prec 0$$

$$\Psi^T \begin{bmatrix} \star \\ \star \\ \star \\ \star \\ \star \\ \star \end{bmatrix}^T \begin{bmatrix} 0 & Y & & & & & \\ Y & 0 & & & & & \\ & & \tilde{Q} & \tilde{S} & & & \\ & & \tilde{S}^T & \tilde{R} & & & \\ & & & & -\frac{1}{\gamma} I & 0 & \\ & & & & 0 & \gamma I & \end{bmatrix} \begin{bmatrix} -A^T & -C_u^T & -C_p^T \\ I & 0 & 0 \\ -B_u^T & -D_u^T & -D_{pu}^T \\ 0 & I & 0 \\ -B_p^T & -D_{up}^T & -D_p^T \\ 0 & 0 & I \end{bmatrix} \Psi \succ 0$$

where Φ and Ψ are bases for the null spaces for $\begin{bmatrix} C & F_u & F_p \end{bmatrix}$ and $\begin{bmatrix} B^T & E_u^T & E_p^T \end{bmatrix}$ respectively. Concerning the notation: $\star$ indicate terms that are induced by symmetry.

We then interconnect the model from Section 2 with the performance weights and insert the weighted open loop system matrices into the formulation in Theorem 18 from which we get a performance level, $\gamma = 35.2$ which is a substantial decrease in performance.

This reduction in performance level is expected to be caused by the design algorithm and the underlying assumptions, i.e. not because of intrinsic properties of the design problem. There can be many reasons for this decrease in performance level. First of all the assumption that the scheduling variable can vary arbitrarily fast might be restrictive and thereby conservative. For this application, an average level for the effective wind speed is used as scheduling variable. In nominal operation this variable will change

rather slowly, however large and sudden variations might occur in extreme weather conditions, e.g. wind gusts. These special cases will naturally also have a limit as to how fast the wind speed level can change – from a physical point of view. Because of this it is suggested to investigate methods for including a limit on the rate of variation in order to understand how conservative the assumption is.

Another possible cause of conservatism is that the restrictions $Q \prec 0$ and $\tilde{R} \prec 0$ are included for technical reasons in order to reduce the number of LMIs from checking the whole parameter region to only checking the vertices. It is however easily checked that this restriction is not very conservative in the specific application. If we disregard the two restrictions and perform another synthesis we obtain a performance level, $\gamma = 35.1$, which is very similar to the level with the restriction.

Finally the reduction in performance level from LTI synthesis to LPV synthesis might come from numerical difficulties in the optimisation problem. In practice the LMIs in Theorem 18 are transformed into a convex optimisation problem (semidefinite programming) which is solved by a numerical algorithm. Essentially this algorithms minimises γ subject to the non-negativeness of a 83×83 matrix function which is affine in its 253 variables. Such an optimisation problem is highly nonlinear and from previous studies it has been observed that it is very dependent on the numerical conditioning of the problem formulation and for example the choice of state space realisation has a huge impact on the obtainable performance level. The numerics have been tested for a few different realisations without any positive result, but it is still not possible to rule out the numerics as a cause for the performance decrease. In order to do so a more thorough and systematic investigation of the choice of realisation and other numerical issues need to be performed.

3.3. Construction of LPV controller

In Section 3.2 it was shown how to calculate a possible performance level, γ and associated multiplier, X, Y, P, and $\tilde{P}$. The optimal performance level from the design algorithm was 35 times worse than what is obtainable locally with an LTI controller. A performance decrease in that scale is not expected to be acceptable, but it is not yet understood if the cause is due to numerical difficulties or because of the assumption of arbitrary fast parameter variations. Because of this it has been decided to do an investigation of the controller construction to understand the controller behaviour and the numerical properties of this algorithm.

In the step from the analysis LMIs (14) with introduced controller variables to the synthesis LMIs in Theorem 18 the controller variables were eliminated. For the implementation we naturally need the controller variables and for doing this we can observe that the proof for the elimination of variables is constructive [9, 5]. This means that we can in fact construct the controller from the variables provided by solving the synthesis problem in Theorem 18.

The first step in the construction procedure is to extend the multiplier X, Y, P, and $\tilde{P}$ to the respective multiplier used for the closed loop analysis, i.e. we need to determine the matrices indicated by $\star$ to satisfy

$$\mathcal{X} = \begin{bmatrix} X & \star \\ \star & \star \end{bmatrix} \quad , \quad \mathcal{X}^{-1} = \begin{bmatrix} Y & \star \\ \star & \star \end{bmatrix} \tag{15a}$$

$$\mathcal{P} = \begin{bmatrix} Q & \star & S & \star \\ \star & \star & \star & \star \\ S^T & \star & R & \star \\ \star & \star & \star & \star \end{bmatrix} \quad , \quad \mathcal{P}^{-1} = \begin{bmatrix} \tilde{Q} & \star & \tilde{S} & \star \\ \star & \star & \star & \star \\ \tilde{S}^T & \star & \tilde{R} & \star \\ \star & \star & \star & \star \end{bmatrix} \tag{15b}$$

Then on the basis of these extended variables we can construct the scheduling function, $\Delta_c(v)$ to satisfy (14a), i.e.

$$\begin{bmatrix} \Delta(v) & 0 \\ 0 & \Delta_c(v) \\ I & 0 \\ 0 & I \end{bmatrix}^T \mathcal{P} \begin{bmatrix} \Delta(v) & 0 \\ 0 & \Delta_c(v) \\ I & 0 \\ 0 & I \end{bmatrix} \succ 0$$

where $\Delta_c(v)$ is the scheduling function to be determined. Via a few Schur complements and rearrangements of the matrix inequality we arrive at

$$\begin{bmatrix} M_{11}(\Delta(v)) & M_{12}(\Delta(v))^T + \Delta_c(v)^T \\ M_{12}(\Delta(v)) + \Delta_c(v) & M_{22}(\Delta(v)) \end{bmatrix} \succ 0$$

where $M(\Delta(v))$ is a rational function of $\Delta(v)$ as shown in [8]. Then we can simply choose $\Delta_c(v) = -M_{12}(\Delta(v))$ to zero out the off-diagonal terms.

The final part is the construction of the LTI part of the controller. This is done by solving (14b) with the controller variables inserted. We note that the inner terms of the matrix inequality is known and the outer terms are linear in the controller variables. In fact we can describe this part of the controller construction as determining

$$K = \begin{bmatrix} A_c & B_c \\ C_c & D_c \end{bmatrix}$$

for the quadratic matrix inequality (QMI)

$$\begin{bmatrix} I \\ N + L \cdot K \cdot R \end{bmatrix}^T \Pi \begin{bmatrix} I \\ N + L \cdot K \cdot R \end{bmatrix} \prec 0$$

in which all other variables have already been determined. This problem can be solved algebraically as in [5] or alternatively the QMI can be rewritten as an LMI by a Schur complement and then the LMI is solved using convex optimisation methods.

When applying the procedure for the construction of a controller for the operation of wind turbines in the full load region we observe numerical issues for all three steps in the algorithm. The most critical part is the construction of the extended multipliers in (15). In Table 2 the conditioning of the variables from the synthesis procedure are illustrated and it is clear that the construction of the extended scalings will be ill-conditioned when based on these variables.

It has been investigated if the inclusion of a bound on the norm of each variable will increase the conditioning of the algorithm. This has been done by fixing the performance level to $\gamma_{construct} = 36.9$ which is a tolerated decrease in performance level of

Variable	X	Y	P	$\tilde{P}$
Norm	$5.12 \cdot 10^{6}$	$1.10 \cdot 10^{7}$	$7.38 \cdot 10^{8}$	$9.07 \cdot 10^{5}$
Conditioning	$2.06 \cdot 10^{12}$	$3.30 \cdot 10^{8}$	$2.21 \cdot 10^{11}$	$3.67 \cdot 10^{7}$

Table 2: Conditioning of multipliers from synthesis. Note that all four variables are poorly conditioned leading to ill-conditioning in the construction algorithm.

5% compared with the optimum value. Then the synthesis is performed with an extra constraint on the four variables

$$||X|| < \alpha_1 \cdot \beta \quad , \quad ||Y|| < \alpha_2 \cdot \beta \quad , \quad ||P|| < \alpha_3 \cdot \beta \quad , \quad ||\tilde{P}|| < \alpha_4 \cdot \beta$$

in which various combinations of α_1, α_2, α_3, and α_4 have been investigated and β has been used in the optimisation problem as the variable to minimise, i.e. the largest scaled norm of the four variables has been minimised with the constraint that the performance level is reduced by no more than 5%.

Several different combinations of scalings of the norm bounds have been investigated without observing results that are stable in the numerics. In fact the construction algorithm seems very sensitive to the choice of scalings, α_i, and to give an understanding of the controller construction, the results are presented for the trade-off which seems to give the best numerical performance. In Table 3 the conditioning of the variables are illustrated and it can be observed that the variables are still very poorly conditioned.

Variable	X	Y	P	$\tilde{P}$
Norm	$1.11 \cdot 10^{8}$	$6.36 \cdot 10^{5}$	$6.80 \cdot 10^{8}$	$5.34 \cdot 10^{4}$
Conditioning	$4.64 \cdot 10^{13}$	$1.17 \cdot 10^{6}$	$2.14 \cdot 10^{11}$	$6.72 \cdot 10^{6}$

Table 3: Conditioning of multipliers from bounding variables. The variables are still ill-conditioned, but provides the best numerical performance.

The extended multipliers have then been constructed with the conditioning illustrated in Table 4 from which it can be seen that the conditioning is very poor which means inverting the extended multipliers is not reliable from a numerical point of view which makes it very difficult to verify if the constructed extended multipliers satisfy the conditions in (15).

Variable	$\mathcal{X}$	$\mathcal{Y}$	$\mathcal{P}$	$\tilde{\mathcal{P}}$
Norm	$1.11 \cdot 10^{8}$	$8.10 \cdot 10^{5}$	$8.46 \cdot 10^{8}$	$6.59 \cdot 10^{7}$
Conditioning	$9.02 \cdot 10^{13}$	$9.02 \cdot 10^{13}$	$3.15 \cdot 10^{16}$	$5.58 \cdot 10^{16}$

Table 4: Conditioning of extended multipliers from bounding variables.

The construction steps two and three require the inversion of matrices that depend on $\mathcal{X}$ and $\mathcal{P}$ which are both very ill-conditioned. This means that the construction of the scheduling function and the LTI part of the controller is very sensitive to small errors (rounding and similar) in these variables due to the ill-conditioning. For the LTI part

of the controller, the problem can be resolved by using the optimisation based method, however this cannot be done for the scheduling function because this would require solving an infinite number of LMIs.

The resulting controller has been analysed by LTI methods to understand the local properties along the trajectory of operating conditions. In Figure 8 a magnitude plot of the LPV controller is illustrated for the two extreme points and the nominal value together with LTI controllers designed for each of the three operating conditions. The main observation is that all three samples of the LPV controller has significantly smaller gains when compared with the LTI controllers – especially at the extreme points. The large difference in gains between the nominal operating condition and the extreme values is investigated further in Figure 9 from which it can be seen that the controller gain decreases rapidly as the operating condition deviates from the nominal condition. Such small gains in the controller essentially means that the wind turbine is left almost to operate in open loop when the operating condition approaches the two extreme values. This behaviour is very bad and leads to instability in the lower wind speed range and very poor tracking in the high wind speed range.

Close to the nominal operating condition an LTI analysis shows a performance level of $\gamma_{nom} = 3.38$ which is good when considering the guaranteed LPV performance level of 36.9. For lower wind speeds (and very high wind speeds) the LTI analysis however shows that the closed loop is unstable which indicates that the construction of the scheduling function has failed due to numerical errors.

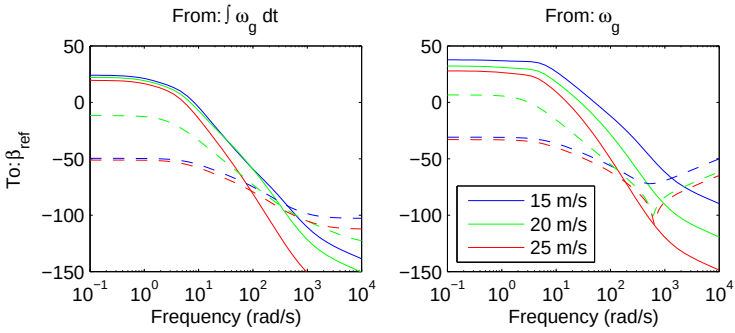


Figure 8: Comparison between LTI controllers (solid) and sample of LPV controller (dashed). Observe the very low gain of the LPV controller.

4. Conclusions

This report has considered the design of a linear parameter varying (LPV) controller for a wind turbine operating in full load. The design is based on a linear fractional transformation (LFT) of the weighted LPV model of the wind turbine developed. The applied design algorithm has been developed a few years ago and constructs an LFT of a LPV controller to satisfy a given performance bound measured by the energy gain.

The algorithms consists of two steps. First an optimal performance level is determined and associated multipliers (used to evaluate the performance) are calculated. For the particular application we observe a significant decrease in performance level (a factor of 35), and it is at the moment not clear if the performance decrease is caused by numerical issues or by an underlying assumption of arbitrary fast parameter variations.

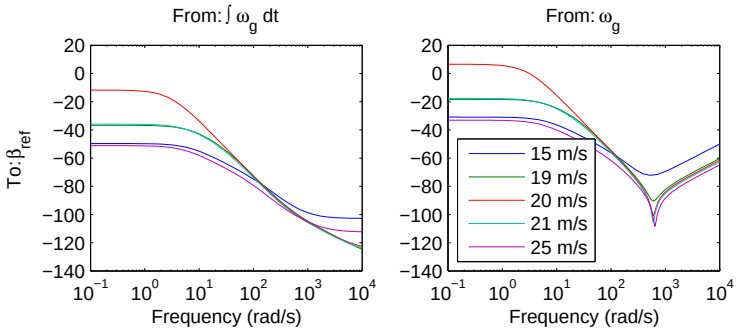


Figure 9: Sample of LPV controller along design trajectory. Controller very low for gains far from the nominal operating condition.

The second step in the algorithm is to calculate the controller variables from the obtained multipliers and performance level. For this step in the algorithm it has been observed that the controller construction is very sensitive to the conditioning of a construction of extended multipliers. From the construction of LTI controllers it is well-known that the poor conditioning of the construction of the extended multiplier (often denoted Lyapunov function) is related to construction of a controller that is not minimal in the number of state variables. It is expected that the issue is related what has been observed for the construction of the extended multipliers for the LPV controller. In this case, however, the minimality refers not only to the number of state vectors but also to the size of the scheduling block. Essentially this means that we expect that the issue can be resolved by adapting the methods from design of low-rank controllers to LPV control with reduced size of the scheduling function – which results in a non-convex optimisation problem.

For the construction of the scheduling function and LTI part of the controller, the construction of the scheduling function has been experienced to very challenging if the extended multipliers are poorly conditioned. This also indicates that a way to go forward is to investigate the minimality of the scheduling function. Such an investigation has not been done and it must be concluded that no satisfactory LPV controller has been obtained from the applied method.

To conclude it has been determined that neither step in the available design procedure is appropriate for the design of gain-scheduled controllers for the operation of wind turbines in full load operation. In order to get satisfactory performance a method must first and foremost be determined with which we can obtain a performance level similar to local LTI controllers. To do this we recommend the gridding method for the following reasons: First of all it is a method with which the number of variables (under the same assumptions) are significantly reduced which should improve the conditioning of the optimisation algorithm. Furthermore the assumption of arbitrary fast parameter variations can be relaxed in order to take into account the rate of variation. Hereby we expect that both issues related to the optimisation problem can be addressed. The drawback of the method is that it is only approximative, i.e. based on sampling the parameter region. This means that the obtainable performance level might be too optimistic when comparing to what is really possible, but by varying the grid size it is expected that we

can find a fixed point which is very close to the true value.

Regarding the construction of the controller it is concluded that the use of extended scalings is not appropriate for a numerically sound algorithm. For the LFT approach this means that it should be investigated how to construct the LTI part and scheduling function directly from the multipliers obtained from the optimisation problem. For the gridding method such a method has been proposed in [4, 1] for which we expect the ideas can be extended to the LFT method.

REFERENCES

[1] Pierre Apkarian and Richard J. Adams. Advanced gain-scheduling techniques for uncertain systems. *Trans. on Control Systems Technology*, 6(1):21–32, 1998. 10.1109/87.654874.

[2] Greg Becker and Andy Packard. Robust performance of linear parametrically varying systems using parametrically-dependent linear feedback. *Systems and Control Letters*, 23:205–215, 1994. http://dx.doi.org/10.1016/0167-6911(94)90006-X.

[3] F.D. Bianchi, R.J. Mantz, and C.F. Christiansen. Control of variable-speed wind turbines by LPV gains scheduling. *Wind Energy*, 7:1–8, 2004.

[4] Pascal Gahinet. Explicit controller formulas for lmi-based $\mathcal{H}_\infty$ synthesis. *Automatica*, 32(7):1007–1014, 1996. 10.1016/0005-1098(96)00033-7.

[5] Anders Helmerssonn. IQC synthesis based on inertia constraints. In *IFAC World Congress*, 1999.

[6] Fabien Lescher, H. Camblong, O. Curea, and R. Briand. LPV control of wind turbines for fatigue loads reduction using intelligent micro sensors. In *American Control Conference*, pages 6061–6066, 2007.

[7] Fabien Lescher, Jing Yun Zhao, and Pierre Borne. Robust gain scheduling controller for pitch regulated variable speed wind turbine. *Studies in Informatics and Control*, 14(4):299–315, 2005.

[8] Carsten W. Scherer. *Robust Mixed Control and LPV Control with Full Block Scalings*. SIAM, 2000.

[9] Carsten W. Scherer. LPV control and full block multipliers. *Automatica*, 37:361–375, 2001. http://dx.doi.org/10.1016/S0005-1098(00)00176-X.

[10] Chen Wang and George Weiss. Self-scheduled LPV control of a wind driven doubly-fed induction generator. In *Conf. on Decision and Control*, pages 1246–1251, 2006. 10.1109/CDC.2006.377474.

Report G

Manual for LPV Design Tool

Contents

Technical report.

Manual for LPV design tool

Version 1.0

Kasper Zinck Østergaard
September 13, 2007

1. Introduction

This manual describes how to use a set of tools for designing linear parameter varying controllers using the grid based method. The tool is proprietary (not public domain) and is implemented for controller design with single, scalar parameter dependency. The implemented algorithms are based on the LPV method for general parameter dependency developed in [3, 2, 1]. To make it applicable to medium-large scale systems, the numerical modifications described in Paper C and Paper D been applied.

To install the tool, unzip the files into a directory on the Matlab path. It should be noted that the tool requires the Control Systems Toolbox and the Robust Control Toolbox. Also Simulink is required for using the full potential of the graphical part of the tool.

In this manual, first the graphical user-interface is presented to give an overview of the possibilities in the tool. Then an introduction to the tool is given by an example going though the basic steps of designing a linear parameter varying controller.

2. Preparation for LPV design

Before the design tool can be used for controller design, the user must prepare a file for creation of LTI models at selected operating conditions. Further to get the full potential of the parts related to Simulink simulations, a few components should be included in the Simulink model.

2.1. Model file

A Matlab function is necessary for the generation of the generalised plant. The function must take only one argument which is an array of parameter values. Then it returns a structure containing the parameter values and an LTI open loop model for each parameter value. Further the structure can contain other variables used for simulation purposes.

For the open loop model it is important that the control/performance channels are selected, also the inputs and outputs should be named. A structure for the model file is proposed below. Note that if no parameter is entered, a dummy state space model is returned with appropriate dimension and all outputs

2.2. Simulink model

For the Simulink simulation, two variables will be available. First of all the controller is available as the variable, `ctrl`. For the simulations with frozen parameters, `ctrl`, will be an `ss` object and for the interpolated controller, `ctrl` will be a cell array of `ss` objects. To implement the interpolated controller, the block “interpolated ss model” from

```
function LTImod = model_fcn(parm)
 if nargin == 0
  % If no argument: Return empty state space model with
  % correct size input names and performance/control
  % channels
  n = ...; m = ...;
  LTImod=ss([],zeros(0,nw+nu),zeros(nz+ny,0),
                                  zeros(nz+ny,nw+nu));
  LTImod.InputNames  = {'w_1',..,'w_nw',u_1',..,'u_nu'};
  LTImod.OutputNames = {'z_1',..,'z_nz',y_1',..,'y_ny'};
  LTImod.InputGroups.Performance  = 1:nw;
  LTImod.InputGroups.Control      = nw + (1:nu);
  LTImod.OutputGroups.Performance = 1:nz;
  LTImod.OutputGroups.Control     = nz + (1:ny);
 else
  % Normal case: return structure containing parameter
  % values and cell array of open loop models in state
  % space form
  LTImod.parm = parm;
  for I=1:length(parm)
   A = ...; B = ...; C = ...; D = ...;
   LTImod.G{I} = ss(A,B,C,D);
   LTImod.G{I}.InputNames  =
                       {'w_1',..,'w_nw',u_1',..,'u_nu'};
   LTImod.G{I}.OutputNames =
                       {'z_1',..,'z_nz',y_1',..,'y_ny'};
   LTImod.G{I}.InputGroups.Performance  = 1:nw;
   LTImod.G{I}.InputGroups.Control      = nw + (1:nu);
   LTImod.G{I}.OutputGroups.Performance = 1:nz;
   LTImod.G{I}.OutputGroups.Control     = nz + (1:ny);
  end
 end
end
```

Figure 1: Matlab code dump to illustrate the structure of the model file: `model.m`.

the “LPV Design Blockset” can be applied – the blockset is included in the tool. This block uses linear interpolation of the system matrices to implement a LPV controller from the set of controllers.

If the simulation time should be updated in the GUI, the block “update gui” should be included in the Simulink model as in Figure 2. The zero-order-hold is implemented to control the rate with which the GUI is updated.

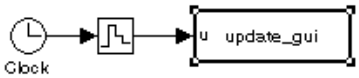


Figure 2: A way to update the simulation time in the GUI.

3. Presentation of graphical user interface

In Figure 3 a snapshot of the graphical user interface (GUI) is given. The red boxes mark different components of the GUI numbered from one to six. In the following sections, these six components will be described one by one.

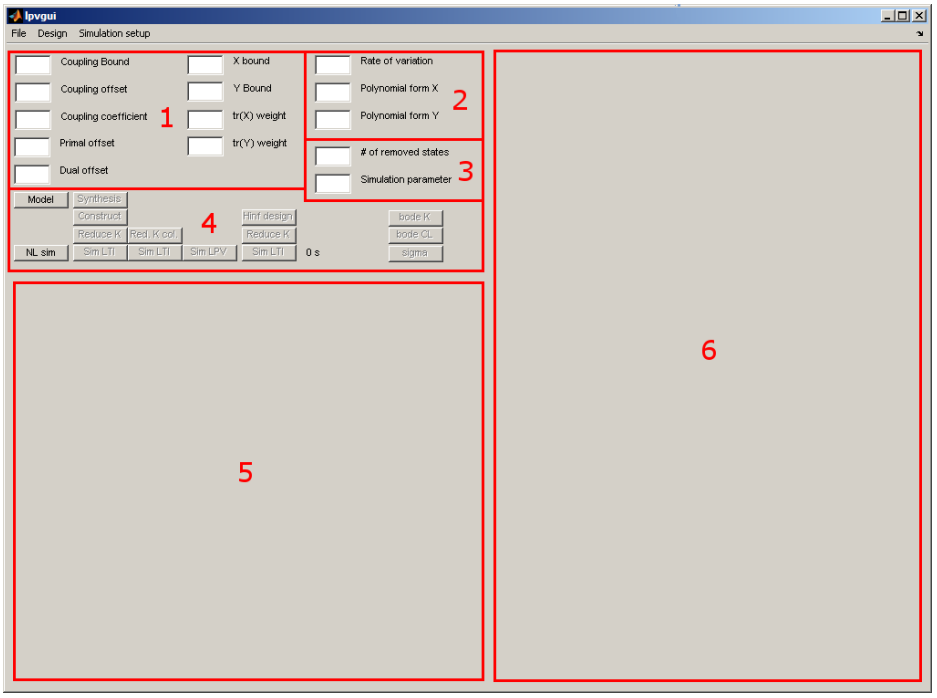


Figure 3: Illustration of graphical user interface.

3.1. LMI Bounds, offset and coefficients

The first component marked in Figure 3 is for constraining the synthesis linear matrix inequalities (LMIs) and variables in order to make the controller construction possible in practice.

Coupling bound (β_c) is the upper bound on the coupling condition

Coupling offset (ϵ_c) is the lower bound on the coupling condition

Coupling coefficient (c_c) is the coefficient for separating X from Y^{-1}

X bound (β_X) is the upper bound on X

Y bound (β_Y) is the upper bound on Y

Primal offset (ϵ_p) is the upper bound on the primal LMI

Dual offset (ϵ_p) is the upper bound on the dual LMI

The control problem is formulated as minimising γ in the set of LMIs given by (1) with a detailed interpretation described in Paper C based on the results in [3, 2, 1]

$$X \succ \beta_X I \quad , \quad Y \succ \beta_Y I \quad , \quad \beta_c I \succ \begin{bmatrix} Y & c_c I \\ c_c I & X \end{bmatrix} \succ \epsilon_c \tag{1a}$$

$$\begin{bmatrix} C(\delta_i)_\perp & 0 \\ F(\delta_i)_\perp & 0 \\ 0 & I \end{bmatrix}^T \begin{bmatrix} XA + A^T X & XB_p & C_p^T \\ B_p^T X & -\gamma I & D^T \\ C_p & D & -\gamma I \end{bmatrix} \begin{bmatrix} C(\delta_i)_\perp & 0 \\ F(\delta_i)_\perp & 0 \\ 0 & I \end{bmatrix} \prec -\epsilon_p \tag{1b}$$

$$\begin{bmatrix} B_p(\delta_i)_\perp^T & 0 \\ E(\delta_i)_\perp^T & 0 \\ 0 & I \end{bmatrix}^T \begin{bmatrix} AY + YA^T & YC_p^t & B_p \\ C_p Y & -\gamma I & D \\ B_p^T & D^T & -\gamma I \end{bmatrix} \begin{bmatrix} B_p(\delta_i)_\perp^T & 0 \\ E(\delta_i)_\perp^T & 0 \\ 0 & I \end{bmatrix} \prec -\epsilon_d \tag{1c}$$

If no value is entered in the respective text fields it means that the bound, offset or coefficient is not in use.

Remark 19. *The bounds and offset should be non-negative and the coefficient, c_c, should be larger and equal to one for the LMIs to guarantee stability and performance.*

For the optimisation problem, the trace of X and Y can be included in the optimisation problem to reduce the size of the two variables. This is done in the two fields

tr(X) weight A weight on the trace of X between 0 and 1.

tr(Y) weight A weight on the trace of Y between 0 and 1.

This means that a large weight (close to one) will result in an X or Y with small diagonal components whereas a small weight (close to zero) will result in a large X or Y.

3.2. Rate of variation

The second marked box in Figure 3 is used to specify the tolerated rate of variation of the scheduling parameter – in the field "Rate of variation". If left blank or "inf" is entered it means arbitrary fast parameter variations – otherwise the rate of variation is specified by a positive number.

When designing with a limit on the rate of variation the matrices X and Y must be parameter dependent. This is implemented by polynomial parameter dependency by specifying a vector containing the exponents in the fields: "Polynomial form X" and "Polynomial form Y".

If we for instance want to design for

$$X(p) = X_0 + p\,X_1 + p^2\,X_2 \quad , \quad Y(p) = Y_0 + p^3\,Y_1$$

we can do this as indicated in Figure 4.

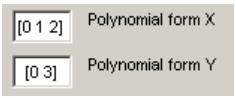


Figure 4: Example of choice of basis function for X and Y.

Remark 20. *Note that for implementation purposes only one of the variables can be chosen as non-constant if the rate limit is non-zero. Otherwise the controller will be dependent on the time derivative of the scheduling parameter. This dependence on the time derivative has not been included in the tool.*

3.3. Number of removed states and simulation parameter

The third marked box in Figure 3 contains two text fields. One for the number of fast states to remove from the constructed controller. If the text field is empty no states will be removed.

The other text field is for the choice of simulation parameter. If the designer wishes to run simulations directly from the tool, this field can be used to setup the simulation for the chosen parameter value.

3.4. Controller design buttons

The fourth box in Figure 3 contains various buttons for designing and simulating LPV and LTI $\mathcal{H}_\infty$ controllers. These buttons will be presented one by one according to Figure 5. Each button will be referred to by its column and row, e.g. the button labelled "Hinf design" will be referred to as button E2.

Button A1: Model creation

This button labelled "Model" will be used to create the generalised plant from which the controller design is performed. The model is created by a function of the form

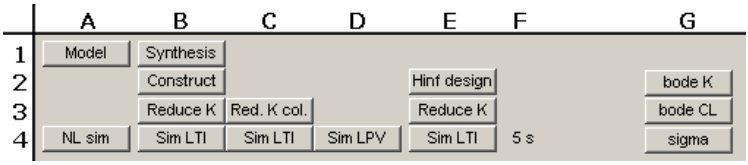


Figure 5: Buttons for controller design and simulation.

```
function LTImod = model_fcn(parm)
  ...
end
```

The function should be designed to return two different objects `LTImod` depending on whether `parm` is passed as a parameter or not:

No parameter `parm` passed:

`LTImod` is an `ss` object with the input/output dimensions of the generalised plant. Inputs and outputs should be labelled, e.g.

```
LTImod.InputNames = {'w1','w2','u1'}
```

Performance channel must be chosen, e.g.

```
LTImod.InputGroups.Performance = 1:2;
```

Control channel must be chosen, e.g.

```
LTImod.InputGroups.Control = 3;
```

Parameter `parm` passed as a vector of real valued numbers:

`LTImod` is a structure with at least two components: `LTImod.G` and `parm`. `LTImod.G` is a cell-array of `ss` objects with the input/output dimensions of the generalised plant. `LTImod.parm` is the parameter vector
Inputs and outputs should be labelled, e.g.

```
LTImod.G{I}.InputNames = {'w1','w2','u1'}
```

Performance channel must be chosen, e.g.

```
LTImod.G{I}.InputGroups.Performance = 1:2;
```

Control channel must be chosen, e.g.

```
LTImod.G{I}.InputGroups.Control = 3;
```

Button B1: LPV synthesis

By pressing this button, the synthesis problem of minimising γ while satisfying the LMIs in (1). The optimisation problem is solved by using `mincx` from the Robust Control Toolbox.

Button B2: Controller construction

Based on the variables, X and Y, obtained from the synthesis, a set of LTI controllers are designed. One for each operating condition.

The different representations of $X_{cl} = X_1^{-1}X_2$ and choice of construction LMI are selected from the menu bar.

Button E2: $\mathcal{H}_\infty$ controller design

This button is used for designing LTI controllers at each operating point. The controllers are designed by using `hinfsyn` from the Robust Control Toolbox. The $\mathcal{H}_\infty$ controller design is mainly used for investigating the difference between LTI and LPV design, i.e. how restrictive the assumption of arbitrary rate of variation is.

Button B3, C3, E3: Controller order reduction

The design methods are based on full order controllers which will typically have some fast modes that are not implementable. These buttons implement a method for removing the states related to these fast modes by diagonalising the system matrix for the controller.

The buttons B3 and C3 are for model reduction of the controllers from the LPV method in respectively a local and global coordinate system. Button E3 is for controller order reduction in local coordinates of the controller from the $\mathcal{H}_\infty$ method.

It should be noted that by simply diagonalising the controllers system matrix and removing the fast modes this way, there is a risk that the frequency response might be altered significantly. In this case it is suggested to use alternative methods for controller order reduction.

Button A4, B4, C4, D4, E4: Closed loop simulation

These buttons start a Simulink simulation of the different controllers according to:

A4 Simulation of controller designed using alternative methods, used for comparison.

B4 Simulation of LPV controller reduced in local coordinates with frozen parameters

C4 Simulation of LPV controller reduced in global coordinates with frozen parameters

D4 Simulation of LPV controller reduced in global coordinates.

E4 Simulation of LTI $\mathcal{H}_\infty$ controller reduced in local coordinates

The progress of the simulation is shown in the text field: F4

The Simulink models used for the simulation model are selected from the menu bar

Button G1, G2, G3

These buttons are used to plot graphs of the controller and closed loop for a selected operating condition. G1 is for open loop bode plot of the controller, G2 shows a magnitude plot of the closed loop and inverse weights, and G3 shows the singular values of the weighted closed loop.

3.5. Panel for entering parameters, weights, and selecting operating conditions

The red box in Figure 3 marked five is for entering information about parameter values and output weights for each operating condition selected for the design. It is assumed that both input and outputs weights are diagonal.

When opening the GUI for the first time nothing will be displayed in this area until a model file has been loaded (See Section 3.7). In Figure 6 an example of the view is displayed for a model with two performance outputs: “int We” and “B r” and one performance input “V”. These names are determined automatically from the performance channel of the open loop system generated by `model.m`.

Figure 6: Panel for entering parameters, associated weighs, and for selecting operating conditions for synthesis, construction, Gramian-based balancing, order reduction, simulation.

The left most text field is for entering the parameter for the considered operating condition. Then in the other fields the weights can be specified by entering the transfer function used for the scalar component in the weight. If we for example want to weight “int We” by a constant term: 31 and “B r” by a first order filter: $5\frac{s+1}{s+5}$ and “V” by $\left(\frac{s+5}{s+0.1}\right)^2$ this can be done as shown in Figure 7.

Figure 7: Example of panel with entered values.

In the right part of Figure 6 there are two check boxes and three radio buttons. The check boxes are used to choose if the particular operating condition should be used for respectively synthesis and controller construction. This way the computationally demanding optimisation problem can be run for a more coarse grid than the controller construction.

The two first radio buttons are used for selecting which operating condition is used as basis for respectively Gramian based balancing of the open loop system, and for controller order reduction. If no operating condition is selected for the balancing it means that no a-priory balancing will be performed.

The rightmost radio button is used to select which controller will be used for simulations when using the LTI controller.

In Figure 6 only one row is displayed for entering information about the operating conditions, however when entering data into a row, another row will be displayed below. This way it is possible to enter as many operating conditions as desired.

3.6. Illustration of design results

The sixth red box in Figure 3 is used for illustration of results from the synthesis. It will be described in the example (Section 5) how to interpret the tables displayed in this area.

3.7. Menu bar

The menu bar has three menus: File, Design, and Simulation setup. In the following these menus will be described.

File menu

The file menu has four menu items

Load setup reads data from a selected `.mat` file and updates the weights, parameters, choice of bounds, etc. according to the data

Save setup Save data regarding weights, parameters, choice of bounds, etc. to a selected `.mat` file.

Save controller data stores controller variables, synthesis result, etc. as a structure `ctrl_dat` in a selected file.

Design

The design menu has three menu items

Model file selects the file used to generate the generalised plant.

Hoo Design selects if "hinflmi" from the Mathworks Robust Control Toolbox should be used for $\mathcal{H}_\infty$ controller design or if the method in this toolbox is used.

Simulation setup

This menu is for setting up the Simulink model used for closed loop simulation. The menu has three menu items:

LPV controller is for the interpolated controller.

LTI controller is for the LPV controller with frozen parameters and for the $\mathcal{H}_\infty$ controller.

Other controller is for a controller designed using alternative methods.

Each menu item opens a window in which the Simulink model can be selected and in which the init and stop functions can be entered. In these functions, the term PARAM can be used to represent the initial parameter value so that the two functions can be made dependent on operating condition.

4. LPV design from command prompt

It is also possible to use the tool for designing controllers from the command prompt. The commands to use are listed in the following and for a description of the command options and outputs can be seen using the Matlab `help` function.

lpvsynthesis.m Runs the optimisation problem and returns X, Y, and γ.

constructK.m Constructs the set of controllers for the chosen operating points.

rmfast.m Removes the selected number of fast modes from the designed controller(s).

5. Example

This section presents the tool by giving a simple example of designing a linear parameter varying controller for an LTI plant. The purpose is to make the controller aggressive for large tracking errors and slow for small tracking errors.

5.1. Preparation steps

The example will consider the control of a double integrator with the control formulation as in Figure 8. The objective will be to track the reference, r, and integral action has been included to obtain asymptotic tracking. The setup in Figure 8 leads to the following

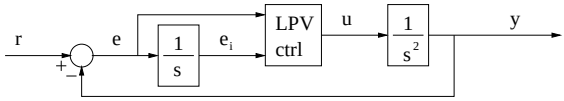


Figure 8: Block diagram of model and controller structure.

model of the interconnection used for controller design

$$\dot{x}_1 = u \quad , \quad \dot{x}_2 = x_1 \quad , \quad \dot{x}_3 = r - x_2$$
$$e = r - x_2 \quad , \quad e_i = x_3 \quad , \quad y = x_2$$

The input for the performance channels is chosen as r and the performance outputs have been chosen to be e and y to trade off sensitivity with complementary sensitivity. The generalised plant is then as in (2).

$$\left[\begin{array}{c} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \hline e \\ y \\ \hline e \\ e_i \end{array}\right] = \left[\begin{array}{ccc|c|c} 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 \\ \hline 0 & -1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ \hline 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{array}\right] \left[\begin{array}{c} x_1 \\ x_2 \\ x_3 \\ \hline r \\ \hline u \end{array}\right] \tag{2}$$

This open loop model is entered into the Matlab file “model.m” according to the specifications of Figure 1. Also Simulink models “ltimod.mdl” and “lpvmod.mdl” are created with the extra blocks according to Section 2.2. Output variables are also stored in the workspace to allow for plotting later on.

5.2. GUI interface

The GUI is now opened by issuing the command `lpvgui`. Now all input fields are empty and no fields are available for entering performance weights. The design is performed by the following steps:

Step 1: The model file `model.m` is selected from the menu item “model file” in the design menu.

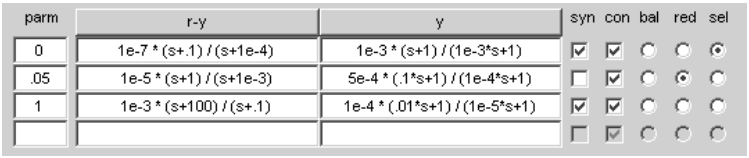


Figure 9: Entering of weights.

Step 2: Parameter and weights are entered. For the specific application the weights have been selected from an iterative procedure to get what is shown in Figure 9.

Step 3: Select operating points used for synthesis and mark if Gramian-based balancing should be performed for the open loop model. In this example the two operation conditions for the parameter equal to zero and one, respectively, have been selected for synthesis, and no balancing is to be performed. This can also be seen from Figure 9.

Step 4: Create model and perform synthesis without bounds. Done by pressing first the "Model" button and subsequently the "Synthesis" button. Results from the synthesis are displayed in the right hand side of the GUI and in Figure 10 it is shown for the particular example. The three topmost lines show respectively the polynomial expansion of $X(p)$ and $Y(p)$, the size of optimisation problem, and computation time and obtained performance level.

The two topmost tables show information about the primal dual LMIs. The first two columns show the parameter values with the first one in a scaled version. The third and fourth column show the minimum and maximum eigenvalue of the LMI and the fifth column display the conditioning of the LMI. Finally the last column shows a green figure which illustrates that the condition is satisfied – a red figure is displayed if the LMI is not satisfied.

The bottom table display information about variables relevant for the controller construction. The columns are interpreted similar to what is displayed above.

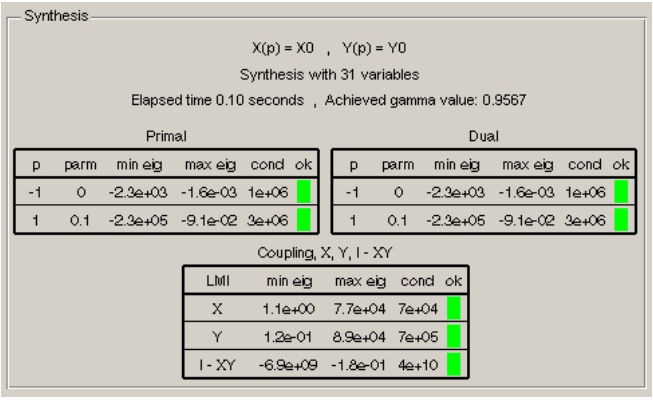


Figure 10: Presentation of synthesis results.

Step 5: Construct controller by pressing the "Construct" button. If controller construction fails return to step 4 but with introduced bounds on relevant variables. For this simple application it has not been necessary to improve numerical performance by bounding variables.

When performing the controller construction, four different pages are illustrated in the right side of the GUI – selected by the buttons panel above the information. The first page shows information about some of the critical variables used in controller construction. If the controller construction fails, these variables can be used to backtrack what went wrong and for example what variable to bound in the synthesis in order to make the controller construction possible.

The two topmost tables display variables X and Y and variables deducted from these two variables. The top two rows show respectively the smallest and largest singular value of each variable. The bottom table display similar information together with the condition number for variables deducted from the coupling condition.

The construction of the input and output matrices for the controller is done by solving two different LMIs on the basis of the constructed variables in Figure 11. Properties of these two LMIs are displayed in Figure 12 which can be interpreted similar to the tables in Figure 10.

An LTI closed loop analysis is performed by sampling the LPV controller at the grid points. The result of this analysis is presented in the third page which is reproduced in Figure 13 for the particular example. The topmost table illustrate the largest real value for the closed loop poles to indicate stability properties. Further the local $\mathcal{H}_\infty$ norm is shown at each grid point and a green/red marker is illustrated if the closed loop is stable/unstable. The bottom table display the norm of the controller variables which give an indication if the controller has high gains. Such large gains can typically be removed either by bounding X and/or Y or by controller order reduction.

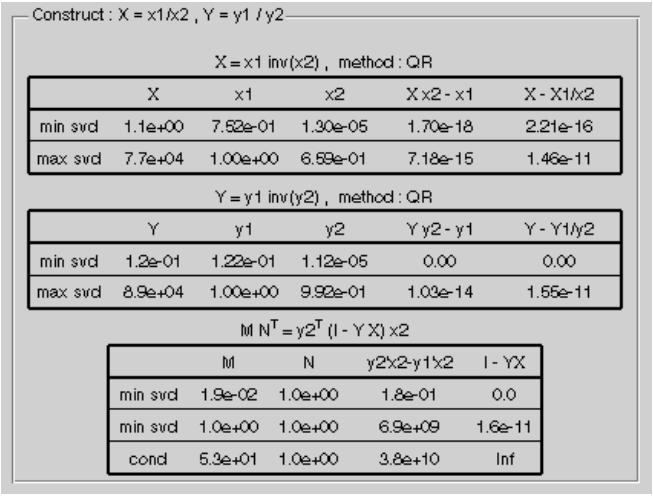


Figure 11: Information about variables used in controller construction.

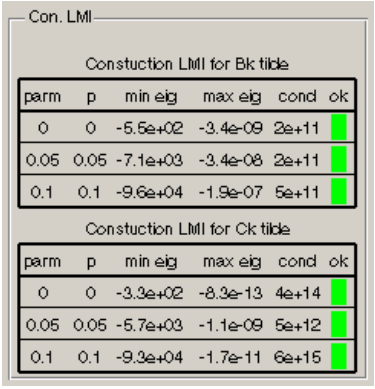


Figure 12: Illustration of construction LMIs.

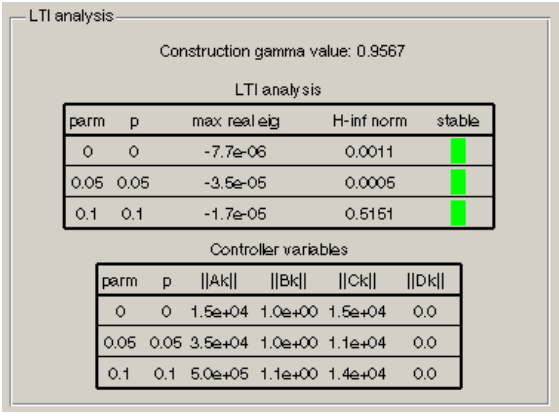


Figure 13: LTI analysis of closed loop.

Step 7: Controller order reduction. From the selection of an operating condition (done by clicking on an appropriate radio button in the "sel" column) it can be seen that the constructed controller has two modes that are significantly faster than the other modes. This can be seen from the left column of Figure 14.

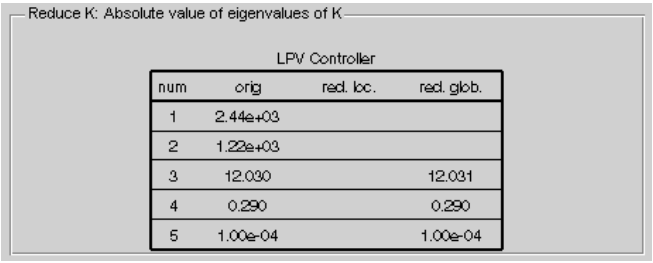


Figure 14: Controller order reduction.

These modes are removed by entering "2" in the text field "# of removed states". The order reduction can now be done in local coordinates by pressing "Reduce K" or in global coordinates by pressing "Red. K. col.". We choose the global coordinates because this is the only option that will enable simulation of the LPV controller. The result can be seen in Figure 14 in which it can be seen that the corner frequency of the other poles are not affected significantly by the reduction in global coordinates. The effect can also be seen from bode plots by pressing either the "bode K" or "bode CL" buttons and selecting an operating point to view (by the radio button "sel").

Step 8: Simulate closed loop response. First a simulation model is chosen for both LTI and LPV simulations from the appropriate menu item in the Simulation setup menu. The choices for the LTI and LPV setup are shown in Figure 15 from which it can be seen that the `stopfcn` is used for plotting simulation results in a comparison.

The result of the controller design can be seen in simulations as illustrated in Figure 16 which shows a simulation of the designed controller with the dotted lines representing the three constructed controllers with frozen parameters and the solid line is LPV controller. The horizontal black, dashed lines illustrate the design points (param = $|e|$) and it can be observed how the aggressiveness of the controller changes as a function of tracking error.

From Figure 16 it can be seen that when the tracking error is large (above 0.1), the controller is aggressive and follows the behaviour of the fast controller with frozen parameters. When the tracking error decreases below 0.1 the controller changes behaviour and has a much slower behaviour. Such a behaviour can be advantageous if we want a fast step response without having overshoot or simply if we want aggressive behaviour when the error is large and small control effort when the error is smaller.

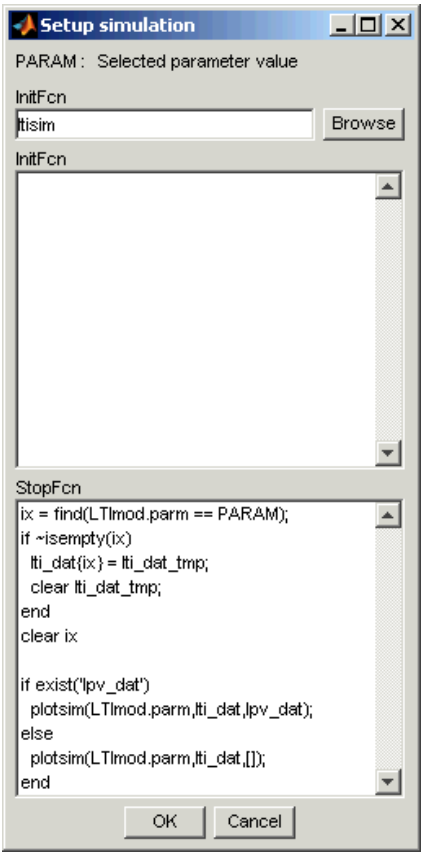


Figure 15-a: Simulation setup for LTI simulations.

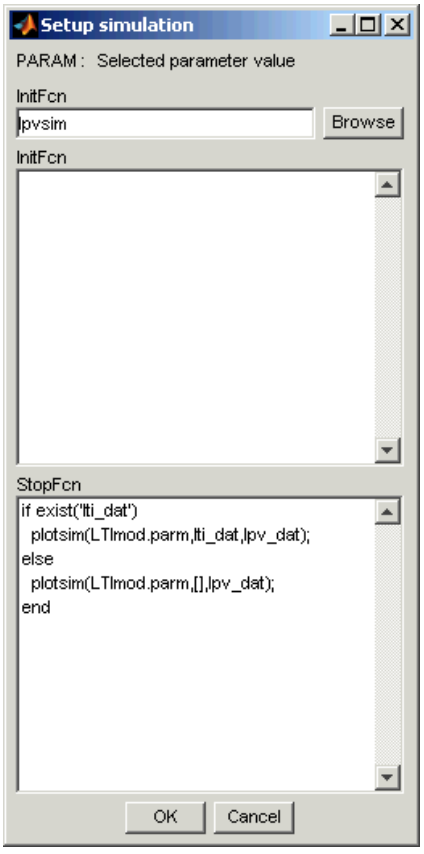


Figure 15-a: Simulation setup for LPV simulations.

Figure 15: Setup for LTI and LPV simulations. InitFcn is used to initialise parameters necessary for the simulation – controller parameters are automatically loaded. StopFcn is used for automatic post-processing of simulation results – In the example it is used to plot simulation data (by plotsim)

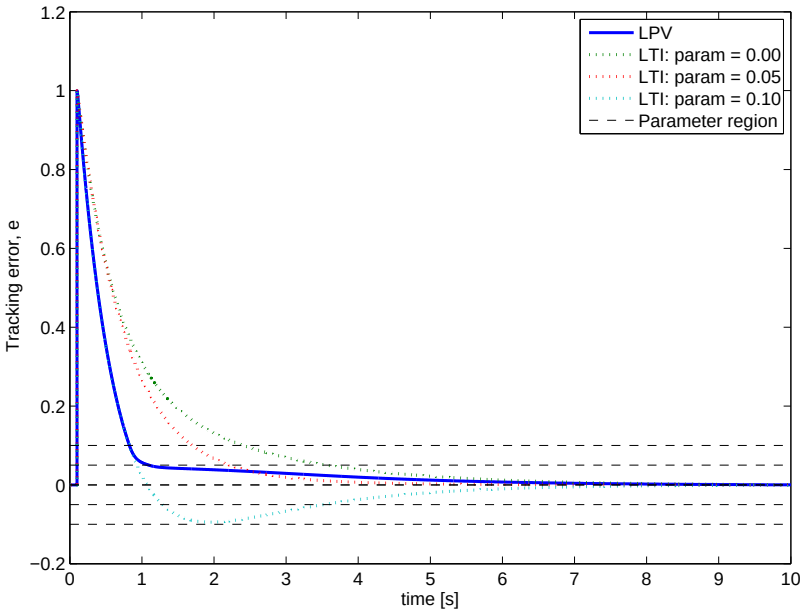


Figure 16: Simulation results. Comparison between frozen parameter controllers and LPV controller.

The model file (`model.m`), Simulink files (`ltisim.mdl` and `lpvsim.mdl`) and a setup (setup.mat file) for the GUI can be found together with this manual. Note that the designer is free to choose alternative filenames which will enable the possibility of having different versions in the same directory.

Alternatively the design can be done from the command prompt by the following commands. In this command line example, the model is initialised directly in the lines of code. This could also be done calling `model.m` if this file has been created. No simulation is done within this sample code, and this could be done by standard means.

```
parm = [0 0.05 0.1];
for I=1:length(parm)
  plantW{I} = ...;
end
cmd.offset = [0 0 0 1e4 1e4 0];
cmd.coeff = 1;
[gam,X,Y] = lpvsynthesis_const(plantW,parm,cmd);
cmd.cX = 3;
cmd.method = 3;
ctrl = lpvconstructXY_const(plantW,parm,X,Y,gam,cmd);
for I=1:length(parm)
  ctrl{I} = rmfast(ctrl{I},1);
end
```

REFERENCES

[1] Pierre Apkarian and Richard J. Adams. Advanced gain-scheduling techniques for uncertain systems. *Trans. on Control Systems Technology*, 6(1):21–32, 1998. http://dx.doi.org/10.1109/87.654874.

[2] Pascal Gahinet. Explicit controller formulas for lmi-based $\mathcal{H}_\infty$ synthesis. *Automatica*, 32(7):1007–1014, 1996. http://dx.doi.org/10.1016/0005-1098(96)00033-7.

[3] Pascal Gahinet and Pierre Apkarian. A linear matrix inequality approach to $\mathcal{H}_\infty$ control. *Int. J. of Robust and Nonlinear Control*, 4:421–448, 1994. http://dx.doi.org/10.1002/rnc.4590040403.