DECEMBER 2018

M.Sc. in Electrical and Electronics Engineering

ARİF YILDIZ

REPUBLIC OF TURKEY

UNIVERSITY OF GAZİANTEP

GRADUATE SCHOOL OF NATURAL & APPLIED SCIENCES

OPTIMAL CONTROL OF A WIND TURBINE BY USING LINEAR MATRIX INEQUALITY METHOD

M. Sc. THESIS

IN

ELECTRICAL AND ELECTRONICS ENGINEERING

By

ARİF YILDIZ

DECEMBER 2018

Optimal Control of a Wind Turbine by Using Linear Matrix Inequality Method

M.Sc. Thesis

in

Electrical and Electronics Engineering

University of Gaziantep

Supervisor

Assist. Prof. Dr. Tolgay KARA

By

Arif YILDIZ

December 2018

REPUBLIC OF TURKEY
UNIVERSITY OF GAZİANTEP
GRADUATE SCHOOL OF NATURAL & APPLIED SCIENCES
ELECTRICAL AND ELECTRONICS ENGINEERING DEPARTMENT

Name of the thesis: Optimal Control of a Wind Turbine by Using Linear Matrix Inequality Method

Name of the student: Arif YILDIZ

Exam date: 27/12/2018

Approval of the Graduate School of Natural and Applied Sciences

Prof. Dr. Ahmet Necmeddin YAZICI
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Prof. Dr. Ergun ERÇELEBİ
Head of Department

This is to certify that we have read this thesis and that in our consensus opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Assist. Prof. Dr. Tolgay KARA
Supervisor

Examining Committee Members:	Signature
Assoc. Prof. Dr. Ahmet Mete VURAL	…………………..
Assist. Prof. Dr. Tolgay KARA	…………………..
Assist. Prof. Dr. Murat FURAT	…………………..

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Arif YILDIZ

ABSTRACT

OPTIMAL CONTROL OF A WIND TURBINE BY USING LINEAR MATRIX INEQUALITY METHOD

YILDIZ, Arif

M.Sc. in Electrical and Electronics Engineering

Supervisor: Asst. Prof. Dr. Tolgay KARA

December 2018

57 pages

Linear matrix inequalities (LMIs) are very important tools that allow us to numerically solve control engineering problems by means of computer tools and optimization techniques. LMIs are convex constraints for a semidefinite programming (SDP) which is subfield of convex optimization. In this thesis, LMI structure and problems related with LMIs were studied. Optimal control problem was written in the form of LMI eigenvalue problem. With the help of the relevant computer tools, controller design was done for linear systems by solving LMI problem. A wind turbine control problem was selected as a case study in order to reveal effectiveness of proposed optimal control solution. A nonlinear turbine model was linearized at an operating point. For linear wind turbine model, a controller design was performed by using linear quadratic regulator (LQR) method. In this design, the LQR problem was written as the LMI constraint of the SDP. By numerically solving the optimization problem, a state feedback gain matrix which minimizes the performance index was calculated. The designed controller was compared with the baseline gain scheduled PI controller on the nonlinear wind turbine model in the computer simulation. Results reveal that proposed optimal control solution performs satisfactorily well and optimal LMI based LQR control proves to be a suitable alternative in controlling nonlinear wind turbine systems.

Key Words: Linear matrix inequality, linear quadratic regulator, wind turbine.

ÖZET

BİR RÜZGAR TÜRBİNİNİN LİNEER MATRİS EŞİTSİZLİK YÖNTEMİ KULLANILARAK OPTİMAL KONTROLÜ

YILDIZ, Arif

Yüksek Lisans Tezi, Elektrik-Elektronik Müh. Bölümü

Tez Yöneticisi: Dr. Öğr. Üyesi Tolgay KARA

Aralık 2018

57 sayfa

Doğrusal dizey eşitsizlikleri (DDE'ler) kontrol mühendisliği problemlerini bilgisayar araçları ve optimizasyon tekniği yardımıyla nümerik olarak çözmemizi sağlayan çok önemli araçlardır. DDE'ler dış bükey optimizasyonun alt alanı olan yarı tanımlı programlama için dış bükey kısıtlardır. Bu tezde DDE'nin yapısına ve DDE ile ilgili problemlere çalışıldı. Optimal kontrol problemi LMI özdeğer problemi biçiminde yazıldı. İlgili bilgisayar araçlarının yardımıyla DDE problemi çözdürülerek lineer sistemler için kontrolör tasarımı yapıldı. Önerilen optimal kontrol çözümün etkililiğini göstermek için bir örnek çalışma olarak bir rüzgar türbin kontrol problemi seçildi. Bir doğrusal olmayan rüzgâr türbin modeli bir operasyon noktasında doğrusallaştırıldı. Doğrusal model için doğrusal kareli düzenleyici (DKD) yöntemi kullanılarak bir kontrolör tasarımı yapıldı. Bu tasarımda DKD problemi yarı tanımlı programlamanın DDE kısıtı şeklinde yazıldı. Optimizasyon probleminin nümerik olarak çözdürülmesiyle performans endeksini minimize eden bir durum geri besleme kazanç matrisi hesaplatıldı. Tasarlanan kontrolör, kazanç planlama PI kontrolör ile doğrusal olmayan rüzgâr türbin modelinin bilgisayar benzetimin ortamında kıyaslandı. Sonuçlar şunu gösterdi: önerilen optimal kontrol çözümü tatmin edici bir şekilde iyi performans sergiler ve optimal DDE temelli DKD doğrusal olmayan rüzgar türbin sisteminin kontrolünde uygun bir alternatif olduğunu kanıtlar.

Anahtar Kelimeler: Doğrusal dizey eşitsizlikleri, optimal kontrol, rüzgâr türbini.

ACKNOWLEDGEMENTS

I express my deep gratitude and appreciation to my supervisor Assist. Prof. Dr. Tolgay KARA for his guidance, excellent advice and cooperation during the course of this work. He has followed my work step by step so that I am extremely grateful and indebted to him for his expert extended to me. His timely and efficient contribution helped me shape this thesis into its final form.

I would like to express my deep appreciation to my family and my best friend Sedef USLU for their support to me during my whole time study.

TABLE OF CONTENTS

Page

LIST OF TABLES

Page

LIST OF FIGURES

Page

LIST OF SYMBOLS

LMI	Linear Matrix Inequality
SDP	Semidefinite Programming
LQR	Linear Quadratic Regulator
ARE	Algebraic Riccati Equation
EVP	Eigenvalue Problem
GEVP	Generalized Eigenvalue Problem
$V(x)$	Lyapunov function
x_{eq}	Equilibrium state
LTI	Linear Time Invariant
ISE	Integral of Square of Error
J	Performance index function
Q	State weighting matrix
R	Input weighting matrix
x_0	Initial state
MIMO	Multi Input Multi Output
K	State feedback gain matrix
C_p	Power coefficient
VAWT	Vertical Axis Wind Turbine
HAWT	Horizontal Axis Wind Turbine
TSR	Tip Speed Ratio
NREL	National Renewable Energy Laboratory

DOF	Degree of Freedom
FAST	Fatigue, Aerodynamics, Structures and Turbulence
A	State matrix
B	Control input gain matrix
B_d	Disturbance input gain matrix
C	Output matrix
Δx	Small perturbation in state vector
$\Delta \dot{x}$	Small perturbation in time derivative of state vectors
Δu	Small perturbation in control input vector
Δu_d	Small perturbation in disturbance input vector
Δy	Small perturbation in measured output vectors
$\Delta \Omega$	Perturbed rotor speed
$\Delta \beta$	Perturbed collective pitch angle
Δw	Perturbed hub height horizontal wind speed
Ω_0	Rotor speed at the operating point
β_0	Collective pitch angle at the operating point
w_0	Hub height horizontal wind speed at the operating point
PI	Proportional Integral
PID	Proportional Integral Derivative
GSPI	Gain Scheduling Proportional Integral
$T_c(s)$	Closed loop transfer function
$P(s)$	Characteristic equation
GK	Gain correction factor
SISO	Single Input Single Output

CHAPTER ONE
INTRODUCTION

1.1 The History of the LMI

The History of the LMIs began in 1890, when Lyapunov published his doctoral thesis. Consider the following homogeneous LTI system:

$$\dot{x} = Ax \tag{1.1}$$

The system given in Equation 1.1 is stable, if a matrix $P = P^T > 0$ provides the following inequality

$$A^T P + PA < 0 \tag{1.2}$$

Equation 1.2 called Lyapunov inequality is a special form of LMI. The first LMI could be solved analytically. Select a matrix $Q = Q^T > 0$ and equate Equation 1.2 to negative of this selected matrix. Solve the following equation:

$$A^T P + PA = -Q \tag{1.3}$$

If the system in Equation 1.1 is stable, the solution of Equation 1.3 guarantees that $P = P^T$ is positive definite. In summary, the first LMI in history was used to analyze the stability of a system. The next big step came in the 1940s. In the Soviet Union, Lur'e, Postnikov and others, applied Lyapunov's method to some special practical problems in particular, the problem of a non-linearity in the actuator and the stability problem of a control system in control engineering. Stability criterion was in the LMI formulation. For small systems (second, third order), these inequalities were reduced to polynomial inequalities controlled by manually. In summary, Lur'e and others have applied Lyapunov's method to practical control engineering for the first time. The next breaking point came in the early 1960s. Yakubovich, Popov, Kalman and other researchers have succeeded in reducing the solution of the rising LMIs in the problem of Lur'e to simple graphical criterion. This is called Positive Real Lemma showed PR Lemma. This criterion could be applied to higher order systems to systems which have more than one nonlinearity. PR Lemma and extensions were intensively studied in the second half of the 1960's and there were relevant ideas: passivity, the small-gain criteria and quadratic optimal control. In the 1970s, it was known that LMI in PR Lemma is not solely graphical, but solved through a definite algebraic Riccati equation (ARE) solution. In quadratic optimal control, J.C. Willems led LMI in a 1971 article [1]:

$$\begin{bmatrix} A^T P + PA + Q & PB + C^T \\ B^T P + C & R \end{bmatrix} \geq 0 \tag{1.4}$$

By 1971, researchers knew a few methods for solving the special types of LMI: direct (for small systems), graphical methods and solution of Lyapunov or Riccati equations. These methods are closed-form or analytical solutions. Most control researchers and engineers think that the Riccati equation has analytical solution but it depends entirely on the size of the problem and on not having a specific problems. “The LMIs that arise in system and control theory can be formulated as convex optimization problems that are amenable to computer solutions.” This observation has some important consequences. Most importantly, we can reliably solve most LMIs that are not analytical solutions. This observation was clearly made by a few researchers. Pyatnitski and Skorodinski reduced Lur’e’s original problem to a convex optimization problem that LMIs required. It was solved using the ellipsoid algorithm. Finally, powerful and efficient interior point methods were developed to solve the LMIs rising in system and control theory. In 1984, N. Karmarkar described a new linear programming algorithm. This algorithm is very efficient in practice when compared with the ellipsoid method. Then in 1988 Nesterov and Nemirovski developed the interior point method. It was directly applied to convex problems involving LMIs.

1.2 Literature Review

Linear matrix inequalities (LMIs) have been frequently used in the areas of state feedback optimal control and robust control in control engineering. The power of LMIs is attributed to the development of efficient convex interfaces and the development of solvers. This is a great source of motivation. The success of LMIs in solving convex optimization problems by using computer tools is highlighted in [2] and [3]. In addition, the use of LMI techniques in student projects is also mentioned in these articles. In [4], the LMI technique is used for the design of robust state feedback control in control education for mechatronics. The robust control method is considered in the MSc study programs. Design of robust control for modular servo system of mechatronics laboratory plant using Lyapunov stability theory leading to LMI solution is discussed. Robust control theory is frequently used in pulse width modulation (PWM) based dc-dc converters. In [5], a LMI based robust H_∞ control design to control the dc-dc boost converter is presented in the photovoltaic system. A 6 kW PV system is considered an application to test the designed controller. The designed method considers the converter uncertainties and the model as a convex polytope, and achieves H_∞ performance, disturbance rejection at the same time system stability. A robust LQR control with a LMI approach for a PWM based dc-dc converter is designed in [6]. The conventional LQR control does not make sure of robust stability, if the system is highly uncertain. The use of LMI is recommended to optimize the performance index in order to implement LQR control in uncertain converter cases. As a result, a new robust method is derived for the dc-dc converter. LMI-LQR is compared to the classic LQR control when a boost converter is designed. In [7], an LMI approach is presented for the decentralized stability of multi machine power systems. LMI based state feedback optimal controller design for asymmetric structures with seismically induced stiffness uncertainties is designed in [8]. Various control methods for power regulation in wind turbines, mitigating structural loads and reducing metal fatigue have been proposed in the literature. In [9], robust collective pitch controller is designed for the National Renewable Energy Laboratory (NREL) 5 MW nonlinear wind turbine utilizing LMI techniques. Polytopic model is used. The LMI approach allows to add constraints such as H_∞, H_2, H_∞ / H_2 trade-off and pole assignment in design. In [10] an advanced LMI based LQR design is also proposed for voltage control of the grid-connected wind farm. The LQR problem is reformulated to find a common Lyapunov function. This is achieved by expressing the control optimization problem in terms of the LMI constraints. The solution of LMI equations requires the form of a Lyapunov function. This function not only ensures stability but also provides the desired performance.

1.3 Statement of the Problem

Linear matrix inequalities are powerful design tools in control engineering. Design features and constraints can be expressed in terms of LMI. Once formulated as LMI, this problem can be solved by convex optimization methods. Various control engineering problems can be written the LMI form and written as the constraint of the optimization problem. In this study, Lyapunov stability analysis and optimal control problem is discussed for linear systems. Stability analysis and state feedback controller design are done by using LMI formulation. Special attention is given to wind turbine blade pitch control design to regulate generator speed by using LMI based optimal controller.

1.4 Thesis Contribution

Main contribution of the thesis is to design LMI based optimal controller for a wind turbine and discussion of the simulation results.

1.5 Thesis Structure

This thesis is organized as follows. Chapter one presents the thesis in general. LMI history, literature review, problem statement, contribution and structure of the thesis are given. Chapter two introduces LMI formulation, LMI problems and Lyapunov stability method for linear systems. In addition, Lyapunov stability method is formulated as LMI. An introduction to the state feedback control system and linear quadratic regulator (LQR) is given and LMI formulation of LQR problem is discussed in chapter three. A wind turbine model selected as a case study in chapter four. Information about wind energy is provided at the introduction level. Problem definition, modeling and linearization issues are discussed. An optimal controller is designed by using the LMI formulation. At the end of the chapter, comparison is made between the optimal and baseline controllers. In the end, conclusion and possible future studies are given in chapter five.

CHAPTER TWO

LINEAR MATRIX INEQUALITIES (LMIs) FOR CONTROL SYSTEMS

2.1 Linear Matrix Inequalities (LMIs) for Control Systems

2.1.1 Linear matrix inequalities

A linear matrix inequality (LMI) has the following form: [11]

$$F(x) \triangleq F_0 + \sum_{i=1}^{m} x_i F_i > 0 \tag{2.1}$$

where $x \in R^m$ is the variable and the symmetric matrices $F_i = F_i^{\ T} \in R^{nxn}, i = 1, \dots, m$ are given. One of the main feature of a LMI is convexity [12] and nonlinear (convex) inequalities can be converted to LMI form using Schur complements. The structure of the LMI is given below:

$$\begin{bmatrix} Q(x) & S(x) \\ S(x)^T & R(x) \end{bmatrix} > 0 \tag{2.2}$$

where $Q(x) = Q(x)^T, R(x) = R(x)^T$, and $S(x)$ depend on x, equivalent to

$$R(x) > 0, \quad Q(x) - S(x)R(x)^{-1}S(x)^T > 0 \tag{2.3}$$

The set of nonlinear inequalities in Equation 2.3 can be represented as the LMI in Equation 2.2. In some control problems, variables can be matrices. Consider the following Lyapunov inequality:

$$A^T P + PA < 0 \tag{2.4}$$

where $A \in R^{nxn}$ is given and $P = P^T$ is variable. Equation 2.4 is a LMI in variable P. Another example, consider the following quadratic matrix inequality:

$$A^T P + PA - (PB + C^T)R^{-1}(B^T P + C) + Q \geq 0 \tag{2.5}$$

where $A, B, C, Q = Q^T, R = R^T > 0$ are given matrices of appropriate sizes, and $P = P^T$ is the variable. It can be represented as LMI by using Schur complements

$$\begin{bmatrix} A^TP + PA + Q & PB + C^T \\ B^TP + C & R \end{bmatrix} \geq 0 \tag{2.6}$$

2.1.2 Optimization problems involving LMIs

A LMI is convex constraint in a semidefinite programming (SDP) [13] which is a subfield of convex optimization is in the following form:

$$\begin{cases} minimize & c^Tx \\ subject\ to & x_1F_1 + \cdots + x_nF_n + G \leq 0\,, \\ & Ax = b \end{cases}$$

where $G, \ldots, F_n \in R^{kxk}$ symmetric matrices, and $A \in R^{pxn}$. The inequality constraint in SDP is a LMI. There are three generic LMI problems which are feasibility, eigenvalue, generalized eigenvalue. LMI in Equation 2.1 said to be feasible if there exists a x^{feas} such that $F(x^{feas}) > 0$. Lyapunov stability problem can be considered as LMI feasibility problem. Minimizing the maximum eigenvalue of a matrix subject to LMI constraint is an eigenvalue problem (EVP) which is following form:

$$\begin{cases} minimize & c^Tx \\ subject\ to & F(x) > 0 \end{cases}$$

where F an affine function of x. Another form of EVP is as the following:

$$\begin{cases} minimize & \gamma \\ subject\ to & A(x, \gamma) > 0 \end{cases}$$

Consider following EVP problem:

$$\begin{cases} minimize & \gamma \\ subject\ to & \begin{bmatrix} -A^TP - PA - C^TC & PB \\ B^TP & \gamma I \end{bmatrix} > 0, \\ & P > 0 \end{cases}$$

where the matrices $A \in R^{nxn}, B \in R^{nxp}$, and $C \in R^{mxn}$ are given, P and γ are the variables of optimization. Finally, the generalized eigenvalue problem (GEVP) is the following form:

$$\begin{cases} minimize & \gamma_{max}(A(x), B(x)) \\ subject\ to & (x) > 0, \quad C(x) > 0 \end{cases}$$

2.2 Lyapunov Stability Method for Linear Systems

In 1892, the Russian mathematician Alexandr Mikhailovich Lyapunov published his doctoral dissertation [14] entitled "The General Problem of the Stability of Motion". Two methods were defined by Lyapunov in his doctoral dissertation for stability analysis. These are first and second (direct) methods. Second method is used effectively. It can be applied to all sorts of systems which are linear, nonlinear, time varying and time invariant. In addition, it does not require analytical solution of state equations. Instead of that, sign behavior of a function V defined by Lyapunov is analyzed. The function V is a function of state variables and time, V(x, t). It is fixed sign scalar function. In this study, we assume that the function V does not depend on time explicitly. The definition of Lyapunov function $V(x)$ is as follows:

$$V(x) > 0 \tag{2.7}$$

$$\frac{dV(x)}{dt} < 0 \tag{2.8}$$

$$V(x_{eq}) = 0 \tag{2.9}$$

where x_{eq} equilibrium state. As it is seen in Equation 2.7, $V(x)$ is positive definite. Its time derivative is negative definite and it is zero at equilibrium state x_{eq}. If a linear system has a Lyapunov function, the system is asymptotically stable in equilibrium state x_{eq}. Consider the following linear time invariant (LTI) free system:

$$\dot{x} = Ax, \ x_{eq} = 0 \tag{2.10}$$

where $x \in R^n$ is state vector, $A \in R^{nxn}$ is system matrix. It is assumed that A is nonsingular and therefore the only equilibrium state is $x_{eq} = 0$ (origin in state space). For LTI system in Equation 2.10 possible Lyapunov function [15] is

$$V(x) = x^T P x \tag{2.11}$$

where $P = P^T$ is positive definite matrix. Time derivative of $V(x)$ along any trajectory is

$$\frac{dV(x)}{dt} = \dot{x}^T P x + x^T P \dot{x} \tag{2.12}$$

$$= (Ax)^T P x + x^T P (Ax) \tag{2.13}$$

$$= x^T(A^TP + PA)x \tag{2.14}$$

The system given in Equation 2.10 is assumed to be asymptotically stable therefore Equation 2.14 must be negative definite.

$$x^T(A^TP + PA)x < 0 \tag{2.15}$$

$$A^TP + PA < 0 \tag{2.16}$$

Equation 2.16 is called Lyapunov inequality. If there exists a matrix $P = P^T > 0$ providing Equation (2.16), state of equilibrium x_{eq} of the system is asymptotically stable. Since matrix P is not unique, Lyapunov inequality can be solved algebraically to find any positive definite P matrix. Let us write Equation 2.16 as the following:

$$A^TP + PA = -Q \tag{2.17}$$

where $Q = Q^T > 0$ is any matrix. When Equation 2.17 is solved analytically if a matrix $P = P^T > 0$ is found, state of equilibrium x_{eq} of the system is asymptotically stable.

Example 2.1 In this example, Lyapunov stability analysis is done. Consider following second order homogeneous LTI system:

$$\frac{dx}{dt} = \begin{bmatrix} -1 & 0 \\ 2 & -5 \end{bmatrix} x,\ x_{eq} = 0 \tag{2.18}$$

Let us select matrix Q in Equation 2.17 as identity matrix which is simplest positive definite symmetric matrix.

$$A^TP + PA = -I \tag{2.19}$$

$$\begin{bmatrix} -1 & 2 \\ 0 & -5 \end{bmatrix} \begin{bmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{bmatrix} + \begin{bmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{bmatrix} \begin{bmatrix} -1 & 0 \\ 2 & -5 \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}$$

$$P = \begin{bmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{bmatrix} = \begin{bmatrix} 0.5666 & 0.0333 \\ 0.0333 & 0.2 \end{bmatrix}$$

$$p_{11} = 0.5666 > 0, \qquad p_{11} * p_{22} - {p_{12}}^2 = 0.1122 > 0$$

Since determinants of all successive principal minors of the symmetric matrix P are positive, it is positive definite. Therefore equilibrium state x_{eq} of the system is asymptotically stable.

2.3 Formulating the Stability Problem with Linear Matrix Inequalities (LMIs)

Consider free LTI system in Equation 2.10. It is asymptotically stable system, if there exists a matrix as $P = P^T > 0$ satisfying Lyapunov inequality in Equation 2.16. Lyapunov inequality is a LMI in matrix P. It can be written explicitly in LMI form in Equation 2.1. Consider the following matrix P and system matrix A:

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix},\ P = \begin{bmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{bmatrix}$$

$$A^T P + PA = \begin{bmatrix} a_{11} & a_{21} \\ a_{12} & a_{22} \end{bmatrix} \begin{bmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{bmatrix} + \begin{bmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} < 0 \tag{2.20}$$

$$\begin{bmatrix} 2p_{11}a_{11} + 2p_{12}a_{21} & p_{12}(a_{11} + a_{22}) + p_{22}a_{21} + p_{11}a_{12} \\ p_{12}(a_{11} + a_{22}) + p_{22}a_{21} + p_{11}a_{12} & 2p_{12}a_{12} + 2p_{22}a_{22} \end{bmatrix} < 0$$

(2.21)

$$p_{11} \begin{bmatrix} 2a_{11} & a_{12} \\ a_{12} & 0 \end{bmatrix} + p_{12} \begin{bmatrix} 2a_{21} & a_{22} + a_{11} \\ a_{22} + a_{11} & 2a_{12} \end{bmatrix} + p_{22} \begin{bmatrix} 0 & a_{21} \\ a_{21} & 2a_{22} \end{bmatrix} <0 \tag{2.22}$$

Equation 2.22 is explicit form of Lyapunov inequality. This equation is in the form of LMI in Equation 2.1. Elements of matrix P are variables of LMI, so Lyapunov stability problem for linear system can be considered LMI feasibility problem. LMI $A^T P + PA < 0$ is feasible and linear system in Equation 2.10 is asymptotically stable if there exists a $P = P^T > 0$ satisfying LMI in Equation 2.12.

Example 2.2 In this example Lyapunov stability problem is solved as LMI feasibility problem. Consider second order LTI system which is given in Example 2.1. LMI constraints are the following:

$$A^T P + PA < 0,\ P = P^T > 0 \tag{2.23}$$

where

$$A = \begin{bmatrix} -1 & 0 \\ 2 & -5 \end{bmatrix}, \qquad P = \begin{bmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{bmatrix}$$

There are commercial and noncommercial softwares to solve SDP. For example Matrix Laboratory (Matlab) / LMIToolbox [16] is a commercial computer software. Yalmip [17] which is developed to model and solve SDP problems is a fast and noncommercial computer software and SeDuMi solver [18] is used for solution of

LMIP. Yalmip and SeDuMi are used to solve SDP problems. A computer software is written as in Table 2.1.

Table 2.1 Computer software for LMI feasibility problem

```
yalmip('clear');
A=[-1 0;2 -5];
P=sdpvar(2,2);
F=[P>=0, A'*P+P*A<=0];
optimize(F);
Pfeasible=value(P);
```

Command 'yalmip('clear')' is to clean up some memory structures and old solution structures. 'A' is system matrix which is available. Command 'sdpvar' is one of the most important commands used to define optimization variables. Yalmip assumes that square matrices are symmetric. 'F' defines set of LMIs. Command 'optimize' is used to solve SDP. Diagnostics of 'optimize' command is the optimize (constraints, objective, options) where 'constraints', 'objective' and 'options' state the conditions that variables must satisfy, function or variable to minimize and usage options of the command respectively. Solution of the SDP given in Table 2.1 is the following

$$P_{feasible} = \begin{bmatrix} 0.7305 & 0.0551 \\ 0.0551 & 0.1844 \end{bmatrix}$$

where $P_{feasible}$ is one of the solutions satisfying Lyapunov inequality.

$$p_{11} = 0.7305 > 0, \qquad p_{11} * p_{22} - {p_{12}}^2 = 0.1317 > 0$$

Since determinants of all successive principal minors of the matrix P are positive, state of equilibrium x_{eq} of the system is asymptotically stable. Lyapunov stability analysis was done for LTI free system in Equation 2.7. Lyapunov stability analysis is done for state feedback control system. Consider the following LTI non homogeneous system:

$$\dot{x} = Ax + Bu, \; x_{eq} = 0 \tag{2.24}$$

where A is system matrix, B is input matrix and x_{eq} is state of equilibrium. State feedback control law is the following:

$$u = -Kx \tag{2.25}$$

where K is static gain matrix. Substituting Equation 2.22 into Equation 2.21,

$$\dot{x} = Ax - BKx = (A - BK)x$$

where $(A - BK) \in R^{nxn}$ is closed loop system matrix. Lyapunov inequality in Equation 2.10 is written in terms of $(A - BK)$

$$(A - BK)^T P + P(A - BK) < 0 \tag{2.26}$$

Equation 2.26 is Lyapunov inequality for state feedback control system combining the open-loop system in Equation 2.24 with the control law in Equation 2.25. Since Equation 2.26 is not convex, it is not a LMI in K and P. Let us pre-multiply and post-multiply it by P^{-1}

$$P^{-1}(A - BK)^T + (A - BK)P^{-1} < 0 \tag{2.27}$$

Substitute $P = Q^{-1}$ in Equation 2.27 where $Q = Q^T$

$$Q(A - BK)^T + (A - BK)Q < 0 \tag{2.28}$$

Equation 2.28 can be expanded as

$$QA^T - QK^T B^T + AQ - BKQ < 0 \tag{2.29}$$

Define $Y = KQ$ and substitute in Equation 2.29.

$$AQ + QA^T - BY - Y^T B^T < 0 \tag{2.30}$$

Equation 2.30 is a LMI in Q and Y. If it is feasible in Q and Y, equilibrium state x_{eq} of the system in Equation 2.21 is asymptotically stable. It can be said that control input $u = -Kx = -YQ^{-1}x$ stabilizes the closed loop state feedback control system.

Example 2.3 In this example Lyapunov stability problem for the following system is solved as LMI feasibility problem. Consider the following fourth order LTI non homogeneous state feedback control system.

$$\dot{x} = Ax + Bu,\ u = -Kx,\ x_{eq} = 0 \tag{2.31}$$

The system and input matrices are given as follows:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 20.58 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -0.49 & 0 & 0 & 0 \end{bmatrix},\ B = \begin{bmatrix} 0 \\ -1 \\ 0 \\ 0.5 \end{bmatrix}$$

Eigenvalues which are poles of the characteristic equation are $\lambda_1 = \lambda_2 = 0, \lambda_3 = 4.5365,\ \lambda_4 = -4.5365$. Open loop system has multiple poles at the origin and a pole on the right hand side of imaginary axis in complex plane, so the system is not stable. If following LMI is feasible in Q and Y, state of equilibrium x_{eq} of the system is stable.

$$AQ + QA^T - BY - Y^T B^T < 0, Q = Q^T > 0 \tag{2.32}$$

A computer software m-file is written to solve LMI feasibility problem as follows,

Table 2.2 Computer software for LMI feasibility problem

```
yalmip('clear');
A=[0 1 0 0;20.58 0 0 0;0 0 0 1;-0.49 0 0 0];
B=[0;-1;0;0.5];
Q=sdpvar(4,4);
Y=sdpvar(1,4);
F=[Q>=0];
F=[F,A*Q+Q*A'-B*Y-Y'*B'<=0];
optimize(F);
P=inv(value(Q));
K=value(Y)*inv(value(Q));
eig(A-B*K);
```

$$P = \begin{bmatrix} 74.6262 & 14.3458 & 7.4112 & 7.5138 \\ 14.3458 & 3.7962 & 1.5203 & 2.0140 \\ 7.4112 & 1.5203 & 1.3991 & 0.9099 \\ 7.5138 & 2.0140 & 0.9099 & 1.6226 \end{bmatrix}$$

$$K = [-109.8288 \quad -16.6327 \quad -7.9557 \quad -7.6694]$$

Eigenvalues of closed loop state feedback control system are $\lambda_{1,2} = -5.9768 \pm j6.1971,\ \lambda_{3,4} = -0.4222 \pm j0.9346$. Real part of the eigenvalues are negative, so control input, $u = -Kx$ stabilizes the system.

CHAPTER THREE
STATE FEEDBACK CONTROL AND QUADRATIC OPTIMAL REGULATOR

3.1 State Feedback Control

3.1.1 State space model of LTI system

In conventional control, transfer function model is used to explain relationship between input dynamic and output dynamic. It can be used for only single input single output (SISO) LTI systems under zero initial conditions. In modern control, state space representation model is used widely since a system which is either SISO and LTI or multi input multi output (MIMO) time varying and nonlinear can be represented in state space plane. A system is represented by a set of first order differential equations in state space. States are time domain variables which give an idea about future of the system. If state space model and values of states are known at initial time t_0, future values of states can also be known. Generally state variables are system dynamics such as position, speed, voltage, temperature, pressure, and similar. A dynamical LTI system in state space is the following:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \vdots \\ \dot{x}_n \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} b_{11} & \dots & b_{1p} \\ b_{21} & \dots & b_{2p} \\ \vdots & \vdots & \vdots \\ b_{n1} & \dots & b_{np} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_p \end{bmatrix} \tag{3.1}$$

$$\dot{x} = Ax + Bu \tag{3.2}$$

Output equation is the following:

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & \dots & c_{1n} \\ c_{21} & c_{22} & \dots & c_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ c_{r1} & c_{r2} & \dots & c_{rn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} d_{11} & \dots & d_{1p} \\ d_{21} & \dots & d_{2p} \\ \vdots & \vdots & \vdots \\ d_{r1} & \dots & d_{rp} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_p \end{bmatrix} \tag{3.3}$$

$$y = Cx + Du \tag{3.4}$$

where $x \in R^n$ and $u \in R^p$ are state and input vectors respectively; $A \in R^{nxn}$, $B \in R^{nxp}$, $C \in R^{rxn}$ and $D \in R^{rxp}$ are matrices. A is system matrix which shows how system dynamics affect each other. B is input matrix which shows how system inputs affect states. C is output matrix which shows how states affect outputs.

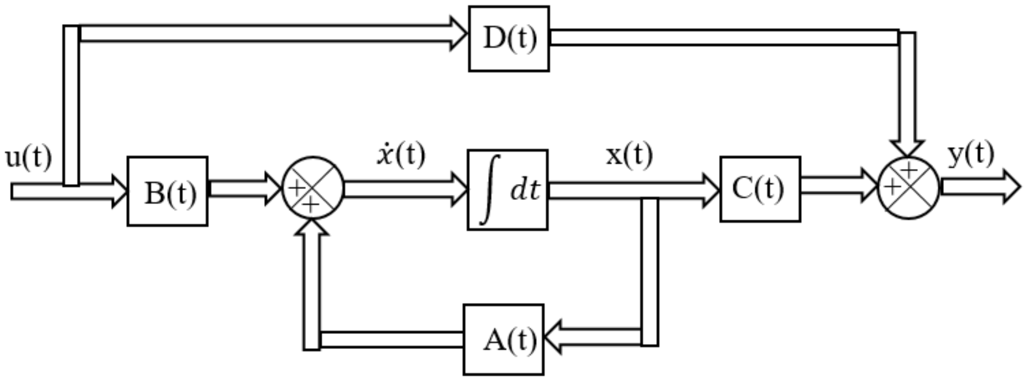

Figure 3.1 Block diagram of state space model of a LTI system

Example 3.1 In this example, modelling of the second order linear system in state space is done. Consider the following mass damper spring system.

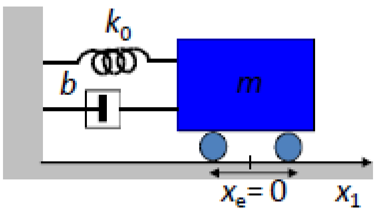

Figure 3.2 Mass damper spring system

Linear governing equation is the following

$$m\ddot{x} + b\dot{x} + k_0 x = 0 \tag{3.5}$$

$$x = x_1, \text{ position of the cart.} \tag{3.6}$$

$$\dot{x}_1 = x_2, \text{ speed of the cart.} \tag{3.7}$$

$$\dot{x}_2 = -\frac{1}{m}[k_0 x_1 + b x_2] \tag{3.8}$$

State space representation of the system is the following:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k_0}{m} & -\frac{b}{m} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{3.9}$$

where $m = 5\,kg$, $k_0 = 150\,N/m$, and $b = 15\,N\,s/m$. If numerical values are substituted into state space model, it becomes

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -30 & -3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{3.10}$$

3.1.2 Stability analysis of a LTI system in state space

Eigenvalues of the system matrix A are roots of characteristic equation of the system. If real parts of the eigenvalues are negative, the system is stable. Characteristic equation is the following:

$$|\lambda I - A| = 0 \tag{3.11}$$

Example 3.2 In this example, characteristic equation and eigenvalues of the system are found. Consider the following system matrix:

$$A = \begin{bmatrix} 0 & 1 \\ -30 & -3 \end{bmatrix}$$

Characteristic equation is

$$|\lambda I - A| = \left|\begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} - \begin{bmatrix} 0 & 1 \\ -30 & -3 \end{bmatrix}\right| = \left|\begin{bmatrix} \lambda & -1 \\ 30 & \lambda + 3 \end{bmatrix}\right| = \lambda(\lambda + 3) + 30 =$$

$$\lambda^2 + 3\lambda + 30 = 0 \tag{3.12}$$

Roots of the characteristic equation are $\lambda_{1,2} = -3 \pm j5.2678$. Since real part of the roots is negative, the system is stable.

3.1.3 Solution of LTI state equations

State equation is made up of a system of first order differential equations. Consider following LTI homogeneous state equation. Laplace transform method [15] is applied to solve state equation.

$$\dot{x} = Ax \tag{3.13}$$

Taking Laplace transform of both sides of Equation 3.13.

$$sX(s) - x(0) = AX(s) \tag{3.14}$$

$$(sI - A)X(s) = x(0) \tag{3.15}$$

$$X(s) = (sI - A)^{-1}x(0) \tag{3.16}$$

Taking inverse Laplace transform of $X(s)$

$$x(t) = L^{-1}[(sI - A)^{-1}]x(0) = e^{At}x(0) = \Phi(t)x(0) \tag{3.17}$$

where $\Phi(t) \in R^{nxn}$ is called state transition matrix.

$$\Phi(t) = e^{At} = L^{-1}[(sI - A)^{-1}] \tag{3.18}$$

Example 3.3 In this example, we solve LTI homogeneous state equations. Consider the following second-order LTI homogeneous system.

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 10 & -3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{3.19}$$

State matrix is the following:

$$A = \begin{bmatrix} 0 & 1 \\ 10 & -3 \end{bmatrix}$$

$$(sI - A) = \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} - \begin{bmatrix} 0 & 1 \\ 10 & -3 \end{bmatrix} = \begin{bmatrix} s & -1 \\ s - 10 & 3 \end{bmatrix} \tag{3.20}$$

Inverse of $(sI - A)$ is the following

$$(sI - A)^{-1} = \frac{1}{(s+5)(s-2)} \begin{bmatrix} s+3 & 1 \\ 10 & s \end{bmatrix} = \begin{bmatrix} \frac{s+3}{(s+5)(s-2)} & \frac{1}{(s+5)(s-2)} \\ \frac{10}{(s+5)(s-2)} & \frac{s}{(s+5)(s-2)} \end{bmatrix} \tag{3.21}$$

State transition matrix is the following:

$$\Phi(t) = e^{At} = L^{-1}[(sI - A)^{-1}] = \begin{bmatrix} \frac{2}{7}e^{-5t} + \frac{5}{7}e^{2t} & -\frac{1}{7}e^{-5t} + \frac{1}{7}e^{2t} \\ -\frac{10}{7}e^{-5t} + \frac{10}{7}e^{2t} & \frac{5}{7}e^{-5t} + \frac{2}{7}e^{2t} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{3.22}$$

$$x_1 = \frac{2}{7}e^{-5t} + \frac{5}{7}e^{2t},\ x_2 = -\frac{10}{7}e^{-5t} + \frac{10}{7}e^{2t} \tag{3.23}$$

Consider LTI nonhomogeneous state equation in Equation 3.2. It is written as

$$\dot{x}(t) - Ax(t) = Bu(t) \tag{3.24}$$

Pre-multiply both sides of Equation 3.24 by e^{-At}, it is obtained

$$e^{-At}[\dot{x}(t) - Ax(t)] = \frac{d}{dt}[e^{-At}x(t)] = e^{-At}Bu(t) \tag{3.25}$$

Integrating Equation 3.25 between 0 and t

$$e^{-At}x(t) - x(0) = \int_0^t e^{-A\tau}Bu(\tau)d\tau \tag{3.26}$$

$$x(t) = e^{At}x(0) + \int_0^t e^{A(t-\tau)}Bu(\tau)d\tau \tag{3.27}$$

Equation 3.27 can be written in terms of $\Phi(t)$, state transition matrix,

$$x(t) = \Phi(t)x(0) + \int_0^t \Phi(t-\tau)Bu(\tau)d\tau \tag{3.28}$$

3.1.4 Controllability

Consider LTI system in Equation 3.2. It is said to be completely state controllable if rank of the following nxn matrix is n [19]:

$$[B \quad AB \quad \ldots \quad A^{n-1}B] \tag{3.29}$$

This means that Equation 3.29 has n linearly independent column vector. Equation 3.29 is called controllability matrix.

Example 3.4 In this example, controllability of the following second-order LTI nonhomogeneous system is investigated:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 10 & -3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u \tag{3.30}$$

Controllability matrix is

$$[B \quad AB] = \begin{bmatrix} 1 & 0 \\ 0 & 10 \end{bmatrix}$$

Controllability matrix has two linearly dependent column vectors. Determinant of the controllability matrix is not zero. That is inverse of controllability matrix exists. System is completely state controllable.

3.1.5 Pole placement method

Consider regulator control problem which is found a control input u to carry states in initial condition to its equilibrium states x_{eq}. Control diagram is the following:

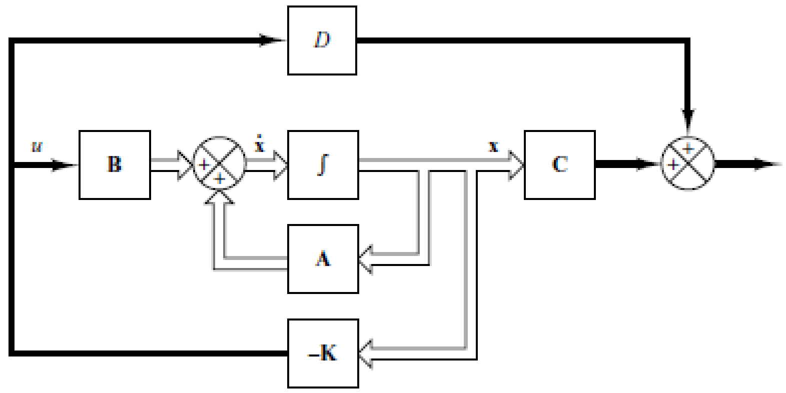

Figure 3.3 Block diagram of state feedback closed loop regulator control system

LTI nonhomogeneous system is the following:

$$\dot{x} = Ax + Bu \tag{3.31}$$

Control law is the following:

$$u = -Kx \tag{3.32}$$

where $x \in R^n$. It assumed that control input u is not a vector. Dimension of matrix K which is called gain matrix becomes R^{1xn}. As it is seen in Equation 3.32, control input u consists of scalar multiplication of elements of gain matrix K. Control input u changes the system matrix A according to desired poles. To design a controller, first desired poles which satisfy design specification should be determined. Then arbitrary pole placement method is applied. The method requires that the system in Equation 3.31 to be completely state controllable. Ackermann's formula [15] is an arbitrary pole placement method. Substitute Equation 3.32 into Equation 3.31. System matrix becomes

$$\dot{x} = Ax - BKx = (A - BK)x \tag{3.33}$$

Desired closed loop poles are $s_1,\ s_2,\ \ldots,\ s_n$. Characteristic equation is the following:

$$|sI - A + BK| = (s - s_1) \ldots (s - s_n) = s^n + \alpha_1 s^{n-1} + .. + \alpha_{n-1} s + \alpha_n = 0 \tag{3.34}$$

Ackermann's formula is the following:

$$K = [0 \quad 0 \quad \ldots \quad 1][B \quad AB \quad \ldots \quad A^{n-1}B]^{-1} \Phi(A) \tag{3.35}$$

where $\Phi(A) = A^n + \alpha_1 A^{n-1} + \cdots \alpha_{n-1} A + \alpha_n I$.

Example 3.5 In this example, state feedback controller is designed by using Ackermann's formula. Consider the following second-order LTI nonhomogeneous system:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -30 & -3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u \tag{3.36}$$

Desired poles are $s_{1,2} = -1 \pm j2$. Characteristic equation is

$$(s + 1 - j2)(s + 1 + j2) = s^2 + 2s + 5, \alpha_1 = 2, \alpha_2 = 5 \tag{3.37}$$

$$[B \quad AB]^{-1} = \begin{bmatrix} 1 & 0 \\ 0 & -\frac{1}{30} \end{bmatrix}$$

$$\Phi(A) = A^2 + 2A + 5I = \begin{bmatrix} -25 & -1 \\ 30 & -22 \end{bmatrix}$$

$$K = [0 \quad 1][B \quad AB]^{-1}\Phi(A) = \begin{bmatrix} -1 & \frac{22}{30} \end{bmatrix}$$

3.2 Quadratic Optimal Regulator

In this subsection, optimal control system is considered in view of performance index. The performance index function is based on integral of square of error (ISE) [20] which is integral of transient errors $x - x_d$ from initial time t_0 to final time t_f. Performance index function is

$$J_{ISE} = \int_{t_0}^{t_f} e^2 dt = \int_{t_0}^{t_f} (x - x_d)^2 dt \tag{3.38}$$

Quadratic optimal regulator regulates state variables to its equilibrium state x_{eq}. For LTI system, origin in state space is only equilibrium state. For regulator control problem, desired state x_d is equilibrium state. Since desired state is origin, error is instantaneous values of state variables. Equation 3.38 leads to

$$J_{ISE} = \int_{t_0}^{t_f} x^2 dt = \int_{t_0}^{t_f} (x^T x) dt \tag{3.39}$$

where $x \in R^n$. Minimization of Equation 3.39 is called least-transient error [19] problem. If the control input u is included in Equation 3.38 then ISE for control signal

$$J_{ISE} = \int_{t_0}^{t_f} (u - u_d)^2 dt = \int_{t_0}^{t_f} u^2 dt = \int_{t_0}^{t_f} (u^T u) dt \tag{3.40}$$

where $u \in R^r$. In Equation 3.40 error is difference of instantaneous control input u from desired control input u_d. Desired control input u_d is zero. Thus error is instantaneous control input. Minimization of Equation 3.40 is called least-effort [19] problem. If Equation 3.39 and Equation 3.40 are combined, the performance index is

$$J = \int_{t_0}^{t_f} (x^T Q x + u^T R u) dt \tag{3.41}$$

where $Q \in R^{nxn}$ is positive semidefinite state weighting and $R \in R^{rxr}$ is positive definite input weighting matrices respectively. Q improves transient response of control system but more control energy is required. R minimizes control input but transient error increases. Consider LTI system in Equation 3.31 and control law in Equation 3.32. Control input in Equation 3.32 is optimal control input if it minimizes performance index in Equation 3.41. Substitute Equation 3.32 into Equation 3.31

$$\dot{x} = Ax - BKx = (A - BK)x \tag{3.42}$$

Substitute Equation 3.32 into Equation 3.41

$$J = \int_{t_0}^{t_f} (x^T Q x + x^T K^T R K x) dt = \int_{t_0}^{t_f} x^T (Q + K^T R K) x dt \tag{3.43}$$

Let us set [15]

$$x^T (Q + K^T R K) x = -\frac{d}{dt} (x^T P x) \tag{3.44}$$

where $P = P^T \in R^{nxn}$ is positive definite matrix.

$$J = \int_{t_0}^{t_f} x^T (Q + K^T R K) x dt = -x^T P x \big|_{t_0}^{t_f} = -x^T(t_f) P x(t_f) + x^T(t_0) P x(t_0) \tag{3.45}$$

We assume that state feedback control input stabilizes the system in Equation 3.31. Every trajectories of the system in state space approaches its equilibrium state. That is at final time t_f, $x(t_f)$ is zero. $x(t_0) = x_0$ is initial state vector of the system at initial time t_0. Performance index J in Equation 3.45 becomes

$$J = {x_0}^T P x_0 \tag{3.46}$$

Equation 3.44 is rewritten by expanding right hand side.

$$x^T (Q + K^T R K) x = -\dot{x}^T P x - x^T P \dot{x} = -x^T [(A - BK)^T P + P(A - BK)] x \tag{3.47}$$

$$-(Q + K^T RK) = (A - BK)^T P + P(A - BK) \quad (3.48)$$

$$(A^T - K^T B^T)P + P(A - BK) + Q + K^T RK = 0 \quad \text{Let } R = T^T T \quad (3.49)$$

$$A^T P + PA + [TK - (T^T)^{-1} B^T P]^T [TK - (T^T)^{-1} B^T P] - PBR^{-1} B^T P + Q = 0 \quad (3.50)$$

To minimize performance index J, state feedback gain matrix should be

$$K = R^{-1} B^T P \quad (3.51)$$

K is called optimal state feedback gain matrix. Substitute Equation 3.51 into Equation 3.50. It is obtained that

$$A^T P + PA - PBR^{-1} B^T P + Q = 0 \quad (3.52)$$

Equation 3.52 is called reduced-matrix Riccati equation. For given A and B matrices and chosen Q and R matrices, solution of Riccati equation in Equation 3.52 gives a positive definite symmetric matrix P. Then minimized performance index J in Equation 3.46 can be calculated.

Example 3.6 State feedback controller is designed by using arbitrary pole placement and LQR methods respectively. Consider the following inverted pendulum system in Figure 3.4.

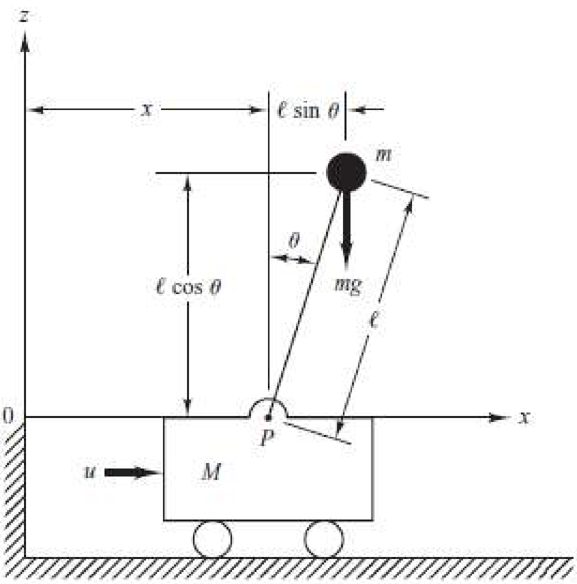

Figure 3.4 Inverted pendulum free body diagram

Dynamics of the system are nonlinear. We assume that θ is within a certain linear interval and starts in linear range. The system is linearized at equilibrium state, x_{eq}=0. Linear governing equations are the following:

$$Ml\frac{d^2\theta}{dt^2} = (M+m)g\theta - u \tag{3.53}$$

$$M\frac{d^2x}{dt^2} = u - mg\theta \tag{3.54}$$

where $M = 2\ kg,\ m = 0.1\ kg,\ l = 0.5\ m,\ g = 9.8\frac{m}{s^2}$. Numerical values are substituted into governing equations.

$$\frac{d^2\theta}{dt^2} = 20.58\theta - u \tag{3.55}$$

$$\frac{d^2x}{dt^2} = -0.49\theta + 0.5u \tag{3.56}$$

Let us define state variables x_1, x_2, x_3, x_4 as follows:

$x_1 = \theta$, angle of the pendulum (3.57)

$x_2 = \frac{d\theta}{dt}$, pendulum angular velocity (3.58)

$x_3 = x$, cart displacement (3.59)

$x_4 = \frac{dx}{dt}$, cart velocity (3.60)

The system model is the following

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 20.58 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -0.49 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ -1 \\ 0 \\ 0.5 \end{bmatrix} u \tag{3.61}$$

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u \tag{3.62}$$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 20.58 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -0.49 & 0 & 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 0 \\ -1 \\ 0 \\ 0.5 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, D = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Characteristic equation is

$$|\lambda I - A| = \left|\begin{bmatrix} \lambda & 0 & 0 & 0 \\ 0 & \lambda & 0 & 0 \\ 0 & 0 & \lambda & 0 \\ 0 & 0 & 0 & \lambda \end{bmatrix} - \begin{bmatrix} 0 & 1 & 0 & 0 \\ 20.58 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -0.49 & 0 & 0 & 0 \end{bmatrix}\right| = \left|\begin{bmatrix} \lambda & -1 & 0 & 0 \\ -20.58 & \lambda & 0 & 0 \\ 0 & 0 & \lambda & -1 \\ 0.49 & -1 & 0 & \lambda \end{bmatrix}\right|$$

(3.63)

$$\lambda^2(\lambda^2 - 20.5798) = 0 \quad (3.64)$$

Eigenvalues of the system are $\lambda_1 = \lambda_2 = 0, \lambda_3 = 4.5365,\ \lambda_4 = -4.5365$. There is a pole which is on the right hand side of the imaginary axis in the complex plane. System is not stable. For stabilization of the system and desired performance we need to design a controller. If the system is completely state controllable, arbitrary pole placement can be done. Controllability matrix is the following:

$$[B \vdots AB \vdots A^2B \vdots A^3B] = \begin{bmatrix} 0 & -1 & 0 & -20.58 \\ -1 & 0 & -20.58 & 0 \\ 0 & 0.5 & 0 & 0.49 \\ 0.5 & 0 & 0.49 & 0 \end{bmatrix}$$

Since rank of controllability matrix is 4, arbitrary pole placement can be done. Design criteria for this system is

- settling time less than 5 seconds
- Maximum overshoot M_p less than 10%

To satisfy design criteria, desired closed loop poles should be calculated.

$$s_d = \sigma \pm j\omega_d \quad (3.65)$$

$$t_s = 4T = \frac{4}{\sigma} \leq 5,\ \sigma \geq 0.8 \ \text{ Let } \sigma = 1 \quad (3.66)$$

$$M_p = e^{-(\frac{\sigma}{\omega_d})\pi} * 100 \leq 10, \omega_d = 1.36 \quad (3.67)$$

Desired poles are $s_d = -1 \pm j1.36$. Other poles are chosen arbitrarily as $s_{3,4} = -10$. A computer software is written to find a gain matrix according to desired poles.

Table 3.1 Computer software to design controller by using pole placement method

```
clc; clear all;
A=[0 1 0 0;20.58 0 0 0;0 0 0 1;-0.49 0 0 0];
B=[0;-1;0;0.5];
C=[1 0 0 0;0 0 1 0];
D=[0;0];
CO=ctrb(A,B);
r=rank(CO);
p1=-1-1.36*i;
p2=-1+1.36*i;
p3=-10;
p4=-10;
P=[p1;p2;p3;p4];
K=acker(A,B,P);
sys=ss(A-B*K,B,C,D);
T=0:0.1:6;
x0=[0.1;0;0;0];
[y,t,x]=initial(sys,x0,T);
subplot(2,1,1);plot(t,y);grid;
xlabel('t in sec');ylabel('Output, y');
subplot(2,1,2);plot(t,-K*x');grid;
xlabel('t in sec');ylabel('Control input, u');
```

Gain matrix is

$$K = [-177.9746 \quad -35.0985 \quad -29.0479 \quad -26.1969]$$

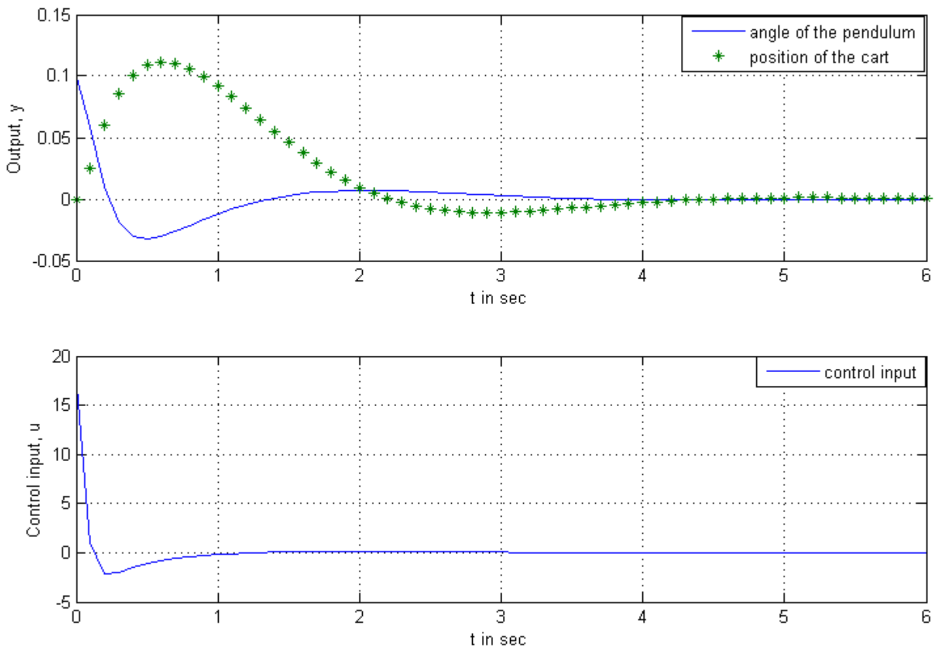

Figure 3.5 Initial value response of inverted pendulum

Settling time is about 4.3 seconds. It is seen that performance criterion is satisfied in Figure 3.5. In pole placement method, performance of control system can be improved but more control energy is required. To control both performance and control energy of the system, it is needed to use an optimal control method like LQR. For desired performance, optimal control energy can be found. A state feedback controller is designed for Example 3.6 by using LQR method. A computer software is written to design state feedback controller. State and input weighting matrices are chosen as the following:

$$Q = \begin{bmatrix} 1000 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1000 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \qquad R = [1]$$

Table 3.2 A computer software to design controller by using LQR method

```
A=[0 1 0 0;20.58 0 0 0;0 0 0 1;-0.49 0 0 0];
B=[0;-1;0;0.5];
C=[1 0 0 0;0 0 1 0];
D=[0;0];
x0=[0.1;0;0;0];
R=[1];
Q=[1000 0 0 0;0 1 0 0;0 0 1000 0;0 0 0 1];
K=lqr(A,B,Q,R);
sys=ss(A-B*K,B,C,D);
T=0:0.1:6;
[y,t,x]=initial(sys,x0,T);
subplot(2,1,1);
plot(t,y);grid;
xlabel('t in sec');
ylabel('Output, y');
subplot(2,1,2);
plot(t,-K*x'),grid;
xlabel('t in sec');
ylabel('Control input, u');
```

Gain matrix is found as the following:

$$K = [-131.7439 \quad -28.6616 \quad -31.6228 \quad -26.7820]$$

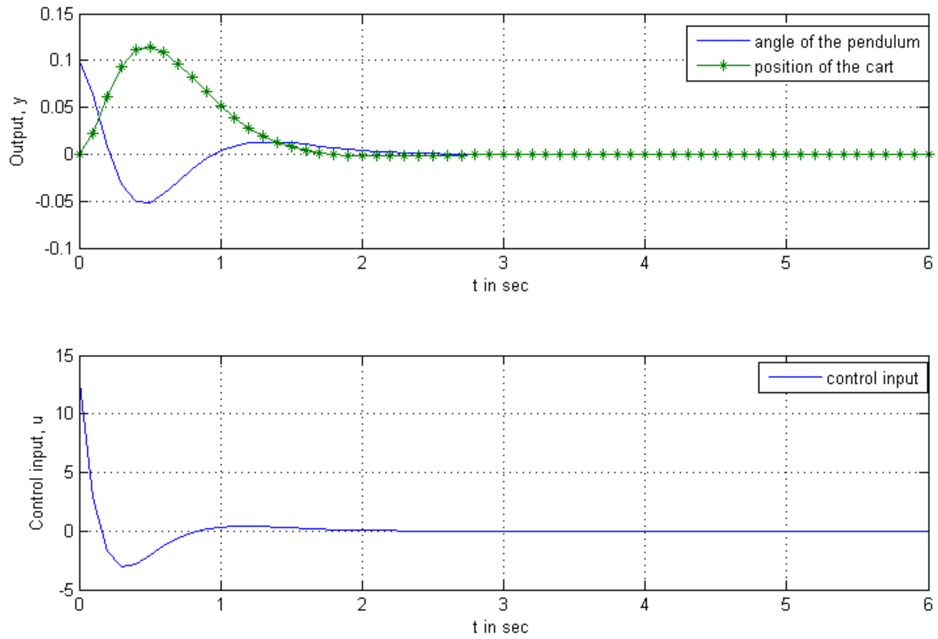

Figure 3.6 LQR method initial response of inverted pendulum control system

Settling time is about 2.7 seconds. Using less control energy, performance of the system is improved.

3.3 LMI Formulation of Quadratic Optimal Controller

Consider the LTI system in Equation 3.2. State control law is $u = -Kx$. LQR problem: find a control input u to minimize performance index $J = {x_0}^T P x_0$ for an initial condition x_0. $P = P^T > 0$ which is satisfies algebraic Riccati equation $A^T P + PA - PBR^{-1}B^T P + Q = 0$ minimizing performance index J. Optimal gain matrix is $K = R^{-1}B^T P$. Output energy is ${x_0}^T P x_0$ [11]. Upper bound of energy is γ. Output energy should be less than upper bound.

$$ {x_0}^T P x_0 < \gamma \tag{3.68} $$

Let $W = P^{-1}$. Equation 3.68 can be written as

$$ \gamma - {x_0}^T W^{-1} x_0 > 0 \tag{3.69} $$

Equation 3.69 can be written as LMI by using Schur complement

$$ \begin{bmatrix} \gamma & {x_0}^T \\ x_0 & W \end{bmatrix} > 0 \tag{3.70} $$

Quadratic matrix inequality [1] is the following:

$$A^T P + PA - PBR^{-1}B^T P + Q \geq 0 \tag{3.71}$$

Post-multiplication and pre-multiplication of Equation 3.71 by P^{-1} yields

$$P^{-1}A^T + AP^{-1} - BR^{-1}B^T + P^{-1}QP^{-1} \geq 0 \tag{3.72}$$

Write W instead of P^{-1} in Equation 3.72.

$$WA^T + AW - BR^{-1}B^T + WQW \geq 0 \tag{3.73}$$

Equation 3.73 can be written as LMI by using Schur complement.

$$\begin{bmatrix} WA^T + AW - BR^{-1}B^T & W \\ W & -Q^{-1} \end{bmatrix} \leq 0 \tag{3.74}$$

Equations 3.70 and 3.74 are LMIs. They can be used in semidefinite programming (SDP) as constraints to minimize γ. A SDP for optimal control is the following

$$\begin{cases} minimize & \gamma \\ subject\ to & \begin{bmatrix} \gamma & {x_0}^T \\ x_0 & W \end{bmatrix} > 0 \\ \begin{bmatrix} WA^T + AW - BR^{-1}B^T & W \\ W & -Q^{-1} \end{bmatrix} \leq 0, W > 0 & \end{cases}$$

A computer software is written to solve the following SDP.

Table 3.3 Computer software to solve SDP

```
yalmip('clear');
A=[0 1 0 0;20.58 0 0 0;0 0 0 1;-0.49 0 0 0];
B=[0;-1;0;0.5];
Q=[1000 0 0 0;0 1 0 0;0 0 1000 0;0 0 0 1];
R=[1];
x0=[0.1;0;0;0];
W=sdpvar(4,4);
Gamma=sdpvar(1,1);
F=[W>=0];
F=[F,[(W*A'+A*W-B*inv(R)*B') W;W -inv(Q)]<=0];
F=[F,[Gamma x0';x0 W]>=0];
optimize(F,Gamma)
P=inv(value(W))
Kopt=inv(R)*B'*P
Klqr=lqr(A,B,Q,R)
```

Gamma is 18.8642. It is minimum value of the upper bound of the energy.

$$P = 1000 \times \begin{bmatrix} 1.8864 & 0.4102 & 0.9064 & 0.5571 \\ 0.4102 & 0.0949 & 0.2109 & 0.1324 \\ 0.9064 & 0.2109 & 0.8474 & 0.3585 \\ 0.5571 & 0.1324 & 0.3585 & 0.2113 \end{bmatrix}$$

$$K_{opt} = [-131.6808 \quad -28.6618 \quad -31.6228 \quad -26.7956]$$

$$K_{lqr} = [-131.6808 \quad -28.6618 \quad -31.6228 \quad -26.7956]$$

Gain matrices are same. Solution of Riccati equation is a special solution of LMI.

CHAPTER FOUR
CASE STUDY: WIND TURBINE COLLECTIVE PITCH CONTROL

4.1 Introduction

Worldwide energy demand is increasing day by day. Renewable energy is an important issue all over the world. There has been tremendous progress in the wind power industry over the past decade and it is the most reliable renewable energy source these days. It also grows in its size and power for a better cost schedule in wind power plants. Since the early 1990s, there has been a growing interest in wind power in the area of renewable energy. Another reason for the increase in investment in wind energy is environmental concerns. Countries that produce their energy from coal and oil is more self-sufficient with alternatives such as wind power. There is no CO_2 emission from power plants that generate electricity from wind sources so it does not cause greenhouse effect. Wind turbines have some disadvantages. Wind turbines bring some noise when they are working. In modern wind turbines, producers have succeeded in reducing almost all mechanical noises. Now they work on reducing the aerodynamic noise of the rotating blades. Noise and visual impact will be less important in the future when more offshore wind turbines are deployed. Another problem is that when nature only provides enough wind, wind energy can be produced. This is not a problem for countries with large electricity networks.

Theoretically, the wind speed could be reduced to zero, the resulting energy would be the following [21]:

$$E = \frac{1}{2}mw^2 = \frac{1}{2}\rho A w^3 \tag{4.1}$$

where m is mass flow, w is wind speed, ρ is air density, A is area of rotor disk and E is maximum available kinetic energy. Practically the wind speed cannot be reduced to zero so a power coefficient C_p is defined. The power coefficient C_p is the ratio between the actual power detected and the maximum available power in the wind. A theoretical maximum, called Betz limit, is determined for the power coefficient, $C_{pmax} = 0.593$. Modern wind turbines operate close to $C_p = 0.5$. Hence the Equation

4.1 was revised as follows:

$$E = \frac{1}{2}\rho A C_p w^3 \tag{4.2}$$

If the blades are connected to a vertical shaft, the wind turbine is called the vertical axis wind turbine (VAWT). If the blades are connected to a horizontal shaft, the wind turbine is called the horizontal axis wind turbine (HAWT). HAWT dominates the market for commercial wind turbines. This study deals with HAWT. An HAWT is shown in Figure 4.1 [22]. Such turbines are described in terms of rotor diameter, number of blades, tower height, rated power and control strategies. The height of the tower and rotor diameter are important because the wind speed increases with altitude and swept area increases with rotor diameter. Rated power the maximum power allowed to produce. Control strategies are gaining importance for not exceeding the rated power and reducing the structural loads on the turbine.

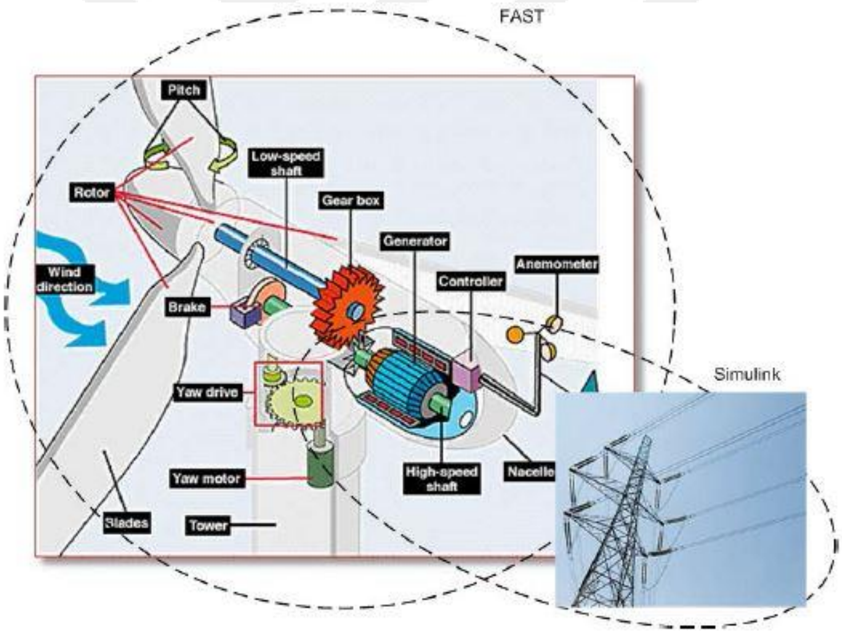

Figure 4.1 Main components of a HAWT

The number of blades is generally two or three. Two-bladed wind turbines are cheaper than three-bladed wind turbines. However three-bladed wind turbines are aerodynamically more efficient than two-bladed wind turbine. In this study, a model defined in technical report [23] is used. In model, the rated rotor speed is 12.1 rpm while the rated generator speed is 1173.7 rpm. Therefore, a gearbox is placed between the low speed shaft and the high speed shaft. Gearbox is not the only option. For example, turbines with multipole generators rotate very slowly and do not need a gearbox. A sensor measuring the wind direction is mounted on the wind turbine. This signal works with the yaw motor that constantly turns nacelle into the wind. Rotor is the wind turbine element that has been experiencing the biggest improvements in recent years. Airfoil was used on the first modern wind turbine blades. These airfoils are optimized for wind turbines. Different materials are used in the structure of the blades. These blades should be strong enough and stiff, have a high fatigue endurance limit and cheap. Today the blades are made from reinforced plastic fiberglass. Other materials are used such as laminated wood.

4.2 Statement of the Problem

In a wind turbine, the main purpose of the controller is to mitigate the fatigue loads that damage the turbine while improving its performance. Generator torque speed graph is divided into five control regions as shown in Figure 4.2. These control regions are 1, $1+\frac{1}{2}$, 2, $2+\frac{1}{2}$, 3. Region 1 is the control region before the cut-in wind speed. In this region, the generator torque is zero and no power is generated from the wind. The rotor is accelerated for startup. Region 2 is the region where power capture from the wind is maximized. Here, the optimum tip speed ratio (TSR) is set at which the power coefficient, C_p is maximum. In Region 3, the wind speed is above the rated wind speed. The control task in this region is to keep the generator torque constant. At the same time the pitch controller regulates the generator speed at rated value to capture the rated power. Region $1+\frac{1}{2}$ is the linear transition region between 1 and 2. A lower limit is placed on the speed of the generator in this region. Region $2+\frac{1}{2}$ is the linear transition region between 2 and 3. This region is needed to limit the generator speed to the speed at which the power is rated.

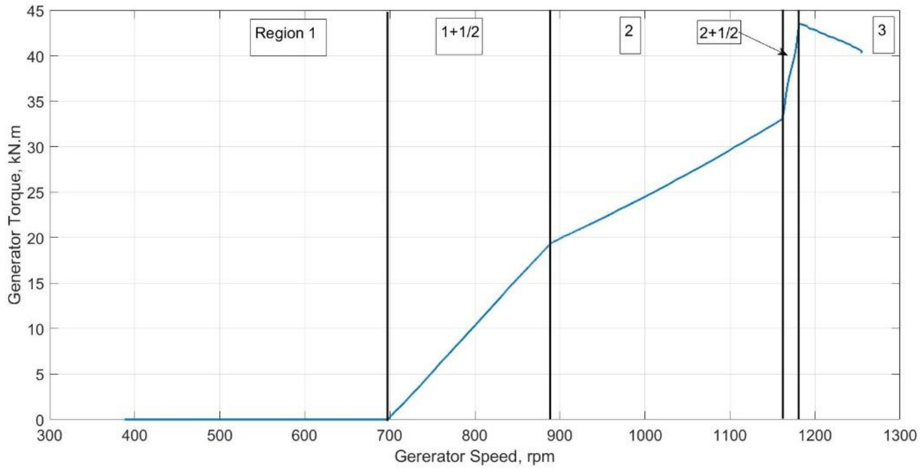

Figure 4.2 Torque versus speed response of the variable speed controller

FAST (Fatigue, Aerodynamics, Structures and Turbulence) simulations were performed at a 12.1 rpm rotor speed, at a ramp wind speed between 4 m/s and 25 m/s and at a given rotor collective blade pitch angle of 0.0 degree. For this wind turbine, the region $1 + \frac{1}{2}$ was defined as the speed range of the generator at 670 rpm, 30% more than this value at 871 rpm. The region $2 + \frac{1}{2}$ was defined as the speed range of the generator at 1161,963 rpm, rated generator speed, 1173.7 rpm.

Some turbines provide passive control as in fixed pitch stall control machines. In these machines, the blades are designed to limit the power via the blade stall in region 3 so there is no dynamic pitch control. In region 2, the generator speed is fixed. Typically, only start and stop operations are performed. Variable pitch mechanism is often used to provide better turbine power control. The blade pitch is regulated to obtain constant power in region 3. But the pitch mechanism must be fast in the presence of gusts and turbulences to ensure good power regulation.

Power coefficient $C_p(\lambda, \beta)$ is the function of λ: TSR, and β: blade pitch angle. TSR is expressed as the following:

$$\lambda = \frac{\Omega R}{w} \tag{4.3}$$

where Ω is rotor speed, w is wind speed and R is radius of the rotor. In region 2, to maximize the power output, the turbine rotation speed must be varied with the wind speed to maintain optimum TSR. Regardless of the wind speed, the turbine should operate on the TSR that gives the maximum C_p. TSR in Equation 4.3 is a function of rotor speed and wind speed. In order to maintain peak C_p, the rotor speed must change as the wind speed changes. This allows the turbine to operate near optimal C_p. This is impossible with constant speed machines. In variable speed machines, the rotor speed can be changed by controlling the generator torque. It is not practical to maintain a maximum C_p at a wide range of wind speeds. Rated wind speed is the speed at which the machine reaches maximum power output. Using the pitch mechanism at speeds above the rated wind speed, the power must be kept constant at rated power. Figure 4.3 shows the TSR versus C_p at different blade pitch angles for a NREL 5 MW wind turbine. In a definite TSR, each curve has a maximum C_p.

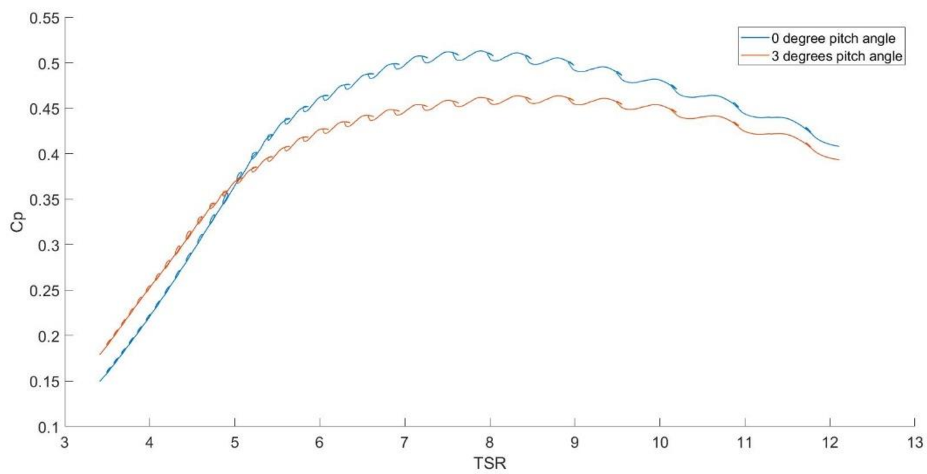

Figure 4.3 NREL 5 MW wind turbine plot of power coefficient versus TSR

Wind power is directly proportional to the cube of wind speed. If we do not control the power, this causes the generator and the power electronic system to overheat. Figure 4.4 shows NREL 5 MW wind turbine power curve.

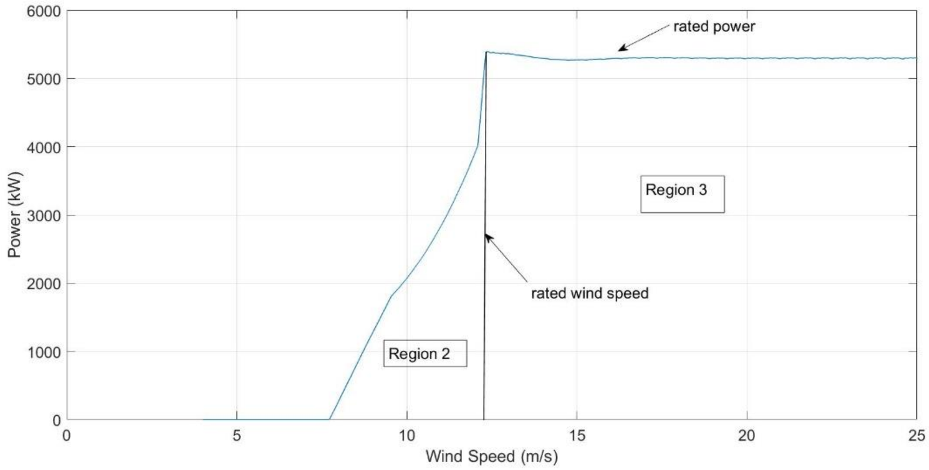

Figure 4.4 NREL 5 MW wind turbine plot of power versus wind speed

Turbine power is sustained at rated power in region 3. In variable speed machines, a constant generator torque is applied and the turbine rotation speed is maintained at the desired value using the blade pitch mechanism in region 3. Constant power generation is a great ability of variable speed machine. So all modern machines work together with variable speed and variable pitch control mechanism.

Most major commercial wind turbines have at least 3 possible control actions: blade pitch, generator torque and machine yaw. Blade pitch control action is the most effective way to control aerodynamic loads. In some machines, blade pitch angles can be set independently of each other. This control method is called individual pitch control. In other machines the pitch angle of each blade is set identically. This control method is called collective pitch control. The generator torque control action is often used to maintain a maximum C_p in region 2. It is used to add damping to the drive-train torsion modes of the turbine in region 3 as well. Yaw control action is used to respond to wind direction changes. The yaw error signal is derived from the wind direction sensor mounted on the nacelle. According to this signal, a control signal is calculated and transmitted to the yaw motor driver. Other control goals such as changing the turbine operation status, fault diagnostics, starting and stopping the machine are performed with the supervisory control system. For example, the machine parking brake is released to start a variable speed machine and blade pitch angle is reduced to a value that allows aerodynamic torque to accelerate the released rotor from full feather (90 degrees pitch angle). It should be designed with lighter and more flexible components (blades, drive train, tower etc.) to reduce the weight and cost of wind turbines. While wind turbines become larger and more flexible, structural dynamic loads and instabilities increase. Therefore, the control system should be designed specifically to mitigate these loads. Modern control algorithms should be developed. It should be tested analytically and experimentally. Supervisory control, yaw control or generator torque control are not defined in this thesis because these control methods do not have an automatic control algorithm. Instead we develop an automatic control algorithm on blade pitch control in region 3 for a variable speed variable pitch wind turbine.

4.3 Mathematical Model of the Wind Turbine

4.3.1 Model definition

The model used is a three-blade, upwind, variable-speed 5MW wind turbine model. The properties are given in Table 4.1. See [23] more features.

Table 4.1 Wind turbine specifications

Rating	5 MW
Rotor Orientation	Upwind
Control	Variable speed, collective pitch
Hub height	90 m
Rotor diameter	126 m
Cut in, rated, cut out wind speed	3 m/s, 11.4 m/s, 25 m/s
Cut-in, rated rotor speed	6.9 rpm, 12.1 rpm
Gearbox ratio	97
Rated generator speed	1173.7
Rotor, tower, nacelle mass	110 ton, 347.4 ton, 240 ton
Peak power coefficient	0.482
TSR at peak power coefficient	7.55
Rotor collective blade pitch angle at peak power coefficient	0 degree
Maximum generator torque	47402.91 Nm
Minimum blade pitch setting	0 degrees
Maximum blade pitch setting	90 degrees
Maximum absolute blade pitch rate	8 degrees / seconds

Important step in control design is the development of a simplified dynamic model. It should be decided what should be the complexity of the model. If it is too simple, significant dynamics leading to design of an unstable closed loop system or control algorithms that do not provide the desired performance is not included. If the model is too complex, the control system is too complicated to design, test, and debug. The simplified model used for the control system design must be based on the control objectives. For example, if we use blade pitch to keep turbine speed at rated speed in region 3, we need a simple model. A dynamic code like FAST [24] developed by NREL in the nonlinear turbine model is useful in design and control system simulations. It is recently modified to produce linear state space models of the turbine. FAST provides many DOFs. In our study, following DOFs were considered:

- Generator DOF (q_1)
- Drive-train rotational flexibility DOF (q_2)

Each DOF is presented as a linear model at a precise operating point according to linear first order model. Figure 4.5 shows the synthesis of the FAST model used in simulations.

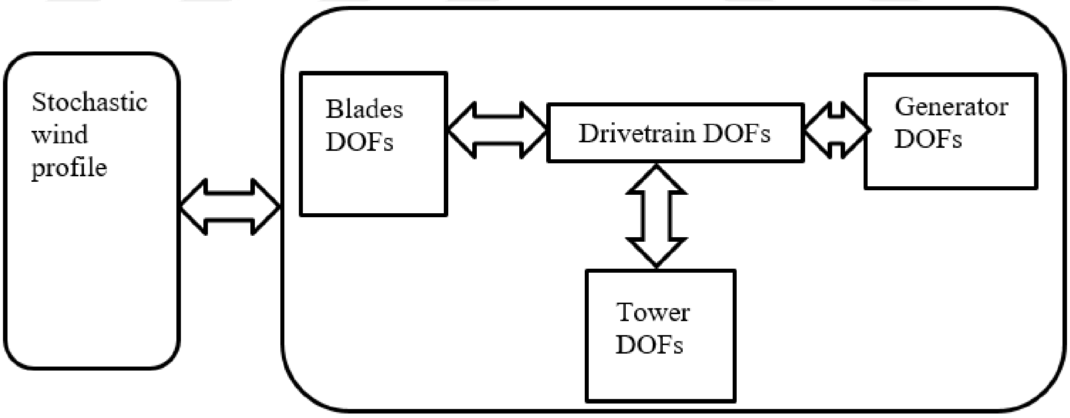

Figure 4.5 FAST model components

4.3.2 Linearization

Following simplified LTI state space model which are produced by FAST dynamic codes is used in control design.

$$\begin{cases} \Delta\dot{x} = A\Delta x + B\Delta u + B_d\Delta u_d \\ \Delta y = C\Delta x + D\Delta u + D_d\Delta u_d \end{cases} \tag{4.4}$$

where A is state matrix, B and B_d are control input and disturbance input gain matrices, C relates measured output Δy to the turbine states, D and D_d relate the measured output to the control input and the disturbance input respectively. $\Delta x, \Delta\dot{x}, \Delta u, \Delta u_d$ and Δy show small perturbations in state vector, time derivative of state vector, control input vector, disturbance input vector and measured output vector respectively from operating point values. The first step in the linearization process is to determine a steady state operation point which is a set of displacements, speeds, accelerations, and similar. The operation point depends on rotor azimuth so periodic linear models are produced around the rotor disk at certain points and averaged. For LTI control, a more accurate model can be obtained, if linearization is made in enough azimuth steps. We use a special averaging tool defined in reference [25].

Important issue in linearization is how B and B_d matrices change with the operation point. These gain matrices are related to partial derivatives of rotor aerodynamic torque according to blade pitch angle and wind speed. For various wind speeds, the rotor aerodynamic torque versus pitch angle is shown in Figure 4.6.

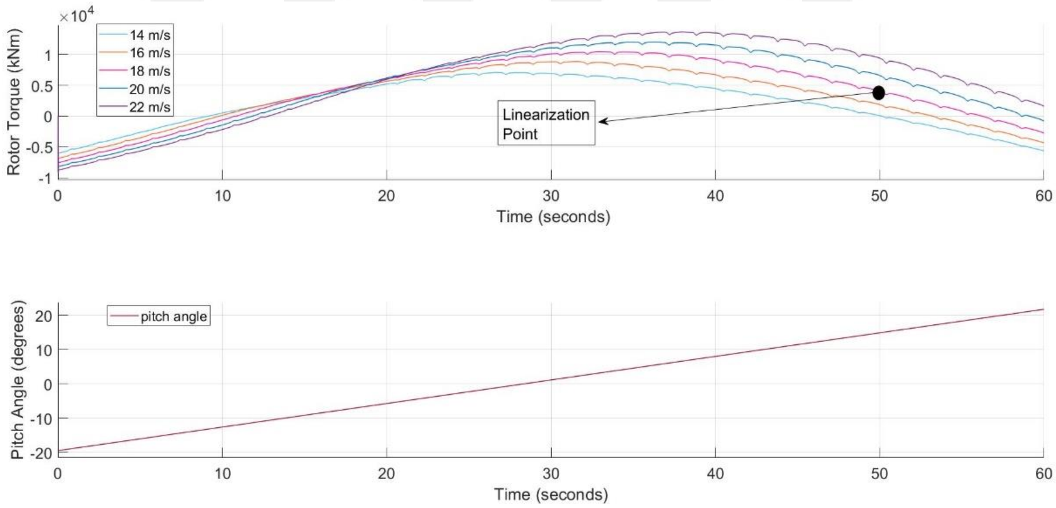

Figure 4.6 Variation of control input gains with pitch angle

An operation point is chosen in region 3. In Figure 4.6, this operation point is the point where pitch angle, wind speed and rotor speed are 14.92 degrees, 18 m/s and 12.1 rpm respectively. At this operating point, the torque versus time curve has a large negative slope. This means that matrix B is nonzero. If an operation point with a smaller pitch angle is selected for linearization, the slope of the torque versus pitch angle curve is reduced. If the 18 m/s wind speed curve is selected, the slope of the curve at the operating point where the pitch angle is 1 degree is approximately zero. If this operation point is chosen, it is impossible to approach the trim solution. It exhibits an unstable equilibrium point. At this point, the pitch input gain matrix is zero because the slope of the curve is zero. To ensure stability, operating point must be carefully examined. Variations of matrix B_d are less important because the value of this matrix is generally positive in a wide range of operations. Performance is only good for small perturbations from the operation point. Therefore the linear controller must be tested for a wide range of operation points of the turbine far from the linearization point. Firstly a controller should be designed at the operating point. Now control design tools are discussed.

4.4 Optimal Control Design by Using LMI Formulation

In this section, a state feedback rotor collective pitch controller is designed by using LMI based LQR method for NREL 5 MW onshore wind turbine in region 3. Performance of this controller is simulated in computer simulation on linearized model and FAST nonlinear wind turbine model.

4.4.1 State space representation of the linear model

Nonlinear model is linearized at the point where rotor speed is $\Omega_0 = 12.1\,rpm$, rotor collective pitch angle is $\beta_0 = 14.92°$ and hub height wind speed is $w_0 = 18\,m/s$. 2 DOFs in Appendix A page A2 was enabled during the linearization process and the nonlinear model was run at the linearization point for 60 seconds until a steady solution was reached. Then a linear model was obtained at each 10 degree rotor angle. 36 linear models were obtained for one revolution of the rotor. These models were averaged using a computer software prepared in FAST. FAST input file with linearization part is shown in Appendix A, page A1. The first order linear state space model was obtained as the follows

$$\Delta\dot{x} = \begin{bmatrix} \Delta\dot{x}_1 \\ \Delta\dot{x}_2 \\ \Delta\dot{x}_3 \\ \Delta\dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -6.3106e-17 & 172.6 & -7.215 & 1.2370 \\ -6.3981e-04 & -195.0917 & 6.9485 & -1.6643 \end{bmatrix} \begin{bmatrix} \Delta x_1 \\ \Delta x_2 \\ \Delta x_3 \\ \Delta x_4 \end{bmatrix} +$$

$$\begin{bmatrix} 0 \\ 0 \\ 5.4562e-18 \\ -1.3195 \end{bmatrix} \Delta u + \begin{bmatrix} 0 \\ 0 \\ -5.6320e-18 \\ 0.0304 \end{bmatrix} \Delta u_d \tag{4.5}$$

$$\Delta y = [0 \quad 0 \quad 1 \quad 0]\begin{bmatrix}\Delta x_1\\ \Delta x_2\\ \Delta x_3\\ \Delta x_4\end{bmatrix} + [0]\Delta u + [0]\Delta u_d \tag{4.6}$$

Table 4.2 States information table

States	States type	Average operating point	Desired states
$\Delta x_1(t)$	Perturbed rotor azimuth (rad)	3.1710	0
$\Delta x_2(t)$	Perturbed drive-train displacement (rad)	0.0047	0
$\Delta x_3(t)$	Perturbed rotor speed (rad/s)	1.2635	0
$\Delta x_4(t)$	Perturbed drive-train velocity (rad/s)	$1.1242e-07$	0

Table 4.3 Input information table

Inputs	Input type	Initial value	Desired value
$\Delta u(t)$	Perturbed control input (rotor collective pitch angle) (rad)	0.26	Calculated by controller
$\Delta u_d(t)$	Perturbed disturbance input (hub-height horizontal wind speed) (m/s)	18	Uncontrolled input

4.4.2 Validation of the linear model

To test the accuracy of the linear model in this subsection, 1 and 2 degrees perturbation step collective pitch angle Δu are applied to the linear and non-linear models in the computer simulation. Figure 4.7 and Figure 4.8 show linear and nonlinear open loop wind turbine Simulink models with stair test input respectively.

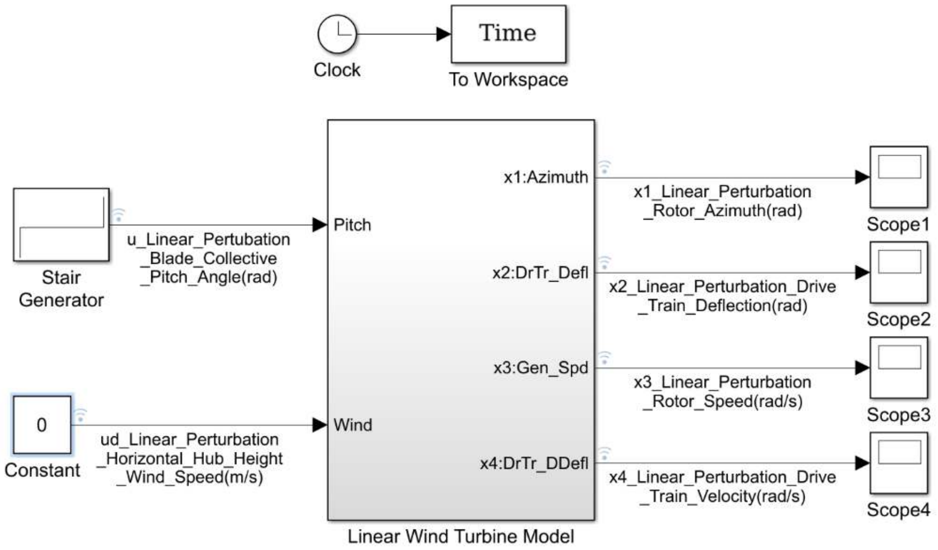

Figure 4.7 Open loop Simulink model of the linear model with stair test input

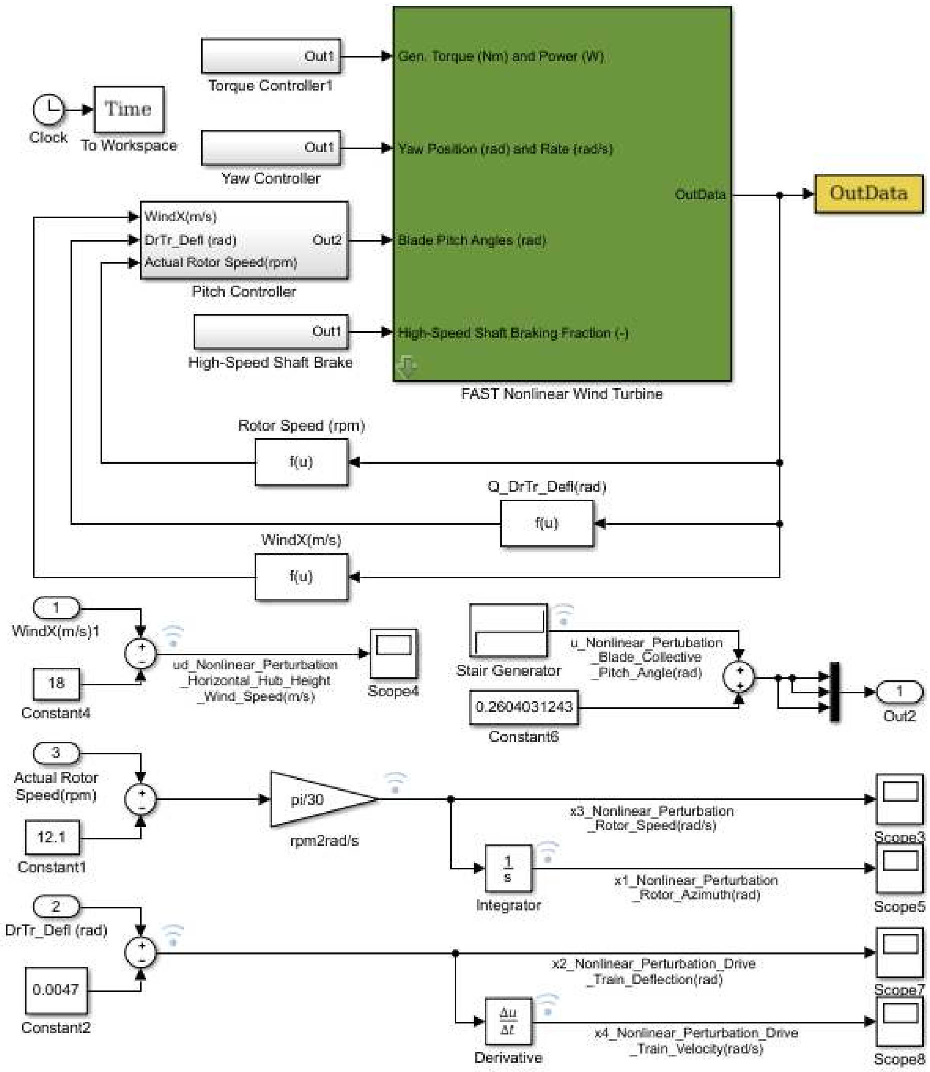

Figure 4.8 Open loop Simulink model of the FAST model with stair test input

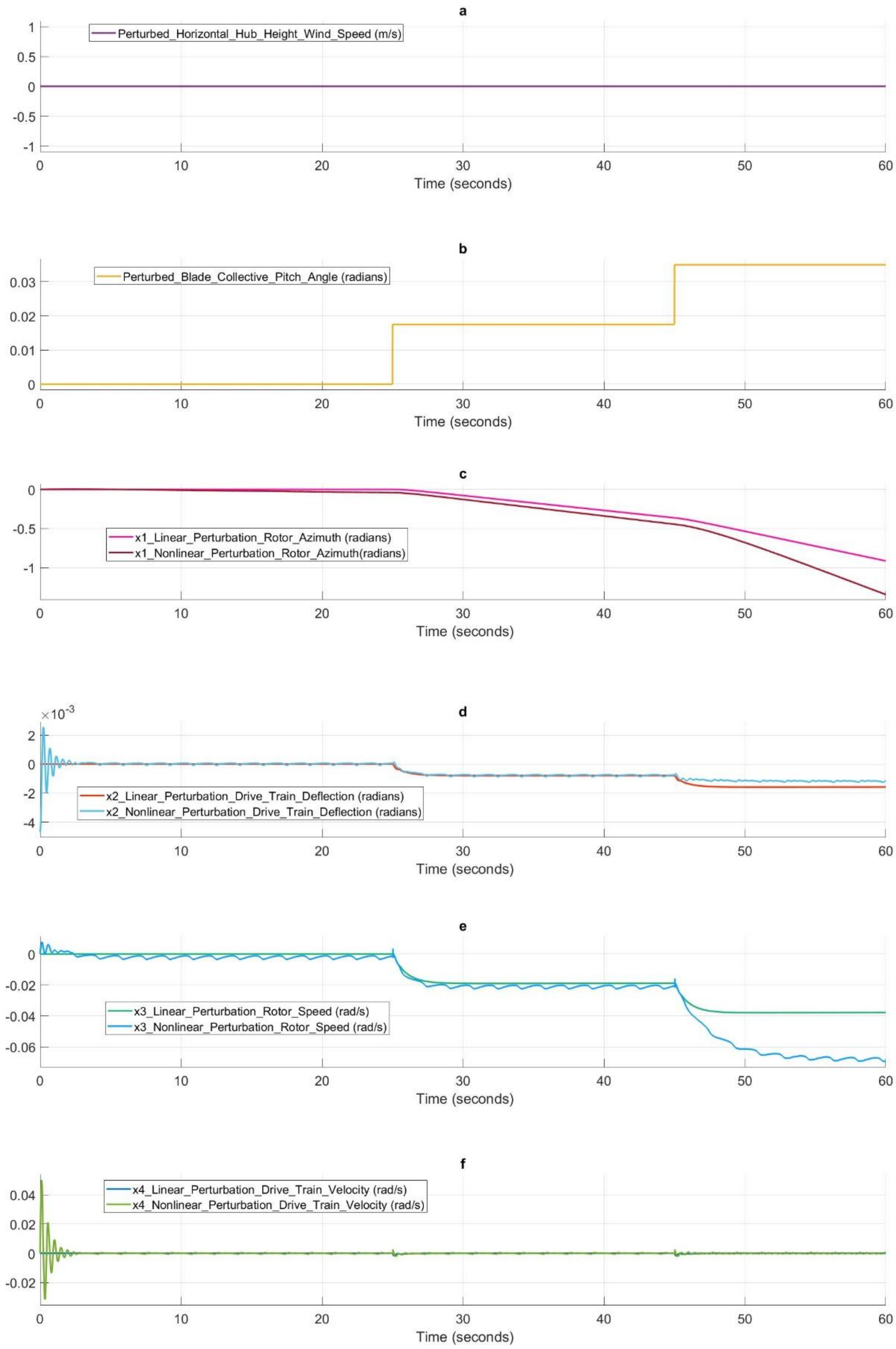

Figure 4.9 Linear model validation

In order to compare linear and nonlinear models, perturbations were found by first subtracting the dynamics of the nonlinear model from the operation point. The units of these dynamics were then changed according to the units of the linear model given in Table 4.1. These operations which are inside of the 'Pitch Controller' subsystem are shown in lower part of the Figure 4.8. Simulation results are given in Figure 4.9.

The above simulation results were made by activating 2 DOFs which are generator and drivetrain. As can be seen in Figure 4.9.a, disturbance input perturbation Δu_d is zero. The reason for this is to examine the responses of linear and nonlinear open loop systems to the control input perturbation Δu. As shown in Figure 4.9.b, 0.0175 radians (approximately 1 degree) of control input was applied in 25 seconds, and then 0.035 radians (approximately 2 degrees) of control input applied in 45 seconds. Figure 4.9.c,d,e,f show the responses of dynamics Δx_1, Δx_2, Δx_3, and Δx_4 to control input Δu for linear and non-linear models. As it is seen in these figures, when 1 degree control input perturbation Δu is applied, the responses of the linear and nonlinear open loop models are approximately the same. But when 2 degree control input perturbation Δu is applied, the responses of the linear and nonlinear models differ. They differ from each other as they move away from the point of operation, which is a natural result of linearization. The linear model is valid at and near the operation point. These figures show us that the nonlinear model around the operation point is linearized correctly. Now a linear controller can be designed according to the linear model in Equation 4.5.

4.4.3 State feedback optimal controller design

In this subsection, LMI based LQR state feedback controller is designed. Consider the linear model given in Equation 4.5. It is in the following form:

$$\Delta\dot{x} = A\Delta x + B\Delta u + B_d\Delta u_d \tag{4.7}$$

State feedback control law is the following:

$$\Delta u = -K\Delta x \tag{4.8}$$

Closed loop system becomes

$$\Delta\dot{x} = (A - BK)\Delta x + B_d\Delta u_d \tag{4.9}$$

The LQR problem is written in Subsection 3.3 as the constraints of SDP as follows:

$$\begin{cases} minimize & \gamma \\ subject\ to & \begin{bmatrix} \gamma & {x_0}^T \\ x_0 & W \end{bmatrix} > 0 \\ \begin{bmatrix} WA^T + AW - BR^{-1}B^T & W \\ W & -Q^{-1} \end{bmatrix} \leq 0, W > 0 \end{cases}$$

Appropriate state and input weighting matrices are selected respectively as follows:

$$Q = \begin{bmatrix} 1000 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, R = [1]$$

SDP problem solved with the following computer software

Table 4.4 Computer software to solve SDP

```
clear all; clc;
yalmip('clear');
load('E:\ARIF\TEZ\Simulations\Optimal
Control\Model_2DOF_Gen_DrTr_18mps\Dogrusallastirma\data_2DOF_Gen_DrT
r_18mps.mat');
A=AvgAMat;
Bu=AvgBMat(:,end);
Q=[1000 0 0 0;0 10 0 0;0 0 1 0;0 0 0 1];
R=[1];
x0=[1;0;0;0];
W=sdpvar(4,4);
Gamma=sdpvar(1,1);
F=[W>=0];
F=[F,[(W*A'+A*W-Bu*inv(R)*Bu') W;W -inv(Q)]<=0];
F=[F,[Gamma x0';x0 W]>=0];
optimize(F,Gamma);
Gamma=value(Gamma)
P=inv(value(W))
Kopt=inv(R)*Bu'*P
Klqr=lqr(A,Bu,Q,R)
Eigenvalues=eig(A-Bu*Kopt)
```

Gamma is 289.0256. It is minimum value of the upper bound of the energy.

$$P = \begin{bmatrix} 289.0256 & 203.3422 & 34.3411 & 23.9659 \\ 203.3422 & 219.4258 & 26.9088 & 21.7260 \\ 32.3411 & 26.9088 & 6.2588 & 5.3256 \\ 23.9659 & 21.7260 & 5.3256 & 4.8762 \end{bmatrix}$$

$$K_{opt} = [-31.6223 \quad -28.6669 \quad -7.0270 \quad -6.4340]$$

Closed system eigenvalues are $-4.0295 \pm 13.4661i, -4.6549 \pm 3.8449i$. We have designed full state feedback controller by accepting that all situations can be measured.

4.4.4 Simulations on linear and FAST nonlinear turbine models

To test the performance of the full state feedback linear controller in this subsection, 1 and 2 m/s perturbation hub-height wind speeds Δu_d are applied to the linear and nonlinear models in the computer simulation. Figure 4.10 and Figure 4.11 show closed loop linear and nonlinear wind turbine Simulink models with stair disturbance test input respectively.

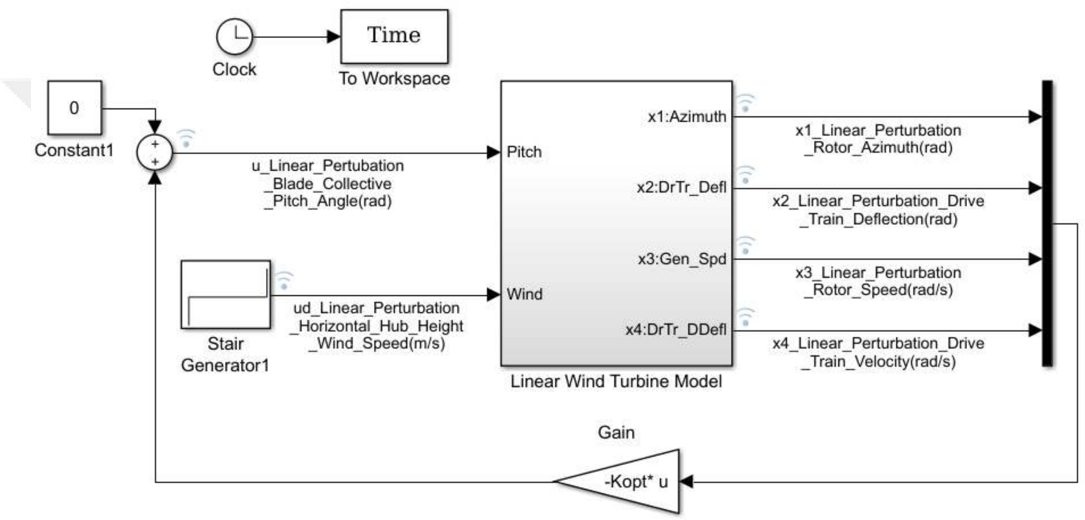

Figure 4.10 Linear model with stair test input and full state feedback controller

The upper part of the Figure 4.11 shows the state feedback collective pitch controller connected with the FAST model. The signals entering the controller are the rotor speed and the drive train deflection. Lower part of the figure shows detail of the controller which is inside of the 'Pitch Controller' subsystem. Simulation results aregiven in Figure 4.12.

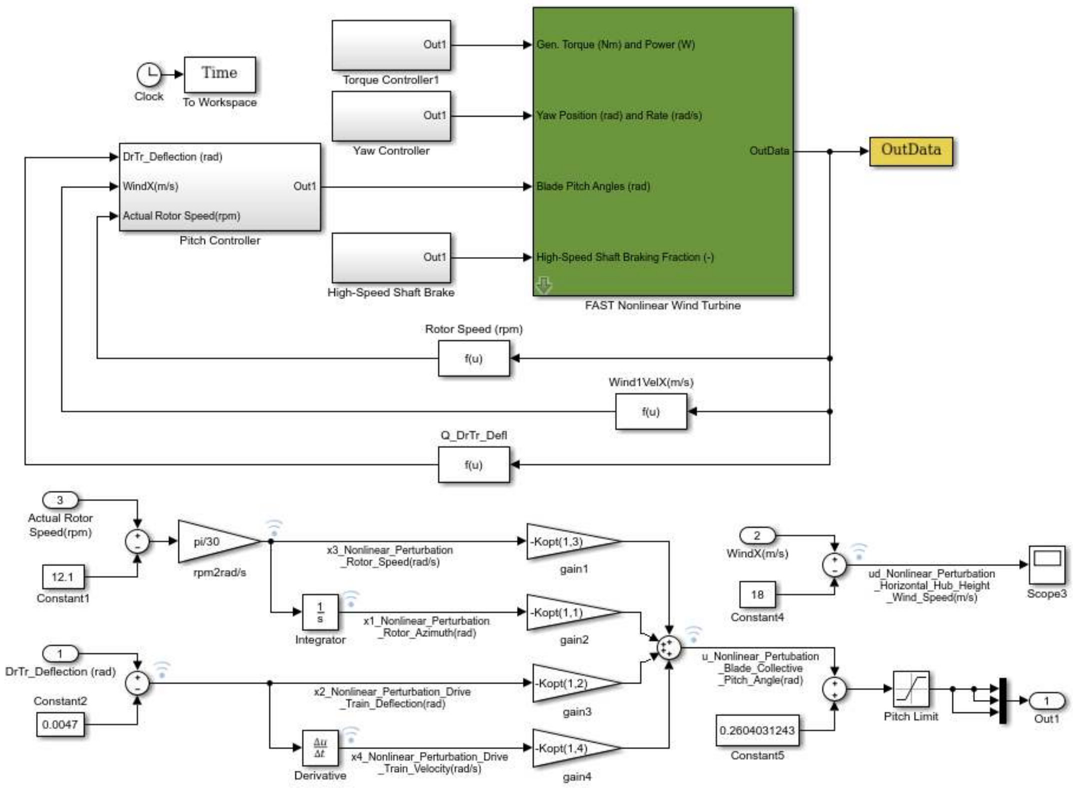

Figure 4.11 FAST model with stair test input and full state feedback controller

The simulation results were obtained by activating 2 DOFs which are generator and drivetrain. As shown in Figure 4.12.a 1 m/s perturbed disturbance input Δu_d was applied in 25 seconds, and then 2 m/s perturbed disturbance input Δu_d applied in 45 seconds. Figure 4.12.b, c, d, e, f show the control input Δu and the responses of dynamics Δx_1, Δx_2, Δx_3, and Δx_4 to disturbance input Δu_d for linear and non-linear models. As it is seen in these figures, the responses of the linear and nonlinear closed loop models are approximately the same. Now this controller is simulated with a stochastic turbulence full field wind model profile developed by NREL TurbSim wind simulator [26] on FAST nonlinear model. This wind profile covers operation points in region 3 and its average is 18 m/s. See Appendix A, page A3 for wind input file. Unstructured model uncertainties added to simulation. 6 DOFs have been activated in simulation even though the controller design is based on 2 DOF. The simulation results are shown in Figure 4.13. Wind speed, blade pitch angle, drivetrain deflection, generator speed, torque, and power are shown in these figures respectively.

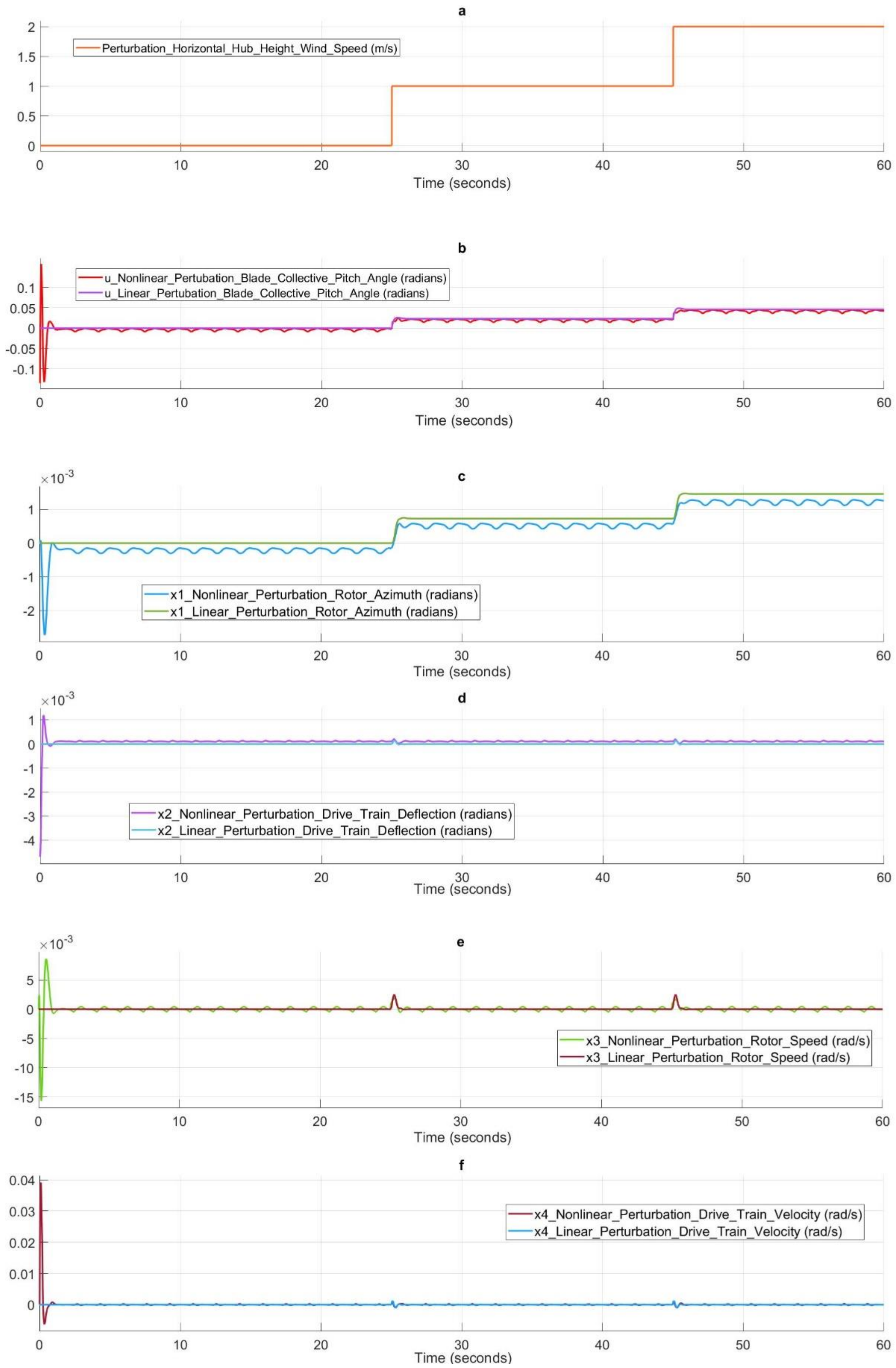

Figure 4.12 Response to 1 m/s and 2 m/s stair disturbance input

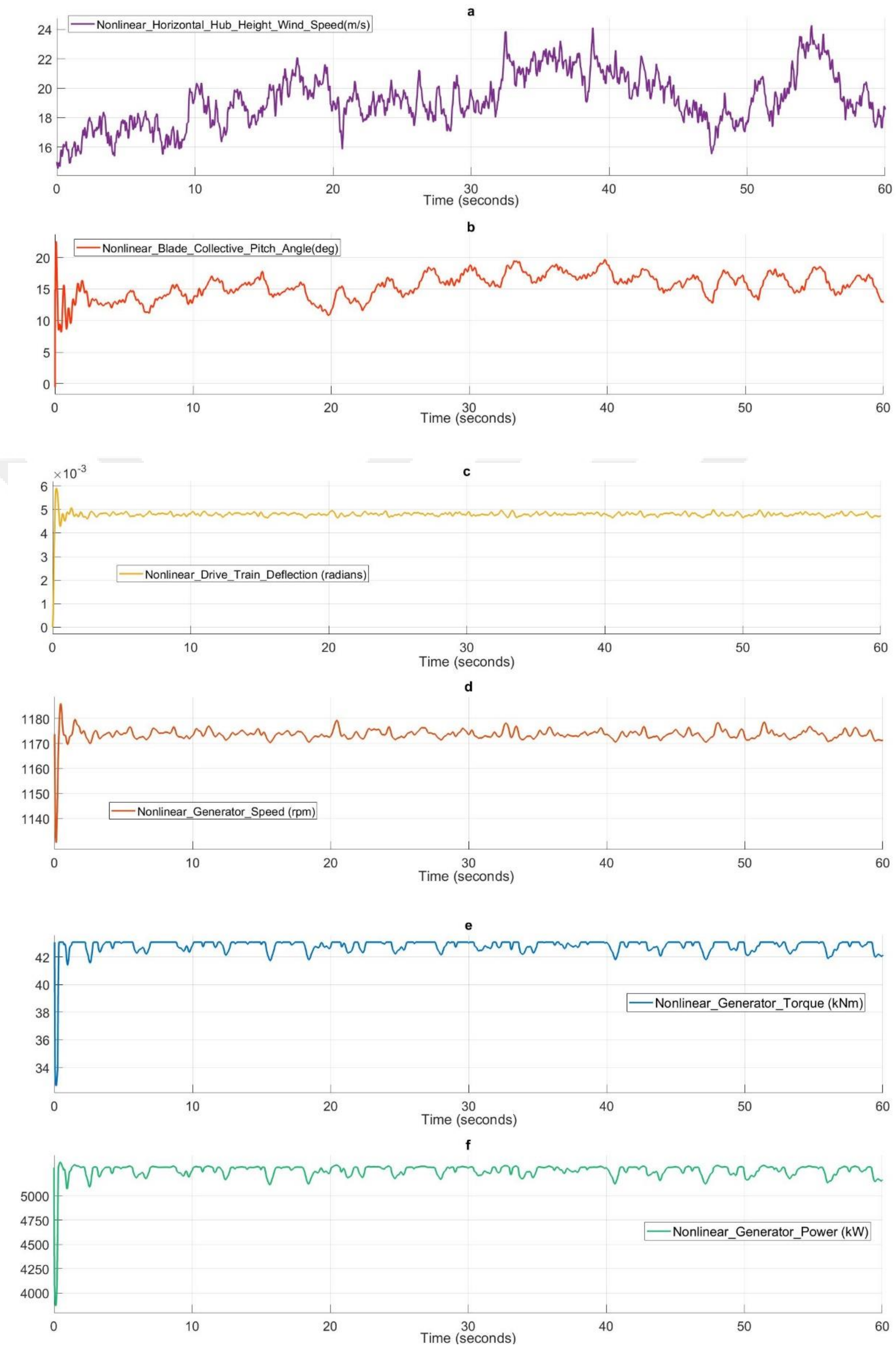

Figure 4.13 Response to 18 m/s mean turbulence disturbance input

In this section, a baseline gain scheduled PI (GSPI) rotor collective pitch control is designed for NREL 5 MW onshore wind turbine in region 3. Performance of this controller is simulated on FAST nonlinear wind turbine Simulink model.

4.5.1 Controller design

One of the purposes of blade pitch control is to regulate rotor speed at rated speed in region 3. Rotor speed is regulated to maintain rated generator torque. PI is SISO control system so a single DOF which is generator DOF is used. For generator DOF, a simple linear model has the following form:

$$\Delta\dot{\Omega} = A\Delta\Omega + B\Delta\beta + B_d\Delta w \tag{4.10}$$

where Ω, β and w are rotor speed, blade pitch angle and hub-height wind speed disturbance respectively. The variables in Equation 4.10 are perturbed variables which are small deviations from their equilibrium point. The expression of the PID control is as follows:

$$u(t) = K_p e(t) + K_I \int e(t)dt + K_d \dot{e}(t) \tag{4.11}$$

where $u(t)$ is control input, $e(t)$ is error. K_p, K_I and K_D are proportional, integral, derivative gains respectively. The aim in this control design is to find the appropriate values of these gains for a stable and desired performance closed loop system. $u(t)$ is perturbed pitch angle $\Delta\beta(t)$ and $e(t)$ is perturbed rotor speed $\Delta\Omega$. Equation 4.11 is rewritten in terms of $\Delta\beta$, $\Delta\Omega$.

$$\Delta\beta(t) = K_p\Delta\Omega(t) + K_i \int \Delta\Omega(t)dt + K_d\Delta\dot{\Omega}(t) \tag{4.12}$$

In order to design the control system, firstly closed loop system must be written in s-domain. Taking Laplace transform of both sides of Equation 4.12,

$$\Delta\beta(s) = K_p\Delta\Omega(s) + K_i\frac{1}{s}\Delta\Omega(s) + K_d s\Delta\Omega(s) \tag{4.13}$$

Where $\Delta\Omega(s)$ and $\Delta\beta(s)$ are Laplace transforms of $\Omega(t)$ and $\Delta\beta(t)$ respectively. Taking Laplace transform of both sides of Equation 4.10,

$$s\Delta\Omega(s) = A\Delta\Omega(s) + B\Delta\beta(s) + B_d\Delta w(s) \tag{4.14}$$

Where $\Delta w(s)$ is the Laplace transform of $\Delta w(t)$. Take the term $A\Delta\Omega(s)$ to the left-hand side of Equation 4.14:

$$\Delta\Omega(s)(s-A) = B\Delta\beta(s) + B_d\Delta w(s) \tag{4.15}$$

Substitute Equation 4.13 into Equation 4.15.

$$\Delta\Omega(s)(s-A) = B(K_p\Delta\Omega(s) + K_i\frac{1}{s}\Delta\Omega(s) + K_d s\Delta\Omega(s)) + B_d\Delta w(s) \tag{4.16}$$

In order to examine the stability and performance of the closed loop system, transfer function $T_c(s)$ should be determined between $\Delta\Omega(s)$ and $\Delta w(s)$:

$$T_c(s) = \frac{\Delta\Omega(s)}{\Delta w(s)} = \frac{B_d s}{(1-BK_d)s^2 + (-A-BK_p)s + (-BK_i)} \tag{4.17}$$

Characteristic equation of closed loop system is the denominator of closed loop transfer function $T_c(s)$. Characteristic equation $P(s)$ is

$$P(s) = (1-BK_d)s^2 + (-A-BK_p)s + (-BK_i) \tag{4.18}$$

In order for the closed loop system to be stable, the roots of the characteristic equation in Equation 4.18 must be located on the left-hand side of the complex plane. In this characteristic equation, the coefficients of s must be positive.

$$1-BK_d > 0,\ -A-BK_p > 0,\ -BK_i > 0 \tag{4.19}$$

It is needed to determine operation point in Region 3 to find turbine parameters A, B and B_d. Wind speed, rotor speed and blade pitch angle are chosen at the operating point as follows $w_0 = 18$ m/s, $\Omega_0 = 12.1$ rpm and $\beta_0 = 14.92$ degrees respectively. These parameters at the operation point can be determined by the running of the FAST linearization code. Linearization results are $A = -0.3099$, $B = -1.7246$ and $B_d = 0.034$. Characteristic equation becomes

$$P(s) = (1+1.7246K_d)s^2 + (0.3099+1.7246K_p)s + (1.7246K_i) \tag{4.20}$$

The coefficients of all terms in the characteristic equation must be positive for a stable system.

$$K_d > -0.58,\ K_p > -0.18,\ K_i > 0 \tag{4.21}$$

The gains selected according to the above conditions make the stable closed loop system. However, these conditions do not give information about performance. The question is how we select the gains that meet the above requirements while also delivering the desired performance. The characteristic equation in Equation 4.18 expressed in the form below helps us.

$$s^2 + 2\delta\omega s + \omega^2 = 0 \tag{4.22}$$

Where δ is damping ratio and ω is natural frequency.

$$2\delta\omega = \frac{-A-BK_p}{1-BK_d}, \text{ and } \omega^2 = \frac{-BK_i}{1-BK_d}. \tag{4.23}$$

We get K_p and K_i like this

$$K_i = \frac{-\omega^2(1-BK_d)}{B} \text{ and } K_p = -\frac{A}{B} - \frac{2\delta\omega(1-BK_d)}{B} \tag{4.24}$$

We choose values for δ=0.806 and ω=2.853. By keeping one of these gains fixed, we can find the other two gains. Let us choose $K_d = 0$. Using Equation 4.24 we can calculate the other two gains as follows:

$K_i = 4.72$ and $K_p = 2.49$

4.5.2 Gain scheduling

Linear controller was designed at a single operating point. The performance of this controller is good around the operation point, which may deteriorate as it moves away from the operating point. Resulting performance is not the desired performance. This is because pitch control input gain B changes with blade pitch angle and wind speed. It depends directly on the partial derivative of the aerodynamic torque according to the blade pitch angle. Variation of aerodynamic torque with various wind speeds and blade pitch angle is shown in Figure 4.5. The slope of this curve gives the control input gain. The slope of the curve decreases as the pitch angle decreases, so control input gain is reduced. Linear controller is designed with 14.92 degrees pitch angle. The controller's performance is reduced at low pitch angles. The solution to this problem is achieved by scheduling the gains of the controller with blade pitch angle. Each PI gain is multiplied by a function GK in the form:

$$GK(\beta) = \frac{1}{1+\frac{\beta}{\beta_k}} \tag{4.25}$$

Where GK is dimensionless gain correction factor and depends on blade pitch angle. Value of β_k calculated at [23] is as 6.302336 degrees. GSPI gains are written as follows.

$K_i(\beta) = K_i GK(\beta) = 4.72GK(\beta)$ and $K_p(\beta) = K_p GK(\beta) = 2.49GK(\beta)$

4.5.3 Simulations on FAST nonlinear turbine model

Closed loop system is simulated to verify the performance of the controller. Figure 4.14 shows Simulink model for this controller. The upper part of the figure shows the baseline pitch controller connected with the FAST nonlinear turbine model. The signal supplied into the 'Pitch Controller' subsystem is the rotor speed. To find the perturbed rotor speed rotor speed should be subtracted at the point of operation from the actual rotor speed. Then it should be converted from rpm to rad/s. The lower part of the figure shows the details of the controller.

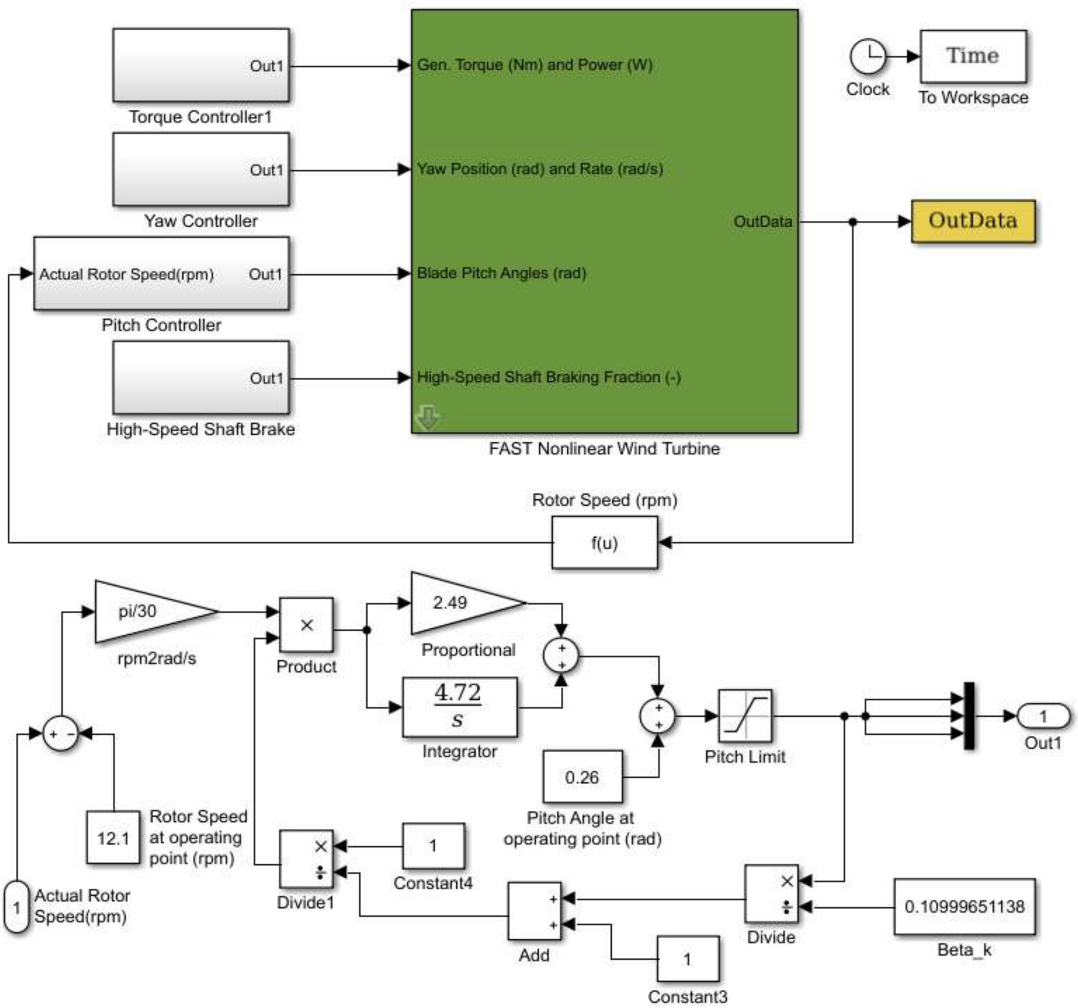

Figure 4.14 Simulink model of the baseline pitch controller

Perturbed rotor speed was multiplied separately by the proportional and integral parts of the GSPI controller. Then perturbed blade pitch angle in radian is found, collecting the products. First, the blade pitch angle at the operating point was converted to radians (14.92 degrees = 0.26 radian). Then it was added with perturbed blade pitch angle. A lower limit of 0 radian and an upper limit of 1.57 radian (90 degrees) are applied to the pitch angle. This controller is simulated with a stochastic turbulence full field 18 m/s mean wind model. The wind profile covers operation points in region 3. Wind input file is shown in Appendix A, page A3. Unstructured model uncertainties added to simulation. 6 DOFs have been activated in simulation even though the controller design is based on 1 DOF. The simulation results are shown in Figure 4.15. Wind speed, pitch angle, drivetrain deflection, generator speed, torque, and power were shown in these figures respectively.

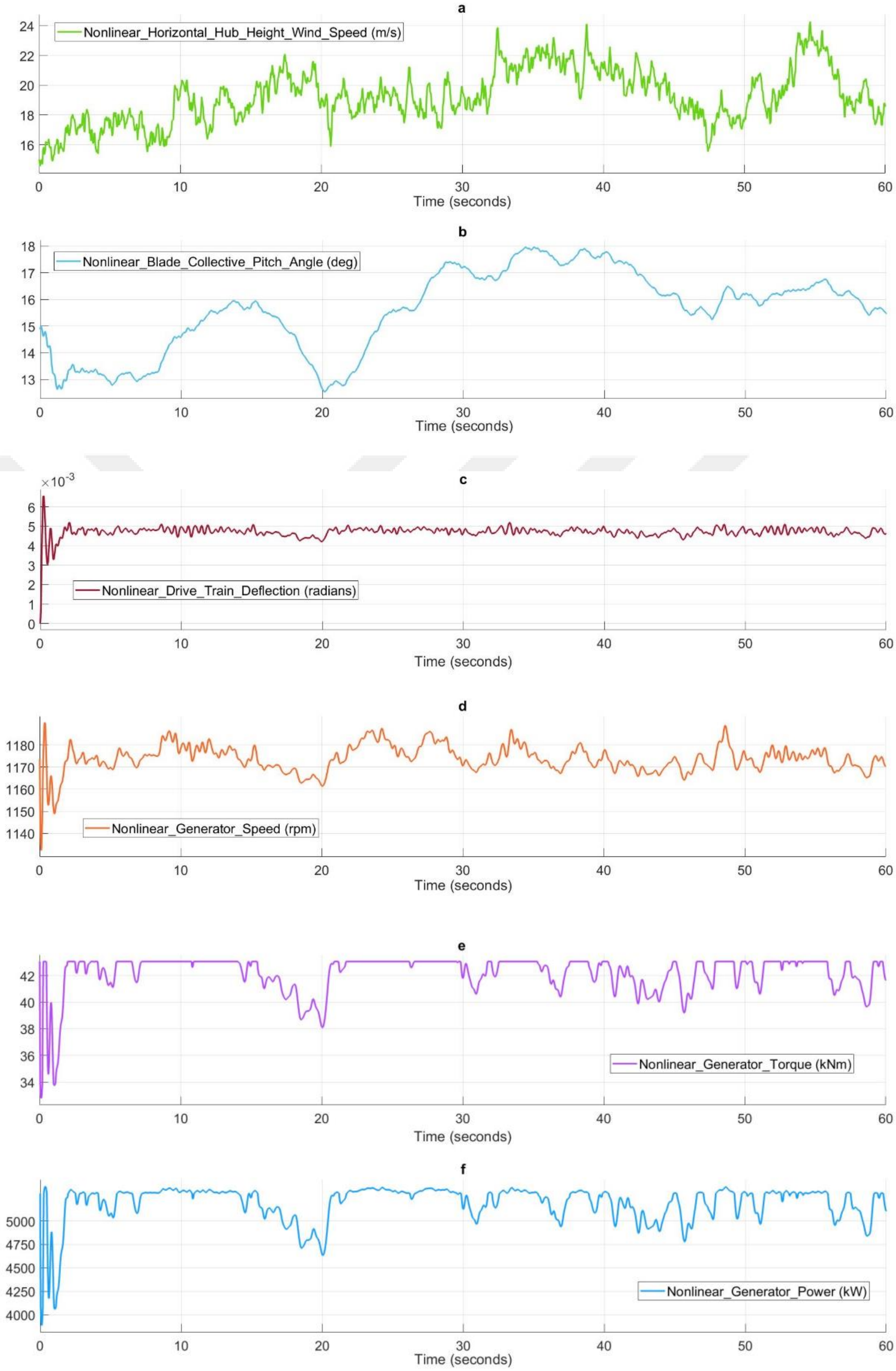

Figure 4.15 Response to 18 m/s mean turbulence disturbance input

4.6 Comparison of the Result

4.6.1 Performance index

LMI based LQR controller and baseline GSPI controller are compared by using following performance indexes, which are ISE and the mean of the square of the error (MSE),

$$ISE = T\sum_{i=1}^{n} {e_i}^2 \tag{4.26}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n} {e_i}^2. \tag{4.27}$$

where T is sample time and n is number of sample.

4.6.2 Comparison of the results

Unstructured model uncertainties added to simulation. 6 DOFs have been activated. Simulations were performed on FAST nonlinear wind turbine model for both controllers. Wind speed, pitch angle, drivetrain deflection, generator speed, torque, and power were shown in Figure 4.16 respectively.

Table 4.5 Comparison table of performance indexes for controllers

States	MSE (GSPI)	MSE (LMI)	ISE (GSPI)	ISE (LMI)
Collective Pitch Angle	2.6827	4.0046	160.9609	240.2756
Drive Train Deflection	2.7436e-05	2.3791e-05	0.0016	0.0014
Generator Speed	35.3777	7.0528	$2.1227e+03$	423.1704
Generator Torque	2.7506	0.5146	165.0333	30.8738
Generator Power	49938	9644.2	$2.9963e+06$	$5.7865e+05$

As shown in Figure 4.16.a, a horizontal turbulence mean 18 m/s wind speed is applied to both control systems. The results presented in Figure 4.16.b, c, d, e, f and Table 4.4 reveal that the accuracy of the LMI based LQR controller is better than that of GSPI, and it can respond to disturbance variations more quickly.

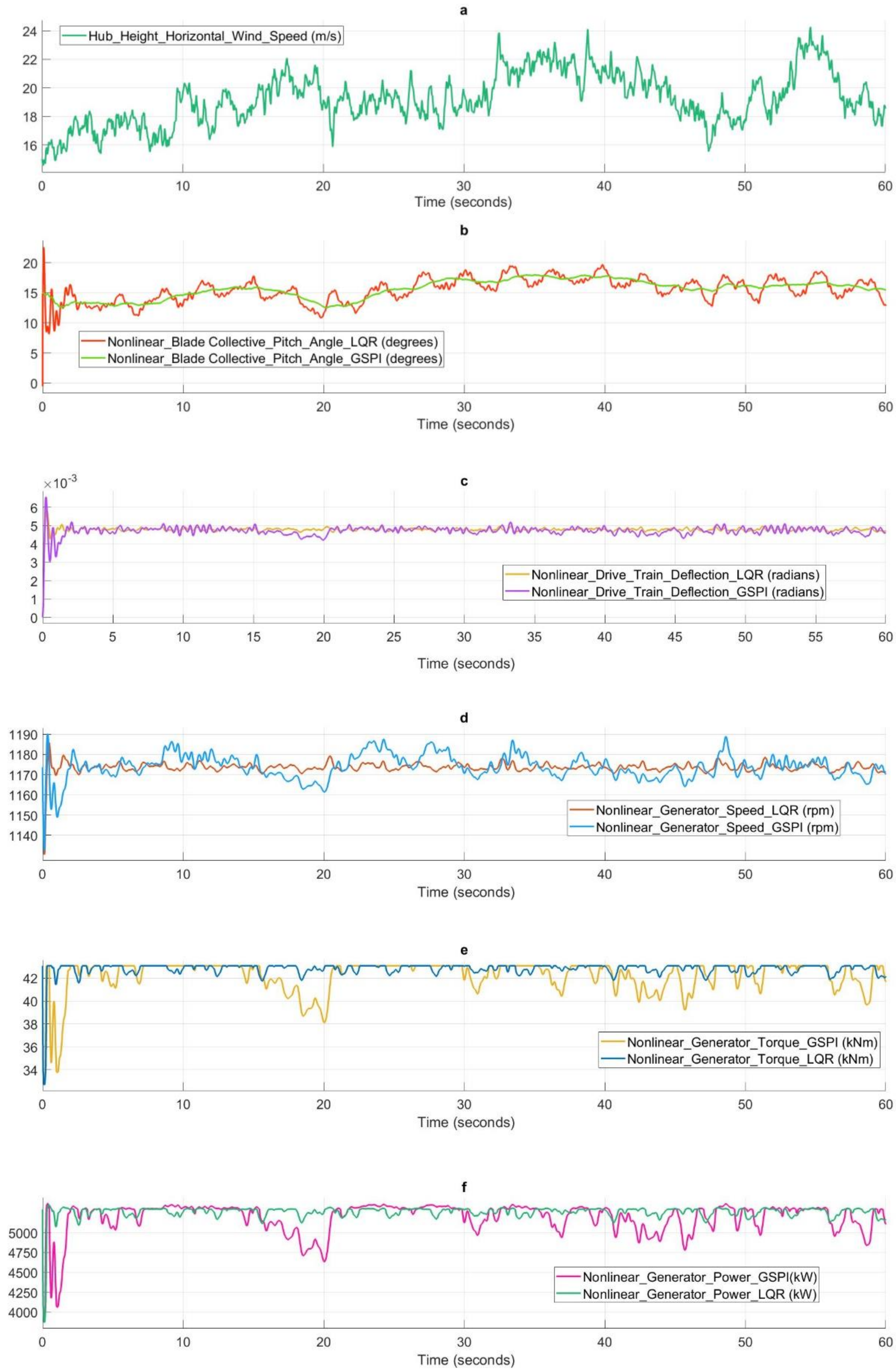

Figure 4.16 Comparison of LMI based LQR and GSPI controllers

CHAPTER FIVE

CONCLUSION AND FUTURE STUDIES

5.1 Conclusion

In this thesis, the answers to the following questions are investigated.

1. What is LMI structure and LMI problemFAST?
2. What is the importance of LMI for control engineering?
3. What is the history of LMI?
4. What is the importance of Lyapunov stability for LMIs?
5. How can a control engineering problem be expressed as an optimization problem involving LMI constraints?
6. How can an optimization problem be solved by using computer tools?

In order to increase the effectiveness of the study and to answer the questions above, the problem of wind turbine blade pitch control is considered as the optimal control problem. The performances of the controllers designed using the LMI on the linear model are tested, which has been proven to be accurate, under the stochastic disturbance input on the nonlinear model, and the results are compared with the baseline GSPI controller shown by figures and performance index table. The simulation results show that the LMI based LQR controller has much better performance in the generator speed and generator power regulation in comparison to the baseline GSPI controller despite the fact that LMI based LQR controller requires more control effort. The results reveal that LMI based LQR controller is a good alternative in optimal control of wind turbines with its good accuracy and quick response to disturbances.

5.2 Future studies

Almost every system in nature has a nonlinear characteristic. Stability analysis and controller design for nonlinear systems are difficult subjects. Lyapunov's second method is the only method available to perform this analysis. Therefore it is a good research area to study Lyapunov's second method and Lyapunov functions. LMIs are important tools for transition from analytical world to numerical world for control engineering. In order to better understand the LMIs, we need to look deeper into the historical development, the interior point method and the geometric interpretation. This is also a good research topic to design more complex control algorithms such as adaptive control, sliding mode control and robust control of control engineering using LMIs. In this study, we assume that all dynamics can be measured for feedback, which is not practical. Some of the dynamics should be observed and observer design is an important part of more complex control algorithms. Therefore observer design is also a part of the future work of this study.

REFERENCES

[1] Jan C. Willems. (1971). Least Squares Stationary Optimal Control and the Algebraic Riccati Equation. *IEEE Transactions On Automatic Control*. **AC-16**, 621-634.

[2] Pavel Pakshin, Julia Emelianova. (2016). An Experience of Using LMI Technique in Control Education. *IFAC-PapersOnLine*. **49-6**, 115-120.

[3] Pavel V. Pakshin, Julia P. Emelianova, Alexander Yu. Mazurov. (2012). An Experience of Using LMI Technique in Control Student Projects. *Proceeding of the 9th IFAC Symposium Advances in Control*. 472-477.

[4] Maria Hypiusova, Alena Kozakova. (2016). Control Education for Mechatronics: Robust State Feedback Controller Design. *IFAC-PapersOnLine*. **49-6**, 320-325.

[5] Majid Fard, Mohammad Aldeen. (2016). Robust control design for a boost converter in a photovoltaic system. *2016 IEEE 7th International Symposium on Power Electronics for Distributed Generation Systems (PEDG)*.

[6] Carlos Olalla, Ramon Levya, Abdelali El Aroudi, Isabelle Queinnec. (2009). Robust LQR Control for PWM Converters: An LMI Approach. *IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS*. **56-7**, 2548-2558.

[7] Dragoslav D. Siljac, Dusan M. Stipanovic and Aleksander I. Zecevic. (2002). Robust Decentralized Turbine/Governor Control Using Linear Matrix Inequalities. *IEEE TRANSACTIONS ON POWER SYSTEMS*. **17-3**, 715-722.

[8] Mert Sever, Gulay Goktas, Hakan Yazivi, Ibrahim Beklan Kucukdemiral. (2015). Disturbance Attenuation of Asymmetric Structure by LMI Based Optimal State Feedback Controller with Saturated Actuators. *2015 6th International Conference on Modeling, Simulation, and Applied Optimization (ICMSAO)*.

[9] H.M. Hassan, A.L. ElShafei, W.A. Farag, M.S. Saad. (2012). A robust LMI-based pitch controller for large wind turbines. *Renewable Energy*. **44**, 63-71.

[10] Hee-Sang Ko, Juri Jatskevich, Guy Dumont, Gi-Gap Yoon. (2008). An Advanced LMI-based-LQR design for voltage control of grid-connected wind farm. *Electric Power Systems Research*. **78**, 539-546.

[11] Stephen Boyd, Laurent El Ghaoui, Eric Feron, Venkataramanan Balakrishnan. (1994). Linear Matrix Inequalities in System and Control Theory. Volume 15: 1st edition. Philadelphia, Pennsylvania: Society for Industrial and Applied Mathematics.

[12] Jeremy G. VanAntwerp, Richard D. Braatz*. (2000). A Tutorial on Linear and Bilinear Matrix Inequalities. *Journal of Process Control.* **10,** 363-385

[13] Stephan Boyd, Lieven Vandenberghe. (2004). Convex Optimization. Seventh Printing with Corrections 2009. Cambridge: University Press.

[14] Aleksandr M. Lyapunov. (1992). The General Problem of the Stability of Motion. *International Journal of Control*. **55(3)**. 531-773.

[15] Ogata K. (1997). Modern Control Engineering. 3rd edition. New Jersey: Tom Robbins.

[16] Pascal Gahinet, Arkadi Nemirovski, Alan J. Laub, Mahmoud Chilali. (1995). LMI Control Toolbox User’s Guide. Version 1.

[17] Johan Löfberg. (2004). YALMIP: A Toolbox for Modeling and Optimization in MATLAB. *IEEE International Symposium on Computer Aided Control Systems*. 284-289.

[18] Jos F. Sturm. (1998). Using SeDuMi 1.02, a MATLAB Toolbox for Optimization Over Symmetric Cones. *Optimization Method and Software.* **11(2)**, 625-623

[19] William L. Brogan. (1991). Modern Control Theory. Third edition. New Jersey: Prentice-Hall, Inc.

[20] Richard C. Dorf, Robert H. Bishop. (2011). Modern Control systems. Twelfth edition. New Jersey: Prentice-Hall, Inc.

[21] Martin O. L. Hansen. (2008). Aerodynamics of Wind Turbines. Second edition. Padstow: Earthscan.

[22] M. Singh, E. Muljadi, J. Jonkman, V. Gevorgian, I. Girsang and J. Dhupia. (2014). Simulation for Wind Turbine Generators-with FAST and Matlab-Simulink Modules. National Wind Technology Center.

[23] J. Jonkman, S. Butterfield. W. Musial and G. Scott. (2009). Definition of a 5-MW Wind Turbine for Offshore System Development. Colorado: National Renewable Energy Laboratory (NREL).

[24] Bonnie Jonkman, Jason Jonkman. (2016). FAST v8.16.00a-bjj. Colorado: National Renewable Energy Laboratory (NREL).

[25] Gunjit S. Bir (2008). User's Guide to MBC3 (Multi-blade Coordinate Transformation Utility for 3-Bladed Wind Turbines), *National Renewable Energy Laboratory (NREL)*. **3**.

[26] Andy Platt, Bonnie Jonkman, and Jason Jonkman. (2016). InflowWind User's Guide. National Wind Technology Center.

APPENDIX

Appendix A: FAST Files

A1: FAST input file

```
------- FAST v8.16.* INPUT FILE ------------------------------------------------
FAST Certification Test #18: NREL 5.0 MW Baseline Wind Turbine (Onshore)
---------------------- SIMULATION CONTROL --------------------------------------
True         Echo            - Echo input data to <RootName>.ech (flag)
"FATAL"       AbortLevel      - Error level when simulation should abort (string) {"WARNING", "SEVERE", "FATAL"}
         60   TMax            - Total run time (s)
    0.00625   DT              - Recommended module time step (s)
          1   InterpOrder     - Interpolation order for input/output time history (-) {1=linear, 2=quadratic}
          1   NumCrctn        - Number of correction iterations (-) {0=explicit calculation, i.e., no corrections}
      99999   DT_UJac         - Time between calls to get Jacobians (s)
      1E+06   UJacSclFact     - Scaling factor used in Jacobians (-)
---------------------- FEATURE SWITCHES AND FLAGS ------------------------------
          1   CompElast       - Compute structural dynamics (switch) {1=ElastoDyn; 2=ElastoDyn + BeamDyn for blades}
          1   CompInflow      - Compute inflow wind velocities (switch) {0=still air; 1=InflowWind; 2=external from OpenFOAM}
          2   CompAero        - Compute aerodynamic loads (switch) {0=None; 1=AeroDyn v14; 2=AeroDyn v15}
          1   CompServo       - Compute control and electrical-drive dynamics (switch) {0=None; 1=ServoDyn}
          0   CompHydro       - Compute hydrodynamic loads (switch) {0=None; 1=HydroDyn}
          0   CompSub         - Compute sub-structural dynamics (switch) {0=None; 1=SubDyn}
          0   CompMooring     - Compute mooring system (switch) {0=None; 1=MAP++; 2=FEAMooring; 3=MoorDyn; 4=OrcaFlex}
          0   CompIce         - Compute ice loads (switch) {0=None; 1=IceFloe; 2=IceDyn}
---------------------- INPUT FILES ---------------------------------------------
"E:\ARIF\TEZ\Fastv8\CertTest\5MW_Baseline\NRELOffshrBsline5MW_Onshore_ElastoDyn.dat"    EDFile          - Name of file
containing ElastoDyn input parameters (quoted string)
"E:\ARIF\TEZ\Fastv8\CertTest\5MW_Baseline\NRELOffshrBsline5MW_BeamDyn.dat"    BDBldFile(1)    - Name of file
containing BeamDyn input parameters for blade 1 (quoted string)
"E:\ARIF\TEZ\Fastv8\CertTest\5MW_Baseline\NRELOffshrBsline5MW_BeamDyn.dat"    BDBldFile(2)    - Name of file
containing BeamDyn input parameters for blade 2 (quoted string)
"E:\ARIF\TEZ\Fastv8\CertTest\5MW_Baseline\NRELOffshrBsline5MW_BeamDyn.dat"    BDBldFile(3)    - Name of file
containing BeamDyn input parameters for blade 3 (quoted string)
"E:\ARIF\TEZ\Fastv8\CertTest\5MW_Baseline\NRELOffshrBsline5MW_InflowWind_12mps.dat"    InflowFile      - Name of file
containing inflow wind input parameters (quoted string)
"E:\ARIF\TEZ\Fastv8\CertTest\5MW_Baseline\NRELOffshrBsline5MW_Onshore_AeroDyn15.dat"    AeroFile        - Name of file
containing aerodynamic input parameters (quoted string)
"E:\ARIF\TEZ\Fastv8\CertTest\5MW_Baseline\NRELOffshrBsline5MW_Onshore_ServoDyn.dat"    ServoFile       - Name of file
containing control and electrical-drive input parameters (quoted string)
"unused"      HydroFile       - Name of file containing hydrodynamic input parameters (quoted string)
"unused"      SubFile         - Name of file containing sub-structural input parameters (quoted string)
"unused"      MooringFile     - Name of file containing mooring system input parameters (quoted string)
"unused"      IceFile         - Name of file containing ice input parameters (quoted string)
---------------------- OUTPUT --------------------------------------------------
True          SumPrint        - Print summary data to "<RootName>.sum" (flag)
          5   SttsTime        - Amount of time between screen status messages (s)
      99999   ChkptTime       - Amount of time between creating checkpoint files for potential restart (s)
"default"     DT_Out          - Time step for tabular output (s) (or "default")
          0   TStart          - Time to begin tabular output (s)
          2   OutFileFmt      - Format for tabular (time-marching) output file (switch) {1: text file [<RootName>.out], 2: binary file
[<RootName>.outb], 3: both}
True          TabDelim        - Use tab delimiters in text tabular output file? (flag) {uses spaces if false}
"ES10.3E2"    OutFmt          - Format used for text tabular output, excluding the time channel.  Resulting field should be 10
characters. (quoted string)
---------------------- LINEARIZATION -------------------------------------------
False         Linearize       - Linearization analysis (flag)
         36   NLinTimes       - Number of times to linearize (-) [>=1] [unused if Linearize=False]
        60.14,60.28,60.42,60.56,60.70, 60.84, 60.98,
61.12,61.26,61.40,61.54,61.68,61.82,61.96,62.10,62.24,62.38,62.52,62.66,62.80,62.94,63.08,63.22,63.36,63.50,63.64,63.78,6
```

```
3.92,64.06,64.20,64.34,64.48,64.62,64.76,64.90,65.04   LinTimes       - List of times at which to linearize (s) [1 to NLinTimes]
[unused if Linearize=False]
        1   LinInputs       - Inputs included in linearization (switch) {0=none; 1=standard; 2=all module inputs (debug)} [unused if
Linearize=False]
        1   LinOutputs      - Outputs included in linearization (switch) {0=none; 1=from OutList(s); 2=all module outputs (debug)}
[unused if Linearize=False]
False       LinOutJac       - Include full Jacobians in linearization output (for debug) (flag) [unused if Linearize=False; used only if
LinInputs=LinOutputs=2]
False       LinOutMod       - Write module-level linearization output files in addition to output for full system? (flag) [unused if
Linearize=False]
--------------------- VISUALIZATION ------------------------------------------
        0   WrVTK           - VTK visualization data output: (switch) {0=none; 1=initialization data only; 2=animation}
        1   VTK_type        - Type of VTK visualization data: (switch) {1=surfaces; 2=basic meshes (lines/points); 3=all meshes
(debug)} [unused if WrVTK=0]
true        VTK_fields      - Write mesh fields to VTK data files? (flag) {true/false} [unused if WrVTK=0]
       15   VTK_fps         - Frame rate for VTK output (frames per second){will use closest integer multiple of DT} [used only if
WrVTK=2]
```

A2: FAST input ElastoDyn file

```
------- ELASTODYN v1.03.* INPUT FILE -------------------------------------------
NREL 5.0 MW Baseline Wind Turbine for Use in Offshore Analysis. Properties from Dutch Offshore Wind Energy Converter
(DOWEC) 6MW Pre-Design (10046_009.pdf) and REpower 5M 5MW (5m_uk.pdf)
---------------------- SIMULATION CONTROL --------------------------------------
False         Echo        - Echo input data to "<RootName>.ech" (flag)
        3   Method      - Integration method: {1: RK4, 2: AB4, or 3: ABM4} (-)
"DEFAULT"     DT          - Integration time step (s)
---------------------- ENVIRONMENTAL CONDITION ---------------------------------
    9.80665   Gravity     - Gravitational acceleration (m/s^2)
---------------------- DEGREES OF FREEDOM --------------------------------------
False         FlapDOF1    - First flapwise blade mode DOF (flag)
False         FlapDOF2    - Second flapwise blade mode DOF (flag)
False         EdgeDOF     - First edgewise blade mode DOF (flag)
False         TeetDOF     - Rotor-teeter DOF (flag) [unused for 3 blades]
True          DrTrDOF     - Drivetrain rotational-flexibility DOF (flag)
True          GenDOF      - Generator DOF (flag)
False         YawDOF      - Yaw DOF (flag)
True          TwFADOF1    - First fore-aft tower bending-mode DOF (flag)
True          TwFADOF2    - Second fore-aft tower bending-mode DOF (flag)
True          TwSSDOF1    - First side-to-side tower bending-mode DOF (flag)
True          TwSSDOF2    - Second side-to-side tower bending-mode DOF (flag)
False         PtfmSgDOF   - Platform horizontal surge translation DOF (flag)
False         PtfmSwDOF   - Platform horizontal sway translation DOF (flag)
False         PtfmHvDOF   - Platform vertical heave translation DOF (flag)
False         PtfmRDOF    - Platform roll tilt rotation DOF (flag)
False         PtfmPDOF    - Platform pitch tilt rotation DOF (flag)
False         PtfmYDOF    - Platform yaw rotation DOF (flag)
---------------------- INITIAL CONDITIONS --------------------------------------
        0   OoPDefl     - Initial out-of-plane blade-tip displacement (meters)
        0   IPDefl      - Initial in-plane blade-tip deflection (meters)
    14.92   BlPitch(1)  - Blade 1 initial pitch (degrees)
    14.92   BlPitch(2)  - Blade 2 initial pitch (degrees)
    14.92   BlPitch(3)  - Blade 3 initial pitch (degrees) [unused for 2 blades]
        0   TeetDefl    - Initial or fixed teeter angle (degrees) [unused for 3 blades]
        0   Azimuth     - Initial azimuth angle for blade 1 (degrees)
     12.1   RotSpeed    - Initial or fixed rotor speed (rpm)
        0   NacYaw      - Initial or fixed nacelle-yaw angle (degrees)
        0   TTDspFA     - Initial fore-aft tower-top displacement (meters)
        0   TTDspSS     - Initial side-to-side tower-top displacement (meters)
        0   PtfmSurge   - Initial or fixed horizontal surge translational displacement of platform (meters)
        0   PtfmSway    - Initial or fixed horizontal sway translational displacement of platform (meters)
        0   PtfmHeave   - Initial or fixed vertical heave translational displacement of platform (meters)
        0   PtfmRoll    - Initial or fixed roll tilt rotational displacement of platform (degrees)
        0   PtfmPitch   - Initial or fixed pitch tilt rotational displacement of platform (degrees)
        0   PtfmYaw     - Initial or fixed yaw rotational displacement of platform (degrees)
---------------------- TURBINE CONFIGURATION -----------------------------------
        3   NumBl       - Number of blades (-)
       63   TipRad      - The distance from the rotor apex to the blade tip (meters)
      1.5   HubRad      - The distance from the rotor apex to the blade root (meters)
     -2.5   PreCone(1)  - Blade 1 cone angle (degrees)
     -2.5   PreCone(2)  - Blade 2 cone angle (degrees)
     -2.5   PreCone(3)  - Blade 3 cone angle (degrees) [unused for 2 blades]
```

0 HubCM - Distance from rotor apex to hub mass [positive downwind] (meters)
0 UndSling - Undersling length [distance from teeter pin to the rotor apex] (meters) [unused for 3 blades]
0 Delta3 - Delta-3 angle for teetering rotors (degrees) [unused for 3 blades]
0 AzimB1Up - Azimuth value to use for I/O when blade 1 points up (degrees)
-5.0191 OverHang - Distance from yaw axis to rotor apex [3 blades] or teeter pin [2 blades] (meters)
1.912 ShftGagL - Distance from rotor apex [3 blades] or teeter pin [2 blades] to shaft strain gages [positive for upwind rotors] (meters)
-5 ShftTilt - Rotor shaft tilt angle (degrees)
1.9 NacCMxn - Downwind distance from the tower-top to the nacelle CM (meters)
0 NacCMyn - Lateral distance from the tower-top to the nacelle CM (meters)
1.75 NacCMzn - Vertical distance from the tower-top to the nacelle CM (meters)
-3.09528 NcIMUxn - Downwind distance from the tower-top to the nacelle IMU (meters)
0 NcIMUyn - Lateral distance from the tower-top to the nacelle IMU (meters)
2.23336 NcIMUzn - Vertical distance from the tower-top to the nacelle IMU (meters)
1.96256 Twr2Shft - Vertical distance from the tower-top to the rotor shaft (meters)
87.6 TowerHt - Height of tower above ground level [onshore] or MSL [offshore] (meters)
0 TowerBsHt - Height of tower base above ground level [onshore] or MSL [offshore] (meters)
0 PtfmCMxt - Downwind distance from the ground level [onshore] or MSL [offshore] to the platform CM (meters)
0 PtfmCMyt - Lateral distance from the ground level [onshore] or MSL [offshore] to the platform CM (meters)
0 PtfmCMzt - Vertical distance from the ground level [onshore] or MSL [offshore] to the platform CM (meters)
0 PtfmRefzt - Vertical distance from the ground level [onshore] or MSL [offshore] to the platform reference point (meters)
---------------------- MASS AND INERTIA --
0 TipMass(1) - Tip-brake mass, blade 1 (kg)
0 TipMass(2) - Tip-brake mass, blade 2 (kg)
0 TipMass(3) - Tip-brake mass, blade 3 (kg) [unused for 2 blades]
56780 HubMass - Hub mass (kg)
115926 HubIner - Hub inertia about rotor axis [3 blades] or teeter axis [2 blades] (kg m^2)
534.116 GenIner - Generator inertia about HSS (kg m^2)
240000 NacMass - Nacelle mass (kg)
2.60789E+06 NacYIner - Nacelle inertia about yaw axis (kg m^2)
0 YawBrMass - Yaw bearing mass (kg)
0 PtfmMass - Platform mass (kg)
0 PtfmRIner - Platform inertia for roll tilt rotation about the platform CM (kg m^2)
0 PtfmPIner - Platform inertia for pitch tilt rotation about the platform CM (kg m^2)
0 PtfmYIner - Platform inertia for yaw rotation about the platform CM (kg m^2)
---------------------- BLADE ---
17 BldNodes - Number of blade nodes (per blade) used for analysis (-)
"NRELOffshrBsline5MW_Blade.dat" BldFile(1) - Name of file containing properties for blade 1 (quoted string)
"NRELOffshrBsline5MW_Blade.dat" BldFile(2) - Name of file containing properties for blade 2 (quoted string)
"NRELOffshrBsline5MW_Blade.dat" BldFile(3) - Name of file containing properties for blade 3 (quoted string) [unused for 2 blades]
---------------------- ROTOR-TEETER --
0 TeetMod - Rotor-teeter spring/damper model {0: none, 1: standard, 2: user-defined from routine UserTeet} (switch) [unused for 3 blades]
0 TeetDmpP - Rotor-teeter damper position (degrees) [used only for 2 blades and when TeetMod=1]
0 TeetDmp - Rotor-teeter damping constant (N-m/(rad/s)) [used only for 2 blades and when TeetMod=1]
0 TeetCDmp - Rotor-teeter rate-independent Coulomb-damping moment (N-m) [used only for 2 blades and when TeetMod=1]
0 TeetSStP - Rotor-teeter soft-stop position (degrees) [used only for 2 blades and when TeetMod=1]
0 TeetHStP - Rotor-teeter hard-stop position (degrees) [used only for 2 blades and when TeetMod=1]
0 TeetSSSp - Rotor-teeter soft-stop linear-spring constant (N-m/rad) [used only for 2 blades and when TeetMod=1]
0 TeetHSSp - Rotor-teeter hard-stop linear-spring constant (N-m/rad) [used only for 2 blades and when TeetMod=1]
---------------------- DRIVETRAIN --
100 GBoxEff - Gearbox efficiency (%)
97 GBRatio - Gearbox ratio (-)
8.67637E+08 DTTorSpr - Drivetrain torsional spring (N-m/rad)
6.215E+06 DTTorDmp - Drivetrain torsional damper (N-m/(rad/s))
---------------------- FURLING ---
False Furling - Read in additional model properties for furling turbine (flag) [must currently be FALSE)
"unused" FurlFile - Name of file containing furling properties (quoted string) [unused when Furling=False]
---------------------- TOWER ---
20 TwrNodes - Number of tower nodes used for analysis (-)
"NRELOffshrBsline5MW_Onshore_ElastoDyn_Tower.dat" TwrFile - Name of file containing tower properties (quoted string)
---------------------- OUTPUT --
True SumPrint - Print summary data to "<RootName>.sum" (flag)
1 OutFile - Switch to determine where output will be placed: {1: in module output file only; 2: in glue code output file only; 3: both} (currently unused)
True TabDelim - Use tab delimiters in text tabular output file? (flag) (currently unused)
"ES10.3E2" OutFmt - Format used for text tabular output (except time). Resulting field should be 10 characters. (quoted string) (currently unused)
0 TStart - Time to begin tabular output (s) (currently unused)
1 DecFact - Decimation factor for tabular output {1: output every time step} (-) (currently unused)
0 NTwGages - Number of tower nodes that have strain gages for output [0 to 9] (-)

10, 19, 28 TwrGagNd - List of tower nodes that have strain gages [1 to TwrNodes] (-) [unused if NTwGages=0]
3 NBlGages - Number of blade nodes that have strain gages for output [0 to 9] (-)
5, 9, 13 BldGagNd - List of blade nodes that have strain gages [1 to BldNodes] (-) [unused if NBlGages=0]
OutList - The next line(s) contains a list of output parameters. See OutListParameters.xlsx for a listing of available output channels, (-)
"OoPDefl1" - Blade 1 out-of-plane and in-plane deflections and tip twist
"IPDefl1" - Blade 1 out-of-plane and in-plane deflections and tip twist
"TwstDefl1" - Blade 1 out-of-plane and in-plane deflections and tip twist
"BldPitch1" - Blade 1 pitch angle
"Azimuth" - Blade 1 azimuth angle
"RotSpeed" - Low-speed shaft and high-speed shaft speeds
"GenSpeed" - Low-speed shaft and high-speed shaft speeds
"TTDspFA" - Tower fore-aft and side-to-side displacements and top twist
"TTDspSS" - Tower fore-aft and side-to-side displacements and top twist
"TTDspTwst" - Tower fore-aft and side-to-side displacements and top twist
"Spn2MLxb1" - Blade 1 local edgewise and flapwise bending moments at span station 2 (approx. 50% span)
"Spn2MLyb1" - Blade 1 local edgewise and flapwise bending moments at span station 2 (approx. 50% span)
"RootFxb1" - Out-of-plane shear, in-plane shear, and axial forces at the root of blade 1
"RootFyb1" - Out-of-plane shear, in-plane shear, and axial forces at the root of blade 1
"RootFzb1" - Out-of-plane shear, in-plane shear, and axial forces at the root of blade 1
"RootMxb1" - In-plane bending, out-of-plane bending, and pitching moments at the root of blade 1
"RootMyb1" - In-plane bending, out-of-plane bending, and pitching moments at the root of blade 1
"RootMzb1" - In-plane bending, out-of-plane bending, and pitching moments at the root of blade 1
"RotTorq" - Rotor torque and low-speed shaft 0- and 90-bending moments at the main bearing
"LSSGagMya" - Rotor torque and low-speed shaft 0- and 90-bending moments at the main bearing
"LSSGagMza" - Rotor torque and low-speed shaft 0- and 90-bending moments at the main bearing
"YawBrFxp" - Fore-aft shear, side-to-side shear, and vertical forces at the top of the tower (not rotating with nacelle yaw)
"YawBrFyp" - Fore-aft shear, side-to-side shear, and vertical forces at the top of the tower (not rotating with nacelle yaw)
"YawBrFzp" - Fore-aft shear, side-to-side shear, and vertical forces at the top of the tower (not rotating with nacelle yaw)
"YawBrMxp" - Side-to-side bending, fore-aft bending, and yaw moments at the top of the tower (not rotating with nacelle yaw)
"YawBrMyp" - Side-to-side bending, fore-aft bending, and yaw moments at the top of the tower (not rotating with nacelle yaw)
"YawBrMzp" - Side-to-side bending, fore-aft bending, and yaw moments at the top of the tower (not rotating with nacelle yaw)
"TwrBsFxt" - Fore-aft shear, side-to-side shear, and vertical forces at the base of the tower (mudline)
"TwrBsFyt" - Fore-aft shear, side-to-side shear, and vertical forces at the base of the tower (mudline)
"TwrBsFzt" - Fore-aft shear, side-to-side shear, and vertical forces at the base of the tower (mudline)
"TwrBsMxt" - Side-to-side bending, fore-aft bending, and yaw moments at the base of the tower (mudline)
"TwrBsMyt" - Side-to-side bending, fore-aft bending, and yaw moments at the base of the tower (mudline)
"TwrBsMzt" - Side-to-side bending, fore-aft bending, and yaw moments at the base of the tower (mudline)
"Q_DrTr" - Displacement of drivetrain rotational-flexibility DOF (rad)
"QD_DrTr" - Velocity of drivetrain rotational-flexibility DOF (rad/s)
"QD_TFA1" - Velocity of 1st tower fore-aft bending mode DOF (m/s)
"Q_B1F1" - Displacement of 1st flapwise bending-mode DOF of blade 1 (m)
"Q_B2F1" - Displacement of 1st flapwise bending-mode DOF of blade 2 (m)
"Q_B3F1" - Displacement of 1st flapwise bending-mode DOF of blade 3 (m)
"QD_B1F1" - Velocity of 1st flapwise bending-mode DOF of blade 1 (m/s)
"QD_B2F1" - Velocity of 1st flapwise bending-mode DOF of blade 2 (m/s)
"QD_B3F1" - Velocity of 1st flapwise bending-mode DOF of blade 3 (m/s)

END of input file (the word "END" must appear in the first 3 columns of this last OutList line)

A3: FAST input InflowWind file

------- InflowWind v3.01.* INPUT FILE ---
12 m/s turbulent winds on 31x31 FF grid and tower for FAST CertTests #18, #19, #21, #22, #23, and #24

False Echo - Echo input data to <RootName>.ech (flag)
3 WindType - switch for wind file type (1=steady; 2=uniform; 3=binary TurbSim FF; 4=binary Bladed-style FF; 5=HAWC format; 6=User defined)
0 PropagationDir - Direction of wind propagation (meteoroligical rotation from aligned with X (positive rotates towards -Y) -- degrees)
1 NWindVel - Number of points to output the wind velocity (0 to 9)
0 WindVxiList - List of coordinates in the inertial X direction (m)
0 WindVyiList - List of coordinates in the inertial Y direction (m)
90 WindVziList - List of coordinates in the inertial Z direction (m)

```
================== Parameters for Steady Wind Conditions [used only for WindType = 1]
=========================
       18   HWindSpeed     - Horizontal windspeed                          (m/s)
       90   RefHt          - Reference height for horizontal wind speed      (m)
      0.2   PLexp          - Power law exponent                            (-)
================== Parameters for Uniform wind file   [used only for WindType = 2]
============================
"Wind/TurbSim-Denemeler_Shear_1_2_Basamak.hh"    Filename       - Filename of time series data for uniform wind field.
(-)
       90   RefHt          - Reference height for horizontal wind speed              (m)
   125.88   RefLength      - Reference length for linear horizontal and vertical sheer (-)
================== Parameters for Binary TurbSim Full-Field files   [used only for WindType = 3] ==============
"Wind/90m_18mps_twr.bts"    Filename       - Name of the Full field wind file to use (.bts)
================== Parameters for Binary Bladed-style Full-Field files   [used only for WindType = 4] =========
"Wind/90m_12mps_twr"    FilenameRoot   - Rootname of the full-field wind file to use (.wnd, .sum)
False         TowerFile      - Have tower file (.twr) (flag)
================== Parameters for HAWC-format binary files  [Only used with WindType = 5]
=====================
"wasp\Output\basic_5u.bin"    FileName_u     - name of the file containing the u-component fluctuating wind (.bin)
"wasp\Output\basic_5v.bin"    FileName_v     - name of the file containing the v-component fluctuating wind (.bin)
"wasp\Output\basic_5w.bin"    FileName_w     - name of the file containing the w-component fluctuating wind (.bin)
       64   nx             - number of grids in the x direction (in the 3 files above) (-)
       32   ny             - number of grids in the y direction (in the 3 files above) (-)
       32   nz             - number of grids in the z direction (in the 3 files above) (-)
       16   dx             - distance (in meters) between points in the x direction    (m)
        3   dy             - distance (in meters) between points in the y direction    (m)
        3   dz             - distance (in meters) between points in the z direction    (m)
       90   RefHt          - reference height; the height (in meters) of the vertical center of the grid (m)
  -------------   Scaling parameters for turbulence   ---------------------------------------------------------
        1   ScaleMethod    - Turbulence scaling method   [0 = none, 1 = direct scaling, 2 = calculate scaling factor based on a
desired standard deviation]
        1   SFx            - Turbulence scaling factor for the x direction (-)   [ScaleMethod=1]
        1   SFy            - Turbulence scaling factor for the y direction (-)   [ScaleMethod=1]
        1   SFz            - Turbulence scaling factor for the z direction (-)   [ScaleMethod=1]
       12   SigmaFx        - Turbulence standard deviation to calculate scaling from in x direction (m/s)    [ScaleMethod=2]
        8   SigmaFy        - Turbulence standard deviation to calculate scaling from in y direction (m/s)    [ScaleMethod=2]
        2   SigmaFz        - Turbulence standard deviation to calculate scaling from in z direction (m/s)    [ScaleMethod=2]
  -------------   Mean wind profile parameters (added to HAWC-format files)   ---------------------------------
        5   URef           - Mean u-component wind speed at the reference height (m/s)
        2   WindProfile    - Wind profile type (0=constant;1=logarithmic,2=power law)
      0.2   PLExp          - Power law exponent (-) (used for PL wind profile type only)
     0.03   Z0             - Surface roughness length (m) (used for LG wind profile type only)
====================== OUTPUT ==================================================
False         SumPrint     - Print summary data to <RootName>.IfW.sum (flag)
              OutList      - The next line(s) contains a list of output parameters.  See OutListParameters.xlsx for a listing of available
output channels, (-)
"Wind1VelX"               X-direction wind velocity at point WindList(1)
"Wind1VelY"               Y-direction wind velocity at point WindList(1)
"Wind1VelZ"               Z-direction wind velocity at point WindList(1)
END of input file (the word "END" must appear in the first 3 columns of this last OutList line)
---------------------------------------------------------------------------------------
```