Analysis and Control of Quasi-Polynomial DAE Systems

Ph.D. Thesis

Barna Pongrácz

Supervisor: Professor Katalin Hangos

Consultant: Gábor Szederkényi, PhD

Information Science PhD School
Department of Computer Science
University of Pannonia
Veszprém, Hungary

Process Control Research Group
Computer and Automation Research Institute
Hungarian Academy of Sciences
Budapest, Hungary

2008

Analysis and Control of Quasi-Polynomial DAE Systems

Értekezés doktori (PhD) fokozat elnyerése érdekében

Írta:
Pongrácz Barna

Készült a Pannon Egyetem Informatikai Tudományok Doktori Iskolája keretében

Témavezető: Dr. Hangos Katalin

Elfogadásra javaslom (igen / nem)

(aláírás)

A jelölt a doktori szigorlaton .....%-ot ért el

Veszprém, ...
a Szigorlati Bizottság elnöke

Az értekezést bírálóként elfogadásra javaslom:

Bíráló neve: ..... (igen / nem)

(aláírás)

Bíráló neve: ..... (igen / nem)

(aláírás)

A jelölt az értekezés nyilvános vitájában .....%-ot ért el

Veszprém, ...
a Bíráló Bizottság elnöke

A doktori (PhD) oklevél minősítése .....

.....
Az EDT elnöke

Tartalmi kivonat

Kvázipolinom alakú DAE modellek analízise és irányítása

A dinamikus rendszerek állapottér-modelljei rendszerint közönséges differenciálegyenlet-rendszer alakúak. Eme leírási mód rendkívül előnyös mind a dinamikus analízis, mind a szabályzótervezés szempontjából. Ugyanakkor a dinamikus rendszermodellek *általános alakja* differenciál-algebrai egyenletrendszer (DAE) alakú, ahol a dinamikát differenciálegyenletek és algebrai egyenletek vegyes halmaza írja le. Sajnos a DAE modellek kezeléséhez igen szegényes matematikai eszköztár adott.

A disszertáció elsődleges célja, hogy egy új megközelítéssel szélesítse ezt az eszköztárat. A szerző ennek érdekében ötvözi a DAE modell reprezentációt a - "hagyományos", közönséges differenciálegyenlet-rendszer alakú modellek esetén már bevált - kvázipolinomiális (QP) formával.

Az értekezés első része egy gázturbina QP modellje két különböző zéró dinamikájának stabilitás-analízisével foglalkozik. A szerző az egyik zéró dinamikára sikerrel alkalmazza az irodalomból ismert, rejtett QP-DAE modellekre kifejlesztett módszert, amellyel megbecsüli az adott állandósult állapot stabilitási környezetét is.

A szerző a fenti eredményeket felhasználva választ szabályzó struktúrát a gázturbina irányítására. Három különböző szabályzót tervez a fordulatszám szabályozására, ezek különböző szabályozási célokat valósítanak meg. Szimulációk segítségével a szerző megmutatja, hogy a szabályzott rendszerek robusztusak mind a környezeti zavarások változásaival, mind a modell paraméterek bizonytalanságaival szemben. Míg az első két szabályzó esetében a terhelés időbeni változását ismertnek tekinti, a harmadik, újszerű megközelítésben a szabályzó ezt a mennyiséget képes adaptívan megbecsülni.

A disszertáció utolsó témaköre egy nehéz matematikai problémával, nemminimális differenciálegyenlet-rendszerek (rejtett DAE-k) mozgásállandóinak (invariánsainak) megkeresésével foglalkozik QP alakú modellek esetén. A szerző megmutatja, hogy QP modellek esetén ez a probléma egyszerűen megoldható lineáris algebrai módszerekkel. Bemutat egy olyan polinomiális idejű, heurisztikus lépésektől mentes algoritmust, amellyel a QP alakú invariánsok visszakereshetők. A szerző megmutatja, hogy a kifejlesztett algoritmus magasabb rendű modellek esetén jóval hatékonyabb, mint az irodalomból ismert, szintén QP modellekre tervezett QPSI algoritmus.

Abstract

Analysis and control of quasi-polynomial DAE systems

The aim of this dissertation is to alloy dynamical models in differential-algebraic equations' (DAE) form with the advantageous quasi-polynomial (QP) formalism in order to develop new methods for the analysis and controller design for DAE models.

The local stability of two zero dynamics of the QP model of a gas turbine is investigated, using a known method for non-minimal QP (hidden QP-DAE) models.

Based on this analysis, different controllers are designed to control the rotational speed of the gas turbine. One of the controllers uses a novel approach, where the load torque is adaptively estimated.

Finally, a simple polynomial time algorithm is constructed to determine QP-type invariants of non-minimal QP (hidden QP-DAE) systems.

Auszug

Analysis und Kontrolle des Quasi-Polynominal DAE Systems

Das Ziel dieser Dissertation ist, die dynamische Models von differential-algebraische Gleichungsform (DAE) mit der vorteilhaften quasipolynominal (QP) Form zu legieren, um damit neue Methode entwickelt werden kann für die Analysis und Kontrolle von DAE Models.

Die lokale Stabilität wurde von zwei Zerodynamik von einem Kleinleistung Gasturbine QP- Model, durch einen bekannten Methode geprüft, bezüglich einer Nicht-minimal QP (versteckte QP-DAE) Model.

Aufgrund dieser Analysis die verschiedene Reglers wurde für die Regelung des Drehzahl Gasturbine geplant. Einer des Reglers benutzt neuartige Zutritt, er schätzt die Lastdrehmoment adaptiv.

Eine einfache Algorytmus mit Polynomial Zeit wurde schliesslich gefertigt, für die Bestimmung der Invarianten Typ QP von nicht-minimal QP (versteckte QP-DAE) Systems.

Acknowledgement

First and foremost, I would like to express my sincere gratitude to Prof. Katalin Hangos, my supervisor, for her continuous and cordial support, wisdom and patience she guided my research throughout my undergraduate and doctoral studies. I would like to thank dr. Gábor Szederkényi, my consultant, who always helped me with pleasure whenever I turned to him. I'm grateful to dr. Piroska Ailer for her kind collaboration and co-authoring in gas turbine research.

I would like to thank Prof. József Bokor, Head of Systems and Control Laboratory, Computer and Automation Research Institute who made my research possible. I am grateful for all the help I have received from every member of the Laboratory. I would like to thank Prof. István Győri and Prof. Béla Lantos, and also my research fellows and colleagues, dr. Balázs Kulcsár, dr. Erzsébet Németh, Zsuzsa Weinhandl, Tamás Péni, Gábor Rödönyi, Dávid Csercsik, Csaba Fazekas and dr. Attila Magyar for their help in my studies and research.

I would like to express my indebtedness to my family: my father, my wife and my little daughter for their love, encouragement and patience. This work could not be accomplished without their selfless support.

Contents

1 Introduction **1**
- 1.1 Motivation and background . . . 1
- 1.2 Problem statement and aims . . . 3
- 1.3 The structure of the thesis . . . 4
- 1.4 Notations . . . 5

2 Basic notions and literature review **7**
- 2.1 State equations in the form of ordinary differential equations (ODEs) 7
  - 2.1.1 Linear time invariant (LTI) system models . . . 7
  - 2.1.2 Nonlinear input-affine system models . . . 8
  - 2.1.3 Asymptotic stability of nonlinear input-affine systems . . . 8
  - 2.1.4 Reachability and minimality of input-affine systems . . . 10
  - 2.1.5 Invariants (first integrals) of ODE models . . . 12
  - 2.1.6 Control of nonlinear input-affine systems . . . 12
- 2.2 State equations in the form of differential algebraic equations (DAEs) 20
  - 2.2.1 The general DAE representation of lumped parameter system models . . . 21
  - 2.2.2 The differential index as the measure of complexity . . . 22
  - 2.2.3 Stability of DAEs . . . 22
  - 2.2.4 Reachability of DAEs . . . 23
  - 2.2.5 Control of DAE systems . . . 23
- 2.3 Quasi-polynomial (QP) models . . . 24
  - 2.3.1 QP models . . . 24
  - 2.3.2 Quasi-monomial transformations . . . 25
  - 2.3.3 Embedding into QP form . . . 25
  - 2.3.4 The QP-ODE form of lumped parameter system models . . . 27
  - 2.3.5 Zero dynamics of QP-ODE models . . . 27
  - 2.3.6 Invariants of QP models . . . 28
  - 2.3.7 Non-minimal autonomous QP models as DAE models . . . 29
  - 2.3.8 The QP-DAE form of controlled lumped parameter system models . . . 29
  - 2.3.9 Lotka-Volterra (LV) form of QP models . . . 30
  - 2.3.10 Local quadratic stability of LV systems . . . 31

3 Stability analysis of the zero dynamics of a low power gas turbine model in QP form **34**
3.1 Dynamic model of the gas turbine . . . 34
3.1.1 Modelling assumptions . . . 36
3.1.2 Conservation balances . . . 36
3.1.3 Conservation balances in intensive form . . . 37
3.1.4 Constitutive (algebraic) equations . . . 37
3.1.5 Dynamic model in nonlinear input affine QP-ODE form . . . 39
3.2 Local stability of the gas turbine with the turbine inlet total pressure held constant . . . 42
3.2.1 Zero dynamics for turbine inlet total pressure . . . 42
3.2.2 Local quadratic stability . . . 43
3.2.3 Estimation of the quadratic stability region . . . 44
3.2.4 Discussion . . . 46
3.3 Local stability of the gas turbine with the rotational speed held constant 47
3.4 Summary . . . 48

4 Controller design for a low-power gas turbine **50**
4.1 Literature review . . . 50
4.1.1 State space based control methods . . . 50
4.1.2 Control of gas turbines . . . 51
4.2 Control structure selection . . . 52
4.3 LQ servo controller design based on input-output linearization . . . 54
4.3.1 Control problems and aims . . . 54
4.3.2 Input-output linearization . . . 54
4.3.3 Servo controller with stabilizing feedback . . . 55
4.3.4 Simulation results . . . 57
4.4 MPT controller design based on the LQ controlled plant . . . 60
4.4.1 Control problems and aims . . . 60
4.4.2 Design of the LQ-MPT controller . . . 61
4.4.3 Simulation results . . . 61
4.5 Design of an LQ servo controller with adaptive load torque estimation 63
4.5.1 Control problems and aims . . . 63
4.5.2 Input-output linearization with load torque estimation . . . 63
4.5.3 Servo controller with stabilizing feedback . . . 66
4.5.4 Simulation results . . . 68
4.5.5 Robustness . . . 68
4.6 Summary . . . 70

5 Determining invariants (first integrals) of QP-ODEs **72**
5.1 An algorithm for determining a class of invariants in QP-ODEs . . . 72
5.1.1 The examined class of invariants . . . 72
5.1.2 The underlying principle of the algorithm . . . 73
5.1.3 The basic algorithm for retrieving single invariants . . . 74
5.1.4 Retrieval of multiple first integrals . . . 77
5.1.5 Implementation and computational properties of the algorithms 77

5.2 Algebraic properties of the algorithm 78
5.2.1 The effect of quasi-monomial transformations 78
5.2.2 The effect of algebraic equivalence transformations 79
5.3 Discussion 79
5.4 Examples 80
5.4.1 Example 5.1: A fed-batch fermentation process 80
5.4.2 Example 5.2: A Rikitake system 83
5.4.3 Example 5.3: An electric circuit 84
5.4.4 Example 5.4: A computational example with multiple first integrals 85
5.5 Summary 86

6 Conclusions **87**
6.1 Main contributions 87
6.2 Application areas, directions of future research 89
6.3 Publications 90

A Nomenclature, constants and coefficients of the gas turbine model **92**

B Zero dynamics of the gas turbine model for the rotational speed **95**

C. Tézisek magyar nyelven **97**

List of Figures

2.1 General scheme of feedback control 13

3.1 The main parts of a gas turbine 35
3.2 Schematic diagram of the DEUTZ T216 gas turbine 35
3.3 Quadratic stability neighborhood estimation 44
3.4 Transients started from corner points 45
3.5 Quadratic stability neighborhood estimation 46
3.6 Phase diagrams for the system with four different rotational speed values 48
3.7 Phase diagrams for the system with four different load torque values near a typical rotational speed value 48
3.8 Phase diagrams for the system with four different load torque values near minimal and maximal rotational speed values 49

4.1 The input-output linearized plant 56
4.2 LQ servo controller on the I/O linearized plant 58
4.3 Reference signal tracking for the rotational speed 58
4.4 Robustness of the LQ servo controller 59
4.5 LQ and MPT controllers on the I/O linearized plant 62
4.6 The effect of step-like changes of load torque on system variables . . . 63
4.7 LQ servo controller on the adaptively I/O linearized plant 67
4.8 Adaptive estimation of the load torque 68
4.9 Robustness of the adaptive LQ servo controller I. 69
4.10 Robustness of the adaptive LQ servo controller II. 70

5.1 The fed-batch fermenter 80
5.2 The two-disk dynamo system 83
5.3 The LC circuit 84

List of Tables

3.1 State, input, output and disturbance variables of the gas turbine model 41

A.1 Constants of the model of the DEUTZ T216 type gas turbine 93
A.2 Coefficients of the model of the DEUTZ T216 type gas turbine 94

Chapter 1

Introduction

If you can't come across the mountain, walk round it.
If you can't walk round, fly over it.
If you can't do this, sit down, and think over
whether it is really important to get to the other side.
If so, start to dig a tunnel.
/Maria Fontaine/

1.1 Motivation and background

The analysis and control of nonlinear dynamical systems is an undoubtedly challenging field of systems science. Its significance is well characterized by the number of different practical areas it is applied day by day, for instance in process systems [45], electric power generation [54], robotics [16], transportation systems such as wheeled vehicles [36], aircrafts [77], submarines [78] or spacecrafts [101], and in many other areas of modern life.

A large number of concentrated parameter nonlinear systems can be mathematically described by finite dimensional nonlinear *input-affine* state space models, where the state equation is a set of ordinary differential equations (ODEs) [46]. This is a definitely advantageous kind of representation, since there are a large number of theoretically well founded tools available for dynamical (stability, controllability, observability) analysis and also for controller design that can be applied thereon (see e.g. [88],[49],[71]). However, the generality of this representation form needs generally applicable (and therefore sometimes less powerful or non-constructive) methods for both dynamic analysis and controller synthesis. Although there are several model classes that have more specialized and practically applicable analysis and synthesis tools that rely on the special model structure, these are usually narrower classes of nonlinear dynamic models and therefore they suffer from the lack of representability.

Fortunately, there is a special physically nonlinear model class in the literature that fit to the nature of certain model classes. The class of quasi-polynomial (QP) models that are also called generalized Lotka-Volterra models has proved to be a unifying nonlinear model class that the majority of the nonlinear models with smooth nonlinearities can be algorithmically transformed to [47]. It is known that QP

system models have several advantageous features that simplify nonlinear dynamical analysis and control techniques: they possess a simple candidate structure for their Lyapunov function [40], [92], moreover their special structure can be exploited in controller design as well [69].

For the modelling, dynamic analysis and control of nonlinear dynamical systems, the special nonlinear nature of the system can be achieved directly: in thermodynamics, mechanics and engineering sciences there is a far amount of nonlinear model information, and one can find ways to exploit this knowledge base. For this purpose a really multi-disciplinary approach is needed that integrates the engineering knowledge of the application domain with the appropriate results of nonlinear systems and control theory. Such an approach is traditionally called a grey box methodology [19].

With this grey-box approach, the prior physical insight allows one to model nonlinear dynamical systems. A systematic seven step modelling procedure can be found in [46], where the dynamical system is modelled by a *mixed* set of differential *and* algebraic equations (DAEs): the differential equations originate from conservation balances for the extensive conserved quantities while the algebraic constitutive equations complete the model. This shows that the general form of the state equations of nonlinear concentrated parameter system models is a system of DAEs. In fortunate cases, the algebraic equations can be eliminated by variable substitution resulting in state equations in ODE form - the *differential index* of the DAE state equation (i.e. the minimal number of necessary time differentiations to be performed on the algebraic equations to get an ODE state equation) [59] is equal to one in these cases.

However, it occurs quite often that this elimination cannot be performed (strongly nonlinear index-one models or "high index" models, where the differential index is at least two) and the state equation remains in DAE form, and therefore the methods of dynamical analysis and controller synthesis for systems described by state space models in input-affine ODE form cannot be applied. Unfortunately, the dynamical analysis and controller design tools that can be found in the literature generally deal with only very narrow classes, e.g. controller design for DAE models which are only nonlinear in their differential variables [59], or the extension of stability theorems to models where the number of the differential and the algebraic equations are equal [84]. Moreover, the time-differentiation of the algebraic part of these DAEs yield to ODE models that are inherently non-minimal excluding the application of most of the standard dynamical analysis and controller design tools.

While the development of hundreds of years in the theory of differential equations gives a well-founded basis for 'standard' ODE models, the lack of theoretical background for DAEs makes the analytical treatment of these models difficult. Since a rising number of dynamical models - e. g. models of strongly nonlinear complex systems or high precision models that have an emerging role in a lot of practical fields with an ever-growing demand for punctuality - fall into the class of non-substitutable DAE models, the lack of widely applicable general analysis and controller design methods is a serious and emerging problem that calls for a solution.

1.2 Problem statement and aims

It has been discussed that the QP structure supplies several advantages for both the dynamical analysis and controller synthesis in case of 'classical' nonlinear ODE state-space models, without the loss of representability. This motivates us to alloy the DAE form with the QP formalism.

This thesis represents the very first steps towards the analysis and control of dynamical systems modelled by nonlinear input-affine DAEs. Three related main topics will be presented that are common in attempting to exploit the advantages of the quasi-polynomial formalism.

The Lyapunov stability analysis of steady state operating points of nonlinear models is not obvious even in the ODE case, since there is no generally applicable constructive method for finding an appropriate Lyapunov function [87]. Lyapunov function candidates can also be used for the estimation of the stability neighborhood (the domain of attraction) of an operating point. There are various methods for doing this for the classical smooth nonlinear ODE state space case [39], however they have limitations in their applicability. The most popular Zubov's method guarantees the exact knowledge of the domain of attraction, however it needs the solution of a partial differential equation and therefore it is hardly applicable in the general nonlinear case [53]. Based on Zubov's result, an iteratively computed rational type Lyapunov function is proposed in [97] to estimate the domain of attraction, however its degree (and also the order of the nonlinear model) is limited by the computational complexity of the method. Note that there are well applicable methods developed to narrower system classes, e.g. quadratic Lyapunov functions for polynomial systems [26].

For a special class of rank-deficient QP models, namely Lotka-Volterra models - which are *hidden QP-DAEs* - there is an algorithmic procedure for finding a quadratic Lyapunov function and estimating the attractor of a steady state point [68]. The first topic aims to *apply* this method to different zero dynamics of a strongly nonlinear quasi-polynomial model of a low-power gas turbine.

The second topic is directed towards designing controllers of different type for the low power gas turbine. This topic is QP-specific in the sense that the gas turbine model is a QP-DAE with substitutable algebraic equations. Although standard controller design methods are to be applied, the special QP structure is to be exploited again.

Finding constants of motion (invariants) of ODE systems is a very complex topic of mathematics, sometimes requiring intimate mathematical skills, usually with a lot of heuristic operations to be performed [3],[82],[90]. Furthermore, it has a great theoretical and practical importance in systems and control theory, because ODE state space models that are not integrable (i.e. have invariant(s)) are generally not reachable (controllable) [49]. The last topic aims to develop a new, algorithmizable method for finding QP type invariants of QP state space models.

1.3 The structure of the thesis

This thesis consists of 6 chapters and an Appendix with two parts. The scientific contributions are contained in Chapters 3,4,5 - each one gives the material of one scientific thesis. The structure of the thesis is organized as follows:

After this introductory chapter, **Chapter 2** describes the three main representation types of lumped parameter systems (ODE, DAE and QP state space models) together with their fundamental properties, and introduces the related notions and tools to be used. Key notions, properties and techniques are linked with a literature review.

Chapter 3 concerns with the stability analysis of two zero dynamics of a low power gas turbine model. First, the strongly nonlinear QP-DAE model of the gas turbine taken from literature is presented in Section 3.1. The quadratic stability analysis of the gas turbine with the turbine inlet total pressure held constant is performed in Section 3.2, with the stability region estimation of the operating point. Section 3.3 concerns with the stability analysis of the gas turbine with the rotational speed held constant.

Chapter 4 is dedicated to controller design for the low power gas turbine. After a literature review (Section 4.1), control structure selection for the gas turbine is performed in Section 4.2 based on the results of Chapter 3. Then, different input-output linearization based controllers are designed: an LQ servo controller in Section 4.3 to track a prescribed reference signal for the rotational speed, and an LQ+MPT controller in Section 4.4 to keep the rotational speed and its time-derivative between predefined bounds. A novel, adaptive input-output linearization based LQ servo controller is designed in Section 4.5 to track a reference signal for the rotational speed, while the most important environmental disturbance of the gas turbine, namely the load torque is estimated dynamically. The performance and robustness of these controllers are tested via simulations performed in the MATLAB/SIMULINK computation environment.

Chapter 5 describes a new algorithmic method based on linear algebra that is capable of finding QP type invariants (first integrals) of non-minimal QP-ODE (i.e. hidden QP-DAE) state space models. Sections 5.1.3 and 5.1.4 describe the retrieval algorithm for single and multiple first integrals, respectively. The computational properties and the implementation of these algorithms in the MATLAB computational environment is discussed in Section 5.1.5. Section 5.2 concerns with the algebraic invariance properties of the algorithms under different transformations. Finally the operation of the algorithms are demonstrated on several examples in Section 5.4.

Chapter 6 contains the new scientific results in the form of three Theses, the possible directions of future research and the list of own publications.

Appendix A contains the explanation of the variables, coefficients and the values of the nominal parameters of the low power gas turbine model.

Appendix B presents the zero dynamics of the low power gas turbine for the rotational speed.

Appendix C contains the new scientific results in Hungarian.

1.4 Notations

The most important notations and acronyms used throughout the Thesis are enlisted in this section.

Basic mathematical notations

$x \in S$	x is an element of S
$X \subset S$	X is a subset of S
R	the set of real numbers
R^+	the set of positive real numbers
R_0^+	the set of non-negative real numbers
$w \in R^n$	w is a real vector with n components
w_i	i-th component of vector w
w^T	transpose of vector w
$f \in \mathbb{R}^n \mapsto \mathbb{R}^m$	function that assigns vectors of $\mathbb{R}^m$ to vectors of $\mathbb{R}^n$
$\frac{\partial f}{\partial x}$	general Jacobian matrix of function $f \in \mathbb{R}^n \mapsto \mathbb{R}^m$, $x \mapsto f(x)$
$\frac{\partial f}{\partial x_i}$	partial derivative of function $f \in \mathbb{R}^n \mapsto \mathbb{R}^m$, $x \mapsto f(x)$ by x_i
$M \in R^{n \times m}$	M is a real $n \times m$ matrix
$M_{i,j}$	(i,j)-th element of matrix M
M^T	transpose of matrix M

Acronyms

AFLT	Adaptive Feedback Linearization Theorem
DAE	differential algebraic equation
LMI	linear matrix inequality
LTI	linear time invariant
LQ	linear quadratic
LV	Lotka-Volterra
MPT	Multi-Parametric Toolbox (used for controller design)
ODE	ordinary differential equation
QM	quasi-monomial
QMT	quasi-monomial transformation
QP	quasi-polynomial
RHS	right-hand side
SISO	single input - single output

Notations related to state space models

d	vector of external disturbances
f	state function of nonlinear input-affine system models
g	input function of single input nonlinear input-affine system models
h	output function of input-affine system models
u	vector of system inputs
x	vector of system states
y	vector of system outputs
A	coefficient matrix of QP models, or state matrix of LTI models
A_{LV}	parameter matrix of LV models
B	exponent matrix of QP models, or input matrix of LTI models
C	exponent matrix of QMTs, or output matrix of LTI models
U	vector of state variables of LV models, or vector of QMs of QP models
λ	parameter vector of QP models
λ_{LV}	parameter vector of LV models

Subscripts

w_d	refers to dimensionless version of a system variable w
w_{max}	refers to maximal value of a system variable w
w_{min}	refers to minimal value of a system variable w
w_{ref}	refers to reference value of a system variable w
w_0	refers to initial value of a system variable w

Superscripts

$\bar{w}$	centered version of a system variable w
$\widehat{w}$	estimated version of a system variable w
w^*	steady state value of a system variable w
$\dot{w} = \frac{dw}{dt}$	time-derivative of a system variable w

Chapter 2

Basic notions and literature review

The purpose of this chapter to get the Reader acquainted with those notions and techniques that should be necessary for the understanding of the following chapters. These notions and tools are collected around the three main system model forms (ODE, DAE and QP) that will be used in this Thesis.

2.1 State equations in the form of ordinary differential equations (ODEs)

The majority of nonlinear lumped parameter systems with smooth nonlinearities can be represented in the form of ordinary differential equations (ODEs). As we will see, this purely differential form has the great advantage that dynamic analysis and plenty of controller design techniques are developed for it [45]. This section concerns with the dynamical (stability and reachability) analysis and shows several controller design techniques for systems represented in ODE form.

2.1.1 Linear time invariant (LTI) system models

First, let us get acquainted with the LTI state space representation consisting of an ODE state equation and a linear output equation:

$$\dot{x} = Ax + Bu \tag{2.1}$$
$$y = Cx + Dy \tag{2.2}$$

where $x \in \mathbb{R}^n,\ u \in \mathbb{R}^p,\ y \in \mathbb{R}^q$ are the vectors of state, input and output variables, respectively, $A \in \mathbb{R}^{n\times n}, B \in \mathbb{R}^{n\times p}, C \in \mathbb{R}^{q\times n}$ are constant matrices. It is important to note that an n-th order state space model is never unique: there are infinitely many n-th order state space models describing the same system [44].

The importance of LTI system models is twofold: a lot of nonlinear ODEs can be transformed to an LTI one with an appropriate control input function, giving rise to the application of linear controllers on the linearized model, moreover LTI models are often derived from nonlinear state space models by local linearization around a steady state operating point.

2.1.2 Nonlinear input-affine system models

As an advantageous mathematical representation for lumped parameter systems, the nonlinear input-affine state space model is proposed [45].

Nonlinear state space models are described by vector fields which are constructions in vector calculus [29]. A vector field associates a vector to every point of an Euclidean space. Thus, a vector field H that associates a d_2 dimensional real vector to each point of the d_1 dimensional real coordinate space can be represented as a vector valued function: $H : \mathbb{R}^{d_1} \rightarrow \mathbb{R}^{d_2}$.

A vector field H is called *affine* in $w = [w_1, \ldots, w_k]^T$ if its vector valued function representation $H(v, w)$ can be written in the form

$$H(v, w) = F(v) + \sum_{j=1}^{k} G_j(v) w_j$$

where F and $G_j, j = 1, \ldots, k$ are arbitrary vector valued functions. The term input-affine system model denotes a system model that is affine in the input variables.

Let us denote the vector of states by $x \in \chi$, where χ is an open subset of $\mathbb{R}^n$, the vector of system inputs by $u \in \mathbb{R}^p$ and the vector of system outputs by $y \in \mathbb{R}^q$. The general input-affine form consists of a state equation in the form of an ordinary differential equation (ODE), and an output equation [45]:

$$\dot{x} = \frac{dx}{dt} = f(x) + \sum_{i=1}^{p} g_i(x) u_i \tag{2.3}$$

$$y = h(x) \tag{2.4}$$

where $f, g_i \in \mathbb{R}^n \mapsto \mathbb{R}^n,\ i = 1, \ldots, p$ and $h \in \mathbb{R}^n \mapsto \mathbb{R}^q$ are smooth nonlinear vector fields (i.e. their vector valued function representations are smooth and nonlinear), and $u = [u_1, \ldots, u_p]^T$.

2.1.3 Asymptotic stability of nonlinear input-affine systems

This section deals with the notion and investigation methods of asymptotic stability. Together with the local (eigenvalue checking) method, the main objective is to discuss how to investigate the asymptotic stability of nonlinear systems by means of the widely used Lyapunov-technique [49].

Determine the input u of the nonlinear input affine state equation (2.3) as a function of x (i.e. apply a control law that is in the form of $u = \psi(x)$) or simply truncate it by setting the input to zero ($u = 0$). Then (2.3) becomes an autonomous ODE that can be written in a 'controlled' ('closed loop') or 'truncated' form

$$\dot{x} = \frac{dx}{dt} = f_a(x) \tag{2.5}$$

where $f_a \in \mathbb{R}^n \mapsto \mathbb{R}^n$ is again a smooth nonlinear vector field.

Consider the autonomous nonlinear ODE in (2.5). We call x^* an equilibrium point (or steady state point) of (2.5) if it fulfills the equation

$$f_a(x^*) = 0$$

e.g. $\frac{d}{dt}x^* = 0$. Denote $x_0 = x(0)$ the initial condition of (2.5).

- An equilibrium point x^* of (2.5) is called *stable* in Lyapunov sense, if for arbitrary $\epsilon > 0$ there is a $\delta > 0$ such that if $||x_0 - x^*|| < \delta$ then $||x(t) - x^*|| < \varepsilon$ for every $t > 0$, where $|| \cdot ||$ is a suitable vector norm.

- An equilibrium point x^* is called *asymptotically stable*, if it is stable in Lyapunov sense, and $lim_{t\to\infty}x(t) = x^*$.

- If x^* is not stable, it is called *unstable*.

- We call x^* locally (asymptotically) stable, if there is a neighborhood $U \neq \mathbb{R}^n$ of x^*, where x^* is (asymptotically) stable.

- If x^* is (asymptotically) stable in R^n, then x^* is a globally (asymptotically) stable equilibrium of (2.5).

As we can see, stability (asymptotic stability) is a property of equilibrium points in case of nonlinear ODEs. However, the stability (asymptotic stability) of the n-th order LTI system model (2.1)-(2.2) is a *system property,* since it is realization independent: any other n-th order state space models describing the same system as (2.1)-(2.2) are stable (asymptotically stable). Moreover, stability (asymptotic stability) is always global in the LTI case. Note that if (2.1)-(2.2) is *asymptotically* stable, then its equilibrium point is *unique*: it is the origin of the state space, i.e. $x^* = 0 \in \mathbb{R}^n$.

The asymptotic stability of LTI system models can easily be checked by the eigenvalues of its state matrix A: (2.1)-(2.2) is asymptotically stable if and only if $Re(\lambda_i(A)) < 0,\ i = 1, \ldots, n$, where the eigenvalues $\lambda_i, i = 1, \ldots, n$ of A are the solutions of the equation

$$det(\lambda I - A) = 0$$

where $I \in \mathbb{R}^{n\times n}$ is the unit matrix [44]. In addition, if $Re(\lambda_i(A)) > 0$ for some i, then (2.1)-(2.2) is unstable. It is important to note that if $Re(\lambda_i(A) \leq 0$ and there is at least one eigenvalue with zero real part, then stability can be checked in the following way: Let (2.1)-(2.2) have s different eigenvalues with zero real parts. If the eigenvectors belonging to these s eigenvalues span an s-dimensional space, then (2.1)-(2.2) is stable (not asymptotically!), otherwise it is unstable. Note that the (non-asymptotic) stability of LTI system models is also a system property.

This eigenvalue-checking method can be extended to nonlinear input affine system models: if x^* is an equilibrium of (2.5) and the locally linearized model of (2.5) around x^* is asymptotically stable (unstable), then x^* is a *locally* asymptotically stable (unstable) equilibrium of (2.5). Although this method is easily applicable, it does not give us any information about the asymptotic stability neighborhood of the equilibrium point, moreover it can prove asymptotic stability and instability, but it does not give us any information if the locally linearized model has eigenvalue(s) with zero real part(s).

Lyapunov theorem

The most widely used technique for the (global) stability analysis of nonlinear input-affine systems is the so-called Lyapunov technique. For proving asymptotic stability of the nonlinear system (2.5) Lyapunov functions are used. These functions can be regarded as "generalized energy" functions, since they are scalar valued, positive definite functions. If the system (2.5) is asymptotically stable, then this energy decreases with time. Therefore a Lyapunov function $V(x)$ has to fulfill the following criteria:

1. V is a scalar function of the state variables of (2.5):

$$V \in \mathbb{R}^n \to \mathbb{R}_0^+$$

2. $V(x)$ is positive definite at the equilibrium x^*:

$$V(x) > 0 \text{ if } x \neq x^*, \quad V(x^*) = 0$$

3. Its time-derivative is negative definite at the equilibrium x^* :

$$\frac{dV}{dt} = \frac{\partial V}{\partial x}\frac{dx}{dt} < 0 \text{ if } x \neq x^*, \qquad \frac{dV}{dt} = 0, \text{ if } x = x^*$$

The Lyapunov-theorem states that *the system (2.5) is asymptotically stable if there exists a Lyapunov function with the properties above.*

Note that finding an appropriate Lyapunov function is not constructive in general. However, there are system classes where Lyapunov functions can be given constructively, e.g. in LTI case [44], or for linear parameter varying systems (see e.g. a control relevant application in [86], or for Hamiltonian systems [43]).

2.1.4 Reachability and minimality of input-affine systems

In this section another dynamical property, namely the reachability will be discussed.

An LTI state space model in the form of (2.1)-(2.2) is called *reachable*, if it is possible to drive an arbitrary state $x(t_1) \in \chi$ to an arbitrary state $x(t_2) \in \chi$ with an appropriate input function, in finite time $t = t_2 - t_1$. Reachability is always *global* in the LTI case.

The input-affine state space model (2.3)-(2.4) is said to be *locally* reachable around the state $x(t_1) \in \chi$, if there exists a neighborhood $U \subset \chi$ of $x(t_1)$ such that it is possible to drive $x(t_1)$ to an arbitrary state $x(t_2) \in U$ with an appropriate control input function, in finite time $t = t_2 - t_1$.

The reachability of state space models is a necessary condition for the application of most of the controller design techniques.

An LTI system model in the form of (2.1)-(2.2) is reachable if and only if its controllability matrix $\mathcal{C} = [BAB \dots A^{n-1}B]$ is of full rank. Note that reachability of LTI system models is also termed as controllability, but to avoid any trouble about notions the term 'reachability' will be used uniformly throughout this Thesis.

The reachability of the nonlinear input-affine system model (2.3)-(2.4) can also be investigated by an algorithm [49].

Before presenting this algorithm we have to get acquainted with some notions. Let $f, g \in \mathbb{R}^n \to \mathbb{R}^n$ be two smooth vector fields. Denote the general Jacobian matrix of f by $\frac{\partial f}{\partial x}$:

$$\frac{\partial f}{\partial x} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}, \quad \text{where } f = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_n \end{bmatrix}$$

A *Lie-bracket* (or Lie product) of f and g is another vector field $[f, g] \in \mathbb{R}^n \to \mathbb{R}^n$ defined as

$$[f, g](x) = \frac{\partial g}{\partial x} f(x) - \frac{\partial f}{\partial x} g(x)$$

A *distribution* is a function that assigns a subspace of R^n for each point $x \in U$, where U is an open subset of R^n. A distribution Δ can be regarded as a subspace depending on x, that is spanned by some vector fields $\delta_1, \ldots, \delta_d$ at each point x:

$$\Delta(x) = \text{span}\{\delta_1(x), \ldots, \delta_d(x)\}$$

The algorithm proposed by [49] computes the so-called reachability distribution, using Lie-brackets.

- The initial step is:
$$\Delta_0 = \text{span}\{g_i, i = 1, \ldots, q\}$$
- The k-th step of the algorithm is the following. Let $\Delta_{k-1} = \text{span}\{\delta_j^{(k-1)}, j = 1, \ldots, d\}$. Then Δ_k is computed as
$$\Delta_k = \text{span}\{\delta_j^{(k-1)}, [f, \delta_j^{(k-1)}], [g_i, \delta_j^{(k-1)}] \ , \quad i = 1, \ldots, p, \ j = 1, \ldots, d\} \qquad (2.6)$$
- The stopping condition of the algorithm is: $dim(\Delta_{k^*}) = dim(\Delta_{k^*-1})$ for some k^*.

The distribution $\Delta = \Delta_{k^*}$ is called the reachability distribution of (2.3)-(2.4). Since the maximal dimension of Δ is n, the algorithm consists of finite steps.

If the dimension of the reachability distribution is maximal at a point $x_0 \in \chi$, i.e. $dim(\Delta(x_0)) = n$, then there is a neighborhood U_0 of x_0 such that (2.3)-(2.4) is reachable locally on U_0. If $dim(\Delta) = n$ independently of x, then (2.3)-(2.4) is (globally) reachable.

A state space model is called *minimal,* if its dynamics is described by the minimum number of state variables (i.e. the dimension of the state space (n) is minimal) [49]. It is known that a state space model is minimal if and only if it is reachable and observable. Observability roughly means that the exact knowledge of the input and output signals and also of the state space model is enough to determine the state trajectories of the system.

Thus, the non-minimality of (2.3)-(2.4) might mean that there are hidden algebraic constraints on the state variables, which is the subject of the next section.

2.1.5 Invariants (first integrals) of ODE models

A function $I : \mathbb{R}^n \mapsto \mathbb{R}$ is called an invariant (constant of motion, first integral or hidden algebraic constraint) of the non-autonomous ODE defined in (2.3) if

$$\frac{d}{dt} I = \frac{\partial I}{\partial x} \cdot \dot{x} = 0. \tag{2.7}$$

The determination of invariants of ODEs has been occupying scientists' mind for the last 100 years. As the most frequent and widely used approaches, methods based on Lie-symmetries [90] and Painlevé analysis [3],[82] has to be mentioned. Unfortunately, these methods are not applicable for arbitrary types of first integrals - see e.g. [76] that concerns with invariants that cannot be determined using Lie-symmetries. Additionally, their often heuristic and generally symbolic nature makes the determination of the invariants difficult.

First integrals play a great role in modern systems and control theory e.g. in the field of canonical representations, reachability and observability analysis [49] and also in the stabilization of nonlinear systems [88],[63]. Moreover, if the given dynamical system is not integrable, then its first integrals (if they exist) give us very useful information about the properties of the solutions and about physically meaningful conserved quantities.

If an ODE state space model has first integral(s), then it is indeed non-minimal because its dynamics can be described with a lower number of state variables. Moreover, its state trajectories evolve on a manifold (i.e. a lower dimensional subset of the state space) determined by its first integral(s). As a consequence, these state space models are not reachable, since only the states on the manifold determined by the invariant(s) can be reached. It has to be emphasized that from the fact that (2.3) with zero inputs ($u_i = 0, i = 1, \ldots, p$) has an invariant does not imply that (2.3) with arbitrary inputs also has this invariant, since e.g. a state feedback may result a completely different autonomous ODE. However, there are invariants that cannot be influenced by the control input.

For systems in the form (2.3-2.4) Isidori proposes a method to determine first integrals [49] that are independent of the control input variable. The complexity of this method is well characterized by the fact that it demands the *symbolic* solution of *systems* of partial differential equations (PDEs). These difficulties are well demonstrated on the reachability analysis of a low (third) order fermentation process model needing the solution of a system of two PDEs. There are other methods that consider model classes that can only represent a narrower class of lumped parameter systems: e.g. positive systems [63], polynomial systems [67], or single n-th order nonlinear ODEs [4].

2.1.6 Control of nonlinear input-affine systems

This section is dedicated to introduce control techniques and control relevant notions that will be used throughout the Thesis.

The *aim* of control is to *modify* a system in such a way that it fulfills a *prescribed control goal*. This modification is usually done by a *feedback*: the input variable u of the system is determined as a function of the signals (the outputs and/or the states)

of the system (see Fig. 2.1). A system with a feedback controller is often called as *closed loop* system, in contrast to the uncontrolled, *open loop* system.

Feedback controllers can be classified by their properties. A feedback can be

- a *state/output* feedback, if it uses the state/output signals of the system;
- *linear/nonlinear* if it is a linear/nonlinear function of the signals (outputs, states) of the system;
- *dynamic*, if it also contains the derivatives of the states/outputs of the system, and *static* otherwise;
- a *full state feedback*, if all components of the state variable vector are used in the computation of the state feedback.

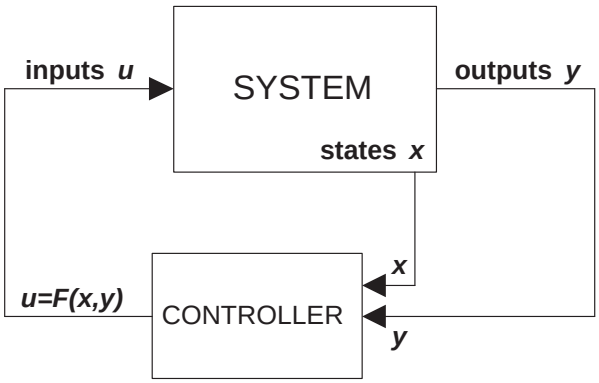

Figure 2.1: General scheme of feedback control

In the following, a few different types of feedback controllers will be described.

Linear quadratic (LQ) and LQ servo controllers

LQ and LQ servo controllers can be applied to LTI models in the form of (2.1)-(2.2), but they can also be applied to (locally or globally) linearized nonlinear input-affine models. (The input-output linearization of nonlinear input affine models will be discussed in the next section.) LQ and LQ servo controllers use linear static feedback. In the following, we focus on LQ controllers with full state feedback.

The *problem statement of LQ control* is the following: Given an LTI state space model in the form (2.1)-(2.2). Minimize the following functional (the control cost)

$$J(x,u) = \frac{1}{2}\int_0^T x^T(t)Qx(t) + u^T(t)Ru(t)dt \tag{2.8}$$

by an appropriate input $u(t), t \in [0,T]$, where Q and R are positive definite, symmetric weighting matrices for the states and inputs, respectively.

If (2.1)-(2.2) is reachable and observable (i.e. from the exact knowledge of system parameters, $u(t)$ and $y(t)$, the states $x(t)$ can be determined), the stationary solution

(i.e. when $T \to \infty$) of this control problem can be obtained by solving the so-called Control Algebraic Ricatti Equation for P:

$$A^T P + PA - PBR^{-1}B^T P + Q = 0 \tag{2.9}$$

It is known that this matrix equation has a unique positive definite symmetric solution K [44]. Then, the static linear full state feedback

$$u(t) = -Kx(t) = -R^{-1}B^T Px(t) \tag{2.10}$$

minimizes the functional (2.8) if $T \to \infty$. The weighting matrices Q and R are the tuning parameters of the LQ controller. The quadratic term $x^T Qx$ in (2.8) penalizes the deviation of the state vector from the reference state $x^* = 0$, while the other quadratic term $u^T Ru$ penalizes the control energy. The magnitude of elements in Q and R can be chosen according to the control aim: choosing a Q that is relatively 'big' compared to R, the deviations from the zero state are smaller, however it needs more input energy (cheap control), while with a Q that is relatively 'small' compared to R the transients of the controlled plant have higher deviations while the control need less energy (expensive control). It is important to note that an LQ controller designed to an arbitrary LTI model guarantees the asymptotic stability of the closed loop system independently of the design parameters Q and R. Another advantage of LQ controllers is that the closed loop system is robust against some model parameter uncertainties and environmental disturbances [45].

While LQ controllers make the system track the prescribed trajectory $x(t) = 0$, LQ servo controllers are designed to track a prescribed reference output signal $y_{ref}(t)$. Consider the simplest case, when (2.1)-(2.2) is a cascade of integrators:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= x_3 \\ &\vdots \\ \dot{x}_{n-1} &= x_n \\ \dot{x}_n &= u \\ y &= x_1 \end{aligned}$$

This can be re-written in matrix-vector form:

$$\frac{dx}{dt} = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} u(t) \tag{2.11}$$

$$y(t) = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \end{bmatrix} x(t) \tag{2.12}$$

Compute an LQ state feedback in the form

$$u(t) = -Kx(t) = -k_1 x_1(t) - k_2 x_2(t) - \ldots - k_n x_n(t)$$

The closed loop system can be written in the form

$$\frac{dx}{dt} = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ -k_1 & -k_2 & -k_3 & \dots & -k_n \end{bmatrix} x(t) \tag{2.13}$$

$$y(t) = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \end{bmatrix} x(t) \tag{2.14}$$

Recall that since this closed loop system is asymptotically stable, it asymptotically tracks the reference state $x^* = 0$.

This closed loop system has to be modified in such a way that $y = x_1$ has to track the prescribed reference signal $y_{ref}(t)$, therefore the state to be tracked is $x^*_{ref}(t) = [y_{ref}(t)\ 0\ \dots\ 0]^T$. Define the tracking error as

$$e(t) = y_{ref}(t) - y(t) = y_{ref}(t) - x_1(t)$$

Modify the LQ control law as

$$\begin{aligned} u(t) &= -k_1(-e(t)) - k_2x_2(t) - \dots - k_nx_n(t) = \\ &= -k_1(x_1(t) - y_{ref}(t)) - k_2x_2(t) - \dots - k_nx_n(t) = -Kx(t) + k_1y_{ref}(t) \end{aligned} \tag{2.15}$$

This control law applied to (2.11) guarantees that $lim_{t\to\infty}e(t) = 0$ if $lim_{t\to\infty}y_{ref}(t)$ exists; and therefore $x_1(t)$ asymptotically tracks $y_{ref}(t)$.

The LQ servo controlled cascade of integrators can be written in the following matrix-vector form:

$$\frac{dx}{dt} = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ -k_1 & -k_2 & -k_3 & \dots & -k_n \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ k_1 \end{bmatrix} y_{ref}(t) \tag{2.16}$$

$$y(t) = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \end{bmatrix} x(t) \tag{2.17}$$

If (2.1)-(2.2) is not a cascade of integrators, an LQ servo controller can be built by defining the tracking error e by a new differential equation:

$$\dot{e} = y_{ref}(t) - y(t) = y_{ref}(t) - Cx(t) \tag{2.18}$$

This yields to the following $(n+1)$ dimensional LTI model:

$$\frac{d}{dt}\begin{bmatrix} e(t) \\ x(t) \end{bmatrix} = \begin{bmatrix} 0 & -C \\ 0_{n\times 1} & A \end{bmatrix}\begin{bmatrix} e(t) \\ x(t) \end{bmatrix} + \begin{bmatrix} 0 \\ B \end{bmatrix} u(t) + \begin{bmatrix} 1 \\ 0_{n\times 1} \end{bmatrix} y_{ref}(t) \tag{2.19}$$

$$y(t) = Cx(t) \tag{2.20}$$

where 0 is a single zero element, $0_{n\times 1}$ is an n dimensional zero vector, while A, B and C are the matrices of the LTI model (2.1)-(2.2). An LQ feedback applied thereon in the form

$$u = -k_ee(t) - k_1x_1(t) - \dots - k_nx_n(t)$$

guarantees the asymptotic stability of the system, moreover - if $y(t) = x_1(t)$ - then $x_1(t)$ asymptotically tracks $y_{ref}(t)$.

Note that the advantageous properties of LQ controllers (stability, robustness) stand for LQ servo controllers, too.

Constrained linear optimal control

LQ controllers do not guarantee that the different signals: the states, outputs, and the control input computed by the controller are bounded, however it is a significant criteria in many physical systems.

The constrained linear control technique (see, e.g. [17], [18], [20]) solves this problem for LTI systems. Constrained linear optimal control considers the following discrete time SISO LTI system class:

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) + Du(k) \end{aligned} \tag{2.21}$$

where $k = 0, 1, \ldots$ is the discrete time, $x(k) \in \mathbb{R}^n$ is the state vector, $u(k) \in \mathbb{R}$ and $y(k) \in \mathbb{R}$ are the input and output respectively. A, B, C and D are real matrices of appropriate dimensions. Note that a discrete time LTI model in the form (2.21) can be obtained from a continuous LTI model (2.1)-(2.2) by time-discretization [44].

The so-called Constrained Finite Time Optimal Control Problem with quadratic cost is to find an input sequence $\{u(0), \ldots, u(N-1)\}$ such that it minimizes the cost function

$$J(x, u) = x(N)^T P_N x(N) + \sum_{k=1}^{N-1} [u(k)^T R_c u(k) + x(k)^T Q_c x(k)] \tag{2.22}$$

subject to the constraints

$$u_{min} \leq u(k) \leq u_{max} \tag{2.23}$$

$$y_{min} \leq y(k) \leq y_{max} \tag{2.24}$$

$$Hx(k) \leq K \tag{2.25}$$

where P_N, Q_c and R_c are positive definite symmetric weighting matrices (design parameters), $H \in \mathbb{R}^{l \times n}$ and $K \in \mathbb{R}^l$ are matrices defining a prescribed polytopic region of the state space inside which the state variables have to evolve.

The constrained linear optimal control problem can be solved by multi-parametric programming. For the numerical solution, the Multi-Parametric Toolbox (MPT) [61] of the MATLAB computational environment has been used.

Input-output linearization of nonlinear input-affine systems [49]

Input-output linearization is the application of an appropriate nonlinear state feedback to obtain a system description which is linear in input-output sense. Consider the general nonlinear input-affine system model in (2.3)-(2.4) with single input and

single output:

$$\dot{x} = \frac{dx}{dt} = f(x) + \sum_{i=1}^{p} g(x)u \tag{2.26}$$

$$y = h(x) \tag{2.27}$$

where $x \in \chi \subset \mathbb{R}^n$, $u, y \in \mathbb{R}$ and $f, g \in \mathbb{R}^n \mapsto \mathbb{R}^n$ and $h \in \mathbb{R}^n \mapsto \mathbb{R}$ are smooth nonlinear vector fields.

The *relative degree* of (2.26)-(2.27) at a point $x_0 \in \chi$ is a nonnegative integer r, that fulfills the following conditions:

1. $L_g L_f^k h(x) = 0$ for all x in a neighborhood of x_0 and $k < r - 1$
2. $L_g L_f^{r-1} h(x) \neq 0$

where $L_f h(x), L_f^k h(x)$ denote Lie derivatives

$$L_f h(x) = L_f^1 h(x) = \frac{\partial h}{\partial x} f(x) \ , \quad L_f^k h(x) = L_f L_f^{k-1} h(x) \ , \quad k = 2, 3, \ldots$$

In other words, the relative degree r is the minimum number of time differentiations that has to be performed on the output y to get the input u explicitly appear.

The zero constrained output dynamics - or *zero dynamics* in brief - of (2.26)-(2.27) is the dynamics of (2.26) that fulfills the constraint

$$y(t) \equiv 0$$

If the relative degree of (2.26)-(2.27) is r in a neighborhood of x_0, then its zero dynamics is $n - r$ dimensional. If $n = r$, then the system has no zero dynamics.

Let the relative degree of the system (2.26)-(2.27) be $r < n$ at a neighborhood of $x_0 \in \chi$. Apply the following nonlinear coordinates transformation in the neighborhood of x_0:

$$z = T(x) = \begin{bmatrix} h(x) \\ L_f h(x) \\ \vdots \\ L_f^{r-1} h(x) \\ \varphi_1(x) \\ \vdots \\ \varphi_{n-r}(x) \end{bmatrix} \tag{2.28}$$

where $\varphi_i(x), \ i = 1, \ldots, n - r$ are chosen in such a way that they fulfill $L_g \varphi_i(x) = 0, \ i = 1, \ldots, n - r$ around x_0.

By applying the following input function

$$u = \alpha(x) + \beta(x) v, \quad \alpha = -\frac{L_f^{r-1} h(x)}{L_g L_f^{r-1} h(x)}, \ \beta = \frac{1}{L_g L_f^{r-1} h(x)}, \tag{2.29}$$

the nonlinear input affine system model in (2.26)-(2.27) can be rewritten in the new coordinates [49]:

$$\begin{aligned}
\dot{z}_1 &= z_2 \\
\dot{z}_2 &= z_3 \\
&\vdots \\
\dot{z}_r &= v \\
\dot{z}_{r+1} &= \Phi_1(z) \\
&\vdots \\
\dot{z}_n &= \Phi_{n-r}(z) \\
y &= z_1
\end{aligned}$$

where v is the new input variable. The transformed system is split into a linear r-dimensional subsystem (a cascade of integrators), and a nonlinear $(n-r)$ dimensional subsystem. Observe that the nonlinear subsystem - which is exactly the *zero dynamics* - is independent of the new input v and does not appear in y, therefore the system in the transformed coordinates is *linear in input-output sense*.

Since this zero dynamics does not depend on the control input, only the linear subsystem can be and should be controlled. This gives rise to the application of linear controllers. The great advantage of this input-output linearized form is that if the zero dynamics is asymptotically stable, then with arbitrary asymptotically stabilizing feedback (e.g. an LQ controller) the whole system becomes asymptotically stable [49].

Also note that if the relative degree of the system is n near $x_0 \in \chi$, the system (2.26)-(2.27) has no zero dynamics and is called *feedback linearizable* near x_0: it can be re-written in the new coordinates as a cascade of integrators.

Adaptive feedback linearization [71]

The adaptive feedback linearization method performs feedback linearization and handles model parameter uncertainties at the same time.

Consider a variant of the input affine single input-single output (SISO) nonlinear input affine state space model in (2.26)-(2.27):

$$\dot{x} = \frac{dx}{dt} = f(x) + \sum_{i=1}^{s} q_i(x)\mu_i + g(x)u \tag{2.30}$$

$$y = h(x) \tag{2.31}$$

where $\mu = [\mu_1, \ldots, \mu_s]^T \in \mathbb{R}^s$ is the vector of *unknown constant parameters*. Denote the *adaptive estimation* of the unknown parameter vector μ by $\hat{\mu} \in \mathbb{R}^s$.

The *problem statement* of the adaptive feedback linearization [71] is the following:

Given a reference model

$$\dot{z}_r = A_r z_r + B_r v_r \ , \quad z_r \in \mathbb{R}^n, \quad A_r = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ -k_1 & -k_2 & -k_3 & \dots & -k_n \end{bmatrix} \tag{2.32}$$

where $B_r \in \mathbb{R}^{n\times 1}$, $A_r \in \mathbb{R}^{n\times n}$, (2.32) is asymptotically stable and reachable. Find an *adaptive feedback linearizing control* for (2.30)-(2.31) which is a dynamic state feedback

$$\frac{d}{dt}\hat{\mu} = \vartheta(x, z_r, v_r, \hat{\mu}) \ , \quad \hat{\mu} \in \mathbb{R}^s, \hat{\mu}(0) = \hat{\mu}_0 \tag{2.33}$$

$$u = u(x, z_r, v_r, \hat{\mu}) \tag{2.34}$$

that for any initial conditions $x(0), \hat{\mu}(0), z_r(0)$, any unknown parameter μ and any bounded $v_r(t)$ fulfills the following requirements:

1. $||x(t)||$ and $||\hat{\mu}(t)||$ are bounded for $t \geq 0$, where $||\cdot||$ is a suitable vector norm;
2. there exist a filtered transformation (i.e. a time-varying coordinates transformation) in the form $x = T(z, z_r(t), \hat{\mu}(t))$ such that

$$lim_{t\to\infty}||x(t) - T(z, z_r(t), \hat{\mu}(t))|| = 0$$

Denote Δ_k the distribution in (2.6) (this is the result of the k-th step of the algorithm for computing the reachability distribution of input-affine systems). The Adaptive Feedback Linearization Theorem [71] states that if

1. the nominal system of (2.30)-(2.31) (i.e. the system with $\mu = 0$) is globally feedback linearizable,
2. the global triangularity conditions $[\mu_i, \delta_j^{(k)}] \subset \Delta_k$, are satisfied, where $i = 1, \dots, s$, $k = 0, \dots, n-2$, and $\Delta_k = span\{\delta_j^{(k)}, j = 1, \dots, d\}$,

then there exists an adaptive feedback linearizing control.

The construction of this adaptive feedback law is not presented here, since it depends on the order n of the system, but will be described in detail on the adaptive controller synthesis applied to the low-power gas turbine in Section 4.5.

However, the assumption that μ is constant can be relaxed, if its time behavior can be modelled by an exosystem with unknown initial condition $\mu(0)$ [71]:

$$\frac{d}{dt}\mu = \Omega(t, x)\mu + \omega(t, x) \tag{2.35}$$

with the property that

1. $\Omega(t, x) + \Omega^T(t, x)$ is negative semidefinite, $t \geq 0$, $\forall x \in \mathbb{R}^n$,
2. the exosystem (2.35) has bounded state trajectories $\mu(t),\ t \geq 0$ if $x(t),\ t \geq 0$ is from a bounded set of $\mathbb{R}^n$.

Linear matrix inequalities

Linear matrix inequalities play an important role in the control (an also in the analysis) of dynamical systems. A (non-strict) linear matrix inequality (LMI) is an inequality of the form [89]:

$$F(x) = F_0 + \sum_{i=1}^{m} x_i F_i \geq 0, \tag{2.36}$$

where $x \in \mathbb{R}^m$ is the variable and $F_i \in \mathbb{R}^{n \times n}$, $i = 0, \ldots, m$ are given symmetric matrices. The inequality symbol in (2.36) stands for the positive semi-definiteness of $F(x)$. LMIs form a convex constraint on the variables i.e. the set

$$\{x \mid F(x) \geq 0\} \tag{2.37}$$

is convex. A wide variety of different problems (linear and convex quadratic inequalities, matrix norm inequalities, convex constraints etc.) can be written as LMIs and there are computationally stable and effective (polynomial time) algorithms for their solution [21], [89].

2.2 State equations in the form of differential algebraic equations (DAEs)

This section concerns with DAE models which are the general representation forms of lumped parameter systems. The stability and reachability of these DAE models are also discussed. A short literature review on the control techniques applied for DAEs can also be found at the end of this section.

Lumped parameter systems are usually modelled by a mixed set of (ordinary) differential and algebraic equations (DAEs). The differential equations describe the dynamics of conserved quantities, while the algebraic equations define static relationships and complete the model.

For example, lumped parameter process systems are intuitively modelled by DAEs, following a systematic modelling procedure based on first engineering principles [46]. The differential equations are balance equations of masses and energies, while the algebraic equations are thermodynamic laws, empirical correlations etc.

Modelling with DAE system models is generally used e.g. for electric circuits, interconnected systems, constrained dynamics etc. DAE system models are also called as singular, semi-state, descriptor or generalized systems, and the notion of singularly perturbed system is also related with DAEs [59].

An equivalent ODE model can be obtained from a DAE model, if - with a symbolic Gauss elimination - the algebraic variables can be expressed from the algebraic equations as functions of the differential variables, since then these equations can be eliminated by substituting them to the differential equations. However, for some DAE models the elimination of the algebraic equations is not possible because of the structure of the algebraic equations (e.g. strong nonlinearities, high number of variables). This gives rise to the (possibly iterative) time-differentiation of the algebraic equations, in order to get a purely differential (ODE) model. Unfortunately,

because of the increased number of differential variables the resulted ODE will be non-minimal, making the dynamical analysis much more difficult, moreover excluding the possibility of the application of standard nonlinear controller design methods. This problem brings up the need to somehow extend the nonlinear dynamic analysis and control methods to DAE system models.

2.2.1 The general DAE representation of lumped parameter system models

As the first attempt, LTI DAEs have been invented in the literature called as "generalized linear systems" or "descriptor systems" (see e.g. [98]):

$$E\dot{x} = Ax + Bu\ , \qquad x(0) = x_0 \tag{2.38}$$

$$y = Cx \tag{2.39}$$

where x, u, y denote the state, input and output vectors of dimensions n,p and q, respectively, while E, A, B, C are matrices of appropriate dimensions and E is a *singular* (non-invertible) matrix (without this condition the model above becomes an LTI ODE model in the form (2.1)-(2.2) by pre-multiplying both sides of (2.38) by E^{-1}). With a decomposition, which is a linear transformation on the state variable vector x, (2.38)-(2.39) can be split to a set of n_1 differential equations and a set of n_2 algebraic equations, where $n_1 + n_2 = n$ [59]. The transformed state variable vector is also split to n_1 differential and n_2 algebraic variables.

However, for the representation of lumped parameter systems the general representation form has to cope with nonlinearities, and also with cases when the algebraic variables cannot be expressed explicitly from the algebraic equations. The following, so-called semi-explicit DAE can fulfill these requirements and therefore it can be regarded as the general representation form of lumped parameter systems:

$$\dot{x} = f(x, z) + \sum_{i=1}^{p} g_i(x, z)u_i \tag{2.40}$$

$$0 = w(x, z) \tag{2.41}$$

$$y = h(x, z) \tag{2.42}$$

where $x \in \mathbb{R}^{n_1}$ and $z \in \mathbb{R}^{n_2}$ are the vectors of the differential and the algebraic variables, respectively, $f, g_i \in \mathbb{R}^{n_1+n_2} \mapsto \mathbb{R}^{n_1}$, $i = 1, \ldots, p$, $w \in \mathbb{R}^{n_1+n_2} \mapsto \mathbb{R}^{n_2}$ and $h \in \mathbb{R}^{n_1+n_2} \mapsto \mathbb{R}^{q}$ are smooth nonlinear vector fields, and $u = [u_1, \ldots, u_p]^T$ is the vector of control input variables. The *state variables* of this DAE consists of *the vector of differential and the vector algebraic variables*. This model is nonlinear and input-affine. The algebraic equations are in implicit form and do not contain control input variables. Moreover, we assume that the initial conditions of the model $(x(t_0), z(t_0))$ fulfill the algebraic equations to avoid impulsive solutions. (This assumption could not be done in case of non-smooth functions f, w and $g_i, i = 1, \ldots, p$, e.g. in modelling electric circuits with switches [98].)

2.2.2 The differential index as the measure of complexity

The *differential index* of a DAE model is a natural number showing the minimal number of time differentiations that have to be performed on the algebraic equations to obtain a purely differential (ODE) model [59].

ODEs can be regarded as special DAEs with differential indices that equal to zero. DAEs containing an algebraic part that from the algebraic variables can expressed explicitly and therefore the algebraic part of the model can be eliminated by substitution to the differential equations have differential indices equal to one. Note that index-1 models have no substitutable algebraic equations in general (there are cases when z cannot be expressed in terms of x because of the nonlinearities and variable cross-connections in the algebraic equations, but a single time-differentiation leads to an ODE state space description), and it is not emphasized in the literature. Systems with indices at least two are called high-index models. As interpreted in [42], the index is "a measure for the distance between the DAE and an ODE".

2.2.3 Stability of DAEs

The constructive method of finding a Lyapunov function for LTI ODE system models is extended to LTI DAEs, as well as the eigenvalue checking technique for proving asymptotic stability or instability of (2.38)-(2.39) is analogous with the LTI ODE case, with the only difference that the "eigenvalues" are the solutions of $det(\lambda E - A) = 0$ [75].

For autonomous nonlinear semi-explicit DAEs (i.e. controlled DAEs in the form (2.40)-(2.42) with no input parts:

$$\dot{x} = f(x,z) \tag{2.43}$$
$$0 = w(x,z) \tag{2.44}$$
$$y = h(x,z) \tag{2.45}$$

there is an analogous extension of local stability analysis by eigenvalue checking, however linearization is not always possible even for index-1 DAEs [84].

Lyapunov technique is discussed in [15] for semi-explicit autonomous DAEs in the form of (2.43)-(2.45) in the index-1 case. Let $[x^* \ z^*]^T \in \mathbb{R}^{n_1+n_2}$ be an equilibrium of (2.43)-(2.45), i.e. $f(x^*, z^*) = w(x^*, z^*) = 0$. If there is a smooth real-valued function V defined in a neighborhood $\mathcal{B} \in \mathbb{R}^{n_1+n_2}$ of the equilibrium point $[x^* \ z^*]^T$ such that

1. $V : \mathcal{B} \mapsto \mathbb{R}$
2. $V(x,z) \geq 0, \forall (x,z) \in \mathcal{B}, V(x^*, z^*) = 0$ (i.e. V is positive semidefinite at $[x^* \ z^*]^T$)
3. $\frac{dV}{dt} = \frac{\partial V}{\partial x} f(x) - \frac{\partial V}{\partial y}\left[\frac{\partial w}{\partial y}\right]^{-1}\left[\frac{\partial w}{\partial x}\right] f(x) \leq 0 \quad \forall (x,z) \in \mathcal{B}$

then the equilibrium is locally *stable* (in Lyapunov sense). If V is positive definite at the equilibrium (i.e. $\geq$ is changed to $>$ in the second condition), then the equilibrium is locally *asymptotically stable*. Note that in the third condition, the Implicit Function Theorem has been used to compute the time-derivative of V.

A considerable result of this work is that algebraic variables are not at all negligible in stability analysis, they are present in the Lyapunov function.

2.2.4 Reachability of DAEs

The reachability of LTI DAEs is carefully discussed in [104], showing that at least two different types have to be defined for handling this question in a precise manner:

1. C-reachability (complete reachability) is defined as any state can be reached from any initial state (recall that the states of the system are the differential and also the algebraic variables).

2. R-reachability means that the system is reachable within the set where $w(x, z) = 0$ is fulfilled. This notion means "reachability in x", since z is an implicit function of x.

It is shown that an LTI DAE (2.38)-(2.39) is R-reachable if and only if the hypermatrix $[sE - A \mid B]$ is of full rank, for all $s \in \mathbb{C}$ [104].

However, for the precise mathematical investigation (as in [49]) of the reachability of nonlinear DAEs no attempts have been taken yet. An important property of DAE systems is that their state trajectories $(x(t), z(t))$ evolve on a manifold which is characterized by their algebraic equations. Thus, the state trajectories of the model (2.40)-(2.42) evolve on a lower dimensional manifold of the $(n_1 + n_2)$-dimensional state space, determined by the set of algebraic constraints $w(x, z) = 0$. From these facts it comes that DAE systems in the form of (2.40)-(2.42) are *not C-reachable*.

For a deeper understanding, consider the case when the algebraic variables z of (2.40)-(2.41) can be explicitly given as a function of the differential ones: $z = w^*(x)$. In this case, z can be considered as a state feedback that can be applied to (2.40) giving an n_1-dimensional representation that only depend on x and $u_i, i = 1, \ldots, p$. We know from control theory that a static state feedback cannot increase the dimension of the state space [49], therefore the (n_1+n_2)-dimensional DAE model (2.40)-(2.42) cannot be reachable.

2.2.5 Control of DAE systems

A lot of papers concern with the control of LTI DAEs, for example the application of pole placement [28], linear quadratic control [51] and robust control [83] has been reported.

Control approaches for nonlinear DAEs are mostly numerical in the literature, using nonlinear programming for optimization, for example in [32]. However, there are attempts to apply the methods and tools of nonlinear control design, e.g. input-output and feedback linearization [59], [103], however these controllers are designed for DAE models that are also affine in their algebraic variables - therefore the nonlinearities are functions of the differential variables only -, moreover the algebraic equations also contain affine input terms.

2.3 Quasi-polynomial (QP) models

In this section special nonlinear ODE models, the class of QP models will be introduced. The main objective is to show the advantageous properties of QP models, and then alloy QP models with DAEs. First, the representability of arbitrary smooth ODEs in QP form by variable embedding is shown. Then, the non-minimality of QP models and their equivalence with DAE models are discussed. After that, a general form for the representation of arbitrary lumped parameter systems is proposed, which is a DAE model in QP form. At the end of this section, a local quadratic stability analysis technique is shown for a special, non-minimal (i.e. DAE-equivalent) class of QP models, namely the Lotka-Volterra models.

The class of quasi-polynomial (QP) system models have an emerging importance in nonlinear systems and control theory, since nonlinear input-affine state space models with smooth nonlinearities can be algorithmically transformed to QP form with a variable embedding procedure (see [48] or Section 2.3.3), moreover recent research indicated that their structure can be exploited well both in dynamical analysis and controller design [69],[93].

The term "quasi-polynomial" in itself denotes a real-valued polynomial with possibly negative and/or fractional powers. For example, a quasi-polynomial of n variables $x_1, \ldots, x_n$ can be characterized as

$$QP(x_1, \ldots, x_n) = \sum_{j=1}^{m} \alpha_j \prod_{j=1}^{m} x^{\beta_{ji}} \tag{2.46}$$

with the coefficients $\alpha_j \in \mathbb{R},\ j = 1, \ldots, m$ and the powers $\beta_{ij} \in \mathbb{R},\ j = 1, \ldots, m,\ i = 1, \ldots, n$.

2.3.1 QP models

A wide class of autonomous ODE models (i.e. (2.3)-(2.4) with $u = 0$ or with $u = \phi(x)$) can be written in the so-called quasi-polynomial (QP) form (i.e. the RHS of state equation consists of quasi-polynomials) [48]:

$$\dot{x_i} = x_i(\lambda_i + \sum_{j=1}^{m} A_{ij} U_j), \quad , i = 1, \ldots, n, \quad m \geq n \tag{2.47}$$

$$y = h(x) \tag{2.48}$$

where the terms

$$U_j = \prod_{k=1}^{n} x_k^{B_{jk}}, \quad j = 1, \ldots, m \tag{2.49}$$

are the so-called *quasi-monomials* (monomials, QMs) of the system, $A \in \mathbb{R}^{n \times m}$ and $B \in \mathbb{R}^{m \times n}$ are constant matrices, $\lambda \in \mathbb{R}^n$ is a constant vector. Note that our variables have concrete physical meaning, i.e. $x_i > 0, \quad i = 1, \ldots, n$. The set of quasi-monomials is denoted by Q:

$$Q = \{\ U_j \mid j = 1, \ldots, m\} \tag{2.50}$$

The QP-ODE in (2.47) is called *inhomogeneous* if it contains linear terms (i.e. $\lambda \neq 0 \in \mathbb{R}^n$). If $\lambda = 0 \in \mathbb{R}^n$, (2.47) can be written in the following *homogeneous* form:

$$\dot{x}_i = x_i \left(\sum_{j=1}^{m} A_{i,j} U_j \right) \ , \quad U_j = \prod_{k=1}^{n} x_k^{B_{j,k}}, \qquad i = 1, \ldots, n \tag{2.51}$$

with its parameter matrices A, B and QMs as defined above.

QP models with $\lambda \neq 0$ can easily be transformed into the form of (2.51) in the following way. Consider the inhomogeneous representation

$$\dot{x}_i = x_i \left(\lambda_i + \sum_{j=1}^{\overline{m}} \overline{A}_{i,j} \overline{U}_j \right) \ , \quad \overline{U}_j = \prod_{k=1}^{n} x_k^{\overline{B}_{j,k}}, \qquad i = 1, \ldots, n \tag{2.52}$$

where A and B are constant matrices of dimensions $n \times \overline{m}$ and $\overline{m} \times n$, respectively, $\lambda \in \mathbb{R}^n$, and $\overline{U}_j,\ j = 1, \ldots, \overline{m}$ are the QMs of the system.

By introducing the unit monomial $U_1 = 1$, (2.52) can be re-written in the form of (2.51), with the following parameter matrices and quasi-monomials:

$$A = \left[\lambda | \overline{A}\right] \in \mathbb{R}^{n \times m} \ , \quad B = \left[\frac{0_{1 \times n}}{\overline{B}} \right] \in \mathbb{R}^{m \times n}$$

$$U_j = 1 \ , \qquad U_j = \overline{U}_{j-1} \ , \quad j = 2, \ldots, m$$

with $0_{1 \times n}$ denoting a null-matrix of size $1 \times n$, and $m = \overline{m} + 1$.

2.3.2 Quasi-monomial transformations

The set of QP system models is closed under special nonlinear state transformations called quasi-monomial transformations (QMTs) [48]. A QMT of the n-dimensional inhomogeneous QP-ODE in (2.47), and also of the n-dimensional homogeneous QP-ODE in (2.51) can be defined as

$$x_i = \prod_{k=1}^{n} \hat{x}_k^{C_{ik}} \ , \quad i = 1, \ldots, n \tag{2.53}$$

where $C \in \mathbb{R}^{n \times n}$ is an arbitrary invertible matrix. The transformed set of equations will also be in QP form with matrices $\hat{B} = BC$ and $\hat{A} = C^{-1}A$. Moreover, for the inhomogeneous model in (2.47), $\hat{\lambda} = C^{-1}\lambda$ also holds.

Another important property of (both homogeneous and inhomogeneous) QP system models that the matrix product $M = B \cdot A$ remains unchanged under QMTs.

2.3.3 Embedding into QP form

Arbitrary input-affine state space models containing non-QP terms can be reformulated by variable embedding to a QP form [47]. Consider the truncated form of the nonlinear state equation (2.3):

$$\dot{x} = f(x), \quad x = [x_1, \ldots, x_n]^T \tag{2.54}$$

This nonlinear ODE can be embedded into QP form if it satisfies two important requirements [47]:

1. The set of nonlinear ODEs should be in the form:

$$\dot{x}_i = QP_i(x, w(x)) \ , \quad i = 1, \ldots, n \tag{2.55}$$

where $QP_i,\ i = 1, \ldots, n$ are arbitrary QP-type functions of x and w, where w is a real-valued non-QP function of x,

2. The partial derivatives of w by $x_i,\ i = 1, \ldots, n$ are QP-type functions of x and w:

$$\frac{dw}{dx_i} = QP_i^{pder}(x, w(x)) \ , \quad i = 1, \ldots, n \tag{2.56}$$

where $QP_i^{pder}\ , i = 1, \ldots, n$ are quasi-polynomials of variables x, w.

The embedding is performed by introducing a *new auxiliary variable*

$$y = w^q(x) \prod_{i=1}^{n} x_i^{p_i}, \qquad q \neq 0. \tag{2.57}$$

For the sake of simplicity, choose the case with $q = 1$ and $p_i = 0,\ i = 1, \ldots, n$:

$$y = w(x) \tag{2.58}$$

With this new variable, (2.55) can be re-written as:

$$\dot{x}_i = QP_i(x, y) = x_i\Big(x_i^{-1} QP_i(x, y)\Big) \ , \quad i = 1, \ldots, n \tag{2.59}$$

Since the product of quasi-polynomials is also a quasi-polynomial, the RHS of (2.59) is a quasi-polynomial function of x and y for all i, $i = 1, \ldots, n$.

Take the time derivative of y:

$$\begin{aligned} \dot{y} &= \sum_{i=1}^{n} \frac{\partial w}{\partial x_i} \dot{x}_i = \sum_{i=1}^{n} QP_i^{pder}(x, y) QP_i(x, y) = \\ &= y\Big(y^{-1} \sum_{i=1}^{n} QP_i^{pder}(x, y) QP_i(x, y)\Big) \end{aligned} \tag{2.60}$$

Since the sums and products of quasi-polynomials are also quasi-polynomials, the RHS of (2.60) is a quasi-polynomial of x and y.

The result (2.59)-(2.60) is an n dimensional QP-ODE in the form (2.47). Note that this set of differential equations is *non-minimal* (i.e. can be described with a lower number of variables), because it has a first integral (given by the equation (2.58)). Also note that the embedding is not unique, since we can choose the parameters p_i, $i = 1, \ldots, n$ and q in (2.57) arbitrarily.

2.3.4 The QP-ODE form of lumped parameter system models

As a consequence of the previous section, nonlinear input-affine models (2.3)-(2.4) of lumped parameter systems can be recast by possible variable embedding(s) into the following general QP form:

$$\dot{x}_i = x_i\Big(\sum_{j=1}^{m} A_{0_{i,j}} U_j + \sum_{k=1}^{p}\sum_{j=1}^{m} A_{k_{i,j}} U_j u_k\Big) , \quad i = 1, \ldots, n \tag{2.61}$$

$$y_i = Y_{i,j} U_j , \quad i = 1, \ldots, q \tag{2.62}$$

where $U_j = \prod_{i=1}^{n} x_k^{B_{j,i}}$, $j = 1, \ldots, m$ are the quasi-monomials of the system (containing the unit monomial if necessary), $x \in \mathbb{R}^n$ is the vector of state variables (also containing the embedded ones), $u \in \mathbb{R}^p$ and $y \in \mathbb{R}^q$ is the vector of input and output variables, respectively, $A_k \in \mathbb{R}^{n \times m}$, $k = 0, \ldots, p$, $B \in \mathbb{R}^{m \times n}$ and $Y \in \mathbb{R}^{q \times m}$ are constant matrices. This kind of representation is advantageous in the sense that the QP structure can be exploited, however the dimension is increased by the number of embedded variables.

2.3.5 Zero dynamics of QP-ODE models

In this section, the zero dynamics of SISO QP models is considered, trying to find a condition which guarantees that the zero dynamics is also in QP from.

Let (2.61)-(2.62) be a SISO model (i.e. $r = q = 1$), with a relative degree r that is constant in a neighborhood of x_0. Denote the state, input and output functions of this SISO model by f, g and h, respectively. The zeroing input (that is necessary for the determination of the zero dynamics) can be computed from the r-th time derivative of y:

$$y^{(n)} = L_f^r h(x) + L_g L_f^{r-1} h(x) u \equiv 0$$

Since the set of quasi-polynomials is closed under differentiation, addition and multiplication, the equation above is of QP type. However, the *zeroing input*

$$u = -\frac{L_f^r h(x)}{L_g L_f^{r-1} h(x)}$$

- and therefore the zero dynamics - is non-QP in general, unless the denominator is a quasi-monomial.

Consider now a special SISO class of (2.61)-(2.62) where the input function $g(x) = b = [b_1 \ldots b_n]^T$ is a constant vector, and the output function $h(x)$ is an affine function of the state variables:

$$\dot{x}_i = x_i\Big(\sum_{j=1}^{m} A_{ij} \prod_{k=1}^{n} x_k^{B_{jk}}\Big) + b_i u = x_i \cdot f_i^{QP}(x_1, \ldots, x_n) + b_i u \tag{2.63}$$

$$i = 1, \ldots, n, \quad m \geq n$$

$$y = C(x - x^*) = \sum_{i=1}^{n} c_i (x_i - x_i^*) \tag{2.64}$$

where x^* contains the steady state values of x.

Consider the constrained output

$$0 \equiv C(x - x_i^*) = \sum_{i=1}^{n} c_i(x_i - x_i^*) \tag{2.65}$$

Take the time derivative of (2.65):

$$0 = L_f h(x) + L_g h(x) u \tag{2.66}$$

where

$$L_f h(x) = \frac{\partial h}{\partial x} f(x) = \sum_{i=1}^{n} c_i f_i(x) \ , \quad L_g h(x) = \frac{\partial h}{\partial x} g(x) = \sum_{i=1}^{n} c_i b_i$$

where f_i denotes the i-th coordinate function of f.

Observe that the relative degree of (2.63)-(2.64) is *globally equal to* 1 independently of x, if and only if the condition on the system parameters

$$\sum_{i=1}^{n} c_i b_i \neq 0 \tag{2.67}$$

is fulfilled. In this case, the zeroing input can be expressed from (2.66). Applying this zeroing input to (2.63) gives the zero dynamics of (2.63)-(2.64) which is an autonomous QP-ODE:

$$\dot{x_i} = x_i \sum_{j=1}^{m} \left(K_i A_{ij} \prod_{k=1}^{n} x_k^{B_{jk}} - \sum_{l=1,\ l \neq i}^{n} \sum_{j=1}^{m} \frac{b_i}{K} c_l A_{lj} \frac{x_l}{x_i} \prod_{k=1}^{n} x_k^{B_{jk}} \right) \ , \quad i = 1, \ldots, n \tag{2.68}$$

where $K = \sum_{i=1}^{n} c_i b_i \ , \quad K_i = 1 - \frac{c_i b_i}{K} \ , \quad i = 1, \ldots, n.$

As we can see, the zero dynamics of (2.63)-(2.64) is also in QP form, however the set of quasi-monomials is different. Note that, however, the relative degree is a local property in general, but because of the LTI input and affine output terms, the $r = 1$ case is a global property of this model class if the condition (2.67) is fulfilled.

2.3.6 Invariants of QP models

The invariant(s) of a QP-ODE (if they exist) are not necessarily quasi-polynomials of its state variables. For example,

$$I = y - w(x) \tag{2.69}$$

is a first integral of the QP-ODE described in (2.59)-(2.60). Since $w(x)$ is an arbitrary non-QP function, I is a non-QP invariant of this QP-ODE.

However, the theoretical background of the existence of *quasi-polynomial* invariants is well-founded. In [33] algebraic tools are applied to find semi-invariants and invariants in quasi-polynomial systems. In [34] it is shown that the existence of polynomial-type semi-invariants in the corresponding Lotka-Volterra systems is

a necessary condition for the existence of quasi-polynomial invariants. A general method is given in the same paper for the symbolical checking of this necessary condition with numerous valuable examples. Moreover, a computer-algebraic software package called QPSI has been implemented for the determination of quasi-polynomial invariants and the corresponding model parameter relations [35]. On the other hand, the QPSI approach also have a serious drawback: as a consequence of the symbolical treatment, the QPSI method have difficulties in finding invariants of QP systems if the number of quasi-monomials is high (more than four).

2.3.7 Non-minimal autonomous QP models as DAE models

It has been shown that an ODE with algebraic constraint(s) on its state variables is non-minimal (See Section 2.1.4), moreover it is equivalent with a DAE. This is now discussed for non-minimal QP-ODEs.

Consider the n-th order QP-ODE defined by (2.51):

$$\dot{x}_i = x_i \left(\sum_{j=1}^{m} A_{i,j} U_j \right) \quad , \quad U_j = \prod_{k=1}^{n} x_k^{B_{j,k}}, \qquad i = 1, \ldots, n$$

Let there be k linearly independent algebraic constraints between its variables $x_1, \ldots, x_n$:

$$I_i = F_i(x_1, \ldots, x_n) = constant_i \quad , \qquad i = 1, \ldots, k$$

These algebraic constraints are first integrals of the homogeneous QP-ODE above, since their time-derivative is equal to zero. Thus, the QP-ODE above can be re-written in the following DAE form:

$$\begin{aligned} \dot{x}_i &= x_i \left(\sum_{j=1}^{m} A_{i,j} U_j \right) \quad , \quad U_j = \prod_{k=1}^{n} x_k^{B_{j,k}}, \qquad i = 1, \ldots, n-k \\ I_i &= F_i(x_1, \ldots, x_n) = constant_i \quad , \qquad i = 1, \ldots, k \end{aligned}$$

Note that this DAE is generally *not in QP form*, since its invariants are not necessarily quasi-polynomials. It is possible that none of the invariants $I_1, \ldots, I_k$ are of QP type. In this case, we call (2.51) with the invariants $I_1, \ldots, I_k$ *minimal in QP sense*, because it has no QP type first integrals. Note that ODE models that are minimal in QP sense are generally non-minimal models (i.e. they may have non-QP first integrals).

On the other hand, it is also possible that all k invariants of the ODE model above are of QP type, and therefore both the differential and the algebraic equations are in QP form. From now on, these models are called as QP-DAE system models.

2.3.8 The QP-DAE form of controlled lumped parameter system models

This section offers a general representation form for lumped parameter systems by alloying the structure of QP and DAE models.

Lumped parameter systems are naturally modelled by DAEs, as it has been discussed in Section 2.2. The *truncated or controlled* semi-explicit form of QP-DAEs (i.e. the DAE model (2.40)-(2.41) with $u_i = 0,\ i = 1, \ldots, p$) can be transformed to a QP-DAE model by substituting as much algebraic equations to the differential ones as possible, and simultaneously embedding all the non-QP nonlinearities of the DAE model as described in Section 2.3.3. The result is a system of equations consisting of $n - k$ differential, and k algebraic equations; both of them of QP type:

$$\dot{x}_i = x_i \left(\sum_{j=1}^{m} \overline{A}_{i,j} U_j \right) , \quad U_j = \prod_{k=1}^{n} x_k^{B_{j,k}}, \qquad i = 1, \ldots, n-k \tag{2.70}$$

$$0 = \sum_{j=1}^{m} \overline{A}_{i,j} U_j , \qquad i = n-k+1, \ldots, n \tag{2.71}$$

The result is a semi-explicit QP-DAE - the algebraic equations are in implicit form.

In some fortunate cases, this semi-explicit QP-DAE above can be transformed to the following *explicit* form:

$$\dot{x}_i = x_i \left(\sum_{j=1}^{m} \overline{A}_{i,j} U_j \right) , \quad U_j = \prod_{k=1}^{n} x_k^{B_{j,k}}, \qquad i = 1, \ldots, n-k \tag{2.72}$$

$$x_{k+1}^{c_1} = QP_1(x_1, \ldots, x_k) \tag{2.73}$$

$$x_{k+2}^{c_2} = QP_2(x_1, \ldots, x_k, x_{k+1}) \tag{2.74}$$

$$\vdots$$

$$x_n^{c_k} = QP_k(x_1, \ldots, x_k, x_{k+1}, \ldots, x_{n-1}) \tag{2.75}$$

where $QP_1, \ldots, QP_k$ are quasi-polynomial functions, while $c_1, \ldots, c_k$ are real numbers. The algebraic equations of the model above can be substituted, resulting an n-th order ODE. It is important to note that this substitution may destroy the QP structure, since the resulted ODE may contain fractional or negative powers of quasi-polynomials.

2.3.9 Lotka-Volterra (LV) form of QP models

In the following, the LV form of QP-ODEs is presented, which is an advantageous but generally non-minimal type of representation. The equivalence of the class of non-minimal LV models with a class of DAE models is also presented here.

QP system models defined by (2.47) can be represented in their so-called Lotka-Volterra form, where the state variables are exactly the quasi-monomials:

$$\dot{U}_\ell = U_\ell \Big(\lambda_{LV_\ell} + \sum_{j=1}^{m} A_{LV_{\ell,j}} U_j \Big), \qquad \ell = 1, \ldots, m \tag{2.76}$$

where $\lambda_{LV} = B \cdot \lambda \in \mathbb{R}^m$, $A_{LV} = B \cdot A \in \mathbb{R}^{m \times m}$. Although LV models are one of the simplest nonlinear autonomous state space representations, LV models computed from QP models are generally non-minimal (when $m > n$).

We also have to note that since all QMs depend on n variables, but the number of QMs are $m \geq n$, we can construct a basis of n quasi-monomials from which all other $m-n$ quasi-monomials can be obtained. This indicates $m-n$ algebraic constraints between the QMs, which are of quasi-monomial type (see [48] or pages 22-23. of [P8] for a more verbose explanation and a numerical example). Since the state variables of the Lotka-Volterra model are its QMs, it has additional $m-n$ algebraic constraints between its state variables, and therefore the system trajectories can evolve on an n dimensional manifold (surface) $\mathcal{M}$ of $\mathbb{R}^m$.

This indicates the fact that LV models are hidden DAEs, and can be represented in the following QP-DAE form:

$$\dot{U}_\ell = U_\ell\Big(\lambda_{LV_\ell} + \sum_{j=1}^{m} A_{LV_{\ell,j}} U_j\Big), \qquad \ell = 1, \ldots, n \tag{2.77}$$

$$U_{n+k} = QM_k(U_1, \ldots, U_n), \qquad k = 1, \ldots, m-n \tag{2.78}$$

where $QM_k,\ k = 1, \ldots, m-n$ are quasi-monomial functions of the quasi-monomials $U_1, \ldots, U_n$

2.3.10 Local quadratic stability of LV systems

Local stability of LV models is investigated in this section with quadratic Lyapunov functions [68]. At the end of this section, a method for non-minimal LV system models will be established.

Let U be the vector of LV variables by $U = [U_1, \ldots, U_m]^T$. Perform a coordinates shift on the LV-equations, i.e. $\overline{U} = U - U^*$, where U^* is a vector containing the operating point values of the LV variables, and therefore $\overline{U}$ is the centered LV variable vector. Then the LV-equations in (2.76) in the transformed coordinates have the form

$$\dot{\overline{U}} = \langle U \rangle \cdot A_{LV} \cdot \overline{U} = \langle \overline{U} + U^* \rangle \cdot A_{LV} \cdot \overline{U} \tag{2.79}$$

where $\langle U \rangle$ denotes the diagonal quadratic matrix built from the vector U:

$$\langle U \rangle_{i,i} = U_i,\ \langle U \rangle_{i,j} = 0,\ i, j = 1, \ldots, m,\ i \neq j$$

moreover $\langle \overline{U} \rangle$ and $\langle U^* \rangle$ are built similarly from $\overline{U}$ and U^*, respectively, where the equilibrium $U^* = U^*(\lambda_1, \ldots, \lambda_m)$ is the solution of the equation

$$\lambda_\ell = -\sum_{j=1}^{m} A_{LV_{\ell,j}} U_j^*, \quad \ell = 1, \ldots, m \tag{2.80}$$

Let the quadratic Lyapunov function candidate be given in the following form :

$$V(\overline{U}) = \overline{U}^T P \overline{U} \tag{2.81}$$

where P is a positive definite symmetric matrix of size $m \times m$. The time derivative of V is given by

$$\begin{aligned} \dot{V} &= \overline{U}^T P \dot{\overline{U}} + \dot{\overline{U}}^T P \overline{U} = \\ &= \overline{U}^T P (\langle \overline{U} \rangle + \langle U^* \rangle) A_{LV} \overline{U} + \overline{U}^T A_{LV}^T (\langle \overline{U} \rangle + \langle U^* \rangle) P \overline{U} = \\ &= \overline{U}^T \left\{ P \langle \overline{U} \rangle A_{LV} + P \langle U^* \rangle A_{LV} + A_{LV}^T \langle \overline{U} \rangle P + A_{LV}^T \langle U^* \rangle P \right\} \overline{U} \end{aligned}$$

The non-increasing nature of the quadratic Lyapunov function in a neighborhood $\mathcal{N}$ of the origin is equivalent with the validity of the following matrix inequality, which is *bilinear* in its variables P and $\langle\overline{U}\rangle$ [68]:

$$P\langle\overline{U}\rangle A_{LV} + P\langle U^*\rangle A_{LV} + A_{LV}^T\langle\overline{U}\rangle P + A_{LV}^T\langle U^*\rangle P \leq 0 \tag{2.82}$$

where $[\overline{U}_1, \ldots, \overline{U}_m]^T \in \mathcal{N}$.

Therefore the quadratic stability region can be estimated by first solving

$$P\langle U^*\rangle A_{LV} + A_{LV}^T\langle U^*\rangle P \leq 0 \tag{2.83}$$

for P, and then (with this P) solving

$$P\langle\overline{U}\rangle A_{LV} + P\langle U^*\rangle A_{LV} + A_{LV}^T\langle\overline{U}\rangle P + A_{LV}^T\langle U^*\rangle P \leq 0 \tag{2.84}$$

for $\langle\overline{U}\rangle$, which defines a *convex* neighborhood of the origin.

Since the quasi-monomials depend on $n = dim(x)$ variables as defined in (2.49), the set of independent quasi-monomials U_{indep} contains n elements, and the remainder $m-n$ quasi-monomials can be determined from the elements of U_{indep}. It means that we only have to determine n values in $\langle\overline{U}\rangle$ and then compute the other $m-n$ values therefrom.

Extension to non-minimal LV system models

The transformation of a QP model into LV form leads to a considerable increase of the state space dimension, since the number of the quasi-monomials is much higher than that of the QP variables in most cases. As a consequence, the LV system model is non-minimal, constrained with $m-n$ algebraic equations. Thus, the presented method has to be extended for non-minimal LV system models. For this purpose a theorem in [88] can be applied:

Let U^* be an equilibrium point of $\dot{U} = f(U)$, and let $V : \mathcal{U} \to \mathbb{R}$ a C^1 function which is positive semidefinite at U^*, that is

$$V(U^*) = 0, V(U) \geq 0. \tag{2.85}$$

Furthermore, suppose that $\dot{V}(U) \leq 0$ for all $U \in \mathcal{U}$. Let K be the largest positively invariant set contained in $\{x|V(x) = 0\}$. If U^* is asymptotically stable conditionally to K, then U^* is a stable equilibrium of $\dot{U} = f(U)$.

Let us consider the case when the eigenvalues of the matrix $J_{LV} = \langle U^*\rangle \cdot A_{LV}$ correspond to Jordan blocks of size 1. Then we can find a similarity transformation T such that

$$\widetilde{J}_{LV} = T^{-1}J_{LV}T = diag(J_{s_{n\times n}}, 0_{(m-n)\times(m-n)}) \tag{2.86}$$

where $J_{s_{n\times n}}$ and $0_{(m-n)\times(m-n)}$ are quadratic matrices, $n = rank(J_{LV})$. If the eigenvalues of J_s have strictly negative real parts, then the Lyapunov inequality

$$J_s^T P_{stab} + P_{stab} J_s \leq 0 \tag{2.87}$$

has a positive definite solution P_{stab}, from which the solution set of the LMI in (2.83) can be determined [21]):

$$\begin{aligned}\{P \mid {J_{LV}}^T P + P J_{LV} \leq 0,\ P > 0\} = \\ = \left\{ T^{-T} \hat{P} T^{-1} \;\middle|\; \begin{array}{l} \hat{P} = \mathrm{diag}(P_{stab}, P_r) \\ P_{stab} > 0,\ P_r = 0 \\ J_s^T P_{stab} + P_{stab} J_s \leq 0 \end{array} \right\}\end{aligned} \tag{2.88}$$

By that, we obtain a positive *semi*definite quadratic Lyapunov function candidate. According to the theorem above, the asymptotic stability of U^* on the largest invariant set of the manifold $\mathcal{M}$ contained in $ker(P)$ (i.e. where $\overline{U}^T P \overline{U} = (U - U^*)^T P (U - U^*) = 0$) implies (local) stability of U^*.

Chapter 3

Stability analysis of the zero dynamics of a low power gas turbine model in QP form

Gas turbines are important and widely used prime movers in transportation systems. Besides of this area gas turbines are found in power systems where they are the main power generators [31]. Therefore the analysis and control of gas turbines is of great practical importance.

A strongly nonlinear state space description of a low power gas turbine has been developed based on first engineering principles [7]. Because of the strong nonlinearities, however, the nonlinear dynamic analysis (e.g. stability analysis) of the developed model can only be performed with difficulty [10].

In this chapter, we intend to investigate the local stability of two different zero dynamics of the low power gas turbine model developed in [P4]. The basis of the analysis is a detailed gas turbine model that has been developed previously [7].

The following sections give the material of the thesis. First, the stability of the zero dynamics for the turbine inlet total pressure is investigated, where the method described in Section 2.3.10 can be applied, enabling us to estimate the quadratic stability region, as well. Finally, the stability of the zero dynamics for the rotational speed is investigated in a different way.

3.1 Dynamic model of the gas turbine

In this section, a dynamic model is developed for a low power gas turbine. The same model can be found in [P4] as a result of model simplification of an earlier gas turbine model taken from [7].

Figure 3.1 shows the general layout of a gas turbine. The main parts of a gas turbine include the compressor, the combustion chamber and the turbine. The operation principle of gas turbines is roughly the following. Air is drawn into the engine by the compressor, which compresses it and then delivers it to the combustion chamber. Within the combustion chamber, the air is mixed with fuel and the mixture is ignited, producing a rise in temperature and hence an expansion of the gases. These

gases pass through the turbine, which is designed to extract sufficient energy from them to keep the compressor rotating, so that the engine is self sustaining.

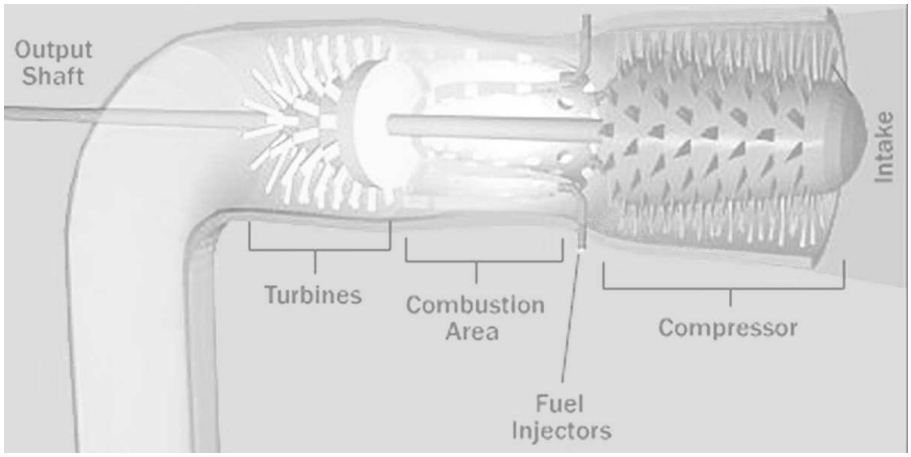

Figure 3.1: The main parts of a gas turbine

A real DEUTZ T216 type low-power gas turbine has been used for our studies. The equipment is installed in the Budapest University of Technology and Economics, Department of Aircraft and Ships on a test-stand. As it is shown by the schematic diagram in Fig. 3.2, two components has also been built to the gas turbine: air is drawn into the engine through the *inlet duct,* while hot gases are exhausted from the turbine through the *gas deflector*.

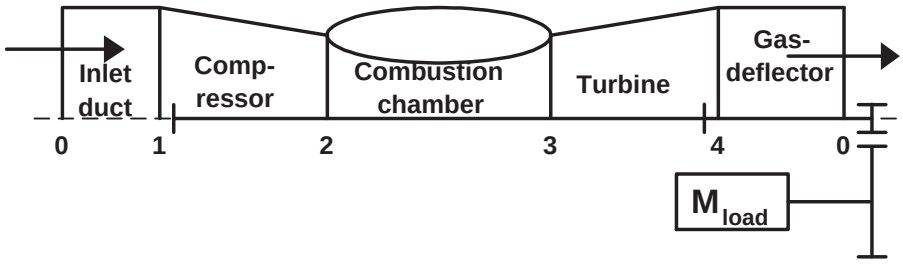

Figure 3.2: Schematic diagram of the DEUTZ T216 gas turbine

A third order nonlinear dynamic model of the DEUTZ T216 gas turbine is developed in [10]. In order to be able to investigate the dynamic properties of the model and base controller design thereon, this nonlinear model had to be simplified. This simplification has been done in [P4] using two model simplifying assumptions. This latter model is presented here, showing the modelling procedure in detail.

Note that a comprehensive list of the notations to be used in this section is given in the Nomenclature of the gas turbine model that can be found in Appendix A.

3.1.1 Modelling assumptions

In order to get a low order dynamic model suitable for control purposes the following modelling assumptions should be made.

General assumptions

- Constant physico-chemical properties are assumed in each main part of the gas turbine, such as specific heat at constant pressure and at constant volume, specific gas constant and adiabatic exponent.
- Heat loss (heat transmission, heat conduction, heat radiation) is neglected.

Other assumptions

- In the inlet duct a constant pressure loss coefficient (σ_I) is assumed.
- In the inlet and in the outlet of the compressor the mass flow rates are the same: $\nu_{Cin} = \nu_{Cout} = \nu_C$, and there is no energy storage effect: $U_2 = constant$.
- In the combustion chamber constant pressure loss coefficient (σ_{Comb}) and constant efficiency of combustion (η_{comb}) are assumed; the enthalpy of fuel is neglected, and the combustion chamber is assumed to be a perfectly stirred region (balance volume). It means that a finite dimensional concentrated parameter model is developed and the value of the variables within this balance volume is equal to that at its outlet.
- In the inlet and in the outlet of the turbine the mass flow rates are the same: $\nu_{Tin} = \nu_{Tout} = \nu_T$, and there is no energy storage effect: $U_4 = constant$.
- In the gas-deflector a constant pressure loss coefficient (σ_N) is assumed.
- The isentropic efficiencies of the compressor (η_C) and the turbine (η_T) are assumed to be constants.
- The physico-chemical properties (such as specific heats at constant pressure and at constant volume, specific gas constants and adiabatic exponents) of air - in the compressor - and hot gas - in the turbine - are assumed to have the same values: $c_p = c_{pair} = c_{pgas}$, $c_v = c_{vair} = c_{vgas}$, $R = R_{air} = R_{gas}$, and $\kappa = \kappa_{air} = \kappa_{gas}$.

3.1.2 Conservation balances

The nonlinear state equations are derived from the laws of conservation principles [8]. Dynamic equations come from the conservation balances constructed for the overall mass m and internal energy U [46]. The development of the model equations is performed in the following steps.

Conservation balance of the total mass:

$$\frac{dm}{dt} = \nu_{in} - \nu_{out}$$

Conservation balance of the internal energy, where the heat energy flows and the power terms are also taken into account:

$$\frac{dU}{dt} = \nu_{in} i_{in} - \nu_{out} i_{out} + Q + P$$

We can transform the above energy conservation equation to its intensive variable form by considering the dependence of the internal energy on the measurable temperature:

$$\frac{dU}{dt} = c_v \frac{d}{dt}(Tm) = c_v T \frac{dm}{dt} + c_v m \frac{dT}{dt}$$

From the two equations above we get a state equation for the temperature as state variable:

$$\frac{dT}{dt} = \frac{\nu_{in} i_{in} - \nu_{out} i_{out} + Q + P - c_v T(\nu_{in} - \nu_{out})}{c_v m}$$

The ideal gas equation ($pV = mRT$) is used together with the two balance equations above to develop an alternative state equation for the pressure:

$$\frac{dp}{dt} = \frac{R}{c_v V}(\nu_{in} i_{in} - \nu_{out} i_{out} + Q + P)$$

Conservation balance of the mechanical energy of the compressor-turbine shaft:

$$\frac{dE_{shaft}}{dt} = \nu_T c_p (T_3 - T_4)\eta_{mech} - \nu_C c_p (T_2 - T_1) - 2\pi \frac{3}{50} M_{load}$$

Note that the quantities T, p and i are *total* parameters of the stream.

3.1.3 Conservation balances in intensive form

These dynamic equations have to be transformed into their intensive variable form to contain measurable quantities. Therefore the set of transformed differential balances include the dynamic mass balance for the combustion chamber, the pressure form of the state equation derived from the internal energy balance for the combustion chamber and the intensive form of the overall mechanical energy balance expressed for the rotational speed n [5].

$$\frac{dm_{Comb}}{dt} = \nu_C + \nu_{fuel} - \nu_T \tag{3.1}$$

$$\frac{dp_3}{dt} = \frac{R}{c_v V_{Comb}}(\nu_C c_p T_2 - \nu_T c_p T_3 + Q_f \eta_{Comb} \nu_{fuel}) \tag{3.2}$$

$$\frac{dn}{dt} = \frac{1}{4\pi^2 \Theta n}(\nu_T c_p (T_3 - T_4)\eta_{mech} - \nu_C c_p (T_2 - T_1) - 2\pi \frac{3}{50} n M_{load}) \tag{3.3}$$

3.1.4 Constitutive (algebraic) equations

Some constitutive equations are also needed to complete the nonlinear gas turbine model [7].

1. Two equations come from the modelling assumptions for the total pressures after the compressor and the turbine:

$$p_2 = \frac{p_3}{\sigma_{Comb}} \tag{3.4}$$

$$p_4 = \frac{p_1}{\sigma_I \sigma_N} \tag{3.5}$$

2. The second is the ideal gas equation which is used for the combustion chamber:

$$T_3 = \frac{p_3 V_{Comb}}{m_{Comb} R} \tag{3.6}$$

3. The third type of constitutive equations describes the total temperature after the compressor (T_2), and the total temperature after the turbine (T_4).

   - The total temperature after the compressor is found by using the isentropic efficiency of the compressor η_C in the following manner:

$$T_2 = T_1 \left(1 + \frac{1}{\eta_C}\left(\left(\frac{p_2}{p_1}\right)^{\frac{\kappa-1}{\kappa}} - 1\right)\right) \tag{3.7}$$

   - The total temperature after the turbine is found similarly by using the isentropic efficiency of the turbine η_T:

$$T_4 = T_3 \left(1 - \eta_T \left(1 - \frac{1}{\left(\frac{p_3}{p_4}\right)^{\frac{\kappa-1}{\kappa}}}\right)\right) \tag{3.8}$$

4. The fourth type of constitutive equations describes the mass flow rate of the compressor and the turbine.

$$\nu_C = \beta A_1 q(\lambda_1) \frac{p_1}{\sqrt{T_1}} \tag{3.9}$$

$$\nu_T = \beta A_3 q(\lambda_3) \frac{p_3}{\sqrt{T_3}} \tag{3.10}$$

   In the equations above, $q(\lambda_1)$ and $q(\lambda_3)$ (the dimensionless mass flow rate of the compressor and the turbine) can be calculated as follows:

$$q(\lambda_1) = F_1\left(\frac{n}{\sqrt{\frac{T_1}{288.15\ K}}}, \frac{p_2}{p_1}\right)$$

$$q(\lambda_3) = F_2\left(\tau \frac{n}{\sqrt{T_3}}, \frac{p_3}{p_4}\right)$$

   In the equations above, $q(\lambda_1)$ is the function of the corrected rotational speed and the compressor pressure ratio, and $q(\lambda_3)$ is the function of the dimensionless velocity and the turbine pressure ratio.

The unknown static parameters of the dynamic model are the unknown coefficients of the polynomials approximating the characteristics of the compressor and the turbine, which can be identified by the least squares method using static measurements. The dynamic parameters of the model (V_{Comb}, Θ) can be identified by using measured step response functions and a simulation model containing the unknown parameters in a nonlinear form [10].

3.1.5 Dynamic model in nonlinear input affine QP-ODE form

All constitutive equations are substitutable to the differential equations, so the final form of the nonlinear DAE model (3.1)-(3.10) of the gas turbine is in the form of three ordinary differential equations obtained by substituting (3.4)-(3.10) to (3.1), (3.2) and (3.3):

$$\begin{aligned}
\frac{dm_{Comb}}{dt} &= \nu_C + \nu_{fuel} - \nu_T && (3.11)\\
\frac{dp_3}{dt} &= \frac{R}{V_{Comb}c_v}\Big(\nu_C c_p T_1\Big(1+\frac{1}{\eta_C}\Big(\Big(\frac{p_3}{p_1\sigma_{Comb}}\Big)^{\frac{\kappa-1}{\kappa}}-1\Big)\Big) \\
&\quad -\nu_T c_p \frac{p_3 V_{Comb}}{m_{Comb}R} + Q_f\eta_{comb}\nu_{fuel}\Big) && (3.12)\\
\frac{dn}{dt} &= \frac{1}{4\pi^2\Theta n}\Big(\nu_T c_p \frac{p_3 V_{Comb}}{m_{Comb}R}\eta_T\eta_{mech}\Big(1-\Big(\frac{p_1}{p_3\sigma_I\sigma_N}\Big)^{\frac{\kappa-1}{\kappa}}\Big) \\
&\quad -\nu_C c_p\frac{T_1}{\eta_C}\Big(\Big(\frac{p_3}{p_1\sigma_{Comb}}\Big)^{\frac{\kappa-1}{\kappa}}-1\Big)\Big) \;-\; \frac{3M_{load}}{100\pi\Theta} && (3.13)
\end{aligned}$$

where

$$\nu_C = \beta\ A_1\ \frac{p_1}{\sqrt{T_1}}\Big(\alpha_1\frac{n}{\sqrt{\frac{T_1}{288.15}}}\frac{p_3}{p_1\sigma_{Comb}}+\alpha_2\frac{n}{\sqrt{\frac{T_1}{288.15}}}+\alpha_3\frac{p_3}{p_1\sigma_{Comb}}+\alpha_4\Big) \qquad (3.14)$$

$$\nu_T = \beta\ A_3\ \frac{p_3}{\sqrt{\frac{p_3V_{Comb}}{m_{Comb}R}}}\Big(\gamma_1\frac{\tau n}{\sqrt{\frac{p_3V_{Comb}}{m_{Comb}R}}}\frac{p_3\sigma_I\sigma_N}{p_1}+\gamma_2\frac{\tau n}{\sqrt{\frac{p_3V_{Comb}}{m_{Comb}R}}}+\gamma_3\frac{p_3\sigma_I\sigma_N}{p_1}+\gamma_4\Big) \qquad (3.15)$$

This dynamic model is in the form of the input-affine state equation (2.3), where

$$f(x)=\begin{bmatrix} f_1(x_1,x_2,x_3,d_1,d_2)\\ f_2(x_1,x_2,x_3,d_1,d_2)\\ f_3(x_1,x_2,x_3,d_1,d_2,d_3)\end{bmatrix},\quad g(x)=\begin{bmatrix} g_1(x)\\ g_2(x)\\ g_3(x)\end{bmatrix}=\begin{bmatrix} b_1\\ b_2\\ 0\end{bmatrix}, \qquad (3.16)$$

and can be re-written in the special input-affine QP-ODE form that has been described in (2.63):

$$\begin{aligned}
\dot{x}_1 &= x_1\Big(c_{1,1}d_2^{-1}\cdot x_1^{-1}x_2x_3 + c_{1,2}d_2^{0.5}\cdot x_1^{-1}x_3 + c_{1,3}d_2^{-0.5}\cdot x_1^{-1}x_2 + c_{1,4}d_1d_2^{-0.5}\cdot x_1^{-1}\\
&+ c_{1,5}d_1^{-1}\cdot x_2x_3 + c_{1,6}\cdot x_3 + c_{1,7}d_1^{-1}\cdot x_1^{-0.5}x_2^{1.5} + c_{1,8}\cdot x_1^{-0.5}x_2^{0.5}\Big) \;+\; b_1u = \\
&= x_1\cdot f_1^{QP}(x_1,x_2,x_3,d_1,d_2) + b_1u && (3.17)\\
\dot{x}_2 &= x_2\Big(c_2d_2^{0.5} + c_{2,1}d_1d_2^{0.5}\cdot x_2^{-1} + c_{2,2}d_1^{-0.2857}d_2^{0.5}\cdot x_2^{0.2857} + c_{2,3}\cdot x_1^{-0.5}x_2^{0.5} +\\
&+ c_{2,4}d_1^{0.7143}\cdot x_2^{-0.7143}x_3 + c_{2,5}d_1^{-0.2857}\cdot x_2^{0.2857}x_3 + c_{2,6}d_1^{-1}\cdot x_2x_3 + c_{2,7}\cdot x_3 +\\
&+ c_{2,8}d_1^{0.7143}d_2^{0.5}\cdot x_2^{-0.7143} + c_{2,9}d_1^{-1}\cdot x_1^{-0.5}x_2^{1.5} + c_{2,10}d_1\cdot x_2^{-1}x_3\Big) \;+\; b_2u \;= \\
&= x_2\cdot f_2^{QP}(x_1,x_2,x_3,d_1,d_2) + b_2u && (3.18)\\
\dot{x}_3 &= x_3\Big(c_{3,1}d_1^{-1}\cdot x_2^2x_3^{-1} + c_{3,2}d_1^{-0.7143}\cdot x_2^{1.7143}x_3^{-1} + c_{3,3}d_1^{0.2857}\cdot x_2^{0.7143}x_3^{-1} +
\end{aligned}$$

$$
\begin{aligned}
&+ \quad c_{3,4}d_1^{0.5}\cdot x_2x_3^{-2} + c_{3,5}d_1^{-1}\cdot x_1^{-0.5}x_2^{2.5}x_3^{-2} + c_{3,6}d_1^{-0.7143}\cdot x_1^{-0.5}x_2^{2.2143}x_3^{-2} + \\
&+ \quad c_{3,7}\cdot x_1^{-0.5}x_2^{1.5}x_3^{-2} + c_{3,8}d_1^{0.2857}\cdot x_1^{-0.5}x_2^{1.2143}x_3^{-2} + c_{3,9}d_1^{-0.2857}\cdot x_2^{1.2857}x_3^{-1} + \\
&+ \quad c_{3,10}d_1d_2^{0.5}\cdot x_3^{-2} + c_{3,11}d_1^{0.7143}d_2^{0.5}\cdot x_2^{0.2857}x_3^{-2} + c_{3,12}d_1^{0.7143}\cdot x_2^{0.2857}x_3^{-1} + \\
&+ \quad c_{3,13}\cdot x_2x_3^{-1} + c_{3,14}d_1^{0.7143}d_2^{0.5}\cdot x_2^{1.2857}x_3^{-2} + (c_{3,15}d_1 + c_{3,16}d_3)\cdot x_3^{-1}\Big) \;= \\
&\quad = x_3\cdot f_3^{QP}(x_1,x_2,x_3,d_1,d_2,d_3)
\end{aligned}
\tag{3.19}
$$

where the scalar valued control input

$$
u = \nu_{fuel} \tag{3.20}
$$

is the mass flowrate of fuel,

$$
x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} m_{Comb} \\ p_3 \\ n \end{bmatrix} \tag{3.21}
$$

is the vector of state variables containing the mass in combustion chamber, the turbine inlet total pressure and the rotational speed, respectively, while

$$
d = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix} = \begin{bmatrix} p_1 \\ T_1 \\ M_{load} \end{bmatrix} \tag{3.22}
$$

is the vector of environmental disturbances containing the compressor inlet total pressure, the compressor inlet total temperature, and the load torque, respectively, and $c_{i,j}$'s and b_i's are the coefficients of the model that can be found in Table A.2 in Appendix A. The measurable output variables of the system are

$$
y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} T_4 \\ p_3 \\ n \end{bmatrix} = \begin{bmatrix} o_1\cdot x_1^{-1}x_2 + o_2d_1^{0.2857}\cdot x_2^{-0.2857} \\ x_2 \\ x_3 \end{bmatrix} \tag{3.23}
$$

where the coefficients o_1 and o_2 can be found in Table A.2 in Appendix A. Note that the second and third output variables are identical to the second and third state variables, respectively, while the first output is the turbine outlet total temperature which is chosen instead of the first state variable, since it is directly measurable, and x_1 can be expressed from it algebraically.

It has to be emphasized that the state equations (3.17)-(3.19) do not only depend on the state variables, but also on the elements of the vector of environmental disturbances. Also note that the input enters the model equations linearly. The state, input, output and environmental disturbance variables are explained in Table 3.1, while a comprehensive list is given in the Nomenclature of the gas turbine model that can be found in Appendix A.

The unknown parameters of the model have been estimated using measured data following the method described in [10], their estimated values together with their units can be found in Table A.1.

The dynamical model of the gas turbine is valid within the following operating domain:

Table 3.1: State, input, output and disturbance variables of the gas turbine model (see Nomenclature for further details)

Notation	**Variable name/Units**
$x_1 = m_{Comb}$	mass in combustion chamber $[kg]$
$x_2 = y_2 = p_3$	turbine inlet total pressure $[Pa]$
$x_3 = y_3 = n$	rotational speed $[1/s]$
$u = \nu_{fuel}$	mass flowrate of fuel $[kg/s]$
$y_1 = T_4$	turbine outlet total temperature $[K]$
$d_1 = p_1$	compressor inlet total pressure $[Pa]$
$d_2 = T_1$	compressor inlet total temperature $[K]$
$d_3 = M_{load}$	load torque $[Nm]$

$$\begin{aligned} 0.00305 \text{ kg} &= x_{1min} \leq x_1 \leq x_{1max} = 0.00835 \text{ kg} \\ 154837 \text{ Pa} &= x_{2min} \leq x_2 \leq x_{2max} = 325637 \text{ Pa} \\ 650 \frac{1}{s} &= x_{3min} \leq x_3 \leq x_{3max} = 833.33 \frac{1}{s} \end{aligned}$$

From now on, this domain is denoted by $\mathcal{X}$. Note that the structure of the state equations (3.17)-(3.19) is such that for every constant reference control input and every constant disturbance variable vector there is a unique steady state point in $\mathcal{X}$ [6].

The typical values of the environmental disturbances are

$$p_1 = 101189.15 \text{ Pa} \tag{3.24}$$
$$T_1 = 288.15 \text{ K} \tag{3.25}$$
$$M_{load} = 50 \text{ Nm} \tag{3.26}$$

while they may vary between the following minimum and maximum values:

$$\min p_1 = 90000Pa \quad , \quad \max p_1 = 110000Pa \tag{3.27}$$
$$\min T_1 = 268.15K \quad , \quad \max T_1 = 303.15K \tag{3.28}$$
$$\min M_{load} = 0Nm \quad , \quad \max M_{load} = 150Nm \tag{3.29}$$

Sensitivity analysis in [10] showed that the model is particularly sensitive to the change of its three parameters: the efficiency of combustion (η_{comb}), the volume of the combustion chamber (V_{Comb}) and the inertial moment (Θ). The values of these parameters fall between the following extrema:

$$\min V_{Comb} = 0.0053m^3 \quad , \quad \max V_{Comb} = 0.0061m^3 \tag{3.30}$$
$$\min \Theta = 0.0003kgm^2 \quad , \quad \max \Theta = 0.0005kgm^2 \tag{3.31}$$
$$\min \eta_{comb} = 0.74768 \quad , \quad \max \eta_{comb} = 0.82048 \tag{3.32}$$

while the nominal values of these parameters can be found in Table A.1 in Appendix A.

3.2 Local stability of the gas turbine with the turbine inlet total pressure held constant

In this section, the zero dynamics of the system for turbine inlet total pressure is computed first and then local stability analysis is applied thereon. At the end of the section, simulation results are presented.

3.2.1 Zero dynamics for turbine inlet total pressure

The quadratic stability of the zero dynamics for the turbine inlet total pressure will be investigated at the operating point belonging to the constant input value $u^* = 0.009913$ and to the typical values of the environmental disturbances (see (3.24)-(3.26)):

$$x^* = [x_1^*,\ x_2^*,\ x_3^*]^T = [0.00528\ kg,\ 223587.2\ Pa,\ 750\ 1/s]^T$$

Let us consider the case when the pressure p_3 is held constant ($x_2 = x_2^*$) with an appropriate control input. This controlling goal can be re-phrased as finding a "zeroing input" for the artificial output $y_{ZD} = x_2 - x_2^*$, e.g. the zero dynamics of the model (3.17)-(3.19) for the output y_{ZD}:

$$y_{ZD} = x_2 - x_2^* = 0 \tag{3.33}$$

Since (3.17)-(3.19), (3.33) is in the form of (2.63)-(2.64) and the condition (2.67) is fulfilled, the relative degree equals to one globally, and the zero dynamics can be computed using (2.68). The resulted zero dynamics is an autonomous QP-ODE with two state variables and eight quasi-monomials:

$$\dot{x}_{1d} = x_{1d}(A_{11}x_{1d}^{-1} + A_{12}x_{3d} + A_{13}x_{1d}^{-1/2} + A_{14}x_{1d}^{-3/2} + A_{17}x_{1d}^{-1}x_{3d}) \tag{3.34}$$

$$\dot{x}_{3d} = x_{3d}(A_{25}x_{3d}^{-1} + A_{26}x_{3d}^{-2} + A_{28}x_{1d}^{-1/2}x_{3d}^{-2}) \tag{3.35}$$

where the constants A_{ij} are the elements of A. This parameter matrix and the exponent matrix B are the following:

$$A = \begin{bmatrix} -244.48 & 8.224 & -178.7 & 4.42 & 0 & 0 & 93.645 & 0 \\ 0 & 0 & 0 & 0 & -130.41 & 291.58 & 0 & 241.57 \end{bmatrix}$$

$$B^T = \begin{bmatrix} -1 & 0 & -\frac{1}{2} & -\frac{3}{2} & 0 & 0 & -1 & -\frac{1}{2} \\ 0 & 1 & 0 & 0 & -1 & -2 & 1 & -2 \end{bmatrix}$$

while the vector of quasi-monomials is

$$U = [U_1, \ldots, U_8]^T = \left[x_{1d}^{-1},\ x_{3d},\ x_{1d}^{-1/2},\ x_{1d}^{-3/2},\ x_{3d}^{-1},\ x_{3d}^{-2},\ x_{1d}^{-1}x_{3d},\ x_{1d}^{-1/2}x_{3d}^{-2}\right]^T \tag{3.36}$$

The subscript d in $x_{1d},\ x_{3d}$ denotes *dimensionless* state variables which were introduced to avoid computational problems caused by the huge difference in the order of magnitude of x_1 and x_3. The dimensionless state variables are computed as

$$x_{id} = \frac{x_i}{x_{imax} - x_{imin}}\ ,\quad i = 1, 3 \tag{3.37}$$

The parameter matrix of the Lotka-Volterra model can be easily computed by matrix multiplication:

$$A_{LV} = B \cdot A \in \mathbb{R}^{8\times 8} \tag{3.38}$$

Choosing $\{U_1 = x_1^{-1},\ U_2 = x_3\}$ as a base, all the LV-variables can be computed, defining a manifold where the state trajectories can evolve:

$$\begin{aligned}\mathcal{M} = \Big\{ &(U_1, U_2, U_3, \ldots, U_8) \in \mathbb{R}^8 \mid U_3 = U_1^{1/2},\ U_4 = U_1^{3/2}, \\ &U_5 = U_2^{-1},\ U_6 = U_2^{-2},\ U_7 = U_1U_2,\ U_8 = U_1^{1/2}U_2^{-2} \Big\}\end{aligned} \tag{3.39}$$

3.2.2 Local quadratic stability

Since $A_{LV} \in \mathbb{R}^{8\times 8}$ is rank-deficient ($rank(A_{LV}) = 2$), the extension of the quadratic stability analysis to non-minimal LV systems described in Section 2.3.10 is applied. The steady state of the LV variable vector (U^*) can be computed from x^* using (3.36) and (3.37). The centered LV variable vector is then defined as $\overline{U} = U - U^*$.

The quadratic Lyapunov function is searched in the form

$$V(\overline{U}) = \overline{U}^T P \overline{U}$$

where P is computed by (2.88) using the similarity transformation T in (2.86), where P in the transformed coordinates is given as $\hat{P} = diag(P_{stab}, P_r)$ (see (2.88)). In our case, $P_r = 0 \in \mathbb{R}^{6\times 6}$, while the positive definite symmetric matrix $P_{stab} \in \mathbb{R}^{2\times 2}$ fulfills the LMI in (2.87). Instead of solving the LMI in (2.87), a Lyapunov equation in the form

$$J_s^T P_{stab} + P_{stab} J_s = -Q_{stab} \tag{3.40}$$

has been solved for P_{stab}, where

$$Q_{stab} = \begin{bmatrix} 10 & -50 \\ -50 & 500 \end{bmatrix} \in \mathbb{R}^{2\times 2}$$

is positive definite and symmetric. With the resulted P_{stab}, the matrix $\hat{P}$ has been construed and transformed back to the original coordinates using the similarity transformation T. The resulted P matrix in the original coordinates is the following:

$$P = \begin{bmatrix} 28.589 & -0.962 & 20.899 & -0.517 & 20.021 & -44.763 & -10.951 & -37.086 \\ -0.962 & 0.032 & -0.703 & 0.017 & -0.673 & 1.506 & 0.368 & 1.248 \\ 20.899 & -0.703 & 15.277 & -0.378 & 14.635 & -32.721 & -8.005 & -27.11 \\ -0.517 & 0.017 & -0.378 & 0.009 & -0.362 & 0.809 & 0.198 & 0.67 \\ 20.021 & -0.673 & 14.635 & -0.362 & 324.46 & -725.45 & -7.669 & -601.04 \\ -44.763 & 1.506 & -32.721 & 0.809 & -725.45 & 1622 & 17.146 & 1343.8 \\ -10.951 & 0.368 & -8.005 & 0.198 & -7.669 & 17.146 & 4.195 & 14.206 \\ -37.086 & 1.248 & -27.11 & 0.67 & -601.04 & 1343.8 & 14.206 & 1113.4 \end{bmatrix}$$

which is indeed symmetric and positive semi-definite with $rank(P) = 2$ and fulfills the LMI in (2.83).

Since V is positive semi-definite at U^*, it is in the form of (2.85). The intersection of sets $\mathcal{M}$ and $\ker(P)$ is a set made of two isolated points: one of them is

the operating point, the other is outside of the valid operating domain of the gas turbine model - in terms of x, this point is at ($x_1 = 0.3577$ kg, $x_3 = 452$ $1/s$). It means that in the valid operating domain of the gas turbine, the largest positively invariant set of $\{x \mid V(x) = 0\}$ is $K = \{U^*\}$. Consequently, the operating point U^* is locally asymptotically stable with respect to K showing that U^* is a (locally) stable equilibrium.

3.2.3 Estimation of the quadratic stability region

Since we have P which fulfills (2.83) and LMIs form convex constraints on their variables, we only have to solve (2.84) for $\langle\overline{U}\rangle$ to find a convex stability region of the origin. Note that $U_3, \ldots, U_8$ are determined by U_1 and U_2, therefore we only have to change these two variables to find this convex polygon. In our case, the convex polygon is a pentagon depicted in Fig. 3.3 by dashed line. It is a closed curve, however its right half is left out from the picture because of its big size. The operating domain (i.e. where the model is valid) is a box (dotted line), containing the operating point (denoted by an 'x').

A quadratic stability neighborhood is an ellipsoid determined by the level-curves of $\overline{U}^T P \overline{U} = c$ where $c \in \mathbb{R}$ is a constant. This ellipsoid is eight-dimensional in our case. Fortunately, we only have to consider its projection to the two-dimensional manifold $\mathcal{M}$ defined in (3.39). Since $U_3, \ldots, U_8$ are determined by U_1 and U_2 we substitute them into $\overline{U}^T P \overline{U} = c$ and get a nonlinear implicit relationship between U_1 and U_2. This equation gives closed Lyapunov level-curves. As Fig. 3.3 shows,

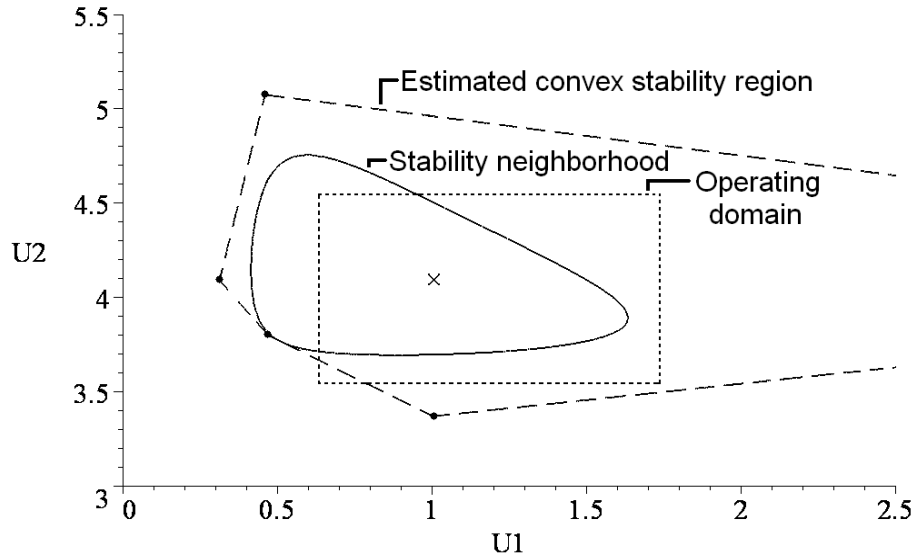

Figure 3.3: Quadratic stability neighborhood estimation

the biggest level curve fitting in the convex region surrounds the quadratic stability neighborhood (solid line). The size of the stability neighborhood is approximately 56 % of the operating domain.

Note that the P matrix of the Lyapunov function is chosen such that the estimated stability *region* is maximized as much as possible. This is done by the appropriate choice of the elements of $Q_{stab} \in \mathbb{R}^{2\times 2}$ in (3.40): one of the diagonal elements can be fixed arbitrarily, then - since Q_{stab} is symmetric and therefore the off-diagonal elements are identical - there are only two elements that can be chosen. Because of the lack of any constructive methods, this is made by a heuristical 'tuning' of these two elements of Q_{stab}.

Simulation results

The MATLAB/SIMULINK model of the zero dynamics for the turbine inlet total pressure has been built not just to confirm previous results, but to test the remainder part of the operating domain.

Simulations suggest that the zero dynamics is stable not only in the estimated quadratic stability region but also in the whole operating domain. It means that the quadratic Lyapunov function candidate was not able to indicate the whole stability *neighborhood*, however the convex stability *region* almost covers the whole operating domain. It is because the applied method does not give any tools to optimize the choice of the Lyapunov function regarding to the magnitude of the stability *neighborhood*, and therefore this quadratic Lyapunov function candidate has been chosen by maximizing the convex stability *region*, and most likely it was not the 'best choice'. Figure 3.4 shows the transients of $x_1 = m_{Comb}$ and $x_3 = n$ started from the

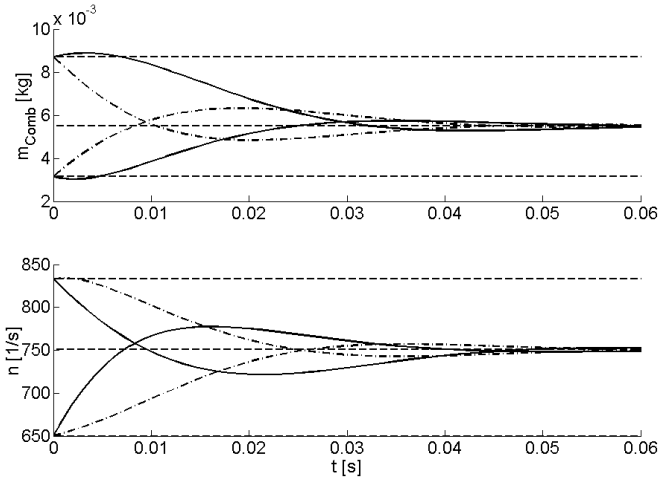

Figure 3.4: Transients started from corner points

four corners of the operating domain. Transients started from $(m_{Comb_{min}}, n_{min})$ and $(m_{Comb_{max}}, n_{max})$ are denoted by solid, transients started from $(m_{Comb_{min}}, n_{max})$ and $(m_{Comb_{max}}, n_{min})$ are denoted by dash-dot lines.

Figure 3.4 shows that all transients are stable, but m_{Comb} has overshoots that leave the operating domain (dashed line) in two cases of four (solid line). However, these overshoots are not critical because of their negligible magnitude.

3.2.4 Discussion

The local quadratic stability neighborhood of the steady states of several zero dynamics belonging to different constant p_3 values has also been estimated.

It is known that the steady states of the turbine model lay on a one dimensional curve in the operating domain [6]. This one dimensional set of steady states is depicted by dashed line in Fig. 3.5, where the system of coordinates is spanned again by the first two monomials U_1 and U_2.

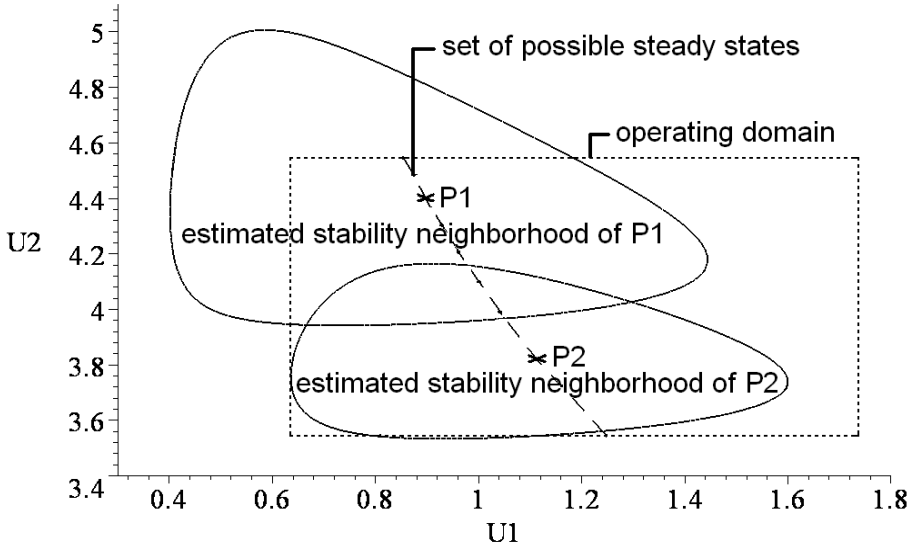

Figure 3.5: Quadratic stability neighborhood estimation

This figure also shows the biggest estimated stability neighborhoods (denoted by solid lines) of the steady states of two different zero dynamics. These steady states are denoted by P_1 and P_2. Simulations in MATLAB/SIMULINK showed again that the range of stability is much wider than the estimated one in both cases.

The conservativeness of the quadratic stability estimation method can be deduced from the following facts:

- The solution of two LMIs instead of one BMI may reduce the estimated stability *region*,
- Finding the biggest convex stability region is heuristical,
- The fact that one finds a convex region that is larger than a former one does not imply that the biggest Lyapunov level curve that fits in it (i.e. the estimated stability neighborhood) is larger than the former convex region's biggest level curve.

3.3 Local stability of the gas turbine with the rotational speed held constant

In this section, the local stability of the zero dynamics of (3.17)-(3.19) for the rotational speed is investigated. Thus, the artificial output is chosen as

$$y_{ZD} = h_{ZD}(x) = x_3 - x_3^* = 0 \tag{3.41}$$

The state space model (3.17)-(3.19), (3.41) is in the form (2.63)-(2.64), however the condition (2.67) is not fulfilled. This means that the relative degree of the model (3.17)-(3.19), (3.41) is globally greater than 1. On the other hand, it is also true that

$$L_g L_f h_{ZD}(x) \neq 0, \quad \forall\, x \in \mathcal{X}$$

therefore $r = 2$ in the whole operating domain of the gas turbine model. This means that the first and second derivative of the output can be written as

$$\dot{y}_{ZD} = L_f h_{ZD}(x) = 0 \tag{3.42}$$

$$\ddot{y}_{ZD} = L_f^2 h_{ZD}(x) + L_g L_f h_{ZD}(x) u = 0 \tag{3.43}$$

Using the constraints $x_3 = x_3^*$ and (3.42), and computing the zeroing input from (3.43), we can write the one-dimensional zero dynamics as a function of x_2, i.e.

$$\dot{x}_2 = \phi(x_2) \tag{3.44}$$

The nonlinear function ϕ is a rather long expression to give here in detail, but it can be found in Appendix B. Although ϕ is a function of only one variable, it is hard to analytically treat the sign of it. Thus, we apply a simple graphical method to examine the stability of the zero dynamics, namely the phase diagram.

The phase diagram of the scalar ODE in (3.44) shows $\dot{x}_2$ as a function of x_2. A steady state x_2^* of (3.44) occurs, where the curve $(x_2, \dot{x}_2)$ crosses the horizontal axis (since $\dot{x}_2 = 0$ here). Moreover, if $\dot{x}_2 > 0\ \forall x_2 < x_2^*$ and $\dot{x}_2 < 0\ \forall x_2 > x_2^*$, the unique steady state x_2^* is globally stable. In our case, if these conditions are fulfilled in the operating domain ($x_{2min} \leq x_2 \leq x_{2max}$), then x_2^* is stable in the operating domain.

Figure 3.6 shows four phase diagrams of the zero dynamics belonging to different constant values of the rotational speed, near the typical value of the load torque: $M_{load} = 50\ Nm$. In all four cases, the equilibrium point x_2^* (the point where the curve crosses the horizontal axis) is unique and stable in the operating domain of the gas turbine. Although Fig. 3.6 illustrates all zero dynamics with the same load torque value, phase diagrams of zero dynamics with a large number of different load torque values (between 0 Nm and 150 Nm) - and also a large number of different rotational speed values in the whole range of the operating domain - showed us that the operating points are unique and stable in the whole operating domain $\mathcal{X}$ in all cases. In Fig. 3.7 the phase diagrams belong to four different M_{load} values, while the rotational speed is uniformly set to a typical steady state value (750 $1/s$). In Fig. 3.8 the phase diagrams belong to four different M_{load} values, while the rotational speed is uniformly set to its minimum (phase diagrams upstairs) and to its maximum (phase diagrams downstairs). The uniqueness and asymptotic stability of the operating points is well demonstrated in these cases.

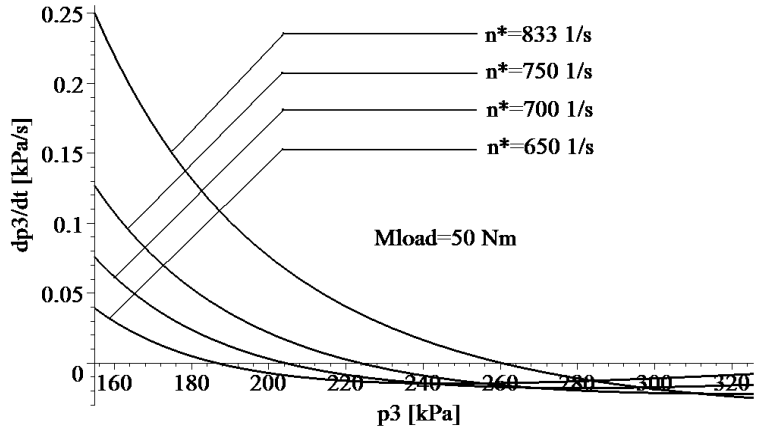

Figure 3.6: Phase diagrams for the system with four different rotational speed values

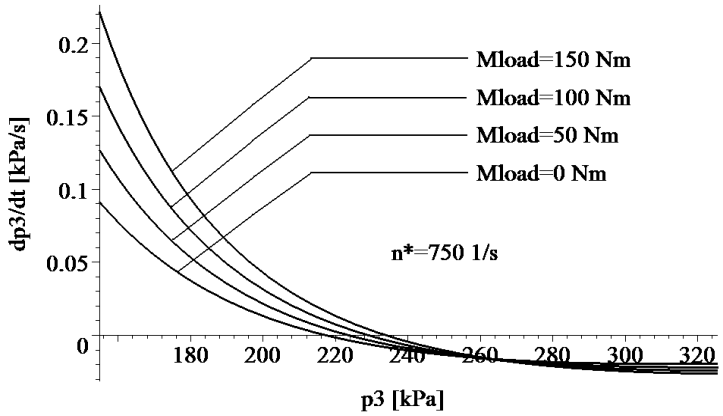

Figure 3.7: Phase diagrams for the system with four different load torque values near a typical rotational speed value

3.4 Summary

In this chapter, the local stability of two different zero dynamics of the low power gas turbine model has been investigated. First, as the basis of the analysis, a strongly nonlinear third order dynamic model has been presented.

The results of stability analysis of the two different zero dynamics constitute the material of the joint thesis, which can be summarized as follows. For the zero dynamics for the turbine inlet total pressure, the quadratic stability analysis method for LV systems has been applied, with stability neighborhood estimation. The estimated stability neighborhood covers 56 % of the operating domain. Simulations confirmed theoretical results, and showed the conservativeness of this estimation since the range of stability is proved to be wider than the estimated one.

The quadratic stability analysis of the steady states of several zero dynamics belonging to different steady state values of the turbine inlet total pressure gave

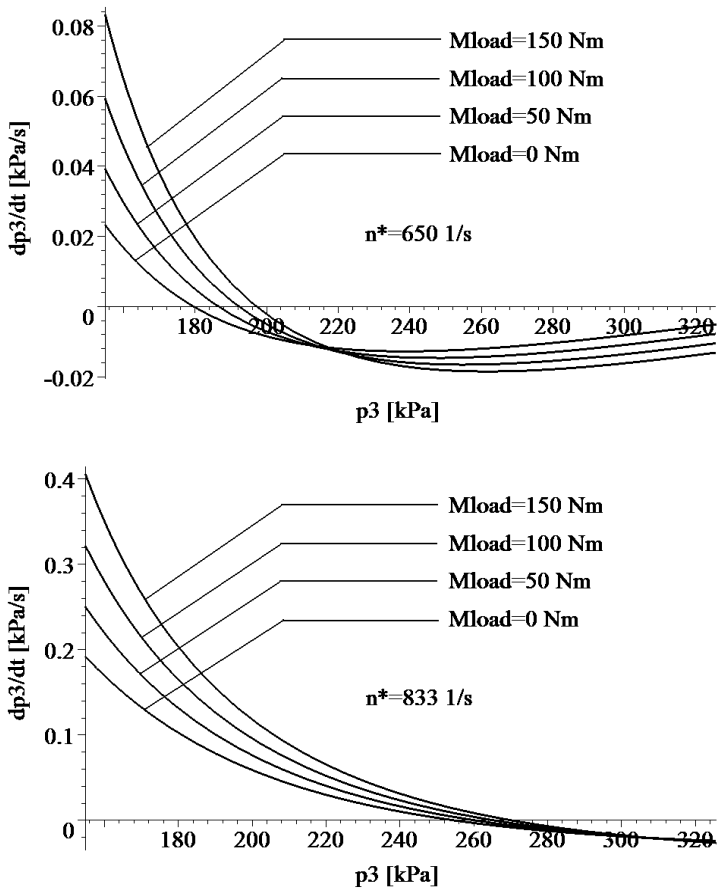

Figure 3.8: Phase diagrams for the system with four different load torque values near minimal and maximal rotational speed values

similar results. The possible causes of the conservativeness of the estimation method have also been discussed.

In case of the zero dynamics for the rotational speed, the former method could not be applied. The stability of this one dimensional zero dynamics is investigated via phase diagrams, and it is found to be stable in the whole operating domain near arbitrary constant values of the rotational speed and also of the load torque, which is the most versatile environmental disturbance.

The importance of the analysis done in this chapter is that it is a preliminary step for selecting an appropriate controller structure for the low power gas turbine.

Chapter 4

Controller design for a low-power gas turbine

In this chapter, three controllers are designed for the low-power gas turbine. All of them are based on input-output linearization, however they are designed to solve different control problems. These control aims are responsible for the differences between the structure and type of controllers.

Note that the results of the controller design procedure are theoretical only, since the controllers haven't been tested on the real pilot-plant of the gas turbine: the performance of the controllers have been checked via MATLAB/SIMULINK simulation experiments. The inputs of the controller design procedure are the controlling goals, while the result is the controller that has been tuned via simulations.

This chapter is organized as follows. After a literature review, the selection of an appropriate control structure is developed for the gas turbine. Then three controllers are designed and tested via simulations. Finally, the performance of the controlled systems are discussed and compared.

4.1 Literature review

The first part of this section gives an overview on the most important 'classical' state space based control methods, without the need of being exhaustive because of the space limitations. Control applications are cited from the fields of transportation systems (as the main application area of gas turbines) and process systems (since the gas turbine has been modelled as a process system in Section 3.1). The second part of this section deals with gas turbine control.

4.1.1 State space based control methods

The simplest controllers are designed to LTI models in the form (2.1)-(2.2). The most commonly used techniques are based on LQ optimal controller design (see Section 2.1.6) such as linear quadratic Gaussian (LQG) controllers [23], where an additional state observer asymptotically estimates the state variables, or LQG with loop transfer recovery (LQG-LTR) technique to recover some of the robustness properties of LQ controllers ([57],[77]). Various types of $\mathcal{H}_\infty$ optimal controllers [105] are also

widely used to guarantee not just the asymptotic stability, but also the robustness of the controlled plant (see e.g. [36],[37],[38]).

It is also widespread to put linear controllers to nonlinear systems: the linear controller is usually designed to the locally linearized version of the nonlinear model around a steady state operating point (see e.g. [91]). However, these controllers can guarantee the stability and robustness of the controlled plant only in a neighborhood of unknown magnitude around the steady state operating point. Although proportional-integral-derivative (PID) controllers [13] are not necessarily based on state space models, they have to be mentioned here because of their important role in practice, e.g. in process control [66].

As a first step towards nonlinear control, the gain scheduling technique [64] uses linear controllers designed to locally linearized models at different well-chosen operating points. The controller applied is a linear combination of these linear controllers, depending on the actual value of the state variable.

However 'classical' nonlinear control theory concerns with controller design to systems represented by nonlinear input-affine state space models, there are narrower model classes that help the controller design procedure by their special structure. For example, linear parameter varying (LPV) systems [89] are well investigated with a plenty of different controllers that can be designed constructively thereon (e.g. [79],[86]). Another example is a nonlinear controller that exploits the special QP structure of the system model in [69]. Another widely used method called backstepping [73] also needs a special nonlinear model structure [24].

For general nonlinear input-affine systems, control Lyapunov function theory is a promising approach: the static nonlinear feedback is computed using a well-chosen Lyapunov function of the closed loop system [55].

For SISO nonlinear input-affine systems, sliding mode control can also be applied: this is a robust control technique that forces the state trajectories to a lower dimensional manifold, namely the sliding manifold [81]. For SISO nonlinear input-affine systems with non-maximal relative degree, the application of the input-output linearization technique results in a system that is linear in input-output sense (see Section 2.1.6), giving rise to the application of linear controllers. However, input-output linearization has a main drawback: it is very sensitive with respect to unmodelled dynamics, and therefore to environmental disturbances and model parameter uncertainties [49] (model matching problem).

This calls for the design of controllers that adaptively estimate time-varying disturbances and/or uncertain parameters. As a consequence, state space based adaptive control methods [71] are frequently applied in the control of transportation systems [50],[70],[78].

A good comparison of most of the methods discussed in this section can be found in a case study [95], where these techniques are applied to control a continuous fermentation process.

4.1.2 Control of gas turbines

Control techniques applied for gas turbines are most often based on linear controllers. PID controllers are frequently used; see e.g. a variant of PID controllers in [74], or

a PI controller designed and tuned to a locally linearized gas turbine model in [56]. Another group of linear controllers are variants of linear quadratic (LQ) controllers, e.g. in [99], [100]. An LQ servo controller is applied to track a reference signal in [80]. LQG/LTR technique [14] and robust controller design [12] has also been performed for gas turbines. In [102], a gain-scheduling controller is designed, using seven well-chosen operating points.

Nonlinear control approaches include model predictive control [25],[52] or soft computing methods [62] such as neural networks [60], genetic algorithms [65] and fuzzy controllers [27]. In [58], mathematical programming is proposed for optimal gas turbine control. However, none of the above mentioned studies examine the robustness of the proposed controllers with respect to the changing environmental conditions and uncertain physical model parameters.

The application of classical nonlinear state-space methods in gas turbine control is not frequent, although several nonlinear control solutions seem to be promising from other application areas. For example, nonlinear adaptive control schemes can be applied to physically similar models in transportation engineering (see e.g. the nonlinear adaptive tracking control of an induction motor with uncertain load torque [72] or a robust backstepping-based control method with actuator failure compensation, applied to a nonlinear aircraft model [96]).

The immediate previous study to our work has used the same gas turbine and turbine model as discussed in Chapter 3 to construct a nonlinear Lyapunov-function based controller thereon [9]. In a PhD thesis [6], this nonlinear controller is compared to an LQ servo controller, as a reference case known from the literature. As a result of the comparison it is pointed out that the system controlled by the nonlinear control Lyapunov-function based controller exhibits similar or better qualitative and quantitative behavior, than the system controlled by the LQ servo controller. However, the design of the nonlinear control Lyapunov-function based controller included some key heuristically performed steps that were strongly specific to the pilot-plant gas turbine model. Therefore, the need to apply an alternative technique led to the application of an input-output linearization based controller [11].

The main possibly time-varying disturbance during the operation of gas turbines is the load torque. In the case of gas turbines, similarly to other rotating machines like induction motors [41] or diesel engines [106], the value of the load torque gives very important state and diagnostic information about the system. Moreover, the knowledge of load torque can largely contribute to the design of more efficient control schemes [106]. However, the instrumental measurement of load torque is not always possible in practice, therefore dynamic model-based identification and estimation methods are often required to solve this problem.

4.2 Control structure selection

As it has been described in Chapter 3, the gas turbine can be controlled via the mass flowrate of fuel, therefore it is the control input ($u = \nu_{fuel}$), and for the system outputs measurable variables are chosen: the turbine outlet total temperature (T_4), the turbine inlet total pressure (p_3) and the rotational speed (n). It has also been

discussed that two state variables coincide with the measured outputs, while the only non-measurable state, the mass in combustion chamber (m_{Comb}) can be algebraically computed from the state variables. This gives rise to the application of state feedback controllers. Because of the strongly nonlinear nature of the gas turbine model, the controllers to be designed will be based on input-output linearization.

It is known that other nonlinear controller design methods such as sliding mode control or backstepping usually show better robustness against model uncertainties. However, in our case the following properties of the gas turbine system called for applying feedback linearization. Firstly, the algebraic computation of the feedback law for our complex nonlinear model seemed to be the only feasible alternative compared to other nonlinear methods. Secondly, linearization with the given input-output structure introduces the rotational speed and its time-derivative as state variables after the necessary coordinates transformation and this is advantageous and useful from a physical and engineering point of view. Thirdly, the linearization-based controller can serve as a basis for other more realistic nonlinear design schemes where the uncertainties are further handled analytically.

In Chapter 3, the stability of two different zero dynamics - one for the turbine inlet total pressure, another for the rotational speed - of the low power gas turbine model has been investigated. These two zero dynamics are closely related to the zero dynamics of the input-output linearized plant, depending on whether the linearization is performed for the turbine inlet total pressure or for the rotational speed. The rotational speed has been chosen to be the variable that the linearization is performed for, because of the following considerations:

- The control of the rotational speed is more widespread in practice,
- the rotational speed is more suitable for measurements,
- a servo controller can be built for the rotational speed which is practical and usable in real problems,
- the stability of the zero dynamics for the rotational speed proved to be stable in the whole operating domain independently of the values of the rotational speed and of the load torque.

For the input-output linearized plant, a stabilizing linear controller has to be applied. Because of the sensitivity of input-output linearization against the change of environmental disturbances and model parameter uncertainties (i.e. the model matching problem), this controller should be a robust one. In the following sections, three different controllers are designed to the input-output linearized plant in order to solve different control problems. The first one is an LQ servo controller to solve reference tracking for the rotational speed and to guarantee robustness. The second is an LQ controller with an additional constrained linear optimal controller to guarantee that the rotational speed and its time-derivative remain in a bounded domain. Unlike the previous ones, the third, LQ servo controller concerns with the more realistic case when the load torque is unknown: its time function has to be estimated. This can be done by *adaptive* input-output linearization. The controller should solve reference tracking and guarantee the robustness of the controlled plant.

4.3 LQ servo controller design based on input-output linearization

In this section, we perform input-output linearization and design an LQ servo controller to the input-output linearized model. As a first attempt, we will assume that the time evolution (signal) of the load torque is known.

4.3.1 Control problems and aims

The gas turbine is to be used as a prime mover for a transportation system, therefore the controller has to fulfill several requirements.

- The gas turbine has to drive a transportation system with changeable velocity, under different circumstances. This indicates the *first control aim*: the rotational speed ($x_3 = n$) of the gas turbine has to follow a prescribed (piecewise constant) reference signal, with a settling time between 1 s and 2 s.
- Two of three environmental disturbances, as the compressor inlet total pressure and temperature ($d_1 = p_1$ and $d_2 = T_1$, respectively) are directly measurable quantities of the ambience. The third one, namely the load torque is the most versatile disturbance, moreover it has the largest effect on the rotational speed. In this case we assume that the time evolution of the load torque is known, meaning that all the environmental disturbances are given. Moreover, there are additional parameter uncertainties in the model. The *second control aim* is that the reference tracking has to be robust against both environmental disturbances and model parameter uncertainties. This implies that the rotational speed remains inside its operating domain (in case of a proper reference signal).
- The *third control aim* applies to the control input variable: the actuator has to be defended against saturation, i.e.

$$0\ kg/s \leq u = \nu_{fuel} \leq 0.03\ kg/s\ .$$

4.3.2 Input-output linearization

Since the relative degree of the gas turbine model (3.17)-(3.19) with the rotational speed as output ($y = h_{ZD}(x) = x_3$) and the mass flowrate of fuel as input is uniformly equal to $r = 2$ in the operating domain, the input-output linearizing feedback can be written in the standard form of (2.29):

$$u = \alpha(x) + \beta(x)v$$

with the new external input v, and $\alpha, \beta \in \mathbb{R}^3 \mapsto \mathbb{R}$:

$$\alpha(x) = -\frac{L_f^2 h_{ZD}(x)}{L_g L_f h_{ZD}(x)} = -\frac{\frac{\partial f_3}{\partial x_1} + \frac{\partial f_3}{\partial x_2} + \frac{\partial f_3}{\partial x_3}}{b_1 \frac{\partial f_3}{\partial x_1} + b_2 \frac{\partial f_3}{\partial x_2}} \tag{4.1}$$

$$\beta(x) = \frac{1}{L_g L_f h_{ZD}(x)} = \frac{1}{b_1 \frac{\partial f_3}{\partial x_1} + b_2 \frac{\partial f_3}{\partial x_2}} \tag{4.2}$$

where f and g are the state and input functions of the input-affine gas turbine model, while $f_i,\ i = 1, \ldots, 3$ are the coordinate functions of f, as defined in (3.16).

Note that α and β also depend on the environmental disturbances collected in vector d (3.22). Since the first two environmental disturbances (p_1 and T_1) are directly measurable, the only problem is with the third one, namely the load torque (M_{load}). In the following, it will be assumed that the time function of the load torque is known (e.g. it is changed by the user) and therefore it can be used in the computation of the linearizing feedback.

Let us use the following notations

$$\bar{z}_1 = x_3 - x_3^* \tag{4.3}$$
$$\bar{z}_2 = \dot{x}_3 \tag{4.4}$$
$$\bar{v} = v - v^* \tag{4.5}$$
$$\bar{y} = y - y^* = y - x_3^* \tag{4.6}$$

where v^* is the necessary constant input value corresponding to the steady state x_3^*. Using (4.3)-(4.5) the linearized system model can be written as a double integrator in the new coordinates:

$$\begin{bmatrix} \dot{\bar{z}}_1 \\ \dot{\bar{z}}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \bar{z}_1 \\ \bar{z}_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \bar{v} \tag{4.7}$$

while the zero dynamics is given by

$$\dot{x}_2 = \Phi(z_1, z_2, x_2), \tag{4.8}$$

where $\Phi(0, 0, x_2) = \phi(x_2)$ (that is the zero dynamics in (3.44)), and the output equation is

$$\bar{y} = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} \bar{z}_1 \\ \bar{z}_2 \end{bmatrix} \tag{4.9}$$

Since the input-output linearizing feedback is a state feedback, the state variables of the system have to be determined from the measured output variables. The state variables x_2 and x_3 are directly measurable, while x_1 can be determined from the first component of the vector of measurable output variables in (3.23) using the measured value of d_1:

$$x_1 = \frac{o_1 x_2}{y_1 - o_2 d_1^{0.2857} x_2^{-0.2857}} = \frac{o_1 y_2}{y_1 - o_2 d_1^{0.2857} y_2^{-0.2857}}$$

where the coefficients o_1 and o_2 can be found in Table A.2 in Appendix A. It means that a static estimator is used to determine x_1. The structure of the input-output linearized plant is shown in Fig. 4.1.

4.3.3 Servo controller with stabilizing feedback

Since the model in (4.7),(4.9) is a cascade of integrators (i.e. it is in the form of (2.11)-(2.12) with $n = 2$), it would be enough to apply an LQ state feedback design to get a second order controlled system in the form (2.16)-(2.17) that tracks a

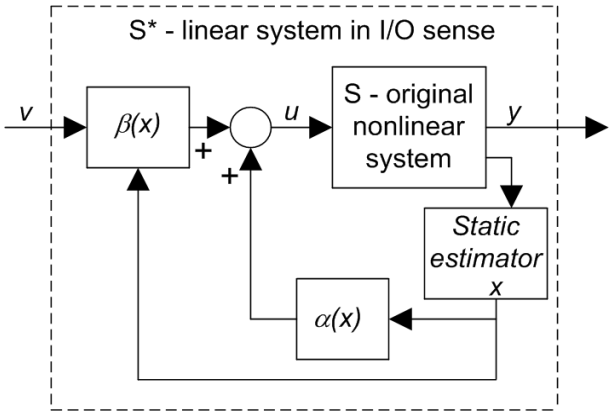

Figure 4.1: The input-output linearized plant

piecewise constant reference signal. However, the change of model parameters and/or environmental disturbances highly influences the input-output linearization (this is the model matching problem discussed in Section 4.1.1), resulting the occurrence of additional nonlinear terms in the linearized plant in (4.7),(4.9). These additional nonlinearities then spoil reference tracking (the controlled plant tracks the reference signal with a large offset).

Therefore (4.7)-(4.9) is extended with another differential equation for the tracking error proposed in (2.18):

$$\frac{d}{dt}e = \bar{v}_{ref} - \bar{y} = \bar{v}_{ref} - \bar{z}_1 \tag{4.10}$$

where v_{ref} is the reference signal to be tracked, $\bar{v}_{ref} = v_{ref} - y^*$ is its centered version, and e is the tracking error, which is zero at steady state, meaning that $y = v_{ref}$ (at steady state). Observe that the system defined by (4.7), (4.9) and (4.10) is an LTI system with two input variables: the centered external reference signal ($\bar{v}_{ref}$) and the stabilizing feedback $\bar{v}$:

$$\begin{aligned}
\frac{d}{dt}\begin{bmatrix} e \\ \bar{z}_1 \\ \bar{z}_2 \end{bmatrix} &= \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} e \\ \bar{z}_1 \\ \bar{z}_2 \end{bmatrix} + \begin{bmatrix} 0 & 1 \\ 0 & 0 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} \bar{v} \\ \bar{v}_{ref} \end{bmatrix} \\
\bar{y} &= \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} e \\ \bar{z}_1 \\ \bar{z}_2 \end{bmatrix}
\end{aligned}$$

In order to stabilize the system, we apply an LQ controller with state ($Q \in \mathbb{R}^{3\times3}$) and input ($R \in \mathbb{R}^{1\times1}$) weighting matrices of the cost functional (2.8) in the following way: we fix R to the unit value, since only the relationship between the magnitude of elements of Q and R is that matters. In Q, we neglect the cross-effects between state variables that results in a diagonal Q. Thus, we have three parameters to tune the controller, with respect to the control aims.

The tuning of the controller (i.e. the appropriate choice of these three parameters) has been performed via simulations. First, the elements of Q has been tuned to

reach the appropriate settling time. Then, the magnitude of the elements of Q are increased in order to guarantee robustness against model parameter uncertainties and environmental disturbances. The tuning of the controller resulted the weighting matrices

$$Q = \begin{bmatrix} 5 \cdot 10^9 & 0 & 0 \\ 0 & 1.5 \cdot 10^9 & 0 \\ 0 & 0 & 10^3 \end{bmatrix}, \quad R = \begin{bmatrix} 1 \end{bmatrix} \tag{4.11}$$

The solution of the corresponding Control Algebraic Ricatti Equation (2.9) is computed in MATLAB, as well as the joint stabilizing feedback gain K (2.10), therefore the LQ state feedback is in the following form:

$$\bar{v} = -K \cdot \begin{bmatrix} e \\ \bar{z}_1 \\ \bar{z}_2 \end{bmatrix}$$

where

$$K = \begin{bmatrix} -70711 & 39241 & 281.93 \end{bmatrix}.$$

With this stabilizing feedback, the state space model of the input-output linearized model with the centered servo control input $\bar{v}_{ref}$ is the following:

$$\frac{d}{dt}\begin{bmatrix} e \\ \bar{z}_1 \\ \bar{z}_2 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & 1 \\ 70711 & -39241 & -281.93 \end{bmatrix}\begin{bmatrix} e \\ \bar{z}_1 \\ \bar{z}_2 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}\bar{v}_{ref} \tag{4.12}$$

$$y = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} e \\ \bar{z}_1 \\ \bar{z}_2 \end{bmatrix} \tag{4.13}$$

The signal flow diagram of the controlled plant is depicted in Fig. 4.2.

It is known from [49] that the asymptotic stability of (4.12) and the asymptotic stability of (4.8) guarantees the asymptotic stability of the whole plant (4.8),(4.12). It is easy to see that at steady state (i.e. when $e = \frac{de}{dt} = 0$) the zero dynamics in (3.44) and the zero dynamics of the linearized model with servo controller is identical. But in our case, the rotational speed tracks a reference signal, therefore the local asymptotic stability of zero dynamics with arbitrary x_3 and $\dot{x}_3$ values would guarantee the stability of the servo controlled plant. This hypothesis is not proved analytically but checked at a lot of *steady state* points (near several constant x_3 and M_{load} values) in Section 3.3 and found to be locally asymptotically stable for all given reference input u^*. However, this hypothesis could not be checked in non-steady state cases (i.e. when $\dot{x}_3$ - or equivalently $\bar{z}_2$ - is not equal to zero).

4.3.4 Simulation results

The MATLAB/SIMULINK model of the simplified gas turbine model with the input-output linearizing, servo and stabilizing feedback has also been built in order to test the control method. In simulations we used the built-in 'ode45' differential

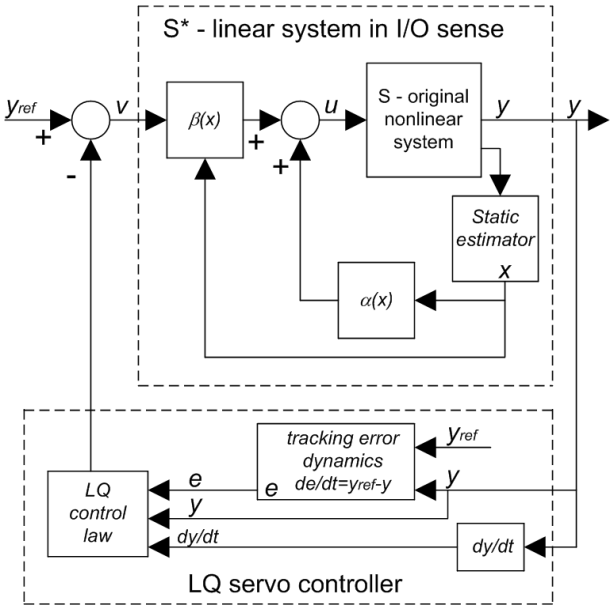

Figure 4.2: LQ servo controller on the I/O linearized plant

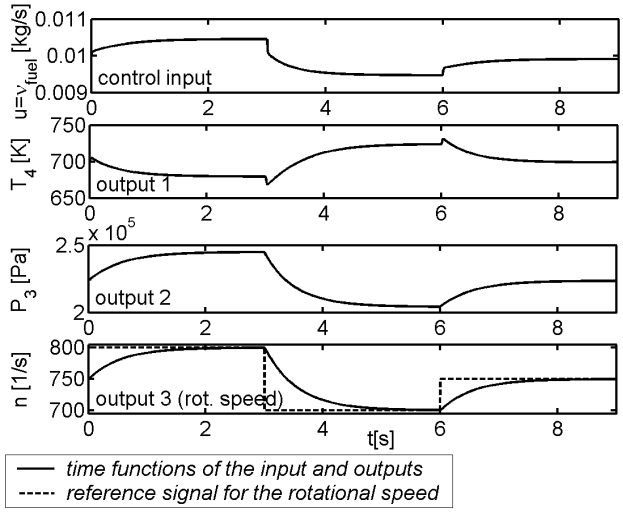

Figure 4.3: Reference signal tracking for the rotational speed

equation solver, which is based on a Runge-Kutta (4,5) formula [30] with variable step size (between 10^{-6} s and 10^{-1} s).

In Fig. 4.3, the subfigures show the time function of the control input and measured output variables - ν_{fuel}, T_4, p_3 and n respectively -, near nominal parameter and typical environmental disturbance values (i.e. using the nominal model). The rotational speed (solid line in 4^{th} subfigure) is started from $n = 750$ $1/s$ and tracks a piecewise constant reference signal (dashed line in 4^{th} subfigure). The settling time of the transient is about $t_s = 1.7$ s. Simulations have showed that the controlled plant is asymptotically stable and tracks the reference signal with linear transients.

Robustness

Simulations have also showed that the plant is robust against not just the change of all environmental disturbance variables (in vector d), but also against uncertain model parameters.

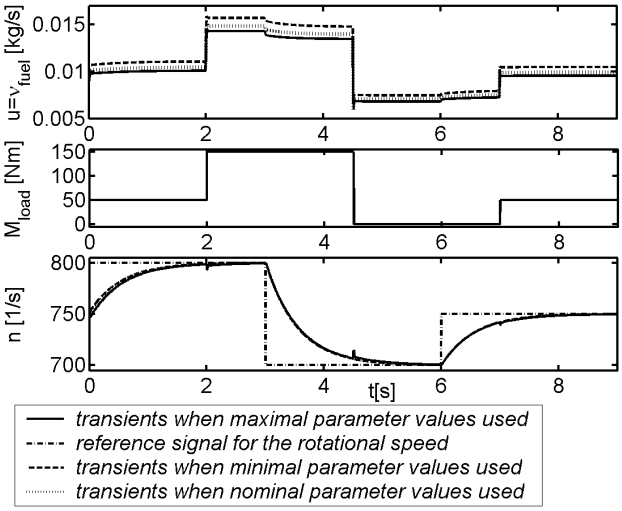

Figure 4.4: Robustness of the LQ servo controller (curves overlap in subfigures 2 and 3)

Figure 4.4 shows the reference tracking of the plant when those three model parameters which have a significant effect on the dynamical behavior of the plant ($V_{Comb}, \Theta, \eta_{comb}$) together with the first two environmental disturbances (p_1 and T_1) are uniformly set to their maximal/minimal/nominal values and the most important environmental disturbance - the load torque (2^{nd} subfigure) - is changed in three steps. (The minimal and maximal values of the environmental disturbances and of the model parameters can be found in (3.27)-(3.29) and (3.30)-(3.32), respectively, while the typical values of the environmental disturbances are given in (3.24)-(3.26), and the nominal values of model parameters are contained in Table A.1 in Appendix A.)

The system trajectories are started from the same initial state (with $n = 750\ 1/s$) as before. The trajectories of the rotational speed corresponding to the maximal/ minimal/ nominal values of uncertain model parameters and environmental disturbances are denoted by solid/dashed/dotted lines, respectively in the 3^{rd} subfigure. The same notations are used for the related control input functions in the 1^{st} subfigure. The change of the reference signal is the same as in the previous simulation and it is successfully tracked. At $t = 0$, the load torque is set to its typical value: $M_{load} = 50\ Nm$, and at $t = 2s$, $4.5s$, $7s$ step-like changes are performed on it. The effect of model parameter uncertainties and the environmental disturbances p_1 and T_1 can be observed in the beginning of the time function of the rotational speed, showing that the LQ servo controller successfully suppresses their influence: the rotational speed trajectories overlap (3^{rd} subfigure), the only slight difference is between the sufficient control inputs (1^{st} subfigure). The change of load torque

(as external disturbance) causes little drops/overshoots in the rotational speed, but the controlled plant turns back to the prescribed reference value in a very short time. Note that the control input never reaches its saturation bounds (0 kg/s and 0.03 kg/s).

4.4 MPT controller design based on the LQ controlled plant

In this section, a constrained linear optimal controller is designed to the input-output linearized and LQ-stabilized plant. As we will see, the control aims are quite different from the previous one, since the controlled plant is to be used to generate electric power. Again, it is assumed that the time function of the load torque is known, therefore the input-output linearized plant in Section 4.3.2 is used to design the LQ and MPT controllers thereon.

4.4.1 Control problems and aims

In this case an industrial low-power gas turbine drives an electric generator. Some important properties and parameters of the construction are the following:

- The generator requires 50 $Hz = 50\ 1/s$ constant rotational speed to generate the regular alternating current (A.C.). Between the gas turbine and the generator there is a gearbox which has fix 3/50 deceleration gears. Because of this fact the *first control aim* has to be formulated: the rotational speed of the gas turbine has to be constant: $n = x_3 = 833.33\ 1/s$.

- The total pressure and temperature in the compressor inlet (d_1 and d_2, respectively) are correctly measurable quantities of the ambience, and the load torque (d_3) is proportional to the effective power of the system. Therefore all disturbance variables are measurable. The *second control aim* is that the rotational speed should be robust against environmental disturbances and model parameter uncertainties.

- The *third control aim* is to defend the actuator against saturation, i.e.

$$0\ kg/s \leq u = \nu_{fuel} \leq 0.03\ kg/s\ .$$

- The *fourth control aim*: we have some additional constraints on the state variables. The operating domain of the rotational speed is modified:

$$828.33\ 1/s \leq n \leq 838.33\ 1/s\ .$$

  In order to protect the gas turbine against too high temperature and pressure values, the time derivative of the rotational speed (which is the state $\bar{z}_2$ of the input-output linearized system) has to be bounded:

$$-500\ 1/s^2\ \leq \dot{n} \leq 500\ 1/s^2\ .$$

Note that the slight increase of the upper bound of the operating domain of the rotational speed from 833.33 $1/s$ to 838.33 $1/s$ is not crucial from the viewpoint of the behavior of the gas turbine model, however it is compulsory because of the change of environmental disturbances, mainly the change of the load torque.

4.4.2 Design of the LQ-MPT controller

First, an LQ controller is designed to the input-output linearized plant (4.7)-(4.9). Since the rotational speed is to be held at the constant value 833.33 $1/s$, it would be enough to design an LQ (but not servo) controller thereon. However, because of the model matching problem, an LQ servo controller is needed again with the tracking error e, and a constant reference input $v_{ref} = 833.33\ 1/s$.

In the design of this LQ servo controller, the weighting matrices (4.11) of the previous LQ servo controller designed in Section 4.3.3 can be re-used. As a consequence, the only difference occurs between the input terms of the controlled systems: since the servo control input is set to a constant value ($v_{ref} = 833.33$) its centered version $\bar{v}_{ref}$ is equal to zero, therefore the associated input term is left out from (4.12)-(4.13), and substituted by an additional input term v_{MPT} which is the control input to be supplied by the constrained linear optimal controller:

$$\frac{d}{dt}\begin{bmatrix} e \\ \bar{z}_1 \\ \bar{z}_2 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & 1 \\ 70711 & -39241 & -281.93 \end{bmatrix}\begin{bmatrix} e \\ \bar{z}_1 \\ \bar{z}_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} v_{MPT} \quad (4.14)$$

$$y = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} e \\ \bar{z}_1 \\ \bar{z}_2 \end{bmatrix} \quad (4.15)$$

The MPT controller is needed to satisfy the fourth control aim. The implementation of the constrained linear optimal controller requires time-discretization of the LQ controlled plant which has been done by zero order hold sampling [44], with sample time 0.01 s.

The MPT controller solves the Constrained Finite Time Optimal Control Problem with the quadratic cost defined in (2.22). The prediction horizon of (2.22) is set to $N = 3$, while its weighting matrices $Q_c \in \mathbb{R}^{3\times3}$, $R_c \in \mathbb{R}^{1\times1}$ and $P_N \in \mathbb{R}^{3\times3}$ are chosen as follows:

$$Q_c = \begin{bmatrix} 10^{10} & 0 & 0 \\ 0 & 10^{15} & 0 \\ 0 & 0 & 10^{10} \end{bmatrix}, \quad R_c = \begin{bmatrix} 1 \end{bmatrix}, \quad P_N = 0$$

For the numerical solution of the constrained linear optimal control problem, the Multi-Parametric Toolbox (MPT) [61] of the MATLAB computational environment is used.

The designed control configuration is shown in Fig. 4.5.

4.4.3 Simulation results

The MATLAB/SIMULINK model of the gas turbine model with input-output linearizing, LQ servo and constrained linear optimal controller has been used for the

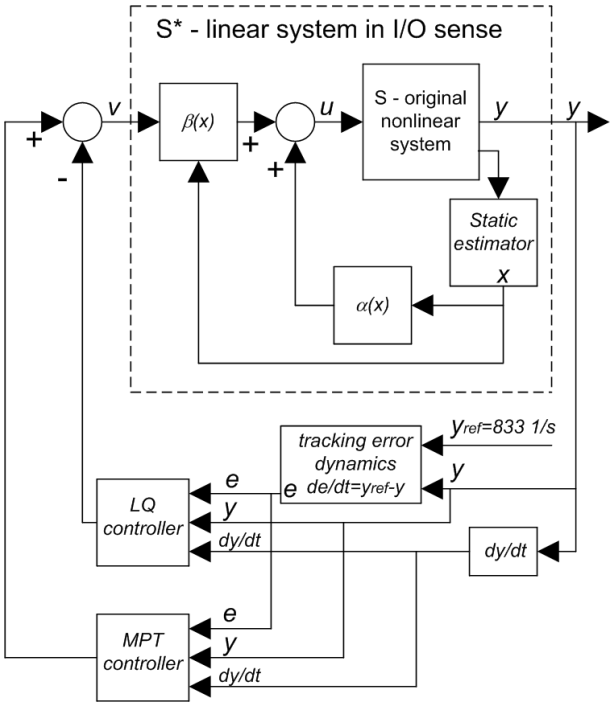

Figure 4.5: LQ and MPT controllers on the I/O linearized plant

simulation investigations. Simulation results have showed that the closed-loop system is robust against the change of disturbance variables.

Fig. 4.6 shows the change of the rotational speed (2^{nd} subfigure) and the change of the time derivative of the rotational speed (3^{rd} subfigure) in the case when the load torque is changed between 50 Nm and 150 Nm (4^{th} subfigure). The initial value of the rotational speed is the desired 833.33 $1/s$. Between $t = 1$ s and 10 s step-like changes are performed on the load torque. As it is shown in the 2^{nd} subfigure, these changes have no significant effect on the rotational speed.

Observe that the rotational speed and its time derivative never reach their minimal and maximal values: they stay within the predefined operation domain, so our constraints are fulfilled. Moreover, the controller successfully defends the actuator against saturation (see the control input ν_{fuel} in the 1^{st} subfigure).

Note that if the time function of the load torque does not contain sudden changes, the MPT controller can always compensate its effect on the time-derivative of the rotational speed ($\dot{n}$). However, since the control input supplied by the MPT controller appears in the differential equation of $\dot{n}$, non-smooth changes (steps) in the load torque have to have a maximal magnitude of 20 Nm to fulfill the constraint on $\dot{n}$.

Also note that 'worst case' simulations have also been performed to investigate robustness, and the controlled plant proved to be robust against environmental disturbances and model parameter uncertainties in all cases.

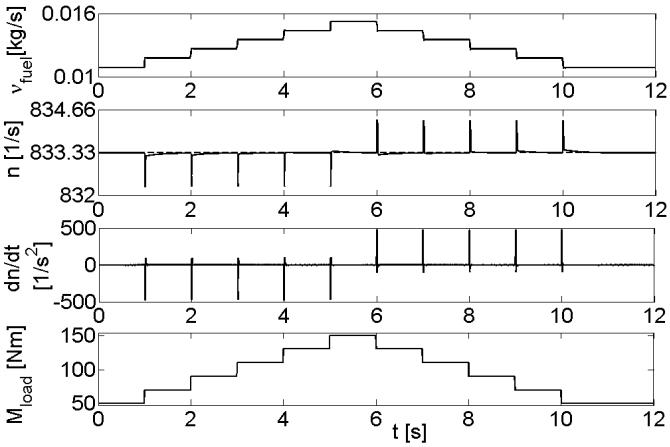

Figure 4.6: The effect of step-like changes of load torque on system variables

4.5 Design of an LQ servo controller with adaptive load torque estimation

In the previous sections it has been assumed that the time function of the load torque is known. This section relaxes this assumption and deals with the more realistic case when the load torque has to be estimated. For this purpose, an appropriate controller structure is designed, which is based on adaptive input-output linearization.

4.5.1 Control problems and aims

Since this controller will be designed to regulate a gas turbine that is a prime mover of a transportation system, the control aims are similar to the controller designed in Section 4.3:

- The *first control aim* is that the rotational speed should follow a piecewise constant reference signal with a settling time between 1 s and 2 s;
- the *second control aim* is that the plant has to be robust against environmental disturbances and model parameter uncertainties;
- the *third control aim* is to avoid the saturation of the actuator:

$$0\ kg/s\ \leq u = \nu_{fuel} \leq 0.03\ kg/s\ ;$$

- while the *fourth control aim* gives the novelty of the controller: since the time evolution of the rotational speed is unknown, the controller should estimate it.

4.5.2 Input-output linearization with load torque estimation

Until now it has been assumed that the external disturbances are known exactly. The first two elements of the disturbance vector ($d_1 = p_1$ and $d_2 = T_1$) are assumed

to be measurable and constant, so the only difficulty is with the computation of the load torque ($d_3 = M_{load}$). This problem is solved by designing an adaptive control law that uses the estimation of d_3.

Let us denote the typical value of d_3 in (3.26) by d_3^{typ}. Then the load torque can be rewritten as:

$$d_3 = d_3^{typ} + \mu$$

where μ is the deviance of d_3 from its typical value. Assume now that d_3 (and therefore μ) is a constant. Using the fact that M_{load} appears additively in (3.13), we can write the derivative of $y = h_{ZD}(x) = x_3$ as

$$\dot{y} = \dot{x}_3 = f_3(x, d_3^{typ}) + l_1\mu$$

where

$$l_1 = -\frac{3}{100\pi\Theta}$$

and

$$\begin{aligned} f_3(x, d_3^{typ}) &= \frac{1}{4\pi^2\Theta n}\left(\nu_T c_p \frac{p_3 V_{Comb}}{m_{Comb} R}\eta_T\eta_{mech}\Big(1 - \Big(\frac{p_1}{p_3\sigma_I\sigma_N}\Big)^{\frac{\kappa-1}{\kappa}}\Big)\right. \\ &- \left.\nu_C c_p \frac{T_1}{\eta_C}\Big(\Big(\frac{p_3}{p_1\sigma_{Comb}}\Big)^{\frac{\kappa-1}{\kappa}} - 1\Big)\right) - \frac{3}{100\pi\Theta} d_3^{typ} \end{aligned}$$

with the algebraic constraints (3.14)-(3.15) and the definition of x in (3.21). The second time-derivative of y is:

$$\ddot{y} = \sum_{i=1}^{3} \frac{\partial f_3(x,d) + l_1\mu}{\partial x_i}\Big(f_i(x,d) + g_i(x)u\Big) + \frac{\partial f_3(x,d) + l_1\mu}{\partial x_3} l_1\mu + l_1\dot{\mu}$$

where f_i and g_i are the coordinate functions of f and g respectively, as defined in (3.16). Using that μ is a constant, this equation becomes

$$\begin{aligned} \ddot{y} &= \sum_{i=1}^{3} \frac{\partial f_3(x,d)}{\partial x_i}\Big(f_i(x,d) + g_i(x)u\Big) + \frac{\partial f_3(x,d)}{\partial x_3} l_1\mu = \\ &= L_f^2 h_{ZD}(x) + L_g L_f h_{ZD}(x)u + \psi(x, d^{typ})\mu \quad , \quad \psi(x, d^{typ}) = l_1 \frac{\partial f_3(x, d^{typ})}{\partial x_3} \end{aligned}$$

where d^{typ} denotes the disturbance vector with its typical values (3.24)-(3.26). Applying the feedback (4.1)-(4.2) and using the notations in (4.3)-(4.6) the system model reads

$$\dot{\bar{z}}_1 = \bar{z}_2 \tag{4.16}$$

$$\dot{\bar{z}}_2 = \psi(x, d^{typ})\mu + \bar{v} \tag{4.17}$$

$$\dot{x}_2 = \Phi(z_1, z_2, x_2) \tag{4.18}$$

$$\bar{y} = \bar{z}_1 \tag{4.19}$$

Comparing this model to the model (4.7)-(4.9), the only difference is the nonlinear term in the second differential equation. To cancel the effect of this nonlinearity, an adaptive controller is designed that estimates the value of μ.

Observe that the model (4.16)-(4.17) can be written in the following form:

$$\dot{\bar{z}} = \widetilde{f}(\bar{z}) + \widetilde{q}(x, d^{typ})\mu + \widetilde{g}(\bar{z})\bar{w}$$

where

$$\widetilde{f}(\bar{z}) = \begin{bmatrix} \bar{z}_2 \\ 0 \end{bmatrix} \ , \ \widetilde{g}(\bar{z}) = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \ , \ \widetilde{q}(x, d^{typ}) = \begin{bmatrix} 0 \\ \psi(x, d^{typ}) \end{bmatrix}.$$

Since this model together with the output equation (4.19) is in the form (2.30)-(2.31), the Adaptive Feedback Linearization Theorem (AFLT) in Section 2.1.6, in Paragraph *Adaptive feedback linearization* can be applied to it, which will serve as a theoretical basis for the controller design.

It is important to note that this theorem will be applied to the model (4.16)-(4.19) instead of the open loop gas turbine model. Its is easy to see that the feedback linearization of the model (4.16)-(4.19) is equivalent to the input-output linearization of the model defined by (4.16)-(4.19) and the additional zero dynamics (4.8). Also note that although the feedback that will be computed feedback linearizes (4.16)-(4.19) **globally**, this feedback will only be applied to the gas turbine model inside its operating domain $\mathcal{X}$.

Since the nominal system (i.e. (4.16)-(4.19) with $\mu = 0$) is globally feedback linearizable, the first condition of AFLT is satisfied. Furthermore, the dimension of the model (4.16)-(4.19) is $\widetilde{n} = 2$, which means that only $[\widetilde{q}, \widetilde{g}] \subset G_0$ has to be checked, since $G_0 = span\{\widetilde{g}\}$:

$$[\widetilde{q}, \widetilde{g}] = \frac{\partial \widetilde{g}}{\partial \bar{z}}\widetilde{q} - \frac{\partial \widetilde{q}}{\partial \bar{z}}\widetilde{g} = -\begin{bmatrix} 0 & 0 \\ \frac{\partial \psi}{\partial \bar{z}_1} & \frac{\partial \psi}{\partial \bar{z}_2} \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ -\frac{\partial \psi}{\partial \bar{z}_2} \end{bmatrix} \subset span\left\{ \begin{bmatrix} 0 \\ 1 \end{bmatrix} \right\} = G_0$$

Since both conditions are satisfied, the adaptive feedback linearizing control can be computed in the following way (see pages 119-120. in [71]). Define a reference model

$$\begin{bmatrix} \dot{\bar{z}}_{r1} \\ \dot{\bar{z}}_{r2} \end{bmatrix} = A_r \begin{bmatrix} \bar{z}_{r1} \\ \bar{z}_{r2} \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} v_r \quad , \quad A_r = \begin{bmatrix} 0 & 1 \\ -k_1 & -k_2 \end{bmatrix}$$

where $k_1, \ k_2$ are chosen in such a way that A_r is a stability matrix (i.e. it has eigenvalues with strictly negative real parts). Denote the estimated value of μ by $\widehat{\mu}$ and the estimation error by $\Delta\mu$:

$$\Delta\mu = \mu - \widehat{\mu} \tag{4.20}$$

Define the following control input function:

$$\bar{v} = -\psi(x, d^{typ})\widehat{\mu} - k_1\bar{z}_1 - k_2\bar{z}_2 + v_r \tag{4.21}$$

where v_r is the common input variable of both the controlled system model and the reference model, and substitute it to the state equations (4.16)-(4.17):

$$\dot{\bar{z}}_1 = \bar{z}_2 \tag{4.22}$$

$$\dot{\bar{z}}_2 = -k_1\bar{z}_1 - k_2\bar{z}_2 + \psi(x, d^{typ})\Delta\mu + v_r \tag{4.23}$$

Define the reference error e as

$$e = \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} = \begin{bmatrix} \bar{z}_1 \\ \bar{z}_2 \end{bmatrix} - \begin{bmatrix} \bar{z}_{r1} \\ \bar{z}_{r2} \end{bmatrix}$$

Then the reference error dynamics reads

$$\begin{bmatrix} \dot{e}_1 \\ \dot{e}_2 \end{bmatrix} = A_r \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \psi(x, d^{typ}) \end{bmatrix} \Delta\mu$$

where the dynamics of μ is not determined yet. Let P be the positive definite solution of the Lyapunov equation

$$A_r^T P + P A_r = -I$$

where $I \in \mathbb{R}^{2\times 2}$ is the identity matrix. Consider the following positive definite Lyapunov function candidate:

$$V = e^T P e + \gamma \Delta\mu^2 \ , \quad \gamma \in \mathbb{R}^+$$

The time derivative of V is given by

$$\frac{d}{dt}V = -e_1^2 - e_2^2 + 2 \begin{bmatrix} e_1 & e_2 \end{bmatrix} P \begin{bmatrix} 0 \\ \psi(x, d^{typ}) \end{bmatrix} \Delta\mu + \frac{2}{\gamma}\Delta\mu \frac{d}{dt}\Delta\mu \ .$$

By choosing the following adaptation error dynamics

$$\frac{d}{dt}\Delta\mu = -\gamma \begin{bmatrix} e_1 & e_2 \end{bmatrix} P \begin{bmatrix} 0 \\ \psi(x, d^{typ}) \end{bmatrix},$$

the time derivative of V becomes

$$\dot{V} = -e_1^2 - e_2^2$$

which is negative definite, therefore V is a Lyapunov function and the adaptation error asymptotically converges to zero.

Differentiating (4.20) by time and using that μ is a constant, the adaptation law can be determined from the adaptation error dynamics:

$$\frac{d}{dt}\widehat{\mu} = \gamma \begin{bmatrix} e_1 & e_2 \end{bmatrix} P \begin{bmatrix} 0 \\ \psi(x, d^{typ}) \end{bmatrix} = -\frac{d}{dt}\Delta\mu \tag{4.24}$$

Thus, the controller designed to (4.16)-(4.17) with the control input (4.21) and adaptation law (4.24) successfully performs the adaptive feedback linearization and stabilization of (4.16)-(4.17), moreover, it gives an (asymptotically converging) estimation $\widehat{\mu}$ of the unknown parameter μ.

4.5.3 Servo controller with stabilizing feedback

Our aim is to build a controller that tracks the reference signal y_{ref} that is our prescribed value for the rotational speed. For this purpose, an LQ servo controller

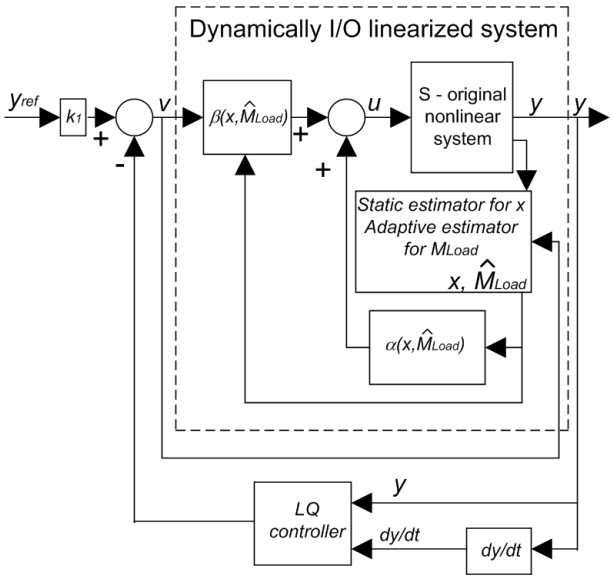

Figure 4.7: LQ servo controller on the adaptively I/O linearized plant

is designed to the adaptively input-output linearized plant. The structure of the controlled plant is shown in Fig. 4.7.

Observe that in the control input defined in (4.21), the parameters k_1 and k_2 have not been determined yet. These parameters can be computed as the result of a simple LQ design to the double integrator model (4.7). Additionally, by choosing $v_r = k_1\bar{y}_{ref}$, an LQ servo controller is designed, and therefore the controlled plant will track a prescribed piecewise constant reference signal. Since $\lim_{t\to\infty}\Delta\mu = 0$, the only steady state operating point of (4.22)-(4.23) is ($\bar{z}_1 = \bar{y}_{ref},\ \bar{z}_2 = 0$) which is unique, and - because of the LQ servo controller - it is asymptotically stable.

The tuning parameters of the controller are the positive definite state and input weighting matrices ($Q \in \mathbb{R}^{2\times 2}$ and $R \in \mathbb{R}^{1\times 1}$, respectively), and the adaptation gain $\gamma \in \mathbb{R}^+$ in (4.24). Let $R = [1]$ be fixed. For the sake of simplicity, Q is restricted to be diagonal. Thus, we have three scalar parameters (the two diagonal elements of Q and the adaptation gain γ) to tune the controller according to the control goals.

The tuning of parameters has been successfully performed via simulations in MATLAB/SIMULINK using piecewise constant reference signals, and 'worst case' disturbances/uncertainties. First, the relative magnitude of the two diagonal elements of Q has been tuned according to the prescribed settling time. Then the magnitude of the elements of Q with respect to R, and - simultaneously - the adaptation gain γ have been determined according to the robustness of the controlled plant. The resulted design parameters are:

$$Q = \begin{bmatrix} 3\times 10^5 & 0 \\ 0 & 1.5\times 10^5 \end{bmatrix}, \quad R = \begin{bmatrix} 1 \end{bmatrix}, \quad \gamma = 25$$

It is important to note that time-varying parameters are also allowed, if they can be modelled by an exosystem in the form (2.35). Since only such $\mu(t)$ functions will be applied during simulations that are with $\Omega = 0$ in (2.35) - therefore $\Omega^T +$

$\Omega = 0$ is indeed negative semidefinite -, moreover $\mu(t)$ is always chosen bounded independently of the values of x, the conditions on (2.35) are fulfilled.

4.5.4 Simulation results

Simulations have been performed again in the MATLAB/SIMULINK software environment. Simulations on the nominal model (i.e. with typical disturbance and nominal parameter values that can be found in (3.24)-(3.26) and in Table A.1 in Appendix A, respectively) have showed that the controlled nominal plant asymptotically tracks a piecewise reference signal with linear transients and with settling time about $t_s = 1.7\ s$.

The operation of the adaptive controller is shown in Fig. 4.8. The 2^{nd} subfigure shows the estimation of the load torque. The time function of the load torque (dashed line) consists of linear and constant pieces, that are followed well by the estimated value (solid line). The estimation contains some little drops/overshoots at time instances $2, 4, 5.5, 7\ s$ caused by the fast changes in the load torque function. The 3^{rd} subfigure shows the reference tracking for the rotational speed. This reference signal is successfully tracked, and large changes in the load torque cause only transients of small magnitude in the rotational speed. The 1^{st} subfigure shows the related control input function. It is important to mention that the drops/overshoots in the estimation of the load torque does not appear nor in the rotational speed nor in the control input.

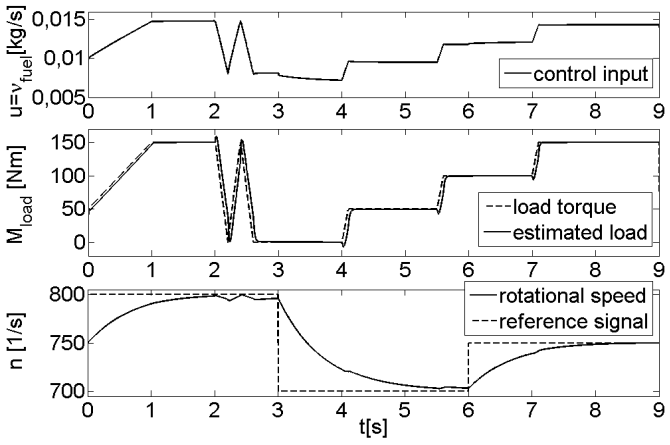

Figure 4.8: Adaptive estimation of the load torque

4.5.5 Robustness

To test the proposed control scheme under more realistic circumstances, we now relax the original assumptions that all the model parameters and disturbances (with the exception of the load torque) are known.

Figure 4.9 demonstrates the 'worst case' behavior of the plant against model parameter uncertainties and environmental disturbances: This simulation shows

the reference tracking when three model parameters having a significant effect on the dynamical behavior of the controlled plant are uniformly set to their maximal/minimal/nominal values together with the first two environmental disturbances: the turbine inlet total pressure and temperature ($d_1 = p_1$ and $d_2 = T_1$, respectively). (The minimal and maximal values of the environmental disturbances and of the model parameters can be found in (3.27)-(3.29) and (3.30)-(3.32), respectively, while the typical values of the environmental disturbances are given in (3.24)-(3.26), and the nominal values of model parameters are contained in Table A.1 in Appendix A.) The load torque (2^{nd} subfigure) is simultaneously set to its minimal and maximal value (0 Nm and 150 Nm, respectively).

The prescribed rotational speed trajectory (dash-dot line in the 3^{rd} subfigure) differs from the former ones in order to show not just the transient, but also the steady state behavior (between 3 s and 6.5 s) of the plant in a more realistic environment.

The trajectories of the rotational speed corresponding to the maximal/ minimal/ nominal values of uncertain model parameters are denoted by solid/dashed/dotted lines, respectively in the 3^{rd} subfigure. The same line styles are used for the related control input functions in the 1^{st} subfigure. The effect of model parameter uncertainties can be observed in the beginning of the time function of the rotational speed, showing that the LQ servo controller successfully suppresses their influence: the rotational speed trajectories overlap (3^{rd} subfigure), the only slight difference is between the related control input functions (1^{st} subfigure). Although the changes of the load torque (as external disturbance) are non-smooth with the possible largest magnitude, they cause only tiny drops/overshoots in the time function of the rotational speed, and the controlled plant turns back to the prescribed reference value in a very short time. It is also visible that the control input is far from saturation since

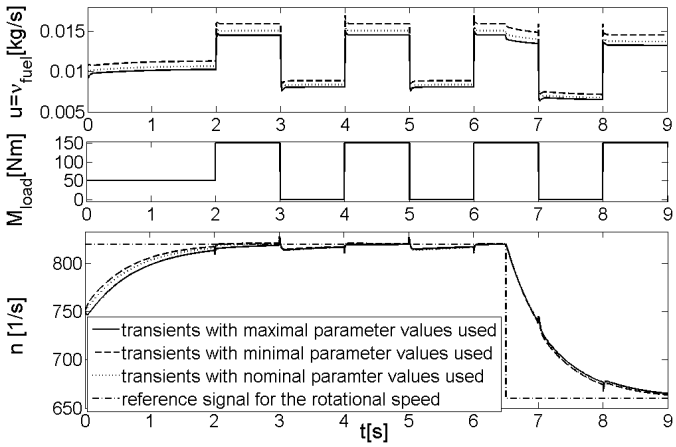

Figure 4.9: Robustness of the adaptive LQ servo controller I. (curves overlap in subfigures 2 and 3)

it is between 0.005 kg/s and 0.017 kg/s, while the saturation bounds are 0 kg/s and 0.03 kg/s.

Another case study is performed and illustrated in Fig. 4.10 in order to show

the effect of environmental disturbances not just on reference tracking, but also on the adaptive estimation of the load torque. For this purpose, the turbine inlet total pressure and temperature are set to their extremal values. Four different simulations are reported here: with minimal p_1 and T_1 values (denoted by dotted line), with minimal p_1 and maximal T_1, and with maximal p_1 and minimal T_1 values (both denoted by dashed lines), and with maximal p_1 and T_1 values (denoted by solid line). The only significant difference is between the control input functions (1^{st} subfigure).

The time-function of the load torque consists of linear and constant pieces (dash-dot line in the 2^{nd} subfigure). Large drops/overshoots in the estimation of the load torque only occur between 1 s and 8 s, caused by step-like changes in the load torque. It is also shown that there is no significant difference between the estimations of the load torque belonging to different extrema of p_1 and T_1 (these four curves overlap).

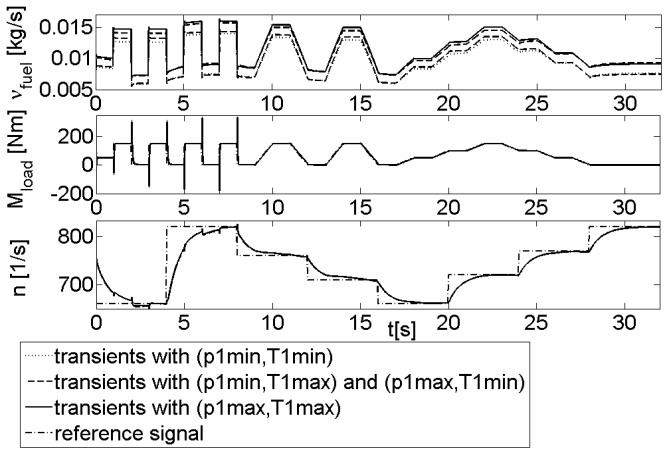

Figure 4.10: Robustness of the adaptive LQ servo controller II. (curves overlap in subfigures 2 and 3)

The reference signal for the rotational speed (dash-dot line in the 3^{rd} subfigure) is a staircase-like piecewise constant function which is successfully tracked. The little drops/overshoots between 1 s and 8 s are caused by the sudden changes of the highest magnitude in the time function of the load torque. The time functions of the rotational speed belonging to different extrema of p_1 and T_1 overlap during the whole simulation. Note that the control input is far from saturation just like in the former simulation studies.

4.6 Summary

In this chapter, the rotational speed of the gas turbine has been chosen to be controlled via input-output linearization based linear controllers. Three different controllers have been built in order to fulfill different requirements:

(a) An LQ servo controller is designed to track a piecewise constant reference signal for the rotational speed, with known time-function of the load torque;

(b) an LQ+MPT controller is designed to keep the rotational speed on a constant value, and also keep the rotational speed and its time derivative between bounded values, with known time-function of the load torque;

(c) a novel adaptive LQ servo controller is designed to track a piecewise constant reference signal for the rotational speed, with *unknown* time function of the load torque.

The tuning of controllers has been performed in the MATLAB/SIMULINK computation environment, and their time-behavior has been tested via worst-case simulations. All three controllers have proven to guarantee the robustness of the controlled plant against model parameter uncertainties and environmental disturbances.

However, the main difference between these controllers is indicated by their applicability under realistic circumstances. In cases (a) and (b) the time function of the load torque has been assumed to be known or directly measurable, and therefore it could be used directly in the computation of the input-output linearizing feedback.

Unfortunately, in most cases the load torque - which is the most versatile environmental disturbance that has the largest impact on the time behavior of the gas turbine - is an unmeasurable disturbance of the environment. The advanced controller (c) handles this more realistic case with a novel approach that has not been yet applied to gas turbines: the load torque is estimated by a dynamic feedback supplied by a state space based adaptive controller. Exploiting the fact that the load torque occurs as an additive disturbance in the gas turbine model, *adaptive* input-output linearization is applied: the dynamically estimated value of the load torque is used in the input-output linearizing feedback.

Worst case simulation experiments in MATLAB/SIMULINK have showed that the controller properly estimates the time function of the load torque, moreover the reference tracking for the rotational speed is robust against all environmental disturbances and model parameter uncertainties. An important observation is that in spite that the load torque is only estimated in this case, the controlled plant shows similar robustness properties than in case (a).

Chapter 5

Determining invariants (first integrals) of QP-ODEs

As it has been discussed in Chapter 2.1.5, the majority of methods for finding invariants (hidden algebraic relationships) of ODEs concern with narrow model classes, or contain non-algorithmizable (heuristical) steps. The purpose of this chapter is to introduce a new numerically effective method for finding QP type invariants of QP-ODEs that meets the conditions of representability and algorithmizability, and additionally provides a feasible alternative of the QPSI method discussed in Section 2.3.6 by its simplicity and numerical approach. The implementation of the algorithm in the MATLAB numerical computational software environment is also presented. The algebraic properties (i.e. invariance under transformations of different type) are also discussed to determine the applicability of the new method. The operation of the algorithm is demonstrated on several physical and a numerical example.

5.1 An algorithm for determining a class of invariants in QP-ODEs

In the following, an algorithm will be presented which is capable to retrieve a class of single QP-type invariants of QP-ODEs described in Section 5.1.1. With a slight modification, another algorithm is given which is able to retrieve multiple first integrals from a QP-ODE model. If the QP-ODE is non-minimal *in QP sense*, the latter algorithm can determine QP-type first integrals, i.e. the result will be a QP-DAE in the form of (2.72)-(2.75). Then, the MATLAB implementation and computational properties of these algorithms are discussed.

5.1.1 The examined class of invariants

Consider the state equation (2.61) of the general input-affine QP-ODE state space model. By regarding the inputs $u_k, k = 1, \ldots, p$ of (2.61) as *arbitrary unknown parameters*, this equation can be re-written in the form of the homogeneous QP-ODE (2.51).

To find the input-independent quasi-polynomial invariants of (2.61) we will concern with the quasi-polynomial invariants of the homogeneous QP-ODE (2.51) - which is exactly (2.61) with parameter inputs - that can be written in the following special form:

$$I = F(x) - x_i^{\frac{1}{\beta}}, \quad \beta \in \mathbb{R} \tag{5.1}$$

where

$$F(x) = \sum_{k=1}^{p} c_k \prod_{j=1, j\neq i}^{n} x_j^{\alpha_{kj}}, \quad c_k, \alpha_{kj} \in \mathbb{R} \tag{5.2}$$

It's clear that (5.1) can be rewritten as

$$x_i^{\frac{1}{\beta}} = F(x) + c_0, \quad c_0 \in \mathbb{R} \tag{5.3}$$

This is a narrower class of invariants than the one examined in [33] since it contains those first integrals from where at least one of the variables can be expressed explicitly. However, many types of first integrals (e.g. conserved mechanical, thermodynamical or electrical energy) in physical system models belong to this class.

5.1.2 The underlying principle of the algorithm

Consider a set of n QP differential equations in the *homogeneous* form of (2.51), i.e.

$$\dot{x}_i = x_i \left(\sum_{j=1}^{m} A_{i,j} U_j \right) \ , \quad U_j = \prod_{k=1}^{n} x_k^{B_{j,k}}, \qquad i = 1, \ldots, n$$

Without the loss of generality we can assume that $m \geq n$, $\mathrm{Rank}(B) = n$ (i.e. the exponent matrix B is of full rank), moreover $\mathrm{Rank}(A) = n$ (the coefficient matrix A is of full rank. If these assumptions are not fulfilled, the QP-ODE has QM-type invariants, and an equivalent lower dimensional QP-ODE model that fulfill all these requirements can be determined by a three-step procedure based on QMTs, as described in [48], pp. 2418-2419.

Moreover, we can assume without restriction of generality that $i = n$ in (5.3) (because the QP form of the equations is preserved under permutation of the differential variables) i.e. the following algebraic dependence is present in (2.51)

$$x_n^{\frac{1}{\beta}} = c_0 + \sum_{\ell=1}^{L} c_\ell V_\ell \tag{5.4}$$

where $\beta, c_\ell \in \mathbb{R},\ \ell = 0, \ldots, L,\ \beta \neq 0$, and

$$V_\ell = \prod_{k=1}^{n-1} x_k^{\alpha_{\ell k}}, \quad \alpha_{\ell k} \in \mathbb{R},\ \ell = 1, \ldots, L,\ k = 1, \ldots, n-1 \tag{5.5}$$

It is clear that (5.4) is equivalent to the existence of a first integral of the form (5.1)–(5.2).

Taking the time derivative of (5.4) we obtain

$$\dot{x}_n = \beta \left(c_0 + \sum_{\ell=1}^{L} c_\ell V_\ell \right)^{\beta-1} \cdot \sum_{\ell=1}^{L} c_\ell \dot{V}_\ell \tag{5.6}$$

Using (5.4) and the fact that the monomials V_ℓ, $\ell = 1, \ldots, L$ do not depend on x_n we can further write

$$\dot{x}_n = \beta x_n^{\frac{\beta-1}{\beta}} \sum_{\ell=1}^{L} c_\ell \cdot \sum_{i=1}^{n-1} \frac{\partial V_\ell}{\partial x_i} \dot{x}_i \tag{5.7}$$

Finally, we can rewrite (5.7) to the standard QP form as

$$\dot{x}_n = x_n \left(\sum_{\ell=1}^{L} \sum_{i=1}^{n-1} \beta \cdot c_\ell \cdot \alpha_{\ell i} \cdot V_\ell \cdot x_n^{-\frac{1}{\beta}} \sum_{j=1}^{m} A_{i,j} U_j \right) \tag{5.8}$$

It is easy to see that the monomials in (5.8) (denoted by $R_{\ell j}$) are the following

$$R_{\ell j} = V_\ell \cdot U_j \cdot x_n^{-\frac{1}{\beta}} = x_1^{\alpha_{\ell 1}+B_{j1}} \cdot x_2^{\alpha_{\ell 2}+B_{j2}} \cdot \ldots \cdot x_{(n-1)}^{\alpha_{\ell,n-1}+B_{j,n-1}} \cdot x_n^{-\frac{1}{\beta}} \tag{5.9}$$

$$j = 1, \ldots, m, \quad \ell = 1, \ldots, L \tag{5.10}$$

while the coefficients of the monomials are

$$\gamma_{\ell j} = \sum_{i=1}^{n-1} \beta c_\ell \alpha_{\ell i} A_{i,j} \tag{5.11}$$

$$i = 1, \ldots, n-1, \quad j = 1, \ldots, m, \quad \ell = 1, \ldots, L \tag{5.12}$$

where the subscript i refers to that the partial differentiation in (5.7) has been performed by x_i.

Now, the aim of our algorithm is to determine β, the coefficients c_ℓ and the exponents $\alpha_{\ell i}$, $\ell = 1, \ldots, L$, $i = 1, \ldots n-1$ in (5.4)–(5.5) using the special form of (5.8) and that of the monomials in (5.9).

5.1.3 The basic algorithm for retrieving single invariants

The *input required by the algorithm* consists of the matrices A and B of the QP model in its *homogeneous form* defined in (2.51).

The *operational condition of the algorithm* is that, consistently to our preliminary assumptions, matrices A and B are of full rank.

Without the loss of generality we can assume that the explicit variable of the possible first integral is the last differential variable x_n. By a simple permutation of variables, each variable can be checked whether it is the explicit variable of a first integral.

In the following, let us denote the i-th row and j-th column of a matrix W by $W_{i,\cdot}$ and $W_{\cdot,j}$ respectively.

1. *Determination of the monomial candidates*
   To find a first integral in the form (5.4), one has to use the relationship (5.9) defined between the monomials $U_j, j = 1, \ldots, m$ of the original differential equations and the monomials $R_{\ell j}, \ell = 1, \ldots, L, j = 1, \ldots, m$ of the ODE for the algebraically dependent variable x_n.

   The first step is dedicated to collect these two groups of monomials, and then to determine the monomial candidates of the first integral using (5.9). Since the exponents of the j-th monomial of a QP-ODE is given as the j-th row vector of matrix B, the first thing to do is to gather the exponents of those monomials that occur in the first $(n-1)$ differential equations and construct the matrix $B^{(U)}$ from them. Let us denote the matrix created form A by deleting its n-th row by A^*. Now construct $B^{(U)}$ in the following way:

   Let $B^{(U)} = B$
   Mark those rows $B^{(U)}_{j,\cdot}$, $j = 1, \ldots, m$, *for which* $A^*_{\cdot,j} = 0$
   Delete the marked rows from $B^{(U)}$

   Similarly, collect the row vectors containing the exponents of monomials of the ODE for x_n to $B^{(R)}$:

   Let $B^{(R)} = B$
   Mark those rows $B^{(R)}_{j,\cdot}$, $j = 1, \ldots, m$, *for that* $A_{n+1,j} = 0$
   Delete the marked rows from $B^{(R)}$

   As a result, $B^{(U)} \in \mathbb{R}^{m_U \times n}$, $B^{(R)} \in \mathbb{R}^{m_R \times n}$, where m_U, m_R denote the number of monomials in the first $(n-1)$, and in the n-th differential equations, respectively. Note that there may be monomials (as row vectors) that appear in both $B^{(U)}$ and $B^{(R)}$.

   As (5.9) shows, the exponents of the monomials U_j, $j = 1, \ldots, m$ in the original differential equations and of the monomials $V_\ell, \ell = 1, \ldots, L$ in the algebraic equation are added up in the resulted monomials $R_{\ell j}$. This allows us to determine V_ℓ by simply *dividing* $R_{\ell j}$ by U_j for some j. This operation is equivalent to subtracting each exponent row vector corresponding to the monomials U_j (stored as row vectors of $B^{(U)}$) from the row vectors determining $R_{\ell j}$ (stored as row vectors of $B^{(R)}$).

   Therefore the next step is that the algorithm determines the exponent row vectors by subtracting each row of $B^{(U)}$ from each row of $B^{(R)}$, and construct the matrix $B^{(V)}$ made of the resulted row vectors:

$$B^{(V)} \in \mathbb{R}^{(m_U \cdot m_R) \times n} \;:\; B^{(V)}_{k,\cdot} = B^{(R)}_{j,\cdot} - B^{(U)}_{i,\cdot} \;, \quad k = (j-1) \times m_U + i$$

   Finally, make sure that each monomial candidate is coded only once in $B^{(V)}$:

   Delete repeated rows from $B^{(V)}$ *so that all rows are different*

   As a result, $B^{(V)} \in \mathbb{R}^{m_V \times n}$ contains all the monomial candidates of the first integral, where $m_V \leq m_U \cdot m_R$ denotes the number of these monomial candidates. However, taking into account all possible monomial candidates may

cause a huge redundancy but this guarantees that the exponent vectors of *all monomials of the first integral are contained* in $B^{(V)}$.

2. *Determination of β*
   To have a QP-type first integral from which x_n can be given explicitly, *the exponents of x_n in all of its monomials have to be identical.* This step classifies the exponent row vectors of the monomial candidates of the first integral by their last element.

   Compute how many different last elements of the row vectors of $B^{(V)}$ have and denote this number by S. Now make S different sets $\mathcal{B}_k,\ k = 1, \ldots, S$ and collect all the row vectors of $B^{(V)}$ having identical last elements into the same sets, while row vectors with different last elements into different sets. The result is a system of sets, where *the elements of each set are exponent row vectors belonging to the same β.*

   Now set the value of k to $k = 1$.

3. *Determination of the coefficients*
   The last step to be performed is to search for a first integral with monomial candidates belonging to the same $\mathcal{B}_k$. Since the exponents of the monomial candidates are already given, only their coefficients have to be determined. If these coefficients exist, the first integral exists for the current β, and it is completely determined by the algorithm.

   Denote the number of elements of $\mathcal{B}_k$ by L, the candidate for being the number of quasi-monomials in (5.4). Then the first integral candidate is given by the monomials described by the elements of $\mathcal{B}_k$ with unknown coefficients $c_1, \ldots, c_L$. Perform time-differentiation by simply applying (5.8) to it, with monomials and coefficients described in (5.9) and (5.11), respectively. Then match the monomials of this time-derivative and the monomials of the n-th differential equation, and determine the coefficients $\gamma_{\ell j}\ , \ell = 1, \ldots, L,\ j = 1 \ldots, m$ therefrom. Then try to solve the linear set of equations characterized in (5.11) for $c_1, \ldots, c_L$.

   Three cases are possible:

   (a) If (5.11) cannot be solved and $k < S$, increase k by one and jump to Step 3.

   (b) If (5.11) cannot be solved and $k = S$ the algorithm stops without finding a first integral.

   (c) If (5.11) can be solved, then the first integral is successfully determined and the algorithm stops.

This algorithm is capable of finding QP type first integrals which are explicit in (at least) one of their variables, moreover it operates without any heuristic steps. The properties of the algorithm are thoroughly discussed in Section 5.2.

5.1.4 Retrieval of multiple first integrals

The multiple retrieval algorithm comes from the basic algorithm by two simple modifications:

1. The first modification is on Step 3.(c), where the algorithm *does not stop* but stores the first integral found, increases s by one, and jumps back to Step 3.

2. Similarly to the basic algorithm, the multiple retrieval algorithm can find first integrals which are explicit in x_n. The second modification is the application of an outer loop performed n times applying a variable index swapping in each steps: in the k-th swapping

$$x_k^{new} = x_n, \quad x_n^{new} = x_k, \quad x_i^{new} = x_i, \quad i = 1, \ldots, n-1, \quad i \neq k$$

   where the superscript *'new'* refers to the variables with swapped indices. With this modification, the multiple retrieval algorithm can find first integrals which are explicit in at least one of their variables.

Care has to be taken to first integrals that are explicit in more than one of their variables, because the multiple retrieval algorithm finds these algebraic dependencies several times. Thus, after running this algorithm, the linear independence of invariants found has to be checked. By this multiple retrieval algorithm, a QP-DAE equivalent of the original QP-ODE can be determined. This QP-DAE is in the special form defined by the equations (2.72)-(2.75).

5.1.5 Implementation and computational properties of the algorithms

The basic and multiple retrieval algorithms are implemented in the MATLAB computation environment [1]. The functions

`[TEXTFORM,C,EXPONENTS,MESSAGE]=ALG_BASIC(A,B)`

`[TEXTFORM,WHICHONE,C,EXPONENTS,MESSAGE]=ALG_MULTIPLE(A,B)`

call the basic and multiple retrieval algorithms, respectively. The inputs A and B are the matrices of the QP-ODE model in its homogeneous form (2.51), while the output `TEXTFORM` gives the explicit form of invariant(s) found in a readable (text) format, while the outputs `WHICHONE`, `C` and `EXPONENTS` give the index of the explicit variable, the coefficients and the exponents of the first integral, respectively.

Both algorithms are of polynomial complexity. Recall that the number of the sets $\mathcal{B}_k$ was denoted by S in Step 3 of Section 5.1.3. Furthermore, let u denote the maximal element size of $\mathcal{B}_k$, $k = 1, \ldots, S$. The retrieval of a *single* invariant with the basic algorithm is of order

$$\Theta\Big(S \times u^3 \times m^3\Big)$$

where m denotes the number of monomials of the QP-ODE. The retrieval of *multiple* first integrals with the multiple retrieval algorithm is of order

$$\Theta\Big(S \times u^3 \times m^3 \times n\Big)$$

A rough upper estimation of these orders are $\Theta(m^{10})$ and $\Theta(m^{10} \times n)$, respectively, where m is the number of monomials, while n is the number of equations in (2.47).

Although MATLAB gives numerical solutions, there is a possibility to find first integrals even in parametric form (see Example 5.2 - Rikitake system) because for solving sets of linear equations, the basic and the multiple retrieval algorithms use the '`linsolve`' command of the built-in Maple kernel of MATLAB Symbolic Math Toolbox because of its efficiency [2].

5.2 Algebraic properties of the algorithm

Being truncated state-space models, QP-ODEs are inherently not unique in the sense that there are different QP-ODE models describing the same physical system: there are coordinates transformations of the variables that transform the model to an equivalent QP-ODE. Moreover there might be another transformations that give a different, but equivalent QP-ODE model.

This section concerns with the invariance of the retrieval process under two of these transformations performed on the QP-ODE model: under (nonlinear) quasi-monomial (QM) state transformations, and under algebraic equivalence transformations.

5.2.1 The effect of quasi-monomial transformations

Consider a QMT with an arbitrary invertible transformation matrix $C \in \mathbb{R}^{n\times n}$. The inverse of this transformation is also a QMT characterized by C^{-1} [22]. It is shown in [34] that a QP invariant of the form

$$I = \sum_{i=1}^{N} F_i \prod_{k=1}^{n} x_k^{E_{i,k}} \tag{5.13}$$

has the following form in the transformed coordinates:

$$I = \sum_{i=1}^{N} F_i \prod_{k=1}^{n} \hat{x}_k^{[EC^{-1}]_{i,k}} \tag{5.14}$$

which means that (5.14) has the same quasi-monomial terms as (5.13). It is easy to see from (5.1) and (5.14) that the explicitness of $x_i^{\frac{1}{\beta}}$ is generally not preserved under QMTs. Assuming that the index of the explicit variable is $i = n$ in (5.1), it is clear that a QMT with the following special C matrix preserves x_n as the explicit variable:

$$C = \left[\begin{array}{c|c} C_{11} & C_{12} \\ \hline 0 & C_{22} \end{array}\right]$$

where $C_{11} \in \mathbb{R}^{(n-1)\times(n-1)},\ C_{12} \in \mathbb{R}^{(n-1)\times 1},\ C_{22} \in \mathbb{R},\ 0 \in \mathbb{R}^{1\times(n-1)}$.

5.2.2 The effect of algebraic equivalence transformations

It is important to remark that there are cases when the sum of the coefficients in (5.8) is such that too much information is lost (i.e. too many terms are cancelled) to be able to obtain candidates for the parameters of (5.4)–(5.5) by the proposed algorithm. Of course, this problem can be caused by an unlucky equivalent algebraic transformation of the right hand side of the differential equations, as we will see at the end of Example 5.1 in Section 5.4. This problem is caused by the *ambiguity* of the algebraic form of non-minimal QP-ODE models caused by QP-type first integrals defining algebraic transformations that are performed on the RHS of these QP-ODE models. However (as it is also shown in Example 5.1), there is a good hope to finally find the first integral if it is explicit in at least one more variable.

5.3 Discussion

As it is discussed in Section 2.3.6, another method is applied to find invariants of QP systems in [33], moreover a software package called QPSI has also been implemented for this purpose [35]. The main differences between the approach of QPSI and our method are the following. Firstly, the classes of invariants that can be found by QPSI and our algorithm are different. Our method is able to retrieve only those invariants that can be written as an explicit function of a power of a differential variable. This is a narrower class of first integrals than QPSI can handle, but this restriction is not so severe in practice (see Section 5.1.1). Secondly, our method does not use the Lotka-Volterra representation of QP systems, but it searches for invariants directly in the original QP system model. Additionally, our algorithm is not based on finding semi-invariants. Thirdly, although our algorithm itself can handle the model parameters symbolically, the implementation has been done in a numerical computational environment (MATLAB) while QPSI works in a symbolic software environment (Maple). It follows from the above mentioned differences that our algorithm works effectively even in those cases when the number of monomials is high (see the examples in Section 5.4).

Though the method presented in this chapter can retrieve only those invariants that can be given as an explicit function of a power of a differential variable, it is possible to extend the retrieval process to a more general class of QP invariants. It can be seen from the description in Section 5.1, that the first step of the algorithm does not use the special explicit structure of the invariants but the second and third steps are completely dependent on this property. However, this dependence can be relaxed a bit on the price of more computation time in the following way. In Step 2, all the exponent row vectors $b^{(V)}$ of the monomial candidates are put to the *same set* $\mathcal{B}_1$ *irrespectively of their last elements.* Then the first integral candidate can be written *in its implicit form* with the monomials described by the exponent rows vectors of $\mathcal{B}_1$, and with unknown coefficients of the monomials. These coefficients can be determined by the time-differentiation of this first integral candidate.

It can be shown that the algorithm with the modifications above is capable to retrieve the invariant(s) that may not possess the explicit structure, only after an appropriate QM state transformation provided that there were no any algebraic

equivalence transformations applied to the model. This extension also generates a significant redundance of monomial candidates possibly causing numerical problems in Step 3 during the determination of the monoms' coefficients of the first integral.

5.4 Examples

In this section, the operation of the basic algorithm is illustrated on the model of a fed-batch fermentation process, and on the model of a simple electric circuit. The multiple retrieval algorithm is also demonstrated on the Rikitake system and on a computational example. The computations have been performed by the MATLAB procedures described in Section 5.1.5.

5.4.1 Example 5.1: A fed-batch fermentation process

In this example we use the model of an isotherm fed-batch fermentation process. Fermenters are used for producing different kinds of biomass (e.g. baker's yeast, antibiotics, etc.) in various branches of industry.

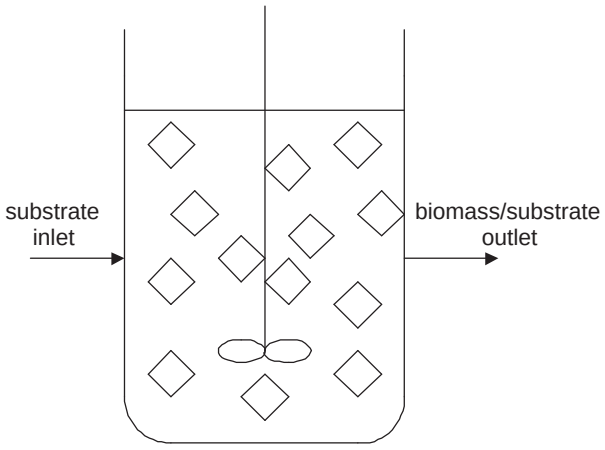

Figure 5.1: The fed-batch fermenter

A fed-batch fermentation process with a bi-linear reaction characteristics [O4] is used as a case study which is depicted in Fig. 5.1 and described by the following physical model:

$$\dot{x}_1 = K_r x_1 x_2 - \frac{x_1}{x_3} F \tag{5.15}$$

$$\dot{x}_2 = -\frac{1}{Y} K_r x_1 x_2 + \frac{S_F - x_2}{x_3} F \tag{5.16}$$

$$\dot{x}_3 = F \tag{5.17}$$

The variables of the model and their units in square brackets are

x_1	biomass concentration	[g/l]
x_2	substrate concentration	[g/l]
x_3	volume	[l]
F	feed flow rate	[l/h].

The constant parameters and their typical values are the following

$Y = 0.5$	yield coefficient	
$K_r = 1$	kinetic parameter	[g/l]
$S_F = 10$	influent substrate concentration	[g/l]

The feed flow-rate F is the physical control input variable and will be treated as a constant parameter during the retrieval process. The model (5.15)–(5.17) can be given as a QP-ODE in its homogeneous form (2.51):

$$\dot{x}_1 = x_1(K_r x_2 - F x_3^{-1}) \tag{5.18}$$

$$\dot{x}_2 = x_2(-\frac{1}{Y} K_r x_1 + S_F F x_2^{-1} x_3^{-1} - F x_3^{-1}) \tag{5.19}$$

$$\dot{x}_3 = x_3(F x_3^{-1}) \tag{5.20}$$

The A and B matrices of the QP model are:

$$A = \begin{bmatrix} K_r & -F & 0 & 0 \\ 0 & -F & -\frac{K_r}{Y} & S_F F \\ 0 & F & 0 & 0 \end{bmatrix}, \; B = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \\ 0 & -1 & -1 \end{bmatrix} \tag{5.21}$$

Apply the basic algorithm to search for a first integral. First set x_1 the variable for which the first integral is searched for. It needs to put x_1 to the 3-rd place, which can be easily done with the following change of coordinates: $x_1^{new} = x_3, \; x_2^{new} = x_2, \; x_3^{new} = x_1$.

Now apply the basic algorithm:

Step 1. The system matrices in the new coordinates are:

$$A^{new} = \begin{bmatrix} 0 & F & 0 & 0 \\ 0 & -F & -\frac{K_r}{Y} & S_F F \\ K_r & -F & 0 & 0 \end{bmatrix}, \; B^{new} = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \\ -1 & -1 & 0 \end{bmatrix} \tag{5.22}$$

The matrices $B^{(U)}$ and $B^{(R)}$ containing the exponent row vectors of the monomials respectively in the first two, and in the third differential equations are:

$$B^{(U)} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 0 & 1 \\ -1 & -1 & 0 \end{bmatrix} \quad , \quad B^{(R)} = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix}$$

Now subtract each row of $B^{(U)}$ from each row of $B^{(R)}$:

$$\begin{array}{lcccccccc} b_1^{(V)} & = & \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} & - & \begin{bmatrix} -1 & 0 & 0 \end{bmatrix} & = & \begin{bmatrix} 1 & 1 & 0 \end{bmatrix} \\ b_2^{(V)} & = & \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} & - & \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} & = & \begin{bmatrix} 0 & 1 & -1 \end{bmatrix} \\ b_3^{(V)} & = & \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} & - & \begin{bmatrix} -1 & -1 & 0 \end{bmatrix} & = & \begin{bmatrix} 1 & 2 & 0 \end{bmatrix} \\ b_4^{(V)} & = & \begin{bmatrix} -1 & 0 & 0 \end{bmatrix} & - & \begin{bmatrix} -1 & 0 & 0 \end{bmatrix} & = & \begin{bmatrix} 0 & 0 & 0 \end{bmatrix} \\ b_5^{(V)} & = & \begin{bmatrix} -1 & 0 & 0 \end{bmatrix} & - & \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} & = & \begin{bmatrix} -1 & 0 & -1 \end{bmatrix} \\ b_6^{(V)} & = & \begin{bmatrix} -1 & 0 & 0 \end{bmatrix} & - & \begin{bmatrix} -1 & -1 & 0 \end{bmatrix} & = & \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} \end{array}$$

Since the last elements of these row vectors should give $-\frac{1}{\beta}$, these elements must be non-zero, meaning that there are only two feasible exponent vectors: $b_2^{(V)}$ and $b_5^{(V)}$.
Step 2. Since the last components of both exponent vectors are identical, (both vectors belong to the same $\beta = 1$) there is only one set $\mathcal{B}_1$ containing these two exponent vectors: $\mathcal{B}_1 = \{b_2, b_5\}$.
Step 3. The parametric first integral is

$$x_3^{new} = c_1(x_1^{new})^{-1} + c_2 x_2^{new} + c_0 \tag{5.23}$$

Its time-derivative does not provide solution for $c_1,\ c_2$ because of the phenomena described in Section 5.2.2. However, this can be written in the original coordinates as

$$x_1 = c_1 x_3^{-1} + c_2 x_2 + c_0 \tag{5.24}$$

By arranging it to an implicit form:

$$x_3(x_1 - c_2 x_2 - c_0) = c_1 \tag{5.25}$$

and taking its time-derivative gives an equation that from the coefficients c_1 and c_3 can be uniquely determined, and therefore the implicit form of the first integral is retrieved:

$$x_3(\frac{1}{Y}x_1 - x_2 + S_F) = \frac{S_F}{c_0} \tag{5.26}$$

where c_0 comes from the initial conditions of the model.

Note that another algorithm based on the construction of Lie-algebras has been applied to a similar fermenter model with the same structure but slightly different reaction kinetics in [94], giving the same final result. As a comparison, this new method provides a computationally more advantageous way of retrieval mainly because it does not require the analytic solution of partial differential equations which was the case in [94].

Now take a look at the four monomials $U_1, \ldots, U_4$ of the model:

$$U_1 = x_2, \quad U_2 = x_3^{-1}, \quad U_3 = x_1, \quad U_4 = x_2^{-1}x_3^{-1} \tag{5.27}$$

To see the connection with [33], the monomials of the first integral (5.26) can be written as a product of a term which is a quasi-monomial function of the monomials, and another term which is a polynomial function of the monomials:

$$U_2^{-2}\Big(\frac{1}{Y}U_2U_3 - U_1U_2 + S_F U_2\Big) = \frac{S_F}{c_0} \tag{5.28}$$

Observe that the polynomial term is of second order.

This example shows the effect of an algebraic equivalence transformation described in Section 5.2.2. If one chooses x_3 as the explicit variable of the first integral, then the retrieval process is not successful. To understand the exact problem, let us express x_3 form the first integral (5.26):

$$x_3 = \Big(\frac{c_0}{S_F}(\frac{1}{Y}x_1 - x_2 + S_F)\Big)^{-1} \tag{5.29}$$

then take its time-derivative:

$$\dot{x}_3 = -x_3^2(-\frac{1}{Y}\dot{x}_1 - \dot{x}_2) = x_3(\frac{c_0F}{S_FY}x_1 + c_0F - \frac{c_0F}{S_F}x_2) \tag{5.30}$$

and then compare this to (5.20). These two differential equations are *equivalent*, since

$$\frac{c_0F}{S_FY}x_1 + c_0F - \frac{c_0F}{S_F}x_2 = Fx_3^{-1} \tag{5.31}$$

according to (5.26). This is the case when an algebraic equivalence transformation defined by (5.31) is applied to the third model equation (5.30) resulting (5.20).

5.4.2 Example 5.2: A Rikitake system

Rikitake modelled the generation of the earth's magnetic field as a simple coupled disk dynamo system [85] as depicted in Figure 5.2. According to [33], the dynamic

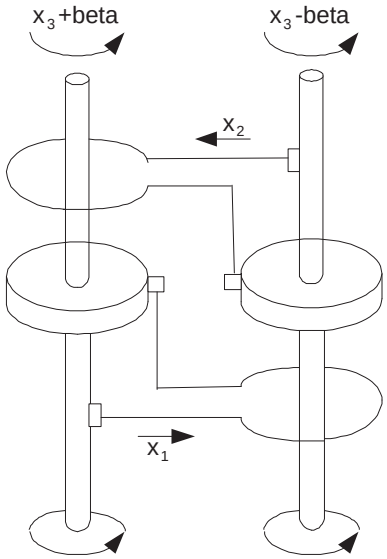

Figure 5.2: The two-disk dynamo system

model equations of the Rikitake model are:

$$\dot{x}_1 = -\mu x_1 + x_2(x_3 + \beta) \tag{5.32}$$
$$\dot{x}_2 = -\mu x_2 + x_1(x_3 - \beta) \tag{5.33}$$
$$\dot{x}_3 = \alpha - x_1x_2 \tag{5.34}$$

where x_1 and x_2 are the dimensionless currents of the dynamo coils, x_3 is the dimensionless average of the angular velocities of the two dynamo disks (which are $x_3 + \beta$ and $x_3 - \beta$, respectively), and α, β, μ are real parameters.

Consider the case when $\alpha = \mu = 0$ and write it into QP form:

$$\dot{x}_1 = x_1(x_1^{-1}x_2x_3 + \beta x_1^{-1}x_2) \tag{5.35}$$
$$\dot{x}_2 = x_2(x_1x_2^{-1}x_3 - \beta x_1x_2^{-1}) \tag{5.36}$$
$$\dot{x}_3 = x_3(x_1x_2x_3^{-1}) \tag{5.37}$$

The application of the basic algorithm gives the following first integral:

$$x_2^2 = c_1 x_1^2 - 2\beta(1+c_1)x_3 + (1-c_1)x_3^2 + c_0 \ , \quad c_1 \in \mathbb{R} \tag{5.38}$$

where the constant c_0 comes from the initial conditions of the model while c_1 is an arbitrary real parameter. This first integral is equivalent with two non-parametric first integrals:

$$\begin{aligned} I_1 &= x_3^2 + 2\beta x_3 - x_1^2 \\ I_2 &= x_3^2 - 2\beta x_3 - x_2^2 \end{aligned}$$

We note that the first integral found in [33] is the linear combination $I_1 - I_2$.

Consider now the case when the angular velocities of the two dynamo disks are equal (i.e. $\beta = 0$). In this case (5.38) can be written as

$$-c_1 x_1^2 + x_2^2 + (c_1 - 1)x_3^2 = c_0 \ , \quad c_1 \in \mathbb{R} \tag{5.39}$$

describing the conserved energy of the system. Moreover, from (5.39)-(5.39) it comes that the *absolute value* of the two dimensionless currents are equal ($x_1^2 = x_2^2$), and the dimensionless angular velocity is equal to the absolute value of the currents ($x_3^2 = x_1^2 (= x_2^2)$).

The MATLAB implementation of this example uses a constant β value and applies the multiple retrieval algorithm to show that the first integral comes back with all variables chosen as the explicit one.

5.4.3 Example 5.3: An electric circuit

Consider the simple electric circuit in Figure 5.3 consisting of two inductors and a capacitor. Assume that the inductors have linear characteristics with inductances

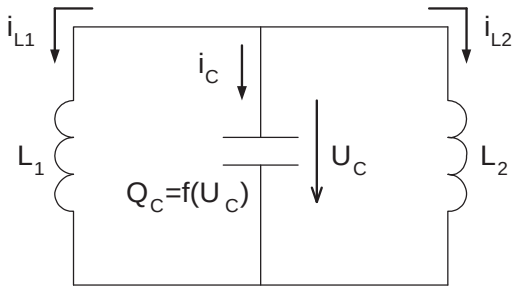

Figure 5.3: The LC circuit

L_1 and L_2. Denote the currents of the inductors by i_{L1} and i_{L2} respectively. Let the capacitor have a nonlinear QM-type characteristic:

$$Q_C = f(U_C)$$

where Q_C is the charge and U_C is the voltage of the capacitor and f is a QM-type function of U_C. The partial derivative of f is also a QM-type function of U_C:

$$\frac{\partial f}{\partial U_C} = k U_C^{\alpha} \ , \quad \alpha \in \mathbb{N}, k \in \mathbb{R}$$

Denote the current of the capacitor by i_C. The dependence

$$i_C = \frac{dQ_C}{dt} = \frac{\partial Q_C}{\partial U_C} \cdot \frac{dU_C}{dt}$$

gives

$$\frac{dU_C}{dt} = \frac{1}{\frac{\partial f}{\partial U_C}} \cdot i_C$$

Using Kirchhoff's laws and choosing the variables $x_1 = i_{L1}$, $x_2 = i_{L2}$ and $x_3 = U_C$ gives a three dimensional QP-ODE model:

$$\dot{x}_1 = \frac{1}{L_1} x_3 = x_1 \left(\frac{1}{L_1} x_1^{-1} x_3\right) \tag{5.40}$$

$$\dot{x}_2 = \frac{1}{L_2} x_3 = x_2 \left(\frac{1}{L_2} x_2^{-1} x_3\right) \tag{5.41}$$

$$\dot{x}_3 = \frac{1}{k} x_3^{-\alpha}(-x_1 - x_2) = x_3\left(-\frac{1}{k} x_1 x_3^{-\alpha-1} - \frac{1}{k} x_2 x_3^{-\alpha-1}\right) \tag{5.42}$$

The basic algorithm gives the following algebraic dependence:

$$x_3^{\alpha+2} = \Big(\frac{(\alpha+2)(L_1^2 - L_1 L_2)}{2kL_2} + \frac{L_1^2}{L_2^2} P\Big) x_1^2 - \Big(\frac{2L_1}{L_2} P + \frac{(\alpha+2)L_1}{k}\Big) x_1 x_2 + P x_2^2$$

where $P \in \mathbb{R}$ is an arbitrary parameter. This first integral is equivalent to two non-parametric ones:

$$x_3^{\alpha+2} = \frac{(\alpha+2)(L_1^2 - L_1 L_2)}{2kL_2} x_1^2 - \frac{(\alpha+2)L_1}{k} x_1 x_2$$

$$x_3^{\alpha+2} = -\frac{\alpha+2}{k}\Big(\frac{L_1}{2} x_1^2 + \frac{L_2}{2} x_2^2\Big)$$

If the capacitor has a linear characteristics (i.e. $\alpha = 0$), the latter first integral describes the conservation of the electrical energy since it can be re-written in the form

$$\frac{1}{2} L_1 x_1^2 + \frac{1}{2} L_2 x_2^2 + \frac{1}{2} k x_3^2 = const.$$

with k being the capacitance.

The MATLAB implementation of this example uses constant values of L_1, L_2, k and α, and gives the invariant in its parametric form.

5.4.4 Example 5.4: A computational example with multiple first integrals

Consider the following QP-ODE system:

$$\dot{x}_1 = x_1(x_2) \tag{5.43}$$

$$\dot{x}_2 = x_2(x_1^2) \tag{5.44}$$

$$\dot{x}_3 = x_3(-21 x_1^{10} x_2^{-7} x_3^{-1} + 24 x_1^6 x_2^{-5} x_3^{-1}) \tag{5.45}$$

$$\dot{x}_4 = x_4(10 x_1^{10} x_2^{21} x_3^2 x_4^{-1} + 20 x_1^{12} x_2^{20} x_3^2 x_4^{-1} - 42 x_1^{20} x_2^{13} x_3 x_4^{-1} + \\ + 48 x_1^{16} x_2^{15} x_3 x_4^{-1}) \tag{5.46}$$

The application of the multiple retrieval algorithm gives back three first integrals:

$$x_4 = x_1^{10} x_2^{20} x_3^2 \tag{5.47}$$

$$x_3 = 3x_1^8 x_2^{-7} + 4x_1^6 x_2^{-6} \tag{5.48}$$

$$x_1^2 = 2x_2 \tag{5.49}$$

The MATLAB implementation of this example uses the multiple retrieval algorithm successfully. The efficiency of the algorithm in the case of multiple first integrals is clearly visible in this example.

5.5 Summary

In this chapter, a numerically effective algorithm based on simple matrix operations is proposed for the determination of a class of first integrals in quasi-polynomial systems.

Two versions of the algorithm are presented: the first one can retrieve a single invariant of a QP-ODE, while the other is designed to determine multiple (arbitrary number of) first integrals. Both versions are polynomial-time, and can handle quasi-polynomial systems of arbitrary order with arbitrarily high number of quasi-monomials. The algorithm has been successfully implemented and tested in the MATLAB numeric computational software environment.

The invariance properties of the algorithm have been discussed under quasi-monomial transformations and also under algebraic equivalence transformations. The operation and the effectiveness of the algorithm has been illustrated on several examples taken from the fields of physics and biochemistry, and also on a purely numerical example.

According to its applicability, simplicity and effectiveness even in cases of high-dimensional models, this algorithm is proved a feasible alternative of the QPSI approach.

Chapter 6

Conclusions

6.1 Main contributions

The new scientific results presented are summarized in the following Theses.

Thesis 1 *Stability analysis of the zero dynamics of a low power gas turbine model in QP form* (Chapter 3)
(**[P1],[P2],[P3],[P4]**)
The local stability of two different zero dynamics of a third order nonlinear low power gas turbine model taken from literature [7] has been investigated.

For the *zero dynamics for the turbine inlet total pressure* as output, a quadratic stability analysis method for LV systems known from literature [68] has been applied to a typical operating point, with stability neighborhood estimation. The estimated stability neighborhood covers approximately 56 % of the operating domain. Simulations confirmed theoretical results, moreover the trajectories started out of the quadratic stability neighborhood showed the conservativeness of this estimation since the range of stability is proved to be wider than the estimated one. By the application of this stability analysis method to different zero dynamics belonging to different steady states of the turbine inlet total pressure, the results have been generalized and the possible causes of the conservativeness of the estimation method have been discussed.

The stability of the one dimensional *zero dynamics for the rotational speed* has been investigated with phase diagrams. The equilibrium of this zero dynamics found to be unique and stable in the whole operating domain, independently of the arbitrary constant values of the rotational speed and of the load torque, which is the most versatile environmental disturbance.

The results of this Thesis give a basis for control structure selection for the low power gas turbine.

Thesis 2 *Controller design for a low-power gas turbine* (Chapter 4)
(**[P2],[P3],[P4],[P5],[P6]**)
The rotational speed of the low power gas turbine has been chosen to be controlled via three different input-output linearization based linear controllers:

(a) An LQ servo controller that tracks a piecewise constant reference signal for the rotational speed, with known time-function of the load torque;

(b) an LQ+MPT controller that keeps the rotational speed at a constant value and the states between bounded values, with measurable load torque;

(c) a novel adaptive LQ servo controller that is an extension of (a): the time evolution of the *load torque is unknown, and is estimated*.

Simulation experiments showed that all controllers guarantee robustness against model parameter uncertainties and environmental disturbances.

In cases (a) and (b) the time function of the load torque has been assumed to be known, although in most cases it is an unmeasurable disturbance of the environment. The advanced controller (c) handles this more realistic case with a novel approach: the load torque is estimated by a dynamic feedback supplied by a state space based adaptive controller and used in the input-output linearizing feedback.

Worst case simulations in MATLAB/SIMULINK have showed that the controller properly estimates the time function of the load torque, moreover the reference tracking for the rotational speed is robust against all environmental disturbances and model parameter uncertainties. In spite that the load torque is only estimated, the controlled plant shows the same robustness as (a) with known load torque.

The importance of this result is well characterized by the fact that this approach has not been applied to gas turbines yet, however the load torque is the most important environmental disturbance because of its non-measurability, versatility and impact on the time behavior of the gas turbine.

Thesis 3 *Determining invariants (first integrals) of QP-ODEs* (Chapter 5) (**[P7],[P8],[P9],[P10],[P11]**)

A new algorithm has been developed that is able to determine a wide class of QP-type invariants of non-minimal QP-ODE systems. The operation of this algorithm is based on simple matrix operations, and does not contain any heuristic steps.

Two versions of this algorithm has been presented: the former is designed to determine single, the latter is to determine multiple first integrals. Both versions of the algorithm are of polynomial-time, and can retrieve invariants from arbitrarily high dimensional QP-ODEs with arbitrarily high number of quasi-monomials.

Both versions of the algorithm has been implemented in the MATLAB numeric computational environment, and their operation has been tested successfully on the mathematical model of several physical systems.

The invariance properties of the algorithm have been investigated under quasi-monomial and algebraic equivalence transformations. The capabilities and limitations of the algorithm have also been discussed, and compared with another

known method, the QPSI method [35]. According to its applicability, simplicity and effectiveness even in cases of high-dimensional models with arbitrary high number of quasi-monomials, the proposed algorithm is proved to be a feasible alternative of the QPSI approach.

6.2 Application areas, directions of future research

Since a detailed nonlinear model of a real low-power gas turbine has been used for the stability analysis and controller design thereon, the implementation of the controllers designed, especially the one that estimates the load torque on an experimental testbed (on a real turbine) is the first step towards the application of theoretical results. Then, this novel controller should be implemented to other gas turbines.

There are a lot of possible directions of future research connected to the analysis and control of the low-power gas turbine:

- For the stability analysis of the zero dynamics for the turbine inlet total pressure, another method could also be applied, e.g. *bilinear* matrix inequalities, being hopefully less conservative than the method based on LMIs.

- Although the one dimensional zero dynamics of the gas turbine for the rotational speed has a very complicated algebraic structure, its phase diagram is a relatively simple curve. The identification of the phase curve as a polynomial depending on two constant parameters (rotational speed and load torque) might give the possibility of the analytical treatment of stability investigation.

- The robustness of the controllers could only be checked via simulations, however an analytical treatment of uncertainties would be desirable. Further work should be focused on obtaining a structured description of the key parametric and dynamic uncertainties of the gas turbine model that would guarantee better circumstances for robust controller design.

Since arbitrary classical nonlinear input-affine models can be transformed to a QP-ODE, the algorithm for retrieving invariants could be an effective and easily applicable tool to determine QP-type first integrals.

The possible directions of future research related to this topic are:

- The algorithm should be re-implemented in a symbolic computer-algebra system (e.g. MAPLE) in order to handle parametric models.

- Moreover, this algorithm should be applied constructively in control oriented nonlinear system analysis and feedback design - for instance by combining it with control Lyapunov function based techniques. It is known that invariants with appropriate properties can be utilized in solving stabilization problems for dynamical systems. By prescribing an appropriate QP-type invariant for the closed loop system, the necessary control input function can be computed in the form of a nonlinear state feedback.

6.3 Publications

The results of this thesis were published in journals, presented in conferences, or published in the form of research reports as enlisted below. The relevant **Thesis** is indicated in parentheses.

[P1] B. Pongrácz, G. Szederkényi, P. Ailer, and K. M. Hangos. Stability of zero dynamics of a low-power gas turbine. In *Proceedings of the 12th Mediterranean Control Conference - MED'04,* Turkey, 2004. *On CD.* (**Thesis 1**)

[P2] B. Pongrácz. Nonlinear stability analysis and control of a low power gas turbine. In *Proceedings of the 2nd PhD Mini-Symposium,* pages 60-62, Veszprém, Hungary, 2004. (**Theses 1,2**)

[P3] B. Pongrácz, P. Ailer, G. Szederkényi, and K. M. Hangos. Nonlinear zero dynamics analysis and control of a low power gas turbine. In *Proceedings of the 5th PhD Workshop on Systems and Control - a Young Generation Viewpoint,* pages 112-116, Balatonfüred, Hungary, 2004. (**Theses 1,2**)

[P4] B. Pongrácz, P. Ailer, G. Szederkényi, and K. M. Hangos. Nonlinear zero dynamics analysis and control of a low power gas turbine. *Technical report SCL-003/2005, Computer and Automation Research Institute, HAS*, 2005.
`http://daedalus.scl.sztaki.hu/pdf/research reports/SCL-003-2005.pdf`
(**Theses 1,2**)

[P5] P. Ailer, B. Pongrácz, and G. Szederkényi. Constrained control of a low power industrial gas turbine based on input-output linearization. In *Proceedings of the International Conference on Control and Automation - ICCA'05,* 2005. *On CD.* (**Thesis 2**)

[P6] B. Pongrácz, P. Ailer, G. Szederkényi, and K. M. Hangos. Nonlinear reference tracking control of a gas turbine with load torque estimation. *International Journal of Adaptive Control and Signal Processing,* in print.
DOI: 10.1002/acs.1020 (**Thesis 2**)
Impact factor: 0.580

[P7] A. Magyar, G. Ingram, B. Pongrácz, G. Szederkényi, and K. M. Hangos. On some properties of quasi-polynomial differential equations and differential-algebraic equations. *Technical report SCL-010/2003, Computer and Automation Research Institute, HAS*, 2003.
`http://daedalus.scl.sztaki.hu/PCRG/works/SCL-010-2003.pdf`
(**Thesis 3**)

[P8] B. Pongrácz, G. Ingram, and K. M. Hangos. The structure and analysis of QP-DAE system models. *Technical report SCL-004/2004, Computer and Automation Research Institute, HAS*, 2004.
`http://daedalus.scl.sztaki.hu/pdf/research reports/SCL-004-2004.pdf`
(**Thesis 3**)

[P9] B. Pongrácz, G. Szederkényi, and K. M. Hangos. An algorithm for determining invariants in quasi-polynomial systems. In *Proceedings of the 6th PhD Workshop on Systems and Control - a Young Generation Viewpoint,* Izola, Slovenia, 2005. *On CD.* (**Thesis 3**)

[P10] B. Pongrácz, G. Szederkényi, and K. M. Hangos. An algorithm for determining a class of invariants in quasi-polynomial systems. *Technical report SCL-002/2005, Computer and Automation Research Institute, HAS*, 2005. `http://daedalus.scl.sztaki.hu/pdf/research reports/SCL-002-2005.pdf` (**Thesis 3**)

[P11] B. Pongrácz, G. Szederkényi, and K. M. Hangos. An algorithm for determining a class of invariants in quasi-polynomial systems. *Computer Physics Communications,* **175:**204-211, 2006. DOI: 10.1016/j.cpc.2006.03.003 (**Thesis 3**)
Impact factor: 1.595

Another group of publications is not directly connected to this thesis:

[O1] B. Pongrácz. An algorithm for transforming a class of DAE models into a purely differential form. In *Proceedings of the 2nd PhD Workshop on Systems and Control - a Young Generation Viewpoint,* pages 31-40, Balatonfüred, Hungary, 2001.

[O2] B. Pongrácz, G. Szederkényi, and K. M. Hangos. The effect of algebraic equations on the stability of process systems. In *Proceedings of the 3rd International PhD Workshop on Systems and Control*, Strunjan, Slovenia, 2002. *On CD.*

[O3] B. Pongrácz, G. Szederkényi, and K. M. Hangos. The effect of algebraic equations on the stability of process systems. *Technical Report, SCL-003/2002 Computer and Automation Research Institute, HAS*, 2002. `http://daedalus.scl.sztaki.hu/PCRG/works/SCL-003-2002.pdf`

[O4] B. Pongrácz. Stability analysis techniques for process models in DAE form - The effect of algebraic equations. In *Proceedings of the 1st PhD Mini-Symposium*, pages 55-57, Veszprém, Hungary, 2003.

[O5] B. Pongrácz, G. Szederkényi, and K. M. Hangos. The effect of algebraic equations on the stability of process systems modelled by differential algebraic equations. In *Proceedings of the 13th European Symposium on Computer Aided Process Engineering - ESCAPE-13,* pages 857-862, Finland, 2003.

Appendix A

Nomenclature, constants and coefficients of the gas turbine model

Nomenclature of the gas turbine model

Variables and Constants

A	area $[m^2]$
E	mechanical energy $[J]$
M	torque $[Nm]$
P	power $[W]$
Q	heat flow $[J/s] = [W]$
Q_f	lower thermal value of fuel $[J/kg]$
R	specific gas constant $[J/(kg\ K)]$
T	temperature $[K]$
U	internal energy $[J]$
V	volume $[m^3]$
c	specific heat $[J/(kg\ K)]$
i	specific enthalpy $[J/kg]$
m	mass $[kg]$
n	rotational speed $[1/s]$
p	pressure $[Pa]$
q	dimensionless mass flow rate $[-]$
t	time $[s]$
α_1, α_2	coefficients of $q(\lambda_1)$ $[s]$
α_3, α_4	coefficients of $q(\lambda_1)$ $[-]$
β	specific parameter of air and gas $[\sqrt{K}s/m]$
$\gamma_1, \gamma_2, \gamma_3, \gamma_4$	coefficients of $q(\lambda_3)$ $[-]$
η	efficiency $[-]$
Θ	inertial moment $[kg\ m^2]$
κ	adiabatic exponent $[-]$
λ	dimensionless speed $[-]$
ν	mass flow rate $[kg/s]$
σ	pressure loss coefficient $[-]$
τ	turbine velocity coefficient $[\sqrt{K}s]$

Nomenclature of the gas turbine model

Subscripts

$_0$	inlet duct inlet
$_1$	compressor inlet
$_2$	compressor outlet
$_3$	turbine inlet
$_4$	turbine outlet
$_C$	refers to compressor
$_{Comb}$	refers to combustion chamber
$_{comb}$	refers to combustion
$_{fuel}$	refers to fuel
$_I$	refers to inlet duct
$_{in}$	refers to inlet
$_{load}$	load
$_{mech}$	mechanical
$_N$	refers to gas deflector
$_{out}$	refers to outlet
$_p$	refers to constant pressure
$_{schaft}$	refers to schaft
$_T$	refers to turbine
$_v$	refers to constant volume

Table A.1: Constants of the model of the DEUTZ T216 type gas turbine

Not.	**Value**	**Not.**	**Value**
c_p	1004.5 $J/(kg\ K)$	c_v	717.5 $J/(kg\ K)$
A_1	0.0058687 m^2	A_3	0.0117056 m^2
Q_f	42.8 MJ/kg	R	287 $J/(kg\ K)$
T_0	288.15 K	V_{Comb}	0.005675 m^3
α_1	0.00035319 s	α_2	0.0011097 s
α_3	-0.4611	α_4	0.16635
β	0.0404184 $\sqrt{K}s/m$	γ_1	-0.033728
γ_2	0.004458	γ_3	0.048847
γ_4	0.15542	η_C	0.67585
η_{comb}	0.79161	η_{mech}	0.9801
η_T	0.85677	Θ	0.0004 $kg\ m^2$
κ	1.4	σ_{Comb}	0.93739
σ_I	0.98879	σ_N	0.96687
τ	0.028071 $\sqrt{K}s$		

Table A.2: Coefficients of the model of the DEUTZ T216 type gas turbine

$$
\begin{aligned}
b_1 &= 1 \\
b_2 &= RQ_f\eta_{comb}/(V_{Comb}c_v) \\
c_{1,1} &= \beta A_1\alpha_1\sqrt{T_0}/(\sigma_{Comb}) \\
c_{1,2} &= \beta A_1\alpha_2\sqrt{T_0} \\
c_{1,3} &= \beta A_1\alpha_3/\sigma_{Comb} \\
c_{1,4} &= \beta A_1\alpha_4 \\
c_{1,5} &= -\beta A_3 R\gamma_1\tau\sigma_I\sigma_N/V_{Comb} \\
c_{1,6} &= -\beta A_3 R\gamma_2\tau/V_{Comb} \\
c_{1,7} &= -\beta A_3\gamma_3\sigma_I\sigma_N\sqrt{R}/\sqrt{V_{Comb}} \\
c_{1,8} &= -\beta A_3\gamma_4\sqrt{R}/\sqrt{V_{Comb}} \\
c_2 &= R\beta c_p A_1\alpha_3(1-\eta_C^{-1})/(V_{Comb}c_v\sigma_{Comb}) \\
c_{2,1} &= R\beta c_p A_1\alpha_4(1-\eta_C^{-1})/(V_{Comb}c_v) \\
c_{2,2} &= R\beta c_p A_1\alpha_3/(V_{Comb}c_v\eta_C\sigma_{Comb}^{1+(\kappa-1)/\kappa}) \\
c_{2,3} &= -R\beta c_p A_3\gamma_4/(\sqrt{RV_{Comb}}c_v) \\
c_{2,4} &= R\beta c_p A_1\alpha_2\sqrt{T_0}/(V_{Comb}c_v\eta_C\sigma_{Comb}^{(\kappa-1)/\kappa}) \\
c_{2,5} &= R\beta c_p A_1\alpha_1\sqrt{T_0}/(V_{Comb}c_v\sigma_{Comb}^{1+(\kappa-1)/\kappa}\eta_C) \\
c_{2,6} &= -R\beta c_p A_3\gamma_1\tau\sigma_I\sigma_N/(V_{Comb}c_v) \\
c_{2,7} &= R\beta c_p(A_1\alpha_1\sqrt{T_0}(1-\eta_C^{-1}) - A_3\gamma_2\tau\sigma_{Comb})/(V_{Comb}c_v\sigma_{Comb}) \\
c_{2,8} &= R\beta c_p A_1\alpha_4/(V_{Comb}c_v\eta_C\sigma_{Comb}^{(\kappa-1)/\kappa}) \\
c_{2,9} &= -R\beta c_p A_3\gamma_3\sigma_I\sigma_N/(\sqrt{RV_{Comb}}c_v) \\
c_{2,10} &= R\beta c_p A_1\alpha_2\sqrt{T_0}(1-\eta_C^{-1})/(V_{Comb}c_v) \\
c_{3,1} &= \beta A_3 c_p\eta_T\eta_{mech}\ \gamma_1\tau\sigma_I\sigma_N/(4\pi^2\Theta) \\
c_{3,2} &= -\beta A_3 c_p\eta_T\eta_{mech}\gamma_1\tau\sigma_I^{-1/\kappa}\sigma_N^{-1/\kappa}/(4\pi^2\Theta) \\
c_{3,3} &= -\beta A_3 c_p\eta_T\eta_{mech}\gamma_2\tau/(4\pi^2\Theta\sigma_I^{(\kappa-1)/\kappa}\sigma_N^{(\kappa-1)/\kappa}) \\
c_{3,4} &= \beta A_1 c_p\alpha_3/(4\pi^2\Theta\sigma_{Comb}\eta_C) \\
c_{3,5} &= \beta A_3 c_p\eta_T\eta_{mech}\gamma_3\sigma_I\sigma_N\sqrt{V_{Comb}}/(4\pi^2\Theta\sqrt{R}) \\
c_{3,6} &= -\beta A_3 c_p\eta_T\eta_{mech}\gamma_3\sigma_I^{-1/\kappa}\sigma_N^{-1/\kappa}\sqrt{V_{Comb}}/(4\pi^2\Theta\sqrt{R}) \\
c_{3,7} &= \beta A_3 c_p\eta_T\eta_{mech}\gamma_4\sqrt{V_{Comb}}/(4\pi^2\Theta\sqrt{R}) \\
c_{3,8} &= -\beta A_3 c_p\eta_T\eta_{mech}\gamma_4\sqrt{V_{Comb}}/(4\pi^2\Theta\sigma_I^{(\kappa-1)/\kappa}\sigma_N^{(\kappa-1)/\kappa}\sqrt{R}) \\
c_{3,9} &= -\beta A_1 c_p\alpha_1\sqrt{T_0}/(4\pi^2\Theta\sigma_{Comb}^{1+(\kappa-1)/\kappa}\eta_C) \\
c_{3,10} &= \beta A_1 c_p\alpha_4/(4\pi^2\Theta\eta_C) \\
c_{3,11} &= -\beta A_1 c_p\alpha_4/(4\pi^2\Theta\sigma_{Comb}^{(\kappa-1)/\kappa}\eta_C) \\
c_{3,12} &= -\beta A_1 c_p\alpha_2\sqrt{T_0}/(4\pi^2\Theta\eta_C\sigma_{Comb}^{(\kappa-1)/\kappa}) \\
c_{3,13} &= \beta c_p(A_3\eta_T\eta_C\sigma_{Comb}\eta_{mech}\gamma_2\tau + A_1\alpha_1\sqrt{T_0})/(4\pi^2\Theta\sigma_{Comb}\eta_C) \\
c_{3,14} &= -\beta A_1 c_p\alpha_3/(4\pi^2\Theta\sigma_{Comb}^{1+(\kappa-1)/\kappa}\eta_C) \\
c_{3,15} &= \beta\sqrt{T_0}A_1 c_p\alpha_2/(4\pi^2\Theta\eta_C) \\
c_{3,16} &= -3/(100\pi\Theta) \\
o_1 &= V_{Comb}(1-\eta_T)/R \\
o_2 &= V_{Comb}\eta_T\sigma_I^{(1-\kappa)/\kappa}\sigma_N^{(1-\kappa)/\kappa}/R
\end{aligned}
$$

Appendix B

Zero dynamics of the gas turbine model for the rotational speed

Recall the zero dynamics of the gas turbine model for the rotational speed in (3.44), which is a scalar differential equation of the turbine inlet total pressure (x_2):

$$\dot{x}_2 = \phi(x_2)$$

This zero dynamics will be detailed here in terms of x_{2d} which is the *dimensionless* version of the turbine inlet total pressure. Recall that the dimensionless versions of the state variables can be computed as

$$x_{id} = \frac{x_i}{x_{imax} - x_{imin}} \ , \quad i = 1, 2, 3 \ .$$

Near its scalar variable x_{2d}, the zero dynamics contains two additional parameters, namely the load torque M_{load} and the steady state value of the rotational speed in its dimensionless form, denoted by x^*_{3d}:

$$\begin{aligned}
\mathbf{\frac{dx_{2d}}{dt}} \ = \ & -10.212x^*_{3d}x_{2d} + 34.465x^*_{3d}x_{2d}^{1.2857} - 16.479x^*_{3d} + 60.138x^*_{3d}x_{2d}^{0.28571} - 131.43x_{2d}^{1.5}x_{1d}^{-0.5} + \\
& +67.255x_{2d} - 13.474 + 49.175x_{2d}^{0.28571} + 9.3824x^*_{3d}x_{2d}^2 - 66.659x_{2d}^{2.5}x_{1d}^{-0.5} - 245.45x_{2d}^{1.2857} - \\
& -\Big\{\Big[\big(-17.483x^*_{3d}x_{2d} + 1.4321x^*_{3d}\big)e_5 + 1575.8x_{1d}^{-0.5}x_{2d}^{0.5}e_4e_5 + 2008300 \cdot x_{2d}^{1.5}x_{1d}^{-1.5}(e_3 - 1)e_4\Big] \cdot \\
& \cdot\Big[148.4x^*_{3d}x_{2d} + 258.95x^*_{3d} - 1056.9x_{2d} + 221.74 - 2881.6x_{2d}^{0.5}x_{1d}^{0.5}e_4\Big] + \\
& +\Big[\Big(-17.483x^*_{3d}x_{1d} - 1.4321x^*_{3d}x_{2d}^{-1}x_{1d} + 248.43x_{2d}^{0.5}x_{1d}^{0.5}\Big)e_5 + 1575.8x_{2d}^{-0.5}x_{1d}^{0.5}e_4e_5 + \\
& +\Big(2008300x_{2d}^{0.5}x_{1d}^{-0.5}(1 - e_3) + 428780x_{2d}^{0.21429}x_{1d}^{-0.5}\Big)e_4 - \big(165.6x^*_{3d} - 1179.3\big)e_2 - \\
& -204680x_{2d}^{-0.71429}e_1\Big]\Big[\big(-386.46 + 1410.4x_{2d}^{0.28571}\big)e_1 - 4037.7x_{2d}^{1.5}x_{1d}^{-0.5}e_4\Big] + \\
& +\frac{1}{{x^*_{3d}}^2}\Big[-\Big(28.332x_{2d}^{0.5}x_{1d}^{0.5}e_4e_5 - 12.769e_1e_2 - 0.62132x^*_{3d}M_{Load}\Big) + \\
& +x^*_{3d}\Big(\big(-16.904x_{2d}x_{1d} + 1.3846x_{1d}\big)e_5 - \big(80.057x_{2d} + 139.69\big)e_2 - 47458M_{Load}\Big)\Big] \cdot \\
& \cdot\Big[3151.6x_{2d}^{0.5}x_{1d}^{0.5}e_4e_5 - 1420.4e_1e_2 - 69.115x^*_{3d}M_{Load}\Big]\Big\} \ \Big/ \\
& \Big/ \ \Big\{0.064381\Big[\big(-17.483x^*_{3d}x_{2d} + 1.4321x^*_{3d}\big)e_5 + 1575.8x_{2d}^{0.5}x_{1d}^{-0.5}e_4e_5 + \\
& +2008300x_{2d}^{1.5}x_{1d}^{-1.5}(e_3 - 1)e_4\Big] + 4.7712\Big[\big(-17.483x^*_{3d}x_{1d} - 1.4321x^*_{3d}x_{2d}^{-1}x_{1d} + \\
& +248.43x_{2d}^{0.5}x_{1d}^{0.5}\big)e_5 + 1575.8x_{2d}^{-0.5}x_{1d}^{0.5}e_4e_5 - \big(165.6x^*_{3d} - 1179.3\big)e_2 + \\
& +\Big(2008300x_{2d}^{0.5}x_{1d}^{-0.5}(1 - e_3) + 428780x_{2d}^{0.21429}x_{1d}^{-0.5}\Big)e_4 - 204680x_{2d}^{-0.71429}e_1\Big]\Big\} \ ,
\end{aligned}$$

where x_{1d} (the dimensionless version of x_1) is given explicitly:

$$\begin{aligned} x_{1d} &= \left[255.54{x^*_{3d}}^{-1}x_{2d}^{2.5} - 222.89{x^*_{3d}}^{-1}x_{2d}^{2.21429} + 503.85{x^*_{3d}}^{-1}x_{2d}^{1.5} - 439.46{x^*_{3d}}^{-1}x_{2d}^{1.21429}\right]^2 \cdot \\ &\cdot\left[35.966x_{2d}^2 - 189.80{x^*_{3d}}^{-1} - 135.97x_{2d} + 2.5697x_{2d}^{0.71429} + 157.37x_{2d}^{1.2857} - 232.12 + 274.61x_{2d}^{0.2857} + \right. \\ &\left. + 947.32{x^*_{3d}}^{-1}x_{2d} - 1120.7{x^*_{3d}}^{-1}x_{2d}^{1.2857} - 31.371x_{2d}^{1.71429} + 224.53{x^*_{3d}}^{-1}x_{2d}^{0.2857} + 0.13022M_{Load}\right]^{-2}, \end{aligned}$$

and the expressions $e_1, \ldots, e_5$ are:

$$\begin{aligned} e_1 &= 0.11659x^*_{3d}x_{2d} + 0.20344x_{3d} - 0.83029x_{2d} + 0.16635 \\ e_2 &= -426.35 + 504.36x_{2d}^{0.28571} \\ e_3 &= 0.14323 + 0.74731x_{2d}^{-0.28571} \\ e_4 &= -0.011095x^*_{3d}x_{2d}^{0.5}x_{1d}^{0.5} + 0.000889x^*_{3d}x_{2d}^{-0.5}x_{1d}^{0.5} + 0.078826x_{2d} + 0.15542 \\ e_5 &= 637.22x_{2d}x_{1d}^{-1}(1 - e_3) \quad . \end{aligned}$$

The substitution of x_{1d} and $e_1, \ldots, e_5$ to the long differential equation for x_{2d} gives the zero dynamics of the gas turbine model for the rotational speed.

C. Függelék

Tézisek magyar nyelven

1. tézis *Egy kisteljesítményű gázturbina QP modellje zéró dinamikáinak stabilitásvizsgálata* (3. fejezet)
(**[P1],[P2],[P3],[P4]**)
Egy kisteljesítményű gázturbina irodalomból vett harmadrendű nemlineáris QP modellje [7] két különböző zéró dinamikájának lokális stabilitását analizáltam.

A turbina belépő nyomásra mint kimenetre *vonatkozó zéró dinamika* lokális stabilitását LV rendszerekre kifejlesztett módszer [68] segítségével vizsgáltam meg. A stabilitási környezet megbecslésével megmutattam, hogy a zéró dinamika stabil egy jellemző munkapont nagy környezetében (a működési tartomány 56 %-a). Szimulációk segítségével megmutattam, hogy a becslés konzervatív: a valódi stabilitási tartomány nagyobb, mint a becsült. A fenti módszer alkalmazásával más állandósult állapotokhoz tartozó zéró dinamikák stabilitási környezeteinek becslésével az eredményeket általánosítottam és feltártam a becslés konzervativitásának lehetséges okait.

A fordulatszámra vonatkozó zéró dinamikát nem-QP formája és algebrai komplexitása miatt fázisdiagramok segítségével vizsgáltam meg. Az egyensúlyi pont egyértelműnek és stabilnak bizonyult a teljes működési tartományon, tetszőleges konstans fordulatszám- és terhelésértékek esetén.

E tézis eredményei szolgáltak alapul a kisteljesítményű gázturbina szabályzóstruktúrájának megválasztásánál.

2. tézis *Szabályzótervezés a kisteljesítményű gázturbinára* (4. fejezet)
(**[P2],[P3],[P4],[P5],[P6]**)
Input-output linearizáláson alapuló szabályzóstruktúrát választottam a gázturbina fordulatszámának szabályozására. Három szabályzót terveztem különböző szabályozási célok megvalósítására:

(a) LQ-szervó szabályzó, amely egy szakaszonként konstans fordulatszám referenciajelet követ, a terhelési nyomaték időfüggvénye ismert;

(b) LQ+MPT szabályzó, amely a fordulatszámot és annak idő szerinti deriváltját előre definiált korlátok között tartja, a terhelési nyomaték mérhető;

(c) egy új, adaptív LQ-szervó szabályzó, amely (a) kiterjesztése: *a terhelési nyomaték időfüggvénye ismeretlen, amelyet a szabályzó adaptívan becsül.*

Szimulációk segítségével megmutattam, hogy mindhárom szabályzó garantálja a zárt rendszer robusztusságát mind a modell paraméterek bizonytalanságaival, mind a környezeti zavarásokkal szemben.

Az (a) és (b) esetben a terhelési nyomatékot ismertnek tekintettem, holott legtöbb esetben ez nem mérhető környezeti zavarás. A (c) szabályzóval ezt a valósághű esetet kezeltem egy újszerű megközelítéssel: a terhelést egy dinamikus visszacsatolás becsli egy adaptív input-output linearizáló szabályzó segítségével.

MATLAB/SIMULINK környezetben végzett 'legrosszabb eset'-re vonatkozó szimulációk segítségével megmutattam, hogy a szabályzó helyesen becsli a terhelési nyomaték időfüggvényét, továbbá a fordulatszám jelkövetése robusztus mind a környezeti zavarásokkal, mind a modell paraméter bizonytalanságokkal szemben. Annak ellenére, hogy a terhelési nyomaték csupán becsült, a szabályzott rendszer az (a) esettel (ismert terhelési nyomaték) megegyező robusztusságot mutat.

Ezen eredmény fontosságát jól mutatja, hogy ezt a megközelítést gázturbinákra még nem alkalmazták, annak ellenére, hogy a terhelési nyomaték a legfontosabb, egyedüli nem mérhető, változékony környezeti zavarás, amely a gázturbina időbeni viselkedésére jelentős hatással bír.

3. tézis *QP rendszerek invariánsainak (első integráljainak) meghatározása* (5. fejezet)
(**[P7],[P8],[P9],[P10],[P11]**)
Kifejlesztettem egy új, egyszerű mátrix-vektor műveleteken alapuló, heurisztikus lépésektől mentes algoritmust, amely QP rendszerek QP típusú, explicit alakra hozható invariánsainak megkeresésére alkalmas.

Ezen algoritmus két különböző verzióját mutattam be: ezek egy, illetve több invariáns megkeresésére alkalmasak. Megmutattam, hogy mindkét változat polinomiális idejű, magas dimenziójú és nagy monomszámú QP modellek esetén is hatékony.

Az algoritmus mindkét változatát implementáltam MATLAB környezetben, működésüket sikeresen teszteltem számos fizikai rendszer matematikai modelljén.

Megvizsgáltam az algoritmus invariancia tulajdonságait két különböző transzformációval szemben, megmutatván ezzel az algoritmus képességeit és korlátait. Könnyű alkalmazhatósága, egyszerűsége és hatékonysága miatt - különösen nagy monomszámú QP modellek esetén -, az általam készített algoritmus mind futási idő, mind találati hatékonyság szempontjából jobbnak bizonyult, mint az irodalomból ismert, szintén QP rendszerekre tervezett QPSI invariáns kereső algoritmus [35].

Bibliography

[1] *Matlab User's Guide*. The Math Works, Inc., Natick, Massachusetts, USA, 2000.

[2] *Symbolic Math Toolbox User's Guide*. The Math Works, Inc., Natick, Massachusetts, USA, 2005.

[3] M. J. Ablowitz, A. Ramani, and H. Segur. A connection between nonlinear evolution equations and ordinary differential equations of P-type I. *Journal of Mathematical Physics*, 21:715–721, 1980.

[4] B. Abraham-Shrauner. Hidden symmetries, first integrals and reduction of order of nonlinear ordinary differential equations. *Journal of Nonlinear Mathematical Physics*, 9(Suppl.2):1–9, 2002.

[5] P. Ailer. Nonlinear mathematical modeling and control design developed for gas turbine. In I. Zobory, editor, *Proceedings of the 7th Mini Conference on Vehicle System Dynamics, Identification and Anomalies*, pages 465–472, Budapest, Hungary, 2000. Budapest University of Technology and Economics.

[6] P. Ailer. Modelling and nonlinear control of a low-power gas turbine. PhD Thesis, In Hungarian. *Budapest University of Technology and Economics, Department of Aircraft and Ships*, 2002.
`http://daedalus.scl.sztaki.hu/PCRG/works/PhD-Ailer-2002.pdf`.

[7] P. Ailer, I. Sánta, G. Szederkényi, and K. M. Hangos. Nonlinear model-building of a low-power gas turbine. *Periodica Polytechnica Ser. Transportation Engineering*, 29(1-2):117–135, 2001.

[8] P. Ailer, G. Szederkényi, and K. M. Hangos. Modeling and nonlinear analysis of a low-power gas turbine. Technical report, SCL-001/2001 *Computer and Automation Research Institute, Hungarian Academy of Sciences*, 2001.
`http://daedalus.scl.sztaki.hu/ps/research reports/Modeling and nonlinear analysis of a low-power gas turbine.ps`.

[9] P. Ailer, G. Szederkényi, and K. M. Hangos. Model-based nonlinear control of a low-power gas turbine. In E. F. Camacho, L. Basanez, and J.A. de la Puente, editors, *Proceedings of the 15th IFAC World Congress on Automatic Control*, Barcelona, Spain, 2002. Elsevier Science. *On CD*.

[10] P. Ailer, G. Szederkényi, and K. M. Hangos. Parameter-estimation and model validation of a low-power gas turbine. In M. H. Hamza, editor, *Proceedings of the Modelling, Identification and Control'2002 Conference*, pages 604–609, Innsbruck, Austria, 2002. ACTA Press.

[11] P. Ailer, G. Szederkényi, and K. M. Hangos. Nonlinear control of a low-power gas turbine based on input-output linearization. In *Proceedings of the 4th International Workshop on Information Technologies and Control*, Spa Libverda, Czech Republic, 2003. *On CD.*

[12] A. E. Ariffin and N. Munro. Robust control analysis of a gas-turbine aero-engine. *IEEE Transactions on Control Systems Technology*, 5(2):178–188, 1997.

[13] K. J. Aström and T. Hägglund. *Advanced PID Control.* ISA, 2005.

[14] M. Athans, P. Kapasouros, E. Kappos, and H. A. Spang. Linear-Quadratic Gaussian with Loop-Transfer Recovery methodology for the F-100 engine. *IEEE Journal of Guidance and Control*, 9(1):45–52, 1986.

[15] R. E. Beardmore and Y. H. Song. Differential-algebraic equations: A tutorial review. *International Journal of Bifurcation and Chaos*, 8:1399–1411, 1998.

[16] K. Belda and J. Böhm. Predictive control of redundant parallel robots and trajectory planning. In R. Neugebauer, editor, *Proceedings of the Parallel Kinematic Machines in Research and Practice. PKS 2006*, pages 497–513, IWU, Chemnitz, Germany, 2006.

[17] A. Bemporad, F. Borrelli, and M. Morari. Explicit solution of LP-based model predictive control. In *Proceedings of the 39th IEEE Conference on Decision and Control*, pages 823–828, Sydney, Australia, 2000.

[18] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopulous. The explicit linear quadratic regulator for constrained systems. *Automatica*, 38(1):3–20, 2002.

[19] T. Bohlin and S. F. Greabe. Issues in nonlinear stochastic grey box identification. *International Journal of Adaptive Control and Signal Processing*, 9:465–490, 1995.

[20] F. Borrelli. *Constrained Optimal Control of Linear and Hybrid Systems, vol. 290 of Lecture Notes in Control and Information Sciences.* Springer, Berlin, Germany, 2003.

[21] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory.* SIAM studies in Applied Mathematics, Philadelphia, USA, 1994.

[22] L. Brenig and A. Goriely. Painlevé analysis and normal forms. In E. Tournier, editor, *London Mathematical Society Lecture Note Series*, volume 193, pages 211–237. Cambridge University Press, Cambridge, UK, 1994.

[23] W. L. Brogan. *Modern Control Theory*. Prentice Hall, 1991.

[24] D. de Bruin, A. A. H. Damen, A. Y. Pogromsky, and P. P. J. van den Bosch. Backstepping control for lateral guidance of all-wheel steered multiple articulated vehicles. In *Proceedings of the IEEE Intelligent Transportation Systems Conference*, pages 95–100, Dearborn, United States, 2000.

[25] B. J. Brunell, R. R. Bitmead, and A. J. Connolly. Nonlinear model predictive control of an aircraft gas turbine engine. In *Proceedings of the 41st IEEE Conference on Decision and Control*, volume 4(10-13), pages 4649–4651, Las Vegas, Nevada, USA, 2002.

[26] G. Chesi, A. Garulli, A. Tesi, and A. Vicino. LMI-based computation of optimal quadratic Lyapunov functions for odd polynomial systems. *International Journal of Robust and Nonlinear Control*, 15(1):35–49, 2005.

[27] A. J. Chipperfield, B. Bica, and P. J. Fleming. Fuzzy scheduling control of a gas turbine aero-engine: a multiobjective approach. *IEEE Transactions on Industrial Electronics*, 49:536–548, 2005.

[28] D. Cobb. Feedback and pole placement in descriptor variable systems. *International Journal of Control*, 33:1135–1146, 1981.

[29] S. J. Colley. *Vector Calculus*. Prentice Hall, 2001.

[30] J. R. Dormand and P. J. Prince. A family of embedded Runge-Kutta formulae. *Journal of Computational and Applied Mathematics*, 6:19–26, 1980.

[31] C. Evans. Testing and modelling aircraft gas turbines: An introduction and overview. In *Proceedings of UKACC International Conference on Control'98*, pages 1361–1366, Swansea, UK, 1998.

[32] W. F. Feehery and P. I. Barton. Dynamic optimization with state variable path constraints. *Computers and Chemical Engineering*, 22:1241–1256, 1998.

[33] A. Figueiredo, T. M. Rocha Filho, and L. Brenig. Algebraic structures and invariant manifolds of differential systems. *Journal of Mathematical Physics*, 39:2929–2946, 1998.

[34] A. Figueiredo, T. M. Rocha Filho, and L. Brenig. Necessary conditions for the existence of quasi-polynomial invariants: the quasi-polynomial and Lotka-Volterra systems. *Physica A*, 262:158–180, 1999.

[35] T. M. Rocha Filho, A. Figueiredo, and L. Brenig. [QPSI] A Maple package for the determination of quasi-polynomial symmetries and invariants. *Computer Physics Communications*, 117:263–272, 1999.

[36] P. Gáspár, I. Szászi, and J. Bokor. Robust control design for mechanical systems using mixed μ synthesis. *Periodica Polytechnica ser. Mechanical Engineering*, 30(1–2):37–52, 2002.

[37] P. Gáspár, I. Szászi, and J. Bokor. Design of robust controllers for active vehicle suspensions. In *Proceedings of the 15th IFAC World Congress*, Barcelona, Spain, 2002. *On CD*.

[38] P. Gáspár, I. Szászi, and J. Bokor. Design of robust controllers for active vehicle suspension using the mixed μ synthesis. *Vehicle System Dynamics*, 40(4):193–228, 2003.

[39] R. Genesio, M. Tartaglia, and A. Vicino. On the estimation of asymptotic stability regions: State of the art and new proposals. *IEEE Transactions on Automatic Control*, 30(8):747–755, 1985.

[40] I. M. Gléria, A. Figueiredo, and T. M. Rocha Filho. A numerical method for the stability analysis of quasi-polynomial vector fields. *Nonlinear Analysis*, 52:329–342, 2003.

[41] A. Goedtel, I. N. da Silva, and P. J. A. Serni. Load torque identification in induction motor using neural networks technique. *Electric Power Systems Research*, 77:35–45, 2007.

[42] M. Gunther and U. Feldmann. The DAE-index in electric circuit simulation. *Mathematics and Computers in Simulation*, 39:573–582, 1995.

[43] K. M. Hangos, J. Bokor, and G. Szederkényi. Hamiltonian view on process systems. *AIChE Journal*, 47:1819–1831, 2001.

[44] K. M. Hangos, J. Bokor, and G. Szederkényi. *Computer Controlled Systems*. (Ed: Veszprémi Egyetemi Kiadó), ISBN: 963 9220 94 9, VE 24/2002, Veszprém, Hungary, 2002.

[45] K. M. Hangos, J. Bokor, and G. Szederkényi. *Analysis and Control of Nonlinear Process Systems*. Springer-Verlag, London, UK, 2004.

[46] K. M. Hangos and I. T. Cameron. *Process Modelling and Model Analysis*. Academic Press, London, UK, 2001.

[47] B. Hernández-Bermejo and V. Fairén. Nonpolynomial vector fields under the Lotka-Volterra normal form. *Physics Letters A*, 206:31–37, 1995.

[48] B. Hernández-Bermejo, V. Fairén, and L. Brenig. Algebraic recasting of nonlinear systems of ODEs into universal formats. *Journal of Physics A: Mathematical and General*, 31:2415–2430, 1998.

[49] A. Isidori. *Nonlinear Control Systems*. Springer, Berlin, Germany, 1995.

[50] H. Jain, V. Kaul, and N. Ananthkrishnan. Parameter estimation of unstable, limit cycling systems using adaptive feedback linearization: example of delta wing roll dynamics. *Journal of Sound and Vibration*, 287:939–960, 2005.

[51] E. Jonckheere. Variational calculus for descriptor problems. *IEEE Transactions on Automatic Control*, 33:491–496, 1988.

[52] F. Jurado and J. Carpio. Improving distribution system stability by predictive control of gas turbines. *Energy Conversion and Management*, 47:2961–2973, 2006.

[53] E. Kaslika, A. M. Balint, and St. Balint. Methods for determination and approximation of the domain of attraction. *Nonlinear Analysis*, 60(4):703–717, 2005.

[54] K. Kawai, Y. Takizawa, and S. Watanabe. Advanced automation for power-generation plants - past, present and future. *Control Engineering Practice*, 7(11):1405–1411, 1999.

[55] P. V. Kokotovic and M. Arcak. Constructive nonlinear control: a historical perspective. *Automatica*, 37(5):637–662, 2001.

[56] C. D. Kong and S. C. Chung. Real time linear simulation and control for small aircraft turbojet engine. *KSME International Journal*, 13:656–666, 1999.

[57] B. Kulcsár. LQG/LTR controller design for an aircraft model. *Periodica Polytechnica ser. Transportation Engineering*, 28(1–2):131–142, 2000.

[58] G. G. Kulikov and H. A. Thompson. *Dynamic Modelling of Gas Turbines: Identification, Simulation, Condition Monitoring and Optimal Control*. Springer-Verlag, 2004.

[59] A. Kumar and P. Daoutidis. *Control of nonlinear differential algebraic equation systems*. Chapman and Hall/CRC, London, UK, 1999.

[60] Z. Kurd and T. P. Kelly. Using safety critical artificial neural networks in gas turbine aero-engine control. In *Lecture Notes in Computer Science*, volume 3688, pages 136–150, 2005.

[61] M. Kvasnica, P. Grieder, M. Baotic, and F. J. Christophersen. *Multi-Parametric Toolbox (MPT)*. Automatic Control Laboratory, Swiss Federal Institute of Technology, Zurich, Switzerland, 2004.

[62] B. Lantos. *Fuzzy systems and genetic algorithms*. Műegyetemi Kiadó, Budapest, Hungary, 2002.

[63] P. De Leenheer and D. Aeyels. Stabilization of positive systems with first integrals. *Automatica*, 38:1583–1589, 2002.

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

[65] S. T. Lin and L. W. Yeh. Intelligent control of the F-100 turbofan engine for full flight envelope operation. *International Journal of Turbo & Jet-Engines*, 22:201–213, 2005.

[66] B. Lipták. *Instrument engineers' handbook. Process control*. Chilton Book Company, Radnor, Pennsylvania, USA, 1995.

[67] J. Llibre and X. Zhang. Polynomial first integrals for quasi-homogeneous polynomial differential systems. *Nonlinearity*, 15:1269–1280, 2002.

[68] A. Magyar, G. Szederkényi, and K. M. Hangos. Quadratic stability of process systems in generalized Lotka-Volterra form. In *Proceedings of the 6th IFAC Symposium on Nonlinear Control (NOLCOS 2004)*, Stuttgart, Germany, 2004. *On CD.*

[69] A. Magyar, G. Szederkényi, and K. M. Hangos. Quasi-polynomial system representation for the analysis and control of nonlinear systems. In P. Horacek, M. Simandl, and P. Zitek, editors, *Proceedings of the 16th IFAC World Congress*, pages 1–6, paper ID: Tu–A22–TO/5. Prague, Czech Republic, 2005.

[70] R. Marino, S. Peresada, and P. Tomei. Adaptive output feedback control of current-fed induction motors with uncertain rotor resistance. *Automatica*, 34:617–624, 1998.

[71] R. Marino and P. Tomei. *Nonlinear Control Design: Geometric, Adaptive, and Robust.* Prentice Hall, London, UK, 1995.

[72] R. Marino, P. Tomei, and C. M. Verrelli. Adaptive control for speed-sensorless induction motors with uncertain load torque and rotor resistance. *International Journal of Adaptive Control and Signal Processing*, 19(9):661–685, 2005.

[73] H. J. Marquez. *Nonlinear Control Systems: Analysis and Design.* Wiley-Interscience, New Jersey, USA, 2003.

[74] J. Mu, D. Rees, and G. P. Liu. Advanced controller design for aircraft gas turbine engines. *Control Engineering Practice*, 13:1001–1015, 2004.

[75] P. C. Müller. Descriptor systems: pros and cons of system modelling by differential-algebraic equations. *Mathematics and Computers in Simulation*, 53:273–279, 2000.

[76] C. Muriel and J. L. Romero. C-infinity symmetries and reduction of equations without Lie point symmetries. *Journal of Lie Theory*, 13:167–188, 2003.

[77] R. C. Nelson. *Flight Stability and Automatic Control.* Prentice Hall, London, UK, 1998.

[78] P. Oliveira, A. Pascoal, and I. Kaminer. A nonlinear vision based tracking system for coordinated control of marine vehicles. In *Proceedings of the 15th IFAC World Congress*, Barcelona, Spain, 2002. *On CD.*

[79] T. Péni and J. Bokor. Trajectory tracking control for a class of LPV systems based on dynamic inversion and passivity. In *Proceedings of the 2004 WSEAS/IASME International Conference*, Corfu, Greece, 2004. *On CD.*

[80] R. A. Perez. Model reference control of a gas turbine engine. In *Proceedings of the Institution of Mechanical Engineers - Part G: Journal of Aerospace Engineering*, volume 210, pages 291–296, Wisconsin, USA, 1996.

[81] W. Perruquetti and J. P. Barbot. *Sliding Mode Control in Engineering*. Marcel Dekker, Oregon, USA, 2002.

[82] V. Pierce. Painlevé analysis and integrability. *The Nonlinear Journal*, 1:41–49, 1999.

[83] A. Rehm and F. Allgöwer. General quadratic performance analysis and synthesis of differential algebraic equation (DAE) systems. *Journal of Process Control*, 12:467–474, 2002.

[84] S. Reich. On the local qualitative behavior of differential-algebraic equations. *Circuits, Systems, and Signal Processing*, 14(4):427–443, 1995.

[85] T. Rikitake. Oscillations of a system of discs dynamos. In *Proceedings of the Cambridge Philosophical Society*, volume 54, pages 89–105, Cambridge, UK, 1958.

[86] G. Rödönyi, J. Bokor, and B. Lantos. LQG control of LPV systems with parameter-dependent Lyapunov function. In *Proceedings of the 10th IEEE Mediterranean Conference on Control and Automation - MED2002*, Lisboa, Portugal, 2002. *On CD*.

[87] J. La Salle and S. Lefschetz. *Stability by Liapunov's direct method with applications*. Academic Press, New York, USA, 1961.

[88] A. J. van der Schaft. *L2-Gain and Passivity Techniques in Nonlinear Control*. Springer, London, UK, 2000.

[89] C. Scherer and S. Weiland. *Lecture notes DISC course on Linear Matrix Inequalities in Control*. 1999.
`http://www.dcsc.tudelft.nl/~cscherer/2416/lmi.pdf`.

[90] F. Schwartz. Symmetries of differential equations: from Sophus Lie to computer algebra. *SIAM Review*, 30:450–481, 1988.

[91] I. Szászi and B. Kulcsár. Robust control and fault detection filter for aircraft pitch axis. *Periodica Polytechnica ser. Transportation Engineering*, 29(1–2):83–100, 2001.

[92] G. Szederkényi and K. M. Hangos. Global stability and quadratic Hamiltonian structure in Lotka-Volterra and quasi-polynomial systems. *Physics Letters A*, 324:437–445, 2004.

[93] G. Szederkényi, K. M. Hangos, and A. Magyar. On the time-reparametrization of quasi-polynomial systems. *Physics Letters A*, 334:288–294, 2005.

[94] G. Szederkényi, M. Kovács, and K. M. Hangos. Reachability of nonlinear fedbatch fermentation processes. *International Journal of Robust and Nonlinear Control*, 12:1109–1124, 2002.

[95] G. Szederkényi, N. R. Kristensen, K. M. Hangos, and S. B. Jørgensen. Nonlinear analysis and control of a continuous fermentation process. *Computers and Chemical Engineering*, 26:659–670, 2002.

[96] X. Tang, G. Tao, and S. M. Joshi. Adaptive output feedback actuator failure compensation for a class of non-linear systems. *International Journal of Adaptive Control and Signal Processing*, 19(6):419–444, 2005.

[97] A. Vannelli and M. Vidyasagar. Maximal Lyapunov functions and domains of attraction for autonomous nonlinear systems. *Automatica*, 21(1):69–80, 1985.

[98] G. Verghese, B. Levy, and T. Kailath. A generalized state space for singular systems. *IEEE Transactions on Automatic Control*, 26:811–831, 1981.

[99] J. W. Watts, T. E. Dwan, and C. G. Brockus. Optimal state-space control of a gas turbine engine. *Transactions of the ASME, Journal of Engineering for Gas Turbines and Power*, 91-GT-219:1–10, 1991.

[100] J. W. Watts, T. E. Dwan, and R. W. Garman. A model and state-space controllers for an intercooled, regenerated (ICR) gas turbine engine. *Transactions of the ASME, Journal of Engineering for Gas Turbines and Power*, 92-GT-43:1–12, 1992.

[101] B. Wie. *Space Vehicle Dynamics and Control*. AIAA, 1998.

[102] G. Wolodkin, G. J. Balas, and W. L. Garrard. Application of parameter-dependent robust control synthesis to turbofan engines. *Journal of Guidance, Control and Dynamics*, 22(6):833–838, 1999.

[103] L. Xiaoping and S. Celikovsky. Feedback control of affine nonlinear singular control systems. *International Journal of Control*, 68:753–774, 1997.

[104] E. L. Yip and R. F. Sincovec. Solvability, controllability, and observability of continuous descriptor systems. *IEEE Transactions on Automatic Control*, 26:702–707, 1981.

[105] K. Zhou, J. C. Doyle, and K. Glover. *Robust and Optimal Control*. Prentice Hall, London, UK, 1996.

[106] Y. H. Zweiri and L. Seneviratne. Diesel engine indicated and load torque estimation using a non-linear observer. In *Proceedings of the Institution of Mechanical Engineers Part D - Journal of Automobile Engineering*, volume 220, pages 775–785, London, UK, 2006. Professional Engineering Publishing.