NONDETERMINISTIC HYBRID DYNAMICAL SYSTEMS

A DISSERTATION
SUBMITTED TO THE FACULTY OF ENGINEERING
UNIVERSITY OF GLASGOW
IN FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

By
Michael Schinkel
November 2002

Dedication:

To My Father and My Mother

Abstract

This thesis is concerned with the analysis, control and identification of hybrid dynamical systems. The main focus is on a particular class of hybrid systems consisting of linear subsystems. The discrete dynamic, i.e., the change between subsystems, is unknown or nondeterministic and cannot be influenced, i.e. controlled, directly. However changes in the discrete dynamic can be detected immediately, such that the current dynamic (subsystem) is known.
In order to motivate the study of hybrid systems and show the merits of hybrid control theory, an example is given. It is shown that real world systems like Anti Locking Brakes (ABS) are naturally modelled by such a class of linear hybrid systems. It is shown that purely continuous feedback is not suitable since it cannot achieve maximum braking performance. A hybrid control strategy, which overcomes this problem, is presented.
For this class of linear hybrid system with unknown discrete dynamic, a framework for robust control is established. The analysis methodology developed gives a robustness radius such that the stability under parameter variations can be analyzed. The controller synthesis procedure is illustrated in a practical example where the control for an active suspension of a car is designed.
Optimal control for this class of hybrid system is introduced. It is shown how a control law is obtained which minimizes a quadratic performance index. The synthesis procedure is stated in terms of a convex optimization problem using linear matrix inequalities (LMI). The solution of the LMI not only returns the controller but also the performance bound.
Since the proposed controller structures require knowledge of the continuous state, an observer design is proposed. It is shown that the estimation error converges quadratically while minimizing the covariance of the estimation error. This is similar to the Kalman filter for discrete or continuous time systems. Further, we show that the synthesis of the observer can be cast into an LMI, which conveniently solves the synthesis problem.
In order to obtain linear hybrid models in the first place, system identification techniques are used. Theoretical issues are discussed in the last section of the thesis. To support the methodology, convergence conditions are derived. Conditions under which the classification problem can be solved are given. These are conditions on noise level and parameters under

which input/output data can be grouped such that only data which is generated by the same subsystem is collected.

Acknowledgements

With the completion of a thesis one usually looks back on a couple of years hard research work and remembers the many persons who contributed to the success of the thesis. First of all, I would like to thank my advisor Ken Hunt for making it all possible. I never would have thought that I would come to Scotland for a PhD but after his invitation and impressive overview of the "Centre of Systems and Control" I couldn't possibly study anywhere else. I really enjoyed the years here in Glasgow, and therefore I would like to thank Markus Rothe and William Holderbaum and all the others with whom I had a good time. Also thanks to my office mate Yongji Wang.

While conducting research work I was grateful for the long discussions I had with William and Wen-Hua Chen. These controversial discussions helped me a lot and gave me the necessary "feedback". I'm also grateful that I could participate in the long term research project H^2C (Heterogeneous Hybrid Control). Within this framework, I had the opportunity to meet many interesting researchers in the field of hybrid systems. As part of this project it was also possible to visit the "Department of Automatic Control" in Lund and work with Anders Rantzer, which was a great pleasure. I felt very welcome in Lund and therefore I would like to thank everybody in the department.

Through the MacRobertson scholarship it was possible to visit the "Institut für Automatik" with Manfred Morari. It was a successful visit and I'm quite happy that I could work with Giancarlo Ferrari-Trecate, who taught me everything about identification of hybrid systems. I would also like to thank Elaine McNamara and Jane Livingston for proof reading this thesis. Also special thanks for the financial support by:

The European Commission through the Long Term Research Project 28104, H^2C (Heterogeneous Hybrid Control).

The UK Engineering and Physical Sciences Research Council through grant GR/M47256.

Contents

Abstract ii

Acknowledgements iv

I Introduction 1

1 Introduction 2

1.1 The notion of hybrid dynamical systems 2

1.2 Survey over hybrid systems 4

1.2.1 Modelling hybrid systems 4

1.2.2 Analysis of hybrid systems 7

1.2.3 Control of hybrid systems and hybrid control 12

1.2.4 Identification of hybrid systems 17

1.3 Summary 18

2 Outline of the thesis: 20

2.1 Overview of thesis contents 20

2.2 Thesis contributions 24

2.2.1 Publications 25

3 Control of Anti-Lock Brake with heterogenous hybrid controller 27

3.1 History of ABS 27

3.2 Modelling of the longitudinal dynamics 28

3.3 Analysis of the ABS dynamics 32

3.3.1 Stability analysis 32

3.3.2 Computation of the maximum deceleration 33

3.4 Controller design 34

3.5 Simulation 36

3.6 Conclusions 37

II Controller design for Hybrid Systems 39

4 Robust control of hybrid systems 40

4.1 Introduction 40

4.2 Calculation of the stable switched set 42

4.3 On existence of a common state feedback controller 45

4.4 Extension to Multiple State Feedback Control 47

4.5 Active suspension control 50

5 Optimal control of hybrid systems in discrete time 55

5.1 Introduction 56

5.1.1 Example 1: Two different sampling times, same continuous loss function in both 57

5.2 Stable scheduling strategies 61

5.3 Controller design 62

5.4 Controller synthesis using LMI 65

5.4.1 Computations of lower bounds on the optimal cost 68

5.5 Example 70

5.6 Conclusion 71

6 Optimal control of hybrid systems in continuous time **73**

6.1 Introduction 73

6.2 Optimal control of hybrid systems in continuous time 74

6.2.1 Computation of upper bounds on the optimal cost 74

6.2.2 Computations of lower bounds on the optimal cost 77

6.3 Conclusion 78

III Observer design for Hybrid Systems **80**

7 Observer design for hybrid systems **81**

7.1 Introduction 81

7.2 Example 1: Two different sampling rates, asymptotic observer for each sampling rate 83

7.3 Observer for systems with varying sampling rate 86

7.4 Observer synthesis using LMI 87

7.5 Example 89

7.6 Conclusion 91

IV Identification of Hybrid Systems **92**

8 Identification of hybrid systems in discrete time **93**

8.1 Introduction 93

8.2 The identification algorithm 95

8.3 Perfect clustering without outliers 98

8.4 Perfect clustering with outliers 105

8.5 Mixed LDs: a posteriori detection 106

8.6 Example 108

8.7 Conclusions 110

V Conclusions and Outlook **111**

9 Summary and Conclusions **112**

9.1 Overview 112

9.2 Contributions 114

9.3 Open problems 115

VI Appendix **117**

Bibliography **118**

List of Tables

5.1 Unstable sequences 59

7.1 Sequences which do not converge 85

List of Figures

1.1 Trajectory of hybrid system 9

1.2 Bouncing ball example for Zeno executions 11

3.1 Quarter car slip model 29

3.2 $\mu(\lambda)$ for wet and dry surfaces and winter tyres 30

3.3 Approximation of $\mu(\lambda)$ with uncertain piecewise linear functions 31

3.4 Velocity of the car body and the wheel $\omega * r$. 36

3.5 Wheel slip λ 36

3.6 Brake torque $\hat{T}_b$ 37

4.1 Computing the stable switched set 44

4.2 Set of state feedbacks which bring the subsystem into SSS 46

4.3 Common state feedback brings subsystems into SSS 46

4.4 Active suspension of a quarter car 50

4.5 Spring coefficient 51

5.1 Unstable sequence 59

5.2 Unstable sequence 60

7.1 Divergent sequence of the estimation error 86

8.1 Clustered LPV containing an outlier 108

8.2 Clustered LPV after removing the misclassified data points 109

8.3 Projection with and without outlier into 1st and 2nd parameter plane 109

8.4 Projection with and without outlier into 2nd and 3rd parameter plane 110

Part I

Introduction

1 Introduction

This chapter gives an introduction to a class of dynamical systems known as hybrid systems. A survey covering current research results in the field of hybrid dynamical systems is given. This survey contains an overview of current modelling frameworks as well as analysis, controller designs and identification methods. The survey shows that many areas in the field of hybrid systems are quite mature and covered very well. However, little has been found about hybrid systems with unknown discrete dynamic, especially in context of robustness analysis and robust controller design. Also hardly any contributions are found for the optimal control of hybrid systems with unknown discrete dynamics. This is also true for identification of hybrid systems, where nearly no results have been found.

This thesis tries to close some of these gaps in theory and makes contributions to robustness analysis and controller design, optimal control, state estimation and identification with the focus on hybrid systems with unknown discrete dynamics.

1.1 The notion of hybrid dynamical systems

In recent years hybrid dynamical systems have been an increasingly popular subject. One of the reasons is that conventional methods were limited and failed to model, analyze and control such systems. In order to strive for higher accuracy and better performance it has been necessary to pursue research work in the area of hybrid dynamical systems. In this introduction we shall explain the properties of hybrid dynamical systems, usually referred to as hybrid systems, and give the history of the research work carried out. Furthermore, we show the clear advantages of using hybrid systems theory in comparison to the limited possibilities of conventional, purely continuous or purely discrete, theory. Since this is rather a complex topic and its impact is broad, we can only cover some of the current and previous achievements in the field of hybrid systems research. However before we start we need to get some notion of what hybrid systems are.

What is a hybrid system? Roughly speaking, hybrid systems combine two basic dynamic notions, namely continuous dynamics and discrete dynamics. Systems like computers, automata, switches etc., are considered systems with discrete dynamics since their states assume only distinct values. The set of values which can be assumed is often finite. For example a switch can be off or on, i.e. the discrete state can assume two distinct values, while systems like the weather, the motion of planets and the flow of water are considered continuous systems, since their states change continuously. For instance the temperature of our weather does not change from 10 C° to 20 C° discontinuously, it changes continuously and probably smoothly as well, such that all temperatures between 10 C° to 20 C° are assumed, although each temperature might be assumed only for an infinitesimally short time. Research work of previous decades and centuries has focused on describing such systems. This was usually done using differential equations. Later, especially with the invention of digital computers and automation, discrete time systems came into focus. Such systems were then described by difference equations or by logic statements (if, then, else, or, and etc.). For many systems however it is not immediately obvious to which class they should belong, since the decision whether a system should be classified as continuous or discrete depends strongly on the level of abstraction. This is often the case with continuous systems being controlled by discrete inputs, where the continuous dynamics are fast. One simple example is a desk-light, which can be switched on or off. For an observer the system might appear to be discrete, since by his observation the light is on, when switched on and off otherwise. Of course, in between, fast continuous transients take place. In other systems the discrete dynamics are as obvious as the continuous dynamics. A popular example of such systems is the often used electrical radiator. The heating is switched on if the temperature is below a certain threshold and switched off again if it reaches above a specified threshold. The continuous part of the radiator is given by the equation of the electric circuit

$$L\frac{di}{dt} + i \cdot R = V_s \tag{1.1}$$

and the equation for the temperature T

$$dT = \frac{1}{C}dQ \qquad \frac{dQ}{dt} = i^2 \cdot R \tag{1.2}$$

The equations above form the continuous part of the system. The discrete part is formed by a thermostat that could switch the voltage on and off according to some inequalities.

$$V_s = \begin{cases} 0 & T > T_1 \\ V & T \leq T_2 \end{cases} \tag{1.3}$$

With this system involving discrete dynamics (off-on switching) well as the continuous dynamics, the increase or decrease of the temperature can be observed. This is the case since the continuous dynamics are slow and the system admits a hysteresis. It is now clear that there are systems which exhibit discrete as well as continuous dynamics; such systems are called hybrid systems. If we take a closer look we will find many such systems in our homes: washing machines, electric food processors, cars etc.. The hybrid phenomenon is not only limited to technical systems, it can be observed in other physical systems such as mechanical systems with mode transition between slip and stick phases, and in systems with impacts or constraints etc.. This immediately raises the question of how to describe (model) such systems in order to analyze them. Models are desirable for various reasons: one is to capture the behaviour in order to analyze it, another is to make predictions of the behaviour for cases where the real system would be endangered or for situations where it is costly to do experiments. Properties like stability can be assessed, or it can be checked if certain states can be reached, reachability analysis. Often we would like to influence such systems so that they show a desired behaviour, i.e. we would like to control the system. Therefore we need the right methodology which is supported by theory.

In the following we will give a short overview of some frameworks for hybrid systems. It has to be mentioned that there are various frameworks which admit modelling different classes of hybrid systems. Since hybrid systems are a very rich class of dynamical system, which includes linear, nonlinear, constrained, non-constrained, continuous, discrete and logical systems and all kinds of overlaps between these classes, it is clear that it makes sense to focus on various subclasses of hybrid systems. In order to get sensible results, the subclass needs to be small enough to carry additional structure, facilitating detailed analysis of its behavior and controller design. The class needs to be large enough to contain real applications which are of interest. The areas of interest are diverse, since applications come from computer science and control engineering. Therefore frameworks have different foci.

1.2 Survey over hybrid systems

1.2.1 Modelling hybrid systems

Models:

There are various models for hybrid systems; all have in common that they describe continuous as well as discrete dynamics. However the emphasis is different; while some are more concerned with the logical or discrete part, others are more focused on the continuous dynamics. Some classes admit broad analysis but real applications which belong to these classes are rarely found. One such example is the batch integrator system by Tittus (75). Tittus models batch processes. The batch processes consist of continuous flows of material and energy with discrete actuators and sensors. The modelling is done with integrator processes. The class of hybrid systems is very limited but these models are important for the control of batch processes. Using such simple models he is able to derive results showing stability and controllability of the systems. The framework by Branicky (22), on the other hand, admits a very broad class of hybrid systems, such that many other frameworks are contained within it. However in terms of analysis and controller design only very general statements can be made.

Many of the available frameworks for modelling hybrid systems have their origin either in the discrete event community or stem from the dynamic systems community. These frameworks were basically only extended to incorporate the additional dynamic. Examples for modified frameworks from the discrete event community are timed or hybrid Petri-nets (26), hybrid automata (22) etc., while examples of modified frameworks with origin in the dynamic systems community are switched bond graphs (27), etc.. However there are also frameworks which do not result from direct modifications of existing frameworks, such as mixed logical dynamic systems (MLD) (17), (18), or complementarity systems (39). Looking at the numbers of different frameworks, it is not surprising that some frameworks are only capable of modelling a subset of others and this is shown in (22). It was also shown that some frameworks are equivalent (40), such that models from one framework can be transferred into another. This has the advantage that analysis results, which have been derived for one class, might apply to the other as well.

In the following we will briefly review some frameworks; for the interested reader references are given.

Piecewise affine systems:

Piecewise affine systems are without doubt among the earliest classes of systems in the literature which admit the properties of hybrid systems (47), although they are often not directly mentioned in the hybrid systems literature. Since each affine dynamic is valid only on a certain domain X_q (or for a certain time), we have the mixture of continuous dynamics and discrete dynamics, which form a hybrid dynamical system.

$$\dot{x} = A_q x + a_q \quad for \quad x \in X_q \quad q \in Q = \{1, 2, \cdots, N\} \tag{1.4}$$

where $x \in \bigcup X_q \subset \mathbf{R}^n$ is the continuous state of the N affine dynamics $\dot{x} = A_q x + a_q$ and $q \in Q = \{1, 2, \cdots, N\} \subset Z^+$ is the discrete state. The q^{th} affine dynamic is valid as long as $x \in X_q$. Piecewise affine systems often stem from linearization of complex nonlinear systems, therefore their affine dynamics are state dependent, i.e. . Usually piecewise affine systems describe systems with static nonlinearities like relays, diodes, saturations, etc. In these cases it is easy to write the nonlinearity as a piecewise affine dynamic. First attempts have already been made by Kalman (49).

Complementarity systems:

Complementarity systems were first used by van der Schaft and Schumacher (78) to describe hybrid systems. The complementarity conditions are similar to the complementarity conditions which have been used in mathematical programming. Heemels shows various applications of linear complementarity systems in his PhD thesis (39). Electric circuits and constrained mechanical systems are modelled with complementarity systems. The linear complementarity system consists of a linear dynamical part

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned}$$

and the complementarity condition

$$\{y_i = 0 \quad or \quad u_i = 0\}, \quad y_i \geq 0, \quad u_i \geq 0$$

It is obvious that electrical components like diodes transform naturally into this framework. In the forward direction there is no voltage at the diode, only current flow, and in the backward direction there is no current, only voltage. It has been shown that linear complementarity systems are equivalent to MLD systems and piecewise affine systems (40). This makes analysis results like well posedness (39) interesting since they can be transferred to the equivalent classes.

Mixed logical dynamic systems

MLD systems were introduced by Bemporad and Morari (17), (18). The MLD system consists of linear dynamic equations in discrete time subject to linear inequalities involving real and integer variables.

$$
\begin{aligned}
x_{t+1} &= A_t x_t + B_{1t} u_t + B_{2t} \delta_t + B_{3t} z_t \\
y_t &= C_t x_t + D_{1t} u_t + D_{2t} \delta_t + D_{3t} z_t \\
E_{2t} \delta_t &+ E_{3t} z_t \leq E_{1t} u_t + E_{4t} x_t + E_{5t}
\end{aligned}
$$

It is in effect a switched linear system, which is governed by logical conditions expressed by linear inequalities. This framework is capable of describing piecewise linear systems, linear hybrid systems, constrained linear systems, automata driven by dynamical systems, dynamic systems which incorporate nonlinearities that can be approximated by piecewise linear functions, and systems with discrete inputs and qualitative outputs.

Branicky

Branicky's (22) framework is given as a hybrid automaton. He makes a formal definition of a controlled hybrid dynamical system $CHDS$, $Hc = \{Q, \Sigma, A, G, V, C, F\}$,where V contains the discrete controls, C is the collection of controlled jump sets and F the collection of controlled jump destination maps. Under some conditions this is a hybrid dynamical system HDS, $H = \{Q, \Sigma, A, G\}$ where Q is the collection of discrete states, Σ the continuous dynamics, A the autonomous jump sets and G the autonomous jump transition map.

1.2.2 Analysis of hybrid systems

Once a system is described by a mathematical model, analysis of the model can give answers to various questions. One of the most profound questions is: given initial conditions; does there exist a solution and is this solution unique? The existence and uniqueness of solutions is usually referred to as well posedness.

Well posedness:

Little work has been done to give answers for hybrid systems. Results are given in the PhD thesis of Heemels (39), who studied linear complementarity systems and gave conditions for the existence and uniqueness of solutions. A study of existence and uniqueness of solutions of hybrid automata is given by Lygeros (53). Also, Bemporad and Morari state conditions under which the solutions for their MLD framework are well posed (18). Closely related with well posedness is a topic called verification.

Verification:

The term verification or formal verification stems from the computing science community. Their interest was usually to show the correctness of their programs, which is certainly not a trivial matter for most software packages. Some general conditions can be given for finite automata, while in comparison with dynamic systems the class is too broad to derive general conditions. This is mainly because the number of states is finite. Therefore the correctness can be verified by checking each state. In addition methods like theorem proving can be applied. This is usually carried out by applying a set of rules. In this way finite automata can be verified. However in the case of hybrid automata, the verification is much harder, since the continuous evolution of the state has to be considered in order to take discrete transitions. Therefore the verification problem is usually formulated into a reachability problem (50). That is, given initial conditions does there exist a path leading to the desirable final (terminal) states? As one can imagine, it is much harder to derive conditions, therefore it is no wonder that only general statements can be made. Some conditions are given for timed automata (13), such as the reachability problem is decidable.

However for more complex hybrid systems there are no useful conditions, so one has to rely on simulation. In many cases one starts with the terminal state and integrates backwards in time, taking discontinuous transitions and obtaining in this way the set of initial conditions from which the terminal state can be reached.

Besides properties like well posedness and verification, stability of the hybrid system is one of the most fundamental properties that needs to be assessed. That is, are there stable equilibria or not?

Stability analysis

The majority of analysis publications deal with stability analysis. Before one carries out any stability analysis it has to be defined what one means by stability. For continuous systems we are already aware that there are different notions of stability: asymptotic stability, bounded input bounded output stability, quadratic (Lyapunov) stability, etc.. Since hybrid systems consist of continuous as well as discrete dynamics, the notion of stability should contain both dynamics analyzed. However, quite often only the continuous dynamic is analyzed, which in some cases might cause a problem, since the stability of the continuous dynamic does not imply stability of the discrete dynamic and vice versa. A hybrid system can have stable continuous dynamic and exhibit sliding modes or Zeno executions, which are both unstable discrete dynamics. The other way around, it is also quite obvious that stability of the discrete part does not imply stability of the continuous part. A simple example is a hybrid system consisting of only one unstable continuous dynamic. Hence the discrete dynamic is stable, since it is invariant, but the continuous dynamic, i.e. the continuous states will grow beyond all bounds.

A further point which makes the stability analysis of hybrid systems complicated is that stability of all individual subsystems does not imply stability of the hybrid systems. This can be shown by this simple example (21), (67), given the two stable subsystems

$$A_1 = \begin{bmatrix} 0 & 1 \\ -0.01 & -0.2 \end{bmatrix} A_2 = \begin{bmatrix} 0 & 1 \\ -0.2 & -0.01 \end{bmatrix} \tag{1.5}$$

in controller canonical form with eigenvalues in the open left hand side of the complex plane. Thus, both subsystems are individually asymptotically stable. For the hybrid system, dynamic 1 is valid in the first quadrant of the state space and dynamic 2 is valid elsewhere. Simulating the system reveals that it is unstable as depicted in the figure below. The result

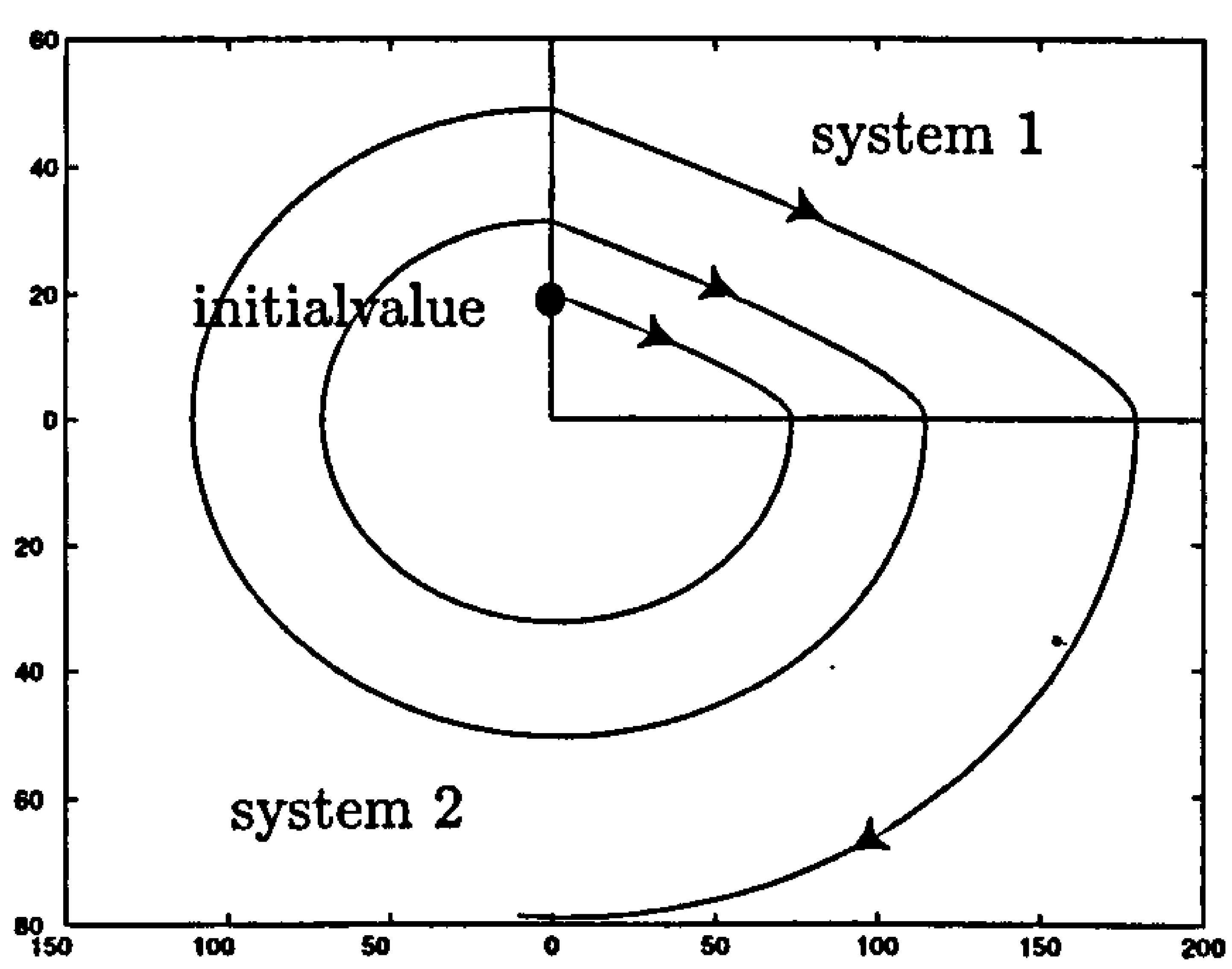


Figure 1.1: Trajectory of hybrid system

is that the trajectory tends to infinity. Hence the system is unstable.

To show stability of the hybrid system, researchers analyzed different classes of hybrid systems and used and developed different methods.

Various results are given for hybrid systems which admit arbitrary switching among stable linear dynamics. One result is that given a set of linear and stable system matrices A_q, $q \in Q$, the hybrid system is stable if the system matrices commute pairwise, i.e. $A_pA_q = A_qA_p$, $\forall q, p \in Q$. This is easy to verify; take $Q = \{1, 2\}$ we can write the solution $e^{A_1t_1}e^{A_2t_2}e^{A_2t_2}\dots e^{A_2t_2}x(0) = e^{A_1(t_1+t_3\cdots t_{k-1})}e^{A_2(t_2+t_3\cdots t_k)}$ thus $x(0) \to 0$ as $t_1+t_2\cdots t_k \to \infty$.

A different condition was given by (71), (70) for a pair of second order asymptotically stable linear systems. Given two matrices A_p, A_q, then every switching sequence is stable, if and only if the matrix pencils $\gamma_\alpha(A_q, A_p)$, $\gamma_\alpha(A_q, A_p^-1)$ are stable. In an extension of Shorten's

work it is shown that his method is less conservative than finding a common quadratic Lyapunov function (72).
However, the most commonly used approach is to construct a Lyapunov function for the hybrid system. If a Lyapunov function can be constructed or if its existence can be proven it shows that the continuous dynamic of the hybrid system is stable. Note that nothing is said about the discrete dynamic. For many classes of hybrid systems, like piecewise affine systems and MLD systems, quadratic Lyapunov functions

$$V(x) = x^T P x \quad P = P^T > 0 \quad A_q^T P + P A_q < 0 \tag{1.6}$$

are obtained as solutions of convex optimization problems. The search of a Lyapunov function is formulated into a linear matrix inequality (LMI) which can be conveniently solved. Note that there are also other methods to compute common quadratic Lyapunov functions as shown by (58). A useful extension to quadratic Lyapunov functions was presented by Johansson (47), (48) and Pettersson (59) and others (21). Two changes can be made to reduce the conservativeness of a quadratic Lyapunov function for the hybrid system. The first relaxation was to introduce the S procedure. Whenever the discrete state q of the hybrid system is a function of the continuous state x, such that one subsystem is valid only in a domain X_q, the Lyapunov inequality can be written as

$$V(x) = x^T P x \quad P = P^T > 0 \quad A_q^T P + P A_q + S_q < 0 \tag{1.7}$$

such that $x^T S x > 0$ when $x \in X_q$ and $x^T S x < 0$ elsewhere. This makes it in general easier to satisfy (1.7). The second relaxation is, in addition, to use piecewise quadratic Lyapunov functions. This Lyapunov function might be discontinuous but decreasing $x^T P_q x \geq x^T P_{q^+} x$ at switching times $f_{q,q^+}^T x = 0$,

$$V(x) = x^T P_q x \quad P_q = P_q^T > 0 \quad A_q^T P_q + P_q A_q + S_q < 0$$

$$P_q - P_{q^+} + f_{q,q^+} t_{q,q^+}^T + t_{q,q^+} f_{q,q^+}^T > 0$$

where q is the predecessor of q^+ and f_{q,q^+}^T describes the switching surface between system q and q^+.
Besides the use of Lyapunov functions, various other methods exist. For an overview, one might read the lecture notes by Liberzon and Morse (51).

Zeno executions

A phenomenon which can be only observed in hybrid systems are the so called Zeno executions. The observed phenomenon is similar to the paradox of the Greek philosopher Zeno (545 BC), who stated the famous paradox with Achilles and the tortoise. In this paradox Achilles, who is a fast runner, wants to catch a tortoise, which is one metre in front of him. Achilles runs twice as fast as the tortoise. Zeno argues that Achilles cannot catch the tortoise. He says every time Achilles reaches the spot where the tortoise was, the tortoise has moved by $\frac{1}{2}$ of the distance Achilles has moved. It seems that Achilles cannot catch the tortoise. Indeed, whenever Achilles reaches the spot where the tortoise was, it has already moved. However, both the distance that Achilles travels and the time that elapses before he reaches the tortoise can be expressed as an infinite geometric series. So, Achilles traverses an infinite number of "distance intervals" before catching the tortoise, but because the "distance intervals" are decreasing geometrically, the total distance that he traverses before catching the tortoise is not infinite. Similarly, it takes an infinite number of time intervals for Achilles to catch the tortoise, but the sum of these time intervals is a finite amount of time.

In hybrid systems, similarly as in the paradox of Zeno, infinite discrete transitions (executions) can occur in finite time. This is undesirable since such models are difficult to simulate. In some cases Zenoness seems to be avoidable when choosing a different model. Take for instance the paradox and write $x = 2 \cdot t$ for the distance covered by Achilles and $x = 1t + 1$ for the tortoise. Then the solution is quite obvious, $t = 1$. A popular example of Zeno executions is the bouncing ball. (46) . The bouncing ball is described in terms of a hybrid

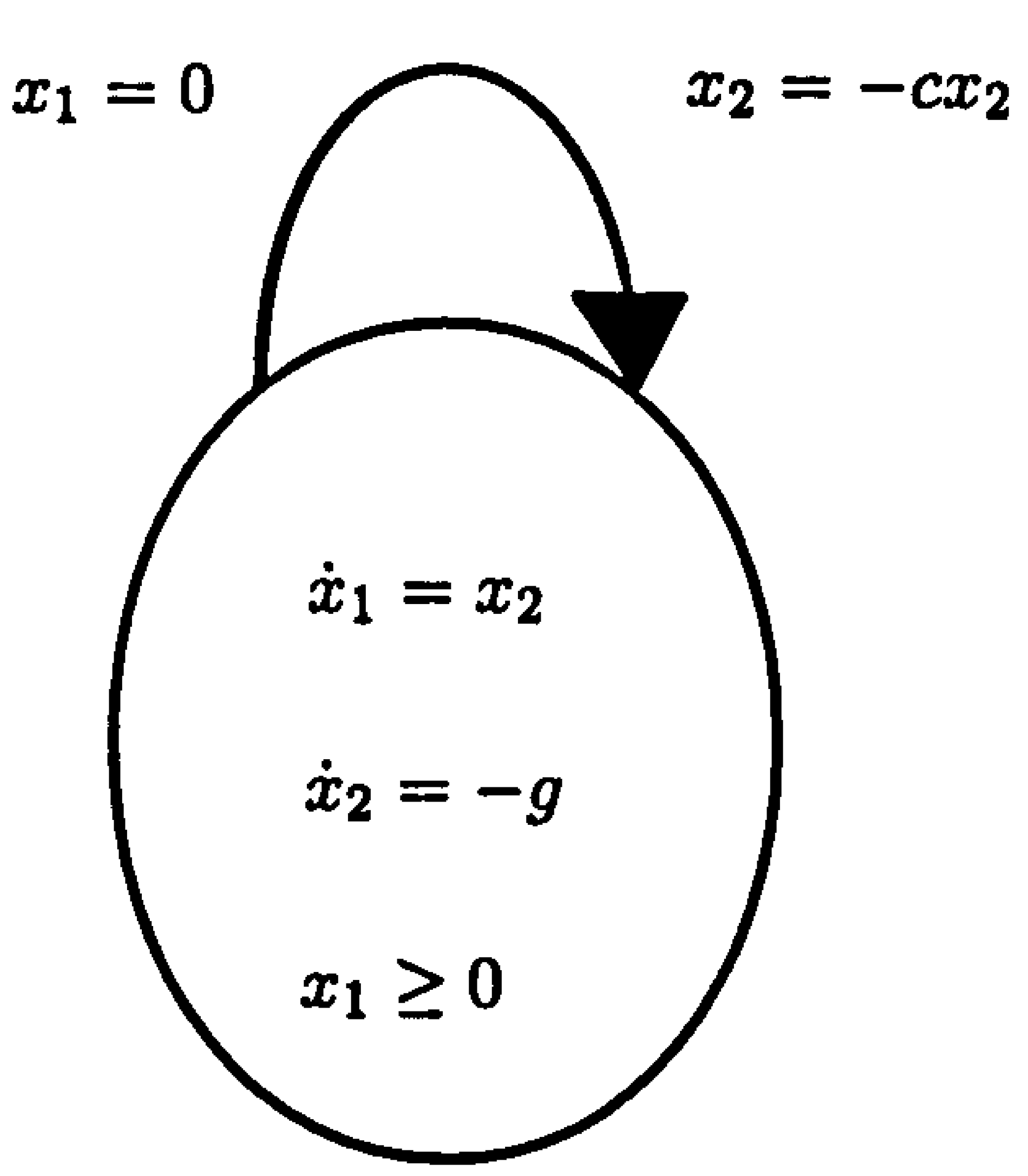


Figure 1.2: Bouncing ball example for Zeno executions

automaton. Inside the bubble (vertex) a second order system is described in form of a differential inclusion. The transition is taken when the ball touches the ground. In case of a transition the state x_2 is set to a value $x_2 = -cx_2$, where $c = (0, 1)$. This is to model the

energy dissipation of the system at the moment when the ball touches the ground.
This phenomenon and especially its detection has been discussed by only a small group of researchers (45). In most publications that deal with hybrid systems this phenomenon is simply neglected.

Controllability

The analysis of controllability for hybrid systems is usually handled in the context of reachability analysis. The hybrid system is controllable with respect to certain terminal states if they can be reached. The proposed methods are of a numerical nature. Bemporad, Ferrari-Trecate and Morari (16) propose tests based on mixed-integer linear programming to show controllability.

Observability

The observability analysis (29) is much simpler since usually one is interested in observing the continuous state. Hence it is sufficient to show that each subsystem is observable. Many of the frameworks admit modelling only linear dynamics in each subsystem. Hence the observability tests derived for linear systems theory apply.
The observer design can be seen as the dual to the state feedback controller design. Therefore the synthesis methods used for controller design can be translated to the observer design. One idea is to find a common quadratic Lyapunov function for all dynamics such that the estimation error converges quadratically. Sometimes a single observer gain can satisfy this requirement. If not, multiple gains are used; in this case the observer gain depends on the discrete state which has to be measured.
Similarly the idea of "Model Predictive Control" (MPC) for hybrid systems can be stated as a dual for observers. Ferrari-Trecate and Mignone (30) propose a state smoothing algorithm for hybrid systems based on Moving Horizon Estimation (MHE). Arguments are made for piecewise affine systems, where sufficient conditions on the time horizon and the penalties on the state at the beginning of the estimation horizon are given to guarantee asymptotic convergence. The MHE is then implemented by solving a Mixed-Integer Quadratic Program.

1.2.3 Control of hybrid systems and hybrid control

Hybrid control does not always involve hybrid dynamics which have to be controlled. In many cases continuous dynamics are considered, which are controlled by a hybrid controller.

There are many reasons for this. One is that the dynamic which is to be controlled might be nonlinear and contains uncertainties. For such systems sliding mode control is very often used. Another reason to use hybrid controllers is that the plant cannot be stabilized by a continuous control law. One such example is the nonholonomic integrator. The term nonholonomic is used for systems which are linear (separately) in the states and in control variables.

$$\begin{aligned} \dot{x}_1 &= u_1 \\ \dot{x}_2 &= u_2 \qquad\qquad \| (u_1, u_2) \| \leq 1 \\ \dot{x}_3 &= x_1 u_2 - x_2 u_1 \end{aligned}$$

The control is closed in a unit ball in $\mathbf{R}^2$. It was shown that the system is globally asymptotically controllable. However, no continuous feedback law $u = k(x)$ exists to stabilize the system (even locally around the origin, as shown by Brockett in 1983). Brockett (24) also gave a condition for dynamic systems $\dot{x} = f(x, u)$ which admit a continuous stabilizing feedback. For every neighbourhood Ω of 0, the set $f(\Omega, U)$ is also a neighbourhood of 0. It is easy to show that no continuous controller exists to stabilize the nonholonomic integrator by using Brockett's condition. For Ω $(0, 0, \epsilon)$ it is easy to see that there is no $\epsilon \neq 0$. Hence Brockett's condition rules out the existence of a continuous feedback.
A further reason to apply hybrid control strategies to continuous systems is that it might be desirable to pursue different objectives during the operation of the system. In such cases heterogenous hybrid controllers are applied.
In some cases where hybrid systems are controlled it does not mean that hybrid control strategies are involved. Some hybrid dynamics can be controlled with purely continuous or discrete control laws. Even then hybrid system theory is needed, since the controlled closed loop system will still be of a hybrid nature.

Sliding mode

Sliding mode control is probably one of the oldest hybrid control strategies. Differential equations with discontinuous right hand sides (hybrid systems) have been studied for a long time. Researchers like Filippov and Utkin developed solution concepts for piecewise continuous dynamics. Their research work was more focused on analysis. However, it was straightforward to use this solution concept for controller synthesis. These days sliding mode control is taught in undergraduate courses where books such as the one by Slotine and Lee (73) are used. The notion of a sliding mode is quite simple. On the boundary of at least two different dynamics, the vector fields point towards the boundary. In this way the trajectory cannot leave the boundary, since whenever it might leave the boundary it is pushed back. The principle of the sliding mode controller is now to define a sliding surface, which is the boundary of two dynamics with vector fields pointing towards the boundary, for the dynamic we want to control. In this way each trajectory for every possible initial condition will go to the sliding surface since the vector fields point towards the sliding surface. Once the sliding surface is reached the trajectory will slide along the surface. Here the solution concept of Filippov is used and the sliding dynamic is described by the differential inclusion of the dynamics which are valid on the neighbour domains. The sliding mode control is often implemented as a state feedback controller with two different state feedback gains. Dependent on the domain, the current state feedback gain is valid.

State feedback control

A lot of work has been done in this area. However, most of the work is only applicable to hybrid systems which admit piecewise linear or piecewise affine dynamics (37). The synthesis is usually based on finding a common quadratic Lyapunov function, i.e. $V(x) = x^T P x$. The synthesis procedure is then cast into a Linear Matrix Inequality (LMI)(20) (62), which are conveniently solved by commonly available tool boxes. Using the Lyapunov inequality

$$(A_q - B_q K_q)^T P + P(A_q - B_q K_q) < 0 \quad \forall q$$

two things have to be found simultaneously. One is the common performance P and K_q which satisfies this inequality.

The solution of an optimal control problem, finding a control input which minimizes a given objective function, is also usually given as a function of the states and results therefore in state feedback.

Optimal control

Optimal control for hybrid systems is closely related to optimal control of continuous or discrete time systems. Sometimes, the popular quadratic objective function used in linear systems is taken (60), (23). Usually small modifications to the standard quadratic objective function are made. In many cases it makes sense to penalize a change of the discrete state, therefore a penalty for mode changes is added. This prevents the system from taking Zeno executions, since infinite mode changes in finite time would mean infinite cost.

In some cases it is suggested to use the suboptimal solutions (43), (44) for the actual implementation, since the optimal solution is hard to obtain. A different approach is to compute upper and lower bounds on the optimal cost (38). By optimizing both, the gap can be reduced and possibly closed such that the optimal input (cost) is found.

Model predictive control

Model predictive control for hybrid systems is quite similar to that of continuous or discrete systems. A performance index is minimized over a finite horizon, subject to the dynamics and constraints of the hybrid model. The performance index is usually 2-norm (quadratic) or infinity norm or mixtures of both. Only the first step of the optimal sequence is applied at time t. At time $t+1$ a new sequence is evaluated to replace the previous one. This online re-planning provides the desired feedback control. The solution of this optimization process is found by solving the mixed integer linear program MILP, for linear performance index, or a mixed integer quadratic program MIQP for quadratic performance. Results have been reported by Bemporad and Morari which use MPC for their MLD systems.

For finite time optimal control of linear hybrid systems with constraints, the optimal control law can be stated as a piecewise linear state feedback controller. Bemporad and Borrelli (15) (14) have shown some results for the MLD systems. The advantage of this approach is that the control law can be stated explicitly as a function of the state space. In this way the state feedback gains can be computed off-line. This allows application of MPC to systems with fast dynamics. An example is the slip control of a car by Borrelli (19).

Scheduling stable switching strategies

This area is concerned with strategies showing how to switch from one mode (subsystem) to another in order to stabilize the system or maintain stability. Malmborg (57) considers the scheduler design for a finite number of linear stable subsystems. It is then easy to compute a Lyapunov function for each subsystem. The scheduler chooses the current system depending on which Lyapunov function achieves the lowest value. In this way it can be guaranteed that the continuous dynamics of the hybrid system are stable.

Another approach is to restrict the time between consecutive switches. A minimum time, referred to as the dwell time, is computed for which a stable subsystem has to be active. Note that this prevents the system from Zeno executions. Michel (84) also develops various switching laws. He shows that for hybrid systems consisting of stable and unstable subsystems, stability can be guaranteed if stable subsystems are active for a minimum time, and unstable subsystems are active for a maximum time.

A more difficult problem is addressed in the scheduler design where the hybrid system consists only of unstable subsystems (83). Wicks, Peletis and DeCarlo (82) considered the problem of two unstable linear subsystems. The only assumption made, but an important one, is that the matrix pencil, $\gamma_\alpha(A_1, A_2)$, contains a stable matrix. Let $\alpha_1 \in (0,1)$ be the value which renders this convex combination $A = \alpha_1 A_1 + (1-\alpha_1)A_2$ to be stable. Then there exists a Lyapunov function for the switched system.

$$\alpha(A_1^T P + PA_1) + (1-\alpha)(A_2^T P + PA_2) < 0 \tag{1.8}$$

This implies that for a nonzero $x \in \mathbf{R}^n$ either 1: $x^T(A_1^T P + PA_1)x < 0$ or 2: $x^T(A_2^T P + PA_2)x < 0$. Hence the switching strategy is to select the subsystem 1 or 2 dependent on which inequality, i.e. 1,2, is negative.

Simultaneous stabilization problem

The simultaneous stabilization problem and strong simultaneous stabilization problem (81) is the problem of stabilizing a number of dynamics with one and the same controller. The requirement of strong simultaneous stabilization (SSSP) is the same but in addition the controller needs to be stable as well. Note that this approach only stabilizes each dynamic individually. It is not guaranteed that changing between the specified dynamics will lead to a stable dynamic. However, with additional arguments like dwell times (see above) stability under switching can also be guaranteed. Results have been presented for state feedback and output feedback (79), (80). The approach is promising for changes that occur only once. One application is safe control in the presence of folds. Possible fold scenarios are modelled and each fold dynamic can be stabilized by the same controller. In this way safety critical systems can be operated in emergency conditions.

Heterogeneous Hybrid Control

One of the most recent developments in hybrid systems is in the area of heterogeneous hybrid control. The word heterogeneous is used to indicate that the control structure changes. With this type of hybrid controller not only the parameters of the controller will change dependent on state, time or input, but also the structure of the controller might change. It is easy to see that such controllers make sense since we might change the objective depending on the situation. Under some operation conditions it might be desirable to be stable and robust no matter how conservative the controller is. In other cases where we do not have to take care of resources, we are only interested in driving the system at maximum throttle. For such cases not only the parameters or weights will change, the whole controller structure will change. One example from the process industry is the use of a time optimal controller together with a PID controller (57). The time optimal controller is used to change from one set point to a different set point as quickly as possible. The objective is to reach the new desired operation condition in minimum time. During the transition from one operating condition to the next we are not concerned with stabilizing the process. As soon as we reach the new operation condition the objective changes. We want to stay in this operating point. In order to achieve the new objective we will switch to the PID controller, which stabilizes the process at this operation point. This example clearly demonstrates that for some processes the objective changes. The controllers which render such requirements have different structures as well as parameters, i.e. hybrid heterogeneous controllers.

Controller synthesis as Game

The idea of this approach is to treat the controller synthesis as a game between the controller and the disturbance (54). In this approach the controller tries to prevent the trajectory from leaving a desired region of the hybrid state space, sometimes called good states, under all possible interferences from the disturbance (55). In this context the notion of controlled invariance is derived. A subset of the hybrid state space W is called controlled invariant if for all states in W there exists a control such that the trajectory stays in W. To derive the notion of maximal controlled invariance the operator controllable predecessor is derived. The controllable predecessor, $Pre_u(K)$, are the states that can be forced to jump into K by some control u. $Pre_d(K)$ are the states that may jump out of K for some disturbance d, while $Reach(Q,O)$ are the states which can be continuously driven into Q avoiding O. With these operators it is then possible by iterative exclusion to find the set of states which can never be left for any action of the disturbance. This set is then referred to as the maximal controlled invariant set.

Applications to aircraft collision avoidance are reported in (61). This is the problem of computing the distance where the aircrafts should change flight mode and fly a circle to avoid collision, given a set of disturbances.

1.2.4 Identification of hybrid systems

Identification of hybrid systems is an emerging topic. Only a few results have been published so far; some pioneer work can be found by Ferrari-Trecate (31), (34), (33). The main achievements so far are in the class of piecewise affine systems. The problem of identifying a piecewise affine system is threefold. One task is to group the identification data such that only data belonging to the same affine map is collected. The second task is to identify the parameters of each affine map, and the third task is to identify the domain on which each affine map is valid. This is already a much more complicated problem than identifying the parameters of only one affine map.

1.3 Summary

Over the last 2 decades, much research work in the area of hybrid systems has been carried out. It has been shown that there are many dynamical systems which incorporate continuous as well as discrete dynamics. Various modelling frameworks have been developed. Some frameworks are able to model various classes of hybrid systems, while limiting the possibility of exploiting structure to derive general conditions for analysis or controller design. Other frameworks have limited applicability but facilitate a lot of structure which makes it easier to derive conditions for stability, reachability etc.. It is fair to say that most hybrid systems are well modelled in the current frameworks. However, not much work has been done modelling hybrid systems incorporating uncertainties or parameter variations.

In terms of analysis most areas have been covered. However, the main focal point has been hybrid systems which consist of linear subsystems. A lot of work has still to be done for hybrid systems incorporating nonlinear dynamics. Only a few attempts have been made to analyze robustness of hybrid systems, with respect to variations in switching or parameter variations.

Control of hybrid systems, and especially hybrid control, has boosted a lot of research. Limitations of continuous feedback were shown nearly 2 decades ago when Brockett showed that for systems like the nonholonomic integrator there does not exist a continuous feedback law which stabilizes the system. This certainly triggered a lot of research work investigating hybrid control strategies. Also, the control of hybrid systems is a quite mature area ranging from sliding mode control to model predictive control of hybrid systems. But again, robust control seems to be neglected and is far away from the maturity that it has for continuous systems. There also seems to be a lack of robust scheduling strategies. The observer synthesis is rarely discussed, which leaves more room for research. Separation principles, between controller and observer, have to be shown for many classes of hybrid systems.

Identifying hybrid systems is definitely an emerging area. A fundamental procedure for identification of hybrid systems, particularly linear hybrid systems, has been established, however many details have to be worked out.

We have seen that the theory of hybrid systems is quite mature with respect to modelling frameworks. However, in terms of analysis, controller and observer design as well as identification, there are still some major gaps to be closed. Robustness issues especially have to be examined, since the control of physical systems usually makes handling of parameter uncertainties and variations inevitable. This thesis tries to develop new methodologies to analyze robustness of hybrid systems with respect to parameter variations. In addition robust controller synthesis for hybrid systems will be investigated.

Many controllers use state feedback which requires knowledge of the current state. In some cases the states are online measurable. However, if that is not possible an observer has to be incorporated, which provides the controller with an estimate of the states. There is a lack of observers, which do not only make the estimation error converge, but are also optimal with respect to minimizing the covariance of the estimation error. It would be nice to have such a dual to the Kalman filter for hybrid systems also. In this thesis we will propose such an optimal observer for hybrid systems to overcome this gap in the theory.

Many contributions need to be made in the field of identifying hybrid systems. In particular theory needs to be developed which supports the proposed methodology. It is also unclear if there are better ways to identify hybrid systems like the currently proposed ones.

The thesis tries to close some of the gaps in hybrid control theory. Especially we are concerned with a usually neglected class of hybrid system, the nondeterministic hybrid system in piecewise linear form. This class of hybrid system consists of linear or affine subsystems and has unknown or nondeterministic discrete dynamic. In this thesis we will be concerned with robust control and optimal control of this class of hybrid system. Since these controllers apply state feedback an optimal observer design is proposed for the case that not all continuous states are measurable. An outline of the work and contributions of the thesis is given in the next chapter.

2 Outline of the thesis:

The thesis is concerned with hybrid systems in piecewise linear or affine form, where the discrete dynamic is unknown. It is shown how such a class of hybrid system can be obtained from time variant nonlinear systems. In an introduction example the performance advances of hybrid feedback over purely continuous feedback is shown.
In order to assess robustness of such hybrid systems an analysis framework is developed. A robust controller design is proposed which can cope with parameter variations.
The thesis shows that optimal control for this class of hybrid system is not straightforward. Since there is no influence on the discrete dynamic, only upper and lower bounds on the optimal cost can be computed. The upper bound is used to design an optimal control law which limits the cost by the upper bound and guarantees stability. Since the robust control as well as the optimal control depends on state feedback, an observer design for this class of hybrid system is proposed.
For analysis and controller design models are needed. In the event that the equations to model the hybrid system are not known they need to be identified. The identification steps are shown together with sufficient conditions under which these results can be obtained.

2.1 Overview of thesis contents

The thesis is structured in 6 main sections, containing in total, 9 chapters.

Section 1: Introduction
The first section consists of chapters 1-3 and contains the introduction to hybrid systems including a survey of previous research work. An example of hybrid control is given. The example shows how a linear hybrid system is obtained form a nonlinear time variant system. Analysis shows that no continuous feedback exists which renders the controlled system to have the desired performance. Consequently a hybrid controller is designed, which stabilizes the dynamics by using a discontinuous feedback. The merits of this discontinuous feedback

are displayed in an illustrative example. After this example the second main section follows.

Section 2: Controller design for hybrid systems

This section is concerned with the control of hybrid systems in piecewise linear (affine) form with unknown discrete dynamic. The section contains 3 chapters (chapters 4-6). The first chapter is concerned with robust control of this class of hybrid system. Robustness analysis and robust control are important since almost all models contain inaccuracies, like parameter variations. Therefore it is important to have robustness analysis and robust controllers for hybrid systems too.

Chapter 4: Robust control of hybrid systems

This chapter introduces a method for controller design of uncertain and parameter-variant linear hybrid systems. The idea is to specify a desired performance, which is represented by a nominal system. Around this nominal system a compact set of systems is obtained which will be robustly stable against switching among members of this set: such a set of systems is then called the stable switched set. It is shown that obtaining the stable switched set is a signomial program. Upper bounds on signomial programs can be easily obtained, which are used to compute the stable switched set. A sufficient condition is given for the existence of a common state feedback controller that stabilizes an uncertain and parameter-variant linear hybrid system on a stable switched set. Further, a synthesis procedure is proposed in terms of a constrained convex optimization problem that places the uncertain and parameter-variant linear subsystems optimally close to the desired nominal system, using one common state feedback controller. An extension is shown for the case that no common state feedback controller exists. The synthesis framework is then applied to a simple example to demonstrate the procedure. It is shown that real systems, like the control of active suspension, transform naturally into uncertain and parameter-variant linear hybrid systems.

Besides robust control, optimal control for hybrid systems is of broad interest. Chapter 5 shows that systems with variations in sampling rate, where the variation is decided by a scheduler, are an example of linear hybrid systems in discrete time. Using this particular set-up it is shown how optimal controllers can be designed.

Chapter 5: Optimal control of hybrid systems in discrete time

This chapter addresses a class of hybrid systems with linear dynamic as they occur in real-time systems with varying sampling rate. In such a set-up the subsystems are obtained by sampling a continuous system at different sampling rates. The switching between the sampling rates is decided by a scheduler. Fast sampling is chosen if enough computational resources are available, and slow sampling otherwise. In order to motivate, an example is given in which a stable continuous system is sampled at two different sampling rates. Two controllers are designed minimizing the same continuous quadratic loss function with the same weights. It is shown that although the design leads to stable controlled closed loop systems, for both discretizations, the resulting system can be unstable due to variations in sampling rate. To avoid that problem two solutions are suggested. The first solution shows how restrictions on variations in sampling rate can be imposed such that only stable sequences are chosen. The second solution presents an optimal controller design in which a bound on the cost, for all possible sampling rate variations, is computed. This results in a piecewise constant state feedback control law and is robustly stable for all variations in sampling rate. The controller synthesis is cast into an LMI, which conveniently solves the synthesis problem. To illustrate the procedure, the introduction example is repeated with the proposed LMI synthesis method and the control law is given, which is robustly stable against variations in sampling rate.

In the following chapter optimal control for linear hybrid systems with unknown discrete dynamic is proposed. This chapter is similar to the previous chapter. The main difference is that derivations are made for hybrid systems in continuous time.

Chapter 6: Optimal control of linear hybrid systems in continuous time

This chapter is concerned with optimal control of linear hybrid systems in continuous time. As in the previous chapters there is no control about the discrete dynamic, i.e. whichever current subsystem is active cannot be influenced by the control action. Further, the discrete dynamic is unknown or nondeterministic but changes can be detected immediately. For this class of system a control input is sought which minimizes the standard quadratic performance index. Since the discrete dynamic is arbitrary, only bounds on the optimal cost can be derived. For the worst case switching sequence an upper bound on the cost can be derived, while a lower bound is found using the best case switching strategy. In order to obtain the lower and upper bound, an LMI is derived which gives the solution by solving a convex optimization problem. It is shown that together with the upper bound a control law is found which is robustly stable for all possible switching sequences, while limiting the cost.

The section controller design for hybrid systems ends with chapter 6. All controllers proposed in this section are for the class of linear hybrid systems with unknown discrete dynamic, and rely on state feedback. In various cases it is not possible to measure all states. Therefore observers are needed, which estimate the current continuous states. Unfortunately there are not many results for hybrid observers in the current literature. The next section tries to overcome this deficit and proposes an optimal observer design for linear hybrid systems.

Section 3: Observer design for hybrid systems

In the previous section controller synthesis methods have been described, which stabilize the hybrid system robustly or where the control input minimizes a quadratic performance index. All those controllers have in common that the control input is given as a function of the states. Some of the controllers depend only on the continuous states while others also depend on the discrete state. For this reason the states need to be available. In some cases the states can be measured as shown in the ABS examples. However, there are various situations where it is impossible to measure the states. In these cases, observers have to be built which give an estimate of the current state. Due to the separation principle of observer and controller, as shown in (56) (29), it can be guaranteed that both converge.

Chapter 7: Observer design for hybrid systems

In this chapter we are concerned with building such an observer. It is first shown that it is not straightforward to design such an observer, and examples demonstrate that the asymptotic Kalman filters can fail. Our design overcomes this problem. It converges robustly under all admissible mode changes. The proposed observer is given by a piecewise linear observer gain. It is further optimal in the sense that it minimizes the covariance of the estimation error. This is similar to the Kalman filter for discrete or continuous time systems. Further, we show that the synthesis of the observer can be cast into an LMI, which conveniently solves the synthesis problem. To demonstrate the synthesis procedure, an example is given.

Analysis and controller design depends on models. These models describe the system's behaviour. In some cases it is possible to obtain such a model from equations, i.e. laws of physics. If this is not possible, models are identified from input-output data. This is usually done by solving an optimization problem, which finds the optimal mathematical model that describes the input-output data best.

Unfortunately there are not many results for identification of hybrid systems. Even for linear hybrid systems there is hardly any work. The next section will address some fundamental questions for identifying linear hybrid systems.

Section 4: Identification of hybrid systems

In this section the identification of hybrid systems in piecewise affine form is discussed. The problem of identifying such systems is threefold: the classification problem, which input-output data belongs to which affine dynamic. The regression problem is to identify the parameters of each affine dynamic and the domain reconstruction problem, reconstructing the area where each dynamic is valid.

Chapter 8: Identification of hybrid systems in discrete time

The problem is solved in a multistage optimization problem. In the first stage the input-output data pairs are collected. In this way local data sets (LD) are generated containing the c nearest neighbors. A model for each LD is fitted. This is done by the least squares method. The model is described by the local parameter vectors (LPV). Dependent on the LD, we get pure LPV if the LD contains data only from the same dynamic. If the LDs contain data from different dynamics we obtain outliers. In the next stage the LPVs which describe the same model are clustered. Clusters which contain LPVs from only one dynamic are called perfect clusters, and clusters which also collect outliers are called mixed clusters. For each cluster one parameter vector is found. The clustering and the identification of the single parameter vector are done in a single optimization procedure.
Conditions are given under which the optimization procedure produces perfect clusters. It is shown that this elementary problem is not trivial. Based on this result methods are proposed to detect outliers. The procedure is explained in an example.

Section 5: Conclusions

A summary of the thesis is given here. Future research directions and open problems are discussed.

2.2 Thesis contributions

The thesis is concerned with analysis, controller and observer design as well as with the identification of hybrid systems in piecewise linear/affine form. The contributions of the thesis are fivefold:

- It was shown how nonlinear systems can be transformed into a class of state dependent uncertain and time-variant piecewise linear systems (chapter 4). For stability analysis a framework was proposed for this class of hybrid system. For fast computation of the stability radius analytical expressions of the upper bound on signomial programmes were given. This work has been published in (67),(68).

- It was shown that sampled data systems with varying sampling time can be viewed as linear hybrid systems (chapter 5). A novel example was given where a continuous system was sampled at two different sampling rates. A controller was obtained minimizing the same continuous loss function for both sampling rates, which led to a piecewise constant state feedback. Although it was assumed widely that such a control law would guarantee stability, it was found that this is not necessarily true. This work has been published in (66), (64).

- For the class of hybrid systems consisting of switched linear dynamics, where the switching logic is unknown but the discrete state is measurable or observable, an optimal controller synthesis was proposed (chapter 5). It was shown that the proposed controller is robustly stable against all possible switching sequences.
  To obtain a performance bound and state feedback gains a Linear Matrix Inequality (LMI) was derived (chapter 5-6). This work has been published in (64).

- The duality between controller design and observer was exploited to design an observer for linear hybrid systems (chapter 7). An observer design was proposed similar to the observer design for purely continuous or purely discrete systems as stated by Kalman. The design does not only guarantee that the estimation error converges quadratically, it also minimizes the covariance of the estimation error. For observer synthesis an LMI has been derived.

- Open problems in identification of hybrid systems in piecewise affine form have been solved (chapter 8). Sufficient conditions have been derived under which the optimization problem of clustering and identification produces non-mixed clusters. Based on this result, methods for outlier detection have been proposed. This work has been submitted in (35).

2.2.1 Publications

In terms of publications the work in this thesis has led to

- two journal papers (68; 80)
- eight refereed conference papers (67; 66; 81; 63; 79; 65; 64; 41)
- one book chapter (42)

[35] G. Ferrari-Trecate and M. Schinkel. Identification of piecewise affine systems: Sufficient conditions for perfect data clustering. *IEEE Trans. on Automatic Control.* submitted.

[41] W. Holderbaum and M. Schinkel. On stability of nondeterministic hybrid systems. In *Proceedings of Wismarer Automatisierungssymposium*, Wismar, 2002.

[42] K. Hunt, Y. Wang, M. Schinkel, and T. Schmitt-Hartmann. *Controller design for hybrid systems using simultaneous D-stabilisation technique and its application to Anti-lock Braking systems (ABS).* "Nonlinear and Hybrid Control in Automotive Applications". Springer-Verlag, London, 2002.

[63] M. Schinkel. A robust and performance orientated controller design approach for non-deterministic hybrid systems. In *Proceedings of European Control Conference*, Porto, 2001. Student session.

[64] M. Schinkel, W. H. Chen, and A. Rantzer. Optimal control for systems with varying sampling rate. In *Proceedings of American Control Conference*, Anchorage, 2002.

[65] M. Schinkel and K. Hunt. Anti-lock braking control using a sliding mode like approach. In *Proceedings of American Control Conference*, Anchorage, 2002.

[66] M. Schinkel and A. Rantzer. Sample data control with varying sampling time. In *Proceedings of European Control Conference*, Porto, 2001.

[67] M. Schinkel, Y. Wang, and K. Hunt. Stable and robust state feedback design for hybrid systems. In *Proceedings of American Control Conference*, Chicago, 2000.

[68] M. Schinkel, Y. Wang, and K. Hunt. Control of uncertain and parameter-variant piecewise linear systems. *International Journal of Control*, 2002.

[79] Y. Wang, M. Schinkel, and K. Hunt. An engineering approach to simultaneous stabilisation and strong simultaneous stabilisation with d stability. In *Proceedings of Conference on Decision and Control*, Tampa, 2001.

[80] Y. Wang, M. Schinkel, T. Schmitt-Hartmann, and K. Hunt. Pid and pid-like controller design by pole assignment within d-stable regions. *Asian Journal of Control*, 2002.

[81] Y. Wang, T. Schmitt-Hartmann, M. Schinkel, and K. Hunt. A new approach to simultaneaous stabilisation with d-stability and its application to control of antilock braking systems. In *Proceedings of European Control Conference*, Porto, 2001.

3 Control of Anti-Lock Brake with heterogenous hybrid controller

In this chapter we will give an introduction to hybrid control systems. First the modelling aspect is covered. Using an example it is shown how physical laws, which incorporate continuous as well as discrete dynamics, are modelled in a hybrid system framework. Analysis is carried out to show that there exists no continuous feedback law which achieves the desired performance. Motivated by this a hybrid controller is designed.
As an example an Anti-Lock Brake (ABS) controller is derived. The dynamics of a braking car need to be viewed as a hybrid system, since discontinuous changes in road condition make the friction coefficient jump. It is shown that the dynamics of an ABS can be conveniently modelled by a linear uncertain hybrid system. The hybrid system consists of linear uncertain subsystems, which are state dependent, i.e. their validity depends on the continuous state. Analysis is carried out to show the highest possible braking performance. It is shown that a continuous feedback law will not achieve the maximum braking performance. Hence, a heterogenous hybrid control law is derived. The controller has similarity with a sliding mode controller. For large control errors the trajectory is brought back to the desired surface with large gains. Around the desired operating point the dynamic is stabilized with a PI controller. The merits of this controller are shown in an example.
Before starting with the braking dynamics a brief history of the design of ABS is given where the advantages of ABS are explained. This chapter has been published in (65), and serves as an illustrative introductory application of hybrid control, it also gives a novel approach to ABS control.

3.1 History of ABS

The origin of anti-lock brake controllers (ABS) lies in the design of the so called anti-skid braking controller. The first anti-skid braking controllers were designed for trains in 1908.

After Bosch received a patent in 1936 for an electro-hydraulic anti-lock system such systems were build into aircraft in the 1940's before their introduction to passenger cars in 1969 when Ford built such a system into their motor-cars. The implemented system was marketed under the name "Sure-Track" and due to shortcomings in performance and poor reliability as well as high price it was taken off the market again. After Bosch managed to overcome these shortcomings with a more sophisticated electronic controller design in 1978 the ABS was again put into a car, this time it was a 1979 Mercedes-Benz. After 1984 the ABS was also reintroduced on the American market. Today ABS comes as a standard in nearly every new car.

The advantages of an ABS can be clearly seen when comparing the emergency braking situation of cars with and without ABS. In emergency braking situations the driver wants to reduce the speed of the car as fast as possible, therefore the driver presses the brake pedal as hard as possible. In cars without ABS the wheels will lock and the car will start sliding. This has undesirable effects. Since the car is sliding the friction between tyre and road will have decreased. Hence the distance after which the car will come to a standstill will increase. The tyre wear is not equally distributed over the whole tyre, since the wheel is locked and the tyre is sliding on the very same tyre part. Another undesired effect is that as soon as the wheels lock the car becomes unsteerable. This might be quite dangerous in the case when the driver wants to avoid an obstacle during the braking manoeuvre. In a car with ABS sensors monitor the rotation of the wheels and as soon as the wheels are about to lock the brake pressure is reduced. Therefore the ABS prevents the wheels from locking. Since the wheels are still rolling steerability is maintained, and a higher friction between street and tyre is achieved which leads to a shorter braking distance.

In the following a nonlinear longitudinal car model is presented. It is shown that the dynamics can be described by a linear uncertain hybrid system. The analysis will assess maximum braking performance and stability issues. It is shown that a continuous feedback law cannot achieve the maximum braking performance considering the uncertainty with which the friction/slip curves are given. To overcome this problem a heterogenous hybrid controller is suggested.

3.2 Modelling of the longitudinal dynamics

In this section we derive the time-varying nonlinear equations of a quarter car model (81). We simplify and linearize them such that we obtain a suitable hybrid representation to carry out linear analysis and controller design.

For the control design we describe the quarter car model as shown in figure 3.1 The equations of motion for the quarter car are given by Newton's second law and the sum of the momenta

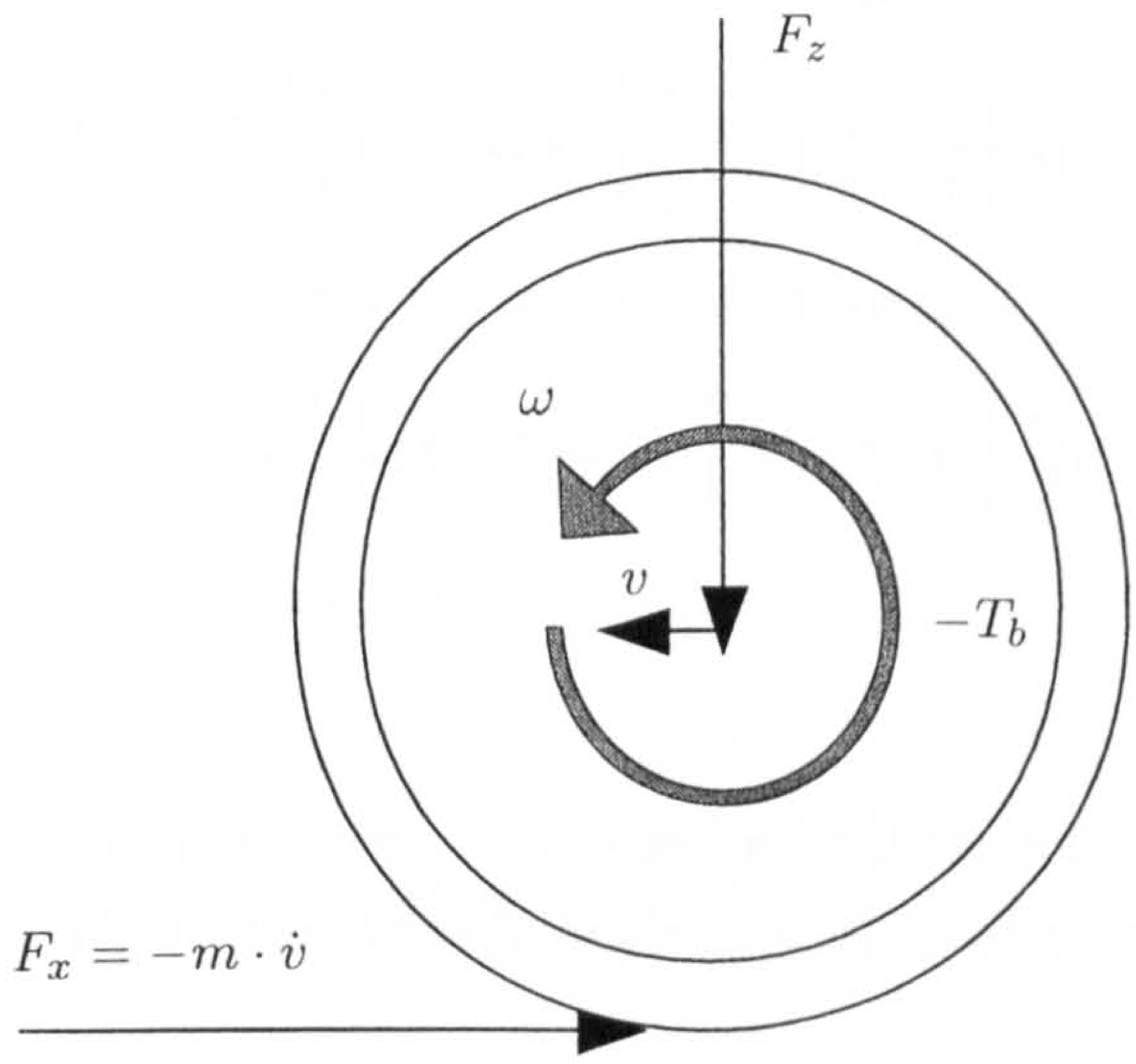


Figure 3.1: Quarter car slip model

at the wheel

$$m\dot{v} = -F_z \tag{3.1}$$

$$J\dot{\omega} = rF_x - T_b sign(\omega) \tag{3.2}$$

where v: vehicle speed, m: vehicle mass, J: wheel inertia, r: wheel radius, λ: tyre slip, μ: friction function between tyre and road, F_x: tyre friction force, F_z: vertical force (dynamic load), T_b: brake torque.

The difference between wheel velocity and car body velocity determines the wheel slip

$$\lambda = \frac{v - \omega r}{v} \tag{3.3}$$

The tyre friction coefficient is given as the ratio of tyre friction force and vertical force

$$\frac{F_x}{F_z} = \mu(\lambda, \mu_H, \alpha, F_z) \tag{3.4}$$

With these equations we can derive the dynamics of the uncontrolled braking quarter car. Using equation (3.1) and substituting for F_z with (3.4) we get

$$\dot{v} = -\frac{1}{m} F_x \mu \tag{3.5}$$

after solving for $\dot{v}$. Solving the slip equation (3.3) for ω and differentiating with respect to time, we get

$$\dot{\omega} = \frac{\dot{v} - \dot{v}\lambda - v\dot{\lambda}}{r}. \tag{3.6}$$

Substituting (3.6) into (3.2) and solving for $\dot{\lambda}$, we get

$$\dot{\lambda} = \frac{\dot{v}(1-\lambda)}{v} - \frac{r^2}{Jv} F_x + \frac{r}{Jv} T_b \tag{3.7}$$

Using the equation for the friction coefficient (3.4) we obtain

$$\dot{\lambda} = -\frac{1}{v}\left[\frac{1}{m}(1-\lambda) + \frac{r^2}{J}\right] F_z\mu(\lambda) + \frac{1}{v}\frac{r}{J}T_b \qquad (3.8)$$

Thus, the time-varying nonlinear equations of a quarter car are:

$$\begin{aligned}\dot{\lambda} &= -\frac{1}{v}\left[\frac{1}{m}(1-\lambda) + \frac{r^2}{J}\right] F_z\mu(\lambda) + \frac{1}{v}\frac{r}{J}T_b \\ \dot{v} &= -\frac{1}{m}F_z\mu(\lambda)\end{aligned}$$

The friction coefficient can vary in a very wide range, depending on factors like a) road surface conditions (dry, wet or icy), b) tyre side slip angle, and c) tyre brand (summer tyre, winter tyre).

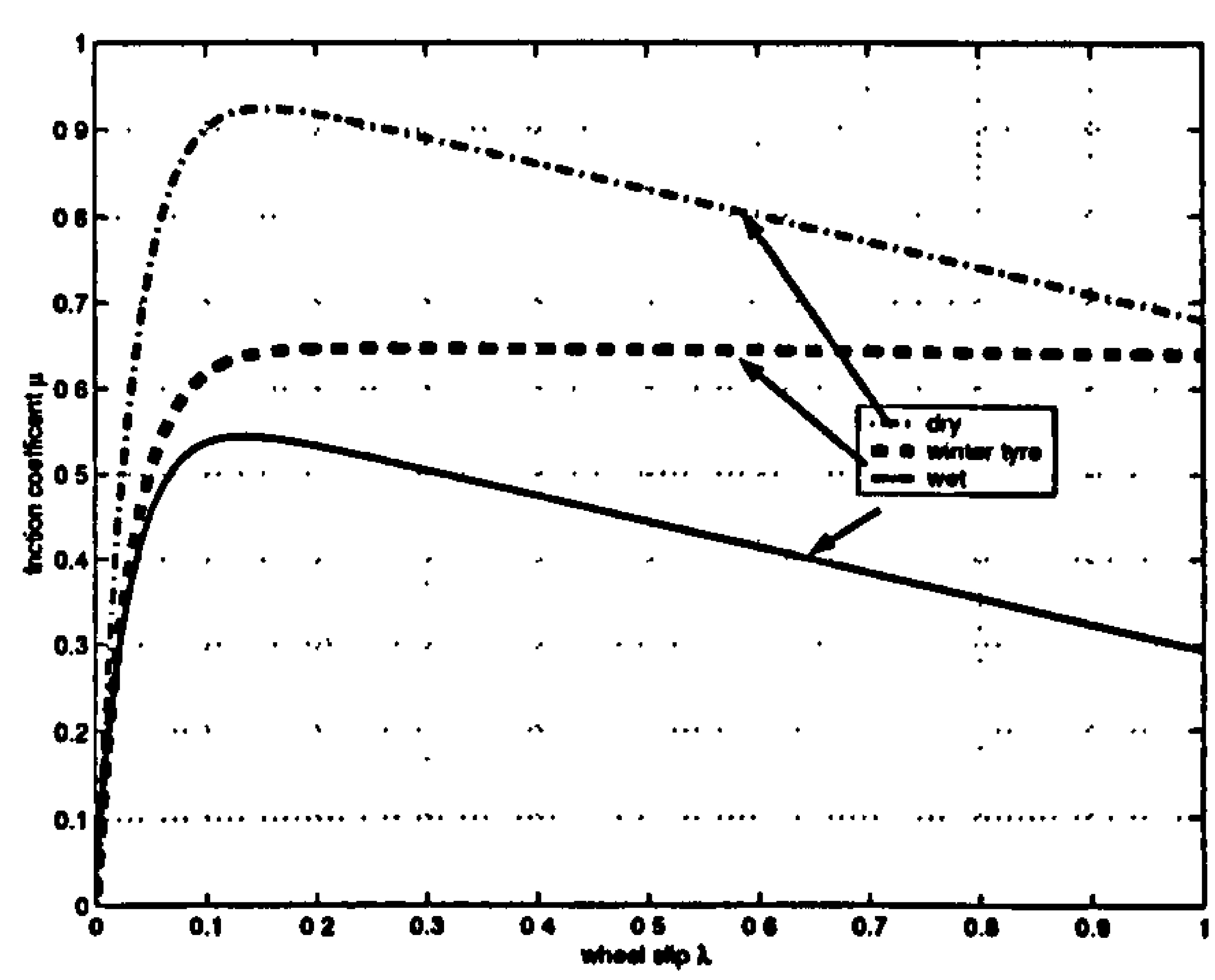


Figure 3.2: $\mu(\lambda)$ for wet and dry surfaces and winter tyres

The qualitative dependence of λ on surface conditions and tyre brand is shown in figure 3.2. The task of the ABS controller is to robustly stabilize the system around the maximum friction, such that minimum braking time, i.e. distance, is needed and the car's steerability is maintained. Before we start with the analysis and controller design we cast the nonlinear equations into piecewise linear equations. This is done by approximating the friction/slip curves by piecewise linear functions. After we have found a piecewise linear representation for the friction/slip curves the non-linear model of the braking quarter car is linearized.
In order to cover all possible dynamics we will approximate the $\mu(\lambda)$ with two piecewise linear functions

$$\mu = a\lambda \qquad for \quad \lambda \leq 0.1 \qquad (3.9)$$

$$\mu = -\frac{1}{4}\lambda + \frac{3}{4} \pm 0.2 \qquad for \quad \lambda > 0.1 \qquad (3.10)$$

where $a \in [5.75, 9.75]$ and the notation ± 0.2 means that any arbitrary, not necessarily fixed, value can be assumed in the interval (-0.2, 0.2). With this approximation we cover most values of μ. The approximation is shown in figure 3.3.

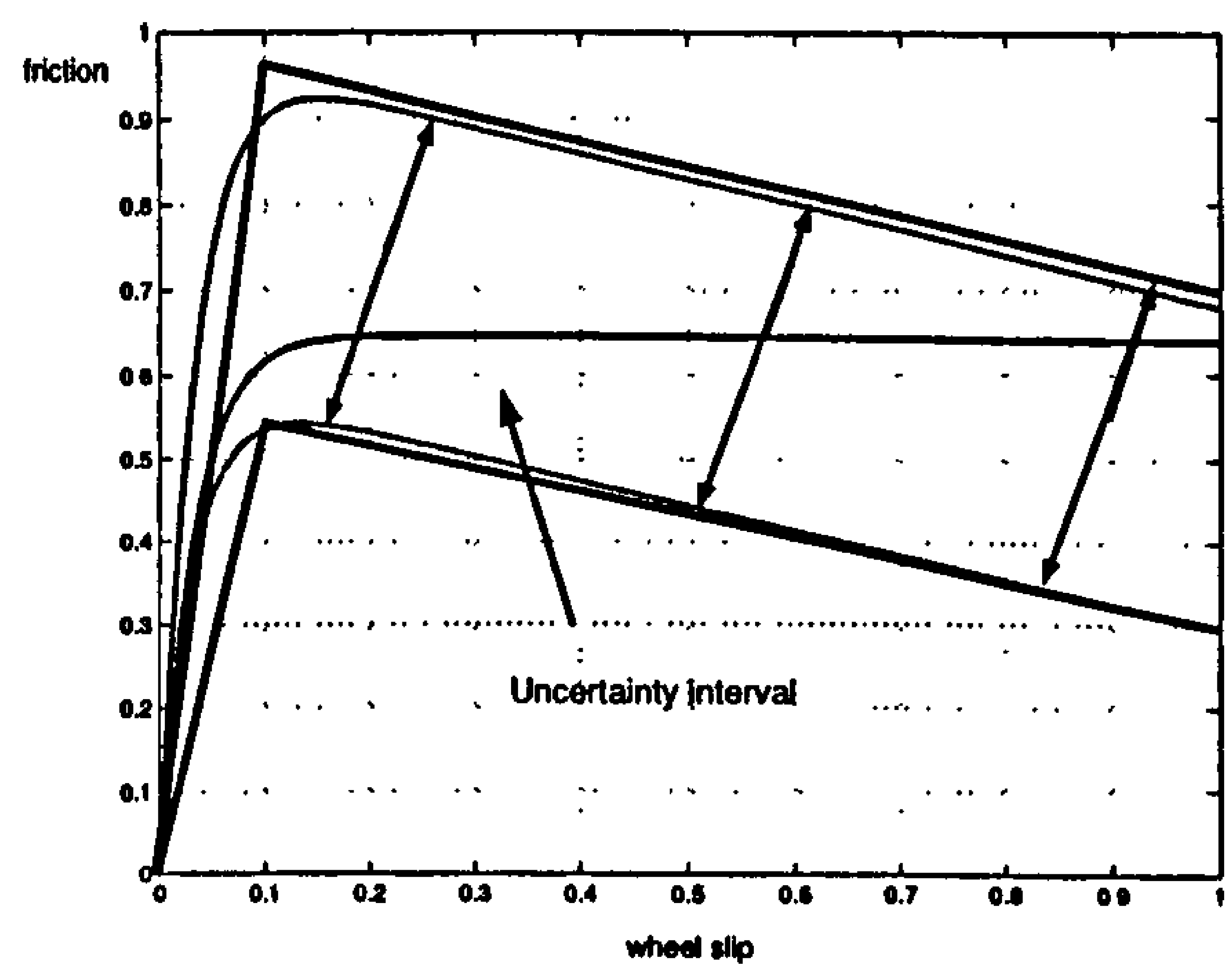


Figure 3.3: Approximation of $\mu(\lambda)$ with uncertain piecewise linear functions

Since road surfaces can change arbitrarily we need to cope with discontinuous jumps of μ, which may take place at arbitrary times. These unpredictable changes of μ are covered by the affine uncertainty term.

We note that since the mass m of a car is quite large, the term $\mid -\frac{1}{v}\left[\frac{1}{m}(1-\lambda)\right] F_z\mu(\lambda) \mid \ll \mid -\frac{1}{v}\left[\frac{r^2}{J}\right] F_z\mu(\lambda) \mid$, such that we will neglect it. For linearization we approximate the system by the first terms of the Taylor series $f(\lambda, v) \approx f(\lambda_{wp}, v_{wp}) + \frac{df}{d\lambda} \mid_{\lambda_{wp}, v_{wp}} (\lambda - \lambda_{wp}) + \frac{df}{dv} \mid_{\lambda_{wp}, v_{wp}} (v - v_{wp})$, such that if we do not use a change of coordinates we will get a linear (affine) system description

$$\left[\begin{array}{ccc} \dot{x} & = & A_q x + E_q + Bu^* \\ y & = & C_q x \\ q & = & f(x) \end{array}\right. \tag{3.11}$$

where $x(t) \in \bigcup \mathbf{X} \subset \mathbf{R}^2$, $q \in \mathbf{Q} = \{1, 2, \cdots, M, \cdots, N\} \subset \mathbf{Z}^+$ are the continuous and discrete states, respectively (Note: that the linearization can be arbitrarily dense, which will influence the number of subsystems N, where $N > M$. However, this has no influence on the following analysis nor on the controller synthesis since we use only the analytic representation). $u(t) \in \mathbf{U} \subset \mathbf{R}^+$ is the control input and A_q, B, C_q are the system, input and output matrices, respectively, of the subsystems. E_q are the affine terms and $f : \mathbf{X} \to \mathbf{Q}$ is the function indicating which subsystem is valid. For each subsystem $q \in \{1, 2, \ldots, M\}$, which are subsystems where $\lambda \leq 0.1$ we have

$$A_q = \begin{bmatrix} 0 & -a\frac{F_z}{m} \\ a\frac{F_z r^2 \lambda_{wp}}{v_{wp}^2 J} & -a\frac{F_z r^2}{v_{wp} J} \end{bmatrix} \tag{3.12}$$

$$E_q = \begin{bmatrix} 0 \\ -a\frac{F_z r^2 \lambda_{wp}}{v_{wp} J} \end{bmatrix} \quad (3.13)$$

and for subsystems $q \in \{M+1, M+2, \ldots, N\}$, which are subsystems where $\lambda > 0.1$ we have

$$A_q = \begin{bmatrix} 0 & \frac{F_z}{4m} \\ \left(-\frac{\lambda_{wp}}{4} + \frac{3}{4}\right)\frac{F_z r^2}{v_{wp}^2 J} \pm 0.2\frac{F_z r^2}{v_{wp}^2 J} & \frac{F_z r^2}{4 v_{wp} J} \end{bmatrix} \quad (3.14)$$

$$E_q = \begin{bmatrix} \left(-\frac{3}{4} \pm 0.2\right)\frac{F_z}{m} \\ \left(\frac{\lambda_{wp}}{4} - \frac{3}{2}\right)\frac{F_z r^2}{v_{wp} J} \pm 0.4\frac{F_z r^2}{v_{wp} J} \end{bmatrix} \quad (3.15)$$

and $B^T = [0, \frac{r}{J}]$, $u^* = u \cdot v$, $x^T = [v, \lambda]$ for $q \in \{1, 2, \ldots, N\}$. We have now cast the time-varying nonlinear system into a linear hybrid system with uncertainty. In the next section we will analyze the dynamics of the braking car by using its hybrid representation.

3.3 Analysis of the ABS dynamics

3.3.1 Stability analysis

For the stability analysis we transform the system matrices into controller canonical form $\tilde{A}_q = T A_q T^{-1}$ with T_1 $\forall q \leq M$ and T_2 $\forall q > M$

$$T_1 = \begin{bmatrix} -\frac{m}{aF_z} & 0 \\ 0 & 1 \end{bmatrix} \quad T_2 = \begin{bmatrix} \frac{4m}{F_z} & 0 \\ 0 & 1 \end{bmatrix} \quad (3.16)$$

such that we get

$$A_q = \begin{bmatrix} 0 & 1 \\ -a^2\frac{F_z^2 r^2 \lambda_{wp}}{m v_{wp}^2 J} & -a\frac{F_z r^2}{v_{wp} J} \end{bmatrix}$$

where $\lambda \leq 0.1$ and

$$A_q = \begin{bmatrix} 0 & 1 \\ \left(-\frac{\lambda_{wp}}{4} + \frac{3}{4}\right)\frac{F_z^2 r^2}{4 m v_{wp}^2 J} \pm 0.2\frac{F_z^2 r^2}{4 m v_{wp}^2 J} & \frac{F_z r^2}{4 v_{wp} J} \end{bmatrix}$$

where $\lambda > 0.1$. Since the system matrices are now given in controller canonical form it is easy to see whether or not the systems are stable. A system is Hurwitz stable if and only if all coefficients in the lowest row of the system matrix in controller canonical form are negative. We now immediately see that the subsystems $q \in \{1, 2, \ldots, M\}$, i.e. subsystems where $\lambda \leq 0.1$, are stable since the coefficients in the lower row of the system matrices are negative for all possible parameter variations. However, the subsystems $q \in \{M+1, M+2, \ldots, N\}$, i.e. systems where $\lambda > 0.1$, are not globally stable. However, we should not forget that none of the subsystems are valid on the whole state space. So we need to check if the subsystems for $\lambda > 0.1$ converge for $0.1 < \lambda \leq 1$ and $0 < v$. Taking the first equation for $\dot{v}$

$$\dot{v} = \frac{F_z\lambda}{4m} + \left(-\frac{3}{4} \pm 0.2\right)\frac{F_z}{m} < 0 \quad \forall\lambda \tag{3.17}$$

we see that the right hand-side of this differential equation remains negative since the maximum value of $\dot{v}$ occurs at $\lambda = 1$, which reduces the equation to

$$\dot{v} = \left(-\frac{1}{2} \pm 0.2\right)\frac{F_z}{m} < 0 \tag{3.18}$$

and it is easy to see that it is negative for all possible values. Hence for all values of λ and v the differential equation (DE) converges to values which belong to subsystems (3.12). The second differential equation for $\dot{\lambda}$

$$\dot{\lambda} = \left(-\frac{3}{4} \pm 0.2\right)\frac{F_z r^2}{vJ} + \frac{F_z r^2\lambda}{4vJ} < 0 \quad \forall\lambda, v \tag{3.19}$$

has a right hand-side which is also negative for all admissible λ and v. This can be easily seen if we substitute for the λ which would make the equation as least negative as possible. This is $\lambda = 1$ which brings the equation into this form:

$$\dot{\lambda} = \left(-\frac{1}{2} \pm 0.2\right)\frac{F_z r^2}{vJ} < 0 \quad \forall v \tag{3.20}$$

It is easy to see that this equation is also negative for all admissible velocities v. Hence also this DE converges (tends) to λ which belong to subsystems (3.12), for all admissible values of the states. We have seen that the subsystems (3.12), (3.14) converge individually, for all admissible initial states, to $x \equiv 0$. In general this does not mean that the whole system is stable. However since the states converge for any initial condition from subsystems (3.14) to states which belong to the subsystems (3.12) and (3.12) converges to zero, $x \to 0$ as $t \to \infty$. Hence the system is stable.
We have seen that the system is stable. It is further desirable to analyze the performance such that we know the maximum deceleration.

3.3.2 Computation of the maximum deceleration

We would like to compute the maximum deceleration. It is expected that the maximum deceleration is $\dot{v} \approx -g$ if the air resistance is neglected. Remark: in general the air resistance should not be neglected since its contribution especially at higher velocities is considerable particularly when the vehicle is equipped with spoilers.
If we look at the friction/slip curve the highest friction occurs at $\lambda = 0.1$. The models which are valid for $\lambda = 0.1$ are the ones described by (3.12). We take the first row of (3.12)

$\dot{v} = -a\frac{F_z}{m} \cdot \lambda$ and equate it at $\lambda = 0.1$, hence we obtain $\dot{v} = -\frac{ag}{10}$ since $F_z = g \cdot m$. For the best possible friction at $\lambda = 0.1$ we obtain $\dot{v} = -0.975 \cdot g \approx -g$.

In the next section we proceed with designing a controller, which will achieve this maximum deceleration.

3.4 Controller design

The objective is to design a controller which decelerates the vehicle as fast as possible and maintains steerability. We have seen that the maximum deceleration is reached at a slip of $\lambda = 0.1$. At such a slip the wheel is far away from being locked, such that we maintain the steerability of the car. We have also seen that it is sensible to approximate the nonlinear car dynamics by (3.12) and (3.14). For (3.12), i.e. subsystems where $\lambda \leq 0.1$, we would like to increase or maintain λ, i.e. we would like $\dot{\lambda} \geq 0$. For (3.14) we would like to reduce λ such that we get better steerability and braking performance, i.e. we would like $\dot{\lambda} < 0$. We compute now the control input space in dependence of the state space. Taking

$$\dot{\lambda} = -\frac{1}{v}\left[\frac{1}{m}(1-\lambda) + \frac{r^2}{J}\right] F_z\mu(\lambda) + \frac{1}{v}\frac{r}{J}T_b \tag{3.21}$$

for $\lambda \leq 0.1$ and $v > 0$ we want to have $\dot{\lambda} \geq 0$. Therefore we take

$$0 \leq -\frac{1}{v}\left[\frac{1}{m}(1-\lambda) + \frac{r^2}{J}\right] aF_z\lambda + \frac{1}{v}\frac{r}{J}T_b \tag{3.22}$$

hence,

$$T_b \geq \frac{aJ}{r}\left[\frac{1}{m}(1-\lambda) + \frac{r^2}{J}\right] F_z\lambda \tag{3.23}$$

Using the simplification as before we obtain

$$T_b \geq arF_z\lambda \tag{3.24}$$

For the maximum value of λ, $\lambda = 0.1$

$$T_b \geq rF_z. \tag{3.25}$$

For subsystems (3.14) we desire a negative $\dot{\lambda}$, i.e. $\dot{\lambda} < 0$, hence

$$0 \geq -\frac{1}{v}\left[\frac{1}{m}(1-\lambda) + \frac{r^2}{J}\right] F_z\mu(\lambda) + \frac{1}{v}\frac{r}{J}T_b \tag{3.26}$$

Simplifying we obtain

$$T_b < \frac{11}{20}rF_z. \tag{3.27}$$

for values of λ which are close to 0.1. It is easy to see that there exists no continuous state feedback controller that achieves the desired performance if it is assumed that the friction can vary arbitrarily. This does not mean that there exists no continuous feedback which stabilizes the system. There are of course continuous controllers which stabilize the system. It can be shown that admissible control inputs are $0 \leq T_b < 0.3rF_z$, which stabilize the system for any initial condition. $T_b \geq 0$ is a technical requirement since the wheels during braking cannot be accelerated. $T_b < 0.3rF_z$ is necessary to ensure that the wheels will not lock.

One possibility is to design a sliding mode controller (77) (73), where the sliding surface is $s = \left(\frac{d}{dt} + K\right) \int_0^t e d\tau$ with $e = \lambda - \lambda_d$, i.e. $\dot{s} = \dot{e} + Ke$. Thus,

$$\dot{s} = -\frac{r^2 F_z \mu(\lambda)}{vJ} + \frac{1}{v}\frac{r}{J}T_b + Ke \tag{3.28}$$

To stay on the surface $\dot{s} = 0$ is required. Solving for T_b and adding the term which forces the trajectory to stay on the surface we get the control input

$$\hat{T}_b = rF_z\mu(\lambda) - \frac{vJ}{r}Ke \tag{3.29}$$

The control input is a function of the friction which is unknown. To overcome this an observer can be designed. However it is known that friction observers have poor performance therefore we would like to pursue a modified strategy.

We suggest a heterogeneous hybrid controller, which has a similar structure as the sliding mode controller. The controller use 3 different control strategies (3.30), (3.31), (3.32): the first one (3.30) is active for λ smaller 0.1 and aims to increase λ.

$$\hat{T}_b = 10rF_z\lambda - \frac{vJ}{r}Ke \tag{3.30}$$

The second one (3.31) stabilize $\lambda \approx 0.1$

$$\hat{T}_b = -\frac{sK_P + K_i}{s}e \tag{3.31}$$

while the third one (3.32) is active for λ larger 0.1 and reduce λ.

$$\hat{T}_b = \left(-\frac{1}{4}\lambda + \frac{3}{4} - 0.2\right) rF_z - \frac{vJ}{r}Ke \tag{3.32}$$

A hysteresis is introduce such that strategy (3.31) become active as soon as strategy (3.30) or (3.32) bring $\lambda = 0.1$. Once the control strategy (3.31) is active it remains active for $0.08 < \lambda < 0.12$. In this way we get vector fields that point towards $\lambda = 0.1$ and in directions of smaller velocities v. To avoid chattering the PI controller stabilizes the dynamics around the desired slip $\lambda_d = 0.1$. It can further handle smaller variations in the friction coefficient. For larger variations controller (3.30) and (3.32) pushes the trajectory back to $\lambda_d = 0.1$ where the PI controller takes over again.

It needs to be mentioned that such a discontinuous control law, besides the advantages of being robust to variations and uncertainty and achieving high braking performance, has the drawback that it might excite unmodelled dynamics such as suspension dynamics. This is undesirable since it reduces passenger comfort.

To illustrate the controller's performance a simulation example is presented.

3.5 Simulation

In the example two discontinuous changes to the nominal friction coefficient 0.8 are made. At $t = 0.5s$ the friction is lowered by 0.3 and at $t = 1.2s$ the friction is increased by 0.28. Figure 3.4 shows the velocity of the car and of its wheel.

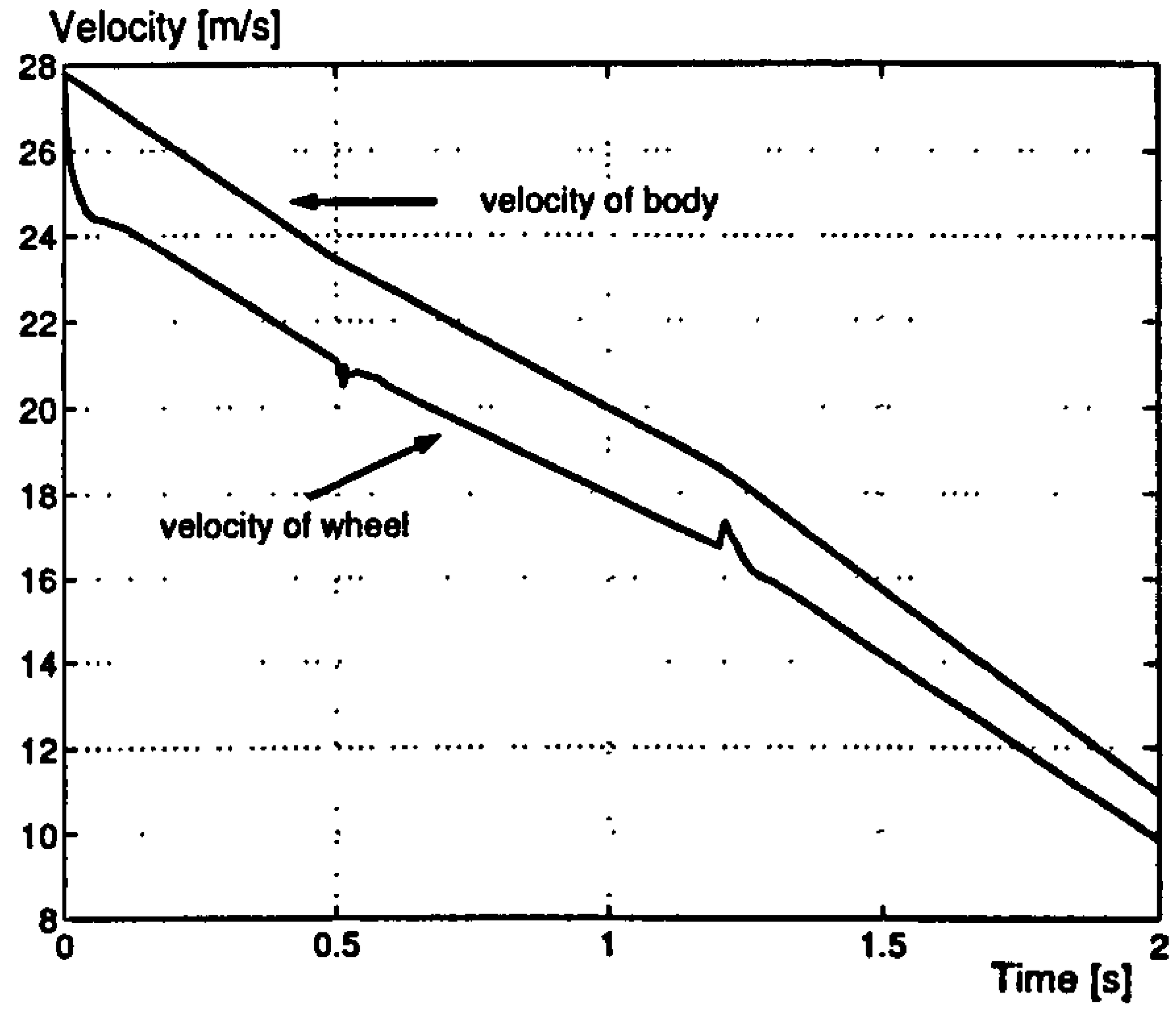


Figure 3.4: Velocity of the car body and the wheel $\omega * r$

Due to the slip the velocity of the wheel is lower than the velocity of the car body. It can also be seen that the variations in slip result in variations of the wheel velocity. Figure 3.5 shows the slip λ, and Figure 3.6 shows the brake torque $\hat{T}_b$.

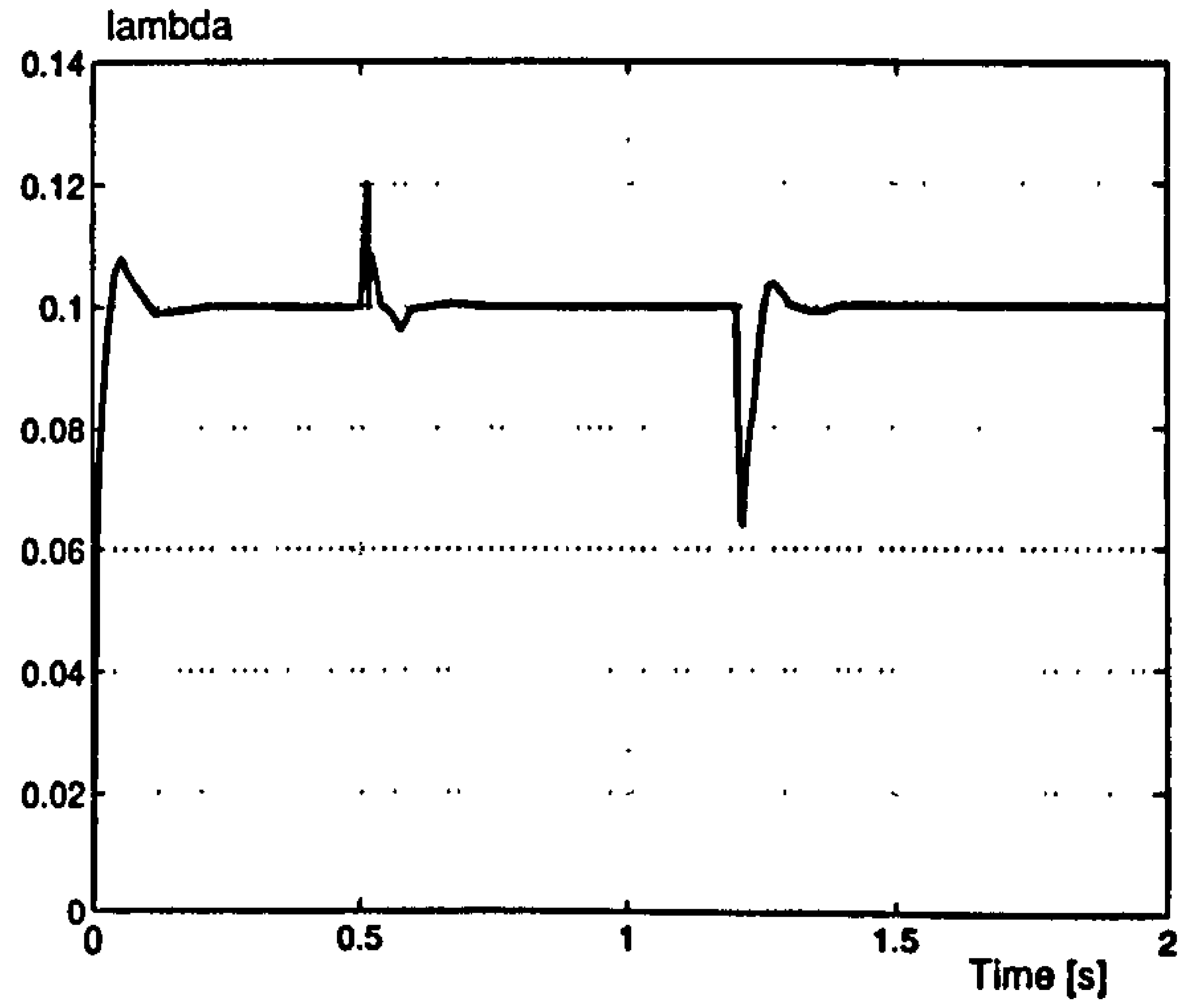


Figure 3.5: Wheel slip λ

In the first instance controller (3.30) is active and brings the slip towards $\lambda = 0.1$ by applying maximum torque. Then the PI controller takes over and stabilizes the slip. At $t = 0.5$ the friction is decreased and the slip increases. Controller (3.32) takes over immediately as $\lambda = 0.12$ and pushes it back where the PI controller takes over again. At time $t = 1.2$ the friction is increased and controller (3.32) brings the slip back.

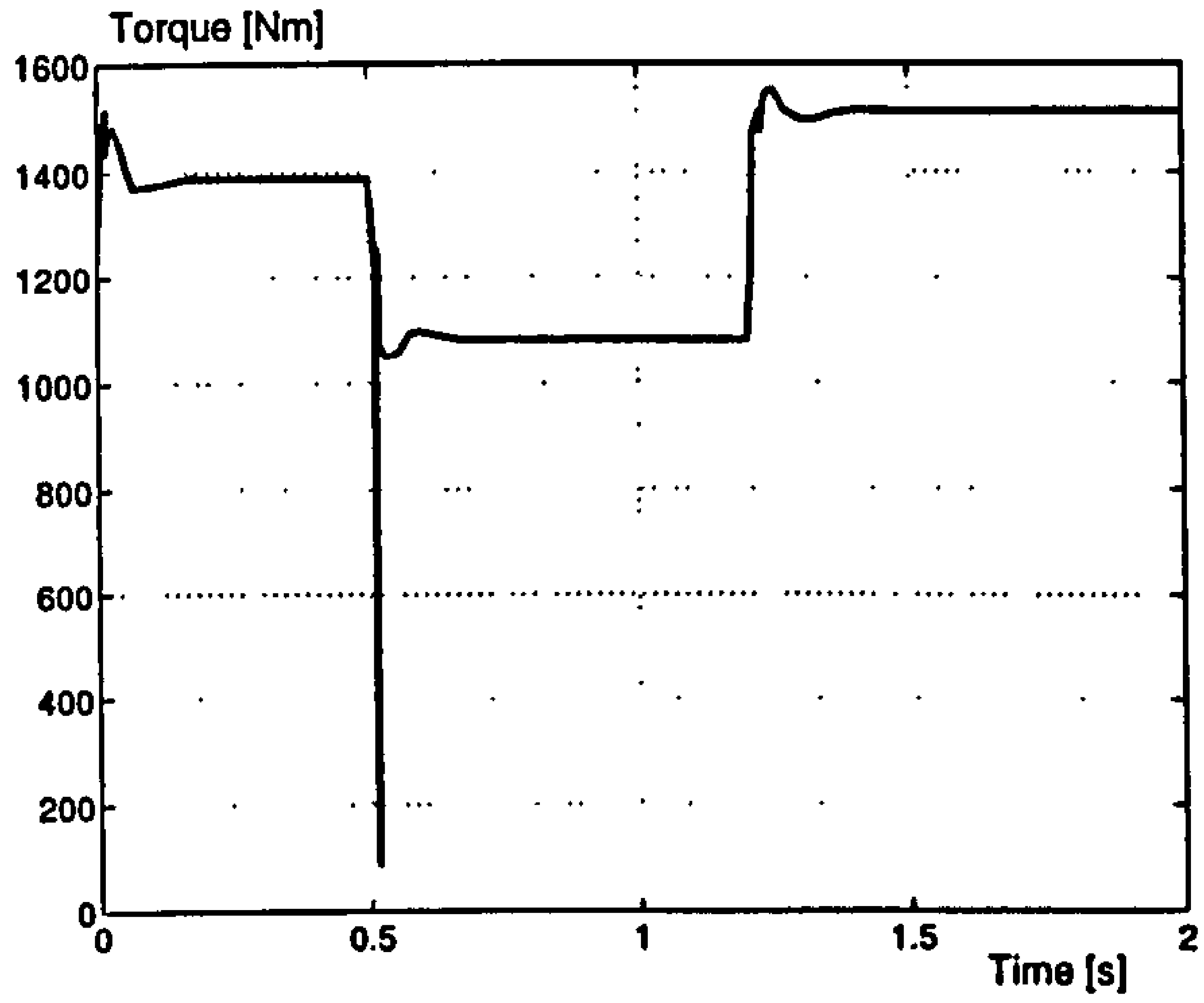


Figure 3.6: Brake torque $\hat{T}_b$

For tuning the controller needs to be implemented in a real car where its performance can be evaluated, since the final trade off between performance and comfort can only be achieved in the real environment. For tuning we suggest varying the coefficient K. With larger K we will get faster dynamics, such that $| \dot{\lambda} |$ will be larger. For smoother control action K needs to be reduced, such that suspension dynamics are excited less.

3.6 Conclusions

After a brief introduction to the history of ABS a nonlinear car model was introduced which captured the longitudinal braking dynamics. It was shown that the dynamics of a braking car can be cast into a linear hybrid system with uncertainty. The uncertainties captured the unpredictable changes in road friction due to changes in surface conditions (wet, dry). It was shown that the dynamics are stable and that the maximum braking performance occurs at $\lambda = 0.1$. The control input space was computed and it was shown that for $\lambda \leq 0.1$ the slip has to be increased in order to increase the friction, i.e. $\dot{\lambda} \geq 0$. For slips $\lambda > 0.1$, the slip has to be reduced to increase the friction and maintain steerability. It was shown that a continuous feedback could not achieve the maximum braking performance given the range of uncertainty.

Therefore it was suggested to design a heterogenous hybrid controller in sliding mode form. In order to avoid the excitation of unmodelled suspension dynamics the controller was chosen such that a relatively smooth transition is possible. The controller can now be tuned towards better performance, i.e. increase the bang bang control, or greater passenger comfort.

Part II

Controller design for Hybrid Systems

4 Robust control of hybrid systems

This chapter introduces a new method for controller design of uncertain and parameter-variant linear hybrid systems. The idea is to specify a desired performance, which is represented by a nominal system. Around this nominal system a compact set of systems is obtained which will be robustly stable against switching among members of this set: such a set of systems is then called the stable switched set. The chapter shows that obtaining the stable switched set is a signomial program. It is shown that upper bounds on signomial programs can be easily obtained, which are used to compute the stable switched set. A sufficient condition is given for the existence of a common state feedback controller that stabilizes the uncertain and parameter-variant linear hybrid systems on a stable switched set. The chapter proposes a synthesis procedure in terms of a constrained convex optimization problem that places the uncertain and parameter-variant subsystems optimally close to the desired nominal system, using one common state feedback controller. An extension is shown for the case that there exists no common state feedback controller for all subsystems. The synthesis framework is then applied to a simple example to demonstrate the procedure.
The contributions of this chapter are the robustness analysis framework, as well as the robust controller design for uncertain and parameter-variant linear hybrid systems. This novel robustness analysis framework for uncertain and parameter-variant linear hybrid systems, together with the robust controller synthesis procedure, has been published in (67), (68).

4.1 Introduction

Hybrid systems in piecewise linear form are often used in practice, since in many cases they stem from linearization of complex nonlinear systems. The controller synthesis is carried out for the linearization, and the stability and performance is verified by linear analysis or simulation. This procedure is justified for nonlinear systems where the nonlinearities can be described by piecewise linear systems (18) or systems with weak nonlinearities where parameter variations are negligible in the operating vicinity of the linearization. However, for a large class of systems, like systems with uncertainty or parameter variations, or systems with strong nonlinearities, it is impractical to carry out analysis or controller synthesis with the linearized systems. In these cases parameter variations have to be considered, or the number of linearizations needs to be increased, which is not practical in many cases. Therefore it is important to use the class of uncertain and parameter-variant linear hybrid systems.

The form of hybrid system that we will use incorporates bounded parametric uncertainties and variations. In this way we can enlarge the validity region around the linearization point and deal with uncertain and time-variant systems. The system is assumed to be in the form:

$$\left[\begin{array}{ccc} \dot{x} & = & A_{\Delta,q}x + B_q u \\ y & = & C_q x \\ q & = & f(x) \end{array}\right. \tag{4.1}$$

$x(t) \in \bigcup X_q \subset \mathbf{R}^n$, $u(t) \in U \subset \mathbf{R}$ are state and input respectively. The matrices $A_{\Delta,q}$, B_q and C_q are according to the dimensions and the parameters of $A_{\Delta,q}$ belong to $S_q = \{a_{q,i} \in \mathbf{R} \ \ |a_{q,i} \in [a^-_{q,i}, a^+_{q,i}]\}$. The function $f : \mathbf{R}^n \to \mathbf{Z}^+$ which governs the discrete state $q \in Q = \{1, 2, \cdots, N\} \subset Z^+$ is a function of the continuous state. Note that we do not assume, as in LPV control (69), that parameters and possibly their rate of variations are on-line measurable. However, for simplicity we assume that $A_{\Delta,q}$ is given or can be transformed by similarity transformation $\tilde{A} = TAT^{-1}$ into controller canonical form. For parameter-invariant systems this is less restrictive than for parameter-variant systems. The controller canonical form is not a real requirement, since any general form of uncertain and parameter-variant piecewise linear system can be handled in our analysis framework while some restrictions apply to the controller synthesis. However, derivations are easier in this way and make the controllability discussion redundant, as well as giving a nice geometric interpretation in parameter space.

In the following we will show how a set of systems is obtained that is robustly stable against switching between members of its set. This is the set of systems which we will refer to as the stable switched set (S^3). It is described by a polyhedral region SSS in the corresponding parameter space. It is shown that finding the S^3 by using quadratic or piecewise quadratic Lyapunov functions corresponds to solving a signomial programming problem.

Definition: $g(x) = \pm \sum_{j=1}^{m} \prod_{i=1}^{n} x_{j,i}^{p_{j,i}}$ is called a signomial function. A signomial programming is then $\min_x g(x)$ subject to constraints. Since signomial programs are non-convex, we take the upper bound to this problem which can be stated analytically. Using this method and increasing the parameter region, usually around a nominal system, gives us an iterative procedure to obtain the stable switched set. Such a set of systems is then used for the controller design of uncertain and parameter-variant linear hybrid systems where we give the sufficient condition for the existence of a common state feedback controller that stabilizes all subsystems of the hybrid system on S^3. If there exists a common state feedback controller we introduce a method that places the parameters optimally close to a specified nominal system, which is an element of the S^3. The controller synthesis is formulated in terms of a constrained convex optimization problem. In cases where a common state feedback law does not exist we introduce a method that finds the minimum number of controllers. To demonstrate the procedures we use a simple uncertain and parameter-variant linear hybrid system, such that it is easy to follow the design steps.

4.2 Calculation of the stable switched set

The aim is to find a set of stable systems around a nominal system, such that stability is maintained under switching among members of the set. The parameters of the normalized systems form a compact set in $\mathbf{R}^n$. Such a set of systems could be viewed as a single parameter-variant system, where the parameter-variant system is stable for all its parameter variations.

In the following we will describe how the stable switched set S^3 is obtained. The procedure starts with an arbitrary stable system, however in the case of analysis of uncertain and parameter-variant hybrid systems the nominal system might be chosen, while in the synthesis case performance requirements might be translated into a nominal system which we will then use. A quadratic Lyapunov function is then computed by solving the Lyapunov inequality for the nominal system. The set of systems is extended by extending the parameter set around the nominal parameters, the Lemmas below are then used to check whether the extended set still belongs to the stable switched set. This will give an iterative procedure to compute larger and larger subsets of the stable switched set. We will now state two lemmas, which will be needed in the sequel.

Lemma 1: Given a signomial function $f_1(a_1, a_2, \dots, a_n)$, $f_1 : \mathbf{R}^n \to \mathbf{R}$, which is a product of its variables and their powers $f_1 = \pm \prod_{i=1}^n a_i^{p_i}$ with $p_i \in \mathbf{N}$, $\forall i \in \{1, 2, \dots, n\}$ on the polyhedral set $S = \{a_i \in \mathbf{R} \quad | \quad a_i \in [a_i^-, a_i^+], \quad sign(a_i^-) = sign(a_i^+)\}$. Define $\bar{a}_i = max(|a_i|)$ and $\underline{a}_i = min(|a_i|)$. Then $\max_{a_i \in S} f_1(a_1, a_2, \dots, a_n) = f_1(\bar{a}_1, \bar{a}_2, \dots, \bar{a}_n)$ is the unique maximum on the set S if f_1 is positive, $\max_{a_i \in S} f_1(a_1, a_2, \dots, a_n) = f_1(\underline{a}_1, \underline{a}_2, \dots, \underline{a}_n))$ otherwise.

Proof: Assume otherwise, then for a positive f_1 there is at least one fixed point $f_1(a_1, a_2, \dots, a_n)$ $> f_1$ $(\bar{a}_1, \bar{a}_2, \dots, \bar{a}_n)$ in the image of f_1, that is $\prod_{i=1}^n a_i^{p_i} > \prod_{i=1}^n a_i^{+p_i}$. By comparison of the coefficients we obtain $a_i < \bar{a}_i$, $\forall\, a_i \neq \bar{a}_i$, so that we arrive at a contradiction. Hence $f_1(\bar{a}_1, \bar{a}_2, \ldots, \bar{a}_n)$ is the unique maximum on the set S if the function is positive. If f_1 is negative, there is at least one fixed point $-f_1(a_1, a_2, \ldots, a_n) \leq -f_1(\underline{a}_1, \underline{a}_2, \ldots, \underline{a}_n))$. By comparison of the coefficients $a_i > \underline{a}_i$, $\forall\, a_i \neq \underline{a}_i$, so that we arrive at contradiction again. Hence $f_1(\underline{a}_1, \underline{a}_2, \ldots, \underline{a}_n))$ is the unique maximum on S if f_1 negative.

□

Lemma 2: Given the signomial function $f_m(a_1, a_2, \ldots, a_n)$, $f_m : \mathbf{R}^n \to \mathbf{R}$, with $f_m = \sum_{j_1=1}^{m_1} \prod_{i=1}^{n} a_i^{p_{j_1 i}} - \sum_{j_2=1}^{m_2} \prod_{i=1}^{n} a_i^{p_{j_2 i}}$, $p_{j_1 i}, p_{j_2 i} \in \mathbf{N}$, and $m_1 + m_2 = m$, defined on the polyhedral set $S = \{a_i \in \mathbf{R} \mid a_i \in [a_i^-, a_i^+],\ sign(a_i^-) = sign(a_i^+)\}$. Then $\max_{a_i \in S} f_m \leq \sum_{j_1=1}^{m_1} \prod_{i=1}^{n} \bar{a}_i^{p_{j_1 i}} - \sum_{j_2=1}^{m_2} \prod_{i=1}^{n} \underline{a}_i^{p_{j_2 i}}$.

Proof 2: By Lemma 1 $\prod_{i=1}^{n} a_i^{p_{j_1 i}} \leq \prod_{i=1}^{n} \bar{a}_i^{p_{j_1 i}}$ and $-\prod_{i=1}^{n} a_i^{p_{j_2 i}} \leq -\prod_{i=1}^{n} \underline{a}_i^{p_{j_2 i}}$ $\forall\, j_1, j_2$ on the set S. Hence $\sum_{j_1=1}^{m_1} \prod_{i=1}^{n} a_i^{p_{j_1 i}} - \sum_{j_2=1}^{m_2} \prod_{i=1}^{n} a_i^{p_{j_2 i}} \leq \sum_{j_1=1}^{m_1} \prod_{i=1}^{n} \bar{a}_i^{p_{j_1 i}} - \sum_{j_2=1}^{m_2} \prod_{i=1}^{n} \underline{a}_i^{p_{j_2 i}}$ on S.

□

According to Sylvester's theorem a symmetric matrix $M = M^T$ is positive definite if all its principal minors M_{11}, $M_{11}M_{22} - M_{12}M_{21}, \ldots$ det (M) are strictly positive (73). Hence we get n inequalities which we need to check, to verify that the Lyapunov inequality $-M = A_q^T P + P A_q < 0$ is satisfied. Taking the left hand side of the n inequalities to be the signomial $f_{m_k,k}(a_1, a_2, \ldots, a_n)$ with $k = 1, 2, \ldots, n$ we have to check that $\max_{a_i \in S} f_{m_k,k} < 0$, $\forall\, k$, which is a signomial program. If $\max_{a_i \in S} f_{m_k,k} < 0$, $\forall\, k$, then the quadratic Lyapunov function holds on the polyhedral set $S = \{a_i \in \mathbf{R} \mid a_i \in [a_i^-, a_i^+],\ sign(a_i^-) = sign(a_i^+)\}$ for a given P.

We can now use Lemma 2 to replace $\max_{a_i \in S} f_{m_k,k} < 0$ by its upper bound. In this way we reduce the number of checks to n, to check if all systems which have their parameters entirely in S belong to the set of stable switched systems. The set of systems that have all their parameters in S will therefore be a subset of the stable switched set, $S \subseteq SSS$, such that switching between members of this subset will lead to a stable switched system. As we can further increase S and repeatedly check if $\max_{a_i \in S} f_{m_k,k} < 0$ is satisfied for all k we will enlarge the set of systems such that we come closer and closer to cover the complete set of stable switched systems.

With this method we have found an iterative procedure to compute a subset of the stable switched set. Figure 4.1 shows the procedure after the second iteration for a third order system. Each system is represented by one point in the parameter space A^n. All the points inside the quarter correspond to systems which belong to the stable switched set.

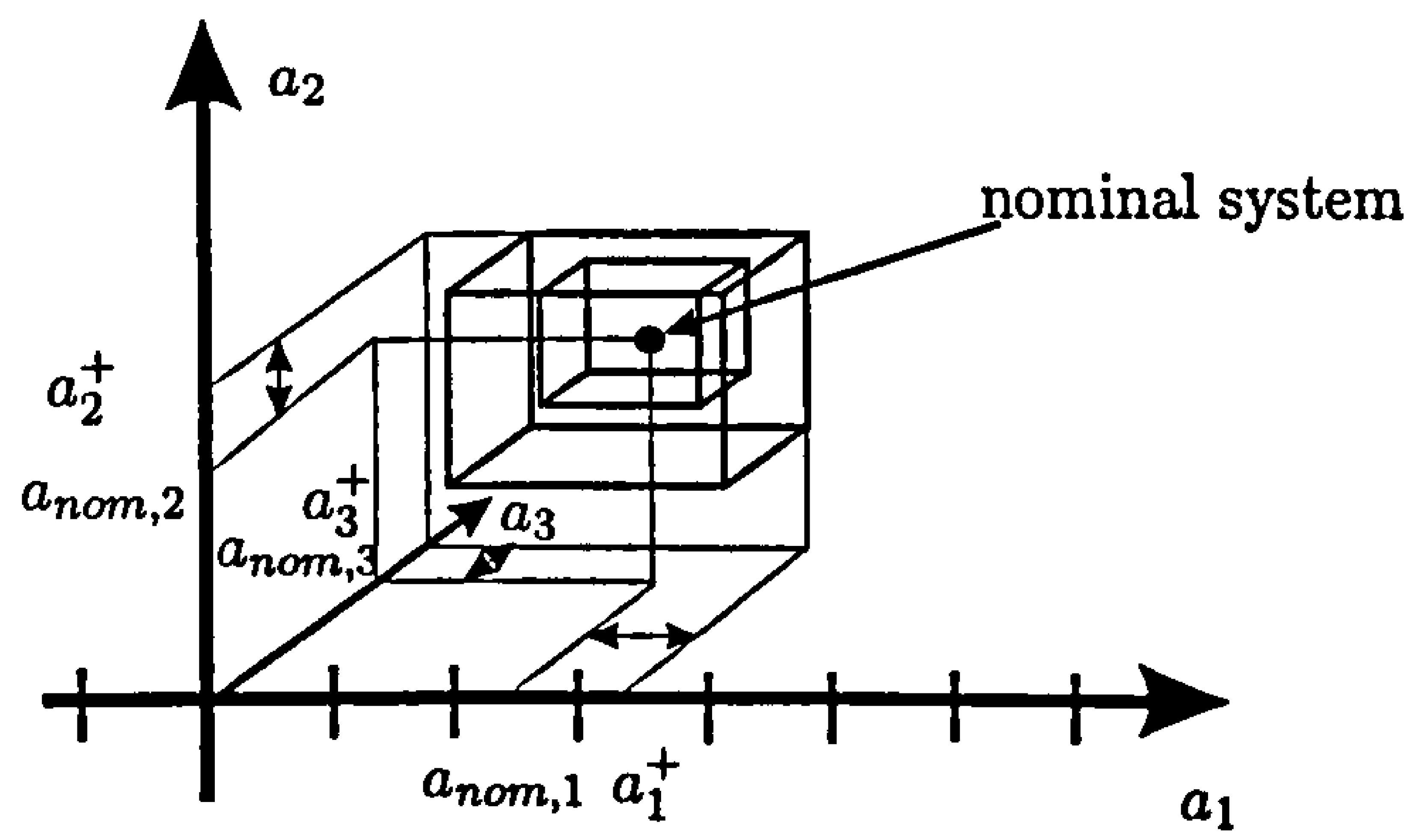


Figure 4.1: Computing the stable switched set

Note that this method is not restricted to common quadratic Lyapunov functions. Whenever the discrete state q of the uncertain and parameter-variant linear hybrid system is a function of the continuous state x, such that one subsystem is valid only in a region X_q, we can make use of the S-procedure as described in (47) (21). The Lyapunov inequality

$$V(x) = x^T P x \quad P = P^T > 0 \quad A_q^T P + P A_q < 0$$

has to hold only for $x \in X_q$. Using the S-procedure relaxes the inequality since $S > 0$ when $x \in X_q$ and $S < 0$ elsewhere. This makes it in general easier to satisfy

$$V(x) = x^T P x \quad P = P^T > 0 \quad A_q^T P + P A_q + S_q < 0.$$

The method can also be applied to piecewise quadratic Lyapunov functions $V(x) = x^T P_q x$ (60) (59), where the Lyapunov function might be discontinuous but decreasing $x^T P_q x \geq x^T P_{q^+} x$ at switching times $f_{q,q^+}^T x = 0$.

$$V(x) = x^T P_q x \quad P_q = P_q^T > 0 \quad A_q^T P_q + P_q A_q + S_q < 0$$

$$P_q - P_{q^+} + f_{q,q^+} t_{q,q^+}^T + t_{q,q^+} f_{q,q^+}^T > 0$$

where q is the predecessor of q^+ and f_{q,q^+}^T describes the switching surface between system q and q^+. The approach can also handle uncertain and parameter-variant affine hybrid systems $\dot{x} = A_{\Delta,q} x + E_{\Delta,q} + Bu$. Here,

$$V(x) = \tilde{x}^T P \tilde{x} \quad P = P^T > 0 \quad \tilde{A}_q^T P + P \tilde{A}_q < 0$$

where we write

$$\tilde{x}^T = [x, 1], \quad \tilde{A}_q = \begin{bmatrix} A_q & E_q \\ 0_{1\times n} & 0 \end{bmatrix}.$$

This translates into $n+1$ signomial programs.
Roughly speaking the extension from quadratic to piecewise quadratic Lyapunov functions is just a matter of handling more inequalities, but remains a signomial programing problem, as shown above. Therefore this matter is not further discussed here since it extends naturally.
A different method to obtain the SSS is given in (20). An iterative procedure is proposed to compute a subset of the stable switched set. Instead of evaluating an explicit algebraic expression an convex optimization problem is solved at each iteration step.
In the following section the problem of existence of a common state feedback controller, which shifts the uncertain and parameter varying subsystems into the SSS, is discussed.

4.3 On existence of a common state feedback controller

In this section the synthesis problem is discussed, to determine whether there exists a single state feedback controller which stabilizes a number of uncertain and parameter-variant subsystems simultaneously. That is, given a finite number N of uncertain and parameter-variant subsystems with parameter uncertainties or variations in the sets $S_q = \{a_{q,i} \in \mathbf{R} \ \ |a_{q,i} \in [a_{q,i}^-, a_{q,i}^+]\}$, $q \in \{1,2,\ldots,N\}$, $i \in \{1,2,\ldots,n\}$ does there exist a single state feedback controller which stabilizes all systems robustly on a given stable switched set?
We will now give the sufficient condition for the existence of a common state feedback controller (Note: the condition is necessary and sufficient if we have obtained the complete S^3, in our case we have just obtained a subset). Let N denote the number of uncertain and parameter-variant systems. Let the elements $R_{i,q}$ of the matrix R be the sets that contain the feasible intervals for a state feedback controller $u = -Dx$, $D = [d_1 \ldots d_n]$.

$$R = \begin{bmatrix} R_{1,1} = (a_1^- - a_{1,1}^-, a_1^+ - a_{1,1}^+) & \cdots & R_{1,N} = (a_1^- - a_{N,1}^-, a_1^+ - a_{N,1}^+) \\ \vdots & \cdots & \vdots \\ R_{n,1} = (a_n^- - a_{1,n}^-, a_n^+ - a_{1,n}^+) & \cdots & R_{n,N} = (a_n^- - a_{N,n}^-, a_n^+ - a_{N,n}^+) \end{bmatrix}$$

where a_i^-, a_i^+, $i \in \{1,2,\ldots,n\}$ are the lower and upper bound respectively of parameters in the previously computed subset of the stable switched set SSS, and $a_{q,i}^-$, $a_{q,i}^+$, $q \in \{1,2,\ldots,N\}$ denote the lower and upper bound of the i^{th} parameter in the q^{th} subsystem.
The matrix R has a very nice geometric interpretation. It consists of the sets which contain all state feedback controllers that translate each S_q such that it is entirely contained in SSS, the stable switched set.

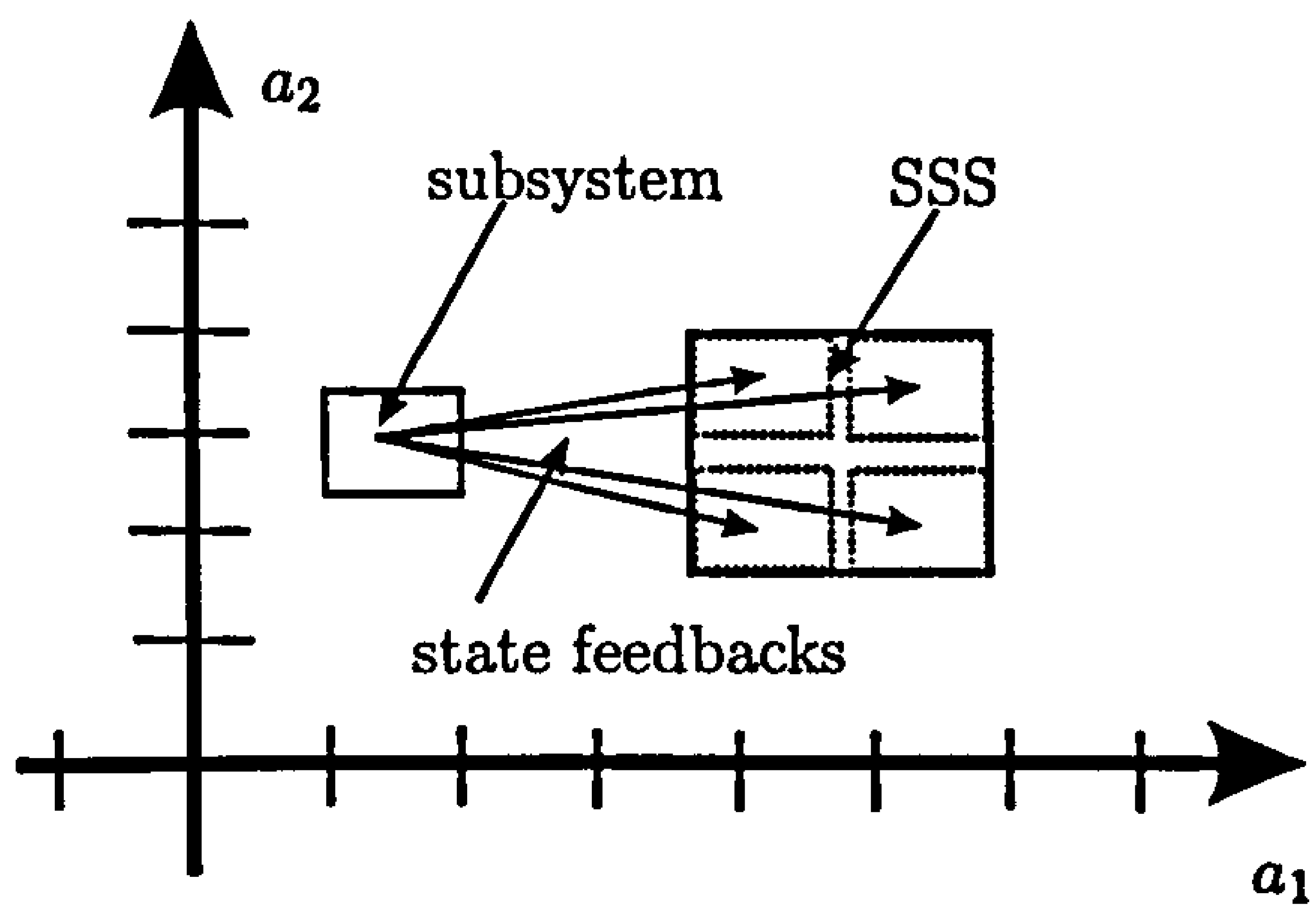


Figure 4.2: Set of state feedbacks which bring the subsystem into SSS

The figure 4.2 sketches how the various state feedbacks bring the parameters S_q of a subsystem q into the parameter region SSS of the S^3. Having obtained the set of all possible state feedback controllers for each subsystem q we now have to intersect these sets to see if there exists a common state feedback controller.

Let $Check_i$ denote the sets that contain the intersection of the sets in each row of R.

$$\begin{array}{lcl} Check_1 & = & R_{1,1} \bigcap \cdots \bigcap R_{1,N} \\ Check_2 & = & R_{2,1} \bigcap \cdots \bigcap R_{2,N} \\ \vdots & = & \vdots \quad \cdots \quad \vdots \\ Check_n & = & R_{n,1} \bigcap \cdots \bigcap R_{n,N} \end{array} \tag{4.2}$$

Hence there exists a control law that uses a single state feedback controller if, and only if (in case we obtained the complete S^3), all sets $Check_i$ are non empty.

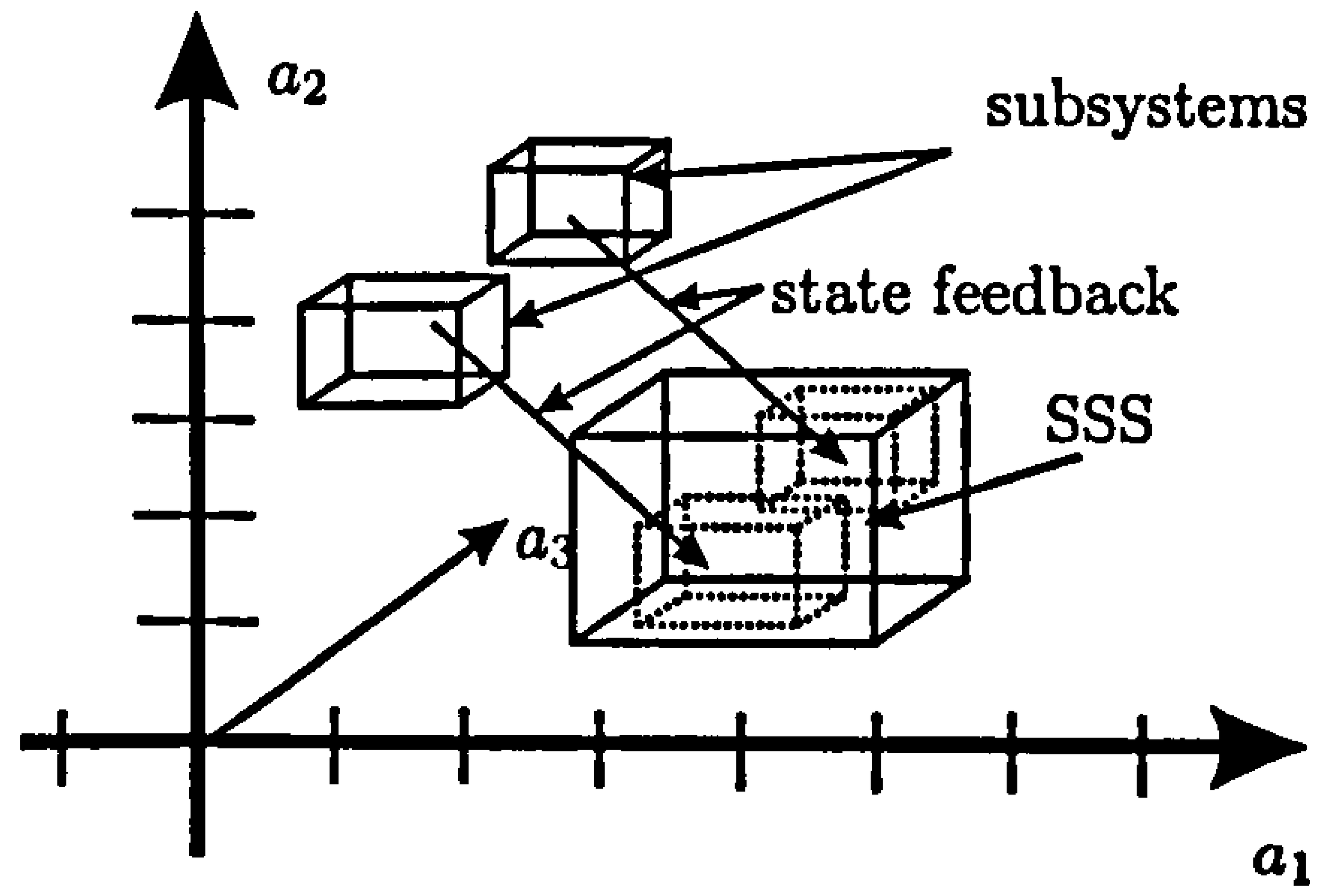


Figure 4.3: Common state feedback brings subsystems into SSS

Assuming that such a controller exists (see figure 4.3), we can design the state feedback controller $u = -Dx$, where $d_1 \dots d_n$ are the coefficients of D, in an optimal manner such that we minimize

$$\min_D J = \min_D I_n^T G I_N \qquad (4.3)$$

subject to

$$\begin{array}{ccc} d_1 \in R_{1,1} & \dots & d_1 \in R_{1,N} \\ \vdots & \dots & \\ d_n \in R_{n,1} & \dots & d_n \in R_{n,N} \end{array} \qquad (4.4)$$

where I_n,I_m are the unit vectors of length n, N respectively. G is defined as follows:

$$G = \begin{bmatrix} \|a_{nom,1} - a_{1,1} - d_1\| & \dots & \|a_{nom,1} - a_{N,1} - d_1\| \\ \vdots & \dots & \vdots \\ \|a_{nom,n} - a_{1,n} - d_n\| & \dots & \|a_{nom,n} - a_{N,n} - d_n\| \end{bmatrix}$$

where $a_{nom,i}$ is the i^{th} parameter of the nominal system and $a_{q,i} = a_{q,i^-} + \frac{a_{q,i^+} - a_{q,i^-}}{2}$ is the mean value of the i^{th} parameter in the q^{th} subsystem. This problem is a constrained convex optimization and can be solved with linear programming. Hence we have formulated the controller design as a convex optimization problem.

4.4 Extension to Multiple State Feedback Control

In some cases is is not possible to find a common state feedback controller that will achieve the required specifications. In other words, at least one set $Check_s$ will be empty. In such cases the number of controllers has to be increased. Extending the design procedure for such a case we find that

$$\begin{aligned} W &= n \cdot (\sum_{i=1}^{N-1} N - i + \sum_{i=1}^{N-2} N - 1 - i + \dots \\ &\quad \dots + \sum_{i=1}^{N-N+2} N - N + 2 - i) \end{aligned} \qquad (4.5)$$

checks need to be carried out in the worst case, where N is the number of subsystems and n is the order of the controlled system, in order to find the minimum number of controllers.

These checks will be carried out as follows: we need to check that $R_{1,1} \bigcap R_{1,2}, \ldots, R_{n,1} \bigcap R_{n,2}$ are all not empty. If that is the case then there exists a common state feedback controller for these two subsystems. If at least one set is empty then there does not exist such a common state feedback controller that shifts the two subsystems simultaneously into the S^3. After performing the $n(\sum_{i=1}^{N-1} N-i)$ checks we may find pairs of subsystems which have a common state feedback controller. If we have not found a single pair we can stop and conclude that we need N state feedback controllers. Otherwise we continue with the pairs that we have found and look for triples while carrying out the next $n(\sum_{i=1}^{N-2} N-1-i)$ checks. Continuing this procedure leads finally to the combination with the least number of state feedback controllers.

Example

The framework that was developed in the previous sections is now applied to a simple example in order to demonstrate the design procedure. We would like to control the following uncertain and parameter-variant piecewise linear system, which consists of the following subsystems:

$$\begin{aligned} A_{\Delta,0} &= \begin{bmatrix} 0 & 1 \\ -1.9 \pm 0.2 & -4.6 \pm 0.1 \end{bmatrix} \\ A_{\Delta,1} &= \begin{bmatrix} 0 & 1 \\ -0.4 \pm 0.1 & -4 \pm 0.2 \end{bmatrix} \\ A_{\Delta,2} &= \begin{bmatrix} 0 & 1 \\ -2.6 \pm 0.2 & -5 \pm 0.1 \end{bmatrix} \\ A_{\Delta,3} &= \begin{bmatrix} 0 & 1 \\ 0.3 \pm 0.1 & -4.5 \pm 0.3 \end{bmatrix} \end{aligned}$$

We also have $B = [0,1]^T$ and the state feedback vector is $D = [d_1, d_2]^T$. Definition: $\pm c$ means that an arbitrary and possibly not fixed element c is added, which belongs to the interval $[-c, c]$, i.e. $c \in [-c, c]$. Due to our specifications the performance of the nominal system should be such that it has two poles at -2.5. Hence our nominal system in controller canonical form is

$$A_{nom} = \begin{bmatrix} 0 & 1 \\ -6.25 & -5 \end{bmatrix} \tag{4.6}$$

To compute the stable switched set we will use a common quadratic Lyapunov function for simplicity, i.e.

$$P = \begin{bmatrix} 11 & 1 \\ 1 & 1.5 \end{bmatrix} \tag{4.7}$$

which fulfills $A_{nom}^T P + PA_{nom} < 0$, $P = P^T > 0$. Using the upper bounds for the two signomial programs we get $S = \{a_1 \in [4.7, 7.1], a_2 \in [4.7, 7.1]\}$. Now we can test if there exists a common state feedback controller that satisfies our specifications. We have

$$R = \begin{bmatrix} R_{1,0} = (3.0, 5.0) & R_{1,1} = (4.4, 6.6) & R_{1,2} = (2.3, 4.3) & R_{1,3} = (5.1, 7.3) \\ R_{2,0} = (0.2, 2.4) & R_{2,1} = (0.9, 2.9) & R_{2,2} = (-0.2, 2) & R_{2,3} = (0.5, 2.3) \end{bmatrix}$$

Applying the formalism for checking we see that $Check_1 = \{\}$ and $Check_2 = (0.9, 2)$. Since $Check_1$ is empty there exist no common state feedback controller. Thus we have to look for the intersections of the sets that are not empty and therefore we need to perform $W = 2 \cdot (\sum_{i=1}^{4-1} 4 - i + \sum_{i=1}^{4-2} 4 - 1 - i) = 18$ checks. The intersections that are not empty belong to the following subsystems: (A_0, A_1), (A_0, A_2); (A_1, A_0), (A_1, A_3); (A_2, A_0); (A_3, A_1). In order to find the minimal number of controllers we need to analyze 4 different possibilities, since we have two times the choice of two possible combinations. We find that the minimum number of controllers that are necessary is 2, for the following non empty sets (A_0, A_2), (A_1, A_3). Since there is just one solution that gives us the minimum of 2 controllers we can now proceed to design the 2 state feedback controllers according to our optimal procedure. Thus,

$$G_1 = \begin{bmatrix} \|6.25 - 1.9 - d_1\| & \|6.25 - 2.6 - d_1\| \\ \|5 - 4.6 - d_2\| & \|5 - 5 - d_2\| \end{bmatrix} \tag{4.8}$$

with the following constraint

$$\begin{matrix} d_1 \in R_{1,0} & d_1 \in R_{1,2} \\ d_2 \in R_{2,0} & d_2 \in R_{2,2} \end{matrix} \tag{4.9}$$

For the first state feedback controller we find a control law with $d_1 = 4$ and $d_2 = 0.2$. The second cost function is given as follows:

$$G_2 = \begin{bmatrix} \|6.25 - 0.4 - d_1\| & \|6.25 + 0.3 - d_1\| \\ \|5 - 4.0 - d_2\| & \|5 - 4.5 - d_2\| \end{bmatrix} \tag{4.10}$$

with the following constraint

$$\begin{matrix} d_1 \in R_{1,1} & d_1 \in R_{1,3} \\ d_2 \in R_{2,1} & d_2 \in R_{2,3} \end{matrix} \tag{4.11}$$

For the second state feedback controller we find a control law with $d_1 = 6.2$ and $d_2 = 0.75$.

4.5 Active suspension control

For over a century designers have developed vehicle suspension systems. Most suspension systems are passive and employ some type of spring in combination with hydraulic or pneumatic shock absorbers. Despite the level of sophistication, passive suspensions can only store and dissipate energy in a pre-determined manner. Therefore it will always be a compromise between passenger ride comfort, handling, and suspension stroke.

The aim of active suspension is to improve passenger comfort while enhancing handling qualities and avoiding suspension strokes. Various controllers have been proposed in recent years, including LQR, fuzzy controllers etc. (76), (52). No matter which controller strategy is used each design has to be robust to parameter variations as nonlinear spring characteristics change due to aging and fabrication differences. Also, large variations in vehicle mass have to be taken into account. Therefore one objective of the controller is to be robust to huge parameter variations. In addition, passenger ride comfort should be improved. This is done by reducing the car's vertical body acceleration, usually referred to as "sky-hook" damping. Another objective is to meet the suspension deflection constraints. In order to prevent the travel limits of the suspension being reached the suspension needs to be stiffer near its travel limits. This is done by feeding back the vehicle suspension deflection to avoid reaching the deflection constraints.

We will now show how the suspension dynamics are modelled by an uncertain and parameter-variant piecewise linear system. For our purposes we assume a quarter car model with parallel connected electric servo drive. The schematics of this setup are displayed in figure 4.4. Here,

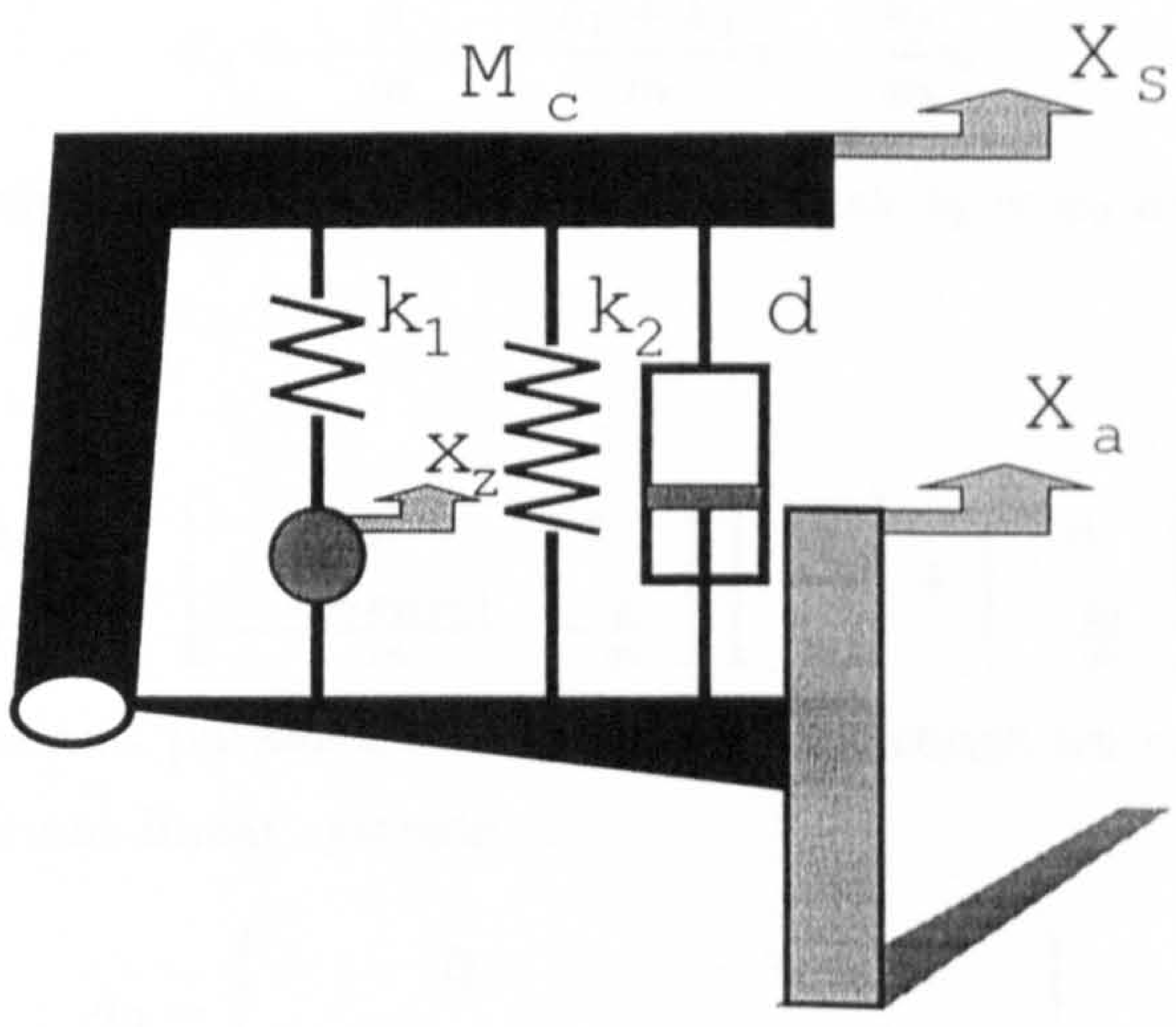


Figure 4.4: Active suspension of a quarter car

$M_c \in [400, 450]$kg is the mass of the quarter car. The spring coefficients are denoted by k_1

and k_2. $k_1 = 25000\frac{\text{N}}{\text{m}}$, while the characteristic of k_2 is given by figure 4.5. The damper

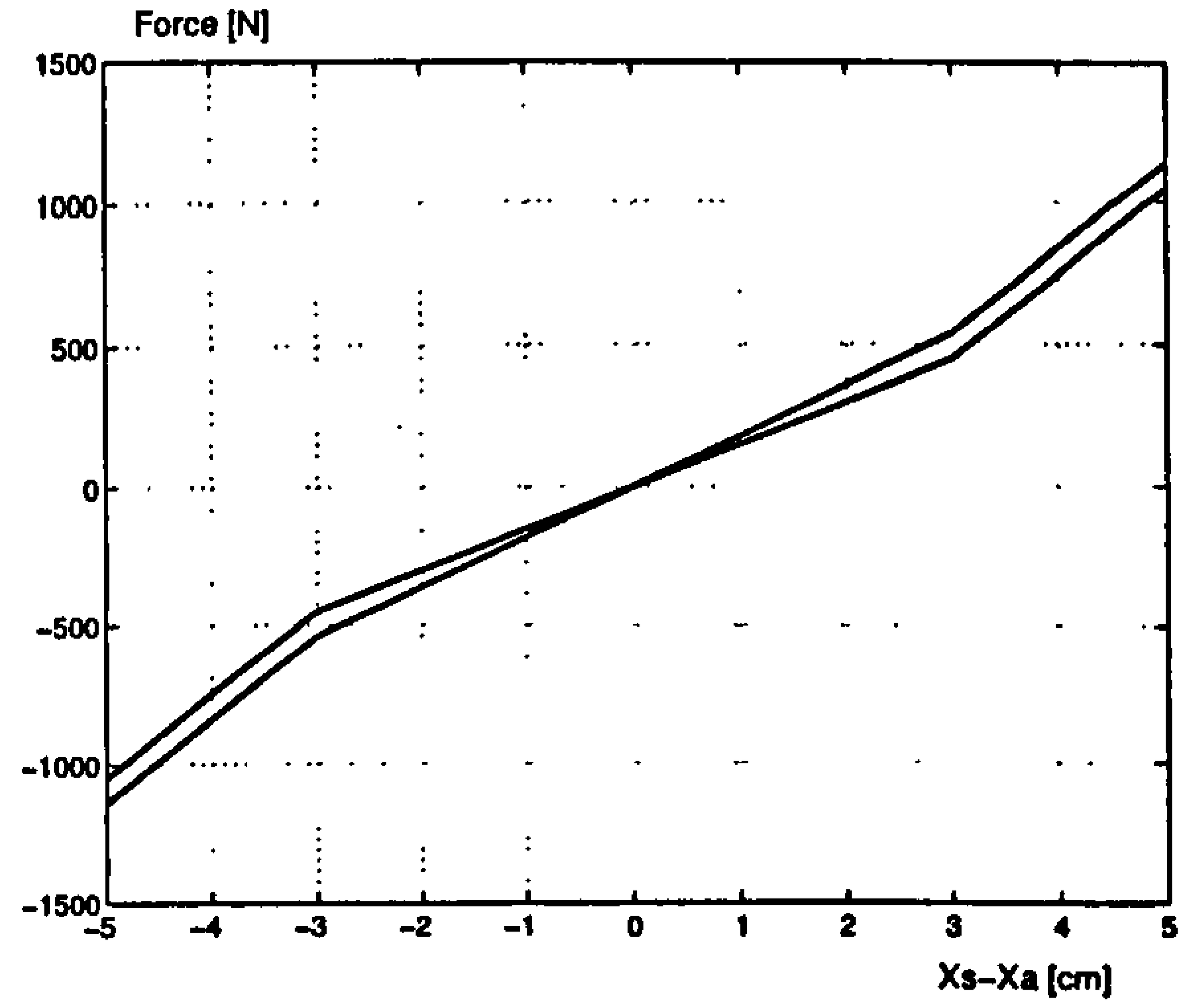


Figure 4.5: Spring coefficient

coefficient is denoted by $d \in [1400, 1500]\frac{\text{Ns}}{\text{m}}$. The suspension travel $x_s - x_a \in [-4\text{cm}, 4\text{cm}]$. The deflection $x_s - x_a$ is measured by a strain gauge on the spring or by a potentiometer at the hinge. To derive the dynamics we write the equation of the forces:

$$\ddot{x}_s = -\frac{d}{m}(\dot{x}_s - \dot{x}_a) - \frac{k_2}{m}(x_s - x_a) - \frac{k_1}{m}(x_s - x_z) \tag{4.12}$$

Using new state variables we take $x_1 = x_s - x_a$, $x_2 = \dot{x}_s - \dot{x}_a$ and the relation $x_z = x_a + u$ where u is the control input (distance change by the servo), we get

$$\ddot{x}_s = -\frac{d}{m}x_2 - \frac{k_1 + k_2}{m}x_1 - \frac{k_1}{m}u \tag{4.13}$$

We can now write the state space equation observing that $\dot{x}_1 = x_2$ and $k_2 = k_2(x_1)$ depend on the deflection

$$\begin{bmatrix} \dot{x}_1 \\ \ddot{x}_s \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k_1+k_2(x_1)}{m} & -\frac{d}{m} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ -\frac{k_1}{m} \end{bmatrix} u \tag{4.14}$$

Taking the uncertainties and parameter variations into account we can derive two different uncertain parameter variant linear systems, i.e.

$$A_0 = \begin{bmatrix} 0 & 1 \\ -98.19 \pm 9.30 & -3.43 \pm 0.32 \end{bmatrix}$$

$$B_0 = \begin{bmatrix} 0 \\ -59.03 \pm 3.47 \end{bmatrix}$$

for $| x_1 |< 3$cm and

$$A_1 = \begin{bmatrix} 0 & 1 \\ -101.94 \pm 13.05 & -3.43 \pm 0.32 \end{bmatrix}$$

$$B_1 = \begin{bmatrix} 0 \\ -59.03 \pm 3.47 \end{bmatrix}$$

for $| x_1 | \geq 3$cm. The notation $\pm c$ means that any arbitrary and not necessarily fixed value can be assumed in the interval $[-c, c]$. The first objective is to keep the vertical acceleration of the car body at a minimum, i.e. $|\dot{x}_s| < \varepsilon$ for some small $\varepsilon > 0$. This means in terms of the closed loop system that the coefficients in the lowest row of $A_{nom,0}$ should be very small. Thus,

$$A_{nom,0} = \begin{bmatrix} 0 & 1 \\ -10 & -2 \end{bmatrix}$$

for $| x_1 |< 2$cm. This softening of the suspension also achieves the second objective, which is to keep good contact between wheels and road, which increases handling performance. The third objective is to prevent the suspension from hitting the travel limits. Therefore the control needs to stiffen up the suspension before travel limits are hit. Thus, we choose

$$A_{nom,1} = \begin{bmatrix} 0 & 1 \\ -200 & -7 \end{bmatrix}$$

for $| x_1 | \geq 2$cm. Computing $P = P^T > 0$, $A^T_{nom}P + PA_{nom} < 0$ we use $A_{nom} = \frac{A_{nom,0}+A_{nom,1}}{2}$ and obtain

$$P = \begin{bmatrix} 4.5432 & 0.0620 \\ 0.0620 & 0.0413 \end{bmatrix} \tag{4.15}$$

Note that the choice of P is not unique. Dependent on P the stable switched set may change. Thus for some P it is easier to fulfill objectives one and two, while for others objective three is easier. Therefore the choice of P is a trade off.

Using the upper bounds of the two signomial programs we get $S = \{a_1 \in [79, 141], a_2 \in [3, 5]\}$. We can already see that there exists a common state feedback which stabilizes the system, since the coefficients $a_{i,1}$ and $a_{i,2}$ for $i \in 1, 2$ of both systems A_0, A_1 are already entirely contained in the S. Computing R as

$$R = \begin{bmatrix} R_{1,0} = (-9.89, 33.51) & R_{1,1} = (-9.89, 26.01) \\ R_{2,0} = (-0.11, 1.25) & R_{2,1} = (-0.11, 1.25) \end{bmatrix}$$

and intersecting, we find that all $Check_i$, $i \in 1, 2$ are non empty. Hence, there exists a common state feedback which robustly stabilizes the dynamics for the given parameter uncertainties and under all parameter variations. However, this is not the only objective. We want to find the state feedback which renders the closed loop system to have similar performance to the desired nominal closed loop dynamics given by $A_{nom,0}$ and $A_{nom,1}$. To find a controller which achieves objectives one and two we minimize

$$G_1 = \begin{bmatrix} \|10 - 98.19 - d_1\| & \|2 - 3.43 - d_2\| \end{bmatrix}^T$$

subject to

$$d_1 \in R_{1,0} \quad d_2 \in R_{2,0}$$

This gives us the state feedback controller $d_1 = -9.89$ and $d_2 = -0.11$. To achieve the third objective we minimize

$$G_2 = \begin{bmatrix} \|200 - 98.19 - d_1\| & \|200 - 101.94 - d_1\| \\ \|7 - 3.43 - d_2\| & \|7 - 3.43 - d_2\| \end{bmatrix}$$

subject to

$$d_1 \in R_{1,0} \quad d_1 \in R_{1,1}$$
$$d_2 \in R_{2,0} \quad d_2 \in R_{2,1}$$

This gives us the state feedback controller $d_1 = 26.01$ and $d_2 = 1.25$. To obtain the final state feedback we need to take into account the gain and the uncertainties of the input matrix. We obtain the state feedback controller by $| d_i |= -\max | b_1 | \cdot \tilde{d}_i$, $i \in 1, 2$. Hence, we get for $| x_1 |< 2$cm, $\tilde{d}_1 = 0.158$ and $\tilde{d}_2 = 0.00176$. For $| x_1 |\geq 2$cm we get $\tilde{d}_1 = -0.4161$ and $\tilde{d}_2 = -0.002$. We have now found the state dependent piecewise linear state feedback gains.

Conclusion

The chapter developed a framework for stability analysis and controller design of uncertain parameter variant piecewise linear systems. The computation of a set of systems, around a nominal system, was proposed, which are robustly stable against switching among members of its set. Such a set of systems was referred to as the stable switched set. It has been shown that using quadratic or piecewise quadratic Lyapunov functions with or without the S-procedure to obtain the stable switched set is a signomial program. Using the upper bounds of signomial programs led to a convenient method to verify that a set of systems belongs to the stable switched set. This gave an iterative procedure to compute larger and larger sets. Since it is desirable to stabilize a piecewise linear system with as few state feedback controllers as possible, we gave the sufficient conditions for the existence of a common state feedback controller to achieve this. For cases where there exists no common state feedback controller we proposed a method to find the minimum number of controllers.

To show the synthesis steps involved, the procedure was applied to a simple uncertain and parameter-variant piecewise linear system. As an example for real systems the control of active suspension was considered. It was shown how the suspension dynamics translate into an uncertain parameter-variant piecewise linear system. Variations in mass and spring as well as damper coefficients made handling of uncertainties and parameter variations inevitable. To improve passenger ride comfort and enhance wheel to surface contact the suspension was softened by the controller for small deviations. To avoid hitting suspension travel limits the control stiffened the suspension for larger deviations. This was achieved by two state dependent linear state feedback controllers which also robustly stabilize the dynamics under all admissible parameter variations.

5 Optimal control of hybrid systems in discrete time

In this chapter it is shown that linear hybrid systems in discrete time result from real-time systems with varying sampling rate. A linear discrete time dynamic is obtained for each sampling rate. The change between sampling rates is decided by a scheduler, which determines the discrete dynamic.
To motivate, an example is given in which a stable continuous system is sampled at two different sampling rates. In this way a hybrid system with two subsystems is obtained. Two controllers are designed minimizing the same continuous quadratic loss function with the same weights. It is shown that the resulting hybrid system can be unstable for some switching sequences, although each individual subsystem is stable. To avoid this problem two solutions are suggested. The first solution shows how restrictions on switching sequences can be imposed such that only stable sequences are chosen. The second solution presents an optimal controller design in which a bound on the cost, for all possible switching sequences, is computed. This results in a piecewise constant state feedback control law and guarantees stability regardless of the switching sequence. The controller synthesis is cast into an LMI, which conveniently solves the synthesis problem. To illustrate the procedure, the introduction example is revisited using the proposed LMI synthesis method and the stable control law is given, which is robustly stable to all switching sequences while limiting the cost.
The contributions of this chapter lie in the instability example and scheduling strategies as well as in the LMI condition for the upper bound on the optimal cost and optimal controller synthesis. This novel example of instability has been published in (66). Solutions by scheduling have been presented in (66), while solutions using the upper bound on the cost for state feedback are published in (66) and (64).

5.1 Introduction

In the previous chapter, analysis and controller design have been investigated under the focus of robustness. In general this is very important since most dynamics incorporate uncertainties. Also, most models approximate reality only up to a curtain accuracy. Therefore controllers need to work for slightly different parameters and setups. Besides the need of robustness a further desire is to have optimality. This is the search for a control law which minimizes a given performance index. In the present chapter we focus on optimal control of linear hybrid systems. In this class of hybrid systems the discrete dynamic, the switching between different continuous dynamics, is not a priori known but can be measured or determined online. From this perspective the class of hybrid systems considered in this chapter is similar to the class considered in the previous chapter but without uncertainties or parameter variations. Further, the continuous dynamics are given in discrete time. Thus, we get a hybrid system as

$$\left[\begin{array}{lcl} x_{k+1} & = & \Phi_q x_k + \Gamma_q u_k \\ y_k & = & C_q x_k \\ q^+ & = & f(\cdot) \end{array}\right. \tag{5.1}$$

where $x \in X \subset \mathbf{R}^n$ is the continuous state in discrete time and $q \in Q = \{1, 2, \ldots, N\} \subset \mathbf{Z}^+$ is the discrete state. q is the current discrete state and q^+ denotes its successor. Φ_q, Γ_q, C_q denote the system, input and output matrices of the discrete time system respectively, which depend on the current discrete state q. $f(\cdot)$ denotes the discrete dynamics, which is decided by a scheduler.

Optimal control of linear hybrid systems is of general interest. In this chapter, however, we pick a particular example of linear hybrid system with continuous dynamic described in discrete time. The example that we take to motivate stems from the area of embedded systems which is of broad interest. In particular we are concerned with the control of sampled data systems which have variations in sampling rate. It is easy to see that such dynamics can be viewed as hybrid systems in discrete time. Each subsystem is given as the discretization of a continuous dynamic at the current sampling rate. The sampling rate, i.e. the discrete dynamic, is decided by a scheduler, which decides whether the system is sampled fast or slow. Sampled data systems with varying sampling rate arise for different reasons. One of them is the optimal usage of central processing unit (CPU) resources (28), (25). Roughly speaking, several tasks are carried out on the same CPU, one of them is to compute the control law. When enough computational resourses are available, the control law is computed more frequently than when the resourses are used for other computations. This leads to variations in sampling rate. Previously, variations in sampling rate were often neglected. In other cases it was assumed that designing a piecewise continuous controller consisting of controllers which are optimal for the current sampling rate would lead to reasonable results. This chapter shows that such assumptions are not justified. Further it is shown that such a control strategy does not guarantee stability.

In the following we will give an example of how variations in sampling time can lead to instability. We proceed by proposing a controller design which results in a piecewise linear state feedback control law and is robustly stable to variations among the prescribed sampling rates. We show how such state feedback controllers can be found using linear matrix inequalities (LMI). We illustrate the design procedure by revisiting the introductory example, where a linear quadratic design approach leads to instability. In the next chapter it is shown that solution concepts derived here, for hybrid systems in discrete time, translate quite naturally into solution concepts for hybrid systems described in continuous time.

5.1.1 Example 1: Two different sampling times, same continuous loss function in both

As an example of instability by scheduling, the real-time control of the following linear continuous system

$$\begin{aligned} \dot{x} &= Ax + bu \\ y &= Cx \end{aligned} \tag{5.2}$$

is considered, where

$$A = \begin{bmatrix} 0 & 1 \\ -10000 & -0.1 \end{bmatrix} \quad b = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad C = [1 \quad 0] \tag{5.3}$$

are the system, input and output matrices. The continuous system is stable with poles in the left hand-side of the complex plane, $p_{1,2} = -0.05 \pm 100i$. In the following, a hybrid system consisting of two discrete time systems is derived from this continuous system. The continuous system is discretized with two different zero order hold circuits, where the sampling rates are $h_1 = 0.002s$, $h_2 = 0.094s$ respectively. The two discretizations, i.e. subsystems, are represented by

$$\begin{aligned} x_{k+1} &= \Phi_q x_k + \Gamma_q u_k \\ y_k &= C_q x_k \end{aligned} \tag{5.4}$$

$$q \in \{1, 2\}$$

where $\Phi_q = e^{Ah_q}$, $\Gamma_q = \int_0^{h_q} e^{As} B ds$ and q denotes the discretized system obtained with sampling time h_q. Both discretizations lead to stable discrete systems with the spectral radius $\rho(\Phi_1) < 1$, $\rho(\Phi_2) < 1$ respectively, where $\rho(\Phi_q)$ gives the largest eigenvalue of Φ_q. It should be clear now, that sampled data system with varying sampling time can be represented as a hybrid system in discrete time. Therefore, and for the sake of compactness, is is understood that each time we refer to "discretization" we also mean "subsystem" and with "variations in sampling rate" we also mean "switching between subsystems" and vice versa.

A discrete linear quadratic optimal controller is designed for both discretizations, minimizing the continuous loss function

$$J = \int_0^\infty (x(t)^T Q_c x(t) + u(t)^T R u(t))dt \tag{5.5}$$

subject to system (5.3) sampled at h_1, h_2, where

$$Q_c = \begin{bmatrix} 20000 & 0 \\ 0 & 20000 \end{bmatrix} \quad R = 50$$

The resulting gain matrices are found by discretizing the loss function (5.5)

$$\begin{aligned} Q_{1,q} &= \int_{kh_q}^{kh_q+h_q} (\Phi_q^T(s,kh_q)Q_c\Phi_q(s,kh_q))ds \\ Q_{12,q} &= \int_{kh_q}^{kh_q+h_q} (\Phi_q^T(s,kh_q)Q_c\Gamma_q(s,kh_q))ds \\ Q_{2,q} &= \int_{kh_q}^{kh_q+h_q} (\Gamma_q^T(s,kh_q)Q_c\Gamma_q(s,kh_q)+R)ds \end{aligned}$$

and solving the discrete algebraic Riccati equation.

$$P_q = \Phi_q^T P_q \Phi_q + Q_{1,q} - (\Phi_q^T P_q \Gamma_q + Q_{12,q})(\Gamma_q^T S(k+1)\Gamma_q + Q_{2,q})^{-1}(\Gamma_q^T P_q \Phi_q + Q_{12,q}^T)$$

The state feedback law $u = -K_q \cdot x$ is then given by

$$K_q = (Q_{2,q} + \Gamma_q^T P_q \Gamma_q)^{-1}(\Gamma_q^T P_q \Phi_q + Q_{12,q}^T)$$

such that we get

$$K_1 = \begin{bmatrix} -195.401 \\ 19.4121 \end{bmatrix} \quad K_2 = \begin{bmatrix} -1296.6 \\ -8.826 \end{bmatrix}$$

For both discretizations the controlled closed loop systems is stable, i.e $\rho(\Phi_1 - \Gamma_1 K_1) < 1$, $\rho(\Phi_2 - \Gamma_2 K_2) < 1$ respectively. However in the case where the system is sampled with h_1 for 1 sampling interval and then the system is sampled with h_2 for 2 sampling intervals repeatedly we find that this sequence is unstable. This can be seen by looking at figure 5.1 or at the spectral radius of the resulting system $\rho((\Phi_2 - \Gamma_2 K_2)^2(\Phi_1 - \Gamma_1 K_1)^1) > 1$. We obtain the spectral radius of the resulting system by writing the solution for sampling at h_1 once, $x_{h_1} = (\Phi_1 + \Gamma_1 K_1)x_0$ and sampling at h_2 twice, $x_{2h_2+h_1} = (\Phi_2 + \Gamma_2 K_2)^2 x_{h_1}$. We can now substitute into each other and obtain $x_{h_2+3h_1} = (\Phi_2 + \Gamma_2 K_2)^2(\Phi_1 + \Gamma_1 K_1)x_0$. Since this is done repeatedly we can think of it as the new system description and take the spectral radius

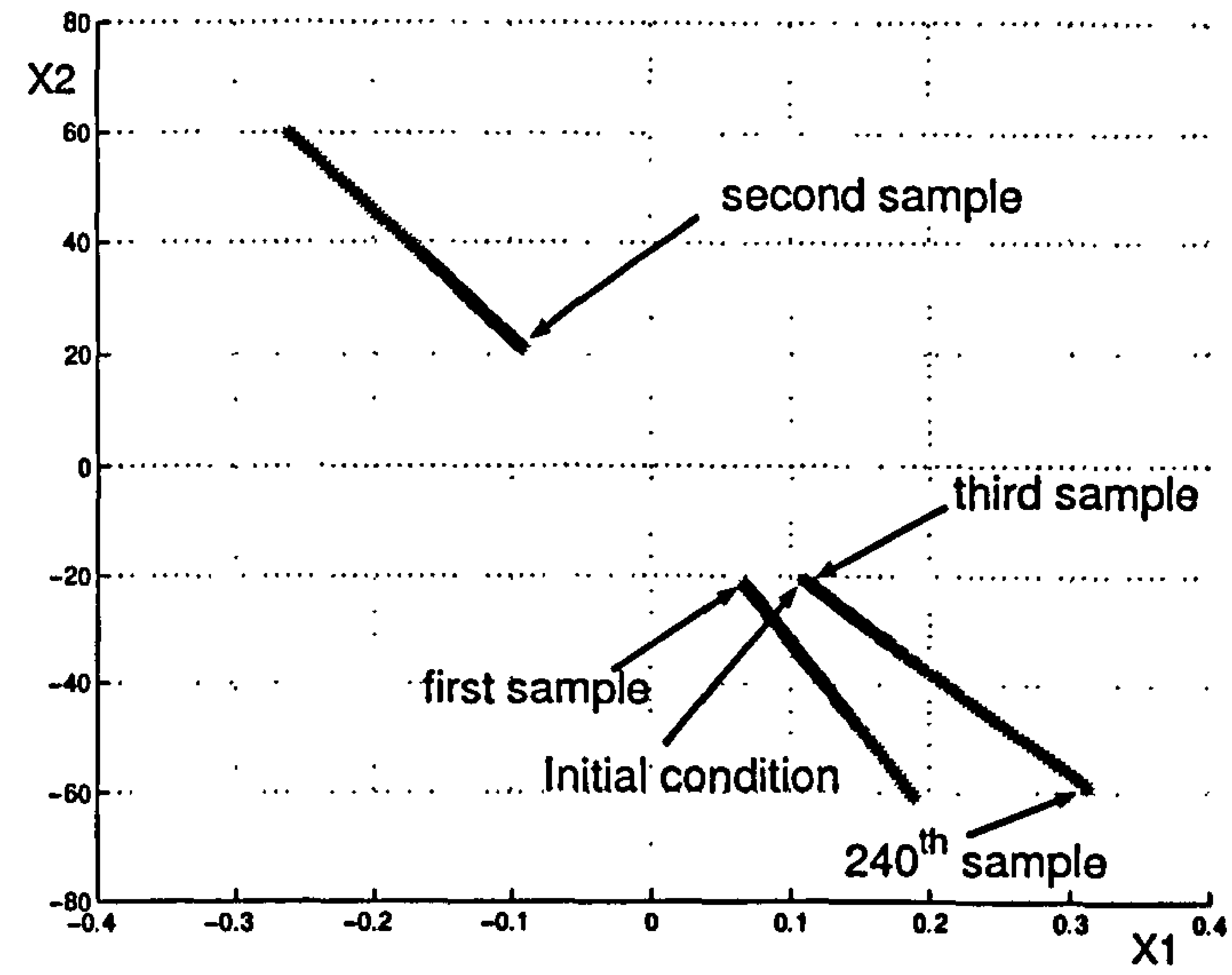


Figure 5.1: Unstable sequence

of it, in this case it is larger than one, hence the resulting system is unstable. The figure 5.1 shows the samples (the discrete points of the continuous trajectory) for the unstable system in the phase-plane. The system 5.3 is sampled for one sampling interval with h_1, i.e. small distance between initial and first sample, and twice with h_2, i.e. larger distance between first, second and third sample. It can be seen that the trajectory enlarges and it gets further away from the origin, i.e. sample 240 is much further away from the origin than the initial sample. It turns out that this is not the only sequence which destabilizes the system. Table 5.1 shows further sequences for which the resulting system is unstable. Figure 5.2 shows another unsta-

	$\rho((\Phi_2 - \Gamma_2 K_2)^{mh_2}(\Phi_1 - \Gamma_1 K_1)^{nh_1}) > 1$				
$n \cdot h_1$	$1 \cdot h_1$	$1 \cdot h_1$	$2 \cdot h_1$	$2 \cdot h_1$	$2 \cdot h_1$
$m \cdot h_2$	$2 \cdot h_2$	$3 \cdot h_2$	$4 \cdot h_2$	$5 \cdot h_2$	$6 \cdot h_2$

Table 5.1: Unstable sequences

ble sequence. The system is sampled once at h_1 and three times at h_2. It is now interesting to check which scheduling sequence lead to unstable scheduled systems. Fortunately the number of possibilities as shown in table 5.1 are limited.

Theorem 1: Given two exponentially stable discrete time closed loop systems, the number of possibly unstable scheduling sequences using repeatedly $(\Phi_2 - \Gamma_2 K_2)^i(\Phi_1 - \Gamma_1 K_1)^l$ are limited

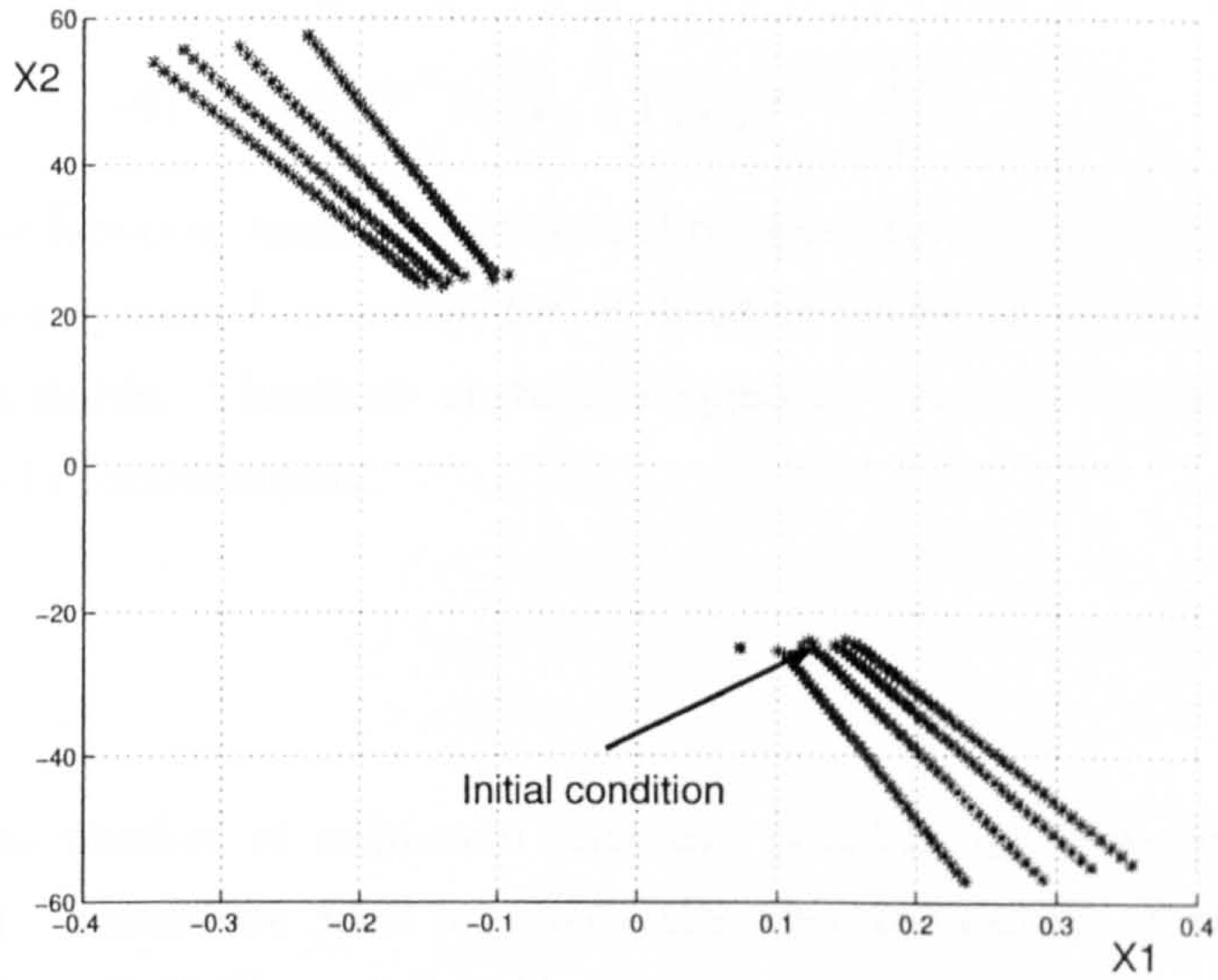


Figure 5.2: Unstable sequence

by $p = (m-1)\cdot(n-1)$, where m and n are sufficiently large positive integers satisfying

$$
\begin{aligned}
(\Phi_1 + \Gamma_1 K_1)^{n^T} P_1 (\Phi_1 + \Gamma_1 K_1)^n - \frac{1}{a} P_2 &< 0 \\
(\Phi_2 + \Gamma_2 K_2)^{m^T} P_1 (\Phi_2 + \Gamma_2 K_2)^m - \frac{1}{a} P_2 &< 0
\end{aligned}
$$

and P_1, $P_2 > 0$ are the associated Lyapunov functions, $a \in \mathbf{R}^+$ being a positive scalar such that $aP_1 > P_2$.

Proof: To show that a sequence is stable it is sufficient to find a Lyapunov function. Since each discrete closed loop system is exponentially stable there exists a Lyapunov function for each system,

$$(\Phi_q + \Gamma_q K_q) P_q (\Phi_q + \Gamma_q K_q) - P_q < 0 \quad P_q = P_q^T > 0 \tag{5.6}$$

$$q \in \{1, 2\}$$

For P_1, $P_2 > 0$, there exists a scalar $a \in \mathbf{R}^+$ such that $aP_1 > P_2$. We can take $x^T a P_1 x$, $x^T P_2 x > 0$ as a piecewise quadratic Lyapunov function. Hence the Lyapunov function decreases while switching from system 1 to system 2 and while staying in each system. However switching from system 2 to 1 there is an energy increase, therefore we need to stay either sufficiently long with system 1

$$(\Phi_1 + \Gamma_1 K_1)^{n^T} P_1 (\Phi_1 + \Gamma_1 K_1)^n - \frac{1}{a} P_2 < 0$$

or with system 2 before switching to system 1

$$(\Phi_2 + \Gamma_2 K_2)^{m^T} P_1 (\Phi_2 + \Gamma_2 K_2)^m - \frac{1}{a} P_2 \quad < \quad 0$$

Then the Lyapunov function decreases between three executive sampling rate changes. Hence all sequences where system 1 is active for at least n cycles or system 2 is active for at least m cycles are stable. Therefore unstable sequences can only consist of the remaining $p = (m-1) \cdot (n-1)$ combinations.

□

By Theorem 1 the number of sequences that are possibly unstable are limited by $p = (m-1) \cdot (n-1)$. Hence we need to check the spectral radius of the p combinations $\rho((\Phi_2 - \Gamma_2 K_2)^i (\Phi_1 - \Gamma_1 K_1)^l) > 1$ $i \in \{1, \ldots, m-1\}$, $l \in \{1, \ldots, n-1\}$ to find all sequences which are unstable.

We have seen that even when two stable discrete systems are obtained from a stable continuous system with state feedbacks minimizing the same continuous loss function, that variation in sampling rate (switching between these two discrete systems) can lead to instability.
In the next sections we will propose scheduling strategies and controller synthesis procedures which will overcome this problem. We will be first concerned to find sensible scheduling strategies which avoid using unstable sequences.

5.2 Stable scheduling strategies

In many cases a hybrid system can not only be controlled by continuous control. It is often also possible to control the discrete dynamics. If this is the case one might choose a performance index which penalizes continuous as well as discrete dynamics. In particular discrete mode changes need to be penalized to avoid Zeno executions. Dealing with hybrid systems where the continuous dynamics are described in discrete time this is of course not an issue.
Unfortunately our application does not allow choice of the discrete dynamic freely since we need to take into account that computational resource changes are anticipated, which require reduction in the sampling rate immediately, i.e. we need to be able to switch from fast to slow sampling at any sampling time. The opposite is of course not required, i.e. we can stay as long as we want with the slower sampling rate. Even if the scheduler decides to sample faster, information can be neglected, such that effectively the system is still sampled slow.

However, it is desirable for better performance to switch back to faster sampling as soon as possible. This fact is exploited for opposing sensible restrictions on the scheduling strategy. We proceeded in computing a minimum time for sampling slow, after switching from fast to slow sampling, before we can sample fast again if computational resources allow it. We show that if such a scheduling strategy is applied the scheduled system is stable.

Theorem 1 implies that if we restrict all possible scheduling sequences such that when slow sampling is chosen we sample slow for at least a minimum time, all sequences are stable. In general a number of different sampling rates is specified such that we can compute a minimum time for each. Let P_1 be associated with h_1 which is the fastest sampling time. Then the minimum sampling times are computed as follows. Pick an $a \in \mathbf{R}^+$ such that $aP_1 \geq P_q$ for all $q \in \{1, 2, \dots, N\}$ then solve iteratively for each m_q which satisfies

$$(\Phi_q + \Gamma_q K_q)^{m_q^T} P_1 (\Phi_q + \Gamma_q K_q)^{m_q} - \frac{1}{a} P_q < 0 \tag{5.7}$$

Hence the minimum times for each sampling rate are given by $m_q h_q$.

In some cases it is not possible to impose restrictions on the scheduling strategies. For these cases the controller design has to be changed such that it is robustly stable against variations in sampling rate. In the next section an optimal controller design is stated which minimizes the loss function over one sampling time and has a terminal penalty, which is greater or equal than the cost of bringing the states to the origin for the worst case variations in sampling rate. We show that this design is stable for all variations in sampling rate and results in a piecewise constant state feedback controller.

5.3 Controller design

For cases where restrictions on sampling rate variations are not desirable a controller has to be found which is robustly stable to variations in sampling rate. One way to achieve this is to find a controller which renders the closed loop systems to have a common Lyapunov function. A state feedback controller can then be constructed using the necessary and sufficient condition for quadratic Lyapunov function for discrete systems:

$$(\Phi_q+\Gamma_q K_q)^T P(\Phi_q+\Gamma_q K_q)-P < 0 \iff (\Phi_q+\Gamma_q K_q)Q(\Phi_q+\Gamma_q K_q)^T-Q < 0 \quad Q = P^{-1} \tag{5.8}$$

with $P = P^T > 0$. Extending the inequality with $I = Q * Q^{-1}$ we get

$$(\Phi_q + \Gamma_q K_q)QQ^{-1}Q(\Phi_q + \Gamma_q K_q)^T - Q < 0 \tag{5.9}$$

Applying Schur's complement to the above expression we obtain

$$\begin{bmatrix} Q & (\Phi_q + \Gamma_q K_q)Q \\ Q(\Phi_q + \Gamma_q K_q)^T & Q \end{bmatrix} > 0 \quad Q > 0 \tag{5.10}$$

Substituting $W_q = K_q Q$ we obtain

$$\begin{bmatrix} Q & \Phi_q Q + \Gamma_q W_q \\ Q\Phi_q^T + W_q^T \Gamma_q^T & Q \end{bmatrix} > 0 \quad Q > 0 \tag{5.11}$$

Note that the above inequality is an LMI with the variables Q, W_q. If the above LMI is feasible we will find our state feedback law $u_k = K_q x_k$ with $K_q = W_q Q^{-1}$ so that we can write our closed loop system

$$x_{k+1} = (\Phi_q + \Gamma_q W_q Q^{-1}) * x_k \tag{5.12}$$

A different way to write (5.8) is

$$[I, 0]\, M \begin{bmatrix} I \\ 0 \end{bmatrix} - [\Phi_q, \Gamma_q]\, M \begin{bmatrix} \Phi_q^T \\ \Gamma_q^T \end{bmatrix} > 0 \quad M > 0 \tag{5.13}$$

where

$$M = \begin{bmatrix} P & PK^T \\ KP & KPK^T \end{bmatrix} \tag{5.14}$$

In this way we can find a state feedback controller which robustly stabilizes the system for the specified variations in sampling rate. However, we do not only seek to stabilize the hybrid system for all possible switching strategies, we further want to minimize the quadratic performance index. The introductory example showed that this might cause problems. To overcome this problem we suggest that instead of minimizing a continuous objective function over the infinite horizon, we minimize only over one sampling period. To compensate for the remaining cost we add a terminal penalty. Minimizing over only one sampling period is more sensible since the sampling rate may change after one sampling period anyway, i.e. after a sampling interval a different subsystem of the hybrid system can be chosen. Since the terminal penalty has to be at least as big as the remaining worst case cost we write the following inequality

$$x(kh)^T P x(kh) \geq \min_u \int_{kh}^{kh+h} (x^T Q_c x + u^T R u)dt + x(kh+h)^T P x(kh+h) \tag{5.15}$$

$$\forall \quad h \in H = \{h_1, h_2, \ldots, h_N\}$$

The solution gives an optimal, piecewise constant state feedback controller for the hybrid system, which is stable regardless of the scheduling.

The first step in solving (5.15) is to discretize the objective function. This is done similarly as in (74). The discretized objective function over one sampling interval with terminal penalty is

$$x(kh)^T P x(kh) \geq \min_u \sum_{i=kh}^{kh} (x(i)^T Q_{1,q} x(i) + 2x(i)^T Q_{12,q} u(i) + u(i)^T Q_{2,q} u(i)) + \\ + x(kh+h)^T P x(kh+h) \quad (5.16)$$

$$\forall \quad q \in \{1, 2, \dots, N\}$$

where

$$Q_{1,q} = \int_{kh_q}^{kh_q+h_q} (\Phi_q^T(s, kh_q) Q_c \Phi_q(s, kh_q)) ds \quad (5.17)$$

$$Q_{12,q} = \int_{kh_q}^{kh_q+h_q} (\Phi_q^T(s, kh_q) Q_c \Gamma_q(s, kh_q)) ds \quad (5.18)$$

$$Q_{2,q} = \int_{kh_q}^{kh_q+h_q} (\Gamma_q^T(s, kh_q) Q_c \Gamma_q(s, kh_q) + R) ds \quad (5.19)$$

and $\Phi_q = e^{Ah_q}$, $\Gamma_q = \int_0^{h_q} e^{As} B ds$ is the fundamental, or the input matrix, of a subsystem.

Theorem 2: If there exists $P = P^T > 0$, K_q, $q \in \{1, 2, \dots, N\}$ such that

$$(\Phi_q + \Gamma_q K_q)^T P (\Phi_q + \Gamma_q K_q) - P + (Q_{1,q} + 2Q_{12,q} K_q + K_q^T Q_{2,q} K_q) \leq 0 \quad (5.20)$$

$$\forall \quad q \in \{1, 2, \dots, N\}$$

then the hybrid system is stable for all scheduling (i.e. switching) strategies among its subsystems and its performance is bounded by $x^T P x$.

Proof: Rearranging (5.16) and taking $x_{k+1} = \Phi_q x_k + \Gamma_q u_k$, $u_k = K_q x_k$ we obtain

$$(\Phi_q + \Gamma_q K_q)^T P (\Phi_q + \Gamma_q K_q) - P + (Q_{1,q} + 2Q_{12,q} K_q + K_q^T Q_{2,q} K_q) \leq 0. \quad (5.21)$$

$V(x) = x^T P x$ serves as a Lyapunov function since $P = P^T > 0$ and

$$\begin{aligned} \Delta V(x) &= (\Phi_q + \Gamma_q K_q)^T P (\Phi_q + \Gamma_q K_q) - P \\ &\leq [I K_q] Q_q [I K_q]^T \end{aligned} \quad (5.22)$$

with

$$Q_q = \begin{bmatrix} Q_{1,q} & Q_{12,q} \\ Q_{12,q}^T & Q_{2,q} \end{bmatrix} \quad \forall \quad q \in Q = \{1, 2, \dots, N\}$$

where Q_q are positive definite for all q and $[IK_q]$ are full rank, therefore $-[IK_q]Q_q[IK_q]$ is negative definite and hence $\Delta V(x) \leq 0$. Hence the hybrid system is stable for all possible scheduling strategies. To show that the performance is bounded by $x^T Px$ we use the fact that $x(kh)^T Px(kh)$ is expressed in terms of $x(kh + h_q)^T Px(kh + h_q)$. Hence

$$x(kh)^T Px(kh) \geq \min_u \sum_{i=kh}^{kh} (x(i)^T Q_{1,q} x(i) + 2x(i)^T Q_{12,q} u(i) + u(i)^T Q_{2,q} u(i)) + \\ + x(kh + h)^T Px(kh + h)$$

and $x(kh + h)^T Px(kh + h)$ can be expressed in terms of $x(kh + h + h)^T Px(kh + h + h)$ as

$$x(kh + h)^T Px(kh + h) \geq \min_u \sum_{i=kh+h}^{kh+h} (x(i)^T Q_{1,q} x(i) + 2x(i)^T Q_{12,q} u(i) + u(i)^T Q_{2,q} u(i)) + \\ + x(kh + h + h)^T Px(kh + h + h)$$

etc.. Substituting recursively we obtain

$$x(kh)^T Px(kh) \quad \geq \quad \min_u \sum_{i=kh}^{\infty} (x(i)^T Q_{1,q} x(i) \; + \; 2x(i)^T Q_{12,q} u(i) \; + \; u(i)^T Q_{2,q} u(i))$$

Hence the performance is bounded by $x^T Px$.

□

We have seen that if we manage to find a controller which satisfies (5.15) and therefore also (5.20), then we can guarantee that the controlled closed loop system is stable for all variations among h_q, $q \in \{1, 2, \dots, N\}$. We will now show how we can formulate the controller synthesis into an LMI, such that we obtain P and K_q.

5.4 Controller synthesis using LMI

We have seen that a system in form (7.2) with its discretizations (7.4) are robustly stable for variations among the prescribed sampling rates h_q, $\forall$ $q \in \{1, 2, \dots, N\}$ and its cost is bounded by $P = P^T > 0$, when we find the state feedback gains K_q, $q \in \{1, 2, \dots, N\}$ which

satisfy (7.15). The remaining problem is to obtain P and K_q.
One possibility is to solve the corresponding Riccati inequalities

$$P \geq \Phi_q^T P \Phi_q + Q_{1,q} - (\Phi_q^T P \Gamma_q + Q_{12,q})(\Gamma_q^T P \Gamma_q + Q_{2,q})^{-1}(\Gamma_q^T P \Phi_q + Q_{12,q}^T) \tag{5.23}$$

$$\forall \quad q \in \{1, 2, \ldots, N\}$$

With the solution P the state feedback gains

$$K_q = (Q_{2,q} + \Gamma_q^T P \Gamma_q)^{-1}(\Gamma_q^T P \Phi_q + Q_{12,q}^T) \tag{5.24}$$

can be obtained. Unfortunately, this has the drawback that (5.23) is not convex. Hence a solution can not be obtained conveniently by solving an LMI. However, we can make use of the following fact

$$(\Phi_q^T P \Gamma_q + Q_{12,q})X + X^T(\Gamma_q^T P \Phi_q + Q_{12,q}^T) + X^T(\Gamma_q^T P \Gamma_q + Q_{2,q})X \geq$$
$$- (\Phi_q^T P \Gamma_q + Q_{12,q})(\Gamma_q^T P \Gamma_q + Q_{2,q})^{-1}(\Gamma_q^T Q_q \Phi_q + Q_{12,q}^T) \tag{5.25}$$

and write (5.23) as a bilinear matrix inequality (BMI)

$$P \geq \Phi_q^T P \Phi_q + Q_{1,q} + (\Phi_q^T P \Gamma_q + Q_{12,q})X + X^T(\Gamma_q^T P \Phi_q + Q_{12,q}^T) + X^T(\Gamma_q^T P \Gamma_q + Q_{2,q})X \tag{5.26}$$

in X and P. The BMI formulation still has the drawback of a non-convex optimization problem.
One way to obtain a convex problem is to use (5.16), and setting $u(kh) \equiv 0 \ \forall \ k$ the equation reduces to

$$x(kh)^T P x(kh) \geq x^T(kh) Q_{1,q} x(kh) + x(kh + h_q)^T P x(kh + h_q) \tag{5.27}$$

$$\forall \quad q \in \{1, 2, \ldots, N\}$$

Since $u(kh) \equiv 0 \ \forall \ k$, $x_{k+1} = \Phi_q x_k + \Gamma_q u_k$ reduces to $x_{k+1} = \Phi_q x_k$ which is substituted into (5.27) to get

$$\Phi_q^T P \Phi_q + Q_{1,q} - P \leq 0 \tag{5.28}$$

$$\forall \quad q \in \{1, 2, \ldots, N\}$$

so that we obtained an LMI in $P = P^T > 0$. The solution to this LMI gives P, which together with (5.24) gives the piecewise constant state feedback law. Thus the controlled closed loop system is robustly stable to arbitrary variation among the prescribed sampling times. This solution has the disadvantage that it is highly restrictive and resorting to suboptimal solutions is also not desirable. One reason why this LMI is restrictive is that the LMI is only feasible for stable open loop dynamics. This is the consequence of the fact that a necessary condition for the LMI to be feasible is $\Phi_q^T P \Phi_q - P \leq 0$. In order to overcome this problem we show how P and the K_q are obtained by solving LMIs which do not require open loop stability. We take (5.20)

$$(\Phi_q + \Gamma_q K_q)^T P(\Phi_q + \Gamma_q K_q) - P + (Q_{1,q} + 2Q_{12,q}K_q + K_q^T Q_{2,q} K_q) \leq 0$$

which we can write as

$$\begin{bmatrix} \Phi_q + \Gamma_q K_q \\ I \\ K_q \end{bmatrix}^T \times \begin{bmatrix} P & 0 & 0 \\ 0 & Q_{1,q} & Q_{12,q} \\ 0 & Q_{12,q}^T & Q_{2,q} \end{bmatrix} \times \begin{bmatrix} \Phi_q + \Gamma_q K_q \\ I \\ K_q \end{bmatrix} - P \leq 0 \quad (5.29)$$

$$\forall \quad q \in \{1, 2, \ldots, N\}$$

Applying Schur's complement to the above expression we obtain

$$\begin{bmatrix} P & (\Phi_q + \Gamma_q K_q)^T & \begin{bmatrix} I & K_q^T \end{bmatrix} \\ (\Phi_q + \Gamma_q K_q) & P^{-1} & 0 \\ \begin{bmatrix} I \\ K_q \end{bmatrix} & 0 & Q_q^{-1} \end{bmatrix} \geq 0$$

$$\forall \quad q \in \{1, 2, \ldots, N\}$$

where

$$Q_q = \begin{bmatrix} Q_{1,q} & Q_{12,q} \\ Q_{12,q}^T & Q_{2,q} \end{bmatrix}$$

Multiplying the above inequality from left and right with

$$\begin{bmatrix} P^{-1} & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix}$$

and setting $W_0 = P^{-1}$, $W_q = K_q P^{-1}$ we obtain the controller synthesis LMIs

$$\begin{bmatrix} W_0 & (\Phi_q W_0 + \Gamma_q W_q)^T & [W_0 \quad W_q^T] \\ \Phi_q W_0 + \Gamma_q W_q & W_0 & 0 \\ \begin{bmatrix} W_0 \\ W_q \end{bmatrix} & 0 & Q_q^{-1} \end{bmatrix} \geq 0 \tag{5.30}$$

$$\forall \quad q \in \{1, 2, \ldots, N\}$$

in $W_0 = W_0^T > 0$ and W_q. The solution of the LMI (5.30) gives the state feedback gains $K_q = W_q W_0^{-1} \; \forall \quad q \in \{1, 2, \ldots, N\}$. Applying the state feedbacks gives a stable closed loop system which is robust against variations among the sampling times $h_q \; \forall \quad q \in \{1, 2, \ldots, N\}$. However, we would not only like to stabilize the system we would further like to minimize the cost for driving the states to the origin for a given objective function (5.15). Therefore we would like to minimize the trace of W_0^{-1}. Unfortunately this is a non-convex optimization problem. Instead of minimizing $Trace(W_0^{-1})$ we minimize

$$\log \det W_0^{-1}$$

subject to (5.30) which is a convex optimization problem.

In this section it was shown how the state feedback synthesis problem which robustly stabilizes a hybrid system for all possible scheduling strategies, while minimizing a quadratic performance index, is cast into an LMI. The solution of the LMI also provides the performance bound for the worst scheduling strategy. It is now interesting to see which performance could be achieved for the best scheduling strategy. Therefore we proceed by computing the lower bound on the optimal cost for driving the states to zero for the best case scheduling strategy.

5.4.1 Computations of lower bounds on the optimal cost

In the previous section we showed how an upper bound on the optimal cost, for driving the states to zero under the worst case scheduling strategy, given initial conditions, is obtained. The interpretation was that no matter which scheduling (switching) strategy is chosen for the hybrid system the performance is better than the computed upper bound. In this section we answer the opposite question: compute the performance bound which is less than the performance given by the best case scheduling strategy. This is the computation of the lower bound of the optimal cost. Note, we compute the lower bound not for controller synthesis, since it does not lead to a stable control strategy, we compute the lower bound for comparison with the upper bound. In this way we can asses how far away we are from optimality. The computation of the lower bound should not be mistaken with computations of the dual of the upper bound, which is a different problem. We take the inverse of the inequality

$$x(kh)^T Px(kh) \leq \min_u \int_{kh}^{kh+h} (x^T Q_c x + u^T Ru)dt + x(kh+h)^T Px(kh+h) \qquad (5.31)$$

$$\forall \quad h \in H = \{h_1, h_2, \dots, h_N\}$$

and sample the objective function such that we obtain

$$x(kh)^T Px(kh) \leq \min_u \sum_{i=kh}^{kh} (x(i)^T Q_{1,q} x(i) + 2x(i)^T Q_{12,q} u(i) + u(i)^T Q_{2,q} u(i)) + \\ + x(kh+h)^T Px(kh+h) \quad (5.32)$$

$$\forall \quad q \in \{1, 2, \dots, N\}$$

where $Q_{1,q}$, $Q_{12,q}$ and $Q_{2,q}$ are obtained by (5.17)-(5.19) as before. Taking (5.32) and replacing $x(kh+h = \Phi_q x(kh) + \Gamma_q u(kh)$ we obtain

$$(\Phi_q x(kh) + \Gamma_q u(kh))^T P(\Phi_q x(kh) + \Gamma_q u(kh)) - x(kh)^T Px(kh) + \\ + (x(kh)^T Q_{1,q} x(kh) + 2x^T Q_{12,q} u(kh) + u(kh)^T Q_{2,q} u(kh)) \geq 0$$

The quadratic from can be written

$$\begin{bmatrix} x(kh)^T \\ u(kh)^T \end{bmatrix}^T \begin{bmatrix} \Phi_q^T P \Phi_q - P + Q_{1,q} & Q_{12,q} + P\Gamma_q \\ Q_{12,q} + P\Gamma_q & \Gamma_q^T P \Gamma_q + Q_{2,q} \end{bmatrix} \begin{bmatrix} x(kh)^T \\ u(kh)^T \end{bmatrix} \geq 0$$

such that we obtain the LMI

$$\begin{bmatrix} \Phi_q^T P \Phi_q - P + Q_{1,q} & Q_{12,q} + P\Gamma_q \\ Q_{12,q} + P\Gamma_q & \Gamma_q^T P \Gamma_q + Q_{2,q} \end{bmatrix} \geq 0$$

$$\forall \quad q \in \{1, 2, \dots, N\}$$

in $P = P^T > 0$. We have seen how a lower bound on the optimal cost for the best case scheduling strategy is computed by solving a convex optimization problem. In the following we will revisit the introductory example and compute the optimal gains which are robust against scheduling.

5.5 Example

We will now demonstrate the synthesis procedure by controlling system (5.2) again. However, since we use the synthesis procedure above we will be certain that the controlled closed loop system is stable and robust against variations among all h_q. We sample the system again with the same sampling rates $h_1 = 0.002s$, $h_2 = 0.094s$. Using $\Phi_q = e^{Ah_q}$, $\Gamma_q = \int_0^{h_q} e^{As} B ds$ $\forall \quad q \in \{1, 2\}$ we obtain

$$\begin{aligned} x_{k+1} &= \Phi_q x_k + \Gamma_q u_k \\ y_k &= C_q x_k \end{aligned}$$

$$q \in \{1, 2\}$$

where

$$\Phi_1 = \begin{bmatrix} 0.9801 & 0.0020 \\ -19.8649 & 0.979 \end{bmatrix} \quad \Gamma_1 = \begin{bmatrix} 0.000 \\ 0.020 \end{bmatrix}$$

$$\Phi_2 = \begin{bmatrix} -0.995 & 0.0002 \\ -2.4660 & -0.9950 \end{bmatrix} \quad \Gamma_2 = \begin{bmatrix} 0.0001995 \\ 0.0002466 \end{bmatrix}$$

For the controller design we want to satisfy

$$x(kh)^T P x(kh) \geq \min_u \int_{kh}^{kh+h} (x^T Q_c x + u^T R u) dt + x(kh+h)^T P x(kh+h)$$

$$\forall \quad h \in H = \{h_1, h_2\}$$

we take the same weights as in the introductory example

$$Q_c = \begin{bmatrix} 20000 & 0 \\ 0 & 20000 \end{bmatrix} \quad R = 50.$$

We then obtain $Q_{1,q}$, $Q_{2,q}$ and $Q_{3,q}$ by solving (5.17)-(5.19), such that we can write

$$Q_q = \begin{bmatrix} Q_{1,q} & Q_{12,q} \\ Q_{12,q}^T & Q_{2,q} \end{bmatrix}$$

$$\forall \quad q \in \{1,2\}$$

$$Q_1 = \begin{bmatrix} 5329.5 & -394.6 & -0.529 \\ -394.6 & 39.5 & 0.0395 \\ -0.529 & 0.0395 & 0.1001 \end{bmatrix}$$

$$Q_2 = \begin{bmatrix} 9381400 & -6.0714 & -938.1423 \\ -6.0714 & 933.2359 & 0.0010 \\ -938.1423 & 0.0010 & 4.7938 \end{bmatrix}$$

We can now solve the state feedback synthesis LMI

$$\begin{bmatrix} W_0 & (\Phi_q W_0 + \Gamma_i W_q)^T & [W_0 \;\; W_q^T] \\ \Phi_q W_0 + \Gamma_q W_q & W_0 & 0 \\ \begin{bmatrix} W_0 \\ W_q \end{bmatrix} & 0 & Q_q^{-1} \end{bmatrix} \geq 0$$

$$\forall \quad q \in \{1,2\}$$

and obtain $W_0 = W_0^T > 0$ and W_1, W_2, which gives the state feedback gains $K_q = W_q W_0^{-1}$ $\forall \quad q \in \{1,2\}$

$$K_1 = \begin{bmatrix} 0.5784 \\ -0.0570 \end{bmatrix} \quad K_2 = \begin{bmatrix} 1765.5 \\ 0.0109 \end{bmatrix}$$

Applying these state feedback gains guarantees stability and robustness against variations in sampling between h_1 and h_2. Further, the cost is bounded by $P = W_0^{-1}$

$$P = \begin{bmatrix} 2870500 & 12.812 \\ 12.812 & 286.9774 \end{bmatrix}$$

5.6 Conclusion

The chapter was concerned with optimal control of linear hybrid systems where the continuous dynamic is described in discrete time. It was shown that sampling a continuous time system at different sampling rates could be viewed as such a class of hybrid system. The subsystems of the hybrid system are determined by the continuous dynamic and the specific sampling rate, such that there is a subsystem associated with each sampling rate. The discrete dynamic, i.e. the choice of sampling rate, is decided by a scheduler. The scheduler manages computational resourses. When resourses are short the control law cannot be computed so frequently. The consequence is that the system is sampled more slowly. However, when resources become available the control law can be computed more frequently, which results in faster sampling. Using this particular example it was shown that problems might occur by applying standard optimal control laws to hybrid systems. To illustrate the problem of stability a controller was designed minimizing the same continuous loss function for two subsystems of a hybrid system. This lead to two stable closed loop systems. However, it was shown that for various switching sequences the hybrid system was unstable.

In order to overcome this shortcoming two solutions where presented. It was shown that restrictions on switching (scheduling) strategy can be imposed which guarantee stability. For cases where such restrictions cannot be imposed a different controller design was proposed. It was suggested that the objective function had to be minimized only over one sampling period instead of minimizing over the infinite horizon. It was shown that when a terminal penalty was added which is greater than or equal to the remaining cost for the worst case variations in sampling rate, the system is robustly stable against these variations.

The synthesis procedure was then formulated in terms of an LMI. In a second example the synthesis procedure using the proposed LMI was carried out on the introductory example. The state feedback gains, which are the solutions of the LMI, were given as well as the performance bound.

6 Optimal control of hybrid systems in continuous time

As in the previous chapter we address the problem of optimal control for a class of hybrid systems. This class of hybrid systems consists of linear dynamics in continuous time, referred to as subsystems, and an unknown or nondeterministic discrete dynamic. Since the discrete dynamic cannot be controlled we are interested in computing upper bounds on the optimal cost for the worst case switching strategy. For best case switching strategies a lower bound on the cost is computed. It is shown that for this class of hybrid system the upper and lower bounds on the cost, for the worst and the best switching strategies respectively, can be obtained conveniently by solving convex optimization problems in the form of a linear matrix inequality (LMI). The chapter shows that obtaining a controller from the upper bound gives a stable control law.

The contributions of this chapter are the LMI conditions for computing the upper bound on the optimal cost for driving the continuous states of a linear hybrid system with unknown discrete dynamic to zero.

6.1 Introduction

In this chapter we consider the optimal control of a class of hybrid system. The hybrid system is given as a finite collection of linear dynamics in continuous time where a discrete function decides which linear dynamic is valid. The hybrid dynamics are described as follows:

$$\left[\begin{array}{rcl} \dot{x} & = & A(q)x + B(q)u \\ y & = & C(q)x \\ q^{+} & = & f(q, x, t) \end{array}\right. \tag{6.1}$$

where $x \in X \subset \mathbf{R}^n$ is the continuous state and $q \in Q = \{1, 2, \ldots, N\} \subset \mathbf{Z}^{+}$ is the discrete state. q is the current discrete state and q^+ denotes its successor. $A(q)$, $B(q)$, $C(q)$ denote the system, input and output matrices respectively, which depend on the current discrete state q. $f(q, x, t)$ denotes the discrete dynamics, which in general can depend on the continuous state and the current discrete state as well as on time. It is assumed that $f(q, x, t)$ is unknown, which is the case for many real systems. One example is the model of a braking car where the braking dynamics depend strongly on road surface conditions (dry, wet, icy), which vary arbitrarily. These variations are modelled by a hybrid system where linear dynamics are derived for different road surface conditions. The fact that changes in road surface conditions are not predictable and occur discontinuously is captured by the unknown or non-deterministic discrete function $f(q, x, t)$. The only assumption that is made on $f(q, x, t)$ in the following, is that some time $t > 0$ elapses between consecutive transitions from the current discrete state q to a new discrete state q^+. The assumption is of a technical nature such that we do not exhibit Zeno executions (45). In general this assumption is satisfied by almost every real system and is therefore not restrictive.

For controller synthesis we want to find a controller which minimizes the following quadratic objective function

$$x^T P x = \min_u \int_0^\infty x^T Q x + u^T R u \quad dt \tag{6.2}$$

subject to our hybrid system (6.1). Since the discrete state cannot be influenced we can only obtain bounds on the optimal cost. For the worst case switching strategy we can compute the upper bound on the optimal cost, which we would like to minimize. While for the best case switching strategy we can obtain a lower bound on the optimal cost. It is shown that the upper and lower bound can be obtained by solving an LMI. Obtaining the upper bound gives a piecewise linear state feedback controller, which is stable for all discontinuous changes among the finite collection of linear dynamics that are assumed by the hybrid system.

6.2 Optimal control of hybrid systems in continuous time

6.2.1 Computation of upper bounds on the optimal cost

With this class of hybrid system it is not possible to control the discrete dynamic, i.e. there is no possibility to influence which subsystem is active. Due to this fact we have to consider all possible switching strategies. For the optimal control problem, which is driving the continuous state to zero from an initial condition while minimizing a performance index, we are interested in finding an upper bound on the cost for the worst case switching strategy. That is, there exists no switching strategy which has higher cost than the upper bound. Since the upper bound is a cost which is greater or equal than the optimal cost (6.2) we write

$$x^T P x \geq \min_u \int_0^\infty x^T Q x + u^T R u \quad dt. \tag{6.3}$$

where Q and R are positive definite weights. Assume that the optimal feedback is given by $u = K_q x$, so that we replace u by $K_q x$

$$x^T P x \geq \int_0^\infty x^T Q x + x^T K_q^T R K_q x \quad dt. \tag{6.4}$$

Differentiating with respect to time we obtain

$$-\dot{x}^T P x - x^T P \dot{x} \geq x^T Q x + x^T K_q^T R K_q x \tag{6.5}$$

and replacing $\dot{x} = (A_q + B_q K_q)x$ we obtain

$$-x^T (A_q + B_q K_q)^T P x - x^T P (A_q + B_q K_q) x \geq x^T Q x + x^T K_q^T R K_q x. \tag{6.6}$$

We can now write the matrix inequality

$$-(A_q + B_q K_q)^T P - P(A_q + B_q K_q) \geq Q + K_q^T R K_q \tag{6.7}$$

Since it is desirable to obtain the K_q for all q and P by solving a convex optimization problem we aim to transform the matrix inequality (6.7) into a linear matrix inequality (LMI), which can then be solved conveniently by standard tools like the MATLAB LMI-toolbox. It is easy to see that the inequality (6.7) is not convex. In the following, steps are carried out to obtain convexity. We first replace P by multiplying from left and right with P^{-1}, to obtain

$$-P^{-1}(A_q + B_q K_q)^T - (A_q + B_q K_q)P^{-1} \geq P^{-1} Q P^{-1} + P^{-1} K_q^T R K_q P^{-1}$$

To recover convexity we make a change of variables and replace $M = P^{-1}$, $W_q = K_q P^{-1}$. Hence

$$-MA_q^T - W_q^T B_q^T - A_q M - B_q W_q \geq MQM + W_q^T R W_q$$

Rearranging the inequality we get

$$0 \geq MA_q^T + W_q^T B_q + A_q M + B_q W_q + \begin{bmatrix} M^T \\ W_q^T \end{bmatrix} \begin{bmatrix} Q & 0 \\ 0 & R \end{bmatrix} \begin{bmatrix} M \\ W_q \end{bmatrix}$$

Applying Schur's complement to the above expression we obtain

$$\begin{bmatrix} MA_q^T + W_q^T B_q + A_q M + B_q W_q & M^T & W_q^T \\ M & -Q^{-1} & 0 \\ W_i & 0 & -R^{-1} \end{bmatrix} \leq 0 \qquad (6.8)$$

which is an LMI in $M = M^T > 0$ and $W_q \ \forall \ q \in \{1, 2, \ldots, N\}$. In order to minimize the upper bound we want to minimize P subject to (6.8) by minimizing the trace of P. However, minimizing the trace of P subject to (6.8) is not a convex optimization problem. Fortunately, we can make use of a trick (20) to overcome this problem and regain convexity. Instead of minimizing the trace of P we minimize

$$\log \det M^{-1}$$

subject to the LMI (6.8) (remember $M^{-1} = P$). The solution of the LMI (6.8) then gives the state feedback gains $K_q = W_q M^{-1} \ \forall \ \ q \in \{1, 2, \ldots, N\}$.

Theorem: If there exists $P = P^T > 0$, K_q, $q \in \{1, 2, \ldots, N\}$ such that

$$(A_q + B_q K_q)^T P + P(A_q + B_q K_q) + Q + K_q^T R K_q \leq 0 \qquad (6.9)$$

$$\forall \ \ q \in \{1, 2, \ldots, N\}$$

then the hybrid system (6.1) is stable and its performance is bounded by $x^T P x$.

Proof: It is easy to show that $V(x) = x^T Px$ serves as a Lyapunov function for the hybrid system (6.1) since $P = P^T > 0$ and

$$\begin{aligned} \dot{V}(x) &= x^T(A_q + B_qK_q)^T Px + x^T P(A_q + B_qK_q)x \\ &\leq -x^T[IK_q]O[IK_q]^T x \end{aligned} \tag{6.10}$$

$$\forall \quad q \in \{1, 2, \ldots, N\} \quad O = \begin{bmatrix} Q & 0 \\ 0 & R \end{bmatrix}$$

where O is positive definite and $[IK_q]$ are full rank, therefore $-[IK_q]O[IK_q]^T$ is negative definite and hence $\dot{V} < 0$.
Since $u = K_q x$ we replace $K_q x$ by u in (6.10) and integrate

$$V(x(0)) - V(x(\infty)) = \quad -\int_0^\infty \dot{V}(x) \quad dt \geq \min_u \int_0^\infty x^T Qx + u^T Ru \quad dt.$$

Since the system is asymptotically stable we know $x \to 0$ as $t \to \infty$. Hence we see that the cost for driving the hybrid system (6.1) from an initial state $x = x(0)$ to $x = 0$ is bounded from above by

$$V(x) = x^T Px \geq \min_u \int_0^\infty x^T Qx + u^T Ru \quad dt.$$

□

In the following we want to get a bound on the best case switching strategy.

6.2.2 Computations of lower bounds on the optimal cost

The lower bound is now computed. This lower bound is a cost which is lower than any cost for any switching sequence. Note, we compute the lower bound not for controller synthesis, since it does not lead to a stable control strategy, we compute the lower bound for comparison with the upper bound. In this way we can asses how close we are to optimality. The computation of the lower bound should be not mistaken for the dual of the upper bound since for the upper bound we consider the worst switching sequence and for the lower bound we consider the best switching sequence. For computation of the lower bound the inequality is reversed

$$x^T P x \leq \min_u \int_0^\infty x^T Q x + u^T R u \quad dt \tag{6.11}$$

Following the steps as above we derive what is known as the Hamilton-Jacobi-Bellman inequality

$$0 \leq \frac{\partial V}{\partial x} f(x,u) + L(x,u) \tag{6.12}$$

where $L(x,u)$ is the time derivative of our objective function $L(x,u) = x^T Q x + u^T R u$ and $\frac{\partial V}{\partial x} f(x,u)$ is the time derivative of the cost (loss) function $\frac{\partial V}{\partial x} f(x,u) = \dot{V}(x) = (A_q x + B_q u)^T P x + x^T P (A_q x + B_q u)$, such we can write

$$0 \leq (A_q x + B_q u)^T P x + x^T P (A_q x + B_q u) + x^T Q x + u^T R u \tag{6.13}$$

Bringing the inequality above into a quadratic form we get

$$\begin{bmatrix} x^T \\ u^T \end{bmatrix}^T \begin{bmatrix} A_q^T P + P A_q + Q & P B_q \\ B_q^T P & R \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} \geq 0$$

which gives us an LMI

$$\begin{bmatrix} A_q^T P + P A_q + Q & P B_q \\ B_q^T P & R \end{bmatrix} \geq 0 \tag{6.14}$$

in $P = P^T > 0$. Solving the above LMI gives then the desired lower bound on the cost.

6.3 Conclusion

This chapter was concerned with the optimal control of a class of hybrid system. This class of hybrid system consists of linear subsystems, describing the continuous dynamics. The discrete dynamic, however, is unknown. Although the discrete dynamic is unknown, the active subsystem can can be immediately detected. The hybrid system can be controlled by a continuous input only. This hybrid framework is of importance since it describes various relevant systems. For optimal control, a quadratic performance index is minimized consisting of the continuous state and the continuous control input. Changes of the discrete state are not penalized since the discrete state cannot be influenced directly. Since the discrete behaviour cannot be influenced, only a bound on the optimal cost (the cost for driving the states from initial conditions to zero) can be attained. From the worst case switching strategy we obtain an upper bound on the optimal cost. While for the best case switching sequence we obtain a lower bound on the optimal cost. The procedure to obtain the upper and lower bounds was formulated into a linear matrix inequality (LMI) which is a convex optimization problem and can be solved with various commonly available tool boxes. It was shown that together with the upper bound we can obtain a piecewise linear state feedback law which is robustly stable against all possible switching sequences.

Part III

Observer design for Hybrid Systems

7 Observer design for hybrid systems

The chapter illustrates problems involving state estimation of hybrid systems. The problems of state estimation are discussed for the class of linear hybrid system where the continuous dynamic is described in discrete time. Such systems play an important role as they occur in many computer control applications using sampled-data control, as we have shown in chapter 5. For many such systems it is not possible to take state measurements at equidistant time intervals. This results ultimately in a hybrid system where the current dynamic (subsystem) is determined by the current sampling rate. It is shown that the state estimate might diverge if an observer is designed, minimizing the covariance of the estimation error for each subsystem (i.e. sampling rate) individually. An observer synthesis is proposed which overcomes this problem. This observer is piecewise linear and converges for all switching sequences, i.e. under all sampling rate variations. The design of the observer is cast into an LMI and gives piecewise linear observer gains. The procedure is then illustrated in an example.
The contributions of this chapter lies in the novel LMI observer synthesis for optimal state observers of linear hybrid systems.

7.1 Introduction

The previous chapters were concerned with the control of hybrid systems. For most of these approaches state feedback was used. This control strategy requires that the states are available. Sometimes it is possible to measure states but generally, of course, this is not the case. Also, sometimes it is considered too expensive to get additional sensors measuring the states. In other cases it is impractical to equip the process with such sensors. Therefore estimates of the states need to be obtained. For this purpose observers are designed. An observer is a dynamic system which has similar structure and parameters to the dynamic which is observed, preferably they are the same. The observer can be viewed as a dynamical system with input and output. The input to the observer is the output of the dynamic usually referred to as $y(t)$, where the t indicates that the output of the dynamic is a function of time. The output of the observer is the estimate $\hat{x}(t)$ of the state $x(t)$. Observers with such structure are called Luenberger observers. The estimates of such observers convergence asymptotically, i.e. the estimation error $e(t) = x(t) - \hat{x}(t)$ vanishes as time goes to infinity, $t \rightarrow \infty$.

In this chapter we are concerned with the design of observers for a special class of hybrid systems. This class has been introduced in chapter 5. The class of hybrid system consists of linear subsystems describing the continuous dynamic in discrete time. The change of subsystem is not a priori known but can be detected immediately. Thus the hybrid system is described by

$$\left[\begin{array}{lcl} x_{k+1} & = & \Phi_q x_k + \Gamma_q u_k + \Psi_q wd_k \\ y_k & = & C_q x_k + vd_k \\ q^+ & = & f(\cdot) \end{array}\right. \tag{7.1}$$

where $x \in X \subset \mathbf{R}^n$ is the continuous state in discrete time and $q \in Q = \{1, 2, \ldots, N\} \subset \mathbf{Z}^+$ is the discrete state. q is the current discrete state and q^+ denotes its successor. Φ_q, Γ_q, C_q denote the system, input and output matrices of the discrete time system, respectively, which depend on the current discrete state q. $\Psi_q wd_k$ is the process noise and vd_k is the measurement noise, its properties are discussed later. $f(\cdot)$ denotes the discrete dynamics.

This class of hybrid system seems to be important since most control applications involve computer control where the control output is computed on a computer and then applied to the control process. Measurements are taken from the process with sensors and sent back to the computer where the new control output is computed. Since computers operate with discrete quantities these continuous signals are sampled. Discrete time models which are equivalent to the continuous time models can be obtained to represent the sampled signals. A hybrid system of such form arises then for many reasons, one is the variation of sampling rate due to computational resource changes as described in chapter 5. Another reason is that measurements are sent via bus systems to controllers. At instances where the measurement should be sent the bus might be busy and the measurement cannot be sent. The measurement is then taken at a later instance where it can be sent. In such a setup variations in sampling can be detected immediately, since the time between the arrival of different measurements can be measured. Hence the change between different subsystems of the hybrid system are detected immediately.

Using this particular example where variations in sampling rate generate a hybrid system, we show how observers can be constructed. It is shown that such configurations cannot be treated directly with methods used for continuous systems. This fact is illustrated by an example which shows that the estimate might diverge when switching between commonly designed asymptotic observers. An observer is proposed which overcomes this problem. The observer is asymptotic such that no on-line inversions are needed to compute the observer gain, which is otherwise a high computational burden. Similar to the Kalman filter, the observer minimizes the covariance of the estimation error asymptotically. Hence the estimate converges for all switching sequences of the hybrid system. The observer has piecewise constant observer gain that depends on the current subsystem, i.e. sampling rate. The observer synthesis is stated as an LMI which gives the desired gains. In an example the synthesis procedure is illustrated.

7.2 Example 1: Two different sampling rates, asymptotic observer for each sampling rate

In this section observer synthesis for sampled data systems is explained. Following the standard procedure of observer synthesis a discrete asymptotic Kalman filter is derived. In an example it is shown that applying such observers to a hybrid system causes problems. For our example the following continuous system is observed

$$\begin{aligned} \dot{x} &= Ax + bu + Gw \\ y &= Cx + v \end{aligned} \tag{7.2}$$

where

$$A = \begin{bmatrix} 0 & 1 \\ -1000 & -0.1 \end{bmatrix} \quad b = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{7.3}$$

$$C = [1 \quad 1] \quad G = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

are the system, input, output and process noise matrices. The noise is expected to have zero mean, i.e. $E[w_k] = E[v_k] = 0$ and the covariance of the noise is given by

$$\begin{aligned} E[Gw_k w_j^T G^T] &= \begin{cases} Qn & k = j \\ 0 & k \neq j \end{cases} \\ E[v_k v_k^T] &= \begin{cases} Rn & k = j \\ 0 & k \neq j \end{cases} \end{aligned}$$

Moreover, the noise is uncorrelated, $E[w_k v_j^T] = 0 \; \forall \; k, j$. The continuous system is stable with poles in the left hand-side of the complex plane $p_{1,2} = -0.05 \pm 31.6i$. In this example we assume that the measurements can be taken at two different sampling rates $h_1 = 0.004$ and $h_2 = 0.08$. For each sampling rate we obtain a discretization by zero order hold, where $\Phi_i = e^{Ah_i}$, $\Gamma_i = \int_0^{h_i} e^{As} B ds$. In this way we obtain a hybrid system with two subsystems. The two discrete subsystems are represented by

$$\begin{aligned} x_{k+1} &= \Phi_q x_k + \Gamma_q u_k + \Psi_q w d_k \\ y_k &= C x_k + v d_k \end{aligned} \tag{7.4}$$

$$q \in \{1,2\}$$

The covariance of the discrete subsystems are denoted by Qd_q, Rd_q. We proceed by designing an asymptotic observer for each subsystem, i.e. for each sampling rate, minimizing the covariance of the estimation error asymptotically. The equations of the observer are given by, (for simplicity we set $u = 0$)

$$\begin{aligned} \hat{x}_{k+1} &= \hat{\Phi}_q \hat{x}_k + L_q \left[y_k - \hat{C}\hat{x}_k\right] \\ \hat{y}_k &= C\hat{x}_k \end{aligned} \tag{7.5}$$

where the hat denotes the estimate. We can now write the equation for the estimation error

$$e_{k+1} = x_{k+1} - \hat{x}_{k+1} = \Phi_i x_k + \Psi_q wd_k - \hat{\Phi}_q \hat{x}_k - L_q \left[Cx_k + vd_k - \hat{C}\hat{x}_k\right] \tag{7.6}$$

Since we take $\Phi_q = \hat{\Phi}_q$, $C = \hat{C}$ we get

$$e_{k+1} = (\Phi_q - L_q C)e_k + \Psi_q wd_k - L_q vd_k \tag{7.7}$$

The objective of the Kalman filter is to generate the optimal estimate from the measured output y_k of the observed system and the previous estimate $\hat{x}_{k-1}$. This is done by minimizing the covariance of the estimation error. This objective is expressed by

$$P_k = E\left[e_k e_k^T\right] \tag{7.8}$$

Replacing the estimation error by (7.7) we obtain

$$P_k = E\left[((\Phi_q - L_q C)e_{k-1} + \Psi_q wd_{k-1} - L_i vd_{k-1}).((\Phi_q - L_q C)e_{k-1} + \Psi_q wd_{k-1} - L_q vd_{k-1})^T\right] \tag{7.9}$$

Using the fact that the expectation operator is linear we write

$$\begin{aligned} P_k = {} & E\left[(\Phi_q - L_q C)e_{k-1}e_{k-1}^T(\Phi_q - L_q C)^T\right] + E\left[\Psi_q wd_{k-1}e_{k-1}^T(\Phi_q - L_q C)^T\right] - \\ & E\left[L_q vd_{k-1}e_{k-1}^T(\Phi_q - L_q C)^T\right] + E\left[(\Phi_q - L_q C)e_{k-1}wd_{k-1}^T\Psi_q^T\right] + \\ & E\left[\Psi_q wd_{k-1}wd_{k-1}^T\Psi_q^T\right] - E\left[L_q vd_{k-1}wd_{k-1}^T\Psi_q^T\right] + \\ & E\left[(\Phi_q - L_q C)e_{k-1}vd_{k-1}^T L_q^T\right] + E\left[\Psi_q wd_{k-1}vd_{k-1}^T L_q^T\right] + E\left[L_q vd_{k-1}vd_{k-1}^T L_q^T\right] \end{aligned}$$

Since the mixed terms are uncorrelated the expression above can be reduced by using

$$\begin{aligned} E[\Psi_q wd_k wd_j^T \Psi_q^T] &= \begin{cases} Qd_q & k = j \\ 0 & k \neq j \end{cases} \\ E[vd_k vd_k^T] &= \begin{cases} Rd_q & k = j \\ 0 & k \neq j \end{cases} \end{aligned}$$

to

$$P_k = E\left[e_k e_k^T\right] = (\Phi_q - L_q C)P_{k-1}(\Phi_q - L_q C)^T + Qd_q + L_q Rd_q L_q^T \tag{7.10}$$

For the asymptotic Kalman filter, which is also referred to as stationary or steady state Kalman filter, $P_k \equiv P_{k-1}$. The advantage of the asymptotic Kalman filter compared to the recursive Kalman filter is that the gain can be computed off-line, which eliminates the online matrix inversion. In order to find the state update gains L_q and the covariance of the estimation error P we have to solve the corresponding algebraic Riccati equation

$$P_q = \Phi_q^T P_q \Phi_q + Qd_q - (C^T P_q \Phi_q)^T (Rd_q + C^T P_q C)^{-1} (C^T P_q \Phi_q) \tag{7.11}$$

The state update gain L_q is then given by

$$L_q = (\Phi_q P_q C^T)(Rd_q + CP_q C^T)^{-1} \tag{7.12}$$

In general it is easy to check if the estimates converge. We only have to check that the spectral radius of $\rho(\Phi_q - L_q C)$ is smaller than one.
In our example we take the covariances to be

$$Qn = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix} \quad Rn = 0.5 \tag{7.13}$$

Solving the Riccati equations we obtain the state update gains

$$L_1 = \begin{bmatrix} -0.0010 \\ 0.1149 \end{bmatrix} \quad L_2 = \begin{bmatrix} 0.0077 \\ -0.8752 \end{bmatrix} \tag{7.14}$$

Both observers converge individually, i.e. $\rho(\Phi_1 - L_1 C) = 0.9439 < 1$, $\rho(\Phi_2 - L_2 C) = 0.5056 < 1$. However, if we use the observers for sequences where the sampling rate changes, we find that there are sequences which do not converge. One of these sequences results if the system is sampled with $4 \times h1$ and then $1 \times h2$ repeatedly. Figure 7.1 shows this sequence. Hence such an observer cannot be applied to a hybrid system, since any possible switching sequence has to be anticipated. For our example this means all possible sampling rate variations between h_1 and h_2 have to be considered. The above sequence shows one sequence where the estimates do not converge. It turns out that there are a number of such sequences which do not converge. Table 7.1 shows these sequences In this section we have seen that problems can

	$\rho((\Phi_2 - L_2 C)^{mh_2} (\Phi_1 - L_1 C)^{nh_1}) > 1$			
$n \cdot h_1$	$4 \cdot h_1$	$5 \cdot h_1$	$6 \cdot h_1$	$7 \cdot h_1$
$m \cdot h_2$	$1 \cdot h_2$	$1 \cdot h_2$	$1 \cdot h_2$	$1 \cdot h_2$

Table 7.1: Sequences which do not converge

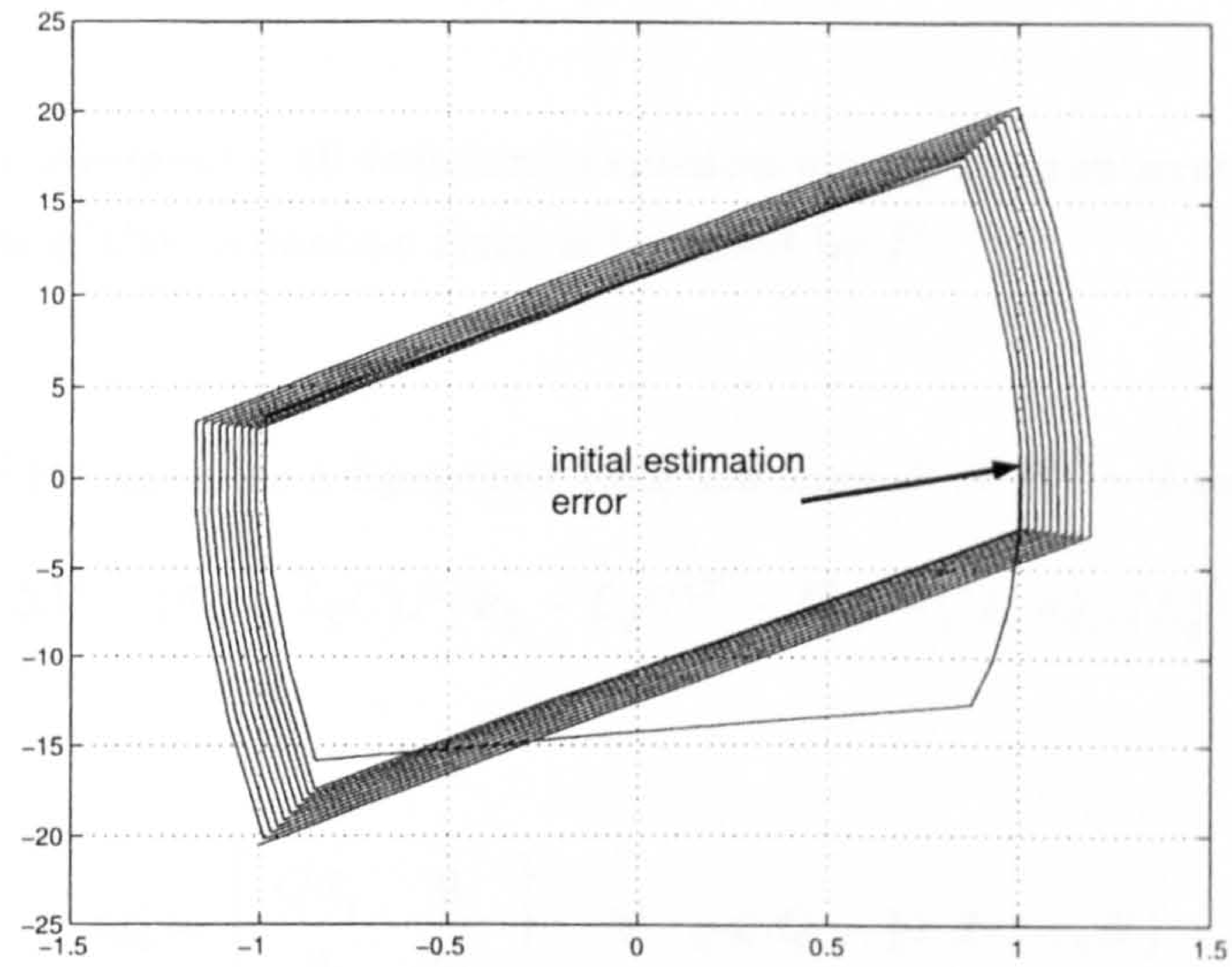


Figure 7.1: Divergent sequence of the estimation error

occur observing the states of a hybrid system using conventional methods. As an example of a hybrid system with two subsystems a continuous process was sampled at two different sampling rates. The variations in sampling rate led to a hybrid system where the discrete dynamic (the change between subsystems) was not a priori known but could be detected immediately. It was shown that the estimates could diverge when switching between Kalman filters which where designed for each subsystem individually. Therefore we need to find a design which is robust to switching among subsystems, i.e. which is robust to variations in sampling rate, such that the estimates do converge. In the next section we propose such a design which overcomes this problem. This observer can be viewed as a Kalman filter for hybrid systems, since it minimizes the covariance of the estimation error. More importantly this observer will be robustly stable to switching between subsystems, such that the estimates are guaranteed to converge asymptotically.

7.3 Observer for systems with varying sampling rate

We will now give the conditions under which the observer is robustly stable to switching between subsystems of the hybrid system.

Theorem 1: If there exists $P = P^T > 0$, L_q, $q \in \{1, 2, \ldots, N\}$ such that

$$(\Phi_q - L_q C)P(\Phi_q - L_q C)^T - P + (Qd_q + L_q Rd_q L_q^T) \leq 0 \tag{7.15}$$

$$\forall \quad q \in \{1, 2, \ldots, N\}$$

then the observer converges for all switching sequences among the q subsystems, $q \in \{1, 2, \ldots, N\}$ and the covariance of the estimation error is bounded by P.

Proof: $V(e) = e^T P e$ serves as a Lyapunov function since $P = P^T > 0$ and

$$\Delta V = (\Phi_q - L_q C) P (\Phi_q - L_q C)^T - P \leq -[I L_q] Q_q [I L_q^T]^T \tag{7.16}$$

with

$$Q_q = \begin{bmatrix} Qd_q & 0 \\ 0 & Rd_q \end{bmatrix} \quad \forall \quad q \in Q = \{1, 2, \ldots, N\}$$

where Q_q are positive definite for all q and $[I L_q]$ are full rank, therefore $-[I L_q] Q_q [I L_q^T]^T$ is negative definite and hence $\Delta V(e) \leq 0$.
From (7.10) we know that the covariance of the estimation error is given by

$$E\left[e_k e_k^T\right] = (\Phi_q - L_q C) P_{k-1} (\Phi_q - L_q C)^T + Qd_q + L_q Rd_q L_q^T. \tag{7.17}$$

Since

$$P \geq E\left[e_k e_k^T\right] = (\Phi_q - L_q C) P_{k-1} (\Phi_q - L_q C)^T + Qd_q + L_q Rd_q L_q^T \tag{7.18}$$

the covariance of the estimation error is bounded by P.

□

We have seen if we manage to find an observer which satisfies (7.15), we can guarantee that the observation converges despite switching among q, $q \in \{1, 2, \ldots, N\}$, subsystems. We will now show how we can formulate the observer synthesis into an LMI, such that we obtain P and L_q.

7.4 Observer synthesis using LMI

We have seen that the estimate converges for all switching sequences among the specified subsystems $q \in \{1, 2, \ldots, N\}$, if we find the L_q and P that satisfy the conditions of Theorem 1. The remaining problem is to find the L_q and P. The difficulty is that this condition cannot be used directly since it is not convex.

The following steps aim to transform the condition such that we obtain a convex problem. We have

$$(\Phi_q - L_qC)P(\Phi_q - L_qC)^T - P + \; + (Qd_q + L_qRd_qL_q^T) \leq 0$$

which we can write as

$$\begin{bmatrix} \Phi_q - L_qC \\ I \\ L_q \end{bmatrix}^T \times \begin{bmatrix} P & 0 & 0 \\ 0 & Qd_q & 0 \\ 0 & 0 & Qd_q \end{bmatrix} \times \begin{bmatrix} (\Phi_q - L_qC)^T \\ I \\ L_q^T \end{bmatrix} - P \leq 0 \qquad (7.19)$$

$$\forall \quad q \in \{1, 2, \ldots, N\}$$

Applying Schur's complement to the above expression we obtain

$$\begin{bmatrix} P & (\Phi_q - L_qC) & \begin{bmatrix} I & L_q \end{bmatrix} \\ (\Phi_q - L_qC)^T & P^{-1} & 0 \\ \begin{bmatrix} I \\ L_q^T \end{bmatrix} & 0 & Q_q^{-1} \end{bmatrix} \geq 0$$

$$\forall \quad q \in \{1, 2, \ldots, N\}$$

where

$$Q_q = \begin{bmatrix} Qd_q & 0 \\ 0 & Qd_q \end{bmatrix}$$

Multiplying from left and right with

$$\begin{bmatrix} P^{-1} & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix}$$

and setting $M_0 = P^{-1}$, $M_q = P^{-1}L_q$ we obtain the observer synthesis LMIs

$$\begin{bmatrix} M_0 & M_0\Phi_q - M_qC & [M_0 \quad M_q] \\ (M_0\Phi_q - M_qC)^T & M_0 & 0 \\ \begin{bmatrix} M_0 \\ M_q \end{bmatrix} & 0 & Q_q^{-1} \end{bmatrix} \geq 0 \qquad (7.20)$$

$$\forall \quad q \in \{1, 2, \ldots, N\}$$

in $M_0 = M_0^T > 0$ and M_q. The solution of the LMI (7.20) gives the observer gains $L_q = M_0^{-1}M_q \ \forall \ \ q \in \{1,2,\dots,N\}$. With these observer gains the estimation error converges robustly for all switching sequences, i.e. under all variations among the sampling times h_q $\forall \ \ q \in \{1,2,\dots,N\}$. However, we would not only like to have an estimate that converges for all switching sequences, we would further like to minimize the covariance of the estimation error, which is P (7.8). Therefore we could minimize the trace of $M_0^{-1} = P$. Unfortunately this is a non-convex optimization problem. Instead of minimizing $Trace(M_0^{-1})$ we minimize

$$\log\det M_0^{-1}$$

subject to (7.20) which is a convex optimization problem ((20)).
We have shown how the observer synthesis problem is cast into an LMI. We will now demonstrate the procedure on the introductory example and give the piecewise linear observer gain for which the estimates converge for all switching sequences, i.e. under all possible variations in sampling rate.

7.5 Example

We will now demonstrate the observer synthesis procedure by building an observer for the system (7.2) again. However, since we use the synthesis procedure above we will be certain that the estimate will converge for all possible switching sequences, i.e. for all variations among all specified sampling rates. We sample the system again with the same sampling rates $h_1 = 0.004s$, $h_2 = 0.08s$. Using $\Phi_q = e^{Ah_q}$, $\Gamma_q = \int_0^{h_q} e^{As}Bds \ \forall \ \ q \in \{1,2\}$ we obtain the hybrid system with the two subsystems described by

$$\begin{aligned} x_{k+1} &= \Phi_q x_k + \Gamma_q u_k + \Psi_q w d_k \\ y_k &= Cx_k + vd_k \end{aligned}$$

$$q \in \{1,2\}$$

where

$$\Phi_1 = \begin{bmatrix} 0.9920 & 0.0040 \\ -3.9885 & 0.9916 \end{bmatrix} \quad \Gamma_1 = \begin{bmatrix} 0.000 \\ 0.004 \end{bmatrix}$$

$$\Phi_2 = \begin{bmatrix} -0.8145 & 0.0181 \\ -18.0891 & -0.8163 \end{bmatrix} \quad \Gamma_2 = \begin{bmatrix} 0.0018 \\ 0.0181 \end{bmatrix}$$

For the observer design we want to satisfy

$$(\Phi_q - L_qC)P(\Phi_q - L_qC)^T - P + (Qd_q + L_iRd_qL_q^T) \leq 0$$

$$\forall \quad h \in H = \{h_1, h_2\}$$

We take the same weights as in the introductory example

$$Qn = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix} \qquad Rn = 0.5$$

such that we can write

$$Q_q = \begin{bmatrix} Qd_q & 0 \\ 0 & Rd_q \end{bmatrix}$$

$$\forall \quad q \in \{1,2\}$$

$$Q_1 = \begin{bmatrix} 0.0020 & -0.0020 & 0 \\ -0.0020 & 0.0067 & 0 \\ 0 & 0 & 125 \end{bmatrix} \quad Q_2 = \begin{bmatrix} 0.0165 & -0.0914 & 0 \\ -0.0914 & 23.4723 & 0 \\ 0 & 0 & 6.25 \end{bmatrix}$$

We can now solve the observer synthesis LMI

$$\begin{bmatrix} M_0 & M_0\Phi_q - M_qC & [M_0 \quad M_q] \\ (M_0\Phi_q - M_qC)^T & M_0 & 0 \\ \begin{bmatrix} M_0 \\ M_q \end{bmatrix} & 0 & Q_q^{-1} \end{bmatrix} \geq 0$$

$$\forall \quad q \in \{1,2\}$$

and obtain $M_0 = M_0^T > 0$ and M_1, M_2, which gives the observer gains $L_q = M_0^{-1}M_q$ $\forall \quad q \in \{1,2\}$:

$$L_1 = \begin{bmatrix} 0.0018 \\ 0.7199 \end{bmatrix} \quad L_2 = \begin{bmatrix} 0.0193 \\ -0.9310 \end{bmatrix}$$

Applying these observer gains guarantees convergence of the state estimate for all possible switching sequences, i.e. robustness against variations in sampling between h_1 and h_2. Further the covariance of the estimation error is bounded by $P = W_0^{-1}$,

$$P = \begin{bmatrix} 0.1275 & -0.4959 \\ -0.4959 & 289.1227 \end{bmatrix}$$

The spectral radius of the observer is $\rho(\Phi_1 - L_1C) = 0.9756 < 1$, $\rho(\Phi_2 - L_2C) = 0.8546 < 1$.

7.6 Conclusion

This chapter was concerned with observer design for a class of hybrid system. In this class of hybrid system the linear continuous dynamic is described in discrete time. The discrete dynamic, i.e. the switching between subsystems, is not a priori known but can be detected immediately. Hybrid systems of such a class arise in sampled-data control with varying sampling time. Many control schemes require knowledge of the current state. Often it is not possible to measure all required states. In these cases observers are used, which give an estimate of the states. Designing such observers, one could proceed with the methodology applied to continuous or discrete time systems. An example showed that applying such approaches directly to a hybrid system is not suitable. In this example an observer was designed for a stable continuous system which is sampled at two different sampling rates. In this way a hybrid system with two subsystems in discrete time was derived. An asymptotic observer was designed, minimizing the covariance of the estimation error for both subsystems individually. It was shown that although the estimates converged individually for each subsystem, i.e. for each sampling rate, it did not for various sequences where the sampling rate was changed repeatedly, i.e. for switching among the two subsystems.

In order to overcome this shortcoming an observer design was presented which converged robustly for all possible switching sequences, i.e for all variations in sampling rate. This observer uses a piecewise linear observer gain, which depends on the current sampling rate, i.e. subsystem. By measuring the time between consecutive output samples it is straight forward to detect the current sampling rate (subsystem).

The observer synthesis was cast into an LMI. The solution of the LMI gives the observer gains as well as the bound on the covariance of the estimation error. In a second example the synthesis procedure using the proposed LMI was carried out on the introductory example. The observer gains, which are the solutions to the LMI, were given, as well as the performance bound.

Part IV

Identification of Hybrid Systems

8 Identification of hybrid systems in discrete time

This chapter is concerned with identification of piecewise linear (affine) maps as they occur in linear hybrid systems. The identification is performed by a series of optimizations. The first step is to build local data sets for each input/output data pair containing the $c - 1$ nearest data pairs. An affine map is fit to each local data set minimizing the least squares error. This creates two types of estimate, the pure one that is obtained from local data sets which contain data from only one map and the outliers containing data from different maps. In the second step the estimates, obtained from the local data sets, are grouped in clusters such that each cluster collects the pure estimates and possibly outliers as well. Such clusters are referred to as perfect clusters and non-mixed clusters, respectively. The parameters of each affine map are computed simultaneously by minimizing the weighted distance between the members of each cluster. The chapter gives the conditions under which perfect and non-mixed clusters are obtained. Further conditions are given for detection of the outliers. It is shown that removing outliers improves the identified affine maps considerably.
The main contribution of this chapter lies in the conditions under which perfect and non-mixed clusters are obtained as well as in the a posteriori detection of outliers. This chapter has been submitted for publication (35).

8.1 Introduction

Previously we have been concerned with the analysis and control of linear hybrid systems. One requirement to carry out such analysis was that a model of the process is available. The design of controllers also requires models. There are basically two different possibilities to obtain such dynamical models. One possibility is to derive a white box model by writing down the equations which govern the process. Laws of physics are used to model real systems which usually involves differential equations, possibly in combination with logical statements.

In the case where no such knowledge is available a model is found by black box identification. This involves the stimulation of the process and measuring the response. With the input and output data a model can be found which describes the system. The difference between model predictions and measured data is minimized in a suitable metric in order to find the model which describes the input/output data as closely as possible. This involves solving an optimization problem. The solution of the optimization problem comprises the parameters and possibly also the structure of the model.

In this chapter we are concerned with the identification of a class of linear hybrid system, as considered in previous chapters. Since in general implementations are done on computers we focus only on hybrid systems where the continuous dynamic is described in discrete time. Thus, we are concerned with hybrid systems modelled by

$$\left[\begin{array}{ccc} x_{k+1} & = & \Phi_q x_k + \Gamma_q u_k \\ y_k & = & C_q x_k \\ q^+ & = & f(x) \end{array}\right. \tag{8.1}$$

where $x \in X \subset \mathrm{R}^n$ is the continuous state in discrete time and $q \in Q = \{1, 2, \ldots, N\}$ is the discrete state. We denote q as the current discrete state and q^+ denotes its successor. Φ_q, Γ_q, C_q denote the system, input and output matrices of the discrete time system, respectively, which depend on the current discrete state q. $f(x)$ denotes the discrete dynamics.

The identification of such a hybrid system aims to reconstruct the piecewise linear, or affine, functions of the hybrid system (32). In addition, the domain of each function, i.e. the area where the individual function is valid, has to be identified. This makes the identification of hybrid systems a challenging problem, since it is not immediately obvious which input/output data pair belongs to which affine map. Therefore three problems have to be solved. The first problem is the classification problem, i.e. which input/output data pair belongs to which function. The second problem is the regression problem, i.e. reconstructing the individual function from the input/output data. The third problem is to reconstruct the domain of each map. A general procedure for solving the identification problem was proposed in (34) (33) and consists of a series of steps. The first step is to group the M input-output data pairs in M local data sets, such that each local data set is labelled by a different input/output data pair and contains the $c-1$ closest input/output data points. In the second step, linear regression is applied to each local data set such that M parameter vectors are obtained. Some local parameter vectors will provide an estimate of the true sub-models. However, it can happen that data points generated from different sub-models are collected in the same local data set. In this case, the corresponding parameter vectors are termed outliers since they do not represent any of the true sub-models. Therefore, together with each parameter vector, there is a measure of confidence one should have on the fact that the parameter vector is not an outlier. In the third step a clustering technique is used. The parameter vectors previously obtained are grouped on the basis of a similarity measure such that there is a cluster of parameter vectors associated with each affine sub-model. Since a local data set is associated to each parameter vector and each local data set is labelled by a single input/output data point, the clusters of parameter vectors can be used to partition the original data points in N sets, one for each sub-model. In the ideal case, each set should then embrace all the data points generated by a single sub-model. In the last step each sub-model is identified by using the final sets of input/output data. The same data sets can be used for estimating the region of validity of each sub-model. It is obvious that the crucial point of the procedure is the clustering step. In the ideal case all the parameter vectors which provide an estimate of the same sub-model should be collected in the same cluster. In this case, we say that perfect clustering is achieved. However, clustering is done through an optimization procedure whose results can be spoiled either by the noise affecting the data points or by the presence of outliers. In the first part of the chapter we summarize the identification procedure proposed in (34). Next, we focus on the clustering step and characterize the cases when perfect clustering is achieved. First, in section 8.3 we prove that in the absence of outliers, if the noise level is sufficiently small perfect clustering results from optimization. Second, in section 8.4, we generalize the previous results to the case when outliers are present.

Finally, based on these result, we present an easy computational method for detecting outliers a posteriori. We demonstrate, through an example, that by using the detection procedure and by removing outliers in the clustering step, the quality of the identification procedure improves considerably.

8.2 The identification algorithm

We now summarize the identification algorithm proposed in (34) and introduce the notation that will be used throughout the chapter. The starting point is to derive an input/output representation of system (8.1). This is necessary since input/output data are used for identification. Thus, consider the N affine maps

$$f_q(x(k)) = \begin{bmatrix} x(k) \\ 1 \end{bmatrix}^T \bar{\theta}_q \tag{8.2}$$

where $x(k)$ denotes the vector of regressors made of past inputs u and outputs y

$$x(k) = \begin{bmatrix} y(k-1)^T & y(k-2)^T & \cdots & y(k-n)^T & u(k-1)^T & \cdots & u(k-m)^T \end{bmatrix}^T \tag{8.3}$$

and $\bar{\theta}_q$ denotes the parameter vector

$$\bar{\theta}_q = \begin{bmatrix} a_{q,1}^T & a_{q,2}^T & \cdots a_{q,n}^T & b_{q,1}^T & \cdots & b_{q,m}^T & g_q \end{bmatrix}^T \tag{8.4}$$

In this chapter we consider, for data generation, the general piecewise affine model

$$y(k) = f(x(k)) + \eta(k) \tag{8.5}$$

$$f(x(k)) = f_q(x(k)) \quad \text{if} \quad x(k) \in \mathcal{X}_q, \quad k \in \{1, \ldots, M\} \tag{8.6}$$

where $\{\mathcal{X}_q, q = 1, \ldots, N\}$ is a polyhedral partition of the set $\mathbf{X}$ of possible regressors, that is assumed polyhedral and bounded. Moreover, $\eta(k)$ are the noise samples which are assumed to be independent and identically distributed in a Gaussian way with $E[\eta(k)] = 0$ and $Var[\eta(k)] = \sigma^2$. The maps $f_q(\cdot)$, each one together with its region of validity $\mathcal{X}_q$, are referred to as sub-models.

We denote the collection of data points by $\Omega = \{(x(k), y(k)), \quad k \in \{1, \ldots, M\}\}$.

The problem of identifying the PWA map is threefold, since the number N of affine maps and the polyhedral domains $\mathcal{X}_q$ as well as the PWA map itself has to be identified. In many cases the number of affine maps is known, which makes the problem easier. If we further know from which affine map which data point is generated the problem becomes trivial.

We can now present the identification algorithm proposed in (34). The method consists of four steps. In the first step local data sets are built.

Local Data sets (LDs): For each data point $(x(j), y(j))$ there is an LD C_j, containing the data point and its $c-1$ nearest neighbours. More precisely,.the LD C_j collects $(x(j), y(j))$ and the $c-1$ distinct datapoints $(\tilde{x}, \tilde{y})$ that satisfy

$$\|x(j) - \tilde{x}\|^2 \leq \|x(j) - \hat{x}\|^2, \quad \forall(\hat{x}, \hat{y}) \in \Omega \backslash C_j. \tag{8.7}$$

Note, that c is a parameter of the algorithm satisfying $c > n + m + 1$. We refer to C_j as pure LD if it collects only data points obtained from a single affine map. An LD is called mixed if not all data points are from the same affine map. Note that the distinction between pure and mixed LDs is conceptual and cannot be done a priori since the regions of the true piecewise affine model are unknown.

Local parameter vectors (LPVs): From each LD a local parameter vector is estimated on the basis of

$$Y_j = \begin{bmatrix} y_1 \\ \vdots \\ y_c \end{bmatrix}, \quad \Phi_j = \begin{bmatrix} x_1 & \cdots & 1 \\ \vdots & \cdots & \vdots \\ x_c & \cdots & 1 \end{bmatrix} \tag{8.8}$$

where $\Phi_j \in \mathbf{R}^{c \times (n+m-1)}$ consists of all $x_k : (x_k, y_k) \in C_j$ and $Y_j \in \mathbf{R}^c$ of all $y_k : (x_k, y_k) \in C_j$. For later purposes, we also define

$$\bar{Y}_j = \begin{bmatrix} f(x((1)) \\ \vdots \\ f(x(c)) \end{bmatrix}, \quad \varepsilon_j = \begin{bmatrix} \eta(1) \\ \vdots \\ \eta(c) \end{bmatrix} \tag{8.9}$$

In other words, $\bar{Y}_j$ is the vector containing the noise free outputs characterizing the data points in Y_j (see (8.5)) and $\varepsilon_j \in \mathbb{R}^c$ is the corresponding noise vector. The local parameter vector is denoted by θ_j and obtained by using least squares estimation,

$$\theta_j = Q_j Y_j \tag{8.10}$$

where $Q_j = (\Phi_j^T \Phi_j)^{-1} \Phi_j^T$. The LPV θ_j can be obtained from a pure or mixed LD. In view of the bijective correspondence between LDs and LPVs, an LPV is called *pure LPV* if it is obtained from a pure LD and *mixed LPV (or outlier)* otherwise.

Local confidence measure: For every LPV its covariance matrix is estimated, as standard in least squares theory, through the formula

$$V_j = \frac{SSR_j}{c-(n+m+1)}(\Phi_j^T \Phi_j)^{-1}, \quad SSR_j = Y_j^T (I - \Phi_j (\Phi_j^T \Phi_j)^{-1} \Phi_j^T) Y_j \tag{8.11}$$

Note that, intuitively, if V_j is the covariance of a pure LPV and V_i is the covariance of a mixed LPV, we expect $V_i > V_j$, since V_i accounts also for the modelling error which increases SSR_i. Therefore we can use V_j^{-1} as a measure of the confidence about LPV θ_j being pure.
Clustering step: The next step aims to partition all LPV in N clusters. In the ideal case each cluster should contain all pure LPVs generated by the same sub-model. The clusters are denoted by $\{\mathcal{D}_q\}_{q=1}^N$ and are disjoint sets that collect all LPV θ_j, i.e. $\{\theta_1, \cdots, \theta_M\} = \cup_{q=1}^N \mathcal{D}_q$. As usual in clustering theory, we find the clusters by solving an optimization problem, i.e. by minimizing the clustering cost

$$J\left(\{\mathcal{D}_q\}_{q=1}^N, \{\mu_q\}_{q=1}^N\right) = \sum_{q=1}^N \sum_{j \in \Im \mathcal{D}_q} \|\theta_j - \mu_q\|^2_{V_j^{-1}} \tag{8.12}$$

where μ_q are the centres of the clusters and the operator $\Im$ acting on the set $\mathcal{D}_q$ gives the collection of indices of the elements belonging to $\mathcal{D}_q$. For instance, if $\mathcal{D}_1 = \{\theta_1, \theta_3, \theta_8\}$, we get $\Im\mathcal{D}_1 = \{1, 3, 8\}$. The clusters that minimize J will be denoted as $\mathcal{D}_q^*$. Note that the distances between centres and LPVs depends on the confidence measures. Intuitively, this means that the clustering results will depend mainly on the LPVs with associated "high" confidence.

Since we assume that the number N of affine sub-models is known, then a proper version of the K-means algorithm can be used for minimizing the cost (8.12). If the number of affine maps is unknown then other clustering technique such as "Growing Neural Gas" (36) can be employed to cluster the LPV and estimate N at the same time.

A cluster $\bar{\mathcal{D}}_q$ is said to be *exact* if it collects all and only the pure LPVs estimated on the basis of pure LDs collecting data points generated by a single sub-model. Without loss of generality we assume that $\bar{\mathcal{D}}_q$ collects all pure LPVs associated to the q^{th} sub-model. Note that exact clusters can be defined independently of the optimization results.

The collection of exact clusters is called $\bar{\mathcal{D}}_{exact}$, i.e. $\bar{\mathcal{D}}_{exact} = \cup_{q=1}^{N} \bar{\mathcal{D}}_q$. Let $\bar{\mathcal{D}}_{mixed}$ be the set of all mixed LPVs. Hence $\{\theta_1, \cdots, \theta_M\} = \cup_{q=1}^{N} \bar{\mathcal{D}}_q \cup \mathcal{D}_{mixed}$.
Definition: The clusters $\{\mathcal{D}_q^*\}_{q=1}^{N}$ are perfect if $\mathcal{D}_q^* \supseteq \bar{\mathcal{D}}_q$, $\forall q \in \{1, \ldots, N\}$.
We note that if $\bar{\mathcal{D}}_{mixed} = \emptyset$ perfect clustering implies that $\mathcal{D}_q^* = \bar{\mathcal{D}}_q$, $\forall q \in \{1, \ldots, N\}$. In the case that mixed LPVs are present (i.e. $\bar{\mathcal{D}}_{mixed} \neq \emptyset$) perfect clustering means that the presence of outliers does not spoil the correct clustering of pure LPVs.

Identification of the sub-models: Note that the following bijective maps hold

$$(x(j), y(j)) \longleftrightarrow C_j \longleftrightarrow \theta_j \tag{8.13}$$

Therefore, once the LPVs have been clustered, we can cluster the original data points in the sets $\{\mathcal{F}_q\}_{q=1}^{N}$ by using (8.13). Then, we can use the points in each final set $\mathcal{F}_q$ for estimating the parameters of the q^{th} sub-model through least squares. Also the regions $\{\mathcal{X}_q\}_{q=1}^{N}$ can be estimated on the basis of the final sets by resorting to multicategory pattern recognition algorithms. For further details we defer the reader to (34).

We will now start to derive the conditions under which perfect clustering can be achieved. For the sake of clarity the derivations are split in two sections. We start considering the case $\bar{\mathcal{D}}_{mixed} = \emptyset$ that is easier and simpler to follow. Then, in section 8.4 this assumption is relaxed. In section 8.5 conditions for detection of outliers are given based on the results of section 8.4. An example illustrating the benefits of a posteriori outlier detection is given in section 8.6.

8.3 Perfect clustering without outliers

In this section we will derive conditions under which perfect clustering can be achieved. Some Lemmas are needed in order to derive the main result of this section, presented in Theorem 1. It will be easier to follow these Lemmas if one is aware of their nice geometrical interpretation. Lemma 1 shows that the LPVs from pure LDs lie in a ball centred around the true LPV (we denote a ball by $B(\bar{\theta}, \rho)$ where $\bar{\theta}$ is the center and ρ is the radius).
Upper bounds for finding the optimal LPV which assume minimum weighted distance to all LPVs in the ball are given in Lemma 3. The geometrical interpretation is that an upper bound is attained if estimates form groups, which lie opposite each other on the circumference

of the ball.

In Lemma 4 a lower bound is derived for the case where mixed clusters are considered. Since the LPVs belong to different maps they belong to different balls as well. A lower bound is attained if we neglect all LPVs of each ball but one, which gives a lower bound by Lemma 2. Then the lowest value for this configuration is attained, for non intersecting balls, if the LPVs lie on the circumference of the ball at the point where a line connecting the two centres of the ball intersect with the circumference of the ball.

In Theorem 1 we then give the condition under which perfect clustering can be achieved. The condition is an upper bound on the noise. Roughly speaking, perfect clustering is attained when the conditions from Lemma 3 yield a lower value than the conditions of Lemma 4.

Let us first assume that there are no estimates based on mixed LD.

We will now show that all pure LPVs characterizing the same sub-model are contained in a ball centered at the true LPV.

Lemma 1: For each $\theta_j \in \bar{\mathcal{D}}_q$ it holds that $\theta_j \in B_q(\bar{\theta}_q, \rho_q)$ where $\rho_q = \max_{j \in \Im \mathcal{D}_q} \| Q_j \varepsilon_j \|$.

Proof: The proof follows directly from the fact that $\forall j \in \Im \bar{\mathcal{D}}_q$

$$\theta_j = Q_j Y_j = Q_j \bar{Y}_j + Q_j \varepsilon_j = \bar{\theta}_q + Q_j \varepsilon_j \tag{8.14}$$

In view of the definition of ρ_q the result follows.

□

In Lemma 2 we recall the fact that we can obtain a lower bound to the optimal clustering cost by neglecting one LPV and solving again the clustering problem.

Lemma 2: Consider the optimization problem

$$\hat{J}^1 = \min_{\mu} J^1(\mu) = \min_{\mu} \sum_{i:\xi_i \in \Xi} \| \xi_i - \mu \| \tag{8.15}$$

where $\xi_i \in \mathbf{R}^n$ and Ξ is a finite collection of vectors ξ_i. Consider also the optimization problem

$$\hat{J}^2 = \min_{\mu} J^2(\mu) = \min_{\mu} \sum_{i:\xi_i \in \Xi \setminus \xi_{i^*}} \| \xi_i - \mu \| \tag{8.16}$$

where ξ_{i^*} is an arbitrary vector in Ξ. Then, the following fact holds

$$\hat{J}^1 \geq \hat{J}^2 \tag{8.17}$$

Proof: Denote with $\hat{\mu}_1$ the minimizer of (8.15). It follows directly from optimality that

$$\hat{J}^1 \geq J^2(\hat{\mu}_1) \geq \hat{J}^2 \tag{8.18}$$

□

Before proceeding we need to introduce some further notation. For a perfect cluster $\bar{\mathcal{D}}_q$, let $\bar{\lambda}_q$ denote the maximum eigenvalue of all maximum eigenvalues in the set of matrices $\{V_j^{-1} : j \in \Im\bar{\mathcal{D}}_q\}$, i.e. $\bar{\lambda}_q = \max_{j\in\Im\bar{\mathcal{D}}_q} \lambda_{max}(V_j^{-1})$ where, for a square matrix A, $\lambda_{max}(A)$ denotes its maximum eigenvalue. Analogously, $\underline{\lambda}_q$ denotes the minimum eigenvalue of all minimum eigenvalues in the set $\{V_j^{-1} : j \in \Im\bar{\mathcal{D}}_q\}$ i.e. $\underline{\lambda}_q = \min_{j\in\Im\bar{\mathcal{D}}_q} \lambda_{min}(V_j^{-1})$.
In Lemma 3 we will derive an upper bound on the optimal clustering cost, in the case of perfect clustering and in the absence of mixed LPVs

Assumption 1: $\bar{\mathcal{D}}_{mixed}$ is empty.

Lemma 3: Let Assumption 1 hold and consider the optimization problem

$$\min_{\{\mu_q\}_{q=1}^N} J(\{\bar{\mathcal{D}}_q\}_{q=1}^N, \quad \{\mu_q\}_{q=1}^N) \tag{8.19}$$

Then, the following upper bound holds

$$\bar{J} = \min_{\{\mu_q\}_{q=1}^N} J(\{\bar{\mathcal{D}}_q\}_{q=1}^N, \quad \{\mu_q\}_{q=1}^N) \leq \sum_{q=1}^N n_q \bar{\lambda}_q \rho_q^2 \tag{8.20}$$

where n_q denotes the cardinality of $\bar{\mathcal{D}}_q$.

Proof: By using the fact that $\| \theta_j - \mu_q \|^2_{V_j^{-1}} \leq \bar{\lambda}_q \| \theta_j - \mu_q \|^2$, we get

$$\min_{\mu_q} J(\{\bar{\mathcal{D}}_q\}_{q=1}^N, \quad \{\mu_q\}_{q=1}^N) \leq \min_{\mu_q} \sum_{q=1}^N \sum_{j\in\Im\bar{\mathcal{D}}_q} \bar{\lambda}_q \| \theta_j - \mu_q \|^2 \tag{8.21}$$

From optimality we get

$$\bar{J} \leq \min_{\mu_q} \sum_{q=1}^N \sum_{j\in\Im\bar{\mathcal{D}}_q} \bar{\lambda}_q \| \theta_j - \mu_q \|^2 \leq \sum_{q=1}^N \sum_{j\in\Im\bar{\mathcal{D}}_q} \bar{\lambda}_q \| \theta_j - \bar{\theta}_q \|^2 \tag{8.22}$$

where the last inequality follow from the choice $\mu_q = \bar{\theta}_q$. Lemma 1 implies that if $\theta_j \in \bar{\mathcal{D}}_q$, we have $\| \theta_j - \bar{\theta}_q \|^2 \leq \rho_q^2$. Hence

$$\min_{\{\mu_q\}_{q=1}^N} J(\{\bar{\mathcal{D}}_q\}_{q=1}^N, \quad \{\mu_q\}_{q=1}^N) \leq \sum_{q=1}^N \sum_{j \in \Im \bar{\mathcal{D}}_q} \bar{\lambda}_q \| \theta_j - \bar{\theta}_q \|^2 \leq \sum_{q=1}^N n_q \bar{\lambda}_q \rho_q^2 \tag{8.23}$$

□

The next aim is to derive a lower bound on the optimal cost in the case of non perfect clustering. A clustering error is committed if two parameter vectors, say $\theta_{\bar{k}} \in \bar{\mathcal{D}}_k$ and $\theta_{\bar{h}} \in \bar{\mathcal{D}}_h$, $k \neq h$, are grouped together in the same cluster $\mathcal{D}_q^*$, found through optimization. The error mentioned above can be represented through the pair of indices $(\bar{k}, \bar{h})$ meaning that $\theta_{\bar{k}}$ and $\theta_{\bar{h}}$ are grouped in the same cluster. The set of possible errors is then given by

$$\mathcal{E} = \left\{(\bar{k}, \bar{h}) : \forall \bar{k} \in \Im \bar{\mathcal{D}}_k, \ \forall \bar{h} \in \Im \bar{\mathcal{D}}_h, \ \forall k, h \in \{1, \ldots, N\}, k \neq h\right\} \tag{8.24}$$

In the next Lemma we derive a lower bound for the optimal clustering cost if at least one error is committed.

Assumption 2: All true parameter vectors are different, i.e. $\bar{\theta}_{q^*} \neq \bar{\theta}_q$, $\forall q, q^* \in \{1, \ldots, N\}$ such that $q^* \neq q$.

A lower bound for the case of mixed clusters is given by Lemma 4. Geometrically we can view this as saying there exists at least one cluster collecting LPVs which belong to two different balls. By Lemma 2 we reduce the problem to two LPVs of the mixed cluster, such that one is in each ball. As stated by Lemma 2 this yields a lower bound. For such a configuration the lowest value of the optimization problem is attained, when the two LPVs lie at the point where a line connecting the two centres of the balls intersect the circumference of the balls.

Lemma 4: Let Assumption 2 hold and let the clusters $\{\mathcal{D}_q\}_{q=1}^N$ be such that at least the clustering error $(\bar{k}, \bar{h})$ is committed, where $\theta_{\bar{k}} \in \bar{\mathcal{D}}_k$ and $\forall \theta_{\bar{h}} \in \bar{\mathcal{D}}_h$. Then, there exists $\beta_{\bar{k},\bar{h}} > 0$ such that if $|\eta(j)| < \beta_{\bar{k},\bar{h}}$, $\forall j \in \{1, \ldots, M\}$, the following inequality holds

$$J(\{\mathcal{D}_q\}_{q=1}^N, \quad \{\mu_q^*\}_{q=1}^N) \geq \frac{\underline{\lambda}_k \underline{\lambda}_h}{\underline{\lambda}_k + \underline{\lambda}_h} \mid T_{\bar{\theta}_k, \bar{\theta}_m} - (\rho_k + \rho_m) \mid^2 \tag{8.25}$$

where μ_q^* is the minimizer and $T_{\bar{\theta}_k, \bar{\theta}_m} = \| \bar{\theta}_k - \bar{\theta}_m \|$ denotes the distance between the corresponding true LPV and ρ_k, ρ_m are defined in Lemma 1.

Proof: Without loss of generality, assume that $\theta_{\bar{k}}$ and $\theta_{\bar{h}}$, are grouped together in the cluster $\mathcal{D}_{\bar{q}}$ Then we have

$$\min_{\{\mu_q\}_{q=1}^N} J(\{\mathcal{D}_q\}_{q=1}^N, \quad \{\mu_q\}_{q=1}^N) \geq \min_{\mu_{\bar{q}}} \sum_{j\in\Im\mathcal{D}_{\bar{q}}} \| \theta_j - \mu_{\bar{q}} \|^2_{V_j^{-1}} \geq \tag{8.26}$$

$$\geq \min_{\mu_{\bar{q}}} \left[\| \theta_{\bar{k}} - \mu_{\bar{q}} \|^2_{V_{\bar{k}}^{-1}} + \| \theta_{\bar{h}} - \mu_{\bar{q}} \|^2_{V_{\bar{h}}^{-1}} \right] \tag{8.27}$$

where the last inequality follows from Lemma 2, applied repeatedly. By using the fact that $\| \theta_{\bar{k}} - \mu_{\bar{q}} \|^2_{V_{\bar{k}}^{-1}} \geq \underline{\lambda}_k \| \theta_{\bar{k}} - \mu_{\bar{q}} \|^2$ and $\| \theta_{\bar{h}} - \mu_{\bar{q}} \|^2_{V_{\bar{h}}^{-1}} \geq \underline{\lambda}_h \| \theta_{\bar{h}} - \mu_{\bar{q}} \|^2$, we obtain

$$\min_{\mu_{\bar{q}}} \left[\| \theta_{\bar{k}} - \mu_{\bar{q}} \|^2_{V_{\bar{k}}^{-1}} + \| \theta_{\bar{h}} - \mu_{\bar{q}} \|^2_{V_{\bar{h}}^{-1}} \right] \geq \min_{\mu_{\bar{q}}} \underline{J}(\mu_{\bar{q}}) \tag{8.28}$$

where

$$\underline{J}(\mu_{\bar{q}}) = [\underline{\lambda}_k \| \theta_{\bar{k}} - \mu_{\bar{q}} \|^2 + \ + \underline{\lambda}_h \| \theta_{\bar{h}} - \mu_{\bar{q}} \|^2] \tag{8.29}$$

We proceed by computing the minimum of $\underline{J}(\mu_{\bar{q}})$. $\underline{J}(\mu_{\bar{q}})$ is convex and the minimizer $\mu^*_{\bar{q}}$ satisfies $\nabla_{\mu_{\bar{q}}} \underline{J}(\mu^*_{\bar{q}}) = 0$. Hence, the minimum is given by

$$\underline{J}(\mu^*_{\bar{q}}) = \frac{\underline{\lambda}_k \underline{\lambda}_h}{\underline{\lambda}_k + \underline{\lambda}_h} \| \theta_{\bar{k}} - \theta_{\bar{h}} \|^2 \tag{8.30}$$

From Lemma 1 it follows that $\theta_{\bar{k}} \in B(\bar{\theta}_k, \rho_k)$ and $\theta_{\bar{h}} \in B(\bar{\theta}_h, \rho_h)$. Consider $T_{\bar{\theta}_k,\bar{\theta}_h} = \| \bar{\theta}_k - \bar{\theta}_h \|$ that is strictly positive in view of Assumption 2. Then, if

$$T_{\bar{\theta}_k,\bar{\theta}_h} - (\rho_k + \rho_h) > 0 \tag{8.31}$$

we get the following relation, $\| \theta_{\bar{k}} - \theta_{\bar{h}} \|^2 \geq | T_{\bar{\theta}_k,\bar{\theta}_h} - (\rho_k + \rho_h) |^2$ from which (8.25) follows. The condition (8.31) represents the fact that the balls $B(\bar{\theta}_k, \rho_k)$ and $B(\bar{\theta}_h, \rho_h)$ do not intersect. Since both ρ_k and ρ_h tend to zero if the bound on the noise $\beta_{\bar{k},\bar{h}}$ goes to zero, we conclude that there exists a positive $\beta_{\bar{k},\bar{h}}$ for which condition (8.31) holds.

□

We combine now the results from Lemmas 3 and 4 in order to guarantee that if the noise is sufficiently small, perfect clustering is achieved.

Theorem 1: Under Assumptions 1 and 2, there exists $\beta > 0$ such that if $\beta > | \eta(j) |$ $\forall j \in \{1, 2, \ldots, M\}$, the clusters $\{\mathcal{D}^*_q\}_{q=1}^N$ that are the optimizers of

$$\min_{\{\mathcal{D}_q\}_{q=1}^N, \{\mu_q\}_{q=1}^N} J(\{\mathcal{D}_q\}_{q=1}^N, \quad \{\mu_q\}_{q=1}^N) \tag{8.32}$$

are perfect.

Proof: From Lemma 3 we get an upper bound on the cost by considering perfect clusters and from Lemma 4 we get a lower bound on the cost when an error $(\bar{k},\bar{h})$ is committed. Therefore the global minimum is attained for perfect clusters if the following inequality holds

$$\sum_{q=1}^{N} n_q \bar{\lambda}_q \rho_q^2 \leq \min_{(\bar{k},\bar{h})\in\mathcal{E}} \frac{\underline{\lambda}_k \underline{\lambda}_h}{\underline{\lambda}_k + \underline{\lambda}_h} \mid T_{\bar{\theta}_k,\bar{\theta}_h} - (\rho_k + \rho_h) \mid^2 \tag{8.33}$$

where, without loss of generality, we assumed that $\theta_{\bar{k}} \in \bar{\mathcal{D}}_k$ and $\theta_{\bar{h}} \in \bar{\mathcal{D}}_h$. The next goal is to get an explicit dependence of (8.33) on β. This is done in four steps.

Step 1. Prove that, $\forall \beta > 0$, if $|\eta(j)| < \beta$, $\forall j \in \{1,\dots,M\}$, then

$$\underline{\gamma}_j \beta^2 < SSR_j \leq \bar{\gamma}_j \beta^2 \tag{8.34}$$

where $\bar{\gamma}_j > 0$, $\underline{\gamma}_j > 0$ are suitable constants.
Let $H_j = (I - \Phi_j(\Phi_j^T\Phi_j)^{-1}\Phi_j^T)$. By direct calculation it is easy to verify that $H_j\bar{Y}_j = 0$. Therefore, by recalling that $Y_j = \bar{Y}_j + \varepsilon_j$, we have $SSR_j = Y_j^T H_j Y_j = \varepsilon_j^T H_j \varepsilon$ from which we get

$$c\beta^2 \lambda_{min}(H_j) \leq SSR_j \leq c\beta^2 \lambda_{max}(H_j) \tag{8.35}$$

The inequality (8.34) directly follows from (8.35).

Step 2. Prove that, $\forall \beta > 0$, if $|\eta(j)| < \beta$, $\forall j \in \{1,\dots,M\}$, then $\forall q \in 1,\dots,N$

$$\bar{\delta}_q \beta^{-2} \geq \underline{\lambda}_q \geq \underline{\delta}_q \beta^{-2}, \tag{8.36}$$

$$\bar{\nu}_q \beta^{-2} \geq \bar{\lambda}_q \geq \underline{\nu}_q \beta^{-2} \tag{8.37}$$

where $\underline{\delta}_q > 0$, $\bar{\delta}_q > 0$, $\underline{\nu}_q > 0$ and $\bar{\nu}_q > 0$ are suitable constants.
Consider the matrix V_j. Then

$$\lambda_{min}(V_j^{-1}) = \lambda_{min}(\Phi_j^T\Phi_j)\frac{c-(n+m+1)}{SSR_j} \tag{8.38}$$

and, from step 1, we get

$$\lambda_{min}(\Phi_j^T\Phi_j)\frac{c-(n+m+1)}{\underline{\gamma}_j\beta^2} \geq \lambda_{min}(\Phi_j^T\Phi_j)\frac{c-(n+m+1)}{SSR_j} \geq \lambda_{min}(\Phi_j^T\Phi_j)\frac{c-(n+m+1)}{\bar{\gamma}_j\beta^2} \tag{8.39}$$

Since, by definition, $\underline{\lambda}_q = \min_{j\in\Im\bar{\mathcal{D}}_q} \lambda_{min}(V_j^{-1})$, we have

$$\min_{j\in\Im\bar{\mathcal{D}}_q} \lambda_{min}(\Phi_j^T\Phi_j)\frac{c-(n+m+1)}{\underline{\gamma}_j\beta^2} \geq \min_{j\in\Im\bar{\mathcal{D}}_q} \lambda_{min}(V_j^{-1}) \geq \min_{j\in\Im\bar{\mathcal{D}}_q} \lambda_{min}(\Phi_j^T\Phi_j)\frac{c-(n+m+1)}{\bar{\gamma}_j\beta^2} \tag{8.40}$$

from which the inequality (8.36) follows. For proving inequality (8.37), start again considering the matrix V_j. Then,

$$\lambda_{max}(V_j^{-1}) = \lambda_{max}(\Phi_j^T\Phi_j)\frac{c-(n+m+1)}{SSR_j} \tag{8.41}$$

and, from step 1, we get

$$\lambda_{max}(\Phi_j^T \Phi_j)\frac{c-(n+m+1)}{\underline{\gamma}_j \beta^2} \geq \lambda_{max}(\Phi_j^T \Phi_j)\frac{c-(n+m+1)}{SSR_j} \geq \lambda_{max}(\Phi_j^T \Phi_j)\frac{c-(n+m+1)}{\bar{\gamma}_j \beta^2} \tag{8.42}$$

Since, by definition, $\bar{\lambda}_q = \max_{j \in \Im \bar{\mathcal{D}}_q} \lambda_{max}(V_j^{-1})$, we have

$$\max_{j \in \Im \bar{\mathcal{D}}_q} \lambda_{max}(\Phi_j^T \Phi_j)\frac{c-(n+m+1)}{\underline{\gamma}_j \beta^2} \geq \max_{j \in \Im \bar{\mathcal{D}}_q} \lambda_{max}(V_j^{-1}) \geq \max_{j \in \Im \bar{\mathcal{D}}_q} \lambda_{max}(\Phi_j^T \Phi_j)\frac{c-(n+m+1)}{\bar{\gamma}_j \beta^2} \tag{8.43}$$

from which the inequality (8.37) follows.

Step 3. Prove that, $\forall \beta > 0$, if $|\eta(j)| < \beta$, $\forall j \in \{1, \ldots, M\}$ then

$$\rho_q \leq \kappa_q \beta \tag{8.44}$$

where $\kappa_q > 0$ is a suitable constant. This inequality immediately follows from the definition of ρ_q

$$\rho_q = \max_{j \in \Im \mathcal{D}_q} \| Q_j \epsilon_j \| \leq \max_{j \in \Im \mathcal{D}_q} \| Q_j \| \| \epsilon_j \| \leq \max_{j \in \Im \mathcal{D}_q} \| Q_j \| \sqrt{c}\beta \tag{8.45}$$

Step 4. We now analyze both sides of (8.33) by combining the results of the previous steps. Consider the term $\bar{\zeta}_{\bar{k},\bar{h}}$ associated with the error $(\bar{k}, \bar{h})$ defined as

$$\bar{\zeta}_{\bar{k},\bar{h}} = \frac{\underline{\lambda}_k \underline{\lambda}_h}{\underline{\lambda}_k + \underline{\lambda}_h} \mid T_{\bar{\theta}_k, \bar{\theta}_h} - (\rho_k + \rho_h) \mid^2 \tag{8.46}$$

Since, under Assumption 2, $T_{\bar{\theta}_k, \bar{\theta}_h} > 0$, it is immediate to verify that

$$\lim_{\beta \to 0} \bar{\zeta}_{\bar{k},\bar{h}} = +\infty \tag{8.47}$$

In fact, for $q \in \{\bar{k}, \bar{h}\}$, we have $\bar{\delta}_q \beta^{-2} \geq \underline{\lambda}_q \geq \underline{\delta}_q \beta^{-2}$ (from step 2) and $\rho_q \leq \kappa_q \beta$ (from step 3). Then, $\forall U > 0$ it is possible to choose β small enough such that $\min_{(\bar{k},\bar{h}) \in \mathcal{E}} \bar{\zeta}_{\bar{k},\bar{h}} > U$. Consider now the term

$$\underline{\zeta}_q = n_q \bar{\lambda}_q \rho_q^2 \tag{8.48}$$

From the fact that $\bar{\nu}_q \beta^{-2} \geq \bar{\lambda}_q \geq \underline{\nu}_q \beta^{-2}$ (step 2) and $\rho_q \leq \kappa_q \beta$ (step 3) we can conclude that $\underline{\zeta}_q$ remains bounded for $\beta \to 0$. Then, the l.h.s. of (8.33) is bounded because it can be written as $\sum_{q=1}^{N} \underline{\zeta}_q$. Therefore, for $\beta > 0$ sufficiently small, inequality (8.33) can be verified since, for $\beta \to 0$ the l.h.s is bounded while the r.h.s. diverges.

□

Theorem 1 states that there exists a bound on the noise for obtaining perfect clusters in the absence of outliers. However, outliers can occur in many real identification problems.

Therefore we relax Assumption 1 and consider the case where outliers are present in the following section. The aim is to show that, if the noise level is small enough, the presence of mixed LPVs does not spoil the accuracy in clustering the pure LPVs.

8.4 Perfect clustering with outliers

We will consider the case of mixed LDs. Mixed LDs contain data from at least two different affine maps.

Lemma 5: Let Assumption 2 hold and consider the mixed LD $\mathcal{C}_j$. Then, the sum of the squared residuals

$$SSR_j = Y_j^T(I - \Phi_j(\Phi_j^T\Phi_j)^{-1}\Phi_j^T)Y_j \tag{8.49}$$

is strictly positive, even in the noiseless case.

Proof: A mixed LD $\mathcal{C}_j$ contains data points from at least two different sub-models, say the q-th and the q^*-th. Under Assumption 2, we have $\bar{\theta}_{q^*} \neq \bar{\theta}_q$, therefore a single affine map fitting all the data points in $\mathcal{C}_j$ does not exist, even for noiseless datapoints.

□

Theorem 2: Let Assumption 2 hold and assume that $\forall q \in \{1\ldots, N\}$, $\bar{\mathcal{D}}_q \neq \emptyset$. Then, there exists $\beta > 0$ such that if $\beta >| \eta(j) | \forall j \in \{1, 2, \ldots, M\}$, the clusters $\{\mathcal{D}_q^*\}_{q=1}^N$ that are the minimizers of

$$\min_{\{\mathcal{D}_q\}_{q=1}^N, \{\mu_q\}_{q=1}^N} J(\{\mathcal{D}_q\}_{q=1}^N, \quad \{\mu_q\}_{q=1}^N) \tag{8.50}$$

are perfect.

Proof: Consider the clusters $\{\mathcal{D}_q\}_{q=1}^N$ and assume that at least the clustering error $(\bar{k}, \bar{h})$, $\theta_{\bar{k}} \in \bar{\mathcal{D}}_k$, $\theta_{\bar{h}} \in \bar{\mathcal{D}}_h$ has been committed. Then, the lower bound on the cost given by Lemma 4

$$\min_{\{\mu_q\}_{q=1}^N} J(\{\mathcal{D}_q\}_{q=1}^N, \quad \{\mu_q\}_{q=1}^N) \geq \frac{\lambda_k\lambda_h}{\lambda_k + \lambda_h} \mid T_{\bar{\theta}_k, \bar{\theta}_h} - (\rho_k + \rho_h) \mid^2 \tag{8.51}$$

still holds. The key point, detailed in the proof of Lemma 4, is to observe that the lower bound is obtained by considering only the contribution of $\theta_{\bar{k}}$ and $\theta_{\bar{h}}$ to the cost functional.

As an upper bound to the cost functional in the case of perfect clusters $\{\mathcal{D}_q\}_{q=1}^N$, by using the same rationale employed for proving Lemma 2, we get

$$\min_{\{\mu_q\}_{q=1}^N} J(\{\mathcal{D}_q\}_{q=1}^N, \{\mu_q\}_{q=1}^N) \le \sum_{q=1}^N n_q \bar{\lambda}_q \rho_q^2 + \sum_{q=1}^N \sum_{j \in \Im\{\mathcal{D}_q \cap \mathcal{D}_{mixed}\}} \| \hat{\theta}_j - \bar{\theta}_q \|_{V_j^{-1}}^2 \tag{8.52}$$

The global minimum of the clustering cost is then attained for perfect clusters, if the following inequality holds

$$\sum_{q=1}^N n_q \bar{\lambda}_q \rho_q^2 + \sum_{q=1}^N \sum_{j \in \Im\{\mathcal{D}_q \cap \mathcal{D}_{mixed}\}} \| \hat{\theta}_j - \bar{\theta}_q \|_{V_j^{-1}}^2 < \min_{(\bar{k},\bar{h}) \in \mathcal{E}} \frac{\lambda_k \lambda_h}{\underline{\lambda}_k + \underline{\lambda}_h} \mid T_{\bar{\theta}_k, \bar{\theta}_h} - (\rho_k + \rho_h) \mid^2 \tag{8.53}$$

In the proof of Theorem 1 we have shown that the r.h.s of (8.53) goes to infinity as β decreases. Moreover, we proved that the term $\sum_{q=1}^N n_q \bar{\lambda}_q \rho_q^2$ remains bounded, for $\beta \to 0$. Therefore, if we show that the term

$$\sum_{q=1}^N \sum_{j \in \Im\{\mathcal{D}_q \cap \mathcal{D}_{mixed}\}} \| \hat{\theta}_j - \bar{\theta}_q \|_{V_j^{-1}}^2 \tag{8.54}$$

remains bounded, as β decreases, the result follows. Note that

$$\| \hat{\theta}_j - \bar{\theta}_q \|_{V_j^{-1}}^2 = (\hat{\theta}_j - \bar{\theta}_q)^T \Phi_j^T \Phi_j \frac{c-(n+m+1)}{SSR_j} (\hat{\theta}_j - \bar{\theta}_q) \tag{8.55}$$

From Lemma 5 we have that, for mixed LPVs, the sum of the squared residuals SSR_j is strictly positive even in the absence of noise. Moreover

$$\|\hat{\theta}_j - \bar{\theta}_q\| = \|Q_j \bar{Y}_j + Q_j \varepsilon_j - \bar{\theta}_q\| \le \|Q_j \bar{Y}_j - \bar{\theta}_q\| + \|Q_j \varepsilon_j\| \le \|Q_j \bar{Y}_j - \bar{\theta}_q\| + \|Q_j\| \sqrt{c}\beta \tag{8.56}$$

thus proving that the quantity (8.54) is bounded, for every finite value of β.

□

We have seen that in the presence of outliers the optimal clusters are also perfect if the noise is sufficiently small. Based on this result we state a method which can detect outliers a posteriori.

8.5 Mixed LDs: a posteriori detection

We will now show that under certain conditions it is possible to detect mixed LDs a posteriori. For this purpose we define the following operators.

Definition The operator In acting on LD returns the collection of all input data points in a LD, $In(\mathcal{C}_j) = \{x(k) : (x(k), y(k)) \in \mathcal{C}_j\}$. The operator *class* acting on a LD returns the index l of the cluster which collects the LPV identified from the LD, $class(\mathcal{C}_j) = l$ if $\theta_j \in \mathcal{D}^*{}_l$.

Definition The cardinality operator # acting on a set returns the number of elements contained in this set.

Lemma 6: If there is a set of indices I, $\#I \geq 2$, such that $\exists\ x(k) \in \cap_{j \in I} In(\mathcal{C}_j)$ with $class(\mathcal{C}_w) \neq class(\mathcal{C}_p)$, for at least two indices $w, p \in I$, and the noise satisfies the assumptions of Theorem 2, then at least one $\mathcal{C}_j$, $j \in I$, is mixed.

Proof: By contradiction, assume that all $\mathcal{C}_j$, $j \in I$ are pure. Under the assumptions of Theorem 2 all pure LPVs are correctly classified. Consider the indices $w, p \in I$ as defined above. Then, both θ_w, θ_p are pure and the fact that $x(k) \in \cap_{j \in I} In(\mathcal{C}_j)$ implies $class(\theta_w) = class(\theta_p)$. But this contradicts the assumption that $class(\mathcal{C}_w) \neq class(\mathcal{C}_p)$. Therefore, at least one $\mathcal{C}_j$, $j \in I$, is mixed.

□

The procedure of Lemma 6 allows determination of a set of LDs $\mathcal{C}_j$, $j \in I$ that are suspected of being mixed. Therefore all the data points in $\bigcup_{j \in I} In(\mathcal{C}_j)$ are suspected of having been misclassified. Usually, we can detect the misclassified data points by looking at the errors between the true and predicted outputs. In fact big errors are likely to be generated from such points. After detecting a misclassified data point we can use different strategies to improve the quality of the identified affine maps. Probably the simplest way is to neglect the point. It seems to be immediately obvious that this will improve the quality of the estimated submodels. However, for reconstructing the regions of the affine maps, the points $x(k)$ belonging to mixed LDs contain useful information. This is clear in view of the fact that if $\mathcal{C}_k$ is a mixed LD, $In(\mathcal{C}_k)$ contains input data points belonging to different sub-models. Therefore it is expected that $x(k)$ is close to the true boundary between different regions. Hence, it will make sense to remove the outliers, reconstruct the affine maps and re-attribute the outliers to the models that most likely generated them, before reconstructing the regions.

8.6 Example

The example demonstrates the outlier detection procedure through the identification of two affine maps. The true LPV of the maps are given by $\bar{\theta}_1 = [0.6, 0.6, 5]$ and $\bar{\theta}_2 = [-0.6, -0.6, 0]$. The variance of the noise is given by $\sigma^2 = 0.04$. Input/output data are generated with 25 input/output data points, and the LDs are created, by collecting the $c = 12$ nearest neighbours. The LPVs are obtained for each LD by using least squares. Then, the LPVs are clustered into two clusters and the coefficients of each affine map is reconstructed. Figure 8.1 shows the LPVs. LPVs which have been clustered as LPVs belonging to the first affine map are depicted by a diamond $\diamond$, while LPVs belonging to the second affine map are depicted by a triangle, $\triangle$. The LPVs which share input-output data, although having been clustered differently, are marked with a light cross $+$. This was done by using the procedure described in Lemma 6 for a posteriori outlier detection.

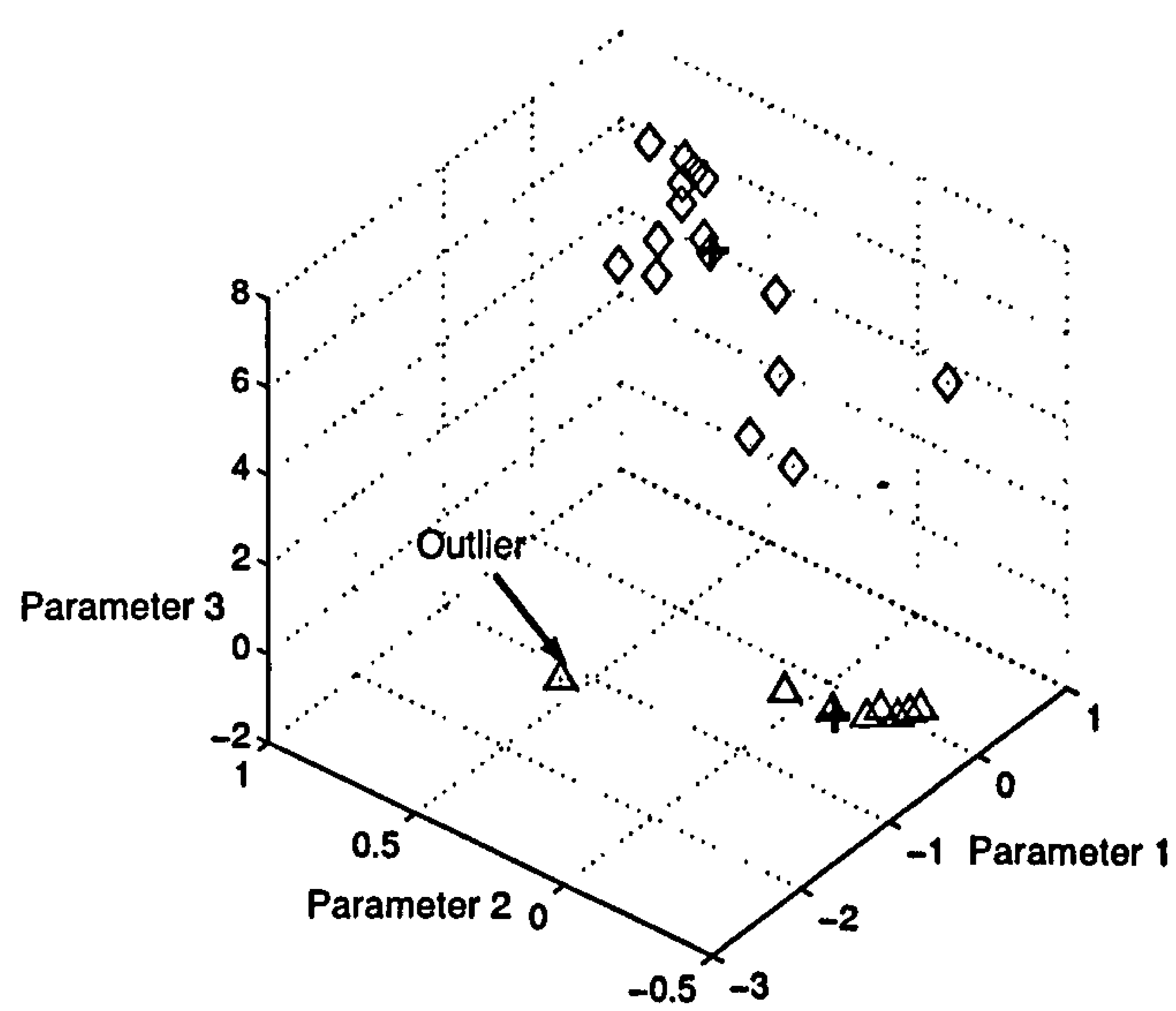


Figure 8.1: Clustered LPV containing an outlier

Comparing the distances between the output samples revealed the misclassified data points. Figure 8.2 gives the result obtained by removing the misclassified data points. Note that the corresponding outlier also disappeared. The identified parameters of the affine map are $\theta_1 = [0.6, 0.55, 5.0]$ and $\theta_2 = [-1.3, -0.4, 0.7]$ when the whole data set is used and $\theta_1 = [0.6, 0.55, 5.0]$ and $\theta_2 = [-0.7, -0.45, 0]$ when the misclassified data point has been removed. By comparing the parameters, we can see that neglecting the misclassified data points considerably improves the identification results.

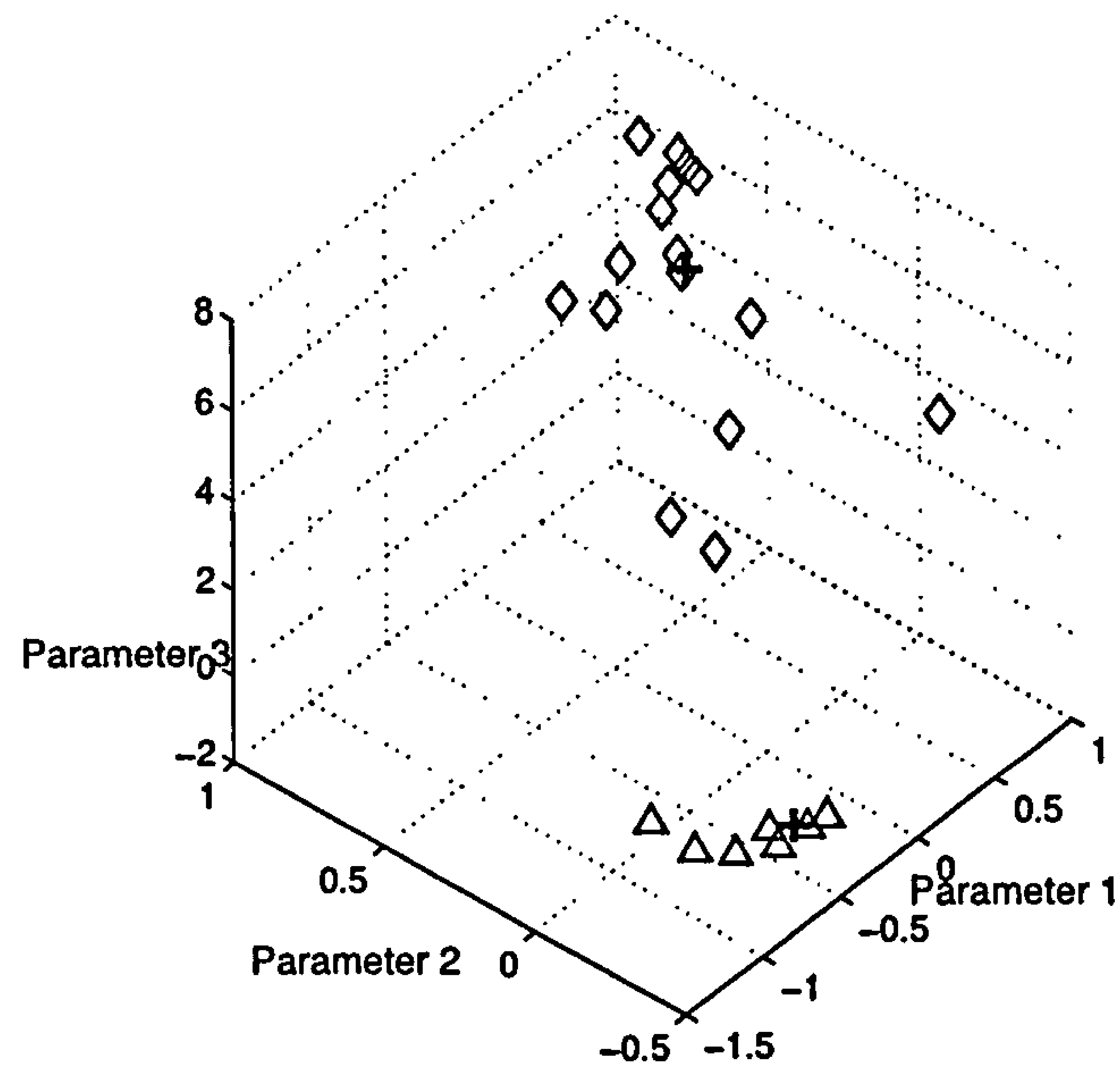


Figure 8.2: Clustered LPV after removing the misclassified data points

The projections of LPVs portrayed 8.1 and 8.2 are given in figures 8.3-8.4. First we display the projections into 1st, 2nd parameter plane 8.3. The first projection contains the outlier while the second projection shows the LPVs obtained after removing the misclassified data points (and then the outlier).

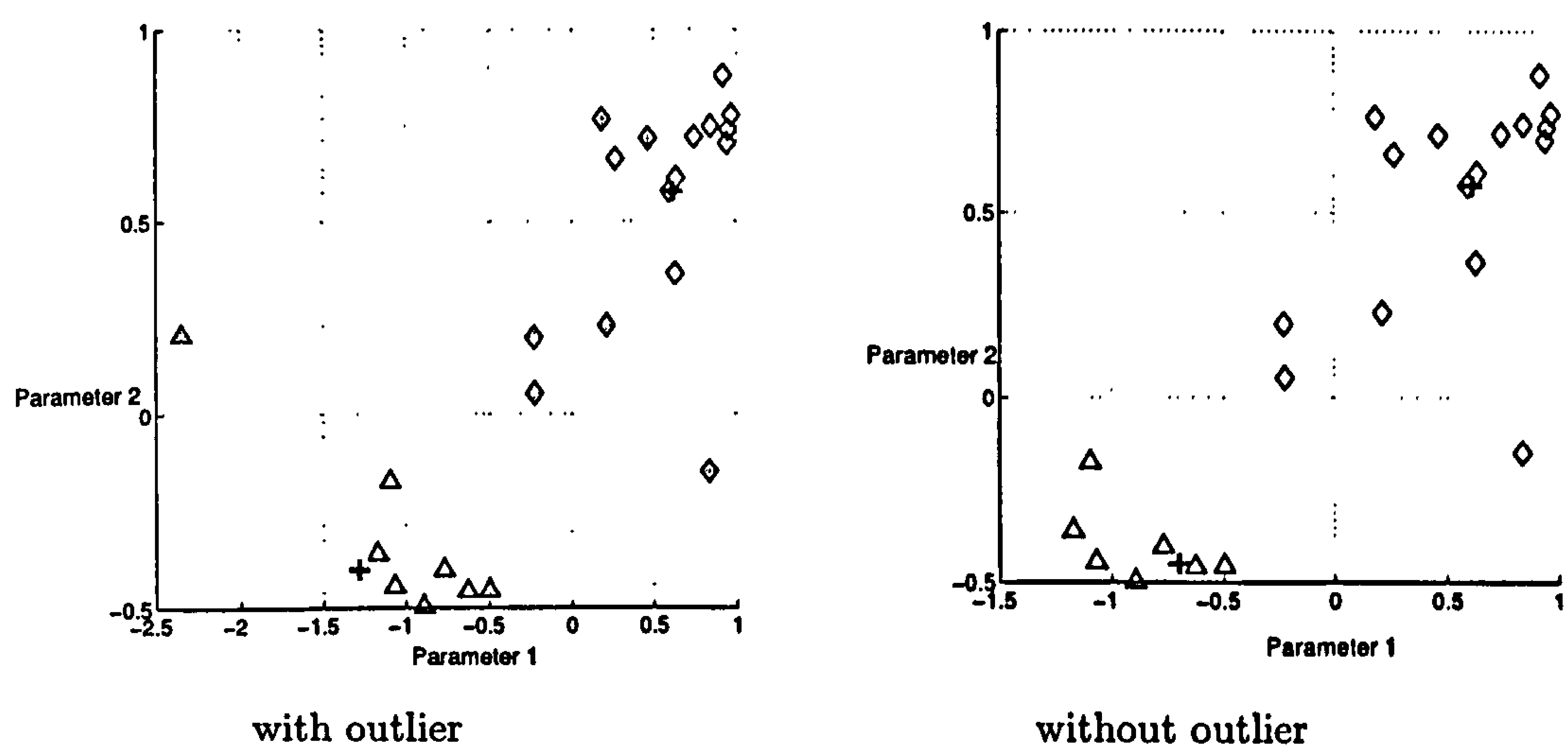
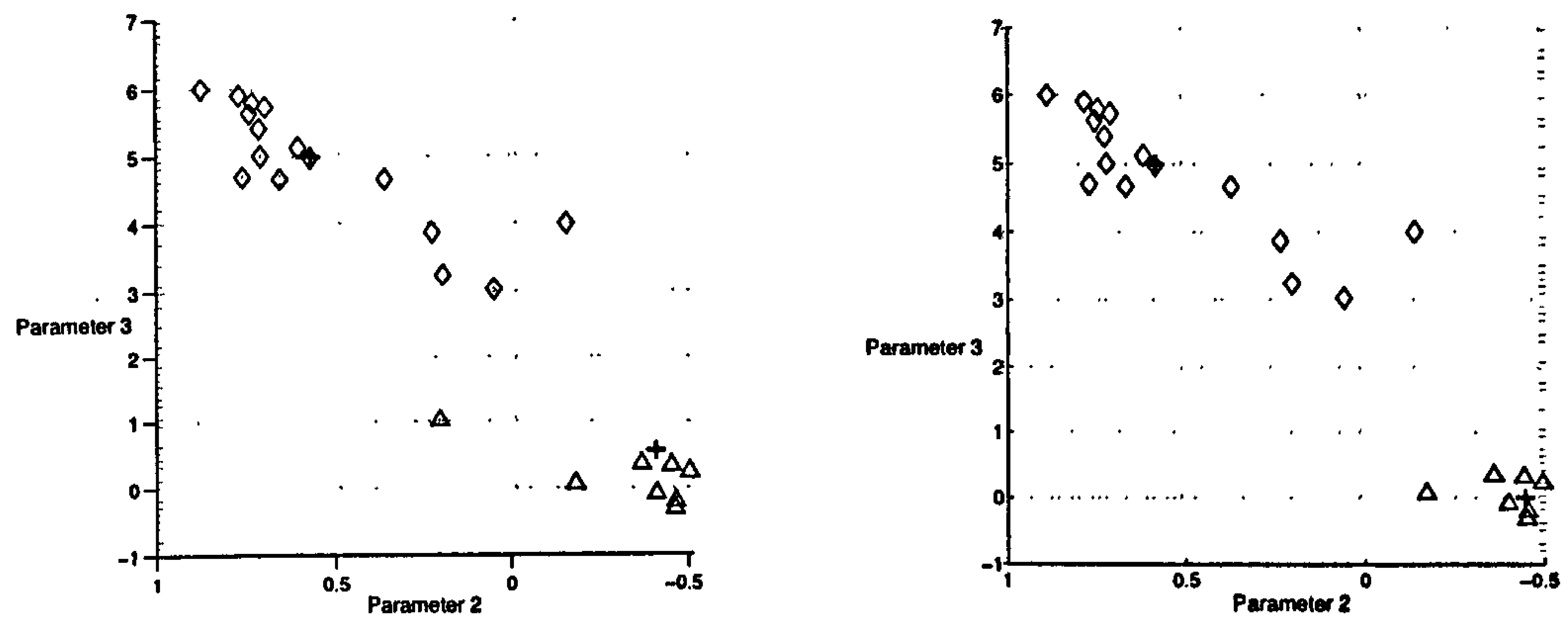


Figure 8.3: Projection with and without outlier into 1st and 2nd parameter plane

The two figures below show the projections of the LPVs into 2nd and 3rd parameter plane.

The left figure is the projection with outlier and the right figure is the projection after removal of the outlier.



with outlier without outlier

Figure 8.4: Projection with and without outlier into 2nd and 3rd parameter plane

8.7 Conclusions

This chapter was concerned with the identification of linear hybrid systems. Identifying such a hybrid system aims to reconstruct the piecewise linear, or affine, functions of the hybrid system. In addition, the domain of each function, i.e. the area where the individual function is valid, has to be identified. This is done in a sequence of steps. In the first step, input-output data pairs are grouped into local data sets. In the second step, linear regression is applied to each local data set such that local parameter vectors are obtained. An outlier is obtained if data points which are generated from different sub-models are collected in the same local data set. In the third step, a clustering technique is used which obtains a local parameter vector for each sub-model by grouping all local parameter vectors which belong to the same subsystem. In the last step the domains of each subsystem are obtained.

The main focus of this chapter was to characterize the cases for which perfect clustering is achieved. First, in section 8.3 we proved that in the absence of outliers, if the noise level is sufficiently small perfect clustering results from optimization. Second, in section 8.4, we generalize the previous results to the case where outliers are present.

Finally, based on these results, we present an easy computational method for detecting outliers a posteriori. We demonstrated this through an example by using the detection procedure and by removing the outlier in the clustering step. The quality of the identification procedure improves considerably.

Part V

Conclusions and Outlook

9 Summary and Conclusions

In this chapter we summarize previous achievements in the field of hybrid systems. After this general overview over hybrid systems we summarize the contents and contributions of this thesis. At the end of this chapter an outlook over future research is given.

9.1 Overview

The past decades and centuries treated dynamical systems as systems with either continuous dynamic or discrete dynamics. Presently, researchers focus on systems, called hybrid systems, combining continuous and discrete dynamics. Whether a dynamic is modelled as continuous, discrete or hybrid often depends on the level of abstraction. For some applications it is certainly sufficient to have a rough approximation of a hybrid dynamic and model it purely continuous or purely discrete. However, not all hybrid dynamics can be treated like this. At the latest this becomes clear when carrying out analysis of such systems. One quickly realizes that analyzing, for instance, the stability of a hybrid system one can draw no conclusions by looking at the purely continuous or purely discrete dynamics only, since stability of the continuous dynamic and stability of the discrete dynamic does not imply stability of the overall hybrid dynamic. This makes it clear that it is necessary to derive hybrid models. An overview of some hybrid frameworks was given in the introduction of this thesis. The frameworks range from very specific ones, which are only capable of modelling a very limited class of systems, to frameworks which encompass a vast number of classes. Frameworks which are restricted to fewer classes, however, are able to exploit the structure in order to give precise conditions for analysis and controller design. On the other hand, frameworks which encompass many classes usually give only poor answers in terms of analysis and controller design. In general, after an intensive literature review, it became clear that there are many different hybrid frameworks around. Some are extensions from continuous systems or from automata and incorporate the additional dynamic. Those classes seem to be fairly good when one dynamic is more dominant than the other. Some other frameworks cannot be linked back to purely continuous or discrete systems. These classes are generally current developments and both dynamics are treated in a more balanced way. Certainly, there is still some improvement needed, however most current applications are well covered by the current frameworks.

In terms of analysis, a lot of research was carried out for hybrid systems consisting of linear subsystems. This seems to be quite natural since linear theory of continuous or discrete systems is better developed than the theory of nonlinear systems. In particular, a lot of research work deals with stability analysis. Conditions are derived for hybrid systems consisting of purely stable, unstable or a mixture of both for linear subsystems. Usually Lyapunov type of arguments are used to show stability.

As it is already difficult to derive specific conditions for nonlinear continuous or discrete systems it is even more difficult for nonlinear hybrid systems. Therefore hardly any results are found for hybrid systems incorporating general nonlinearities.

Robustness analysis is hardly developed for hybrid systems. This area is quite strong in the field of continuous systems where often frequency methods (Bode, Nyquist etc.) are used.

Some results are found for verification and reachability, which are closely related, since often the verification problem can be stated as a reachabiltiy problem. Algorithms in tools like HyTECH, KRONOS, have been implemented to tackle this problem. However, answers to such problems are by no means trivial and it seems there is room for more research work in this area.

There are some results in other analysis areas like controlablility and observability. However, most results are limited to linear hybrid systems.

The control of systems using hybrid control strategies is already quite mature. Nowadays, it is clear that purely discrete or purely continuous feedback is limited. Results like the one by Brockett (24), showing that nonholonomic integrators can only be stabilized by using hybrid feedback, are more than 2 decades old. Many people are aware by now of hybrid control strategies and their advantages, like robustness in the case of sliding mode control. Recent developments are concerned with heterogenous hybrid control. These controllers make it possible to pursue different objectives for different operating conditions. This is needed for complex tasks like flight control, where controllers pursue different objectives, like take off, hovering and landing. Especially the increase of such complex systems and their automation made the usage of hybrid control indispensable.

Controlling hybrid systems does not necessarily mean that hybrid controllers are involved, but it certainly demands knowledge of hybrid systems theory. Applying continuous control to a hybrid system ultimately raises the questions: does there exist a single, continuous, controller which stabilizes the hybrid system? The area of simultaneous stabilization problem (SSP)

(80) deals with this problem. However, for many hybrid systems which have continuous as well as discrete control inputs it is desirable to use both. Results for such systems are mainly found in the area of optimal hybrid control. Here, a performance index penalizing continuous as well as discrete dynamics is used in order to find the continuous and discrete control inputs which minimizes this index. Unfortunately, solutions are not easy to obtain. However, in cases where the underlying dynamic is linear, solutions are found by mixed integer linear or quadratic programs (MILP), (MIQP). But even in these cases solutions are difficult to obtain since the optimization problems remains non-convex.

9.2 Contributions

The literature survey made clear that there are many open problems and areas which are hardly touched. One of them is the robustness analysis and robust controller design for hybrid systems. Also hardly any results are found in the area of state estimation for hybrid systems, although it plays an important role in practice. An emerging topic is identification of hybrid systems. In this area only few results can be found and they are limited to linear hybrid systems. This thesis makes contributions exactly in these fields, while focusing on a particular class of linear hybrid system with unknown discrete dynamic.

The thesis shows that many real systems can be modelled in this class of linear hybrid system with unknown discrete dynamic. Such systems occur for example through human interaction influencing the discrete behavior of a hybrid system, i.e. on or off switching, gear changing, turning dials etc.. In all these cases the discrete states can be detected immediately but modelling the discrete dynamic is impossible. For such a class of linear hybrid system we derived conditions for robustness analysis and robust controller design in the presence of parameter uncertainties and variations. Since in most models parameters are only known with a certain precision it is inevitable to deal with parameter uncertainties. Parameters can also vary dependent on the operating condition, in addition most systems are actually time variant due to effects of ageing. Therefore, it is important to have an analysis framework in which the uncertainties and variations can be assessed. An iterative procedure has been presented in this thesis to compute robustness intervals by solving signomial programs. If the uncertain and time variant parameters of the hybrid system lie entirely in the robustness interval the system is stable for all possible parameter configurations. Taking this a step further a controller synthesis procedure was presented using this analysis framework. It was suggested that specifications and performance requirements are translated into a nominal system. Around this nominal system a robustness interval is computed using the analysis framework. It is then possible to check if there exists a common state feedback controller which shifts the subsystems of the hybrid system into the robustness interval. An extension to multiple controllers was made for the case that it is impossible to achieve this with a single controller.

Besides robustness to parameter variations the thesis is also concerned with optimal control of linear hybrid systems having unknown discrete dynamics. It is shown how an optimal controller can be constructed which is robustly stable to switching among the subsystems and limiting the cost at the same time. Looking at our class of hybrid system it is immediately clear that there is not an associated cost given an initial condition. This is due to the fact that an infinite number of different switching sequences can be chosen. Therefore, only a bound on the optimal cost can be computed. It is shown that computing an upper bound, implying the worst case switching sequence, can be cast into a convex optimization problem in form of an LMI. Solving the LMI gives the performance bound as well as a controller which is robustly stable against all switching sequences. While implying the best case switching sequence which yields a lower bound on the performance. It has been shown that the lower bound can be obtained by solving an LMI as well. Since most controllers are implemented on computers it is important to have discrete time realizations. Therefore, the synthesis procedure was not only stated for hybrid systems in continuous time but also for hybrid systems in discrete time. In this way, a direct implementation on a computer is possible. Since all the proposed control laws require knowledge of the states an observer design was proposed for this class of hybrid systems. This observer converged for all possible switching strategies while minimizing the covariance of the estimation error. To obtain the observer-gains in a convenient way, the observer synthesis was cast into an LMI. In this way, an observer can be obtained which is the hybrid version of the Kalman filter.

The last section of the thesis was concerned with the identification of linear hybrid systems. Identification is necessary to obtain models for analysis and controller design in cases where no analytic model can be derived. In such a case, the system is excited with an input signal and the output is measured. After sufficient input/output data is collected, models are fit using optimization.

Identification of hybrid systems has to solve three problems. The first problem is the classification problem, i.e. which input/output data pair belongs to which subsystem. The second problem is the regression problem, reconstructing the individual dynamic of each subsystem from the input/output data. The third problem is to reconstruct the domain of each subsystem. In chapter 8 these steps are discussed in detail and theory is developed to support this methodology.

9.3 Open problems

In this section we discuss some open problems, which appear to be interesting.

There are various robustness issues which remain open. Besides the robustness analysis in this thesis which is mainly concerned with robustness to parameter uncertainties and variations or with robustness to arbitrary switching there are still open questions: what is the effect of measurement noise and errors in hybrid systems? Can we use switching to suppress noise or detect errors?
In this thesis it was assumed that changes of subsystems can be detected immediately. If this is not the case, what are the consequences of delayed detection? Are hybrid systems sensitive to delayed detection of subsystem changes? It would be certainly useful to compute maximal delays for subsystem change detection which still guarantees stability. Perhaps a ratio between delayed detection and the active time of a subsystem could be derived.

It would be also sensible to extend the class of linear hybrid system with unknown or nondeterministic discrete dynamic, to cases where distributions of the active times of each subsystem can be obtained. Various applications could be modelled in such a framework, gear changes in a car for example. For such a class of hybrid system with stochastic discrete dynamic one could hope to reduce the conservativeness of the current approaches. For instance, less conservative stability results could be obtained, this could also lead to larger robustness radii etc.. In the field of optimal control for hybrid systems with stochastic discrete dynamics one could obtain less conservative bounds than we obtained in chapters 5 and 6 for hybrid systems with unknown discrete dynamics. It seems sensible to investigate such hybrid systems with stochastic behaviour in general, since there are only few contributions in this field.

It was shown in chapter 7 that the asymptotic Kalman filters cannot handle hybrid systems in general. However, it remains open if the time-varying Kalman filter can overcome such problems or if it will diverge for some periodic switching sequences too. If it does not guarantee convergence of the estimate it might have a big impact on many current applications which use time-varying Kalman filters.

In the field of identification there seem to be many interesting problems open. One might investigate how to generate suitable stimuli for hybrid identification, in order to simulate all domains sufficiently. The proposed procedure relies on various parameters like c (chapter 8), the number of input/output data pairs collected in a local data set. One might find a procedure to determine optimal quantities for these parameters. It also seems to be sensible to investigate different methods for obtaining the number of subsystems. Also the order of each subsystem needs to be detected.

Part VI

Appendix

Bibliography

[13] R. Alur and D. Dill. A theory of timed automata. *Theoretical Computer Science*, 126:183–235, 1994.

[14] A. Bemporad, F. Borrelli, and M. Morari. Optimal controllers for hybrid systems: Stability and piecewise linear explicit form. In *Proceedings of Conference on Decision and Control*, Sydney, 2000.

[15] A. Bemporad, F. Borrelli, and M. Morari. Piecewise linear optimal controllers for hybrid systems. In *Proceedings of American Control Conference*, Chicago, 2000.

[16] A. Bemporad, G. Ferrari-Trecate, and M. Morari. Observability and controllability of piecewise affine and hybrid systems. *IEEE Trans. on Automatic Control*, 45(10):1864–1876, October 2000.

[17] A. Bemporad and M. Morari. Control of systems integrating logic, dynamics, and constraints. Technical Report AUT-98-04, Institut fuer Automatik, ETH Zuerich, 1998.

[18] A. Bemporad and M. Morari. Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35:407–527, 1999.

[19] F. Borrelli, A. Bemporad, M. Fodor, and D. Hrovat. A hybrid approach to traction control. In *Proc. 4th International Workshop on Hybrid Systems: Computation and Control*, Roma, 2001.

[20] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*, volume 15 of *Studies in Applied Mathematics*. SIAM, Philadelphia, PA, June 1994.

[21] M.S. Branicky. Multiple lyapunov functions and other analysis tools for switched and hybrid systems. *IEEE Trans. on Automatic Control*, 43(4):475–482, April 1998.

[22] M.S. Branicky, V.S. Borkar, and S.K. Mitter. A unified framework for hybrid control: Background, model, and theory. *IEEE Trans. on Automatic Control*, 43(1):31–45, January 1998.

[23] M.S. Branicky, R. Hebbar, and G. Zhang. A fast matching algorithm for hybrid systems. In *Proceedings of Conference on Decision and Control*, Phoenix, 1999.

[24] R.W. Brockett. *"Asymptotic Stability and Feedback Stabilization," in Differential Geometrics in Control Theory (R. W. Brockett et al., eds.)*, volume 27. Birkhauser Publishers, Boston, MA, 1983.

[25] A. Cervin. Towards the integration of control and real-time scheduling design. Technical Report TFRT-3226, Box 118, S-221 00 Lund, Sweden, 2000. Licentiate thesis.

[26] R. David and H. Alla. Tetri nets for modellinng of dynamic systems a survey. *Automatica*, 30(2):175–202, 1994.

[27] K. Edstrm. *Switched Bond Graphs Simulation and Analysis.* PhD thesis, Dept. of Electrical Engineering, University of Linkping, SE-581 83 Linkping, Sweden, 1999.

[28] J. Eker. *Flexible Embedded Control Systems.* PhD thesis, Dept. of Automatic Control, Lund Institute of Technology, Box 118, S-221 00 Lund, Sweden, 1999.

[29] J. Ezzine and A.H. Haddad. Controllability and observability of hybrid systems. *Int. J. Control*, 49:2045–2055, June 1989.

[30] G. Ferrari-Trecate, D. Mignone, and M. Morari. Moving horizon estimation for hybrid systems. *IEEE Trans. on Automatic Control*, 2002.

[31] G. Ferrari-Trecate and M. Muselli. A new learning method for piecewise linear regression. In *Proceedings of International Conference on Artificial Neural Networks (ICANN02)*, Madrid, Spain, 2002.

[32] G. Ferrari-Trecate, M. Muselli, D. Liberati, and M. Morari. A clustering technique for the identification of piecewise affine systems. In M. Di Benedetto and A. Sangiovanni-Vincentelli, editors, *Proc. 4th International Workshop on Hybrid Systems: Computation and Control*, volume 2034 of *Lecture Notes in Computer Science*, pages 218–231. Springer-Verlag, 2001.

[33] G. Ferrari-Trecate, M. Muselli, D. Liberati, and M. Morari. Identification of piecewise affine and hybrid systems. In *Proceedings of American Control Conference*, pages 3521–3526, Arlington, 2001.

[34] G. Ferrari-Trecate, M. Muselli, D. Liberati, and M. Morari. A clustering technique for the identification of piecewise affine systems. *Automatica*, 2002.

[35] G. Ferrari-Trecate and M. Schinkel. Identification of piecewise affine systems: Sufficient conditions for perfect data clustering. *IEEE Trans. on Automatic Control.* submitted.

[36] B. Fritzke. Growing cell structures - a self-organizing network for unsupervised and supervised learning. *Neural Networks*, 1994.

[37] A. Hassibi and S.P. Boyd. Quadratic stabilization and control of piecewise-linear systems. In *Proceedings of American Control Conference*, volume 6, pages 3659–64, Philadelphia, PA, June 1998.

[38] S. Hedlund and A. Rantzer. Optimal control of hybrid systems. In *Proceedings of Conference on Decision and Control*, Phoenix, 1999.

[39] W.P.M.H. Heemels. *Linear Complementarity Systems.* PhD thesis, Technical University Eindhoven, 1999.

[40] W.P.M.H. Heemels, B. De Schutter, and A. Bemporad. Equivalence of hybrid dynamical models. *Automatica*, 37(7):1085–1091, July 2001.

[41] W. Holderbaum and M. Schinkel. On stability of nondeterministic hybrid systems. In *Proceedings of Wismarer Automatisierungssymposium*, Wismar, 2002.

[42] K. Hunt, Y. Wang, M. Schinkel, and T. Schmitt-Hartmann. *Controller design for hybrid systems using simultaneous D-stabilisation technique and its application to Anti-lock Braking systems (ABS).* "Nonlinear and Hybrid Control in Automotive Applications". Springer-Verlag, London, 2002.

[43] T.A. Johansen, I. Petersen, and O. Slupphaug. On explicit suboptimal lqr with state and input constraints. In *Proceedings of Conference on Decision and Control*, Sydney, 2000.

[44] T.A. Johansen, I. Petersen, and O. Slupphaug. Explicit subopitmal linear quadratic regulation with input and state constraints. *Automatica*, 2002.

[45] K.H. Johansson, M. Egerstedt, J. Lygeros, and S. Sastry. On the regularization of zeno hybrid automata. *System & Control Letters*, 38:141–150, 1999.

[46] K.H. Johansson, J. Lygeros, S. Sastry, and M. Egerstedt. Simulation of zeno hybrid automata. In *Proceedings of Conference on Decision and Control*, Phoenix, 1999.

[47] M. Johansson. *Piecewise Linear Control Systems.* PhD thesis, Dept. of Automatic Control, Lund Institute of Technology, Box 118, S-221 00 Lund, Sweden, 1999.

[48] M. Johansson and A. Rantzer. Computation of piecewise quadratic lyapunov functions for hybrid systems. *IEEE Trans. on Automatic Control*, 43(4):555–559, April 1998.

[49] R. Kalman. Phase-plane analysis of automatic control systems with nonlinear gain elements. *Trans. AIEE Part II: Applications and Industry*, 73(4):383–390, April 1954.

[50] G. Lafferriere, G.J. Pappas, and S. Sastry. Reachablility analysis of hybrid systems using bisimulations. In *Proceedings of Conference on Decision and Control*, Tampa, Florida, USA, 1998.

[51] D. Liberzon and A.S. Morse. Basic problems in stability and design of switched systems. *IEEE Control Systems Magazine*, October 1999.

[52] J.S. Lin. *Nonlinear and adaptive control design of active suspensions.* PhD thesis, University of California, Los Angeles, 1997.

[53] J. Lygeros, K.H Johansson, M. Egerstedt, and S. Sastry. On the existence of executions of hybrid automata. In *Proceedings of Conference on Decision and Control*, Phoenix, 1999.

[54] J. Lygeros, C. Tomlin, and S. Sastry. On controller synthesis for nonlinear hybrid systems. In *Proceedings of Conference on Decision and Control*, Tampa, Florida, USA, 1998.

[55] J. Lygeros, C. Tomlin, and S. Sastry. Controllers for reachablity specifications for hybrid systems. *Automatica*, pages 349–370, March 1999.

[56] X.-J. Ma, Sun, and Y. He. Analysis and design of fuzzy controller and fuzzy observer. *IEEE Trans. Fuzzy Systems*, 6(1):41–51, Feb. 1998.

[57] J. Malmborg. *Analysis and Design of Hybrid Control Systems.* PhD thesis, Dept. of Automatic Control, Lund Institute of Technology, Box 118, S-221 00 Lund, Sweden, 1998.

[58] K.S. Narendra and J. Balakrishnan. A common lyapunov function for stable lti systems with commuting a-matrices. *IEEE Trans. on Automatic Control*, 39:2469–2471, 1994.

[59] S. Pettersson. *Analysis and Design of Hybrid Systems.* PhD thesis, Chalmers University of Technology, 1999.

[60] A. Rantzer and M. Johansson. Piecewise linear quadratic optimal control. In *Proceedings of American Control Conference*, Albuquerque, 1997.

[61] S. Sastry, G. Meyer, C. Tomlin, J. Lygeros, D. Godbole, and G. Pappas. Hybrid systems in air traffic control. In *Proceedings of Conference on Decision and Control*, New Orleans, 1995.

[62] C. Scherer and S. Weiland. *Course on Linear Matrix Inequalities in Control.* Dutch Institute of Systems and Control.

[63] M. Schinkel. A robust and performance orientated controller design approach for non-deterministic hybrid systems. In *Proceedings of European Control Conference*, Porto, 2001. Student session.

[64] M. Schinkel, W. H. Chen, and A. Rantzer. Optimal control for systems with varying sampling rate. In *Proceedings of American Control Conference*, Anchorage, 2002.

[65] M. Schinkel and K. Hunt. Anti-lock braking control using a sliding mode like approach. In *Proceedings of American Control Conference*, Anchorage, 2002.

[66] M. Schinkel and A. Rantzer. Sample data control with varying sampling time. In *Proceedings of European Control Conference*, Porto, 2001.

[67] M. Schinkel, Y. Wang, and K. Hunt. Stable and robust state feedback design for hybrid systems. In *Proceedings of American Control Conference*, Chicago, 2000.

[68] M. Schinkel, Y. Wang, and K. Hunt. Control of uncertain and parameter-variant piece-wise linear systems. *International Journal of Control*, 2002.

[69] J.S. Shamma and M. Athans. Guaranteed properties of gain scheduled control for linear paramete varying plants. *Automatica*, 27:559–564, 1991.

[70] R.N. Shorten. *A Study of Hybrid Dynamical Systems with Application to Automobile Control.* PhD thesis, Dept. Electrical and Electorinic Engineering, Universtiy College, Dublin, 1996.

[71] R.N. Shorten and K. S. Narendra. On the stability and existence of common lyapunov functions for stable linear switching systems. In *Proceedings of Conference on Decision and Control*, Tampa, Florida, USA, 1998.

[72] R.N. Shorten and F. O'Cairbre. On the global attractivity of a class of switching system using a non-quadratic lyapunov approach. In *Proceedings of American Control Conference*, Chicago, 2000.

[73] J.J.E. Slotine and W. Li. *Applied Nonlinear Control.* Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1991.

[74] K.J. Åström and B. Wittenmark. *Computer-Controlled Systems.* Third Edition. Prentice Hall, New Jersey, NJ, 1997.

[75] M. Tittus and B. Egardt. Control design for integrator hybrid systems. *IEEE Trans. on Automatic Control*, 43(4):491–500, April 1998.

[76] A.G. Ulsoz and D. Hrovat. Stability robustness of lqg active suspensions. In *Proceedings of American Control Conference*, San Diego, 1990.

[77] V.I. Utkin. Variable structure systems with sliding modes. *IEEE Trans. on Automatic Control*, 22:212–222, 1977.

[78] A.J. van der Schaft and J.M. Schumacher. The complementary-slackness class of hybrid systems. *Mathematics of Control, Signals and Systems*, 9:266–301, 1996.

[79] Y. Wang, M. Schinkel, and K. Hunt. An engineering approach to simultaneous stabilisation and strong simultaneous stabilisation with d stability. In *Proceedings of Conference on Decision and Control*, Tampa, 2001.

[80] Y. Wang, M. Schinkel, T. Schmitt-Hartmann, and K. Hunt. Pid and pid-like controller design by pole assignment within d-stable regions. *Asian Journal of Control*, 2002.

[81] Y. Wang, T. Schmitt-Hartmann, M. Schinkel, and K. Hunt. A new approach to simultaneaous stabilisation with d-stability and its application to control of antilock braking systems. In *Proceedings of European Control Conference*, Porto, 2001.

[82] M.A. Wicks, P. Peleties, and R.A. DeCarlo. Switched controller synthesis for the quadratic stabilization of a pair of unstable linear systems. *European Journal of Control*, 44(4):140–147, April 1998.

[83] X. Xu and P. J. Antsaklis. Design of stabilizing control laws for second-order switched systems. In *Proceedings of the IFAC World Congress*, Beijing, 1999.

[84] G. Zhai, Bo Hu, K. Yasuda, and A.N. Michel. Stability analysis of switched systems with stable and unstable subsystems: An average dwell time approach. In *Proceedings of American Control Conference*, Chicago, 2000.