UNIVERSITY OF SOUTHAMPTON

FACULTY OF ENGINEERING, SCIENCE AND MATHEMATICS
School of Electronics and Computer Science

Analysis and Design of Classes of Hybrid Control Systems

by

Sallehuddin Mohamed Haris

Thesis for the degree of Doctor of Philosophy

September 2005

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING, SCIENCE AND MATHEMATICS
SCHOOL OF ELECTRONICS AND COMPUTER SCIENCE

Doctor of Philosophy

ANALYSIS AND DESIGN OF CLASSES OF HYBRID CONTROL SYSTEMS

by Sallehuddin Mohamed Haris

This thesis considers the problem of designing stabilizing controllers for hybrid systems. Generally speaking, the term hybrid systems refers to systems that exhibit interaction between continuous-time dynamics and logical events. The class of hybrid systems studied here are plants whose dynamics switch between several linear models. Control systems analysis and design is a non-trivial problem as established methods applicable to continuous-time and discrete-time systems cannot, in general, be extended to hybrid systems. This research focuses on two themes: the application of model reference adaptive control (MRAC) to hybrid systems and the stabilizeability of hybrid systems. In the first of these, we study the performance of output feedback against state feedback and free running against resettable adaptive control for single input single output (SISO) systems. Simulations reveal that output feedback and free running adaptive schemes are incapable of producing stabilizing control in some cases of hybrid systems. Resettable state feedback MRAC, implemented through a multiple model adaptive control structure, was found to provide very good results where the plant and reference outputs approached convergence, even in the presence of some perturbation. The problem of extending the method to multivariable hybrid systems is also investigated. Here, successful implementation of the scheme was found to depend on the decoupleability of both the plant and reference models. We propose guidelines on how this constraint may be overcome. In the second theme, the objective was to identify the stabilizeability of hybrid systems through determination of the existence of control parameters such that all subsystems have a common Lyapunov function. Our work focussed on SISO switching systems with two continuous-time states. For systems modelled in continuous-time, we have derived a method from which this is achieved. A similar method for sampled-data systems has also been found, provided that all subsystems are modelled in the Brunovsky form.

Contents

List of Figures

List of Tables

Acknowledgements

First and foremost, I would like to express my utmost gratitude to Prof. Eric Rogers who supervised me on this research project. Without his patient guidance and support, this thesis would never have materialized. A word of thanks to Wojciech Paszke of *Uniwersytet Zielonogórski* who taught me how to use the Matlab LMI toolbox. Also, I would like to express my appreciation to the administration and technical support staff at ISIS for all their help.

Special thanks are reserved for my family. To my wife, Marlina for her love, encouragement and patient support, especially whilst I was under the high stresses of writing this thesis. Likewise, to my parents, brother and sister, all of whom also gave support, encouragement and wise words.

Not to forget, my gratitude to the Government of Malaysia and *Universiti Kebangsaan Malaysia* who approved my scholarship and study leave to pursue this PhD, and to the taxpaying citizens of Malaysia whose money made it all possible.

Chapter 1

Introduction

In recent times, there has been an increasing interest in the study of hybrid dynamical systems. Hybrid systems arise from interaction between continuous-time dynamics and discrete event systems. In the past, continuous-time and discrete event dynamics were studied separately, the former using differential or difference equations and the latter through automata and Petri net models, for example. For systems where the two dynamic components are tightly coupled, such an approach is not adequate to fully describe the system behaviour. Consequently, the system will not be optimized to meet high performance specifications.

1.1 Defining Hybrid Systems

The distinguishing feature of hybrid dynamical systems is the tight interaction between continuous-time and discrete event dynamics. In the following we illustrate what we mean by this.

1.1.1 Continuous-time Dynamical Systems

In the study of hybrid dynamical systems, little distinction is made between continuous-time and discrete-time dynamics. For brevity, both are considered to be continuous-time systems. In most cases, mathematical models of dynamical systems are represented in the form of differential (for continuous-time systems) or difference equations (for discrete-time systems).

A mass-spring-damper system as shown in Figure 1.1 is an example of a continuous-time dynamical system. If the system is taken to be linear and time-invariant, it can be represented by a continuous-time model in the form of an ordinary differential equation with constant coefficients:

$$m\ddot{x}(t) + b\dot{x}(t) + kx(t) = F(t), \quad x(t_0) = x_0. \tag{1.1}$$

Here, m is the mass, b is the damper coefficient and k is the spring constant. $F(t)$ is an external force acting on the system, $x(t)$ is the displacement from an initial position x_0 taken at time t_0 and $\dot{x}(t)$ and $\ddot{x}(t)$ are respectively, the first and second derivatives of x with respect to time.

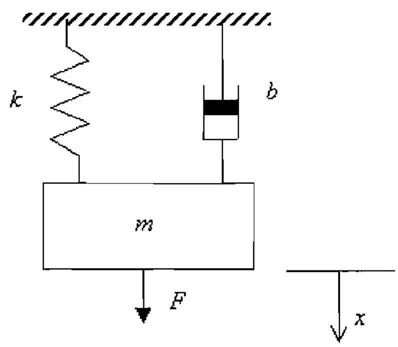


FIGURE 1.1: A mass-spring-damper system

In classical control, Equation (1.1) is often written as a *transfer function* which takes the following form:

$$\frac{X(s)}{F(s)} = \frac{1}{ms^2 + bs + k}. \tag{1.2}$$

Here, s is the Laplace variable and $X(s)$ and $F(s)$ are the Laplace transforms of x and f respectively. When formulating dynamical system models in the transfer function form, the initial condition is taken to be zero. By multiplying both sides with $F(s)$ and taking inverse Laplace transforms, Equation (1.2) provides a way of calculating the system response $x(t)$ for a given input $F(t)$.

Alternatively, dynamical systems can also be represented in the *state space* form

$$\dot{x} = Ax + Bu,$$
$$y = Cx$$

where x is the state vector, u is the input vector, y is the output vector, A is the system matrix, For the system of Figure 1.1, taking $x_1 = x$ and $x_2 = \dot{x}$, the state space model is

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -k/m & -b/m \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1/m \end{bmatrix} F,$$
$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}. \tag{1.3}$$

The advantage of using the state space form is that it allows even very large dynamical systems to be analysed through the use of computer programs.

The objective for control theorists is to shape the transient and steady-state responses of a dynamical system so as to achieve a given desired performance. The control system should also be sufficiently robust such that the performance is maintained even in the presence of disturbances and unmodelled dynamics. There are numerous literature that provide comprehensive treatment of continuous-time dynamical systems, for example Ogata (1998) and Vu and Esfandiari (1998).

1.1.2 Discrete Event Systems

A system can be defined as a group of interacting, interrelated, or interdependent elements forming a complex whole. Often the elements interact to accomplish a set of specific functions. The accomplishment of such functions require logical behaviour or functionality between components within the system. In a manufacturing cell for example, the arrival of a workpiece at a particular machine would signal the machine to start operation. When the job required on the workpiece has been completed, the machine would stop and the workpiece would move on to the next machine in the cell. To ensure the cell functions correctly, it needs to be supervised, either by human operators or by an automated supervisory control system.

With rapid advancements in computer technology, automated computer control of complex systems are increasingly being considered. In view of this, there is a necessity to formalize logical behaviour into mathematical models. This is the goal of *discrete event systems* (DES) modelling. Several methods of modelling DES have been developed. Among these methods are finite state automata, Petri nets and statecharts.

DES models are often visualized graphically. Typically, the system is defined by a set of discrete states which represent an activity or mode of operation. The occurrence of an event leads to a change in the current discrete state. A set of rules defines when the events occur and how the discrete state changes. Figure 1.2 (from Montoya and Boel (2001)) is an example of an automaton representation of the functions a machine.

An automaton is defined by the tuple $G = (Q, E, \delta, D, q_{init}, Q_m)$ where Q is the set of all possible states, E is the event set, $\delta : Q \times E \rightarrow Q$ is the transition function, $D : Q \rightarrow 2^E$ is the active event function, $q_{init} \in Q$ is the initial state and $Q_m \subseteq Q$ is the set of marked states. For the example of Figure 1.2, $Q = \{q_0, q_1, q_2, q_3\}$, $E = \{start, stop, breakdown, repair\}$ and $q_{init} = q_0$. If the current state is q, upon occurrence of event e, the next state will be $\delta(q, e)$. In the example, $\delta(q_0, start) = q_1$ and $\delta(q_1, breakdown) = q_2$. D gives the set of events that are allowed to happen given a particular state. Again, in the example, $D(q_1) = \{stop, breakdown\}$ and $D(q_3) = \{start, repair\}$. Typically, a marked state $q \in Q_m$ is a state $q \in Q$ that has completed a specific task.

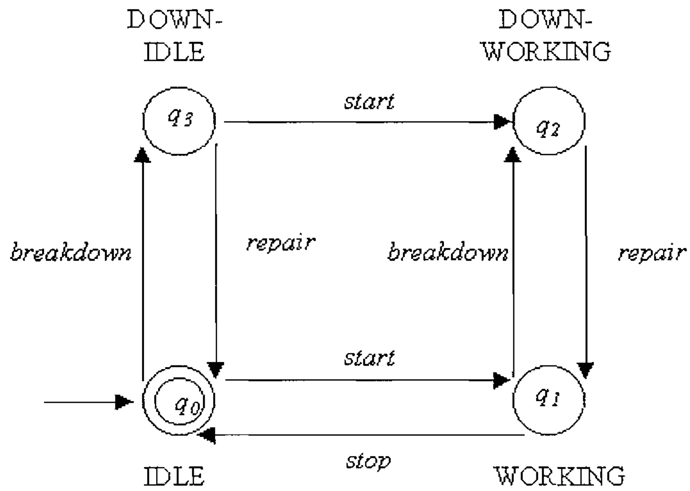


FIGURE 1.2: An automaton describing the functions of a machine (Montoya and Boel, 2001)

In designing DES, it must be ensured that deadlocks or livelocks will not occur, that variables will always remain within a specified safe set and that all tasks will eventually come to a completion. Supervisory control is usually implemented to guarantee these. The interested reader may refer to Montoya and Boel (2001) and the references therein for a more detailed introduction to DES.

1.1.3 Hybrid Dynamical Systems

The term *hybrid dynamical systems* refers to systems that exhibit continuous-time dynamics interacting with logical events. Hybrid systems operate in several different modes where, in each mode, the system obeys a particular set of continuous dynamic equations. The occurrence of a logical event would cause the system to switch from one mode of operation, where the system follows one set of dynamic equations, to a different mode, where another set is obeyed. Figure 1.3 illustrates the state trajectory for a system exhibiting such a phenomenon.

The phenomena of switching between modes can be caused by several factors. It may occur naturally if the plant operates in several different discontinuous modes such as in relays and saturations. It may be caused by the controller, in order to accommodate constraints such as in anti-windup systems and in model predictive control, or in order to implement a sequence such as in programmable logic control. It may also be caused by a simplification of the model, for example when approximating nonlinear phenomenon by piecewise linear functions.

There are many approaches to modelling of hybrid dynamical systems. The vast majority of these approaches are based on a combination of state space and DES models. The

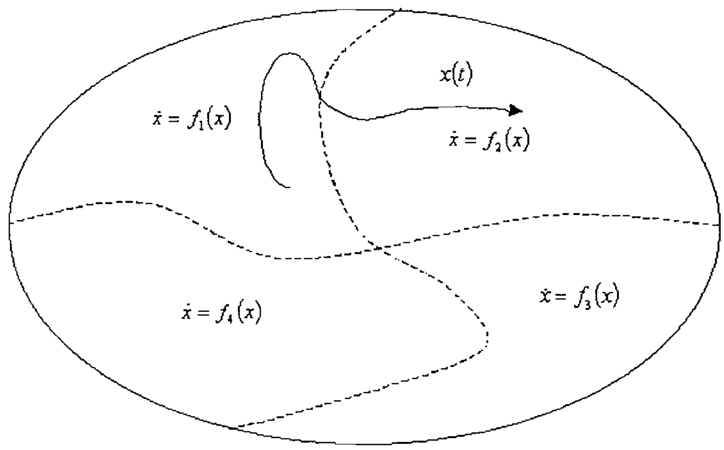


FIGURE 1.3: State trajectory of a hybrid system

balance between the two vary according to applications and objectives. In cases where complexity of the logical structure is the main concern, the model would be more DES orientated. On the other hand, when dynamic response of the system is the dominant issue, the model would typically be biased towards the state space form.

1.2 Motivational Examples

There are several reasons why the study of hybrid systems control has become an active area of research. One reason is that hybrid systems encompass a wide range of engineering systems. In fact, it could be said that hybrid systems are the most general form of dynamical systems, and continuous-time and discrete event systems (DES) are special cases of hybrid systems. Another reason lies in the importance of its applications. The hybrid systems approach has been found to be useful in improving performance in a wide range of applications. The formalism provides a more accurate model for complex systems that in the past were approximated by either strictly continuous or strictly DES models. Examples of complex systems that would benefit from a hybrid systems approach include manufacturing processes, automotive engine control, robotics, road traffic control systems, chemical process plants and air traffic control. Another reason for using the hybrid systems approach is to reduce complexity. A complex system can be abstracted to different levels of abstraction to become more manageable. For example, continuous dynamics could be abstracted at the lower level and logical events at the higher level of a hierarchical control system. Another example is when a nonlinear system is broken up into piecewise linear equations. Below, we provide a few motivating examples on how the hybrid systems approach could be beneficial.

Example 1.1. *Hybrid control of internal combustion engines (Balluchi et al., 1999).*

Increasing environmental awareness and tighter controls on pollution emissions have led to a need for engines with reduced emissions and better fuel consumption. To attain these goals, new control methodologies are being looked into to reduce emissions of internal combustion (IC) engines without compromising on power and performance. Typical automotive IC engines operate periodically in several distinct phases. It has been suggested that such engines should be modelled as hybrid systems where the torque generation process is a discrete-event system while the powertrain behaviour observes continuous dynamics. □

Example 1.2. *Modelling of switching power converters (Escobar et al., 1999).*

Power converters can be regarded as a set of voltage or current sourced subsystems interconnected through switches, whose purpose is to enable energy to be transferred from one subsystem to another. Each subsystem is made up of passive elements (such as inductors, capacitors and resistances), power sources and a load. The discontinuous elements found in these power converters make them operate in several distinct modes. In the past, since they operate at very high frequencies, the dynamics of these converters have been ignored. However, with the ever increasing demands on higher bandwidths and stiffer constraints on harmonic generation, they can no longer be ignored. Consequently, these switching power converters can best be represented by hybrid system models. □

Example 1.3. *Automated highway systems (Lygeros et al., 1998).*

A solution that has been proposed to overcome traffic congestion on highways is to make the highway fully automated. An automated highway system can be viewed as a complex system that can be abstracted to different levels of complexity, controlled by a hybrid hierarchical structure. At the lower level, continuous feedback control laws are used to regulate and track vehicle movements. Supervision is carried out at higher levels of the control hierarchy to ensure optimum use of available road space without neglecting safety of the road users. □

1.3 Scope of Research

In this research, we study hybrid systems where the plant dynamics undergo some sort of switching behaviour. The switch in plant dynamics may be caused by external or internal events. By an external event, we mean an event that can occur randomly without any prior knowledge of the instance of switching. Since the exact moment of switching is not known, some form of mechanism is required to detect the occurrence of a switch. In which case, implementation of the required corrective action may be delayed. An example of such behaviour is when a component of a complex system breaks down.

On the other hand, by "switching due to internal events", we refer to systems where the plant's switching behaviour is determined by its intrinsic properties such as its states.

By tracking the states, the moment of switching is immediately known and corrective action can be applied instantly. As an example, in an automatic gearbox, the amount of torque generated determines which gear is selected. For each gear selected, the plant follows a different dynamical behaviour.

For the class of hybrid systems mentioned above, our research can be divided into two parts. In the first part, we study the possibility of controlling the hybrid plant such that its output follows that of a reference model. We attempt to accomplish this through the use of a combination of fixed and tunable controllers. Our study includes single input single output (SISO) and multiple input multiple output (MIMO) hybrid plants. In the second part of the research, we study the problem of finding stabilizing control laws for a class of SISO hybrid systems. Throughout this thesis, all simulation results were obtained using Matlab 6.5 and Simulink 5.0 running on a Pentium 4, 2GHz computer.

1.4 Contributions to Knowledge

Contributions to knowledge from this thesis can be summarized as follows:

1. An outline of important considerations when implementing direct model reference adaptive control to hybrid systems have been identified. A point of particular interest is that in some cases, free running adaptive control becomes unstable after a switch in plant dynamics has occurred. The inclusion of such a controller therefore provides no guarantee of convergence between plant and reference model responses. On the other hand, good results can be expected from resettable state feedback adaptive controllers.

2. For MIMO hybrid systems, in addition to the points outlined for the SISO case, the following points need also be given attention:

    - All subsystems of the hybrid plant must be decoupleable. Alternatively, pre-compensators to decouple the subsystems could be considered.
    - The reference model should be decoupled and have the plant's interactor matrix as a factor. As the subsystems may not have a common interactor matrix, the use of switchable reference models may be required.

3. A method from which control laws that guarantee stability of a class of hybrid systems (if one exists) has been developed for systems modelled in continuous-time. A similar method for systems modelled in discrete-time has also been developed, provided that the subsystems are in the Brunovsky form.

1.5 Outline of Thesis

This thesis is outlined as follows. In Chapter 2, we review modelling, control and stability issues in hybrid dynamical systems. In Chapter 3, we attempt to control SISO hybrid systems such that the plant output tracks the performance of a reference model. Then, in Chapter 4, we look into issues that need to be addressed in order to extend the idea to MIMO systems. In Chapter 5, the issue of finding control laws that guarantee the stability of plants with switching dynamics is addressed. We end this thesis in Chapter 6 with a summary of our findings and discuss directions for further research.

Chapter 2

Modelling and Control of Hybrid Systems

Hybrid systems have dynamics that are complex and cannot be adequately addressed using methods from linear systems theory. In this chapter, we examine issues concerning modelling, stability and control of hybrid dynamical systems. In Section 2.1, we highlight the many modelling frameworks that have appeared. This is followed, in Section 2.2, by an introduction to the class of piecewise affine systems which is a modelling formalism that has gained popularity among researchers because of its applicability to a wide range of system types. In Section 2.3 we demonstrate how stability concepts from linear systems theory fail to correctly predict the stability of hybrid systems. We also discuss problems in controlling hybrid systems and highlight several methods that have been proposed to tackle the issue. In Section 2.4, we discuss basic concepts of Multiple Model Switching and Control, which is an adaptive type of control that is potentially applicable to practical hybrid systems.

2.1 Modelling Hybrid Systems

Various methods have been taken by researchers in modelling hybrid dynamical systems. In general, most of these approaches are either inclined towards formalisms from discrete event systems (DES) or follow the differential or difference equation models of continuous-time dynamical systems.

Researchers who favour the former are more concerned with the complexity of the logical structure of the system. They focus on issues such as compositionality of the system model, verification and reachability analysis of the states and ensuring that deadlocks and livelocks do not occur. Within this group, hybrid systems are typically modelled based on *Petri net* and *finite automaton* theories. Examples of of this approach include the *hybrid automata* and *event flow formulae* models (van der Schaft and Schumacher, 2001) and the method used by Raisch and his co-workers where the continuous states are abstracted to form a totally DES representation (Moor and Raisch, 1998, 1999b,a; Moor et al., 2001, 2002; Raisch and O'Young, 1997, 1998; Raisch, 2000). In the second group are researchers who concentrate on the continuous-time aspect of the hybrid system. Researchers in this category focus on the transient and steady-state performance as well as on the controllability, observability and stabilizeability of the hybrid system. Some of the formalisms that have been used to model hybrid systems include *linear complementarity systems (LC)* (van der Schaft and Schumacher, 1998), *extended linear complementarity systems (ELC)* (de Schutter and de Moor, 1999), *max-min-plus-scaling systems (MMPS)* (de Schutter and van den Boom, 2001), *mixed logical dynamical systems (MLD)* (Bemporad and Morari, 1999) and *piecewise affine systems (PWA)* (Sontag, 1996).

Other approaches to hybrid system representation have also appeared. One is the method used by Grossman and Larson (1995), where the hybrid system is modelled in the form of algebraic equations. This opens up the possibility of using algebraic tools in the analysis. Another method is to encode the hybrid behaviour implicitly into the programming language without having any explicit model of the system. An example of this is the approach of Benveniste and le Guernic (1990) and Benveniste (1998). For an overview to the many different approaches to modelling hybrid systems, the reader is referred to Labinaz et al. (1997), Boel et al. (1999), Antsaklis and Koutsoukos (2002) and the references therein.

While numerous types of models have been proposed for hybrid dynamical systems, in this thesis, we will only examine the PWA representation. Our interest in PWA systems comes from the fact that the representation can sufficiently model a large number of physical processes, such as systems with static nonlinearities, and can also provide good approximation of nonlinear dynamics through multiple linearizations at different operating points (Bemporad et al., 2000). Several control synthesis techniques have also been suggested for such systems, for example (de Schutter and van den Boom, 2004; Rodrigues and How, 2003; Bemporad et al., 2002). Furthermore, it has been shown that LC, ELC, MMPS, MLD and PWA systems are equivalent (Heemels et al., 2001; Bemporad et al., 2002). In a recent paper (Potocnik et al., 2004), a technique that enables the translation of hybrid automata into PWA systems has been proposed. Hence, methods developed for PWA systems can be adapted for use on a wide range of hybrid system representations.

2.2 Piecewise Affine Systems

Piecewise Affine Systems (PWA) has its roots in the *Piecewise Linear* (PWL) approach which was introduced in (Sontag, 1981) as a theoretical foundation to regulation of nonlinear systems. The extension of this concept to hybrid systems applications is summarized in (Sontag, 1996). This formalism models a class of hybrid systems where the dynamic equations and switching rules are linear functions of the state space. In this representation, the system's state space is first partitioned into a finite number of polyhedral regions. Then, to each polyhedron, a different affine dynamics is assigned.

PWA system models are represented as in Equation (2.1) (Bemporad et al., 2000)

$$\begin{aligned} x(k+1) &= A_i x(k) + B_i u(k) + f_i, \\ y(k) &= C_i x(k) + g_i, \end{aligned} \quad \text{for} \begin{bmatrix} x(k) \\ u(k) \end{bmatrix} \in \chi_i \tag{2.1}$$

where $x(k) \in \mathbb{R}^n$, $u(k) \in \mathbb{R}^m$ and $y(k) \in \mathbb{R}^p$ are the system states, inputs and outputs respectively. $\Sigma_i = \{A_i, B_i, f_i, C_i, g_i\}$, $i = 1, \ldots, s$ defines subsystem i, where f_i and g_i are suitable constant vectors. The cell χ_i is a partition of $\mathbb{R}^n$ in which Σ_i is defined. From here, it can be seen that the PWA form is simply a composition of linear time-invariant state space models. If f_i and g_i are null, then the PWA model becomes a PWL representation.

Ferrari-Trecate et al. (2002) introduced the *PWA logic canonical form* (PWA-LC) which is a variation of the PWA that includes logical states in the representation. The PWA-LC form is as shown by Equation (2.2)

$$\begin{aligned} \begin{bmatrix} x^c \\ x^d \end{bmatrix}(k+1) &= A_i \begin{bmatrix} x^c \\ x^d \end{bmatrix}(k) + B_i \begin{bmatrix} u^c \\ u^d \end{bmatrix}(k) + f_i, \\ \begin{bmatrix} y^c \\ y^d \end{bmatrix}(k+1) &= C_i \begin{bmatrix} x^c \\ x^d \end{bmatrix}(k) + D_i \begin{bmatrix} u^c \\ u^d \end{bmatrix}(k) + g_i, \\ \text{for} \begin{bmatrix} x^c \\ u^c \end{bmatrix}(k) \in \chi_i, \quad x^d &= \tilde{x}_i^d \quad \text{and} \quad u^d = \tilde{u}_i^d \end{aligned} \tag{2.2}$$

where, $x^c \in \mathbb{R}^{n_c}$, $u^c \in \mathbb{R}^{m_c}$, $y^c \in \mathbb{R}^{p_c}$ are continuous states, inputs and outputs respectively and $x^d \in \{0,1\}^{n_d}$, $u^d \in \{0,1\}^{m_d}$, $y^d \in \{0,1\}^{p_d}$ are respectively, their logical counterparts. The subsystem $\Sigma_i = (A_i, B_i, f_i, C_i, D_i, g_i)$, $i = 1, \ldots, s$ have the block structure

$$A_i = \begin{bmatrix} A_i^c & 0 \\ 0 & 0 \end{bmatrix}, \quad B_i = \begin{bmatrix} B_i^c & 0 \\ 0 & 0 \end{bmatrix}, \quad f_i = \begin{bmatrix} f_i^c \\ f_i^d \end{bmatrix},$$

$$C_i = \begin{bmatrix} C_i^c & 0 \\ 0 & 0 \end{bmatrix}, \quad D_i = \begin{bmatrix} D_i^c & 0 \\ 0 & 0 \end{bmatrix}, \quad g_i = \begin{bmatrix} g_i^c \\ g_i^d \end{bmatrix}$$

where $A_i^c \in \mathbb{R}^{n_c \times n_c}$, $B_i^c \in \mathbb{R}^{n_c \times m_c}$, $C_i^c \in \mathbb{R}^{p_c \times n_c}$, $D_i^c \in \mathbb{R}^{p_c \times n_c}$, $f_i^c \in \mathbb{R}^{n_c}$ and $g_i^c \in \mathbb{R}^{p_c}$ are the continuous components and $f_i^d \in \{0,1\}^{n_d}$ and $g_i^d \in \{0,1\}^{p_d}$ are the logical components. The cell $\chi_i \subset \mathbb{R}^{n^c} \times \mathbb{R}^{m^c}$ is the polyhedron in which the subsystem Σ_i is defined. $\bigcup_{i=1}^{s} \chi_i$ defines the admissible continuous states and inputs of the system. To each χ_i is associated a constant logic state vector $\tilde{x}_i^d \in \{0,1\}^{n_d}$ and a constant logic input vector $\tilde{u}_i^d \in \{0,1\}^{m_d}$.

In Equations (2.1) and (2.2) we have kept the discrete-time notation used in the original texts, but in the analyses that follow, unless otherwise stated, subsystems are modelled in continuous-time.

2.3 Control and Stability Issues in Hybrid Systems

Despite the simplicity of the PWA form, control and stability analysis of hybrid systems is not a straight forward matter. The control scheme for a hybrid system must be able to perform the following duties:

1. Ensure that the overall system meets performance specifications.
2. Ensure that the overall system is robust towards noise and unmodelled dynamics.
3. Ensure the stability of the overall system.
4. Ensure that the system does not stray into an unsafe mode.

The first three tasks are common to those of conventional control for continuous-time systems and the fourth is a verification problem that is commonly studied in the realm of discrete event dynamical systems. While established techniques are available for such types of systems, they are not immediately extendable to hybrid dynamical systems.

It is a well known fact that it is possible to have a stable hybrid system totally made out of unstable subsystems. It is also possible to have the converse; an unstable hybrid system completely comprising of stable subsystems. In the following, we show examples of such cases.

Example 2.1. *A stable PWA-LC system comprising of unstable subsystems. This example has been adapted from (Ferrari-Trecate et al., 2002) with a slight variation in the boundaries of regions R_1 and R_2.*

The system is defined by the following subsystem matrices:

$$A_1 = \begin{bmatrix} \cos\left(-\frac{\pi}{8}\right) & -\sin\left(-\frac{\pi}{8}\right) & 0 \\ \sin\left(-\frac{\pi}{8}\right) & \cos\left(-\frac{\pi}{8}\right) & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad B_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad f_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix},$$

$$A_2 = \begin{bmatrix} \cos\left(\frac{\pi}{9}\right) & -\sin\left(\frac{\pi}{9}\right) & 0 \\ \sin\left(\frac{\pi}{9}\right) & \cos\left(\frac{\pi}{9}\right) & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad f_2 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix},$$

$$A_3 = A_4 = \begin{bmatrix} \cos\left(-\frac{\pi}{6}\right) & -\sin\left(-\frac{\pi}{6}\right) & 0 \\ \sin\left(-\frac{\pi}{6}\right) & \cos\left(-\frac{\pi}{6}\right) & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad B_3 = B_4 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad f_3 = f_4 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix},$$

$$A_5 = A_6 = \begin{bmatrix} \cos\left(\frac{\pi}{18}\right) & -\sin\left(\frac{\pi}{18}\right) & 0 \\ \sin\left(\frac{\pi}{18}\right) & \cos\left(\frac{\pi}{18}\right) & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad B_5 = B_6 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad f_5 = f_6 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

The polyhedral regions are defined as

$$\begin{aligned} R_1 &= \left\{(x_1^c, x_2^c) : -x_1^c + x_2^c \leq 0,\ x_1^c \leq 1.3,\ x_2^c \geq 0\right\}, \\ R_2 &= \left\{(x_1^c, x_2^c) : -2.14x_1^c + x_2^c \leq 0,\ 0.7x_1^c + x_2^c \leq 1.7,\ x_1^c - x_2^c < 0\right\}, \\ R_3 &= \left\{(x_1^c, x_2^c) : x_2^c < 0,\ 0 \leq x_1^c \leq 1.3,\ -0.47x_1^c - x_2^c \leq 0.2\right\}. \end{aligned}$$

The cells χ_i, $i = 1, \ldots, 6$ in which subsystems Σ_i operate are defined as

$$\begin{aligned} \chi_1 &= R_1, \\ \chi_2 &= R_1, \\ \chi_3 &= \chi_4 = R_2, \\ \chi_5 &= \chi_6 = R_3. \end{aligned}$$

The logical states associated to the respective cells are

$$\begin{aligned} \tilde{x}_1^d &= \tilde{x}_3^d = \tilde{x}_5^d = 0, \\ \tilde{x}_2^d &= \tilde{x}_4^d = \tilde{x}_6^d = 1. \end{aligned}$$

An examination of the eigenvalues of A_i^c, as shown in Table 2.1, reveals that all six continuous-time subsystems have eigenvalues with positive real parts. It is therefore clear that the individual subsystems, on their own, are unstable.

The system was simulated with the states starting from $x_1^c(0) = 0.95$, $x_2^c(0) = 1$ and $x^d(0) = 0$. The overall system state trajectory obtained from the simulation is shown

Matrix	Eigenvalues
A_1^c	$0.7391 \pm 0.3062i$
A_2^c	$0.7518 \pm 0.2736i$
$A_3^c = A_4^c$	$0.6928 \pm 0.4000i$
$A_5^c = A_6^c$	$0.7878 \pm 0.1389i$

TABLE 2.1: Matrices A_i^c and their eigenvalues

in Figure 2.1. It can be seen that, although the individual subsystems are unstable, the overall system is asymptotically stable as the continuous states converge to the origin.

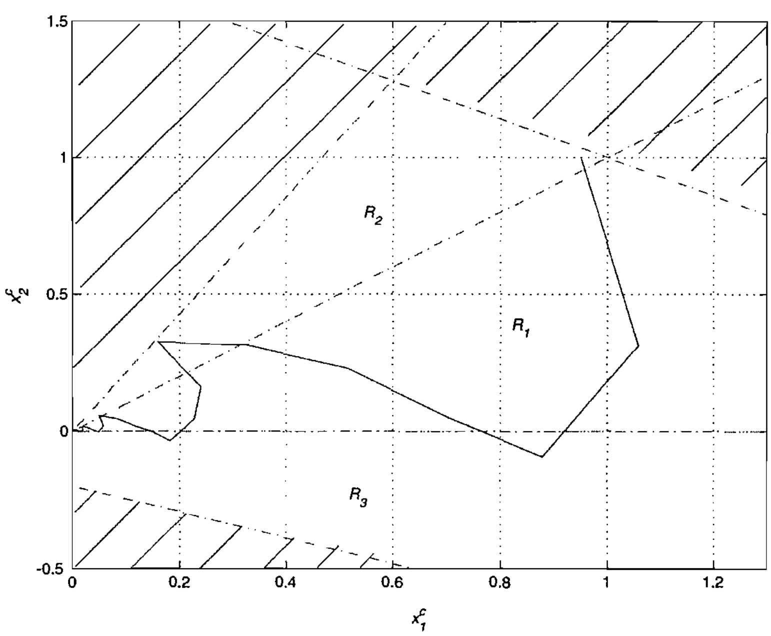


FIGURE 2.1: Phase plane trajectory for the PWA-LC system in Example 2.1

□

Example 2.2. *An unstable PWL system comprising of stable subsystems. This example has been adapted from (Branicky, 1994).*

The system is defined by the following subsystem matrices

$$A_1 = \begin{bmatrix} -0.1 & 1 \\ -10 & -0.1 \end{bmatrix}, \quad B_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix},$$

$$A_2 = \begin{bmatrix} -0.1 & 10 \\ -1 & -0.1 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

The state space is divided into the following polyhedral regions

$$\begin{aligned} R_1 &= \{(x_1, x_2) : x_1 \geq 0,\ x_2 \geq 0\}, \\ R_2 &= \{(x_1, x_2) : x_1 < 0,\ x_2 \geq 0\}, \\ R_3 &= \{(x_1, x_2) : x_1 < 0,\ x_2 < 0\}, \\ R_4 &= \{(x_1, x_2) : x_1 \geq 0,\ x_2 < 0\}. \end{aligned}$$

The subsystems Σ_1 and Σ_2 operate respectively on cells χ_1 and χ_2 defined by

$$\begin{aligned} \chi_1 &= R_2 \cup R_4, \\ \chi_2 &= R_1 \cup R_3. \end{aligned}$$

The eigenvalues of A_i, $i = 1, 2$ as shown in Table 2.2 are identical, with a negative real part. It can therefore be concluded that the individual subsystems are asymptotically stable.

Matrix	Eigenvalue
A_1	-0.1 ± 3.1623i
A_2	-0.1 ± 3.1623i

TABLE 2.2: Matrices A_i and their eigenvalues

The overall system was simulated starting from the initial states of $x_1 = 10^{-7}$ and $x_2 = 10^{-7}$. Figure 2.2 shows the state trajectory obtained from the simulation. Evidently, the overall system is unstable as the states divert away from the origin. □

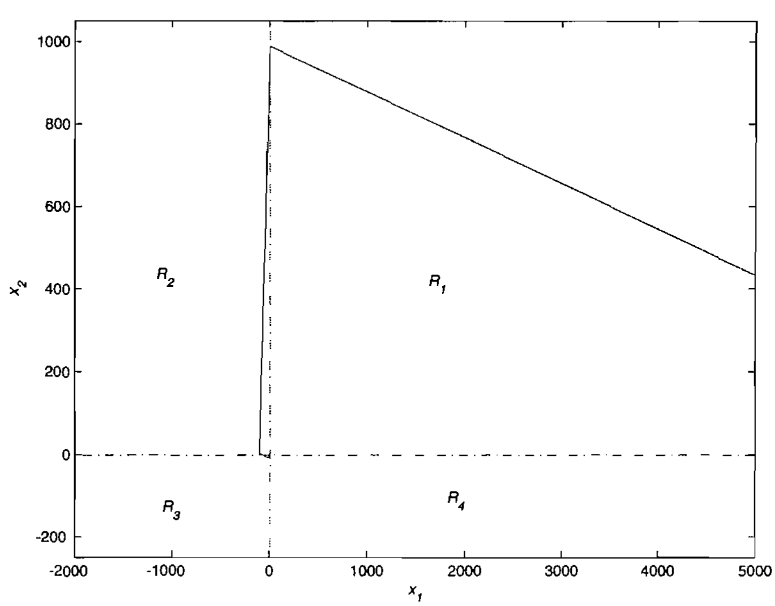


FIGURE 2.2: Phase plane trajectory for the PWL system in Example 2.2

The examples we have presented highlight the fact that stability of hybrid dynamical systems cannot be ascertained from stability analysis of the individual continuous-time subsystems. Further discussion on problems associated with stability of hybrid systems can be found in (Branicky, 1994; Liberzon and Morse, 1999; Leith et al., 2003). It is therefore not surprising that stability analysis of hybrid systems is an area that is currently seeing much research activity. Several approaches for some classes of hybrid systems have appeared in (Hespanha and Morse, 2002; Daafouz et al., 2002; Rodrigues, 2004; Gökçek, 2004; Stefanovic, 2004). An extensive compilation of the various approaches taken by researchers on the subject can be found in (Davrazos and Koussoulas, 2003). In view of the stability problem mentioned here, it is obvious that conventional control design and analysis methods do not sufficiently meet control requirements for hybrid dynamical systems. A different approach is therefore required. We will now discuss some of the work already done by researchers on the topic.

Present work on controller design methods for hybrid systems can broadly be classified as follows:

1. **Verification of the safety property.**

   Safety verification of hybrid dynamical systems involves reachability analysis for sets of continuous states. If it is possible to determine the set of states from where an unsafe mode can be reached, safety of the system could then be verified by ensuring that the system states do not stray into the aforementioned set. Figure 2.3 illustrates the problem. Suppose the hybrid system comprises of two continuous states, $x_{1,2}$ and the state space is partitioned into four polyhedral regions, R_{1-4}. The diagram depicts four trajectories, T_{1-4} starting from four different locations in R_1. Suppose polyhedron R_3 is unsafe (for example, the system would go into an unstable mode if the region is entered). Then, the system must not be allowed to follow trajectories T_2 and T_3. The verification problem would then be to determine the set of continuous states that would result in trajectories entering R_3. For discrete state spaces, efficient algorithms for performing this task have been established. However this is not the case for general continuous-time systems having state dimensions greater than four or five (Mitchell and Tomlin, 2003). Recent advances in this area include the work of Tomlin et al. (2003); Mitchell and Tomlin (2003) where numerical methods for computation of reachable sets based on the solution to some Hamilton-Jacobi-Isaacs Partial Differential Equations have been proposed.

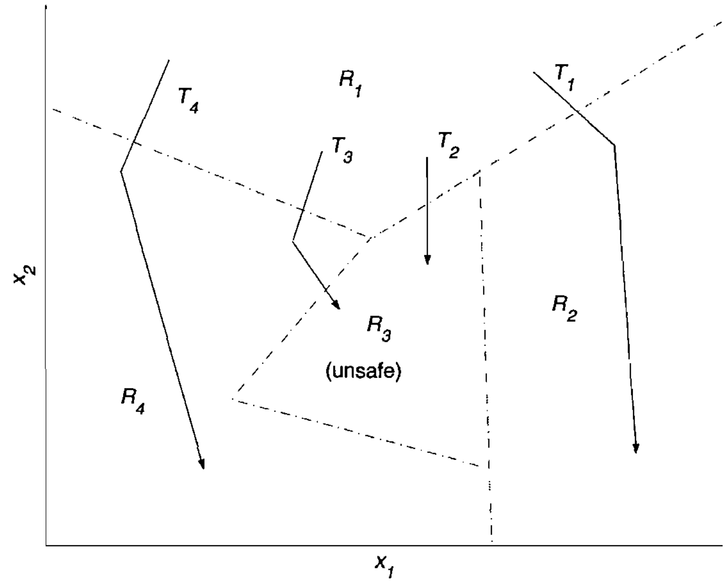


FIGURE 2.3: Safety verification problem: Find the set of initial states that would track trajectories leading into an unsafe region

2. **Piecewise state and output feedback optimal control.**

   Much of recent research on control of hybrid systems have focussed on the class of optimal controllers. Among these, the *Piecewise Linear Quadratic Optimal Control* method was proposed in (Rantzer and Johansson, 2000). In this method, optimal control of PWA systems is achieved by finding piecewise quadratic Lyapunov functions using convex optimization. Computation of such functions is achieved by using the Hamilton-Jacobi-Bellman inequality. A receding horizon technique has been proposed in (Borelli et al., 2003). Here, a combination of dynamic programming and multiparametric quadratic programming techniques are used to compute the state feedback optimal control law for PWA systems. Rodrigues and How (2003) proposed a synthesis method for observer-based state and output feedback control. Their method makes use of mathematical programming in the search of a piecewise quadratic Lyapunov function and a control law for PWA systems. Being observer-based, this method allows for the control of systems where the switching parameters are not measured directly.

3. **Model Predictive Control.**

   A *Model Predictive Control* (MPC) approach was proposed by de Schutter and van den Boom (2004). In this method, several linear programming problems are solved in order to obtain the optimal open loop control law in a receding horizon fashion.

4. **Multiple Model Adaptive Control.**

   Adaptive type control methods based on switching of fixed controller parameters, also known as *Multiple Model Adaptive Control* (MMAC), have recently received the attention of several researchers. A scheme where, based on some performance measurement, a controller is selected from a pool of candidate fixed controllers has been suggested in Hespanha and Morse (2002); Hespanha et al. (2003). The controller that is deemed to produce the best performance would be selected and placed into the control loop. The performance measurement is constantly monitored and if a different controller to the one currently selected is found to perform better (as, in general, would happen should the plant dynamics switch), the former would be moved into the control loop, replacing the latter. A stable realization of the system is achieved either by ensuring that the Youla parameters have a common Lyapunov function or by resetting the states of the time-varying system at switching instances. The *Multiple Model Switching and Tuning* (MMST) technique used by Narendra and co-workers (Narendra and Balakrishnan, 1994, 1997; Narendra, 2000; Narendra et al., 2003) and by Autenrieth and Rogers (1999) is quite similar to that of Hespanha and Morse (2002); Hespanha et al. (2003) in that a switch in the plant dynamics will cause a controller switch where the controller is selected from a pool of candidate controllers. Unlike in Hespanha and Morse (2002); Hespanha et al. (2003), the MMST method makes use of a *Model Reference Adaptive Control* (MRAC) scheme to fine tune the response after switching has taken place.

The methods we have mentioned above do not constitute an exhaustive list of all work that have been done on hybrid systems control. However, these are broadly representative of the many different approaches that have been suggested.

In this research, we are particularly interested in the *switching-based adaptive control* approach. In particular, we see the MMST method as a practical approach in controlling hybrid systems. In Chapter 3, we introduce some modifications to the technique used in (Narendra and Balakrishnan, 1997) and investigate the performance of the modified scheme. Before that, we first discuss the fundamental concepts of MMST.

2.4 Multiple Model Switching and Tuning

Multiple Model Switching and Tuning (MMST) can be attributed to Narendra and Balakrishnan (1997). The concept is a variation from conventional adaptive control and was first proposed to provide control systems with robustness against large and abrupt changes in the plant's operating environment. In the past, conventional adaptive and robust control competed against each other as the best technique to be used in controller design for plants with some uncertainty. In self-tuning adaptive control, a plant model identifier is used in the closed loop, allowing the controller to adapt itself to parameter variations. Robust control, on the other hand, deals with the problem of controller design in the presence of uncertainties. This includes parameter variations and unstructured model uncertainties. It has been recognized that there should be interaction between robust control design, plant identification in the closed-loop and adaptive control. Such interaction has been shown to be enhanced by the use of MMST (Landau, 1999). The increased robustness property of MMST systems makes it ideal to deal with the abrupt switching behaviour experienced by hybrid systems.

2.4.1 Conventional Adaptive Control

Commonly, control systems are designed based on a good understanding of the plant and its environment. The controller parameters are selected to ensure the controlled plant meets performance specifications and is robust to noise and uncertainty in the plant parameters. In conventional control schemes, the controller parameters take up fixed values that are optimized to meet specifications and, at the same time, be insensitive to perturbations. In adaptive control systems, on the other hand, the controller parameters would adjust themselves such that the controlled plant always meets the performance targets. While there are many ways in which adaptive control schemes can be classified, two important classifications are the *self-tuning control* (STC) and *model reference adaptive control* (MRAC) schemes. Sastry and Bodson (1989) distinguished between the two in the following way:

1. STC

   In STC, the plant is assumed to have constant but unknown parameters. If the plant parameters are known, the controller parameters could be set such that performance targets are met. The philosophy behind STC is that, even when the plant parameters are not known, the controller parameters should tune itself to the desired values. This would normally involve the estimation of the plant parameters, and from this estimation, the controller parameters are determined and are automatically tuned to.

   Figure 2.4 illustrates this concept. Here, u_r is the reference input, u is the control input and y is the plant output. In addition to the feedback loop, a secondary loop (shown by dashed lines) is introduced. In this secondary loop, an identifier estimates the plant parameters. Using these estimated parameters, the *Controller Design* block tunes the controller parameters such that y meets its specified target.

2. MRAC

   In MRAC, the objective is to control the plant such that its response matches that of a chosen reference model. In this scheme, the response of the plant is compared to the response of the reference model. The error between the two is then used

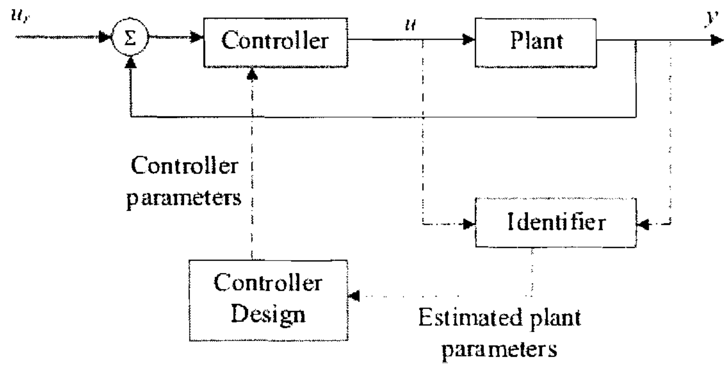


FIGURE 2.4: Self-tuning control

to calculate the controller parameters using a set of rules. Estimation of the plant parameters is not required in order to obtain the controller parameters.

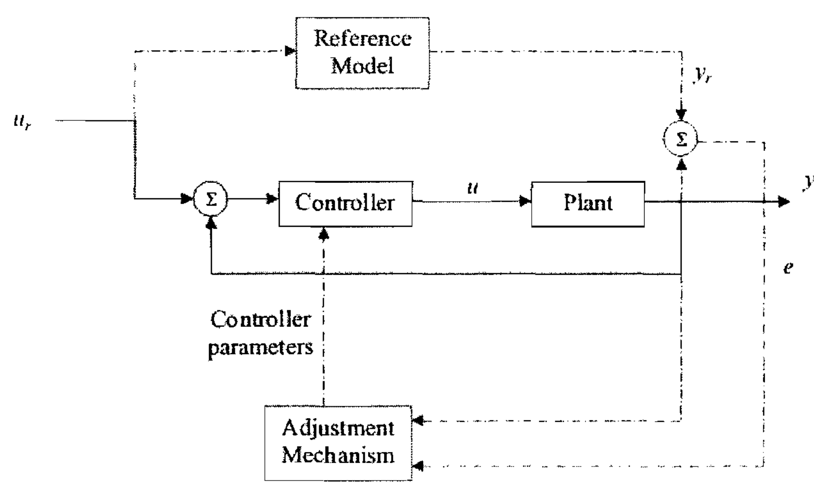


FIGURE 2.5: Model reference adaptive control

Figure 2.5 shows a schematic diagram of the MRAC process. Here, u_r, u and y are as defined in the STC case. A reference model that produces output y_r given an input u_r is introduced into the secondary loop. The objective is to make the plant output follow the reference model output. The error between the two, $e = y - y_r$ is fed into the *Adjustment Mechanism* block which computes the controller parameters such that y and y_r would converge. Controller parameter calculations are performed based on the error signal e. Therefore, no plant identifier is required.

Adaptive control schemes can also be categorized as either *indirect* or *direct*. The indirect method refers to adaptive control schemes where the plant parameters need to be estimated first in order to estimate the controller parameters. Conversely, in the direct scheme the controller parameters are determined without the plant estimation step. From the classification of Sastry and Bodson (1989), MRAC can be regarded as a direct adaptive control scheme and STC can be viewed to be an indirect adaptive control method. It should be noted that there are STC schemes that bypass the estimation step and hence are direct. In the same respect, Narendra and Balakrishnan (1997) used an indirect adaptive control scheme that follows a specified reference model. Practically, little distinction is made between STC and MRAC, and the two are often studied interchangeably. For an introduction to adaptive control, the interested reader is referred to (Åstrom and Wittenmark, 1989; Sastry and Bodson, 1989; Narendra and Annaswamy, 1989; Slotine and Li, 1991), for example. More detailed treatment of the subject can be found in (Tao, 2003; Ioannou and Sun, 1996).

2.4.2 Multiple Model Adaptive Control

MMST is a control scheme that comes under the class of *Multiple Model Adaptive Control* (MMAC). The basic idea behind MMAC is to have a set of candidate models enveloping the entire operating region of the plant. For each candidate model, controller parameters that would produce the required plant performance are determined *a priori*. During system operation, using some form of cost criterion, the candidate model that best matches the plant is determined. Consequently, the controller parameters corresponding to the selected candidate model is placed into the control loop. Upon occurrence of a switch in plant dynamics, a second candidate model may match the new plant parameters better than the one currently selected. As a result, the control loop parameters would switch to the values corresponding to the second candidate model. The process carries on as such, repeating itself at the onset of every switch in plant dynamics. Obviously, if after a switch in plant parameters, the currently selected candidate model still matches the plant better than any other candidate model, then no switching of controller parameters occur.

To illustrate this, Figure 2.6 shows the general architecture of an MMAC system. Here P is the actual plant, P_i and K_i are respectively, the i-th candidate model and its corresponding controller. $i = 1, 2, \ldots, n$ is the index for the set of n candidate models and controllers. u_i is the control input generated by K_i and y_i is the output coming from P_i. u_r is the reference input into the system and y is the actual plant output. $e_i = y - y_i$ is the error between the actual plant output and the i-th candidate model output. Using a pre-determined cost function $J_i = f(e_i)$ to evaluate all candidate models, the candidate model P^* that minimizes $\{J_i\}_{i=1}^{n}$ is determined and the corresponding controller K^* is placed into the control loop. If, after a period of time, P^{**} minimizes $\{J_i\}_{i=1}^{n}$, its corresponding controller K^{**} replaces K^{*} in the control loop.

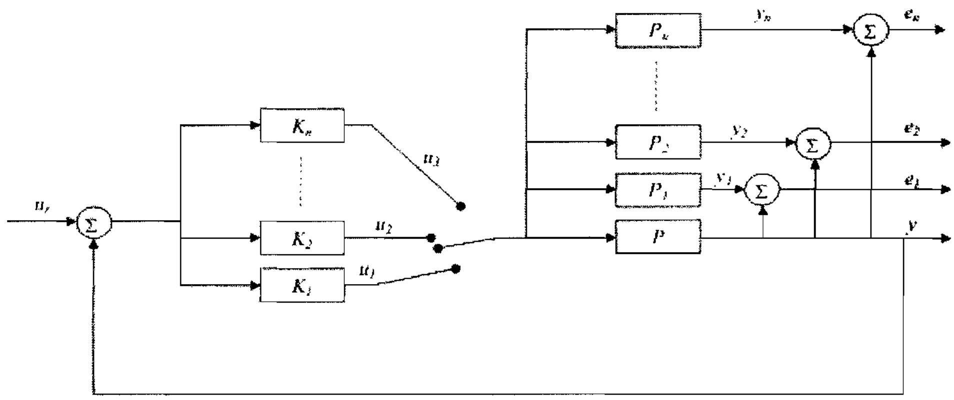


FIGURE 2.6: General architecture of MMAC

Two important problems that need to be addressed when implementing an MMAC system are

1. Determination of the set of candidate models

   It has been shown in Anderson et al. (2000) that given certain mild assumptions on the uncertainty set of linear time-invariant plant models, a finite set of plants can be found such that at least one controller from a finite set of corresponding controllers will control each plant in the uncertainty set satisfactorily.

2. Determination of the switching algorithm

   From the finite set of plant models, the model that best fits the plant in its current operating environment is selected based on some criterion. A performance index could be used for this purpose such as Equation (2.3) which was suggested in Narendra and Balakrishnan (1997).

$$\begin{gathered} J_i\left(t\right) = \alpha e_i^2\left(t\right) + \beta \int_0^t e^{-\lambda(t-\tau)} e_i^2\left(\tau\right) d\tau, \\ \alpha \geq 0, \quad \beta, \lambda > 0. \end{gathered} \tag{2.3}$$

   Here J_i is the performance index for controller K_i. α and β are parameters whose values are chosen respectively, to give the desired instantaneous and long-term accuracy and λ is a forgetting factor.

   Other parameter estimation methods have also been suggested. For example, Petridis and Kehagias (1998) proposed a parameter estimation algorithm for non-linear systems where the parameter values are assumed to change according to a Markovian model switching mechanism. A study on this problem has been reported by Hespanha et al. (2001).

2.4.3 From MMAC to MMST

In addition to the n candidate fixed controllers, the MMST scheme introduced by Narendra and Balakrishnan (1997) and later adopted by Autenrieth and Rogers (1999) also incorporated two model reference adaptive controllers into their schemes. One is a *resettable adaptive controller* (RSA) and the other is a *free running adaptive controller* (FRA). The RSA is an adaptive controller that resets to some known parameter values each time the selected candidate model switches. The parameter values that the RSA controller resets to are exactly those of the fixed controller corresponding to the selected candidate model. This would make the adaptation process start from a point quite close to the ideal parameter values. It would therefore take less time for convergence between plant response and ideal response to occur. The purpose of the FRA is to ensure that the controlled plant would always be stabilized. Since the FRA does not reset, it was thought that the FRA would always eventually converge, something that the RSA might not be able to achieve if resetting takes place too often. However, since in general for the FRA, adaptation starts further away from the ideal parameter values compared to the RSA, the FRA would require a longer time to achieve convergence.

In short, the whole functionality of MMST can be summarized as follows. The plant is first identified through a parameter estimation scheme such as the recursive least-squares algorithm. The response of the estimated plant parameters to the given reference input, is compared to those of the candidate models. A performance index is used to determine which of the candidate models best matches the estimated plant. The fixed controller parameters that correspond to the selected candidate model is placed into the closed-loop system. At the same time, the RSA parameters are re-initialized to those of the selected fixed controller. The RSA parameters then tune themselves and the simulated performance is compared to that of the fixed controller. When the RSA performance has surpassed that of the fixed controller, it is brought into the system, replacing the latter. Each time the plant undergoes a switch, the whole process is repeated. In the meantime, the FRA runs continuously in the background with the purpose of ensuring convergence of the system should the RSA fail to do so. Results from the work of Autenrieth and Rogers (1999) have indicated that MMST, when applied to one class of systems with switching dynamics, is capable of providing superior performance compared to conventional adaptive control schemes.

Chapter 3

Application of MMST to SISO Hybrid Systems

In Section 2.4, we studied the concept of Multiple Model Adaptive Control (MMAC). The fundamental concept of the MMAC scheme is that, based on some performance criterion, the best controller from a pool of candidate controllers is selected and placed in the control loop. If, at a later stage, another candidate controller outperforms the currently selected one, the former is switched in, taking the place of the latter. Throughout plant operation, the performance of each candidate controller is monitored and switching of the selected controller takes place accordingly.

An important issue in the implementation of MMAC is the switching algorithm. Narendra and co-workers (Narendra and Balakrishnan, 1994, 1997; Narendra, 2000; Narendra et al., 2003) preferred the indirect switching method where the plant parameters are first estimated using an estimation algorithm. Performance of the candidate controllers is evaluated based on these estimated parameters. They also extended the MMAC scheme to include indirect self-tuning adaptive control to fine tune system performance after switching has taken place. Alternatively, Safonov and Tsao (1997); Stefanovic (2004) pursued a direct switching method where estimation of the plant parameters is not required. This is done using the *unfalsified control* concept (Safonov and Tsao, 1997).

In this chapter, we study the use of MMAC in controlling certain classes of Single Input Single Output (SISO) hybrid dynamical systems. The hybrid systems considered here are

1. Systems whose plant dynamics switch due to the occurrence of an external event, for example when a switch is closed by a human operator. This type of systems can be modelled in the PWA-LC form.

2. Systems whose plant dynamics switch due to the evolution of the continuous states. Here, the continuous state space is divided into polyhedral regions and the plant

dynamics switch upon movement of the continuous states from one region to another. Such systems can be represented by the PWA or PWL forms.

This chapter proceeds as follows. In Section 3.1, we define the control scheme used in this study. The results of simulations carried out on an example system where switching of plant dynamics occur due to external events are shown in Section 3.2. In Section 3.3, we present simulation results for a system where location of the continuous states determine the plant dynamics. In Section 3.4, we conclude this chapter with a discussion of the results obtained.

3.1 Control Scheme

In this research, we adopt the MMST scheme consisting of several candidate fixed controllers, one resettable adaptive controller (RSA) and one free running adaptive controller (FRA). Unlike in the work of Narendra and Balakrishnan (1994, 1997); Narendra (2000); Narendra et al. (2003) and also that adopted by Autenrieth and Rogers (1999), we chose to use the direct MRAC scheme for our RSA and FRA controllers. Our choice of using direct MRAC comes from the following considerations:

1. Ease of implementation.

   The direct MRAC method bypasses the plant parameter estimation step. The controlled plant output is compared directly to the reference model output and a corrective control signal is applied to the plant. Computation is therefore simplified.

2. Suitability for use with multivariable systems.

   When applied to multivariable systems, computation of controller parameters in indirect adaptive control schemes often involve multiplication of the inverse of certain matrices whose elements are dependent on the estimated plant parameters. These matrices may easily become singular, making the computation intractable. Using the direct MRAC scheme avoids this problem to some extent.

We also examine the use of state feedback against output feedback in our MMST control scheme. It would only be natural that one would expect state feedback to perform better than output feedback. However, we do acknowledge that in most practical situations, not all states are accessible and state estimation through the use of observers is required. We also note that a state estimation scheme would be immensely useful for control of PWA systems where the plant dynamics is determined by the continuous states.

3.1.1 State Feedback Control

The plant to be controlled, Σ_p is defined in state space form as

$$\dot{x}(t) = A_p x(t) + B_p u(t)$$

where $x(t) \in \mathbb{R}^n$ is the state vector and $\dot{x} = \frac{d}{dt}x(t)$, $u(t) \in \mathbb{R}$ is the input, $A_p \in \mathbb{R}^{n\times n}$ is the system matrix, $B_p \in \mathbb{R}^{n\times 1}$ and n is the dimension of the state vector.

Similarly, the reference model, Σ_r is given by

$$\dot{x}(t) = A_r x(t) + B_r u_r(t)$$

where $u_r \in \mathbb{R}$ is the reference input.

1. **Fixed controller parameters**

   Our MMST control scheme is made up of n candidate models, Σ_i, $i = 1, \ldots, n$ defined by

   $$\dot{x}(t) = A_i x(t) + B_i u(t)$$

   where A_i and B_i are determined *a priori*. To each candidate model, the feedback and feedforward parameter matrices, S_i and T_i respectively, are assigned according to the control system structure as illustrated in Figure 3.1. The control law is then

   $$u(t) = S_i x(t) + T_i u_r(t).$$

   The closed loop system is therefore

   $$\dot{x}(t) = (A_i + B_i S_i)x(t) + (B_i T_i)u_r(t).$$

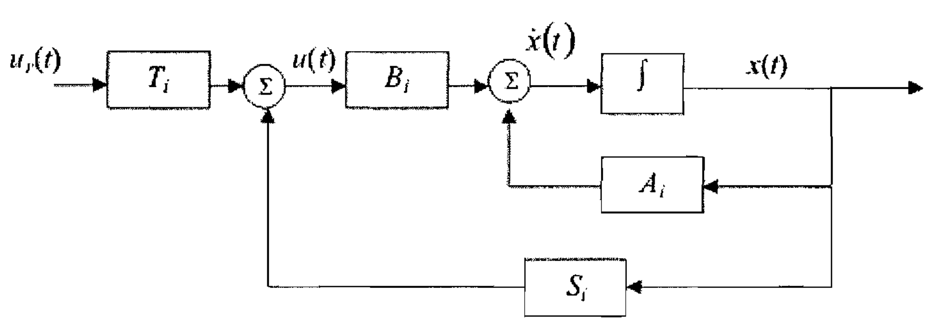


FIGURE 3.1: Structure of state feedback control

Since A_i and B_i are known, the closed loop system will emulate Σ_r if S_i and T_i can be selected such that

$$A_i + B_i S_i = A_r,$$
$$B_i T_i = B_r.$$

Hence, if within the set of candidate models, there exist $\tilde{\Sigma} = \left(\tilde{A}, \tilde{B}\right)$ which perfectly matches the plant $\Sigma_p = (A_p, B_p)$, the closed loop response can be made to follow the reference model response exactly.

2. **Tuning of adaptive controllers**

If there is no exact match between the candidate models and the plant, the closest match is chosen and a tunable adaptive controller sets in to correct the discrepancy. The tunable adaptive controllers follow the MRAC scheme similar to that shown in Figure 2.5.

Two types of tunable adaptive controllers are used: a free-running adaptive (FRA) controller and a resettable adaptive (RSA) controller. For the FRA, the parameters start from an arbitrarily chosen value and are tuned continuously throughout the process. For the RSA, the parameters are reset every time the selected candidate controller switches. The parameters reset to the pre-determined parameter values of the selected fixed candidate controller and then, are tuned such that the system response is forced to emulate that of the reference model. To illustrate the function of the RSA, suppose the currently selected model is $\Sigma_1 = (A_1, B_1)$ and the parameters in the control loop are S_1 and T_1. At an instant of time t^*, the selected model switches to $\Sigma_2 = (A_2, B_2)$ and the controller parameters switch to S_2 and T_2. The RSA parameters would immediately reset to the values of S_2 and T_2. These parameters are then tuned so as to track the reference model response. The adaptation law used to tune the parameters is based on Lyapunov's stability theory and is given by Equation (3.1) (Åstrom and Wittenmark, 1989)

$$\frac{d\theta}{dt} = -\gamma e \theta \tag{3.1}$$

where γ is the adaptation rate, e is the error between the plant output and the reference model output and θ is the parameter to be tuned. In this case,

$$\tfrac{dS}{dt} = -\gamma e x,$$

$$\tfrac{dT}{dt} = -\gamma e u_r.$$

3.1.2 Output Feedback Control

Similar to the state feedback control scheme explained in Section 3.1.1, our output feedback MMST scheme is made up of n fixed candidate controllers, an FRA controller and an RSA controller. For output feedback control, we model the system in the transfer function form where the plant Σ_p is represented as

$$y(s) = G_p(s)\,u(s)$$

where $y \in \mathbb{R}$ is the plant output, $u \in \mathbb{R}$ is the plant input, $s \in \mathbb{C}$ is the Laplace variable and

$$G_p(s) = k_p \frac{Z_p(s)}{P_p(s)}.$$

Here, k_p is a non-zero constant and Z_p and P_p are polynomials in s. Similarly, the reference model, Σ_r is represented as

$$y(s) = G_r(s)\,u_r(s)$$

where $u \in \mathbb{R}$ is the reference input. The candidate models, Σ_i, $i = 1, \ldots, n$ are represented as

$$y(s) = G_i(s)\,u(s).$$

Each candidate model is assigned feedback and feedforward gain parameters, K_{Si} and K_{Ti} respectively, as shown in Figure 3.2.

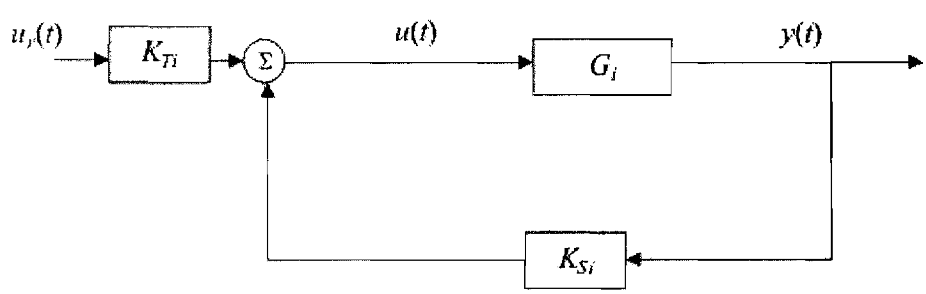


FIGURE 3.2: Structure of output feedback control

Since only the output measurements are used for feedback, the closed loop system, in general, will not match the reference model exactly. It is therefore up to the tunable adaptive controllers to bring the system response in line with that of the reference model. As in the state feedback case, the tunable output feedback adaptive controllers follow an MRAC scheme.

3.2 Switching Due to an External Event

Simulations were performed based on a case-study example from Autenrieth and Rogers (1999). Here, plant switching occurs due to an external event (for example, when a switch is closed). The instance of switching, and the plant models before and after switching are not known.

3.2.1 System Definition

The system is defined in transfer function (for output feedback control) and state space (for state feedback control) forms as follows:

1. **Transfer function form**

   The plant subsystems, Σ_A and Σ_B, in transfer function form, are represented by

   $$\Sigma_A = \frac{0.55}{s^2+1.7s+0.43} \text{ for } t \leq 60s,$$

   $$\Sigma_B = \frac{1}{s^2+1.7s-0.2} \text{ for } t > 60s.$$

   Four candidate models, Σ_i, $i = 1, \ldots, 4$ are used to determine the controller parameters. The four models and the corresponding controller parameters are given in Table 3.1.

i	Σ_i	K_{Si}	K_{Ti}
1	$\frac{0.5}{s^2+1.7s-0.2}$	-2.5	2.1
2	$\frac{0.5}{s^2+1.7s+0.3}$	-1.7	2.3
3	$\frac{1}{s^2+1.7s-0.2}$	-1.3	1.1
4	$\frac{1}{s^2+1.7s+0.3}$	-0.9	1.2

TABLE 3.1: Candidate models and their corresponding controller parameters

   The plant is required to emulate the response of a reference model, Σ_r given by

   $$\Sigma_r = \frac{1}{s^2+1.4s+1}.$$

2. **State space form**

   In state space form, the subsystems $\Sigma_A = (A_A, B_A, C_A)$ and $\Sigma_B = (A_B, B_B, C_B)$

are

$$A_A = \begin{bmatrix} 0 & 1 \\ -1.7 & -0.43 \end{bmatrix}, \; B_A = \begin{bmatrix} 0 \\ 0.55 \end{bmatrix}, \; C_A = \begin{bmatrix} 1 & 0 \end{bmatrix} \quad \text{for } t \leq 60s,$$

$$A_B = \begin{bmatrix} 0 & 1 \\ -1.7 & 0.2 \end{bmatrix}, \; B_B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \; C_B = \begin{bmatrix} 1 & 0 \end{bmatrix} \quad \text{for } t > 60s.$$

The candidate models, Σ_i are

$$A_1 = \begin{bmatrix} 0 & 1 \\ -1.7 & 0.2 \end{bmatrix}, \; B_1 = \begin{bmatrix} 0 \\ 0.5 \end{bmatrix}, \; C_1 = \begin{bmatrix} 1 & 0 \end{bmatrix},$$

$$A_2 = \begin{bmatrix} 0 & 1 \\ -1.7 & -0.3 \end{bmatrix}, \; B_2 = \begin{bmatrix} 0 \\ 0.5 \end{bmatrix}, \; C_2 = \begin{bmatrix} 1 & 0 \end{bmatrix},$$

$$A_3 = \begin{bmatrix} 0 & 1 \\ -1.7 & 0.2 \end{bmatrix}, \; B_3 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \; C_3 = \begin{bmatrix} 1 & 0 \end{bmatrix},$$

$$A_4 = \begin{bmatrix} 0 & 1 \\ -1.7 & -0.3 \end{bmatrix}, \; B_4 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \; C_4 = \begin{bmatrix} 1 & 0 \end{bmatrix}.$$

The reference model, Σ_r is

$$A_r = \begin{bmatrix} 0 & 1 \\ -1.4 & -1 \end{bmatrix}, \; B_r = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \; C_r = \begin{bmatrix} 1 & 0 \end{bmatrix}.$$

The controller parameters for each candidate model are then

$$T_1 = 2, \quad S_1 = \begin{bmatrix} 0.6 & -2.4 \end{bmatrix},$$

$$T_2 = 2, \quad S_2 = \begin{bmatrix} 0.6 & -1.4 \end{bmatrix},$$

$$T_3 = 1, \quad S_3 = \begin{bmatrix} 0.3 & -1.2 \end{bmatrix},$$

$$T_4 = 1, \quad S_4 = \begin{bmatrix} 0.3 & -0.7 \end{bmatrix}.$$

A cost function is used to evaluate each candidate model in order to find which one best matches the plant. The cost function used here is similar to Equation (2.3) and shown again here for brevity:

$$J_i(t) = \alpha e_i^2(t) + \beta \int_0^t e^{-\lambda(t-\tau)} e_i^2(\tau)\, d\tau,$$
$$\alpha \geq 0, \quad \beta, \lambda > 0.$$

Tunable adaptive controllers require a little time for the parameters to tune sufficiently to provide effective control. In view of this, the controllers cannot be allowed to switch too frequently. As such, in our system, after a switch in controller parameters have taken place, no further switch is allowed for a period of 2.3 seconds.

3.2.2 Simulation Results

1. **Free Running Adaptive Control of Switched Systems**

   Figures 3.3(a) and 3.3(b) show the simulation results of a switching plant controlled by free running adaptive state feedback control and free running adaptive output feedback control respectively.

2. **Resettable Adaptive Control of Switched Systems**

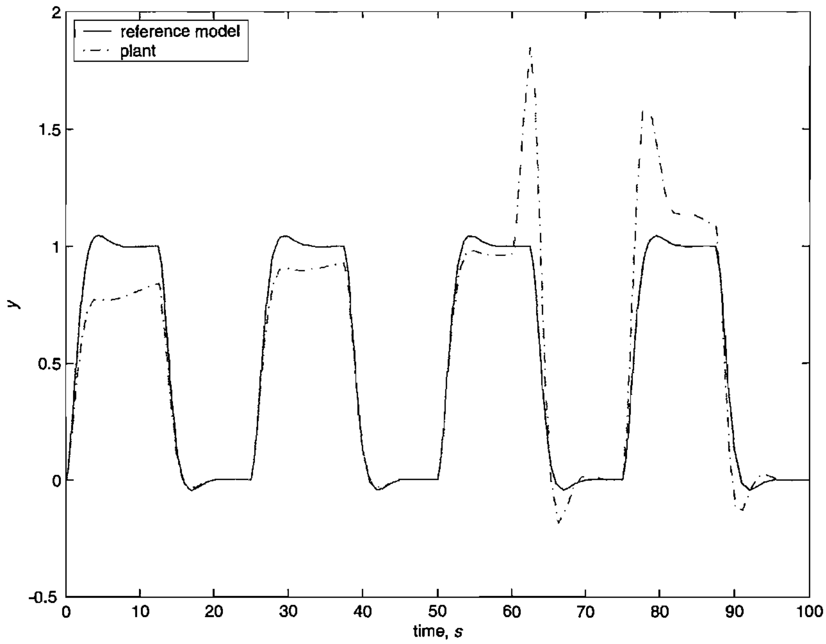
   Figures 3.4(a) and 3.4(b) show the simulation results of a switching plant controlled by resettable adaptive state feedback control and resettable adaptive output feedback control respectively.

3. **Fixed Control of Switched Systems**

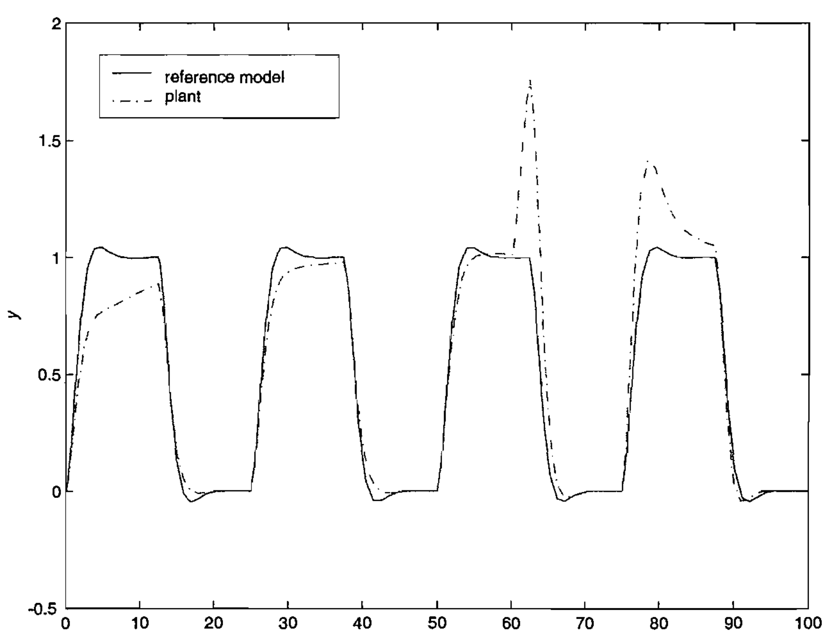
   Figures 3.5(a) and 3.5(b) show the simulation results of a switching plant controlled by fixed state feedback control and fixed output feedback control respectively.

4. **Fixed and Resettable Adaptive Control of Switched Systems**

   Figures 3.6(a) and 3.6(b) show the simulation results of a switching plant controlled by a combination of fixed and resettable adaptive state feedback control and output feedback control respectively.

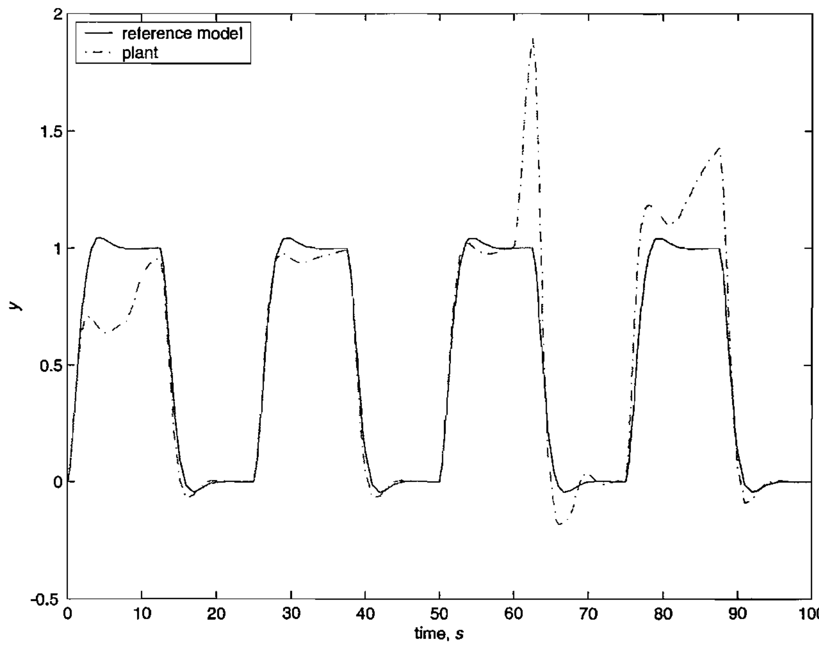


(a) State feedback response

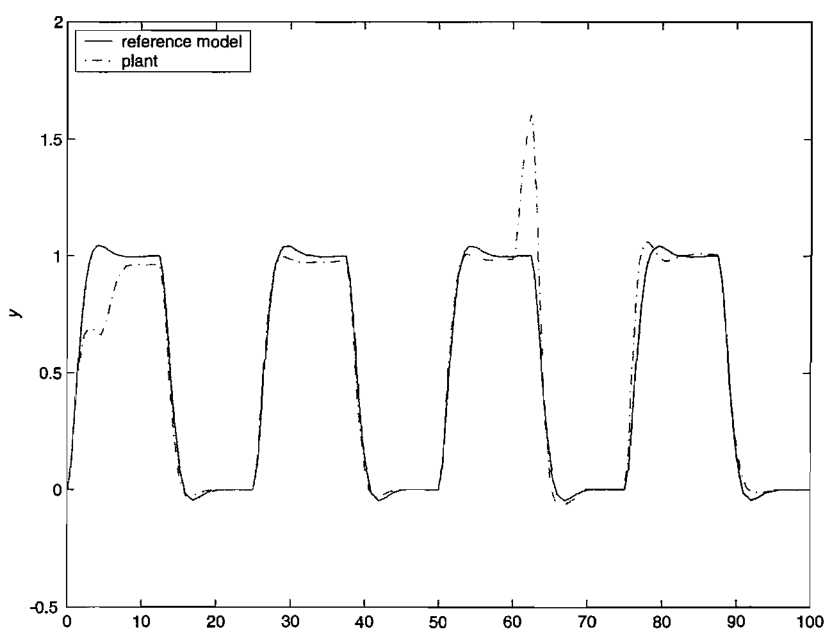


(b) Output feedback response

FIGURE 3.3: Switching due to an external event - FRA control

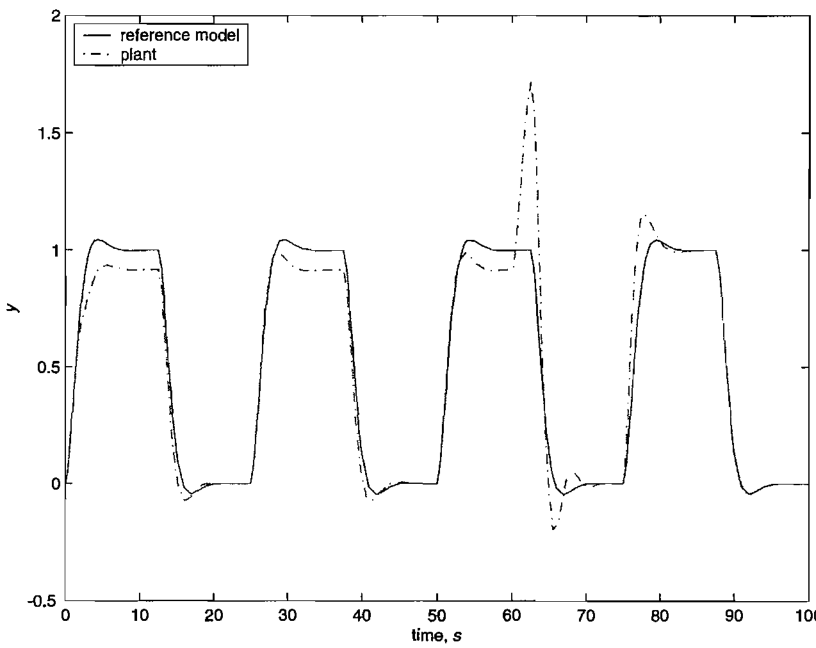


(a) State feedback response

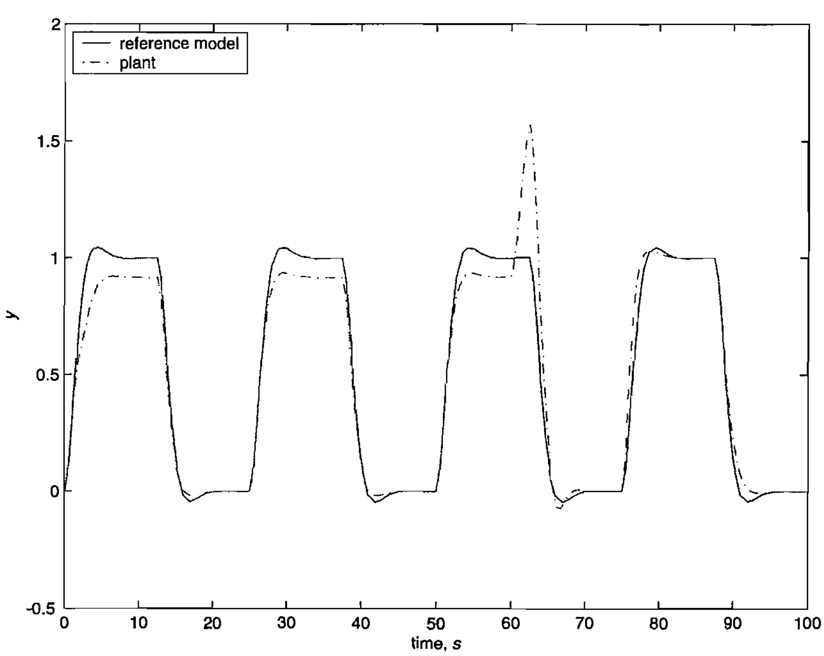


(b) Output feedback response

FIGURE 3.4: Switching due to an external event - RSA control

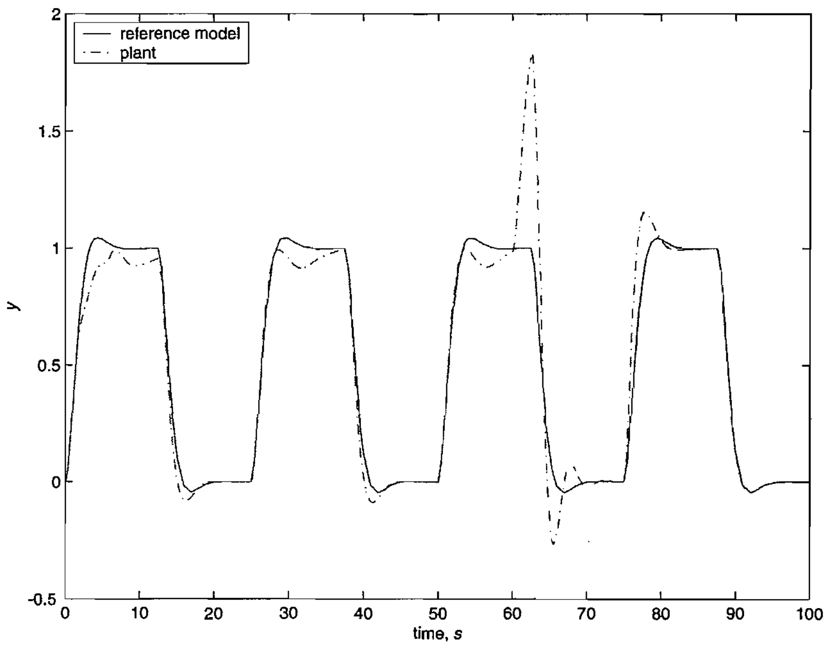


(a) State feedback response

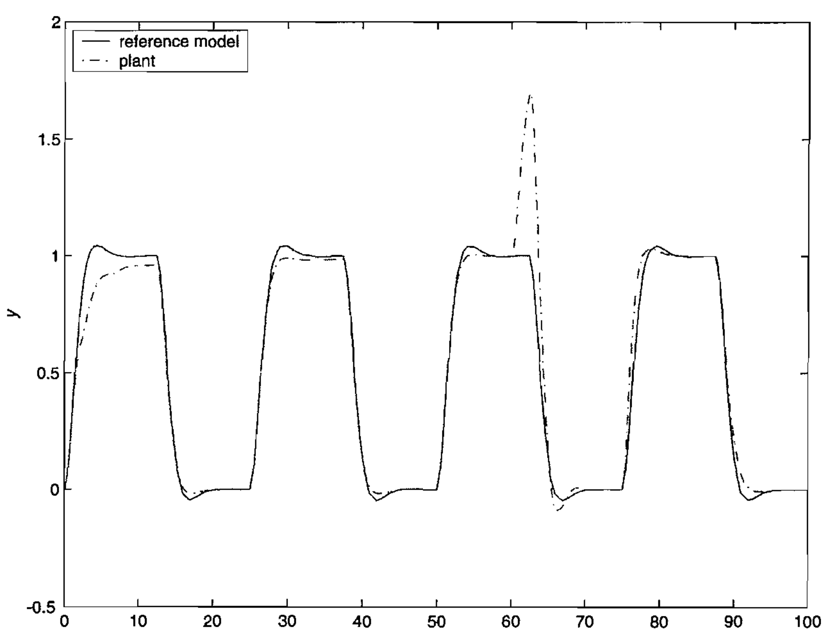


(b) Output feedback response

FIGURE 3.5: Switching due to an external event - Fixed controllers



(a) State feedback response



(b) Output feedback response

FIGURE 3.6: Switching due to an external event - Fixed + RSA control

3.3 Switching Due to the Location of States

In this section, we study the performance of our MMST scheme in a situation where the plant dynamics switch according to the location of the continuous states. Here, it is assumed that the plant dynamics in each polyhedral region are known well and that all states can be tracked such that the instance of switching is unambiguous.

3.3.1 System Definition

We adopt an example PWL system from Borelli et al. (2003). The system is defined in state space PWL representation for state feedback control. We also provide the transfer function equivalent of the subsystems to better illustrate output feedback control.

1. **State space PWL representation**

   The PWL system comprises of two subsystems $\Sigma_1 = (A_1, B_1, C_1)$ and $\Sigma_1 = (A_1, B_1, C_1)$ defined by

$$A_1 = 0.8\begin{bmatrix} \cos(\pi/3) & -\sin(\pi/3) \\ \sin(\pi/3) & \cos(\pi/3) \end{bmatrix}, \quad B_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad C_1 = \begin{bmatrix} 1 & 0 \end{bmatrix},$$

$$A_2 = 0.8\begin{bmatrix} \cos(-\pi/3) & -\sin(-\pi/3) \\ \sin(-\pi/3) & \cos(-\pi/3) \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad C_2 = \begin{bmatrix} 1 & 0 \end{bmatrix}.$$

   In order to construct the controller structure the system matrices need to be in the control canonical form as follows

$$A_1 = \begin{bmatrix} 0 & 1 \\ -0.64 & 0.8 \end{bmatrix}, \quad B_1 = \begin{bmatrix} 0 \\ -0.6928 \end{bmatrix}, \quad C_1 = \begin{bmatrix} 1 & 0 \end{bmatrix},$$

$$A_2 = \begin{bmatrix} 0 & 1 \\ -0.64 & 0.8 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0 \\ 0.6928 \end{bmatrix}, \quad C_2 = \begin{bmatrix} 1 & 0 \end{bmatrix}.$$

   The state space is divided into the following polyhedral regions

$$R_1 = \left\{\left(x, \tfrac{dx}{dt}\right) : x \geq 0\right\},$$
$$R_2 = \left\{\left(x, \tfrac{dx}{dt}\right) : x < 0\right\}.$$

   The subsystems Σ_1 and Σ_2 operate respectively on cells χ_1 and χ_2 defined by

$$\chi_1 = R_1,$$
$$\chi_2 = R_2.$$

The reference model is chosen to be

$$A_r = \begin{bmatrix} 0 & 1 \\ -1 & -1 \end{bmatrix}, \quad B_r = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad C_r = \begin{bmatrix} 1 & 0 \end{bmatrix}.$$

The system is not subjected to any external reference input but the initial conditions are $x = -1$ and $\frac{dx}{dt} = 1$.

In order to track the reference model response, the controller parameters were chosen as follows:

$$T_1 = -1.4434, \quad S_1 = \begin{bmatrix} 0.5196 & 2.5982 \end{bmatrix},$$
$$T_2 = 1.4434, \quad S_2 = \begin{bmatrix} -0.5196 & -2.5982 \end{bmatrix}.$$

2. Transfer function equivalent

In transfer function form, the plant is defined as

$$\Sigma_1 = \frac{-0.6928}{s^2-0.8s+0.64},$$

$$\Sigma_2 = \frac{0.6928}{s^2-0.8s+0.64}$$

and the reference model is

$$\Sigma_r = \frac{1}{s^2+s+1}.$$

The controller parameters for output feedback RSA control are as given in Table 3.2

i	K_{Si}	K_{Ti}
1	0.5196	-1.4434
2	-0.5196	1.4434

TABLE 3.2: Candidate controller parameters

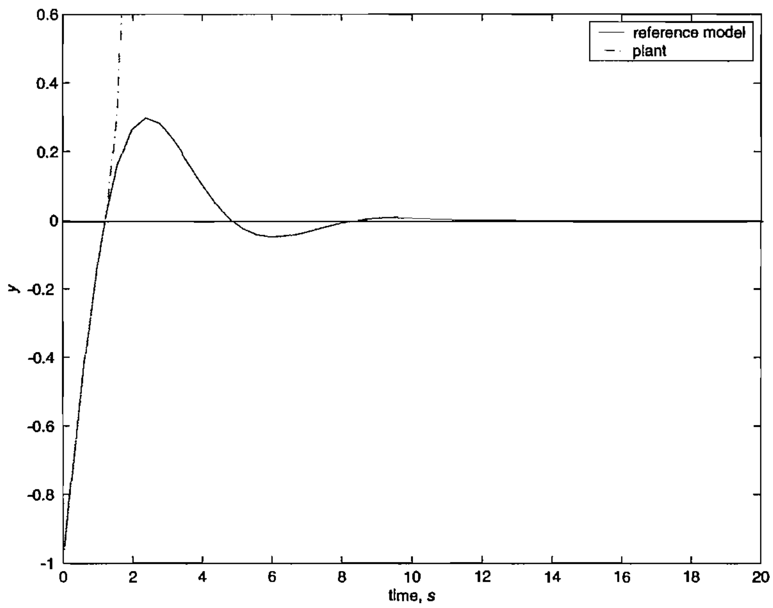
3.3.2 Simulation Results

1. State feedback

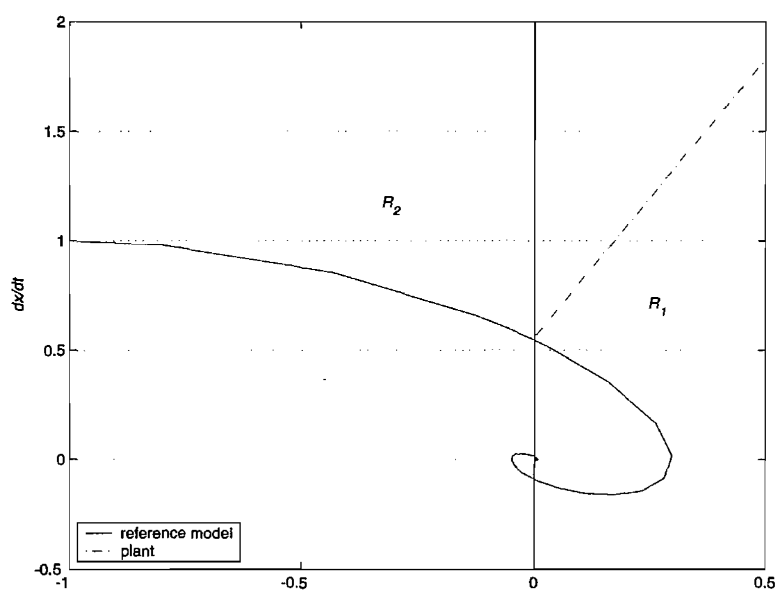
Figure 3.7 shows the simulated response and state trajectory for state feedback FRA control, while Figure 3.8 shows the results for state feedback RSA control. The response for state feedback RSA control when the plant is subjected to an additive perturbation of

$$\Delta = \frac{0.01}{s+0.01}$$

is shown in Figure 3.9.

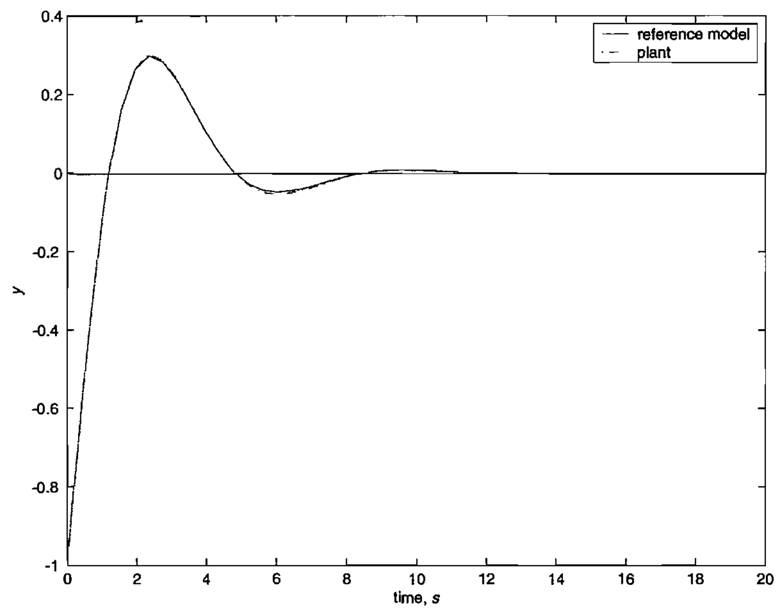


(a) Plant response

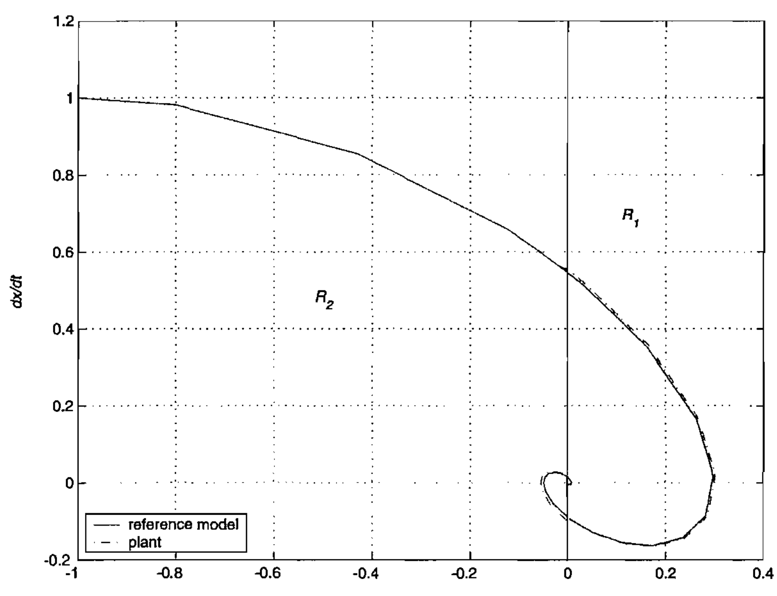


(b) State trajectory

FIGURE 3.7: Switching due the location of states - state feedback FRA control

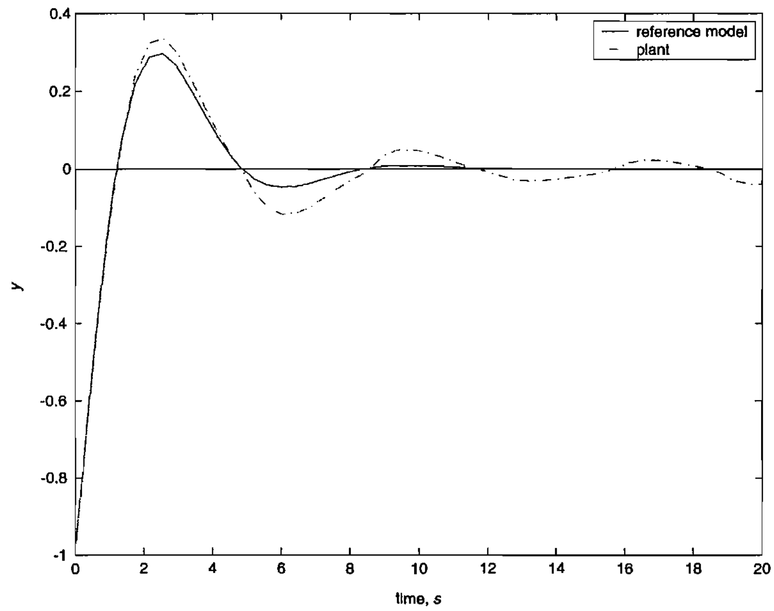


(a) Plant response

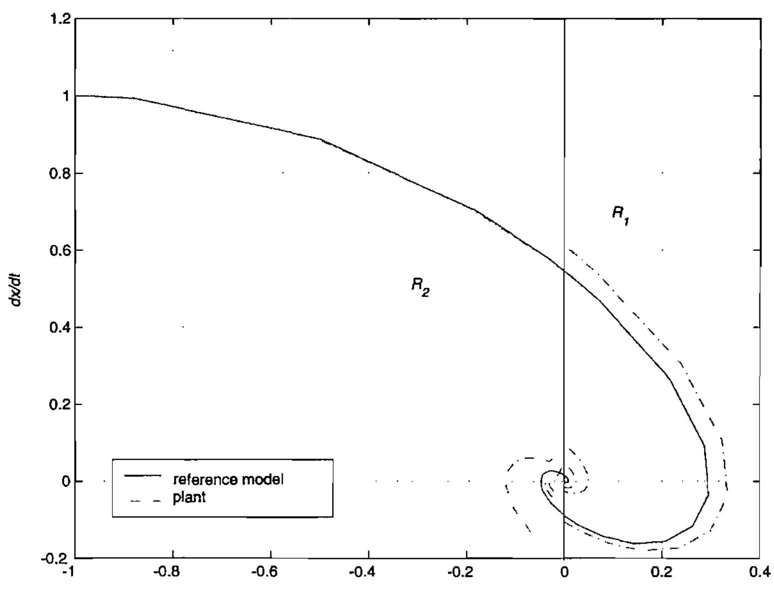


(b) State trajectory

FIGURE 3.8: Switching due to the location of states - state feedback RSA control



(a) Plant response

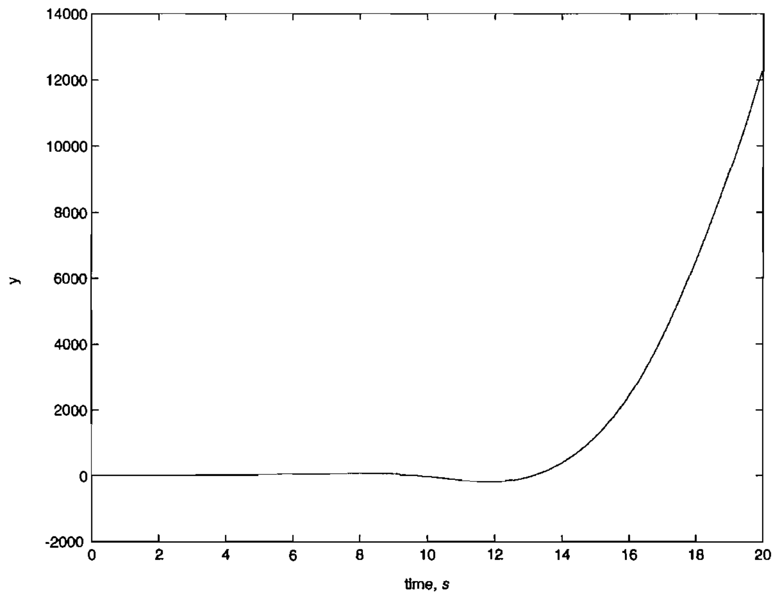


(b) State trajectory

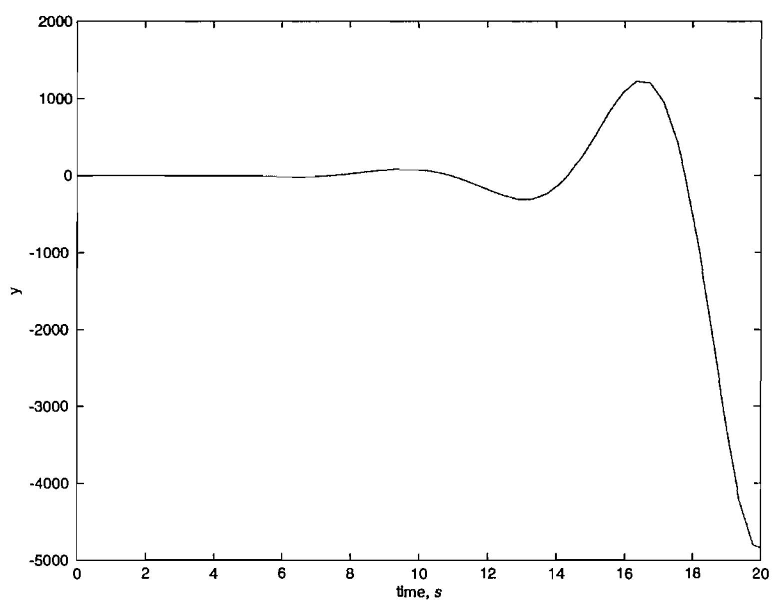
FIGURE 3.9: Switching due to the location of states - state feedback RSA control for plant subjected to perturbation

2. **Output feedback**

Figure 3.10(a) shows the response of a plant controlled by output feedback FRA control while 3.10(b) shows the response when the plant is controlled by an output feedback RSA control scheme.



(a) Free running adaptive control (plant response)



(b) Resettable adaptive control (plant response)

FIGURE 3.10: Switching due to the location of states - output feedback adaptive control

3.4 Conclusion

From the results obtained in Sections 3.2 and 3.3 the following observations can be made:

1. Using the direct adaptive control method as opposed to the indirect method in MMAC is viable. The direct method has the advantage that it is simpler to implement. The direct method also avoids the problem of multiplication by the inverse of singular matrices which may occur in indirect adaptive control when applied to multivariable systems. However, when the plant model is not well known *a priori*, identification is required to determine which candidate model best fits the plant. However, in our scheme, identification is not used to compute the controller parameters.

2. Free running adaptive control does not ensure convergence. This can be seen from Figure 3.7. In this example, before switching occurs, the plant dynamics is Σ_2 and the feedback and feedforward controller parameters are S_2 and T_2 respectively. When x crosses the threshold value of zero, the plant switches to Σ_1. However, at this point, the controller parameters have been optimized to stabilize Σ_2. These parameters, when applied to Σ_1 leads to instability. This can be confirmed by examining the eigenvalues of the closed-loop system immediately before and after switching, as shown in Table 3.3

Time	**Eigenvalues**
Before switching	$-0.5 \pm 0.866i$
After switching	0.1126, 2.4875

TABLE 3.3: Closed-loop eigenvalues immediately before and after switching

Clearly, when the plant dynamics switches, the closed-loop system becomes unstable. In the free running adaptive control scheme, the adaptation occurs slowly and hence, the controller parameters cannot be adjusted quickly enough to prevent the system from becoming unstable. In contrast, for the resettable adaptive control scheme, upon switching, the controller parameters are reinitialized to values optimized for the new plant dynamics. Hence, the closed-loop plant is prevented from becoming unstable as can be seen from Figure 3.8.

In Narendra and Balakrishnan (1994, 1997); Narendra (2000); Narendra et al. (2003) and Autenrieth and Rogers (1999), the free running adaptive controller were put into the MMAC scheme for the purpose of ensuring convergence. However, the example we have provided demonstrates that this is not always true. Hence, we conclude that inclusion of the free running adaptive controller is not necessary.

3. For the system described in Section 3.2, it can be seen that state feedback does not provide any significant improvement over output feedback. In fact, it could be said that output feedback performed better. However, for the system of Section 3.3, it is clear that output feedback control is incapable of providing stability to the system. To understand this, we look at the characteristic polynomials of the two systems before and after switching. This is shown in Table 3.4.

System	**Subsystem**	**Characteristic polynomial**	**Poles**
Section 3.2	Σ_A	$s^2 + 1.7s + 0.43$	-1.3908, -0.3092
	Σ_B	$s^2 + 1.7s - 0.2$	-1.8105, 0.1105
Section 3.3	Σ_1, Σ_2	$s^2 - 0.8s + 0.64$	$0.4 \pm 0.6928i$

TABLE 3.4: Characteristic polynomials of plant subsystems

The points of interest are the unstable subsystems which are Σ_B for the system of Section 3.2 and both Σ_1 and Σ_2 for the system of Section 3.3. The instability of Σ_B is due to the negative value for the x term. Making x positive will stabilize the subsystem. Since this happens to be the parameter that is adjusted in output feedback control, the system is stabilizeable. In contrast, the cause of instability for $\Sigma_{1,2}$ is the negative $\frac{dx}{dt}$ term. This term is not accessed in output feedback control and therefore, the system cannot be stabilized.

4. In Figure 3.8, state feedback RSA control provides perfect reference model tracking for the system of Section 3.3. This is simply because, in this example, it was deemed that the plant dynamics are known. Hence, the candidate models match the plant subsystems exactly and the correct model is always chosen without the need for any cost function evaluation. Figure 3.9 shows the simulated response when the plant is subjected to perturbation. It can be seen that the parameter tuning scheme is capable of bringing the system response towards convergence. However, since the parameters are reset each time the plant dynamics switches, at these instances, the plant response jumps away from the reference target. The tuning scheme then drives the plant towards convergence before another switching instance brings it away again.

5. In the system of Section 3.2 (Figures 3.3 - 3.6), the plant is not known in advance. The instance of switching is also not predictable. As a result, model selection has to be based on the cost function index, evaluated for each candidate model. Since the cost function used takes into account past values, a short period of time is required before the correct candidate model can be identified. In view of this, there is a short delay before the switch at $t = 60s$ is recognized. This accounts for the spikes at the instance of switching observed in all cases, due to a mismatch between the

new plant dynamics and the (unchanged) controller parameters. This is similar to Point 2 except that in this case, the mismatch does not lead to instability.

Chapter 4

Extension to Multivariable Systems

Many real-life control problems involve plants that are multivariable in nature. Unlike single input single output (SISO) systems, control of multiple input multiple output (MIMO) plants is made complicated by cross-couplings between the inputs and outputs. Since each output may be influenced by more than one input, it is often not possible to obtain control parameters that would emulate the reference model in the same manner as was done in Chapter 3 for SISO systems. In this chapter, we study how the ideas from there can be extended to the multivariable case.

The organization of this chapter is as follows. In Section 4.1 we discuss the issue of decoupling multivariable systems by means of static state feedback control. We then extend the discussion to tunable adaptive control of multivariable systems in Section 4.2. In Section 4.3 we simulate the implementation of a switchable static feedback controller on an example MIMO hybrid system. The problem of moving to dynamic feedback for such a system is deliberated in Section 4.4. We conclude this chapter in Section 4.5 with a note on issues to be addressed when implementing the multiple model switching and tuning scheme on MIMO hybrid systems.

4.1 Decoupling by State Feedback

MIMO systems suffer from cross-couplings between the different inputs and outputs. The performance of each output may involve contributions from more than one input. As a result, controllers may need to perform contradicting actions on the same input at the same time in order to meet specifications on more than one output. In view of this, researchers have long sought out methods in which the system could be decoupled such that each output is affected by only one input.

One of the main early works on decoupling control of multivariable systems were those of Falb and Wolovich (1967); Gilbert (1969), which can be summarized as follows. Given a multivariable plant Σ, represented in state space form as

$$\begin{aligned} \tfrac{d}{dt}x(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t), \end{aligned} \tag{4.1}$$

where $u \in \mathbb{R}^{m\times 1}$ is the input vector, $x \in \mathbb{R}^{n\times 1}$ is the state vector, $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$ and $C \in \mathbb{R}^{m\times n}$, it can be determined if there exist a decoupling control law of the form

$$u(t) = Fx(t) + Gu_r(t) \tag{4.2}$$

where $u_r \in \mathbb{R}^{m\times 1}$ is the reference input vector and $F \in \mathbb{R}^{m\times n}$ and $G \in \mathbb{R}^{m\times m}$.

Define

$$\psi_i = \min\left\{j : C_i A^j B \neq 0, \quad j = 0, 1, \ldots, n-1\right\}$$

or

$$\psi_i = n - 1 \quad \text{if } C_i A^j B = 0\ \forall j,$$

where C_i is the i-th row of C,

$$\Psi_i = C_i A^{\psi_i} B,$$

and

$$\Psi = \begin{bmatrix} \Psi_1 \\ \vdots \\ \Psi_m \end{bmatrix}.$$

Then, Σ can be decoupled if and only if Ψ is nonsingular.

Having identified sufficient and necessary conditions for decoupling of multivariable plants by state feedback, Falb and Wolovich (1967) then showed that the set of all pairs F, G that decouple Σ are matrices F and G which satisfy

- rank $\left[\Omega^i(F)\right] = 1$ where

$$\Omega^i(F) = \begin{bmatrix} C_i(A+BF)^{n-1}B \\ C_i(A+BF)^{n-2}B \\ \vdots \\ C_i(A+BF)^{\psi_i}B \\ \emptyset \end{bmatrix} \qquad i = 1, \ldots, m,$$

  and $\emptyset$ is a zero matrix of order consistent with $\left[\Omega^i(F)\right]$ and

- $G = \Lambda\Psi^{-1}$ where Λ is a diagonal nonsingular matrix.

The number of closed loop poles that can be specified is $m + \sum\limits_{i=1}^{m} \psi_i$ and they can be specified by the choice of M_k where

$$M_k = \text{diag}\left[m_k^1, \ldots, m_k^m\right], \quad i = 1, \ldots, m, \quad k = 0, \ldots, \psi_i.$$

(Falb and Wolovich, 1967)

A later development was the concept of *the interactor matrix* introduced by Wolovich and Falb (1976). It was found that systems having a diagonal interactor matrix can be decoupled by static state feedback. Suppose the nonsingular MIMO plant Σ is represented in the transfer function form by the proper rational $m \times m$ transfer matrix $P(s)$. Then, the unique nonsingular $m \times m$ interactor matrix $\xi(s)$ is defined as

$$\xi(s) = L(s)\,\Phi(s)$$

where

$$L(s) = \begin{bmatrix} 1 & 0 & \ldots & 0 \\ l_{21}(s) & 1 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ l_{m1}(s) & l_{m2}(s) & \ldots & 1 \end{bmatrix},$$

$$\Phi(s) = \text{diag}[s^{f_i}], \quad f_i \geq 0, \; i = 1, \ldots m,$$

and $l_{ij}(s)$ is divisible by s (or is zero) such that

$$\lim_{t \to \infty} \xi(s)\,P(s) = K_p$$

with K_p nonsingular. It turns out that the interactor matrix is equivalent to the inverse of the plant's Hermite normal form and K_p is called the *high frequency gain* matrix. Obviously, not all multivariable plants have diagonal interactor matrices. It has been established that by introducing certain pre-compensators, the interactor matrix (or the Hermite normal form) can be diagonalized. Methods by which this can be achieved are found in (Singh and Narendra, 1984; Gibbens et al., 1993; Schwartz and Yan, 1995).

4.2 Model Reference Adaptive Control of Multivariable Systems

Singh and Narendra (1984) outlined *a priori* information that fulfil sufficient conditions for the use of tunable adaptive controllers on multivariable plants. These required information are:

1. The Hermite normal form (or equivalently, the interactor matrix) of the plant transfer matrix is known.

2. The high frequency gain matrix K_p satisfies

$$\Gamma K_p(s) + K_p^T(s)\Gamma = Q > 0$$

for some $\Gamma = \Gamma^T > 0$.

3. An upperbound ν on the observability index of the plant is known.

4. The zeros of the plant are located in the open left half plane.

The importance of the Hermite normal form H (or the interactor matrix) lies in that the reference model P_r needs to be of the form

$$P_r = HP_0$$

where P_0 is an arbitrary proper transfer matrix (Elliott and Wolovich, 1982; Sastry and Bodson, 1989). A diagonal interactor matrix also means that the plant can be decoupled.

The high frequency gain matrix K_p is required in order to specify a Lyapunov based stabilising parameter update law. This requirement has been a major stumbling block in the advancement of multivariable MRAC research, since in adaptive control, the plant transfer matrix is not known. Several methods of getting around this problem has been proposed. Ioannou and Sun (1996) assumed that a matrix S is known such that $K_pS = (K_pS)^T > 0$. Imai et al. (2001, 2004); Costa et al. (2003); Tao (2003) proposed the LDU, SDU and LDS factorizations of K_p where the matrices L is unity lower triangular, D is diagonal, U is unity upper triangular and S is symmetric positive definite. With these factorizations, the requirement of knowledge of K_p is reduced to only the knowledge of the signs of the diagonal elements of D.

It should be noted that the nonsingular matrix K_p is closely related to the plant transfer matrix P and its Hermitian normal form $H(s)$ in that

$$\lim_{s\to\infty} H^{-1}(s)P(s) \triangleq K_p$$

(Sastry and Bodson, 1989).

4.3 Switchable Static Control of Multivariable Switching Systems

Similar to the discussion in Chapter 3 for SISO systems, we would like to control multivariable hybrid systems such that the plant output emulates that of a reference model. In this section, we study the use of state feedback controllers where the control parameters are non-tunable but would switch accordingly with a switch in plant parameters. From

Section 4.1, it is clear that all subsystems of the plant must be decoupleable by state feedback i.e. Ψ for every subsystem needs to be nonsingular. Additionally, the plant can only emulate a reference model if the required feedback and feedforward parameter matrices, F and G respectively, meet the conditions rank $\left[\Omega^i(F)\right] = 1$ and $G = \Lambda\Psi^{-1}$. Clearly the choice of acceptable reference models is limited.

4.3.1 Simulation

Simulations were performed on a multivariable switching system as defined below. In view of the difficulty in finding a reference model that could be emulated by every subsystem, it was decided that a different reference model should be used for every subsystem where each reference model is known to be trackable by that particular subsystem. Using example decoupleable systems from Falb and Wolovich (1967), the system Σ consists of two subsystems $\Sigma_A = \{A_A, B_A, C_A\}$ and $\Sigma_B = \{A_B, B_B, C_B\}$ as follows:

$$A_A = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad B_A = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 1 & 0 \end{bmatrix}, \quad C_A = \begin{bmatrix} 1 & 1 & -1 \\ 0 & 1 & 0 \end{bmatrix} \quad \text{for } t \geq 37s,$$

$$A_B = \begin{bmatrix} 0 & 1 & 0 \\ 2 & 3 & 0 \\ 1 & 1 & 1 \end{bmatrix}, \quad B_B = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad C_B = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{for } t < 37s$$

or, in the transfer matrix format

$$P_A = \begin{bmatrix} \frac{s-1}{s^3-3s^2+3s-1} & \frac{1}{s-1} \\ \frac{1}{s-1} & 0 \end{bmatrix} \quad \text{for } t \geq 37s, \quad P_B = \begin{bmatrix} \frac{s+1}{s^2-3s-2} & 0 \\ \frac{s+1}{s^3-4s^2+s+2} & \frac{1}{s-1} \end{bmatrix} \quad \text{for } t < 37s.$$

The reference model was chosen to be

$$P_A^r = \begin{bmatrix} 0 & \frac{2}{s+2} \\ \frac{2}{s+4} & 0 \end{bmatrix} \quad \text{for } t \geq 37s, \quad P_B^r = \begin{bmatrix} \frac{3s+3}{s^2+6s+3} & 0 \\ 0 & \frac{3}{s+6} \end{bmatrix} \quad \text{for } t < 37s.$$

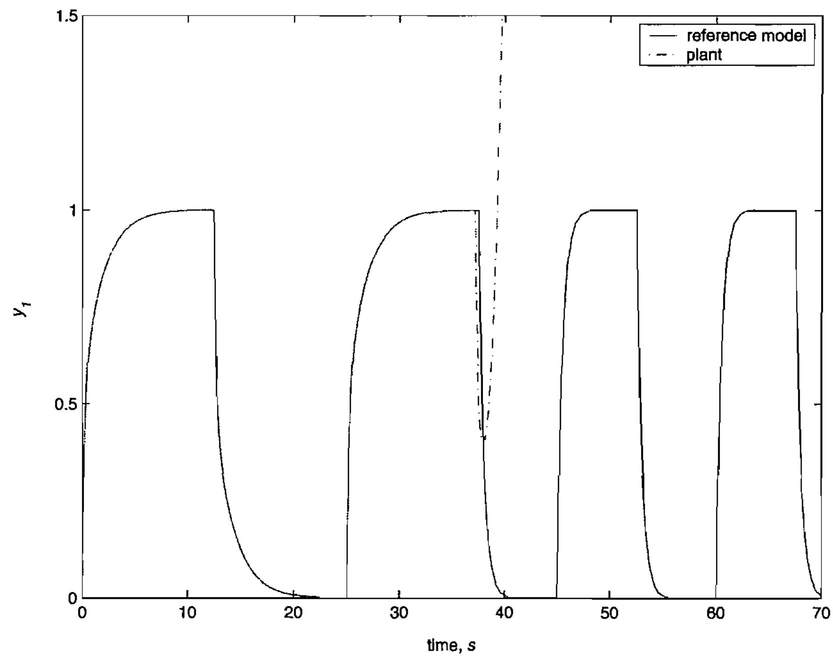
Using methods from Falb and Wolovich (1967), the controller parameters $\{F_A, G_A\}$ for Σ_A and $\{F_B, G_B\}$ for Σ_B were chosen to be

$$F_A = \begin{bmatrix} 0 & -2 & -3 \\ -3 & -5 & 4 \end{bmatrix}, \quad G_A = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$

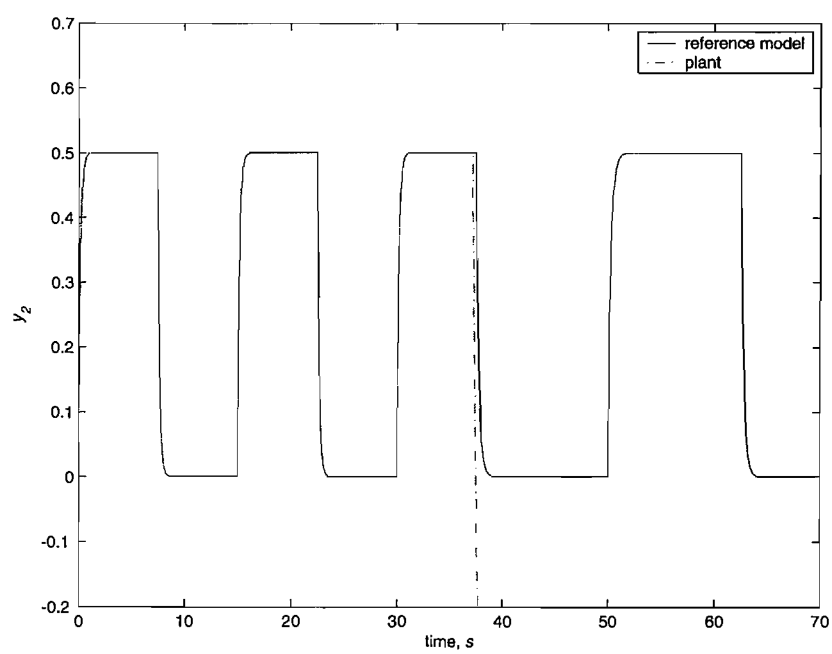
and

$$F_B = \begin{bmatrix} -5 & -9 & 0 \\ -1 & -1 & -7 \end{bmatrix}, \quad G_B = \begin{bmatrix} 3 & 0 \\ 0 & 3 \end{bmatrix}.$$

The simulation results are shown in Figure 4.1



(a) Output 1



(b) Output 2

FIGURE 4.1: Switching from Σ_B to Σ_A at $t = 37s$

4.3.2 Discussion

The results shown by Figure 4.1 indicate that the control parameters $\{F_B, G_B\}$ was successful in controlling Σ_B before the plant switches to Σ_A. After the switch to Σ_A, the system became unstable although the control parameters did switch to $\{F_A, G_A\}$. It appears as if $\{F_A, G_A\}$ was incapable of controlling Σ_A.

Figure 4.2 shows the performance of $\{F_A, G_A\}$ on Σ_A without any plant switching taking place. Curiously, it would appear that perfect performance was obtained. Next, the subsystem sequence was swapped, the system starting out as Σ_A and switches to Σ_B at $t \geq 37s$. The simulation result is shown by Figure 4.3. The response obtained was almost perfect, the only blemish being a spike occurring at the instance of switching for y_1.

These observations could be understood by examining the closed loop transfer matrices of the subsystems. For Σ_A, the closed loop transfer matrix P_A^{cl} is

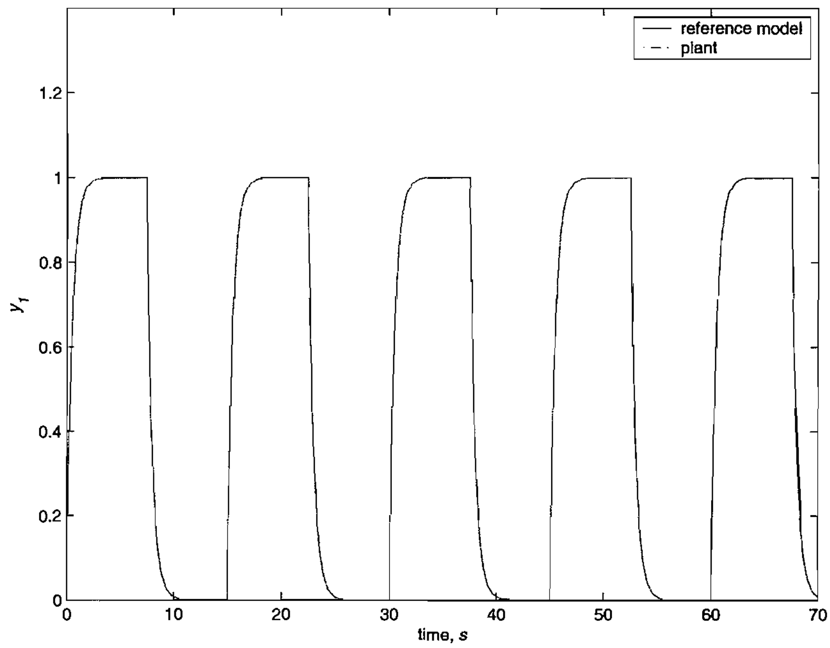
$$P_A^{cl} = \begin{bmatrix} \frac{1.507\times10^{-14}}{(s-1)(s+2)(s+4)} & \frac{2}{s+2} \\ \frac{2(s-1)}{(s-1)(s+4)} & 0 \end{bmatrix}.$$

Comparing this with the reference model

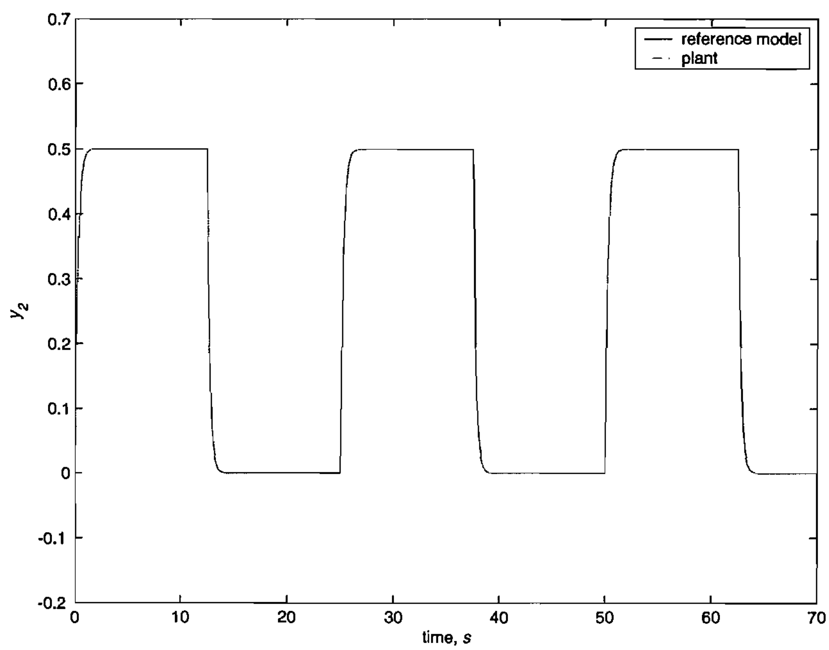
$$P_A^{r} = \begin{bmatrix} 0 & \frac{2}{s+2} \\ \frac{2}{s+4} & 0 \end{bmatrix},$$

it is clear that P_A^{cl} only matches P_A^r if $P_A^{cl}(1,1)$ can be taken as 0 and, for $P_A^{cl}(2,1)$, the pole and zero at $s = 1$ completely cancels each other out. Evidently, although $P_A^{cl}(1,1)$ is very small, it has a contributing effect where the pole at $s = 1$ leads to the instability of y_1. Similarly, the pole-zero cancellation in $P_A^{cl}(2,1)$ does not negate the instability of y_2 caused by the unstable pole. To confirm this, simulations were carried out on an uncontrolled open loop switching plant having the dynamics of P_B^r for $t < 37s$ and P_A^r for $t \geq 37s$. The results shown in Figure 4.4 suggest that the instability has been removed with the absence of the poles at $s = 1$. However, the small value of the numerator term of $P_A^{cl}(1,1)$ and the pole-zero cancellation of $P_A^{cl}(2,1)$ make the effects of the unstable pole unnoticeable only if the system starts out as Σ_A and the system is not perturbed.

Evidently, while the method of Falb and Wolovich (1967) is capable of determining decoupling state feedback parameters for some MIMO plants, the method could lead to closed loop systems that are unstable as it allows for the pole-zero cancellation of unstable poles. It is a well known fact that an unstable pole cannot be cancelled out by a zero at the same location. An alternative approach worth considering is the use of pre-compensators to diagonalize the interactor matrices for each subsystem. By diagonalizing the interactor matrices, the choice of controller parameters and hence, the choice of reference models available to the designer could be increased.

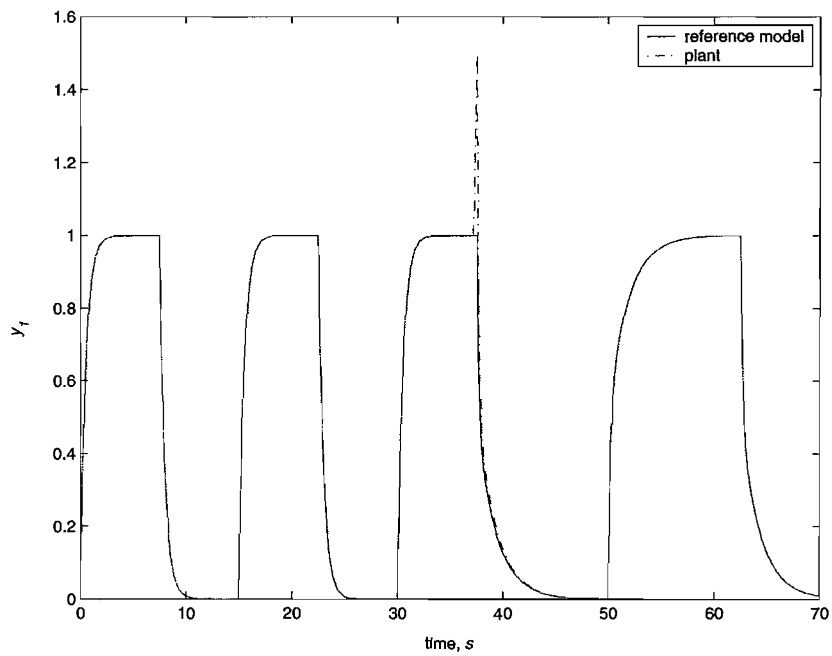


(a) Output 1

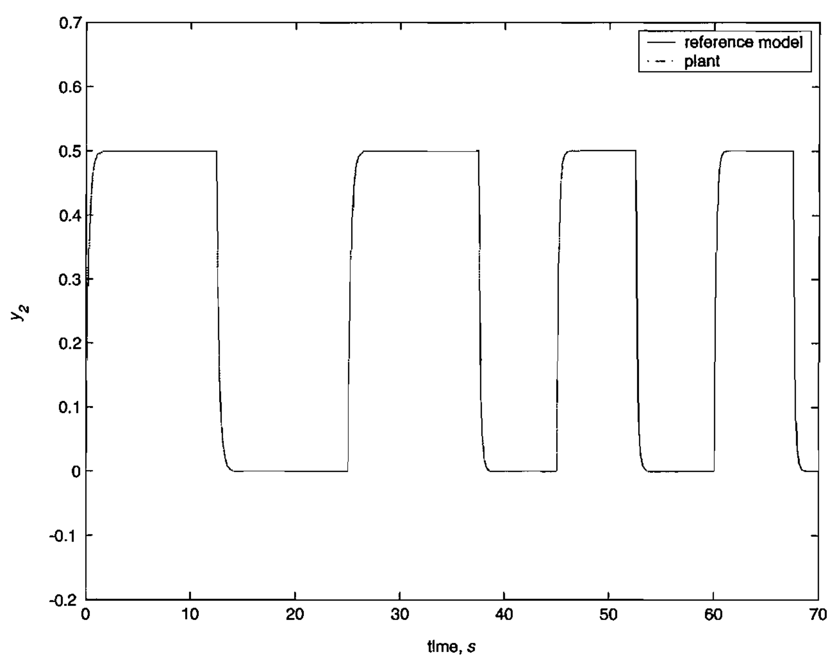


(b) Output 2

FIGURE 4.2: Σ_A response without plant switching

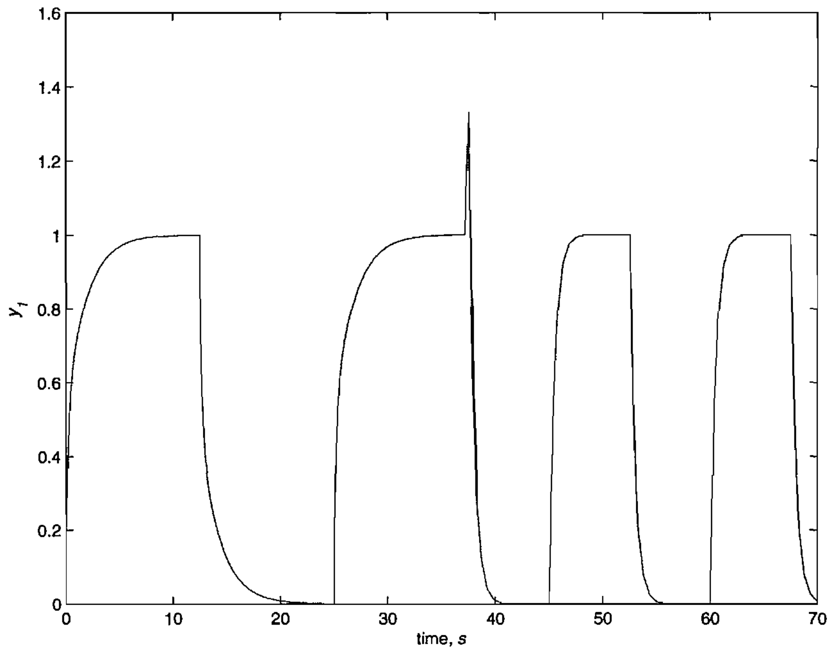


(a) Output 1

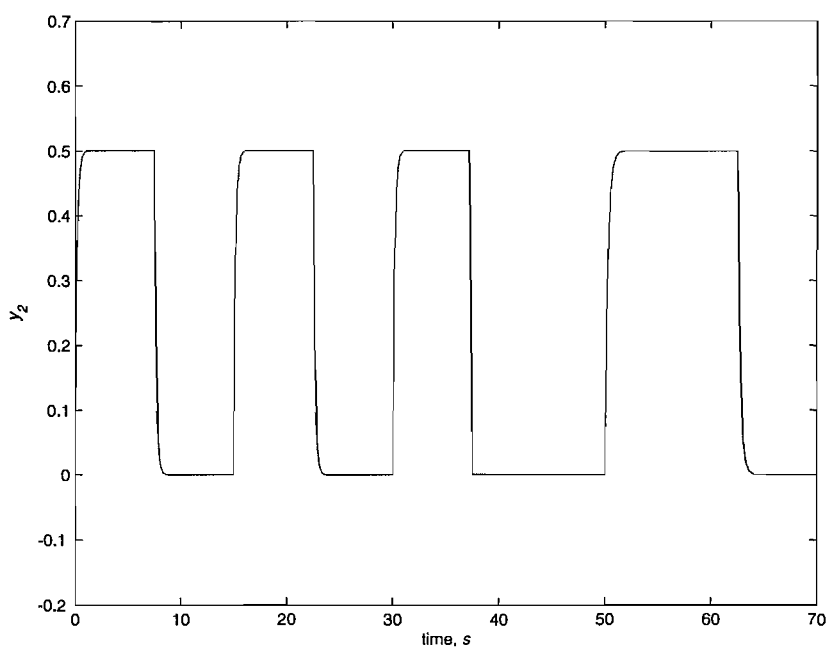


(b) Output 2

FIGURE 4.3: Switching from Σ_A to Σ_B at $t = 37s$



(a) Output 1



(b) Output 2

FIGURE 4.4: Switching from P_B^r to P_A^r at $t = 37s$

4.4 Implementation of Dynamic Controllers

Static state feedback control as discussed in Section 4.3 can only achieve the desired performance if the plant is known well. In cases where the plant is not known well, or in the presence of unmodelled dynamics, it would be desirable to be able to dynamically tune the control parameters so as to bring convergence between plant and reference model outputs.

In order to achieve adaptive control of the multivariable system (4.1) using tunable state feedback, the following assumptions are needed (Tao, 2003)

1. There exist constant matrices $F \in \mathbb{R}^{m\times n}$ and $G \in \mathbb{R}^{m\times m}$, G nonsingular, such that
$$A + BF = A_r, \quad BG = B_r. \tag{4.3}$$
2. The symmetric and positive definite matrix $\Gamma \in \mathbb{R}^{m\times m}$ such that
$$F\Gamma = (F\Gamma)^T = \Gamma^T F^T > 0 \tag{4.4}$$
is known.

From Assumption 2, the following parameter update laws can be shown to be stabilizing (Tao, 2003):

$$F = -\Gamma^T B_r^T P e x^T,$$
$$G = -\Gamma^T B_r^T P e u_r^T$$

where the constant matrix $P \in \mathbb{R}^{n\times n}$, $P = P^T > 0$ satisfies

$$PA_r + A_r^T P = -Q$$

for some constant matrix $A \in \mathbb{R}^{n\times n}$, $Q = Q^T > 0$ and $e = x - x_r$ is the tracking error. Apart from these assumptions, the requirements explained in Section 4.3 which are that every subsystem of the plant must be decoupleable and that the reference model must also be decoupled, must be met too.

Unlike in the SISO case, MIMO systems in general are not normally transformable to the control canonical form

$$\frac{d}{dt}x(t) = \begin{bmatrix} 0 & 1 & \cdots & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ a_1 & a_2 & a_3 & \cdots & a_n \end{bmatrix} x(t) + \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & & \vdots \\ \vdots & & \vdots \\ 0 & \cdots & 0 \\ b_1 & \cdots & b_m \end{bmatrix} u(t).$$

Because of this, obtaining decoupling matrices F and G that satisfy Equation (4.3) is not trivial. An alternative approach would be to place a pre-compensator matrix that diagonalizes the interactor matrix in cascade to the plant as described by Singh and Narendra (1984); Gibbens et al. (1993); Schwartz and Yan (1995). In cases where the plant model is not known, the high frequency gain matrix would also not be known. Factorization methods for solving this problem proposed by Imai et al. (2001, 2004); Costa et al. (2003); Tao (2003) should be looked into as possible solutions for use on multivariable hybrid systems. However, due to time constraints, we were not able to pursue these ideas and we leave them as topics for future research.

4.5 Conclusion

We have discussed several issues that are of concern when extending the multiple model switching and tuning (MMST) method to MIMO hybrid systems. The MMST scheme makes use of a combination of fixed control parameters for candidate models of the plant and a resettable adaptive controller with tunable parameters that start from the values of the fixed controllers. With this in mind, we considered the use of the state feedback control structure for the multivariable MMST scheme.

Several points need to be taken into consideration, in addition to those outlined by Singh and Narendra (1984), when an MMST scheme is to be implemented on a multivariable hybrid system comprising of N subsystems. These are:

For systems where the plant subsystems are well known, it must be ensured that

1. All subsystems must be decoupleable. As this can be quite restrictive, the possibility of using pre-compensators to diagonalize the interactor matrices of the subsystems should be looked into.
2. The reference model should be decoupled and have the plant's interactor matrix as a factor. As it could be quite difficult to find a reference model that meets these requirements for all subsystems, a possible solution would be to use a different reference model for each subsystem.
3. There exist matrices F_i and G_i, $i = 1, \ldots, N$ such that Equation (4.3) is satisfied for the i-th subsystem Σ_i.
4. The F_i and G_i matrices must stabilize as well as decouple Σ_i.
5. There exist Γ_i such that Equation (4.4) is satisfied for Σ_i.

For systems where the plant subsystems are not well known, the following additional considerations need also be taken into account.

6. To get around the problem of unknown high frequency gain matrices of the subsystems, the factorization methods proposed by Imai et al. (2001, 2004); Costa et al. (2003); Tao (2003) should be considered.

7. Singh and Narendra (1984) showed that the interactor matrices of the subsystems can be diagonalized with only the knowledge of the relative degrees of the elements of the subsystems transfer matrix.

Clearly, there is an abundance of research work that needs to be done before a practical implementation of MMST on multivariable hybrid systems can be achieved. We have provided several points of note for this purpose, which we were unable to pursue due to time constraints. These are therefore left as topics for further research.

Chapter 5

Stabilizeability of Switched Systems

One way to stabilize a switched system is to find a quadratic Lyapunov function (also known as quadratic stabilization in the literature). For a switched system with a number of linear models, this problem becomes one of finding a common quadratic Lyapunov function for a set of matrices. Some progress has been made in this area, e.g. (Branicky, 1998; Liberzon et al., 1999; Ooba and Funahashi, 1997; Shorten and Narendra, 2000; Decarlo et al., 2000) but the problem is still open and in this chapter we give some new results.

We propose, for a class of switching systems, a method by which it could be determined if there exists a common Lyapunov function for all subsystems. The class of switching systems studied here are restricted to systems that take the following form:

$$\Delta x = f_i\left(x, u\right), \quad i = 1, 2, \ldots, k+1 \tag{5.1}$$

where $\Delta x = \frac{d}{dt}x$ for continuous-time systems and $\Delta x = x\left(k+1\right)$ for discrete-time systems, $x \in \mathbb{R}^n$ and $u \in \mathbb{R}^m$. In this study, we restrict our discussion to systems where $n = 2$ and $m = 1$.

The organization of this Chapter is as follows. In Section 5.1, we present results for the case when the subsystems are modelled in continuous-time. For the case when the subsystems are modelled in discrete-time, the results are given in Section 5.2. The chapter is concluded in Section 5.3 with a discussion on both these results.

5.1 Stabilizeability of continuous-time switched systems

Consider the switching system denoted by

$$\dot{x} = A_{\sigma(x,t)}x + B_{\sigma(x,t)}u_{\sigma(x,t)} \tag{5.2}$$

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$ and $\sigma(x,t) : \mathbb{R}^n \times [0,\infty) \to \Lambda$ is an arbitrary mapping unless where stated. We assume $\Lambda = \{1, 2, \ldots, N\}$.

The problem is to ensure quadratic stabilization of (5.2), i.e. we require the existence of a state feedback control law and a quadratic Lyapunov function $x^T Mx$, $M = M^T > 0$ (written from this point onwards as $M > 0$) such that the feedback switching models share $x^T Mx$ as a common quadratic Lyapunov function. Formally, we have the following problem definition.

Definition 5.1. System (5.2) is said to be quadratically stabilizeable with observeability $\sigma(x,t)$ if there exists a common Lyapunov function $x^T Mx$ with $M > 0$, and a set of state feedback control laws $u_i = A_i + B_i K_i$ such that $A_i + B_i K_i$, $i = 1, 2, \ldots, N$, share a common quadratic Lyapunov function $x^T Mx$.

Clearly a necessary condition here is that each model is stabilizeable. In actual fact, however, the following result shows that we can replace this assumption by controllability.

Lemma 5.2. *If a single-input system of the form considered here, written (A, b), is stabilizeable but not controllable then for any $M > 0$ there exists a suitable state feedback law such that $x^T Mx$ is a quadratic Lyapunov function of the closed loop system.*

Proof. We assume $b = \begin{bmatrix} 0 & 1 \end{bmatrix}^T$, which incurs no loss of generality. The uncontrollable system then becomes

$$\dot{x}(t) = \begin{bmatrix} a_{11} & 0 \\ a_{21} & a_{22} \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t).$$

Also, since stabilizeability holds, $a_{11} < 0$ and, without loss of generality, we assume $a_{11} = -1$. The closed loop matrix $\widetilde{A} = A + bK$ is

$$\widetilde{A} = \begin{bmatrix} -1 & 0 \\ \alpha & \beta \end{bmatrix},$$

with α and β chosen arbitrarily.

Assume now that $M = \begin{bmatrix} 1 & m_2 \\ m_2 & m_3 \end{bmatrix}$. Then, it is required to prove that we can select α and β such that

$$M\widetilde{A} + \widetilde{A}^T M < 0.$$

where

$$Q := M\widetilde{A} + \widetilde{A}^T M = \begin{bmatrix} 2(\alpha m_2 - 1) & \alpha m_3 + (\beta - 1)m_2 \\ \alpha m_3 + (\beta - 1)m_2 & 2\beta m_3 \end{bmatrix}.$$

Hence, since $m_3 > 0$, we need to choose α and β such that $\det(Q) > 0$, i.e. such that

$$-(\alpha m_3 - \beta m_2)^2 + 2\alpha m_2 m_3 + 2\beta m_2^2 - m_2^2 - 4\beta m_3 > 0.$$

Choosing

$$\alpha = \frac{\beta m_2}{m_3}, \quad \beta < -\frac{m_2^2}{4\det(M)} < 0$$

ensures that this condition holds and the proof is complete. □

The following result comes from the well known fact that a single input system can be uniquely transformed to its canonical form.

Lemma 5.3. *Let (A, b) be a single input linear system. Then, there exists a unique state transformation matrix C which converts the system into the Brunovsky canonical form*

$$CAC^{-1} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ & & & \cdots & \\ & & & \cdots & 1 \\ a_1 & a_2 & a_3 & \cdots & a_n \end{bmatrix}, \quad Cb = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ \vdots \\ 1 \end{bmatrix}$$

where

$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} b & Ab & A^2b & \ldots & A^{n-1}b \end{bmatrix}^{-1} A^n b$$

with

$$D_n = b,$$

$$D_{i-1} = AD_i - a_i b; \quad i = n, n-1, \ldots, 2,$$

$$D = \begin{bmatrix} D_1 & D_2 & D_3 & \ldots & D_n \end{bmatrix}$$

and

$$C = D^{-1}$$

Lemma 5.4. *Given*

$$M = \begin{bmatrix} m_1 & m_2 \\ m_2 & m_3 \end{bmatrix} > 0,$$

there exists a feedback control law $u = Kx = \begin{bmatrix} k_1 & k_2 \end{bmatrix} x$ *such that*

$$\widetilde{A} = A + bK = \begin{bmatrix} 0 & 1 \\ \alpha & \beta \end{bmatrix}$$

(i.e. the resulting closed loop system) has M *defining its quadratic Lyapunov function if, and only if,* $m_2 > 0$.

Proof. No loss of generality arises from assuming $m_1 = 1$. Then, for $M > 0$ to have the required property, we require that $M\widetilde{A} + \widetilde{A}^T M < 0$, or equivalently,

$$Q := \begin{bmatrix} 2\alpha m_2 & 1 + \alpha m_3 + \beta m_2 \\ 1 + \alpha m_3 + \beta m_2 & 2(m_2 + \beta m_3) \end{bmatrix} < 0.$$

Moreover, β is the trace of $\widetilde{A}$ and hence, $\beta < 0$ is a necessary condition for closed loop stability. Also, $m_2 = 0$ is not allowed, and $\alpha m_2 < 0$ is necessary.

Suppose now that $m_2 < 0$ and $\alpha > 0$. Then, for $Q < 0$ to hold, we require that $\det(Q)$ is positive. It is also easy to see that the maximum value of this determinant occurs when $\beta m_2 = \alpha m_3 - 1$, and that it is negative. Hence, $m_2 > 0$ is a necessary condition for $Q < 0$.

If $m_2 > 0$ and $\alpha < 0$, we can simply choose any $\alpha < 0$ and let $\beta = \frac{\alpha m_3 - 1}{m_2}$. Then, $\det(Q) = -4\alpha(m_3 - m_2^2) > 0$ and sufficiency is established. □

Definition 5.5. A matrix

$$M = \begin{bmatrix} m_1 & m_2 \\ m_2 & m_3 \end{bmatrix}$$

is said to be 'an equivalence' to the controllable canonical form if $m_2 > 0$.

Lemma 5.6. *Given a nonsingular matrix*

$$C = \begin{bmatrix} c_1 & c_2 \\ c_3 & c_4 \end{bmatrix},$$

there exists 'an equivalence' matrix $M > 0$ *such that* $C^T MC$ *also has this property if, and only if, the quadratic equation*

$$c_3 c_4 x^2 + (c_2 c_3 + c_1 c_4)x + c_1 c_2 > 0 \tag{5.3}$$

has positive solution $x > 0$.

Proof. To prove sufficiency, we proceed as follows.

Given $M > 0$ with $m_2 > 0$, and calculating $\widetilde{M} = C^T M C$, we have

$$H(m_1, m_2, m_3) := \widetilde{m}_{12} = c_3 c_4 m_3 + (c_2 c_3 + c_1 c_4) m_2 + c_1 c_2 m_1,$$

and from Equation (5.3), $H(1, x, x^2) > 0$. Hence, by continuity, there exists a small enough $\epsilon > 0$ such that $H(1, x, x^2 + \epsilon) > 0$. Setting $m_1 = 1$, $m_2 = x$ and $m_3 = x^2 + \epsilon$ now gives an M with the required property.

To prove necessity, we can assume, without loss of generality, that there exists

$$M = \begin{bmatrix} 1 & m_2 \\ m_2 & m_3 \end{bmatrix} > 0$$

such that both M and $C^T M C$ have the desired property, i.e. $m_2 > 0$ and $H(1, x, x^2) > 0$. Now, if $c_3 c_4 > 0$, clearly (5.3) has a positive solution. If $c_3 c_4 < 0$, then, since $m_3 > m_2^2$,

$$H(m_1, m_2, m_2^2) \geq H(m_1, m_2, m_3) > 0, \tag{5.4}$$

and m_2 is a positive solution of Equation (5.3). □

For a given state transformation matrix C, the solution of Equation (5.3) consists of one or two open intervals, or it could be empty. We use I to denote the set of solutions.

Still with the single input assumption, let $\Lambda = \{1, 2, \ldots, N\}$ be a finite set. Then, for each switched model, we denote the state transformation matrix which converts it to the Brunovsky canonical form by C_i, $i = 1, 2, \ldots, N$. Moreover, by Lemma 5.3, each transformation matrix is uniquely defined.

Now, let $z_i = C_i x$, $i = 1, 2, \ldots, N$, and set $T_i = C_1 C_{i+1}^{-1}$, $i = 1, 2, \ldots, N-1$. In this notation, T_i is the state transformation matrix from z_{i+1} to z_1, i.e.

$$z_1 = C_1 x = C_1 C_{i+1}^{-1} z_{i+1} = T_i z_{i+1}.$$

Next, we classify $T_i = \left(t^i\right)_{j,k}$ into the following three categories:

$$\begin{aligned} S_p &= \{i \in \Lambda \mid t_3^i t_4^i > 0\}, \\ S_n &= \{i \in \Lambda \mid t_3^i t_4^i < 0\}, \\ S_z &= \{i \in \Lambda \mid t_3^i t_4^i = 0\}. \end{aligned}$$

Then, $\Lambda = S_p \cup S_n \cup S_z$. Also, for $i \in S_z$, define

$$\left.\begin{aligned} r_i &= t_2^i t_3^i + t_1^i t_4^i \\ s_i &= t_1^i t_2^i \end{aligned}\right., \quad i \in S_z, \tag{5.5}$$

and, for the other cases, define

$$p_i = \frac{t_1^i}{t_3^i}, \quad q_i = \frac{t_2^i}{t_4^i}, \quad i \in S_p \cup S_n. \tag{5.6}$$

For each $i \in S_z$, we define a linear form as

$$L_i = r_i x + s_i, \quad i \in S_z$$

and, for each $i \in S_p$ or $i \in S_n$, we define a quadratic form as

$$Q_i = x^2 + (p_i + q_i)x + p_i q_i, \quad i \in S_p \cup S_n. \tag{5.7}$$

Then, by Lemma 5.6, we may obtain (or solve) x from

$$\begin{aligned} Q_i(x) &< 0, \quad i \in S_n, \\ Q_i(x) &> 0, \quad i \in S_p, \\ L_i(x) &> 0, \quad i \in S_z. \end{aligned}$$

Note also that the polynomial $Q_i(x)$ of Equation (5.7) has roots $\{-p_i, -q_i\}$ and hence, we can define the solution set as follows:

If $i \in S_p$, we define an open set as

$$I_i = (-\infty, \min(-p_i, -q_i) \cup (\max(-p_i, -q_i), \infty).$$

If $i \in S_n$, we define an open set as

$$I_i = (\min(-p_i, -q_i) \cup (\max(-p_i, -q_i)).$$

If $i \in S_z$, and since T_i is nonsingular, it follows that $r_i \neq 0$. Hence, we can define an open set as

$$I_i = \begin{cases} (-\frac{s_i}{r_i}, \infty), & r_i > 0, \\ (-\infty, -\frac{s_i}{r_i}), & r_i < 0. \end{cases}$$

Hence, the following result now follows immediately from Lemma 5.6.

Theorem 5.7. *1. A sufficient condition for the switched system (5.1) to be quadratically stabilizeable is*

$$I = \bigcap_{i=1}^{N-1} I_i \neq \emptyset. \tag{5.8}$$

2. If all $i \in S_p$, $i = 1, \cdots, N-1$, then the switched system (5.1) is always stabilizeable.

3. If all $i \in S_n$, $i = 1, \cdots, N-1$, (5.8) is also necessary.

Proof. To prove 1, choose $m_2 \in I$ and $m_3 = m_2^2 + \epsilon$. Then, it follows immediately that, for small enough ϵ, the corresponding matrix (in z_1 co-ordinates)

$$M = \begin{bmatrix} 1 & m_2 \\ m_2 & m_3 + \epsilon \end{bmatrix}$$

becomes (for suitably chosen controls) the common quadratic Lyapunov function.

In the case of 2, choose m_2 large enough and then, $m_2 \in I$.

In the case of 3, the inequality (5.4) from the proof of Lemma 5.6 shows that $m_2 \in I$, and therefore, $I \neq \emptyset$. □

In the case of $N = 2$, the above result is, from Lemma 5.6, necessary and sufficient. In general however, this property does not hold as the example given in Section 5.1.2 demonstrates. The following result gives such conditions.

Theorem 5.8. *Let $\Lambda = \{1, 2, \ldots, N\}$. Then, the system considered here is quadratically stabilizeable if, and only if, there exists a positive x such that*

$$\begin{aligned} \min_{i \in S_n} Q_i(x) &< 0, \\ \min_{i \in S_p} Q_i(x) &> \min_{i \in S_n} Q_i(x), \\ L_i(x) &> 0, \quad I \in S_z. \end{aligned} \tag{5.9}$$

Proof. Assume that there is a quadratic Lyapunov function in z_1 coordinates which is expressed as

$$M_1 = \begin{bmatrix} 1 & m_2 \\ m_2 & m_3 \end{bmatrix}$$

and, by Lemma 5.4, $m_2 > 0$. It is easy to see from the proof of Lemma 5.6 that M_1 is a common quadratic Lyapunov function for the other models if, and only if, $H_i(1, m_2, m_3) > 0$, $i = 1, 2, \ldots, N-1$, or

$$\begin{aligned} m_3 + (p_i + q_i)m_2 + p_i q_i &> 0, \quad i \in S_p, \\ m_3 + (p_i + q_i)m_2 + p_i q_i &< 0, \quad i \in S_n, \\ r_i m_2 + s_i &> 0, \quad i \in S_z. \end{aligned} \tag{5.10}$$

Also, since $m_3 > m_2^2$, we can rewrite the first two equations in (5.10) as

$$\begin{aligned} e + m_2^2 + (p_i + q_i)m_2 + p_i q_i &> 0, \quad i \in S_p, \\ e + m_2^2 + (p_i + q_i)m_2 + p_i q_i &< 0, \quad i \in S_n \end{aligned} \tag{5.11}$$

where $e > 0$, and the necessity of (5.9) is obvious.

To prove sufficiency, assume that there is a solution x such that $\min_{i \in S_p} Q_i(x) > 0$. Then, we can choose $m_2 = x$ and $m_3 = x^2 + \epsilon$, with $\epsilon > 0$ small enough to ensure that (5.10) holds. Otherwise, set $w = \min_{i \in S_p} Q_i(x) \leq 0$. Then choose $m_2 = x$ and

$$m_3 = x^2 + \frac{1}{2}\left(\min_{i \in S_p} Q_i(x) - \max_{i \in S_p} Q_i(x)\right) - w,$$

and it is easy to see that (5.10) holds. Hence, the matrix

$$M = \begin{bmatrix} 1 & m_2 \\ m_2 & m_3 \end{bmatrix}$$

in this case meets the requirement. □

Return now to Theorem 5.7. Then, in fact, it has been proven that the system is stabilizeable if all $i \in S_p$ or, if all $i \in S_n$, $i = 1, \ldots, N-1$, then (5.8) is also necessary. Moreover, since I is easily computed, it is easy to use. The example from Section 5.1.2 shows that, in general, this condition is not necessary. When $N \leq 3$, however, we have the following:

Corollary 5.9. *If $N \leq 3$, then (5.8) is also necessary.*

Proof. For the case when $N = 2$, it was proved in Lemma 5.6. To prove this result for the case when $N = 3$, we construct T_1 and T_2. If both 1 and 2 are in S_p (or S_n), the result has been proven in Theorem 5.7. Without loss of generality, we can assume that $1 \in S_p$ and $2 \in S_n$ and to establish necessity, we assume $I_1 \cap I_2 = \emptyset$. Then

$$p_1 \geq p_2 \geq q_2 \geq q_1.$$

Again, without loss of generality, we can assume $p_1 \geq q_1$ and $p_2 \geq q_2$. Now, it is obvious that

$$Q_2(x) \geq Q_1(x), \quad x \in (q_2, p_2), \tag{5.12}$$

and the result follows immediately. □

5.1.1 Control Design

To begin, first note that the first inequality in (5.9) is equivalent to

$$\min\{p_j, q_j\} < x < \max\{p_j, q_j\}, \quad j \in S_n$$

and the second inequality in this set is equivalent to

$$(p_i + q_i - p_j - q_j)x + p_i q_i - p_j q_j > 0, \quad i \in S_p, \ j \in S_n$$

(or each Q_i, $i \in S_p$ is greater than each Q_j, $j \in S_n$). Also, since we seek a positive solution, we have the following result.

Corollary 5.10. *The switched system considered here is quadratically stabilizeable if, and only if, the following set of linear inequalities have a solution*

$$\begin{aligned} \min\{p_j, q_j\} < x < \max\{p_j, q_j\}, &\quad j \in S_n, \\ (p_i + q_i - p_j - q_j)\, x + p_i q_i - a_j q_j > 0, &\quad i \in S_p,\ j \in S_n, \\ r_i x + s_i > 0, &\quad i \in S_z, \\ x > 0. & \end{aligned} \tag{5.13}$$

Recalling the proof of Lemma 5.4, a stabilizing control law is easily constructed.

Theorem 5.11. *Let (A, b) be a canonical system of the form*

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ a_{21} & a_{22} \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \tag{5.14}$$

and

$$M = \begin{bmatrix} m_1 & m_2 \\ m_2 & m_3 \end{bmatrix}$$

with $m_2 > 0$. Then $x^T M x$ is a quadratic Lyapunov function for the closed loop system under the action of the control law

$$u = kx = \begin{pmatrix} (\alpha - a_{21}) & \left(\frac{\alpha m_3 - m_1}{m_2} - a_{22}\right) \end{pmatrix} x \tag{5.15}$$

where $\alpha < 0$ is an arbitrary real number.

Note that Equation (5.15) is not unique.

The following is a systematic procedure for implementing the design method just developed.

- **Step 1** Use Lemma 5.3 to obtain the state transition matrices C_i such that in the coordinate frame z_i, the i-th switching model is in the Brunovsky canonical form.
- **Step 2** Define another set of state transformation matrices $T_i = C_1 C_{i+1}^{-1}$, $i = 1, \ldots, N-1$, such that

$$z_1 = T_i z_{i+1}, \quad i = 1, \ldots, N-1.$$

- **Step 3** Calculate a_i and b_i by (5.6) if $i \in S_p \cup S_n$, and c_i and d_i by (5.5) if $i \in S_z$.
- **Step 4** Construct the system of inequalities (5.13) and find a solution $x = x_0$. *Note: If there is no solution, the problem considered has no solution.*

- **Step 5** Using the inequalities (5.11), set $m_2 = x_0$ to find a positive solution $e > 0$. Set $m_3 = m_2^2 + e$. Construct a positive definite matrix

$$M_1 = \begin{bmatrix} 1 & m_2 \\ m_2 & m_3 \end{bmatrix} > 0$$

which is a common quadratic Lyapunov function for all switching models with certain feedback control laws. *Note: If Step 4 has a solution, then there exist solutions for the inequalities (5.11).*

- **Step 6** Convert M_1 to each canonical coordinate system using

$$M_{i+1} = T_i^T M_1 T_i, \quad i = 1, \ldots, N-1.$$

Convert model (A_i, b_i) to its canonical representation

$$\widetilde{A}_i = C_i A C_i^{-1}, \quad \widetilde{b}_i = \begin{bmatrix} 0 & 1 \end{bmatrix}^T, \quad i = 1, \ldots, N.$$

Use (5.15) to construct the feedback gains $k_i, \quad i = 1, \ldots, N$.

- **Step 7** Revert to the original coordinate x, where the controls are

$$K_i = k_i C_i, \quad i = 1, \ldots, N.$$

The common quadratic Lyapunov function for all closed loop switching models is then

$$M = C_1^T M_1 C_1.$$

5.1.2 Example — Part 1

Consider the case when $\Lambda = \{1, 2, 3, 4\}$, with the four switching models given by

$$A_1 = \begin{bmatrix} -4 & -2 \\ 9 & 5 \end{bmatrix}, \quad b_1 = \begin{bmatrix} -1 \\ 2 \end{bmatrix},$$

$$A_2 = \begin{bmatrix} 2 & -\frac{1}{3} \\ -3 & 0 \end{bmatrix}, \quad b_2 = \begin{bmatrix} 5 \\ -6 \end{bmatrix},$$

$$A_3 = \begin{bmatrix} -7 & -6 \\ 9.5 & 8 \end{bmatrix}, \quad b_3 = \begin{bmatrix} 6 \\ -7 \end{bmatrix},$$

$$A_4 = \begin{bmatrix} -12 & -11 \\ 13 & 12 \end{bmatrix}, \quad b_4 = \begin{bmatrix} -7 \\ 8 \end{bmatrix}.$$

We wish to investigate if this system is quadratically stabilizeable. Let z_i, $i = 1, 2, 3, 4$ denote the canonical models of $1, 2, 3, 4$ respectively, i.e. with z_i as the state vector model, i has the Brunovsky canonical form.

Let

$$z_i = C_i x, \quad i = 1, 2, 3, 4.$$

Then using Lemma 5.3 we obtain C_i as follows:

$$C_1 = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}, \quad C_2 = \begin{bmatrix} -2 & -\frac{5}{3} \\ 1 & \frac{2}{3} \end{bmatrix},$$

$$C_3 = \begin{bmatrix} \frac{7}{6} & 1 \\ \frac{4}{3} & 1 \end{bmatrix}, \quad C_4 = \begin{bmatrix} \frac{8}{3} & \frac{7}{3} \\ -\frac{5}{3} & -\frac{4}{3} \end{bmatrix}.$$

The state transformation matrices between z_1 and z_{i+1}, given by $T_i = C_1 C_{i+1}^{-1}$, $i = 1, 2, 3$ are

$$T_1 = \begin{bmatrix} 1 & 4 \\ -1 & -1 \end{bmatrix}, \quad T_2 = \begin{bmatrix} -4 & 5 \\ 2 & -1 \end{bmatrix},$$

$$T_3 = \begin{bmatrix} -3 & -6 \\ 1 & 1 \end{bmatrix}.$$

In the case of T_1: $t_3 t_4 = 1$, $1 \in S_p$, and $p_1 = \frac{t_1}{t_3} = -1$, $q_1 = \frac{t_2}{t_4} = -4$. Hence,

$$I_1 = (-\infty, 1) \cup (4, \infty).$$

Likewise,

$$I_2 = (2, 5)$$

and

$$I_3 = (-\infty, 3) \cup (6, \infty).$$

Hence,

$$I_3 \cap I_2 \cap I_3 = \emptyset.$$

Now, suppose that we use the state feedback control laws $u = K_i x$ with

$$\begin{aligned} K_1 &= \begin{bmatrix} -11.2143 & -8.2143 \end{bmatrix}, \\ K_2 &= \begin{bmatrix} -36 & -23.3333 \end{bmatrix}, \\ K_3 &= \begin{bmatrix} -18 & -13.5 \end{bmatrix}, \\ K_4 &= \begin{bmatrix} 58 & 46 \end{bmatrix}. \end{aligned}$$

Also, construct the quadratic Lyapunov function $x^T M x$ using

$$M = \begin{bmatrix} 31.75 & 26.25 \\ 26.25 & 21.75 \end{bmatrix} > 0.$$

Then, it is easily verified that

$$M(A_i + b_i K_i) + (A_i + b_i K_i)^T M < 0,$$

i.e. this system is quadratically stabilizeable even though the condition of Theorem 5.7 is not satisfied.

5.1.3 Example — Part 2

Here we consider again the above example, but using our newly obtained algorithm. In this algorithm, Steps 1–3 have been completed in Part 1. To complete Step 4, note that since $2 \in S_n$, $a_2 = -2$ and $b_2 = -5$, and the corresponding inequality is

$$2 = \min\{-a-2, -b_2\} < x < \max\{-a_2, -b_2\} = 5.$$

Now, consider $i = 1 \in S_p$ and $j = 2 \in S_n$. Then, since $a_1 = -1$ and $b_1 = -4$, the corresponding inequality is

$$(a_1 + b_1 - a_2 - b_2)x + a_1 b_1 - a_2 b_2 = 2x - 6 > 0.$$

Finally, when $i = 3 \in S_p$ and $j = 2 \in S_n$, $a_3 = -3$ and $b_3 = -6$ lead to the inequality

$$(a_3 + b_3 - a_2 - b_2)x + a_3 b_3 - a_2 b_2 = -2x + 8 > 0.$$

The complete set of inequalities (5.13) is

$$\begin{aligned} 2 < \;& x < 5, \\ 2x - 6 \;&>\; 0, \\ -2x + 8 \;&>\; 0, \\ x \;&>\; 0 \end{aligned}$$

with solution $3 < x < 4$, and we know that any solution x can be used to construct a common Lyapunov function. Choosing, for example, $x_0 = 3.5$, (5.11) becomes

$$\begin{aligned} e + 12.5 + (-1-4)\times 3.5 + 4 &> 0, \\ e + 12.5 + (-3-6)\times 3.5 + 18 &> 0, \\ e + 12.5 + (-2-5)\times 3.5 + 10 &> 0, \\ e &> 0 \end{aligned}$$

with solution $1 < e < 2$. Any solution here can be used to construct a common quadratic Lyapunov function and here, we set $e = 1.5$.

Given this value of e, set $m_2 = x_0$, $m_3 = x_0^2 + e$, to obtain

$$M_1 = \begin{bmatrix} 1 & 3.5 \\ 3.5 & 14 \end{bmatrix}.$$

To implement Steps 6 and 7, convert (A_1, b_1) to z_1 coordinates to yield

$$\widetilde{A}_1 = \begin{bmatrix} 0 & 1 \\ 2 & 1 \end{bmatrix}, \quad \widetilde{b}_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

Also, using (5.15), set, for example, $\alpha = -1 < 0$, and hence,

$$\widetilde{k}_1 = \begin{bmatrix} -3 & -\frac{37}{7} \end{bmatrix},$$

or, in the original coordinate system x,

$$K_1 = \widetilde{k}_1 C_1 = \begin{bmatrix} -11.2857 & -8.2857 \end{bmatrix}.$$

Repeating this last calculation now gives

$$\begin{aligned} K_2 &= \begin{bmatrix} -18 & -11.34 \end{bmatrix}, \\ K_3 &= \begin{bmatrix} -28 & -21 \end{bmatrix}, \\ K_4 &= \begin{bmatrix} 28 & 22 \end{bmatrix}. \end{aligned}$$

To verify this result, first compute

$$M_0 = C_1^T M_1 C_1 = \begin{bmatrix} 32 & 26.5 \\ 26.5 & 22 \end{bmatrix}$$

and then, we find that

$$M_0(A_i + b_i K_i) + (A_i + b_i K_i)^T M_0 < 0, \; i = 1, 2, 3, 4.$$

5.2 Stabilizeability of discrete-time switched systems

In this section, ideas developed in Section 5.1 for continuous-time systems are extended to the discrete-time case. We follow a similar approach to the one used for the continuous-time case, making adjustments where necessary.

We begin by defining a discrete-time switching system as

$$x(k+1) = A_{\sigma(x,t)}x(k) + B_{\sigma(x,t)}u_{\sigma(x,t)}(k) \tag{5.16}$$

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$ and $\sigma(x,t) : \mathbb{R}^n \times [0,\infty) \to \Lambda$ is an arbitrary mapping unless where stated. We assume $\Lambda = \{1, 2, \ldots, N\}$.

Similar to Section 5.1, the problem is to ensure quadratic stabilization of (5.16), i.e. we require the existence of a state feedback control law and a quadratic Lyapunov function $x^T Mx$, $M > 0$ such that the feedback switching models share $x^T Mx$ as a common quadratic Lyapunov function. Definition 5.1 is applicable to (5.16) and is stated again here for brevity.

Definition 5.12. System (5.16) is said to be quadratically stabilizeable with observeability $\sigma(x,t)$ if there exists a common Lyapunov function $x^T Mx$ with $M > 0$, and a set of state feedback control laws $u_i = A_i + B_iK_i$ such that $A_i + B_iK_i$, $f = 1, 2, \ldots, N$ share a common quadratic Lyapunov function $x^T Mx$.

Again, we require that each model is stabilizeable and again, this assumption can be replaced by controllability. We now prove Lemma 5.2, stated again here as Lemma 5.13, for the discrete-time case.

Lemma 5.13. *If a single-input system of the form considered here, written (A, b), is stabilizeable but not controllable then for any $M > 0$, there exists a suitable state feedback law such that $x^T Mx$ is a quadratic Lyapunov function of the closed loop system.*

Proof. We assume $b = \begin{bmatrix} 0 & 1 \end{bmatrix}^T$, which incurs no loss of generality. The system then becomes

$$x(k+1) = \begin{bmatrix} a_{11} & 0 \\ a_{21} & a_{22} \end{bmatrix} x(k) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(k).$$

Also, since stabilizeability holds, $|a_{11}| < 1$ and, without loss of generality, we assume $a_{11} = -\frac{1}{2}$. The closed loop matrix $\widetilde{A} = A + bK$ is

$$\widetilde{A} = \begin{bmatrix} -\frac{1}{2} & 0 \\ \alpha & \beta \end{bmatrix}$$

with α and β chosen arbitrarily.

Assume now that $M = \begin{bmatrix} 1 & m_2 \\ m_2 & m_3 \end{bmatrix}$. Then, it is required to prove that we can select α and β such that

$$\widetilde{A}^T M \widetilde{A} - M < 0$$

where

$$Q := \widetilde{A}^T M \widetilde{A} - M = \begin{bmatrix} \alpha^2 m_3 - \alpha m_2 - \frac{3}{4} & \alpha\beta m_3 - \left(\frac{1}{2}\beta + 1\right) m_2 \\ \alpha\beta m_3 - \left(\frac{1}{2}\beta + 1\right) m_2 & \left(\beta^2 - 1\right) m_3 \end{bmatrix}.$$

Hence, we need to choose α and β such that $Q(1,1) < 0$ and $\det(Q) > 0$. Since $m_3 > 0$ choosing

$$\alpha^2 m_3 - \alpha m_2 > \frac{3}{4} \quad \text{and} \quad \beta < 1$$

ensures that this condition holds and the proof is complete. □

Lemma 5.14. *Given*

$$M = \begin{bmatrix} m_1 & m_2 \\ m_2 & m_3 \end{bmatrix} > 0,$$

there exists a feedback control law $u = Kx = \begin{bmatrix} k_1 & k_2 \end{bmatrix} x$ *such that*

$$\widetilde{A} = A + bK = \begin{bmatrix} 0 & 1 \\ \alpha & \beta \end{bmatrix}$$

(i.e. the resulting closed loop system) has M *defining its quadratic Lyapunov function if* $m_2 < 0$ *and* $m_3 > 1$.

Proof. No loss of generality arises from assuming $m_1 = 1$. Then, for $M > 0$ to have the required property, we require that $\widetilde{A}^T M \widetilde{A} - M < 0$, or equivalently,

$$Q := \begin{bmatrix} \alpha^2 m_3 - 1 & \alpha\beta m_3 + (\alpha - 1) m_2 \\ \alpha\beta m_3 + (\alpha - 1) m_2 & \left(\beta^2 - 1\right) m_3 + 2\beta m_2 + 1 \end{bmatrix} < 0. \tag{5.17}$$

β is the trace of $\widetilde{A}$ and hence, $|\beta| < 1$ is a necessary condition for closed loop stability. Also, we note that $\alpha^2 m_3 - 1 < 0$ and therefore, $0 < m_3 < \frac{1}{\alpha^2}$ is necessary.

Equation (5.17) can be ensured to hold by first making

$$\alpha\beta m_3 = (1 + \alpha) m_2. \tag{5.18}$$

Then, since we require $\det(Q) > 0$, i.e.

$$-\frac{\left(\alpha^2 m_3 - 1\right)\left(\alpha^2 \left(m_3^2 - m_3\right) + \left(3\alpha^2 + 4\alpha\right) m_2^2 - m_2\right)}{\alpha^2 m_3} - 4\alpha^2 m_2^2 > 0. \tag{5.19}$$

Clearly now, if $m_2 < 0$ and $m_3 > 1$, there exist some α for which (5.19) holds. Choosing $\beta = \frac{(1+\alpha)m_2}{\alpha m_3}$ ensures $\det(Q) > 0$. □

5.2.1 Example

Consider the case when $\Lambda = \{1, 2, 3, 4\}$, with the four switching models in the Brunovsky form, given by

$$A_1 = \begin{bmatrix} 0 & 1 \\ -1.1052 & 2.1262 \end{bmatrix}, \quad b_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix},$$

$$A_2 = \begin{bmatrix} 0 & 1 \\ -1.2214 & 2.2325 \end{bmatrix}, \quad b_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix},$$

$$A_3 = \begin{bmatrix} 0 & 1 \\ -1.1052 & 2.0947 \end{bmatrix}, \quad b_3 = \begin{bmatrix} 0 \\ 1 \end{bmatrix},$$

$$A_4 = \begin{bmatrix} 0 & 1 \\ -1.0000 & 2.0100 \end{bmatrix}, \quad b_4 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

Take for example,

$$M = \begin{bmatrix} 1 & -0.3 \\ -0.3 & 1.2 \end{bmatrix} > 0.$$

From (5.19), it is found that

$$-0.6240 < \alpha < -0.2619$$

and β is found accordingly from (5.18). Hence, by taking

$$K_1 = \begin{bmatrix} 0.5052 & -1.9595 \end{bmatrix},$$

$$K_2 = \begin{bmatrix} 0.9583 & -1.5325 \end{bmatrix},$$

$$K_3 = \begin{bmatrix} 0.7206 & -1.6947 \end{bmatrix},$$

$$K_4 = \begin{bmatrix} 0.5000 & -1.7600 \end{bmatrix},$$

it can easily be verified that the Lyapunov function

$$(A_i + b_i K_i)^T M (A_i + b_i K_i) - M < 0$$

for all $i \in \{1, 2, 3, 4\}$.

5.3 Conclusion

In this Chapter, we addressed the problem of finding stabilizing controllers for single input single output switching systems comprising of subsystems that are second order linear time-invariant. It turns out that the problem lies on finding a common Lyapunov function for all subsystems.

For switching systems modelled in continuous-time, we have developed a method by which it could be determined

1. if there can exist a common Lyapunov function for all controlled subsystems $A_i + b_iK_i$ for all i,
2. for the quadratic Lyapunov function $V = x^TMx$, what elements are permissible for M such that $V > 0$ and $\frac{d}{dt}V < 0$, and
3. given a matrix M with permissible elements such as in Point 2, what values of k_i for each i are allowed.

For the case when the subsystems are modelled in discrete-time, the problem becomes more involved. Given a switching system where the subsystems are specified in the Brunovsky form, sufficient conditions for the matrix M such that M specifies the Lyapunov function for the controlled subsystem have been found. Consequently, values of k_i such that all controlled subsystems $A_i + b_iK_i$ have a common Lyapunov function can easily be found.

For the discrete-time case, the following problems still need to be resolved:

1. The necessary conditions for M to specify the Lyapunov function.
2. The sufficient and necessary conditions such that M can be transformed into other canonical coordinate systems.

Due to time constraints, we were unable to pursue these and we leave them as topics for further research.

Chapter 6

Conclusions and Future Research

In this chapter, we summarize the main conclusions that have come out of this research work. The main focus has been to investigate the problem of developing stabilizing controllers for the class of hybrid systems where the system dynamics show some form of switching behaviour. Our study concentrated on two main themes listed below, along with the relevant findings in brief.

1. **Control of hybrid systems through Model Reference Adaptive Control (MRAC).**

   The use of MRAC to control hybrid systems is potentially advantageous in that the designer is able to determine the system response by selection of an appropriate reference model. Since safety verification is a cause for concern in hybrid systems control, the system response can be guided away from unsafe regions by proper selection of the reference model. In view of this, we investigated the performance of hybrid systems under MRAC control for both, the single input single output (SISO) and the multiple input multiple output (MIMO) cases.

   For the SISO case, we studied the use of direct MRAC on hybrid systems, in particular, the performance of free running adaptive control against resettable adaptive control. Our findings reveal that, while direct MRAC was capable of producing the required response, there is a need for the control parameters to be reset to appropriate values at the onset of a switch in plant dynamics. Failure to do so could cause a mismatch between the control parameters and the new plant dynamics, which could lead to an unstable closed loop system. To perform the resetting action, the Multiple Model Adaptive Control (MMAC) structure was found to be particularly useful. Uncertainty of state parameters in regions close to subsystem polyhedral boundaries could result in the wrong control parameters being selected and could potentially lead to disastrous consequences. Therefore, robustness of the system at the polyhedral boundaries is an open problem for further research.

The performance of state feedback adaptive control against output feedback was also studied. It was found that there are instances when the control parameters of an output feedback system fails to be tuned to produce the required response. This is due to the inability of output feedback to access and tune some states. If such states are unstable, then the system is not stabilizeable. Obviously state feedback is desirable, and in many hybrid systems, knowledge of the states is essential in determining the continuous mode of the system. To achieve this in practise requires the use of state observers. Therefore, control of hybrid systems in the presence of observers is a subject to be studied in future.

Given that most real-life dynamical systems are multivariable in nature, it would be desirable to implement the MRAC scheme to this type of systems. Unlike in the SISO case, coupling is an inherent problem here. In order to implement MRAC to MIMO systems, both the plant and reference model need to be decoupled and both have to have identical interactor matrices. The problem is further compounded in hybrid systems since a switch in plant dynamics also generally means a change in the interactor matrix. It would be unrealistic to have systems where the reference model and all plant subsystems have identical interactor matrices. Consequently, a more practical approach would be to switch the reference model as the plant dynamics switch such that at any instance of time, the plant and reference model would always have identical interactor matrices. Our research indicate that this way, multivariable hybrid systems can be controlled through MMAC provided that the plant and reference models are adequately decoupled.

While decoupling may be achieved through state feedback, it was found that this method is unreliable as it is possible that decoupling is achieved through cancellation of unstable poles by zeros at the same location. As an alternative, diagonalization of the interactor matrix could be performed by insertion of an appropriate pre-compensator (Singh and Narendra, 1984). Future research should study the implementation of such pre-compensators for multivariable hybrid systems.

2. **Determination of the stabilizeability of controllable hybrid systems.**

   Our work in this area concentrated on SISO systems with two continuous states. The study encompassed systems modelled in continuous-time, as well as those modelled in discrete-time.

   For the continuous-time case, a method for the determination of the existence of a common Lyapunov function that is shared by all subsystems given appropriate control parameters, have been developed. Having identified the existence of such a function, and by appropriate transformations to and from the Brunovsky form, the required control parameters can be identified. Future research should seek to extend this method to more general switching systems with different dimensions.

   For the discrete-time case, given a system described in the Brunovsky form, sufficient conditions for the positive definite matrix M such that M specifies the

Lyapunov function of the system in closed loop with appropriate control laws have been established. Then, for each subsystem, the required control parameters can always be found such that all closed loop subsystems have a common Lyapunov function. Future work on this area should seek to develop a method for finding stabilizing controllers for systems specified in other canonical forms, similar to the one that we have developed for continuous-time systems.

Bibliography

B.D.O. Anderson, T.S. Brinsmead, F. de Bruyne, J. Hespanha, D. Liberzon, and A.S. Morse. Multiple model adaptive control. part 1: Finite controller coverings. *International Journal of Robust and Nonlinear Control*, 10:909–929, 2000.

P.J. Antsaklis and X.D. Koutsoukos. Hybrid systems: Review and recent progress. In T. Samad and G. Balas, editors, *Software-Enabled Control: Information Technologies for Dynamical Systems*. IEEE Press, 2002.

K.J. Åstrom and B. Wittenmark. *Adaptive Control*. Addison-Wesley, 1989.

T. Autenrieth and E. Rogers. Performance enhancements for a class of multiple model adaptive control schemes. *International Journal of Adaptive Control and Signal Processing*, 13:105–127, 1999.

A. Balluchi, M.D. Di Benedetto, C. Pinello, C. Rossi, and A. Sangiovanni-Vincentelli. Hybrid control in automotive applications: the cut-off control. *Automatica*, 35:519–535, 1999.

A. Bemporad, F. Borelli, and M. Morari. On the optimal control law for linear discrete time hybrid systems. In C.J. Tomlin and M.R. Greenstreet, editors, *Hybrid Systems: Computation and Control*, volume 2289 of *Lecture Notes in Computer Science*, pages 105–119. Springer-Verlag, 2002.

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

A. Bemporad and M. Morari. Control of systems integrating logic, dynamics and constraints. *Automatica*, 35:407–427, 1999.

A. Benveniste. Compositional and uniform modelling of hybrid systems. *IEEE Transactions on Automatic Control*, 43:579–583, 1998.

A. Benveniste and P. le Guernic. Hybrid dynamical systems and the SIGNAL language. *IEEE Transactions on Automatic Control*, 35:535–546, 1990.

R.K. Boel, B. de Schutter, G. Nijsse, J.M. Schumacher, and J.H. van Schuppen. Approaches to modelling, analysis and control of hybrid systems. *Journal A*, 40(4):16–27, 1999.

F. Borelli, M. Baotic, A. Bemporad, and M. Morari. An efficient algorithm for computing the state feedback optimal control law for discrete time hybrid systems. In *Proceedings of the American Control Conference*, pages 4717–4722, Denver, USA, 2003.

M.S. Branicky. Stability of switched and hybrid systems. In *Proceedings of the IEEE Conference on Decision and Control*, 1994.

M.S. Branicky. Multiple Lyapunov functions and other analysis tools for switched hybrid systems. *IEEE Transactions on Automatic Control*, 43(4):475–482, 1998.

R.R. Costa, L. Hsu, A.K. Imai, and P. Kokotović. Lyapunov-based adaptive control of MIMO systems. *Automatica*, 39:1251–1257, 2003.

J. Daafouz, G. Millerioux, and C. Iung. A poly-quadratic stability based approach for linear switched systems. *International Journal of Control*, 75(16/17):1302–1310, 2002.

G. Davrazos and N.T. Koussoulas. A review of stability results for switched and hybrid systems. In *Proceedings of the 2003 Mediterranean Conference on Control and Automation*, 2003.

B. de Schutter and B. de Moor. The extended linear complementarity problem and the modeling and analysis of hybrid systems. In P. Antsaklis, W. Kohn, M. Lemmon, A. Nerode, and S. Sastry, editors, *Hybrid Systems V*, volume 1567 of *Lecture Notes in Computer Science*, pages 70–85. Springer-Verlag, 1999.

B. de Schutter and T.J.J. van den Boom. Model predictive control for max-min-plus-scaling systems. In *Proceedings of the 2001 American Control Conference*, pages 319–324, 2001.

B. de Schutter and T.J.J. van den Boom. MPC for continuous piecewise-affine systems. *Systems and Control Letters*, 52:179–192, 2004.

R.A. Decarlo, M.S. Branicky, S. Pettersson, and B. Lennartson. Perspectives and results on the stability and stabilizability of hybrid systems. *Proceedings of the IEEE*, 88(7): 1069–1082, 2000.

H. Elliott and W.A. Wolovich. A parameter adaptive control structure for linear multivariable systems. *IEEE Transactions on Automatic Control*, AC-27(2):340–352, 1982.

G. Escobar, A.J. van der Schaft, and R. Ortega. A (hamiltonian) viewpoint in the modeling of switching power converters. *Automatica*, 35:445–452, 1999.

P.J. Falb and W.A. Wolovich. Decoupling in the design and synthesis of multivariable control systems. *IEEE Transactions on Automatic Control*, AC-12(6):651–659, 1967.

G. Ferrari-Trecate, F.A. Cuzzola, and M. Morari. Analysis of discrete-time pwa systems with logic states. In C.J. Tomlin and M.R. Greenstreet, editors, *Hybrid Systems: Computation and Control*, volume 2289 of *Lecture Notes in Computer Sciences*, pages 194–208. 2002.

P.W. Gibbens, C.A. Schwartz, and M. Fu. Achieving diagonal interactor matrix for multivariable linear systems with uncertain parameters. *Automatica*, 29:1547–1550, 1993.

E.G. Gilbert. The decoupling of multivariable systems by state feedback. *SIAM Journal of Control*, 7(1):50–63, 1969.

C. Gökçek. Stability analysis of periodically switched linear systems using floquet theory. *Mathematical Problems in Engineering*, 1:1–10, 2004.

R.L. Grossman and R.G. Larson. An algebraic approach to hybrid systems. *Journal of Theoretical Computer Science*, 138:101–112, 1995.

W.P.M.H. Heemels, B. de Schutter, and A. Bemporad. Equivalence of hybrid dynamical models. *Automatica*, 37(7):1085–1092, 2001.

J. Hespanha, D. Liberzon, A.S. Morse, B.D.O. Anderson, T.S. Brinsmead, and F. de Bruyne. Multiple model adaptive control. part 2: switching. *International Journal of Robust and Nonlinear Control*, 11:479–496, 2001.

J.P. Hespanha, D. Liberzon, and A.S. Morse. Overcoming the limitations of adaptive control by means of logic-based switching. *Systems and Control Letters*, 49:49–65, 2003.

J.P. Hespanha and A.S. Morse. Switching between stabilizing controllers. *Automatica*, 38:1905–1917, 2002.

A.K. Imai, R.R. Costa, L. Hsu, G. Tao, and P. Kokotović. Multivariable MRAC using high frequency gain matrix factorization. In *Proceedings of the 40th IEEE Conference on Decision and Control*, pages 1193–1198, Orlando, Florida, USA, 2001. IEEE.

A.K. Imai, R.R. Costa, L. Hsu, G. Tao, and P.V. Kokotović. Multivariable adaptive control using high frequency gain matrix factorization. *IEEE Transactions on Automatic Control*, 49(7):1152–1157, 2004.

P.A. Ioannou and J. Sun. *Robust Adaptive Control*. Prentice Hall, New Jersey, 1996.

G. Labinaz, M.M. Bayoumi, and K. Rudie. A survey of modeling and control of hybrid systems. *Annual Reviews in Control*, 21:79–92, 1997.

I.D. Landau. From robust control to adaptive control. *Control Engineering Practice*, 7: 1113–1124, 1999.

D.J. Leith, R.N. Shorten, W.E. Leithead, O. Mason, and P. Curran. Issues in the design of switched linear control systems: A benchmark study. *International Journal of Adaptive Control and Signal Processing*, 17:103–118, 2003.

D. Liberzon, J.P. Hespanha, and A.S. Morse. Stability of switched systems: a Lie-algebraic condition. *Systems and Control Letters*, 37(3), 1999.

D. Liberzon and A.S. Morse. Basic problems in stability and design of switched systems. *IEEE Control Systems Magazine*, 19(5):59–70, 1999.

J. Lygeros, D.N. Godbole, and S. Sastry. Verified hybrid controllers for automated vehicles. *IEEE Transactions on Automatic Control*, 43(4):522–539, 1998.

I.M. Mitchell and C.J. Tomlin. Overapproximating reachable sets by Hamilton-Jacobi projections. *Journal of Scientific Computing*, 19(1-3), 2003.

F.J. Montoya and R.K. Boel. Introduction to discrete event systems. In A. Baños, F. Lamnabhi-Lagarrigue, and F.J. Montoya, editors, *Advances in the Control of Nonlinear Systems*, volume 264. 2001.

T. Moor, J.M. Davoren, and J. Raisch. Modular supervisory control of a class of hybrid systems in a behavioural framework. In *Proceedings of the European Control Conference ECC2001*, pages 870–875, Porto, Portugal, 2001.

T. Moor and J. Raisch. Estimating reachable states of hybrid systems via l-complete approximations. In V.B. Bajic, editor, *Proceedings of SSCC '98 - Advances in Systems, Signals, Control and Computers*, volume III, pages 30–34, Durban, South Africa, 1998. IAAMSAD and the South African Branch of the Academy of Nonlinear Sciences.

T. Moor and J. Raisch. Discrete control of switched linear systems. In *Proceedings of the European Control Conference*, 1999a.

T. Moor and J. Raisch. Supervisory control of hybrid systems within a behavioural framework. *Systems and Control Letters*, 38:157–166, 1999b.

T. Moor, J. Raisch, and S. O'Young. Discrete supervisory control of hybrid systems based on l-complete approximations. *Discrete Event Dynamic Systems: Theory and Applications*, 12:83–107, 2002.

K.S. Narendra. Adaptive control of discrete-time systems using multiple models. *IEEE Transactions on Automatic Control*, 45(9):1669–1686, 2000.

K.S. Narendra and A.M. Annaswamy. *Stable Adaptive Systems*. Prentice Hall, New Jersey, 1989.

K.S. Narendra and J. Balakrishnan. Improving transient response of adaptive control systems using multiple models and switching. *IEEE Transactions on Automatic Control*, 39(9):1861–1866, 1994.

K.S. Narendra and J. Balakrishnan. Adaptive control using multiple models. *IEEE Transactions on Automatic Control*, 42(2):171–187, 1997.

K.S. Narendra, O.A. Driollet, M. Feiler, and K. George. Adaptive control using multiple models, switching and tuning. *International Journal of Adaptive Control and Signal Processing*, 17:87–102, 2003.

K. Ogata. *System Dynamics.* Prentice Hall, New Jersey, 3rd edition, 1998.

T. Ooba and Y. Funahashi. Two conditions concerning common QLFs for linear systems. *IEEE Transactions on Automatic Control*, 42(5):719–721, 1997.

V. Petridis and Ath. Kehagias. A multi-model algorithm for parameter estimation of time-varying nonlinear systems. *Automatica*, 34(4):469–475, 1998.

B. Potocnik, G. Music, and B. Zupancic. A new technique for translating discrete hybrid automata into piecewise affine systems. *Mathematical and Computer Modelling of Dynamical Systems*, 10(1):41–57, 2004.

J. Raisch. Discrete abstractions of continuous systems - an input/output point of view. *Mathematical and Computer Modelling of Dynamical Systems*, 6(1):6–29, 2000.

J. Raisch and S.D. O'Young. A totally ordered set of discrete abstractions for a given hybrid or continuous system. In Antsaklis, editor, *Hybrid Systems IV*, Lecture Notes in Computer Sciences, pages 342–360. 1997.

J. Raisch and S.D. O'Young. Discrete approximation and supervisory control of continuous systems. *IEEE Transactions on Automatic Control*, 43(4):569–572, 1998.

A. Rantzer and M. Johansson. Piecewise linear quadratic optimal control. *IEEE Transactions on Automatic Control*, 45(4):629–637, 2000.

L. Rodrigues. Stability analysis of piecewise-affine systems using controlled invariant sets. *Systems and Control Letters*, 53:157–169, 2004.

L. Rodrigues and J.P. How. Observer-based control of piecewise affine systems. *International Journal of Control*, 76(5):459–477, 2003.

M.G. Safonov and T-C. Tsao. The unfalsified control concept and learning. *IEEE Transactions on Automatic Control*, 42(6):843–847, 1997.

S. Sastry and M. Bodson. *Adaptive Control: Stability, Convergence and Robustness.* Prentice Hall, New Jersey, 1989.

C.A. Schwartz and A. Yan. Comments on 'Achieving diagonal interactor matrix for multivariable linear systems with uncertain parameters'. *Automatica*, 31:161–164, 1995.

R.N. Shorten and K.S. Narendra. Necessary and sufficient conditions for the existence of a common quadratic Lyapunov function for M-stable linear second-order systems. In *Proceedings of the American Control Conference 2000*, pages 359–363, 2000.

R.P. Singh and K.S. Narendra. Prior information in the design of multivariable adaptive controllers. *IEEE Transactions on Automatic Control*, AC-29(12):1108–1111, 1984.

J.E. Slotine and W. Li. *Applied Nonlinear Control.* Prentice Hall, New Jersey, 1991.

E.D. Sontag. Nonlinear regulation: The piecewise linear approach. *IEEE Transactions on Automatic Control*, AC-26(2):346–358, 1981.

E.D. Sontag. Interconnected automata and linear systems: A theoretical framework in discrete-time. In R. Alur, T.A. Henzinger, and E.D. Sontag, editors, *Hybrid Systems III*, volume 1066 of *Lecture Notes in Computer Sciences*, pages 436–448. 1996.

M. Stefanovic. Stability and convergence in adaptive systems. In *Proceedings of the 2004 American Control Conference*, 2004.

G. Tao. *Adaptive Control Design and Analysis.* Wiley-Interscience, New Jersey, 2003.

C.J. Tomlin, I. Mitchell, A.M. Bayen, and M. Oishi. Computational techniques for the verification of hybrid systems. *Proceedings of the IEEE*, 91(7):986–1001, 2003.

A.J. van der Schaft and J.M. Schumacher. Complementarity modeling of hybrid systems. *IEEE Transactions on Automatic Control*, 43(4):483–490, 1998.

A.J. van der Schaft and J.M. Schumacher. Modelling and analysis of hybrid dynamical systems. In A. Baños, F. Lamnabhi-Lagarrigue, and F.J. Montoya, editors, *Advances in the Control of Nonlinear Systems*, volume 264 of *Lecture Notes in Control and Information Sciences*, pages 195–224. Springer-Verlag, 2001.

H.V. Vu and R.S. Esfandiari. *Dynamic Systems - Modeling and Analysis.* McGraw-Hill, 1998.

W.A. Wolovich and P.J. Falb. Invariants and canonical forms under dynamic compensation. *SIAM Journal of Control and Optimization*, 14(6):996–1008, 1976.