The University of Sheffield

# Adaptive Backstepping and Sliding Mode Control of Uncertain Nonlinear Systems

by

Miguel Rios-Bolívar

**A thesis submitted**
**in support of the application**
**for the degree of Doctor of Philosophy**
**at the University of Sheffield**

March 1997

Applied Mathematics Section, School of Mathematics and Statistics
The University of Sheffield

*Dedicated to the memory of my father, and to Hilia, Rosanna, Miguel Angel and Christian for their love and patience.*

# Acknowledgements

I would like to thank Prof. A. S. I. Zinober for advice and encouragement throughout this study, and Prof. H. Sira-Ramírez for contributing with many helpful discussions.

I would also like to thank CONICIT-Venezuela for providing financial support.

# Summary

## Adaptive Backstepping and Sliding Mode Control of Uncertain Nonlinear Systems

**by**

**Miguel Rios-Bolívar**

The development of adaptive control design techniques for nonlinear systems with parametric uncertainty has been intensively studied in recent years. The recently developed adaptive *backstepping* technique has provided a systematic solution to the problem of designing static adaptive controllers for uncertain nonlinear systems transformable into the triangular Parametric Strict Feedback and Parametric Pure Feedback forms.

The adaptive backstepping technique has been adopted in this thesis as the control design approach and a number of new algorithms have been developed for the design of dynamical controllers for the regulation and tracking of deterministic and adaptive control systems. The combination of adaptive backstepping and Sliding Mode Control has also been proposed to design robust adaptive strategies for uncertain systems with disturbances. The class of adaptive backstepping nonlinear systems has been broadened to *observable minimum phase* systems which are *not* necessarily transformable into triangular forms. The design of output feedback control, when only the output is measured, has also been studied for a class of uncertain systems transformable into the *adaptive generalized observer canonical form.*

Since the equations arising from these new algorithms are too complicated to be computed by hand, a symbolic algebraic toolbox has been developed. This toolbox implements the proposed algorithms for the design of static (dynamic) deterministic (adaptive) controllers, and automatically generates MATLAB code programs for computer simulation.

# Contents

# List of Figures

# Chapter 1

# Control Regulation of Uncertain Nonlinear Systems

## 1.1 Introduction

Mathematical models of practical systems usually contain inaccuracies which may arise from actual uncertainty (for instance, imperfect knowledge of parameter values or disturbances) or from the purposeful choice of a simplified representation of the system's dynamics. Modelling imprecisions due to inaccuracies in the terms actually included in the model, are called *structured* (or parametric) uncertainties. In contrast, *unstructured* uncertainties (or unmodelled dynamics) correspond to inaccuracies in the system order. Modelling imprecisions can produce adverse effects on the performance of linear and nonlinear control systems. Therefore, it is important that any control design scheme dealing with uncertainty provides enough robustness to guarantee a satisfactory performance under these conditions. Traditionally, two complementary approaches are used to deal with uncertainty: *adaptive control* and *robust control.*

Adaptive control is more convenient for plants containing parametric uncertainty when no information about the bounds of the unknown parameters is known *a priori*. An estimator (or identifier) provides on-line estimation of the unknown parameters. In its forty years of existence, adaptive control has become a well-established discipline with a wide spectrum of algorithms and many practical applications. Recently, a new family of adaptive control algorithms called *backstepping* has been developed. This new control scheme allows the systematic design of adaptive controllers for nonlinear systems containing "unmatched" parametric uncertainty. It will be the main subject to be studied in this thesis.

Robust control is usually employed for plants with unstructured uncertainty when some information (bounds) about the uncertainty is known. The typical robust controller is composed of a nominal part, obtained via feedback linearization, pole-placement, or any other design technique to stabilize the known part of the plant; and additional terms aim at dealing with model uncertainty.

A simple and popular robust approach to the deterministic control of uncertain systems is the *Sliding Mode Control* technique, which is based upon the special behaviour of Variable Structure Systems in the so-called *sliding regime.* The basic idea of sliding mode control is to drive system trajectories towards a stable manifold and ideally keep them on it using a discontinuous feedback control. The controlled plant in the sliding regime is totally invariant to matched disturbances and parameter variations with known bounds, thus maintaining stability and consistent performance. An equivalent approach used in the context of power electronics and, more particularly, power conversion, is the so-called Pulse-Width-Modulated control technique. It has been shown that Sliding Mode Control and Pulse-Width-Modulated control are equivalent control techniques with similar robustness properties [104].

In this introductory chapter, both adaptive schemes and robust (sliding mode control and pulse-width-modulated control) techniques are introduced, and additionally, some motivating examples are considered. In Section 1.4 the topics studied in this thesis are introduced.

## 1.2 Adaptive Control of Uncertain Systems

It was in the early 1950s, when classical scalar frequency-domain methods were well established and broadly used, that the field of adaptive control was born. At that time there was considerable interest in the design of autopilots for high-performance aircraft, which operated at a wide range of conditions such as different speeds and altitudes. The existing feedback theory was appropriate to design an efficient controller at any one operating condition, but could not cope with problems under changing operating conditions. Adaptive control was proposed as a way of automatically adjusting the controller parameters in the face of changing aircraft dynamics. Since then, a lot of research has been carried out on the design of adaptive controllers to stabilize plants with internal and/or external uncertainties. Generally speaking, the basic objective of adaptive control is to achieve consistent performance of a system in the presence of uncertainties or unknown variations in plant parameters. Since such parameter uncertainties or changing conditions occur in many practical problems, adaptive control is applicable in many industrial contexts.

Adaptive control is now a mature discipline. The intense research performed in this area has made available a wide variety of concepts and algorithms, which have attracted the attention of many practitioners. The main reason for the popularity of adaptive control is associated with its clearly defined objective: to control plants with unknown parameters. A very detailed survey of the progress of adaptive control since its birth has been presented by Narendra [79].

Depending on the way in which the update law is implemented, adaptive schemes are classified as *direct* or *indirect*. When the estimate corresponds to the parameters of the controlled plant, and the controller design is then based on the identified parameters, the scheme is called *indirect*. Alternatively, *direct* schemes estimate the parameters of the controller. Additionally, adaptive controllers can be classified as *Lyapunov-based* or *estimation-based*, according to the type of parameter update law and the corresponding proof of stability and convergence.

Because of the wide variety of parameter update laws, such as gradient and least-squares algorithms, estimation-based designs are broadly applicable in adaptive linear control ([35],[97],[80]). This acceptance is also supported by the applicability of the "separation principle" to linear systems, which allows one to treat the identifier as a separate module and guarantee its properties independent of the controller module. However, the stability analysis of estimation-based schemes is usually intricate and is conclusive only in the case of normalized update laws. Lyapunov-based design provides elegant stability proofs, although, until recently its applicability was restricted to linear plants with relative degree one or two.

One traditional approach in adaptive control consists of ignoring the uncertainty and treating the estimated parameters as if they were the true parameters for designing the controller via a deterministic scheme, such as pole-placement or optimal control. This approach is commonly called *certainty equivalence* and involves the separation of the estimation and control problems. However, it is not obvious that certainty equivalence controllers will achieve stabilization of the closed-loop system. Additionally, existing adaptive techniques generally require a *linear parameterization* of the plant dynamics, i.e. the parametric uncertainty needs be expressed linearly in terms of a set of unknown parameters.

In the 1960s when the first attempts were made to describe adaptive systems using state equations, the fact that adaptive systems are generally nonlinear came to the fore. This can be illustrated with the following example.

**Example 1.1** Consider the system

$$\begin{aligned} \dot{x} &= A(\sigma,\theta)x + B(\sigma,\theta)u \\ y &= C(\sigma,\theta)x \end{aligned} \tag{1.1}$$

with $u(t) \in \Re$, $x(t) \in \Re^n$, $y(t) \in \Re^m$, $\sigma \in \Re^{m_1}$, and $\theta(t) \in \Re^{m_2}$, where $u, x, y$ are respectively the input, state and output, $\sigma$ is an unknown constant vector, and $\theta(t)$ a time-varying control parameter vector which is adjusted using the measured signals of the system by

$$\dot{\theta} = g(y, \theta, t) \tag{1.2}$$

Note that the components $\theta_i(t)$ of $\theta(t)$, $i = 1, 2, \ldots, m_2$ are no longer parameters, but are state variables of the system (1.1)-(1.2). Hence, any adaptive system is merely a nonlinear feedback system involving estimation and control ([79]). Indeed, if a linear plant contains unknown parameters without any information regarding their bounds, it cannot be stabilized by a linear controller. This is true even for the following simple example.

**Example 1.2** Consider the scalar linear plant

$$\dot{x} = u + \theta x \tag{1.3}$$

where $u$ is the control and $\theta$ is an unknown constant. Its stabilization can be achieved by the following adaptive controller

$$u = -\hat{\theta}x - cx \tag{1.4}$$

$$\dot{\hat{\theta}} = kx^2 \tag{1.5}$$

where $c$ and $k$ are positive design parameters. Its stability properties can be checked by examining the derivative of the Lyapunov function

$$V(x, \hat{\theta}) = \frac{1}{2}x^2 + \frac{1}{2k}(\theta - \hat{\theta})^2 \tag{1.6}$$

which turns out to be nonpositive

$$\dot{V} = -cx^2 \leqslant 0. \tag{1.7}$$

Consequently, $V(x, \theta)$ evaluated along the solutions of the closed-loop system

$$\dot{x} = -(c + \hat{\theta})x + \theta x \tag{1.8}$$

$$\dot{\hat{\theta}} = kx^2 \tag{1.9}$$

is a nonincreasing function of time. This proves that $x(t)$ and $\hat{\theta}$ remain bounded for all $t \geqslant 0$, by Theorem A.1 (in Appendix A). It can be proved that $\lim_{t\to\infty} x(t) = 0$ by using the Corollary A.1.

### 1.2.1 Matching Conditions

In the last few years the problem of designing adaptive nonlinear controllers for plants containing both unknown parameters and known nonlinearities has been of increasing interest in the control community. Linear parameterization and full-state feedback are common assumptions to deal with these plants. Most of the adaptive nonlinear control results have been obtained for linearly parameterized systems of the form

$$\dot{\zeta} = f_0(\zeta) + \sum_{i=1}^{p} \theta_i f_i(\zeta) + \left[ g_0(\zeta) + \sum_{i=1}^{p} \theta_i g_i(\zeta) \right] u \tag{1.10}$$

where $\zeta \in \Re^n$ is the state, $u \in \Re$ is the scalar input, $\theta = [\theta_1, \ldots, \theta_p]^T$ is the vector of unknown parameters, and $f_i, g_i$, $0 \leqslant i \leqslant p$, are smooth vector fields in a neighbourhood of the origin $\zeta = 0$ with $f_i(0) = 0$, $0 \leqslant i \leqslant p$, $g_0(0) \neq 0$.

An important concept associated with the control of plants with uncertainties is that of *matching* conditions. A system satisfies the matching conditions strictly if both uncertainty and control input appear in the same dynamic equation. The controlled plant (1.3) obviously satisfies the matching conditions. The design of an adaptive controller for a matched system is quite straightforward as illustrated by the following example.

**Example 1.3**

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= u + \theta\varphi(x_1) \end{aligned} \tag{1.11}$$

where $u$ is the control, $\theta$ an unknown constant and $\varphi$ a known nonlinear function. By choosing the control

$$u = -k_1 x_1 - k_2 x_2 - \hat{\theta}\varphi(x_1) \tag{1.12}$$

where $\hat{\theta}$ is an estimate of the unknown parameter, and $k_1$ and $k_2$ are positive design parameters, the system is transformed into

$$\dot{x} = Ax + W(x)(\theta - \hat{\theta}) \tag{1.13}$$

with

$$A = \begin{bmatrix} 0 & 1 \\ -k_1 & -k_2 \end{bmatrix}, \qquad W = \begin{bmatrix} 0 \\ \varphi \end{bmatrix} \tag{1.14}$$

This form suggests that a parameter update law can be designed using a result of adaptive linear control [80]. This update law is

$$\dot{\hat{\theta}} = W^T(x)Px \tag{1.15}$$

where $P = P^T > 0$ is chosen to satisfy $PA + A^T P = -I$. The stability of the equilibrium $x = 0$, $\hat{\theta} = \theta$ of the closed-loop system (1.13)-(1.15) is established by checking the derivative of the Lyapunov function

$$V = x^T P x + (\theta - \hat{\theta})^2 \tag{1.16}$$

which is $\dot{V} = -\|x\|^2 \leqslant 0$ along the solutions of the closed-loop system.

A concept associated with the matching conditions is that of *uncertainty level* [57], which indicates the number of integrators separating the uncertainty from the control input. In other words, suppose the uncertain parameters are in one equation, the uncertainty level is the number of differentiations with respect to time required to reach the control input explicitly. Therefore, systems satisfying the matching conditions strictly have uncertainty level zero. It is said that systems satisfy the *Extended Matching Conditions* (EMC) when the uncertainty level is one, i.e., the uncertainty is separated from the control input by one integrator only. Kanellakopoulos, Kokotović and Marino [46] have developed control schemes for the stabilization of this class of systems. These algorithms are called *uncertainty-constrained schemes* because they require that the system satisfies the EMC as structural conditions. The EMC were reformulated in [13] as "strong linearizability" conditions. For full-state feedback linearizable systems, the EMC is necessary and sufficient for the existence of a parameter-independent diffeomorphism $x = \phi(\zeta)$ which transforms the system (1.10) into

$$\begin{aligned} \dot{x}_i &= x_{i+1} \qquad\qquad i = 1, \ldots, n-2 \\ \dot{x}_{n-1} &= x_n + \sum_{i=1}^{l} \theta_i(x)\gamma_i(x) \\ \dot{x}_n &= \alpha_0(x) + \sum_{i=1}^{l} \theta_i(x)\alpha_i(x) + \left[\beta_0(x) + \sum_{i=1}^{l} \theta_i \beta_i(x)\right] u. \end{aligned} \tag{1.17}$$

In order to illustrate the properties of the EMC-based schemes of [46] consider the following example.

**Example 1.4** Consider the second order system

$$\begin{aligned} \dot{x}_1 &= x_2 + \theta\gamma(x_1) \\ \dot{x}_2 &= u \end{aligned} \tag{1.18}$$

with $\gamma(x_1)$ a known smooth nonlinear function of $x_1$. Note that this system is already in the form (1.17). The design procedure of [46] employs an estimate $\hat{\theta}$ of the unknown parameter $\theta$ and replaces $x_2$ with the new state

$$\hat{x}_2 = x_2 + \hat{\theta}\gamma(x_1) \tag{1.19}$$

The system (1.18) is then rewritten as

$$\begin{aligned} \dot{x}_1 &= \hat{x}_2 + (\theta - \hat{\theta})\gamma(x_1) \\ \dot{\hat{x}}_2 &= u + \dot{\hat{\theta}}\gamma(x_1) + \hat{\theta}\dot{\gamma}\hat{x}_2 + (\theta - \hat{\theta})\hat{\theta}\dot{\gamma}\gamma(x_1). \end{aligned} \tag{1.20}$$

The control can now be designed as a function of $\hat{\theta}$ and $\dot{\hat{\theta}}$, since the derivative $\dot{\hat{\theta}}$ will be explicitly defined from the update law. By using the control

$$u = -k_1 x_1 - k_2 \hat{x}_2 - \dot{\hat{\theta}}\gamma(x_1) - \hat{\theta}\dot{\gamma}\hat{x}_2 \tag{1.21}$$

the system (1.18) is transformed into

$$\begin{aligned} \dot{x}_1 &= \hat{x}_2 + (\theta - \hat{\theta})\gamma(x_1) \\ \dot{\hat{x}}_2 &= -k_1 x_1 - k_2 \hat{x}_2 + (\theta - \hat{\theta})\hat{\theta}\dot{\gamma}\gamma(x_1) \end{aligned} \tag{1.22}$$

which is linear in the parameter error $\theta - \hat{\theta}$. The gains $k_1$ and $k_2$ are chosen to place the eigenvalues of the system matrix

$$A = \begin{bmatrix} 0 & 1 \\ -k_1 & -k_2 \end{bmatrix} \tag{1.23}$$

at some desired stable locations. The resulting form is

$$\dot{\hat{x}} = A\hat{x} + W(\hat{x}, \hat{\theta})(\theta - \hat{\theta}) \tag{1.24}$$

with $\hat{x} := [x_1 \; \hat{x}_2]^T$ and

$$W(\hat{x}, \hat{\theta}) = \begin{bmatrix} \gamma(x_1) \\ \hat{\theta}\dot{\gamma}\gamma(x_1) \end{bmatrix} \tag{1.25}$$

This form is the same as (1.13), which suggests the use of the update law

$$\dot{\hat{\theta}} = W(\hat{x}, \hat{\theta})^T P \hat{x} \tag{1.26}$$

which is the same update law used in Example 1.3. The stability of the equilibrium point $\hat{x} = 0, \hat{\theta} = \theta$ of the closed-loop system is established in the same way as for Example 1.3, by using the Lyapunov function (1.16). Since the feedback linearization of (1.18) can be achieved for all $x$ and $\theta$, the stability result is global. Moreover, from the LaSalle invariance theorem (Theorem A.2), the largest invariant set of (1.24)-(1.26) contained in the set where $\dot{V} = 0$, is

$$M = \{(\hat{x}, \hat{\theta}) : \hat{x} = 0, (\theta - \hat{\theta})W(0, \hat{\theta}) = 0\} \tag{1.27}$$

Then, if $W(0, \hat{\theta}) \neq 0$, $(0, \theta)$ can be shown to be an *asymptotically stable* equilibrium point of (1.24)-(1.26).

Since the nonlinear function $\gamma(x_1)$ can be any smooth function, this example demonstrates that EMC-based schemes guarantee stability properties *independently* of the type of nonlinearities. Many practical systems satisfy the EMC and their stabilization can be achieved by this scheme [46]. However, implicit expressions for the control and the update law (the derivative $\dot{\hat{\theta}}$ is defined in terms of the control $u$, and $u$ is in turn defined in terms of $\dot{\hat{\theta}}$) may result from the EMC-based schemes (for an example showing this problem, see [57]). This difficulty can be avoided by considering the overparameterization approach proposed by Pomet and Praly [82]. Using their scheme, various update laws are designed for the same set of unknown parameters. A limitation of this approach is the loss of exponential stability.

The first approach to cope with the stabilization problem of systems that do not satisfy the EMC was proposed by Taylor *et al* [117]. It uses a high-gain feedback control to induce a two-time separation property, so that the slow subsystem satisfies the EMC and the fast stable dynamics is treated as unmodelled dynamics. EMC-based schemes are not applicable to systems that do not satisfy the EMC because second derivatives of the parameter estimate $\ddot{\hat{\theta}}$ would invariably appear in the control, since the uncertainty level would be at least two. The adaptive *backstepping* approach proposed by Kanellakopoulos, Kokotović and Morse [48] removed this structural obstacle and allowed Lyapunov-based designs to be applied to wide classes of uncertain nonlinear systems.

As an alternative to EMC-based schemes, *nonlinearity-constrained schemes* can be used to stabilize uncertain nonlinear systems. These schemes are applicable to systems that do not satisfy the EMC, but they impose restrictions on the type of nonlinearity. For example, the schemes of Nam and Arapostathis [78] and Sastry and Isidori [98] combine feedback linearization with adaptation techniques from adaptive linear control. However, to achieve global stability, these schemes require that the nonlinearities be restricted by linear growth conditions. In contrast, the adaptive backstepping design allows the stabilization of uncertain nonlinear systems without restrictions being placed on the system nonlinearities.

### 1.2.2 Adaptive Backstepping Design

The adaptive backstepping design procedure broke the extended matching barrier. It was developed in a series of enlightening contributions, regarding state feedback and output feedback control, by Kanellakopoulos, Kokotović and Morse ([47]-[50],[57]). The aim of their work was to provide a systematic framework for the design of regulation and tracking strategies suitable for a large class of state linearizable nonlinear systems exhibiting constant, but unknown, parameter values. For instance, this approach achieves the global regulation of the benchmark example

$$\begin{aligned} \dot{x}_1 &= x_2 + \theta x_1^2 \\ \dot{x}_2 &= x_3 \\ \dot{x}_3 &= u \end{aligned} \tag{1.28}$$

which obviously does not satisfy the EMC.

The initial largest class of uncertain systems for which backstepping provided feedback solutions was that of *parametric pure-feedback (PPF)* systems [48]. This class is well represented by the third order system

$$\begin{aligned} \dot{x}_1 &= x_2 + \varphi_1^T(x_1, x_2)\theta \\ \dot{x}_2 &= x_3 + \varphi_2^T(x_1, x_2, x_3)\theta \\ \dot{x}_3 &= u + \varphi_3^T(x_1, x_2, x_3)\theta \end{aligned} \tag{1.29}$$

where $\theta \in \Re^p$ is a constant unknown vector. PPF systems are characterized by both linear parameterization and the structure of the known nonlinearities $\varphi_1, \varphi_2$, and $\varphi_3$. The function $\varphi_1$ must not depend on $x_3$, and a further implicit function restriction is imposed on the dependence of $\varphi_1$ on $x_2$, and of $\varphi_2$ on $x_3$ [48]. This restriction is automatically satisfied by the subclass of *parametric strict-feedback (PSF) systems*, in which $\varphi_1$ does not depend on $x_2$, and $\varphi_2$ does not depend on $x_3$. Global regulation and tracking properties are guaranteed for PSF systems. The benchmark problem (1.28) is an example of an uncertain nonlinear system in PSF form.

The basic idea of backstepping is to design a controller for (1.29) by following a step-by-step procedure which interlaces, at each step, the change of coordinates required for feedback linearization, and the construction of parameter update laws required for adaptation. This is achieved by considering some of the state variables as "virtual controls" and designing intermediate "control laws". For instance, in (1.29) the first virtual control is $x_2$. It is used to stabilize the first equation as a separate system. Since $\theta$ is unknown, this task is solved with an adaptive controller consisting of the control law $\alpha(x_1)$ and the update law $\dot{\hat{\theta}} = \tau(x_1)$, which is obtained by Lyapunov-based design. In the next step the state $x_3$ is the virtual control which is used to stabilize the subsystem consisting of the first two equations of (1.29). The overparameterized algorithm in [48] treats the parameter $\theta$ as a new parameter and assigns to it a new estimate with a new update law. This overparameterization was removed in the adaptive backstepping algorithm with *tuning functions* [61], by treating $\dot{\hat{\theta}}$ in the first step not as an update law but only as the function $\tau(x_1)$. This tuning function is used in subsequent recursive steps and the discrepancy $\dot{\hat{\theta}} - \tau(x_1)$ is compensated with additional terms in the controller.

The proof of stabilization and tracking properties achieved by both the overparameterized algorithm and backstepping with tuning functions is a direct consequence of the recursive procedure, during which an additive Lyapunov function is constructed for the entire system, including the parameter estimates.

This systematic procedure and its stability and convergence properties are described fully in Chapter 2.

## 1.3 Robust Control of Uncertain Systems

As an alternative to adaptive control, the design of robust non-adaptive nonlinear state feedback control to yield stability for nonlinear systems with uncertainties, has been the subject of considerable research over the last fifteen years. The first attempts to deal with uncertainty were studied in the setting of non-cooperative games in the early 1970's. At that time, Leitmann and Gutman [37, 68] developed the notion of treating uncertainty as "the other player" in a two-person game. Thus, one player seeks to assure a desired outcome, say stability, in the presence of another player whose control actions he does not know or knows imperfectly.

Gutman [36] developed a discontinuous *min-max* control which asymptotically stabilizes nonlinear uncertain systems under the matching conditions. Thus, to assure the desired behaviour of the closed-loop system, an infinitely (in the ideal case) fast switching mechanism is required. The non-ideal but fast switching control yields high-frequency dynamics called *chattering.* This chattering may be undesirable in some systems because unmodelled high-frequency plant dynamics may be exciting resulting in unforeseen instabilities.

In order to avoid mathematical difficulties associated with non-classical differential equations and, more importantly, to obviate the occurrence of *chattering*, Corless and Leitmann [19] introduced a class of continuous state feedback control under the same matching conditions. This guaranteed stability from a classical differential equation setting, but at the expense of giving up asymptotic stability for *uniform ultimate boundedness*, which at least for systems with matched uncertainties, assures behaviour arbitrarily close to asymptotic stability.

Lyapunov stability theory is very useful in the design of stabilizing controls for uncertain systems. It can be used in a constructive fashion to obtain feedback controls which guarantee desired (or acceptable) stability. For instance, consider the uncertain system

$$\dot{x} = f(t,x) + g(t,x,u,\theta) \tag{1.30}$$

where $x$ is the state, $u$ the control and $\theta \in \Theta$ the uncertainty. The *nominal* system

$$\dot{x} = f(t,x) \tag{1.31}$$

is Lyapunov stable with a Lyapunov function $V(t,x)$. The problem then is to determine a feedback control $u = \alpha(t,x)$ such that $V(t,x)$ is also a Lyapunov function for

$$\dot{x} = f(t,x) + g(t,x,\alpha(t,x),\theta) \tag{1.32}$$

for all $\theta \in \Theta$.

Stabilizing feedback controls are readily obtained if the so-called matching conditions are fulfilled ([69],[19]). Several results regarding robustness in the absence of matching conditions have been proposed ([3],[16]-[17],[86]). These results cope with mismatched uncertainties and are based on the *stability margin* of the stabilized nominal system. For example, if the nominal system can be stabilized, it has been shown that uncertain systems with arbitrarily large unmatched uncertainties can be also stabilized [86]. Thorp and Barmish [120] introduced *generalized matching conditions.* These are structural conditions on the uncertainty which are less restrictive than matching conditions and permit quadratic stabilization via linear control, regardless of the size of most of the uncertain elements. Generalized matching conditions have been extended to nonlinear systems by Qu [84]. Recently, systematic robust control design techniques have been proposed for systems with unmatched uncertainties ([31]-[32], [85]). These approaches were inspired by the recursive backstepping algorithm to provide stabilization of uncertain systems when information on uncertainty bounds is available.

An alternative approach in the control regulation of uncertain systems is the Sliding Mode Control (SMC) technique and its associated feedback control law of Variable Structure Control (VSC) systems. The outstanding feature of VSC is the excellent robustness and invariance properties in the face of disturbances and unmodelled dynamics. On the other hand, a common and popular technique used in the context of power electronic circuits control is that of Pulse Width Modulation (PWM), which has been shown to exhibit robust properties similar to those of VSC systems [104].

### 1.3.1 Sliding Mode Control

Generally speaking, SMC is a high-speed discontinuous control which switches on a manifold, i.e. the gain switches between two values (structures) according to a rule that depends on the value of the state at each time instant. The objective of the switching control law is to drive the state trajectories of the nonlinear plant towards a prescribed *switching (sliding) surface* in the state space, and constrain them to lie upon this surface for all subsequent time. The motion that ideally arises when the system state crosses and re-crosses a switching surface is called *sliding motion*([122],[126]). In the sliding mode the system is totally invariant to a class of matched disturbances and parameter variations with known bounds; thus, the closed-loop system dynamics are wholly characterized by the reduced order dynamics of the selected surface. Therefore, a crucial phase of SMC is to define a sliding surface so that the plant, restricted to the surface, has desired dynamics and properties such as stability to the origin, regulation and tracking.

SMC design is carried out in two phases. The first phase is to choose a sliding surface so that the plant state restricted to the surface has desired dynamics. The second phase is to design a switched control that will drive the plant state to the switching surface and maintain it on the surface thereafter. Usually, a Lyapunov approach is used to achieve this second design phase. The Lyapunov function, which characterizes the motion of the state to the surface, is defined in terms of the surface. For each switched control structure one chooses the control law so that the derivative of this Lyapunov function is negative definite, thus guaranteeing motion of the state trajectory to the surface.

VSC with sliding mode was intensively studied in the 1960's. This was motivated by the introduction of appropriate mathematical tools that allowed the formal description of the sliding mode. The work of Utkin [122] provided notable contributions to this area. Since then SMC has attracted the attention of many researchers and has allowed the introduction of new ideas and the broadening of applications to many practical systems (see, for example, [22], [33], [126]).

In order to illustrate the design tasks carried out during the two phases of SMC design, consider the following single input nonlinear system

$$\dot{x} = f(x) + g(x)u \tag{1.33}$$

where $x \in \Re^n$ is the state, $u \in \Re$ the control input, and $f(x)$ and $g(x)$ are smooth vector fields. The objective of SMC is to design a switching surface

$$\sigma(x) = 0 \tag{1.34}$$

in the state space, together with a switched controller of the form

$$u(x) = \begin{cases} u^+(x) & \text{if} \quad \sigma(x) > 0 \\ u^-(x) & \text{if} \quad \sigma(x) < 0 \end{cases} \tag{1.35}$$

which takes the value depending on the sign of the switching function at $x$. The control values $u^+(x)$ and $u^-(x)$ are chosen so that the tangent vectors of the state trajectories point towards the surface such that the state is driven to and maintained on $\sigma(x) = 0$.

A sliding mode exists on the surface $\sigma(x) = 0$ whenever the "distance" to this surface and the velocity $\dot{\sigma}$ are of opposite signs ([122]), i.e. when

$$\lim_{\sigma \to 0^-} \dot{\sigma} > 0 \qquad \lim_{\sigma \to 0^+} \dot{\sigma} < 0 \tag{1.36}$$

or, equivalently

$$\sigma(x)\dot{\sigma}(x) \leqslant 0. \tag{1.37}$$

If the reduced-order dynamics for $\sigma(x) = 0$ is stable, the VSC design problem for single input systems can be readily solved by finding a control law such that the time derivative of the Lyapunov function

$$V(x) = \frac{1}{2}\sigma^2(x) \tag{1.38}$$

is negative definite. By achieving this, the existence condition (1.37) is automatically fulfilled and, as a consequence, state trajectories invariably converge to the sliding surface and remain on it in a sliding mode (see Figure 1.1). Thus, the problem of existence of the sliding mode becomes a feedback control design problem.

The design of the sliding surface is determined by the system behaviour in the sliding mode. This behaviour depends on the parameters of the switching surface. Nonlinear switching surfaces are non-trivial to design. In contrast, for the linear case the switching surface design problem can be converted into an equivalent state feedback design problem. The design of the switching surface requires one to specify the motion of the state trajectory in a sliding mode, for which the so-called *equivalent control method* has been developed [122].

### Equivalent Control Method

The equivalent control method provides a formal procedure to obtain the sliding equations when the state trajectory lies upon the sliding surface. Equivalent control constitutes an equivalent continuous input which, when applied to the controlled system, produces the motion of the system on the sliding surface for the initial state on the surface. The equivalent control can be found from

$$\dot{\sigma}(x) = 0 \tag{1.39}$$

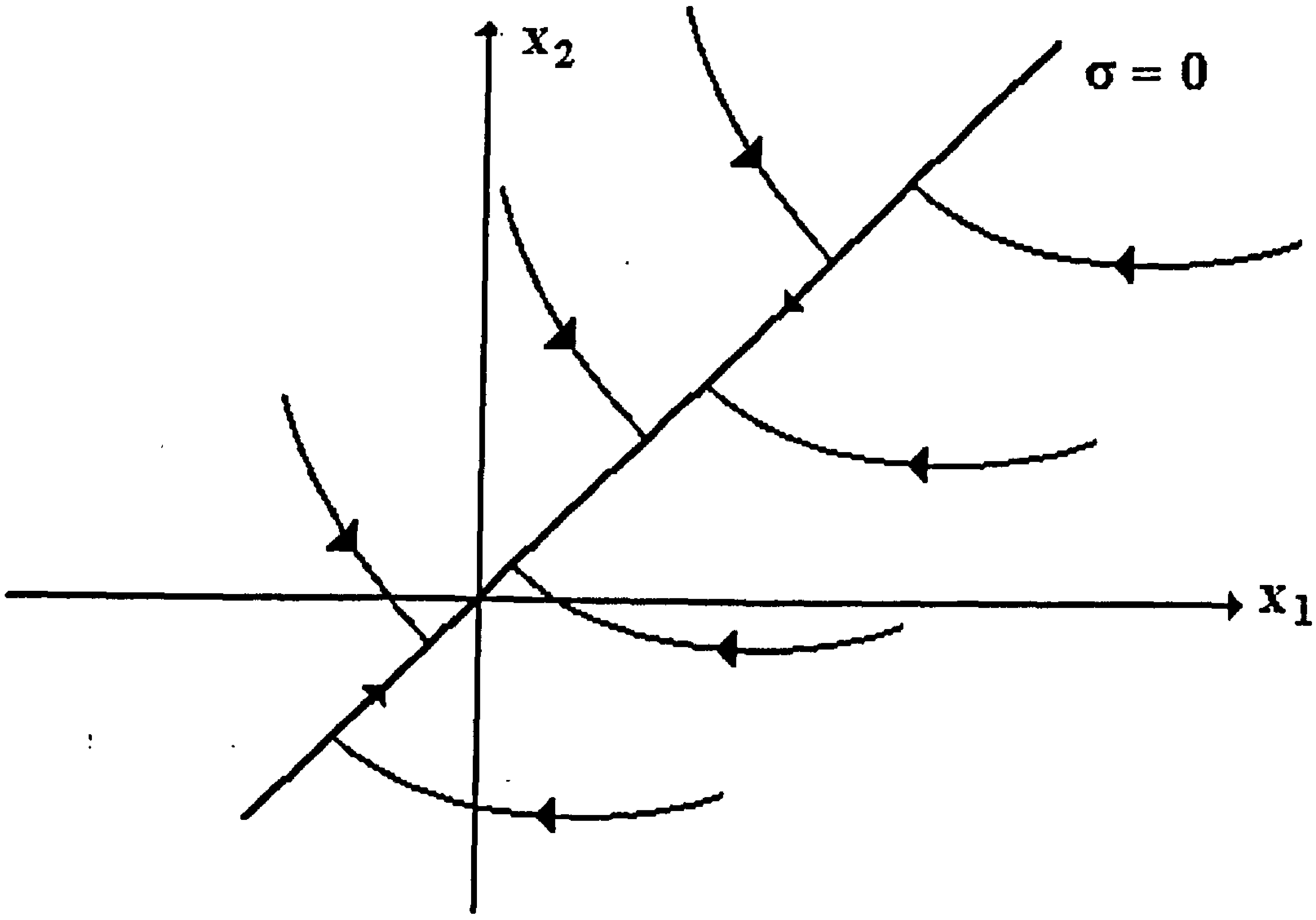


Figure 1.1: (a) Sliding mode in a bidimensional state space.

It can be interpreted as the continuous control law that would maintain $\dot{\sigma} = 0$ if the dynamics were exactly known. For instance, for the system (1.33), $\dot{\sigma}$ yields

$$\dot{\sigma}(x) = \frac{\partial \sigma}{\partial x}\Big(f(x) + g(x)u\Big) \tag{1.40}$$

Hence, the equivalent control that renders $\dot{\sigma} = 0$ is given by

$$u_{eq} = -\left(\frac{\partial \sigma}{\partial x} g(x)\right)^{-1}\left(\frac{\partial \sigma}{\partial x} f(x)\right) \tag{1.41}$$

Note that to guarantee the existence of the equivalent control the condition

$$\frac{\partial \sigma}{\partial x} g(x) \neq 0 \tag{1.42}$$

should be fulfilled, at least locally. Condition (1.42) is also a geometric condition for the design of a sliding surface, since surfaces that do not satisfy (1.42) are not valid for generating a sliding mode. Geometrically, the equivalent control can be constructed as the convex combination of the values $u^+$ and $u^-$ of $u$ on both sides of the surface $\sigma(x)$. This formal justification was derived in the early 1960's by Filippov [25].

By substituting the equivalent control into the original system (1.33), one obtains the equations governing the system response during *ideal sliding motion*

$$\dot{x} = f(x) + g(x)u_{eq}. \tag{1.43}$$

The reduced-order dynamics of (1.43) are characterized by the geometry of the switching surface $\sigma(x)$.

The structure of the actual control $u$ applied to real systems is decomposed into

$$u(x) = u_{eq}(x) + u_n(x) \tag{1.44}$$

where $u_n(x)$ is the discontinuous control used to guarantee stability and drive the state trajectory towards the sliding surface. For the system (1.33), a candidate Lyapunov function is

$$V(x,\sigma) = \frac{1}{2}\sigma^2(x) \tag{1.45}$$

The derivative of (1.45) yields

$$\dot{V} = \sigma\dot{\sigma} = \sigma\frac{\partial\sigma}{\partial x}\left(f(x) + g(x)u\right) \tag{1.46}$$

By decomposing $u$ as (1.44) and using the equivalent control (1.41), one obtains

$$\dot{V} = \sigma\left(\frac{\partial\sigma}{\partial x}g(x)u_n(x)\right). \tag{1.47}$$

Hence, it is convenient to set

$$u_n(x) = -\left(\frac{\partial\sigma}{\partial x}g(x)\right)^{-1}\alpha\ \mathrm{sgn}(\sigma), \qquad \alpha > 0 \tag{1.48}$$

to yield

$$\dot{V} = -\alpha|\sigma|. \tag{1.49}$$

Therefore, exponential convergence of the state trajectories towards the sliding surface is guaranteed.

**Example 1.5** Consider the second order system

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} x_2 \\ \varphi(x) \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u = f(x) + g(x)u \tag{1.50}$$

where $\varphi(x)$ is a known nonlinear function. To determine the structure of an appropriate sliding surface recall that

$$\frac{\partial\sigma}{\partial x}g(x) \neq 0$$

Since $g(x) = [0 \quad 1]^T$, it follows that $\frac{\partial\sigma}{\partial x_2}$ must be nonzero. Hence, by choosing $\frac{\partial\sigma}{\partial x_2} = 1$, it is sufficient to consider sliding surfaces of the form

$$\sigma(x) = \sigma_1(x_1) + x_2 = 0 \tag{1.51}$$

If one presumes that the reduced-order dynamics is given by

$$\dot{x}_1 = -\lambda x_1, \qquad \lambda > 0 \tag{1.52}$$

the switching surface structure (1.51) implies that in the sliding mode

$$x_2 = -\sigma_1(x_1). \tag{1.53}$$

The reduced-order dynamics becomes

$$\dot{x}_1 = x_2 = -\sigma_1(x_1) = -\lambda x_1. \tag{1.54}$$

Hence, the switching surface design is completed by setting

$$\sigma(x) = x_2 + \lambda x_1. \tag{1.55}$$

The equivalent control obtained from (1.41) yields

$$u_{eq}(x) = -\varphi(x) - \lambda x_2. \tag{1.56}$$

To complete the control design task, the constant gain relay control structure $u_n(x)$ is chosen according to (1.48) as

$$u_n(x) = -\alpha \ \mathrm{sgn}(\sigma(x)), \qquad \alpha > 0. \tag{1.57}$$

### Robustness to Matched Disturbances

In order to illustrate the robustness of VSC to disturbances (and/or parameter variations), considered the modified system

$$\dot{x} = f(x) + g(x)\Big(u + \varphi(x,\theta)\Big) \tag{1.58}$$

where $\varphi(x,\theta)$ is a matched unknown function bounded by a positive continuous function

$$|\varphi(x,\theta)| \leqslant \overline{\varphi}(x). \tag{1.59}$$

To incorporate robustness into a VSC design the decomposed control structure (1.44) is convenient. Furthermore, the design of a sliding surface $\sigma(x)$ with stable reduced-order dynamics, when $\sigma(x) = 0$, is vital to guarantee stability and robustness. By considering the Lyapunov function

$$V(x,\sigma) = \frac{1}{2}\sigma^2(x), \tag{1.60}$$

the derivative

$$\dot{V} = \sigma\dot{\sigma} = \sigma\frac{\partial\sigma}{\partial x}\Big[f(x) + g(x)\Big(u(x) + \varphi(x,\theta)\Big)\Big]. \tag{1.61}$$

The use of the equivalent control (1.41) yields

$$\dot{V} = \sigma\frac{\partial\sigma}{\partial x}\Big[g(x)\Big(u_n(x) + \varphi(x,\theta)\Big)\Big]. \tag{1.62}$$

Hence, it is necessary to choose $u_n(x)$ so that (1.62) is nonpositive. By using the discontinuous control

$$u_n(x) = -\left(\overline{\varphi}(x) + \alpha\right)\ \mathrm{sgn}\left(\sigma\frac{\partial\sigma}{\partial x}g(x)\right), \quad \alpha > 0 \tag{1.63}$$

we obtain

$$\begin{aligned}\dot{V} &= -\left(\overline{\varphi}(x) + \alpha\right)\left|\sigma\frac{\partial\sigma}{\partial x}g(x)\right| + \sigma\frac{\partial\sigma}{\partial x}g(x)\varphi(x,\theta)\\ &\leqslant -\alpha\left|\sigma\frac{\partial\sigma}{\partial x}g(x)\right| \leqslant 0\end{aligned} \tag{1.64}$$

for stability.

### Chattering

SMC assures the desired behaviour of the closed-loop system via an infinitely (in the ideal case) fast switching mechanism. In practical applications this ideal condition of infinitely fast switching is not achieved and as a consequence the phenomenon of *chattering* appears. Chattering contains high frequency components which may excite unmodelled plant dynamics. Thus, in order to achieve proper performance of the controller, chattering should be eliminated in certain physical systems. This can be accomplished by smoothing out the control discontinuity in a thin *boundary layer* neighbouring the switching surface ([7, 114])

$$B = \{x\ :\ |\sigma(x)| \leqslant \varepsilon\} \tag{1.65}$$

whose width is $2\varepsilon$. The control law is modified as follows

$$u = \begin{cases} u_{eq} + u_n(x) & |\sigma(x)| \geqslant \varepsilon \\ u_{eq} + u_c(\sigma,x) & |\sigma(x)| < \varepsilon \end{cases} \tag{1.66}$$

where $u_c(\sigma,x)$ is any continuous function satisfying $u_c(0,x) = 0$ and $u_c(\sigma,x) = u_n(x)$ when $|\sigma(x)| = \varepsilon$.

This control guarantees that trajectories are attracted to the boundary layer. Inside the boundary layer, (1.66) provides a continuous approximation to the usual discontinuous control action. Intuitively, the smoothing of control discontinuity inside the boundary layer essentially assigns a low-pass filter structure to the local dynamics, thus eliminating chattering. However, asymptotic stability is no longer guaranteed but ultimate boundedness of trajectories to within an $\varepsilon$-dependent neighbourhood of the origin is assured ([19]).

Recently, the use of dynamical sliding mode controllers have been proposed as an alternative mechanism to reduce undesirable chattering ([30], [105]). By using this approach, the sliding mode control design is carried out for a dynamically extended version of the controlled plant so that, chattering generated at the dynamical controller for the extended system appears reduced at the actual control applied to the original system. The controller structure thus obtained is more complex but chattering is alleviated and asymptotic stability is still guaranteed. This approach will be adopted as the chattering alleviation procedure throughout this thesis.

### 1.3.2 Pulse Width Modulation

We will now describe control via Pulse Width Modulation (PWM). An important subject associated with power supplies and electromechanical systems is power electronics. Power electronics circuits are principally concerned with processing energy and convert electrical energy from the form supplied by a source to the form required by a load. For instance, when the conversion process concludes with mechanical motion, the power circuit converts electric energy to the form required by the electromechanical transducer, for example a DC-motor.

A special class of power electronics circuits is used to change the character of electrical energy: from AC to DC, from one voltage level to another, or in some other way. These circuits are called *converters.* DC-to-DC converters are used extensively in power supplies for electronic equipment to control the flow of energy between two DC systems.

The conversion of DC power is exclusively performed in the switched mode ([53],[101]). Switched converters are power circuits in which semiconductor devices switch at high repetitions (high frequency), compared to the variation of the input and output waveforms, between the two DC terminals. The actual power flow is controlled by the on/off ratio of the respective switches. Examples of their use are the power supplies in computers and other electronic equipment.

The simplest form of a DC-to-DC converter is that of Figure 1.2(a). The switch opens and closes at a frequency $1/T$, with the ratio of the on-time to the period defined as $\mu$. The resulting voltage $V_2$ is a *chopped* version of the input, i.e. a series of pulses having an amplitude of $V_1$ and an average or DC value of $\mu V_1$ as shown in Figure 1.2(b). Usually this DC signal has a substantial amount of ripple (chattering), present not only

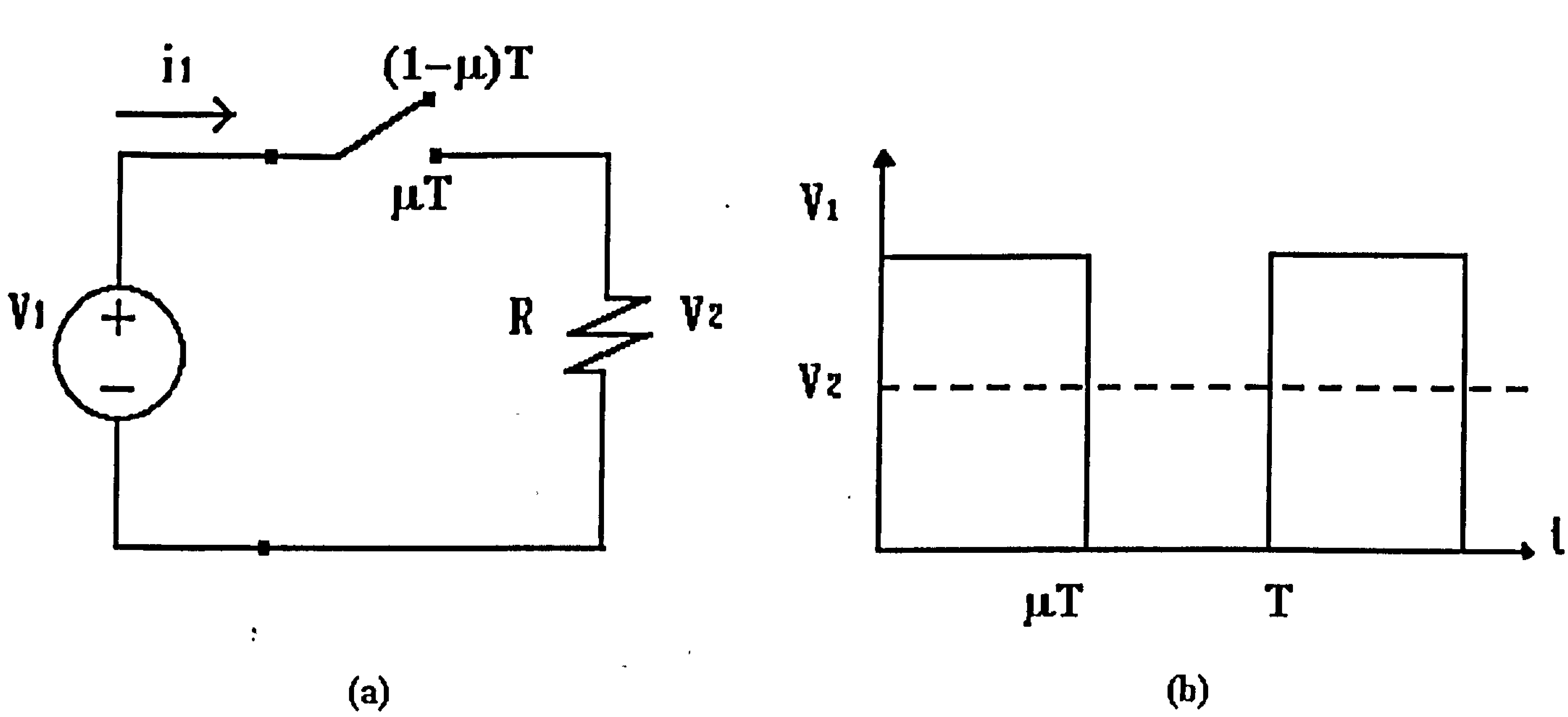


Figure 1.2: (a) The simplest form of a switching DC-to-DC converter. (b) Input and output waveforms.

in the load voltage $V_2$ but also in the source current $i_1$. The high frequencies contained in the ripple can cause both conducted and radiated interference with other apparatus, such as computers or communications equipment ([53]). Therefore, to obtain the desired ripple-free input current and output voltage, one must insert low-pass filters at the input and output.

DC-to-DC converters are called *up, down* or *up/down* converters to describe their ability of increasing, diminishing or both increasing and diminishing the voltage level presents at their input terminals. Example are the bilinear *Boost, Buck* and *Buck-Boost* converters respectively. Their state space models can be represented by the following switched controlled dynamical nonlinear system

$$\dot{\xi} = f(\xi) + g(\xi)u + \eta \tag{1.67}$$

where $f(\cdot)$ and $g(\cdot)$ are smooth vector fields defined on an open set of $\Re^n$, $\eta$ is a constant vector and $u$ denotes the switch position function acting as a control input, and taking values in the binary discrete set $\{0, 1\}$ ([103, 104]).

Switchmode DC-to-DC power converters are usually regulated by means of Pulse-Width-Modulation (PWM) feedback control laws. An idealized PWM control strategy is that in which switching to the $u = 1$ position are assumed to occur at the beginning of each period known as the *duty cycle*, and change to the $u = 0$ position once within the duty cycle according to a switching policy determined by a smooth feedback function of the state vector $\xi$, known as the *duty ratio* and denoted by $\mu(\xi)$. The duty cycle is assumed to be periodic with infinitesimally small period (i.e. infinitely high frequency), and the duty ratio is the fraction of the duty cycle on which the switch position is at $u = 1$ (see Figure 1.3). Hence $0 < \mu(\xi) < 1$. Under the high frequency assumption

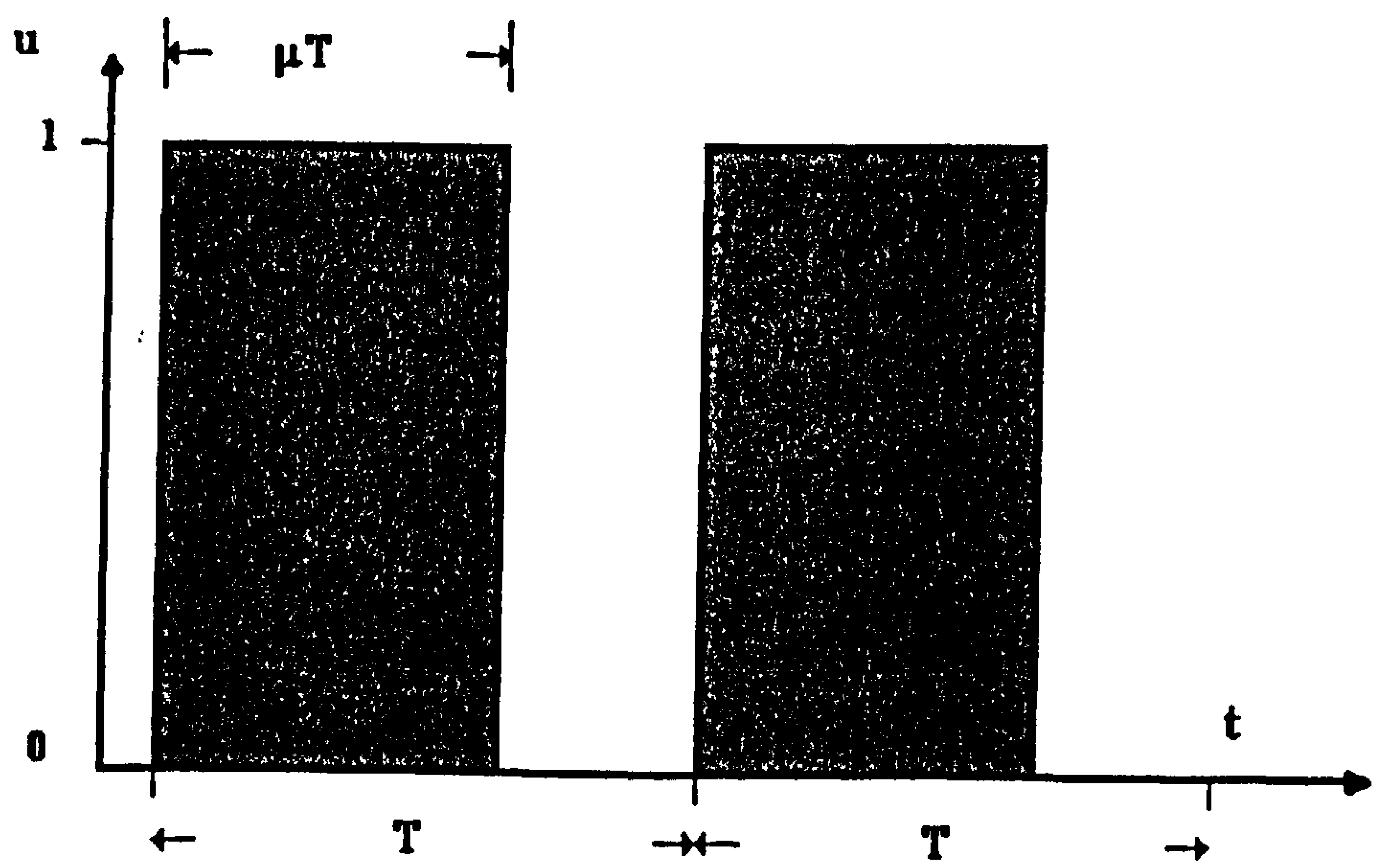


Figure 1.3: Typical duty cycle and duty ratio

it has been rigorously demonstrated in [103] that an *average PWM model* is obtained by formally replacing the switch position function $u$ by the duty ratio function $\mu$. The average model satisfies

$$\dot{x} = f(x) + g(x)\mu + \eta. \tag{1.68}$$

The average model (1.68) has the advantage of reducing any regulation or tracking problem, defined by the converter model (1.67), to a standard nonlinear feedback controller design problem in which the duty ratio function plays the role of the required input variable. However, the designed feedback control law must be limited to vary in the interval $[0, 1]$ for implementation purposes.

A PWM feedback strategy for the specification of the switch position $u$ is given as

$$u = \begin{cases} 1 & \text{for} \quad t_k \leqslant t < t_k + \mu_r(t_k)T \\ 0 & \text{for} \quad t_k + \mu_r(t_k)T \leqslant t < t_k + T \end{cases} \tag{1.69}$$

where $T$ is the sampling period assumed constant, $\mu_r(t_k)$ is the value of the restricted duty ratio function at the sampling instant $t_k$. The restricted duty function is obtained from the duty ratio function $\mu$, computed via feedback, when it is limited to take values in the continuous closed interval $[0, 1]$.

In [104] a general relationship was established for average PWM-controlled responses and ideal sliding mode controlled trajectories. Integral manifolds of average PWM-controlled networks qualify as sliding surfaces on which the corresponding equivalent control coincides with the duty ratio of the PWM control scheme. PWM control laws exhibit robust properties similar to SMC systems.

It is worth emphasizing that the underlying common assumption made in obtaining both average models (average PWM model and ideal sliding dynamics) is the high (infinite) frequency assumption.

## 1.4 Robust Adaptive Control

It was demonstrated in [39] that in the presence of unmodelled dynamics and disturbances, an adaptive controller designed for the ideal situation, i.e. no modelling errors or disturbances, could exhibit instability. Since then, considerable research has been directed towards the development of *robust adaptive control schemes*, which can retain certain stability properties in the presence of a wide class of modelling errors ([40]-[42],[83],[97]).

In order to motivate the need for robustness in adaptive control schemes, consider again the scalar plant (1.3) together with the adaptive controller

$$u = -\hat{\theta}x, \qquad \dot{\hat{\theta}} = kx^2 \tag{1.70}$$

which accomplishes the objective of stabilizing the state $x$ to the origin. Suppose that instead of (1.3), the actual plant is described by

$$\dot{x} = \theta x + u + d \tag{1.71}$$

where $d$ is an unknown bounded disturbance.

It was demonstrated in [38] that for $\hat{\theta}(0) = 5,\ x(0) = 1,\ \theta = 1,\ k = 1$ and

$$d(t) = (1+t)^{-\frac{1}{5}}\left[5 - (1+t)^{-\frac{1}{5}} - 0.4(1+t)^{-\frac{6}{5}}\right] \tag{1.72}$$

the solution of (1.70), (1.71) is given by

$$x(t) = (1+t)^{-\frac{2}{5}} \to 0 \quad \text{as} \quad t \to \infty \tag{1.73}$$

and

$$\hat{\theta}(t) = 5(1+t)^{\frac{2}{5}} \to \infty \quad \text{as} \quad t \to \infty \tag{1.74}$$

i.e. the estimated parameter drifts to infinity with time. This instability phenomenon is called *parameter drift*. By noting that $\hat{\theta}(0) = 5 > 1$ and $\hat{\theta}(t) \geqslant \hat{\theta}(0),\ \forall\, t \geqslant 0$, it is easy to verify that parameter drift can be stopped at any time $t$ by switching off adaptation, i.e. setting $k = 0$ in (1.70). This observation motivated the use of an adaptive law with a *dead-zone*, as a way to counteract parameter drift.

To avoid this and other phenomena that may cause instabilities in adaptive control systems, a number of robust adaptive controllers have been developed. These are based mainly upon the use of the certainty equivalence approach. Attempts have been also made in [41], [38] to unify the existing algorithms, which were recently compiled in [42].

In this thesis the combination of adaptive backstepping and sliding mode control techniques is proposed as a way to provide robustness in the presence of disturbances, and its effectiveness is shown for a number of practical examples. The contents of the thesis is organized as follows:

In Chapter 2 the geometric conditions needed for an uncertain system to be transformable into the PPF and PSF triangular forms are given and the Static Adaptive Backstepping (SAB) algorithm applicable to these systems is described. A deterministic class of triangular linearizable systems is also considered and a Static Deterministic Backstepping (SDB) algorithm is developed for this class of systems.

A natural extension of static feedback linearization is dynamical feedback linearization. It is characterized by the fact that the linearizing control law is dynamical. We describe in Chapter 3 a new Dynamical Deterministic Backstepping (DDB) algorithm which achieves a linear differential relation between the input and the output via dynamical compensation. This algorithm is applicable to deterministic observable minimum phase nonlinear systems and motivates the development of its adaptive version for uncertain systems, described in Chapter 4. This new Dynamical Adaptive Backstepping (DAB) algorithm does not use canonical forms and broadens the applicability of backstepping to *observable minimum phase* uncertain nonlinear systems with nontriangular forms. Chapter 5 presents the application of the DAB algorithm to design duty ratio functions which are employed in PWM control strategies for the regulation of DC-to-DC power converters.

In Chapters 6 and 7 output tracking problems of uncertain systems in the presence of undesirable disturbances are addressed via the combination of adaptive backstepping and SMC. Two new systematic combined algorithms SAB-SMC and DAB-SMC have been developed for PSF and observable minimum phase systems, respectively. Examples illustrate the performance of the combined controllers in tracking tasks.

A limitation of these control design algorithms is associated with the complexity of the equations arising at each step, when the control design is carried out by hand computation. This is true even for low order ($n \leqslant 3$) systems. To overcome this limitation a symbolic algebraic toolbox called BACKDSMC has been developed. This toolbox implements the backstepping and combined backstepping-SMC algorithms described in this thesis via the MATLAB symbolic toolbox. The use of BACKDSMC in the design of static (dynamic) deterministic (adaptive) backstepping controllers is explained in Chapter 8 in a tutorial manner.

A more realistic control design problem is considered in Chapter 9, where only the output of the uncertain nonlinear system is available for measurement. A solution is provided via the design of an adaptive observer and a dynamical adaptive output feedback control for a more restricted class of observable minimum phase systems which are transformable into the *adaptive generalized observer canonical form.*

Some concluding remarks and suggestions for further research are given in Chapter 10. In addition two appendices has been incorporated. Appendix A contains some basic concepts about stability and Appendix B has the full MATLAB code program of the toolbox BACKDSMC.

The author has published a number of papers ([87]-[94]) on the topics covered in this thesis.

# Chapter 2

# Classical Backstepping Control Design

## 2.1 Introduction

The analysis of nonlinear control systems, consisting of the study of the interaction between input and output, and between input and state, allows one to establish analogies with interesting features of linear control systems. For instance, the extension of the notions of controllability and observability to nonlinear systems allows classification and the possibility of developing nonlinear feedback controllers and observers. However, the nontrivial task of analysing nonlinear control systems require the use of complex mathematical tools taken from differential geometry, topology and differential algebra.

An important problem is that of determining whether or not a nonlinear system is linearizable, i.e. can be converted to a controllable linear system via a nonlinear transformation and a feedback control law. Systems exhibiting this desirable property can be stabilized by employing well-known linear control design techniques, such as pole-placement or linear optimal control.

A difficult problem is the stabilization of nonlinear control systems containing parametric uncertainty. Adaptive control has been used for forty years in the regulation of uncertain systems. In dealing with parametric uncertainty of nonlinear systems, one cannot use the separation principle, which is often applied in the case of linear systems to design the update law for identifying unknown parameters as a part of the feedback control. Lyapunov-based techniques are often used to overcome this limitation. This thesis is concerned with the design of adaptive control for single-input single-output nonlinear systems with parametric uncertainty of the form

$$\dot{\zeta} = f_0(\zeta) + \sum_{i=1}^{p} \theta_i f_i(\zeta) + \left[ g_0(\zeta) + \sum_{i=1}^{p} \theta_i g_i(\zeta) \right] u \tag{2.1}$$

where $\zeta \in \Re^n$ is the state, $u \in \Re$ the input, $\theta_1, \ldots, \theta_p$ a set of unknown parameters, and $f_i$, $g_i$, $0 \leqslant i \leqslant p$, are smooth vector fields in a neighbourhood of the origin $\zeta = 0$, with $f_i(0) = 0$, $0 \leqslant i \leqslant p$, $g_0(0) \neq 0$. A number of different techniques to design nonlinear adaptive controllers for this class of systems are described in [55], including the first version of the so-called backstepping algorithm.

This chapter is devoted to describing the classical backstepping approach developed by Kanellakopoulos *et al* [48] and Krstić *et al* [61], and the characterization of the classes of nonlinear systems for which this technique was originally developed. For the sake of completeness the exposition starts by introducing basic mathematical tools for the analysis of nonlinear systems, with emphasis on linearizable systems. Reader familiar with these concepts can omit this introductory part. Then we present recursive procedures (including the basic backstepping algorithm for deterministic systems) to linearize a special subclass of nonlinear systems in triangular form, via a coordinate transformation and a feedback control. Uncertain systems are introduced and geometric conditions are presented to characterize the classes of uncertain nonlinear systems transformable into either the parametric pure-feedback (PPF) form or the parametric strict-feedback (PSF) form. Finally, the classical adaptive backstepping algorithm with tuning functions is described, and its stability and convergence properties are analysed. Examples are given to illustrate the various concepts and control design algorithms.

## 2.2 Feedback Linearization

In this section the geometric characterization of the classes of input-output and state feedback linearizable systems is presented. A limited number of concepts from differential geometry are employed, and simple illustrative examples are considered.

The proofs of theorems, lemmas, propositions and corollaries given in this section have been omitted for the sake of brevity. They can be found in [44, 75, 81], which contain more detailed expositions concerning this subject.

It is well known from linear control theory that a linear system

$$\dot{x} = Ax + Bu \tag{2.2}$$

with $x \in \Re^n$ and $u \in \Re$ is transformable via a nonsingular linear transformation

$$z = Tx, \qquad z \in \Re^n \tag{2.3}$$

and a feedback law

$$u = Kx + v \tag{2.4}$$

into the Brunovsky controller form

$$\dot{z} = T(A+BK)T^{-1}z + TBv = A_c z + B_c v \tag{2.5}$$

with

$$A_c = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ -k_0 & -k_1 & -k_2 & \dots & -k_{n-1} \end{bmatrix}, \quad B_c = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \tag{2.6}$$

if and only if the Kalman controllability condition

$$\text{span}\{B, AB, \dots, A^{n-1}B\} = \Re^n \tag{2.7}$$

holds. Hereafter span$\{S\}$ denotes the space spanned by the set $S$.

A much more complicated problem is to find a nonlinear coordinate transformation and a feedback control for single-input nonlinear systems of the form

$$\dot{x} = f(x) + g(x)u, \quad x \in \Re^n, \quad u \in \Re \tag{2.8}$$

with $f(x)$ and $g(x)$ being smooth vector fields defined on an open subset of $\Re^n$, which places (2.8) into the Brunovsky controller form. This can be illustrated by the following example.

**Example 2.1** Consider the single-input nonlinear system

$$\begin{aligned} \dot{x}_1 &= x_2 + x_1^2 \\ \dot{x}_2 &= x_3 \\ \dot{x}_3 &= u \end{aligned} \tag{2.9}$$

The nonlinear transformation

$$\begin{aligned} z_1 &= x_1 \\ z_2 &= x_2 + x_1^2 \\ z_3 &= x_3 + 2x_1(x_2 + x_1^2) \end{aligned} \tag{2.10}$$

transforms (2.9) into

$$\begin{aligned} \dot{z}_1 &= z_2 \\ \dot{z}_2 &= z_3 \\ \dot{z}_3 &= u + z_2(4z_1^2 + 2z_2) + 2z_1(z_3 - 2z_1z_2) \end{aligned} \tag{2.11}$$

Then the nonlinear state feedback

$$u = -z_2(4z_1^2 + 2z_2) - 2z_1(z_3 - 2z_1z_2) + v \tag{2.12}$$

where $v$ is a reference input, gives the linear closed-loop system

$$\dot{z} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} z + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} v \tag{2.13}$$

which is in the Brunovsky controller form. Thus, the use of a nonlinear coordinate transformation and a state nonlinear feedback control has transformed a nonlinear system into a controllable linear system.

There exists mathematical theory which characterizes, in terms of necessary and sufficient conditions, those nonlinear systems which are *state feedback linearizable* (i.e. transformable into linear and controllable systems by a change of coordinates and nonlinear state feedback) and those which are *input-output linearizable* (i.e. transformable into a system with a linear input-output map). Before characterizing these classes of linearizable systems, some mathematical tools from differential geometry are introduced [44, 75, 81].

Consider single-input single-output nonlinear systems

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= h(x) \end{aligned} \tag{2.14}$$

where $x \in \Re^n$ is the state, $u \in \Re$ the input and $y \in \Re$ the output. $f(x)$ and $g(x)$ are smooth vector fields (i.e. continuous vector fields with continuous derivatives of any order) defined on an open set $R_o \subset \Re^n$ of an equilibrium point $x_o$, i.e. $f(x_o) = 0$, with $g(x_o) \neq 0$. Without loss of generality, it can be assumed that the origin is an equilibrium point of (2.14). Additionally, $h(x)$ is a smooth scalar function also defined on $R_o$.

For a smooth scalar function $h(x)$ of the state $x$, the *gradient* or *differential* of $h$ is defined as the row vector whose $i$-th element is the partial derivative of $h$ with respect to the state coordinate $x_i$, and is denoted by

$$\frac{\partial h}{\partial x}(x) = \left( \frac{\partial h}{\partial x_1} \; \frac{\partial h}{\partial x_2} \; \cdots \; \frac{\partial h}{\partial x_n} \right). \tag{2.15}$$

Similarly, given a vector field $f(x)$, the Jacobian of $f$ is represented by an $n \times n$ matrix of elements $\left(\frac{\partial f}{\partial x}\right)_{ij} = \frac{\partial f_i}{\partial x_j}$ and denoted by

$$\frac{\partial f}{\partial x} = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{pmatrix}. \tag{2.16}$$

Two types of differential operation, involving vector fields and scalar functions, are frequently used in the analysis of nonlinear control systems. The first type of operation involves a real-valued function $h$ and a vector field $f$, both defined on a subset $R_o$ of $\Re^n$.

**Definition 2.1** *Let $h(x)$ be a smooth scalar function defined on a subset $R_o \subset \Re^n$ and $f(x)$ a smooth vector field also defined on $R_o$. The* Lie derivative *of the function $h$ along the vector field $f$ is a scalar function often written as $L_f h$ and defined as*

$$L_f h(x) := \frac{\partial h}{\partial x} f(x) = \left\langle \frac{\partial h}{\partial x}, f(x) \right\rangle = \sum_{i=1}^{n} \frac{\partial h}{\partial x_i} f_i(x). \tag{2.17}$$

Repeated Lie derivatives can be defined recursively by

$$L_f^0 h = h,\ L_f^1 h = L_f h,\ \ldots,\ L_f^i h = L_f \left(L_f^{i-1} h\right). \tag{2.18}$$

Similarly, if $g(x)$ is another vector field defined on $R_o$, then the scalar function $L_g L_f h(x)$ is

$$L_g L_f h(x) = \frac{\partial (L_f h)}{\partial x} g(x). \tag{2.19}$$

The second type of operation involves two vector fields $f$ and $g$.

**Definition 2.2** *Let $f$ and $g$ be two vector fields defined on an open subset $R_o \subset \Re^n$. The* Lie bracket *of $f$ and $g$ is another vector field defined by*

$$[f, g](x) := \frac{\partial g}{\partial x} f(x) - \frac{\partial f}{\partial x} g(x). \tag{2.20}$$

Repeated Lie brackets of a vector field $g$ with the same vector field $f$ is possible. For this case the Lie bracket is commonly written as $ad_f g$ to define the recursive operation

$$ad_f^0 g(x) = g(x), \quad \ldots, \quad ad_f^i g(x) = \left[f, ad_f^{i-1} g(x)\right] \qquad i \geqslant 1. \tag{2.21}$$

**Example 2.2** Consider the two vector fields

$$f(x) = \begin{pmatrix} x_2 + x_1^2 \\ x_3 \\ 0 \end{pmatrix}, \qquad g(x) = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \tag{2.22}$$

Then

$$\begin{aligned} ad_f g(x) &= [f,g](x) = \frac{\partial g}{\partial x}f(x) - \frac{\partial f}{\partial x}g(x) = -\frac{\partial f}{\partial x}g(x) \\ &= -\begin{pmatrix} 2x_1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}\begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ -1 \\ 0 \end{pmatrix} \end{aligned} \tag{2.23}$$

and

$$\begin{aligned} ad_f^2 g(x) &= [f, ad_f g](x) = \frac{\partial (ad_f g)}{\partial x}f(x) - \frac{\partial f}{\partial x}ad_f g(x) = -\frac{\partial f}{\partial x}ad_f g(x) \\ &= -\begin{pmatrix} 2x_1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}\begin{pmatrix} 0 \\ -1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \end{aligned} \tag{2.24}$$

Coordinate transformations in the state space are very useful with regard to certain properties like *reachability* and *observability*, or to solve certain control problems such as *stabilization* or *decoupling*. In nonlinear systems, coordinate transformations are carried out by using diffeomorphisms which are defined as follows:

**Definition 2.3** *An $\Re^n$-valued function of $n$ variables*

$$z = \Phi(x) = \begin{pmatrix} \phi_1(x) \\ \phi_2(x) \\ \vdots \\ \phi_n(x) \end{pmatrix} = \begin{pmatrix} \phi_1(x_1,\ldots,x_n) \\ \phi_2(x_1,\ldots,x_n) \\ \vdots \\ \phi_n(x_1,\ldots,x_n) \end{pmatrix} \tag{2.25}$$

*defined on an open subset $R_o$ of $\Re^n$, is called a* local diffeomorphism *if it has the following properties*

*(i) $\Phi(x)$ is invertible, i.e. there exists a function $\Phi^{-1}(z)$ such that*

$$\Phi^{-1}(\Phi(x)) = x \qquad \forall x \in R_o \tag{2.26}$$

*(ii) $\Phi(x)$ and $\Phi^{-1}(z)$ are both smooth mappings, i.e. have continuous derivatives of any order.*

If $R_o$ is the whole space $\Re^n$, then $\Phi(x)$ is called a *global diffeomorphism*. However, since global diffeomorphisms are rare, one often looks for local diffeomorphisms defined in a local neighbourhood of a given point. Given a nonlinear function $\Phi(x)$, the following result is very useful to check whether or not it is a local diffeomorphism.

**Proposition 2.1** *Suppose $\Phi(x)$ is a smooth function defined on some subset $R_1$ of $\Re^n$. If the Jacobian matrix of $\Phi$ is nonsingular at a point $x = x_o$, then, on a suitable open subset $R_o$ of $R_1$ containing $x_o$, $\Phi(x)$ defines a local diffeomorphism.*

This can be illustrated by the following two examples.

**Example 2.3** Consider the function

$$\begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \Phi(x_1, x_2) = \begin{pmatrix} 3x_1 + x_2^2 x_1 \\ \sin x_2 \end{pmatrix} \tag{2.27}$$

which is defined for all $(x_1, x_2)$ in $\Re^2$. Its Jacobian matrix

$$\frac{\partial \Phi}{\partial x} = \begin{pmatrix} 3 + x_2^2 & 2x_1 x_2 \\ 0 & \cos x_2 \end{pmatrix} \tag{2.28}$$

has rank 2 at $x_o = (0, 0)$. Therefore the function (2.27) defines a local diffeomorphism around the origin.

**Example 2.4** Consider the function

$$\begin{pmatrix} z_1 \\ z_2 \\ z_3 \end{pmatrix} = \Phi(x_1, x_2, x_3) = \begin{pmatrix} x_1 \\ x_2 + x_1^2 \\ x_3 + 2x_1(x_2 + x_1^2) \end{pmatrix} \tag{2.29}$$

used as the coordinate transformation in Example 2.1. It is defined for all $(x_1, x_2, x_3)$ in $\Re^3$. The Jacobian matrix

$$\frac{\partial \Phi}{\partial x} = \begin{pmatrix} 1 & 0 & 0 \\ 2x_1 & 1 & 0 \\ 2x_2 + 6x_1^2 & 2x_1 & 1 \end{pmatrix} \tag{2.30}$$

has rank 3 for all $(x_1, x_2, x_3)$ in $\Re^3$. Therefore the function (2.29) defines a global diffeomorphism.

### 2.2.1 Distributions

A vector field $f$, defined on an open set $R_1$ of $\Re^n$, can be intuitively interpreted as a smooth mapping assigning the $n$-dimensional vector $f(x)$ to each point $x$ of $R_1$. Suppose now that $d$ smooth vector fields $f_1, \ldots, f_d$ are given, all defined on the same open set $R_1$, the concept of distribution can be similarly considered as a mapping assigning a vector space to each point $x$ of $R_1$. This can be formally expressed as follows:

**Definition 2.4** *A d-dimensional non-singular smooth* $\mathcal{D}$ distribution *on* $R_1$*, an open subset of* $\Re^n$*, is a map which assigns to each point* $x \in R_1$ *a d-dimensional subspace of* $\Re^n$ *such that for each* $x_o \in R_1$ *there exists a neighbourhood* $R_o$ *of* $x_o$ *and d smooth vector fields* $f_1, \ldots, f_d$ *with the properties*
*(i)* $f_1(x), \ldots, f_d(x)$ *are linearly independent at each* $x$ *in* $R_o$
*(ii)* $\mathcal{D}(x) = \text{span}\{f_1(x), \ldots, f_d(x)\}$ *at each* $x$ *in* $R_o$.
*Moreover, every smooth vector field* $\tau$ *belonging to* $\mathcal{D}$ *can be expressed on* $R_o$ *as*

$$\tau(x) = \sum_{i=1}^{d} c_i(x) f_i(x) \tag{2.31}$$

*where* $c_1(x), \ldots, c_d(x)$ *are smooth real-valued functions of* $x$*, defined on* $R_o$.

An important property associated with distributions is *involutivity* which is defined as follows:

**Definition 2.5** *A distribution* $\mathcal{D}$ *is called* involutive *if, given any two vector fields* $f_i$ *and* $f_j$ *belonging to* $\mathcal{D}$*, their Lie bracket* $[f_i, f_j]$ *also belongs to* $\mathcal{D}$.

Therefore checking whether or not a nonsingular distribution is involutive amounts to checking whether

$$\text{rank}\Big(f_1(x) \ \ldots \ f_d(x)\Big) = \text{rank}\Big(f_1(x) \ \ldots \ f_d(x) \ [f_i, f_j](x)\Big) \tag{2.32}$$

for all $x$ and all $1 \leqslant i, j \leqslant d$.

**Example 2.5** Consider a distribution

$$\mathcal{D} = \text{span}\{f_1, f_2\} \tag{2.33}$$

on $\Re^3$ with

$$f_1(x) = \begin{pmatrix} 0 \\ 0 \\ 1 + x_2^2 \end{pmatrix}, \qquad f_2(x) = \begin{pmatrix} 0 \\ -(1 + x_2^2) \\ 2x_2(x_3 + x_2^2 x_1) \end{pmatrix}. \tag{2.34}$$

This distribution has rank 2 for all $x \in \Re^3$. Since

$$[f_1, f_2](x) = \begin{pmatrix} 0 \\ 0 \\ 4x_2(1 + x_2^2) \end{pmatrix} \tag{2.35}$$

the matrix

$$(f_1\ f_2\ [f_1, f_2]) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & -(1+x_2^2) & 0 \\ 1+x_2^2 & 2x_2(x_3+x_2^2x_1) & 4x_2(1+x_2^2) \end{pmatrix} \quad (2.36)$$

has rank 2 for all $x \in \Re^3$ and thus the distribution is involutive. Indeed, from (2.31) the following relation is satisfied

$$[f_1, f_2](x) = c_1(x)f_1(x) + c_2(x)f_2(x) \quad (2.37)$$

with

$$c_1(x) = 4x_2, \qquad c_2(x) = 0$$

Therefore the vector field $[f_1, f_2](x)$ belongs to the distribution $\mathcal{D}$.

### 2.2.2 The Frobenius Theorem

In this section the integrability of a special set of partial differential equations of first order is analysed. A solution to this problem is very important to establish the class of feedback linearizable nonlinear systems. To this effect the concept of *codistribution* is first introduced.

**Definition 2.6** *A d-dimensional non-singular smooth* codistribution $\mathcal{W}$ *on an open subset* $R_1$ *of the dual space* $(\Re^n)^*$ *spanned by n-dimensional row vector (covector) fields* $\omega_i$, *is a map which assigns to each point* $x \in R_1$ *a d-dimensional subspace of* $(\Re^n)^*$ *such that for each* $x_o \in R_1$ *there exists a neighbourhood* $R_o$ *of* $x_o$, *and d smooth covector fields* $\omega_1, \ldots, \omega_d$ *with the properties*
*(i)* $\omega_1(x), \ldots, \omega_d(x)$ *are linearly independent at each* $x$ *in* $R_o$
*(ii)* $\mathcal{W}(x) = \text{span}\{\omega_1(x), \ldots, \omega_d(x)\}$ *at each* $x$ *in* $R_o$.

If $M$ is a matrix having $n$ independent columns with elements which are smooth functions of $x$, its rows can be regarded as smooth covector fields. Thus, any matrix of this kind identifies a codistribution; the codistribution being generated by the rows.

Codistributions can be constructed from given distributions in the following manner. Given a $d$-dimensional nonsingular distribution $\mathcal{D}$, the codistribution $\mathcal{D}^\perp$ can be constructed by the set of all row vectors $\omega$ which annihilate all the vectors in $\mathcal{D}$, i.e.

$$\mathcal{D}^\perp(x) = \{\omega \in (\Re^n)^* \ : <\omega, f>= 0 \quad \forall f \in \mathcal{D}(x)\} \quad (2.38)$$

The codistribution $\mathcal{D}^{\perp}$ is nonsingular, has dimension $n-d$ locally around each $x_o$, and is spanned by $n-d$ covector fields $\omega_1(x),\ldots,\omega_{n-d}$. By construction, the covector field $\omega_j$ is such that

$$\langle \omega_j(x), f_i(x)\rangle = 0 \quad \forall\, 1 \leqslant i \leqslant d,\ 1 \leqslant j \leqslant n-d \tag{2.39}$$

for all $x$ in $R_o$, i.e. satisfies the equation

$$\omega_j(x)F(x) = 0 \tag{2.40}$$

where $F(x)$ is the $n \times d$ matrix

$$F(x) = (f_1(x)\ \ldots\ f_d(x)). \tag{2.41}$$

Therefore, at any fixed point $x$ in $R_1$, (2.40) can be simply regarded as a linear homogeneous equation in the unknown $\omega_j(x)$.

Suppose now that one is interested in solving differential equations of the form

$$\frac{\partial \lambda_j}{\partial x}(f_1(x)\ \ldots\ f_d(x)) = \frac{\partial \lambda_j}{\partial x}F(x) = 0 \tag{2.42}$$

where $f_i(x)$, $1 \leqslant i \leqslant d$, are smooth vector fields defined on an open set $R_1$ of $\Re^n$. Note that (2.42) has the form of equation (2.40) with

$$\omega_j = \frac{\partial \lambda_j}{\partial x} \tag{2.43}$$

being differentials. In other words, one must find $n-d$ independent row vectors

$$\frac{\partial \lambda_1}{\partial x}, \ldots, \frac{\partial \lambda_{n-d}}{\partial x} \tag{2.44}$$

satisfying (2.42). The solution to this problem is possible when the distribution

$$\mathcal{D}(x) = \mathrm{span}\{f_1(x)\ \ldots\ f_d(x)\} \tag{2.45}$$

is completely integrable, i.e. for each point $x_o$ of $R_1$ there exist a neighbourhood $R_o$ of $x_o$ and $n-d$ real-valued functions $\lambda_1,\ldots,\lambda_{n-d}$, all defined on $R_o$, such that

$$\mathrm{span}\left\{\frac{\partial \lambda_1}{\partial x}\ \cdots\ \frac{\partial \lambda_{n-d}}{\partial x}\right\} = \mathcal{D}^{\perp} \tag{2.46}$$

on $R_o$. The Frobenius Theorem provides necessary and sufficient conditions for complete integrability.

**Theorem 2.1 (Frobenius)** *A nonsingular distribution is completely integrable if and only if it is involutive.*

**Example 2.6** Suppose that one is interested in determining whether or not the partial differential equations

$$\frac{\partial\lambda}{\partial x}\begin{pmatrix} 0 & 0 \\ 0 & -(1+x_2^2) \\ 1+x_2^2 & 2x_2(x_3+x_2^2x_1) \end{pmatrix} = (0 \quad 0) \tag{2.47}$$

are solvable. One needs to find a real-valued function $\lambda(x_1, x_2, x_3)$ which satisfies equation (2.47). The distribution $\mathcal{D} = \text{span}\{f_1, f_2\}$, with

$$f_1(x) = \begin{pmatrix} 0 \\ 0 \\ 1+x_2^2 \end{pmatrix}, \qquad f_2(x) = \begin{pmatrix} 0 \\ -(1+x_2^2) \\ 2x_2(x_3+x_2^2x_1) \end{pmatrix} \tag{2.48}$$

has rank 2 for all $x \in \Re^3$, and is also involutive, as shown in Example 2.5. Therefore, from the Frobenius theorem, $\mathcal{D}$ is integrable and thus the partial differential equations (2.47) are solvable.

With these mathematical tools we can now formulate the conditions for feedback linearization.

### 2.2.3 Input-Output Linearization

Input-output linearization means the generation of a *linear* differential relation between the output and the input. The notion of relative degree is first described.

**Definition 2.7** *The single-input single-output nonlinear system*

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= h(x) \end{aligned} \tag{2.49}$$

*is said to have* relative degree $\rho$ *at a point* $x_o$ *if*
*(i)* $L_gL_f^kh(x) = 0$ *for all* $x$ *in a neighbourhood of* $x_o$ *and all* $k < \rho - 1$, *and*
*(ii)* $L_gL_f^{\rho-1}h(x_o) \neq 0$.

In a simpler interpretation the relative degree $\rho$ is considered to be the least order of the output time derivative which is directly affected by the input $u$. This is illustrated as follows. Denote the output time derivative of order $i$ by

$$y^{(i)}(t) = \frac{d^iy}{dt^i} \tag{2.50}$$

Thus,

$$\begin{aligned} y^{(0)}(t) &= y(t) = h(x(t)), && (2.51)\\ y^{(1)}(t) &= \left\langle \frac{\partial h}{\partial x}, \dot{x} \right\rangle = L_f h + u L_g h && (2.52) \end{aligned}$$

If $\rho = 1$, since by definition $L_g h \neq 0$ in $R_o$, $y^{(1)}$ is directly affected by the input $u$. When $1 < \rho \leqslant n$, by virtue of the definition of $\rho$

$$\begin{aligned} y^{(i)} &= L_f^i h, \qquad 1 \leqslant i \leqslant \rho - 1 && (2.53)\\ y^{(\rho)} &= L_f^\rho h + u L_g L_f^{\rho-1} h. && (2.54) \end{aligned}$$

Hence, $y^{(\rho)}$ is the lowest order time derivative which is directly affected by the input $u$.

The $\rho$ functions $h(x), L_f h(x), \ldots, L_f^{\rho-1} h(x)$ are linearly independent and qualify as a partial set of new coordinate functions around the point $x_o$ [44]. The choice of these new coordinates places system (2.49) into a particularly simple structure.

**Proposition 2.2** *Suppose that the system (2.49) has relative degree $\rho$. Set*

$$\begin{aligned} \phi_1(x) &= h(x)\\ \phi_2(x) &= L_f h(x)\\ &\vdots\\ \phi_\rho(x) &= L_f^{\rho-1} h(x). \end{aligned} \qquad (2.55)$$

*If $\rho$ is strictly less than $n$, it is always possible to find $n - \rho$ functions $\phi_{\rho+1}, \ldots, \phi_n(x)$ such that the mapping*

$$\Phi(x) = \begin{pmatrix} \phi_1(x) \\ \vdots \\ \phi_n(x) \end{pmatrix} \qquad (2.56)$$

*has a Jacobian matrix which is nonsingular at $x_o$ and therefore qualifies as a local coordinate transformation in a neighbourhood of $x_o$. The value of these additional functions at $x_0$ can be fixed arbitrarily. Moreover, it is always possible to choose the $n - \rho$ functions in such a way that, for all $x$ in the neighbourhood of $x_o$,*

$$L_g \phi_i(x) = 0 \qquad \forall\, i \quad s.t. \quad \rho + 1 \leqslant i \leqslant n. \qquad (2.57)$$

The description of the system in the new coordinates $z = \Phi(x)$ can be easily found. The first $\rho - 1$ equations correspond to a chain of integrators and the $\rho$-th equation contains the control input $u$ explicitly, as follows

$$\begin{aligned}
\dot{z}_1 &= z_2 \\
\dot{z}_2 &= z_3 \\
&\vdots \\
\dot{z}_{\rho-1} &= z_\rho \\
\dot{z}_\rho &= a(z) + b(z)u
\end{aligned} \tag{2.58}$$

with

$$b(z) = L_g L_f^{\rho-1} h\left(\Phi^{-1}(z)\right) \tag{2.59}$$

$$a(z) = L_f^\rho h\left(\Phi^{-1}(z)\right). \tag{2.60}$$

The remaining $n-\rho$ equations do not have any special structure and they take the form

$$\begin{aligned}
\dot{z}_{\rho+1} &= q_{\rho+1}(z) + p_{\rho+1}(z)u \\
&\vdots \\
\dot{z}_n &= q_n(z) + p_n(z)u
\end{aligned} \tag{2.61}$$

with

$$q_i(z) = L_f\phi_i\left(\Phi^{-1}(z)\right), \quad p_i(z) = L_g\phi_i\left(\Phi^{-1}(z)\right) \quad \rho+1 \leqslant i \leqslant n. \tag{2.62}$$

Nevertheless, if $\phi_{\rho+1}(x), \ldots, \phi_n(x)$ are chosen so that $L_g\phi_i(x) = 0$, then equations (2.61) are modified to yield the following *normal form*

$$\begin{aligned}
\dot{z}_1 &= z_2 \\
\dot{z}_2 &= z_3 \\
&\vdots \\
\dot{z}_{\rho-1} &= z_\rho \\
\dot{z}_\rho &= a(z) + b(z)u \\
\dot{z}_{\rho+1} &= q_{\rho+1}(z) \\
&\vdots \\
\dot{z}_n &= q_n(z) \\
y &= z_1
\end{aligned} \tag{2.63}$$

Next, a constructive procedure is described to transform a single-input single-output nonlinear system into a linear controllable system via a change of coordinates in the state space, and static state feedback.

Consider a nonlinear system having at some point $x = x_o$ relative degree equal to the dimension of the state space, i.e. $\rho = n$. In this case, the change of coordinates to construct the normal form is given by the output function $h(x)$ and its first $n-1$ derivatives along $f(x)$, i.e.

$$z = \Phi(x) = \begin{pmatrix} \phi_1(x) \\ \phi_2(x) \\ \vdots \\ \phi_n(x) \end{pmatrix} = \begin{pmatrix} h(x) \\ L_f h(x) \\ \vdots \\ L_f^{n-1} h(x) \end{pmatrix} \tag{2.64}$$

No additional functions are needed to complete the transformation. In the new coordinates, the system is described by

$$\begin{aligned} \dot{z}_1 &= z_2 \\ \dot{z}_2 &= z_3 \\ &\vdots \\ \dot{z}_{n-1} &= z_n \\ \dot{z}_n &= a(z) + b(z)u \end{aligned} \tag{2.65}$$

with

$$b(z) = L_g L_f^{n-1} h\left(\Phi^{-1}(z)\right) \tag{2.66}$$

$$a(z) = L_f^n h\left(\Phi^{-1}(z)\right) \tag{2.67}$$

and $z = (z_1, z_2, \ldots, z_n)$. By construction, the function $b(z)$ is nonzero at the point $z_o = \Phi(x_o)$, and at all $z$ in a neighbourhood of $z_o$. By choosing the following state feedback control law

$$u = \frac{1}{b(z)}(-a(z) + v) \tag{2.68}$$

where $v$ is an external reference input, the resulting closed-loop system is governed by the equations

$$\begin{aligned} \dot{z}_1 &= z_2 \\ \dot{z}_2 &= z_3 \\ &\vdots \\ \dot{z}_{n-1} &= z_n \\ \dot{z}_n &= v \\ y &= z_1 \end{aligned} \tag{2.69}$$

which corresponds to a linear controllable system (with linear input-output map) in the Brunovsky controller form. Therefore any nonlinear system with relative degree $n$ at some point $x_o$, can be transformed into a system which is linear and controllable in a neighbourhood of the point $z_o = \Phi(x_o)$,

Additional feedback controls can be imposed on the linear system thus obtained. For instance

$$v = Kz \tag{2.70}$$

with

$$K = (k_0 \quad k_1 \quad \dots \quad k_{n-1}) \tag{2.71}$$

chosen to meet some given control specifications, e.g. to assign a set of eigenvalues in a specific sector or to satisfy an optimality criterion. From (2.68) the nonlinear feedback control law in the original coordinates has the form

$$u = \frac{-L_f^n h(x) + \sum_{i=0}^{n-1} k_i L_f^i h(x)}{L_g L_f^{n-1} h(x)} \tag{2.72}$$

**Example 2.7** In order to illustrate the procedure explained above, consider again the nonlinear system of Example 2.1

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{pmatrix} = \begin{pmatrix} x_2 + x_1^2 \\ x_3 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} u \tag{2.73}$$

with the output

$$y = h(x) = x_1. \tag{2.74}$$

For this system

$$\begin{array}{ll} L_g h(x) = 0, & L_f h(x) = x_2 + x_1^2 \\ L_g L_f h(x) = 0, & L_f^2 h(x) = x_3 + 2x_1(x_2 + x_1^2) \\ L_g L_f^2 h(x) = 1, & L_f^3 h(x) = (6x_1^2 + 2x_2)(x_2 + x_1^2) + 2x_1 x_3 \end{array}$$

and the system has relative degree 3 (i.e. equal to the state space dimension) for all points in $\Re^3$. The system can be transformed into a linear controllable system by means of the feedback control

$$u = -(6x_1^2 + 2x_2)(x_2 + x_1^2) - 2x_1 x_3 + v \tag{2.75}$$

and the change of coordinates

$$\begin{aligned} z_1 &= h(x) = x_1 \\ z_2 &= L_f h(x) = x_2 + x_1^2 \\ z_3 &= L_f^2 h(x) = x_3 + 2x_1(x_2 + x_1^2). \end{aligned} \tag{2.76}$$

In this case the feedback control and the change of coordinates are defined globally, i.e. are valid for all $x \in \Re^3$. In the new coordinates the system is

$$\dot{z} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} z + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} v \tag{2.77}$$

which is linear and controllable.

Note that the change of coordinates (2.76) and the control law (2.75) are, respectively, the same transformation (2.10) and control (2.12) used in Example 2.1.

### 2.2.4 State Feedback Linearization

In the previous section the key point that made it possible to change a nonlinear system into a linear and controllable one (with a linear input-output map), was the existence of an "output" function $h(x)$ for which the system had relative degree equal to $n$ (at $x_o$). The existence of such a function is not only a sufficient condition but also a necessary condition for the existence of a suitable state feedback and a change of coordinates.

More precisely, consider a nonlinear system (without output)

$$\dot{x} = f(x) + g(x)u \tag{2.78}$$

and suppose one is interested in solving the following problem: given a point $x_o$, find (if possible) a neighbourhood $R_1$ of $x_o$, a feedback

$$u = \alpha(x) + \beta(x)v \tag{2.79}$$

defined on $R_1$, and a coordinate transformation $z = \Phi(x)$ such that the corresponding closed-loop system

$$\dot{x} = f(x) + g(x)\alpha(x) + g(x)\beta(x)v \tag{2.80}$$

in the coordinates $z = \Phi(x)$ is linear and controllable, i.e. such that

$$\left[\frac{\partial \Phi}{\partial x}(f(x) + g(x)\alpha(x))\right]_{x=\Phi^{-1}(z)} = Az \tag{2.81}$$

$$\left[\frac{\partial \Phi}{\partial x}(g(x)\beta(x))\right]_{x=\Phi^{-1}(z)} = B \tag{2.82}$$

for some suitable matrix $A \in \Re^{n \times n}$ and vector $B \in \Re^n$ satisfying the controllability condition

$$\text{rank}(B \; AB \; \dots \; A^{n-1}B) = n. \tag{2.83}$$

This problem is the so-called *state feedback linearization.* The following lemma provides necessary and sufficient conditions for the existence of a solution [44].

**Lemma 2.1** *The* state feedback linearization problem *is solvable if and only if there exist a neighbourhood $R_1$ of $x_o$ and a real-valued function $\lambda(x)$ defined on $R_1$ such that the system*

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= \lambda(x) \end{aligned} \tag{2.84}$$

*has relative degree $n$ at $x_o$.*

Regarding the analysis given in Section 2.2.3, the problem of finding a function $\lambda(x)$ such that the relative degree of the system at $x_o$ is $n$, namely a function such that

$$L_g\lambda(x) = L_gL_f\lambda(x) = \ldots = L_gL_f^{n-2}\lambda(x) = 0 \tag{2.85}$$

$$L_gL_f^{n-1}\lambda(x_o) \neq 0, \tag{2.86}$$

is solvable under the conditions given in the following theorem [44].

**Theorem 2.2** *The state feedback linearization problem is solvable near a point $x_o$ (i.e. there exists an "output" function $\lambda(x)$ for which the system has relative degree $n$ at $x_o$) if and only if the following conditions are satisfied*
*(i) the matrix $(g(x_o)\ ad_fg(x_o)\ \ldots ad_f^{n-2}g(x_o)\ ad_f^{n-1}g(x_o))$ has rank $n$*
*(ii) the distribution $\mathcal{G}_{n-2} = \text{span}\left\{g, ad_fg, \ldots, ad_f^{n-2}g\right\}$ is involutive in a neighbourhood of $x_o$.*

It can be shown that condition $(i)$ is equivalent to the controllability condition (2.83) for the linear approximation of the system (2.78) [44]. In fact, the controllability of the linear approximation of the system at $x_o$ is a necessary condition for the solvability of the state feedback linearization problem.

Conditions $(i)$ and $(ii)$ together are sufficient to solve the partial differential equations (2.85)-(2.86), since they are equivalent to the condition that the distributions

$$\mathcal{G}_i = \text{span}\left\{g, ad_fg, \ldots, ad_f^ig\right\}, \quad 0 \leqslant i \leqslant n-1$$

are involutive and of constant rank $i+1$.

To illustrate the use of Theorem 2.2, consider again the linearizable system of Example 2.1, whose associated vector fields are

$$f(x) = \begin{pmatrix} x_2 + x_1^2 \\ x_3 \\ 0 \end{pmatrix}, \quad g(x) = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}. \tag{2.87}$$

It was demonstrated in Example 2.7 that this system is input-output linearizable with the output $y = x_1$. Hence, it is state feedback linearizable and should satisfy conditions $(i)$ and $(ii)$ of Theorem 2.2. In order to verify this, recall that the vector fields $ad_f g(x)$ and $ad_f^2 g(x)$, computed in Example 2.2, are

$$ad_f g(x) = \begin{pmatrix} 0 \\ -1 \\ 0 \end{pmatrix}, \quad ad_f^2 g(x) = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}. \tag{2.88}$$

Then

$$\text{rank}\begin{pmatrix} g(x) & ad_f g(x) & ad_f^2 g(x) \end{pmatrix} = \text{rank}\begin{pmatrix} 0 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{pmatrix} = 3 \tag{2.89}$$

and condition $(i)$ is satisfied. Note that in this case condition $(i)$ holds for all $x \in \Re^3$. Regarding condition $(ii)$, one must verify that the distribution

$$\mathcal{G}_1 = \text{span}\{g(x), ad_f g(x)\} \tag{2.90}$$

is involutive by checking that

$$\text{rank}\Big(g(x) \quad ad_f g(x)\Big) = \text{rank}\Big(g(x) \quad ad_f g(x) \quad [g(x), ad_f g(c)]\Big). \tag{2.91}$$

Noting that both vector fields $g(x)$ and $ad_f g(x)$ have constant entries independent of $x$, $[g(x), ad_f g(x)] = (0\ 0\ 0)^T$. Therefore, (2.90) holds and consequently condition $(ii)$ is also satisfied.

**Example 2.8** Consider the second order system

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} x_2^3 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u \tag{2.92}$$

This system is not feedback linearizable in a neighbourhood of the origin because the linear approximation about the origin is not controllable. This can be verified by checking that condition $(i)$ of Theorem 2.2 is violated. Since

$$ad_f g(x) = [f, g](x) = \frac{\partial g}{\partial x} f(x) - \frac{\partial f}{\partial x} g(x) = -\frac{\partial f}{\partial x} g(x) = \begin{pmatrix} -3x_2^2 \\ 0 \end{pmatrix} \tag{2.93}$$

the rank of the matrix

$$\Big(g(0) \quad ad_f g(0)\Big) = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \tag{2.94}$$

is 1 and therefore condition $(i)$ is not satisfied.

## 2.3 Control Design for Triangular Systems

Feedback linearizable systems are an important class of stabilizable systems because the control design problem is simplified via a linearizing coordinate transformation, which allows the use of well-established control design techniques from linear control theory. An interesting class of feedback linearizable nonlinear systems is that of systems in *triangular form*. The control design problem for these systems is simplified even more, since the linearizing coordinate transformation is easily set up, as shown in the following Corollary.

**Corollary 2.1** *Systems in the triangular form*

$$\begin{aligned} \dot{x}_1 &= x_2 + \phi_1(x_1) \\ \dot{x}_2 &= x_3 + \phi_2(x_1, x_2) \\ &\vdots \\ \dot{x}_k &= x_{k+1} + \phi_k(x_1, \ldots, x_k) \\ &\vdots \\ \dot{x}_{n-1} &= x_n + \phi_{n-1}(x_1, \ldots, x_{n-1}) \\ \dot{x}_n &= \phi_n(x_1, \ldots, x_n) + u \end{aligned} \tag{2.95}$$

*in which $\phi_1, \ldots, \phi_n$ are known smooth functions such that $\phi_i(0) = 0,\ 1 \leqslant i \leqslant n$, are globally linearizable.*

**Proof.** The proof of this corollary requires the verification of conditions ($i$) and ($ii$) of Theorem 2.2. Alternatively, a constructive proof can be carried out by directly computing the linearizing coordinate transformation and feedback control, as follows:

**Step 1.** Choose $z_1 = x_1$ as the first new state coordinate. The time derivative of $z_1$ is

$$\dot{z}_1 = x_2 + \phi_1(x_1) \tag{2.96}$$

By choosing

$$z_2 = x_2 + \phi_1(x_1) \tag{2.97}$$

(2.96) becomes

$$\dot{z}_1 = z_2. \tag{2.98}$$

**Step 2.** The time derivative of $z_2$ is

$$\dot{z}_2 = x_3 + \phi_2(x_1, x_2) + \frac{\partial \phi_1}{\partial x_1}(x_2 + \phi_1(x_1)) := x_3 + \phi_2^*(x_1, x_2) \tag{2.99}$$

By selecting

$$z_3 = x_3 + \phi_2^*(x_1, x_2) \tag{2.100}$$

(2.99) becomes

$$\dot{z}_2 = z_3. \tag{2.101}$$

**Step k** ($3 \leqslant k \leqslant n-1$). The time derivative of $z_k$ is

$$\begin{aligned} \dot{z}_k &= x_{k+1} + \phi_k(x_1, \ldots, x_k) + \sum_{i=1}^{k-1} \frac{\partial \phi_{k-1}^*}{\partial x_i}(x_{i+1} + \phi_i(x_1, \ldots, x_i)) \\ &:= x_{k+1} + \phi_k^*(x_1, \ldots, x_k) \end{aligned} \tag{2.102}$$

By selecting

$$z_{k+1} = x_{k+1} + \phi_k^*(x_1, \ldots, x_k) \tag{2.103}$$

(2.102) becomes

$$\dot{z}_k = z_{k+1} \qquad 3 \leqslant k \leqslant n-1. \tag{2.104}$$

**Step n.** The time derivative of $z_n$ is

$$\dot{z}_n = \phi_n(x_1, \ldots, x_n) + u + \sum_{i=1}^{n-1} \frac{\partial \phi_{n-1}^*}{\partial x_i}(x_{i+1} + \phi_i(x_1, \ldots, x_i)) \tag{2.105}$$

Finally, the linearizing process is completed by choosing the feedback control

$$u = -\phi_n(x_1, \ldots, x_n) - \sum_{i=1}^{n-1} \frac{\partial \phi_{n-1}^*}{\partial x_i}(x_{i+1} + \phi_i(x_1, \ldots, x_i)) + v \tag{2.106}$$

where $v$ is a reference input, or an input control to impose additional control action such as pole-placement or optimal control on the system.

□

**Example 2.9** Consider the nonlinear system in triangular form

$$\begin{aligned} \dot{x}_1 &= x_2 + ax_1^2 \\ \dot{x}_2 &= x_3 \\ \dot{x}_3 &= u \end{aligned} \tag{2.107}$$

with $a$ a known constant. The $\phi_i$ functions are

$$\phi_1 = ax_1^2, \qquad \phi_2 = 0, \qquad \phi_3 = 0. \tag{2.108}$$

Applying the above procedure, one obtains the coordinate transformation

$$\begin{aligned} z_1 &= x_1 \\ z_2 &= x_2 + ax_1^2 \\ z_3 &= x_3 + 2ax_1(x_2 + ax_1^2) \end{aligned} \tag{2.109}$$

and the linearizing feedback control law

$$u = -2ax_1x_3 - (6a^2x_1^2 + 2ax_2)(x_2 + ax_1^2) + v. \tag{2.110}$$

This recursive and systematic procedure is made possible because of the special structure of triangular systems. It allows one to choose the function $\lambda(x) = x_1$ as a linearizing output, for which the system has relative degree $n$, as mentioned in Lemma 2.1. A similar procedure can be performed for more general triangular systems, such as

$$\begin{aligned} \dot{x}_i &= \phi_i(x_1, \ldots, x_{i+1}), \qquad 1 \leqslant i \leqslant n-1 \\ \dot{x}_n &= \phi_n(x_1, \ldots, x_n) + \beta(x_1, \ldots, x_n)u \end{aligned} \tag{2.111}$$

where $\phi_i$, $1 \leqslant i \leqslant n$, and $\beta$ are smooth nonlinear functions such that $\beta(0) \neq 0$, $\phi_i(0) = 0$, $1 \leqslant i \leqslant n$, and $\partial\phi_i/\partial x_{i+1}(0) \neq 0$, $1 \leqslant i \leqslant n-1$. Systems in the form (2.111) are, in general, linearizable locally.

### 2.3.1 Static Deterministic Backstepping (SDB) Algorithm

An alternative design procedure for linearizing *deterministic* (non-adaptive) triangular systems is the backstepping approach proposed by Kanellakopoulos *et al* [48] and Krstić *et al* [64]. It is based upon the use of a quadratic Lyapunov function which is augmented at each step with an additional quadratic term for the stabilization of a subsystem of the system (2.95). At the final step the linearizing procedure is completed by obtaining a coordinate transformation $z = \Phi(x)$ and a *static* feedback control law. The algorithm to be described is identified in this thesis as the SDB algorithm to characterize the static (without derivatives of the control) nature of the control law obtained for deterministic systems of the form (2.95). This systematic procedure is carried out as follows:

**Step 1.** Choose $z_1 = x_1$ as the first new state coordinate. The first subsystem is defined as

$$\dot{z}_1 = x_2 + \phi_1(x_1) \tag{2.112}$$

which will be stabilized with respect to the quadratic Lyapunov function

$$V_1 = \frac{1}{2}z_1^2. \tag{2.113}$$

The time derivative of $V_1$ is

$$\dot{V}_1 = z_1(x_2 + \phi_1(x_1)). \tag{2.114}$$

We consider $x_2$ as a *virtual* control and can choose $x_2 = \alpha_1(x_1)$ to cancel the nonlinearities and make the bracketed term multiplying $z_1$ equal to $-c_1 z_1$, i.e.

$$x_2 = \alpha_1(x_1) := -\phi_1(x_1) - c_1 z_1, \tag{2.115}$$

where $c_1$ is a positive constant. However, since $x_2$ is not the actual control, a new coordinate is defined as the deviation of $x_2$ from its "desired value", i.e.

$$z_2 := x_2 - \alpha_1(x_1). \tag{2.116}$$

Then $\dot{z}_1$ becomes

$$\dot{z}_1 = -c_1 z_1 + z_2, \tag{2.117}$$

and

$$\dot{V}_1 = -c_1 z_1^2 + z_1 z_2. \tag{2.118}$$

The term $z_1 z_2$ in (2.118) will be compensated at the next step.

**Step 2.** Obtain the time derivative of $z_2$ and augment the previous subsystem (2.112) with

$$\dot{z}_2 = x_3 + \phi_2(x_1, x_2) - \frac{\partial \alpha_1}{\partial x_1}(x_2 + \phi_1(x_1)), \tag{2.119}$$

which will be stabilized with respect to the augmented Lyapunov function

$$V_2 = V_1 + \frac{1}{2} z_2^2. \tag{2.120}$$

The time derivative of $V_2$, considering (2.118), is

$$\dot{V}_2 = -c_1 z_1^2 + z_2 \left[ z_1 + x_3 + \phi_2(x_1, x_2) - \frac{\partial \alpha_1}{\partial x_1}(x_2 + \phi_1(x_1)) \right]. \tag{2.121}$$

Then, we consider $x_3$ as a virtual control and can choose $x_3 = \alpha_2(x_1, x_2)$ to cancel the nonlinearities and make the bracketed term multiplying $z_2$ equal to $-c_2 z_2$, i.e.

$$x_3 = \alpha_2(x_1, x_2) := -z_1 - \phi_2(x_1, x_2) + \frac{\partial \alpha_1}{\partial x_1}(x_2 + \phi_1(x_1)) - c_2 z_2, \tag{2.122}$$

where $c_2$ is a positive constant. However, since $x_3$ is not the actual control, a new coordinate is defined as the deviation of $x_3$ from its "desired value", i.e.

$$z_3 := x_3 - \alpha_2(x_1, x_2). \tag{2.123}$$

Then (2.119) becomes

$$\dot{z}_2 = -z_1 - c_2 z_2 + z_3, \tag{2.124}$$

and

$$\dot{V}_2 = -c_1 z_1^2 - c_2 z_2^2 + z_2 z_3. \tag{2.125}$$

**Step k.** Proceeding by induction, the general $k$-th step is as follows. By obtaining the time derivative of $z_k$, the previous subsystem is augmented with

$$\dot{z}_k = x_{k+1} + \phi_k(x_1, \ldots, x_k) - \sum_{i=1}^{k-1} \frac{\partial \alpha_{k-1}}{\partial x_i}(x_{i+1} + \phi_i(x_1, \ldots, x_i)), \tag{2.126}$$

which is stabilized with respect to the augmented Lyapunov function

$$V_k = V_{k-1} + \frac{1}{2} z_k^2. \tag{2.127}$$

The time derivative of $V_k$ is

$$\dot{V}_k = -\sum_{i=1}^{k-1} c_i z_i^2 + z_k \left[ z_{k-1} + x_{k+1} + \phi_k(x_1, \ldots, x_k) - \sum_{i=1}^{k-1} \frac{\partial \alpha_{k-1}}{\partial x_i}(x_{i+1} + \phi_i(x_1, \ldots, x_i)) \right]. \tag{2.128}$$

In a manner similar to previous steps, we consider $x_{k+1}$ as a virtual control and can choose $x_{k+1} = \alpha_k(x_1, \ldots, x_k)$ to cancel the nonlinearities and make the bracketed term multiplying $z_k$ equal to $-c_k z_k$, i.e.

$$x_{k+1} = \alpha_k(x_1, \ldots, x_k) := -z_{k-1} - \phi_k(x_1, \ldots, x_k) + \sum_{i=1}^{k-1} \frac{\partial \alpha_{k-1}}{\partial x_i}(x_{i+1} + \phi_i(x_1, \ldots, x_i)) - c_k z_k \tag{2.129}$$

where $c_k$ is a positive constant. However, since $x_k$ is not the actual control, a new coordinate is defined as the deviation of $x_k$ from its "desired value", i.e.

$$z_{k+1} := x_{k+1} - \alpha_k(x_1, \ldots, x_k). \tag{2.130}$$

Then $\dot{z}_k$ becomes

$$\dot{z}_k = -z_{k-1} - c_k z_k + z_{k+1}, \tag{2.131}$$

and

$$\dot{V}_k = -\sum_{i=1}^{k} c_i z_i^2 + z_k z_{k+1}. \tag{2.132}$$

**Step n.** At this final step the time derivative of $z_n$ is

$$\dot{z}_n = u + \phi_n(x) - \sum_{i=1}^{n-1} \frac{\partial \alpha_{n-1}}{\partial x_i}(x_{i+1} + \phi_i(x_1, \ldots, x_i)), \tag{2.133}$$

which will be stabilized with respect to the augmented Lyapunov function

$$V_n = V_{n-1} + \frac{1}{2} z_n^2 = \frac{1}{2} z^T z. \tag{2.134}$$

The time derivative of $V_n$ is

$$\dot{V}_n = -\sum_{i=1}^{n-1} c_i z_i^2 + z_n \left[ z_{n-1} + u + \phi_n(x) - \sum_{i=1}^{n-1} \frac{\partial \alpha_{n-1}}{\partial x_i}(x_{i+1} + \phi_i(x_1,\ldots,x_i)) \right]. \quad (2.135)$$

Now we select the linearizing control law

$$u = -z_{n-1} - \phi_n(x) + \sum_{i=1}^{n-1} \frac{\partial \alpha_{n-1}}{\partial x_i}(x_{i+1} + \phi_i(x_1,\ldots,x_i)) - c_n z_n \quad (2.136)$$

to cancel the nonlinearities and make the bracketed term multiplying $z_n$ equal to $-c_n z_n$, where $c_n$ is a positive constant. Thus

$$\dot{V}_n = -\sum_{i=1}^{n} c_i z_i^2 \quad (2.137)$$

and the closed-loop system is

$$\dot{z} = A_z z \quad (2.138)$$

with

$$A_z = \begin{bmatrix} -c_1 & 1 & 0 & \ldots & 0 & 0 \\ -1 & -c_2 & 1 & \ldots & 0 & 0 \\ 0 & -1 & -c_3 & \ldots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \ldots & -c_{n-1} & 1 \\ 0 & 0 & 0 & \ldots & -1 & -c_n \end{bmatrix} \quad (2.139)$$

The feedback control (2.136) exhibits a larger control effort than the control law (2.106) because of the incorporation of extra terms $-c_k z_k$ at intermediate steps of the backstepping algorithm to achieve linearization and stabilization simultaneously. This systematic procedure of stabilization is applicable to wide classes of linearizable nonlinear systems [64], and can be summarized as follows:

***SDB Algorithm***

*Coordinate transformation*

$$z_k = x_k - \alpha_{k-1}(x_1, \ldots, x_{k-1}) \qquad 1 \leqslant k \leqslant n \tag{2.140}$$

*with*

$$\begin{aligned}\alpha_{k-1}(x_1, \ldots, x_{k-1}) &= -z_{k-2} - \phi_{k-1}(x_1, \ldots, x_{k-1}) + \sum_{i=1}^{k-2} \frac{\partial \alpha_{k-2}}{\partial x_i}\Big(x_{i+1} + \phi_i(x_1, \ldots, x_i)\Big) \\ &\quad - c_{k-1} z_{k-1}\end{aligned} \tag{2.141}$$

*Feedback control law*

$$u = -z_{n-1} - \phi_n(x) + \sum_{i=1}^{n-1} \frac{\partial \alpha_{n-1}}{\partial x_i}(x_{i+1} + \phi_i(x_1, \ldots, x_i)) - c_n z_n \tag{2.142}$$

**Example 2.10** Consider again the nonlinear system in triangular form of Example (2.9)

$$\begin{aligned}\dot{x}_1 &= x_2 + a x_1^2 \\ \dot{x}_2 &= x_3 \\ \dot{x}_3 &= u\end{aligned} \tag{2.143}$$

with $a$ a known constant. Applying the SDB algorithm one obtains the coordinate transformation

$$\begin{aligned}z_1 &= x_1 \\ z_2 &= x_2 - \alpha_1(x_1) \\ z_3 &= x_3 - \alpha_2(x_1, x_2)\end{aligned} \tag{2.144}$$

with

$$\begin{aligned}\alpha_1(x_1) &= -a x_1^2 - c_1 x_1 \\ \alpha_2(x_1, x_2) &= -x_1 - (2a x_1 - c_1)(x_2 + a x_1^2) - c_2(x_2 + a x_1^2 + c_1 x_1)\end{aligned} \tag{2.145}$$

and the linearizing feedback control law

$$u = -z_2 + \frac{\partial \alpha_2}{\partial x_1}(x_2 + a x_1^2) + \frac{\partial \alpha_2}{\partial x_2} x_3 - c_3 z_3 \tag{2.146}$$

Computer simulations were carried out to assess the performance of the feedback control law (2.146) for stabilization. Figure 2.1 shows the asymptotic convergence of the state trajectories to the origin for the design parameters $c_1 = 3$, $c_2 = 2$ and $c_3 = 1$. Figure 2.2 shows the controlled responses of the state variables for initial conditions set at ten times larger than the initial conditions in Figure 2.1.

Adaptive versions of the SDB algorithm can be applied to nonlinear systems with uncertainties, as shown in the next section.

## 2.4 Adaptive Backstepping Control Design

Many adaptive nonlinear control schemes have been proposed for single-input linearizable systems that are linear in the unknown parameters

$$\dot{\zeta} = f_0(\zeta) + \sum_{i=1}^{p} \theta_i f_i(\zeta) + \left[ g_0(\zeta) + \sum_{i=1}^{p} \theta_i g_i(\zeta) \right] u \qquad (2.147)$$

where $\zeta \in \Re^n$ is the state, $u \in \Re$ the control, $\theta_1, \ldots, \theta_p$ a set of unknown parameters, and $f_i$, $g_i$, $0 \leqslant i \leqslant p$, smooth vector fields in a neighbourhood of the origin $\zeta = 0$ with $f_i(0) = 0$, $0 \leqslant i \leqslant p$, $g_0(0) \neq 0$.

The adaptive backstepping design approach developed by Kanellakopoulos *et al* ([48]-[50],[56, 57]) extended the class of nonlinear systems for which adaptive controllers can be systematically designed. It overcame the structural restriction associated with

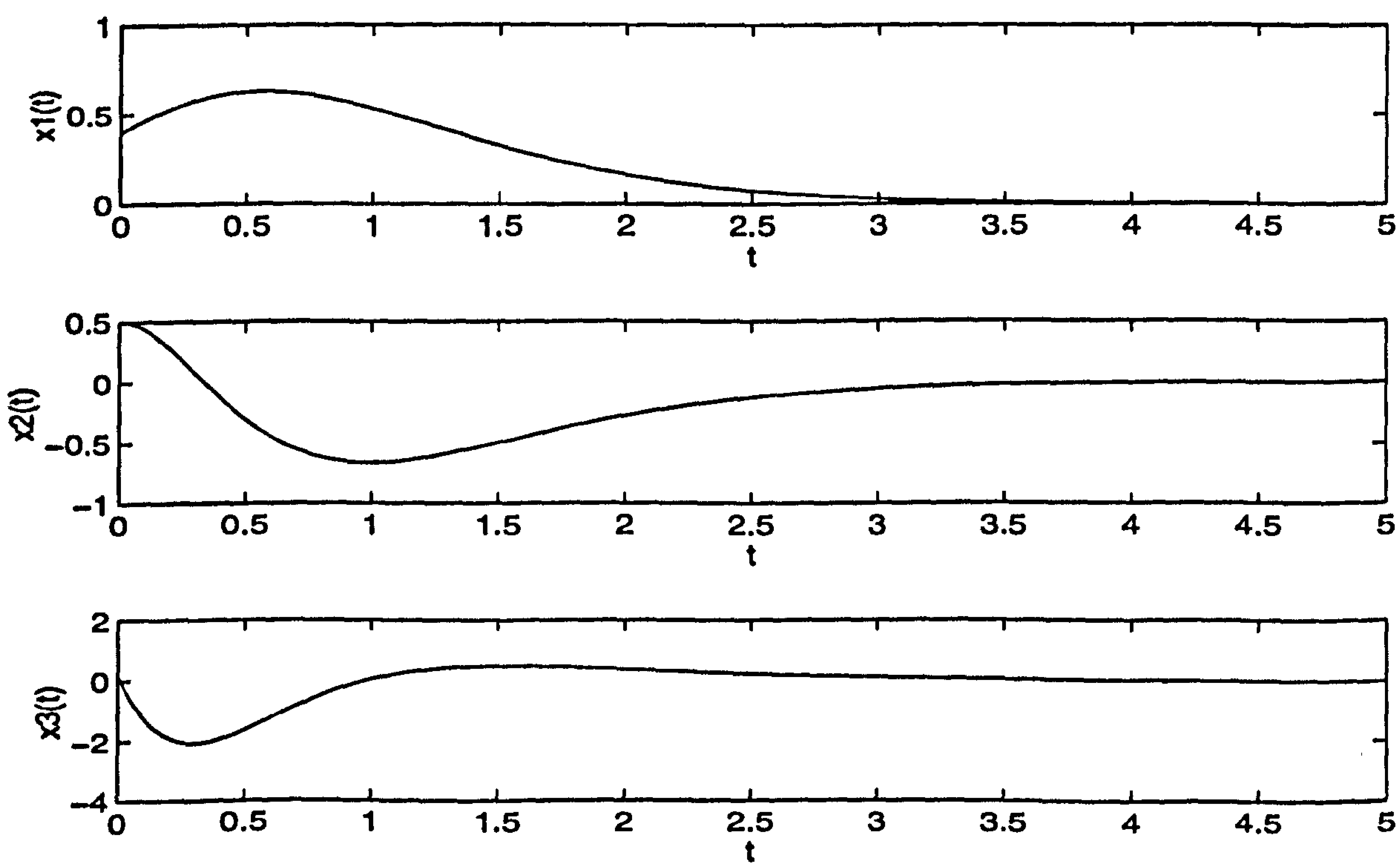


Figure 2.1: State variable responses of a triangular system regulated via SDB control

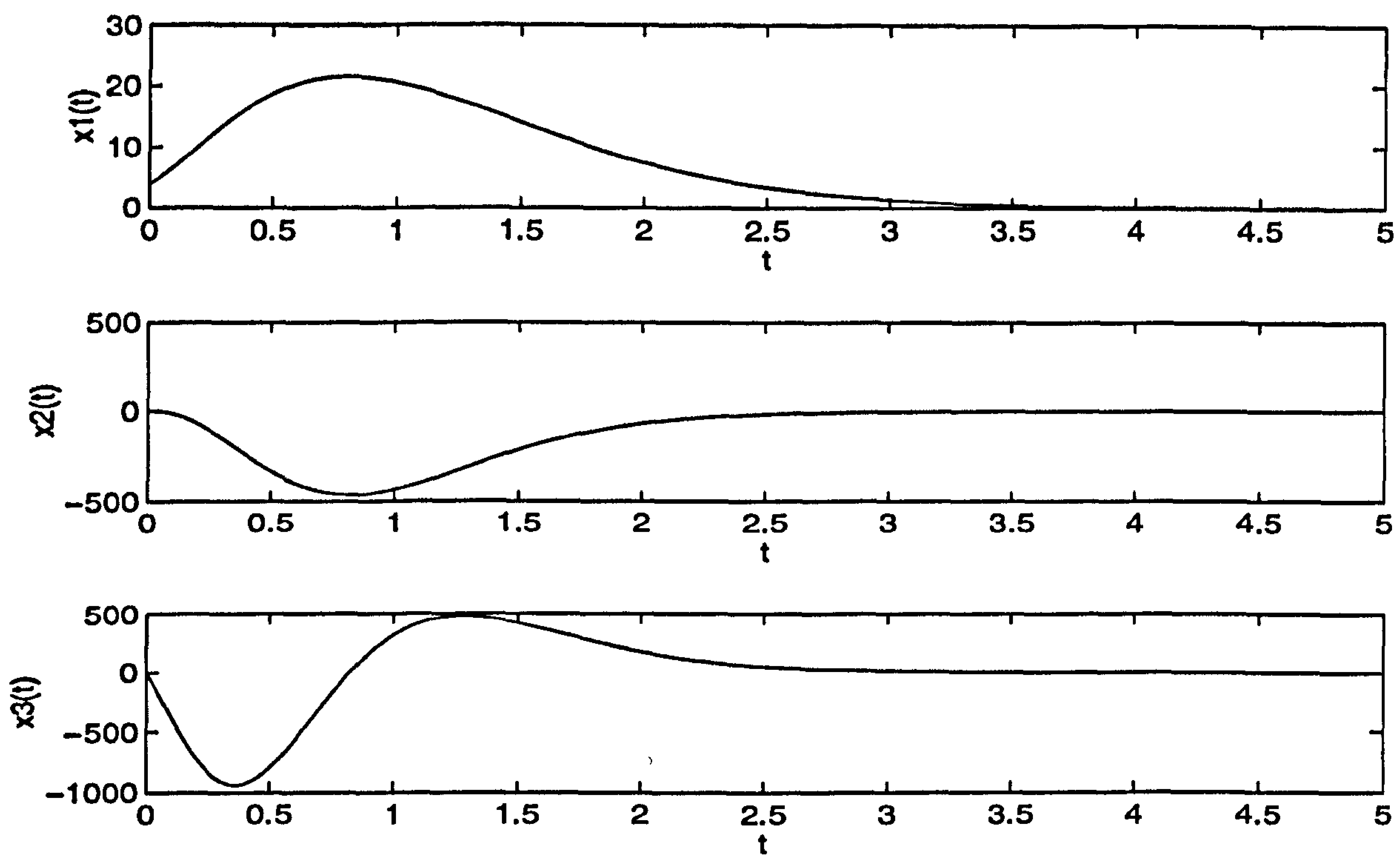


Figure 2.2: State variable responses of a triangular system regulated via SDB control for different initial conditions

the matching conditions of previous design schemes and enlarged the potentiality of Lyapunov-based designs to control nonlinear systems with unknown constant parameters.

The technique uses a step-by-step procedure which interlaces at each step a virtual control for a subsystem of the controlled plant, a linearizing change of coordinates and the construction of an update law for the unknown parameters. The stability proof is constructive and simple, since it is based on the use of a quadratic Lyapunov function which is updated at each step as in Section 2.3.1.

The geometric conditions characterizing the class of systems to which backstepping is applicable do not constrain the class of nonlinear functions present in the system. Instead it is required that the nonlinear system be transformable into the PPF form. In general, local stabilization is achieved for this class of uncertain systems, and an estimate of the region of attraction is also provided. Furthermore in the case of systems transformable into the more restrictive PSF form, backstepping guarantees *global* regulation and tracking properties.

### 2.4.1 Parametric Pure-Feedback Systems

The class of pure-feedback systems is constituted by those systems of the form (2.147) which can be transformed via a parameter-independent diffeomorphism $x = \Phi(\zeta)$ into the parametric pure-feedback systems

$$\begin{aligned} \dot{x}_1 &= x_2 + \varphi_1^T(x_1, x_2)\theta \\ \dot{x}_2 &= x_3 + \varphi_2^T(x_1, x_2, x_3)\theta \\ &\vdots \\ \dot{x}_{n-1} &= x_n + \varphi_{n-1}^T(x_1, \ldots, x_n)\theta \\ \dot{x}_n &= \varphi_0(x) + \varphi_n^T(x)\theta + \left[\beta_0(x) + \beta^T(x)\theta\right] u \end{aligned} \tag{2.148}$$

with

$$\varphi_0(0) = 0,\ \varphi_1(0) = \ldots = \varphi_n(0) = 0,\ \beta_0(0) \neq 0.$$

Necessary and sufficient conditions for the existence of such a diffeomorphism were provided in [48, 57] and are given in the following theorem.

**Theorem 2.3** *A diffeomorphism $x = \Phi(\zeta)$ with $\Phi(0) = 0$, transforming (2.147) into (2.148), exists in a neighbourhood $R_0$ of the origin if and only if the following conditions are satisfied in $R_0$:*
*(i)* Feedback Linearization condition. *The distributions*

$$\mathcal{G}_i = \text{span}\left\{g_0, ad_{f_0}g_0, \ldots, ad_{f_0}^i g_0\right\}, \quad 0 \leqslant i \leqslant n-1 \tag{2.149}$$

*are involutive and of constant* rank $i+1$
*(ii)* Pure-feedback condition

$$\begin{aligned} g_i &\in \mathcal{G}_0, \\ [X, f_i] \in \mathcal{G}_{j+1}, \quad \forall X &\in \mathcal{G}_j, \quad 0 \leqslant j \leqslant n-2,\ 1 \leqslant i \leqslant p. \end{aligned} \tag{2.150}$$

Note that condition $(i)$ is sufficient for the existence of a diffeomorphism $x = \Phi(\zeta)$ which transforms the system

$$\dot{\zeta} = f_0(\zeta) + g_0(\zeta)u, \qquad f_0(0) = 0,\ g_0(0) \neq 0 \tag{2.151}$$

into the system

$$\begin{aligned} \dot{x}_i &= x_{i+1}, \qquad 1 \leqslant i \leqslant n-1 \\ \dot{x}_n &= \varphi_0(x) + \beta_0(x)u, \end{aligned} \tag{2.152}$$

with

$$\varphi_0(0) = 0, \qquad \beta_0(0) \neq 0.$$

The pure-feedback condition guarantees the structure of the nonlinear functions $\varphi_i$ in the $x$-coordinates.

The term "parametric pure-feedback" indicates that the nonlinearities in the vector-valued function $\varphi_i$ can depend only on the state variables $x_1, \ldots, x_i$ for $1 \leqslant i \leqslant n$. An important subclass of these systems are parametric strict-feedback systems, which have desirable properties in the context of adaptive control.

### 2.4.2 Parametric Strict-Feedback Systems

Global stability properties of an adaptive system in the extended space of the states *and* parameter estimates are important for theoretical and practical reasons. Systems with these properties exhibit better robustness to disturbances and unmodelled dynamics than systems with a finite region of attraction.

In order to characterize the class of PSF systems, consider the following assumption concerning the part of the system (2.147) that does *not* contain unknown parameters

**Assumption 2.1** *There exists a* global *diffeomorphism* $x = \Phi(\zeta)$, *with* $\Phi(0) = 0$, *transforming the system*

$$\dot{\zeta} = f_0(\zeta) + g_0(\zeta)u \tag{2.153}$$

*into the system*

$$\begin{aligned} \dot{x}_i &= x_{i+1}, \qquad 1 \leqslant i \leqslant n-1 \\ \dot{x}_n &= \varphi_0(x) + \beta_0(x)u, \end{aligned} \tag{2.154}$$

*with*

$$\varphi_0(0) = 0, \quad \beta_0(x) \neq 0 \qquad \forall x \in \Re^n.$$

Necessary and sufficient conditions to transform system (2.147) into the parametric strict-feedback form were established in [48, 57] and are given in the following theorem.

**Theorem 2.4** *Under Assumption 2.1 the system (2.147) is globally diffeomorphically equivalent through* $x = \Phi(\zeta)$ *to the parametric strict-feedback system*

$$\begin{aligned} \dot{x}_1 &= x_2 + \varphi_1^T(x_1)\theta \\ \dot{x}_2 &= x_3 + \varphi_2^T(x_1, x_2)\theta \\ &\vdots \end{aligned} \tag{2.155}$$

$$\begin{aligned}\dot{x}_{n-1} &= x_n + \varphi_{n-1}^T(x_1,\ldots,x_{n-1})\theta \\ \dot{x}_n &= \varphi_0(x) + \varphi_n^T(x)\theta + \beta_0(x)u\end{aligned}$$

*if and only if the following* parametric strict-feedback condition *holds globally*

$$\begin{gathered} g_i \equiv 0 \\ [X, f_i] \in \mathcal{G}_j, \quad \forall X \in \mathcal{G}_j, \quad 0 \leqslant j \leqslant n-2,\ 1 \leqslant i \leqslant p \end{gathered} \tag{2.156}$$

*with* $\mathcal{G}_j$, $0 \leqslant j \leqslant n-1$, *as defined in (2.149).*

Note that the PSF form (2.155) is a special case of the system (2.147). If the unknown parameters are assumed to be known, (2.155) has the same form as the triangular systems (2.95), and the recursive SDB algorithm given in Section 2.3.1 can be used for global linearization (stabilization). Then, for $\theta$ unknown, if a proposed adaptive controller does not achieve global stability, this is clearly due to adaptation. Adaptive backstepping controllers preserve the global stabilization property for this class of systems.

The adaptive backstepping algorithm proposed by Kanellakopoulos *et al* [48] requires multiple estimates of the same parameter, which is impractical for high-order systems with numerous unknown parameters. This overparameterization was eliminated by Krstić *et al* [61] by the introduction of tuning functions. This method allows the strengthening of the stability and convergence properties of the resulting adaptive system.

### 2.4.3 Static Adaptive Backstepping (SAB) Algorithm

The main improvement achieved by the incorporation of tuning functions is the reduction of the adaptive controller to a minimum structure; the number of parameter estimates being equal to the number of unknown parameters. This reduction in the order of the closed-loop dynamics guarantees strong stability and convergence properties, as shown below. At each step of the recursive algorithm a tuning function is designed, as a potential update law, for compensation purposes. The final tuning function is used as the actual parameter update law. The static adaptive control law is also obtained at the final step.

It is worthwhile stressing that some authors call adaptive controllers consisting of an update law for estimation of the unknown parameters and a feedback control law *dynamical* controllers. Hereafter, we use the terms *dynamical adaptive control* for a feedback law involving the control $u$ and its derivatives, together with an update law for estimation of the unknown parameters; and *static adaptive control* for a feedback control law *without* derivatives of $u$, along with an update law.

In order to motivate the presentation of the general SAB algorithm with tuning functions proposed by Krstić *et al* [61] for the design of static adaptive controllers for PSF and PPF systems, two simple examples of backstepping design are presented first.

**Example 2.11** Consider the problem of designing an adaptive controller for the scalar system

$$\dot{x}_1 = u + \varphi_1^T(x_1)\theta \tag{2.157}$$

where $\theta = [\theta_1, \ldots, \theta_p]^T$ is an unknown constant parameter vector and the vector-valued nonlinear function $\varphi_1(x_1) = [\varphi_{11}, \ldots, \varphi_{1p}]^T$, is known and smooth. Note that this system is already in parametric strict-feedback form. Therefore, the control objective is to stabilize globally the state $x_1$ to a desired equilibrium point $X_1$. Define the error coordinate $z_1 = x_1 - X_1$, the deviation of the state variable $x_1$ from its desired equilibrium, and consider the quadratic Lyapunov function

$$V_1 = \frac{1}{2}z_1^2 + \frac{1}{2}(\theta - \hat{\theta})^T\Gamma^{-1}(\theta - \hat{\theta}) \tag{2.158}$$

where $\Gamma$ is a positive definite adaptation gain matrix, $\hat{\theta}$ is an estimate of the unknown parameters and $\theta - \hat{\theta}$ the corresponding estimate error. The time derivative of (2.158) yields

$$\dot{V}_1 = z_1\Big[u + \varphi_1^T(x_1)\theta\Big] + (\theta - \hat{\theta})^T\Gamma^{-1}\Big(-\dot{\hat{\theta}}\Big). \tag{2.159}$$

The linear parameterization of (2.157) allows one to add (and substract) $\hat{\theta}$ to (from) the bracketed term multiplying $z_1$, to obtain

$$\dot{V}_1 = z_1\Big[u + \varphi_1^T(x_1)\hat{\theta} + \varphi_1^T(x_1)(\theta - \hat{\theta})\Big] + (\theta - \hat{\theta})^T\Gamma^{-1}\Big(-\dot{\hat{\theta}}\Big). \tag{2.160}$$

By grouping terms, (2.160) can be rewritten as

$$\dot{V}_1 = z_1\Big[u + \omega_1^T(x_1)\hat{\theta}\Big] + (\theta - \hat{\theta})^T\Gamma^{-1}\Big(-\dot{\hat{\theta}} + \Gamma\omega_1(x_1)z_1\Big) \tag{2.161}$$

with the regressor vector defined as

$$\omega_1(x_1) = \varphi_1(x_1). \tag{2.162}$$

To achieve the objective of stabilizing the equilibrium $x_1 = X_1$ (or, equivalently, $z_1 = 0$), the time derivative of the Lyapunov function (2.161) must yield

$$\dot{V}_1 = -c_1 z_1^2 \tag{2.163}$$

with $c_1 > 0$ a constant design parameter. This is achieved by choosing the *update law*

$$\dot{\hat{\theta}} = \Gamma\omega_1(x_1)z_1 \tag{2.164}$$

and the *control law*

$$u = \alpha(x_1, \hat{\theta}) = -\omega_1^T(x_1)\hat{\theta} - c_1 z_1. \tag{2.165}$$

The closed-loop adaptive system is

$$\begin{aligned} \dot{z}_1 &= -c_1 z_1 + \omega_1^T(x_1)(\theta - \hat{\theta}) \\ \dot{\hat{\theta}} &= \Gamma\omega_1(x_1)z_1. \end{aligned} \tag{2.166}$$

The unknown parameters in system (2.157) are "matched". Nevertheless, it is not because of this condition that the adaptive control design is possible. This can be shown by applying the same procedure to the following system, satisfying the *extended matching condition* and obtained by augmenting system (2.157) with an integrator.

**Example 2.12** Consider the problem of designing an adaptive controller for the system

$$\begin{aligned} \dot{x}_1 &= x_2 + \varphi_1^T(x_1)\theta \\ \dot{x}_2 &= u \end{aligned} \tag{2.167}$$

to stabilize the state variable $x_1$ to the desired equilibrium $X_1$.

Note that the problem of finding an adaptive control to stabilize $x_1$ to the equilibrium $X_1$ for the subsystem

$$\dot{x}_1 = x_2 + \varphi_1^T(x_1)\theta \tag{2.168}$$

is exactly the same control design problem as solved in Example 2.11, if $x_2$ was the control input. This suggests the possibility of taking advantage of the design carried out in Example 2.11 by regarding $x_2$ as a *virtual control* to stabilize (2.168) with the Lyapunov function (2.159). However, since $x_2$ is not a control, one must go back to the design in Example 2.11 and consider the fact that $x_2$ is actually a state variable. Thus, the time derivative of (2.159) yields

$$\dot{V}_1 = z_1\Big[x_2 + \omega_1^T(x_1)\hat{\theta}\Big] + (\theta - \hat{\theta})^T\Gamma^{-1}\Big(-\dot{\hat{\theta}} + \tau_1\Big) \tag{2.169}$$

with

$$\tau_1 = \Gamma\omega_1(x_1)z_1 \tag{2.170}$$

being considered as a *tuning function*, instead of the update law because of the presence of the additional state variable. The stabilization of (2.168) would be readily possible, if $\tau_1$ were the update law, $x_2$ the actual control, and the relation

$$x_2 = \alpha(x_1, \hat{\theta}) = -\omega_1^T(x_1)\hat{\theta} - c_1 z_1 \tag{2.171}$$

were satisfied. However, since (2.171) is in general not valid, the deviation of $x_2$ from its desired value is considered as a new error coordinate

$$z_2 := x_2 - \alpha(x_1, \hat{\theta}). \tag{2.172}$$

As a consequence, the objective of stabilizing $x_1$ to the equilibrium $X_1$ is equivalent to stabilizing $(z_1, z_2)^T$ to the origin. This new error coordinate $z_2$ allows one to obtain the time derivative of (2.158)

$$\dot{V}_1 = -c_1 z_1^2 + z_1 z_2 + (\theta - \hat{\theta})^T \Gamma^{-1}\Big( -\dot{\hat{\theta}} + \tau_1\Big) \tag{2.173}$$

and

$$\dot{z}_1 = -c_1 z_1 + z_2 + \omega_1^T(x_1)(\theta - \hat{\theta}). \tag{2.174}$$

Then

$$\dot{z}_2 = u - \frac{\partial \alpha_1}{\partial x_1}\left(x_2 + \varphi_1^T(x_1)\theta\right) - \frac{\partial \alpha_1}{\partial \hat{\theta}}\dot{\hat{\theta}}, \tag{2.175}$$

and augmenting the Lyapunov function as

$$V_2 = V_1 + \frac{1}{2}z_2^2, \tag{2.176}$$

the time derivative of the augmented Lyapunov function yields

$$\begin{aligned} \dot{V}_2 &= -c_1 z_1^2 + z_2\left[z_1 + u - \frac{\partial \alpha_1}{\partial x_1}\left(x_2 + \varphi_1^T(x_1)\theta\right) - \frac{\partial \alpha_1}{\partial \hat{\theta}}\dot{\hat{\theta}}\right] \\ &\quad + (\theta - \hat{\theta})^T \Gamma^{-1}\left(-\dot{\hat{\theta}} + \tau_1\right) \end{aligned} \tag{2.177}$$

By adding (and substracting) the parameter estimate $\hat{\theta}$ to (from) the bracketed term multiplying $z_2$, (2.177) can be rewritten as

$$\begin{aligned} \dot{V}_2 &= -c_1 z_1^2 + z_2\left[z_1 + u - \frac{\partial \alpha_1}{\partial x_1}x_2 + \omega_2^T\hat{\theta} - \frac{\partial \alpha_1}{\partial \hat{\theta}}\dot{\hat{\theta}}\right] \\ &\quad + (\theta - \hat{\theta})^T \Gamma^{-1}\left(-\dot{\hat{\theta}} + \tau_1 + \Gamma\omega_2 z_2\right) \end{aligned} \tag{2.178}$$

with the regressor vector defined as

$$\omega_2 = -\frac{\partial \alpha_1}{\partial x_1}\varphi_1(x_1) \tag{2.179}$$

The control objective of global stabilization is achieved if (2.178) has the form

$$\dot{V}_2 = -c_1 z_1^2 - c_2 z_2^2 \tag{2.180}$$

with $c_2 > 0$ a design parameter. This is achieved by choosing the *update law*

$$\dot{\hat{\theta}} := \tau_2 = \tau_1 + \Gamma\omega_2 z_2 = \Gamma(\omega_1 z_1 + \omega_2 z_2) \tag{2.181}$$

and the *control law*

$$u = -z_1 + \frac{\partial \alpha_1}{\partial x_1} x_2 - \omega_2^T \hat{\theta} + \frac{\partial \alpha_1}{\partial \hat{\theta}} \tau_2 - c_2 z_2. \tag{2.182}$$

The adaptive closed-loop system is

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} -c_1 & 1 \\ -1 & -c_2 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \begin{bmatrix} \omega_1^T \\ \omega_2^T \end{bmatrix} (\theta - \hat{\theta}) \tag{2.183}$$

$$\dot{\hat{\theta}} = \Gamma W(z, \hat{\theta}) z \tag{2.184}$$

where the regressor matrix $W$ is composed of the regressor vectors

$$W(z, \hat{\theta}) = [\omega_1 \quad \omega_2]. \tag{2.185}$$

The general SAB algorithm (backstepping with tuning functions) proposed by Krstić *et al* [61] for the adaptive regulation of the "output" $y = x_1$ of a PSF system to a desired set point $y_r$, can be summarized as follows:

***SAB Algorithm***

***Coordinate transformation***

$$\begin{aligned} z_1 &= x_1 - y_r \\ z_k &= x_k - \alpha_{k-1}(x_1, \ldots, x_{k-1}, \hat{\theta}) \qquad 2 \leqslant k \leqslant n \end{aligned} \tag{2.186}$$

*with*

$$\begin{aligned} \alpha_k(x_1, \ldots, x_k, \hat{\theta}) &:= -z_{k-1} + \sum_{i=1}^{k-1} \frac{\partial \alpha_{k-1}}{\partial x_i} x_{i+1} + \frac{\partial \alpha_{k-1}}{\partial \hat{\theta}} \tau_k - \omega_k^T \hat{\theta} \\ &\quad + \left( \sum_{i=1}^{k-2} z_{i+1} \frac{\partial \alpha_i}{\partial \hat{\theta}} \right) \Gamma \omega_k - c_k z_k \end{aligned} \tag{2.187}$$

$$\omega_k(x_1, \ldots, x_k, \hat{\theta}) := \varphi_k - \sum_{i=1}^{k-1} \frac{\partial \alpha_{k-1}}{\partial x_i} \varphi_i \tag{2.188}$$

$$\tau_k(x_1, \ldots, x_k, \hat{\theta}) := \tau_{k-1} + \Gamma \omega_k z_k = \Gamma \sum_{i=1}^{k} \omega_i z_i \tag{2.189}$$

*Parameter update law*

$$\dot{\hat{\theta}} = \tau_n(x, \hat{\theta}) = \Gamma W z \tag{2.190}$$

*with*

$$W(z, \hat{\theta}) = [\omega_1, \ \ldots, \ \omega_n]; \qquad z = (z_1, \ldots, z_n)^T \tag{2.191}$$

*Adaptive control law*

$$\begin{aligned} u &= \frac{1}{\beta_0(x)} \Bigg[ - z_{n-1} - \varphi_0(x) + \sum_{i=1}^{n-1} \frac{\partial \alpha_{n-1}}{\partial x_i} x_{i+1} - \omega_n^T \hat{\theta} + \frac{\partial \alpha_{n-1}}{\partial \hat{\theta}} \tau_n \\ &\quad + \left( \sum_{i=1}^{n-2} z_{i+1} \frac{\partial \alpha_i}{\partial \hat{\theta}} \right) \Gamma \omega_n - c_n z_n \Bigg] \end{aligned} \tag{2.192}$$

The steps leading to this general algorithm are described below. The design procedure first stabilizes the first equation of (2.155), by considering $x_2$ as a virtual control and designing the first tuning function. At each subsequent step the controlled system is augmented by one equation. Thus, at the $k$-th step, a subsystem of order $k$ is stabilized with respect to a quadratic Lyapunov function by selecting a stabilizing function $\alpha_k$ and a tuning function $\tau_k$. At the final step the resulting tuning function is used as the actual update law and the adaptive feedback control is formulated. The design parameters $c_i$ below are positive.

**Step 1.** Define the error variable

$$z_1 := x_1 - y_r \tag{2.193}$$

whose time derivative is

$$\dot{z}_1 = x_2 + \varphi_1^T(x_1)\theta. \tag{2.194}$$

By adding and subtracting the estimate $\hat{\theta}$ of $\theta$ in (2.194), $\dot{z}_1$ can be rewritten as

$$\dot{z}_1 = x_2 + \varphi_1^T(x_1)\hat{\theta} + \varphi_1^T(x_1)\tilde{\theta} \tag{2.195}$$

with $\tilde{\theta} := \theta - \hat{\theta}$ defined as the estimate error. The subsystem (2.195) can be stabilized with respect to the Lyapunov function

$$V_1(z_1, \tilde{\theta}) = \frac{1}{2}z_1^2 + \frac{1}{2}\tilde{\theta}^T\Gamma^{-1}\tilde{\theta}, \tag{2.196}$$

with $\Gamma = \Gamma^T > 0$ an adaptation gain matrix. The time derivative of $V_1$ is

$$\dot{V}_1 = z_1\left[x_2 + \omega_1^T\hat{\theta}\right] + \tilde{\theta}^T\Gamma^{-1}\left(-\dot{\hat{\theta}} + \Gamma\omega_1 z_1\right), \tag{2.197}$$

with the regressor vector

$$\omega_1(x_1) := \varphi_1(x_1). \tag{2.198}$$

One can eliminate the estimate error $\tilde{\theta}$ from $\dot{V}_1$ with the update law $\dot{\hat{\theta}} = \tau_1$ defined as

$$\tau_1(x_1) := \Gamma\omega_1 z_1. \tag{2.199}$$

Then, treating $x_2$ as the control, one would have $\dot{V}_1 = -c_1 z_1^2$ with the virtual control $x_2 = \alpha_1$ defined as

$$\alpha_1(x_1, \hat{\theta}) := -\omega_1^T\hat{\theta} - c_1 z_1. \tag{2.200}$$

However, since $x_2$ is not the control, $x_2 \neq \alpha_1$ and the second error variable

$$z_2 := x_2 - \alpha_1 = x_2 + \omega_1^T\hat{\theta} + c_1 z_1, \tag{2.201}$$

is defined as the deviation of the state variable $x_2$ from its desired trajectory. Thus the closed-loop form of $\dot{z}_1$ becomes

$$\dot{z}_1 = -c_1 z_1 + z_2 + \omega_1^T\tilde{\theta} \tag{2.202}$$

and, since $\tau_1$ is not considered as an update law but the first tuning function, the presence of $\tilde{\theta}$ is tolerated in

$$\dot{V}_1 = -c_1 z_1^2 + z_1 z_2 + \tilde{\theta}^T\Gamma^{-1}\left(-\dot{\hat{\theta}} + \tau_1\right) \tag{2.203}$$

The second term in (2.203) will be cancelled at the next step.

**Step 2.** Consider now the time derivative of the error variable $z_2$

$$\dot{z}_2 = x_3 - \frac{\partial \alpha_1}{\partial x_1} x_2 + \omega_2^T(x_1, x_2, \hat{\theta})\theta - \frac{\partial \alpha_1}{\partial \hat{\theta}}\dot{\hat{\theta}} \tag{2.204}$$

with the regressor vector

$$\omega_2(x_1, x_2, \hat{\theta}) := \varphi_2 - \frac{\partial \alpha_1}{\partial x_1}\varphi_1 \tag{2.205}$$

By adding and subtracting $\hat{\theta}$ in (2.204), $\dot{z}_2$ can be rewritten as

$$\dot{z}_2 = x_3 - \frac{\partial \alpha_1}{\partial x_1} x_2 + \omega_2^T(x_1, x_2, \hat{\theta})\hat{\theta} - \frac{\partial \alpha_1}{\partial \hat{\theta}}\dot{\hat{\theta}} + \omega_2^T(x_1, x_2, \hat{\theta})\tilde{\theta} \tag{2.206}$$

which can be stabilized with respect to the augmented Lyapunov function

$$V_2 = V_1 + \frac{1}{2}z_2^2 = \frac{1}{2}z_1^2 + \frac{1}{2}z_2^2 + \frac{1}{2}\tilde{\theta}^T\Gamma^{-1}\tilde{\theta}. \tag{2.207}$$

The time derivative of $V_2$ is

$$\begin{aligned}\dot{V}_2 = -c_1 z_1^2 + z_2 &\left[z_1 + x_3 - \frac{\partial \alpha_1}{\partial x_1}x_2 + \omega_2^T\hat{\theta} - \frac{\partial \alpha_1}{\partial \hat{\theta}}\dot{\hat{\theta}}\right] \\ &+\tilde{\theta}^T\Gamma^{-1}\Big(-\dot{\hat{\theta}} + \tau_1 + \Gamma\omega_2 z_2\Big).\end{aligned} \tag{2.208}$$

The estimate error $\tilde{\theta}$ can be eliminated from $\dot{V}_2$ with the update law $\dot{\hat{\theta}} = \tau_2$ defined as

$$\tau_2(x_1, x_2, \hat{\theta}) := \tau_1 + \Gamma\omega_2 z_2 = \Gamma(z_1\omega_1 + \omega_2 z_2). \tag{2.209}$$

If additionally $x_3$ were the control, one would make $\dot{V}_2 = -c_1 z_1^2 - c_2 z_2^2$ with the virtual control $x_3 = \alpha_2$ defined as

$$\alpha_2(x_1, x_2, \hat{\theta}) := -z_1 + \frac{\partial \alpha_1}{\partial x_1}x_2 - \omega_2^T\hat{\theta} + \frac{\partial \alpha_1}{\partial \hat{\theta}}\tau_2 - c_2 z_2. \tag{2.210}$$

Since $x_3$ is not the control, $x_3 \neq \alpha_2$ and the third error variable

$$z_3 := x_3 - \alpha_2 = x_3 + z_1 - \frac{\partial \alpha_1}{\partial x_1}x_2 + \omega_2^T\hat{\theta} - \frac{\partial \alpha_1}{\partial \hat{\theta}}\tau_2 + c_2 z_2 \tag{2.211}$$

is defined as the deviation of the state variable $x_3$ from its desired trajectory. Thus the closed-loop form of $\dot{z}_2$ is

$$\dot{z}_2 = -z_1 - c_2 z_2 + z_3 + \omega_2^T\tilde{\theta} - \frac{\partial \alpha_1}{\partial \hat{\theta}}\Big(\dot{\hat{\theta}} - \tau_2\Big) \tag{2.212}$$

and, since $\tau_2$ is considered the second tuning function, $\dot{V}_2$ becomes

$$\dot{V}_2 = -c_1 z_1^2 - c_2 z_2^2 + z_2 z_3 - z_2\frac{\partial \alpha_1}{\partial \hat{\theta}}\Big(\dot{\hat{\theta}} - \tau_2\Big) + \tilde{\theta}^T\Gamma^{-1}\Big(-\dot{\hat{\theta}} + \tau_2\Big) \tag{2.213}$$

The third term in (2.213) will be cancelled at the next step.

**Step 3.** Considering (2.211), the time derivative of the error variable $z_3$ is

$$\dot{z}_3 = x_4 - \frac{\partial \alpha_2}{\partial x_1}x_2 - \frac{\partial \alpha_2}{\partial x_2}x_3 + \omega_3^T(x_1, x_2, x_3, \hat{\theta})\theta - \frac{\partial \alpha_2}{\partial \hat{\theta}}\dot{\hat{\theta}} \tag{2.214}$$

with the regressor vector

$$\omega_3(x_1, x_2, x_3, \hat{\theta}) := \varphi_3 - \frac{\partial \alpha_2}{\partial x_1}\varphi_1 - \frac{\partial \alpha_2}{\partial x_2}\varphi_2. \tag{2.215}$$

By adding and subtracting $\hat{\theta}$ in (2.204), $\dot{z}_3$ can be rewritten as

$$\dot{z}_3 = x_4 - \frac{\partial \alpha_2}{\partial x_1}x_2 - \frac{\partial \alpha_2}{\partial x_2}x_3 + \omega_3^T(x_1, x_2, x_3, \hat{\theta})\hat{\theta} - \frac{\partial \alpha_2}{\partial \hat{\theta}}\dot{\hat{\theta}} + \omega_3^T(x_1, x_2, x_3, \hat{\theta})\tilde{\theta}, \tag{2.216}$$

which can be stabilized with respect to the augmented Lyapunov function

$$V_3 = V_2 + \frac{1}{2}z_3^2 = \frac{1}{2}z_1^2 + \frac{1}{2}z_2^2 + \frac{1}{2}z_3^2 + \frac{1}{2}\tilde{\theta}^T\Gamma^{-1}\tilde{\theta}. \tag{2.217}$$

The time derivative of $V_3$ is

$$\begin{aligned} \dot{V}_3 = & -c_1z_1^2 - c_2z_2^2 - z_2\frac{\partial \alpha_1}{\partial \hat{\theta}}\left(\dot{\hat{\theta}} - \tau_2\right) \\ & + z_3\left[z_2 + x_4 - \frac{\partial \alpha_2}{\partial x_1}x_2 - \frac{\partial \alpha_2}{\partial x_2}x_3 + \omega_3^T\hat{\theta} - \frac{\partial \alpha_2}{\partial \hat{\theta}}\dot{\hat{\theta}}\right] \\ & + \tilde{\theta}^T\Gamma^{-1}\left( - \dot{\hat{\theta}} + \tau_2 + \Gamma\omega_3z_3\right). \end{aligned} \tag{2.218}$$

One can eliminate the estimate error $\tilde{\theta}$ from $\dot{V}_3$ with the tuning function $\dot{\hat{\theta}} = \tau_3$ defined as

$$\tau_3(x_1, x_2, x_3, \hat{\theta}) := \tau_2 + \Gamma\omega_3z_3 = \Gamma(\omega_1z_1 + \omega_2z_2 + \omega_3z_3). \tag{2.219}$$

Notice that choosing a virtual control law to make the bracketed term multiplying $z_3$ equal to $-c_3z_3$ does not cancel the third term in (2.218). Nevertheless, noting that

$$\dot{\hat{\theta}} - \tau_2 = \dot{\hat{\theta}} - \tau_3 + \tau_3 - \tau_2 = \dot{\hat{\theta}} - \tau_3 + \Gamma\omega_3z_3, \tag{2.220}$$

one can rewrite (2.218) as

$$\begin{aligned} \dot{V}_3 = & -c_1z_1^2 - c_2z_2^2 - z_2\frac{\partial \alpha_1}{\partial \hat{\theta}}\left(\dot{\hat{\theta}} - \tau_3\right) + \tilde{\theta}^T\Gamma^{-1}\left( - \dot{\hat{\theta}} + \tau_3\right) \\ & + z_3\left[z_2 + x_4 - \frac{\partial \alpha_2}{\partial x_1}x_2 - \frac{\partial \alpha_2}{\partial x_2}x_3 + \omega_3^T\hat{\theta} - \frac{\partial \alpha_2}{\partial \hat{\theta}}\dot{\hat{\theta}} - z_2\frac{\partial \alpha_1}{\partial \hat{\theta}}\Gamma\omega_3\right] \end{aligned} \tag{2.221}$$

Thus the tuning function $\dot{\hat{\theta}} = \tau_3$ would also eliminate the third term in (2.221) and, if $x_3$ were the control, one would make $\dot{V}_3 = -c_1z_1^2 - c_2z_2^2 - c_3z_3^2$ with the virtual control $x_4 = \alpha_3$ defined as

$$\alpha_3(x_1, x_2, x_3, \hat{\theta}) := -z_2 + \frac{\partial \alpha_2}{\partial x_1}x_2 + \frac{\partial \alpha_2}{\partial x_2}x_3 - \omega_3^T\hat{\theta} + \frac{\partial \alpha_2}{\partial \hat{\theta}}\tau_3 + z_2\frac{\partial \alpha_1}{\partial \hat{\theta}}\Gamma\omega_3 - c_3z_3. \tag{2.222}$$

Since $x_4$ is not the control, $x_4 \neq \alpha_3$ and the third error variable

$$z_4 := x_4 - \alpha_3 = x_4 + z_2 - \frac{\partial \alpha_2}{\partial x_1} x_2 - \frac{\partial \alpha_2}{\partial x_2} x_3 + \omega_3^T \hat{\theta} - \frac{\partial \alpha_2}{\partial \hat{\theta}} \tau_3 - z_2 \frac{\partial \alpha_1}{\partial \hat{\theta}} \Gamma \omega_3 + c_3 z_3 \quad (2.223)$$

is defined as the deviation of the state variable $x_4$ from its desired trajectory. Thus, the closed-loop form of $\dot{z}_3$ is

$$\dot{z}_3 = -z_2 - c_3 z_3 + z_4 + \omega_3^T \tilde{\theta} - \frac{\partial \alpha_2}{\partial \hat{\theta}} \left( \dot{\hat{\theta}} - \tau_3 \right) + z_2 \frac{\partial \alpha_1}{\partial \hat{\theta}} \Gamma \omega_3 \quad (2.224)$$

and, since $\tau_3$ is considered the third tuning function, $\dot{V}_3$ becomes

$$\dot{V}_3 = -\sum_{i=1}^{3} c_i z_i^2 + z_3 z_4 - \left( z_2 \frac{\partial \alpha_1}{\partial \hat{\theta}} + z_3 \frac{\partial \alpha_2}{\partial \hat{\theta}} \right) \left( \dot{\hat{\theta}} - \tau_3 \right) + \tilde{\theta}^T \Gamma^{-1} \left( -\dot{\hat{\theta}} + \tau_3 \right). \quad (2.225)$$

**Step k.** Proceeding by induction, the time derivative of the error variable $z_k$ is

$$\dot{z}_k = x_{k+1} - \sum_{i=1}^{k-1} \frac{\partial \alpha_{k-1}}{\partial x_i} x_{i+1} + \omega_k^T (x_1, \ldots, x_k, \hat{\theta}) \theta - \frac{\partial \alpha_{k-1}}{\partial \hat{\theta}} \dot{\hat{\theta}} \quad (2.226)$$

with the regressor vector

$$\omega_k(x_1, \ldots, x_k, \hat{\theta}) := \varphi_k - \sum_{i=1}^{k-1} \frac{\partial \alpha_{k-1}}{\partial x_i} \varphi_i. \quad (2.227)$$

By adding and subtracting $\hat{\theta}$ in (2.226), $\dot{z}_k$ can be rewritten as

$$\begin{aligned} \dot{z}_k &= x_{k+1} - \sum_{i=1}^{k-1} \frac{\partial \alpha_{k-1}}{\partial x_i} x_{i+1} + \omega_k^T (x_1, \ldots, x_k, \hat{\theta}) \hat{\theta} - \frac{\partial \alpha_{k-1}}{\partial \hat{\theta}} \dot{\hat{\theta}} \\ &\quad + \omega_k^T (x_1, \ldots, x_k, \hat{\theta}) \tilde{\theta} \end{aligned} \quad (2.228)$$

which can be stabilized with respect to the augmented Lyapunov function

$$V_k = V_{k-1} + \frac{1}{2} z_k^2. \quad (2.229)$$

The time derivative of $V_k$ is

$$\begin{aligned} \dot{V}_k &= -\sum_{i=1}^{k-1} c_i z_i^2 - \left( \sum_{i=1}^{k-2} z_{i+1} \frac{\partial \alpha_i}{\partial \hat{\theta}} \right) \left( \dot{\hat{\theta}} - \tau_{k-1} \right) \\ &\quad + z_k \left[ z_{k-1} + x_{k+1} - \sum_{i=1}^{k-1} \frac{\partial \alpha_{k-1}}{\partial x_i} x_{i+1} + \omega_k^T \hat{\theta} - \frac{\partial \alpha_{k-1}}{\partial \hat{\theta}} \dot{\hat{\theta}} \right] \\ &\quad + \tilde{\theta}^T \Gamma^{-1} \left( -\dot{\hat{\theta}} + \tau_{k-1} + \Gamma \omega_k z_k \right) \end{aligned} \quad (2.230)$$

One can eliminate $\tilde{\theta}$ from $\dot{V}_k$ with the tuning function $\dot{\hat{\theta}} = \tau_k$ defined as

$$\tau_k(x_1,\ldots,x_k,\hat{\theta}) := \tau_{k-1} + \Gamma\omega_k z_k = \Gamma\sum_{i=1}^{k}\omega_i z_i. \tag{2.231}$$

Furthermore noting that

$$\dot{\hat{\theta}} - \tau_{k-1} = \dot{\hat{\theta}} - \tau_k + \tau_k - \tau_{k-1} = \dot{\hat{\theta}} - \tau_k + \Gamma\omega_k z_k \tag{2.232}$$

$\dot{V}_k$ can be rewritten as

$$\begin{aligned}\dot{V}_k = &-\sum_{i=1}^{k-1} c_i z_i^2 - \left(\sum_{i=1}^{k-2} z_{i+1}\frac{\partial\alpha_i}{\partial\hat{\theta}}\right)\left(\dot{\hat{\theta}} - \tau_k\right) + \tilde{\theta}^T\Gamma^{-1}\left(-\dot{\hat{\theta}} + \tau_k\right)\\ &+z_k\Bigg[z_{k-1} + x_{k+1} - \sum_{i=1}^{k-1}\frac{\partial\alpha_{k-1}}{\partial x_i}x_{i+1} - \frac{\partial\alpha_{k-1}}{\partial\hat{\theta}}\dot{\hat{\theta}}\\ &+\omega_k^T\hat{\theta} - \left(\sum_{i=1}^{k-2} z_{i+1}\frac{\partial\alpha_i}{\partial\hat{\theta}}\right)\Gamma\omega_k\Bigg]\end{aligned} \tag{2.233}$$

Then, if $x_{k+1}$ were the control, one would make $\dot{V}_k = -\sum_{i=1}^{k} c_i z_i^2$ with the virtual control $x_{k+1} = \alpha_k$ defined as

$$\begin{aligned}\alpha_k(x_1,\ldots,x_k,\hat{\theta}) \;:=\; & -z_{k-1} + \sum_{i=1}^{k-1}\frac{\partial\alpha_{k-1}}{\partial x_i}x_{i+1} + \frac{\partial\alpha_{k-1}}{\partial\hat{\theta}}\tau_k - \omega_k^T\hat{\theta}\\ &+\left(\sum_{i=1}^{k-2} z_{i+1}\frac{\partial\alpha_i}{\partial\hat{\theta}}\right)\Gamma\omega_k - c_k z_k.\end{aligned} \tag{2.234}$$

Since $x_{k+1}$ is not the control, $x_{k+1} \neq \alpha_k$ in general, and the $k$-th error variable

$$\begin{aligned}z_{k+1} \;:=\; & x_{k+1} - \alpha_k\\ =\; & x_{k+1} + z_{k-1} - \sum_{i=1}^{k-1}\frac{\partial\alpha_{k-1}}{\partial x_i}x_{i+1} - \frac{\partial\alpha_{k-1}}{\partial\hat{\theta}}\tau_k + \omega_k^T\hat{\theta}\\ &-\left(\sum_{i=1}^{k-2} z_{i+1}\frac{\partial\alpha_i}{\partial\hat{\theta}}\right)\Gamma\omega_k + c_k z_k\end{aligned} \tag{2.235}$$

is defined as the deviation of the state variable $x_{k+1}$ from its desired trajectory. Thus the closed-loop form of $\dot{z}_k$ is

$$\dot{z}_k = -z_{k-1} - c_k z_k + z_{k+1} + \omega_k^T\tilde{\theta} - \frac{\partial\alpha_{k-1}}{\partial\hat{\theta}}\left(\dot{\hat{\theta}} - \tau_k\right) + \left(\sum_{i=1}^{k-2} z_{i+1}\frac{\partial\alpha_i}{\partial\hat{\theta}}\right)\Gamma\omega_k \tag{2.236}$$

and, since the $k$-th tuning function $\tau_k$ is considered instead of an update law, the time derivative of $V_k$ becomes

$$\dot{V}_k = -\sum_{i=1}^{k} c_i z_i^2 + z_k z_{k+1} - \left(\sum_{i=1}^{k-2} z_{i+1}\frac{\partial\alpha_i}{\partial\hat{\theta}}\right)\left(\dot{\hat{\theta}} - \tau_k\right) + \tilde{\theta}^T\Gamma^{-1}\left(-\dot{\hat{\theta}} + \tau_k\right) \tag{2.237}$$

**Step n.** Using the definition $z_n := x_n - \alpha_{n-1}$, the time derivative of the error variable $z_n$ is

$$\dot{z}_n = \varphi_0(x) + \beta_0(x)u - \sum_{i=1}^{n-1} \frac{\partial \alpha_{n-1}}{\partial x_i} x_{i+1} + \omega_n^T(x,\hat{\theta})\theta - \frac{\partial \alpha_{n-1}}{\partial \hat{\theta}}\dot{\hat{\theta}} \tag{2.238}$$

with the last regressor vector defined as

$$\omega_n(x,\hat{\theta}) := \varphi_n - \sum_{i=1}^{n-1} \frac{\partial \alpha_{n-1}}{\partial x_i}\varphi_i \tag{2.239}$$

By adding and subtracting $\hat{\theta}$ in (2.238), $\dot{z}_n$ can be rewritten as

$$\dot{z}_n = \varphi_0(x) + \beta_0(x)u - \sum_{i=1}^{n-1} \frac{\partial \alpha_{n-1}}{\partial x_i} x_{i+1} + \omega_n^T(x,\hat{\theta})\hat{\theta} - \frac{\partial \alpha_{n-1}}{\partial \hat{\theta}}\dot{\hat{\theta}} + \omega_n^T(x,\hat{\theta})\tilde{\theta}, \tag{2.240}$$

which can be stabilized with respect to the augmented Lyapunov function

$$V_n = V_{n-1} + \frac{1}{2}z_n^2 = \frac{1}{2}z^T z + \frac{1}{2}\tilde{\theta}^T\Gamma^{-1}\tilde{\theta}. \tag{2.241}$$

The time derivative of $V_n$ is

$$\begin{aligned}\dot{V}_n = & -\sum_{i=1}^{n-1} c_i z_i^2 - \left(\sum_{i=1}^{n-2} z_{i+1}\frac{\partial \alpha_i}{\partial \hat{\theta}}\right)\left(\dot{\hat{\theta}} - \tau_{n-1}\right) \\ & + z_n\left[z_{n-1} + \varphi_0(x) + \beta_0(x)u - \sum_{i=1}^{n-1}\frac{\partial \alpha_{n-1}}{\partial x_i}x_{i+1} + \omega_n^T\hat{\theta} - \frac{\partial \alpha_{n-1}}{\partial \hat{\theta}}\dot{\hat{\theta}}\right] \\ & + \tilde{\theta}^T\Gamma^{-1}\left(-\dot{\hat{\theta}} + \tau_{n-1} + \Gamma\omega_n z_n\right)\end{aligned} \tag{2.242}$$

One can eliminate $\tilde{\theta}$ from $\dot{V}_n$ with the final tuning function, i.e. the *actual update law*

$$\begin{aligned}\dot{\hat{\theta}} := \tau_n(z,\hat{\theta}) &= \tau_{n-1} + \Gamma\omega_n z_n \\ &= \Gamma W(z,\hat{\theta})z,\end{aligned} \tag{2.243}$$

where the regressor matrix $W$ is composed of the regressor vectors $\omega_1, \ldots, \omega_n$

$$W(z,\hat{\theta}) = [\omega_1, \ \ldots, \ \omega_n]. \tag{2.244}$$

Noting that

$$\dot{\hat{\theta}} - \tau_{n-1} = \tau_n(z,\hat{\theta}) - \tau_{n-1} = \Gamma\omega_n z_n, \tag{2.245}$$

$\dot{V}_n$ can be rewritten as

$$\begin{aligned}\dot{V}_n = -\sum_{i=1}^{n-1} c_i z_i^2 + z_n\Bigg[ & z_{n-1} + \varphi_0(x) + \beta_0(x)u - \sum_{i=1}^{n-1}\frac{\partial \alpha_{n-1}}{\partial x_i}x_{i+1} + \omega_n^T\hat{\theta} \\ & -\frac{\partial \alpha_{n-1}}{\partial \hat{\theta}}\tau_n - \left(\sum_{i=1}^{n-2} z_{i+1}\frac{\partial \alpha_i}{\partial \hat{\theta}}\right)\Gamma\omega_n\Bigg]\end{aligned} \tag{2.246}$$

In order to achieve the goal

$$\dot{V}_n = -\sum_{i=1}^{n} c_i z_i^2 \tag{2.247}$$

with all $c_i > 0$, the control $u$ should be chosen to make the bracketed term multiplying $z_n$ equal to $-c_n z_n$ as follows

$$\begin{aligned} u &= \frac{1}{\beta_0(x)}\Bigg[-z_{n-1} - \varphi_0(x) + \sum_{i=1}^{n-1}\frac{\partial\alpha_{n-1}}{\partial x_i}x_{i+1} - \omega_n^T\hat{\theta} + \frac{\partial\alpha_{n-1}}{\partial\hat{\theta}}\tau_n \\ &\quad + \left(\sum_{i=1}^{n-2} z_{i+1}\frac{\partial\alpha_i}{\partial\hat{\theta}}\right)\Gamma\omega_n - c_n z_n\Bigg]. \end{aligned} \tag{2.248}$$

The overall closed-loop form of the *error system* is

$$\dot{z} = A_z(z,\hat{\theta})z + W^T(z,\hat{\theta})\tilde{\theta} \tag{2.249}$$

$$\dot{\hat{\theta}} = \Gamma W(z,\hat{\theta})z, \tag{2.250}$$

where the matrix $A_z(z,\hat{\theta})$ has the following skew-symmetric form

$$A_z = \begin{bmatrix} -c_1 & 1 & 0 & \dots & 0 & 0 \\ -1 & -c_2 & 1+\varrho_{2,3} & \dots & \varrho_{2,n-1} & \varrho_{2,n} \\ 0 & -1-\varrho_{2,3} & -c_3 & \dots & \varrho_{3,n-1} & \varrho_{3,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & -\varrho_{2,n-1} & -\varrho_{3,n-1} & \dots & -c_{n-1} & 1+\varrho_{n-1,n} \\ 0 & -\varrho_{2,n} & -\varrho_{3,n} & \dots & -1-\varrho_{n-1,n} & -c_n \end{bmatrix} \tag{2.251}$$

with

$$\varrho_{i,j} = -\frac{\partial\alpha_{i-1}}{\partial\hat{\theta}}\Gamma\omega_j. \tag{2.252}$$

The skew-symmetric form of the matrix $A_z$ is of paramount importance for the stability of the system(2.249)-(2.250), since the relation

$$A_z(z,\hat{\theta}) + A_z^T(z,\hat{\theta}) = -2\begin{bmatrix} c_1 & 0 & \dots & 0 \\ 0 & c_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & c_n \end{bmatrix}, \qquad \forall(z,\hat{\theta}) \in \Re^{n+p} \tag{2.253}$$

yields (2.247) with the quadratic Lyapunov function (2.241). The stability properties of the equilibrium point $(z,\tilde{\theta}) = (0,0)$ of the system (2.249)-(2.250) will be analysed in Section 2.4.4.

A characterization of the above described algorithm from a *passivity* perspective has been presented by Kokotović *et al* [58]. Full-state measurement and observed-based versions of backstepping for tracking adaptive control of nonlinear uncertain systems have been developed by Kanellakopoulos *et al* [48, 49], Kokotović [56] and Marino *et al* [76]. Also, the nonlinear design of adaptive backstepping controllers for linear systems with uncertainties have been studied by Krstić *et al* [63, 62] and an early application of the backstepping design has been reported by Dawson *et al* [21].

**Example 2.13 (Benchmark Example)** Consider the third order system [64]

$$\begin{aligned} \dot{x}_1 &= x_2 + \varphi^T(x_1)\theta \\ \dot{x}_2 &= x_3 \\ \dot{x}_3 &= u \end{aligned} \tag{2.254}$$

where $\theta$ is an unknown constant parameter vector and $\varphi$ is a known vector with smooth nonlinear function entries. The system (2.254) has *uncertainty level* two, i.e. two integrators separate the uncertainty from the control input. Therefore, the uncertainty is *unmatched.* This system is already in the PSF form and the backstepping algorithm with tuning function can be applied directly. For the regulation of the output $y = x_1$ to the desired set-point $y_r$ the backstepping algorithm gives the error variables

$$\begin{aligned} z_1 &= x_1 - y_r \\ z_2 &= x_2 - \alpha_1(x_1, \hat{\theta}) \\ z_3 &= x_3 - \alpha_2(x_1, x_2, \hat{\theta}) \end{aligned} \tag{2.255}$$

with the stabilizing functions $\alpha_1$ and $\alpha_2$ given by

$$\begin{aligned} \alpha_1 &= -\varphi^T(x_1)\hat{\theta} - c_1 z_1 \\ \alpha_2 &= -z_1 + \frac{\partial \alpha_1}{\partial x_1}(x_2 + \varphi^T(x_1)\hat{\theta}) + \frac{\partial \alpha_1}{\partial \hat{\theta}}\tau_2 - c_2 z_2 \end{aligned} \tag{2.256}$$

The tuning functions obtained at the successive steps are

$$\begin{aligned} \tau_1 &= \Gamma z_1 \omega_1 = \Gamma z_1 \varphi \\ \tau_2 &= \tau_1 + \Gamma z_2 \omega_2 = \tau_1 - \Gamma z_2 \frac{\partial \alpha_1}{\partial x_1}\varphi \\ \tau_3 &= \tau_2 + \Gamma z_3 \omega_3 = \tau_2 - \Gamma z_3 \frac{\partial \alpha_2}{\partial x_1}\varphi \end{aligned} \tag{2.257}$$

The parameter update law yields

$$\dot{\hat{\theta}} = \tau_3 = \Gamma\left(z_1\varphi - z_2\frac{\partial \alpha_1}{\partial x_1}\varphi - z_3\frac{\partial \alpha_2}{\partial x_1}\varphi\right) \tag{2.258}$$

and the feedback control law is

$$u = -z_2 + \frac{\partial \alpha_2}{\partial x_1}(x_2 + \varphi^T \hat{\theta}) + \frac{\partial \alpha_2}{\partial x_2} x_3 + \frac{\partial \alpha_2}{\partial \hat{\theta}} \tau_3 - z_2 \frac{\partial \alpha_1}{\partial \hat{\theta}} \Gamma \frac{\partial \alpha_2}{\partial x_1} \varphi - c_3 z_3 \tag{2.259}$$

The closed-loop system in the $(z, \tilde{\theta})$-coordinate system yields

$$\dot{z} = \begin{bmatrix} -c_1 & 1 & 0 \\ -1 & -c_2 & 1 - \frac{\partial \alpha_2}{\partial x_1}\varphi^T \Gamma \varphi \\ 0 & -1 + \frac{\partial \alpha_2}{\partial x_1}\varphi^T \Gamma \varphi & -c_3 \end{bmatrix} z + \begin{bmatrix} 1 \\ \frac{\partial \alpha_1}{\partial x_1} \\ \frac{\partial \alpha_2}{\partial x_1} \end{bmatrix} \varphi^T \tilde{\theta} \tag{2.260}$$

$$\dot{\tilde{\theta}} = \Gamma \varphi \left[1,\ -\frac{\partial \alpha_1}{\partial x_1},\ -\frac{\partial \alpha_2}{\partial x_1}\right] z \tag{2.261}$$

### 2.4.4 Stability and Convergence Properties

The global stability of the equilibria $(z, \tilde{\theta}) = (0, 0)$ of the system (2.249)-(2.250) and $(x, \hat{\theta}) = (X, \theta)$ of the original system, are established in this section (see also [64]).

From Theorem A.1 the global stability of the equilibrium $(z, \tilde{\theta}) = (0, 0)$ follows from the fact that the derivative $\dot{V}_n$ of the Lyapunov function $V_n$ along the solutions of (2.249)-(2.250) is nonpositive. Moreover, from the LaSalle Invariance Theorem (Theorem A.2), it follows that the $(n+p)$-dimensional state $(z(t), \tilde{\theta}(t))$ converges to the largest invariant set $M$ of (2.249)-(2.250) contained in $E = \{(z, \tilde{\theta}) \in \Re^{n+p} \mid z = 0\}$, i.e. the set where $\dot{V}_n = 0$. This proves, in particular, that $z(t) \to 0$ as $t \to \infty$.

On the invariant set $M$ we have $z \equiv 0$ and $\dot{z} \equiv 0$. Thus, by setting $z = 0$ and $\dot{z} = 0$ in (2.249)-(2.250), $\dot{\hat{\theta}} = 0$ and

$$W^T(z, \hat{\theta})(\theta - \hat{\theta}) = 0 \qquad \forall (z, \hat{\theta}) \in M. \tag{2.262}$$

From (2.227) and (2.244) it is seen that

$$W^T(z, \hat{\theta}) := N(z, \hat{\theta}) F^T(x) = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ -\frac{\partial \alpha_1}{\partial x_1} & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ -\frac{\partial \alpha_{n-1}}{\partial x_1} & \cdots & -\frac{\partial \alpha_{n-1}}{\partial x_{n-1}} & 1 \end{bmatrix} F^T(x) \tag{2.263}$$

where

$$F(x) = [\varphi_1(x_1),\ \varphi_2(x_1, x_2),\ \ldots,\ \varphi_n(x)] \tag{2.264}$$

Since $N(z, \hat{\theta})$ is nonsingular for all $(z, \hat{\theta}) \in M$, (2.262) and (2.263) imply

$$F^T(x)(\theta - \hat{\theta}) = 0 \tag{2.265}$$

on $M$. Since $z_1 = x_1 - y_r = 0$ on $M$, the equilibrium is $x_1 = y_r = X_1$, and from (2.265)

$$\varphi_1^T(X_1)(\theta - \hat{\theta}) = 0 \tag{2.266}$$

on $M$. Furthermore, recalling from (2.200) that $\alpha_1 = -\varphi_1^T\hat{\theta} - c_1 z_1$, and since $z_2 = x_2 - \alpha_1 = 0$ and $\alpha_1 = -\varphi_1^T(X_1)\hat{\theta}$ on $M$, the relation $x_2 = -\varphi_1^T(X_1)\hat{\theta} = X_2$ is obtained on $M$. Therefore using (2.265)

$$\varphi_2^T(X_1, X_2)(\theta - \hat{\theta}) = 0 \tag{2.267}$$

on $M$. By proceeding in the same manner for the remaining state variables, it can be shown that $x_i = X_i$ and $\varphi_i^T(X_1, \ldots, X_i)(\theta - \hat{\theta}) = 0$ on $M$ for $i = 1, \ldots, n$. Thus the largest invariant set $M$ in $E$ is contained in

$$\begin{aligned} M &\subset \{(z, \tilde{\theta}) \in \Re^{n+p} \mid z = 0,\ F_e^T\tilde{\theta} = 0\} \\ &= \{(x, \hat{\theta}) \in \Re^{n+p} \mid x = X,\ F_e^T\hat{\theta} = F_e^T\theta\} \end{aligned} \tag{2.268}$$

where $F_e = F(X)$. The two equivalent expressions in (2.268) and the convergence of $(z(t), \tilde{\theta})$ to $M$ prove that $x(t) \to X$ as $t \to \infty$. The convergence of the parameter estimates $\hat{\theta}$ to the true unknown parameters depends on the dimension of $M$ which equals $p - \text{rank}\{F_e\}$. When $\text{rank}\{F_e\} = p$, then dim $M = 0$, i.e. $M$ becomes the equilibrium point $x = X$, $\hat{\theta} = \theta$. Thus the parameter estimates converge to their true values, so that the equilibrium $x = X$, $\hat{\theta} = \theta$ is *globally asymptotically stable*.

The above facts lead to the following theorem (see [64]):

**Theorem 2.5** *The closed-loop system consisting of the plant (2.155), the controller (2.248) and the update law (2.243) has a globally stable equilibrium* $(x, \hat{\theta}) = (X, \theta)$. *Furthermore, its state* $(x(t), \hat{\theta}(t))$ *converges to the* $(p - \text{rank}\{F_e\})$*-dimensional equilibrium manifold* $M$ *given by (2.268), which means, in particular, that*

$$\lim_{t \to \infty} x(t) = X. \tag{2.269}$$

*If* $y_r = 0$ *and* $F(0) = 0$, *then* $\lim_{t\to\infty} x(t) = 0$. *The equilibrium* $(x, \hat{\theta}) = (X, \theta)$ *is globally asymptotically stable if and only if* $\text{rank}\{F_e\} = p$.

**Proof:** Given constructively before the statement of Theorem 2.5.

□

Regarding Theorem 2.5, the stability properties of Example 2.13 can be readily established. Consider the particular case in which $\theta$ is a scalar uncertain constant parameter and $\varphi_1(x_1) = x_1^2$, i.e. the uncertain system is

$$\begin{aligned} \dot{x}_1 &= x_2 + x_1^2\theta \\ \dot{x}_2 &= x_3 \\ \dot{x}_3 &= u \end{aligned} \tag{2.270}$$

Since dim $\theta = p = 1$, this example corresponds to the simplest case. If the equilibrium value for $x_1$ is $X_1 \neq 0$, then the manifold $M$ is the single point $x_1 = X_1$, $x_2 = X_2 = -X_1^2\theta$, $x_3 = 0$, $\hat{\theta} = \theta$, which is a *globally asymptotically stable* equilibrium. However, if the equilibrium value is $X_1 = 0$, global stabilization is still guaranteed but the parameter estimate does not converge to the unknown true parameter value. This is shown in Figures 2.3, 2.4 and 2.5 showing the state trajectories of the system (2.270) regulated to the origin for different initial conditions; design parameters $c_1 = 3$, $c_2 = 1$, $c_3 = 5$, $\gamma = 0.1$; and a nominal unknown parameter $\theta = 1$. The parameter estimate $\hat{\theta}$ converges to different values for different initial conditions. Nevertheless, the control objective of regulation to the origin is accomplished for all the initial conditions (global). The lack of convergence of the parameter estimate $\hat{\theta}$ to the unknown true parameter value is due to $\varphi_1(0) = 0$ and thus the rank condition is not satisfied.

We consider a second equilibrium point $x = (1, -1, 0)$ of the system (2.270). For this case the equilibrium point $(x, \tilde{\theta}) = (1, -1, 0, 0)$ of the closed-loop system is stabilized globally and *asymptotically*. This is shown in Figures 2.6, 2.7 and 2.8, which were obtained for the same initial conditions and design parameters used in the case of regulation to the origin. The parameter estimate converges asymptotically to the unknown true parameter value, and the state variables to their desired equilibrium values. This global asymptotic regulation is achieved because $\varphi_1(1) = 1$ and thus the rank condition is satisfied. The oscillatory behaviour exhibited by $\hat{\theta}$ and the state variables in Figure 2.8 can be eliminated by selecting appropriate values for the design parameters. This is illustrated in Figure 2.9 in which the design parameters were chosen to be $c_1 = 5$, $c_2 = 5$, $c_3 = 6$ and $\gamma = 0.01$, and the responses show asymptotic stability.

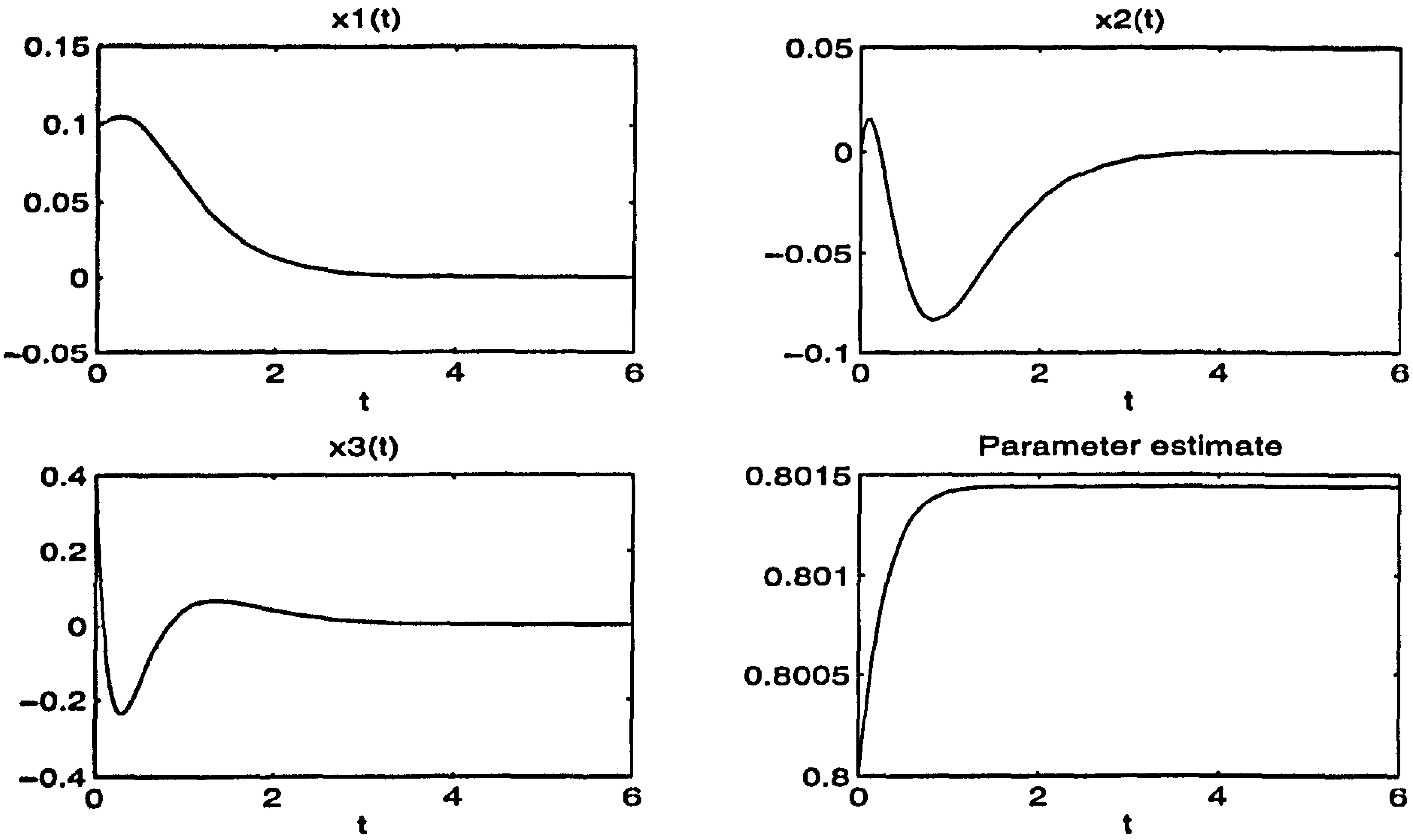


Figure 2.3: Controlled state variables and parameter estimate of the Benchmark example in regulation to the origin

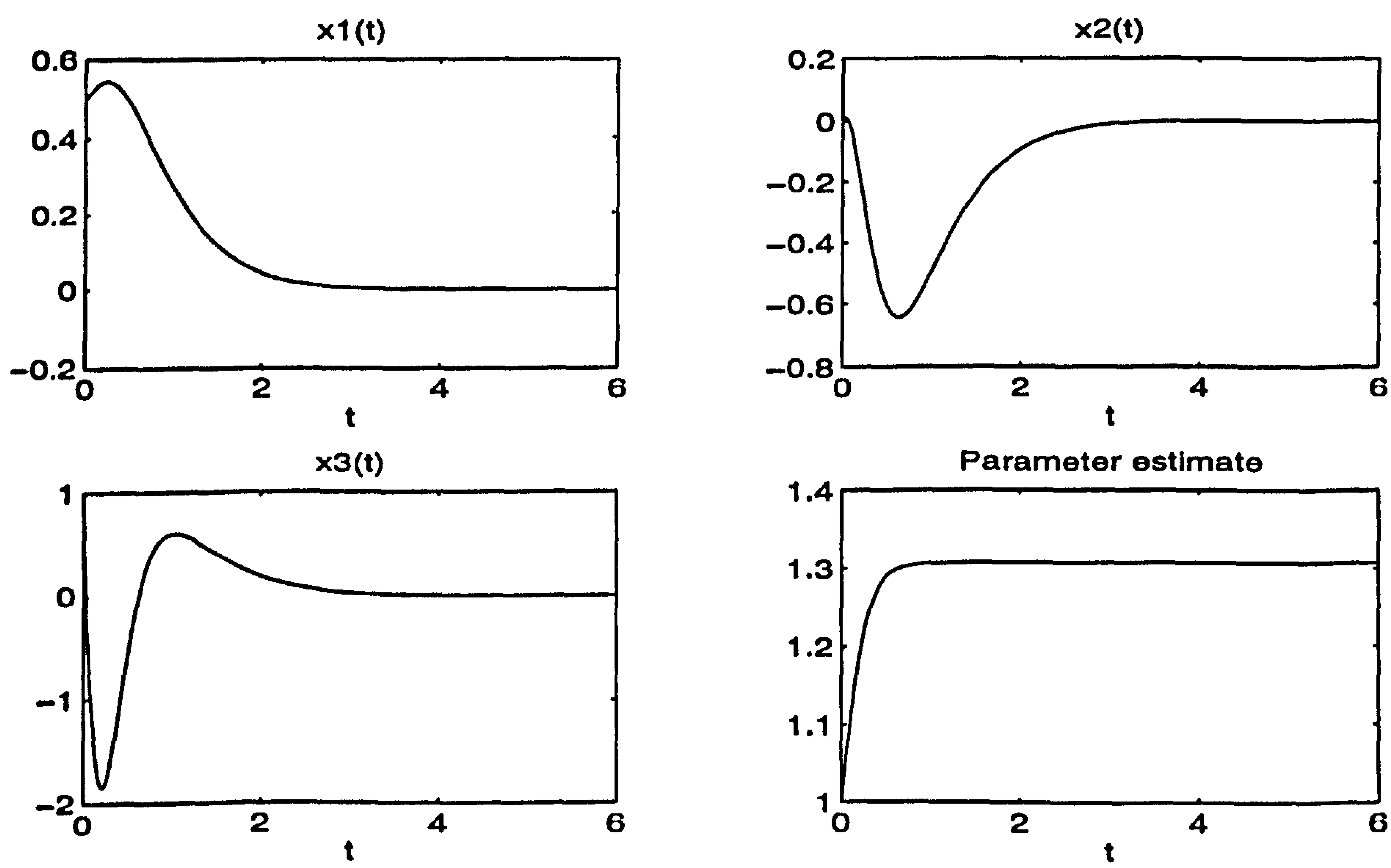


Figure 2.4: Controlled state variables and parameter estimate of the Benchmark example for the initial conditions $x_1 = 0.5$, $x_2 = 0$, $x_3 = 0.8$ and $\hat{\theta} = 1$

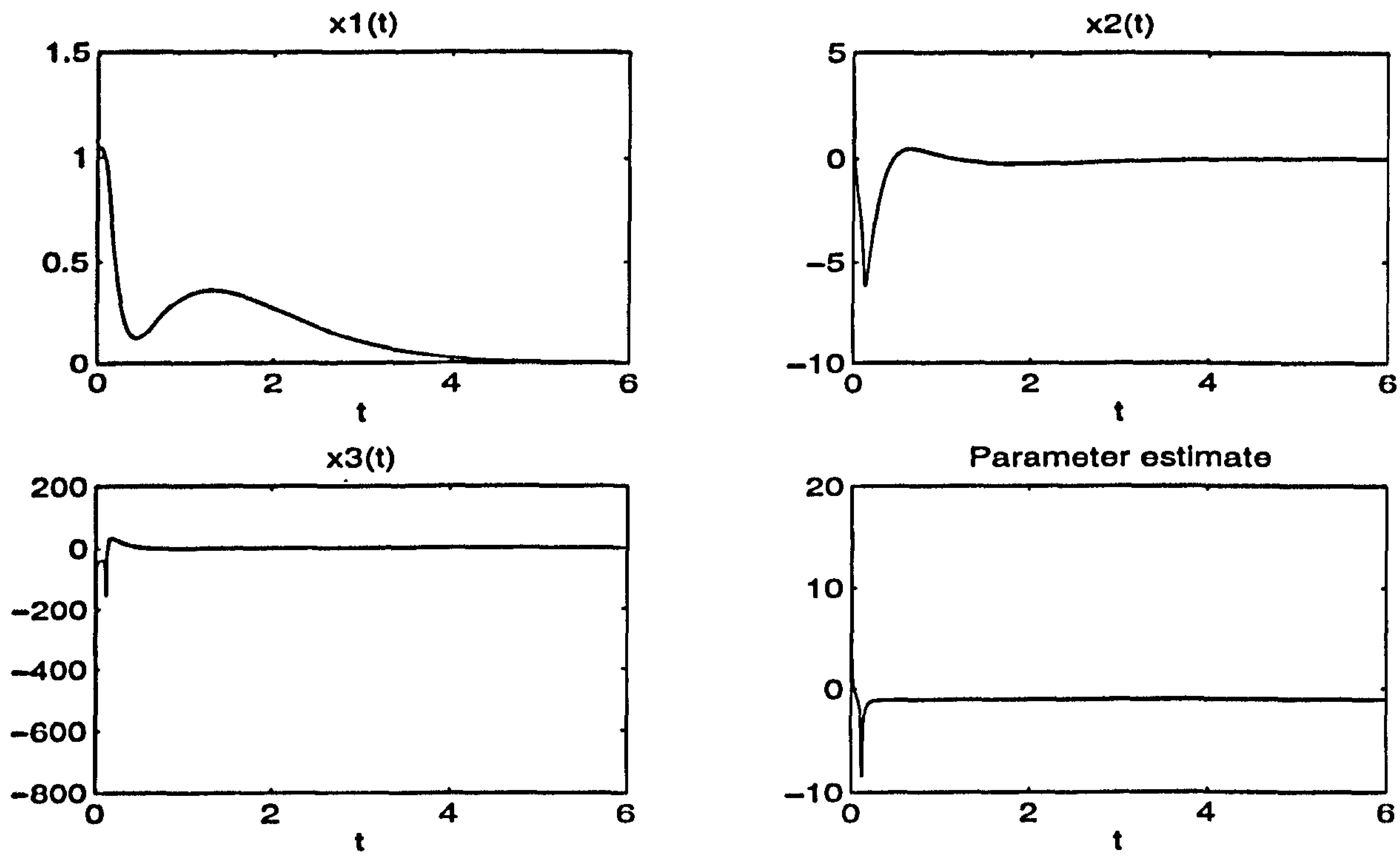


Figure 2.5: Controlled state variables and parameter estimate of the Benchmark example for the initial conditions $x_1 = 1$, $x_2 = 3$, $x_3 = 2$ and $\hat{\theta} = 2$

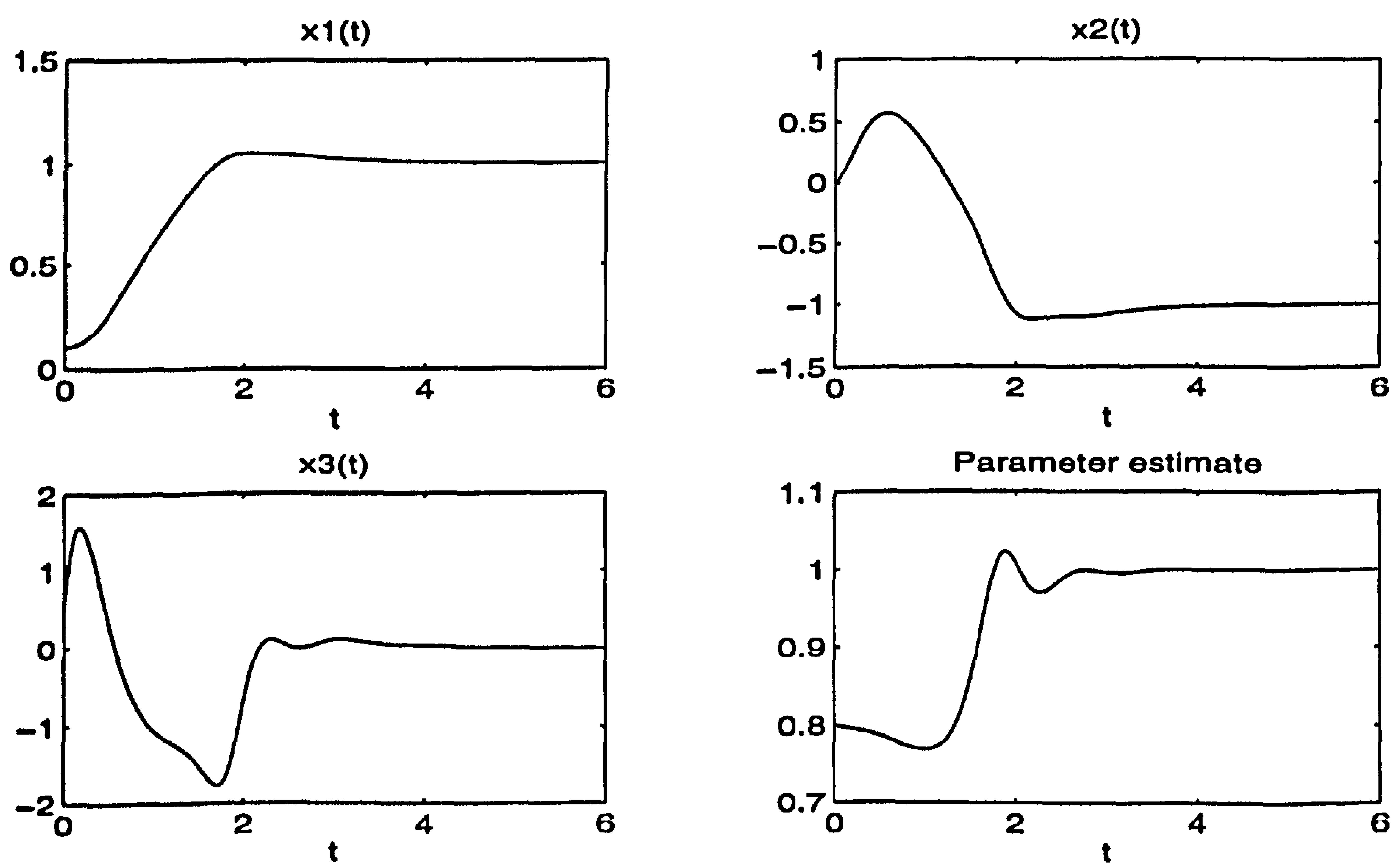


Figure 2.6: Controlled state variables and parameter estimate of the Benchmark example in regulation to the equilibrium point $(x, \hat{\theta}) = (1, -1, 0, 1)$

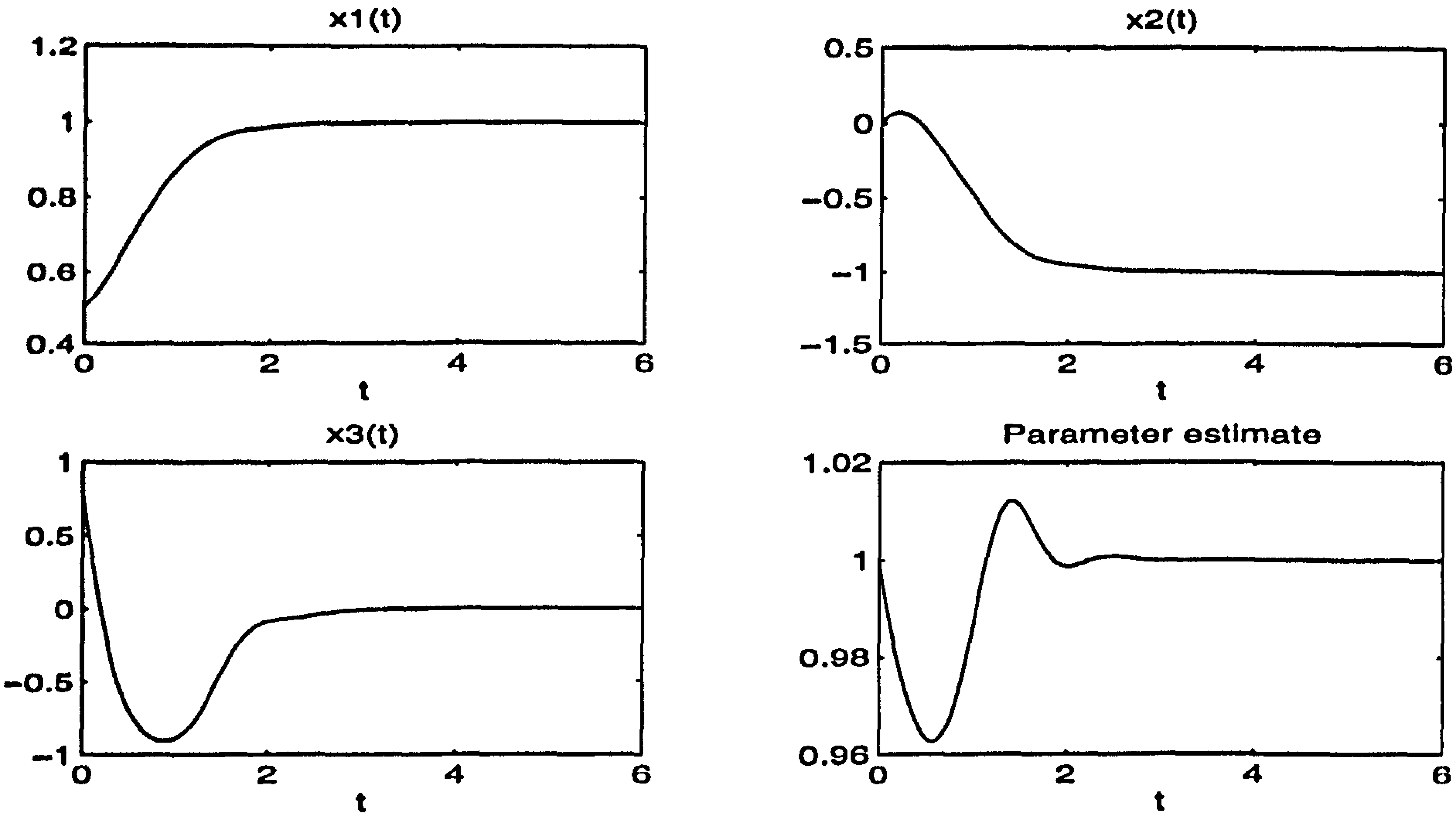


Figure 2.7: Controlled state variables and parameter estimate of the Benchmark example for the initial conditions $x_1 = 0.5$, $x_2 = 0$, $x_3 = 0.8$ and $\hat{\theta} = 1$

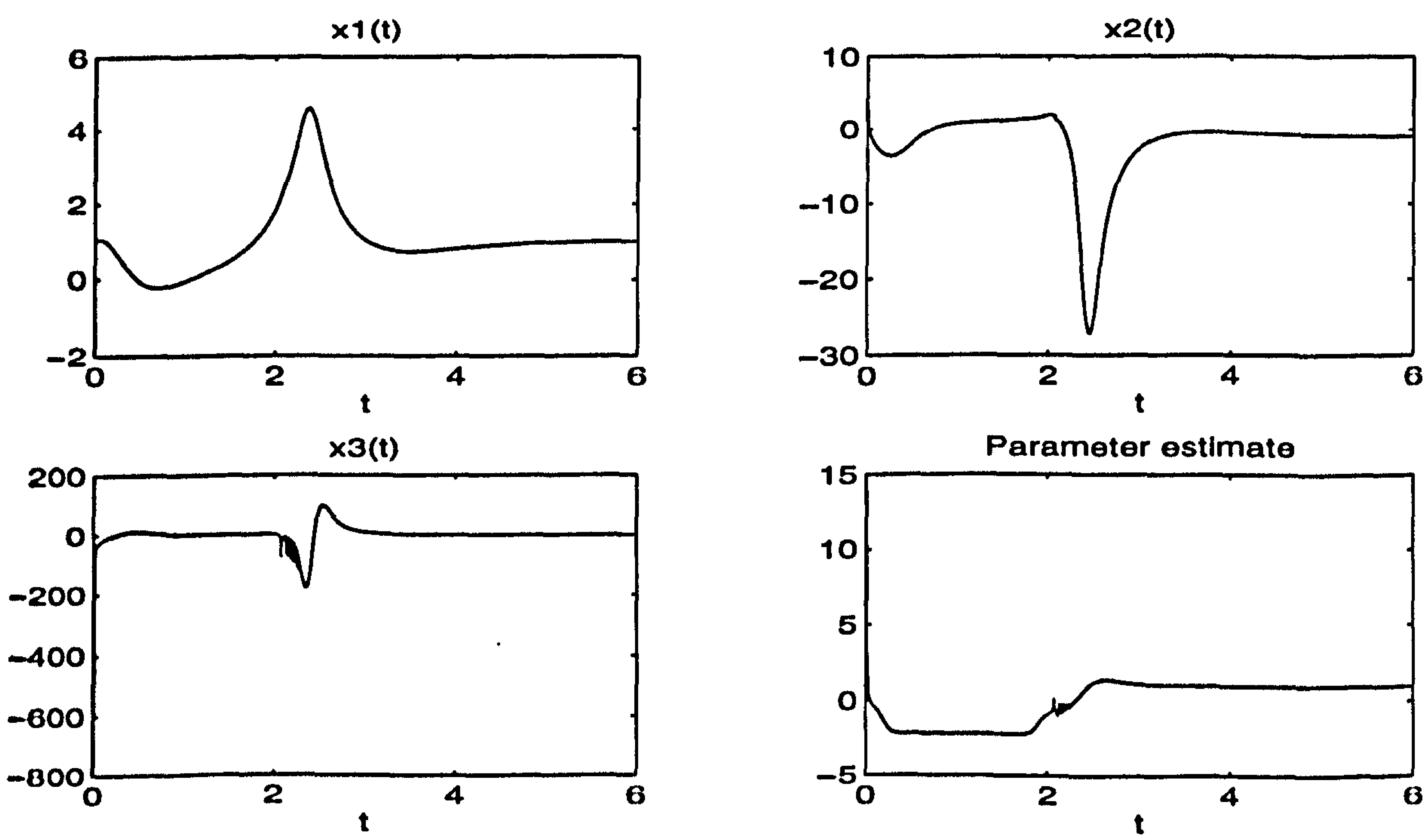


Figure 2.8: Controlled state variables and parameter estimate of the Benchmark example for the initial conditions $x_1 = 1$, $x_2 = 3$, $x_3 = 2$ and $\hat{\theta} = 2$

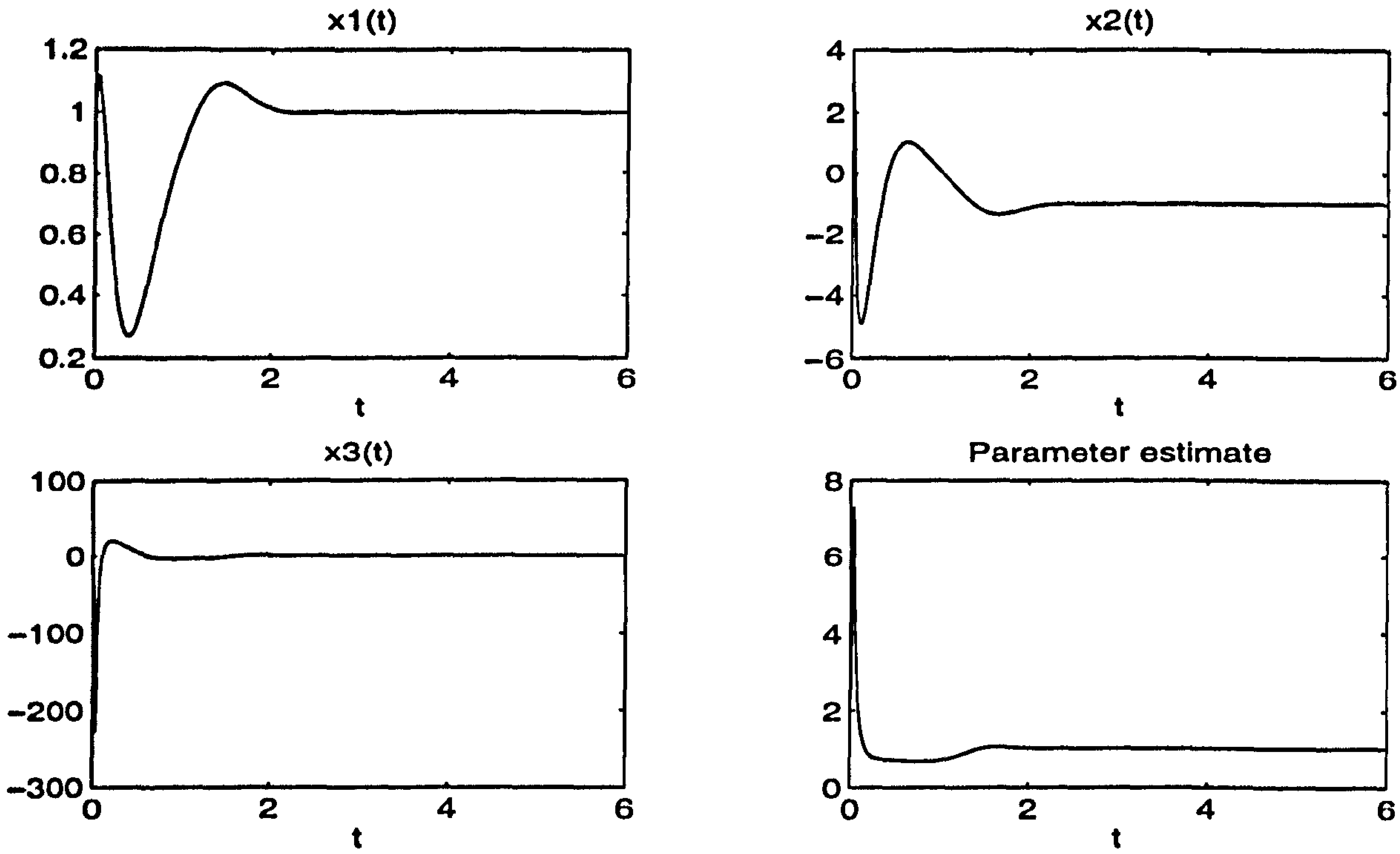


Figure 2.9: Controlled state variables and parameter estimate of the Benchmark example with the design parameters $c_1 = 5$, $c_2 = 5$, $x_3 = 6$ and $\gamma = 0.01$

Further analysis of convergence of the parameter estimates can be carried out, without loss of generality, by considering the regulation of the second order system

$$\begin{aligned} \dot{x}_1 &= x_2 + \varphi_1^T(x_1)\theta \\ \dot{x}_2 &= u + \varphi_2^T(x)\theta \end{aligned} \tag{2.271}$$

to $X_1 = 0$. After applying the backstepping design and using Theorem 2.5, the point

$$\begin{bmatrix} x_1 \\ x_2 \\ \hat{\theta} \end{bmatrix} = \begin{bmatrix} 0 \\ -\varphi_1^T(0)\theta \\ \theta \end{bmatrix} \tag{2.272}$$

is a globally stable equilibrium, and the state of the closed-loop system converges to the equilibrium manifold

$$M = \left\{ (x, \hat{\theta}) \in \Re^{2+p} \;\middle|\; \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ -\varphi_1^T(0)\theta \end{bmatrix}, \begin{bmatrix} \varphi_1^T(0) \\ \varphi_2^T(0, -\varphi_1^T(0)\theta) \end{bmatrix} (\theta - \hat{\theta}) = 0 \right\}. \tag{2.273}$$

Now, if $p = 2$, three different cases can be distinguished.

- Rank$\{F_e\} = 2$. In this case $F_e^T$ can be represented by

$$\begin{bmatrix} \varphi_1^T(0) \\ \varphi_2^T(0, -\varphi_1^T(0)\theta) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}. \tag{2.274}$$

  The manifold $M$ is the single point $x_1 = 0,\ x_2 = -\theta_2,\ \hat{\theta}_1 = \theta_1,\ \hat{\theta}_2 = \theta_2$, which is a globally *asymptotically* stable equilibrium.

- Rank$\{F_e\} = 1$. In this case $F_e^T$ can be represented, for instance, by

$$\begin{bmatrix} \varphi_1^T(0) \\ \varphi_2^T(0, -\varphi_1^T(0)\theta) \end{bmatrix} = \begin{bmatrix} -1 & 1 \\ 0 & 0 \end{bmatrix}. \tag{2.275}$$

  The manifold $M$ is the linear variety $x_1 = 0,\ x_2 = \theta_1 - \theta_2,\ \hat{\theta}_2 - \hat{\theta}_1 = \theta_2 - \theta_1$. Neither of the parameter estimates converge to the actual parameter value, but they converge jointly to the line $\hat{\theta}_2 = \hat{\theta}_1 + \theta_2 - \theta_1$ in the plane $x_1 = 0,\ x_2 = \theta_1 - \theta_2$.

- Rank$\{F_e\} = 0$. In this case $F_e^T$ is represented by

$$\begin{bmatrix} \varphi_1^T(0) \\ \varphi_2^T(0, -\varphi_1^T(0)\theta) \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}. \tag{2.276}$$

  The manifold $M$ is the plane $x = 0$ and corresponds to the case of the weakest convergence properties because one cannot guarantee that the parameter estimates converge to any submanifold in $M$.

## 2.5 Example: Flexible-Joint Manipulator

Consider a flexible-joint mechanism which consists of a link driven by a motor through a torsional spring (a single-link flexible-joint robot) in the vertical plane ([72, 114]). The system dynamics can be written in a state space representation as

$$\begin{aligned} \dot{\zeta}_1 &= \zeta_2 \\ \dot{\zeta}_2 &= -\frac{k}{j_l}(\zeta_1 - \zeta_3) - \frac{mgl}{j_l}\sin(\zeta_1) \\ \dot{\zeta}_3 &= \zeta_4 \\ \dot{\zeta}_4 &= \frac{k}{j_m}(\zeta_1 - \zeta_3) + \frac{1}{j_m}u \end{aligned} \tag{2.277}$$

where the state variables $\zeta = [\zeta_1, \dot{\zeta}_1, \zeta_2, \dot{\zeta}_2]^T$ are the angular positions and velocities of the link and the motor shaft, respectively. The control input $u$ is the torque applied to the motor shaft, and the system parameters are $m$ mass of the link; $g$ acceleration due to gravity; $l$ distance from the motor shaft to the center of mass of the link; $k$ torsional spring constant; and $j_l, j_m$ moments of inertia about the motor shaft of the link and the motor respectively.

Suppose that one wishes to control this system under parametric uncertainty conditions. In particular consider the stabilization of this mechanism when the mass of the link $m$ is assumed constant but unknown, due to the manipulator driving loads of variable mass. Under these conditions the application of the backstepping algorithm with tuning functions can be analysed in order to get an adaptive controller to stabilize this system.

Clearly (2.277) is not in the PSF form (2.155). Nevertheless, one may investigate whether or not this system is transformable into (2.155) by firstly rewriting its dynamic equations in the form (2.147)

$$\dot{\zeta} = f_0(\zeta) + g_0 u + f_1(\zeta)\theta \tag{2.278}$$

where $\theta = m$ and the vector fields $f_0$, $g_0$ and $f_1$ are defined by

$$f_0(\zeta) = \begin{bmatrix} \zeta_2 \\ -\frac{k}{j_l}(\zeta_1 - \zeta_3) \\ \zeta_4 \\ \frac{k}{j_m}(\zeta_1 - \zeta_3) \end{bmatrix}, \quad g_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{j_m} \end{bmatrix}, \quad f_1(\zeta) = \begin{bmatrix} 0 \\ -\frac{gl}{j_l}\sin(\zeta_1) \\ 0 \\ 0 \end{bmatrix} \tag{2.279}$$

Before checking whether or not the parametric strict-feedback conditions (2.156) of Theorem 2.4 are satisfied, one must verify that Assumption 2.1 is satisfied. By noting that the system

$$\dot{\zeta} = f_0(\zeta) + g_0 u \tag{2.280}$$

with $f_0$ and $g_0$ defined by (2.279), is a linear system in triangular form, Assumption 2.1 is obviously satisfied, by Corollary 2.1. In fact because of the triangular form of (2.280), the coordinate transformation $x = \Phi(\zeta)$ which transforms (2.280) into (2.154) is found by following the recursive procedure described in Section 2.3. Such a transformation is given by

$$x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \Phi(\zeta) = \begin{pmatrix} \zeta_1 \\ \zeta_2 \\ -\frac{k}{j_l}(\zeta_1 - \zeta_3) \\ -\frac{k}{j_l}(\zeta_2 - \zeta_4) \end{pmatrix}. \tag{2.281}$$

The vector fields $g_0$, $ad_{f_0}g_0$ and $ad^2_{f_0}g_0$ which characterize the distributions $\mathcal{G}_j$, $j = 0, 1, 2$ are

$$g_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{j_m} \end{bmatrix}, \quad ad_{f_0}g_0 = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{j_m} \\ 0 \end{bmatrix}, \quad ad^2_{f_0}g_0 = \begin{bmatrix} 0 \\ \frac{k}{j_l j_m} \\ 0 \\ \frac{k}{j_m^2} \end{bmatrix}. \tag{2.282}$$

In this case the Lie brackets of the vector fields (2.282) and $f_1$ yield

$$[g_0, f_1] = [ad_{f_0}g_0, f_1] = [ad^2_{f_0}g_0, f_1] = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{2.283}$$

and therefore the parametric strict-feedback condition (2.156) is satisfied. Thus, by applying the coordinate transformation (2.281), the system (2.277) is transformed into the following system in parametric strict-feedback form

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= x_3 + \varphi(x_1)\theta \\ \dot{x}_3 &= x_4 \\ \dot{x}_4 &= a_0 x_3 + a_1\varphi(x_1)\theta + bu \end{aligned} \tag{2.284}$$

where the nonlinear function $\varphi$ and the constant known parameters are defined by

$$a_1 = \frac{k}{j_l} \tag{2.285}$$

$$a_0 = -\left(a_1 + \frac{k}{j_m}\right) \tag{2.286}$$

$$b = \frac{a_1}{j_m} \tag{2.287}$$

$$\varphi(x_1) = -\frac{gl}{j_l}\sin(x_1) \tag{2.288}$$

The application of the backstepping algorithm with tuning functions for the regulation of the output $y = x_1$ to the desired set-point $y_r$ gives the error variables

$$\begin{aligned} z_1 &= x_1 - y_r \\ z_2 &= x_2 - \alpha_1(x_1) \\ z_3 &= x_3 - \alpha_2(x_1, x_2, \hat{\theta}) \\ z_4 &= x_4 - \alpha_3(x_1, x_2, x_3, \hat{\theta}), \end{aligned} \tag{2.289}$$

with the stabilizing functions $\alpha_i$, $i = 1, 2, 3$, given by

$$\begin{aligned}
\alpha_1 &= -c_1 z_1 \\
\alpha_2 &= -z_1 - \varphi\hat{\theta} + \frac{\partial \alpha_1}{\partial x_1} x_2 - c_2 z_2 \\
\alpha_3 &= -z_2 + \frac{\partial \alpha_2}{\partial x_1} x_2 + \frac{\partial \alpha_2}{\partial x_2}(x_3 + \varphi\hat{\theta}) + \frac{\partial \alpha_2}{\partial \hat{\theta}} \tau_3 - c_3 z_3
\end{aligned} \tag{2.290}$$

The tuning functions obtained at the successive steps are

$$\begin{aligned}
\tau_2 &= \gamma z_2 \omega_2 = \gamma z_2 \varphi \\
\tau_3 &= \tau_2 + \gamma z_3 \omega_3 = \tau_2 - \gamma z_3 \frac{\partial \alpha_2}{\partial x_2} \varphi \\
\tau_4 &= \tau_3 + \gamma z_4 \omega_4 = \tau_3 + \gamma z_4 \left( a_1 - \frac{\partial \alpha_3}{\partial x_2} \right) \varphi
\end{aligned} \tag{2.291}$$

where $\gamma$ is a scalar adaptation gain. The parameter update law is

$$\dot{\hat{\theta}} = \tau_4 = \gamma\varphi \left[ z_2 - z_3 \frac{\partial \alpha_2}{\partial x_2} + z_4 \left( a_1 - \frac{\partial \alpha_3}{\partial x_2} \right) \right] \tag{2.292}$$

and the designed feedback control yields

$$u = \frac{1}{b} \left[ -z_3 - a_0 x_3 - \omega_4 \hat{\theta} + \sum_{i=1}^{3} \frac{\partial \alpha_3}{\partial x_i} x_{i+1} + \frac{\partial \alpha_3}{\partial \hat{\theta}} \tau_4 + z_3 \frac{\partial \alpha_2}{\partial \hat{\theta}} \gamma \omega_4 - c_4 z_4 \right]. \tag{2.293}$$

The closed-loop system in the $(z, \tilde{\theta})$-coordinates is

$$\dot{z} = \begin{bmatrix} -c_1 & 1 & 0 & 0 \\ -1 & -c_2 & 1 & 0 \\ 0 & -1 & -c_3 & 1 - \gamma \frac{\partial \alpha_2}{\partial \hat{\theta}} \omega_4 \\ 0 & 0 & -1 + \gamma \frac{\partial \alpha_2}{\partial \hat{\theta}} \omega_4 & -c_4 \end{bmatrix} z + \begin{bmatrix} 0 \\ 1 \\ \frac{\partial \alpha_2}{\partial x_2} \\ a_1 - \frac{\partial \alpha_3}{\partial x_2} \end{bmatrix} \varphi\tilde{\theta} \tag{2.294}$$

$$\dot{\hat{\theta}} = \gamma\varphi \left[ 0,\ 1,\ -\frac{\partial \alpha_2}{\partial x_2},\ a_1 - \frac{\partial \alpha_3}{\partial x_2} \right] z. \tag{2.295}$$

Simulations were performed for the single-link manipulator regulated by the above adaptive feedback control. The following known parameters were used for simulation purposes:

$$j_l = 0.05\,\text{N-m-sec}^2 \quad ; \quad j_m = 0.007\,\text{N-m-sec}^2 \quad ; \quad l = 1\,\text{m} \quad ; \quad k = 200\,\text{N-m/rad}$$

while the nominal *unknown* mass was chosen to be $m = 2\,\text{kg}$. In order to drive the load suitably fast to the desired equilibrium position, the design parameters were selected to be $c_1 = 8$, $c_2 = 10$, $c_3 = 5$, $c_4 = 10$ and $\gamma = 0.005$. Figure 2.10 shows the angles of the link and the motor shaft, when the controlled link is moved from the initial condition $\phi_1(0) = 0.0873$ radians to the desired position $\phi_{1e} = 1.2217$ radians, corresponding to a link displacement of 1.1344 radians, i.e. 65 degrees, for a torque $U = 18.4180$ N-m applied to the motor shaft. For this desired angular position of the link, the equilibrium value for $x_3$ is

$$x_3(U) = X_3 = \frac{k}{j_l}(\xi_3(U) - \xi_1(U)) = \frac{U}{j_l} = 368.36 \text{ sec}^{-2}$$

Figure 2.10 also shows the corresponding controlled angular velocities of the link and the motor shaft and the estimate $\hat{\theta}$ of the unknown parameter $\theta$. Since the rank condition

$$\text{rank}\{F_e\} = \text{rank} \begin{bmatrix} 0 \\ \frac{gl}{j_l}\sin(x_1) \\ 0 \\ \frac{a_1 gl}{j_l}\sin(x_1) \end{bmatrix}_{x_1 = X_1} = 1$$

is satisfied, convergence of the parameter estimate $\hat{\theta}$ to the unknown true mass value is guaranteed. Therefore global asymptotic stabilization of the equilibrium point $(x, \tilde{\theta}) = (X_1, 0, X_3, 0, 0)$ of the closed-loop system is achieved, as shown in Figure 2.10. Digital simulations were carried out for a link displacement of 120 degrees, i.e. from the initial position $\phi(0) = 0.0873$ radians to the desired position $X_1 = \phi_e = 2.1817$ radians, for different initial conditions of the link velocity, angular position and velocity of the motor shaft. Figures 2.11, 2.12 and 2.13 show the global asymptotic stability of this new equilibrium point. The same design parameters were used in all the computer simulations.

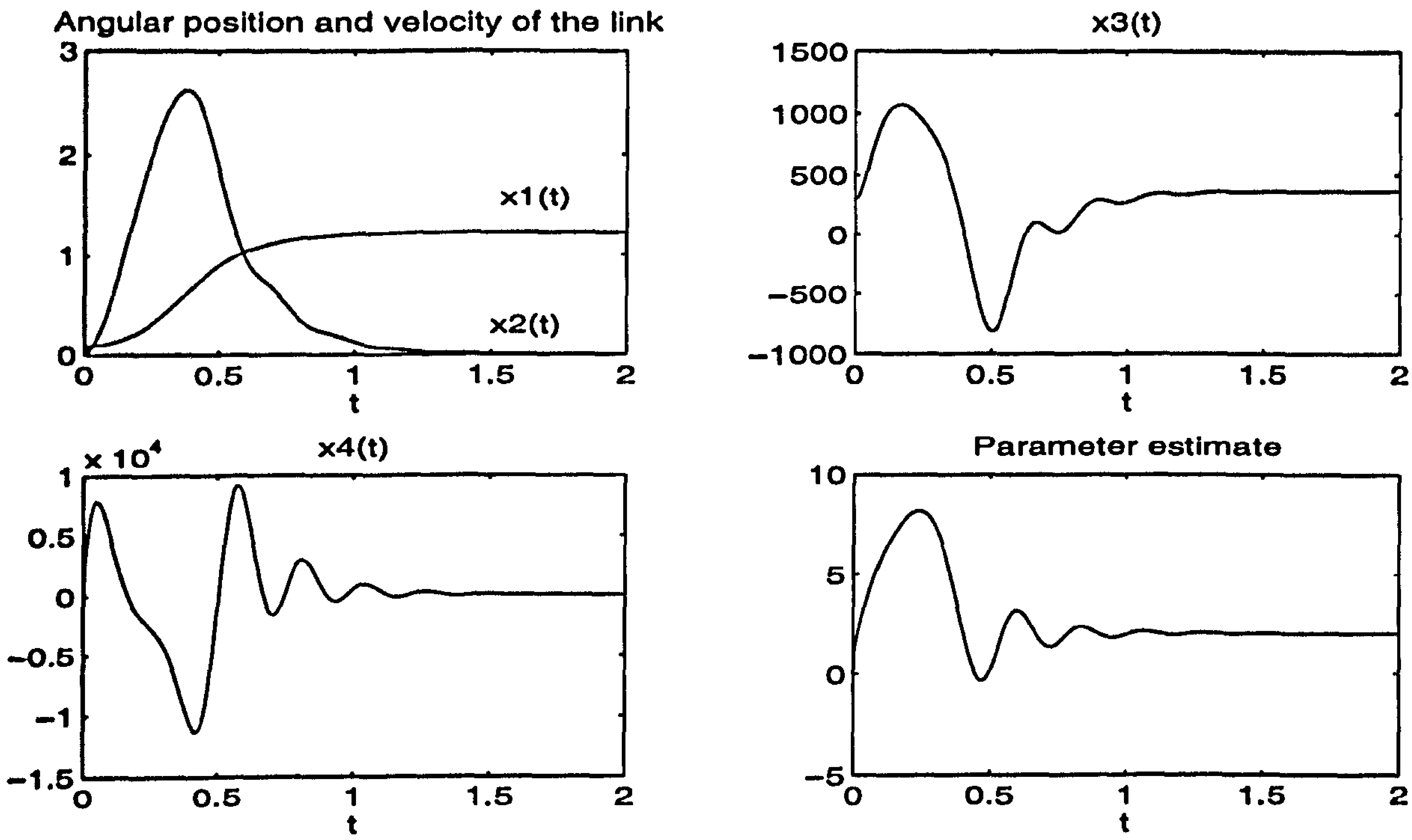


Figure 2.10: Controlled state variables and parameter estimate for a link displacement of 65 degrees

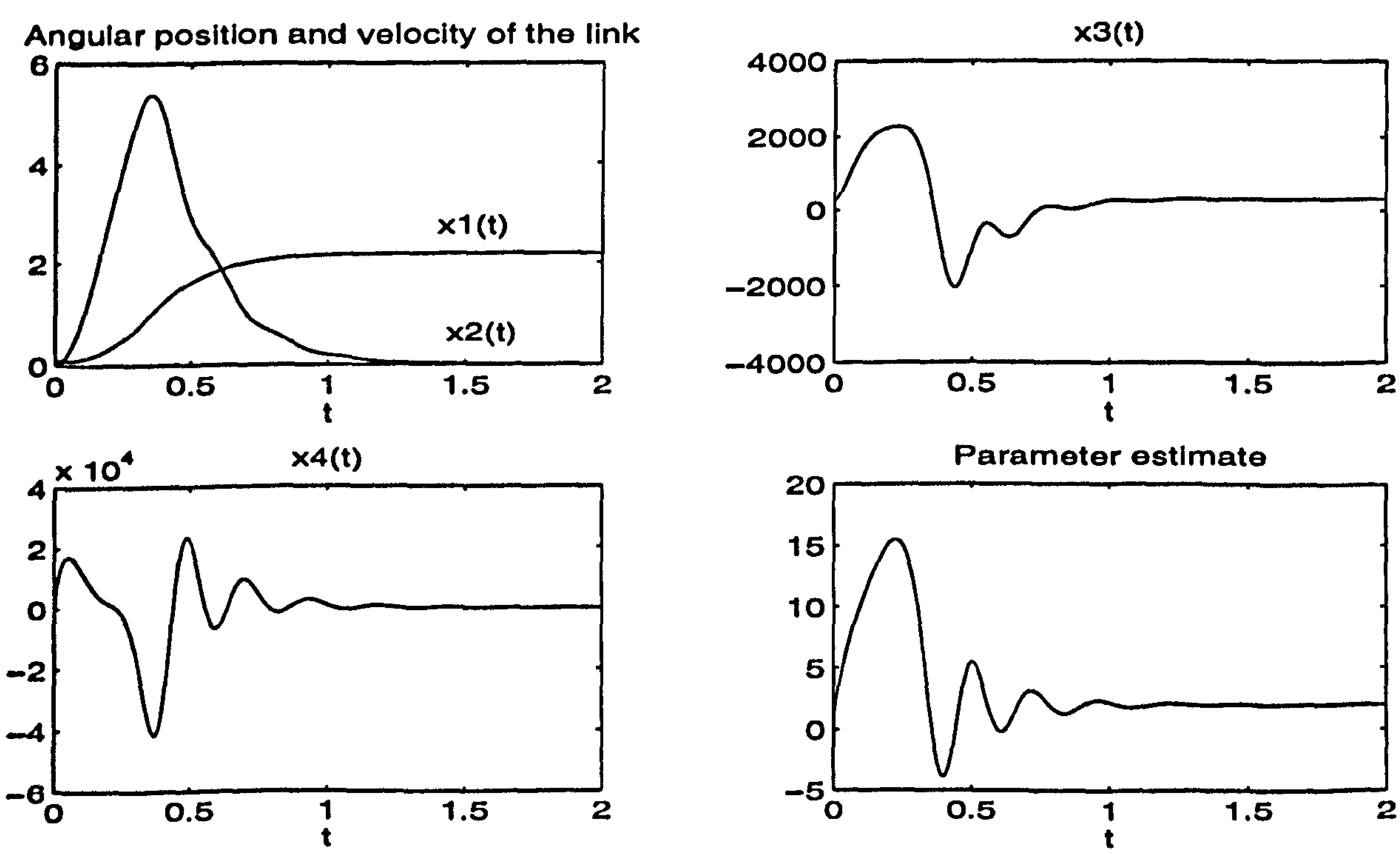


Figure 2.11: Controlled state variables and parameter estimate for a link displacement of 120 degrees

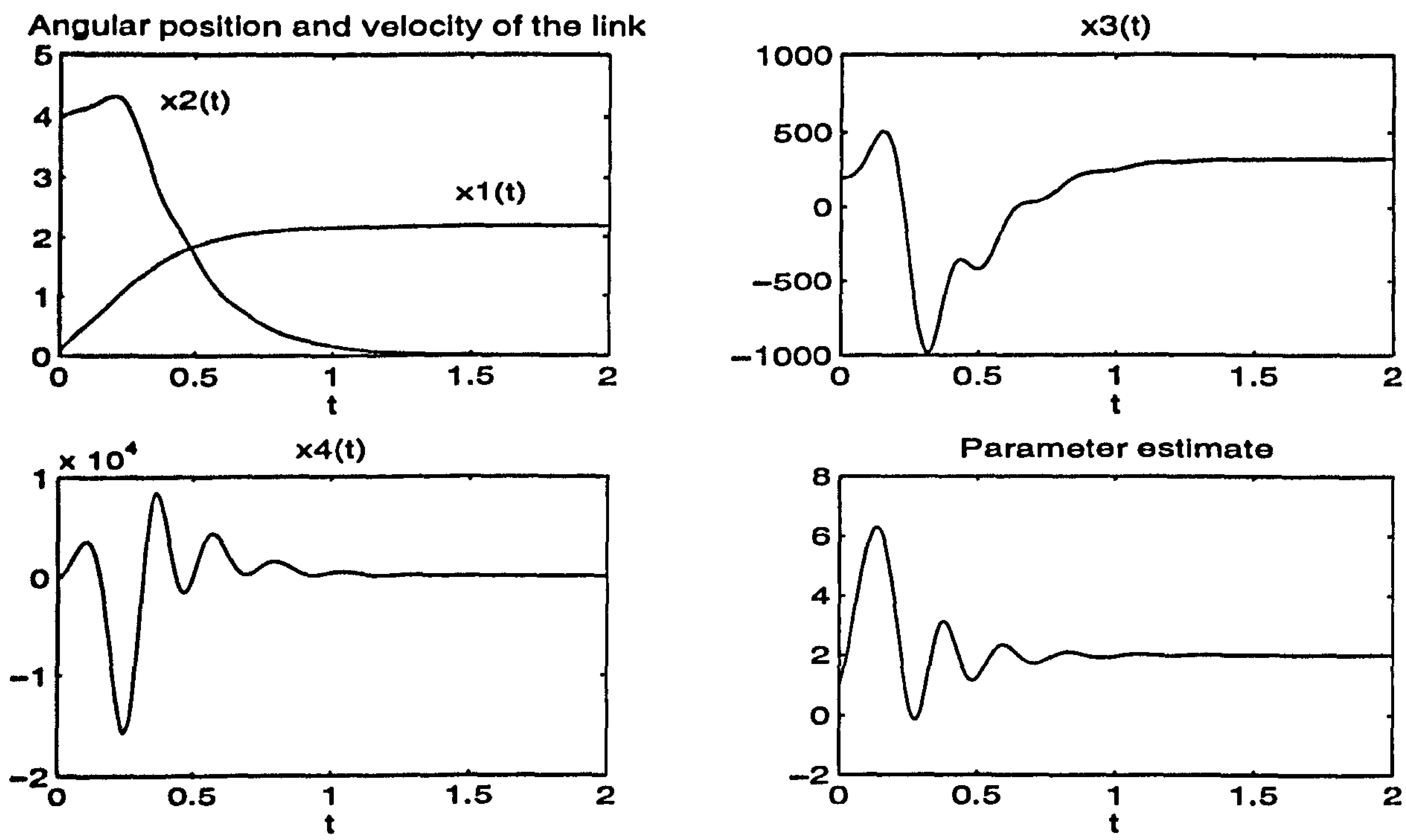


Figure 2.12: Controlled state variables and parameter estimate for a link displacement of 120 degrees and initial conditions $x_1 = 0.0873$, $x_2 = 4$, $x_3 = 200$, $x_4 = 10$ and $\hat{\theta} = 1$

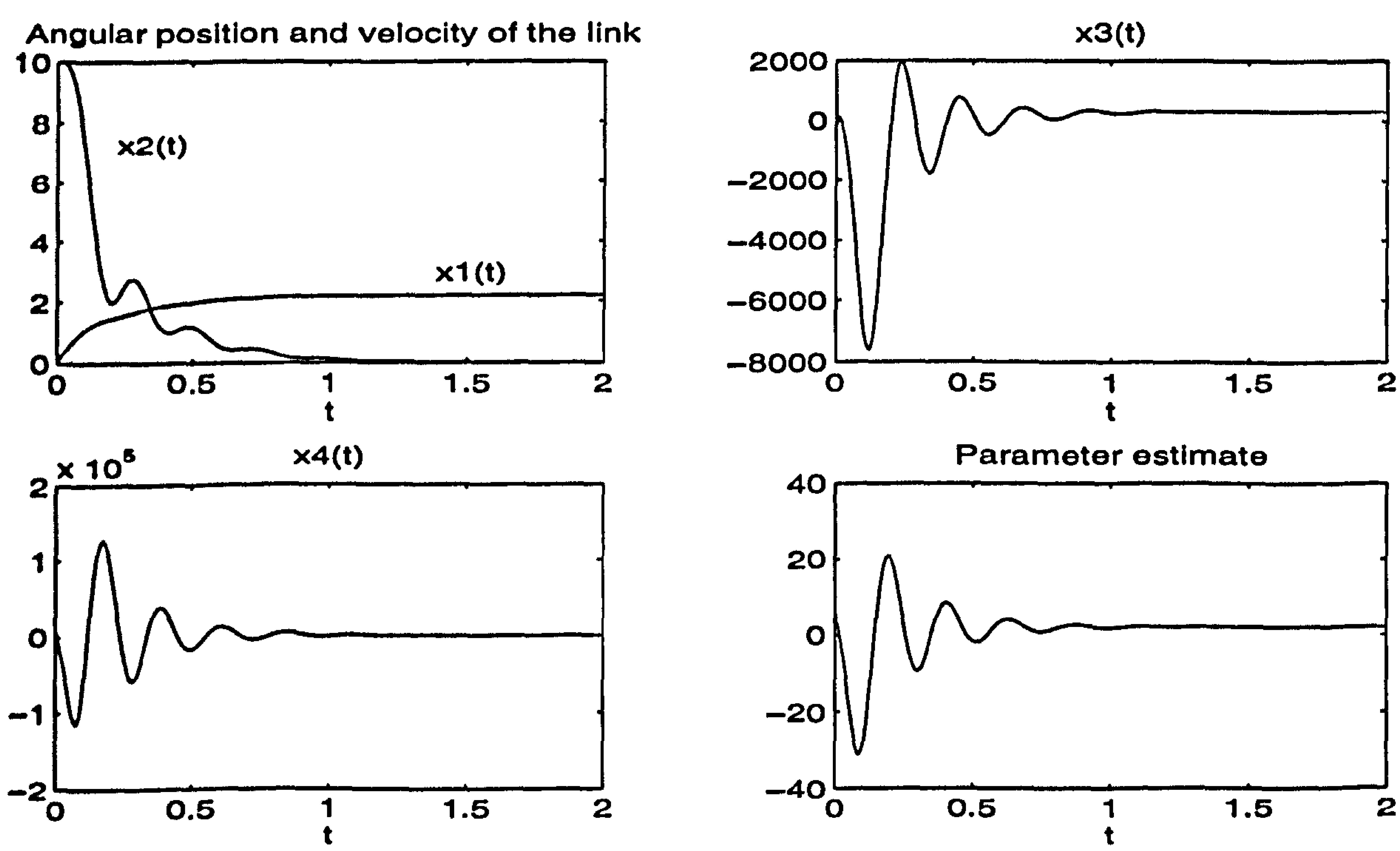


Figure 2.13: Controlled state variables and parameter estimate for a link displacement of 120 degrees and initial conditions $x_1 = 0.0873$, $x_2 = 10$, $x_3 = 100$, $x_4 = 100$ and $\hat{\theta} = 5$

# Chapter 3

# Dynamical Deterministic Feedback Control

## 3.1 Introduction

In this chapter the stabilization of nonlinear systems via *dynamical* deterministic (non-adaptive) controllers is studied. The systematic procedure described here is a new extension of the design of static feedback control for input-output linearizable nonlinear systems. The extended class of deterministic nonlinear systems for which this procedure is applicable is characterized by *observable minimum phase affine nonlinear systems.*

We first need to consider the problem of stabilizing minimum phase nonlinear systems locally and globally. Then dynamical compensation is presented as a natural extension of the static stabilization of these systems. Finally a new deterministic backstepping algorithm for the design of dynamical feedback controllers is described, and examples illustrate the applicability of the design procedure.

## 3.2 Stabilization of Minimum Phase Systems

The stabilization of completely linearizable systems discussed Chapter 2, corresponds to the simplest case of regulation of nonlinear systems. In this section the stabilization of *partially linearizable* nonlinear systems is studied. In particular nonlinear systems with relative degree $\rho$ *less than the system order*, with respect to an output $y = h(x)$, are considered. Consider a nonlinear single-input single-output system described by

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= h(x) \end{aligned} \tag{3.1}$$

where $f(x)$ and $g(x)$ are smooth vector fields in $\Re^n$, and $h(x)$ is a smooth scalar function. Suppose $x = 0$ is an equilibrium point of the vector field $f(x)$, i.e. $f(0) = 0$ and $h(0) = 0$. Assume also that the system (3.1) has relative degree $\rho < n$ at $x = 0$. Then there exists a neighbourhood $R_0$ of $x = 0$ in $\Re^n$ and a local change of coordinates $z = \Phi(x)$ defined on $R_0$, satisfying $\Phi(0) = 0$, such that in the new coordinates, the system is described by the equations in the normal form (see Section 2.2.3 for details) [44]

$$\begin{aligned} \dot{z}_1 &= z_2 \\ \dot{z}_2 &= z_3 \\ &\vdots \\ \dot{z}_{\rho-1} &= z_\rho \\ \dot{z}_\rho &= a(z) + b(z)u \\ \dot{z}_{\rho+1} &= q_{\rho+1}(z) \\ &\vdots \\ \dot{z}_n &= q_n(z) \\ y &= z_1. \end{aligned} \tag{3.2}$$

In order to write (3.2) in a slightly more compact manner, set

$$\xi = \begin{pmatrix} h(x) \\ L_f h(x) \\ \vdots \\ L_f^{\rho-1} h(x) \end{pmatrix} = \begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_\rho \end{pmatrix}, \qquad \eta = \begin{pmatrix} z_{\rho+1} \\ z_{\rho+2} \\ \vdots \\ z_n \end{pmatrix}. \tag{3.3}$$

The normal form of a single-input single-output nonlinear system with $\rho < n$ at $x = 0$, can be rewritten as

$$\begin{aligned} \dot{z}_1 &= z_2 \\ \dot{z}_2 &= z_3 \\ &\vdots \\ \dot{z}_{\rho-1} &= z_\rho \\ \dot{z}_\rho &= a(\xi,\eta) + b(\xi,\eta)u \\ \dot{\eta} &= q(\xi,\eta) \\ y &= z_1 \end{aligned} \tag{3.4}$$

Consider now the problem of *zeroing the output*, namely finding, if any exist, pairs formed by an initial state $x^0$ and by an input function $u^0(\cdot)$, defined for all $t$ in a neighbourhood of $t = 0$, such that the corresponding output $y(t)$ of the system is identically zero for all $t$ in a neighbourhood of $t = 0$ [44].

Recalling that in the normal form

$$y(t) = z_1(t), \tag{3.5}$$

the constraint $y(t) = 0$ for all $t$ implies

$$\dot{z}_1(t) = \dot{z}_2(t) = \ldots = \dot{z}_\rho(t) = 0, \tag{3.6}$$

i.e. $\xi(t) = 0$ for all $t$. In addition the input $u(t)$ must necessarily be the unique solution of the equation

$$0 = a(0, \eta) + b(0, \eta)u. \tag{3.7}$$

As far as the variable $\eta(t)$ is concerned, it is clear that, since $\xi(t)$ is identically zero, its behaviour is governed by the differential equation

$$\dot{\eta}(t) = q(0, \eta(t)). \tag{3.8}$$

We can now conclude that, in order to have the output $y(t)$ of the system identically zero, necessarily the initial state must be such that $\xi(0) = 0$, whereas $\eta(0) = \eta^0$ can be chosen arbitrarily. Depending upon the value $\eta^0$, the unique input capable of keeping $y(t)$ identically zero for all times is

$$u(t) = -\frac{a(0, \eta(t))}{b(0, \eta(t))} \tag{3.9}$$

where $\eta(t)$ denotes the solution of the differential equation (3.8) with initial condition $\eta(0) = \eta^0$.

The dynamics of equation (3.8) correspond to the dynamics describing the "internal" behaviour of the system when input and initial conditions have been chosen in such a way as to constrain the output to remain identically zero. These dynamics are called the *zero dynamics* of the system. The trajectories of (3.8) can also be interpreted as autonomous trajectories of an appropriate closed-loop system. This can be shown by imposing the feedback control law

$$u(t) = \frac{(-a(0, \eta(t)) + v)}{b(0, \eta(t))}, \tag{3.10}$$

with $v$ a new input, to a nonlinear system in the normal form (3.4). The closed-loop system thus obtained is described by the equations

$$\begin{aligned} \dot{\xi} &= A\xi + Bv \\ \dot{\eta} &= q(\xi, \eta) \\ y &= C\xi \end{aligned} \tag{3.11}$$

with

$$A = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \cdot & \cdot & \cdot & \dots & \cdot \\ 0 & 0 & 0 & \dots & 1 \\ 0 & 0 & 0 & \dots & 0 \end{pmatrix}, \qquad B = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}$$

$$C = (1\ 0\ \dots\ 0).$$

If the linear subsystem is initially at rest and no input is applied, then $y(t) = 0$ for all values of $t$, and the corresponding internal dynamics of the whole closed-loop system are exactly those of equation (3.8), namely, the zero dynamics. The concept of the zero dynamics of a nonlinear system plays a role analogously to that of the "zeros" of the transfer function in a linear system. In fact the linear approximation at $\eta = 0$ of the zero dynamics of a system coincides with the zero dynamics of a linear approximation of the system at $x = 0$ (see [44]).

The interpretation of equation (3.8), the dynamics describing the internal behaviour of the system when the output is forced to track the output $y(t) = 0$ exactly, can be extended to the case in which the output to be tracked is any arbitrary function. Consider the problem of finding, if any exist, pairs consisting of an initial state $x^0$ and an input function $u(\cdot)$, defined for all $t$ in a neighbourhood of $t = 0$, such that the corresponding output $y(t)$ of the system coincides exactly with $y_r(t)$ for all $t$ in a neighbourhood of $t = 0$. Proceeding as before, the relation $y(t) = y_r(t)$ necessarily implies

$$z_i(t) = y_r^{(i-1)}(t), \quad 1 \leqslant i \leqslant \rho \quad \forall t \tag{3.12}$$

Setting

$$\xi_r(t) = \operatorname{col}\left(y_r(t), y_r^{(1)}(t), \dots, y_r^{(\rho-1)}(t)\right) \tag{3.13}$$

the input $u(t)$ must necessarily satisfy

$$y_r^{(\rho)} = b(\xi_r(t), \eta(t)) + a(\xi_r(t), \eta(t))u(t) \tag{3.14}$$

where $\eta(t)$ is a solution of the differential equation

$$\dot{\eta}(t) = q(\xi_r(t), \eta(t)). \tag{3.15}$$

Thus, if an output $y(t)$ has to track $y_r(t)$ exactly, then necessarily the initial state of the system must be set to a value such that $\xi(0) = \xi_r(0)$, whereas $\eta(0) = \eta^0$ can be chosen arbitrarily. Under these conditions the unique input capable of keeping $y(t) = y_r(t)$ is

$$u(t) = \frac{y_r^{(\rho)} - a(\xi_r(t), \eta(t))}{b(\xi_r(t), \eta(t))} \qquad \forall\ t \tag{3.16}$$

where $\eta(t)$ denotes the solution of the differential equation (3.15) with initial condition $\xi(0) = \xi_r(0)$.

Note that the internal dynamics (3.15) and the equation (3.16) describe a system with input $\xi_r(t)$, output $u(t)$ and state $\eta(t)$ which can be interpreted as a realization of the inverse of the original system [44].

**Definition 3.1 (Minimum Phase)** *The system (3.1) with $\rho < n$ and zero dynamics (3.8) is said to be* minimum phase *if $\eta = 0$ is an asymptotically stable equilibrium point of (3.8). A system which is not minimum phase is said to be non-minimum phase.*

**Example 3.1** Consider the zero dynamics of the system

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{pmatrix} = \begin{pmatrix} x_2 + x_1^2 \\ x_3 \\ x_1 x_2^2 - x_3^3 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} u \tag{3.17}$$

$$y = x_1.$$

Transform (3.17) into the normal form by proceeding as described in Section 2.2.3. Thus, taking successive time derivatives of the output one obtains

$$L_g h(x) = 0, \quad L_f h(x) = x_2 + x_1^2, \quad L_g L_f h(x) = 1. \tag{3.18}$$

Note that the relative degree of (3.17) is 2. So we can take $y = h(x)$ and $\dot{y} = L_f h(x)$ as the first two coordinates of the transformation placing (3.17) into the normal form. The third coordinate can be chosen in this case as $x_3$. Therefore the normal form can be calculated by using the mapping

$$\begin{aligned} z_1 &= x_1 \\ z_2 &= x_2 + x_1^2 \\ z_3 &= x_3 \end{aligned} \tag{3.19}$$

which is a globally defined coordinate transformation. With these new coordinates, (3.17) is transformed into the following form

$$\begin{aligned} \dot{z}_1 &= z_2 \\ \dot{z}_2 &= z_3 + 2z_1 z_2 + u \\ \dot{z}_3 &= -z_3^3 + z_1(z_2 - z_1^2)^2 \end{aligned} \tag{3.20}$$

The constraint $y(t) = 0$ for all $t$ imposes $z_1(t) = z_2(t) = 0$ for all $t$, and the state must necessarily be governed by the zero dynamics

$$\dot{z}_3 = -z_3^3. \tag{3.21}$$

Since $z_3 = 0$ is an asymptotically stable equilibrium point of the zero dynamics (3.21), system (3.17) is minimum phase.

Definition 3.1 is analogous to the notion of minimum phase for the case of linear systems in which a system is said to be minimum phase if all its transmission zeros have negative real parts.

### 3.2.1 Local Asymptotic Stabilization

The application of the notion of zero dynamics to the local stabilization of minimum phase nonlinear systems was firstly described by Byrnes and Isidori ([8]-[11]) and is summarized in this section.

Consider a nonlinear system in the normal form (3.4) and impose a feedback of the form

$$u = \frac{1}{b(\xi,\eta)}(-a(\xi,\eta) - c_0\xi_1 - c_1\xi_2 - \ldots - c_{\rho-1}\xi_\rho) \tag{3.22}$$

where $c_0, c_1, \ldots, c_{\rho-1}$ are chosen positive real numbers. Recalling the notation in (3.3), the choice of the feedback control (3.22) yields the closed-loop system

$$\begin{aligned} \dot{\xi} &= (A+BK)\xi \\ \dot{\eta} &= q(\xi,\eta) \end{aligned} \tag{3.23}$$

with

$$A+BK = \begin{pmatrix} 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & \ldots & 0 \\ \cdot & \cdot & \cdot & \ldots & \cdot \\ 0 & 0 & 0 & \ldots & 1 \\ -c_0 & -c_1 & -c_2 & \ldots & -c_{\rho-1} \end{pmatrix}. \tag{3.24}$$

The matrix $A+BK$ has a Hurwitz characteristic polynomial

$$p(s) = c_0 + c_1 s + \ldots + c_{\rho-1}s^{\rho-1} + s^\rho \tag{3.25}$$

i.e. all its roots have negative real parts. From this form of the equations describing the closed-loop system, the following property is deduced (see [44]):

**Proposition 3.1** *Suppose the equilibrium $\eta = 0$ of the zero dynamics of the system is locally asymptotically stable and all the roots of the polynomial $p(s)$ have negative real parts. Then the feedback control (3.22) locally asymptotically stabilizes the closed-loop system (3.23) at the equilibrium $(\xi,\eta) = (0,0)$.*

The linear approximation of the zero dynamics at $\eta = 0$ is characterized by the matrix

$$Q = \left[\frac{\partial q(\xi,\eta)}{\partial \eta}\right]_{(\xi,\eta)=(0,0)} \tag{3.26}$$

If $Q$ has its eigenvalues in the left half complex plane, then the result stated in Proposition 3.1 would have been a trivial consequence of the *Principle of Stability of the First Approximation*, because the linear approximation of (3.23) has the form [44]

$$\begin{pmatrix} \dot{\xi} \\ \dot{\eta} \end{pmatrix} = \begin{pmatrix} A & 0 \\ * & Q \end{pmatrix} \begin{pmatrix} \xi \\ \eta \end{pmatrix}. \tag{3.27}$$

However, Proposition 3.1 establishes a stronger result, because it relies only upon the assumption that $\eta = 0$ is an asymptotically stable equilibrium of the zero dynamics of the system, and this does not necessarily require the asymptotic stability of the linear approximation (i.e. all eigenvalues of $Q$ having negative real parts). In other words, the result may hold also in the case of some eigenvalues of $Q$ with zero real parts.

In order to design a stabilizing control law, there is no need to know explicitly the expression of the normal form, only that the system has zero dynamics with a locally asymptotically stable equilibrium at $\eta = 0$. In the original coordinates the stabilizing control law (3.22) assumes the form

$$u = \frac{1}{L_g L_f^{\rho-1} h(x)} \Big( - L_f^{\rho} h(x) - c_0 h(x) - c_1 L_f h(x) - \ldots - c_{\rho-1} L_f^{\rho-1} h(x) \Big). \tag{3.28}$$

If an output function is not defined, the zero dynamics is also not defined. However, it may happen that one is able to design a suitable dummy output whose associated zero dynamics have an asymptotically stable equilibrium. In this case a control law of the form discussed above will guarantee asymptotic stability.

**Example 3.2** Consider the system already discussed in Example 3.1. Its linear approximation at $x = 0$ is described by matrices $A$ and $B$ of the form

$$A = \left[\frac{\partial f}{\partial x}\right]_{x=0} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}, \qquad B = g(0) = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \tag{3.29}$$

and has one uncontrollable mode corresponding to the eigenvalue $\lambda = 0$. However, its zero dynamics

$$\dot{z}_3 = -z_3^3 \tag{3.30}$$

has an asymptotically stable equilibrium at $z_3 = 0$. Thus, from Proposition 3.1, the control law

$$u = -z_3 - 2z_1 z_2 - c_0 z_1 - c_1 z_2 \tag{3.31}$$

stabilizes the equilibrium $x = 0$. This control law, rewritten in the original coordinates, assumes the form

$$u = -x_3 - 2x_1(x_2 + x_1^2) - c_0 x_1 - c_1(x_2 + x_1^2) \tag{3.32}$$

### 3.2.2 Global Asymptotic Stabilization

Global stabilization of nonlinear systems has been intensively studied in the recent years (see [11],[64],[77],[119]). We consider in this section a special class of single-input single-output nonlinear systems which can be stabilized via state feedback. In particular the systems to be considered are those which can be transformed by means of a globally defined change of coordinates and/or feedback, into a system having the special normal form [44]

$$\begin{aligned} \dot{z} &= f_0(z, \xi_1) \\ \dot{\xi}_1 &= \xi_2 \\ &\vdots \\ \dot{\xi}_{\rho-1} &= \xi_\rho \\ \dot{\xi}_\rho &= u. \end{aligned} \tag{3.33}$$

Note that a system in the normal form of equation (3.4) can be converted via feedback into a system of the form

$$\begin{aligned} \dot{\xi}_1 &= \xi_2 \\ &\vdots \\ \dot{\xi}_{\rho-1} &= \xi_\rho \\ \dot{\xi}_\rho &= u \\ \dot{\eta} &= q(\xi, \eta) \end{aligned} \tag{3.34}$$

In addition, if the normal form (3.4) is defined globally, so also is the feedback yielding the normal form (3.34). The form of (3.33) is a special case of the normal form (3.34) in which $q(\xi, \eta)$ depends only upon the component $\xi_1$ of the vector $\xi$.

In order to simplify the analysis, consider initially the case $\rho = 1$. Then (3.33) has the form

$$\begin{aligned} \dot{z} &= f(z, \xi) \\ \dot{\xi} &= u \end{aligned} \tag{3.35}$$

with $(z,\xi) \in \Re^{n-1} \times \Re$, and $f(0,0) = 0$. Suppose the subsystem

$$\dot{z} = f(z,0) \tag{3.36}$$

has a *global asymptotically stable equilibrium* at $z = 0$. Then, in view of a converse Lyapunov theorem, there exists a smooth positive definite and radially unbounded function $V(z)$ such that $(\partial V/\partial z)f(z,0)$ is negative for each nonzero $z$. Using this property, it is easy to show that (3.35) can be stabilized globally and asymptotically. Observe that the function $f(z,\xi)$ can be written in the form

$$f(z,\xi) = f(z,0) + p(z,\xi)\xi \tag{3.37}$$

where $p(z,\xi)$ is a smooth function. It suffices to observe that the difference

$$\bar{f}(z,\xi) := f(z,\xi) - f(z,0) \tag{3.38}$$

is a smooth function vanishing at $\xi = 0$, and we can express $\bar{f}(z,\xi)$ as

$$\bar{f}(z,\xi) = \int_0^1 \frac{\partial \bar{f}(z,s\xi)}{\partial s} ds = \int_0^1 \left[\frac{\partial \bar{f}(z,\zeta)}{\partial \zeta}\right]_{\zeta=s\xi} \xi ds \tag{3.39}$$

Now consider the positive definite and radially unbounded function

$$W(z,\xi) = V(z) + \frac{1}{2}\xi^2 \tag{3.40}$$

and observe that

$$\begin{pmatrix} \frac{\partial W}{\partial z} & \frac{\partial W}{\partial \xi} \end{pmatrix} \begin{pmatrix} f(z,\xi) \\ u \end{pmatrix} = \frac{\partial V}{\partial z} f(z,0) + \frac{\partial V}{\partial z} p(z,\xi)\xi + \xi u \tag{3.41}$$

Choosing

$$u = u(z,\xi) = -\xi - \frac{\partial V}{\partial z} p(z,\xi) \tag{3.42}$$

yields

$$\begin{pmatrix} \frac{\partial W}{\partial z} & \frac{\partial W}{\partial \xi} \end{pmatrix} \begin{pmatrix} f(z,\xi) \\ u \end{pmatrix} < 0 \tag{3.43}$$

for all nonzero $(z,\xi)$. Therefore from Theorem A.1 it is concluded that the closed-loop system

$$\begin{aligned} \dot{z} &= f(z,\xi) \\ \dot{\xi} &= u(z,\xi) \end{aligned} \tag{3.44}$$

has a globally asymptotically stable equilibrium at $(z,\xi) = (0,0)$. In other words it has been shown that, if (3.36) has a globally asymptotically stable equilibrium at z=0, then the equilibrium $(z,\xi) = (0,0)$ of the system (3.35) can be rendered globally asymptotically stable by means of a smooth feedback law $u = u(z,\xi)$.

This result can be easily extended by showing that, for the purpose of stabilizing the equilibrium $(z,\xi) = (0,0)$ of (3.35), it suffices to assume that the equilibrium $z = 0$ of

$$\dot{z} = f(z,\xi) \tag{3.45}$$

is stabilizable by means of a smooth law $\xi = \alpha(z)$. In other words the state variable $\xi$ is used as a virtual control to stabilize (3.45) in a manner similar to the backstepping approach [44].

**Lemma 3.1** *Consider a system of the form (3.35). Suppose there exists a smooth real-valued function*

$$\xi = \alpha(z) \tag{3.46}$$

*with* $\alpha(0) = 0$ *and a smooth real-valued function* $V(z)$*, which is positive definite and radially unbounded, such that*

$$\frac{\partial V}{\partial z} f(z,\alpha(z)) < 0 \tag{3.47}$$

*for all nonzero* $z$*. Then there exists a smooth static feedback control law* $u = u(z,\xi)$ *with* $u(0,0) = 0$ *and a smooth real-valued function*

$$W(z,\xi) = V(z) + \frac{1}{2}\xi^2 \tag{3.48}$$

*which is positive definite and radially unbounded, such that*

$$\begin{pmatrix} \frac{\partial W}{\partial z} & \frac{\partial W}{\partial \xi} \end{pmatrix} \begin{pmatrix} f(z,\xi) \\ u(z,\xi) \end{pmatrix} < 0 \tag{3.49}$$

*for all nonzero* $(z,\xi)$*.*

**Proof.** It suffices to consider the globally defined change of variables

$$y = \xi - \alpha(z) \tag{3.50}$$

which transforms (3.35) into

$$\begin{aligned} \dot{z} &= f(z, y + \alpha(z)) \\ \dot{y} &= -\frac{\partial \alpha}{\partial z} f(z, y + \alpha(z)) + u. \end{aligned} \tag{3.51}$$

The feedback law

$$u = \frac{\partial \alpha}{\partial z} f(z, y + \alpha(z)) + u_1 \tag{3.52}$$

with $u_1$ an input to allow additional control actions, yields

$$\begin{aligned} \dot{z} &= f(z, y + \alpha(z)) \\ \dot{y} &= u_1 \end{aligned} \tag{3.53}$$

which satisfies the hypotheses on which the previous construction is based.

□

For the general case of nonlinear systems with relative degree $\rho > 1$, the repeated use of Lemma 3.1 allows one to derive the following result concerning global stabilization of a system in the form (3.33).

**Theorem 3.1** *Consider a system of the form (3.33). Suppose there exists a smooth real-valued function*

$$\xi = \alpha(z) \tag{3.54}$$

*with $\alpha(0) = 0$ and a smooth real valued function $V(z)$, which is positive definite and radially unbounded, such that*

$$\frac{\partial V}{\partial z} f_0(z, \alpha(z)) < 0 \tag{3.55}$$

*for all nonzero $z$. Then there exists a smooth static feedback control law*

$$u = u(z, \xi_1, \ldots, \xi_\rho) \tag{3.56}$$

*with $u(0) = 0$, which globally asymptotically stabilizes the equilibrium $(z, \xi_1, \ldots, \xi_\rho) = (0, 0, \ldots, 0)$ of the corresponding closed-loop system.*

A special case for which the results of Theorem 3.1 hold is when $\alpha(z) = 0$, i.e. when $\dot{z} = f_0(z, 0)$ has a globally asymptotically stable equilibrium at $z = 0$. This is the case of a system whose zero dynamics has a globally asymptotically stable equilibrium at $z = 0$, i.e. the case of a globally minimum phase system.

**Example 3.3** Consider the system (3.17) of Example 3.1. Its normal form, obtained by using the transformation (3.19), is

$$\begin{aligned} \dot{z}_1 &= z_2 \\ \dot{z}_2 &= z_3 + 2z_1 z_2 + u \\ \dot{z}_3 &= -z_3^3 + z_1(z_2 - z_1^2)^2 \end{aligned} \tag{3.57}$$

This system is an example of the special case for which Theorem 3.1 holds with $\alpha = 0$, because the zero dynamics

$$\dot{z}_3 = -z_3^3 \tag{3.58}$$

has a globally asymptotically stable equilibrium at $z_3 = 0$. Therefore the control law (3.32) designed in Example 3.2 globally stabilizes the equilibrium $x = 0$.

**Example 3.4** Consider the problem of stabilizing globally asymptotically the equilibrium $(x_1, x_2, x_3) = (0, 0, 0)$ of the nonlinear system

$$\begin{aligned} \dot{x}_1 &= x_2 + x_1^2 \\ \dot{x}_2 &= x_3 + x_2^2 x_1 \\ \dot{x}_3 &= u \end{aligned} \tag{3.59}$$

Note that this system has the triangular form (2.95) and thus the recursive control design of Section 2.3 can be used. However, for illustrative purposes, we will use the procedure explained above in this section for its stabilization. Since no output function has been specified, a "dummy output" of the form

$$y = x_3 - \alpha(x_1, x_2) \tag{3.60}$$

can be used. This output yields a relative degree $\rho = 1$ at each $x \in \Re^3$, and a two-dimensional zero dynamics. The dynamics obtained by imposing the constraint $y = 0$ on (3.59), is given by

$$\begin{aligned} \dot{x}_1 &= x_2 + x_1^2 \\ \dot{x}_2 &= \alpha(x_1, x_2) + x_2^2 x_1 \end{aligned} \tag{3.61}$$

From the discussion above, one must find, if possible, a function $\alpha(x_1, x_2)$ which globally stabilizes the equilibrium $(x_1, x_2) = (0, 0)$ of (3.61). Then there exists an input $u(x_1, x_2, x_3)$ that globally asymptotically stabilizes the equilibrium $(x_1, x_2, x_3) = (0, 0, 0)$ of (3.59). Noting the triangular form of (3.61), one can use the systematic procedure described in Section 2.3 to find the coordinate transformation

$$\begin{aligned} z_1 &= x_1 \\ z_2 &= x_2 + x_1^2 \end{aligned} \tag{3.62}$$

and the control law

$$\alpha(x_1, x_2) = -x_2^2 x_1 - 2x_1(x_2 + x_1^2) - c_0 x_1 - c_1(x_2 + x_1^2) \tag{3.63}$$

which globally asymptotically stabilizes the equilibrium $(x_1, x_2) = (0, 0)$ of (3.61). In fact, the closed-loop system expressed in the coordinates (3.62), has the form

$$\dot{z} = Az = \begin{pmatrix} 0 & 1 \\ -c_0 & -c_1 \end{pmatrix} z \tag{3.64}$$

where $c_0$, $c_1 > 0$. In order to obtain the input that globally stabilizes the equilibrium $x = 0$ of (3.59), it is necessary to use the construction indicated in the proof of Lemma 3.1. With the change of variables (3.60), the system (3.59) is transformed into

$$\begin{aligned} \dot{x}_1 &= x_2 + x_1^2 \\ \dot{x}_2 &= y + \alpha(x_1, x_2) + x_2^2 x_1 \\ \dot{y} &= u - \frac{\partial \alpha}{\partial x_1}(x_2 + x_1^2) - \frac{\partial \alpha}{\partial x_2}\Big(y + \alpha(x_1, x_2) + x_2^2 x_1\Big) \end{aligned} \tag{3.65}$$

Choosing a preliminary feedback

$$u = \frac{\partial \alpha}{\partial x_1}(x_2 + x_1^2) + \frac{\partial \alpha}{\partial x_2}\Big(y + \alpha(x_1, x_2) + x_2^2 x_1\Big) + u_1 \tag{3.66}$$

yields

$$\dot{x}_1 = x_2 + x_1^2$$

$$\dot{x}_2 = y + \alpha(x_1, x_2) + x_2^2 x_1 \tag{3.67}$$

$$\dot{y} = u_1 \tag{3.68}$$

which has the form of (3.35)

$$\begin{aligned} \dot{z} &= f(z, \xi) \\ \dot{\xi} &= u_1 \end{aligned} \tag{3.69}$$

with

$$z = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, \qquad f(z, \xi) = \begin{pmatrix} x_2 + x_1^2 \\ \alpha(x_1, x_2) + x_2^2 x_1 \end{pmatrix} + \xi \begin{pmatrix} 0 \\ 1 \end{pmatrix} \tag{3.70}$$

$\dot{z} = f(z, 0)$ has a globally asymptotically stable equilibrium at $z = 0$. As a consequence this system can be globally asymptotically stabilized by means of a feedback control law $u_1 = u_1(z, \xi)$ similar to (3.42).

## 3.3 Dynamical Feedback Stabilization

As shown in Section 2.2 the study of transformations of affine nonlinear systems

$$\dot{x} = f(x) + g(x)u \tag{3.71}$$

with $f(0) = 0$ into linear controllable systems

$$\dot{z} = Az + Bv \tag{3.72}$$

provides a classification of nonlinear systems, and a simplification of the analysis and control of those systems (3.71) which are transformable into the form (3.72).

A natural generalization of static feedback transformations is given by *dynamic* state feedback transformations

$$\begin{aligned} \dot{\omega} &= a(x,\omega) + b(x,\omega) \\ u &= \alpha(x,\omega) + \beta(x,\omega) \end{aligned} \tag{3.73}$$

Dynamical compensation was introduced in [102] and also investigated in [24] in the study of input-output decoupling of nonlinear systems with outputs $y = h(x)$. In [44] sufficient conditions were given for which a system (3.71) with outputs can be simultaneously input-output decoupled and linearized by a dynamic compensator. Charlet *et al* [14, 15] addressed the problem of transforming a nonlinear multi-input system without outputs into a linear controllable one via dynamic feedback and extended state space diffeomorphism. They also gave sufficient conditions for which a class of multi-input systems is dynamic feedback linearizable.

In this section dynamical feedback stabilization via input-output linearization of nonlinear systems with relative degree less than the system order is presented. Consider a single-input single-output nonlinear system

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= h(x) \end{aligned} \tag{3.74}$$

where $f(x)$ and $g(x)$ are smooth vector fields defined on an open set $R_0 \subset \Re^n$, and $h(x)$ is a smooth scalar function also defined on $R_0$. Assume that the origin $x = 0$ is an equilibrium point of the vector field $f(x)$, i.e. $f(0) = 0$ and $h(0) = 0$. Assume also that system (3.74) has relative degree $\rho < n$ at $x = 0$. Consider the operator

$$\begin{aligned} \mathcal{L}_h^0(x) &= h(x) \\ \mathcal{L}_h^i(x) &= \frac{\partial\left(\mathcal{L}_h^{i-1}(x)\right)}{\partial x} f(x) \qquad\qquad 1 \leqslant i \leqslant \rho - 1 \\ \mathcal{L}_h^\rho(x, v_1) &= \frac{\partial\left(\mathcal{L}_h^{\rho-1}(x)\right)}{\partial x}\Big(f(x) + g(x)v_1\Big) \\ \mathcal{L}_h^i(x, v_1, \ldots, v_{i-\rho+1}) &= \frac{\partial\left(\mathcal{L}_h^{i-1}(x, v_1, \ldots, v_{i-\rho})\right)}{\partial x}\Big(f(x) + g(x)v_1\Big) \\ &\quad + \sum_{j=1}^{i-\rho} \frac{\partial\left(\mathcal{L}_h^{i-1}(x, v_1, \ldots, v_{i-\rho})\right)}{\partial v_j} v_{j+1} \quad \rho + 1 \leqslant i \leqslant n - 1 \end{aligned} \tag{3.75}$$

defined on an extended state space $(x, v) \in \Re^n \times \Re^{n-\rho}$. This operator allows one to express the output $y$ and its first $n-1$ derivatives as functions of $x$, $u$ and the derivatives of $u$

$$\Phi(x, v_1, \ldots, v_{n-\rho}) = \begin{bmatrix} y \\ y^{(1)} \\ \vdots \\ y^{(n-1)} \end{bmatrix} = \begin{bmatrix} \mathcal{L}_h^0 \\ \mathcal{L}_h^1 \\ \vdots \\ \mathcal{L}_h^{n-1} \end{bmatrix}. \tag{3.76}$$

**Assumption 3.1** *The system (3.74) is* observable, *i.e. the mapping (3.76) satisfies the rank condition*

$$\text{rank}\frac{\partial \Phi(\cdot)}{\partial x} = n \tag{3.77}$$

$\forall x \in R_0$ *and* $\forall v_1, \ldots, v_{n-p}$.

**Assumption 3.2** *The system (3.74) is* minimum phase *in* $R_0$.

**Lemma 3.2** *With Assumption 3.1, the mapping (3.76) is a local change of coordinates which places system (3.74) into the form*

$$\begin{aligned} \dot{z}_1 &= z_2 \\ \dot{z}_2 &= z_3 \\ &\vdots \\ \dot{z}_{n-1} &= z_n \\ \dot{z}_n &= a_{n-\rho}(z, u, u^{(1)}, \ldots, u^{(n-\rho-1)}) + b(z)u^{(n-\rho)} \\ y &= z_1 \end{aligned} \tag{3.78}$$

*with* $b(z) \neq 0$ *in* $R_0$.

**Proof.** Recall from Section 2.2.3 that an independent set of functions can be formed by the output and its first $\rho - 1$ derivatives $\{y, y^{(1)}, \ldots, y^{(\rho-1)}\}$. It can be used as a partial coordinate transformation to place system (3.74) into the form

$$\begin{aligned} \dot{z}_1 &= z_2 \\ \dot{z}_2 &= z_3 \\ &\vdots \\ \dot{z}_{\rho-1} &= z_\rho \\ \dot{z}_\rho &= a(x) + b(x)u \end{aligned} \tag{3.79}$$

with

$$a(x) = L_f^\rho h(x), \quad b(x) = L_g L_f^{\rho-1} h(x)$$

Using the set $\{y, y^{(1)}, \ldots, y^{(\rho-1)}\}$ as the first $\rho$ functions of the mapping (3.76), the remaining $n-\rho$ functions are defined by using the operator (3.75)

$$\begin{aligned} z_{\rho+1} &= a(x) + b(x)u \\ z_{\rho+1+i} &= a_i(x, u, \ldots, u^{(i-1)}) + b(x)u^{(i)} \quad 1 \leqslant i \leqslant n-\rho-1 \end{aligned} \tag{3.80}$$

where

$$a_i(x, u, \ldots, u^{(i-1)}) := \frac{da_{i-1}}{dt} + \frac{db}{dt}u^{(i-1)} \tag{3.81}$$

In view of Assumption 3.1 the mapping $z = (z_1, \ldots, z_n)^T = \Phi(x, u, \ldots, u^{(n-\rho-1)})$ constitutes an invertible local change of coordinates. The proof is completed by observing that the relations (3.80) are obtained by computing the time derivative of the previous relation (derivative of the output), and noting that

$$a_{n-\rho}(z, u, \ldots, u^{(n-\rho-1)}) = \frac{da_{n-\rho-1}\left(\Phi^{-1}(z)\right)}{dt} + \frac{db\left(\Phi^{-1}(z)\right)}{dt}u^{(n-\rho-1)} \tag{3.82}$$

$$b(z) = \Big[b(x)\Big]_{x=\Phi^{-1}(z)} \tag{3.83}$$

□

It is now possible to state the following theorem.

**Theorem 3.2** *With Assumptions 3.1 and 3.2 there exists a dynamical controller*

$$\begin{aligned} \dot{v}_1 &= v_2 \\ &\vdots \\ \dot{v}_{n-\rho-1} &= v_{n-\rho} \\ \dot{v}_{n-\rho} &= -\frac{1}{b(z)}\left(a_{n-\rho}(z, v_1, \ldots, v_{n-\rho-1}) + \sum_{i=1}^{n} c_{i-1}z_i - \tilde{u}\right) \\ u &= v_1 \end{aligned} \tag{3.84}$$

*with $\tilde{u}$, a new input, which locally linearizes the dynamics in the $z$ coordinates with eigenvalues arbitrarily placed.*

**Proof.** Assumption 3.2 guarantees the asymptotic stability of the internal dynamics of system (3.74). Then Assumption 3.1 allows one to transform the system (3.74) into (3.78) by virtue of Lemma 3.2. So, for the system (3.78) the dynamical controller

$$u^{(n-\rho)} = -\frac{1}{b(z)}\left(a_{n-\rho}(z, v_1, \ldots, v_{n-\rho-1}) + \sum_{i=1}^{n} c_{i-1}z_i - \tilde{u}\right) \tag{3.85}$$

yields the linearized closed-loop system in the $z$ coordinates

$$\dot{z} = Az + B\tilde{u} = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \cdot & \cdot & \cdot & \dots & \cdot \\ 0 & 0 & 0 & \dots & 1 \\ -c_0 & -c_1 & -c_2 & \dots & -c_{n-1} \end{pmatrix} z + \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix} \tilde{u} \tag{3.86}$$

where the design parameters $c_i$, $i = 0, \dots, n-1$, can be selected to assign the roots of the characteristic polynomial

$$p(s) = c_0 + c_1 s + \dots + c_{n-1} s^{n-1} + s^n \tag{3.87}$$

at arbitrary locations of the open left-half complex plane. The proof is completed by defining in (3.85) and (3.78) the extended coordinates

$$v_i = u^{(i-1)}, \qquad 1 \leqslant i \leqslant n - \rho \tag{3.88}$$

which characterizes the set of state variables of the dynamical compensator (3.84).

□

**Example 3.5** Consider the nonlinear system of Example 3.1

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{pmatrix} = \begin{pmatrix} x_2 + x_1^2 \\ x_3 \\ x_1 x_2^2 - x_3^3 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} u \tag{3.89}$$

$$y = x_1.$$

This system has relative degree $\rho = 2$. Therefore using the partial transformation

$$\begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \begin{pmatrix} y \\ y^{(1)} \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 + x_1^2 \end{pmatrix} \tag{3.90}$$

one obtains

$$\begin{aligned} \dot{z}_1 &= z_2 \\ \dot{z}_2 &= x_3 + 2x_1(x_2 + x_1^2) + u. \end{aligned} \tag{3.91}$$

It was shown in Example 3.1 that the system (3.89) is globally minimum phase. Taking

$$z_3 = x_3 + 2x_1(x_2 + x_1^2) + u \tag{3.92}$$

gives a globally defined control-dependent transformation

$$z = \Phi(x,u) = \begin{pmatrix} x_1 \\ x_2 + x_1^2 \\ x_3 + 2x_1(x_2 + x_1^2) + u \end{pmatrix} \tag{3.93}$$

satisfying the observability condition (3.77). Its inverse is defined as

$$x = \Phi^{-1}(z,u) = \begin{pmatrix} z_1 \\ z_2 - z_1^2 \\ z_3 - 2z_1z_2 - u \end{pmatrix} \tag{3.94}$$

The transformed system takes the form

$$\begin{aligned} \dot{z}_1 &= z_2 \\ \dot{z}_2 &= z_3 \\ \dot{z}_3 &= z_1(z_2 - z_1^2)^2 - (z_3 - 2z_1z_2 - u)^3 + 2(z_2^2 + z_1z_3) + \dot{u} \end{aligned} \tag{3.95}$$

Then the dynamical controller

$$\dot{u} = -z_1(z_2 - z_1^2)^2 + (z_3 - 2z_1z_2 - u)^3 - 2(z_2^2 + z_1z_3) - c_0z_1 - c_1z_2 - c_2z_3 \tag{3.96}$$

linearizes system (3.95) and places its eigenvalues in desired positions of the open left-half complex plane. Hence the equilibrium $(x_1, x_2, x_3) = (0,0,0)$ is globally asymptotically stabilized.

This systematic procedure is suitable for affine minimum phase nonlinear systems which satisfy the observability condition (3.77). Zeitz [125] analysed the representation of *non-affine* systems

$$\begin{aligned} \dot{x} &= f(x,u) \\ y &= h(x,u) \end{aligned} \tag{3.97}$$

into an extended state space formed by a set of higher-order differential equations involving the inputs, outputs and their respective derivatives. A similar approach was proposed in [123] by van der Schaft. The common idea of both approaches is based upon successive differentiation of the outputs and the extension of the system (3.97) by incorporating equations depending on the inputs, outputs and their derivatives. In addition, assuming that the system (3.95) satisfies an observability condition defined by using an operator similar to (3.75), Zeitz introduced the following *Generalized Observability Canonical Form*

$$\begin{aligned} \dot{z}_1 &= z_2 \\ \dot{z}_2 &= z_3 \\ &\vdots \\ \dot{z}_{n-1} &= z_n \\ \dot{z}_n &= c(z, u, u^{(1)}, \ldots, u^{(n-\rho)}) \end{aligned} \tag{3.98}$$

Fliess ([26]-[28]) obtained the same canonical form by using a differential algebraic approach. In this setting new generalized controller canonical forms for linear and nonlinear systems were proposed, for which dynamical compensators can be designed to achieve local (global) stabilization. Indeed the relation

$$c(z, u, u^{(1)}, \ldots, u^{(n-\rho)}) = -\sum_{i=1}^{n} c_{i-1} z_i + v \tag{3.99}$$

defines a linearizing dynamic state feedback for systems in the form (3.98), provided that the non-singularity condition

$$\frac{\partial c(z, u, u^{(1)}, \ldots, u^{(n-\rho)})}{\partial u^{(n-\rho)}} \neq 0 \tag{3.100}$$

is satisfied, at least locally. Note that the condition (3.100) is always satisfied, by construction, for the class of affine nonlinear systems (3.74).

The problem of stabilizing nonlinear systems via dynamical feedback control has attracted the attention of many researchers (see, for instance, [20, 124]). Recently a lot of attention has been paid to *global* and *semi-global* stabilization of nonlinear systems via dynamical state feedback and output feedback ([77, 118, 119, 121]).

## 3.4 Dynamical Deterministic Backstepping (DDB) Algorithm

In this section we describe our new extension to the SDB algorithm given in Section 2.3.1 for triangular systems. The DDB algorithm adopts the same systematic procedure of the SDB algorithm, which is based upon Lyapunov-based control design, and is applicable to the wider class of deterministic observable minimum phase nonlinear systems. The controlled plant does *not* need to be transformable to a triangular form or contain a cascade form. The stability results are in general local, but global results can be achieved if all the assumed conditions are satisfied globally.

Consider a single-input single-output nonlinear system

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= h(x) \end{aligned} \tag{3.101}$$

where $f(x)$ and $g(x)$ are smooth vector fields defined on an open set $R_0 \subset \Re^n$, and $h(x)$ is a smooth scalar function also defined on $R_0$. Assume that the origin $x = 0$ is an equilibrium point of the vector field $f(x)$, i.e. $f(0) = 0$ and $h(0) = 0$. Assume also that the system (3.101) has relative degree $\rho < n$ at $x = 0$. If the system (3.101) satisfies the Assumptions 3.1 and 3.2, the following DDB algorithm can be used for regulation of the output $h(x)$ to a desired set point $y_r$ via a dynamical controller.

***DDB Algorithm***

*Coordinate transformation*

$$\begin{aligned} z_1 &= h_0(x) = y - y_r = h(x) - y_r \\ z_j &= h_{j-1}(x) = z_{j-2} + \frac{\partial h_{j-2}}{\partial x} f(x) + c_{j-1} z_{j-1} \qquad 2 \leqslant j \leqslant \rho \\ z_k &= h_{k-1}(x, v_1, \ldots, v_{k-\rho}) = z_{k-2} + \frac{\partial h_{k-2}}{\partial x}\Big(f(x) + g(x)v_1\Big) + \sum_{i=1}^{k-\rho-1} \frac{\partial h_{k-2}}{\partial v_i} v_{i+1} \\ &\qquad + c_{k-1} z_{k-1} \qquad \rho + 1 \leqslant k \leqslant n \end{aligned} \tag{3.102}$$

*Dynamical control law*

$$\begin{aligned} \dot{v}_1 &= v_2 \\ \dot{v}_2 &= v_3 \\ &\vdots \\ \dot{v}_{n-\rho-1} &= v_{n-\rho} \\ \dot{v}_{n-\rho} &= \left(\frac{\partial h_{n-1}}{\partial v_{n-\rho}}\right)^{-1} \left[ -z_{n-1} - \frac{\partial h_{n-1}}{\partial x}\Big(f(x) + g(x)v_1\Big) - \sum_{i=1}^{n-\rho-1} \frac{\partial h_{n-1}}{\partial v_i} v_{i+1} - c_n z_n \right] \end{aligned} \tag{3.103}$$

Since we consider affine minimum phase nonlinear systems, the condition

$$\frac{\partial h_{n-1}}{\partial v_{n-\rho}} = \frac{\partial h_{\rho-1}}{\partial x} g(x) \neq 0 \tag{3.104}$$

is in general satisfied locally. The following steps describe the procedure leading to the general DDB algorithm above.

**Step 1.** Choose $z_1 = h_0 := h(x) - y_r$ as the first new state coordinate. The first subsystem is defined as

$$\dot{z}_1 = \frac{\partial h_0}{\partial x}\Big(f(x) + g(x)u\Big) \tag{3.105}$$

If the relative degree $\rho > 1$,

$$\frac{\partial h_0}{\partial x} g(x) = 0 \tag{3.106}$$

and (3.105) can be rewritten as

$$\dot{z}_1 = \frac{\partial h_0}{\partial x} f(x). \tag{3.107}$$

We stabilize (3.107) with respect to the quadratic Lyapunov function

$$V_1 = \frac{1}{2} z_1^2. \tag{3.108}$$

The time derivative of $V_1$ is

$$\dot{V}_1 = z_1 \left(\frac{\partial h_0}{\partial x} f(x)\right) \tag{3.109}$$

If the bracketed term multiplying $z_1$ equals $-c_1 z_1$, i.e.

$$\frac{\partial h_0}{\partial x} f(x) = -c_1 z_1 \tag{3.110}$$

where $c_1$ is a positive constant, we can achieve $\dot{V}_1 = -c_1 z_1^2$. However, since (3.110) is not satisfied, a new coordinate is defined as

$$z_2 = h_1(x) := \frac{\partial h_0}{\partial x} f(x) + c_1 z_1 \tag{3.111}$$

Then $\dot{z}_1$ becomes

$$\dot{z}_1 = -c_1 z_1 + z_2 \tag{3.112}$$

and

$$\dot{V}_1 = -c_1 z_1^2 + z_1 z_2 \tag{3.113}$$

The term $z_1 z_2$ in (3.113) will be compensated at the next step.

**Step j** $(2 \leqslant j \leqslant \rho - 1)$. By induction, this general $j$-th step characterizes the first $\rho - 1$ steps in which the control input $u$ does not appear explicitly. By obtaining the time derivative of $z_j$, the previous subsystem is augmented with

$$\dot{z}_j = \frac{\partial h_{j-1}}{\partial x}\Big(f(x) + g(x)u\Big) = \frac{\partial h_{j-1}}{\partial x} f(x) \tag{3.114}$$

which will be stabilized with respect to the augmented Lyapunov function

$$V_j = V_{j-1} + \frac{1}{2} z_j^2 \tag{3.115}$$

The time derivative of $V_j$ is

$$\dot{V}_j = -\sum_{i=1}^{j-1} c_i z_i^2 + z_j \left[ z_{j-1} + \frac{\partial h_{j-1}}{\partial x} f(x) \right] \tag{3.116}$$

If the bracketed term multiplying $z_j$ equals $-c_j z_j$, i.e.

$$z_{j-1} + \frac{\partial h_{j-1}}{\partial x} f(x) = -c_j z_j \tag{3.117}$$

where $c_j$ is a positive constant, we can achieve $\dot{V} = -\sum_{i=1}^{j} c_i z_i^2$. However, since (3.117) is not satisfied, a new coordinate is defined as

$$z_{j+1} = h_j := z_{j-1} + \frac{\partial h_{j-1}}{\partial x} f(x) + c_j z_j \tag{3.118}$$

Then $\dot{z}_j$ becomes

$$\dot{z}_j = -z_{j-1} - c_j z_j + z_{j+1} \tag{3.119}$$

and

$$\dot{V}_j = -\sum_{i=1}^{j} c_i z_i^2 + z_j z_{j+1} \tag{3.120}$$

**Step k** ($\rho \leqslant k \leqslant n-1$). This general $k$-th step characterizes the steps in which the control input $u$ and its derivatives appear explicitly. By obtaining the time derivative of $z_k$ the previous subsystem is augmented with

$$\dot{z}_k = \frac{\partial h_{k-1}}{\partial x}\Big( f(x) + g(x) v_1 \Big) + \sum_{i=1}^{k-\rho} \frac{\partial h_{k-1}}{\partial v_i} v_{i+1} \tag{3.121}$$

Note that we have substituted in (3.121) the control $u$ and its derivatives $\dot{u}, \ddot{u}, \ldots$ by the state variables $v_1$, $v_2, \ldots$ respectively, which define the dynamical compensator (3.103). The subsystem (3.121) can be stabilized with respect to the augmented Lyapunov function

$$V_k = V_{k-1} + \frac{1}{2} z_k^2 \tag{3.122}$$

The time derivative of $V_k$ is

$$\dot{V}_k = -\sum_{i=1}^{k-1} c_i z_i^2 + z_k \left[ z_{k-1} + \frac{\partial h_{k-1}}{\partial x}\Big( f(x) + g(x) v_1 \Big) + \sum_{i=1}^{k-\rho} \frac{\partial h_{k-1}}{\partial v_i} v_{i+1} \right] \tag{3.123}$$

If the bracketed term multiplying $z_k$ equals $-c_k z_k$, i.e.

$$z_{k-1} + \frac{\partial h_{k-1}}{\partial x}\Big( f(x) + g(x) v_1 \Big) + \sum_{i=1}^{k-\rho} \frac{\partial h_{k-1}}{\partial v_i} v_{i+1} = -c_k z_k \tag{3.124}$$

where $c_k$ is a positive design parameter, we can achieve $\dot{V} = -\sum_{i=1}^{k} c_i z_i^2$. However, since (3.124) is not satisfied, a new coordinate is defined as

$$z_{k+1} = h_k := z_{k-1} + \frac{\partial h_{k-1}}{\partial x}\Big(f(x) + g(x)v_1\Big) + \sum_{i=1}^{k-\rho} \frac{\partial h_{k-1}}{\partial v_i} v_{i+1} + c_k z_k \tag{3.125}$$

Then $\dot{z}_k$ becomes

$$\dot{z}_k = -z_{k-1} - c_k z_k + z_{k+1} \tag{3.126}$$

and

$$\dot{V}_k = -\sum_{i=1}^{k} c_i z_i^2 + z_k z_{k+1} \tag{3.127}$$

**Step n.** At this final step the time derivative of $z_n$ is

$$\dot{z}_n = \frac{\partial h_{n-1}}{\partial x}\Big(f(x) + g(x)v_1\Big) + \sum_{i=1}^{n-\rho-1} \frac{\partial h_{n-1}}{\partial v_i} v_{i+1} + \frac{\partial h_{n-1}}{\partial v_{n-\rho}} \dot{v}_{n-\rho} \tag{3.128}$$

Then the whole system can be stabilized with respect to the augmented Lyapunov function

$$V_n = V_{n-1} + \frac{1}{2} z_n^2. \tag{3.129}$$

The time derivative of $V_n$ is

$$\dot{V}_n = -\sum_{i=1}^{n-1} c_i z_i^2 + z_n \left[ z_{n-1} + \frac{\partial h_{n-1}}{\partial x}\Big(f(x) + g(x)v_1\Big) + \sum_{i=1}^{n-\rho-1} \frac{\partial h_{n-1}}{\partial v_i} v_{i+1} + \frac{\partial h_{n-1}}{\partial v_{n-\rho}} \dot{v}_{n-\rho} \right] \tag{3.130}$$

Now, a linearizing control law

$$z_{n-1} + \frac{\partial h_{n-1}}{\partial x}\Big(f(x) + g(x)v_1\Big) + \sum_{i=1}^{n-\rho-1} \frac{\partial h_{n-1}}{\partial v_i} v_{i+1} + \frac{\partial h_{n-1}}{\partial v_{n-\rho}} \dot{v}_{n-\rho} = -c_n z_n \tag{3.131}$$

can be chosen to cancel the nonlinearities and make the bracketed term multiplying $z_n$ equal to $-c_n z_n$, where $c_n$ is a positive constant. Thus

$$\dot{V}_n = -\sum_{i=1}^{n} c_i z_i^2 \tag{3.132}$$

and the closed-loop system is

$$\dot{z} = A_z z \tag{3.133}$$

with

$$A_z = \begin{bmatrix} -c_1 & 1 & 0 & \dots & 0 & 0 \\ -1 & -c_2 & 1 & \dots & 0 & 0 \\ 0 & -1 & -c_3 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & -c_{n-1} & 1 \\ 0 & 0 & 0 & \dots & -1 & -c_n \end{bmatrix} \tag{3.134}$$

The control law (3.131) completes the design of the dynamical compensator (3.103).

**Example 3.6** Consider the problem of stabilizing the origin of the system of Example 3.1

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{pmatrix} = \begin{pmatrix} x_2 + x_1^2 \\ x_3 \\ x_1 x_2^2 - x_3^3 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} u \qquad (3.135)$$
$$y = x_1$$

This system has relative degree $\rho = 2$ and is globally minimum phase, as shown in Example 3.1. After applying our DDB algorithm we obtain the globally defined control-dependent transformation

$$\begin{aligned} z_1 &= h_0(x) = x_1 \\ z_2 &= h_1(x) = x_2 + x_1^2 + c_1 x_1 \qquad (3.136) \\ z_3 &= h_2(x, v_1) = x_1 + (2x_1 + c_1)(x_2 + x_1^2) + x_3 + v_1 + c_2(x_2 + x_1^2 + c_1 x_1) \end{aligned}$$

Therefore the observability condition is satisfied globally and the inverse of the transformation (3.136) is defined as

$$\begin{aligned} x_1 &= z_1 \\ x_2 &= z_2 - z_1^2 - c_1 z_1 \qquad (3.137) \\ x_3 &= z_3 - z_1 - (2z_1 + c_1)(z_2 - c_1 z_1) - v_1 - c_2 z_2 \end{aligned}$$

The dynamical controller is obtained as

$$\dot{v}_1 = -z_2 - \frac{\partial h_2}{\partial x_1}(x_2 + x_1^2) - \frac{\partial h_2}{\partial x_2}(x_3 + v_1) - x_1 x_2^2 + x_3^3 - c_3 z_3 \qquad (3.138)$$

with

$$\begin{aligned} \frac{\partial h_2}{\partial x_1} &= 1 + 2(x_2 + x_1^2) + (2x_1 + c_1)(2x_1 + c_2) \qquad (3.139) \\ \frac{\partial h_2}{\partial x_2} &= 2x_1 + c_1 + c_2 \end{aligned}$$

Computer simulations were performed to assess the performance of the dynamical controller (3.138) in the regulation of the origin of the system (3.135). Figures 3.1 and 3.2 show the asymptotic convergence of the state variables to the origin for the initial conditions $(x(0), v_1(0)) = (4, 3, 5, 0.5)$ and $(x(0), v_1(0)) = (8, 6, 10, 0.5)$ respectively. The design parameters used for both computer simulations were $c_1 = 4$, $c_2 = 3$ and $c_3 = 10$.

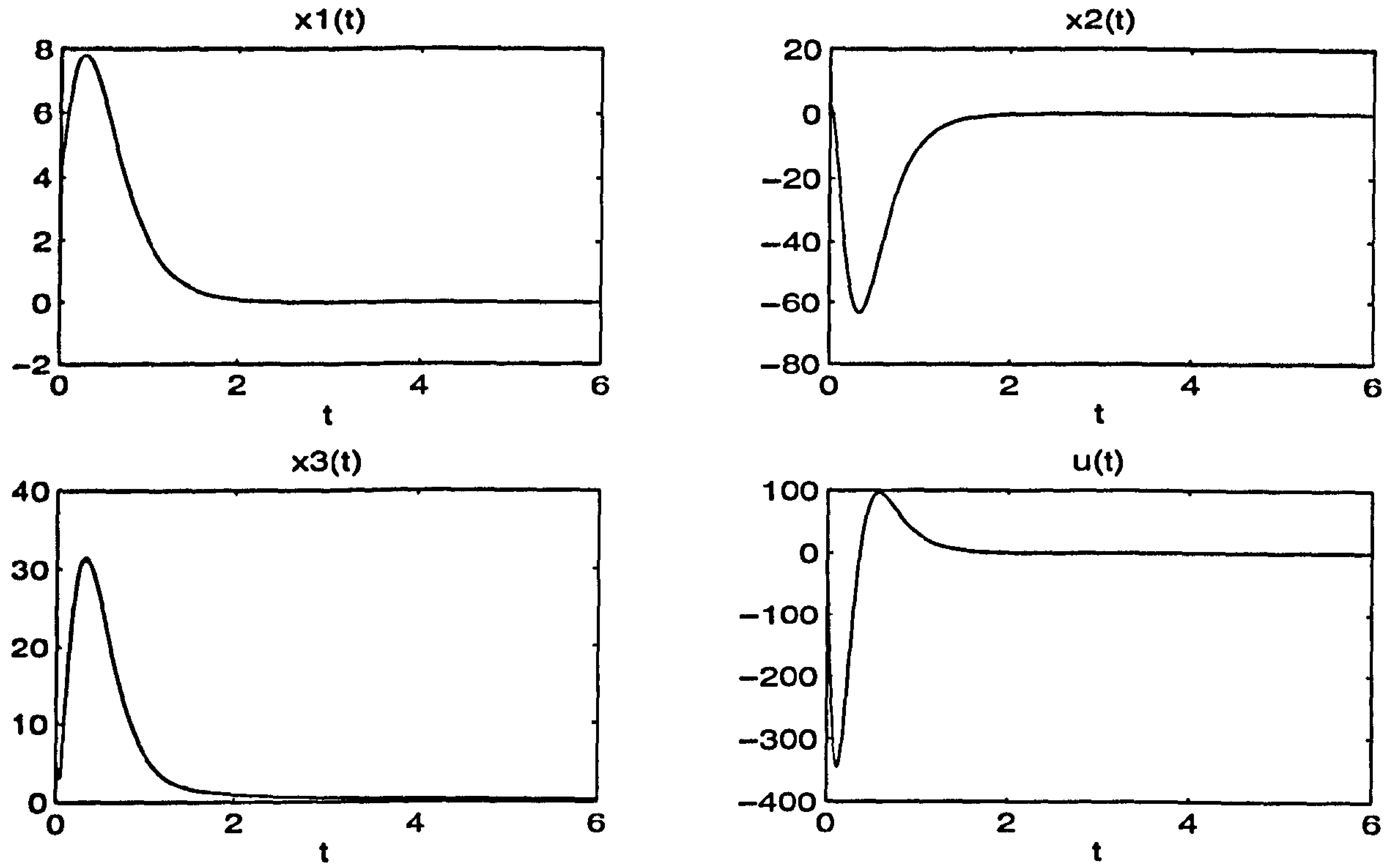


Figure 3.1: Controlled responses of the state variables of a triangular system

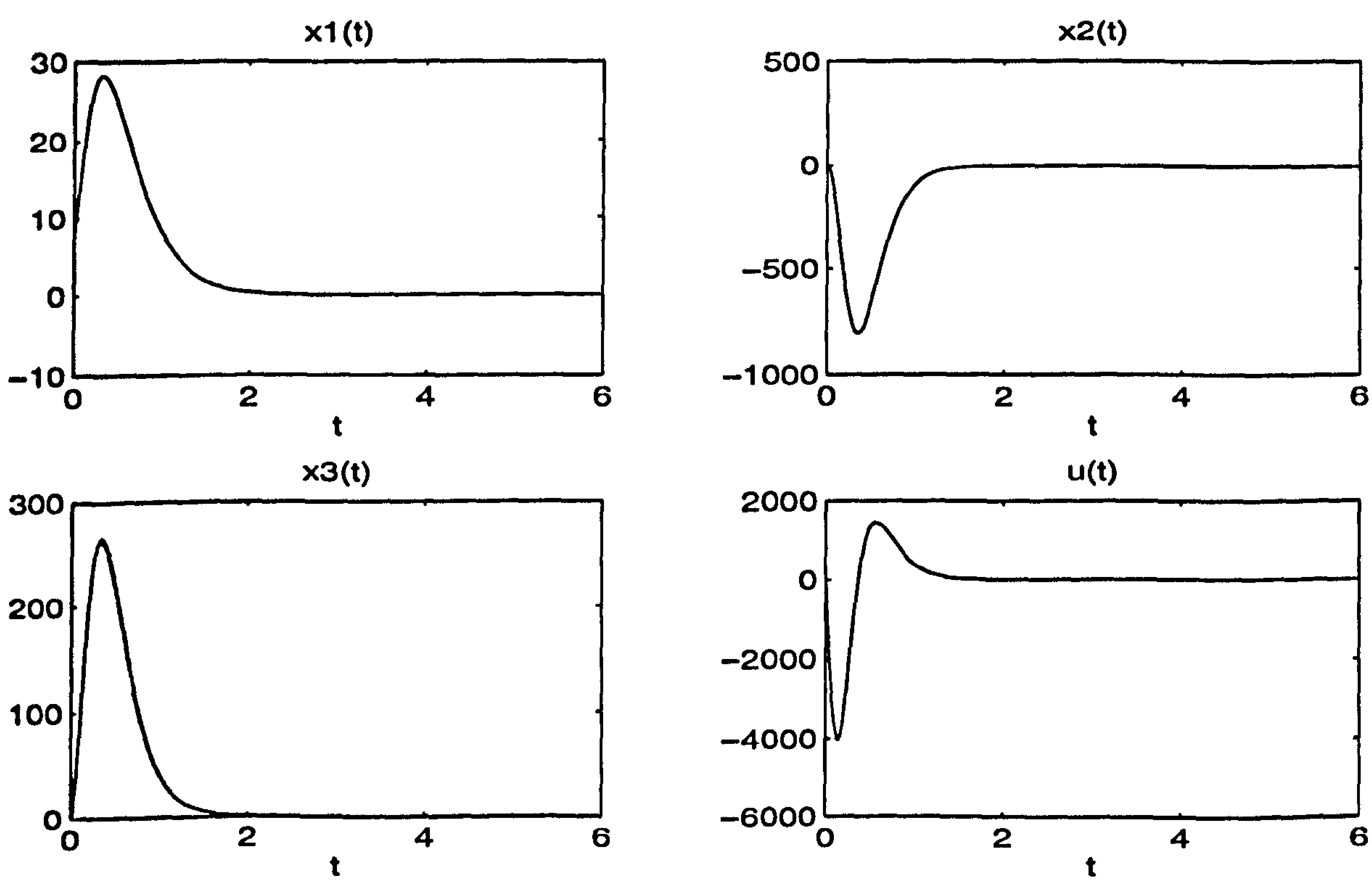


Figure 3.2: Controlled responses of the state variables of a triangular system for different initial conditions

**Example 3.7** We consider now an example corresponding to a nonlinear system in *nontriangular* form

$$\begin{aligned} \dot{x}_1 &= x_2 + x_1 x_3 \\ \dot{x}_2 &= x_3 \\ \dot{x}_3 &= u \end{aligned} \tag{3.140}$$

It was shown by Marino and Tomei [75] that this system is input-output linearizable locally with respect to the nonlinear output

$$y = x_1 \exp(-x_2) \tag{3.141}$$

We use here the output function (3.141) and apply our DDB algorithm to design a linearizing controller. We obtain the following nonlinear transformation

$$\begin{aligned} z_1 &= h_0(x) = x_1 \exp(-x_2) - y_r \\ z_2 &= h_1(x) = x_2 \exp(-x_2) + c_1\Big(x_1 \exp(-x_2) - y_r\Big) \\ z_3 &= h_2(x, v_1) = x_1 \exp(-x_2) + x_3(1 - x_2)\exp(-x_2) + c_1 x_2 \exp(-x_2) \\ &\qquad + c_2\Big[x_2 \exp(-x_2) + c_1\Big(x_1 \exp(-x_2) - y_r\Big)\Big] \end{aligned} \tag{3.142}$$

and the *static* feedback control

$$u = \left(\frac{\partial h_2}{\partial x_3}\right)^{-1} \left[-z_2 - \frac{\partial h_2}{\partial x_1}(x_2 + x_1 x_3) - \frac{\partial h_2}{\partial x_2} x_3 - c_3 z_3\right] \tag{3.143}$$

with

$$\begin{aligned} \frac{\partial h_2}{\partial x_1} &= (1 + c_1 c_2)\exp(-x_2) \\ \frac{\partial h_2}{\partial x_2} &= \exp(-x_2)\Big(-x_1 - x_3(2 - x_2) + c_1(1 - x_2) + c_2(1 - x_2 - c_1 x_1)\Big) \\ \frac{\partial h_2}{\partial x_3} &= \exp(-x_2)(1 - x_2) \end{aligned} \tag{3.144}$$

Note that, as shown by Marino and Tomei [75], the value $x_2 = 1$ is a singular value for the feedback control law (3.143). Therefore local input-output linearization is achieved. Computer simulations were carried out for the stabilization of the system (3.140) to the origin. Figure 3.3 shows the asymptotic controlled responses of the state variables for the design parameters $c_1 = 2$, $c_2 = 2$ and $c_3 = 3$. A different equilibrium point $x = (2, 0, 0)$ was also considered. Figure 3.4 shows the asymptotic behaviour of the state variables to this equilibrium point using the same initial conditions and design parameters used in Figure 3.3.

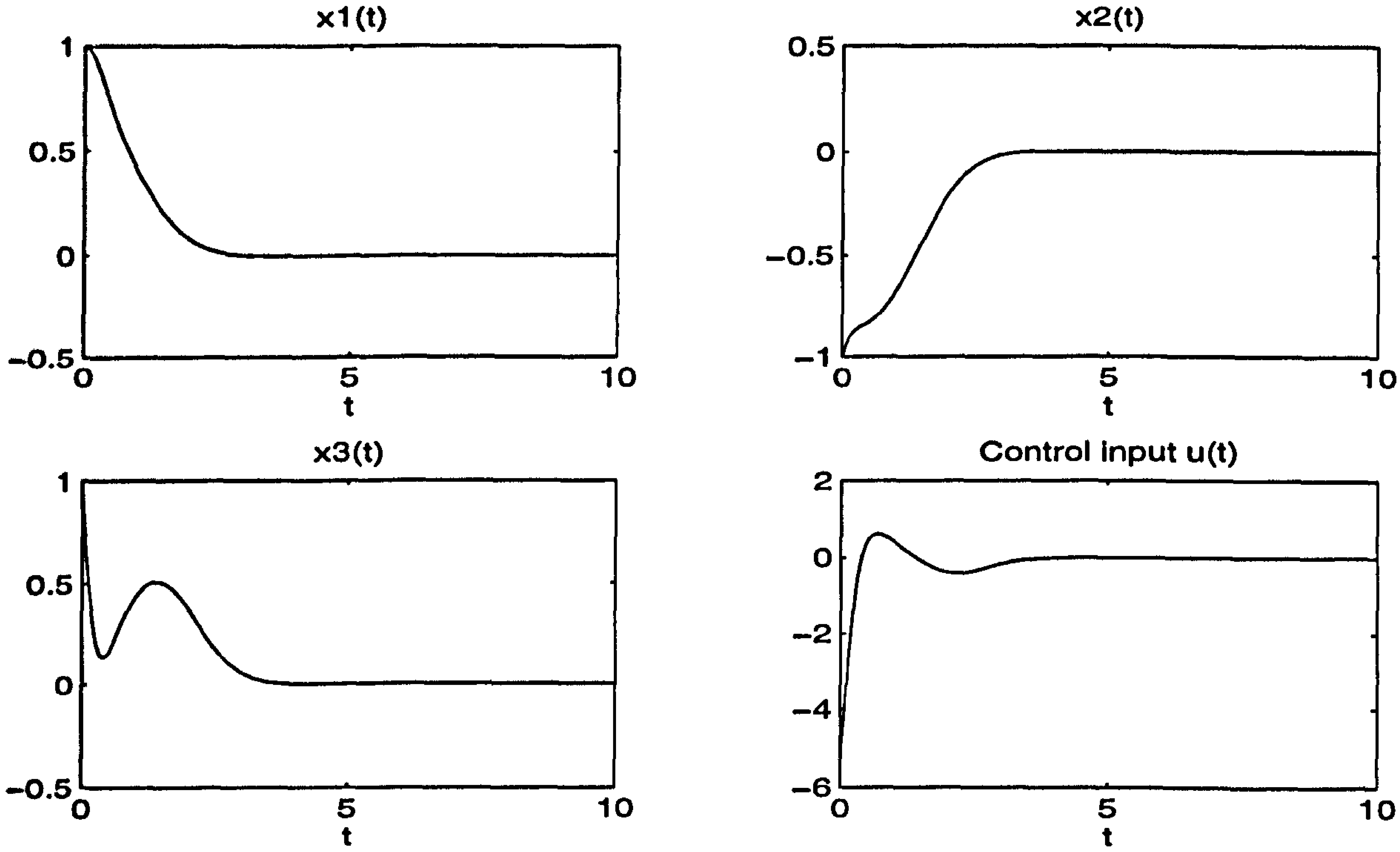


Figure 3.3: Controlled responses of an input-output linearizable system in regulation to the origin

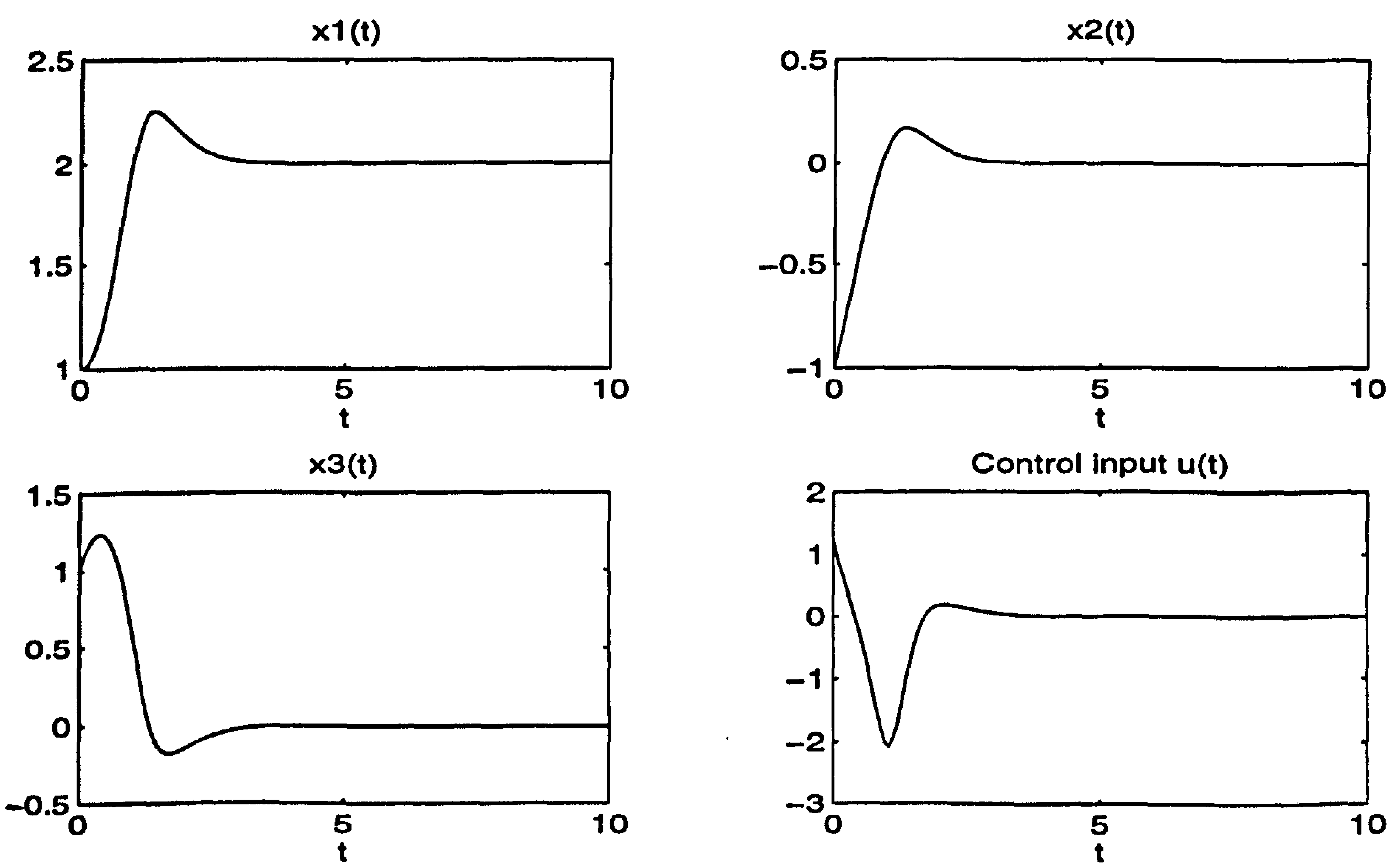


Figure 3.4: Controlled responses of an input-output linearizable system in regulation to the equilibrium point $x = (2, 0, 0)$

## 3.5 Concluding Remarks

The systematic DDB algorithm has been developed for observable minimum-phase systems. It follows an input-output linearization procedure adopting the backstepping ideas. Note that the DDB algorithm leads to the design of static controllers for input-output linearizable systems, i.e. systems with relative degree $\rho = n$, as shown in Example 3.7. In Chapter 4.1 we extend this approach to observable minimum phase nonlinear systems with uncertain parameters, by developing a dynamical adaptive backstepping algorithm with tuning functions.

# Chapter 4

# Dynamical Adaptive Backstepping Control

## 4.1 Introduction

The Static Adaptive Backstepping (SAB) algorithm of Section 2.4 is appropriate for a large class of feedback linearizable uncertain nonlinear systems which are transformable into the triangular canonical forms; parametric strict and parametric pure feedback forms. However, it cannot deal with nonlinear systems having a *nontriangular* form.

In this chapter a new systematic approach is presented for the design of Dynamical Adaptive Backstepping (DAB) controllers for the control of nontriangular systems. This new adaptive control design procedure is based upon a combination of dynamical input-output linearization and the adaptive backstepping algorithm. It does not require any canonical form but, instead, it does require that the plant be *observable* and *minimum-phase.* This new approach broadens the class of systems for which backstepping is applicable. It includes the SAB algorithm for PSF and PPF forms as a particular case.

We start by showing that for some systems for which the SAB algorithm may fail to guarantee asymptotic stability, dynamical adaptive controllers can achieve asymptotic stabilization. This fact motivates the need for the use of extended dynamical controllers to overcome this limitation of the SAB algorithm. Then the DAB algorithm developed by Rios-Bolívar *et al* [87] is described, and its stability and convergence properties are analysed. Two examples, including a system in PSF form and another system in nontriangular form, are used to illustrate its applicability to both PSF and observable minimum phase nonlinear systems with parametric uncertainty.

## 4.2 Dynamical Adaptive Backstepping (DAB) Algorithm

As shown in Section 2.4, the SAB algorithm with tuning functions achieves asymptotic stabilization of nonlinear systems in PPF or PSF forms. These forms can be seen as special structural triangular forms of nonlinear systems which are input-output linearizable, when all the parameters are known, by specifying the output $y = x_1$. Seto *et al.* [100] extended the class of uncertain nonlinear systems for which backstepping is applicable. However, to guarantee global asymptotic stability, they considered only systems which can be transformed into a triangular form. Here, a general backstepping algorithm for a class of observable nonlinear systems with nontriangular forms and dynamically input-output linearizable is developed. The algorithm is a new adaptive version of the systematic control design procedure given in Section 3.4 and is based upon a suitable combination of dynamical input-output linearization and the backstepping controller design method (see Rios-Bolívar *et al* [87]).

Consider a single-input single-output nonlinear system with parametric uncertainty of the form

$$\begin{aligned} \dot{x} &= f_0(x) + \Psi(x)\theta + \Big(g_0(x) + \varphi(x)\theta\Big)u \\ y &= h(x) \end{aligned} \tag{4.1}$$

where $x \in \Re^n$ is the state; $u, y \in \Re$ the input and output respectively, and $\theta = [\theta_1, \ldots, \theta_p]^T$ is a vector of unknown parameters. $f_0$, $g_0$ and the columns of the matrices $\Psi, \varphi \in \Re^{n \times p}$ are smooth vector fields in a neighbourhood $R_0$ of the origin $x = 0$ with $f_0(0) = 0$, $g_0(0) \neq 0$, and $h$ is a smooth scalar function also defined in $R_0$. It is assumed that the system (4.1) has relative degree $\rho$ strictly less than the system order.

Kanellakopoulos *et al.* [48] has presented necessary and sufficient conditions to transform (4.1) globally into the following *parametric strict-feedback normal form*

$$\begin{aligned} \dot{\xi}_i &= \xi_{i+1} + \varphi_i^T(\xi_1, \ldots, \xi_i, \xi^r)\theta \qquad 1 \leqslant i \leqslant \rho - 1 \\ \dot{\xi}_\rho &= \varphi_0(\xi) + \varphi_\rho^T(\xi)\theta + \beta_0(\xi)u \\ \dot{\xi}^r &= \Phi_0(y, \xi^r) + \Phi^T(y, \xi^r)\theta \qquad \xi^r \in \Re^{n-\rho} \\ y &= \xi_1 \end{aligned} \tag{4.2}$$

Assuming that the $\xi^r$-subsystem satisfies a bounded-input bounded-state condition with respect to $y$ as its input, the problem of tracking a bounded reference signal $y_r(t)$ with its first $\rho$ derivatives continuous and bounded, was solved by applying $\rho$ steps of the adaptive backstepping algorithm. However, even though

$$\lim_{t \to \infty} (y(t) - y_r(t)) = 0 \tag{4.3}$$

is achieved, only boundedness of the remaining variables is guaranteed in general. This is illustrated by the following example.

**Example 4.1** Consider the problem of adaptively regulating the nonlinear system

$$\begin{aligned} \dot{x}_1 &= x_2 + \theta x_1^2 \\ \dot{x}_2 &= x_3 + u \\ \dot{x}_3 &= -x_3 + \theta(1 + x_1) \\ y &= x_1 \end{aligned} \tag{4.4}$$

where $\theta$ is a constant but unknown scalar parameter, to take the output $y = x_1$ to zero. System (4.4) is already in the parametric strict-feedback normal form and is globally minimum-phase. The application of two steps of the SAB algorithm with tuning functions yields the error variables

$$\begin{aligned} z_1 &= x_1 \qquad (4.5) \\ z_2 &= x_2 - \alpha_1(x_1, \hat{\theta}) = x_2 + \hat{\theta} x_1^2 + c_1 x_1 \end{aligned}$$

The tuning functions are obtained as

$$\tau_1 = \gamma z_1 x_1^2 \tag{4.6}$$

$$\tau_2 = \tau_1 - \gamma z_2 \frac{\partial \alpha_1}{\partial x_1} x_1^2 = \gamma \left( z_1 - z_2 \frac{\partial \alpha_1}{\partial x_1} \right) x_1^2 \tag{4.7}$$

and the adaptive controller, formed by the parameter update law $\dot{\hat{\theta}} = \tau_2$ and the feedback control

$$u = -z_1 - x_3 + \frac{\partial \alpha_1}{\partial x_1}(x_2 + \hat{\theta} x_1^2) + \frac{\partial \alpha_1}{\partial \hat{\theta}} \tau_2 - c_2 z_2, \tag{4.8}$$

yields the closed-loop system

$$\begin{aligned} \dot{z} &= \begin{bmatrix} -c_1 & 1 \\ -1 & -c_2 \end{bmatrix} z + \begin{bmatrix} 1 \\ \frac{\partial \alpha_1}{\partial x_1} \end{bmatrix} z_1^2 \tilde{\theta} \\ \dot{x}_3 &= -x_3 + \theta(1 + z_1) \\ \dot{\tilde{\theta}} &= \gamma z_1^2 \begin{bmatrix} 1 & -\frac{\partial \alpha_1}{\partial x_1} \end{bmatrix} z \end{aligned} \tag{4.9}$$

where $z = [z_1 \; z_2]^T$ and $\tilde{\theta} = \theta - \hat{\theta}$. Note that, from the dynamic equations (4.4), the stabilization of the output $y = x_1$ also guarantees that $x_3$ converges to the unknown parameter $\theta$. Therefore, $(z, x_3, \tilde{\theta}) = (0, \theta, 0)$ is an equilibrium point of (4.9). In order to establish the stability properties of this equilibrium, recall that when applying backstepping the Lyapunov function

$$V = \frac{1}{2}z_1^2 + \frac{1}{2}z_2^2 + \frac{1}{2\gamma}\tilde{\theta}^2 \tag{4.10}$$

is used and the Lyapunov derivative

$$\dot{V} = -c_1 z_1^2 - c_2 z_2^2 \tag{4.11}$$

is obtained. Then, from the LaSalle theorem (Theorem A.2),

$$\lim_{t\to\infty} z = [z_1 \quad z_2]^T = 0, \tag{4.12}$$

which also guarantees that $x_1 \to 0$ as $t \to \infty$. Also recall from Section 2.4.4 that to guarantee convergence of the estimate $\hat{\theta}$ to the actual unknown parameter $\theta$, the rank condition

$$\text{rank}\left[x_1^2\right]_{(z, x_3, \tilde{\theta})=(0,\theta,0)} = 1 \tag{4.13}$$

must be satisfied. Since the output $x_1$ is stabilized to zero, the rank condition (4.13) is violated and convergence to the unknown parameter is not guaranteed. Hence asymptotic stabilization of the equilibrium $(z, x_3, \tilde{\theta}) = (0, \theta, 0)$ is not achieved. Computer simulations confirm this assertion. Figures 4.1 and 4.2 show the stabilization of the state variables to the desired equilibrium values, i.e. $x_1 = 0$, $x_2 = 0$ and $x_3 = \theta$. However, the parameter estimate $\hat{\theta}$ does not converge to the unknown true parameter value. Therefore asymptotic stabilization of the equilibrium point $(z, x_3, \tilde{\theta}) = (0, \theta, 0)$ of the closed-loop system (4.9) is not achieved.

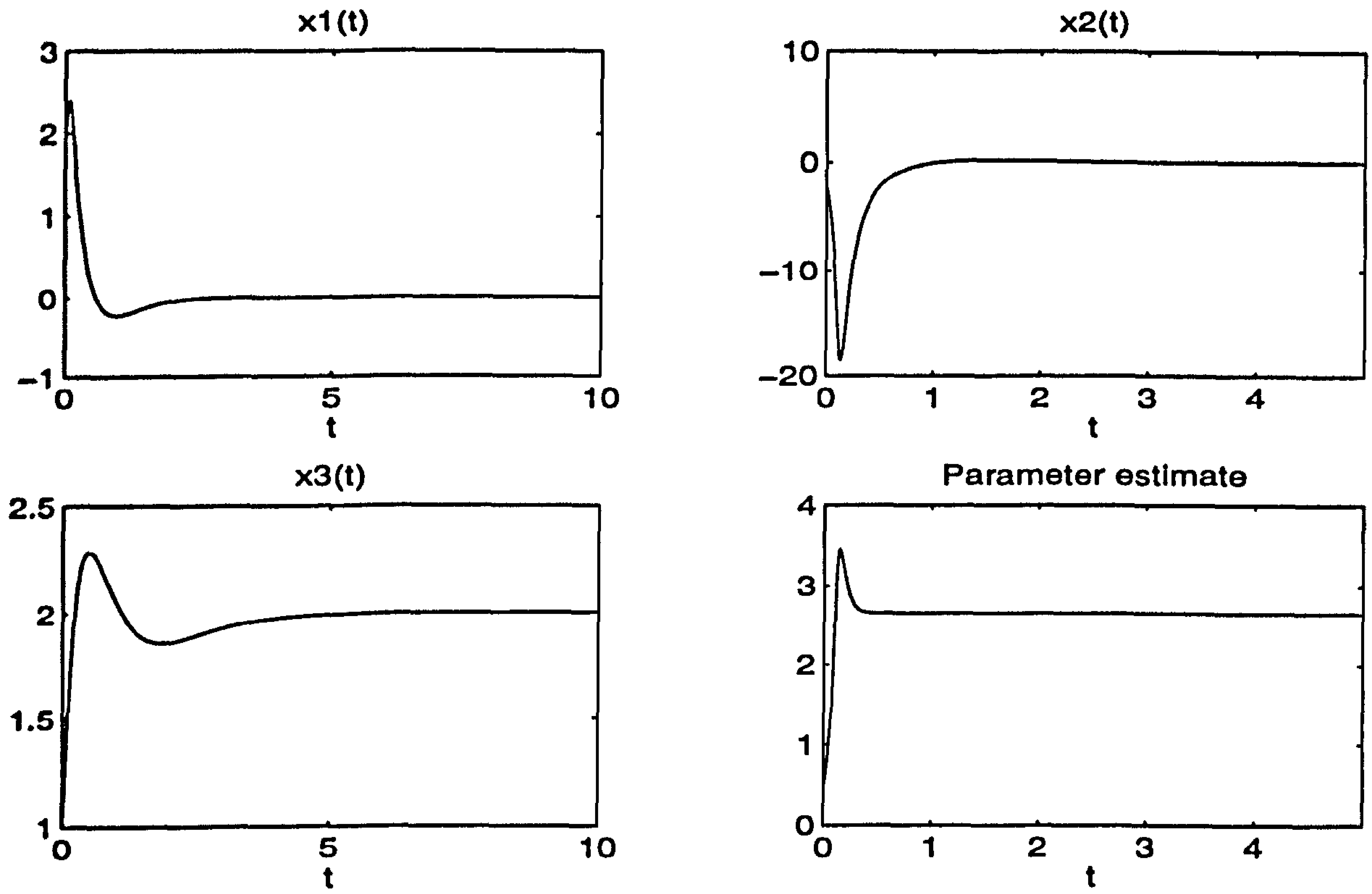


Figure 4.1: Controlled responses of a nonlinear system in PSF normal form

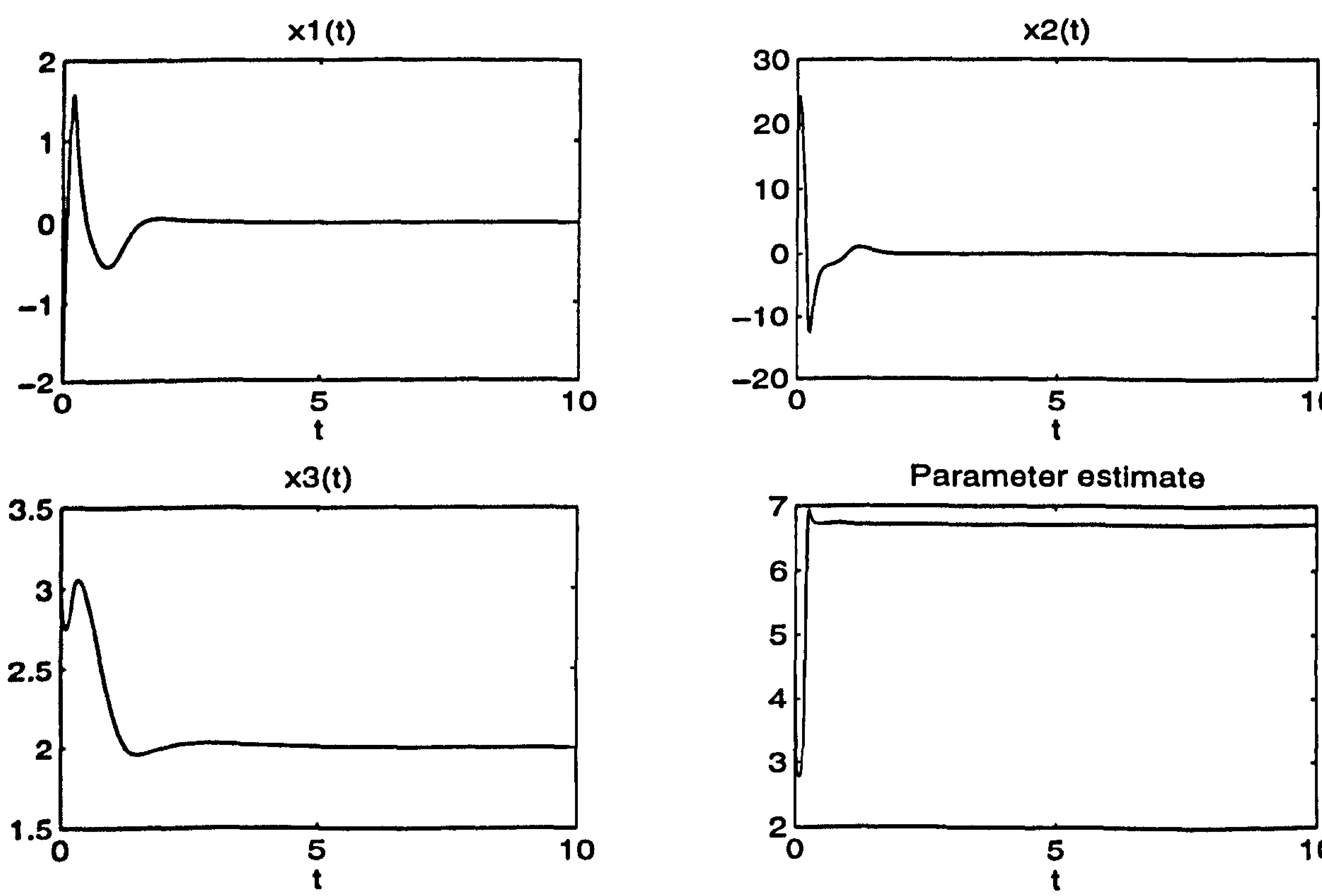


Figure 4.2: Controlled responses of a nonlinear system in PSF normal form for different initial conditions

This lack of asymptotic stabilization occurs because the SAB algorithm does not use the third equation of (4.4) which would allow the rank condition (4.13) to hold. It will be proved below that a dynamical adaptive backstepping control achieves the asymptotic stabilization of this equilibrium.

In order to overcome this limitation of the SAB algorithm and also to broaden the applicability of this technique to systems with nontriangular forms which are *not* transformable into either the parametric pure-feedback form or parametric strict-feedback form, a general DAB algorithm to design dynamical adaptive controllers via backstepping has been developed by Rios-Bolívar *et al* [87]. The steps yielding this adaptive controller follow an input-output linearization procedure and, moreover, the control input and its derivatives may appear at intermediate steps of the recursive design method. This procedure becomes equivalent to the traditional backstepping algorithm in [61] when the output corresponds to a linearizing function of the system, i.e. the relative degree is equal to the system order, $\rho = n$.

For the purpose of characterizing the class of uncertain nonlinear systems for which our algorithm is applicable, consider the adaptive version of the operator (3.75) on the system (4.1)

$$
\begin{aligned}
\mathcal{L}_h^0(x) &= h(x) \\
\mathcal{L}_h^1(x,\hat{\theta}) &= \frac{\partial h(x)}{\partial x}(f_0 + \Psi\hat{\theta}) \\
\mathcal{L}_h^i(x,\hat{\theta}) &= \frac{\partial\left(\mathcal{L}_h^{i-1}(x,\hat{\theta})\right)}{\partial x}(f_0 + \Psi\hat{\theta}) \\
&\quad + \frac{\partial\left(\mathcal{L}_h^{i-1}(x,\hat{\theta})\right)}{\partial\hat{\theta}}\tau_i(x,\hat{\theta}) \qquad 2 \leqslant i \leqslant \rho - 1 \\
\mathcal{L}_h^\rho(x,\hat{\theta},v_1) &= \frac{\partial\left(\mathcal{L}_h^{\rho-1}(x,\hat{\theta})\right)}{\partial x}\Big(f_0(x) + \Psi(x)\hat{\theta} + (g_0(x) + \varphi(x)\hat{\theta})v_1\Big) \\
&\quad + \frac{\partial\left(\mathcal{L}_h^{\rho-1}(x,\hat{\theta})\right)}{\partial\hat{\theta}}\tau_\rho(x,\hat{\theta},v_1) \\
\mathcal{L}_h^i(x,\hat{\theta},v_1,\ldots,v_{i-\rho+1}) &= \frac{\partial\left(\mathcal{L}_h^{i-1}(\cdot)\right)}{\partial x}\Big(f_0(x) + \Psi(x)\hat{\theta} + (g_0(x) + \varphi(x)\hat{\theta})v_1\Big) \\
&\quad + \frac{\partial\left(\mathcal{L}_h^{i-1}(\cdot)\right)}{\partial\hat{\theta}}\tau_i(x,\hat{\theta},v_1,\ldots,v_{i-\rho+1}) \\
&\quad + \sum_{j=1}^{i-\rho}\frac{\partial\left(\mathcal{L}_h^{i-1}(\cdot)\right)}{\partial v_j}v_{j+1} \qquad \rho + 1 \leqslant i \leqslant n - 1
\end{aligned}
\tag{4.14}
$$

where the $\tau_i$'s tuning functions are designed at each step of the design algorithm.

This allows us to express the output $y$ and its first $n-1$ derivatives as functions of $x$, $u$ and the derivatives of $u$

$$\Phi(x, \hat{\theta}, v_1, \ldots, v_{n-\rho}) = \begin{bmatrix} y \\ y^{(1)} \\ \vdots \\ y^{(n-1)} \end{bmatrix} = \begin{bmatrix} \mathcal{L}_h^0 \\ \mathcal{L}_h^1 \\ \vdots \\ \mathcal{L}_h^{n-1} \end{bmatrix}. \tag{4.15}$$

**Assumption 4.1** *System (4.1) is* observable, *i.e. the mapping (4.15) satisfies the rank condition*

$$\text{rank}\frac{\partial \Phi(\cdot)}{\partial x} = n \tag{4.16}$$

*in $R_0$.*

**Assumption 4.2** *System (4.1) is* minimum phase *in $R_0$.*

Note that the ability of a system in the form (4.1) to exhibit the observability property may depend upon the unknown parameters $\theta$. Nevertheless, in that case, one can assume that Assumption 4.1 is still valid in the neighbourhood $R_0$ for the nominal, but unknown, value of the parameters.

For Assumptions 4.1 and 4.2, i.e. observable minimum phase nonlinear systems of the form (4.1), the general problem of adaptively tracking a bounded desired reference signal $y_r(t)$ with smooth and bounded derivatives, can be solved using the DAB algorithm summarized as follows:

***DAB Algorithm***

*Coordinate transformation*

$$\begin{aligned} z_1 &:= y - y_r(t) = h^{(0)}(x) - y_r(t) \\ z_k &:= \hat{h}^{(k-1)}(x, \hat{\theta}, v_1, \ldots, v_{(k-\rho)}, t) - y_r^{(k-1)} + \alpha_{k-1}(x, \hat{\theta}, v_1, \ldots, v_{(k-\rho)}, t) \quad 2 \leqslant k \leqslant n \end{aligned} \tag{4.17}$$

*with*

$$\begin{aligned} \hat{h}^k &= \frac{\partial \hat{h}^{(k-1)}}{\partial \hat{\theta}} \tau_k + \frac{\partial \hat{h}^{(k-1)}}{\partial x} \left[ f_0 + \Psi \hat{\theta} + (g_0 + \varphi \hat{\theta}) v_1 \right] \\ &\quad + \sum_{i=1}^{k-\rho-1} \frac{\partial \hat{h}^{(k-1)}}{\partial v_i} v_{i+1} + \frac{\partial \hat{h}^{(k-1)}}{\partial t} \end{aligned} \tag{4.18}$$

$$\omega_k = \left( \frac{\partial \hat{h}^{(k-1)}}{\partial x} + \frac{\partial \alpha_{k-1}}{\partial x} \right) (\Psi + \varphi u) \tag{4.19}$$

$$\begin{aligned} \alpha_k &= z_{k-1} + \left( \sum_{i=2}^{k-1} z_i \frac{\partial \hat{h}^{(i-1)}}{\partial \hat{\theta}} + \sum_{i=3}^{k-1} z_i \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}} \right) \Gamma \omega_k^T + \sum_{i=1}^{k-\rho-1} \frac{\partial \alpha_{k-1}}{\partial v_i} v_{i+1} \\ &\quad + \frac{\partial \alpha_{k-1}}{\partial x} \left[ f_0 + \Psi \hat{\theta} + (g_0 + \varphi \hat{\theta}) v_1 \right] + \frac{\partial \alpha_{k-1}}{\partial \hat{\theta}} \tau_k + \frac{\partial \alpha_{k-1}}{\partial t} + c_k z_k \end{aligned} \tag{4.20}$$

$$\tau_k = \Gamma \sum_{i=1}^{k} \omega_k^T z_k \tag{4.21}$$

*Parameter update law*

$$\dot{\hat{\theta}} = \tau_n = \Gamma W^T z = \Gamma \left[ \omega_1^T \; \omega_2^T \; \ldots \; \omega_n^T \right] z \tag{4.22}$$

*Dynamical adaptive control law*

$$\begin{aligned} \dot{v}_1 &= v_2 \\ &\vdots \\ \dot{v}_{n-\rho-1} &= v_{n-\rho} \\ \dot{v}_{n-\rho} &= \frac{1}{\left( \frac{\partial \hat{h}^{(n-1)}}{\partial v_{n-\rho}} + \frac{\partial \alpha_{n-1}}{\partial v_{n-\rho}} \right)} \Bigg[ - z_{n-1} - \sum_{i=2}^{n-1} \left( \frac{\partial \hat{h}^{(n-1)}}{\partial \hat{\theta}} + \frac{\partial \alpha_{n-1}}{\partial \hat{\theta}} \right) z_i \Gamma \omega_n^T + y_r^{(n)}(t) \\ &\quad - \left( \frac{\partial \hat{h}^{(n-1)}}{\partial x} + \frac{\partial \alpha_{n-1}}{\partial x} \right) \left( f_0 + \Psi \hat{\theta} + (g_0 + \varphi \hat{\theta}) v_1 \right) \\ &\quad - \frac{\partial \hat{h}^{(n-1)}}{\partial t} - \frac{\partial \alpha_{n-1}}{\partial t} - \left( \frac{\partial \hat{h}^{(n-1)}}{\partial \hat{\theta}} + \frac{\partial \alpha_{n-1}}{\partial \hat{\theta}} \right) \tau_n \\ &\quad - \sum_{i=1}^{n-\rho-1} \left( \frac{\partial \hat{h}^{(n-1)}}{\partial v_i} + \frac{\partial \alpha_{n-1}}{\partial v_i} \right) v_{i+1} - c_n z_n \Bigg] \end{aligned} \tag{4.23}$$

The following steps lead to the general DAB algorithm summarized above.

**Step 1.** Define the output tracking error as

$$z_1 := y - y_r(t) = h(x) - y_r(t) \tag{4.24}$$

whose time derivative is given by

$$\dot{z}_1 = h^{(1)}(x, \theta) - \dot{y}_r(t) = \frac{\partial h}{\partial x}\Big[f_0 + \Psi\theta + (g_0 + \varphi\theta)u\Big] - \dot{y}_r(t) \tag{4.25}$$

If the relative degree $\rho$ with respect to $u$ is greater than one,

$$\frac{\partial h}{\partial x}\Big(g_0(x) + \varphi(x)\theta\Big) = 0 \tag{4.26}$$

For the sake of generality, it is assumed that the relative degree $\rho$ is greater than one. Nevertheless, this algorithm is also applicable to systems with $\rho = 1$. By adding to and subtracting from the actual value of the parameters $\theta$ their estimated values $\hat{\theta}$, (4.25) can be rewritten as

$$\dot{z}_1 = \hat{h}^{(1)}(x, \hat{\theta}) - \dot{y}_r(t) + \omega_1(\theta - \hat{\theta}) \tag{4.27}$$

with

$$\hat{h}^{(1)}(x, \hat{\theta}) = \frac{\partial h}{\partial x}\Big(f_0(x) + \Psi(x)\hat{\theta}\Big) \tag{4.28}$$

$$\omega_1 = \frac{\partial h}{\partial x}\Psi(x) \tag{4.29}$$

Consider the quadratic Lyapunov function

$$V_1 = \frac{1}{2}z_1^2 + \frac{1}{2}(\theta - \hat{\theta})^T\Gamma^{-1}(\theta - \hat{\theta}) \tag{4.30}$$

where $\Gamma = \Gamma^T > 0$ is a matrix of adaptation gains. The time derivative of $V_1$ is

$$\dot{V}_1 = z_1\left(\hat{h}^{(1)}(x, \hat{\theta}) - \dot{y}_r(t)\right) + (\theta - \hat{\theta})^T\Gamma^{-1}(-\dot{\hat{\theta}} + \Gamma\omega_1^T z_1) \tag{4.31}$$

One can achieve $\dot{V}_1 = -c_1 z_1^2$ with $c_1$ a positive scalar design constant, by choosing the tuning function

$$\dot{\hat{\theta}} = \tau_1 = \Gamma\omega_1^T z_1 \tag{4.32}$$

if the relation

$$\hat{h}^{(1)}(x, \hat{\theta}) - \dot{y}_r(t) = -c_1 z_1 \tag{4.33}$$

is satisfied. The expression (4.33) represents a desired algebraic relation for which effective stabilization of the output tracking error would be possible in combination with the estimation update law (4.32). However, since (4.33) is not valid from the outset and $\tau_1$

is not considered as an update law but rather as the first tuning function, the deviation is taken as the second error variable, i.e.

$$z_2 := \hat{h}^{(1)}(x,\hat{\theta}) - \dot{y}_r(t) + \alpha_1 \tag{4.34}$$

with

$$\alpha_1 = c_1 z_1 \tag{4.35}$$

The closed-loop form is

$$\dot{z}_1 = -c_1 z_1 + z_2 + \omega_1(\theta - \hat{\theta}) \tag{4.36}$$

and

$$\dot{V}_1 = -c_1 z_1^2 + z_1 z_2 + (\theta - \hat{\theta})^T \Gamma^{-1}(-\dot{\hat{\theta}} + \tau_1). \tag{4.37}$$

By induction one obtains the following $j$-th generic step which characterizes the first steps prior to the explicit appearance of the control input in the transformed dynamical system.

**Step j** $(2 \leqslant j \leqslant \rho - 1)$

$$\begin{aligned}\dot{z}_j = \hat{h}^{(j)}(x,\hat{\theta},t) - y_r^{(j)}(t) + \frac{\partial \alpha_{j-1}}{\partial x}(f_0 + \Psi\hat{\theta}) + \frac{\partial \alpha_{j-1}}{\partial \hat{\theta}}\tau_j + \frac{\partial \alpha_{j-1}}{\partial t} \\ + \omega_j(\theta - \hat{\theta}) + \left(\frac{\partial \hat{h}^{(j-1)}}{\partial \hat{\theta}} + \frac{\partial \alpha_{j-1}}{\partial \hat{\theta}}\right)(\dot{\hat{\theta}} - \tau_j)\end{aligned} \tag{4.38}$$

with

$$\hat{h}^{(j)}(x,\hat{\theta},t) = \frac{\partial \hat{h}^{(j-1)}}{\partial x}(f_0 + \Psi\hat{\theta}) + \frac{\partial \hat{h}^{(j-1)}}{\partial \hat{\theta}}\tau_j + \frac{\partial \hat{h}^{(j-1)}}{\partial t} \tag{4.39}$$

$$\omega_j = \left(\frac{\partial \hat{h}^{(j-1)}}{\partial x} + \frac{\partial \alpha_{j-1}}{\partial x}\right)\Psi(x) \tag{4.40}$$

and $\tau_j$ the corresponding tuning function defined at this step. By augmenting the Lyapunov function

$$V_j = V_{j-1} + \frac{1}{2}z_j^2 = \frac{1}{2}\sum_{i=1}^{j} z_i^2 + \frac{1}{2}(\theta - \hat{\theta})^T \Gamma^{-1}(\theta - \hat{\theta}) \tag{4.41}$$

and time derivative is

$$\begin{aligned}\dot{V}_j = -\sum_{i=1}^{j-1} c_i z_i^2 + (\theta - \hat{\theta})^T \Gamma^{-1}(-\dot{\hat{\theta}} + \tau_{j-1} + \Gamma\omega_j^T z_j) \\ + z_j\left(\frac{\partial \hat{h}^{(j-1)}}{\partial \hat{\theta}} + \frac{\partial \alpha_{j-1}}{\partial \hat{\theta}}\right)(\dot{\hat{\theta}} - \tau_j)\end{aligned}$$

$$+\left(\sum_{i=2}^{j-1} z_i \frac{\partial \hat{h}^{(i-1)}}{\partial \hat{\theta}} + \sum_{i=3}^{j-1} z_i \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}}\right)(\dot{\hat{\theta}} - \tau_{j-1})$$
$$+ z_j\Big[z_{j-1} + \hat{h}^{(j)}(x,\hat{\theta},t) - y_r^{(j)}(t) + \frac{\partial \alpha_{j-1}}{\partial t}$$
$$+ \frac{\partial \alpha_{j-1}}{\partial \hat{\theta}}\tau_j + \frac{\partial \alpha_{j-1}}{\partial x}(f_0 + \Psi\hat{\theta})\Big] \tag{4.42}$$

The parameter estimate error $(\theta - \hat{\theta})$ can be eliminated from $\dot{V}_j$ by choosing the update law

$$\dot{\hat{\theta}} = \tau_j = \tau_{j-1} + \Gamma\omega_j^T z_j \tag{4.43}$$

However, $\tau_j$ will instead be used as a new tuning function. Thus, noting that

$$\dot{\hat{\theta}} - \tau_{j-1} = \dot{\hat{\theta}} - \tau_j + \tau_j - \tau_{j-1} = \dot{\hat{\theta}} - \tau_j + \Gamma\omega_j^T z_j, \tag{4.44}$$

one can rewrite $\dot{V}_j$ as

$$\dot{V}_j = -\sum_{i=1}^{j-1} c_i z_i^2 + (\theta - \hat{\theta})^T\Gamma^{-1}(-\dot{\hat{\theta}} + \tau_j)$$
$$+\left(\sum_{i=2}^{j} z_i \frac{\partial \hat{h}^{(i-1)}}{\partial \hat{\theta}} + \sum_{i=3}^{j} z_i \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}}\right)(\dot{\hat{\theta}} - \tau_j)$$
$$+ z_j\Bigg[\left(\sum_{i=2}^{j-1} z_i \frac{\partial \hat{h}^{(i-1)}}{\partial \hat{\theta}} + \sum_{i=3}^{j-1} z_i \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}}\right)\Gamma\omega_j^T + \hat{h}^{(j)}(x,\hat{\theta},t) - y_r^{(j)}(t)$$
$$+ \frac{\partial \alpha_{j-1}}{\partial x}(f_0 + \Psi\hat{\theta}) + \frac{\partial \alpha_{j-1}}{\partial \hat{\theta}}\tau_j + \frac{\partial \alpha_{j-1}}{\partial t} + z_{j-1}\Bigg] \tag{4.45}$$

One can achieve $\dot{V}_j = -\sum_{i=1}^{j} c_i z_i^2$, with the $c_i$'s being positive scalar design constants, if $\tau_j$ is the update law and the relation

$$\left(\sum_{i=2}^{j-1} z_i \frac{\partial \hat{h}^{(i-1)}}{\partial \hat{\theta}} + \sum_{i=3}^{j-1} z_i \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}}\right)\Gamma\omega_j^T + \hat{h}^{(j)}(x,\hat{\theta},t) - y_r^{(j)}(t)$$
$$+ \frac{\partial \alpha_{j-1}}{\partial x}(f_0 + \Psi\hat{\theta}) + \frac{\partial \alpha_{j-1}}{\partial \hat{\theta}}\tau_j + \frac{\partial \alpha_{j-1}}{\partial t} + z_{j-1} = -c_j z_j \tag{4.46}$$

is satisfied. Since (4.46) is not valid from the outset, its deviation is taken as the $(j+1)$-th error variable

$$z_{j+1} := \hat{h}^{(j)}(x,\hat{\theta},t) - y_r^{(j)}(t) + \alpha_j(x,\hat{\theta},t) \tag{4.47}$$

with

$$\alpha_j = z_{j-1} + \left(\sum_{i=2}^{j-1} z_i \frac{\partial \hat{h}^{(i-1)}}{\partial \hat{\theta}} + \sum_{i=3}^{j-1} z_i \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}}\right)\Gamma\omega_j^T + \frac{\partial \alpha_{j-1}}{\partial \hat{\theta}}\tau_j$$
$$+ \frac{\partial \alpha_{j-1}}{\partial x}(f_0(x) + \Psi(x)\hat{\theta}) + \frac{\partial \alpha_{j-1}}{\partial t} + c_j z_j \tag{4.48}$$

obtaining the closed-loop form for $\dot{z}_j$ as

$$\dot{z}_j = -z_{j-1} - c_j z_j + z_{j+1} + \omega_j(\theta - \hat{\theta}) + \left(\frac{\partial \hat{h}^{(j-1)}}{\partial \hat{\theta}} + \frac{\partial \alpha_{j-1}}{\partial \hat{\theta}}\right)(\dot{\hat{\theta}} - \tau_j)$$
$$- \left(\sum_{i=2}^{j-1} z_i \frac{\partial \hat{h}^{(i-1)}}{\partial \hat{\theta}} + \sum_{i=3}^{j-1} z_i \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}}\right)\Gamma \omega_j^T \tag{4.49}$$

and

$$\dot{V}_j = -\sum_{i=1}^{j} c_i z_i^2 + z_j z_{j+1} + (\theta - \hat{\theta})^T \Gamma^{-1}(-\dot{\hat{\theta}} + \tau_j)$$
$$+ \left(\sum_{i=2}^{j} z_i \frac{\partial \hat{h}^{(i-1)}}{\partial \hat{\theta}} + \sum_{i=3}^{j} z_i \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}}\right)(\dot{\hat{\theta}} - \tau_j). \tag{4.50}$$

Now the steps containing the control input and its derivatives are summarized in the following generic step.

**Step k** ($\rho \leqslant k \leqslant n-1$)

$$\dot{z}_k = \hat{h}^{(k)}(x, \hat{\theta}, u, \ldots, u^{(k-\rho)}, t) - y_r^{(k)}(t) + \frac{\partial \alpha_{k-1}}{\partial t} + \frac{\partial \alpha_{k-1}}{\partial \hat{\theta}}\tau_k$$
$$+ \frac{\partial \alpha_{k-1}}{\partial x}\left[f_0 + \Psi\hat{\theta} + (g_0 + \varphi\hat{\theta})u\right] + \sum_{i=1}^{k-\rho} \frac{\partial \alpha_{k-1}}{\partial u^{(i-1)}} u^{(i)} + \omega_k(\theta - \hat{\theta})$$
$$+ \left(\frac{\partial \hat{h}^{(k-1)}}{\partial \hat{\theta}} + \frac{\partial \alpha_{k-1}}{\partial \hat{\theta}}\right)(\dot{\hat{\theta}} - \tau_k) \tag{4.51}$$

with

$$\hat{h}^{(k)}(x, \hat{\theta}, u, \ldots, u^{(k-\rho)}, t) = \frac{\partial \hat{h}^{(k-1)}}{\partial \hat{\theta}}\tau_k + \frac{\partial \hat{h}^{(k-1)}}{\partial x}\left[f_0 + \Psi\hat{\theta} + (g_0 + \varphi\hat{\theta})u\right]$$
$$+ \sum_{i=1}^{k-\rho} \frac{\partial \hat{h}^{(k-1)}}{\partial u^{(i-1)}} u^{(i)} + \frac{\partial \hat{h}^{(k-1)}}{\partial t} \tag{4.52}$$

$$\omega_k = \left(\frac{\partial \hat{h}^{(k-1)}}{\partial x} + \frac{\partial \alpha_{k-1}}{\partial x}\right)(\Psi + \varphi u) \tag{4.53}$$

and $\tau_k$ the tuning function defined at this step. By augmenting the Lyapunov function

$$V_k = V_{k-1} + \frac{1}{2}z_k^2 = \frac{1}{2}\sum_{i=1}^{k} z_i^2 + \frac{1}{2}(\theta - \hat{\theta})^T \Gamma^{-1}(\theta - \hat{\theta}) \tag{4.54}$$

and its time derivative is

$$\dot{V}_k = -\sum_{i=1}^{k-1} c_i z_i^2 + (\theta - \hat{\theta})^T \Gamma^{-1}\left(-\dot{\hat{\theta}} + \tau_{k-1} + \Gamma \omega_k^T z_k\right)$$

$$
\begin{aligned}
&+ z_k \left( \frac{\partial \hat{h}^{(k-1)}}{\partial \hat{\theta}} + \frac{\partial \alpha_{k-1}}{\partial \hat{\theta}} \right) (\dot{\hat{\theta}} - \tau_k) \\
&+ \left( \sum_{i=2}^{k-1} z_i \frac{\partial \hat{h}^{(i-1)}}{\partial \hat{\theta}} + \sum_{i=3}^{k-1} z_i \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}} \right) (\dot{\hat{\theta}} - \tau_{k-1}) \\
&+ z_k \Bigg[ z_{k-1} + \hat{h}^{(k)} - y_r^{(k)} + \frac{\partial \alpha_{k-1}}{\partial \hat{\theta}} \tau_k + \sum_{i=1}^{k-\rho} \frac{\partial \alpha_{k-1}}{\partial u^{(i-1)}} u^{(i)} \\
&\qquad + \frac{\partial \alpha_{k-1}}{\partial t} + \frac{\partial \alpha_{k-1}}{\partial x} \Big[ f_0 + \Psi \hat{\theta} + (g_0 + \varphi \hat{\theta}) u \Big] \Bigg]
\end{aligned} \tag{4.55}
$$

The parameter estimate error $(\theta - \hat{\theta})$ can be eliminated from $\dot{V}_k$ by choosing the update law

$$
\dot{\hat{\theta}} = \tau_k = \tau_{k-1} + \Gamma \omega_k^T z_k. \tag{4.56}
$$

However $\tau_k$ will instead be used as a new tuning function. Thus, noting that

$$
\dot{\hat{\theta}} - \tau_{k-1} = \dot{\hat{\theta}} - \tau_k + \tau_k - \tau_{k-1} = \dot{\hat{\theta}} - \tau_k + \Gamma \omega_k^T z_k, \tag{4.57}
$$

$\dot{V}_k$ can be rewritten as

$$
\begin{aligned}
\dot{V}_k = &- \sum_{i=1}^{k-1} c_i z_i^2 + (\theta - \hat{\theta})^T \Gamma^{-1} (-\dot{\hat{\theta}} + \tau_k) \\
&+ \left( \sum_{i=2}^{k} z_i \frac{\partial \hat{h}^{(i-1)}}{\partial \hat{\theta}} + \sum_{i=3}^{k} z_i \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}} \right) (\dot{\hat{\theta}} - \tau_k) \\
&+ z_k \Bigg[ \left( \sum_{i=2}^{k-1} z_i \frac{\partial \hat{h}^{(i-1)}}{\partial \hat{\theta}} + \sum_{i=3}^{k-1} z_i \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}} \right) \Gamma \omega_k^T + \hat{h}^{(k)} - y_r^{(k)} \\
&\qquad + \frac{\partial \alpha_{k-1}}{\partial \hat{\theta}} \tau_k + \frac{\partial \alpha_{k-1}}{\partial x} \Big[ f_0 + \Psi \hat{\theta} + (g_0 + \varphi \hat{\theta}) u \Big] \\
&\qquad + \sum_{i=1}^{k-\rho} \frac{\partial \alpha_{k-1}}{\partial u^{(i-1)}} u^{(i)} + \frac{\partial \alpha_{k-1}}{\partial t} + z_{k-1} \Bigg]
\end{aligned} \tag{4.58}
$$

One can achieve $\dot{V}_k = -\sum_{i=1}^{k} c_i z_i^2$, with the $c_i$'s being positive scalar design constants, if $\tau_k$ were the update law and the relation

$$
\begin{aligned}
&z_{k-1} + \left( \sum_{i=2}^{k-1} z_i \frac{\partial \hat{h}^{(i-1)}}{\partial \hat{\theta}} + \sum_{i=3}^{k-1} z_i \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}} \right) \Gamma \omega_k^T + \hat{h}^{(k)} - y_r^{(k)} + \sum_{i=1}^{k-\rho} \frac{\partial \alpha_{k-1}}{\partial u^{(i-1)}} u^{(i)} \\
&+ \frac{\partial \alpha_{k-1}}{\partial \hat{\theta}} \tau_k + \frac{\partial \alpha_{k-1}}{\partial x} \Big[ f_0 + \Psi \hat{\theta} + (g_0 + \varphi \hat{\theta}) u \Big] + \frac{\partial \alpha_{k-1}}{\partial t} = -c_k z_k
\end{aligned} \tag{4.59}
$$

were satisfied. However, since (4.59) is not valid from the outset, its deviation is taken as the $(k+1)$-th error variable

$$
z_{k+1} := \hat{h}^{(k)}(x, \hat{\theta}, u, \ldots, u^{(k-\rho)}, t) - y_r^{(k)} + \alpha_k(x, \hat{\theta}, u, \ldots, u^{(k-\rho)}, t) \tag{4.60}
$$

with

$$\alpha_k = z_{k-1} + \left(\sum_{i=2}^{k-1} z_i \frac{\partial \hat{h}^{(i-1)}}{\partial \hat{\theta}} + \sum_{i=3}^{k-1} z_i \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}}\right) \Gamma \omega_k^T + \sum_{i=1}^{k-\rho} \frac{\partial \alpha_{k-1}}{\partial u^{(i-1)}} u^{(i)}$$
$$+ \frac{\partial \alpha_{k-1}}{\partial x}\left[f_0 + \Psi\hat{\theta} + (g_0 + \varphi\hat{\theta})u\right] + \frac{\partial \alpha_{k-1}}{\partial \hat{\theta}}\tau_k + \frac{\partial \alpha_{k-1}}{\partial t} + c_k z_k. \quad (4.61)$$

We obtain the closed-loop form

$$\dot{z}_k = -z_{k-1} - c_k z_k + z_{k+1} + \omega_k(\theta - \hat{\theta})^T + \left(\frac{\partial \hat{h}^{(k-1)}}{\partial \hat{\theta}} + \frac{\partial \alpha_{k-1}}{\partial \hat{\theta}}\right)(\dot{\hat{\theta}} - \tau_k)$$
$$- \left(\sum_{i=2}^{k-1} z_i \frac{\partial \hat{h}^{(i-1)}}{\partial \hat{\theta}} + \sum_{i=3}^{k-1} z_i \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}}\right) \Gamma \omega_k^T \quad (4.62)$$

and

$$\dot{V}_k = -\sum_{i=1}^{k} c_i z_i^2 + z_k z_{k+1} + (\theta - \hat{\theta})^T \Gamma^{-1}(-\dot{\hat{\theta}} + \tau_k)$$
$$+ \left(\sum_{i=2}^{k} z_i \frac{\partial \hat{h}^{(i-1)}}{\partial \hat{\theta}} + \sum_{i=3}^{k} z_i \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}}\right)(\dot{\hat{\theta}} - \tau_k) \quad (4.63)$$

**Step n.** The design of both the actual update law and the dynamical adaptive output tracking controller is completed at this final step. Using the definition (4.60)

$$\dot{z}_n = \hat{h}^{(n)}(x, \hat{\theta}, u, \ldots, u^{(n-\rho)}, t) - y_r^{(n)}(t) + \frac{\partial \alpha_{n-1}}{\partial t} + \frac{\partial \alpha_{n-1}}{\partial \hat{\theta}}\tau_n$$
$$+ \frac{\partial \alpha_{n-1}}{\partial x}\left[f_0 + \Psi\hat{\theta} + (g_0 + \varphi\hat{\theta})u\right] + \sum_{i=1}^{n-\rho} \frac{\partial \alpha_{n-1}}{\partial u^{(i-1)}} u^{(i)} + \omega_n(\theta - \hat{\theta})$$
$$+ \left(\frac{\partial \hat{h}^{(n-1)}}{\partial \hat{\theta}} + \frac{\partial \alpha_{n-1}}{\partial \hat{\theta}}\right)(\dot{\hat{\theta}} - \tau_n) \quad (4.64)$$

with

$$\hat{h}^{(n)}(x, \hat{\theta}, u, \ldots, u^{(n-\rho)}, t) = \frac{\partial \hat{h}^{(n-1)}}{\partial \hat{\theta}}\tau_n + \frac{\partial \hat{h}^{(n-1)}}{\partial x}\left[f_0 + \Psi\hat{\theta} + (g_0 + \varphi\hat{\theta})u\right]$$
$$+ \sum_{i=1}^{n-\rho} \frac{\partial \hat{h}^{(n-1)}}{\partial u^{(i-1)}} u^{(i)} + \frac{\partial \hat{h}^{(n-1)}}{\partial t} \quad (4.65)$$

$$\omega_n = \left(\frac{\partial \hat{h}^{(n-1)}}{\partial x} + \frac{\partial \alpha_{n-1}}{\partial x}\right)(\Psi + \varphi u) \quad (4.66)$$

and $\tau_n$ the tuning function defined at this final step. Augmenting the Lyapunov function

$$V_n = V_{n-1} + \frac{1}{2}z_n^2 = \frac{1}{2}z^T z + \frac{1}{2}(\theta - \hat{\theta})^T \Gamma^{-1}(\theta - \hat{\theta}) \quad (4.67)$$

and its time derivative is

$$\begin{aligned}\dot{V}_n = &-\sum_{i=1}^{n-1} c_i z_i^2 + (\theta-\hat{\theta})^T\Gamma^{-1}\Big(-\dot{\hat{\theta}} + \tau_{n-1} + \Gamma\omega_n^T z_n\Big)\\ &+ z_n\left(\frac{\partial \hat{h}^{(n-1)}}{\partial\hat{\theta}} + \frac{\partial\alpha_{n-1}}{\partial\hat{\theta}}\right)(\dot{\hat{\theta}} - \tau_n)\\ &+\left(\sum_{i=2}^{n-1} z_i\frac{\partial \hat{h}^{(i-1)}}{\partial\hat{\theta}} + \sum_{i=3}^{n-1} z_i\frac{\partial\alpha_{i-1}}{\partial\hat{\theta}}\right)(\dot{\hat{\theta}} - \tau_{n-1})\\ &+ z_n\Bigg[z_{n-1} + \hat{h}^{(n)} - y_r^{(n)} + \frac{\partial\alpha_{n-1}}{\partial\hat{\theta}}\tau_n + \sum_{i=1}^{n-\rho}\frac{\partial\alpha_{n-1}}{\partial u^{(i-1)}}u^{(i)}\\ &\qquad + \frac{\partial\alpha_{n-1}}{\partial t} + \frac{\partial\alpha_{n-1}}{\partial x}\Big[f_0 + \Psi\hat{\theta} + (g_0+\varphi\hat{\theta})u\Big]\Bigg]\end{aligned} \tag{4.68}$$

At this final step one can eliminate the parameter estimate error $(\theta-\hat{\theta})$ from $\dot{V}_n$ with the update law

$$\dot{\hat{\theta}} = \tau_n = \tau_{n-1} + \Gamma\omega_n^T z_n = \Gamma W^T z \tag{4.69}$$

where the regressor matrix $W^T$ is composed of the regressor vectors as follows

$$W^T = \Big[\omega_1^T\ \omega_2^T\ \dots\ \omega_n^T\Big]. \tag{4.70}$$

Then, noting that

$$\dot{\hat{\theta}} - \tau_{n-1} = \dot{\hat{\theta}} - \tau_n + \tau_n - \tau_{n-1} = \dot{\hat{\theta}} - \tau_n + \Gamma\omega_n^T z_n, \tag{4.71}$$

$\dot{V}_n$ can be rewritten as

$$\begin{aligned}\dot{V}_n = &-\sum_{i=1}^{n-1} c_i z_i^2 + (\theta-\hat{\theta})^T\Gamma^{-1}(-\dot{\hat{\theta}} + \tau_n)\\ &+\left(\sum_{i=2}^{n} z_i\frac{\partial \hat{h}^{(i-1)}}{\partial\hat{\theta}} + \sum_{i=3}^{n} z_i\frac{\partial\alpha_{i-1}}{\partial\hat{\theta}}\right)(\dot{\hat{\theta}} - \tau_n)\\ &+ z_n\Bigg[\left(\sum_{i=2}^{n-1} z_i\frac{\partial \hat{h}^{(i-1)}}{\partial\hat{\theta}} + \sum_{i=3}^{n-1} z_i\frac{\partial\alpha_{i-1}}{\partial\hat{\theta}}\right)\Gamma\omega_n^T + \hat{h}^{(n)} - y_r^{(n)}\\ &\qquad + \frac{\partial\alpha_{n-1}}{\partial\hat{\theta}}\tau_n + \frac{\partial\alpha_{n-1}}{\partial x}\Big[f_0 + \Psi\hat{\theta} + (g_0+\varphi\hat{\theta})u\Big]\\ &\qquad + \sum_{i=1}^{n-\rho}\frac{\partial\alpha_{n-1}}{\partial u^{(i-1)}}u^{(i)} + \frac{\partial\alpha_{n-1}}{\partial t} + z_{n-1}\Bigg]\end{aligned} \tag{4.72}$$

In order to achieve

$$\dot{V} = \dot{V}_n = -\sum_{i=1}^{n} c_i z_i^2 \leqslant 0 \tag{4.73}$$

one must make the bracketed term multiplying $z_n$ equal to $-c_n z_n$, i.e.

$$z_{n-1} + \left( \sum_{i=2}^{n-1} z_i \frac{\partial \hat{h}^{(i-1)}}{\partial \hat{\theta}} + \sum_{i=3}^{n-1} z_i \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}} \right) \Gamma \omega_n^T + \hat{h}^{(n)}(x, \hat{\theta}, u, \ldots, u^{(n-\rho)}, t) \\ - y_r^{(n)} + \frac{\partial \alpha_{n-1}}{\partial x} \left[ f_0 + \Psi \hat{\theta} + (g_0 + \varphi \hat{\theta}) u \right] \\ + \frac{\partial \alpha_{n-1}}{\partial t} + \sum_{i=1}^{n-\rho} \frac{\partial \alpha_{n-1}}{\partial u^{(i-1)}} u^{(i)} + \frac{\partial \alpha_{n-1}}{\partial \hat{\theta}} \tau_n = -c_n z_n \tag{4.74}$$

The control function $u$ can be obtained implicitly as the solution of the nonlinear time-varying differential equation defined by (4.74). Note that the control law (4.74) can be rewritten in the form of the dynamical controller (4.23) by replacing the control input $u$ and its derivatives $\dot{u}, \ddot{u}, \ldots$ by the extended state variables $v_1, v_2, v_3, \ldots$ respectively. The overall closed-loop *error system* has the form

$$\dot{z} = A_z z + W(\theta - \hat{\theta}) \tag{4.75}$$

$$\dot{\hat{\theta}} = \Gamma W^T z \tag{4.76}$$

where $A_z$ has the following skew-symmetric form

$$A_z = \begin{bmatrix} -c_1 & 1 & 0 & \cdots & 0 & 0 \\ -1 & -c_2 & 1 + \varrho_{2,3} & \cdots & \varrho_{2,n-1} & \varrho_{2,n} \\ 0 & -1 - \varrho_{2,3} & -c_3 & \cdots & \varrho_{3,n-1} & \varrho_{3,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & -\varrho_{2,n-1} & -\varrho_{3,n-1} & \cdots & -c_{n-1} & 1 + \varrho_{n-1,n} \\ 0 & -\varrho_{2,n} & -\varrho_{3,n} & \cdots & -1 - \varrho_{n-1,n} & -c_n \end{bmatrix} \tag{4.77}$$

with

$$\varrho_{i,j} = \left( \frac{\partial \hat{h}^{(i-1)}}{\partial \hat{\theta}} + \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}} \right) \Gamma \omega_j^T. \tag{4.78}$$

## 4.3 Analysis of Stability and Convergence

The stability of the equilibrium $(z, \tilde{\theta}) = (0, 0)$, with $\tilde{\theta} = \theta - \hat{\theta}$, is now established for our new algorithm. Since the time derivative of the Lyapunov function $V$ along the solutions of (4.75)-(4.76) is nonpositive, uniform stability of the equilibrium $(z, \tilde{\theta}) = (0, 0)$ is guaranteed. Moreover, by virtue of the LaSalle-Yoshizawa Theorem (Theorem A.3), it follows further that, as $t \to \infty$, all solutions converge to the manifold characterized by $z = 0$, namely

$$\lim_{t \to \infty} \dot{V} = -\sum_{i=1}^{n} c_i z_i^2 = 0 \tag{4.79}$$

This proves, in particular, that $\hat{\theta}$ is bounded and $z(t) \to 0$. Consequently,

$$\lim_{t\to\infty} \Big[y(t) - y_r(t)\Big] = 0, \tag{4.80}$$

i.e. asymptotic tracking is achieved. Furthermore, from (4.70) and the definitions of the regressor vectors given in (4.53), it is seen that the components of the regressor matrix $W$ depend in general upon $t, x, z, \hat{\theta}, v_1, \dots, v_{n-\rho}$

$$W := O(t, z, \hat{\theta}, v_1, \dots, v_{n-\rho})F(x,u) = \begin{bmatrix} \dfrac{\partial h}{\partial x} \\ \dfrac{\partial \hat{h}^{(1)}}{\partial x} + \dfrac{\partial \alpha_1}{\partial x} \\ \vdots \\ \dfrac{\partial \hat{h}^{(n-1)}}{\partial x} + \dfrac{\partial \alpha_{n-1}}{\partial x} \end{bmatrix} F(x,u) \tag{4.81}$$

where

$$F(x,u) = \Psi(x) + \varphi(x)u \tag{4.82}$$

The matrix $O(\cdot)$ in (4.81) is the partial derivative of the observability mapping with respect to $x$, namely

$$O(t, z, \hat{\theta}, v_1, \dots, v_{n-\rho}) = \left[\frac{\partial \Phi(\cdot)}{\partial x}\right]_{x=\Phi^{-1}(z, v_1, \dots, v_{n-\rho})} \tag{4.83}$$

which by Assumption 4.1 is nonsingular. Moreover, if $\lim_{t\to\infty} y_r^{(i)}(t) = 0$, i=0,...,$n$, holds, and $F(0,0) = 0$, $z = 0$ is an equilibrium point of the system (4.75)-(4.76) and, then, $\lim_{t\to\infty} x(t) = 0$.

The above facts prove the following theorem.

**Theorem 4.1** *The closed-loop adaptive system consisting of the plant (4.1), the dynamical controller defined by (4.74) and the update law (4.69), has a locally uniformly stable equilibrium at* $(z, \tilde{\theta}) = (0,0)$ *and* $\lim_{t\to\infty} z(t) = 0$, *which means that asymptotic tracking is achieved, i.e.*

$$\lim_{t\to\infty} \Big[y(t) - y_r(t)\Big] = 0 \tag{4.84}$$

*Moreover, if* $\lim_{t\to\infty} y_r^{(i)} = 0,\ i = 0, \dots, n$ *and* $F(0,0) = 0$; $\lim_{t\to\infty} x(t) = 0$.

Theorem 4.1 guarantees local asymptotic tracking in general. Nevertheless global asymptotic tracking can be achieved if Assumptions 4.1 and 4.2 are satisfied globally.

The closed-loop system (4.75)-(4.76) exhibits better parameter estimate convergence when

$$\text{rank}[F(x,u)]_{(x,u)=(X,U)} = p \tag{4.85}$$

and a *persistency of excitation condition*, similar to that formulated by Sastry and Bodson [97], is satisfied by the regressor matrix $W$.

**Example 4.2** Consider the nonlinear system already discussed in Example 4.1

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{pmatrix} = \begin{pmatrix} x_2 \\ x_3 \\ -x_3 \end{pmatrix} + \begin{pmatrix} x_1^2 \\ 0 \\ 1+x_1 \end{pmatrix} \theta + \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} u \tag{4.86}$$

$$y = x_1$$

where $\theta$ is an unknown constant parameter. It will be shown that the objective of regulating the output to zero is achieved by a dynamical adaptive controller, designed according to the algorithm above. In addition, the stability of the equilibrium $(X_1, X_2, X_3, \hat{\theta}, U) = (0, 0, \theta, \theta, -\theta)$ of (4.86) is also achieved. Since the relative degree of (4.86) is 2, the first two steps of the traditional backstepping and the above described dynamical backstepping algorithms are identical. Therefore, one can use the results of Example 4.1 as partial results for this example and complete the design of the dynamical controller by applying the new third step. Thus the function $\tau_2$ obtained in Example 4.1 is not considered the update law anymore but an intermediate tuning function. The third error variable is defined as

$$z_3 = u - \alpha_2(x_1, x_2, x_3, \hat{\theta}) \tag{4.87}$$

where

$$\alpha_2(x_1, x_2, x_3, \hat{\theta}) = -z_1 - x_3 + \frac{\partial \alpha_1}{\partial x_1}(x_2 + \hat{\theta} x_1^2) + \frac{\partial \alpha_1}{\partial \hat{\theta}} \tau_2 - c_2 z_2 \tag{4.88}$$

Consequently the closed-loop form of $\dot{z}_2$ is

$$\dot{z}_2 = -z_1 - c_2 z_2 + z_3 + \omega_2(\theta - \hat{\theta}) - \frac{\partial \alpha_1}{\partial \hat{\theta}}\left(\dot{\hat{\theta}} - \tau_2\right) \tag{4.89}$$

with

$$\omega_2 = -\frac{\partial \alpha_1}{\partial x_1} x_1^2 \tag{4.90}$$

The Lyapunov function defined for the parameter estimate error $(\theta - \hat{\theta})$ and the first two error variables is

$$V_2 = \frac{1}{2} z_1^2 + \frac{1}{2} z_2^2 + \frac{1}{2\gamma}(\theta - \hat{\theta})^2 \tag{4.91}$$

and

$$\dot{V}_2 = -c_1 z_1^2 - c_2 z_2^2 + z_2 z_3 - z_2 \frac{\partial \alpha_1}{\partial \hat{\theta}}\left(\dot{\hat{\theta}} - \tau_2\right) + \frac{(\theta - \hat{\theta})}{\gamma}\left(-\dot{\hat{\theta}} + \tau_2\right) \tag{4.92}$$

The third term in (4.92) will be cancelled at the next step.

**Step 3.** From (4.87) the time derivative of $z_3$ is

$$\dot{z}_3 = \dot{u} - \frac{\partial \alpha_2}{\partial x_1}(x_2 + \hat{\theta} x_1^2) - \frac{\partial \alpha_2}{\partial x_2}(x_3 + u) - \frac{\partial \alpha_2}{\partial x_3}\left(-x_3 + \hat{\theta}(1 + x_1)\right) - \frac{\partial \alpha_2}{\partial \hat{\theta}}\dot{\hat{\theta}} + \omega_3(\theta - \hat{\theta}) \tag{4.93}$$

with

$$\omega_3 = -\frac{\partial \alpha_2}{\partial x_1}x_1^2 - \frac{\partial \alpha_2}{\partial x_3}(1+x_1) \tag{4.94}$$

The Lyapunov function is augmented as

$$V_3 = V_2 + \frac{1}{2}z_3^2 = \frac{1}{2}z^T z + \frac{1}{2\gamma}(\theta - \hat{\theta})^2 \tag{4.95}$$

where $\gamma$ is a scalar adaptation gain. The time derivative of $V_3$ is

$$\begin{aligned} \dot{V}_3 &= -c_1 z_1^2 - c_2 z_2^2 - z_2 \frac{\partial \alpha_1}{\partial \hat{\theta}}\Big(\dot{\hat{\theta}} - \tau_2\Big) + \frac{(\theta - \hat{\theta})}{\gamma}\Big( -\dot{\hat{\theta}} + \tau_2 + \gamma\omega_3 z_3\Big) \\ &\quad + z_3\Big[z_2 + \dot{u} - \frac{\partial \alpha_2}{\partial x_1}(x_2 + \hat{\theta}x_1^2) - \frac{\partial \alpha_2}{\partial x_2}(x_3 + u) \\ &\qquad - \frac{\partial \alpha_2}{\partial x_3}\Big( -x_3 + \hat{\theta}(1+x_1)\Big) - \frac{\partial \alpha_2}{\partial \hat{\theta}}\dot{\hat{\theta}}\Big] \end{aligned} \tag{4.96}$$

The parameter estimate error $(\theta - \hat{\theta})$ is eliminated from (4.96) by selecting the update law

$$\dot{\hat{\theta}} = \tau_3 := \tau_2 + \gamma\omega_3 z_3 = \gamma[\omega_1\ \omega_2\ \omega_3]z \tag{4.97}$$

with

$$\omega_1 = x_1^2, \qquad \omega_2 = -\frac{\partial \alpha_1}{\partial x_1}x_1^2. \tag{4.98}$$

Noting that

$$\dot{\hat{\theta}} - \tau_2 = \tau_3 - \tau_2 = \gamma\omega_3 z_3 \tag{4.99}$$

$\dot{V}_3$ can be rewritten as

$$\begin{aligned} \dot{V}_3 &= -c_1 z_1^2 - c_2 z_2^2 + z_3\Big[z_2\Big(1 - \gamma\frac{\partial \alpha_1}{\partial \hat{\theta}}\omega_3\Big) + \dot{u} - \frac{\partial \alpha_2}{\partial x_1}(x_2 + \hat{\theta}x_1^2) - \frac{\partial \alpha_2}{\partial x_2}(x_3 + u) \\ &\qquad - \frac{\partial \alpha_2}{\partial x_3}\Big( -x_3 + \hat{\theta}(1+x_1)\Big) - \frac{\partial \alpha_2}{\partial \hat{\theta}}\tau_3\Big] \end{aligned} \tag{4.100}$$

Then the dynamical controller is chosen such that the bracketed term multiplying $z_3$ equals $-c_3 z_3$, namely

$$\begin{aligned} \dot{u} &= -z_2\Big(1 - \gamma\frac{\partial \alpha_1}{\partial \hat{\theta}}\omega_3\Big) + \frac{\partial \alpha_2}{\partial x_1}(x_2 + \hat{\theta}x_1^2) + \frac{\partial \alpha_2}{\partial x_2}(x_3 + u) \\ &\quad + \frac{\partial \alpha_2}{\partial x_3}\Big( -x_3 + \hat{\theta}(1+x_1)\Big) + \frac{\partial \alpha_2}{\partial \hat{\theta}}\tau_3 - c_3 z_3. \end{aligned} \tag{4.101}$$

The error coordinate transformation $z = \Phi(x, \hat{\theta}, u)$ constructed in this manner is

$$\begin{aligned} z_1 &= x_1 \\ z_2 &= x_2 + \hat{\theta}x_1^2 + c_1 x_1 \\ z_3 &= x_3 + u + x_1 + (2\hat{\theta}x_1 + c_1)(x_2 + \hat{\theta}x_1^2) + c_2(x_2 + \hat{\theta}x_1^2 + c_1 x_1) \\ &\quad + \gamma x_1^4\Big[x_1 + (2\hat{\theta}x_1 + c_1)(x_2 + \hat{\theta}x_1^2 + c_1 x_1)\Big] \end{aligned} \tag{4.102}$$

and its associated Jacobian matrix has the following triangular form

$$\frac{\partial \Phi(\cdot)}{\partial x} = \begin{bmatrix} 1 & 0 & 0 \\ * & 1 & 0 \\ * & * & 1 \end{bmatrix} \tag{4.103}$$

Therefore the observability condition (4.16) is satisfied globally and the inverse of the transformation (4.102) is defined globally as

$$\begin{aligned} x_1 &= z_1 \\ x_2 &= z_2 - \hat{\theta} z_1^2 - c_1 z_1 \\ x_3 &= z_3 - u - z_1 - (2\hat{\theta} z_1 + c_1)(z_2 - c_1 z_1) - c_2 z_2 - \gamma z_1^4 \Big[ z_1 + (2\hat{\theta} z_1 + c_1) z_2 \Big] \end{aligned} \tag{4.104}$$

By Theorem 4.1, the equilibrium $(X^T, \hat{\theta}, U) = (0, 0, \theta, \theta, -\theta)$ is globally stable. We have verified this result by computer simulations for a nominal "unknown" parameter set at $\theta = 2$. Global asymptotic behaviour of the dynamically controlled state variable $x_1$ and boundedness of the parameter estimate is obtained for all the initial conditions shown in Figures 4.3, 4.4 and 4.5. The design parameters were $c_1 = 3$, $c_2 = 4$, $c_3 = 3$ and $\gamma = 1$. If the initial conditions are chosen close to the desired equilibrium values, small positive values of the design parameters may achieve good performance. Otherwise, i.e. for initial conditions chosen far from the desired equilibrium values, the design parameters should be increased for a suitably fast convergence. However, this can produce adverse effects such as oscillations in the transient behaviour of the state variables and parameter estimate. These oscillations can be reduced by decreasing some design parameters. Therefore there is a trade off between performance and fast convergence of the variables to the desired values. This is shown in Figure 4.6, which was obtained with the design parameters $c_1 = 6$, $c_2 = 7$, $c_3 = 8$ and $\gamma = 0.001$. The stability proof above does not guarantee that the parameter estimate converges to the actual unknown parameter value. This is shown in Figure 4.7 which was obtained for different initial conditions, nevertheless global stabilization is achieved for all the initial conditions.

## 4.4 Example: DAB Control of Nontriangular Systems

A limitation of the traditional adaptive backstepping algorithm [48], [61] has been its applicability only to systems which can be transformed to a triangular form. In particular the system should be transformable into either the PPF form or PSF form to guarantee its stabilization via backstepping. The above new algorithm does not require any particular form and can be applied to nonlinear systems *not* in a triangular form and which are not transformable into these canonical forms. This feature is illustrated in the following example which considers a nontriangular system.

**Example 4.3** Consider the following third order nonlinear system

$$\dot{x} = f_0(x) + f_1(x)\theta + g_0(x)u = \begin{pmatrix} x_2 \\ x_3 \\ 0 \end{pmatrix} + \begin{pmatrix} x_1 x_3 \\ 0 \\ 0 \end{pmatrix} \theta + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} u \tag{4.105}$$

where $\theta$ is a *positive* unknown parameter. Note that $\dot{x} = f_0(x) + g_0(x)u$ is already in linear form; therefore to check whether or not this system is transformable into either

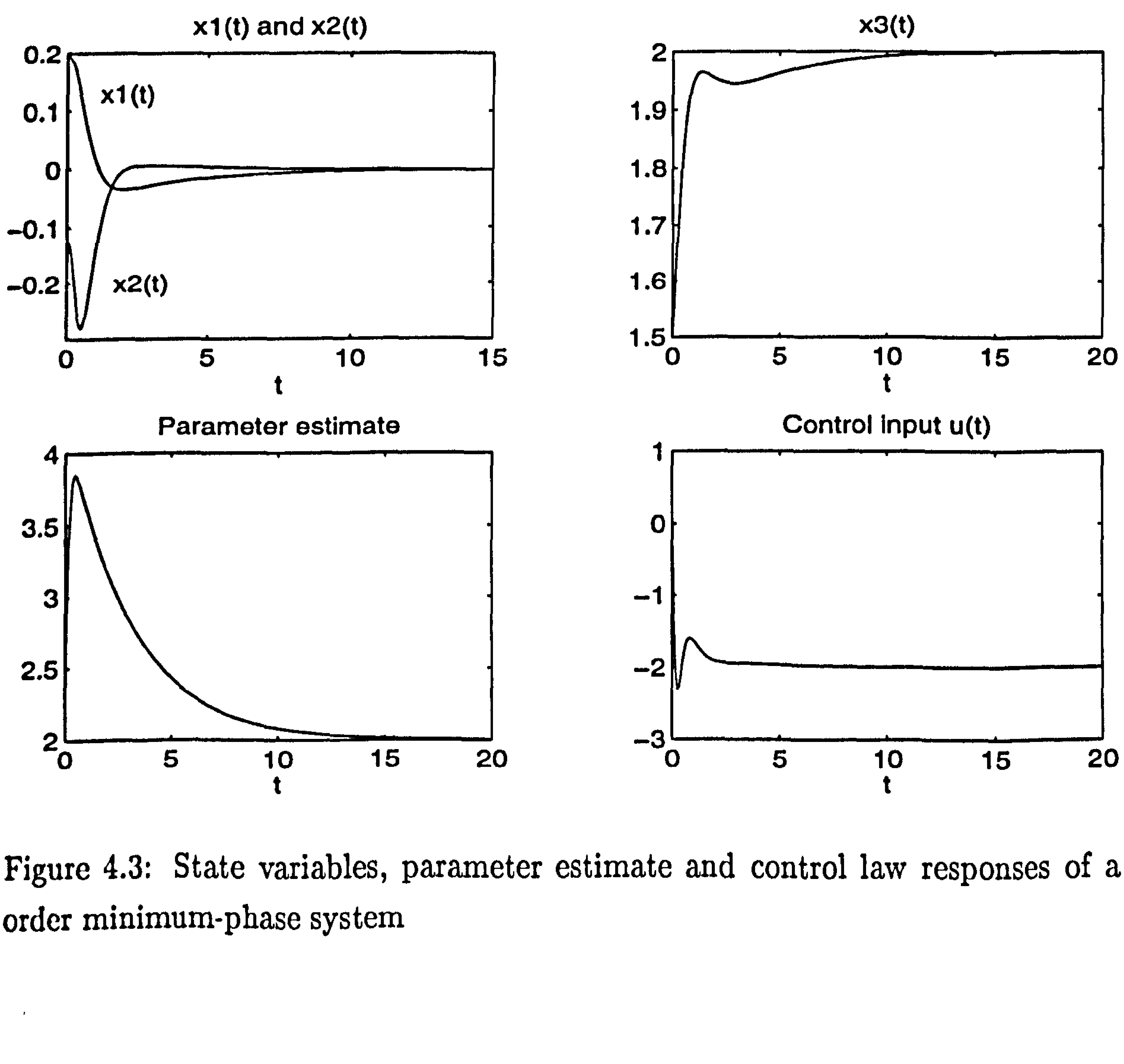


Figure 4.3: State variables, parameter estimate and control law responses of a third order minimum-phase system

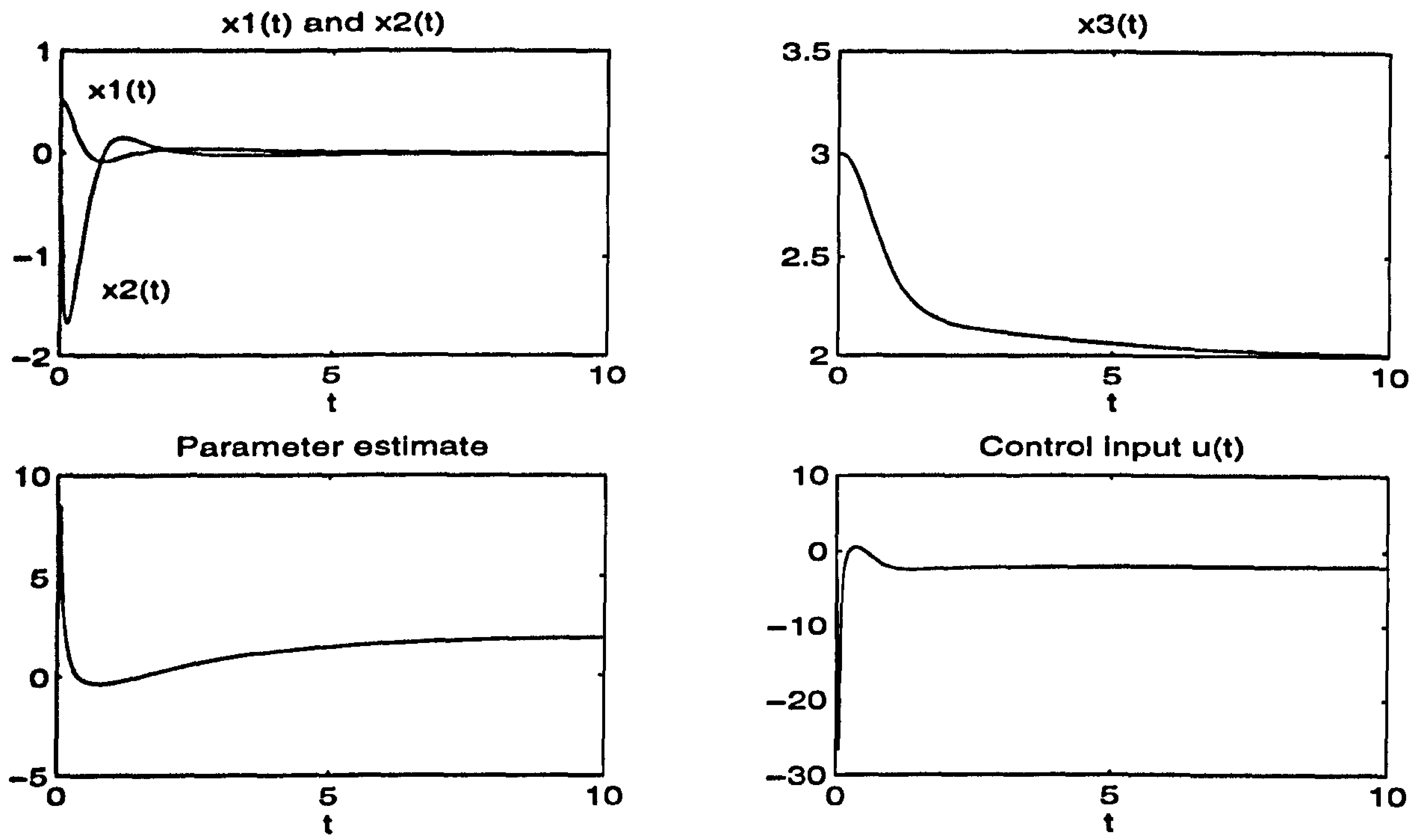


Figure 4.4: Controlled state variables, parameter estimate and control law responses of a third order minimum-phase nonlinear system for the initial conditions $(x, \hat{\theta}, u)(0) = (0.5, -0.5, 3, 1.5, -1)$

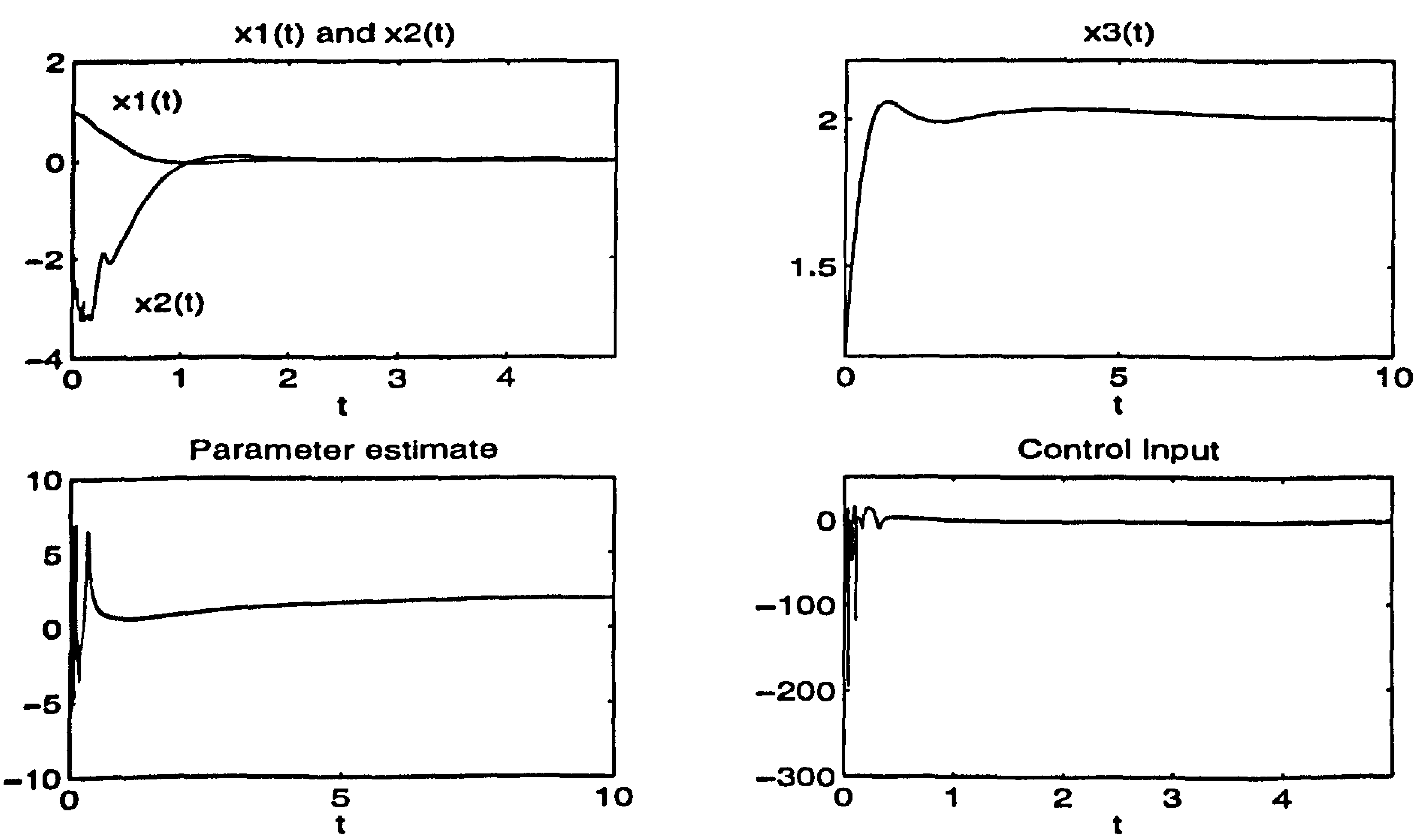


Figure 4.5: Controlled state variables, parameter estimate and control law responses of a third order minimum-phase nonlinear system for the initial conditions $(x, \hat{\theta}, u)(0) = (1, -2, 1.2, 0.6, -1)$

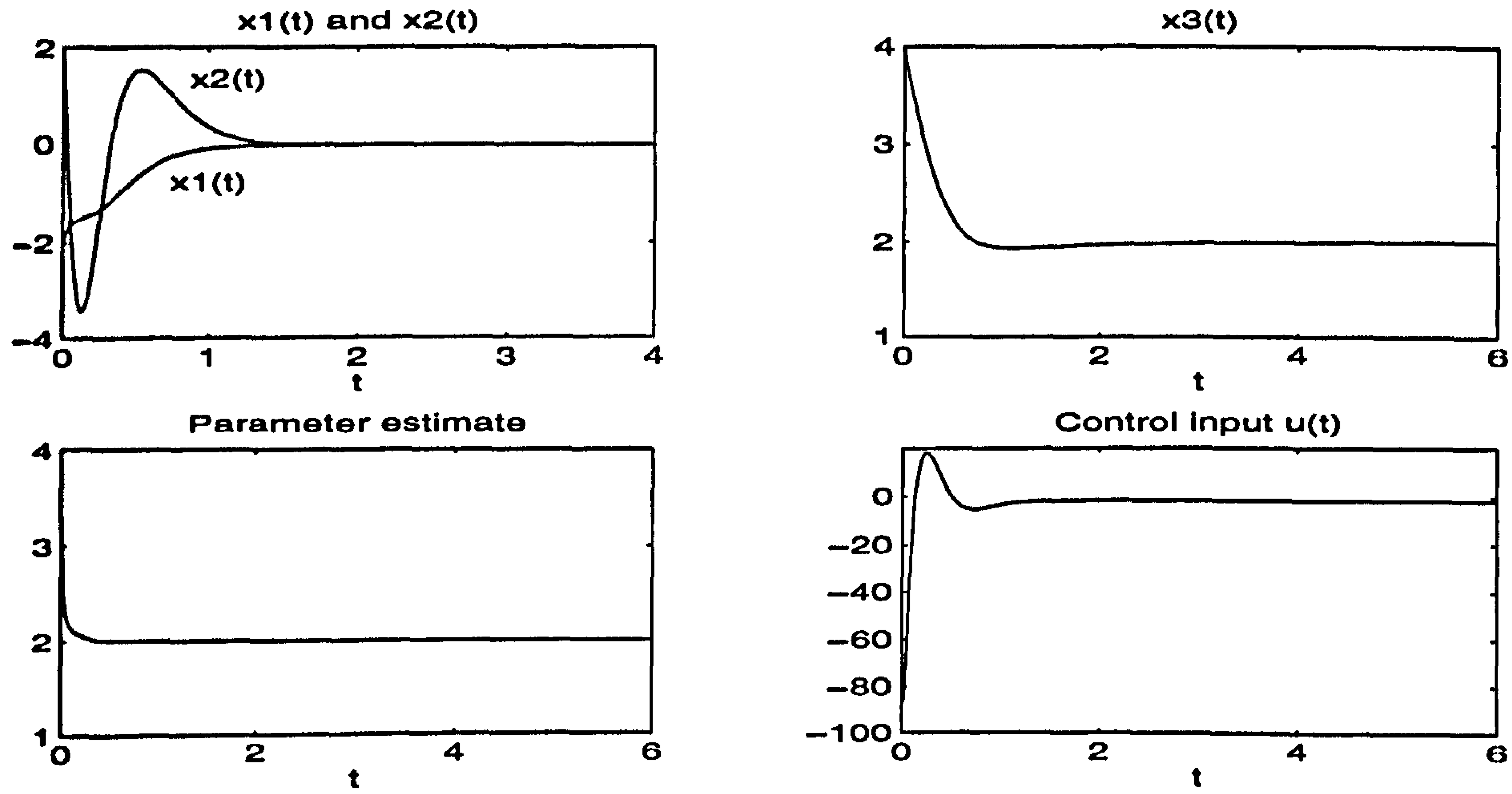


Figure 4.6: Controlled state variables, parameter estimate and control law responses of a third order minimum-phase nonlinear system for the initial conditions $(x, \hat{\theta}, u)(0) = (-2, 2, 4, 4, -3)$

the PPF form or the PSF form, it suffices to compute the vector field $ad_{f_0}g_0$

$$ad_{f_0}g_0 = -\begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \tag{4.106}$$

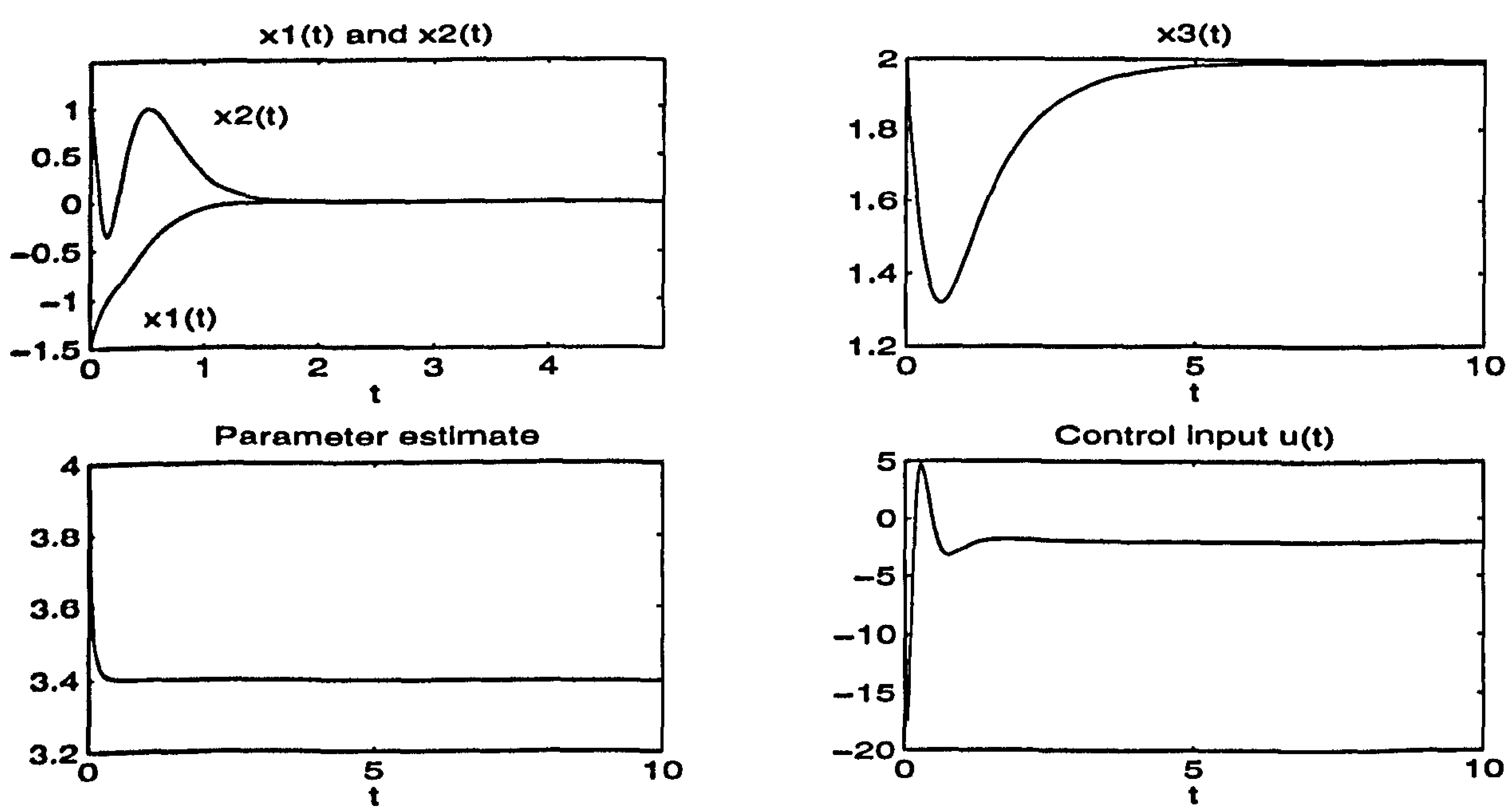


Figure 4.7: Controlled state variables, parameter estimate and control law responses of a third order minimum-phase nonlinear system for the initial conditions $(x, \hat{\theta}, u)(0) = (-1.5, 1, 2, 4, 0)$

which together with $g_0$, allows the definition of the distributions

$$\mathcal{G}_0 = \text{span}\{g_0\}, \qquad \mathcal{G}_1 = \text{span}\{g_0, ad_{f_0} g_0\} \tag{4.107}$$

The Lie bracket $[g_0, f_1]$

$$[g_0, f_1] := \frac{\partial f_1}{\partial x} g_0 - \frac{\partial g_0}{\partial x} f_1 = \begin{pmatrix} x_1 \\ 0 \\ 0 \end{pmatrix} \tag{4.108}$$

violates the parametric strict-feedback condition of Theorem 2.4 and therefore the system (4.105) is not transformable into the PSF form. The Lie bracket $[g_0, f_1]$ also violates the parametric pure-feedback condition of Theorem 2.3. Hence, traditional backstepping is not applicable to this system. Nevertheless we recall from Section 3.4 that, for $\theta = 1$, the system 4.61 is locally input-output linearizable with respect to the output function

$$y = x_1 \exp(-x_2) \tag{4.109}$$

Thus, for this output function, we can apply the DAB algorithm for the regulation of the output to the desired set point $y_r$, to obtain the coordinate transformation

$$\begin{aligned} \dot{z}_1 &= x_1 \exp(-x_2) - y_r \\ \dot{z}_2 &= h_1(x, \hat{\theta}) = x_2 \exp(-x_2) + x_1 x_3 (\hat{\theta} - 1) \exp(-x_2) + c_1 z_1 \\ \dot{z}_3 &= h_2(x, \hat{\theta}, u) = z_1 + \frac{\partial h_1}{\partial x_1}(x_2 + \hat{\theta} x_1 x_3) + \frac{\partial h_1}{\partial x_2} x_3 + \frac{\partial h_1}{\partial x_3} u + \frac{\partial h_1}{\partial \hat{\theta}} \tau_2 + c_2 z_2 \end{aligned} \tag{4.110}$$

with the regressor functions

$$\begin{aligned} \omega_1 &= x_1 x_3 \exp(-x_2) \\ \omega_2 &= \frac{\partial h_1}{\partial x_1} x_1 x_3 \\ \omega_3 &= \frac{\partial h_2}{\partial x_1} x_1 x_3 \end{aligned} \tag{4.111}$$

and the tuning functions

$$\begin{aligned} \tau_1 &= \gamma \omega_1 z_1 \\ \tau_2 &= \gamma(\omega_1 z_1 + \omega_2 z_2) \\ \tau_3 &= \gamma[\omega_1 \ \omega_2 \ \omega_3] z \end{aligned} \tag{4.112}$$

where $\gamma$ is a scalar adaptation gain. The update law is

$$\dot{\hat{\theta}} = \tau_3 \tag{4.113}$$

and the dynamical adaptive control law is given by

$$\dot{u} = \frac{1}{\frac{\partial h_2}{\partial u}}\left[-z_2\left(1+\frac{\partial h_1}{\partial \hat{\theta}}\gamma\omega_3\right) - \frac{\partial h_2}{\partial x_1}(x_2+\hat{\theta}x_1x_3) - \frac{\partial h_2}{\partial x_2}x_3 - \frac{\partial h_2}{\partial x_3}u - \frac{\partial h_2}{\partial \hat{\theta}}\tau_3 - c_3z_3\right] \tag{4.114}$$

Note that in this case

$$\frac{\partial h_2}{\partial u} = \frac{\partial h_1}{\partial x_3} = (\hat{\theta}-1)x_1\exp(-x_2) \tag{4.115}$$

Thus $\hat{\theta} = 1$ and $x_1 = 0$ are singular values for the designed controller. Therefore local stabilization is possible for equilibrium points far from these singularity values. Furthermore, from the known information that $\theta > 0$, we can select initial conditions with $\hat{\theta}(0) > 1$ and an adaptation gain small (or zero) to guarantee that $\hat{\theta} > 1$ and thus guaranteeing stabilization. We carried out computer simulations for a nominal unknown parameter $\theta = 1$ and a desired set point $y_r = 2$. Figure 4.8 shows the controlled trajectories and parameter estimate for the design parameters $c_1 = 4$, $c_2 = 3$, $c_3 = 2$ and $\gamma = 0.001$. The regulation of the state variables to the desired equilibrium point $x = (2, 0, 0)$ is achieved whilst $\hat{\theta}$ converges to a value greater than one. Figure 4.9 shows that by choosing the initial condition $\hat{\theta}(0) > 1$ we can turn off the adaptation process by selecting $\gamma = 0$, and still achieve stabilization of the state variables to the desired equilibrium values.

Consider the alternative output function

$$y = x_1 \tag{4.116}$$

for the regulation of the system (4.61). Before applying the DAB algorithm one must check whether or not the system with this new output is minimum phase. Suppose that the parameter $\theta$ is known and define the change of coordinates

$$z := \Phi(x) = \begin{pmatrix} x_1 \\ x_2 + \theta x_1 x_3 \\ x_2 \end{pmatrix} \tag{4.117}$$

to obtain the normal form

$$\begin{aligned}
\dot{z}_1 &= z_2 \\
\dot{z}_2 &= (1+\theta z_2)\frac{(z_2 - z_3)}{\theta z_1} + \theta z_1 u \\
\dot{z}_3 &= \frac{(z_2 - z_3)}{\theta z_1}
\end{aligned} \tag{4.118}$$

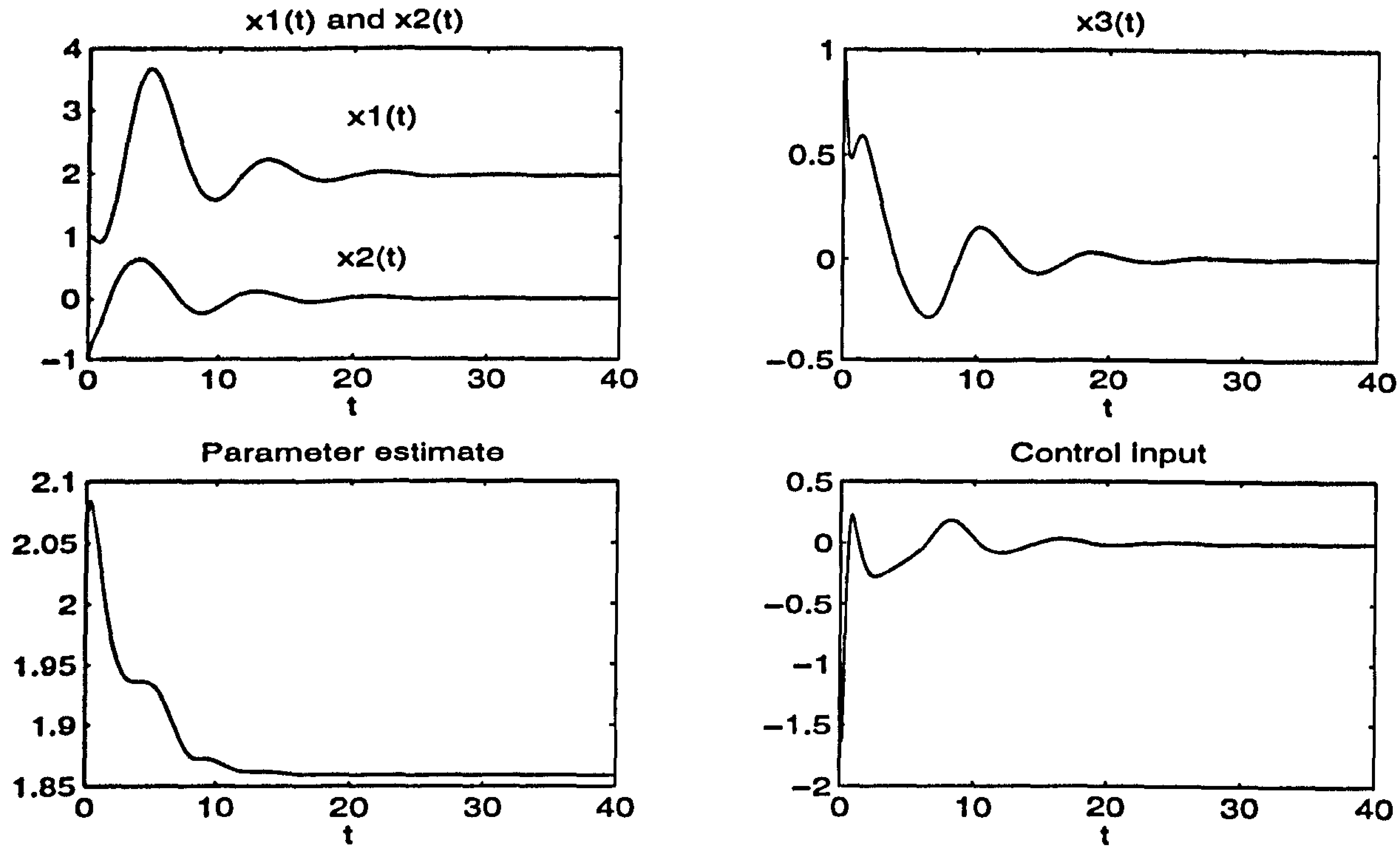


Figure 4.8: State variables, parameter estimate and control responses of a nontriangular system regulated by a DAB controller

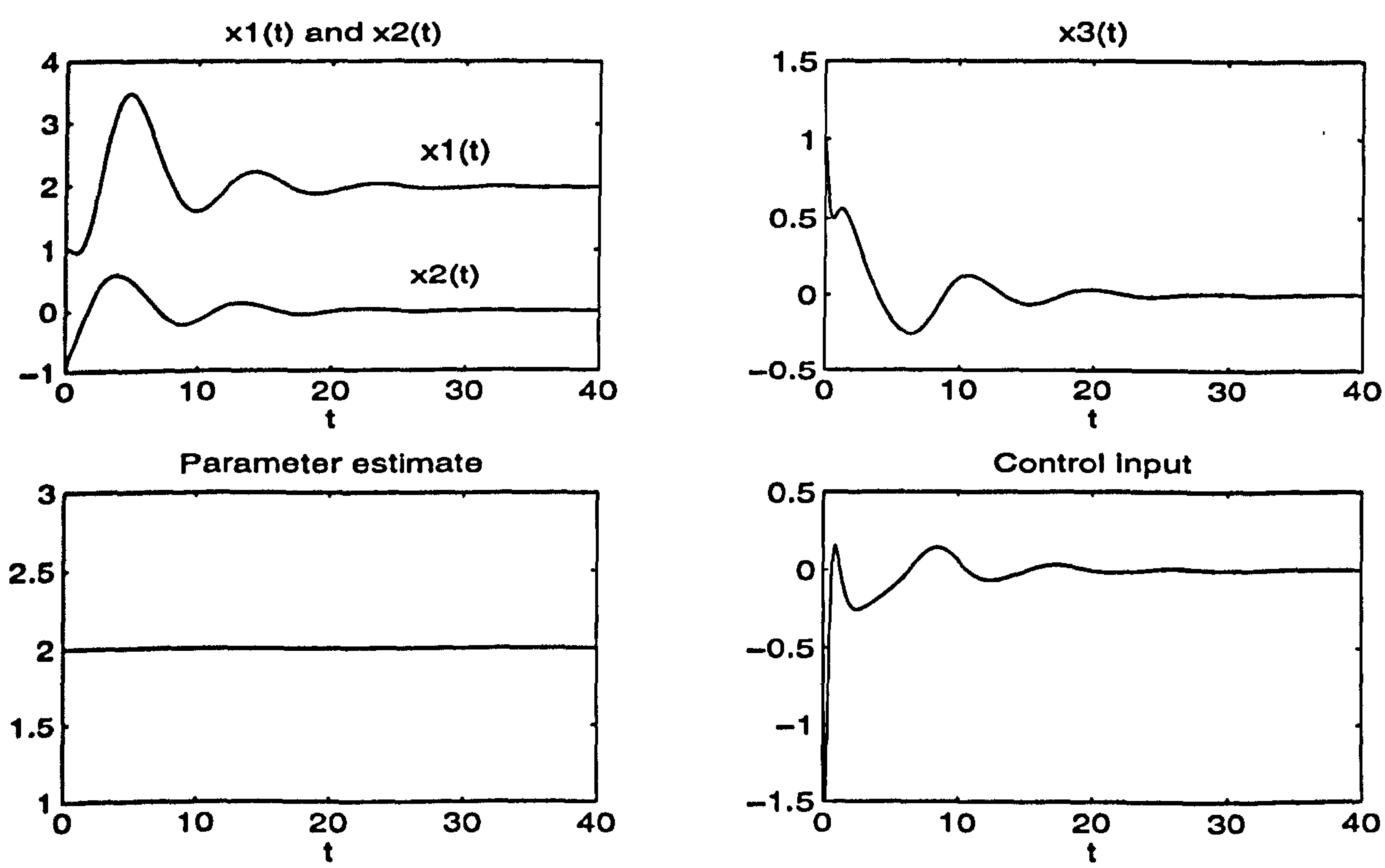


Figure 4.9: State variables, parameter estimate and control responses of a nontriangular system regulated by a DAB controller with adaptation turned off

Note that the relative degree is 2 for the corresponding output if the equilibrium has coordinates $z_1 = x_1 = X_1 \neq 0$. Therefore, from the dynamical equations (4.105) only local stabilization of equilibria of the form $(x_1, x_2, x_3) = (X_1, 0, 0)$ is possible for this system. To check that (4.117) is a valid transformation, the Jacobian matrix

$$\frac{\partial\Phi(x)}{\partial x} = \begin{pmatrix} 1 & 0 & 0 \\ \theta x_3 & 1 & \theta x_1 \\ 0 & 1 & 0 \end{pmatrix} \tag{4.119}$$

is obtained. This matrix has rank 3 if the coordinate $x_1$ is different from zero. For equilibria with $x_1 \neq 0$, the minimum phase property is proved by noting that the third equation of the normal form (4.118) can be reduced to the internal dynamics

$$\dot{z}_3 = -\frac{z_3}{\theta X_1} \tag{4.120}$$

and $z_3 = 0$ is an asymptotically stable equilibrium of the internal dynamics if $X_1$ is positive. Hence, we conclude that the system (4.105) is locally minimum phase for equilibria of the form $(x_1, x_2, x_3) = (X_1, 0, 0)$ with $X_1 > 0$. The application of the new DAB algorithm gives the control-dependent error variables

$$\begin{aligned} z_1 &= x_1 - X_1 \\ z_2 &= \alpha_1(x, \hat{\theta}) = x_2 + \omega_1\hat{\theta} + c_1(x_1 - X_1) \\ z_3 &= \alpha_2(x, \hat{\theta}, u) = x_1 - X_1 + x_3 + (c_1 + \hat{\theta}x_3)x_2 + \omega_2\hat{\theta} + \hat{\theta}x_1 u + \omega_1\tau_2 + c_2 z_2 \end{aligned} \tag{4.121}$$

with the regressor functions

$$\begin{aligned} \omega_1 &= x_1 x_3 \\ \omega_2 &= (c_1 + x_3\hat{\theta})x_1 x_3 \\ \omega_3 &= \frac{\partial\alpha_1}{\partial x_1} x_1 x_3 \end{aligned} \tag{4.122}$$

The validity of the transformation (4.121) is verified by computing its Jacobian matrix at the equilibrium point $(x^T, u) = (X_1, 0, 0, 0)$

$$\left[\frac{\partial\Phi}{\partial x}\right]_{(X_1,0,0,0)} = \begin{pmatrix} 1 & 0 & 0 \\ c_1 & 1 & \hat{\theta}X_1 \\ 1 & c_1 & 1 + c_1\hat{\theta}X_1 \end{pmatrix} \tag{4.123}$$

This matrix has rank 3 at the equilibrium point, hence (4.121) defines a local coordinate transformation. This result also verifies the observability condition. The tuning functions obtained at the successive steps are

$$\begin{aligned} \tau_1 &= \gamma\omega_1 z_1 \\ \tau_2 &= \tau_1 + \gamma\omega_2 z_2 = \gamma(\omega_1 z_1 + \omega_2 z_2) \\ \tau_3 &= \tau_2 + \gamma\omega_3 z_3 = \gamma[\omega_1\ \omega_2\ \omega_3]z \end{aligned} \tag{4.124}$$

where $\gamma > 0$ is a scalar adaptation gain. The resulting adaptive controller has the parameter update law $\dot{\hat{\theta}} = \tau_3$ and the dynamical feedback control

$$\dot{u} = \frac{1}{\frac{\partial \alpha_2}{\partial u}}\left[ - z_2(1+\gamma\omega_1\omega_3) - \frac{\partial \alpha_2}{\partial x_1}(x_2 + x_1 x_3 \hat{\theta}) - \frac{\partial \alpha_2}{\partial x_2}x_3 - \frac{\partial \alpha_2}{\partial x_3}u - \frac{\partial \alpha_2}{\partial \hat{\theta}}\tau_3 - c_3 z_3\right] \quad (4.125)$$

Note that

$$\frac{\partial \alpha_2}{\partial u} = \hat{\theta} x_1 \quad (4.126)$$

and, as discussed above, only equilibria with coordinate $X_1 > 0$ should be considered to avoid singularities of the control law. In this case the parameter estimate value $\hat{\theta} = 0$ is a singular value for the designed controller. The closed-loop system in the $(z, \tilde{\theta})$-coordinates yields

$$\dot{z} = \begin{bmatrix} -c_1 & 1 & 0 \\ -1 & -c_2 & 1+\gamma\omega_1\omega_3 \\ 0 & -1-\gamma\omega_1\omega_3 & -c_3 \end{bmatrix} z + \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} \tilde{\theta} \quad (4.127)$$

$$\dot{\tilde{\theta}} = \gamma\,[\omega_1\ \omega_2\ \omega_3]\,z \quad (4.128)$$

By virtue of Theorem 4.1 asymptotic stabilization of the output to the desired value $X_1$ is achieved. Computer simulations were carried out for the nominal value of the unknown parameter set at $\theta = 1$ and a desired output $X_1 = 2$. Figure 4.10 shows the satisfactory asymptotic behaviour of the controlled responses for the design parameters $c_1 = 2$, $c_2 = 3$, $c_3 = 3$ and $\gamma = 1$. Figure 4.11 shows the state variables and parameter estimate for different initial conditions and an adaptation gain $\gamma = 0.001$. Note that, since $\theta = 0$ is a singular value, we can select initial conditions with $\hat{\theta}(0) > 0$ and assign small values to the adaptation gain to achieve stabilization avoiding singularities.

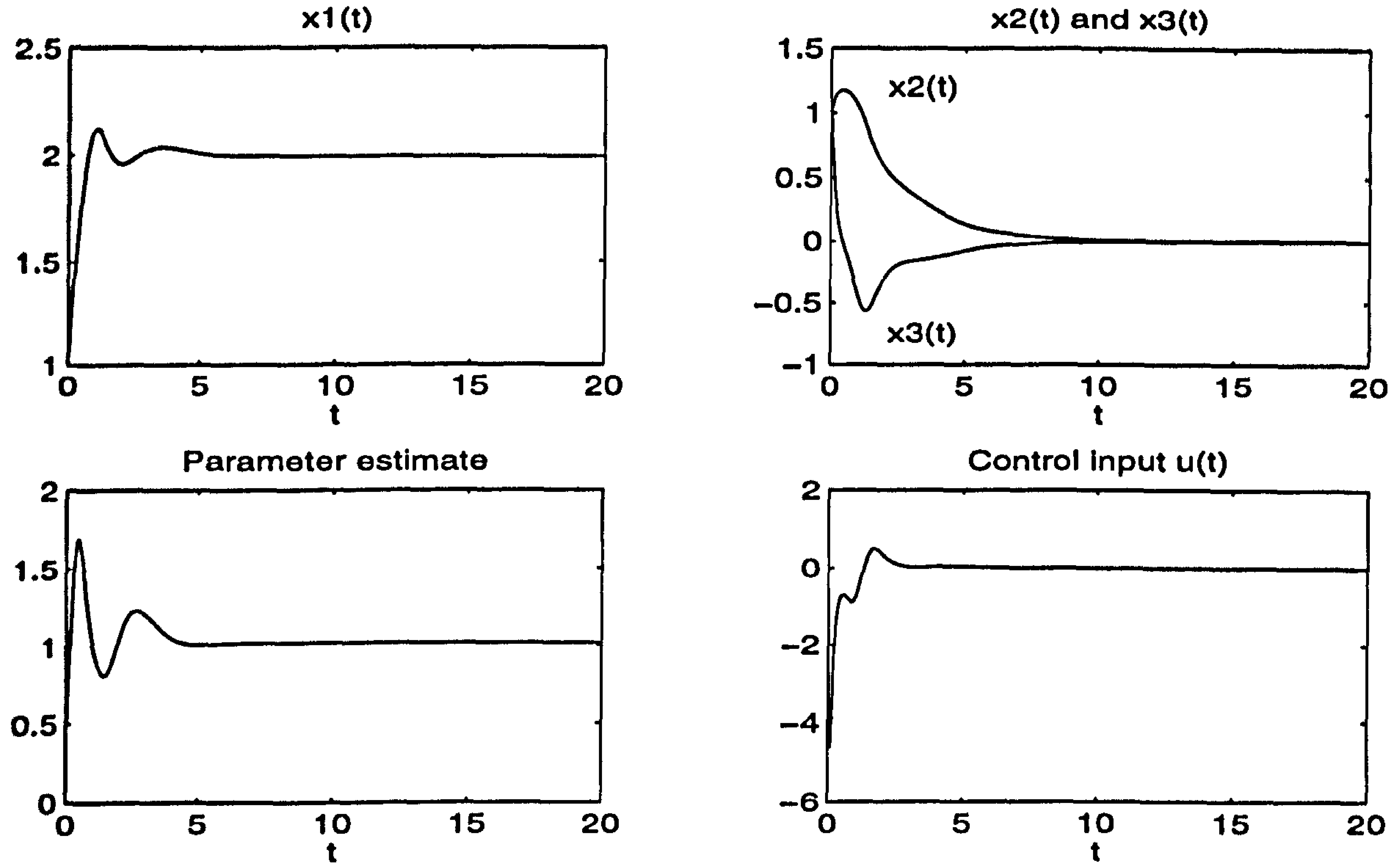


Figure 4.10: State variables, parameter estimate and control responses of a nontriangular system regulated by a DAB controller

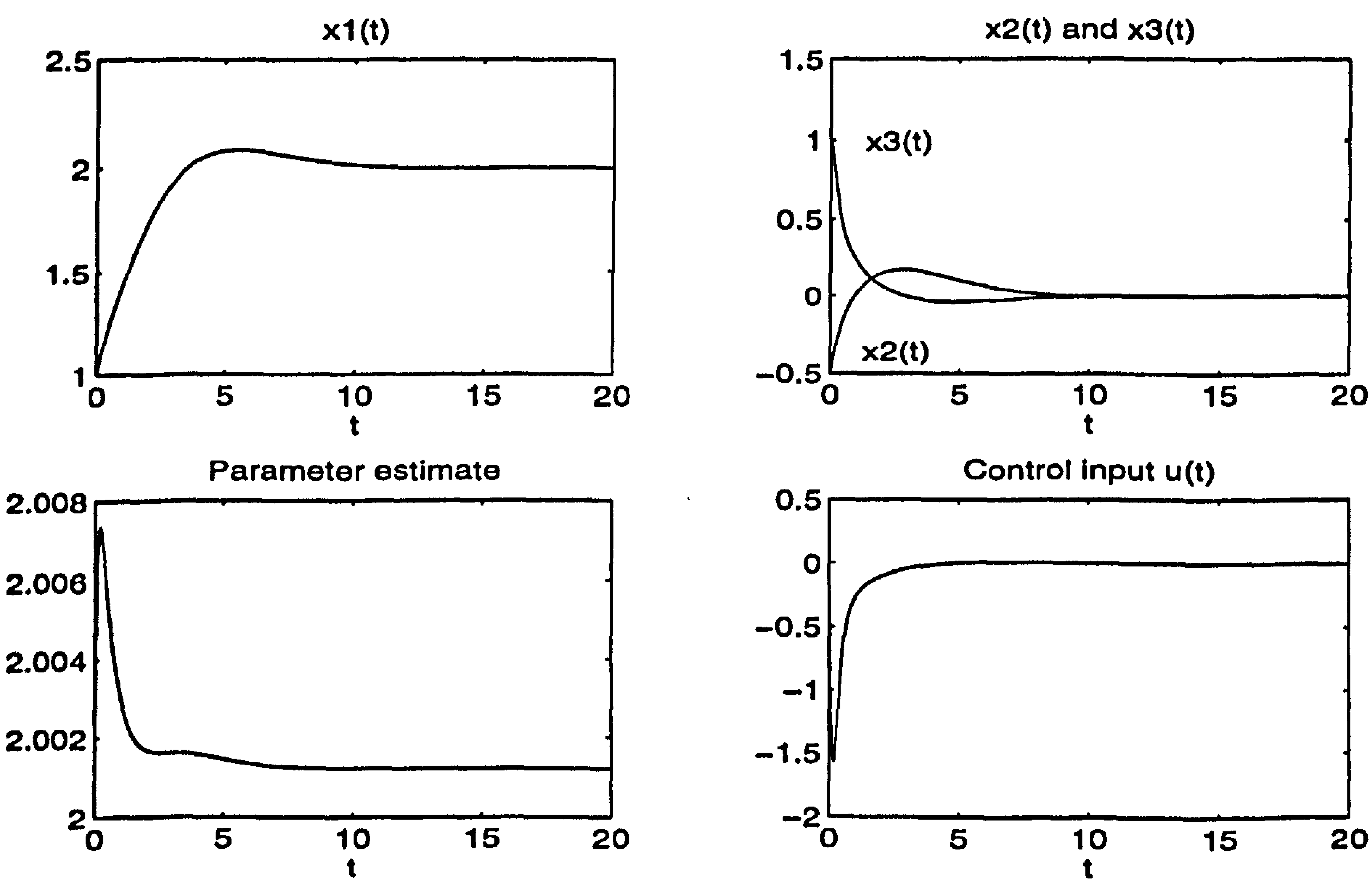


Figure 4.11: State variables, parameter estimate and control responses of a nontriangular system regulated by a DAB controller for the initial conditions $(x, \hat{\theta}, u)(0) = (1, -0.5, 1, 2, 0.2)$

# Chapter 5

# DAB Control of Power Converters

## 5.1 Introduction

We consider in this chapter the application of the DAB algorithm described in Section 4.2 to practical systems with a discontinuous form. We study the design of dynamical adaptive Pulse-Width-Modulated (PWM) controllers with robust stability properties for the regulation of DC-to-DC power converters in the presence of perturbation inputs.

Since, for practical reasons, the regulation of DC-to-DC power converters is traditionally performed by means of Pulse-Width-Modulation (PWM) feedback control strategies, we propose the use of this technique in combination with the DAB algorithm for the robust control of power converters of the Boost and Buck-Boost types. This new approach extends the application of the DAB algorithm described in Section 4.2 to nonlinear systems with a discontinuous form whose average models (infinite frequency assumption) are observable and minimum phase.

The design of adaptive control strategies for the regulation of power converters in the presence of parametric uncertainty is very important in practical applications concerning regulated power supplies. The combined PWM-DAB approach proposed here provides a systematic solution (see [112]). Computer simulations are used to illustrate the performance of the backstepping controllers.

## 5.2 DC-to-DC Power Converters with Uncertainties

DC-to-DC power converters are electronic systems used in the conversion of energy. They can be modelled as nonlinear systems of discontinuous nature. As stated in Section 1.3.2 the state space models of switchmode DC-to-DC power converters can be represented by the following switch-controlled dynamical nonlinear system [104]

$$\dot{\xi} = f(\xi) + g(\xi)u + \eta \tag{5.1}$$

where $f(\cdot)$ and $g(\cdot)$ are smooth vector fields defined on an open set $R_0$ of $\Re^n$, $\eta$ is a constant vector and $u$ denotes the switch position function, acting as a control input, which takes one of the two values in the binary set $\{0, 1\}$. Since the power converters to be considered in this section are of the Boost and Buck-Boost types (see Figure 5.1), the vector fields $f(\cdot)$ and $g(\cdot)$ are defined on an open set $R_0$ of $\Re^2$. The mathematical model of the Boost converter describing the input inductor current $I(t)$ and the output capacitor voltage $V(t)$ has the form

$$\begin{aligned} \dot{I}(t) &= -\frac{1}{L}(1-u)V(t) + \frac{E}{L} \\ \dot{V}(t) &= \frac{1}{C}(1-u)I(t) - \frac{1}{RC}V(t) \end{aligned} \tag{5.2}$$

whereas the corresponding to the Buck-Boost converter is given by

$$\begin{aligned} \dot{I}(t) &= \frac{1}{L}(1-u)V(t) + \frac{E}{L}u \\ \dot{V}(t) &= -\frac{1}{C}(1-u)I(t) - \frac{1}{RC}V(t) \end{aligned} \tag{5.3}$$

where $L$, $C$ and $R$ are respectively the inductance, capacitance and resistance values of the circuit components. The quantity $E$ represents the constant value of the external voltage source and the control input function $u$ is the *switch position function* taking values in the binary set $\{0, 1\}$.

Defining the state vector

$$\begin{pmatrix} \xi_1(t) \\ \xi_2(t) \end{pmatrix} = \begin{pmatrix} I(t) \\ V(t) \end{pmatrix} \tag{5.4}$$

the following relations can be identified for the Boost converter

$$f(\xi) = \begin{pmatrix} -\frac{1}{L}\xi_2 \\ \frac{1}{C}\xi_1 - \frac{1}{RC}\xi_2 \end{pmatrix}, \quad g(\xi) = \begin{pmatrix} \frac{1}{L}\xi_2 \\ -\frac{1}{C}\xi_1 \end{pmatrix}, \quad \eta = \begin{pmatrix} \frac{E}{L} \\ 0 \end{pmatrix} \tag{5.5}$$

and the Buck-Boost converter

$$f(\xi) = \begin{pmatrix} \frac{1}{L}\xi_2 \\ -\frac{1}{C}\xi_1 - \frac{1}{RC}\xi_2 \end{pmatrix}, \quad g(\xi) = \begin{pmatrix} -\frac{1}{L}\xi_2 + \frac{E}{L} \\ \frac{1}{C}\xi_1 \end{pmatrix} \quad \eta = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{5.6}$$

for the general model (5.1).

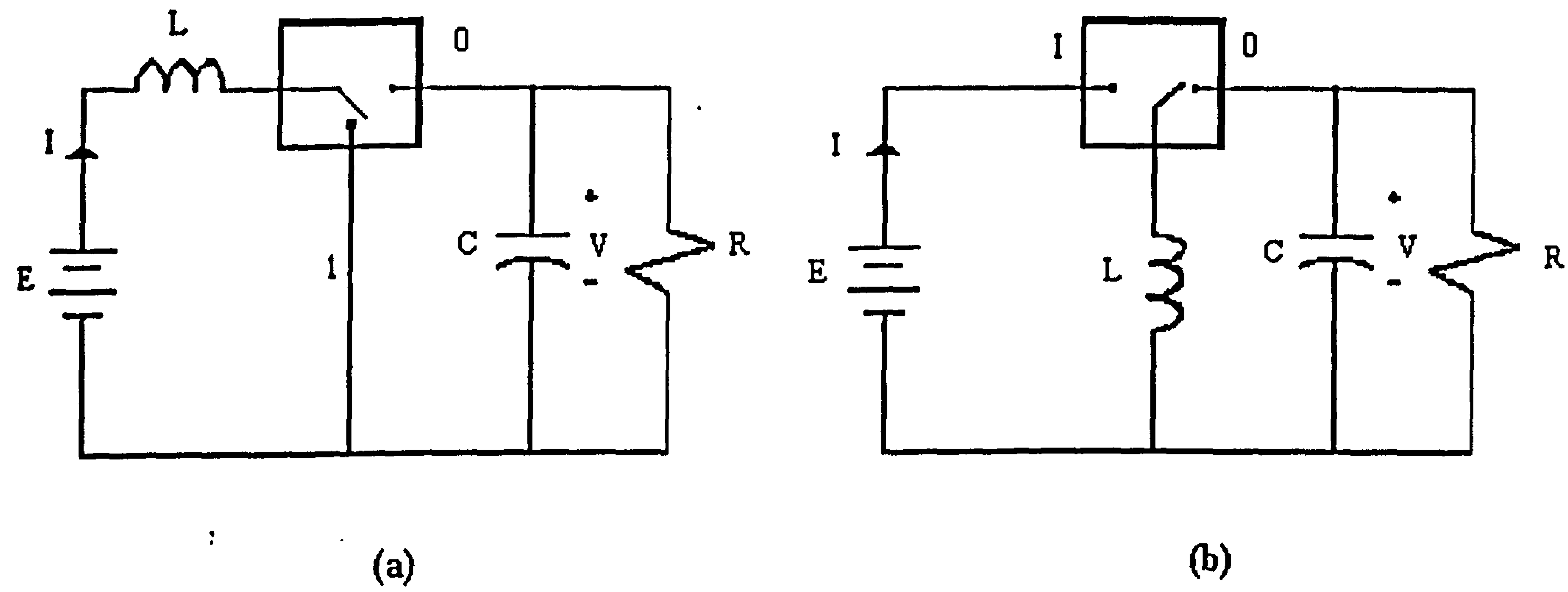


Figure 5.1: (a) Boost converter circuit. (b) Buck-Boost converter circuit

PWM feedback control strategies consist, ideally, of switching; to the $u = 1$ position at the beginning of each duty cycle (period) and changing to the $u = 0$ position once within the duty cycle. The fraction of the duty cycle for which the switch position is at $u = 1$ is known as the duty ratio $\mu$. Since the conversion of DC power is performed by semiconductor devices which switch at high frequency, it is assumed that the duty cycle has an infinitesimally small period. This assumption allows one to characterize ***average*** PWM models of power converters. These are obtained by replacing in (5.1) the discontinuous switch position $u$ by the duty ratio function $\mu$ [104, 107]

$$\dot{\xi} = f(\xi) + g(\xi)\mu + \eta. \tag{5.7}$$

The state vector thus obtained has an average connotation. This simplification has the advantage of reducing the system problem defined by (5.1) to a standard control design problem in which the duty cycle function, taking values in the continuous interval $0 \leqslant \mu \leqslant 1$, plays the role of the control input. However, the control function generated by a nonlinear control design technique may exhibit values outside this interval. This drawback is overcome by incorporating a limiting function of the form

$$\mu_r(t) = \begin{cases} 1 & \text{for} \quad \mu(t) \geqslant 1 \\ \mu(t) & \text{for} \quad 0 < \mu(t) < 1 \\ 0 & \text{for} \quad \mu(t) \leqslant 0 \end{cases} \tag{5.8}$$

between the controller and the pulse width modulator (see Figure 5.2), bounding the duty ratio function values to the closed interval $[0,1]$. The above physical restriction on the values of the duty ratio function results in *local* stabilization of the state variables and is a well-known limitation of linear and nonlinear feedback control designs for DC-to-DC power converters (see, for instance, [53, 101, 104, 107, 113]).

A PWM feedback control strategy for the specification of the switch position function $u$, occurring at regularly sampled instants of time, is given by

$$u = \begin{cases} 1 & \text{for} \quad t_k \leqslant t < t_k + \mu_r(t_k)T \\ 0 & \text{for} \quad t_k + \mu_r(t_k)T \leqslant t < t_k + T \end{cases} \tag{5.9}$$

$$t_k + T = t_{k+1}, \quad k = 0, 1, \ldots$$

Under the simplifying assumption that all the circuit parameters are perfectly known, a number of feedback control strategies has been proposed for the regulation of power supplies ([103],[104],[107],[110]). This assumption, however, may be invalid in practical situations. More frequently one has a partial knowledge of the values of the converter circuit components and of the external voltage source. Commonly, ageing effects on such components alter the known nominal values of the circuit parameters. These issues justify the need for an adaptive control strategy for the feedback regulation of switch-controlled devices delivering constant power to loads.

Adaptive feedback control of DC-to-DC power supplies has been proposed by Sira-Ramírez *et al* [113]. The adaptive PWM strategy adopted in that contribution was based upon an extension of the results of Sastry and Isidori [98] to discontinuous feedback control. Sira-Ramírez *et al* proposed in [109] an overparameterized adaptive backstepping scheme based upon the backstepping algorithm in [48]. However, due to overparameterization, the dimensionally increased adaptive controller exhibited poor parameter convergence. More recently our new non-overparameterized DAB algorithm described in Section 4.2 has been applied to the regulation of switch-controlled power supplies (see [111, 112]). This approach yields asymptotic stabilization of power supplies when all the circuit parameters are unknown, as will be shown in the next section.

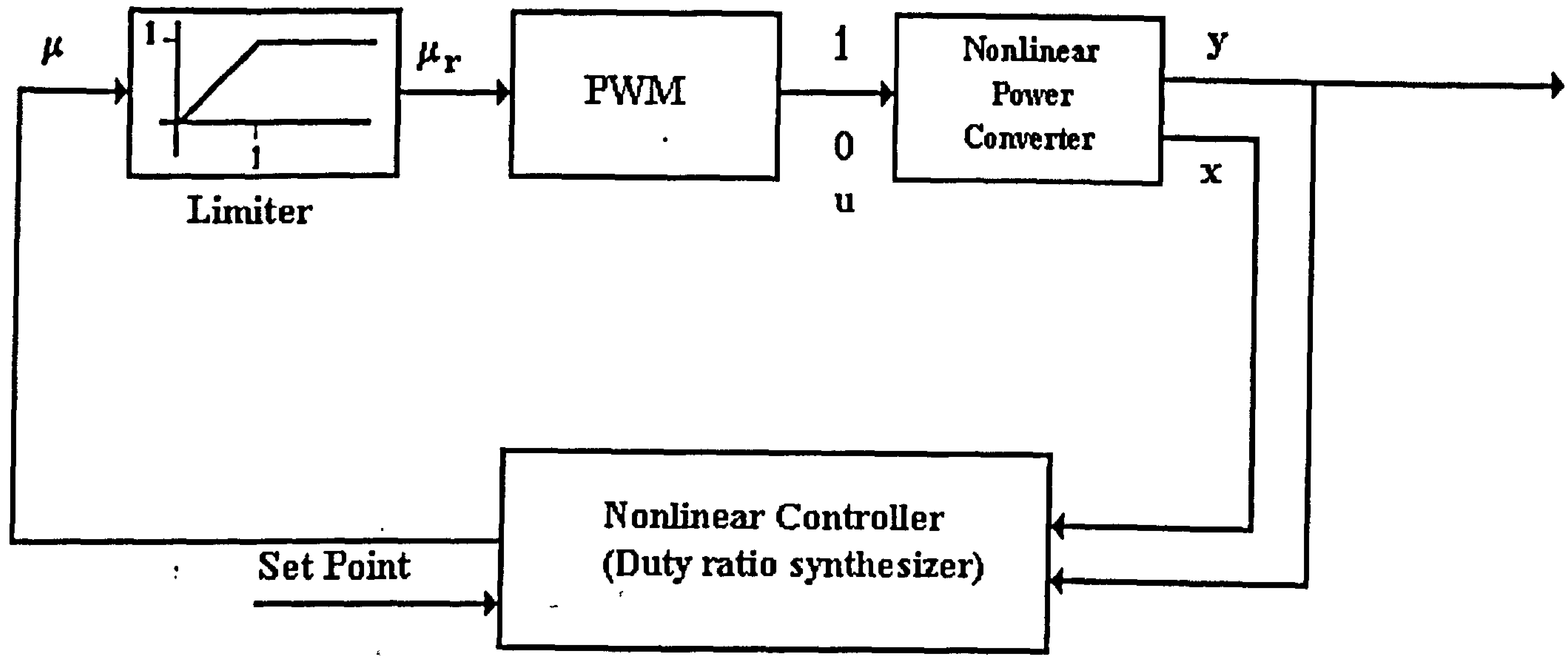


Figure 5.2: Feedback regulation scheme for nonlinear PWM switch-controlled systems.

## 5.3 Adaptive PWM Control of Power Converters

In order to apply the DAB algorithm, an output function should be specified. Traditionally two possibilities exist for the regulated output function: either the input inductor current $I(t)$ or the output capacitor voltage $V(t)$. When the latter is taken as the regulated output, both the Boost and the Buck-Boost converters are non-minimum phase [107]. In such a non-minimum phase case the DAB algorithm of Section 4.2 leads to an *unstable* adaptive controller. Hence, regulation of power converters via the input inductor current is preferred. Thus, indirect feedback regulation of the output capacitor voltage is accomplished.

Dynamical compensation has been previously proposed for the regulation problem of DC-to-DC power converters when full-knowledge of the circuit parameters is assumed [107] and under uncertainty conditions [109]. Since the relative degree of the average models considered here is 1, the adaptive controller arising from the DAB algorithm of Section 4.2 is dynamical in nature. Moreover, it is worth emphasizing that the average PWM models of both the Boost and Buck-Boost converters examined in this section are *not* transformable into the PPF and PSF forms. When applying the DAB algorithm, we consider the control input $u$ as a state variable of the dynamical compensator. So the

PWM models of these converters can be regarded as nontriangular forms of the extended system (power converter plus dynamical compensator).

### 5.3.1 Adaptive PWM Control of the Boost Converter

Consider the average model of the Boost converter circuit with the input inductor current as the regulated output

$$\begin{aligned} \dot{x}_1 &= -\frac{1}{L}(1-\mu)x_2 + \frac{E}{L} \\ \dot{x}_2 &= \frac{1}{C}(1-\mu)x_1 - \frac{1}{RC}x_2 \\ y &= x_1 \end{aligned} \tag{5.10}$$

where $x_1$ is the average input inductor current $I(t)$, $x_2$ the average output capacitor voltage $V(t)$ and $\mu$ the duty ratio function acting as the control input. Denote the values of the parameters defining the circuit equations (5.10) as

$$\theta_1 = \frac{1}{L}, \qquad \theta_2 = \frac{1}{C}, \qquad \theta_3 = \frac{1}{RC}, \qquad \theta_4 = \frac{E}{L}. \tag{5.11}$$

For a constant value of the duty ratio function $\mu = U$, $0 < U < 1$, the equilibrium values of the average PWM model are readily obtained from (5.10) and (5.11) as

$$X_1(U) = \frac{\theta_3\theta_4}{\theta_1\theta_2(1-U)^2}, \qquad X_2(U) = \frac{\theta_4}{\theta_1(1-U)} \tag{5.12}$$

Assuming that the actual values of all the circuit parameters are unknown, the primary objective is the adaptive feedback regulation of the average input inductor current $x_1(t)$ towards a known and constant equilibrium value $X_1 = x_1(U)$. This value corresponds to some constant value, $U$, of the (actual) duty ratio function.

The average model (5.10) can be rewritten as

$$\begin{aligned} \dot{x}_1 &= \varphi_1^T(x_2,\mu)\theta \\ \dot{x}_2 &= \varphi_2^T(x,\mu)\theta \\ y &= x_1 \end{aligned} \tag{5.13}$$

where $\theta = [\theta_1, \theta_2, \theta_3, \theta_4]^T$ and

$$\varphi_1(x_2,\mu) = \begin{bmatrix} -(1-\mu)x_2 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \quad \varphi_2(x,\mu) = \begin{bmatrix} 0 \\ (1-\mu)x_1 \\ -x_2 \\ 0 \end{bmatrix}. \tag{5.14}$$

Since the function $\varphi_1$ depends on the control input $\mu$, the average model (5.13) is not in PPF form. We now design a dynamical adaptive controller for the average Boost converter model (5.13). Once the adaptive controller expressions are found, the average state variables $x_1$, $x_2$ appearing in the feedback controller and the parameter adaptation laws, are replaced by the actual (non-averaged) variable $I(t)$, $V(t)$ respectively. This procedure has been shown to be valid in many linear and nonlinear regulation schemes proposed for DC-to-DC power supplies, including those which are based on adaptive control ([53, 104, 107, 109, 113]).

**Step 1.** Define the error variable $z_1$

$$z_1 := x_1 - X_1 \tag{5.15}$$

whose time derivative is given by

$$\dot{z}_1 = \omega_1(x_2, \mu)\hat{\theta} + \omega_1(x_2, \mu)(\theta - \hat{\theta}) \tag{5.16}$$

with the regressor vector $\omega_1$ defined by

$$\omega_1(x_2, \mu) = \varphi_1^T(x_2, \mu) \tag{5.17}$$

Consider the quadratic Lyapunov function

$$V_1 = \frac{1}{2}z_1^2 + \frac{1}{2}(\theta - \hat{\theta})^T\Gamma^{-1}(\theta - \hat{\theta}) \tag{5.18}$$

where $\Gamma = \Gamma^T > 0$ is a matrix of adaptation gains. The time derivative of $V_1$ is obtained as

$$\dot{V}_1 = z_1\Big[\omega_1(x_2, \mu)\hat{\theta}\Big] + (\theta - \hat{\theta})^T\Gamma^{-1}\Big[ - \dot{\hat{\theta}} + \Gamma\omega_1^T(x_2, \mu)z_1\Big] \tag{5.19}$$

One can achieve $\dot{V}_1 = -c_1z_1^2$ with $c_1$ a positive design parameter, by choosing the update law

$$\dot{\hat{\theta}} = \tau_1 = \Gamma\omega_1^T(x_2, \mu)z_1 \tag{5.20}$$

if the relation

$$\omega_1(x_2, \mu)\hat{\theta} = -c_1z_1 \tag{5.21}$$

is satisfied. However, since (5.21) is not valid and $\tau_1$ is not considered an update law but the first tuning function, the second error variable is defined as

$$z_2 := \omega_1(x_2, \mu)\hat{\theta} + c_1z_1 \tag{5.22}$$

The closed-loop form for $\dot{z}_1$ is

$$\dot{z}_1 = -c_1z_1 + z_2 + \omega_1(x_2, \mu)(\theta - \hat{\theta}) \tag{5.23}$$

and

$$\dot{V}_1 = -c_1 z_1^2 + z_1 z_2 + (\theta - \hat{\theta})^T \Gamma^{-1} \Big( -\dot{\hat{\theta}} + \tau_1 \Big) \tag{5.24}$$

**Step 2.** The time derivative of $z_2$ is obtained as

$$\dot{z}_2 = \omega_2 \hat{\theta} + \omega_2(\theta - \hat{\theta}) + \hat{\theta}_1 x_2 \dot{\mu} + \omega_1 \dot{\hat{\theta}} \tag{5.25}$$

with the regressor vector

$$\omega_2(x, \hat{\theta}, \mu) = \frac{\partial \varphi_1^T}{\partial x_2} \hat{\theta} \varphi_2^T + c_1 \varphi_1^T = \begin{bmatrix} -c_1(1-\mu)x_2 \\ \hat{\theta}_1(1-\mu)^2 x_1 \\ \hat{\theta}_1(1-\mu)x_2 \\ c_1 \end{bmatrix}^T \tag{5.26}$$

Define the augmented Lyapunov function

$$V_2 = V_1 + \frac{1}{2} z_2^2 \tag{5.27}$$

whose time derivative is given by

$$\dot{V}_2 = -c_1 z_1^2 + z_2 \Big[ z_1 + \omega_2 \hat{\theta} + \hat{\theta}_1 x_2 \dot{\mu} + \omega_1 \dot{\hat{\theta}} \Big] + (\theta - \hat{\theta}) \Gamma^{-1} \Big( -\dot{\hat{\theta}} + \tau_1 + \Gamma \omega_2^T z_2 \Big) \tag{5.28}$$

The control design is completed by choosing the update law

$$\dot{\hat{\theta}} = \tau_2 = \tau_1 + \Gamma \omega_2^T z_2 = \Gamma \Big[ \omega_1^T \quad \omega_2^T \Big] z \tag{5.29}$$

and the dynamical controller

$$\dot{\mu} = \frac{1}{\hat{\theta}_1 x_2} \Big[ - z_1 - \omega_2 \hat{\theta} - \omega_1 \tau_2 - c_2 z_2 \Big] \tag{5.30}$$

to achieve

$$\dot{V}_2 = -c_1 z_1^2 - c_2 z_2^2 \tag{5.31}$$

The duty ratio function $\mu$ can be obtained as the solution of the nonlinear differential equation defined by (5.30). Note that the set of error variables

$$z = \Phi(x, \hat{\theta}, \mu) := \begin{pmatrix} x_1 - X_1 \\ -\hat{\theta}_1(1-\mu)x_2 + \hat{\theta}_4 + c_1(x_1 - X_1) \end{pmatrix} \tag{5.32}$$

qualifies as a coordinate transformation of the average state variables $x_1$ and $x_2$. The Jacobian matrix of this transformation is obtained as

$$\frac{\partial \Phi(x, \hat{\theta}, \mu)}{\partial x} = \begin{bmatrix} 1 & 0 \\ c_1 & -\hat{\theta}_1(1-\mu) \end{bmatrix} \tag{5.33}$$

which is non-singular everywhere except at persistently saturated values of the duty ratio function $\mu = 1$. This condition represents an unstable open-loop situation in the adopted PWM setup. For this reason, asymptotically stable behaviour can be guaranteed locally for the original state variables as long as asymptotically stable behaviour is achieved for the transformed variables $z_1$ and $z_2$. The local non-singularity of the Jacobian matrix is equivalent to the local *observability* of the average system (5.13). Furthermore, since the rank of the matrix

$$\begin{bmatrix} \varphi_1^T \\ \varphi_2^T \end{bmatrix}_{(X,U)} = \begin{bmatrix} -(1-\mu)x_2 & 0 & 0 & 1 \\ 0 & (1-\mu)x_1 & -x_2 & 0 \end{bmatrix}_{(X,U)} \tag{5.34}$$

is always less than the number of unknown parameters ($p = 4$), convergence of the parameter estimate $\hat{\theta}$ to the true unknown value of the parameters is not accomplished. Nevertheless, bounded values of $\hat{\theta}$ are guaranteed.

In summary the adaptive controller for the average system is given by (5.30) together with the update law for the parameters (5.29). The duty ratio synthesizer for the PWM regulated system is obtained by replacing the average state variables $x_1$, $x_2$ appearing in the controller expressions by the actual state variables $I(t)$ and $V(t)$, respectively.

## 5.3.2 Adaptive PWM Control of the Buck-Boost Converter

Consider the average bilinear model of the Buck-Boost converter circuit with the input inductor current as the regulated output

$$\begin{aligned} \dot{x}_1 &= \frac{1}{L}(1-\mu)x_2 + \frac{E}{L}\mu \\ \dot{x}_2 &= -\frac{1}{C}(1-\mu)x_1 - \frac{1}{RC}x_2 \\ y &= x_1 \end{aligned} \tag{5.35}$$

where $x_1$ is the average input inductor current $I(t)$, $x_2$ the average output capacitor voltage $V(t)$ and $\mu$ the duty ratio function acting as the control input. Denoting the values of the circuit parameters as

$$\theta_1 = \frac{1}{L}, \qquad \theta_2 = \frac{1}{C}, \qquad \theta_3 = \frac{1}{RC}, \qquad \theta_4 = \frac{E}{L}, \tag{5.36}$$

the equilibrium values of the average PWM model for a constant value of the duty ratio function $\mu = U$, $0 < U < 1$, are obtained from (5.35) and (5.36) as

$$X_1(U) = \frac{\theta_3\theta_4}{\theta_1\theta_2(1-U)^2}, \qquad X_2(U) = -\frac{\theta_4}{\theta_1(1-U)}. \tag{5.37}$$

Assuming that the actual values of all the circuit parameters are unknown, the control objective is the adaptive feedback regulation of the average input inductor current $x_1(t)$ towards a known and constant equilibrium value $X_1 = x_1(U)$ corresponding to some constant value $U$ of the duty ratio function. Under these conditions the average model (5.35) can be rewritten as

$$\begin{aligned} \dot{x}_1 &= \varphi_1^T(x_2,\mu)\theta \\ \dot{x}_2 &= \varphi_2^T(x,\mu)\theta \\ y &= x_1 \end{aligned} \tag{5.38}$$

where $\theta = [\theta_1, \theta_2, \theta_3, \theta_4]^T$ and

$$\varphi_1(x_2,\mu) = \begin{bmatrix} (1-\mu)x_2 \\ 0 \\ 0 \\ \mu \end{bmatrix}, \quad \varphi_2(x,\mu) = \begin{bmatrix} 0 \\ (1-\mu)x_1 \\ -x_2 \\ 0 \end{bmatrix}. \tag{5.39}$$

Since the function $\varphi_1$ depends on the control input $\mu$, the average model (5.38) is not in PPF form.

Proceeding in the same manner as for the Boost converter, the dynamical backstepping algorithm yields the error coordinate transformation

$$z = \Phi(x,\hat{\theta},\mu) := \begin{pmatrix} x_1 - X_1 \\ \hat{\theta}_1(1-\mu)x_2 + \hat{\theta}_4\mu + c_1(x_1 - X_1) \end{pmatrix} \tag{5.40}$$

and the tuning functions

$$\begin{aligned} \tau_1 &= \Gamma\omega_1^T z_1 \qquad (5.41)\\ \tau_2 &= \tau_1 + \Gamma\omega_2^T z_2 = \Gamma\begin{bmatrix}\omega_1^T & \omega_2^T\end{bmatrix} z \end{aligned}$$

with the regressor vectors

$$\begin{aligned} \omega_1(x_2,\mu) &= \varphi_1^T(x_2,\mu) \qquad (5.42)\\ \omega_2(x,\hat{\theta},\mu) &= \frac{\partial\varphi_1^T}{\partial x_2}\hat{\theta}\varphi_2^T + c_1\varphi_1^T = \begin{bmatrix} c_1(1-\mu)x_2 \\ \hat{\theta}_1(1-\mu)^2 x_1 \\ \hat{\theta}_1(1-\mu)x_2 \\ c_1\mu \end{bmatrix}^T. \end{aligned}$$

The adaptive controller is defined by the update law $\dot{\hat{\theta}} = \tau_2$ and the dynamical feedback law

$$\dot{\mu} = \frac{1}{\hat{\theta}_4 - \hat{\theta}_1 x_2}\Big[-z_1 - \omega_2\hat{\theta} - \omega_1\tau_2 - c_2 z_2\Big] \tag{5.43}$$

The duty ratio function $\mu$ can be obtained as the solution of the nonlinear differential equation defined by (5.43).

In order to check the observability condition, the Jacobian matrix of the transformation (5.40)

$$\frac{\partial\Phi(x,\hat{\theta},\mu)}{\partial x} = \begin{bmatrix} 1 & 0 \\ c_1 & -\hat{\theta}_1(1-\mu) \end{bmatrix} \tag{5.44}$$

which, as in the case of the Boost converter, is non-singular everywhere except at persistently saturated values of the duty ratio function $\mu = 1$. Hence, asymptotically stable behaviour of the state variables is guaranteed locally. Furthermore, for the same reasons as for the Boost converter, bounded values of $\hat{\theta}$ are guaranteed but convergence to the actual unknown parameters is not accomplished.

### 5.3.3 Simulation Results

Simulations were carried out to assess the adaptively controlled behaviour of both converters. In order to test the robustness of the proposed regulation scheme with respect to external perturbation inputs, perturbed models of the actual PWM controlled converter were used in the simulations. The perturbed models included an external stochastic perturbation input entering additively to the external source voltage represented by $E$. Thus, for the Boost converter the adaptive dynamical controller was applied to the perturbed model

$$\begin{aligned} \dot{I}(t) &= -\frac{1}{L}(1-u)V(t) + \frac{E+\nu}{L} \\ \dot{V}(t) &= \frac{1}{C}(1-u)I(t) - \frac{1}{RC}V(t) \end{aligned} \tag{5.45}$$

i.e. the external voltage source $E$ was assumed to include an unmatched additive noisy input $\nu$ affecting the behaviour of the converter during the "on" and "off" stages of the switching.

In the Buck-Boost converter simulations the perturbed model was taken to be

$$\begin{aligned} \dot{I}(t) &= \frac{1}{L}(1-u)V(t) + \frac{E+\nu}{L}u \\ \dot{V}(t) &= -\frac{1}{C}(1-u)I(t) - \frac{1}{RC}V(t) \end{aligned} \tag{5.46}$$

The following supposedly "unknown" values of the circuit parameters were used for simulation purposes

$$C = 181.82\ \mu\text{F} \ ; \ L = 0.27\ \text{mH} \ ; \ R = 2.44\ \Omega \ ; \ E = 14.667\ \text{Volts}$$

for both converters. These values of the circuit components yield the following actual values of the model parameters

$$\theta_1 = 3600 \ ; \ \theta_2 = 5500 \ ; \ \theta_3 = 2250 \ ; \ \theta_4 = 52800$$

The sampling frequency was set to be 1 KHz and the random noise amplitude was set to be 2.44 Volts (16 % of the value of E).

Figure 5.3 depicts the dynamic adaptively regulated state responses of the Boost converter. The figure also shows the evolution of the estimated parameter values as obtained from the designed update law, the duty ratio function and the switching actions on a shorter time interval. The desired equilibrium value for the average input inductor current was set to be $I = 15.75$ amps. The resulting steady state equilibrium value for the average output capacitor voltage is $V = 23.77$ Volts. The duty ratio value corresponding to this equilibrium is $\mu = U = 0.38$. The regulated state variables are seen to converge asymptotically towards the desired equilibrium values.

Figure 5.4 shows the dynamic adaptively regulated state responses of the Buck–Boost converter, and also the evolution of the estimated parameters arising from the update law, the duty ratio function and the switching actions on a shorter time interval. The chosen equilibrium value for the average input inductor current was $I = 22.5$ amps. The resulting value for the average capacitor voltage is $V = -22$ Volts, with duty ratio $\mu = U = 0.6$. The regulated state variables converge asymptotically towards the desired equilibrium values.

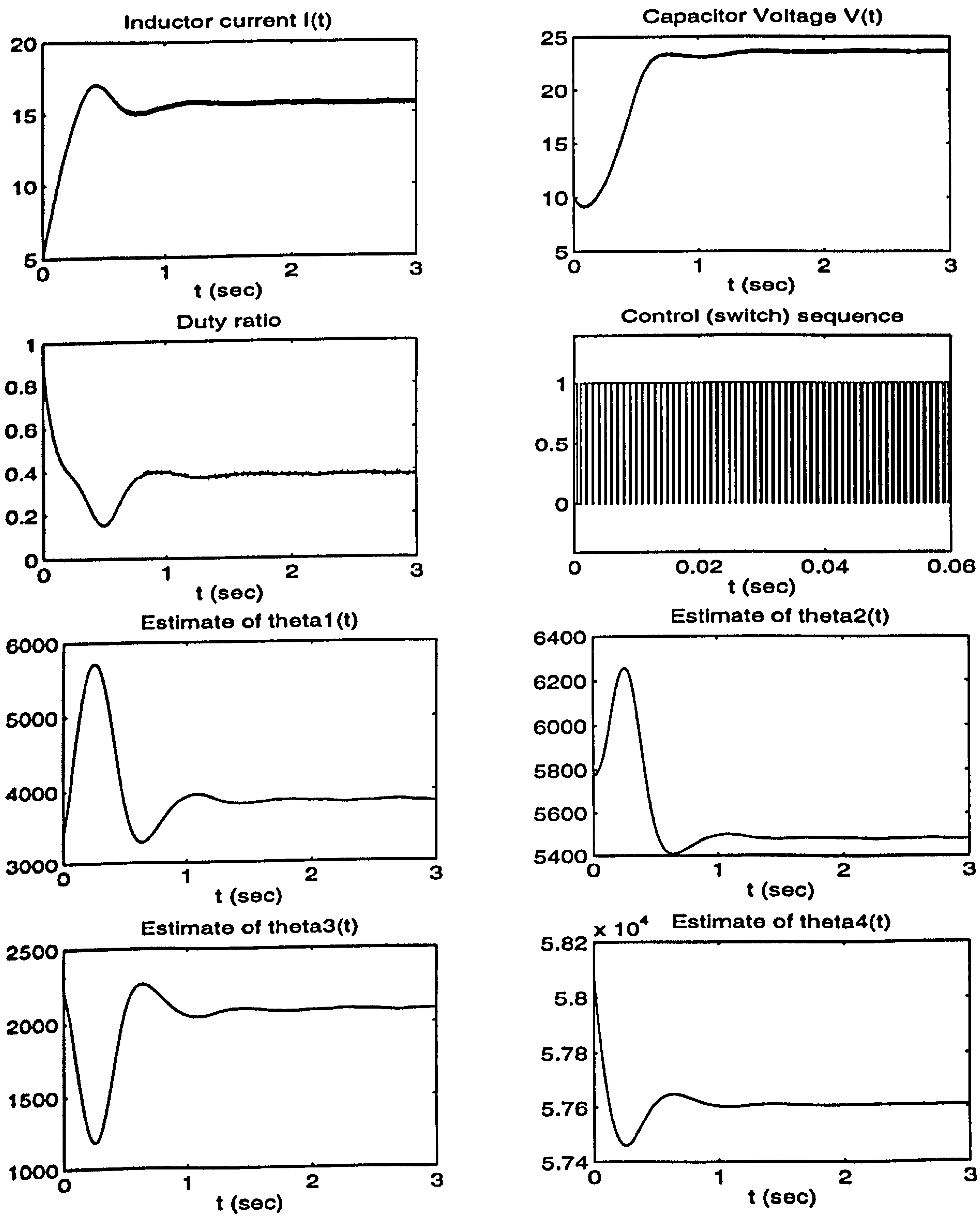


Figure 5.3: Controlled state trajectories of the perturbed Boost converter, duty ratio function, control (switch) sequence and evolution of the parameter estimate.

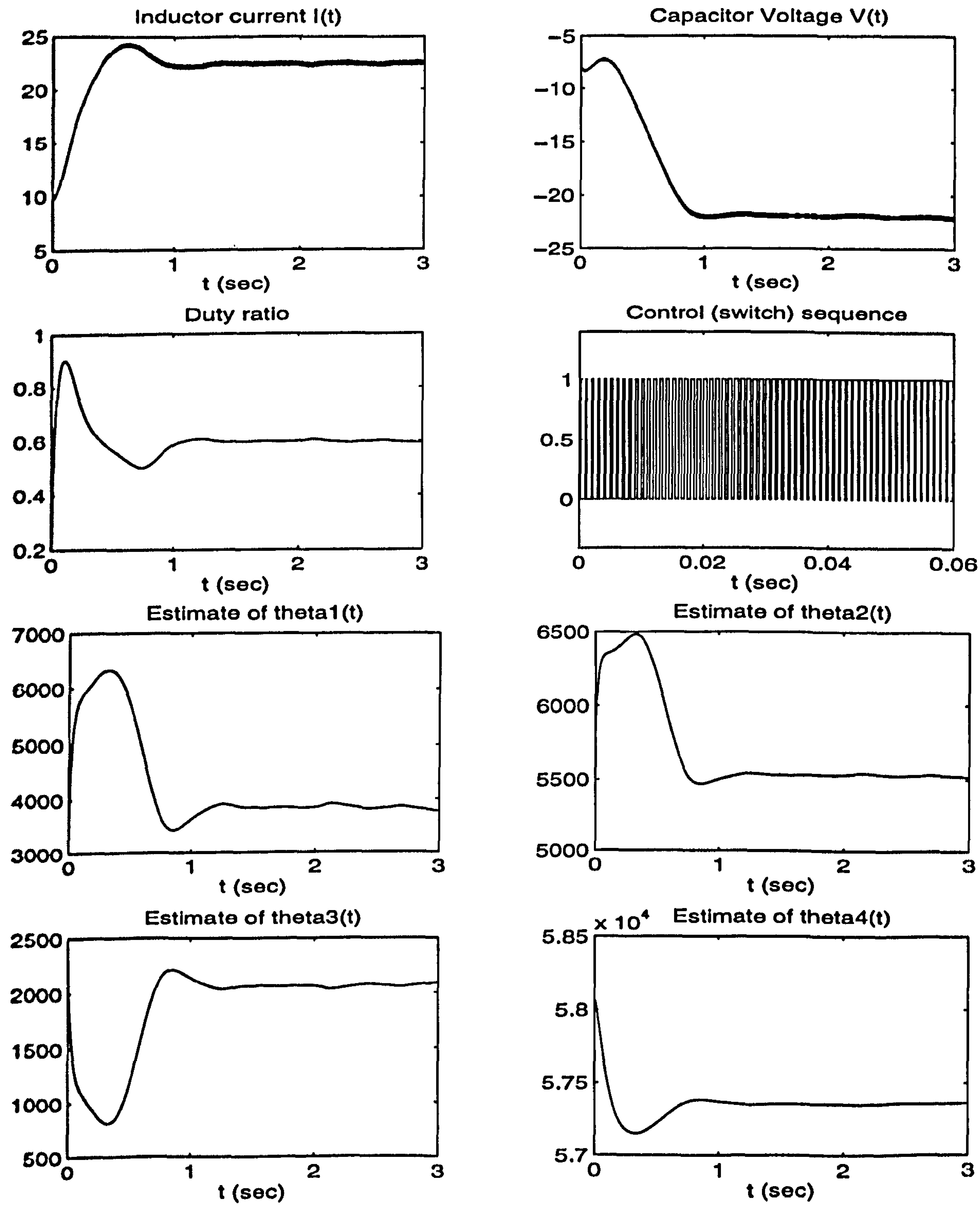


Figure 5.4: Controlled state trajectories of the perturbed Buck-Boost converter, duty ratio function, control (switch) sequence and evolution of the parameter estimate.

# Chapter 6

# SAB and Sliding Mode Control of PSF Systems with Disturbances

## 6.1 Introduction

As demonstrated in Section 1.4, an adaptive system may possibly exhibit instability in the presence of disturbances. This drawback motivates the need for robustness in adaptive control schemes. On the other hand, Sliding Mode Control (SMC) is a well-known deterministic technique for the design of robust controllers for uncertain systems with disturbances ([122],[126]).

In this chapter a combination of adaptive control and SMC is proposed to cope with tracking problems of uncertain systems in the presence of undesirable disturbances. In particular we propose the SAB algorithm described in Section 2.4.3 and SMC to design static adaptive sliding mode tracking controllers for systems transformable into the PSF form. These new combined SAB-SMC algorithm allows one to design robust controllers to achieve global asymptotic stability of uncertain systems in PSF form, and motivates the design of tracking controllers for broader classes of uncertain systems which will be considered in Chapter 7.

We first describe the systematic SAB-SMC algorithm and give sufficient conditions on the design parameters to guarantee asymptotic stability of the closed-loop system. Then a third order uncertain nonlinear system is used to illustrate the applicability of the design algorithm, and computer simulations illustrate the performance of the adaptive sliding mode controller in tracking tasks.

## 6.2 SAB-SMC Algorithm

The combination of adaptive control and SMC has been used in regulation and tracking problems involving uncertain nonlinear systems. For instance, Slotine and Li solved in [114] the tracking problem of robot manipulators with matched uncertainties. A direct adaptive SMC scheme was proposed by Su [116] for nonlinear robotic systems with bounded time-varying parameters. More recently, a combination of the adaptive backstepping algorithm with tuning functions (the SAB algorithm) proposed by Krstić *et al* [61] and SMC has been explored by various researchers for the robust adaptive control of nonlinear systems with unmatched disturbances. Sira-Ramírez and Llanes-Santiago [108] combined these two techniques for the control of two-dimensional systems in PSF form. The approach proposed by Rios-Bolívar and Zinober [88] and Rios-Bolívar *et al* [91] deals with the general tracking problem for PSF systems, which will be described below. A different combined approach has been proposed by Karsenti and Lamnabhi-Lagarrigue [52] for a class of nonlinear parameterized uncertain systems with a canonical form called *non-pure parametric feedback form.* This form preserves the triangular form of parametric pure-feedback systems, but includes nonlinear relations for the unknown parameters. Another combined backstepping-SMC strategy has been used by Chien and Fu [18] in the context of output-feedback control.

Consider the class of uncertain nonlinear systems transformable into the PSF form

$$
\begin{aligned}
\dot{x}_1 &= x_2 + \varphi_1^T(x_1)\theta \\
\dot{x}_2 &= x_3 + \varphi_2^T(x_1, x_2)\theta \\
&\vdots \\
\dot{x}_{n-1} &= x_n + \varphi_{n-1}^T(x_1, \ldots, x_{n-1})\theta \\
\dot{x}_n &= \varphi_0(x) + \varphi_n^T(x)\theta + \beta_0(x)u \\
y &= x_1
\end{aligned}
\tag{6.1}
$$

where $\beta_0$ and $\varphi_i$, $i = 0, \ldots, n$, are smooth nonlinear functions of their arguments, with $\beta_0(0) \neq 0 \; \forall x \in \Re^n$.

The problem of robustly adaptively tracking a known bounded reference signal $y_r(t)$ with continuous and bounded derivatives $y_r^{(i)}(t)$, $i = 1, \ldots, n$ will be solved in this section via the following combined SAB-SMC algorithm.

***SAB-SMC Algorithm***

*Coordinate transformation*

$$\begin{aligned} z_1 &:= y - y_r(t) = x_1 - y_r(t) \\ z_k &= x_k - \alpha_{k-1}(x_1, \ldots, x_{k-1}, \hat{\theta}, t) - y_r^{(k)}(t) \qquad 2 \leqslant k \leqslant n \end{aligned} \tag{6.2}$$

*with*

$$\begin{aligned} \alpha_k(x_1, \ldots, x_k, \hat{\theta}, t) &:= -z_{k-1} + \sum_{i=1}^{k-1} \frac{\partial \alpha_{k-1}}{\partial x_i} x_{i+1} + \frac{\partial \alpha_{k-1}}{\partial \hat{\theta}} \tau_k + \frac{\partial \alpha_{k-1}}{\partial t} - \omega_k^T \hat{\theta} \\ &\quad + \left( \sum_{i=1}^{k-2} z_{i+1} \frac{\partial \alpha_i}{\partial \hat{\theta}} \right) \Gamma \omega_k - c_k z_k \end{aligned} \tag{6.3}$$

$$\omega_k(x_1, \ldots, x_k, \hat{\theta}, t) := \varphi_k - \sum_{i=1}^{k-1} \frac{\partial \alpha_{k-1}}{\partial x_i} \varphi_i \tag{6.4}$$

$$\tau_k(x_1, \ldots, x_k, \hat{\theta}, t) := \tau_{k-1} + \Gamma \omega_k z_k = \Gamma \sum_{i=1}^{k} \omega_i z_i \tag{6.5}$$

*Define the sliding surface*

$$\sigma = k_1 z_1 + k_2 z_2 + \ldots + k_{n-1} z_{n-1} + z_n = 0 \tag{6.6}$$

*and choose the design parameters $k_i$, $i = 1, \ldots, n-1$, such that the polynomial*

$$p(s) = k_1 + k_2 s + \ldots + k_{n-1} s^{n-2} + s^{n-1} \tag{6.7}$$

*in the complex variable $s$ is Hurwitz.*

*Parameter update law*

$$\begin{aligned} \dot{\hat{\theta}} := \tau_n &= \tau_{n-1} + \Gamma \sigma \left( \omega_n + \sum_{i=1}^{n-1} k_i \omega_i \right) \\ &= \Gamma \left[ \sum_{i=1}^{n-1} \omega_i z_i + \sigma \left( \omega_n + \sum_{i=1}^{n-1} k_i \omega_i \right) \right] \end{aligned} \tag{6.8}$$

*Adaptive sliding mode control law*

$$\begin{aligned} u = \frac{1}{\beta_0(x)} \Bigg[ &- \varphi_0(x) + \sum_{i=1}^{n-1} \frac{\partial \alpha_{n-1}}{\partial x_i} x_{i+1} - \omega_n^T \hat{\theta} + \frac{\partial \alpha_{n-1}}{\partial \hat{\theta}} \tau_n + \frac{\partial \alpha_{n-1}}{\partial t} + y_r^{(n)}(t) \\ &- \sum_{i=1}^{n-1} k_i \left( -z_{i-1} - c_i z_i + z_{i+1} - \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}} \left( \tau_n - \tau_i \right) + \left( \sum_{j=1}^{i-2} z_{j+1} \frac{\partial \alpha_j}{\partial \hat{\theta}} \right) \Gamma \omega_i \right) \\ &+ \left( \sum_{i=1}^{n-2} z_{i+1} \frac{\partial \alpha_i}{\partial \hat{\theta}} \right) \Gamma \left( \omega_n + \sum_{i=1}^{n-1} k_i \omega_i \right) - \lambda \left( \sigma + \beta \operatorname{sgn}(\sigma) \right) \Bigg] \end{aligned} \tag{6.9}$$

We now describe the steps leading to the general SAB-SMC algorithm summarized above. The first $n-1$ steps of the computational procedure leading to the adaptive controller are exactly the same of those of the SAB algorithm with tuning functions [61] for tracking tasks. At the final step an adaptive sliding surface is defined in terms of the tracking error variables, and a discontinuous control law is also synthesized to assure convergence of the state trajectories to the sliding surface. The resulting adaptive SMC guarantees global robust tracking for PSF systems even in the presence of bounded disturbances.

In order to distinguish the differences of the backstepping algorithm in tracking tasks considered here and the SAB algorithm given in Section 2.4.3 for regulation problems, we describe the first step in detail.

**Step 1.** Define the tracking error variable $z_1$ as

$$z_1 := y - y_r(t) = x_1 - y_r(t) \tag{6.10}$$

whose time derivative is given by

$$\dot{z}_1 = x_2 + \varphi_1^T(x_1)\theta - \dot{y}_r(t). \tag{6.11}$$

Adding (and subtracting) the estimate $\hat{\theta}$ of $\theta$ to (from) (6.11), $\dot{z}_1$ can be rewritten as

$$\dot{z}_1 = x_2 + \omega_1^T\hat{\theta} - \dot{y}_r(t) + \omega_1^T\tilde{\theta} \tag{6.12}$$

where $\tilde{\theta} = \theta - \hat{\theta}$ and $\omega_1$ is the regressor vector defined by

$$\omega_1(x_1) = \varphi_1(x_1). \tag{6.13}$$

Consider the Lyapunov function

$$V_1 = \frac{1}{2}z_1^2 + \frac{1}{2}\tilde{\theta}^T\Gamma^{-1}\tilde{\theta} \tag{6.14}$$

where $\Gamma = \Gamma^T > 0$ is a matrix of adaptation gains. The time derivative of $V_1$ yields

$$\dot{V}_1 = z_1\Big[x_2 + \omega_1^T\hat{\theta} - \dot{y}_r(t)\Big] + \tilde{\theta}^T\Gamma^{-1}\Big( -\dot{\hat{\theta}} + \Gamma\omega_1 z_1\Big) \tag{6.15}$$

One can achieve $\dot{V}_1 = -c_1 z_1^2$ with $c_1 > 0$ a design parameter, by choosing the update law

$$\dot{\hat{\theta}} = \tau_1 = \Gamma\omega_1 z_1 \tag{6.16}$$

if the relation

$$x_2 + \omega_1^T\hat{\theta} - \dot{y}_r(t) = -c_1 z_1 \tag{6.17}$$

is satisfied. However, since (6.17) is not valid in general and $\tau_1$ is not considered an update law but the first tuning function, the second error variable is defined as

$$z_2 = x_2 - \alpha_1(x_1, \hat{\theta}, t) - \dot{y}_r(t) \tag{6.18}$$

where

$$\alpha_1(x_1, \hat{\theta}, t) = -\omega_1^T \hat{\theta} - c_1 z_1. \tag{6.19}$$

The closed-loop form of $\dot{z}_1$ becomes

$$\dot{z}_1 = -c_1 z_1 + z_2 + \omega_1^T \tilde{\theta} \tag{6.20}$$

and

$$\dot{V}_1 = -c_1 z_1^2 + z_1 z_2 + \tilde{\theta}^T \Gamma^{-1}\Big( -\dot{\hat{\theta}} + \tau_1 \Big). \tag{6.21}$$

Note that in this case $z_1$, $\alpha_1$ and $z_2$ are time-dependent variables. In fact, all the $z_i$'s and $\alpha_i$'s will be time-dependent variables, as shown below. The following generic step summarizes the first $n-1$ steps of the backstepping algorithm for tracking tasks.

**Step k** ($2 \leqslant k \leqslant n-1$). The time derivative of the error variable $z_k$ is

$$\dot{z}_k = x_{k+1} - \sum_{i=1}^{k-1} \frac{\partial \alpha_{k-1}}{\partial x_i} x_{i+1} + \omega_k^T \theta - \frac{\partial \alpha_{k-1}}{\partial \hat{\theta}} \dot{\hat{\theta}} - \frac{\partial \alpha_{k-1}}{\partial t} - y_r^{(k)}(t) \tag{6.22}$$

with the regressor vector

$$\omega_k := \varphi_k - \sum_{i=1}^{k-1} \frac{\partial \alpha_{k-1}}{\partial x_i} \varphi_i. \tag{6.23}$$

By adding and subtracting $\hat{\theta}$ in (6.22), $\dot{z}_k$ can be rewritten as

$$\dot{z}_k = x_{k+1} - \sum_{i=1}^{k-1} \frac{\partial \alpha_{k-1}}{\partial x_i} x_{i+1} + \omega_k^T \hat{\theta} - \frac{\partial \alpha_{k-1}}{\partial \hat{\theta}} \dot{\hat{\theta}} - \frac{\partial \alpha_{k-1}}{\partial t} - y_r^{(k)}(t) + \omega_k^T \tilde{\theta} \tag{6.24}$$

which can be stabilized with respect to the augmented Lyapunov function

$$V_k = V_{k-1} + \frac{1}{2} z_k^2. \tag{6.25}$$

The time derivative of $V_k$ is

$$\begin{aligned} \dot{V}_k = & -\sum_{i=1}^{k-1} c_i z_i^2 - \left( \sum_{i=1}^{k-2} z_{i+1} \frac{\partial \alpha_i}{\partial \hat{\theta}} \right) \Big( \dot{\hat{\theta}} - \tau_{k-1} \Big) \\ & + z_k \left[ z_{k-1} + x_{k+1} - \sum_{i=1}^{k-1} \frac{\partial \alpha_{k-1}}{\partial x_i} x_{i+1} + \omega_k^T \hat{\theta} - \frac{\partial \alpha_{k-1}}{\partial \hat{\theta}} \dot{\hat{\theta}} - \frac{\partial \alpha_{k-1}}{\partial t} - y_r^{(k)}(t) \right] \\ & + \tilde{\theta}^T \Gamma^{-1} \Big( -\dot{\hat{\theta}} + \tau_{k-1} + \Gamma \omega_k z_k \Big) \end{aligned} \tag{6.26}$$

One can eliminate $\tilde{\theta}$ from $\dot{V}_k$ with the update law $\dot{\hat{\theta}} = \tau_k$ defined as

$$\tau_k := \tau_{k-1} + \Gamma\omega_k z_k = \Gamma\sum_{i=1}^{k}\omega_i z_i. \tag{6.27}$$

Furthermore, noting that

$$\dot{\hat{\theta}} - \tau_{k-1} = \dot{\hat{\theta}} - \tau_k + \tau_k - \tau_{k-1} = \dot{\hat{\theta}} - \tau_k + \Gamma\omega_k z_k \tag{6.28}$$

$\dot{V}_k$ can be rewritten as

$$\begin{aligned}\dot{V}_k = &-\sum_{i=1}^{k-1} c_i z_i^2 - \left(\sum_{i=1}^{k-2} z_{i+1}\frac{\partial\alpha_i}{\partial\hat{\theta}}\right)\left(\dot{\hat{\theta}} - \tau_k\right) + \tilde{\theta}^T\Gamma^{-1}\left(-\dot{\hat{\theta}} + \tau_k\right) \\ &+ z_k\left[z_{k-1} + x_{k+1} - \sum_{i=1}^{k-1}\frac{\partial\alpha_{k-1}}{\partial x_i}x_{i+1} - \frac{\partial\alpha_{k-1}}{\partial\hat{\theta}}\dot{\hat{\theta}} - \frac{\partial\alpha_{k-1}}{\partial t} - y_r^{(k)}(t)\right. \\ &\left. + \omega_k^T\hat{\theta} - \left(\sum_{i=1}^{k-2} z_{i+1}\frac{\partial\alpha_i}{\partial\hat{\theta}}\right)\Gamma\omega_k\right]\end{aligned} \tag{6.29}$$

Then if $x_{k+1}$ is considered a virtual control, one can make $\dot{V}_k = -\sum_{i=1}^{k} c_i z_i^2$ with the virtual control $x_{k+1} = \alpha_k + y_r^{(k)}(t)$ defined as

$$\begin{aligned}\alpha_k(x_1,\ldots,x_k,\hat{\theta},t) \;:=\; &-z_{k-1} + \sum_{i=1}^{k-1}\frac{\partial\alpha_{k-1}}{\partial x_i}x_{i+1} + \frac{\partial\alpha_{k-1}}{\partial\hat{\theta}}\tau_k + \frac{\partial\alpha_{k-1}}{\partial t} - \omega_k^T\hat{\theta} \\ &+\left(\sum_{i=1}^{k-2} z_{i+1}\frac{\partial\alpha_i}{\partial\hat{\theta}}\right)\Gamma\omega_k - c_k z_k.\end{aligned} \tag{6.30}$$

which would make the bracketed term multiplying $z_k$ equal to $-c_k z_k$. However, since $x_{k+1}$ is not the actual control, the $k$-th error variable is defined as

$$z_{k+1} = x_{k+1} - \alpha_k(x_1,\ldots,x_k,\hat{\theta},t) - y_r^{(k)}(t) \tag{6.31}$$

Thus the closed-loop form of $\dot{z}_k$ is

$$\dot{z}_k = -z_{k-1} - c_k z_k + z_{k+1} + \omega_k^T\tilde{\theta} - \frac{\partial\alpha_{k-1}}{\partial\hat{\theta}}\left(\dot{\hat{\theta}} - \tau_k\right) + \left(\sum_{i=1}^{k-2} z_{i+1}\frac{\partial\alpha_i}{\partial\hat{\theta}}\right)\Gamma\omega_k \tag{6.32}$$

and, since $\tau_k$ is not considered an update law but rather the $k$-th tuning function

$$\dot{V}_k = -\sum_{i=1}^{k} c_i z_i^2 + z_k z_{k+1} - \left(\sum_{i=1}^{k-2} z_{i+1}\frac{\partial\alpha_i}{\partial\hat{\theta}}\right)\left(\dot{\hat{\theta}} - \tau_k\right) + \tilde{\theta}^T\Gamma^{-1}\left(-\dot{\hat{\theta}} + \tau_k\right). \tag{6.33}$$

The control design is completed at the next final step.

**Step n.** Using the definition (6.31) for $k = n-1$, the time derivative of the error variable $z_n$ is

$$\dot{z}_n = \varphi_0(x) + \beta_0(x)u - \sum_{i=1}^{n-1} \frac{\partial \alpha_{n-1}}{\partial x_i} x_{i+1} + \omega_n^T \theta - \frac{\partial \alpha_{n-1}}{\partial \hat{\theta}} \dot{\hat{\theta}} - \frac{\partial \alpha_{n-1}}{\partial t} - y_r^{(n)}(t) \qquad (6.34)$$

where the last regressor vector is defined as

$$\omega_n := \varphi_n - \sum_{i=1}^{n-1} \frac{\partial \alpha_{n-1}}{\partial x_i} \varphi_i \qquad (6.35)$$

By adding and subtracting $\hat{\theta}$ in (6.34), $\dot{z}_n$ can be rewritten as

$$\dot{z}_n = \varphi_0(x) + \beta_0(x)u - \sum_{i=1}^{n-1} \frac{\partial \alpha_{n-1}}{\partial x_i} x_{i+1} + \omega_n^T(x,\hat{\theta})\hat{\theta} - \frac{\partial \alpha_{n-1}}{\partial \hat{\theta}} \dot{\hat{\theta}} - \frac{\partial \alpha_{n-1}}{\partial t} - y_r^{(n)}(t) + \omega_n^T(x,\hat{\theta})\tilde{\theta}. \qquad (6.36)$$

Consider the *sliding surface* $\sigma$ defined in terms of the error variables as

$$\sigma = k_1 z_1 + k_2 z_2 + \ldots + k_{n-1} z_{n-1} + z_n = 0 \qquad (6.37)$$

where the positive scalar design parameters $k_i$, $i = 1, \ldots, n-1$, are chosen in such a manner that the polynomial

$$p(s) = k_1 + k_2 s + \ldots + k_{n-1} s^{n-2} + s^{n-1} \qquad (6.38)$$

in the complex variable $s$ is Hurwitz. Then the Lyapunov function is augmented as follows

$$V_n = V_{n-1} + \frac{1}{2}\sigma^2 = \frac{1}{2}\sum_{i=1}^{n-1} z_i^2 + \frac{1}{2}\sigma^2 + \frac{1}{2}\tilde{\theta}^T \Gamma^{-1} \tilde{\theta}. \qquad (6.39)$$

The time derivative of $V_n$ is

$$\begin{aligned}
\dot{V}_n = & -\sum_{i=1}^{n-1} c_i z_i^2 + z_n z_{n-1} - \left(\sum_{i=1}^{n-2} z_{i+1} \frac{\partial \alpha_i}{\partial \hat{\theta}}\right)\left(\dot{\hat{\theta}} - \tau_{n-1}\right) \\
& + \sigma \Bigg[ \varphi_0(x) + \beta_0(x)u - \sum_{i=1}^{n-1} \frac{\partial \alpha_{n-1}}{\partial x_i} x_{i+1} + \omega_n^T \hat{\theta} - \frac{\partial \alpha_{n-1}}{\partial \hat{\theta}} \dot{\hat{\theta}} - \frac{\partial \alpha_{n-1}}{\partial t} - y_r^{(n)}(t) \\
& \quad + \sum_{i=1}^{n-1} k_i \Bigg( - z_{i-1} - c_i z_i + z_{i+1} - \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}}\left(\dot{\hat{\theta}} - \tau_i\right) + \left(\sum_{j=1}^{i-2} z_{j+1} \frac{\partial \alpha_j}{\partial \hat{\theta}}\right) \Gamma \omega_i \Bigg) \Bigg] \\
& + \tilde{\theta}^T \Gamma^{-1} \left[ -\dot{\hat{\theta}} + \tau_{n-1} + \Gamma \sigma \left( \omega_n + \sum_{i=1}^{n-1} k_i \omega_i \right) \right]
\end{aligned} \qquad (6.40)$$

One can eliminate $\tilde{\theta}$ from $\dot{V}_n$ by choosing the update law

$$\begin{aligned}\dot{\hat{\theta}} := \tau_n &= \tau_{n-1} + \Gamma\sigma\Big(\omega_n + \sum_{i=1}^{n-1} k_i\omega_i\Big) \\ &= \Gamma\Big[\sum_{i=1}^{n-1}\omega_i z_i + \sigma\Big(\omega_n + \sum_{i=1}^{n-1} k_i\omega_i\Big)\Big] \qquad (6.41)\end{aligned}$$

Noting that

$$\dot{\hat{\theta}} - \tau_{n-1} = \tau_n - \tau_{n-1} = \Gamma\sigma\Big(\omega_n + \sum_{i=1}^{n-1} k_i\omega_i\Big) \qquad (6.42)$$

$\dot{V}_n$ can be rewritten as

$$\begin{aligned}\dot{V}_n = &-\sum_{i=1}^{n-1} c_i z_i^2 + z_n z_{n-1} \\ &+\sigma\Bigg[\varphi_0(x) + \beta_0(x)u - \sum_{i=1}^{n-1}\frac{\partial\alpha_{n-1}}{\partial x_i}x_{i+1} + \omega_n^T\hat{\theta} - \frac{\partial\alpha_{n-1}}{\partial\hat{\theta}}\dot{\hat{\theta}} - \frac{\partial\alpha_{n-1}}{\partial t} - y_r^{(n)}(t) \\ &+\sum_{i=1}^{n-1}k_i\Big(-z_{i-1} - c_i z_i + z_{i+1} - \frac{\partial\alpha_{i-1}}{\partial\hat{\theta}}\Big(\tau_n - \tau_i\Big) + \Big(\sum_{j=1}^{i-2} z_{j+1}\frac{\partial\alpha_j}{\partial\hat{\theta}}\Big)\Gamma\omega_i\Big) \\ &-\Big(\sum_{i=1}^{n-2} z_{i+1}\frac{\partial\alpha_i}{\partial\hat{\theta}}\Big)\Gamma\Big(\omega_n + \sum_{i=1}^{n-1} k_i\omega_i\Big)\Bigg] \qquad (6.43)\end{aligned}$$

To achieve

$$\dot{V}_n = -\sum_{i=1}^{n-1} c_i z_i^2 + z_n z_{n-1} - \lambda\sigma^2 - \lambda\beta|\sigma| \qquad (6.44)$$

the control $u$ should be chosen to make the bracketed term multiplying $\sigma$ equal to $-\lambda(\sigma + \beta\,\mathrm{sgn}(\sigma))$, where $\lambda$ and $\beta$ are positive design parameters and sgn is the *signum* function. We can select

$$\begin{aligned}u = &\frac{1}{\beta_0(x)}\Bigg[-\varphi_0(x) + \sum_{i=1}^{n-1}\frac{\partial\alpha_{n-1}}{\partial x_i}x_{i+1} - \omega_n^T\hat{\theta} + \frac{\partial\alpha_{n-1}}{\partial\hat{\theta}}\tau_n + \frac{\partial\alpha_{n-1}}{\partial t} + y_r^{(n)}(t) \\ &-\sum_{i=1}^{n-1}k_i\Big(-z_{i-1} - c_i z_i + z_{i+1} - \frac{\partial\alpha_{i-1}}{\partial\hat{\theta}}\Big(\tau_n - \tau_i\Big) + \Big(\sum_{j=1}^{i-2} z_{j+1}\frac{\partial\alpha_j}{\partial\hat{\theta}}\Big)\Gamma\omega_i\Big) \\ &+\Big(\sum_{i=1}^{n-2} z_{i+1}\frac{\partial\alpha_i}{\partial\hat{\theta}}\Big)\Gamma\Big(\omega_n + \sum_{i=1}^{n-1} k_i\omega_i\Big) - \lambda\Big(\sigma + \beta\,\mathrm{sgn}(\sigma)\Big)\Bigg] \qquad (6.45)\end{aligned}$$

## 6.3 Analysis of Stability

Note that we can rewrite the time derivative of the Lyapunov function (6.44) as

$$\dot{V}_n = -z^T Q z - \lambda\beta|\sigma| \qquad (6.46)$$

where $Q$ is a symmetric matrix with the following form

$$Q = \begin{bmatrix} c_1 + \lambda k_1^2 & \lambda k_1 k_2 & \dots & \lambda k_1 k_{n-1} & \lambda k_1 \\ \lambda k_2 k_1 & c_2 + \lambda k_2^2 & \dots & \lambda k_2 k_{n-1} & \lambda k_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \lambda k_{n-1} k_1 & c_2 + \lambda k_{n-1} k_2 & \dots & c_{n-1} + \lambda k_{n-1}^2 & -\frac{1}{2} + \lambda k_{n-1} \\ \lambda k_1 & \lambda k_2 & \dots & -\frac{1}{2} + \lambda k_{n-1} & \lambda \end{bmatrix} \tag{6.47}$$

Noting that the determinants of the principal minors of $Q$ are all positive and have the form

$$\prod_{i=1}^{d} c_i + \lambda \sum_{i=1}^{d} (c_1 c_2 \dots c_{i-1} k_i^2 c_{i+1} \dots c_d) > 0, \qquad 1 \leqslant d \leqslant n-1, \tag{6.48}$$

a sufficient condition to guarantee that $Q$ is positive definite is

$$|Q| = \left[ -\frac{1}{4} + \lambda (c_{n-1} + k_{n-1}) \right] \prod_{i=1}^{n-2} c_i - \frac{1}{4} \lambda \sum_{i=1}^{n-2} (c_1 \dots c_{i-1} k_i^2 c_{i+1} \dots c_{n-2}) > 0. \tag{6.49}$$

The design procedure above guarantees that $\sigma\dot{\sigma} \leqslant 0$, and, thus, a sliding mode is generated on the sliding surface $\sigma = 0$.

Since the right hand side of the differential equations which characterize the closed-loop system are discontinuous, the Lipschitz continuity condition is violated and, consequently, we cannot use standard theorems for existence of solutions. Nevertheless, a stability proof could be constructed by using the *integral invariance principle* for differential inclusions formulated by Ryan [96]. The integral invariance principle was firstly introduced by Byrnes and Martin [12] as a generalization of the LaSalle invariance principle. Then, Ryan considered in [96] the control design problem in a more general setting that allows time variation in the differential equations, possible non-uniqueness of solutions and *discontinuous* feedback strategies.

When the perturbed system incorporates a matched disturbance with known bounds, classical disturbance rejection techniques in the setting of sliding mode control (see [122], [126]), can be used to reject the undesirable perturbation. However, we consider here the effect of an *unmatched* bounded disturbance without information regarding bounds, and show that the closed-loop system obtained by using the procedure above, exhibits a certain degree of insensitivity to such disturbances (robustness) as a consequence of the sliding mode generated. This is illustrated in the following example.

## 6.4 Robust Tracking Control of PSF Systems

**Example 6.1** Consider the unperturbed third order system in PSF form

$$\begin{aligned} \dot{x}_1 &= x_2 + x_1^2\theta \\ \dot{x}_2 &= x_3 \\ \dot{x}_3 &= u \\ y &= x_1 \end{aligned} \tag{6.50}$$

where $\theta$ is an unknown constant parameter. The control objective is the tracking of a known bounded reference signal $y_r(t)$ with bounded derivatives even in the presence of disturbances. From the application of the backstepping-SMC algorithm explained above, the following tracking error variables are obtained

$$\begin{aligned} z_1 &= x_1 - y_r(t) \\ z_2 &= x_2 - \alpha_1(x_1, \hat{\theta}, t) - \dot{y}_r(t) \\ z_2 &= x_3 - \alpha_2(x_1, x_2, \hat{\theta}, t) - \ddot{y}_r(t) \end{aligned} \tag{6.51}$$

$$\tag{6.52}$$

where

$$\alpha_1(x_1, \hat{\theta}, t) = -\omega_1\hat{\theta} - c_1 z_1 \tag{6.53}$$

$$\alpha_2(x_1, x_2, \hat{\theta}, t) = -z_1 + \frac{\partial \alpha_1}{\partial x_1}x_2 - \omega_2\hat{\theta} + \frac{\partial \alpha_1}{\partial \hat{\theta}}\tau_2 + \frac{\partial \alpha_1}{\partial t} - c_2 z_2$$

with the regressor vectors defined as

$$\begin{aligned} \omega_1 &= x_1^2 \\ \omega_2 &= -\frac{\partial \alpha_1}{\partial x_1}x_1^2 \\ \omega_3 &= -\frac{\partial \alpha_2}{\partial x_1}x_1^2 \end{aligned} \tag{6.54}$$

Defining the adaptive sliding surface

$$\sigma = k_1 z_1 + k_2 z_2 + z_3 = 0 \qquad k_1,\ k_2 > 0 \tag{6.55}$$

the tuning functions are obtained as

$$\begin{aligned} \tau_1 &= \gamma\omega_1 z_1 \\ \tau_2 &= \tau_1 + \gamma\omega_2 z_2 \\ \tau_3 &= \tau_2 + \gamma\sigma(\omega_3 + k_1\omega_1 + k_2\omega_2) \end{aligned} \tag{6.56}$$

where the update law is $\dot{\hat{\theta}} = \tau_3$, and the discontinuous adaptive control law is

$$\begin{aligned} u &= \frac{\partial \alpha_2}{\partial x_1} x_2 + \frac{\partial \alpha_2}{\partial x_2} x_3 - \omega_3 \hat{\theta} + \frac{\partial \alpha_2}{\partial \hat{\theta}} \tau_3 + y_r^{(3)}(t) - k_1(-c_1 z_1 + z_2) - k_2(-z_1 - c_2 z_2 + z_3) \\ &\quad + (k_2 + z_2)\gamma\sigma \frac{\partial \alpha_1}{\partial \hat{\theta}} (\omega_3 + k_1\omega_1 + k_2\omega_2) - \lambda\Big(\sigma + \beta\,\mathrm{sgn}(\sigma)\Big). \end{aligned} \tag{6.57}$$

Using the update law and the adaptive controller thus designed, the time derivative of the Lyapunov function

$$V = \frac{1}{2}z_1^2 + \frac{1}{2}z_2^2 + \frac{1}{2\gamma}\tilde{\theta}^2 + \frac{1}{2}\sigma^2 \tag{6.58}$$

becomes

$$\dot{V} = -z^T Q z - \lambda\beta|\sigma| \tag{6.59}$$

with

$$Q = \begin{bmatrix} c_1 + \lambda k_1^2 & \lambda k_1 k_2 & \lambda k_1 \\ \lambda k_2 k_1 & c_2 + \lambda k_2^2 & -\frac{1}{2} + \lambda k_2 \\ \lambda k_3 k_1 & -\frac{1}{2} + \lambda k_2 & \lambda \end{bmatrix} \tag{6.60}$$

Thus, if the design parameters satisfy the condition (6.49), global asymptotic tracking is achieved, namely

$$\lim_{t\to\infty} z_1(t) = \Big(y(t) - y_r(t)\Big) = 0, \tag{6.61}$$

whilst a sliding mode is generated on the sliding surface $\sigma = 0$.

A *perturbed* model of the third order PSF system (6.50) was used in the computer simulations. The additive perturbation was an unmatched external stochastic input $v$ entering at the first differential equation

$$\begin{aligned} \dot{x}_1 &= x_2 + x_1^2\theta + v \\ \dot{x}_2 &= x_3 \\ \dot{x}_3 &= u \\ y &= x_1 \end{aligned} \tag{6.62}$$

The reference signal to be tracked was chosen to be

$$y_r(t) = \sin(10t) \tag{6.63}$$

and the perturbation input $v = 4\sin(80t)$.

Computer simulations were carried out for a nominal parameter value $\theta = 1$. Figure 6.1 shows the tracking performance, the state variable responses and the evolution of both the parameter estimate and the sliding function in the absence of the perturbation.

Figure 6.2 depicts the robust asymptotic tracking achieved by the combined controller in the presence of the perturbation. The design parameters used to obtain both figures were $c_1 = 8$, $c_2 = 8$, $k_1 = 5$, $k_2 = 15$, $\lambda = 3$, $\beta = 5$ and $\gamma = 0.001$. Computer simulations were also performed for an adaptive controller obtained by using the SAB algorithm of Section 2.4.3. Figures 6.3 and 6.4 show the tracking performance and evolution of the controlled variables in the absence and presence of the perturbation respectively for the design parameters $c_1 = 5$, $c_2 = 5$, $c_3 = 8$ and $\gamma = 1$. The responses obtained by the combined SAB-SMC controller exhibited a more robust behaviour and reduced tracking error in comparison with the SAB controlled responses. Aditionally the parameter estimate function is more insensitive to the perturbation in the case of the combined SAB-SMC. This aspect is very important in preserving asymptotic tracking.

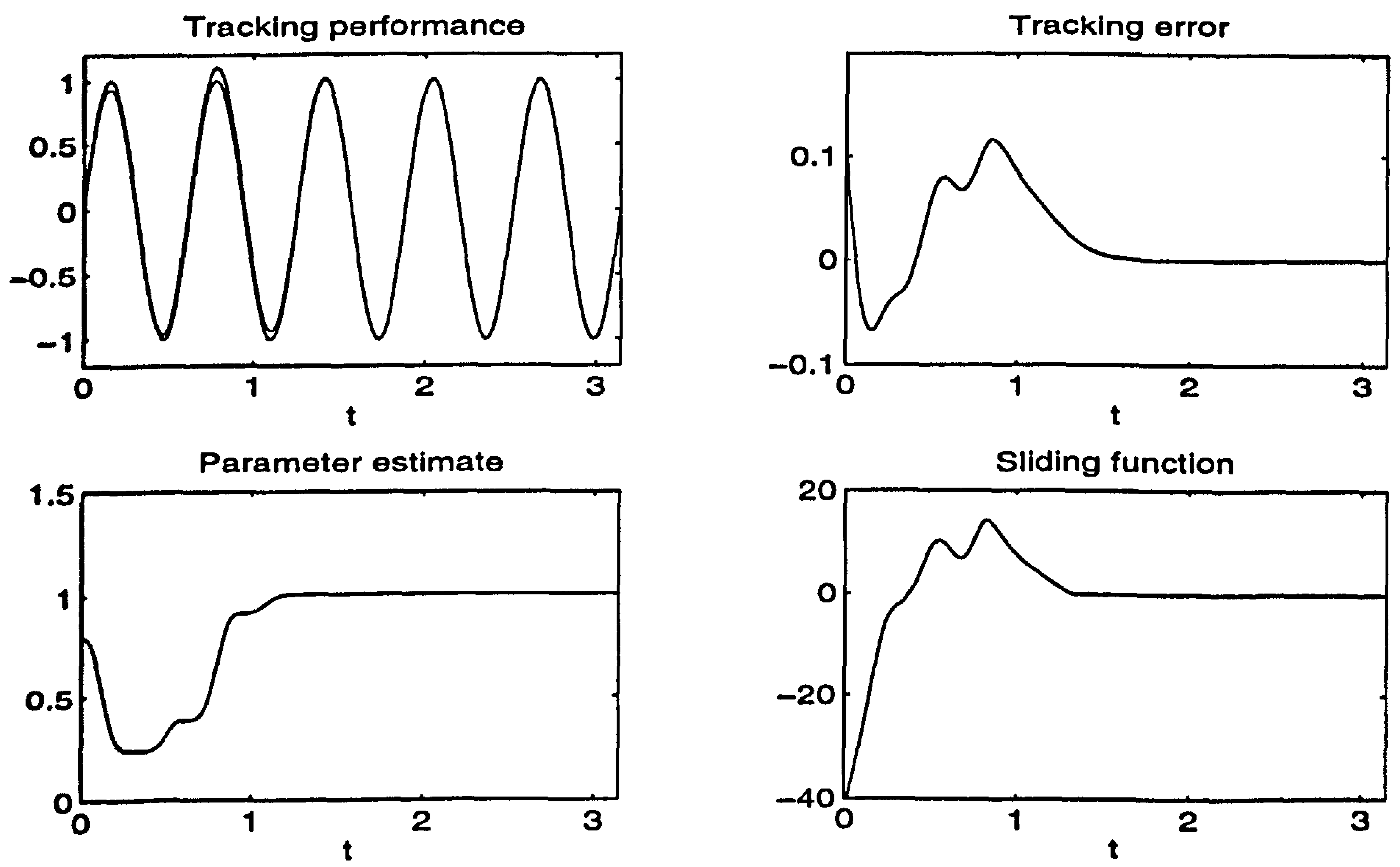


Figure 6.1: Controlled responses of the third order system with the combined SAB-SMC algorithm in the absence of the perturbation

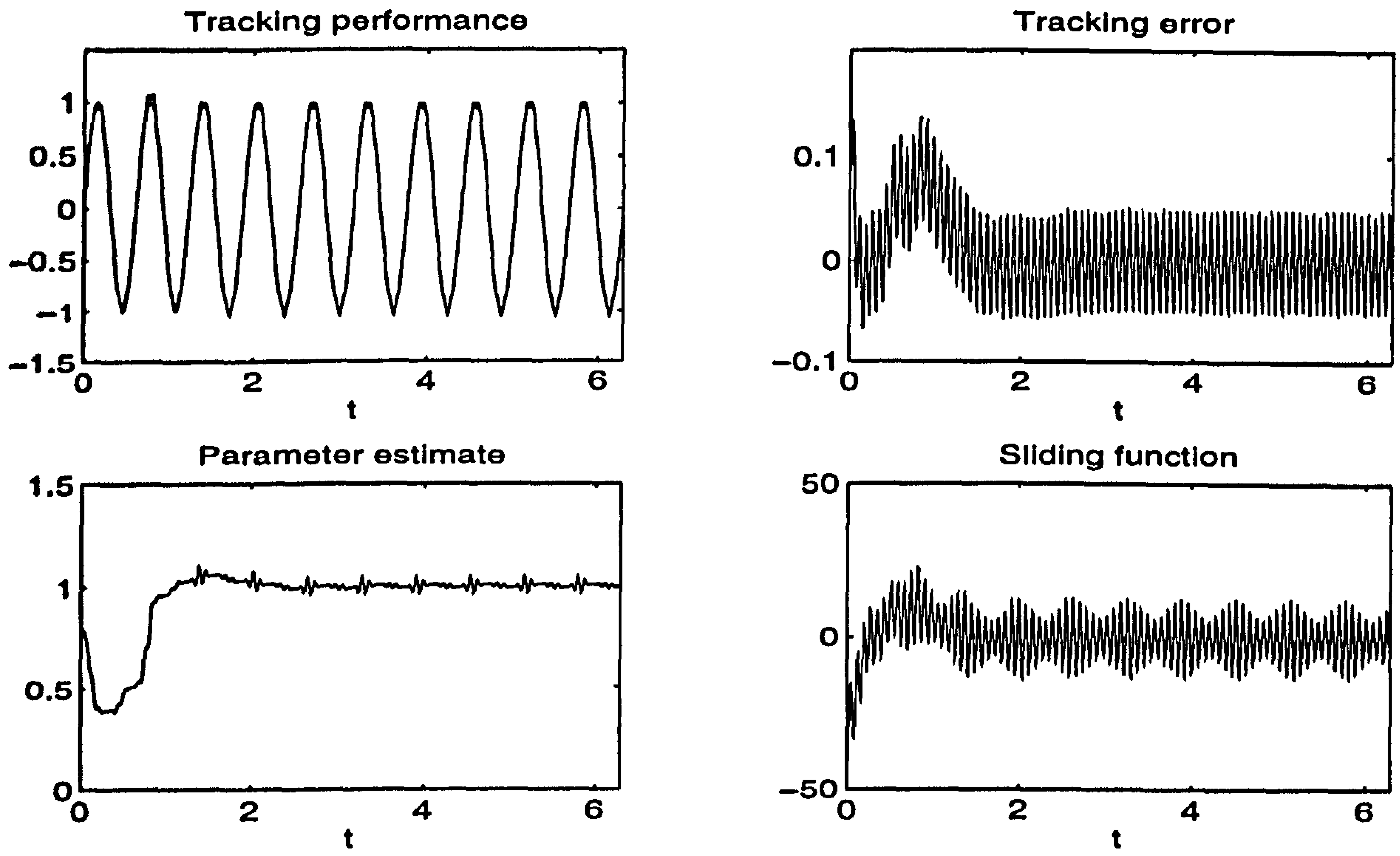


Figure 6.2: Controlled responses of the third order system with the combined SAB-SMC algorithm in the presence of the perturbation

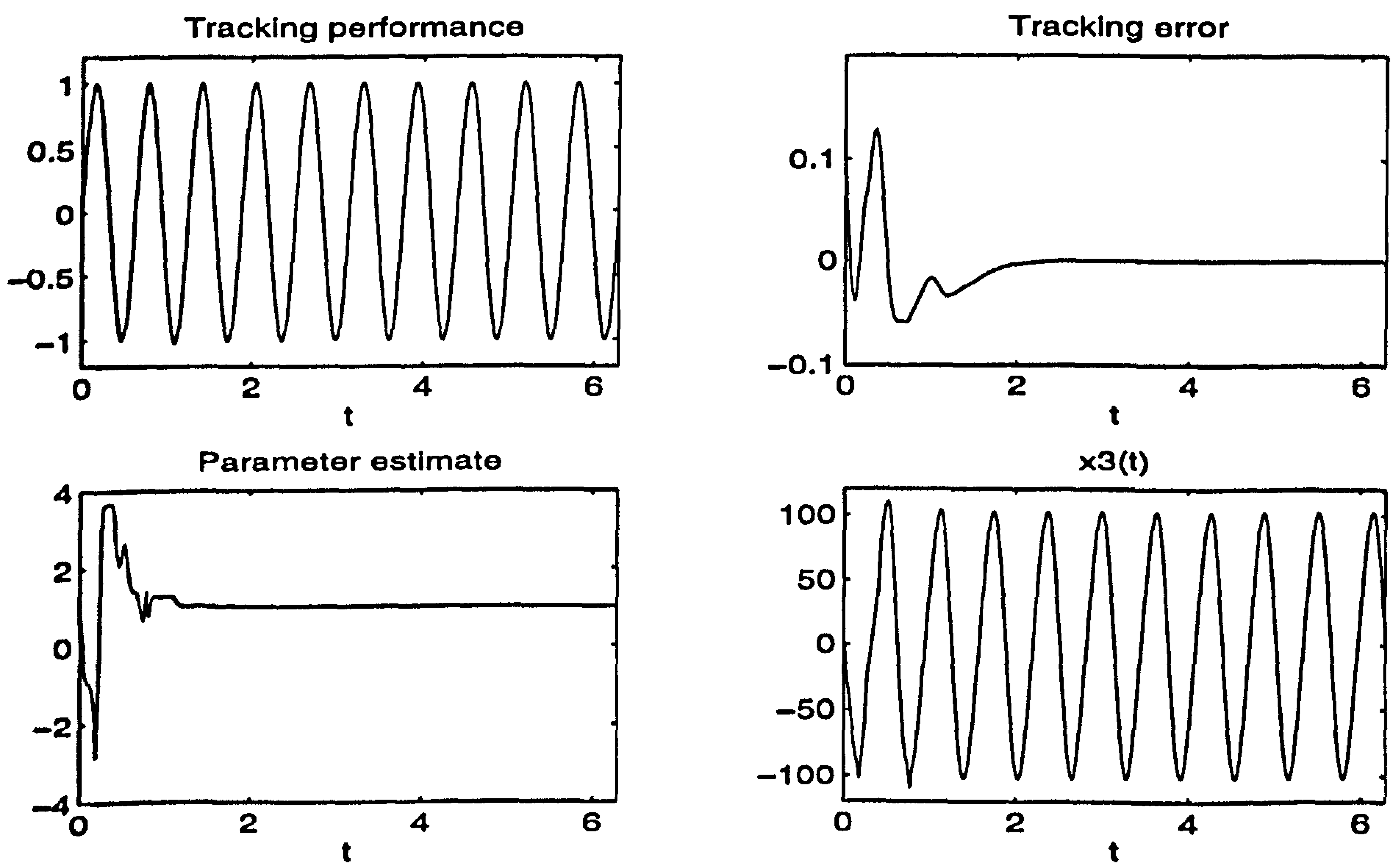


Figure 6.3: Controlled responses of a third order system with the SAB algorithm in the absence of the perturbation

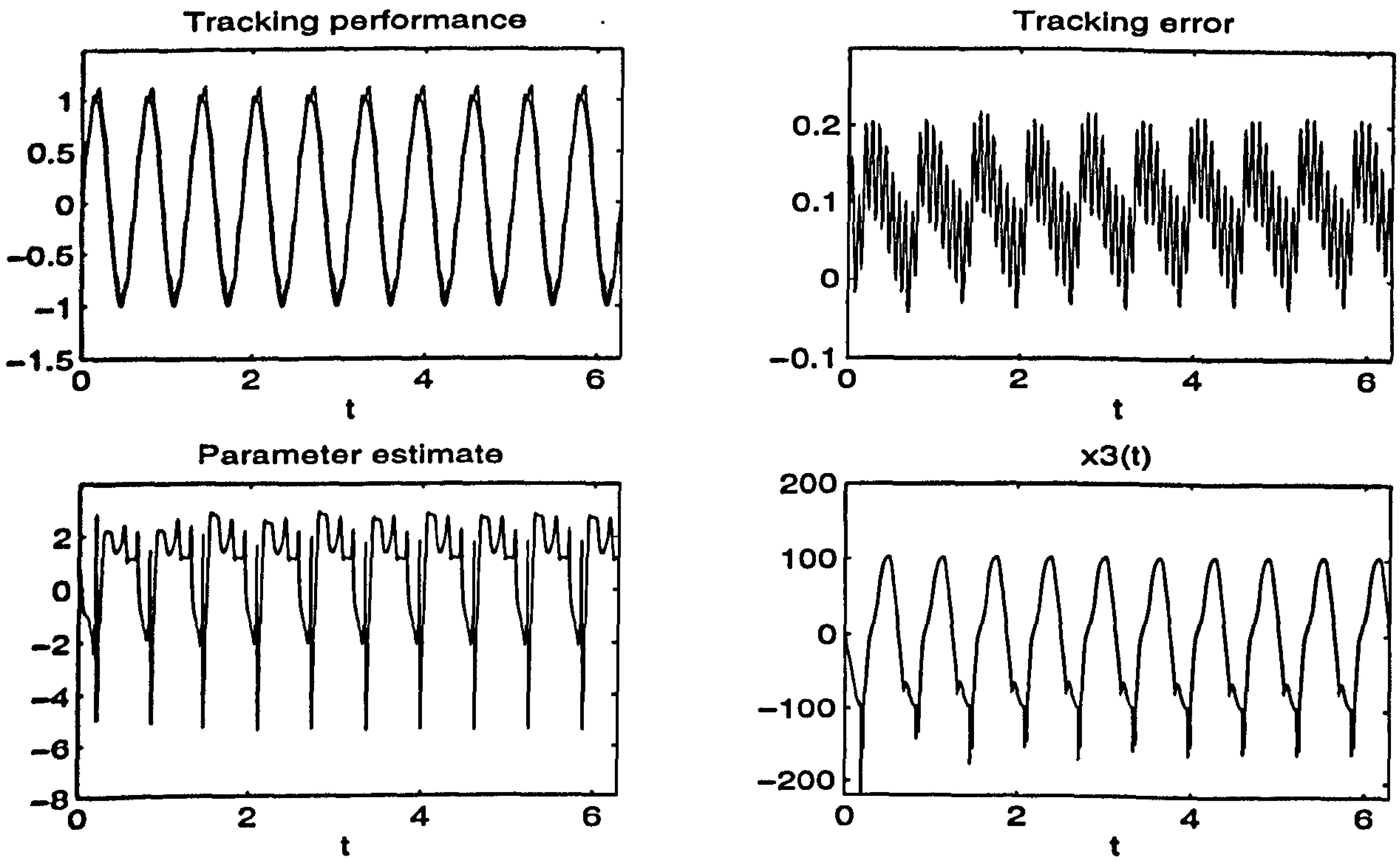


Figure 6.4: Controlled responses of a third order system with the SAB algorithm in the presence of the perturbation

# Chapter 7

# DAB and SMC of Uncertain Nonlinear Systems with Disturbances

## 7.1 Introduction

The results of Section 6.2 are applicable only to linearizable uncertain nonlinear systems which are transformable into the PSF form. We consider in this chapter the extension of these results to a broader class of nonlinear systems with parametric uncertainty. In particular, we propose a new combined adaptive sliding mode control approach based upon the DAB algorithm given in Section 4.2 and SMC to deal with uncertain systems which are not necessarily in the PSF and PPF forms. The controller thus obtained exhibits robust behaviour in the presence of undesirable perturbation inputs.

The design technique proposed here follows a similar procedure to that of Section 6.2 and the same sufficient condition on the design parameters, guarantees asymptotic tracking and stability of the controlled state variables. As in the case of the DAB algorithm described in Section 4.2, the design procedure is applicable to observable minimum phase uncertain nonlinear systems and the resulting stability is in general local.

We start the chapter describing the earlier proposed approach for the design of dynamical sliding mode controllers for tracking tasks of non-adaptive systems. Then the combined DAB-SMC algorithm is presented and its stability properties are analysed. Finally a field-controlled DC motor with five uncertain parameters is used as an illustrative example of the new control design algorithm.

## 7.2 Dynamical SMC Design for Deterministic Nonlinear Systems

Dynamical discontinuous feedback control of the sliding mode type has been proposed by Sira-Ramírez [105, 106] for asymptotic tracking problems in nonlinear systems *without* uncertainty. The dynamical feedback controller is designed by using the *Generalized Observability Canonical Form (GOCF)* (3.98) and achieves asymptotic tracking for a class of non-affine minimum-phase observable systems. As a motivation for the asymptotic tracking problem for uncertain nonlinear systems, the approach in [106] is summarized below.

Consider a single-input single-output non-affine minimum-phase nonlinear system of the form

$$\begin{aligned} \dot{x} &= f(x,u) \\ y &= h(x) \end{aligned} \tag{7.1}$$

Assuming that (7.1) is transformable into the GOCF (see Section 3.3)

$$\begin{aligned} \dot{\xi}_1 &= \xi_2 \\ \dot{\xi}_2 &= \xi_3 \\ &\vdots \\ \dot{\xi}_{n-1} &= \xi_n \\ \dot{\xi}_n &= c(\xi, u, \ldots, u^{(n-\rho)}) \\ y &= \xi_1 \end{aligned} \tag{7.2}$$

where $\rho$ is the relative degree of (7.1). The problem of tracking a bounded reference signal $y_r(t)$ with bounded derivatives can be solved as follows. Defining the tracking error function $e(t)$ as the difference between the actual system output $y(t)$ and the desired bounded output reference signal $y_r(t)$,

$$e(t) = y(t) - y_r(t) \tag{7.3}$$

we obtain

$$\begin{aligned} e^{(i)}(t) &= \xi_{i+1} - y_r^{(i)}(t); \qquad 0 \leqslant i \leqslant n-1 \\ e^{(n)}(t) &= \dot{\xi}_n - y_r^{(n)}(t) = c(\xi, u, \dot{u}, \ldots, u^{(n-\rho)}) - y_r^{(n)}(t). \end{aligned}$$

Then, defining $z_i = e^{(i-1)}$, $i = 1, \ldots, n$, as the components of the error vector $z$, we express the tracking error system in GOCF as

$$\begin{aligned}
\dot{z}_1 &= z_2 \\
\dot{z}_2 &= z_3 \\
&\vdots \\
\dot{z}_{n-1} &= z_n \\
\dot{z}_n &= c(\xi_r(t) + z, u, \dot{u}, \dots, u^{(n-\rho)}) - y_r^{(n)}(t) \\
y_z &= z_1
\end{aligned} \tag{7.4}$$

with

$$\begin{aligned}
\xi_r(t) &= \left[y_r(t), y_r^{(1)}(t), \dots, y_r^{(n-1)}(t)\right]^T \\
z &= [z_1, z_2, \dots, z_n]^T
\end{aligned} \tag{7.5}$$

Under the assumption that the condition

$$\frac{\partial c(\xi_r(t) + z, u, \dots, u^{(n-\rho)})}{\partial u^{(n-\rho)}} \neq 0 \tag{7.6}$$

is satisfied locally, no singularities need to be considered in the design of a dynamical SMC. The sliding surface is specified in terms of the output tracking error coordinates as

$$\sigma = z_n + c_{n-2}z_{n-1} + \dots + c_1 z_2 + c_0 z_1 = 0 \tag{7.7}$$

such that the polynomial

$$p(s) = s^{n-1} + c_{n-2}s^{n-2} + \dots + c_1 s + c_0 \tag{7.8}$$

in the complex variable $s$ is Hurwitz. Finally, the dynamical SMC

$$c(\xi_r(t) + z, u, \dot{u}, \dots, u^{(n-\rho)}) = y_r^{(n)}(t) - \sum_{i=1}^{n-1} c_{i-1}z_{i+1} - \lambda(\sigma + \beta\,\mathrm{sgn}(\sigma)) \tag{7.9}$$

with $\lambda > 0$ and $\beta > 0$, can be obtained in terms of an implicit ordinary differential equation with discontinuous right-hand side. The *ideal sliding dynamics*, obtained from the invariance conditions $\sigma = 0,\ \dot{\sigma} = 0$ [122], is

$$\begin{aligned}
\dot{z}_1 &= z_2 \\
\dot{z}_2 &= z_3 \\
&\vdots \\
\dot{z}_{n-1} &= -\sum_{i=1}^{n-1} c_{i-1}z_i
\end{aligned} \tag{7.10}$$

and exhibits asymptotically stable motion towards the origin of the error coordinates. Consequently, the output tracking error function $z_1$ converges asymptotically to zero, as desired.

This approach is suitable for perfectly known minimum-phase observable nonlinear systems. Recently, Spurgeon and Lu [115] proposed different methods for output tracking of non-minimum phase systems without uncertainty via dynamical compensation and SMC.

## 7.3 DAB-SMC Algorithm

We next consider an adaptive version of the procedure in Section 7.2 which is applicable to a class of minimum-phase observable uncertain nonlinear systems. This adaptive control design method has been proposed by Rios-Bolívar *et al* [92, 94] to achieve robust tracking in the presence of uncertainty and disturbances.

Consider the same class of single-input single-output uncertain nonlinear systems as analysed in Section 4.2, which can be represented by the dynamic equations

$$\begin{aligned} \dot{x} &= f_0(x) + \Psi(x)\theta + \Big(g_0(x) + \varphi(x)\theta\Big)u \\ y &= h(x) \end{aligned} \tag{7.11}$$

where $x \in \Re^n$ is the state, $u, y \in \Re$ the input and output respectively and $\theta = [\theta_1, \ldots, \theta_p]^T$ a vector of unknown parameters. $f_0$, $g_0$ and the columns of the matrices $\Psi, \varphi \in \Re^{n \times p}$ are smooth vector fields in a neighbourhood $R_0$ of the origin $x = 0$, with $f_0(0) = 0$, $g_0(0) \neq 0$, and $h$ is a smooth scalar function also defined in $R_0$. It is assumed that system (7.11) has relative degree $\rho$ strictly less than the system order. The control objective is to drive the system output $y(t)$ to track asymptotically a bounded reference signal $y_r(t)$ with bounded derivatives in the presence of disturbances.

Under Assumptions 4.1 and 4.2, this objective can be accomplished by combining the DAB algorithm of Section 4.2 and SMC. Adopting a similar procedure to that of the previous section, the first $n-1$ steps follow the algorithm described in Section 4.2. At the final step a sliding surface is defined in terms of the tracking error variables and both an update law and a dynamical discontinuous feedback law are also synthesized. The combined DAB-SMC algorithm can be summarized as follows:

***DAB-SMC Algorithm***

*Coordinate transformation*

$$\begin{aligned} z_1 &:= y - y_r(t) = h^{(0)}(x) - y_r(t) \qquad (7.12) \\ z_k &:= \hat{h}^{(k-1)}(x, \hat{\theta}, v_1, \ldots, v_{(k-\rho)}, t) - y_r^{(k-1)} + \alpha_{k-1}(x, \hat{\theta}, v_1, \ldots, v_{(k-\rho)}, t) \quad 2 \leqslant k \leqslant n \end{aligned}$$

*with*

$$\begin{aligned} \hat{h}^k &= \frac{\partial \hat{h}^{(k-1)}}{\partial \hat{\theta}} \tau_k + \frac{\partial \hat{h}^{(k-1)}}{\partial x} \Big[ f_0 + \Psi \hat{\theta} + (g_0 + \varphi \hat{\theta}) v_1 \Big] \\ &\quad + \sum_{i=1}^{k-\rho-1} \frac{\partial \hat{h}^{(k-1)}}{\partial v_i} v_{i+1} + \frac{\partial \hat{h}^{(k-1)}}{\partial t} \qquad (7.13) \end{aligned}$$

$$\omega_k = \left( \frac{\partial \hat{h}^{(k-1)}}{\partial x} + \frac{\partial \alpha_{k-1}}{\partial x} \right) (\Psi + \varphi u) \qquad (7.14)$$

$$\begin{aligned} \alpha_k &= z_{k-1} + \left( \sum_{i=2}^{k-1} z_i \frac{\partial \hat{h}^{(i-1)}}{\partial \hat{\theta}} + \sum_{i=3}^{k-1} z_i \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}} \right) \Gamma \omega_k^T + \sum_{i=1}^{k-\rho-1} \frac{\partial \alpha_{k-1}}{\partial v_i} v_{i+1} \\ &\quad + \frac{\partial \alpha_{k-1}}{\partial x} \Big[ f_0 + \Psi \hat{\theta} + (g_0 + \varphi \hat{\theta}) v_1 \Big] + \frac{\partial \alpha_{k-1}}{\partial \hat{\theta}} \tau_k + \frac{\partial \alpha_{k-1}}{\partial t} + c_k z_k \qquad (7.15) \end{aligned}$$

$$\tau_k = \Gamma \sum_{i=1}^{k} \omega_k^T z_k \qquad 1 \leqslant k \leqslant n-1 \qquad (7.16)$$

*Define the sliding surface*

$$\sigma = k_1 z_1 + k_2 z_2 + \ldots + k_{n-1} z_{n-1} + z_n = 0 \qquad (7.17)$$

*with the design parameters* $k_i$, $i = 1, \ldots, n-1$, *chosen such that the polynomial*

$$p(s) = k_1 + k_2 s + \ldots + k_{n-1} s^{n-2} + s^{n-1} \qquad (7.18)$$

*in the complex variable* $s$ *is Hurwitz.*

*Parameter update law*

$$\dot{\hat{\theta}} = \tau_n = \tau_{n-1} + \Gamma \sigma \Big( \omega_n^T + \sum_{i=1}^{n-1} k_i \omega_i^T \Big) \qquad (7.19)$$

*DAB-SMC Algorithm (cont.)*

*DAB-SMC law*

$$\begin{aligned}
\dot{v}_1 &= v_2 \\
&\vdots \\
\dot{v}_{n-\rho-1} &= v_{n-\rho} \\
\dot{v}_{n-\rho} &= \frac{1}{\left(\frac{\partial \hat{h}^{(n-1)}}{\partial v_{n-\rho}} + \frac{\partial \alpha_{n-1}}{\partial v_{n-\rho}}\right)} \Bigg[ -\sum_{i=2}^{n-1} \left( \frac{\partial \hat{h}^{(n-1)}}{\partial \hat{\theta}} + \frac{\partial \alpha_{n-1}}{\partial \hat{\theta}} \right) z_i \Big( \Gamma \omega_n^T + \sum_{i=1}^{n-1} k_i \omega_i^T \Big) + y_r^{(n)}(t) \\
&\quad - \left( \frac{\partial \hat{h}^{(n-1)}}{\partial x} + \frac{\partial \alpha_{n-1}}{\partial x} \right) \Big( f_0 + \Psi \hat{\theta} + (g_0 + \varphi \hat{\theta}) v_1 \Big) - \frac{\partial \hat{h}^{(n-1)}}{\partial t} - \frac{\partial \alpha_{n-1}}{\partial t} \\
&\quad - \left( \frac{\partial \hat{h}^{(n-1)}}{\partial \hat{\theta}} + \frac{\partial \alpha_{n-1}}{\partial \hat{\theta}} \right) \tau_n - \sum_{i=1}^{n-\rho-1} \left( \frac{\partial \hat{h}^{(n-1)}}{\partial v_i} + \frac{\partial \alpha_{n-1}}{\partial v_i} \right) v_{i+1} \\
&\quad - \sum_{i=1}^{n-1} k_i \left( -z_{i-1} - c_i z_i + z_{i+1} - \left( \sum_{j=2}^{i-1} z_j \frac{\partial \hat{h}^{(j-1)}}{\partial \hat{\theta}} + \sum_{j=3}^{i-1} z_j \frac{\partial \alpha_{j-1}}{\partial \hat{\theta}} \right) \Gamma \omega_i^T \right) \\
&\quad + \sum_{i=1}^{n-1} k_i \left( \frac{\partial \hat{h}^{(i-1)}}{\partial \hat{\theta}} + \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}} \right) (\tau_n - \tau_i) - \lambda \Big( \sigma + \beta \, \mathrm{sgn}(\sigma) \Big) \Bigg]
\end{aligned} \tag{7.20}$$

Since the final step is the only one different from the DAB algorithm given in Section 4.2, it is described next.

**Step n.** At this step we obtain the update law and the *dynamical adaptive sliding mode* tracking controller. After the $k = n-1$ step of the DAB algorithm, the transformed system is

$$\begin{aligned}
\dot{z}_1 &= -c_1 z_1 + z_2 + \omega_1(\theta - \hat{\theta}) \\
\dot{z}_2 &= -z_1 - c_2 z_2 + z_3 + \omega_2(\theta - \hat{\theta}) + \frac{\partial \hat{h}^{(1)}}{\partial \hat{\theta}} (\dot{\hat{\theta}} - \tau_2) \\
&\vdots \\
\dot{z}_k &= -z_{k-1} - c_k z_k + z_{k+1} + \omega_k(\theta - \hat{\theta}) + \left( \frac{\partial \hat{h}^{(k-1)}}{\partial \hat{\theta}} + \frac{\partial \alpha_{k-1}}{\partial \hat{\theta}} \right) (\dot{\hat{\theta}} - \tau_k) \\
&\quad - \left( \sum_{i=2}^{k-1} z_i \frac{\partial \hat{h}^{(i-1)}}{\partial \hat{\theta}} + \sum_{i=3}^{k-1} z_i \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}} \right) \Gamma \omega_k^T \qquad (7.21) \\
&\vdots \\
\dot{z}_n &= \hat{h}^{(n)}(x, \hat{\theta}, u, \ldots, u^{(n-\rho)}, t) - y_r^{(n)}(t) + \alpha_n(x, \hat{\theta}, u, \ldots, u^{(n-\rho-1)}, t) + \omega_n(\theta - \hat{\theta}) \\
&\quad + \left( \frac{\partial \hat{h}^{(n-1)}}{\partial \hat{\theta}} + \frac{\partial \alpha_{n-1}}{\partial \hat{\theta}} \right) (\dot{\hat{\theta}} - \tau_n)
\end{aligned}$$

$$\tau_{n-1} = \Gamma\sum_{i=1}^{n-1}\omega_i^T z_i$$

and the time derivative of $V_{n-1}$ is

$$\begin{aligned}\dot{V}_{n-1} &= -\sum_{i=1}^{n-1} c_i z_i^2 + z_{n-1}z_n + \left(\sum_{i=2}^{n-1} z_i\frac{\partial\hat{h}^{(i-1)}}{\partial\hat{\theta}} + \sum_{i=3}^{n-1} z_i\frac{\partial\alpha_{i-1}}{\partial\hat{\theta}}\right)\left(\dot{\hat{\theta}} - \tau_{n-1}\right)\\ &\quad +(\theta-\hat{\theta})^T\Gamma^{-1}\left(-\dot{\hat{\theta}} + \tau_{n-1}\right)\end{aligned} \tag{7.22}$$

We now define the sliding surface

$$\sigma = k_1 z_1 + k_2 z_2 + \ldots + k_{n-1}z_{n-1} + z_n = 0 \tag{7.23}$$

with the positive design parameters $k_i$, $i = 1,\ldots,n-1$, chosen in such a manner that the polynomial

$$p(s) = k_1 + k_2 s + \ldots + k_{n-1}s^{n-2} + s^{n-1} \tag{7.24}$$

in the complex variable $s$ is Hurwitz. Extending the Lyapunov function as

$$V_n = V_{n-1} + \frac{1}{2}\sigma^2 = \frac{1}{2}\sum_{i=1}^{n-1} z_i^2 + \frac{1}{2}\sigma^2 + \frac{1}{2}(\theta-\hat{\theta})^T\Gamma^{-1}(\theta-\hat{\theta}) \tag{7.25}$$

the time derivative of $V_n$ is

$$\begin{aligned}\dot{V}_n &= -\sum_{i=1}^{n-1} c_i z_i^2 + z_{n-1}z_n + \left(\sum_{i=2}^{n-1} z_i\frac{\partial\hat{h}^{(i-1)}}{\partial\hat{\theta}} + \sum_{i=3}^{n-1} z_i\frac{\partial\alpha_{i-1}}{\partial\hat{\theta}}\right)\left(\dot{\hat{\theta}} - \tau_{n-1}\right)\\ &\quad +\sigma\Bigg[\hat{h}^{(n)} - y_r^{(n)}(t) + \alpha_n + \left(\frac{\partial\hat{h}^{(n-1)}}{\partial\hat{\theta}} + \frac{\partial\alpha_{n-1}}{\partial\hat{\theta}}\right)\left(\dot{\hat{\theta}} - \tau_n\right)\\ &\quad\quad +\sum_{i=1}^{n-1} k_i\left(-z_{i-1} - c_i z_i + z_{i+1} - \left(\sum_{j=2}^{i-1} z_j\frac{\partial\hat{h}^{(j-1)}}{\partial\hat{\theta}} + \sum_{j=3}^{i-1} z_j\frac{\partial\alpha_{j-1}}{\partial\hat{\theta}}\right)\Gamma\omega_i^T\right)\\ &\quad\quad -\sum_{i=1}^{n-1} k_i\left(\frac{\partial\hat{h}^{(i-1)}}{\partial\hat{\theta}} + \frac{\partial\alpha_{i-1}}{\partial\hat{\theta}}\right)\left(\dot{\hat{\theta}} - \tau_i\right)\Bigg]\\ &\quad +(\theta-\hat{\theta})^T\Gamma^{-1}\left(-\dot{\hat{\theta}} + \tau_{n-1} + \Gamma\sigma\left(\omega_n^T + \sum_{i=1}^{n-1} k_i\omega_i^T\right)\right)\end{aligned} \tag{7.26}$$

We can eliminate $(\theta - \hat{\theta})$ from $\dot{V}_n$ by choosing the update law

$$\dot{\hat{\theta}} = \tau_n = \tau_{n-1} + \Gamma\sigma\left(\omega_n^T + \sum_{i=1}^{n-1} k_i\omega_i^T\right). \tag{7.27}$$

Now, noting that

$$\dot{\hat{\theta}} - \tau_{n-1} = \tau_n - \tau_{n-1} = \Gamma\sigma\left(\omega_n^T + \sum_{i=1}^{n-1} k_i\omega_i^T\right) \tag{7.28}$$

$\dot{V}_n$ can be rewritten as

$$\begin{aligned}\dot{V}_n &= -\sum_{i=1}^{n-1} c_i z_i^2 + z_{n-1} z_n \\ &+\sigma\Bigg[\hat{h}^{(n)} - y_r^{(n)}(t) + \alpha_n + \left(\sum_{i=2}^{n-1} z_i \frac{\partial \hat{h}^{(i-1)}}{\partial \hat{\theta}} + \sum_{i=3}^{n-1} z_i \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}}\right)\Gamma\left(\omega_n^T + \sum_{i=1}^{n-1} k_i \omega_i^T\right) \\ &\quad + \sum_{i=1}^{n-1} k_i \left(-z_{i-1} - c_i z_i + z_{i+1} - \left(\sum_{j=2}^{i-1} z_j \frac{\partial \hat{h}^{(j-1)}}{\partial \hat{\theta}} + \sum_{j=3}^{i-1} z_j \frac{\partial \alpha_{j-1}}{\partial \hat{\theta}}\right)\Gamma \omega_i^T\right) \\ &\quad - \sum_{i=1}^{n-1} k_i \left(\frac{\partial \hat{h}^{(i-1)}}{\partial \hat{\theta}} + \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}}\right)(\tau_n - \tau_i)\Bigg] \end{aligned} \tag{7.29}$$

Finally, to achieve

$$\dot{V}_n = -\sum_{i=1}^{n-1} c_i z_i^2 + z_{n-1} z_n - \lambda \sigma^2 - \lambda \beta \,|\sigma| \tag{7.30}$$

the bracketed term multiplying $\sigma$ should be $-\lambda(\sigma+\beta\ \mathrm{sgn}(\sigma))$, where $\lambda$ and $\beta$ are positive design parameters and sgn is the *signum* function, namely

$$\begin{aligned}&\hat{h}^{(n)} - y_r^{(n)}(t) + \alpha_n + \left(\sum_{i=2}^{n-1} z_i \frac{\partial \hat{h}^{(i-1)}}{\partial \hat{\theta}} + \sum_{i=3}^{n-1} z_i \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}}\right)\Gamma\left(\omega_n^T + \sum_{i=1}^{n-1} k_i \omega_i^T\right) \\ &+ \sum_{i=1}^{n-1} k_i \left(-z_{i-1} - c_i z_i + z_{i+1} - \left(\sum_{j=2}^{i-1} z_j \frac{\partial \hat{h}^{(j-1)}}{\partial \hat{\theta}} + \sum_{j=3}^{i-1} z_j \frac{\partial \alpha_{j-1}}{\partial \hat{\theta}}\right)\Gamma \omega_i^T\right) \\ &- \sum_{i=1}^{n-1} k_i \left(\frac{\partial \hat{h}^{(i-1)}}{\partial \hat{\theta}} + \frac{\partial \alpha_{i-1}}{\partial \hat{\theta}}\right)(\tau_n - \tau_i) = -\lambda\Big(\sigma + \beta\ \mathrm{sgn}(\sigma)\Big) \end{aligned} \tag{7.31}$$

The update law (7.27) together with the dynamical discontinuous adaptive feedback law (7.31) achieve a sliding mode on the sliding surface (7.23). Note that (7.30) can be rewritten as

$$\dot{V}_n = -z^T Q z - \lambda \beta |\sigma| \tag{7.32}$$

with Q being the same as (6.47). Thus, the sufficient condition (6.49) also applies in this case to guarantee that $Q$ is positive definite. However, since the right hand side of the differential equations which characterize the closed-loop system are discontinuous, we cannot use standard theorems for existence of solutions. Instead, the use of the recently introduced integral invariance principle for differential inclusions [96] provides a convenient setting for the construction of a stability proof.

Note that the discontinuous feedback control law (7.31) can be rewritten in the form of the dynamical adaptive sliding mode control law (7.20) by replacing the control input $u$ and its derivatives $\dot{u}, \ddot{u}, \ldots$ by the state variables $v_1, v_2, v_3, \ldots$ respectively and solving for $\dot{v}_{n-\rho}$.

An important advantage arises from the dynamical adaptive sliding mode control: *the output tracking error function $z_1(t)$ asymptotically approaches zero with substantially reduced chattering* [92, 94]. A slightly modified version of the above algorithm has been used by Ahmed-Ali and Lamnabhi-Lagarrigue [2] for tracking problems in some examples of practical interest.

## 7.4 Example: Field-Controlled DC Motor

Consider the following nonlinear dynamical model of a field-controlled DC motor [106]

$$\begin{aligned} \dot{x}_1 &= -\frac{R_a}{L_a}x_1 - \frac{K}{L_a}x_2u + \frac{V_a}{L_a} \\ \dot{x}_2 &= -\frac{B}{J}x_2 + \frac{K}{J}x_1u \\ y &= x_2 \end{aligned} \tag{7.33}$$

where $x_1$ represents the armature circuit current and $x_2$ is the angular velocity of the rotating axis. $V_a$ is a fixed voltage applied to the armature circuit and $u$ is the field winding input voltage, acting as the control input. The constants $R_a, L_a$, and $K$ represent the resistance, the inductance in the armature circuit and the constant torque, respectively. The parameters $J$ and $B$ are the moment of inertia and the associated viscous damping coefficient of the load. The control objective is to track a known reference trajectory $y_r(t)$ with bounded derivatives.

We assume that all the parameters are unknown and rewrite (7.33) as

$$\begin{aligned} \dot{x}_1 &= -\theta_1x_1 - \theta_2x_2u + \theta_3 \\ \dot{x}_2 &= -\theta_4x_2 + \theta_5x_1u \\ y &= x_2 \end{aligned} \tag{7.34}$$

with

$$\theta_1 = \frac{R_a}{L_a}, \quad \theta_2 = \frac{K}{L_a}, \quad \theta_3 = \frac{V_a}{L_a}, \quad \theta_4 = \frac{B}{J}, \quad \theta_5 = \frac{K}{J} \tag{7.35}$$

It was shown in [106] that (7.33) is locally minimum-phase in the region characterized by the conditions

$$R_aB > K^2U^2, \qquad V_a^2 \geqslant 4R_aBX_2^2(U) \tag{7.36}$$

where $U$ is a constant equilibrium input voltage and $X_2(U)$ is the corresponding equilibrium of the angular velocity $x_2$. Therefore we assume that the nominal unknown values of the system parameters satisfy the conditions (7.36) and apply the algorithm described above to design an adaptive dynamical sliding mode controller for tracking the desired trajectory. Note that the relative degree $\rho$ is 1 and therefore the time derivative of the control input will appear at the second step of the design algorithm.

**Step 1.** By defining the output variable error $z_1 = x_2 - y_r(t)$, the time derivative of $z_1$ is given by

$$\dot{z}_1 = \omega_1\hat{\theta} + \omega_1(\theta - \hat{\theta}) - \dot{y}_r(t) \tag{7.37}$$

with

$$\omega_1 = \begin{bmatrix} 0 & 0 & 0 & -x_2 & x_1 u \end{bmatrix}. \tag{7.38}$$

Consider the Lyapunov function

$$V_1 = \frac{1}{2}z_1^2 + \frac{1}{2}(\theta - \hat{\theta})^T\Gamma^{-1}(\theta - \hat{\theta}) \tag{7.39}$$

whose time derivative is

$$\dot{V}_1 = z_1\left[\omega_1\hat{\theta} - \dot{y}_r(t)\right] + (\theta - \hat{\theta})^T\Gamma^{-1}\left(-\dot{\hat{\theta}} + \Gamma\omega_1^T z_1\right) \tag{7.40}$$

We can achieve $\dot{V}_1 = -c_1 z_1^2$ with the update law

$$\dot{\hat{\theta}} = \tau_1 = \Gamma\omega_1^T z_1 \tag{7.41}$$

if the expression

$$\omega_1\hat{\theta} - \dot{y}_r(t) = -c_1 z_1 \tag{7.42}$$

is satisfied. However, since (7.42) is not valid and $\tau_1$ is not considered an update law but the first tuning function, we define the second error variable as

$$z_2 = \omega_1\hat{\theta} - \dot{y}_r(t) + c_1 z_1 \tag{7.43}$$

obtaining the following closed-loop form for $\dot{z}_1$

$$\dot{z}_1 = -c_1 z_1 + z_2 + \omega_1(\theta - \hat{\theta}) \tag{7.44}$$

and

$$\dot{V}_1 = -c_1 z_1^2 + z_1 z_2 + (\theta - \hat{\theta})^T\Gamma^{-1}\left(-\dot{\hat{\theta}} + \tau_1\right). \tag{7.45}$$

**Step 2.** At this final step we define the sliding surface

$$\sigma = k_1 z_1 + z_2 = 0 \tag{7.46}$$

and the augmented Lyapunov function

$$V_2 = V_1 + \frac{1}{2}\sigma^2. \tag{7.47}$$

The time derivative of $V_2$ is given by

$$\begin{aligned}\dot{V}_2 &= -c_1 z_1^2 + z_1 z_2 + (\theta - \hat{\theta})^T \Gamma^{-1}\Big( -\dot{\hat{\theta}} + \tau_1 + \Gamma\sigma(\omega_2^T + k_1\omega_1^T)\Big) \\ &\quad +\sigma\Big[\omega_2\hat{\theta} + \omega_1\dot{\hat{\theta}} + \hat{\theta}_5 x_1 \dot{u} - \ddot{y}_r(t) - c_1\dot{y}_r(t) + k_1(-c_1 z_1 + z_2)\Big]\end{aligned} \tag{7.48}$$

with

$$\omega_2 = \Big[ -\hat{\theta}_5 x_1 u \quad -\hat{\theta}_5 x_2 u^2 \quad \hat{\theta}_5 u \quad -(c_1 - \hat{\theta}_4)x_2 \quad (c_1 - \hat{\theta}_4)x_1 u\Big] \tag{7.49}$$

To eliminate $(\theta - \hat{\theta})$ from $\dot{V}_2$ we choose the update law

$$\dot{\hat{\theta}} = \tau_2 = \tau_1 + \Gamma\sigma(\omega_2^T + k_1\omega_1^T) = \Gamma\Big[\omega_1^T z_1 + \sigma(\omega_2^T + k_1\omega_1^T)\Big] \tag{7.50}$$

The control function $u$ can be readily obtained as the solution of the following nonlinear time-varying differential equation

$$\dot{u} = \frac{1}{\hat{\theta}_5 x_1}\Big[ -\omega_2\hat{\theta} - \omega_1\tau_2 + \ddot{y}_r(t) + c_1\dot{y}_r(t) - k_1(-c_1 z_1 + z_2) - \lambda\Big(\sigma + \beta\,\mathrm{sgn}(\sigma)\Big)\Big] \tag{7.51}$$

and then

$$\dot{V}_2 = -c_1 z_1^2 + z_1 z_2 - \lambda\sigma^2 - \lambda\beta|\sigma|. \tag{7.52}$$

Note that the term $\hat{\theta}_5 x_1$ in the denominator of (7.51) characterizes the local nature of the controller obtained for this example. Thus, this control is applicable only in a region for which the condition $\hat{\theta}_5 x_1 \neq 0$ is satisfied. Moreover, the control-dependent transformation $z = \Phi(x, \hat{\theta}, u, t)$ defined by

$$z = \Phi(x,\hat{\theta},u,t) = \begin{pmatrix} x_2 - y_r(t) \\ -x_2\hat{\theta}_4 + x_1 u\hat{\theta}_5 + c_1(x_2 - y_r(t)) - \dot{y}_r(t)\end{pmatrix} \tag{7.53}$$

has the associated Jacobian matrix

$$\frac{\partial\Phi(x,\hat{\theta},u,t)}{\partial x} = \begin{bmatrix} 0 & 1 \\ u\hat{\theta}_5 & c_1 - \hat{\theta}_4\end{bmatrix} \tag{7.54}$$

Thus $u = 0$ corresponds to a singularity of the transformation (7.53) and, hence, stabilization or tracking tasks that imply polarity reversals in the field winding input voltage must be handled using a different technique. We consider here tracking tasks which guarantees non-singularities of the transformation (7.53).

From (6.49) the sufficient condition on the design parameters which guarantees asymptotic tracking behaviour is

$$\lambda(c_1 + k_1) > \frac{1}{4}. \tag{7.55}$$

In order to test the robustness of the proposed scheme with respect to external perturbation inputs, we simulated a perturbed model including a bounded external stochastic perturbation input $\nu$ applied to the voltage of the armature circuit $V_a$

$$\begin{aligned} \dot{x}_1 &= -\theta_1 x_1 - \theta_2 x_2 u + \theta_3 + \nu \\ \dot{x}_2 &= -\theta_4 x_2 + \theta_5 x_1 u \\ y &= x_2 \end{aligned} \tag{7.56}$$

Simulations of a tracking task were performed for a DC motor with the following parameter values

$$R_a = 7\ \Omega \quad ; \quad L_a = 120 \text{ mH} \quad ; \quad V_a = 5 \text{ V}$$

$$K = 1.41 \times 10^{-2} \text{ N-m/A}, \quad B = 6.04 \times 10^{-6} \text{ N-m-s/rad}, \quad J = 1.06 \times 10^{-6} \text{ N-m-s}^2\text{/rad}$$

and the design parameters were

$$\lambda = 1, \quad \beta = 1, \quad k_1 = 1, \quad c_1 = 100, \quad \Gamma = I_5$$

The desired output reference trajectory $y_r(t)$ was considered to allow a smooth transition of the angular velocity $x_2$ between two equilibrium values $X_2 = 300$ and $X_2^* = 200$

$$y_r(t) = \begin{cases} X_2 & \text{for} \quad 0 \leqslant t < t_0 \\ X_2^* + (X_2 - X_2^*)\exp(-2(t - t_0)^2) & \text{for} \quad t \geqslant t_0 \end{cases} \tag{7.57}$$

It may be verified that, according to the nominal values of the parameters, the initial and final angular velocities are located in the minimum phase region of the system. Figure 7.1 shows the satisfactory robust asymptotic tracking achieved by the combined DAB-SMC approach. Computer simulations were also carried out for the controlled system in the absence of perturbations. Figure 7.2 shows the asymptotic tracking behaviour exhibited in the absence of perturbations. This example illustrate the satisfactory performance exhibited by the controller obtained by the DAB-SMC algorithm. The tracking of the desired reference output is achieved with almost full insensitivity to the disturbance, and considerably reduced chattering.

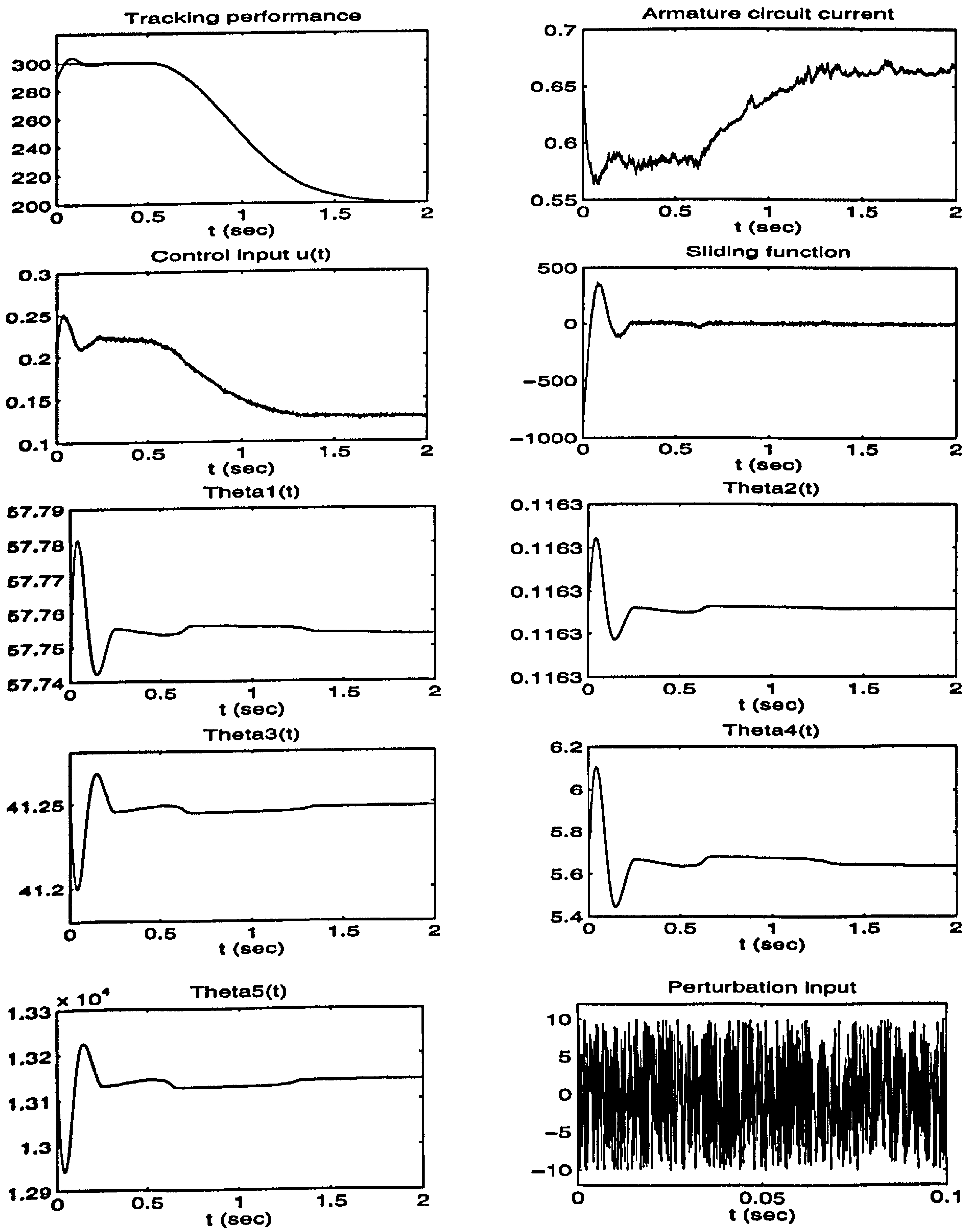


Figure 7.1: Tracking performance, state trajectories of the perturbed system, evolution of the parameter estimate and the sliding surface function and perturbation signal

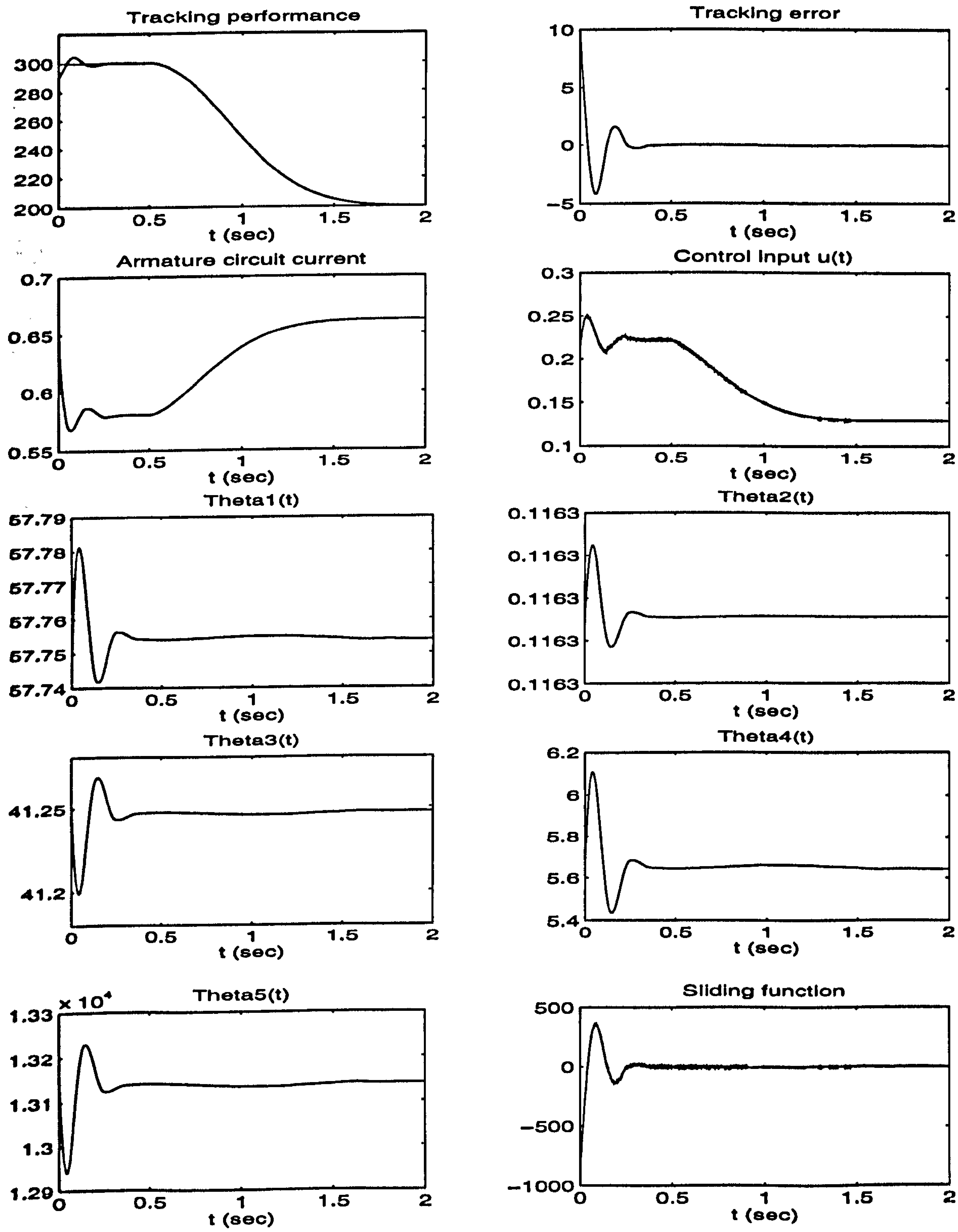


Figure 7.2: Tracking performance, controlled responses of the system and evolution of the parameter estimate and the sliding surface function in the absence of perturbations

# Chapter 8

# Backstepping Design via Symbolic Algebra Computation

## 8.1 Introduction

The computer technology advances of the last fifteen years have allowed the development of a number of computer software systems intended for numerical and symbolic computations. Thus, computer packages such as MACSYMA, MAPLE and MATHEMATICA provide in general the following capabilities

- numerical and symbolic algebra computation
- data analysis and graphical visualization
- high-level programming

The availability of these packages has allowed the development of useful toolboxes for implementing systematic schemes for the analysis and design of feedback control systems. For instance, some of the toolboxes so far developed include the design of nonlinear observers [6], analysis and control design for affine and non-affine systems [34, 23], SMC design via zero dynamics [67], modelling and nonlinear control design [5], analysis of nonlinear systems via algebraic system theory [1], and analysis and design based on flatness [95].

These tools simplify the development of systematic and recursive control design methods which can be implemented by using these computer programs, so that the design of stabilizing controllers may be carried out more efficiently. The various backstepping control design algorithms given in the previous chapters provide a systematic framework for the design of tracking and regulation strategies suitable for large classes of both deterministic systems and uncertain systems. A very appealing aspect of these backstepping design algorithms is that they follow a systematic step-by-step algorithm. However, the equations arising from these design algorithms are usually very complicated for hand computation even for low order ($n \leqslant 3$) systems. This justifies the need for a symbolic computation toolbox which implements the various backstepping design algorithms and computes the required controller without the possible errors of hand computation.

The SAB algorithm with tuning functions for PPF and PSF systems developed by Krstić *et al* [61] has been implemented using Mathematica [5]. In this chapter we describe our new symbolic toolbox which implements the new DAB algorithm of Section 4.2 and the combined DAB-SMC of Section 7.3, which have been developed by Rios-Bolívar *et al* [87, 92]. The implementation of these algorithms allows the design of both static and dynamic adaptive tracking controllers following the basic ideas of backstepping with tuning functions, without the need for transformation into the above-mentioned restricted feedback forms. These algorithms have been implemented using the MATLAB Symbolic Toolbox [89, 90]. MATLAB has been chosen for the implementation of this toolbox due to the availability of efficient tools for numerical integration and other toolboxes for computer simulations of control systems. The MATLAB Symbolic Algebra Toolbox incorporates much of the MAPLE system.

We start by describing the algorithms and the various classes of nonlinear systems for which they are applicable, as well as the features of the new symbolic toolbox. Then the use of this toolbox is explained in a tutorial manner by using three examples corresponding to three different classes of systems.

## 8.2 Symbolic Toolbox for Backstepping Design

In this section we describe the symbolic toolbox developed via MATLAB, which implements the DAB algorithm given in Section 4.2 and the combined DAB-SMC algorithm of Section 7.3. The backstepping control design algorithms described in the previous chapters allows the design of various types of static and dynamic controllers. These controllers are applicable to several classes of nonlinear systems, as shown in Table 8.1.

| *Design algorithm* | *Type of controller* | *Class of system* |
|---|---|---|
| SDB | Static deterministic | Triangular form |
| SAB | Static adaptive | PSF and PPF |
| DDB | Dynamic deterministic | Observable minimum phase (I/O linearizable) |
| DAB | Dynamic adaptive | Uncertain observable minimum phase |
| SAB-SMC | Static adaptive sliding mode | PSF and PPF |
| DAB-SMC | Dynamic adaptive sliding mode | Uncertain observable minimum phase |

Table 8.1: Backstepping control design algorithms

It is worthwhile stressing that the DDB and DAB algorithms incorporate the SDB and SAB algorithms as particular cases, respectively. Also the algorithms developed for deterministic systems (identified with D in the second letter of the acronyms) can be seen as particular cases of their correspondent adaptive version; those in which the nonlinear functions multiplying the uncertain parameter vector $\theta$ are zero. Thus implementation of all the algorithms in Table 8.1 is achieved by implementing only two algorithms: the DAB and DAB-SMC algorithms.

The MATLAB Symbolic Toolbox contains a collection of tools (functions) that can be used for manipulating and solving symbolic expressions. There are tools to combine, simplify, differentiate, integrate, and solve algebraic and differential equations. Other tools allow the derivation of exact results in linear algebraic matrix operations such as inverses, determinants, canonical forms and eigenvalues of symbolic matrices without the errors introduced by numerical computations. The tools in the MATLAB Symbolic Toolbox are built upon the MAPLE computer package. Thus, the symbolic operations in MATLAB are actually performed by MAPLE; then the results are transferred to the MATLAB environment.

Our backstepping-SMC design toolbox (BACKDSMC) has been developed for the synthesis of tracking and regulating adaptive (and non-adaptive) controllers for a large class of observable minimum-phase nonlinear systems, requiring a minimum of effort by the user. It has the following features [89, 90]

- automatises the backstepping control design process
- does not use transformations into canonical forms

- allows the design of a number of adaptive and robust adaptive-SMC controllers
- does not require the user to have expert knowledge of the backstepping design technique
- automatically generates MATLAB code programs for computer simulation of the closed-loop systems.

The types of controllers designed by BACKDSMC for regulation and tracking problems include

- static and dynamical non-adaptive linearizing controllers for deterministic systems (SDB and DDB algorithms)
- static and dynamical adaptive backstepping controllers for uncertain systems (SAB and DAB algorithms)
- robust static and dynamical combined backstepping-SMC for uncertain systems (combined SAB-SMC and DAB-SMC algorithms)

Our three examples in Section 8.3 illustrate the design of the types of controllers above. The outputs generated by BACKDSMC consist of the *feedback control law*, the *coordinate transformation* placing the system into the error coordinates, the *parameter update law* for uncertain systems, the *sliding surface* for the combined backstepping-SMC design, and the MATLAB *code programs* for simulation (see Figure 8.1).

The user needs to provide the nonlinear functions of the mathematical model of the system written in the general form

$$\begin{aligned} \dot{x} &= f_0(x) + \Psi(x)\theta + \Big(g_0(x) + \varphi(x)\theta\Big)u \\ y &= h(x) \end{aligned} \tag{8.1}$$

where $x \in \Re^n$ is the state; $u, y \in \Re$ the input and output respectively and $\theta = [\theta_1, \ldots, \theta_p]^T$ a vector of unknown parameters. $f_0$, $g_0$ and the columns of the matrices $\Psi, \varphi \in \Re^{n \times p}$ are known smooth vector fields and $h$ is a smooth scalar function. In addition, the symbolic desired output should also be supplied by the user. Depending on the nature of $y_r$ the problem to be solved is either regulation or tracking. Thus, when $y_r$ is constant the designed controller is for regulation, otherwise $y_r$ is a time-dependent function and the controller is designed for tracking tasks.

To define symbolic expressions in MATLAB the command SYM is used. Symbolic expressions in the MATLAB Symbolic Toolbox are character strings or arrays of character strings that represent numbers, functions, operators and variables. The variables

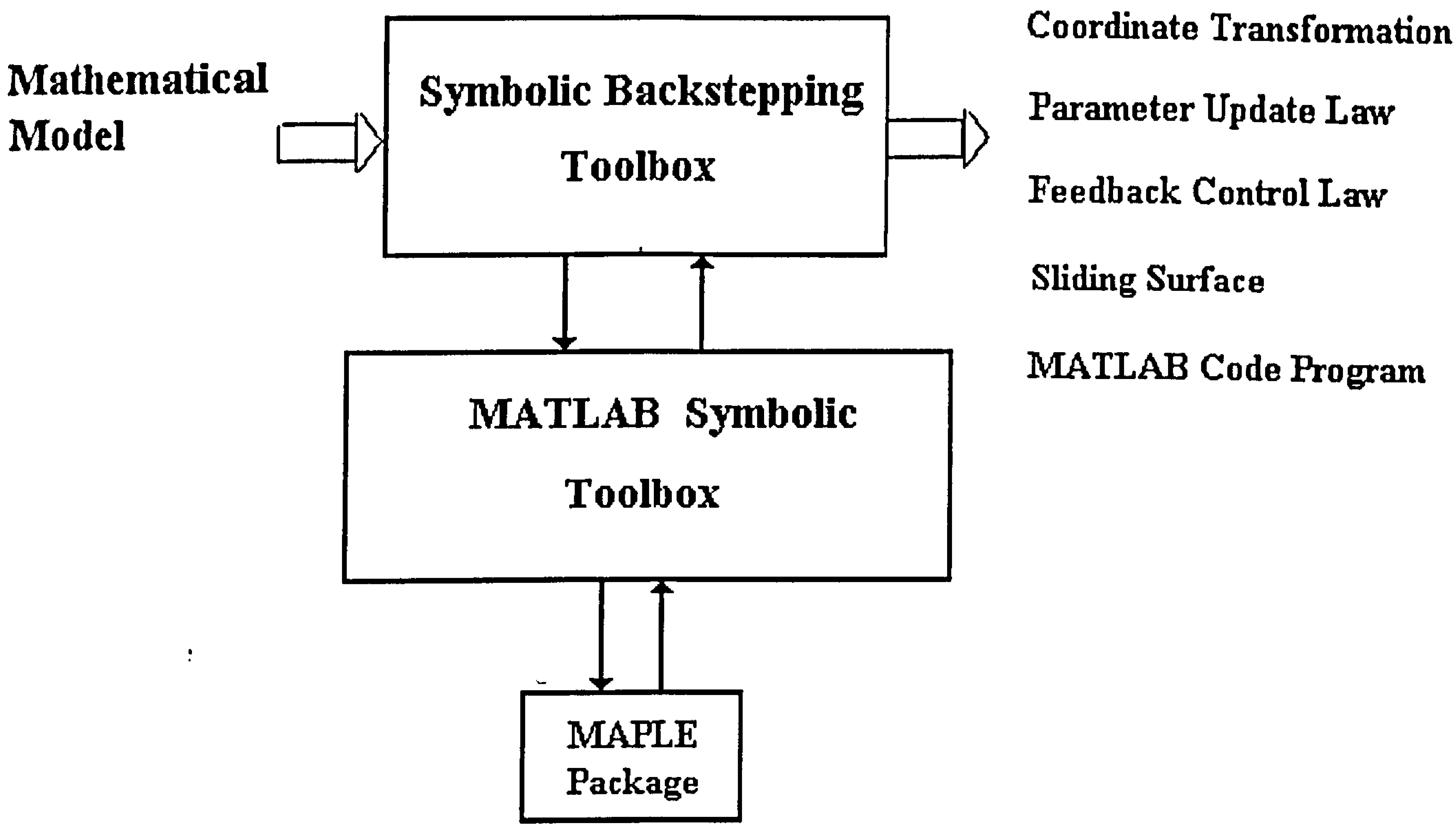


Figure 8.1: Symbolic backstepping-SMC control design toolbox.

are not required to have predefined values. The following MATLAB commands allows one to define symbolic expressions. For instance,

```
h='sin(x1)'      % scalar function
```

defines the symbolic scalar function $h = \sin(x_1)$ depending on the symbolic variable $x1$. A vector field of the form

$$f(x) = \begin{pmatrix} x_1^2 \\ x_3 \\ 0 \end{pmatrix}$$

is defined by the following MATLAB command

```
f=sym('[x1^2;x3;0]')      % vector field
```

whereas a symbolic matrix of the form

$$\Psi(x) = \begin{bmatrix} x_1 x_2^2 & x_2 \\ \cos(2x_1) & \dfrac{x_2}{1+x_1} \end{bmatrix}$$

is defined by

```
Psi=sym('[x1*x2^2,x2;cos(2*x1),x2/(1+x1)]')      % symbolic matrix
```

To identify the state variables, BACKDSMC needs them be specified by placing a number immediately after (without spaces) the symbol 'x'. So the first state variable should be 'x1', the second one 'x2', and so on. Character strings which are not identified as state variables, symbolic functions or arithmetic operators are treated by BACKDSMC as constants. A detailed explanation regarding the creation and manipulation of symbolic expressions is available in the MATLAB User's Guide [71].

Once the symbolic expressions characterizing the system model (8.1) have been created, BACKDSMC can be called in any of the forms below, depending on the design requirements of the user.

[C,TAU]=BACKDSMC(f,g,h,yd,phi,psi)
designs the adaptive backstepping controller for the uncertain nonlinear system characterized by f,g,h,phi and psi for regulation or tracking to the desired output yd (SDB, DDB, SAB and DAB algorithms).

[C,TAU,Z]=BACKDSMC(f,g,h,yd,phi,psi)
gives, additionally, the transformation Z placing the original system into error coordinates.

[C,TAU,Z]=BACKDSMC(f,g,h,yd,phi,psi,'modfile')
generates the m-file 'modfile.m' containing the system model and the adaptive controller for numerical simulation purposes.

[C,TAU,Z]=BACKDSMC(f,g,h,yd,phi,psi,'modfile','runfile')
generates, additionally, the m-file 'runfile.m', which runs 'modfile.m' and provides nominal values of the unknown parameters, design parameter values and initial states for simulation purposes.

[C,TAU,Z,SIGMA]=BACKDSMC(f,g,h,yd,phi,psi,'modfile','runfile')
i.e. containing a fourth output argument, allows one to design an adaptive sliding mode control (SMC) to generate a SLIDING MODE on the sliding surface SIGMA=$k_1 z_1 + k_2 z_2 + \ldots + k_{n-1} z_{n-1} + z_n = 0$, with $k_i$'s the design parameters and $z_i$'s the error coordinates of the transformation Z.

This information is obtained in the MATLAB command window by typing

help backdsmc

The full MATLAB code program of the toolbox BACKDSMC is included in Appendix B. In the next section, various examples are given to illustrate the use of the BACKDSMC toolbox.

## 8.3 Use of the Symbolic Toolbox BACKDSMC

We present in this section three different examples to illustrate the use of the symbolic toolbox BACKDSMC corresponding to various classes of systems. These are given in a tutorial manner, including the series of MATLAB commands required to define the symbolic equations which characterize the dynamical system and the arguments needed by BACKDSMC. The results are presented in the form of m-files (MATLAB code programs), which can be used directly for numerical integration.

### 8.3.1 Control Design for Deterministic Nontriangular Systems

**Example 8.1** Consider the third order system without uncertainties

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{pmatrix} = \begin{pmatrix} x_2 + x_1 x_3 \\ x_3 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} u \tag{8.2}$$

which is *not* in *triangular* form. As stated in Section 3.4, this system belongs to the class of input-output linearizable systems. The linearizing output is

$$y = x_1 \exp(-x_2) \tag{8.3}$$

The DDB algorithm was applied in Section 3.4 to obtain the following nonlinear transformation

$$\begin{aligned} z_1 &= h_0(x) = x_1 \exp(-x_2) - y_r \\ z_2 &= h_1(x) = x_2 \exp(-x_2) + c_1\Big(x_1 \exp(-x_2) - y_r\Big) \\ z_3 &= h_2(x, v_1) = x_1 \exp(-x_2) + x_3(1 - x_2)\exp(-x_2) + c_1 x_2 \exp(-x_2) \\ &\qquad + c_2\Big[x_2 \exp(-x_2) + c_1\Big(x_1 \exp(-x_2) - y_r\Big)\Big] \end{aligned} \tag{8.4}$$

and the *static* feedback control

$$u = \left(\frac{\partial h_2}{\partial x_3}\right)^{-1} \left[-z_2 - \frac{\partial h_2}{\partial x_1}(x_2 + x_1 x_3) - \frac{\partial h_2}{\partial x_2} x_3 - c_3 z_3\right] \tag{8.5}$$

with

$$\begin{aligned} \frac{\partial h_2}{\partial x_1} &= (1 + c_1 c_2)\exp(-x_2) \\ \frac{\partial h_2}{\partial x_2} &= \exp(-x_2)\Big(-x_1 - x_3(2 - x_2) + c_1(1 - x_2) + c_2(1 - x_2 - c_1 x_1)\Big) \\ \frac{\partial h_2}{\partial x_3} &= \exp(-x_2)(1 - x_2) \end{aligned} \tag{8.6}$$

We apply BACKDSMC to obtain the linearizing static control (8.5). The symbolic expressions which characterize the mathematical model (8.2) are defined by the following series of MATLAB commands

```
f=sym('[x2+x1*x3;x3;0]')        % f(x)
g=sym('[0;0;1]')                % g(x)
phi=sym('[0;0;0]')              % phi(x)
psi=sym('[0;0;0]')              % psi(x)
h='x1*exp(-x2)'                 % h(x)
yd='0'                          % desired output yd
```

The above static backstepping controller is obtained by specifying the command line

```
[c,tau,z]=backdsmc(f,g,h,yd,phi,psi)
```

To obtain the m-files required for computer simulations two optional arguments may be added to the command line

$$[\mathrm{c},\mathrm{tau},\mathrm{z}] = \mathrm{backdsmc}(\mathrm{f},\mathrm{g},\mathrm{h},\mathrm{yd},\mathrm{phi},\mathrm{psi},'\mathrm{notrian}','\mathrm{notrianr}') \tag{8.7}$$

The m-file 'notrian.m', generated by the command (8.7), contains the system equations and the control law as shown below:

```
function xdot=notrian(t,x);
global c;
%
%    Control law
%
u1=(-c(2)*c(1)/exp(x(2))*x(2)-2/exp(x(2))*x(2)+x(3)*c(1)/exp(x(2))*...
x(2)-x(3)*c(1)/exp(x(2))-x(3)^2/exp(x(2))*x(2)+2*x(3)^2/exp(x(2))+...
x(3)*c(2)/exp(x(2))*x(2)-x(3)*c(2)/exp(x(2))-c(3)*c(1)/exp(x(2))*x(2)...
+c(3)*x(3)/exp(x(2))*x(2)-c(3)*x(3)/exp(x(2))-c(3)*c(2)/exp(x(2))*...
x(2)-c(3)*c(2)*c(1)*x(1)/exp(x(2))-c(3)*x(1)/exp(x(2))-c(1)*x(1)/...
exp(x(2)))/(-exp(-x(2))*x(2)+exp(-x(2)));
%
%    System equations
%
xdot(1)=x(2)+x(1)*x(3);
xdot(2)=x(3);
xdot(3)=u1;
```

The m-file 'notrianr.m' is generated to run the numerical integrations of the closed-loop system in 'notrian.m' and allows one to specify the initial conditions, the parameter design values and the initial and final times for simulations, as follows:

```
%
% File notrianr.m
%
% This program runs notrian.m
%
% Right hand side values to be input by user
%
global c;
%
%   Parameter values
%
c=[;;];
t0=;        % Initial time
tf=;        % Final time
%
%    Initial conditions
%
x0=[,,];
[t,x]=ode23('notrian',t0,tf,x0);
```

When the system contains uncertain parameters the nominal "unknown" values are also given in 'notrianr.m'. As shown in Section 3.4 the above controller linearizes the system dynamics and the closed-loop system has a skew-symmetric form. The computer simulations shown in Section 3.4 were obtained by using the m-files 'notrian.m' and 'notrianr.m'. BACKDSMC automatically selects the standard Runge-Kutta procedure *ode23* as ordinary differential equations (ODE) solver. Nevertheless, user may choose another ODE solver from the various ODE solvers available in the recently released *ode-suite* for MATLAB and SIMULINK. This suite contains five different routines, including two routines for *stiff* problems. In fact, we used the routine *ode15s* from this suite, for the numerical integration of sliding mode controlled systems in this thesis.

### 8.3.2 Adaptive Control Design for PSF Systems

**Example 8.2** Consider the flexible-joint manipulator model of Section 2.5 with unknown link mass and transformed into the PSF form

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= x_3 + \varphi(x_1)\theta \\ \dot{x}_3 &= x_4 \\ \dot{x}_4 &= a_0 x_3 + a_1\varphi(x_1)\theta + bu \\ y &= x_1 \end{aligned} \tag{8.8}$$

where the nonlinear function $\varphi$ and the constant parameters $a_0$, $a_1$ and $b$ are all known and defined by

$$a_1 = \frac{k}{j_l} \tag{8.9}$$

$$a_0 = -\left(a_1 + \frac{k}{j_m}\right) \tag{8.10}$$

$$b = \frac{a_1}{j_m} \tag{8.11}$$

$$\varphi(x_1) = -\frac{gl}{j_l}\sin(x_1) \tag{8.12}$$

This example allows one to illustrate the performance of BACKDSMC for the design of adaptive backstepping controllers for *uncertain systems in PSF form*, namely the implementation of the SAB algorithm described in Section 2.4.3. We will show that applying BACKDSMC we obtain both the update control law (2.292) and the static adaptive feedback law (2.293) of Section 2.5, which adaptively stabilizes the output to the desired link angular position $y_d = 1.22$ radians. The symbolic expressions which characterize the mathematical model (8.8) are defined by the following sequence of MATLAB commands:

```
f=sym('[x2;x3;x4;a0*x3]')                              % f(x)
g=sym('[0;0;0;b]')                                     % g(x)
phi=sym('[0;-g*l*sin(x1)/jl;0;-a1*g*l*sin(x1)/jl]')    % phi(x)
psi=sym('[0;0;0;0]')                                   % psi(x)
h='x1'                                                 % h(x)
yd='1.22'                                              % desired output yd
```

The following MATLAB command runs BACKDSMC for the design of the static adaptive backstepping controller and creates the m-file 'robotmod.m'

```
[c,tau,z]=backdsmc(f,g,h,yd,phi,psi,'robotmod')
```

The generated m-file 'robotmod.m' contains the system equations, the update law for the unknown parameter and the adaptive control law as follows:

```
function xdot=robotmod(t,x);
global c ad theta g l jl a0 a1 b;
%
%  Auxiliary variables
%
th1=x(5);
%
%  Update law
%
tau1=-(x(2)+c(1)*(x(1)-1.22))*ad(1)*g*l*sin(x(1))/jl-((c(1)*x(2)*jl+...
x(3)*jl-th1*g*l*sin(x(1)))/jl+c(2)*(x(2)+c(1)*(x(1)-1.22))+x(1)-1.22)...
*ad(1)*g*l*sin(x(1))/jl*(c(1)+c(2))-(-(x(2)*th1*g*l*cos(x(1))-x(2)*...
c(2)*c(1)*jl-x(2)*jl-c(1)*x(3)*jl+c(1)*th1*g*l*sin(x(1))-c(2)*x(3)*...
jl+c(2)*th1*g*l*sin(x(1))-x(4)*jl)/jl-g*l*sin(x(1))/jl*(-(x(2)+c(1)*...
(x(1)-1.22))*ad(1)*g*l*sin(x(1))/jl-((c(1)*x(2)*jl+x(3)*jl-th1*g*l*...
sin(x(1)))/jl+c(2)*(x(2)+c(1)*(x(1)-1.22))+x(1)-1.22)*ad(1)*g*l*...
sin(x(1))/jl*(c(1)+c(2)))+c(3)*((c(1)*x(2)*jl+x(3)*jl-th1*g*l*...
sin(x(1)))/jl+c(2)*(x(2)+c(1)*(x(1)-1.22))+x(1)-1.22)+x(2)+c(1)*...
(x(1)-1.22))*ad(1)*g*l*sin(x(1))*(-jl*th1*g*l*cos(x(1))+c(2)*c(1)*...
jl^2+2*jl^2+g^2*l^2*sin(x(1))^2*ad(1)+g^2*l^2*sin(x(1))^2*ad(1)*...
c(1)^2+2*g^2*l^2*sin(x(1))^2*ad(1)*c(2)*c(1)+g^2*l^2*sin(x(1))^2*...
ad(1)*c(2)^2+c(3)*jl^2*c(1)+c(3)*jl^2*c(2)+a1*jl^2)/jl^3;
%
%   Control law
%
u1=(-(2*jl^3*x(3)+a0*x(3)*jl^3+x(2)*c(3)*jl^3+x(2)*c(1)*...
jl^3+x(4)*jl^3*c(1)+x(4)*jl^3*c(2)+x(4)*jl^3*c(3)+4*g^2*l^2*...
cos(x(1))*ad(1)*sin(x(1))*x(2)^2*c(2)*c(1)*jl+2*g^2*l^2*cos(x(1))*...
ad(1)*sin(x(1))*x(2)^2*c(2)^2*jl-x(2)*th1*g*l*jl^2*c(1)*cos(x(1))-...
x(2)*th1*g*l*jl^2*c(2)*cos(x(1))+4*x(2)*g^2*l^2*cos(x(1))*ad(1)*...
sin(x(1))*jl*c(1)*x(1)-4.88*x(2)*g^2*l^2*cos(x(1))*ad(1)*sin(x(1))*...
c(1)*jl+2*x(2)*g^2*l^2*cos(x(1))*ad(1)*sin(x(1))*c(1)*x(3)*jl+2*...
x(2)*g^2*l^2*cos(x(1))*ad(1)*sin(x(1))*c(2)*x(3)*jl-3*x(2)*g^3*l^3*...
```

```
cos(x(1))*ad(1)*sin(x(1))^2*c(1)*th1-3*x(2)*g^3*l^3*cos(x(1))*ad(1)*...
sin(x(1))^2*c(2)*th1+2*x(2)*g^2*l^2*cos(x(1))*ad(1)*sin(x(1))*c(2)*...
jl*c(1)^2*x(1)+2*x(2)*g^2*l^2*sin(x(1))^2*ad(1)*c(1)*jl+2*x(2)*g^2*...
l^2*cos(x(1))*ad(1)*sin(x(1))*c(2)^2*jl*c(1)*x(1)-2.44*x(2)*g^2*l^2*...
cos(x(1))*ad(1)*sin(x(1))*c(2)*c(1)^2*jl-2.44*x(2)*g^2*l^2*cos(x(1))...
*ad(1)*sin(x(1))*c(2)^2*c(1)*jl+2*x(2)*g^2*l^2*cos(x(1))*ad(1)*...
sin(x(1))*c(2)*jl*x(1)-2.44*x(2)*g^2*l^2*cos(x(1))*ad(1)*sin(x(1))*...
c(2)*jl+x(2)*g^2*l^2*sin(x(1))^2*ad(1)*c(2)^2*c(1)*jl+x(2)*g^2*l^2*...
sin(x(1))^2*ad(1)*c(2)*jl+x(2)*g^2*l^2*sin(x(1))^2*ad(1)*c(2)*c(1)^2*...
jl-x(2)*c(3)*jl^2*th1*g*l*cos(x(1))-jl^2*th1*g*l*cos(x(1))*x(3)+jl*...
th1^2*g^2*l^2*cos(x(1))*sin(x(1))+c(2)*c(1)*jl^3*x(3)-c(2)*c(1)*jl^2*...
th1*g*l*sin(x(1))+g^2*l^2*sin(x(1))^2*ad(1)*x(3)*jl-g^3*l^3*...
sin(x(1))^3*ad(1)*th1+g^2*l^2*sin(x(1))^2*ad(1)*c(1)^2*x(3)*jl-g^3*...
l^3*sin(x(1))^3*ad(1)*c(1)^2*th1+2*g^2*l^2*sin(x(1))^2*ad(1)*c(2)*...
c(1)*x(3)*jl-2*g^3*l^3*sin(x(1))^3*ad(1)*c(2)*c(1)*th1+g^2*l^2*...
sin(x(1))^2*ad(1)*c(2)^2*x(3)*jl-g^3*l^3*sin(x(1))^3*ad(1)*c(2)^2*...
th1+c(3)*jl^3*c(1)*x(3)-c(3)*jl^2*c(1)*th1*g*l*sin(x(1))+c(3)*jl^3*...
c(2)*x(3)-c(3)*jl^2*c(2)*th1*g*l*sin(x(1))+x(4)*jl*g^2*l^2*...
sin(x(1))^2*ad(1)*c(1)+x(4)*jl*g^2*l^2*sin(x(1))^2*ad(1)*c(2)-2*...
jl^2*th1*g*l*sin(x(1))-th1*a1*g*l*sin(x(1))*jl^2+th1*g*l*jl^2*x(2)^2...
*sin(x(1))+2*g^2*l^2*cos(x(1))*ad(1)*sin(x(1))*x(2)^2*jl+2*g^2*l^2*...
cos(x(1))*ad(1)*sin(x(1))*c(1)^2*x(2)^2*jl+x(2)*c(3)*jl^3*c(2)*c(1))...
/jl^3-(-(x(2)*g*l*cos(x(1))+c(1)*g*l*sin(x(1))+c(2)*g*l*sin(x(1)))/...
jl-g^3*l^3*sin(x(1))^3/jl^3*ad(1)*(c(1)+c(2))-c(3)*g*l*sin(x(1))/jl)...
*(-(x(2)+c(1)*(x(1)-1.22))*ad(1)*g*l*sin(x(1))/jl-((c(1)*x(2)*jl+x(3)...
*jl-th1*g*l*sin(x(1)))/jl+c(2)*(x(2)+c(1)*(x(1)-1.22))+x(1)-1.22)*...
ad(1)*g*l*sin(x(1))/jl*(c(1)+c(2))-(-(x(2)*th1*g*l*cos(x(1))-x(2)*...
c(2)*c(1)*jl-x(2)*jl-c(1)*x(3)*jl+c(1)*th1*g*l*sin(x(1))-c(2)*x(3)*...
jl+c(2)*th1*g*l*sin(x(1))-x(4)*jl)/jl-g*l*sin(x(1))/jl*(-(x(2)+c(1)*...
(x(1)-1.22))*ad(1)*g*l*sin(x(1))/jl-((c(1)*x(2)*jl+x(3)*jl-th1*g*l*...
sin(x(1)))/jl+c(2)*(x(2)+c(1)*(x(1)-1.22))+x(1)-1.22)*ad(1)*g*l*...
sin(x(1))/jl*(c(1)+c(2)))+c(3)*((c(1)*x(2)*jl+x(3)*jl-th1*g*l*...
sin(x(1)))/jl+c(2)*(x(2)+c(1)*(x(1)-1.22))+x(1)-1.22)+x(2)+c(1)*...
(x(1)-1.22))*ad(1)*g*l*sin(x(1))*(-jl*th1*g*l*cos(x(1))+c(2)*c(1)*...
jl^2+2*jl^2+g^2*l^2*sin(x(1))^2*ad(1)+g^2*l^2*sin(x(1))^2*ad(1)*...
c(1)^2+2*g^2*l^2*sin(x(1))^2*ad(1)*c(2)*c(1)+g^2*l^2*sin(x(1))^2*...
```

```
ad(1)*c(2)^2+c(3)*jl^2*c(1)+c(3)*jl^2*c(2)+a1*jl^2)/jl^3)-c(4)*...
(-(x(2)*th1*g*l*cos(x(1))-x(2)*c(2)*c(1)*jl-x(2)*jl-c(1)*x(3)*jl+...
c(1)*th1*g*l*sin(x(1))-c(2)*x(3)*jl+c(2)*th1*g*l*sin(x(1))-x(4)*jl)...
/jl-g*l*sin(x(1))/jl*(-(x(2)+c(1)*(x(1)-1.22))*ad(1)*g*l*sin(x(1))/...
jl-((c(1)*x(2)*jl+x(3)*jl-th1*g*l*sin(x(1)))/jl+c(2)*(x(2)+c(1)*...
(x(1)-1.22))+x(1)-1.22)*ad(1)*g*l*sin(x(1))/jl*(c(1)+c(2)))+c(3)*...
((c(1)*x(2)*jl+x(3)*jl-th1*g*l*sin(x(1)))/jl+c(2)*(x(2)+c(1)*...
(x(1)-1.22))+x(1)-1.22)+x(2)+c(1)*(x(1)-1.22))-(c(1)*x(2)*jl+x(3)*...
jl-th1*g*l*sin(x(1)))/jl-c(2)*(x(2)+c(1)*(x(1)-1.22))-x(1)+1.22-...
(c(1)*x(2)*jl+x(3)*jl-th1*g*l*sin(x(1))+x(2)*c(2)*jl+c(2)*jl*c(1)*...
x(1)-1.22*c(2)*c(1)*jl+x(1)*jl-1.22*jl)*g^2*l^2*sin(x(1))^2/jl^5*...
ad(1)*(-jl*th1*g*l*cos(x(1))+c(2)*c(1)*jl^2+2*jl^2+g^2*l^2*...
sin(x(1))^2*ad(1)+g^2*l^2*sin(x(1))^2*ad(1)*c(1)^2+2*g^2*l^2*...
sin(x(1))^2*ad(1)*c(2)*c(1)+g^2*l^2*sin(x(1))^2*ad(1)*c(2)^2+c(3)*...
jl^2*c(1)+c(3)*jl^2*c(2)+a1*jl^2))/b;
%
%    System equations
%
xdot(1)=x(2);
xdot(2)=x(3)-theta(1)*g*l*sin(x(1))/jl;
xdot(3)=x(4);
xdot(4)=a0*x(3)+u1*b-theta(1)*a1*g*l*sin(x(1))/jl;
%
%    Parameter estimate equations
%
xdot(5)=tau1;
```

The MATLAB program above was used to obtain the computer simulations shown in Section 2.5.

### 8.3.3 Adaptive SMC Design for Observable Minimum Phase Systems

We now present the symbolic design of dynamical adaptive sliding mode tracking controllers via BACKDSMC for observable minimum-phase uncertain nonlinear systems, namely the implementation of the combined DAB-SMC algorithm described in Section 7.3.

**Example 8.3** Consider the mathematical model of the field-controlled DC motor of Section 7.4 when all parameters are assumed unknown

$$\begin{aligned} \dot{x}_1 &= -\theta_1 x_1 - \theta_2 x_2 u + \theta_3 \\ \dot{x}_2 &= -\theta_4 x_2 + \theta_5 x_1 u \\ y &= x_2 \end{aligned} \tag{8.13}$$

with

$$\theta_1 = \frac{R_a}{L_a}, \quad \theta_2 = \frac{K}{L_a}, \quad \theta_3 = \frac{V_a}{L_a}, \quad \theta_4 = \frac{B}{J}, \quad \theta_5 = \frac{K}{J} \tag{8.14}$$

As shown in Section 7.4 this system is locally *observable and minimum-phase* in the region characterized by the conditions

$$R_a B > K^2 U^2, \qquad V_a^2 \geqslant 4 R_a B X_2^2(U) \tag{8.15}$$

where $U$ is a constant equilibrium input voltage and $X_2(U)$ is the corresponding equilibrium of the angular velocity $x_2$. Assuming that the nominal unknown values of the system parameters satisfy the conditions (8.15), we can use BACKDSMC to design an adaptive dynamical sliding mode controller for tracking a desired trajectory. We consider here the same tracking problem of Section 7.4 in which the desired output reference trajectory $y_r(t)$ corresponds to a nonlinear function characterizing a smooth transition of the angular velocity $x_2$ between two equilibrium values $X_2$ and $X_2^*$

$$y_r(t) = \begin{cases} X_2 & \text{for} \quad 0 \leqslant t < t_0 \\ X_2^* + (X_2 - X_2^*)\exp(-3(t-t_0)^2) & \text{for} \quad t \geqslant t_0 \end{cases} \tag{8.16}$$

Regarding the general form (8.1) the symbolic expressions which characterize the mathematical model (8.13) are defined by the following sequence of MATLAB commands

```
f=sym('[0;0]')                          % f(x)
g=sym('[0;0]')                          % g(x)
phi=sym('[-x1,0,1,0,0;0,0,0,-x2,0]')    % phi(x)
psi=sym('[0,-x2,0,0,0;0,0,0,0,x1]')     % psi(x)
h='x2'                                  % h(x)
yd='vf+(vf-vi)*exp(-a*(t-t0)^2)'        % desired output yd
```

The combined DAB-SMC obtained in Section 7.4, which consists of the *sliding surface* (7.46), the *parameter update law* (7.50) and the *dynamical adaptive control law* (7.51) can be designed by running the following MATLAB command

```
[c,tau,z,sigma]=backdsmc(f,g,h,yd,phi,psi,'motrack')
```

Note that the fourth output argument 'sigma' indicates to BACKDSMC that the combined DAB-SMC control design algorithm should be performed. The generated m-file 'motrack.m' contains the system equations, the update law for the unknown parameter, the sliding surface and the adaptive control law as follows:

```
function xdot=motrack(t,x);
global c ad theta k lambda beta a;
%
%  Auxiliary variables
%
u1=x(8);
th1=x(3);
th2=x(4);
th3=x(5);
th4=x(6);
th5=x(7);
%
%  Sliding surface
%
sigma=k(1)*x(2)-k(1)*vf-k(1)*exp(-a*(t-t0)^2)*vf+k(1)*...
exp(-a*(t-t0)^2)*vi-x(2)*th4+u1*x(1)*th5+c(1)*x(2)-c(1)*vf-...
c(1)*exp(-a*(t-t0)^2)*vf+c(1)*exp(-a*(t-t0)^2)*vi+2*a*(t-t0)*...
exp(-a*(t-t0)^2)*vf-2*a*(t-t0)*exp(-a*(t-t0)^2)*vi;
%
%  Update law
%
tau1=-sigma*ad(1)*u1*x(1)*th5;
tau2=-sigma*ad(2)*u1^2*x(2)*th5;
tau3=sigma*ad(3)*u1*th5;
tau4=-(x(2)-vf-exp(-a*(t-t0)^2)*vf+exp(-a*(t-t0)^2)*vi)*ad(4)*x(2)+...
sigma*ad(4)*x(2)*(th4-c(1)-k(1));
tau5=(x(2)-vf-exp(-a*(t-t0)^2)*vf+exp(-a*(t-t0)^2)*vi)*ad(5)*u1*x(1)-...
sigma*ad(5)*u1*x(1)*(th4-c(1)-k(1));
%
%  Control law
%
```

```
control=(-k(1)*(-x(2)*th4+u1*x(1)*th5+2*(vf-vi)*a*(t-t0)*exp(-a...
*(t-t0)^2))-2*(vf-vi)*a*exp(-a*(t-t0)^2)+4*(vf-vi)*a^2*(t-t0)^2*...
exp(-a*(t-t0)^2)-x(2)*th4^2-u1*th5*th3+c(1)*x(2)*th4+ad(4)*x(2)^2*vf...
-2*c(1)*(vf-vi)*a*(t-t0)*exp(-a*(t-t0)^2)+u1*th5*x(1)*th1+u1^2*th5*...
x(2)*th2+th4*u1*x(1)*th5-c(1)*u1*x(1)*th5-ad(4)*x(2)^3+ad(4)*x(2)^2*...
exp(-a*(t-t0)^2)*vf-ad(4)*x(2)^2*exp(-a*(t-t0)^2)*vi+sigma*ad(4)*...
x(2)^2*th4-sigma*ad(4)*x(2)^2*c(1)-sigma*ad(4)*x(2)^2*k(1)-...
u1^2*x(1)^2*ad(5)*x(2)+u1^2*x(1)^2*ad(5)*vf+u1^2*x(1)^2*ad(5)*...
exp(-a*(t-t0)^2)*vf-u1^2*x(1)^2*ad(5)*exp(-a*(t-t0)^2)*vi+u1^2*...
x(1)^2*sigma*ad(5)*th4-u1^2*x(1)^2*sigma*ad(5)*c(1)-u1^2*x(1)^2*...
sigma*ad(5)*k(1)-lambda*(sigma+beta*sign(sigma)))/x(1)*th5;
%
%    System equations
%
xdot(1)=-x(1)*theta(1)-u1*x(2)*theta(2)+theta(3);
xdot(2)=-x(2)*theta(4)+u1*x(1)*theta(5);
%
%    Parameter estimate equations
%
xdot(3)=tau1;
xdot(4)=tau2;
xdot(5)=tau3;
xdot(6)=tau4;
xdot(7)=tau5;
%
%   Dynamic control equations
%
xdot(8)=control;
```

This m-file was used in the computer simulations shown in Section 7.4.

## 8.4 Concluding Remarks

These examples illustrate the use of the symbolic toolbox BACKDSMC in the design of three different classes of systems. In fact, all the m-files used to obtain the computer simulations of all the examples in this thesis were obtained by employing this toolbox.

The designer needs to select the initial conditions, the nominal values of the supposedly unknown parameters, the design parameters $\Gamma$, $c_i$, $k_i$, $\lambda$, and $\beta$, as well as the initial and final times for simulations. The design parameters $k_i$, $\lambda$ and $\beta$ used in the SAB-SMC and DAB-SMC algorithms should be selected as indicated by standard SMC (see Utkin's book [122] and Zinober's book [126]).

On the other hand, the selection of the design parameters $\Gamma$ and $c_i$ depends on the dynamics exhibited by the open-loop system and the desired performance of the closed-loop system. The greater the components of $\Gamma$ and $c_i$ are chosen, the faster the state variables and parameter estimate converge to the desired values. However, high gains may generate undesirable increase of the control and oscillations of the transient response of the variables, as shown in Section 2.4.4.

# Chapter 9

# Output Feedback Control of Uncertain Systems

## 9.1 Introduction

All the control design algorithms described in previous chapters have been developed under the assumption that the full state of the system is measured. We consider now a more realistic control design problem where only the output is available for measurement. The solution to this problem is very difficult because the *separation principle*, which allows one to design state-feedback controllers and observers as two separate modules for linear systems, is not applicable to nonlinear systems.

To provide a solution to this problem additional restrictions are usually needed and the transformation of the plant into a more convenient structure is also required (see, for instance, [65, 66]). We describe here a new approach proposed by Rios-Bolívar *et al* [93], which achieves the design of adaptive observers and output feedback controllers for a class of uncertain systems transformable into the *adaptive generalized observer canonical form*. In this form the nonlinearities multiplying the uncertain parameter vector depend only on the output and the control.

We first introduce *nonlinear damping terms* to compensate the destabilizing effects of the observer errors. Then a procedure is explained for the design of a deterministic backstepping observer/controller system and the concept of *passivity* is presented. Finally the systematic adaptive observer/control backstepping design procedure is described and an example is given for illustrative purposes.

## 9.2 Nonlinear Damping Terms

We describe in this section the use of nonlinear damping terms to guarantee boundedness of nonlinear systems perturbed by bounded and matched disturbances. To motivate the need for damping terms consider the scalar nonlinear system

$$\dot{x} = u + \psi(x)\vartheta(t) \tag{9.1}$$

where $\psi(x)$ is a known nonlinear function and $\vartheta(t)$ is a disturbing function of $t$.

Krstić *et al* [64] have shown that even for an exponentially decaying disturbance of the form

$$\vartheta(t) = \vartheta(0)e^{-kt} \tag{9.2}$$

the application of a linear control $u = -cx$ to a nonlinear system may lead to divergence of the state $x(t)$ to infinity in finite time for initial conditions satisfying

$$\vartheta(0)x(0) > c + k > 0. \tag{9.3}$$

This result shows the adverse effects which an apparently "mild" disturbance can produce in a nonlinear system. To overcome this problem and guarantee bounded $x(t)$ for all bounded $\vartheta(t)$ and for all $x(0)$, the control law $u = -cx$ can be augmented with a *nonlinear damping term* $-d(x)x$ [64]:

$$u = -cx - d(x)x \tag{9.4}$$

$d(x)$ is designed for (9.1) using the quadratic Lyapunov function $V(x) = \frac{1}{2}x^2$ whose time derivative is

$$\dot{V} = xu + x\psi(x)\vartheta(t) = -cx^2 - x^2 d(x) + x\psi(x)\vartheta(t). \tag{9.5}$$

The objective of guaranteeing *global boundedness* of solutions can be equivalently expressed as yielding $\dot{V}$ outside a compact region to be determined. This is achieved by selecting

$$d(x) = \kappa\psi^2(x), \qquad \kappa > 0 \tag{9.6}$$

which yields the control law

$$u = -cx - \kappa x\psi^2(x) \tag{9.7}$$

and

$$\begin{aligned} \dot{V} &= -cx^2 - \kappa x^2\psi^2(x) + x\psi(x)\vartheta(t) \\ &= -cx^2 - \kappa\left(x\psi(x) - \frac{\vartheta(t)}{2\kappa}\right)^2 + \frac{\vartheta^2(t)}{4\kappa} \\ &\leqslant -cx^2 + \frac{\vartheta^2(t)}{4\kappa}. \end{aligned} \tag{9.8}$$

Thus $\dot{V}$ is negative whenever

$$|x(t)| \geqslant \frac{\vartheta(t)}{2\sqrt{\kappa c}} \tag{9.9}$$

Since $\vartheta(t)$ is a bounded disturbance, $\dot{V}$ is negative outside the compact residual set

$$\mathcal{R} = \left\{ x: \quad |x| \leqslant \frac{\|\vartheta\|_\infty}{2\sqrt{\kappa c}} \right\} \tag{9.10}$$

and recalling that $V(x) = \frac{1}{2}x^2$, it can be concluded that $|x(t)|$ decreases whenever $x(t)$ is outside the set $\mathcal{R}$. Hence $x(t)$ is bounded

$$\|x\|_\infty \leqslant \max\left\{ |x(0)|,\ \frac{\|\vartheta\|_\infty}{2\sqrt{\kappa c}} \right\} \tag{9.11}$$

and also $x(t)$ converges to the compact set $\mathcal{R}$ defined in (9.10)

$$\lim_{t\to\infty} \text{dist}\{x(t), \mathcal{R}\} = 0. \tag{9.12}$$

These results show that global boundedness is guaranteed in the presence of bounded disturbances with unknown bounds, regardless of how small the gains $\kappa$ and $c$ are chosen [64]. Note that the size of $\mathcal{R}$ can be reduced by increasing the values of $\kappa$ and $c$.

Moreover, if the disturbance $\vartheta(t)$ converges to zero in addition to being bounded, then the control (9.7) guarantees convergence of $x(t)$ to zero in addition to global boundedness. To show this, consider a nonnegative monotically decreasing function $\overline{\vartheta}(t)$ such that $|\vartheta(t)| \leqslant \overline{\vartheta}(t)$ and $\lim_{t\to\infty} \overline{\vartheta}(t) = 0$. Then, one can obtain [64]

$$|x(t)| \leqslant |x(0)|e^{-ct} + \frac{1}{2\sqrt{\kappa c}} \left(\overline{\vartheta}(0)e^{-\frac{c}{2}t} + \overline{\vartheta}(t/2)\right) \tag{9.13}$$

and, since $\lim_{t\to\infty} \overline{\vartheta}(t/2) = 0$

$$\lim_{t\to\infty} x(t) = 0. \tag{9.14}$$

The use of nonlinear damping terms as a tool to improve performance and guarantee boundedness was firstly proposed by Kanellakopoulos [45] and Kanellakopoulos *et al* [51]. Similar damping terms are used in this chapter to counteract the destabilizing effects of observation errors in the design of output feedback controllers.

## 9.3 Observer/Controller Backstepping Design for Non-Adaptive Observable Systems

Our DDB algorithm involves nonlinear transformations of the controlled plant into error coordinates depending on the control input and its derivatives (see Section 3.4). The control algorithm is applicable to observable minimum-phase nonlinear systems and dynamically input-output linearizable.

Here the practical restriction of full-state measurement is relaxed by assuming that only the output is measured. Firstly an observer is designed for the system placed into the *Generalized Observer Canonical Form* (GORCF) which has been proposed by Keller and Fritz [54]. Then a modified version of the DDB algorithm, incorporating *nonlinear damping terms*, is developed to design dynamical output feedback controllers. Note that we have incorporated an additional $R$ in the acronym to distinguish it from the acronym GOCF to be used in this chapter for the Generalized Observability Canonical Form.

### 9.3.1 Observer Design

The design of state observers with linearizable error dynamics was firstly studied by Krener and Isidori [59], Krener and Respondek [60] and Bestle and Zeitz [4]. The Generalized Observability Canonical Form (GOCF) was proposed by Zeitz [125]. Fliess obtained the same canonical form in the setting of differential algebra (see [27, 28]). Since an observer cannot be synthesized from the GOCF directly, Keller and Fritz [54] used the GOCF to transform the observable system into the GORCF, from which the observer is obtained directly. We adopt this approach for the design of observers and combine it with the DDB algorithm along with nonlinear damping terms for the design of dynamical output controllers.

Consider the nonlinear system with no uncertainties

$$\begin{aligned} \dot{x} &= f_0(x) + g_0(x)u \\ y &= h(x) \end{aligned} \tag{9.15}$$

where $x \in \Re^n$ is the state, $u \in \Re$ the control input and $y \in \Re$ the output. It is assumed that (9.15) is a minimum-phase system with a well-defined relative degree $\rho$, i.e. $1 \leqslant \rho \leqslant n$. Use the operator (3.75) defined in Section 3.3

$$\begin{aligned} \mathcal{L}_h^0(x) &= h(x) \\ \mathcal{L}_h^i(x) &= \frac{\partial\left(\mathcal{L}_h^{i-1}(x)\right)}{\partial x} f_0(x) \qquad 1 \leqslant i \leqslant \rho - 1 \\ \mathcal{L}_h^\rho(x, v_1) &= \frac{\partial\left(\mathcal{L}_h^{\rho-1}(x)\right)}{\partial x}\Big(f_0(x) + g_0(x)v_1\Big) \\ \mathcal{L}_h^i(x, v_1, \ldots, v_{i-\rho+1}) &= \frac{\partial\left(\mathcal{L}_h^{i-1}(x, v_1, \ldots, v_{i-\rho})\right)}{\partial x}\Big(f_0(x) + g_0(x)v_1\Big) \\ &\quad + \sum_{j=1}^{i-\rho} \frac{\partial\left(\mathcal{L}_h^{i-1}(x, v_1, \ldots, v_{i-\rho})\right)}{\partial v_j} v_{j+1} \qquad \rho + 1 \leqslant i \leqslant n - 1 \end{aligned} \tag{9.16}$$

where the functions $v_1, v_2, \ldots$ correspond to $u, \dot{u}, \ldots$. This operator allows one to express the output $y$ and its first $n-1$ derivatives as functions of $x$, $u$ and the derivatives of $u$

$$\Phi(x, v_1, \ldots, v_{n-\rho}) = \begin{bmatrix} y \\ y^{(1)} \\ \vdots \\ y^{(n-1)} \end{bmatrix} = \begin{bmatrix} \mathcal{L}_h^0 \\ \mathcal{L}_h^1 \\ \vdots \\ \mathcal{L}_h^{n-1} \end{bmatrix}. \tag{9.17}$$

We also assume that (9.15) satisfies the observability condition

$$\text{rank}\frac{\partial \Phi(\cdot)}{\partial x} = n \tag{9.18}$$

at least locally. Then (9.15) can be transformed into the GOCF

$$\begin{aligned} \dot{z}_1 &= z_2 \\ &\vdots \\ \dot{z}_{n-1} &= z_n \\ \dot{z}_n &= f(z, u, \dot{u}, \ldots, u^{(n-\rho)}) \\ y &= z_1 \end{aligned} \tag{9.19}$$

A solution to the problem of designing state observers for systems transformable into GOCF, when only the output is measured, has been given by Keller and Fritz [54] using the following GORCF

$$\begin{aligned} \dot{\xi}_1 &= -\alpha_0(y, u, \dot{u}, \ldots, u^{(n-\rho)}) \\ \dot{\xi}_k &= \xi_{k-1} - \alpha_{k-1}(y, u, \dot{u}, \ldots, u^{(n-\rho-k+1)}) \qquad 2 \leqslant k \leqslant n-\rho+1 \\ \dot{\xi}_j &= \xi_{j-1} - \alpha_{j-1}(y) \qquad n-\rho+2 \leqslant j \leqslant n \\ y_\xi &= \xi_n = c(y) \end{aligned} \tag{9.20}$$

where $y$ is the output of (9.15). The GORCF can be obtained from (9.19) if the scalar function $f(z, u, \dot{u}, \ldots, u^{(n-\rho)})$ fulfils a special structural condition derived from the following *Generalized Characteristic Equation* (GCE)

$$\mathcal{L}^n c(y) + \sum_{i=1}^{n} \mathcal{L}^{n-i} \alpha_{n-i}(y, u, \ldots, u^{(i-\rho)}) = 0 \tag{9.21}$$

where $\mathcal{L}$ is the differential operator defined in (9.16).

For simplicity of notation we will restrict the explanation here to second order nonlinear systems. Thus, for a second order system with relative degree $\rho = 1$, the GCE is reduced to

$$\mathcal{L}^2 c(y) + \mathcal{L}\alpha_1(y, u) + \alpha_0(y, u, \dot{u}) = 0 \tag{9.22}$$

Applying the differential operator $\mathcal{L}$ to the system transformed into the GOCF (9.19) we obtain

$$\frac{d^2c(y)}{dy^2}z_2^2 + \frac{dc(y)}{dy}f(z,u,\dot{u}) + \frac{\partial\alpha_1(y,u)}{\partial y}z_2 + \frac{\partial\alpha_1(y,u)}{\partial u}\dot{u} + \alpha_0(y,u,\dot{u}) = 0 \tag{9.23}$$

Hence, the structural condition yields

$$f(z,u,\dot{u}) = \beta_2(z_1)z_2^2 + \beta_1(z_1,u)z_2 + \beta_0(z_1,u,\dot{u}) \tag{9.24}$$

If this condition is fulfilled, the functions $\beta_0$, $\beta_1$ and $\beta_2$ are known and the three unknown functions $c(y), \alpha_1$ and $\alpha_0$ can be determined from the three partial differential equations

$$\begin{aligned} -\beta_2(z_1)\frac{dc(y)}{dy} &= \frac{d^2c(y)}{dy^2} \\ -\beta_1(z_1,u)\frac{dc(y)}{dy} &= \frac{\partial\alpha_1(y,u)}{\partial y} \\ -\beta_0(z_1,u,\dot{u})\frac{dc(y)}{dy} &= \frac{\partial\alpha_1(y,u)}{\partial u}\dot{u} + \alpha_0(y,u,\dot{u}) \end{aligned} \tag{9.25}$$

By rewriting the GORCF (9.20) as follows

$$\begin{aligned} \dot{\xi} &= A\xi + \alpha(y,u,\ldots,u^{(n-\rho)}) \\ y_\xi &= c^T\xi \end{aligned} \tag{9.26}$$

with

$$A = \begin{bmatrix} 0 & 0 & \ldots & 0 & 0 \\ 1 & 0 & \ldots & 0 & 0 \\ \vdots & \vdots & \ldots & \vdots & \vdots \\ 0 & 0 & \ldots & 1 & 0 \end{bmatrix} \quad ; \quad \alpha(\cdot) = \begin{bmatrix} -\alpha_0(\cdot) \\ \vdots \\ -\alpha_{n-1}(\cdot) \end{bmatrix} \tag{9.27}$$

$$c^T = [0,\ldots,0,1]$$

an observer can be readily obtained as

$$\begin{aligned} \dot{\hat{\xi}} &= A\hat{\xi} + \alpha(y,u,\ldots,u^{(n-\rho)}) + K(\xi_n - \hat{\xi}_n) \\ y_{\hat{\xi}} &= c^T\hat{\xi} \end{aligned} \tag{9.28}$$

with $K = [k_1,\ldots,k_n]^T$ a vector of positive gains. Thus the observer error $e = \xi - \hat{\xi}$ exhibits the exponentially stable dynamics

$$\dot{e} = A_0 e \tag{9.29}$$

with

$$A_0 = (A - Kc^T) = \begin{bmatrix} 0 & 0 & \dots & 0 & -k_1 \\ 1 & 0 & \dots & 0 & -k_2 \\ \vdots & \vdots & \dots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & -k_n \end{bmatrix} \tag{9.30}$$

a Hurwitz matrix obtained by selecting the gains $k_i$'s appropriately.

**Example 9.1** Consider the second order nonlinear system

$$\begin{aligned} \dot{x}_1 &= -x_1 + x_2^2 + u \\ \dot{x}_2 &= x_1 x_2 + u \\ y &= x_2 \end{aligned} \tag{9.31}$$

The observability condition (9.18) is satisfied if $x_2 \neq 0$. Therefore only equilibrium points different from the origin can be considered using the proposed method. The control-dependent coordinate transformation

$$\begin{aligned} z_1 &= y = x_2 \\ z_2 &= \dot{y} = x_1 x_2 + u \end{aligned} \tag{9.32}$$

places (9.31) into the GOCF

$$\begin{aligned} \dot{z}_1 &= z_2 \\ \dot{z}_2 &= f(z,u,\dot{u}) = \left(\frac{1}{z_1}\right) z_2^2 - \left(1 + \frac{u}{z_1}\right) z_2 + (1 + z_1)u + z_1^3 + \dot{u} \\ y &= z_1 \end{aligned} \tag{9.33}$$

Note that $f(z,u,\dot{u})$ satisfies the structural condition (9.24) and the functions $\beta_0$, $\beta_1$ and $\beta_2$ are identified as

$$\beta_2(z_1) = \frac{1}{z_1}, \qquad \beta_1(z_1,u) = -\left(1 + \frac{u}{z_1}\right), \qquad \beta_0(z_1,u,\dot{u}) = (1 + z_1)u + z_1^3 + \dot{u} \tag{9.34}$$

Then, solving the partial differential equations (9.25), we obtain the unknown functions

$$c(y) = \ln y, \qquad \alpha_1(y,u) = \ln y - \frac{u}{y}, \qquad \alpha_0(y,u) = -\frac{u}{y} - u - y^2 \tag{9.35}$$

Therefore the coordinate transformation

$$\xi = \begin{bmatrix} \xi_1 \\ \xi_2 \end{bmatrix} = \Phi(x) = \begin{bmatrix} \ln(x_2) + x_1 \\ \ln(x_2) \end{bmatrix} \tag{9.36}$$

places the system (9.31) into the GORCF

$$\begin{aligned} \dot{\xi}_1 &= u\exp(-\xi_2) + \exp(2\xi_2) + u \\ \dot{\xi}_2 &= \xi_1 - \xi_2 + u\exp(-\xi_2) \\ y_\xi &= \xi_2 \end{aligned} \tag{9.37}$$

Finally the observer is readily synthesized as

$$\begin{aligned} \dot{\hat{\xi}}_1 &= u\exp(-\xi_2) + \exp(2\xi_2) + u + k_1(\xi_2 - \hat{\xi}_2) \\ \dot{\hat{\xi}}_2 &= \hat{\xi}_1 - \xi_2 + u\exp(-\xi_2) + k_2(\xi_2 - \hat{\xi}_2) \\ y_{\hat{\xi}} &= \hat{\xi}_2 \end{aligned} \tag{9.38}$$

Note that the coordinate transformation (9.36) is valid locally for positive values of the state variable $x_2$.

This method is convenient when the system order is low. However, with increasing order the structural conditions become stronger [54].

### 9.3.2 Dynamical Deterministic Control Design

We now proceed to the design of a dynamical deterministic backstepping output control by using a slightly modified version of the DDB algorithm, which incorporates nonlinear damping terms. The approach follows a systematic procedure and employs the estimate of the unmeasured state variables and the available output to design a dynamical controller in order to ensure the derivative of the Lyapunov function is nonpositive. For simplicity we restrict our attention to second order systems which are transformable into the GORCF

$$\begin{aligned} \dot{\xi}_1 &= -\alpha_0(\xi_2, u, \dot{u}) \\ \dot{\xi}_2 &= \xi_1 - \alpha_1(\xi_2, u) \\ y_\xi &= \xi_2 \end{aligned} \tag{9.39}$$

As shown in Section 9.3.1 the following dynamics

$$\begin{aligned} \dot{\hat{\xi}}_1 &= -\alpha_0(\xi_2, u, \dot{u}) + k_1(\xi_2 - \hat{\xi}_2) \\ \dot{\hat{\xi}}_2 &= \hat{\xi}_1 - \alpha_1(\xi_2, u) + k_2(\xi_2 - \hat{\xi}_2) \\ y_{\hat{\xi}} &= \hat{\xi}_2 \end{aligned} \tag{9.40}$$

constitutes an observer for the system (9.39). The error system has the following exponential stable dynamics

$$\dot{e} = A_0 e = (A - Kc^T)e \tag{9.41}$$

with

$$A_0 = \begin{pmatrix} 0 & -k_1 \\ 1 & -k_2 \end{pmatrix} \tag{9.42}$$

The problem of tracking a bounded time-dependent desired trajectory $\xi_r(t)$ with bounded derivatives, can be solved by applying the control design algorithm below.

**Step 1.** Define the tracking error

$$z_1 := y_\xi - \xi_r = \xi_2 - \xi_r \tag{9.43}$$

whose time derivative is given by

$$\dot{z}_1 = \xi_1 - \alpha_1(\xi_2, u) - \dot{\xi}_r. \tag{9.44}$$

Since the state variable $\xi_1$ is not measured, we replace it with its estimate and take into account the observation error $e_1 = \xi_1 - \hat{\xi}_1$ to obtain

$$\dot{z}_1 = \hat{\xi}_1 - \alpha_1(\xi_2, u) - \dot{\xi}_r + e_1 \tag{9.45}$$

Consider now the quadratic Lyapunov function

$$V_1 = \frac{1}{2} z_1^2 \tag{9.46}$$

with time derivative

$$\dot{V}_1 = z_1 \left[ \hat{\xi}_1 - \alpha_1(\xi_2, u) - \dot{\xi}_r \right] + z_1 e_1 \tag{9.47}$$

Note that due to the presence of the observation error $e_1$ we are not able to achieve $\dot{V}_1 = -c_1 z_1^2$ by making the bracketed term multiplying $z_1$ equal to $-c_1 z_1$. Nevertheless, we can consider the observation error $e_1$ as a destabilizing perturbation whose effects may be compensated by incorporating an additional nonlinear damping term $d_1 \psi_1^2 z_1$ in the relation involving the bracketed term multiplying $z_1$, as shown in Section 9.2. Thus

$$\hat{\xi}_1 - \alpha_1(\xi_2, u) - \dot{\xi}_r = -c_1 z_1 - d_1 \psi_1^2 z_1 \tag{9.48}$$

where $c_1$ and $d_1$ are positive design parameters, is a desired relation for the bracketed term multiplying $z_1$. Since the term multiplying the observation error $e_1$ in (9.47) is $z_1$, $\psi_1 = 1$. Moreover, since the relation (9.48) is not valid in general, we define the second error variable as

$$z_2 := \hat{\xi}_1 - \alpha_1^*(\xi_2, u, \xi_r, \dot{\xi}_r) \tag{9.49}$$

with

$$\alpha_1^*(\xi_2, u, \xi_r, \dot{\xi}_r) = \alpha_1(\xi_2, u) + \dot{\xi}_r - c_1 z_1 - d_1 z_1. \tag{9.50}$$

Then

$$\begin{aligned}\dot{V}_1 &= -c_1 z_1^2 + z_1 z_2 - d_1 z_1^2 + z_1 e_1 \\ &= -c_1 z_1^2 + z_1 z_2 - d_1 \left(z_1 - \frac{e_1}{2d_1}\right)^2 + \frac{e_1^2}{4d_1} \\ &\leqslant -c_1 z_1^2 + z_1 z_2 + \frac{e_1^2}{4d_1}. \end{aligned} \tag{9.51}$$

**Step 2.** The time derivative of $z_2$ is obtained from (9.39) and (9.40) as

$$\dot{z}_2 = -\alpha_0(\xi_2, u, \dot{u}) + k_1 e_2 - \frac{\partial \alpha_1^*}{\partial \xi_2}(\xi_1 - \alpha_1(\xi_2, u)) - \frac{\partial \alpha_1^*}{\partial u}\dot{u} - \frac{\partial \alpha_1^*}{\partial \xi_r}\dot{\xi}_r - \frac{\partial \alpha_1^*}{\partial \dot{\xi}_r}\ddot{\xi}_r. \tag{9.52}$$

We again replace $\xi_1$ with its estimate $\hat{\xi}_1$ in (9.52) and take into account the observation error $e_1$

$$\dot{z}_2 = -\alpha_0(\xi_2, u, \dot{u}) + k_1 e_2 + \psi_2(\hat{\xi}_1 - \alpha_1(\xi_2, u)) - \frac{\partial \alpha_1^*}{\partial u}\dot{u} - \frac{\partial \alpha_1^*}{\partial \xi_r}\dot{\xi}_r - \frac{\partial \alpha_1^*}{\partial \dot{\xi}_r}\ddot{\xi}_r + \psi_2 e_1 \tag{9.53}$$

with

$$\psi_2 = -\frac{\partial \alpha_1^*}{\partial \xi_2}. \tag{9.54}$$

The Lyapunov function is augmented as

$$V_2 = V_1 + \frac{1}{2} z_2^2 \tag{9.55}$$

and

$$\begin{aligned}\dot{V}_2 \leqslant\; & -c_1 z_1^2 + \frac{e_1^2}{4d_1} + z_2 \Big[ z_1 - \alpha_0(\xi_2, u, \dot{u}) + k_1 e_2 + \psi_2(\hat{\xi}_1 - \alpha_1(\xi_2, u)) \\ & - \frac{\partial \alpha_1^*}{\partial u}\dot{u} - \frac{\partial \alpha_1^*}{\partial \xi_r}\dot{\xi}_r - \frac{\partial \alpha_1^*}{\partial \dot{\xi}_r}\ddot{\xi}_r \Big] + z_2 \psi_2 e_1. \end{aligned} \tag{9.56}$$

Note that the observation error $e_1$ appears again in the derivative of the Lyapunov function. So, we can now choose a dynamical control law with an additional damping term $d_2 \psi_2^2 z_2$ to make the bracketed term multiplying $z_2$ equal to $-c_2 z_2 - d_2 \psi_2^2 z_2$, namely

$$\begin{aligned}-c_2 z_2 - d_2 \psi_2^2 z_2 =\; & z_1 - \alpha_0(\xi_2, u, \dot{u}) + k_1 e_2 + \psi_2(\hat{\xi}_1 - \alpha_1(\xi_2, u)) \\ & - \frac{\partial \alpha_1^*}{\partial u}\dot{u} - \frac{\partial \alpha_1^*}{\partial \xi_r}\dot{\xi}_r - \frac{\partial \alpha_1^*}{\partial \dot{\xi}_r}\ddot{\xi}_r \end{aligned} \tag{9.57}$$

where $c_2$ and $d_2$ are positive design parameters. Thus $\dot{V}_2$ becomes

$$\begin{aligned}\dot{V}_2 &\leqslant -c_1 z_1^2 - c_2 z_2^2 + \frac{e_1^2}{4d_1} - d_2 \psi_2^2 z_2^2 + z_2 \psi_2 e_1 \\ &= -c_1 z_1^2 - c_2 z_2^2 + \frac{e_1^2}{4d_1} - d_2 \left(\psi_2 z_2 - \frac{e_1}{2d_2}\right)^2 + \frac{e_1^2}{4d_2} \\ &\leqslant -c_1 z_1^2 - c_2 z_2^2 + \frac{1}{4}\left(\frac{1}{d_1} + \frac{1}{d_2}\right) e_1^2 \end{aligned} \tag{9.58}$$

We can augment the Lyapunov function with a quadratic term in the observation errors

$$V = V_2 + \left(\frac{1}{d_1} + \frac{1}{d_2}\right) e^T P e \tag{9.59}$$

where $P$ is a symmetric positive definite matrix which satisfies the Lyapunov equation

$$A_0^T P + P A_0 = -I \tag{9.60}$$

Thus we obtain

$$\dot{V} \leqslant -c_1 z_1^2 - c_2 z_2^2 - \frac{3}{4}\left(\frac{1}{d_1} + \frac{1}{d_2}\right) e_1^2 - \left(\frac{1}{d_1} + \frac{1}{d_2}\right) e_2^2 \tag{9.61}$$

Hence, both asymptotic tracking and exponential convergence of the observation errors to zero is achieved.

**Example 9.2** Consider again the second order nonlinear system of Example 9.1

$$\begin{aligned} \dot{x}_1 &= -x_1 + x_2^2 + u \\ \dot{x}_2 &= x_1 x_2 + u \\ y &= x_2 \end{aligned} \tag{9.62}$$

which is transformable into the GORCF

$$\begin{aligned} \dot{\xi}_1 &= u\exp(-\xi_2) + \exp(2\xi_2) + u \\ \dot{\xi}_2 &= \xi_1 - \xi_2 + u\exp(-\xi_2) \\ y_\xi &= \xi_2 \end{aligned} \tag{9.63}$$

and hence an observer for (9.63) is

$$\begin{aligned} \dot{\hat{\xi}}_1 &= u\exp(-\xi_2) + \exp(2\xi_2) + u + k_1(\xi_2 - \hat{\xi}_2) \\ \dot{\hat{\xi}}_2 &= \hat{\xi}_1 - \xi_2 + u\exp(-\xi_2) + k_2(\xi_2 - \hat{\xi}_2) \\ y_{\hat{\xi}} &= \hat{\xi}_2 \end{aligned} \tag{9.64}$$

After applying the modified DDB algorithm above we obtain the coordinate transformation

$$z_1 = \xi_2 - \xi_r \tag{9.65}$$

$$z_2 = \hat{\xi}_1 - \xi_2 + u\exp(-\xi_2) - \dot{\xi}_r + (c_1 + d_1)(\xi_2 - \xi_r) \tag{9.66}$$

and the dynamical controller

$$\begin{aligned} \dot{u} = \exp(\xi_2)\Big[ &- z_1 - u\exp(-\xi_2) - \exp(2\xi_2) - u - k_1(\xi_2 - \hat{\xi}_2) + \ddot{\xi}_r + (c_1 + d_1)\dot{\xi}_r \\ &- \psi_2\Big(\hat{\xi}_1 - \xi_2 + u\exp(-\xi_2)\Big) - c_2 z_2 - d_2 \psi_2^2 z_2\Big] \end{aligned} \tag{9.67}$$

with

$$\psi_2 = -1 + c_1 + d_1 - u\exp(-\xi_2). \tag{9.68}$$

This systematic procedure can be extended to the general case guaranteeing asymptotic tracking and exponential convergence of the observation errors to zero.

A more complicated problem is tracking a desired output when the system contains parametric uncertainty and only the output is measured. Before developing a solution to this problem let us introduce some important concepts.

## 9.4 Passivity

The concept of passivity is very important in adaptive control. We present here some definitions given by Byrnes *et al* [11], Krstić *et al* [64] and Slotine and Li [114]. Consider systems of the form

$$\begin{aligned} \dot{x} &= f(x,t) + g(x,t)u \\ y &= h(x,t) \end{aligned} \tag{9.69}$$

with $x \in \Re^n$, $y \in \Re^m$, $u \in \Re^m$, and $f, g, h$ continuous in $t$ and smooth in $x$. Suppose that $f(0,t) = 0$ and $h(0,t) = 0$ for all $t \geqslant 0$.

**Definition 9.1** *The system (9.69) is said to be* passive *if there exists a continuous ("storage") function* $V : \Re^n \times \Re_+ \to \Re_+$, *which satisfies* $V(0,t) = 0$, $\forall t \geqslant 0$, *such that for all* $u \in C^0$, $x(0) \in \Re^n$, $t \geqslant t_0 \geqslant 0$

$$\int_{t_0}^{t} y^T(\sigma)u(\sigma)d\sigma \geqslant V(x(t),t) - V(x(t_0),t_0). \tag{9.70}$$

**Definition 9.2** *The system (9.69) is said to be* strictly passive *if there exist a continuous nonnegative (storage) function* $V : \Re^n \times \Re_+ \to \Re_+$, *which satisfies* $V(0,t) = 0$, $\forall t \geqslant 0$, *and a positive definite function (dissipation rate)* $\psi : \Re^n \to \Re_+$, *such that for all* $u \in C^0$, $x(0) \in \Re^n$, $t \geqslant t_0 \geqslant 0$

$$\int_{t_0}^{t} y^T(\sigma)u(\sigma)d\sigma \geqslant V(x(t),t) - V(x(t_0),t_0) + \int_{t_0}^{t} \psi(x(\sigma))d\sigma. \tag{9.71}$$

The concepts of passivity and Lyapunov stability are closely related, as shown below.

**Lemma 9.1** *Suppose the system (9.69) is (strictly) passive. If* $V$ *is positive definite, radially unbounded and decrescent, i.e. if there exist class* $K_\infty$ *functions* $\gamma_1$ *and* $\gamma_2$ *such that* $\gamma_1(|x|) \leqslant V(x,t) \leqslant \gamma_2(|x|)$, $\forall (x,t) \in \Re^n \times \Re_+$, *then, for* $u \equiv 0$, *the equilibrium* $x = 0$ *of (9.69) is globally uniformly (asymptotically) stable.*

Since adaptive control is concerned with the design of interconnected identifiers and control laws, the feedback interconnection of passive systems is very important. Consider the two passive systems

$$\begin{aligned} \dot{x}_1 &= f_1(x_1,t) + g_1(x_1,t)u_1 \\ y_1 &= h_1(x_1,t) \end{aligned} \tag{9.72}$$

$$\begin{aligned} \dot{x}_2 &= f_2(x_2,t) + g_2(x_2,t)u_2 \\ y_2 &= h_2(x_2,t) \end{aligned} \tag{9.73}$$

connected by the relations

$$u_1 = -y_2 + v_1, \qquad u_2 = y_1, \tag{9.74}$$

where $v_1$ is a reference input to the interconnected system.

**Theorem 9.1** *Suppose the system (9.72) is (strictly) passive with storage function $V_1$ (and dissipation rate $\psi_1$) independent of $x_2$. Likewise, suppose the system (9.73) is (strictly) passive with storage function $V_2$ (and dissipation rate $\psi_2$) independent of $x_1$. Then the interconnected system (9.72)-(9.73) with input $v_1$ and output $y_1$ is*

1. *strictly passive if both (9.72) and (9.73) are strictly passive*
2. *passive if at least one of the systems (9.72) and (9.73) is passive, but not strictly passive.*

*Moreover, when $v_1 \equiv 0$, if (9.72) is strictly passive and (9.73) is passive, then the equilibrium $x = 0$ is uniformly stable and $\lim_{t\to\infty} x_1(t) = 0$.*

In the case of time-invariant linear system the concept of passivity is equivalent to positive realness [114].

**Definition 9.3** *A rational transfer function $G(s)$ is said to be* positive real *if $G(s)$ is real for all real $s$, and $\mathrm{Re}\{G(s)\} \geqslant 0$ for all $\mathrm{Re}\{s\} \geqslant 0$. If in addition $G(s-\varepsilon)$ is positive real for some $\varepsilon > 0$, then $G(s)$ is said to be* strictly positive real.

Thus the basic difference between positive real (PR) and strictly positive real (SPR) transfer functions is that PR transfer functions may tolerate poles on the $j\omega$ axis, while SPR functions cannot.

We will show that the closed-loop system obtained from the application of both the SAB and DAB algorithms have an important passivity property. Recall from Sections 2.4.3 and 4.2 that the closed-loop system written in the $(z, \tilde{\theta})$ coordinates, has the form

$$\dot{z} = A_z z + W^T \tilde{\theta} \tag{9.75}$$

$$\dot{\tilde{\theta}} = -\Gamma W z \tag{9.76}$$

where $A_z$ has a skew-symmetric form, $\tilde{\theta} = \theta - \hat{\theta}$ is the parameter estimate error and $\Gamma = \Gamma^T > 0$ the adaptation gain matrix. Due to the structure of $A_z$, along the solutions of (9.75) we have

$$\begin{aligned} \frac{d}{dt}\left(\frac{1}{2} z^T z\right) &= -\sum_{i=1}^{n} c_i z_i^2 + z^T W^T \tilde{\theta}, \qquad c_i > 0, \quad 1 \leqslant i \leqslant n \\ &= -\sum_{i=1}^{n} c_i z_i^2 + \tau^T \tilde{\theta}. \end{aligned} \tag{9.77}$$

By integrating (9.77) over $[0, t]$, we get

$$\int_0^t \tau^T(s)\tilde{\theta}(s) ds = \frac{1}{2} z^T(t) z(t) - \frac{1}{2} z^T(0) z(0) + \int_0^t \sum_{i=1}^{n} c_i z_i^2(s) ds. \tag{9.78}$$

By Definition 9.2, (9.78) implies that the system

$$\begin{aligned} \dot{z} &= A_z z + W^T \tilde{\theta} \\ \tau &= W z \end{aligned} \tag{9.79}$$

is *strictly passive* with $\tilde{\theta}$ as its input, the function $\tau$ as its output, $V(z) = \frac{1}{2} z^T z$ as the storage function, and $\psi(z) = \sum_{i=1}^{n} c_i z_i^2$ as the dissipation rate. Moreover, the integrator system

$$-\tilde{\theta} = \frac{\Gamma}{s}\tau = \frac{\tau_n}{s} \tag{9.80}$$

with $\tau_n$ the tuning function obtained at the final step, is *passive* from $\tau_n$ to $-\tilde{\theta}$. Hence, the closed-loop adaptive system represents a negative feedback interconnection of the strictly passive system (9.79) with the passive system (9.80). Thus, Theorem 9.1 indicates that the equilibrium $(z, \tilde{\theta}) = (0, 0)$ is stable and $z \to 0$ as $t \to \infty$.

We will carry out a similar analysis on the stability of the closed-loop system having observer and controller.

## 9.5 Adaptive Observer/Controller Backstepping Design for Observable Uncertain Systems

The design of output feedback controllers in the presence of uncertainty is a challenging problem in adaptive control. This problem is usually tackled by imposing additional restrictions on the class of uncertain systems and resorting to canonical forms. For instance, Kanellakopoulos *et al* [47, 50] and Marino and Tomei [73, 74] have proposed the use of the *output feedback form* and the *adaptive observer form* respectively, for the design of output feedback controllers and observers (or filters). Both canonical forms restrict the nonlinearities multiplying the uncertain parameter, to be dependent of the output only.

We consider here a new approach proposed by Rios-Bolívar *et al* [93] for the design of observers and output feedback controllers for systems with parametric uncertainty, when only the output is available for feedback. We can represent these systems by

$$\begin{aligned} \dot{x} &= f_0(x) + g_0(x)u + \Psi(x)\theta \\ y &= h(x) \end{aligned} \tag{9.81}$$

where $\Psi$ is a known matrix whose entries are smooth nonlinear functions, and $\theta \in \Re^p$ a constant unknown parameter vector. A solution to both observer and tracking control design can be obtained when, after applying the transformation $\xi = \Phi(x)$ which places the nominal system

$$\begin{aligned} \dot{x} &= f_0(x) + g_0(x)u \\ y &= h(x) \end{aligned} \tag{9.82}$$

into the GORCF, the resulting nonlinearities multiplying the uncertain parameter vector $\theta$ depend only upon the measured output and the control, namely

$$\begin{aligned} \dot{\xi} &= A\xi + \alpha(y_\xi, u, \dots, u^{(n-\rho)}) + \Lambda(y_\xi, u)\theta \\ y_\xi &= c^T\xi \end{aligned} \tag{9.83}$$

with $A$, $c$ and $\alpha$ defined as in (9.27), and $\Lambda \in \Re^{n \times p}$ a matrix whose entries are smooth functions in $y_\xi$ and $u$. We will call the structure of the system (9.83) the *adaptive GORCF*. For the transformed system (9.83) we propose the following observer

$$\begin{aligned} \dot{\hat{\xi}} &= A\hat{\xi} + \alpha(y, u, \dots, u^{(n-\rho)}) + K(\xi_n - \hat{\xi}_n) + \Lambda(y_\xi, u)\hat{\theta} \\ y_{\hat{\xi}} &= c^T\hat{\xi} \end{aligned} \tag{9.84}$$

with

$$\Lambda(y_\xi,u)=\begin{bmatrix}\lambda_1(y_\xi,u)\\ \vdots\\ \lambda_n(y_\xi,u)\end{bmatrix}=\begin{bmatrix}\lambda_{1,1}(y_\xi,u) & \dots & \lambda_{1,p}(y_\xi,u)\\ \vdots & & \vdots\\ \lambda_{n,1}(y_\xi,u) & \dots & \lambda_{n,p}(y_\xi,u)\end{bmatrix}. \tag{9.85}$$

The observation error system yields

$$\dot e = A_0 e + \Lambda(y_\xi,u)\tilde\theta \tag{9.86}$$

where $A_0 = A - Kc^T$, $e=\xi-\hat\xi$ and $\tilde\theta=\theta-\hat\theta$.

Since $A_0$ is a Hurwitz matrix, the system

$$\begin{aligned}\dot e &= A_0e+\Lambda(y_\xi,u)\tilde\theta\\ \tau &= \Lambda^T(y_\xi,u)Pe\end{aligned} \tag{9.87}$$

where $P$ is a symmetric positive definite matrix satisfying $A_0^TP+PA_0=-I$, is strictly passive with input $\tilde\theta$, output $\tau$, storage function $V=\frac{1}{2}e^TPe$ and dissipation rate $2e^Te$. However, since only $e_n$ is available, we cannot use $\tau$ as our first tuning function. Nevertheless, from the structure of $A_0$ we notice that $e_n$ has the following relation

$$\dot e_n = -k_ne_n+\lambda_n(y_\xi,u)\tilde\theta+e_{n-1} \qquad k_n>0 \tag{9.88}$$

If $e_{n-1}$ was not present in (9.88), the subsystem

$$\dot e_n = -k_ne_n+\lambda_n(y_\xi,u)\tilde\theta \qquad k_n>0 \tag{9.89}$$

$$\tau = \Gamma\lambda_n^T(y_\xi,u)e_n \tag{9.90}$$

would be strictly passive with input $\tilde\theta$ and output $\tau$. This would allow to choose the gradient type parameter law (see Sastry and Bodson [97])

$$\tau_0=\Gamma\lambda_n^T(y_\xi,u)e_n \tag{9.91}$$

to guarantee asymptotic stability of $e_n$. However, since the presence of the disturbing term $e_{n-1}$, we must incorporate nonlinear damping terms in the design procedure to compensate the destabilizing effect of $e_{n-1}$. Therefore we will use (9.91) as the first tuning function of the design procedure.

For the sake of simplicity we describe below the design procedure for a second order system with relative degree one, already in the adaptive GORCF

$$\begin{aligned}\dot\xi_1 &= -\alpha_0(\xi_2,u,\dot u)+\lambda_1(\xi_2,u)\theta\\ \dot\xi_2 &= \xi_1-\alpha_1(\xi_2,u)+\lambda_2(\xi_2,u)\theta\\ y_\xi &= \xi_2\end{aligned} \tag{9.92}$$

The observer dynamics is

$$\begin{aligned}\dot{\hat{\xi}}_1 &= -\alpha_0(\xi_2, u, \dot{u}) + k_1(\xi_2 - \hat{\xi}_2) + \lambda_1(\xi_2, u)\hat{\theta} \\ \dot{\hat{\xi}}_2 &= \hat{\xi}_1 - \alpha_1(\xi_2, u) + k_2(\xi_2 - \hat{\xi}_2) + \lambda_2(\xi_2, u)\hat{\theta} \\ y_\xi &= \xi_2\end{aligned} \tag{9.93}$$

and the function

$$\tau_0 = \Gamma\lambda_2^T e_2 \tag{9.94}$$

with $\Gamma = \Gamma^T > 0$, is consider our first tuning function.

**Step 1.** The derivative of the tracking error $z_1 = \xi_2 - \xi_r$ is given by

$$\dot{z}_1 = \hat{\xi}_1 - \alpha_1(\xi_2, u) + \lambda_2(\xi_2, u)\hat{\theta} - \dot{\xi}_r + \lambda_2(\xi_2, u)\tilde{\theta} + e_1 \tag{9.95}$$

Using the Lyapunov function

$$V_1 = \frac{1}{2}z_1^2 + \frac{1}{2}\tilde{\theta}\Gamma^{-1}\tilde{\theta} \tag{9.96}$$

the time derivative

$$\dot{V}_1 = z_1\left[\hat{\xi}_1 - \alpha_1(\xi_2, u) + \lambda_2(\xi_2, u)\hat{\theta} - \dot{\xi}_r\right] + z_1 e_1 + \tilde{\theta}\Gamma^{-1}\left(-\dot{\hat{\theta}} + \tau_0 + \Gamma\lambda_2^T z_1\right) \tag{9.97}$$

is obtained. Taking the second tuning function as follows

$$\tau_1 = \tau_0 + \Gamma\lambda_2^T z_1 = \Gamma\lambda_2^T(e_2 + z_1) \tag{9.98}$$

and defining our second error coordinate as

$$z_2 = \hat{\xi}_1 - \alpha_1^*(\xi_2, u, \hat{\theta}, \xi_r, \dot{\xi}_r) \tag{9.99}$$

with

$$\alpha_1^*(\xi_2, u, \hat{\theta}, \xi_r, \dot{\xi}_r) = \alpha_1(\xi_2, u) - \lambda_2(\xi_2, u)\hat{\theta} + \dot{\xi}_r - (c_1 + d_1)(\xi_2 - \xi_r) \tag{9.100}$$

where $c_1$ and $d_1$ are positive design parameters, $\dot{V}_1$ becomes

$$\begin{aligned}\dot{V}_1 &= -c_1 z_1^2 + z_1 z_2 - d_1 z_1^2 + z_1 e_1 + \tilde{\theta}\Gamma^{-1}(-\dot{\hat{\theta}} + \tau_1) \\ &= -c_1 z_1^2 + z_1 z_2 - d_1\left(z_1 - \frac{e_1}{2d_1}\right)^2 + \frac{e_1^2}{4d_1} + \tilde{\theta}\Gamma^{-1}(-\dot{\hat{\theta}} + \tau_1) \\ &\leqslant -c_1 z_1^2 + z_1 z_2 + \frac{e_1^2}{4d_1} + \tilde{\theta}\Gamma^{-1}(-\dot{\hat{\theta}} + \tau_1)\end{aligned} \tag{9.101}$$

**Step 2.** The time derivative of $z_2$ is

$$\begin{aligned}\dot{z}_2 &= -\alpha_0(\xi_2, u, \dot{u}) + k_1 e_2 + \lambda_1\hat{\theta} + \psi_2\Big(\xi_1 - \alpha_1(\xi_2, u) + \lambda_2\theta\Big) \\ &\quad -\frac{\partial\alpha_1^*}{\partial u}\dot{u} - \frac{\partial\alpha_1^*}{\partial\hat{\theta}}\dot{\hat{\theta}} - \frac{\partial\alpha_1^*}{\partial\xi_r}\dot{\xi}_r - \frac{\partial\alpha_1^*}{\partial\dot{\xi}_r}\ddot{\xi}_r\end{aligned} \tag{9.102}$$

with

$$\psi_2 = -\frac{\partial \alpha_1^*}{\partial \xi_2} \tag{9.103}$$

By considering the estimate of $\xi_1$ and $\theta$, (9.102) can be rewritten as

$$\begin{aligned} \dot{z}_2 &= -\alpha_0(\xi_2, u, \dot{u}) + k_1 e_2 + \lambda_1 \hat{\theta} + \psi_2\left(\hat{\xi}_1 - \alpha_1(\xi_2, u) + \lambda_2 \hat{\theta}\right) \\ &\quad -\frac{\partial \alpha_1^*}{\partial u}\dot{u} - \frac{\partial \alpha_1^*}{\partial \hat{\theta}}\dot{\hat{\theta}} - \frac{\partial \alpha_1^*}{\partial \xi_r}\dot{\xi}_r - \frac{\partial \alpha_1^*}{\partial \dot{\xi}_r}\ddot{\xi}_r + \psi_2 e_1 + \psi_2 \lambda_2 \tilde{\theta} \end{aligned} \tag{9.104}$$

Augmenting the Lyapunov function

$$V_2 = V_1 + \frac{1}{2}z_2^2 \tag{9.105}$$

and

$$\begin{aligned} \dot{V}_2 &\leqslant -c_1 z_1^2 + \frac{e_1^2}{4d_1} + \tilde{\theta}\Gamma^{-1}(-\dot{\hat{\theta}} + \tau_1 + \Gamma\lambda_2^T \psi_2 z_2) + \psi_2 e_1 \\ &\quad + z_2\Bigg[z_1 - \alpha_0(\xi_2, u, \dot{u}) + k_1 e_2 + \lambda_1 \hat{\theta} + \psi_2\left(\hat{\xi}_1 - \alpha_1(\xi_2, u) + \lambda_2 \hat{\theta}\right) \\ &\quad -\frac{\partial \alpha_1^*}{\partial u}\dot{u} - \frac{\partial \alpha_1^*}{\partial \hat{\theta}}\dot{\hat{\theta}} - \frac{\partial \alpha_1^*}{\partial \xi_r}\dot{\xi}_r - \frac{\partial \alpha_1^*}{\partial \dot{\xi}_r}\ddot{\xi}_r\Bigg]. \end{aligned} \tag{9.106}$$

Finally, the actual update law for the unknown parameters yields

$$\dot{\hat{\theta}} = \tau_2 = \tau_1 + \Gamma\lambda_2^T \psi_2 z_2 = \Gamma\lambda_2^T(e_2 + z_1 + \psi_2 z_2) \tag{9.107}$$

and the dynamical adaptive control is

$$\begin{aligned} -d_2\psi_2^2 z_2 - c_2 z_2 &= z_1 - \alpha_0(\xi_2, u, \dot{u}) + k_1 e_2 + \lambda_1 \hat{\theta} + \psi_2\left(\hat{\xi}_1 - \alpha_1(\xi_2, u) + \lambda_2 \hat{\theta}\right) \\ &\quad -\frac{\partial \alpha_1^*}{\partial u}\dot{u} - \frac{\partial \alpha_1^*}{\partial \hat{\theta}}\tau_2 - \frac{\partial \alpha_1^*}{\partial \xi_r}\dot{\xi}_r - \frac{\partial \alpha_1^*}{\partial \dot{\xi}_r}\ddot{\xi}_r \end{aligned} \tag{9.108}$$

where $-d_2\psi_2^2 z_2$ is the second nonlinear damping term to compensate the destabilizing effect of the observation error $e_1$. Then

$$\begin{aligned} \dot{V}_2 &\leqslant -c_1 z_1^2 - c_2 z_2^2 - d_2\psi_2^2 z_2^2 + \psi_2 e_1 + \frac{e_1^2}{4d_1} \\ &= -c_1 z_1^2 - c_2 z_2^2 - d_2\left(\psi_2 z_2 - \frac{e_1}{2d_2}\right)^2 + \frac{e_1^2}{4d_2} + \frac{e_1^2}{4d_1} \\ &\leqslant -c_1 z_1^2 - c_2 z_2^2 + \frac{1}{4}\left(\frac{1}{d_1} + \frac{1}{d_2}\right)e_1^2 \end{aligned} \tag{9.109}$$

Augmenting the Lyapunov function with a quadratic term defined on the observation error vector

$$V = V_2 + \left(\frac{1}{d_1} + \frac{1}{d_2}\right)e^T P e \tag{9.110}$$

we obtain

$$\dot{V} \leqslant -c_1 z_1^2 - c_2 z_2^2 - \frac{3}{4}\left(\frac{1}{d_1}+\frac{1}{d_2}\right)e_1^2 - \left(\frac{1}{d_1}+\frac{1}{d_2}\right)e_2^2 + 2\left(\frac{1}{d_1}+\frac{1}{d_2}\right)e^T P\Lambda\tilde{\theta} \quad (9.111)$$

which shows that the passivity property is preserved. Therefore boundedness of all variables of the overall closed-loop system, consisting of the controlled plant, the observer, the controller and the parameter estimator, is guaranteed. This is illustrated in the following example.

**Example 9.3** Consider the following second order uncertain system

$$\begin{aligned} \dot{x}_1 &= -x_1 + x_2^2 + u + \theta x_2^4 \\ \dot{x}_2 &= x_1 x_2 + u + \theta x_2^2 \\ y &= x_2 \end{aligned} \quad (9.112)$$

where $\theta$ is a constant unknown parameter. Note that the nominal system (obtained when $\theta = 0$) coincides with the system (9.31) of Examples 9.1 and 9.2. Therefore, applying the transformation (9.36) to (9.112), yields

$$\begin{aligned} \dot{\xi}_1 &= \frac{u}{y} + y^2 + u + \theta(y + y^4) \\ \dot{\xi}_2 &= \xi_1 - \ln y + \frac{u}{y} + \theta y \\ y_\xi &= \xi_2 \end{aligned} \quad (9.113)$$

or, fully transformed into the $\xi$ coordinates

$$\begin{aligned} \dot{\xi}_1 &= u\exp(-\xi_2) + \exp(2\xi_2) + u + \theta(\exp(\xi_2) + \exp(4\xi_2)) \\ \dot{\xi}_2 &= \xi_1 - \xi_2 + u\exp(-\xi_2) + \theta\exp(\xi_2) \\ y_\xi &= \xi_2 \end{aligned} \quad (9.114)$$

which is obviously in the *adaptive GORCF* (9.83) with

$$\begin{aligned} \alpha_0(\xi_2, u) &= -u\exp(-\xi_2) - \exp(2\xi_2) - u \\ \alpha_1(\xi_2, u) &= \xi_2 - u\exp(-\xi_2) \\ \lambda_1(\xi_2) &= \exp(\xi_2) + \exp(4\xi_2) \\ \lambda_2(\xi_2) &= \exp(\xi_2) \end{aligned} \quad (9.115)$$

The observer is given by

$$\begin{aligned} \dot{\hat{\xi}}_1 &= u\exp(-\xi_2) + \exp(2\xi_2) + u + \hat{\theta}(\exp(\xi_2) + \exp(4\xi_2)) + k_1(\xi_2 - \hat{\xi}_2) \\ \dot{\hat{\xi}}_2 &= \hat{\xi}_1 - \xi_2 + u\exp(-\xi_2) + \hat{\theta}\exp(\xi_2) + k_2(\xi_2 - \hat{\xi}_2) \\ y_{\hat{\xi}} &= \hat{\xi}_2 \end{aligned} \quad (9.116)$$

After applying the systematic procedure described above for tracking a desired output $\xi_r$, we obtain the error coordinate transformation

$$z_1 = \xi_2 - \xi_r \tag{9.117}$$

$$z_2 = \hat{\xi}_1 - \alpha_1^*(\xi_2, u, \hat{\theta}, \xi_r, \dot{\xi}_r) \tag{9.118}$$

with

$$\alpha_1^* = \xi_2 - u\exp(-\xi_2) - \hat{\theta}\exp(\xi_2) + \dot{\xi}_r - (c_1 + d_1)(\xi_2 - \xi_r) \tag{9.119}$$

The update law for the unknown parameter yields

$$\dot{\hat{\theta}} = \gamma\lambda_2(\xi_2)(e_2 + z_1 + \psi_2 z_2) \tag{9.120}$$

with $\gamma$ a positive adaptation gain and

$$\psi_2 := -\frac{\partial\alpha_1^*}{\partial\xi_2} = -1 - u\exp(-\xi_2) + \hat{\theta}\exp(\xi_2) + (c_1 + d_1). \tag{9.121}$$

The dynamical control law is

$$\begin{aligned} \dot{u} = \exp(\xi_2)\Big[ & -z_1 - u\exp(-\xi_2) - \exp(2\xi_2) - u - k_1(\xi_2 - \hat{\xi}_2) - \lambda_1(\xi_2)\hat{\theta} \\ & -\psi_2\Big(\hat{\xi}_1 - \xi_2 + u\exp(-\xi_2) + \hat{\theta}\exp(\xi_2)\Big) - \exp(\xi_2)\tau_2 \\ & +\ddot{\xi}_r + (c_1 + d_1)\dot{\xi}_r - c_2 z_2 - d_2\psi_2^2 z_2\Big] \end{aligned} \tag{9.122}$$

Computer simulations were carried out to illustrate the performance of the dynamical adaptive backstepping control/observer in tracking a desired reference, corresponding to a smooth transition of $x_2$ from $X_2 = 1.5$ to $X_2^* = 3$ for a nominal unknown parameter $\theta = -1$. Since we have a scalar unknown parameter and

$$\lambda_2(X_2^*) = \exp(3) \neq 0 \tag{9.123}$$

the convergence of $\hat{\theta}$ to the unknown parameter value is guaranteed and asymptotic stability of the overall closed-loop system is achieved, as shown in Figure 9.1.

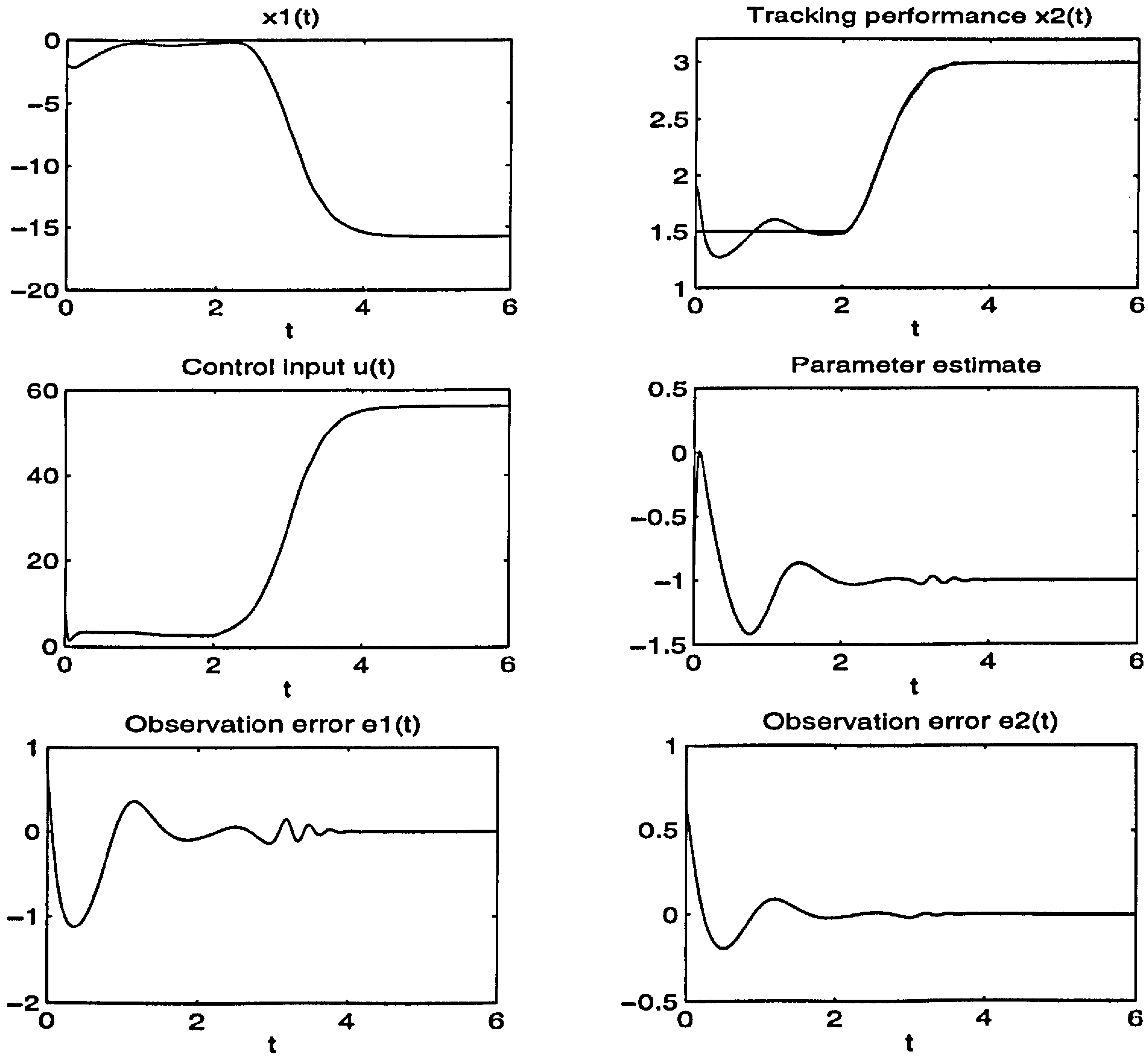


Figure 9.1: Response of $x_1$, tracking performance, control input, observation errors and parameter estimate

# Chapter 10

# Conclusions and Suggestions for Further Research

## 10.1 Conclusions

In this thesis we have studied the control design for the regulation and tracking of uncertain nonlinear systems when no information is available regarding the uncertain parameters. We have adopted the backstepping approach in dealing with these problems, because it is a systematic control design procedure which does not require matching conditions on the uncertain parameters. Additionally the proof of stability, based upon the use of a quadratic Lyapunov function, is simple and constructive.

We have proposed a number of new systematic control design algorithms without the need for the controlled plant to be transformable into canonical forms. Instead of canonical forms, we require that the uncertain nonlinear system be observable and minimum phase, for which a dynamic (static) adaptive (deterministic) controller is designed. We have also combined dynamic (static) adaptive backstepping with sliding mode control via the DAB-SMC (SAB-SMC) algorithm to provide robustness in the presence of disturbances. Some of the benefits achieved by these new backstepping algorithms are:

- the availability of a number of alternative recursive control design algorithms for deterministic and adaptive nonlinear systems

- the DAB algorithm does not require that the system be transformable into either the PSF or PPF (triangular) forms and is applicable to uncertain nontriangular systems; this result extends the applicability of the backstepping approach to a broader class of observable minimum phase uncertain nonlinear systems

- dynamical adaptive controllers obtained using the DAB algorithm may exhibit better performance than classical static backstepping controllers designed via the SAB algorithm, when the differential equations characterizing the zero dynamics contain uncertain parameters, as shown in Section 4.2.
- the adaptive sliding mode controllers designed via the combined DAB-SMC, exhibit excellent robustness properties with considerably reduced chattering in the presence of undesirable disturbances.

A number of examples and simulations have been studied to evaluate the performance of the new backstepping algorithms in regulation and tracking problems. Some applications of practical interest have been also studied. For instance, the regulation of a flexible-joint manipulator, the PWM control regulation of DC-to-DC power converters, and tracking tasks of a field-controlled DC motor have been studied to illustrate the SAB, DAB and DAB-SMC algorithms, respectively.

Even though the recursive design procedures described in this thesis follow systematic step-by-step algorithms, the equations are usually too complicated for hand computation. For this reason we have developed a symbolic algebra computation toolbox which implements the various backstepping design algorithms and computes the required controller obviating the likely errors of hand computation. This toolbox does not require the user to have expert knowledge of the backstepping design technique and automatically generates MATLAB code programs for computer simulation of the closed-loop systems.

In the more realistic context of output feedback control design, when only the output is measured, we have proposed a technique which employs a state observer, a parameter estimator and a dynamical adaptive backstepping controller for a more restricted class of observable minimum phase uncertain systems transformable into the *adaptive generalized observer canonical form*. The output tracking problem may be solved via this approach with boundedness of the variables of the overall closed-loop system. Also the conditions to guarantee asymptotic stability (rather than boundedness) have been analysed.

## 10.2 Suggestions for Further Research

There are several important research directions which should be pursued in the context of the results obtained in this thesis. Some of these are:

### 10.2.1 Extension to Non-Affine Multi-Input Multi-Output Systems

We have studied the design of deterministic (adaptive) control of uncertain affine single-input single-output nonlinear systems. These results should be extended to non-affine multi-input multi-output uncertain nonlinear systems where the theory of *flat systems* proposed by Fliess and co-workers [29] provides a convenient setting.

### 10.2.2 Non-Minimum Phase Uncertain Systems

We have extended in this thesis the applicability of the backstepping approach to a class of observable minimum phase uncertain nonlinear systems. On the other hand some discontinuous control design techniques have recently been proposed by Spurgeon and Lu [115] and Llanes-Santiago and Sira-Ramírez [70] to deal with deterministic non-minimum phase systems. The extension of our results to non-minimum phase systems with uncertain parameters, is a very important problem which could be addressed by the combination of Adaptive Backstepping and SMC.

### 10.2.3 Bounding Functions

We have assumed in this thesis that no information is available regarding the uncertain parameters. This assumption could be removed and one could study uncertainty arising from unknown *bounded* functions or (and) time-varying parameters for which the use of known bounding functions is suggested for the design of the stabilizing control law. In this context the recent results of Freeman and Kokotović [31, 32] and Qu [85] give some additional insight.

### 10.2.4 Extensions in Output Feedback Control

Further generalizations of the second order cases analysed in Chapter 9 may be developed. Also the class of observable minimum phase uncertain nonlinear systems transformable into the *adaptive generalized observer canonical form* (considered in Chapter 9) is rather restrictive. Therefore, a challenging problem is to broaden the class of uncertain nonlinear systems which can be tackled with the control design algorithm proposed in this thesis in the context of output feedback control.

### 10.2.5 Extension to the Discrete-Time Case

The extension of the systematic design algorithms proposed in this thesis to discrete-time systems should be addressed. Additional difficulties arise in this context. Two of them are:

1. the geometric characterization of the classes of nonlinear discrete-time systems which can be stabilized following the systematic procedure

2. even though the plant appears linearly parameterized, terms with nonlinear dependence on the estimate parameters may arise from the direct application of the backstepping approach.

### 10.2.6 Analysis Tools in the Symbolic Toolbox

The symbolic algebra toolbox BACKDSMC implements the various backstepping control design algorithms proposed in this thesis. However, BACKDSMC is so far a design technique only, i.e. it does not incorporate tools for analysis. It would be useful to incorporate tools for the analysis of the uncertain plants with regard to the zero dynamics, transforming the plant into certain convenient canonical forms, establishing the region of minimum phase, determining the region of observability, etc.

### 10.2.7 Simplification of the Symbolic Algebra Output

Usually the expressions characterizing the backstepping controllers obtained from the application of the toolbox BACKDSMC, are very complicated. Additionally, the tools for algebraic simplification available in the MATLAB Symbolic Toolbox often fail to obtain expressions which have been "optimally" simplified. This matter needs attention with regard to practical applications.

### 10.2.8 Simplification of the Control Laws

Further research should be performed concerning the adaptation of the approach to yield less complicated control laws. For instance, one could investigate whether or not the sliding mode control technique gives simpler control laws.

### 10.2.9 Links with Passivity

As demonstrated in Chapter 9, the backstepping design approach is closely related to passivity. Further investigations should be carried out to study these links and their use for the development of new control design techniques. Some new control design procedures using passivity have been proposed in the recently published book by Sepulchre *et al* [99].

# Appendix A

# Basics of Stability Theory

A time-varying dynamic system can be represented by a set of nonlinear differential equations of the form

$$\dot{x} = f(x,t) \tag{A.1}$$

where $x \in \Re^n$ and $f : \Re^n \times \Re_+ \to \Re^n$ is a piecewise continuous in $t$ and Lipschitz in $x$. A solution of (A.1) which starts from the point $x_0$ at time $t_0 \geqslant 0$, denoted as $x(t; x_0, t_0)$, is generally referred to as a *state trajectory* or a *system trajectory.* It is possible for a system trajectory to correspond to only a single point. Such a point is called an *equilibrium point.*

**Definition A.1** *A state $X$ is an equilibrium point of the system if once $x(t; X, t_0)$ is equal to $X$, it remains equal to $X$ for all future time.*

Mathematically this means that the vector $X$ satisfies

$$f(X,t) \equiv 0 \qquad \forall t \geqslant t_0 \tag{A.2}$$

A nonlinear system of the form (A.1) may have several (or many) isolated equilibrium points. Any equilibrium can be translated to the origin 0 by redefining the state $x$ as $z = x - X$, so that the solution under investigation can always be considered to be an equilibrium at the origin. Lyapunov stability concepts describe continuity properties of $x(t; x_0, t_0)$ with respect to the initial state $x_0$.

**Definition A.2** *The equilibrium point* 0 *is* stable *at $t_0$ if for any $\varepsilon > 0$ there exists a $\delta(\varepsilon, t_0)$ such that*

$$\|x_0\| < \delta(\varepsilon, t_0) \Rightarrow \|x(t; x_0, t_0)\| < \varepsilon, \qquad \forall t \geqslant t_0 \tag{A.3}$$

*otherwise, the equilibrium point* 0 *is* unstable.

This definition means that one can keep the state in a ball of arbitrarily small radius $\varepsilon$ by starting the state trajectory in a ball of sufficiently small radius $\delta$.

**Definition A.3** *The equilibrium point* 0 *is* asymptotically stable *at time* $t_0$ *if it is stable and, additionally, there exists an* $r(t_0) > 0$ *such that*

$$\|x_0\| < r(t_0) \Rightarrow \|x(t; x_0, t_0)\| \to 0 \text{ as } t \to \infty. \tag{A.4}$$

If an equilibrium point $X$ is asymptotically stable, it has a *region of attraction*, i.e. a set $\Omega$ of initial states $x_0$ such that $x(t; x_0, t_0) \to X$ as $t \to \infty$. In general, stability properties of nonlinear systems are valid for *local* attraction regions. When the region of attraction is the whole space $\Re^n$, then stability properties are *global.*

**Definition A.4** *The equilibrium point* 0 *is* exponentially stable *if there exists two positive numbers,* $\alpha$ *and* $\lambda$, *such that for sufficiently small* $x_0$,

$$\|x(t; x_0, t_0)\| \leqslant \alpha \|x_0\| e^{-\lambda(t-t_0)}, \qquad \forall t \geqslant t_0. \tag{A.5}$$

The stability properties of *non-autonomous* (time-dependent) systems of the form (A.1) in general depend on the initial time $t_0$. For different $t_0$, different values of $\delta(\varepsilon, t_0)$ and $r(t_0)$ may be required to satisfy the conditions in (A.3) and (A.4) respectively. For practical purposes, it is desirable for the system to have a certain uniformity in its behaviour regardless of when the operation starts. Thus, the equilibrium point 0 in definitions A.3 and A.4 is said to be *uniform stable* and *uniform asymptotic stable* respectively, when $\delta(\varepsilon, t_0) = \delta(\varepsilon)$ and $r(t_0) = r$, i.e. $\delta$ and $r$ are independent of $t_0$.

For regulation tasks the designed system is usually *autonomous* (time-invariant)

$$\dot{x} = f(x). \tag{A.6}$$

The stability properties of autonomous systems are uniform. Also the Lyapunov stability theorems for this kind of systems are easily formulated.

**Theorem A.1** *Let* $x = 0$ *be an equilibrium point of (A.6). If in a ball* $\Omega$ *around the origin, there exists a continuously differentiable positive definite function* $V(x) : \Re^n \to \Re_+$ *such that*

$$\dot{V}(x) = \frac{\partial V}{\partial x} f(x) \leqslant 0, \qquad \forall x \in \Omega, \tag{A.7}$$

*then the equilibrium point* 0 *is locally stable. If the derivative* $V(x)$ *is locally negative definite, i.e.*

$$\dot{V}(0) = 0, \qquad \dot{V}(x) = \frac{\partial V}{\partial x} f(x) < 0, \qquad \forall x \neq 0 \in \Omega, \tag{A.8}$$

*then the equilibrium is asymptotically stable.*

The scalar function $V(x)$ in Theorem A.1 is called a *Lyapunov function.* When $\Omega$ is the whole space $\Re^n$ and the Lyapunov function $V(x)$ is *radially unbounded*, i.e.

$$V(x) \to \infty \quad as \quad \|x\| \to \infty, \tag{A.9}$$

then the equilibrium at the origin is *globally asymptotically stable.*

For autonomous systems *invariant sets* are very important in checking the asymptotic stability of a control system. This is because it often happens that the derivative of the Lyapunov function candidate is only negative semi-definite. In this situation it is still possible to draw conclusions on asymptotic stability with the help of LaSalle's invariance theorem.

**Definition A.5** *A set $M$ is an* invariant set *of (A.6) if any solution $x(t)$ that belongs to $M$ at some time instant $t_1$, must belong to $M$ for all future and past time, i.e.*

$$x(t_1) \in M \Rightarrow x(t) \in M, \quad \forall t \in \Re. \tag{A.10}$$

*A set $\Omega$ is* positively invariant *if this is true for all future time only:*

$$x(t_1) \in \Omega \Rightarrow x(t) \in \Omega, \quad \forall t \geqslant t_1. \tag{A.11}$$

Examples of invariant sets are equilibrium points, their attraction domains and limit cycles.

**Theorem A.2 (LaSalle)** *Let $\Omega$ be a bounded closed (compact) positively invariant set of an autonomous system of the form (A.6). Let $V : \Omega \to \Re_+$ be a continuously differentiable function $V(x)$ such that $\dot{V}(x) \leqslant 0, \quad \forall x \in \Omega$. Let $E = \{x \in \Omega \mid \dot{V}(x) = 0\}$ and let $M$ be the largest invariant set contained in $E$. Then every bounded solution $x(t)$ starting in $\Omega$ converges to $M$ as $t \to \infty$.*

**Corollary A.1 (Global Asymptotic Stability)** *Let $x = 0$ be the* only *equilibrium point of (A.6). Let $V : \Re^n \to \Re_+$ be a continuously differentiable positive definite radially unbounded function $V(x)$ such that $\dot{V}(x) \leqslant 0$, $\forall x \in \Re^n$. Let $E = \{x \in \Re^n \mid \dot{V}(x) = 0\}$, and suppose that no solution other than $x(t) \equiv 0$ can stay forever in $E$. Then the origin is* globally asymptotically stable.

Considering these invariance results, the most favourable case regarding asymptotic stability corresponds to the case when the largest invariant subset $M$ of $E$ is just the origin $x = 0$.

For non-autonomous systems, the main tools to assert convergence of state trajectories are Barbalat's Lemma and the LaSalle-Yoshizawa Theorem. Stability definitions are firstly restated in terms of the so-called class-$K$ functions.

**Definition A.6** *A continuous function* $\gamma : [0, a) \to \Re_+$ *is said to belong to the class* $K$ *if it is strictly increasing and* $\gamma(0) = 0$. *It is said to belong to class* $K_\infty$ *if* $a = \infty$ *and* $\gamma(r) \to \infty$ *as* $r \to \infty$.

**Definition A.7** *A continuous function* $\beta : [0, a) \times \Re_+ \to \Re_+$ *is said to belong to the class* $KL$ *if for each fixed* $s$ *the mapping* $\beta(r, s)$ *belongs to* $K$ *with respect to* $r$, *and for each fixed* $r$ *the mapping* $\beta(r, s)$ *is decreasing with respect to* $s$ *and* $\beta(r, s) \to 0$ *as* $s \to \infty$. *It is said to belong to the class* $KL_\infty$ *if in addition for each fixed* $s$ *the mapping* $\beta(r, s)$ *belongs to the class* $K_\infty$ *with respect to* $r$.

**Definition A.8** *The equilibrium point* $x = 0$ *of (A.1) is*

- uniformly stable, *if there exists a class* $K$ *function* $\gamma(\cdot)$ *and a positive constant* $c$ *independent of* $t_0$ *such that*

$$\|x(t)\| \leqslant \gamma(\|x(t_0)\|), \quad \forall t \geqslant t_0 \geqslant 0, \ \forall x(t_0) \mid \|x(t_0)\| < c; \tag{A.12}$$

- uniformly asymptotically stable, *if there exists a class* $KL$ *function* $\beta(\cdot, \cdot)$ *and a positive contant* $c$ *independent of* $t_0$ *such that*

$$\|x(t)\| \leqslant \beta(\|x(t_0)\|, t - t_0), \quad \forall t \geqslant t_0 \geqslant 0, \ \forall x(t_0) \mid \|x(t_0)\| < c; \tag{A.13}$$

- exponentially stable, *if (A.13) is satisfied with* $\beta(r, s) = kre^{-\alpha s}$, $k > 0$, $\alpha > 0$.

If the conditions in Definition A.8 are satisfied for any initial state $x(t_0)$ and in addition $\gamma \in K_\infty$ and $\beta \in KL_\infty$, the stability properties are said to be *global.*

**Theorem A.3 (LaSalle-Yoshizawa)** *Let* $x = 0$ *be an equilibrium point of (A.1) and suppose* $f$ *is locally Lipschitz in* $x$ *uniformly in* $t$. *Let* $V : \Re^n \times \Re_+ \to \Re_+$ *be a continuously differentiable function such that*

$$\gamma_1(\|x\|) \leqslant V(x, t) \leqslant \gamma_2(\|x\|) \tag{A.14}$$

$$\dot{V} = \frac{\partial V}{\partial t} + \frac{\partial V}{\partial x} f(x, t) \leqslant -W(x) \leqslant 0 \tag{A.15}$$

$\forall t \geqslant 0$, $\forall x \in \Re^n$, *where* $\gamma_1$ *and* $\gamma_2$ *are class* $K_\infty$ *functions and* $W$ *is a continuous function. Then, all solutions of (A.1) are globally uniformly bounded and satisfy*

$$\lim_{t \to \infty} W(x(t)) = 0. \tag{A.16}$$

*In addition, if* $W(x)$ *is positive definite, then the equilibrium* $x = 0$ *is globally uniformly asymptotically stable.*

# Appendix B

# MATLAB Code Program of the Symbolic Toolbox BACKDSMC

```
%
%    Toolbox BACKDSMC
%
%    This toolbox allows the design of deterministic (adaptive)
%    backstepping controllers or combined backstepping-sliding mode
%    controllers (SMC) for observable minimum phase nonlinear systems.
%
%    Programmed by M. Rios-Bolivar      Date: 24/10/96
%
function [control,tau,z,surface]=backdsgn(f,g,h,yd,phi,psi,modfile,runfile);
%
% BACKDSMC  Symbolic backstepping control design via input-output
%     linearization for uncertain nonlinear systems. This program
%     allows the design of either dynamical adaptive backstepping
%     control or a combined dynamical adaptive backstepping with
%     sliding mode control for uncertain nonlinear systems.
%
% [C,TAU]=BACKDSMC(F,G,H,Yd,Phi,Psi) designs the adaptive
%     backstepping controller for the uncertain nonlinear system
%
%      \dot{x}=F(x)+G(x)*u+(Phi(x)+u*Psi(x))*THETA
%            y=H(x)
%
%     where F and G are vector fields, Phi and Psi are matrices
%     of appropriate dimensions, THETA is a vector of unknown parameters,
%     H is a scalar output function, and  Yd is the desired output.
%     (If Yd is a function of time, a tracking controller is designed).
%     C is the designed control and TAU is the update law for the
%     unknown parameters.
%     If the relative degree r is less than the system order n, the
%     controller is dynamic and C corresponds to the  (n-r)-th
%     derivative of u.
%     Note: When all entries of both Phi and Psi are zero,
%       a non-adaptive control is designed.
%
% [C,TAU,Z]=BACKDSMC(F,G,H,Yd,Phi,Psi) gives, additionally, the
%     tranformation Z placing the original system into error coordinates.
%
% [C,TAU,Z]=BACKDSMC(F,G,H,Yd,Phi,Psi,'modfile') generates the
%     m-file 'modfile.m' containing the system model and the adaptive
%     controller for numerical simulation purposes.
%
% [C,TAU,Z]=BACKDSMC(F,G,H,Yd,Phi,Psi,'modfile','runfile') generates,
%     additionally, the m-file 'runfile.m', which runs 'modfile.m' and
%     provides nominal values of the unknown parameters, design parameter
%     values and initial states for simulation purposes.
```

```
%
% [C,TAU,Z,SIGMA]=BACKDSMC(F,G,H,Yd,Phi,Psi,'modfile','runfile'),
%     i.e. containing a fourth output argument, allows to design
%     an adaptive variable structure control to generate a SLIDING MODE
%     on the sliding surface SIGMA=k_1*z_1+k_2*z_2+...k_n-1*z_n-1+z_n,
%     with k_i's the design parameters and z_i's the error coordinates
%     of the transformation Z.
%

%
% Checking dimensions
%
chk=find([6,7,8] == nargin);
if isempty(chk)
    error('Number of input arguments is incorrect')
end
chk=find([1,2,3,4] == nargout);
if isempty(chk)
    error('Number of output arguments is incorrect')
end
if nargout==4
   smc_design=1;      % Sliding Mode Control Design
else
   smc_design=0;
end
df=symsize(f);
dg=symsize(g);
dh=symsize(h);
dyd=symsize(yd);
n=df(1);                    % system order
dphi=symsize(phi);
dpsi=symsize(psi);
p=dphi(2);                  % No. of unknown parameters
%
% check dimensions matching
%
if (df ~= dg) | (dphi ~= dpsi) | (dh ~= [1,1]) | (dh ~= dyd) ...
  | (df(1) ~= dphi(1)) error('Inner matrix dimensions must agree')
end
%
% initialization of the states vector [x1,x2,...,xn]
%
x='x1';
for i=2:n x=[x,',x',int2str(i)]; end
%
% initialization of the vector of unknowns thetahat=[th1,th2,...,thp]
```

```
%
thhat='';
theta='';
adgain=zeros(p);
for i=1:p
    adgain=sym(adgain,i,i,['ad(' ,num2str(i),')']);
    if i==1
        thhat='th1';
        theta='theta(1)';
   else
        thhat=[thhat,',th',num2str(i)];
        theta=[theta,',theta(',num2str(i),')'];
    end
end
u='u1';                              % define u as a symbolic variable
wr=symop(phi,'+',psi,'*',u);          % auxiliar vector wr=phi+psi*u
%
% estimate of the system xhat=f(x)+g(x)u+(phi(x)+psi(x))*theta
%
xhat=symop(f,'+',g,'*',u,'+',wr,'*',transpose(sym(thhat)));
%
% initialization of the relative degrees with respect to theta (hat) and
% the control u (cont) respectively
%
hat=-1;
cont=-1;
tau=sym(p,1,'0');   % initialization of the tuning function
z=symop(h,'-',yd);  % define the first state coordinate transformation
yr=yd;
%
% initialization of a symbolic vector containing the control and its
% derivatives
%
uhat=u;
%
% Design loop
%
for i=1:n
    yd=diff(yd,'t');          % i-th derivative of yd(t)
    zaux=sym(sym(z),1,i);     % previous coordinate transformation
    dzdu='0';                 % initial dz/duhat
%
% check out the presence of theta in z(i) coordinate
%
    if hat==-1
      if findstr('th',zaux) ~= []
```

```
        hat=i-1;              % relative degree respect to theta
      end
    end
%
% check out the presence of u in z(i) coordinate
%
    if cont==-1
      if findstr(u,zaux) ~= []
        cont=i-1;              % relative degree respect to u
        udot='u2';           % differential du/dt
%
% obtain dz/du(du/dt) first time
%
        dzdu=symop(dzdu,'+',diff(zaux,uhat),'*',udot);
        uhat=[uhat,',',udot]; % incorporate du/dt to uhat
      end
    else
%
% vector containing derivatives of u only
%
      udot=transpose(sym([uhat(length(u)+2:size(uhat,2)),',u',...
           num2str(i-cont+1)]));
%
% obtain dz/duhat(duhat/dt) with uhat as a vector
%
      dzdu=symop(dzdu,'+',jacobian(zaux,sym(uhat)),'*',udot);
      uhat=[uhat,',u',num2str(i-cont+1)]; % new derivative of u
    end
    dzdx=jacobian(zaux,sym(x));   % partial dz/dx
%
% obtain partial dz/dthetahat
%
    if p==1
      dzdthhat=diff(zaux,thhat);
    else
      dzdthhat=jacobian(zaux,sym(thhat));
    end
    w=symop(transpose(wr),'*',transpose(dzdx));  % regressor vector
    if  (i==n) & smc_design
       k='k(1)';
       for j=2:n-1  k=[k,',k(',int2str(j),')']; end
       k=sym([k,',1']);
       surface=symop(k,'*',transpose(sym(z)));
       sigma=sym('sigma');
       wadd=sym(1,p,'0');
        for j=1:n-1
```

```
     wadd=symop(wadd,'+',sym(k,1,j),'*',wt(j,:));
        end
        regaux=symop(adgain,'*', '(',w,'+',transpose(wadd),')');
        tau=symop(tau,'+',sigma,'*',regaux);
    else
        tau=symop(tau,'+',zaux,'*',adgain,'*',w);  % tuning function
    end
    if i==1
        wt=transpose(w);
        dzdth=dzdthhat;
        taut=transpose(tau);
    else
        wt=symadrow(wt,transpose(w));
        dzdth=symadrow(dzdth,dzdthhat);
        taut=symadrow(taut,transpose(tau));
    end
    dzdt=diff(zaux,'t');
    sdzdth='0';
    if  i >= 3
       sdzdth=symop(symvxm(sym(z),dzdth,1,i-1),'*',adgain,'*',w);
    end
%
% obtain coordinate z to be added at this step
%
    if i >= 2
      sz=sym(sym(z),1,i-1);       % coordinate z(i-1)
    else
      sz='0';
    end
    c=['c(',num2str(i),')'];      % control gain
    znext=symop(dzdx,'*',xhat,'+',dzdthhat,'*',tau,'+',dzdu);
    if  ((i < n) | ~(smc_design))
         znext=symop(znext,'+',c,'*',zaux,'+',sz);
    end
    znext=symop(znext,'+',dzdt,'+',sdzdth);   % z(i+1)
    if i < n
      z=[z,',',znext];
      yr=[yr,',',yd];
    end
end                          % end of the design loop

yr=transpose(sym(yr));     % transform yr to a symbolic vector
z=transpose(sym(z));        % transform z to a symbolic vector
%
%  Sliding Mode Control Design
%
```

```
if  smc_design
   dsigmadt=symop(symvxm(transpose(z),dzdth,1,n-1),'*',adgain,...
'*',transpose(wadd));
   for i=1:n-1
      c=['c(',num2str(i),')'];     % control gain
      sind=symop(sym(z,i+1,1),'-',c,'*',sym(z,i,1),'+',...
           dzdth(i,:),'*','(',tau,'-',transpose(taut(i,:)),')');
      if i > 1
          sind=symop(sind,'-',sym(z,i-1,1),'-',...
                     symvxm(transpose(z),dzdth,1,i-1),...
                     '*',adgain,'*',transpose(wt(i,:)));
      end
      sind=symop(sym(k,1,i),'*',sind);
      dsigmadt=symop(dsigmadt,'+',sind);
   end
else
   dsigmadt='0';       %  No SMC design
end
znext=symop(znext,'+',dsigmadt);       % Add dsigma/dt

comm=findstr(',',uhat);    % find the positions of ',' in uhat
%
% Obtain substrings containing [u1,u2,...] and [u2,u3,...]
% to be used in the final control
%
if comm ~= []
    lcom=length(comm);
    if lcom > 2
      subhat=transpose(sym(uhat(1:comm(lcom-1)-1)));
      subdot=transpose(sym(uhat(length(u)+2:comm(lcom)-1)));
    elseif lcom == 2
      subhat=uhat(1);
      subdot=uhat(length(u)+2:comm(lcom)-1);
    end
end
uhat=transpose(sym(uhat));    % transform uhat to a symbolic vector
thhat=transpose(sym(thhat));  % transform thhat to a symbolic vector
%
% Obtain the final control
%
if cont==-1
     if findstr(u,znext) ~= []
       cont=n;                        % relative degree n
     end
end
if cont==-1
```

```
  error('control is not available')
elseif cont==n                  % case of linearizable system
  den=sym(symop(sym(dzdx,1,n),'*','(',g,'+',psi,'*',thhat,')'),n,1);
  sum=expand(symop(den,'*',u));
  control=symop(sum,'-',(znext));
else                            % case of no linearizable system
  den=diff(sym(z,n,1),sym(uhat,symsize(uhat,1)-1,1)); %denominator
  if n-cont == 1    % case of relative degree n-1
    dzdu='0';
  else
    if symsize(subhat,1) ~= 1
      dzdu=symop(jacobian(sym(z,n,1),subhat),'*',subdot);
    else
      dzdu=symop(diff(sym(z,n,1),subhat),'*',subdot);
    end
  end
  control=symop('-',dzdx,'*',xhat,'-',dzdthhat,'*',tau,'-',dzdu);
  if  ~smc_design
     control=symop(control,'-',c,'*',zaux,'-',sz);
  end
  control=symop(control,'-',dzdt,'-',sdzdth,'-',dsigmadt);
end
if  smc_design
   disigma='lambda*(sigma+beta*sign(sigma))';
   control=['(',control,'-',disigma,')/',den];
else
   control=symop(control,'/',den);
end
%dispform(control,80);
%
%
%   M-file creation
%
%
if nargin > 6
   fid=fopen([modfile,'.m'],'w','n');  %open file
   fprintf(fid,['function xdot=',modfile,'(t,x);\n']);
   if strcmp(tau,sym(p,1,'0'))==0
        adapt=1;
     else
        adapt=0;
         p=0;
   end
   fprintf(fid,'global c');
   if adapt
       fprintf(fid,' ad theta');
```

```
    end
    if  smc_design
        fprintf(fid,' k lambda beta');
    end
    fprintf(fid,';\n');
%
% the system x=f(x)+g(x)u+(phi(x)+psi(x))*theta
%
    xhat=symop(f,'+',g,'*',u,'+',wr,'*',transpose(sym(theta)));
%
    if  adapt | (cont < n)
        fprintf(fid,'%c\n%s\n%c\n','%','%   Auxiliary variables','%');
    end
    for i=1:n-cont
        fprintf(fid,['u',num2str(i),'=x(',num2str(n+p+i),');\n']);
    end;
    if adapt
        for i=1:p
            fprintf(fid,['th',num2str(i),'=x(',num2str(n+i),');\n']);
        end;
    end
    if  smc_design
      fprintf(fid,'%c\n%s\n%c\n','%','%   Sliding surface','%');
      fprintf(fid,'sigma=');
      cad=surface;
      for j=1:n
          cad=strrep(cad,['x',num2str(j)],['x(',num2str(j),')']);
      end
      wrecback(fid,cad);
    end
    if adapt
        fprintf(fid,'%c\n%s\n%c\n','%','%   Update law','%');
        for i=1:p
            fprintf(fid,['tau',num2str(i),'=']);
            cad=sym(tau,i,1);
            for j=1:n
                cad=strrep(cad,['x',num2str(j)],['x(',num2str(j),')']);
            end
            wrecback(fid,cad);
        end
    end
    fprintf(fid,'%c\n%s\n%c\n','%','%   Control law','%');
    cad=control;
    for i=1:n
        cad=strrep(cad,['x',num2str(i)],['x(',num2str(i),')']);
    end
```

```
    if n==cont
         fprintf(fid,[u,'=']);
    else
         fprintf(fid,'control=');
    end
    wrecback(fid,cad);
    fprintf(fid,'%c\n%s\n%c\n','%','%    System equations','%');
    for i=1:n
         fprintf(fid,['xdot(',num2str(i),')=']);
         cad=sym(xhat,i,1);
         for j=1:n
            cad=strrep(cad,['x',num2str(j)],['x(',num2str(j),')']);
         end
         wrecback(fid,cad);
    end;
    if adapt
         fprintf(fid,'%c\n%s\n%c\n','%',...
                 '%    Parameter estimate equations','%');
         for i=1:p
            fprintf(fid,['xdot(',num2str(n+i),')=tau',num2str(i),';\n']);
         end
    end
    if n > cont
        fprintf(fid,'%c\n%s\n%c\n','%',...
                '%   Dynamic control equations','%');
    end
    for i=1:n-cont-1
       fprintf(fid,['xdot(',num2str(n+p+i),')=x(',num2str(n+p+i+1),');\n']);
    end
    if n > cont
       fprintf(fid,['xdot(',num2str(2*n+p-cont),')=control;\n']);
    end
    fclose(fid);
end
%
%
%      Creation of  main M-file
%
%
if nargin == 8
    fid=fopen([runfile,'.m'],'w','n');  %open file
    fprintf(fid,'%c\n%s\n%c\n','%',...
            ['% This program runs ',modfile,'.m'],'%');
    fprintf(fid,'global c');
    if adapt
        fprintf(fid,' ad theta');
```

```
    end
    if  smc_design
        fprintf(fid,' k lambda beta');
    end
    fprintf(fid,';\n');
    fprintf(fid,'%c\n%s\n%c\n','%','%     Parameter values','%');
    blk=blanks(n-1);
    blk=strrep(blk,' ',' ;');
    if  smc_design
       fprintf(fid,['k=[',blk,'1];\n']);
       blk=blanks(n-2);
       blk=strrep(blk,' ',' ;');
       fprintf(fid,['c=[',blk,' ];\n']);
       fprintf(fid,'%s\n','lambda= ;');
       fprintf(fid,'%s\n','beta= ;');
    else
       fprintf(fid,['c=[',blk,' ];\n']);
    end
    if adapt
         blk=blanks(p-1);
         blk=strrep(blk,' ',' ;');
         fprintf(fid,['ad=[',blk,' ];\n']);
         fprintf(fid,['theta=[',blk,' ];\n']);
    end
    fprintf(fid,'%s\n','t0= ;      %  Initial time');
    fprintf(fid,'%s\n','tf= ;      %  Final time');
    fprintf(fid,'%c\n%s\n%c\n','%','%     Initial conditions','%');
    blk=blanks(2*n+p-cont-1);
    blk=strrep(blk,' ',' ,');
    fprintf(fid,['x0=[',blk,' ];\n']);
    fprintf(fid,['[t,x]=ode23(''',modfile,''',t0,tf,x0);\n']);
    fclose(fid);
end

function [mbigger]=symadrow(m,v);
%
% SYMADROW  appends a new row to a symbolic matrix.
%
% MBIGGER=SYMADROW(M,V) appends the symbolic vector V, as a new row, at
% the end of the matrix M to generate the expanded matrix MBIGGER.
%
%    Programmed by M. Rios-Bolivar   Date:24/10/96
%
[nr,nc]=symsize(m);
lm=length(m);
m_is_scalar=(nr==nc) & (nc==1);
```

```
if m_is_scalar
  mbigger=sym([m,';',v]);
else
  cad=m(1,2:lm-1);
  for i=2:nr
     cad=[cad,';',m(i,2:lm-1)];
  end
  [nrv,ncv]=symsize(v);
  if (nrv==ncv) & (ncv==1)   % v is scalar
    mbigger=sym([cad,';',v]);
  else
    mbigger=sym([cad,';',v(2:length(v)-1)]);
  end
end

function wrecback=wrecback(fid,cad)
%
%   This program writes the character string 'cad' in the file
%   assigned to 'fid' as records of length approximately equal
%   to 'max'
%
%   Programmed by M. Rios-Bolivar    Date: 24/10/96
%
max=65;      %  approximate record length
while cad ~= '',
   l=length(cad);
   if l > max
      p=findstr('+',cad(max+1:l));
  %    p=[p,findstr('-',cad(max+1:l))];
      p=[p,findstr('*',cad(max+1:l))];
      p=[p,findstr('/',cad(max+1:l))];
      pos=min(p);
      if pos==[]
         fprintf(fid,[cad,';\n']);
         cad='';
      else
        fprintf(fid,[cad(1:max+pos),'...\n']);
        cad=cad(max+pos+1:l);
      end
   else
      fprintf(fid,[cad,';\n']);
      cad='';
    end
end
end
```

# Bibliography


[1] Y. Ait Amirat and S. Diop, "Towards an algebraic system theory toolbox." *Proceedings of the 13th IFAC World Congress*, San Francisco (CA), USA, July 1996, vol. F, pp. 299–304.

[2] T. Ahmed-Ali and F. Lamnabhi-Lagarigue, "Adaptive tracking of some uncertain nonlinear systems." *Proc. IEEE Workshop on Flatness*, Sheffield, UK, July 1996.

[3] B. R. Barmish and G. Leitmann, "On ultimate boundedness control of uncertain systems in the absence of matching conditions." *IEEE Transactions on Automatic Control*, vol. 27, pp. 153–158, 1982.

[4] D. Bestle and M. Zeitz, "Canonical form observer design for nonlinear time-variable systems." *International Journal of Control*, vol. 38, pp. 419–431, 1983.

[5] G. L. Blankenship, R. Ghanadan, H. G. Kwatny, C. LaVigna and V. Polyakov, "Tools for integrated modeling, design, and nonlinear control." *IEEE Control Systems Magazine*, vol. 15, no. 2, pp. 65–79, 1995.

[6] J. Birk and M. Zeitz, "Program for symbolic and rule-based analysis and design of nonlinear systems," in *Algebraic Computing in Control. Proceedings of the 1st European Control Conference. Lecture Notes in Control and Information Sciences 165*, G. Jacob and F. Lamnabhi-Lagarrigue, Eds., Springer-Verlag, pp. 115–119, 1991.

[7] J. A. Burton and A. S. I. Zinober, "Continuous approximation of variable structure control." *International Journal of Systems Science*, vol. 17, pp. 876–885, 1986.

[8] C. I. Byrnes and A. Isidori, "Local stabilization of minimum-phase nonlinear systems." *Systems and Control Letters*, vol. 11, pp. 9–17, 1988.

[9] C. I. Byrnes and A. Isidori, "New results and examples in nonlinear feedback stabilization." *Systems and Control Letters*, vol. 12, pp. 437–442, 1989.

[10] C. I. Byrnes and A. Isidori, "Asymptotic stabilization of minimum-phase nonlinear systems." *IEEE Transactions on Automatic Control*, vol. 36, pp. 1122–1137, 1991.

[11] C. I. Byrnes, A. Isidori and J. C. Willems, "Passivity, feedback equivalence, and the global stabilization of minimum phase nonlinear systems." *IEEE Transactions on Automatic Control*, vol. 36, pp. 1228–1240, 1991.

[12] C. I. Byrnes and C. F. Martin, "An integral-invariance principle for nonlinear systems." *IEEE Transactions on Automatic Control*, vol. 40, pp. 983–994, 1995.

[13] G. Campion and G. Bastin, "Indirect adaptive state feedback control of linearly parametrized nonlinear systems," *International Journal of Adaptive Control and Signal Processing*, vol 4, pp. 345–358, 1990.

[14] B. Charlet, J. Lévine and R. Marino, "On dynamic feedback linearization." *Systems and Control Letters*, vol. 13, pp. 143–151, 1989.

[15] B. Charlet, J. Lévine and R. Marino, "Sufficient conditions for dynamic state feedback linearization." *SIAM Journal of Control and Optimization*, vol. 29, pp. 38–57, 1991.

[16] Y. H. Chen, "On the robustness of mismatched uncertain dynamical systems." *ASME Journal of Dynamic Systems, Measurement, and Control*, vol. 109, pp. 29–35, 1987.

[17] Y. H. Chen and G. Leitmann, "Robustness of uncertain systems in the absence of matching assumptions." *International Journal of Control*, vol. 45, pp. 1527–1542, 1987.

[18] C. Chien and L. Fu, "An adaptive variable structure control for a class of nonlinear systems." *Systems and Control Letters*, vol. 21, pp. 49–57, 1993.

[19] M. Corless and G. Leitmann, "Continuous state feedback guaranteeing uniform ultimate boundedness for uncertain dynamical systems." *IEEE Transactions on Automatic Control*, vol. 26, pp. 1139–1144, 1981.

[20] J-M. Coron and L. Praly, "Adding an integrator for the stabilization problem." *Systems and Control Letters*, vol. 17, pp. 89–104, 1991.

[21] D. M. Dawson, J. J. Carroll and M. Schneider, "Integrator backstepping control of a brush DC motor turning a robotic load." *IEEE Transactions on Control Systems Technology*, vol. 2, pp. 233–244, 1994.

[22] R. A. DeCarlo, S. H. Zak, and G. P. Matthews, "Variable structure control of nonlinear multivariable systems: A tutorial." *Proceedings of IEEE*, vol. 76, pp. 212–232, 1988.

[23] B. de Jager, "Symbolic analysis and design for (nonaffine) nonlinear control systems." *Proceedings of the 13th IFAC World Congress*, San Francisco (CA), USA, July 1996, vol. F, pp. 289–294.

[24] J. Descusse and C. H. Moog, "Decoupling with dynamic compensation for strong invertible affine nonlinear systems." *International Journal of Control*, vol. 42, pp. 1387–1398, 1985.

[25] A. G. Filippov, "Application of the theory of equations with discontinuous right-hand side to non-linear problems in automatic control." *Proceedings First IFAC Congress*, London, UK, pp. 923–927, 1961.

[26] M. Fliess, "A note on the invertibility of nonlinear input-output differential systems." *IEEE Transactions on Automatic Control,* vol. 8, pp. 147–151, 1986.

[27] M. Fliess, "Nonlinear control theory and differential algebra," in *Modeling and Adaptive Control*, Ch. I. Byrnes and A. Khurzhansky, Eds., Lecture Notes in Control and Information Sciences, Springer-Verlag, Berlin, 1989.

[28] M. Fliess, "Generalized controller canonical forms for linear and nonlinear dynamics." *IEEE Transactions on Automatic Control,* vol. 35, pp. 994–1001, 1990.

[29] M. Fliess, "Flatness and defect of non-linear systems: introductory theory and examples." *International Journal of Control*, vol. 61, pp. 1327–1361, 1995.

[30] M. Fliess and F. Messager, "Sur la commande en régime glissant." *Comptes Rendus de la Academié des Sciences de Paris*, vol. 313, série I, pp. 951–956, 1991.

[31] R. A. Freeman and P. V. Kokotović, "Design of softer robust nonlinear control laws." *Automatica*, vol. 29, pp. 1425–1437, 1993.

[32] R. A. Freeman and P. V. Kokotović, "Tracking controllers for systems linear in the unmeasured states." *Automatica*, vol. 32, pp. 735–746, 1996.

[33] F. Garofalo and L. Glielmo (Eds.), *Robust Control via Variable Structure and Lyapunov Techniques* Berlin: Springer-Verlag, 1996.

[34] A. Glumineau and L. Graciani, "Symbolic nonlinear analysis and control package." *Proceedings of the 13th IFAC World Congress*, San Francisco (CA), USA, July 1996, vol. F, pp. 295–298.

[35] G. C. Goodwin and K. S. Sin, *Adaptive Filtering Prediction and Control*, Prentice-Hall, Englewood Cliffs, NJ, 1984.

[36] S. Gutman, "Uncertain dynamical systems-Lyapunov min-max approach." *IEEE Transactions on Automatic Control*, vol. 24, pp. 437–443, 1979.

[37] S. Gutman and G. Leitmann, "On a Class of Linear Differential Games." *Journal of Optimization Theory mand Applications*, vol. 17, 1975.

[38] P. A. Ioannou and A. Datta, "Robust Adaptive Control: A Unified Approach." *Proceedings of the IEEE*, vol. 79, pp. 1736–1767, 1991.

[39] P. A. Ioannou and P. V. Kokotović, *Adaptive Systems with Reduced Models*. Springer-Verlag, Berlin, 1983.

[40] P. A. Ioannou and P. V. Kokotović, "Instability analysis and the improvement of robustness of adaptive control." *Automatica*, vol. 20, p. 583–594, 1984.

[41] P. A. Ioannou and J. Sun, "Theory and design of robust direct and indirect adaptive control schemes." *International Journal of Control*, vol. 47, pp. 775–813, 1986.

[42] P. A. Ioannou and J. Sun, *Robust Adaptive Control*, Prentice-Hall, Upper Saddle River, NJ, 1996.

[43] P. A. Ioannou and K. S. Tsakalis, "A robust direct adaptive controller." *IEEE Transactions on Automatic Control*, vol. 31, pp. 1033–1043, 1986.

[44] A. Isidori, *Nonlinear Control Systems*, 3rd edition, Springer-Verlag, New York, 1995.

[45] I. Kanellakopoulos, "Passive adaptive control of nonlinear systems." *International Journal of Adaptive Control and Signal Processing*, vol. 7, pp. 339–352, 1993.

[46] I. Kanellakopoulos, P. V. Kokotović and R. Marino, "An extended direct scheme for robust adaptive nonlinear control." *Automatica*, vol. 27, pp. 247–255, 1991.

[47] I. Kanellakopoulos, P. V. Kokotović and A. S. Morse, "Adaptive output-feedback control of a class of nonlinear systems." *Proceedings of the 30th IEEE Conference on Decision and Control*, Brighton, UK, December 1991, vol. 2, pp. 1082–1087.

[48] I. Kanellakopoulos, P.V. Kokotović and A.S. Morse, "Systematic design of adaptive controllers for feedback linearizable systems." *IEEE Transactions on Automatic Control*, vol. 36, pp. 1241–1253, 1991.

[49] I. Kanellakopoulos, P. V. Kokotović and A. S. Morse, "A toolkit for nonlinear feedback designs." *Systems and Control Letters*, vol. 18, pp. 83–92, 1992.

[50] I. Kanellakopoulos, P. V. Kokotović and A. S. Morse, "Adaptive output-feedback control of systems with ouput nonlinearities." In *Foundations of Adaptive Control*, P. V. Kokotović, Ed., pp. 495–525, 1991.

[51] I. Kanellakopoulos, M. Krstić and P. V. Kokotović, "$\kappa$-adaptive control of output-feedback nonlinear systems." *Proceedings of the 32nd IEEE Conference on Decision and Control*, San Antonio, TX, pp.1061–1066, 1993.

[52] L. Karsenti and F. Lamnabhi-Lagarrigue, "Adaptive feedback control for nonlinear systems using backstepping and sliding-mode designs," submitted for publication.

[53] J. G. Kassakian, M. F. Schlecht and G. C. Verghese, *Principle of Power Electronics*, Addison-Wesley, Massachusetts, 1991.

[54] H. Keller and H. Fritz, "Design of nonlinear observers by a two-steps transformation." In: *Algebraic and Geometric Methods in Nonlinear Control Theory*, M. Fliess and M. Hazewinkel, Eds., D. Reidel Publishing Company, pp 89–98, 1986.

[55] P. V. Kokotović, Ed., *Foundations of Adaptive Control*, Springer-Verlag, Berlin, 1991.

[56] P. V. Kokotović, "The joy of feedback: nonlinear and adaptive." *IEEE Control Systems Magazine*, vol. 12, pp. 7–17, 1992.

[57] P. V. Kokotović, I. Kanellakopoulos, and A. S. Morse, "Adaptive feedback linearization of nonlinear systems." In *Foundations of Adaptive Control*, P. V. Kokotović, Ed., pp. 311–346, 1991.

[58] P.V. Kokotović, M. Krstić and I. Kanellakopoulos, "Backstepping to passivity: Recursive design of adaptive systems." *Proceedings of the 31st. IEEE Conference on Decision and Control*, Tucson, USA, December 1992, pp. 3276–3280.

[59] A. J. Krener and A. Isidori, "Linearization by output injection and nonlinear observers." *Systems and Control Letters*, vol. 3, pp. 47–52, 1983.

[60] A. J. Krener and W. Respondek, "Nonlinear observers with linearizable error dynamics." *SIAM Journal of Control and Optimization*, vol. 23, pp. 197–216, 1985.

[61] M. Krstić, I. Kanellakopoulos and P.V. Kokotović, "Adaptive nonlinear control without overparametrization." *Systems and Control Letters*, vol. 19, pp. 177–185, 1992.

[62] M. Krstić, I. Kanellakopoulos and P. V. Kokotović, "A new generation of adaptive controllers for linear systems." *Proceedings of the 31st. IEEE Conference on Decision and Control*, Tucson, USA, December 1992, pp. 3644–3650.

[63] M. Krstić, I. Kanellakopoulos and P. V. Kokotović, "Nonlinear design of adaptive controllers for linear systems." *IEEE Transactions on Automatic Control*, vol. 39, pp. 738–751, 1994.

[64] M. Krstić, I. Kanellakopoulos and P. V. Kokotović, *Nonlinear and Adaptive Control Design*, New York: John Wiley & Sons, 1995.

[65] M. Krstić and P.V. Kokotović, "Observed-based schemes for adaptive nonlinear state-feedback control." *International Journal of Control*, vol. 59, pp. 1373–1381, 1994.

[66] M. Krstić and P. V. Kokotović, "Adaptive nonlinear design with controller-identifier separation and swapping." *IEEE Transactions on Automatic Control*, vol. 40, pp. 426–440, 1995.

[67] H. G. Kwatny and G. L. Blankenship, "Symbolic tools for variable structure control system design: The zero dynamics." *Proceedings of the IEEE Workshop on Robust Control via Variable Structure and Lyapunov Techniques*, Benevento, Italy, September 1994, pp. 204–209.

[68] G. Leitmann, "A differential game of pursuit and evasion." *International Journal of Non-linear Mechanics*, vol. 4, pp. 1–6, 1969.

[69] G. Leitmann, "Guaranteed asymptotic stability for some linear systems with bounded uncertainties." *ASME Journal of Dynamic Systems, Measurement and Control*, vol. 23, pp 212–216, 1979.

[70] O. Llanes-Santiago and H. Sira-Ramírez, "A controller resetting strategy for the stabilization of DC-to-DC power converters towards non-minimum phase equilib-

ria." Proceedings of the 33th IEEE Conference on Decision and Control, USA, Walt Disney World Resort, Lake Buena Vista, Florida, USA, December 1994.

[71] The student edition of MATLAB: Version 4: User's guide. The MathWorks, Prentice-Hall, Englewood Cliffs, NJ, 1995.

[72] R. Marino and M. W. Spong, "Nonlinear control techniques for flexible joint manipulators: A single link case study. " *Proceedings of the IEEE International Conference on Robotics and Automation*, 1986, pp. 1026–1030.

[73] R. Marino and P. Tomei, "Global adaptive observers and output-feedback stabilization for a class of nonlinear systems." In: *Foundations of Adaptive Control*, P.V. Kokotović, Ed., Springer-Verlag, Berlin, pp. 455–493, 1991.

[74] R. Marino and P. Tomei, "Global adaptive output-feedback control of nonlinear systems." *Proceedings of the 30th IEEE Conference on Decision and Control*, Brighton, U.K., December 1991, pp. 1077–1081.

[75] R. Marino and P. Tomei, *Nonlinear Control Design*, Prentice-Hall, London, 1995.

[76] R. Marino, P. Tomei, I. Kanellakopoulos and P. V. Kokotović, "Adaptive tracking for a class of feedback linearizable systems." *IEEE Transactions on Automatic control*, vol. 39, pp. 1314–1319, 1994.

[77] F. Mazenc, L. Praly and W. P. Dayawansa, "Global stabilization by output feedback: examples and counterexamples." *Systems and Control Letters*, vol. 23, pp. 119–125, 1994.

[78] K. Nam and A. Arapostathis, "A model-reference adaptive control scheme for pure-feedback nonlinear systems." *IEEE Transactions on Automatic Control*, vol 33, pp.803–811, 1988.

[79] K. S. Narendra, "The maturing of adaptive control." In: *Foundations of Adaptive Control*, P.V. Kokotović, Ed., Springer-Verlag, Berlin, pp. 3–36, 1991.

[80] K. S. Narendra and M. Annaswamy, *Stable Adaptive Systems*, Prentice-Hall, Englewood Cliffs, NJ, 1989.

[81] H. Nijmeijer and A. J. van der Schaft, *Nonlinear Dynamical Control Systems*, Springer-Verlag, New York, 1990.

[82] J.-B. Pomet and L. Praly, "Adaptive nonlinear regulation: equation error from the Lyapunov equation." *Proc. 28th IEEE Conference on Decision and Control*, Tampa, USA, December 1989, pp. 1008–1013.

[83] L. Praly, "Robust model reference adaptive controllers–Part 1: Stability Analysis." *Proceedings 23rd IEEE Conference on Decision and Control*, pp. 1009–1014, 1984.

[84] Z. Qu, "Robust control of nonlinear uncertain systems under generalized matching conditions." *Automatica*, vol. 29, pp. 985–998, 1993.

[85] Z. Qu, "A new generic procedure of designing robust control for nonlinear uncertain systems: Beyond the Back-stepping design." *Proceedings of the IEEE Workshop on Robust Control via Variable Structure and Lyapunov Techniques*, Benevento, Italy, September 1994, pp. 174–181.

[86] Z. Qu and J. F. Dorsey, "Robust control of generalized dynamic systems without matching conditions." *ASME Journal of Dynamic Systems, Measurement and Control*, vol. 113, pp. 582–589, 1991.

[87] M. Rios-Bolívar, H. Sira-Ramírez and A. S. I. Zinober, "Output tracking control via adaptive input-output linearization: A backstepping approach." *Proceedings of the 34th IEEE Conference on Decision and Control*, New Orleans, USA, December 1995, vol. 2, pp. 1579–1584.

[88] M. Rios-Bolívar and A. S. I. Zinober, "Sliding mode control for uncertain linearizable nonlinear systems: A backstepping approach," *Proceedings of the IEEE Workshop on Robust Control via Variable Structure and Lyapunov Techniques*, Benevento, Italy, September 1994, pp. 78–85.

[89] M. Rios-Bolívar and A. S. I. Zinober, "Symbolic dynamical adaptive backstepping control design via MATLAB." *Proceedings of the 1996 IEEE Workshop on Variable Structure Systems, VSS'96*, Tokyo, Japan, December 1996, pp. 216–221.

[90] M. Rios-Bolívar and A. S. I. Zinober, "Dynamical adaptive backstepping control design via symbolic computation." *Accepted for 4th European Control Conference, ECC 97*, Brussels, Belgium, July 1997.

[91] M. Rios-Bolívar, A. S. I. Zinober and H. Sira-Ramírez, "Sliding mode output tracking via backstepping for uncertain nonlinear systems." *Proceedings of the 3rd European Control Conference*, Rome, Italy, September 1995, pp. 699–704.

[92] M. Rios-Bolívar, A. S. I. Zinober and H. Sira-Ramírez, "Dynamical sliding mode control via adaptive input-output linearization: A backstepping approach," in: *Robust Control via Variable Structure and Lyapunov Techniques* F. Garofalo and L. Glielmo, Eds., Springer-Verlag. pp. 15–35, 1996.

[93] M. Rios-Bolívar, A. S. I. Zinober and H. Sira-Ramírez, "Dynamical adaptive backstepping control/observer design for uncertain nonlinear systems." *Proceedings of the 13th IFAC World Congress*, San Francisco (CA), USA, July 1996, vol K, pp. 145–150.

[94] M. Rios-Bolívar, A. S. I. Zinober and H. Sira-Ramírez,"Dynamical adaptive sliding mode output tracking control of a class of nonlinear systems." *International Journal of Robust and Nonlinear Control*, 1997 (to appear).

[95] R. Rothfuss and M. Zeitz, "A toolbox for symbolic nonlinear feedback design." *Proceedings of the 13th IFAC World Congress*, San Francisco (CA), USA, July 1996, vol. F, pp. 283–288.

[96] E. P. Ryan, "An integral invariance principle for differential inclusions with applications in adaptive control." *SIAM Journal of Control & optimization*, 1997 (to appear).

[97] S. S. Sastry and M. Bodson, *Adaptive Control: Stability, Convergence and Robustness*, Prentice-Hall, Englewood Cliffs, NJ, 1989.

[98] S. S. Sastry and A. Isidori, "Adaptive control of linearizable systems." *IEEE Transactions on Automatic Control*, vol. 34, pp. 1123–1131, 1989.

[99] R. Sepulche, M. Jankovic and P. Kokotović, *Constructive Control Design*. Springer-Verlag, Berlin, 1996.

[100] D. Seto, A. M. Annaswamy and J. Baillieul, "Adaptive control of nonlinear systems with a triangular structure." *IEEE Transactions on Automatic Control*, vol. 39, pp. 1411–1427, 1994.

[101] R. P. Severns and G. Bloom, *Modern DC-to-DC Switch-mode Power Converter Circuits*, New York: Van Nostrand-Reinhold, 1985.

[102] S. N. Singh, "Decoupling of invertible nonlinear systems with state feedback and precompensation." *IEEE Transactions on Automatic Control*, vol. 25, pp. 1237–1239, 1980.

[103] H. Sira-Ramírez, "Sliding mode control on slow manifolds of DC-to-DC Power converters." *International Journal of Control*, vol. 47, pp. 1323–1340, 1988.

[104] H. Sira-Ramírez, "A geometric approach to pulse-width-modulated control in nonlinear dynamical systems." *IEEE Transactions on Automatic Control*, vol. 34, pp. 184–187, 1989.

[105] H. Sira-Ramírez, "A dynamical variable structure control strategy in asymptotic output tracking problems." *Proceedings of the 30th IEEE Conference on Decision and Control*, Brighton, UK, December 1991, vol. 2, pp. 1373–1378.

[106] H. Sira-Ramírez, "A dynamical variable structure control strategy in asymptotic output tracking problems." *IEEE Transactions on Automatic Control,* vol. 38, pp. 615–620, 1993.

[107] H. Sira-Ramírez and P. Lischinsky-Arenas, "The differential approach in nonlinear dynamical compensator design for DC-to-DC Power converters." *International Journal of Control*, vol. 54, pp. 111–133, 1991.

[108] H. Sira-Ramírez and O. Llanes-Santiago, "Adaptive dynamical sliding mode control via backstepping." *Proceeding of the 32nd IEEE Conference on Decision and Control*, San Antonio, USA, December 1993, vol. 3, pp. 1422–1426.

[109] H. Sira-Ramírez, M. García-Esteban and A. S. I. Zinober, "Dynamical adaptive Pulse-Width-Modulation control of DC–to–DC power converters: A backstepping approach." *International Journal of Control*, 1997 (to appear).

[110] H. Sira-Ramírez and M. Rios-Bolívar, "Sliding Mode Control of dc-to-dc power converters via Extended Linearization." *IEEE Transactions on Circuits and Systems*, vol. 41, pp. 652–661, 1994.

[111] H. Sira-Ramírez, M. Rios-Bolívar and A. S. I. Zinober, "Adaptive input-output linearization for PWM regulation of DC-to-DC power converters." *Proceedings of the 1995 American Control Conference*, Seattle, USA, vol. 1, pp. 81–85.

[112] H. Sira-Ramírez, M. Rios-Bolívar and A. S. I. Zinober, "Adaptive dynamical input-output linearization of DC-to-DC power converters: A Backstepping approach." *International Journal of Robust and Nonlinear Control*, 1997 (to appear).

[113] H. Sira-Ramírez, R. Tarantino-Alvarado and O. Llanes-Santiago,"Adaptive feedback stabilization in PWM-controlled DC-to-DC Power supplies." *International Journal of Control*, vol. 57, pp. 599–625, 1993.

[114] J-J. E. Slotine and W. Li, *Applied Nonlinear Control*, Prentice-Hall, Englewood Cliffs, NJ, 1991.

[115] S. K. Spurgeon and X. Y. Lu, "Output tracking of non-minimum phase systems via sliding mode." *Proceedings of the 3rd European Control Conference*, Rome, Italy, September 1995, pp. 705–710.

[116] C. Su, "Adaptive sliding mode control of nonlinear robotic systems with time-varying parameters." *Systems and Control Letters*, vol. 23, pp. 35–41, 1994.

[117] D. G. Taylor, P. V. Kokotović, R. Marino and I. Kanellakopoulos, "Adaptive regulation of nonlinear systems with unmodeled dynamics." *IEEE Transactions on Automatic Control*, vol. 34, pp. 405–412, 1989.

[118] A. Teel, "Semi-global stabilization of minimum phase nonlinear systems in special normal forms." *Systems and Control Letters*, vol. 19, pp. 187–192, 1992.

[119] A. Teel and L. Praly, "Global stabilizability and observability imply semi-global stabilizability by output feedback." *Systems and Control Letters*, vol. 22, pp. 313–325, 1994.

[120] J. S. Thorp and B. R. Barmish, "On guaranteed stability of uncertain linear systems via linear control." *Journal of Optimization Theory and Applications*, vol. 35, pp. 559–579.

[121] A. Tornambé, "Output feedback stabilization of a class of non-minimum phase nonlinear systems." *Systems and Control Letters*, vol. 19, pp. 193–204, 1992.

[122] V. I. Utkin, *Sliding Modes in Control Optimization*. Springer-Verlag, Berlin, 1992.

[123] A. J. van der Schaft, "Representing a nonlinear state space system as a set of higher order differential equations in the inputs and outputs." *Systems and Control Letters*, vol. 12, pp. 151–160, 1989.

[124] W. Zhan, T. J. Tarn and A. Isidori, "A canonical dynamical extension for noninteraction with stability for affine nonlinear square systems." *Systems and Control Letters*, vol. 17, pp. 177–184, 1991.

[125] M. Zeitz, "Observability canonical (phase-variable) form for non-linear time-variable systems." *International Journal of Systems Science*, vol. 15, pp. 949–958, 1984.

[126] A. S. I. Zinober *Variable Structure and Lyapunov Control.* Springer-Verlag, London, 1994.