STUDY OF SEVERAL PROBLEMS IN NONLINEAR CONTROL BY BACKSTEPPING

by

Zongtao Lu

Department of Electrical Engineering
Lakehead University
Thunder Bay, Ontario

A thesis submitted to the Faculty of Graduate Studies and
Research in partial fulfillment of the requirement for the
Degree of Science in Control Engineering

November, 2005

Your file Votre référence
ISBN: 978-0-494-17017-5
Our file Notre référence
ISBN: 978-0-494-17017-5

Lakehead
UNIVERSITY

OFFICE OF GRADUATE STUDIES

NAME OF STUDENT: Zongtao Lu

DEGREE AWARDED: Master of Science in Control Engineering

ACADEMIC UNIT: Dept. of Electrical Engineering

TITLE OF THESIS: Study of Several Problems in Nonlinear Control by Backstepping

This thesis has been prepared

under my supervision

and the candidate has complied

with the Master's regulations.

Signature of Supervisor

Aug. 18/05

Date

Content

Abstract

Three topics in nonlinear control are studied in this thesis:

Adaptive control is investigated for a class of nonlinearly parameterized systems by backstepping. An adaptive controller is constructed for MIMO nonlinearly parameterized systems with nested triangular form. The design procedure is developed based on the adaptive backstepping design technique. The designed controller guarantees that the corresponding closed-loop system is globally asymptotically stable for any unknown parameters which enter the system nonlinearly.

Adaptive control problem is also investigated for MIMO nonlinear DAE systems with unknown parameters appearing linearly in both differential and algebraic equations. The DAE system is converted into an equivalent ODE system. An adaptive controller is designed by the backstepping technique and the asymptotic stability of the system is guaranteed.

A parallel robotic system is studied as a test-bed to illustrate the backstepping control approach. For comparison, PD control is also designed and implemented on the parallel robot. The given simulation and experimental results are satisfactory for both backstepping and PD control approaches.

Key Words: backstepping, nonlinear control, adaptive control, differential-algebraic equation (DAE) systems, parallel robot

Chapter 1

Introduction

1.1 Background

In this thesis, several topics in nonlinear control are studied by the backstepping technique. The adaptive control problems are investigated by backstepping for a class of ordinary differential equation (ODE) systems and differential algebraic equation (DAE) systems in Chapter 2 and Chapter 3 respectively. In Chapter 4, the controllers for set point controls are designed and implemented on both backstepping and PD control schemes for a parallel robotic system. The backstepping technique plays a key role in this thesis. First of all, we are going to give a brief introduction to the backstepping design technique.

Great progress has been made in nonlinear control since a recursive design procedure, called backstepping, was systematically developed by [10]. For details in this subject, see [14], [34], [28] and references therein. The backstepping technique can be perfectly applied for a class of nonlinear system, specifically, "lower triangular" nonlinear systems. Let's take a simple example to illustrate the basic idea of backstepping. Consider the following 2-th order system

$$\dot{x}_1 = x_2 + \phi_1(x_1) \tag{1.1}$$

$$\dot{x}_2 = u + \phi_2(x_1, x_2) \tag{1.2}$$

where u is the control input, $\phi_1(x_1)$, $\phi_2(x_1, x_2)$ are the functions of x_1 and x_1, x_2 respectively, which are as differentiable as needed and vanish at the origin. Our aim is to design the control input u to stabilize the states x_1, x_2 to the origin. The backstepping technique is a recursive design procedure. In each step, a Lyapunov function candidate is constructed and by choosing a controller α, the derivative of the Lyapunov function candidate is made negative definite.

Step 1: Consider the Lyapunov function candidate

$$V_1 = \frac{1}{2}(x_1)^2$$

Differentiating V_1 with respect to time yields

$$\dot{V}_1 = x_1(x_2 + \phi_1)$$

By introducing a virtual controller

$$\alpha_1(x_1) = -c_1 x_1 - \phi_1(x_1)$$

with c_1 is a positive number,

$$\dot{V}_1 = -c_1(x_1)^2 + x_1(x_2 - \alpha_1)$$

Step 2: Consider the Lyapunov function candidate

$$V_2 = V_1 + \frac{1}{2}(x_2 - \alpha_1)^2$$

Differentiating V_2 with respect to time yields

$$\dot{V}_2 = -c_1(x_1)^2 + x_1(x_2 - \alpha_1) + (x_2 - \alpha_1)\left[u + \phi_2 - \frac{\partial \alpha_1}{\partial x_1}(x_2 + \phi_1)\right]$$

It is not difficult to see that the controller

$$u = -c_2(x_2 - \alpha_1) - x_1 - \phi_2 + \frac{\partial \alpha_1}{\partial x_1}(x_2 + \phi_1) \tag{1.3}$$

makes $\dot{V}_2$

$$\dot{V}_2 = -c_1(x_1)^2 - c_2(x_2 - \alpha_1)^2$$

negative definite, which implies the feedback controller u can stabilize the corresponding closed-loop system (1.1), (1.2), (1.3).

By the developed backstepping technique, several topics are studied in the areas of nonlinear control. The thesis is organized as follows:

Chapter 2 and 3 involve more theoretical research and Chapter 4 is the experimental part. Backstepping is extended in the adaptive control from the linearly parameterized systems to the nonlinearly parameterized systems with nested triangular structure in Chapter 2. The developed methodology based on backstepping can stabilize the multi-input multi-output (MIMO) system with the unknown parameters entering the system nonlinearly.

In Chapter 3, the adaptive control problem is investigated based on the adaptive backstepping for DAE systems with unknown parameters appearing linearly in both differential and algebraic equations. We propose three algorithms to produce a set of new coordinates, in which the original system is expressed in lower triangular form. An adaptive controller is designed by the backstepping technique and the asymptotic stability of the system is guaranteed. As an application example of DAE systems, a constrained manipulator with flexible joints is studied to illustrate the proposed methodology.

In Chapter 4, we take a parallel robot as a test-bed for the nonlinear stabilizing controller designed by backstepping. The experimental setup is introduced and the controller design procedure is shown with the simulation and experimental results.

1.2 Literature Review

The methodology developed in [10] is for linearly parameterized system and in strict feedback form. Up to now, most of the existing results obtained by the adaptive backstepping design method are limited to the linearly parameterized systems with lower triangular form. [23] studied the adaptive control of the multi-input multi-output (MIMO) nonlinear systems with nested triangular structure, which is introduced in [25] for the first time. In [23], the strict feedback condition is relaxed for MIMO nonlinear systems.

On the other hand, by comparison with comprehensive reports on the development in the area of adaptive control with linear parameterization, few papers are published for adaptive control for nonlinearly parameterized systems. In [27] and [40], the problem of the global adaptive control was solved for the nonlinearly parameterized systems with the bound of the nonlinear parameters assumed a priori. Nonlinear terms considered as the functions of unknown parameters are assumed either convex or concave in [12]. In [18], systematic design methods of adaptive control for nonlinearly parameterized systems are presented without imposing any extra conditions such as convex/concave condition or the upper bound on the unknown parameters. However, the results in [18] are limited in single-input single-output (SISO) systems. The extension to the MIMO nonlinearly parameterized systems still remains open. Meanwhile, the nonlinearly parameterized systems with nested triangular form model more general forms of the real systems than the ones with strict triangular forms, which can be considered as a particular case of the ones with nested triangular form. Therefore, it is natural and of practical importance to investigate the problem of adaptive control for MIMO nonlinearly parameterized systems with nested triangular forms, in which the conditions of strict feedback form and linear parameterization are respectively relaxed to nested triangular form and nonlinear parameterization.

In Chapter 2, an adaptive controller is constructed for MIMO nonlinearly parameterized systems with nested triangular form. Based on the recursive adaptive backstepping and inspired by [23], a design procedure is developed for the construction of an adaptive controller. The success of the design procedure is guaranteed by the assumption involved in the functions which the unknown parameters enter nonlinearly. Two lemmas proposed in [18] are applied to separate the unknown parameters linearly from the nonlinear parameterization. Instead of estimating the vector of the original unknown parameters, a new scalar parameter is introduced by the combination of the original unknown parameters and is estimated. The derived adaptive controller guarantees that the global asymptotic stability of the closed-loop system with the estimation of the scalar parameter. Finally, one numerical example is studied and the simulation results are given to illustrate the design methodology proposed in this chapter.

Differential-Algebraic Equation (DAE) systems (also referred to as singular, descriptor, semistate, generalized systems etc.) arise naturally as dynamic models of electrical [32], mechanical [29] and chemical engineering [4] applications. It constitutes an important class of systems of both theoretical interest and practical significance. In theoretical research on DAE systems, most of the work focused on the issues related to solvability and numerical solutions [2] [3]. There was also some work on the topics of feedback linearization [11], observer design [42], disturbance decoupling [22], input-output decoupling [24], output tracking [13] [21], output regulation [9], stabilization [26] [30], robust stabilization [20], over- and underdetermined nonlinear analysis[16]. In practical applications, it is known that mechanical systems with classical holonomic and nonholonomic constraints [41] and robotic systems with kinematic constraints [13] [31] are modeled naturally by DAE systems. DAE systems are also known as dynamic models in power systems [8] and chemical processes [15].

In above mentioned theoretical research and practical applications, the parameters in the DAE systems are normally assumed to be known or given. But generally, it is not the case in practice. For example, for a constrained robotic system, the parameters such as inertia, damping, stiffness and friction coefficients in the dynamic equations are normally unknown or difficult to measure. Therefore, it is necessary to investigate the adaptive control problem of DAE systems with unknown parameters.

In Chapter 3, one methodology will be developed to design a stabilizing feedback controller for the multi-input multi-output (MIMO) DAE systems with unknown parameters. Our aim is to find a change of coordinates to transform the DAE systems into an equivalent ordinary differential equation (ODE) systems with lower triangular structure. As a result, adaptive backstepping is applied to design an adaptive controller for the resulting ODE system.

As an important application of DAE systems, the issues of constrained manipulators have been the focus of research recently [6] [37] [39]. Many robotic motions require contacts to be made with the environment by the end-effector of the robot manipulator. On the other hand, the flexibility of joint transmissions is significant in many applications and proper compensations are required in order to achieve accurate regulation and fast motion tracking [36] [38]. This chapter provides one approach to handle the constrained manipulator with flexible joints. Different from the traditional PD control based on the property of skew symmetry [38] [39], our approach does not require this property, but focuses on the dynamics of the system itself. For some constrained manipulators in which the property of skew symmetry is difficult to obtain, the approach proposed in Chapter 3 can also handle this kind of constrained systems. Following the approach proposed in this chapter, one adaptive controller is designed for a constrained manipulator with flexible joints in Section 3.4. The simulation results show the effectiveness of this approach.

Today, most of the robotic systems that are popular have the links connected sequentially from a fixed base. Typically, all the joints of serial robots are actuated. For serial robots, there is quite a few literature in the dynamics [35], [7] and control results [1], [33]. Different from serial robots, parallel robots have the links connected in series as well as in parallel combinations forming one or more closed-link loops and typically, not all the joints are actuated. The actuators are placed closer to the base or on the base itself. This makes parallel robots have lighter moving parts, which leads to greater efficiency and faster acceleration at the end-effector. Parallel robots also offer greater payload handling capability for the same number of actuators. Therefore, parallel robots are more suitable for fast assembly lines, flight simulators and robotics machining, etc.

In Chapter 4, a parallel robotic manipulator will be studied as an example of the DAE system. A nonlinear controller is designed by the backstepping technique for the parallel robot based on the dynamical model derived in [5]. The designed controller will also be implemented and both of the simulation and experiment results will be shown. For comparison, the simulation and experimental results of the PD controller will be also provided.

Chapter 2

Adaptive Control of MIMO Nonlinearly Parameterized Systems with Nested Triangular Structure

2.1 Introduction

The main content of this chapter is to construct an adaptive controller for MIMO nonlinearly parameterized systems with nested triangular form. Based on the recursive adaptive backstepping and inspired by [23], a design procedure is developed for the construction of an adaptive controller. The success of the design procedure is guaranteed by the assumption involved in the functions which the unknown parameters enter nonlinearly. Two lemmas proposed in [18] are applied to separate the unknown parameters linearly from the nonlinear parameterization. Instead of estimating the vector of the original unknown parameters, a new scalar parameter is estimated, which is introduced by the combination of the original unknown parameters. The derived adaptive controller guarantees that the global asymptotic stability of the closed-loop system with the estimation of the scalar parameter. Finally, one numerical example is studied and the simulation results are given to illustrate the design methodology proposed in this chapter.

2.2 Problem Formulation

Consider a MIMO nonlinear system described by

$$\begin{aligned}
\Sigma^1 &\left\{\begin{array}{l}
\dot{x}_1^1 = x_2^1 + \phi_1^{11}(x_1^1, \theta) + \sum_{s=2}^m x_1^s \phi_1^{1s}(\overline{x}^1, \cdots, \overline{x}^{s-1}, x_1^s, \theta) \\
\vdots \\
\dot{x}_k^1 = x_{k+1}^1 + \phi_k^{11}(x_1^1, \cdots, x_k^1, \theta) + \sum_{s=2}^m x_1^s \phi_k^{1s}(\overline{x}^1, \cdots, \overline{x}^{s-1}, x_1^s, \theta) \\
\vdots \\
\dot{x}_{n_1}^1 = u_1 + \phi_{n_1}^{11}(\overline{x}^1, \theta) + \sum_{s=2}^m x_1^s \phi_{n_1}^{1s}(\overline{x}^1, \cdots, \overline{x}^{s-1}, x_1^s, \theta)
\end{array}\right. \\
&\vdots \\
\Sigma^i &\left\{\begin{array}{l}
\dot{x}_1^i = x_2^i + \phi_1^{ii}(\overline{x}^1, \cdots, \overline{x}^{i-1}, x_1^i, \theta) + \sum_{s=i+1}^m x_1^s \phi_1^{is}(\overline{x}^1, \cdots, \overline{x}^{s-1}, x_1^s, \theta) \\
\vdots \\
\dot{x}_k^i = x_{k+1}^i + \phi_k^{ii}(\overline{x}^1, \cdots, \overline{x}^{i-1}, x_1^i, \cdots, x_k^i, \theta) + \sum_{s=i+1}^m x_1^s \phi_k^{is}(\overline{x}^1, \cdots, \overline{x}^{s-1}, x_1^s, \theta) \\
\vdots \\
\dot{x}_{n_i}^i = u_i + \phi_{n_i}^{ii}(\overline{x}^1, \cdots, \overline{x}^{i-1}, \overline{x}^i, \theta) + \sum_{s=i+1}^m x_1^s \phi_{n_i}^{is}(\overline{x}^1, \cdots, \overline{x}^{s-1}, x_1^s, \theta)
\end{array}\right. \\
&\vdots \\
\Sigma^m &\left\{\begin{array}{l}
\dot{x}_1^m = x_2^m + \phi_1^{mm}(\overline{x}^1, \cdots, \overline{x}^{m-1}, x_1^m, \theta) \\
\vdots \\
\dot{x}_k^m = x_{k+1}^m + \phi_k^{mm}(\overline{x}^1, \cdots, \overline{x}^{m-1}, x_1^m, \cdots, x_k^m, \theta) \\
\vdots \\
\dot{x}_{n_m}^m = u_m + \phi_{n_m}^{mm}(\overline{x}^1, \cdots, \overline{x}^{m-1}, \overline{x}^m, \theta)
\end{array}\right.
\end{aligned} \tag{2.1}$$

where $\overline{x}^i = (x_1^i, \cdots, x_{n_i}^i)^T$ and $\theta \in \mathsf{R}^q$ is the vector of the unknown parameters, which enters the involved functions nonlinearly.

The structure of (2.1) is called nested triangular form, which is introduced in [23]. Note that the nested lower triangular form contains the lower triangular form as a particular case because the former becomes the same as the latter when all the interconnections $\sum_{s=i+1}^m x_1^s \phi_k^{is}$ vanish for $1 \le i \le m$ and $1 \le k \le n_i$. In [23], Liu has studied the problem of adaptive control of MIMO linearly parameterized system with nested triangular form, in which the unknown parameters enter the functions linearly. In the following sections of this chapter, the case that all the involved functions are nonlinear in θ will be discussed. In order to construct Lyapunov functions based on adaptive backstepping for (2.1) with nonlinear parameterization, we are going to separate θ linearly from the involved functions. For this purpose, one assumption will be introduced. Before that, two useful lemmas proposed in [18] will be given first.

Lemma 2.1 *For any real-valued continuous function $f(x,y)$, where $x \in \mathsf{R}^m$, $y \in \mathsf{R}^m$, there are smooth scalar functions $a(x) \geq 0$, $b(y) \geq 0$, $c(x) \geq 1$ and $d(y) \geq 1$, such that*

$$\begin{aligned} |f(x,y)| &\leq a(x) + b(y) \\ |f(x,y)| &\leq c(x)d(y) \end{aligned}$$

Lemma 2.2 *For any positive integers m, n and real-valued function $\pi(x,y) > 0$,*

$$|x|^m |y|^n \leq \frac{m}{m+n}\pi(x,y)\,|x|^{m+n} + \frac{n}{m+n}\pi^{-(m/n)}\,|y|^{m+n}$$

See [18] for the proofs of Lemma 2.1 and Lemma 2.2.

Assumption 2.3 *For $i = 1, \cdots, m$, $s = i+1, \cdots, m$ and $k = 1, \cdots, n_i$,*

$$\phi_k^{ii}(\overline{x}^1, \cdots, \overline{x}^{i-1}, x_1^i, \cdots, x_k^i, \theta) \text{ is smooth and } \phi_k^{ii}(0,\theta) = 0 \tag{2.2}$$

$$\left|\phi_k^{is}(\overline{x}^1, \cdots, \overline{x}^{s-1}, x_1^s, \theta)\right| \leq \left(\sum_{p=1}^{s-1}\sum_{j=1}^{n_p}\left|x_j^p\right| + |x_1^s|\right) b_k^{is}(\overline{x}^1, \cdots, \overline{x}^{s-1}, x_1^s, \theta) \tag{2.3}$$

where b_k^{ii} and b_k^{is} are nonnegative continuous functions.

As for b_k^{ii} and b_k^{is}, with Lemma 2.1, there exist smooth functions $\gamma_k^{ii}(\overline{x}^1, \cdots, \overline{x}^{i-1}, x_1^i, \cdots, x_k^i) \geq 1$, $\gamma_k^{is}(\overline{x}^1, \cdots, \overline{x}^{s-1}, x_1^s) \geq 1$, $c_k^{ii}(\theta) \geq 1$ and $c_k^{is}(\theta) \geq 1$ satisfying

$$\begin{aligned} b_k^{ii}(\overline{x}^1, \cdots, \overline{x}^{i-1}, x_1^i, \cdots, x_k^i, \theta) &\leq \gamma_k^{ii}(\overline{x}^1, \cdots, \overline{x}^{i-1}, x_1^i, \cdots, x_k^i)c_k^{ii}(\theta) \\ b_k^{is}(\overline{x}^1, \cdots, \overline{x}^{s-1}, x_1^s, \theta) &\leq \gamma_k^{is}(\overline{x}^1, \cdots, \overline{x}^{s-1}, x_1^s)c_k^{is}(\theta) \end{aligned}$$

Since θ is a constant, $c_k^{ii}(\theta)$ and $c_k^{is}(\theta)$ are constants as well. Let $\Theta := \sum_{i=1}^{m}\sum_{k=1}^{n_i} c_k^{ii}(\theta) + \sum_{s=i+1}^{m}\sum_{i=1}^{m}\sum_{k=1}^{n_i} c_k^{is}(\theta)$ be a new unknown scalar constant. With the results in [19] and [17], Assumption 2.3 can be rewritten as

$$\left|\phi_k^{ii}(\overline{x}^1,\cdots,\overline{x}^{i-1},x_1^i,\cdots,x_k^i,\theta)\right| \le \left(\sum_{p=1}^{i-1}\sum_{j=1}^{n_p}\left|x_j^p\right| + \sum_{j=1}^{k}\left|x_j^i\right|\right)\gamma_k^{ii}(\overline{x}^1,\cdots,\overline{x}^{i-1},x_1^i,\cdots,x_k^i)\Theta \tag{2.4}$$

$$\left|\phi_k^{is}(\overline{x}^1,\cdots,\overline{x}^{s-1},x_1^s,\theta)\right| \le \left(\sum_{p=1}^{s-1}\sum_{j=1}^{n_p}\left|x_j^p\right| + |x_1^s|\right)\gamma_k^{is}(\overline{x}^1,\cdots,\overline{x}^{s-1},x_1^s)\Theta \tag{2.5}$$

As presented in [18], the new parameter $\Theta \in \mathrm{R}$ is different from the original parameter $\theta \in \mathrm{R}^q$. Instead of estimating θ, we estimate the scalar parameter $\Theta \ge 1$ in our algorithm. Up to now, we separate the parameter Θ from the involved functions, which is positive and only appears linearly in the bounded functions γ_k^{ii} and γ_k^{is}. Taking advantage of the linear-like parameterization condition, we construct an adaptive controller based on adaptive backstepping.

2.3 Main Results

In this section, main results about the adaptive controller design will be given and a constructive proof will be provided.

Theorem 2.4 *Considering the system (2.1), under Assumption 2.3, there exists a smooth adaptive controller*

$$\begin{cases} \dot{\widehat{\Theta}} = \tau(\overline{x}^1,\cdots,\overline{x}^m,\widehat{\Theta}) \\ u_1 = \alpha_{n_1}^1(\overline{x}^1,\widehat{\Theta}) \\ \vdots \\ u_i = \alpha_{n_i}^1(\overline{x}^1,\cdots,\overline{x}^i,\widehat{\Theta}) \\ \vdots \\ u_m = \alpha_{n_m}^1(\overline{x}^1,\cdots,\overline{x}^m,\widehat{\Theta}) \end{cases} \tag{2.6}$$

such that the corresponding closed-loop system (2.1), (2.6) is globally asymptotically stable.

Proof: The proof is based on a recursive procedure. In each step, a Lyapunov function candidate is constructed and by choosing a controller α and three tuning functions τ, λ and Φ, the derivative of the Lyapunov function candidate is made negative definite. For convenience, let $\xi_k^i = x_k^i - \alpha_{k-1}^i$ with $\alpha_0^i = 0$ for $1 \leq i \leq m, 1 \leq j \leq n_i$.

Step 1.1*:* Consider the Lyapunov function candidate

$$V_1^1 = \frac{1}{2}(x_1^1)^2 + \frac{1}{2}(\Theta - \widehat{\Theta})^2$$

Differentiating V_1^1 with respect to time yields

$$\dot{V}_1^1 = x_1^1(x_2^1 + \phi_1^{11} + \sum_{s=2}^{m} x_1^s \phi_1^{1s}) - (\Theta - \widehat{\Theta})\,\dot{\widehat{\Theta}} \tag{2.7}$$

Then, by (2.4)

$$\left| x_1^1 \phi_1^{11} \right| \leq (x_1^1)^2 \gamma_1^{11} \Theta \tag{2.8}$$

(2.7) takes the form

$$\dot{V}_1^1 \leq x_1^1(x_2^1 + x_1^1\gamma_1^{11}\Theta) - (\Theta - \widehat{\Theta})\,\dot{\widehat{\Theta}} + x_1^1 \sum_{s=2}^{m} x_1^s \phi_1^{1s} \tag{2.9}$$

By introducing the virtual controller

$$\alpha_1^1(x_1^1, \widehat{\Theta}) = -x_1^1 \beta_1^1 \tag{2.10}$$

with $\beta_1^1(x_1^1, \widehat{\Theta}) = c_1^1 + \gamma_1^{11}\widehat{\Theta}$,

$$\dot{V}_1^1 \leq -c_1^1(x_1^1)^2 + x_1^1(x_2^1 - \alpha_1^1) + (\tau_1^1 - \dot{\widehat{\Theta}})(\Theta - \widehat{\Theta}) + \sum_{s=2}^{m} x_1^s \Phi_1^{1s} \tag{2.11}$$

with $\tau_1^1(x_1^1) = (x_1^1)^2\gamma_1^{11}$ and $\Phi_1^{1s} = x_1^1\phi_1^{1s}$ for $s = 2, \cdots, m$.

Claim 2.5 *For each $1 \leq r \leq k$ and $2 \leq s \leq m$, there exist a virtual controller $\alpha_r^1(x_1^1, \cdots, x_r^1) = -\xi_r^1 \beta_r^1(x_1^1, \cdots, x_r^1)$ and tuning functions τ_r^1, λ_r^1 and Φ_r^{1s}, such that the time derivative of $V_r^1 = \frac{1}{2}\sum_{j=1}^{r}(x_j^1 - \alpha_{j-1}^1)^2$ satisfies*

$$\dot{V}_r^1 \leq -\sum_{j=1}^{r}\left[c_j^1 - \frac{3}{2}(r-j)\right](\xi_j^1)^2 + \xi_r^1(x_{r+1}^1 - \alpha_r^1) + (\tau_r^1 - \dot{\widehat{\Theta}})(\Theta - \widehat{\Theta} + \lambda_r^1) + \sum_{s=2}^{m} x_1^s \Phi_r^{1s} \tag{2.12}$$

Proof of Claim 2.5

We have already proved that Claim 2.5 holds for $r = 1$. Assume that Claim 1 hold for $r = k$. Now, we show that this claim also holds for $r = k + 1$.

Step$1.k+1$: Consider the Lyapunov function candidate

$$V_{k+1}^1 = V_k^1 + \frac{1}{2}(x_{k+1}^1 - \alpha_k^1)^2$$

Differentiating V_{k+1}^1 with respect to time yields

$$\begin{aligned} \dot{V}_{k+1}^1 \leq & -\sum_{j=1}^{k}\left[c_j^1 - \frac{3}{2}(k-j)\right](\xi_j^1)^2 + \xi_k^1(x_{k+1}^1 - \alpha_k^1) + (\tau_k^1 - \dot{\widehat{\Theta}})(\Theta - \widehat{\Theta} + \lambda_k^1) \\ & + \sum_{s=2}^{m} x_1^s \Phi_k^{1s} + \xi_{k+1}^1 \left[x_{k+2}^1 + \phi_{k+1}^{11} + \sum_{s=2}^{m} x_1^s \phi_{k+1}^{1s} \right. \\ & \left. - \sum_{j=1}^{k} \frac{\partial \alpha_k^1}{\partial x_j^1}(x_{j+1}^1 + \phi_j^{11} + \sum_{s=2}^{m} x_1^s \phi_j^{1s}) - \frac{\partial \alpha_k^1}{\partial \widehat{\Theta}} \dot{\widehat{\Theta}} \right] \end{aligned} \tag{2.13}$$

for $s = 2, \cdots, m$.

By (2.4), it follows that

$$\begin{aligned} & \xi_{k+1}^1 \left[\xi_k^1 + \phi_{k+1}^{11} - \sum_{j=1}^{k} \frac{\partial \alpha_k^1}{\partial x_j^1}(x_{j+1}^1 + \phi_j^{11})\right] \\ \leq & \left|\xi_{k+1}^1\right| \left[\left|\xi_k^1\right| + \sum_{j=1}^{k+1} \left|x_j^1\right| \gamma_{k+1}^{11} + \sum_{j=1}^{k} \left|\frac{\partial \alpha_k^1}{\partial x_j^1}\right| \left(\left|x_{j+1}^1\right| + \sum_{l=1}^{j} \left|x_l^1\right| \gamma_j^{11}\right)\right] \Theta \\ \leq & \left|\xi_{k+1}^1\right| \left[\left|\xi_k^1\right| + \sum_{j=1}^{k+1} \left|x_j^1\right| \widetilde{w}_{k+1}^1\right] \Theta \\ \leq & \left|\xi_{k+1}^1\right| \sum_{j=1}^{k+1} \left|\xi_j^1\right| w_{k+1}^1 \Theta \\ \leq & \frac{1}{2}\sum_{j=1}^{k}(\xi_j^1)^2 + (\xi_{k+1}^1)^2 \rho_{k+1}^1 + A_{k+1}^1(\Theta - \widehat{\Theta}) \end{aligned} \tag{2.14}$$

where

$$\begin{aligned} \widetilde{w}_{k+1}^1(x_1^1, \cdots, x_{k+1}^1) &= \gamma_{k+1}^{11} + \sum_{j=1}^{k} \left|\frac{\partial \alpha_k^1}{\partial x_j^1}\right| \gamma_j^{11} \\ w_{k+1}^1(x_1^1, \cdots, x_{k+1}^1, \widehat{\Theta}) &= (1 + \beta_1^1 + \cdots + \beta_k^1)\widetilde{w}_{k+1}^1 \end{aligned}$$

$$
\begin{aligned}
\rho_{k+1}^1(x_1^1,\cdots,x_{k+1}^1,\widehat{\Theta}) &= \frac{1}{2}+\frac{k+1}{2}(w_{k+1}^1)^2\widehat{\Theta}^2 \\
A_{k+1}^1(x_1^1,\cdots,x_{k+1}^1,\widehat{\Theta}) &= \frac{1}{2\sqrt{1+(\lambda_{k+1}^1)^2}}\sum_{j=1}^{k}(\xi_j^1)^2+(\xi_{k+1}^1)^2\left[\frac{1}{2}+\frac{k+1}{2}(w_{k+1}^1)^2\right]\sqrt{1+(\lambda_{k+1}^1)^2}
\end{aligned}
$$

and λ_{k+1}^1 will be given later. The last inequality of (2.14) comes from Lemma 2.2.

Thus, (2.13) takes the form

$$
\begin{aligned}
\dot{V}_{k+1}^1 \leq\; & -\sum_{j=1}^{k}\left[c_j^1-\frac{3}{2}(k-j)-\frac{1}{2}\right](\xi_j^1)^2+\xi_{k+1}^1(x_{k+2}^1+\xi_{k+1}^1\rho_{k+1}^1) \\
& +(\tau_{k+1}^1-\dot{\widehat{\Theta}})(\Theta-\widehat{\Theta}+\lambda_{k+1}^1)-A_{k+1}^1\lambda_{k+1}^1-\xi_{k+1}^1\frac{\partial\alpha_k^1}{\partial\widehat{\Theta}}\tau_k^1+\sum_{s=2}^{m}x_1^s\Phi_{k+1}^{1s} \quad (2.15)
\end{aligned}
$$

where

$$
\begin{aligned}
\lambda_{k+1}^1 &= \lambda_k^1+\xi_{k+1}^1\frac{\partial\alpha_k^1}{\partial\widehat{\Theta}}=\sum_{j=1}^{k}\xi_{j+1}^1\frac{\partial\alpha_j^1}{\partial\widehat{\Theta}} && (2.16) \\
\Phi_{k+1}^{1s} &= \Phi_k^{1s}+\xi_{k+1}^1(\phi_{k+1}^{1s}-\sum_{j=1}^{k}\frac{\partial\alpha_k^1}{\partial x_j^1}\phi_j^{1s})=\sum_{l=1}^{k+1}\left(\xi_l^1-\sum_{j=l}^{k}\xi_{j+1}^1\frac{\partial\alpha_j^1}{\partial x_l^1}\right)\phi_l^{1s} && (2.17) \\
\tau_{k+1}^1 &= \tau_k^1+A_{k+1}^1=(x_1^1)^2\gamma_1^{11} \\
&\quad +\sum_{l=2}^{k+1}\left(\frac{1}{2\sqrt{1+(\lambda_l^1)^2}}\sum_{j=1}^{l-1}(\xi_j^1)^2+(\xi_l^1)^2\left[\frac{1}{2}+\frac{p}{2}(w_l^1)^2\right]\sqrt{1+(\lambda_l^1)^2}\right) && (2.18)
\end{aligned}
$$

As for the terms $-A_{k+1}^1\lambda_{k+1}^1$ and $-\xi_{k+1}^1\frac{\partial\alpha_k^1}{\partial\widehat{\Theta}}\tau_k^1$ in (2.15), there exist nonnegative smooth functions $\widetilde{\beta}_{k+1}^1(x_1^1,\cdots,x_{k+1}^1,\widehat{\Theta})$ and $\overline{\beta}_{k+1}^1(x_1^1,\cdots,x_{k+1}^1,\widehat{\Theta})$, such that

$$
\begin{aligned}
\left|A_{k+1}^1\lambda_{k+1}^1\right| &\leq \frac{1}{2}\sum_{j=1}^{k}(\xi_j^1)^2+(\xi_{k+1}^1)^2\widetilde{\beta}_{k+1}^1 && (2.19) \\
\left|\xi_{k+1}^1\frac{\partial\alpha_k^1}{\partial\widehat{\Theta}}\tau_k^1\right| &\leq \frac{1}{2}\sum_{j=1}^{k}(\xi_j^1)^2+(\xi_{k+1}^1)^2\overline{\beta}_{k+1}^1 && (2.20)
\end{aligned}
$$

By introducing the virtual controller

$$
\alpha_{k+1}^1(x_1^1,\cdots,x_{k+1}^1,\widehat{\Theta})=-\xi_{k+1}^1\beta_{k+1}^1 \quad (2.21)
$$

with $\beta^1_{k+1}(x^1_1, \cdots, x^1_{k+1}, \widehat{\Theta}) = c^1_{k+1} + \rho^1_{k+1} + \widetilde{\beta}^1_{k+1} + \overline{\beta}^1_{k+1}$,

$$\dot{V}^1_{k+1} \leq -\sum_{j=1}^{k+1}[c^1_j - \frac{3}{2}(k+1-j)](\xi^1_j)^2 + \xi^1_{k+1}(x^1_{k+2} - \alpha^1_{k+1}) + (\tau^1_{k+1} - \dot{\widehat{\Theta}})(\Theta - \widehat{\Theta} + \lambda^1_{k+1}) + \sum_{s=2}^{m} x^s_1 \Phi^{1s}_{k+1} \tag{2.22}$$

This completes the proof of Claim 2.5.

Claim 2.5 holds until $r = n_1$. At *Step* $1.n_1$, the time derivative of $V^1_{n_1} = \frac{1}{2}\sum_{j=1}^{n_1}(x^1_j - \alpha^1_{j-1})^2$ satisfies

$$\dot{V}^1_{n_1} \leq -\sum_{j=1}^{n_1}\left[c^1_j - \frac{3}{2}(n_1 - j)\right](\xi^1_j)^2 + \xi^1_{n_1}(x^1_{n_1+1} - \alpha^1_{n_1}) + (\tau^1_{n_1} - \dot{\widehat{\Theta}})(\Theta - \widehat{\Theta} + \lambda^1_{n_1}) + \sum_{s=2}^{m} x^s_1 \Phi^{1s}_{n_1}$$

with $x^1_{n_1+1} = u_1$.

Claim 2.6 *For each $1 \leq r \leq n_t$ $(1 \leq t \leq i-1)$ and $1 \leq r \leq k$ $(t = i)$, there exist a virtual controller $\alpha^t_r(\overline{x}^1, \cdots, \overline{x}^{t-1}, x^t_1, \cdots, x^t_r) = -\xi^t_r \beta^t_r(\overline{x}^1, \cdots, \overline{x}^{t-1}, x^t_1, \cdots, x^t_r)$ and tuning functions τ^t_r, λ^t_r and Φ^{ts}_r $(t+1 \leq s \leq m)$, such that the time derivative of $V^t_r = \frac{1}{2}\sum_{p=1}^{t-1}\sum_{j=1}^{n_p}(x^p_j - \alpha^p_{j-1})$ $^2 + \frac{1}{2}\sum_{j=1}^{r}(x^t_j - \alpha^t_{j-1})$ ^{2} satisfies*

$$\begin{aligned}
\dot{V}^t_r \;\; \leq \;\; & -\sum_{p=1}^{t-1}\sum_{j=1}^{n_p}[c^p_j - \frac{3}{2}(\sum_{l=p}^{t-1} n_l + r - j)](\xi^p_j)^2 - \sum_{j=1}^{r}[c^t_j - \frac{3}{2}(r-j)](\xi^t_j)^2 + \sum_{p=1}^{t-1}\xi^p_{n_p}(u_p - \alpha^p_{n_p}) \\
& + \xi^t_r(x^t_{r+1} - \alpha^t_r) + (\tau^1_r - \dot{\widehat{\Theta}})(\Theta - \widehat{\Theta} + \lambda^1_r) + \sum_{s=p+1}^{m} x^s_1 \Phi^{ps}_r
\end{aligned} \tag{2.23}$$

Proof of Claim 2.6

Assume Claim 2.6 hold for $t = i$, $r = k$, we show that the claim also holds for $t = i$, $r = k+1$.

Step $i.k+1$: Consider the Lyapunov function candidate

$$V^i_{k+1} = V^i_k + \frac{1}{2}(x^i_{k+1} - \alpha^i_k)^2$$

Differentiating V^i_{k+1} with respect to time yields

$$\dot{V}^i_{k+1} \;\; \leq \;\; -\sum_{p=1}^{i-1}\sum_{j=1}^{n_p}[c^p_j - \frac{3}{2}(\sum_{l=p}^{i-1} n_l + k - j)](\xi^p_j)^2 - \sum_{j=1}^{k}[c^i_j - \frac{3}{2}(k-j)](\xi^i_j)^2 + \sum_{p=1}^{i-1}\xi^p_{n_p}(u_p - \alpha^p_{n_p})$$

$$
\begin{aligned}
&+\xi_k^i(x_{k+1}^i-\alpha_k^i)+(\tau_k^i-\dot{\widehat{\Theta}})(\Theta-\widehat{\Theta}+\lambda_k^i)+\sum_{s=i+1}^m x_1^s\Phi_k^{is}\\
&+(x_{k+1}^i-\alpha_k^i)\Bigg[x_{k+2}^i+\phi_{k+1}^{ii}+\sum_{s=i+1}^m x_1^s\phi_{k+1}^{is}-\sum_{p=1}^{i-1}\sum_{j=1}^{n_p}\frac{\partial\alpha_k^i}{\partial x_j^p}(x_{j+1}^p+\phi_j^{pp}+\sum_{s=p+1}^m x_1^s\phi_j^{ps})\\
&-\sum_{j=1}^k\frac{\partial\alpha_k^i}{\partial x_j^i}(x_{j+1}^i+\phi_j^{ii}+\sum_{s=i+1}^m x_1^s\phi_j^{is})-\frac{\partial\alpha_k^i}{\partial\widehat{\Theta}}\dot{\widehat{\Theta}}\Bigg] \qquad (2.24)
\end{aligned}
$$

where $x_{n_p+1}^p=u_p$ for $p=1,\cdots,i-1$. Rewriting (2.24), we have

$$
\begin{aligned}
\dot V_{k+1}^i \;\le\;& -\sum_{p=1}^{i-1}\sum_{j=1}^{n_p}[c_j^p-\frac{3}{2}(\sum_{l=p}^{i-1}n_l+k-j)](\xi_j^p)^2-\sum_{j=1}^k[c_j^t-\frac{3}{2}(k-j)](\xi_j^i)^2+\sum_{p=1}^{i-1}\xi_{n_p}^p(u_p-\alpha_{n_p}^p)\\
&+(\tau_k^i-\dot{\widehat{\Theta}})(\Theta-\widehat{\Theta}+\lambda_k^i)+\sum_{s=i+1}^m x_1^s\Phi_{k+1}^{is}+\xi_{k+1}^i\Big[\xi_k^i+x_{k+2}^i+\phi_{k+1}^{ii}\\
&-\sum_{p=1}^{i-1}\sum_{j=1}^{n_p}\frac{\partial\alpha_k^i}{\partial x_j^p}(x_{j+1}^p+\phi_j^{pp}+\sum_{s=p+1}^{i}x_1^s\phi_j^{ps})-\sum_{j=1}^k\frac{\partial\alpha_k^i}{\partial x_j^i}(x_{j+1}^i+\phi_j^{ii})-\frac{\partial\alpha_k^i}{\partial\widehat{\Theta}}\dot{\widehat{\Theta}}\Big] \qquad (2.25)
\end{aligned}
$$

where

$$
\begin{aligned}
\Phi_{k+1}^{is} &= \Phi_k^{is}+\xi_{k+1}^i\left(\phi_{k+1}^{is}-\sum_{j=1}^k\frac{\partial\alpha_k^i}{\partial x_j^i}\phi_j^{is}-\sum_{p=1}^{i-1}\sum_{j=1}^{n_p}\frac{\partial\alpha_k^i}{\partial x_j^p}\phi_j^{ps}\right)\\
&= \sum_{l=1}^{k+1}\left(\xi_l^i-\sum_{j=l}^k\xi_{j+1}^i\frac{\partial\alpha_j^i}{\partial x_l^i}\right)\phi_l^{is}+\sum_{l=1}^{n_p}\left(\xi_l^p-\sum_{q=i-p}^{p}\sum_{j=l}^{n_p-1}\xi_{j+1}^q\frac{\partial\alpha_j^q}{\partial x_l^q}-\sum_{j=1}^k\xi_{j+1}^i\frac{\partial\alpha_j^i}{\partial x_l^p}\right)\phi_l^{ps}
\end{aligned}
$$

It follows from (2.4) and (2.5) that

$$
\begin{aligned}
&\xi_{k+1}^i\left[\xi_k^i+\phi_{k+1}^{ii}-\sum_{p=1}^{i-1}\sum_{j=1}^{n_p}\frac{\partial\alpha_k^i}{\partial x_j^p}(x_{j+1}^p+\phi_j^{pp}+\sum_{s=p+1}^{i}x_1^s\phi_j^{ps})-\sum_{j=1}^k\frac{\partial\alpha_k^i}{\partial x_j^i}(x_{j+1}^i+\phi_j^{ii})\right]\\
\le\;& \left|\xi_{k+1}^i\right|\left(\sum_{p=1}^{i-1}\sum_{j=1}^{n_p}\left|x_j^p\right|+\sum_{j=1}^{k+1}\left|x_j^i\right|\right)\widetilde{w}_{k+1}^i\Theta\\
\le\;& \left|\xi_{k+1}^i\right|\left(\sum_{p=1}^{i-1}\sum_{j=1}^{n_p}\left|\xi_j^p\right|+\sum_{j=1}^{k+1}\left|\xi_j^i\right|\right)w_{k+1}^i\Theta\\
\le\;& \frac{1}{2}\left(\sum_{p=1}^{i-1}\sum_{j=1}^{n_p}(\xi_j^p)^2+\sum_{j=1}^k(\xi_j^i)^2\right)+(\xi_{k+1}^i)^2\rho_{k+1}^i+A_{k+1}^i(\Theta-\widehat{\Theta}) \qquad (2.26)
\end{aligned}
$$

with

$$\rho_{k+1}^{i} = \frac{1}{2} + \frac{1}{2}\left(\sum_{j=1}^{i-1} n_j + k + 1\right)(w_{k+1}^{i})^2 \widehat{\Theta}^2 \tag{2.27}$$

$$\begin{aligned} A_{k+1}^{i} &= \frac{1}{2\sqrt{1+(\lambda_{k+1}^{i})^2}}\left(\sum_{p=1}^{i-1}\sum_{j=1}^{n_p}(\xi_j^p)^2 + \sum_{j=1}^{k}(\xi_j^i)^2\right) + \\ &\quad (\xi_{k+1}^{i})^2\left[\frac{1}{2} + \frac{1}{2}\left(\sum_{j=1}^{i-1} n_j + k + 1\right)(w_{k+1}^{i})^2\right]\sqrt{1+(\lambda_{k+1}^{i})^2} \end{aligned} \tag{2.28}$$

Lemma 2.2 is used in the last inequality of (2.26).

Thus, (2.25) takes the form

$$\begin{aligned} \dot{V}_{k+1}^{i} \leq & -\sum_{p=1}^{i-1}\sum_{j=1}^{n_p}[c_j^p - \frac{3}{2}(\sum_{l=p}^{i-1} n_l + k - j) - \frac{1}{2}](\xi_j^p)^2 - \sum_{j=1}^{k}[c_j^t - \frac{3}{2}(k-j) - \frac{1}{2}](\xi_j^i)^2 + \sum_{p=1}^{i-1}\xi_{n_p}^p(u_p - \alpha_{n_p}^p) \\ & +\xi_{k+1}^{i}(x_{k+2}^{i} + \xi_{k+1}^{i}\rho_{k+1}^{i}) + (\tau_{k+1}^{i} - \dot{\widehat{\Theta}})(\Theta - \widehat{\Theta} + \lambda_{k+1}^{i}) + \sum_{s=i+1}^{m} x_1^s \Phi_{k+1}^{is} \\ & -A_{k+1}^{i}\lambda_{k+1}^{i} - \xi_{k+1}^{i}\frac{\partial\alpha_k^i}{\partial\widehat{\Theta}}\tau_k^i \end{aligned} \tag{2.29}$$

where

$$\lambda_{k+1}^{i} = \lambda_k^i + \xi_{k+1}^{i}\frac{\partial\alpha_k^i}{\partial\widehat{\Theta}} \tag{2.30}$$

$$\tau_{k+1}^{i} = \tau_k^i + A_{k+1}^{i} \tag{2.31}$$

As for the terms $-A_{k+1}^{i}\lambda_{k+1}^{i}$ and $-\xi_{k+1}^{i}\frac{\partial\alpha_k^i}{\partial\widehat{\Theta}}\tau_k^i$ in (2.29), there exist nonnegative smooth functions $\widetilde{\beta}_{k+1}^{i}$ and $\overline{\beta}_{k+1}^{i}$, such that

$$\left|A_{k+1}^{i}\lambda_{k+1}^{i}\right| \leq \frac{1}{2}\sum_{p=1}^{i-1}\sum_{j=1}^{n_p}(\xi_j^p)^2 + \sum_{j=1}^{k}(\xi_j^i)^2 + (\xi_{k+1}^{i})^2\widetilde{\beta}_{k+1}^{i} \tag{2.32}$$

$$\left|\xi_{k+1}^{i}\frac{\partial\alpha_k^i}{\partial\widehat{\Theta}}\tau_k^i\right| \leq \frac{1}{2}\sum_{p=1}^{i-1}\sum_{j=1}^{n_p}(\xi_j^p)^2 + \sum_{j=1}^{k}(\xi_j^i)^2 + (\xi_{k+1}^{i})^2\overline{\beta}_{k+1}^{i} \tag{2.33}$$

By introducing the virtual controller

$$\alpha_{k+1}^{i}(\overline{x}^1, \cdots, \overline{x}^{i-1}, x_1^i, \cdots, x_{k+1}^{i}, \widehat{\Theta}) = -\xi_{k+1}^{i}\beta_{k+1}^{i} \tag{2.34}$$

with $\beta^i_{k+1}(\overline{x}^1,\cdots,\overline{x}^{i-1},x^i_1,\cdots,x^i_{k+1},\widehat{\Theta}) = c^i_{k+1}+\rho^i_{k+1}+\widetilde{\beta}^i_{k+1}+\overline{\beta}^i_{k+1}$,

$$\begin{aligned}\dot{V}^i_{k+1} \quad \leq \quad & -\sum_{p=1}^{i-1}\sum_{j=1}^{n_p}[c^p_j-\frac{3}{2}(\sum_{l=p}^{i-1}n_l+k+1-j)](\xi^p_j)^2-\sum_{j=1}^{k+1}[c^t_j-\frac{3}{2}(k+1-j)](\xi^i_j)^2+\sum_{p=1}^{i-1}\xi^p_{n_p}(u_p-\alpha^p_{n_p}) \\ & +\xi^i_{k+1}(x^i_{k+2}-\alpha^i_{k+1})+(\tau^i_{k+1}-\dot{\widehat{\Theta}})(\Theta-\widehat{\Theta}+\lambda^i_{k+1})+\sum_{s=i+1}^{m}x^s_1\Phi^{is}_{k+1}\end{aligned} \tag{2.35}$$

which completes the proof of Claim 2.6.

Claim 2.6 holds for each $r = n_t$ $(1 \leq t \leq m-1)$ and at each Step $t.n_t$ the time derivative of $V^t_{n_t} = \frac{1}{2}\sum_{p=1}^{t}\sum_{j=1}^{n_t}(x^p_j-\alpha^p_{j-1})^2$ satisfies

$$\dot{V}^t_n \leq -\sum_{p=1}^{t}\sum_{j=1}^{n_p}[c^p_j-\frac{3}{2}(\sum_{l=p}^{t}n_l-j)](\xi^p_j)^2+\sum_{p=1}^{t}\xi^p_{n_p}(u_p-\alpha^t_{n_p})+(\tau^t_{n_t}-\dot{\widehat{\Theta}})(\Theta-\widehat{\Theta}+\lambda^t_{n_t})+\sum_{s=t+1}^{m}x^s_1\Phi^{ts}_{n_t} \tag{2.36}$$

and for $t = m$, $r = n_m$ at Step $m.n_m$ the time derivative of $V^m_{n_m} = \frac{1}{2}\sum_{p=1}^{m}\sum_{j=1}^{n_t}(x^p_j-\alpha^p_{j-1})^2$ satisfies

$$\dot{V}^m_{n_m} \leq -\sum_{p=1}^{m}\sum_{j=1}^{n_p}[c^p_j-\frac{3}{2}(\sum_{l=p}^{m}n_l-j)](\xi^p_j)^2+\sum_{p=1}^{m}\xi^p_{n_p}(u_p-\alpha^t_{n_p})+(\tau^m_{n_m}-\dot{\widehat{\Theta}})(\Theta-\widehat{\Theta}+\lambda^m_{n_m}) \tag{2.37}$$

with $x^p_{n_1+1} = u_p$.

By choosing real positive numbers c^p_j to make sure the term $c^p_j - \frac{3}{2}(\sum_{l=p}^{m}n_l - j) > 0$ for $1 \leq p \leq m, 1 \leq j \leq n_p$, it is obvious that the controller

$$\begin{cases} u_1 = \alpha^1_{n_1}(\overline{x}^1,\widehat{\Theta}) \\ \vdots \\ u_i = \alpha^1_{n_i}(\overline{x}^1,\cdots,\overline{x}^i,\widehat{\Theta}) \\ \vdots \\ u_m = \alpha^1_{n_m}(\overline{x}^1,\cdots,\overline{x}^m,\widehat{\Theta}) \end{cases} \tag{2.38}$$

and the estimator

$$\dot{\widehat{\Theta}} = \tau^m_{n_m}(\overline{x}^1,\cdots,\overline{x}^m,\widehat{\Theta}) \tag{2.39}$$

make $\dot{V}^m_{n_m}$ negative definite, therefore guarantee the asymptotic stability of the closed-loop system (2.1), (2.38) and (2.39).

2.4 Numerical Example

In this section, one numerical example is studied to illustrate the design methodology developed in this chapter. Consider one two-input system with nonlinear parameterization

$$\begin{aligned}
\dot{x}_1^1 &= 2x_2^1 + 3x_1^2 \\
\dot{x}_2^1 &= u_1 + \phi_2^{11}(x_1^1, x_2^1, \theta_1) + x_1^2\phi_2^{12}(x_1^2, \theta_2) \\
\dot{x}_1^2 &= x_2^2 + \phi_1^{22}(x_1^1, x_2^1, x_1^2, \theta_3) \\
\dot{x}_2^2 &= u_2
\end{aligned} \tag{2.40}$$

where $\phi_2^{11} = x_1^1[1+(x_2^1)^{2/3}](\theta_1)^{x_1^1}$, $\theta_1 > 0$, $\phi_2^{12} = \ln[1+(\theta_2 x_1^2)^2]$, $\phi_1^{22} = [(x_1^1)^2 + (x_2^1)^2]\left|x_1^2\right|^{\theta_3}$ and the true values of θ_1, θ_2 and θ_3 are 2, 1 and 0.8 respectively.

In the system (2.40), the unknown parameters θ_1, θ_2 and θ_3 enter the functions ϕ_2^{11}, ϕ_2^{12} and ϕ_1^{22} nonlinearly and the subsystem (x_1^1, x_2^1) is nested in the larger subsystem (x_1^1, x_2^1, x_1^2) through the interconnection $x_1^2\phi_2^{12}$.

For the function ϕ_2^{11}, it follows that

$$\begin{aligned}
\left|\phi_2^{11}\right| &\le \left|x_1^1\right| |\theta_1|^{x_1^1} + \left|x_1^1\right| \left|x_2^1\right|^{2/3} |\theta_1|^{x_1^1} \\
&\le \left|x_1^1\right| |\theta_1|^{x_1^1} + \frac{2}{3}\left|x_2^1\right| + \frac{1}{3}\left|x_1^1\right|^3 |\theta_1|^{3x_1^1} \\
&\le \frac{2}{3}\left|x_2^1\right| + \left|x_1^1\right| [e^{\frac{1}{2}(x_1^1)^2 + \frac{1}{2}\ln^2|\theta_1|} + \frac{1}{3}(x_1^1)^2 e^{\frac{9}{2}(x_1^1)^2 + \frac{1}{2}\ln^2|\theta_1|}] \\
&\le \frac{2}{3}\left|x_2^1\right| + \left|x_1^1\right| [e^{\frac{1}{2}(x_1^1)^2} + \frac{1}{3}(x_1^1)^2 e^{\frac{9}{2}(x_1^1)^2}] e^{\frac{1}{2}\ln^2|\theta_1|} \\
&\le (\left|x_1^1\right| + \left|x_2^1\right|)\gamma_2^{11}(x_1^1) e^{\frac{1}{2}\ln^2|\theta_1|}
\end{aligned}$$

with $\gamma_2^{11}(x_1^1) = e^{\frac{1}{2}(x_1^1)^2} + \frac{1}{3}(x_1^1)^2 e^{\frac{9}{2}(x_1^1)^2} \ge 1$ and $e^{\frac{1}{2}\ln^2|\theta_1|} \ge 1$. Lemma 2.2 is applied during the inequality derivation above.

For the function ϕ_2^{12}, $\phi_2^{12} \le |\theta_2|\left|x_1^2\right|$. For the function ϕ_1^{22}, it follows that

$$\left|\phi_1^{22}\right| \le \frac{2}{3}\left|x_1^1\right|^3 + \frac{2}{3}\left|x_2^1\right|^3 + \frac{2}{3}\left|x_1^2\right|^{3\theta_3}$$

$$
\begin{aligned}
&\leq \frac{2}{3}\left|x_1^1\right|^3+\frac{2}{3}\left|x_2^1\right|^3+\frac{2}{3}\left|x_1^2\right|e^{\frac{1}{2}(3\theta_3-1)\ln[1+(x_1^2)^2]}\\
&\leq \frac{2}{3}\left|x_1^1\right|^3+\frac{2}{3}\left|x_2^1\right|^3+\frac{2}{3}\left|x_1^2\right|e^{\frac{1}{2}\ln^2[1+(x_1^2)^2]}e^{\frac{1}{8}(3\theta_3-1)^2}\\
&\leq (\left|x_1^1\right|+\left|x_2^1\right|+\left|x_1^2\right|)\gamma_1^{22}(x_1^1,x_2^1,x_1^2)e^{\frac{1}{8}(3\theta_3-1)^2}
\end{aligned}
$$

where $\gamma_1^{22}(x_1^1,x_2^1,x_1^2)=\frac{2}{3}(x_1^1)^2+\frac{2}{3}(x_2^1)^2+e^{\frac{1}{2}\ln^2[1+(x_1^2)^2]}\geq 1$ and $e^{\frac{1}{8}(3\theta_3-1)^2}\geq 1$.

Let $\Theta=e^{\frac{1}{2}\ln^2|\theta_1|}+|\theta_2|+e^{\frac{1}{8}(3\theta_3-1)^2}\geq 1$ and it follows that

$$
\begin{aligned}
\phi_2^{11} &\leq (\left|x_1^1\right|+\left|x_2^1\right|)\gamma_2^{11}(x_1^1)\Theta\\
\phi_2^{12} &\leq \left|x_1^2\right|\Theta\\
\phi_1^{22} &\leq (\left|x_1^1\right|+\left|x_2^1\right|+\left|x_1^2\right|)\gamma_1^{22}(x_1^1,x_2^1,x_1^2)\Theta
\end{aligned}
\tag{2.41}
$$

Following the design procedure presented in the last section, the adaptive controller of the system (2.40) is constructed as follows.

Step 1.1*:* Define Lyapunov function candidate

$$V_1^1=\frac{1}{2}(x_1^1)^2$$

Differentiating V_1^1 with respect to time gives

$$\dot{V}_1^1=-c_1(x_1^1)^2+2x_1^1(x_2^1-\alpha_1^1)+3x_1^1x_1^2 \tag{2.42}$$

with

$$\alpha_1^1(x_1^1)=-\frac{1}{2}c_1x_1^1$$

Step 1.2*:* Consider the following Lyapunov function candidate

$$V_2^1=V_1^1+\frac{1}{2}(x_2^1-\alpha_1^1)^2+\frac{1}{2}(\Theta-\widehat{\Theta})^2 \tag{2.43}$$

Differentiating V_2^1 with respect to time yields

$$\dot{V}_2^1=-c_1^1(x_1^1)^2+\xi_2^1[2x_1^1+u_1+\phi_2^{11}-\frac{\partial\alpha_1^1}{\partial x_1^1}(2x_2^1+3x_1^2)]+x_1^2(3x_1^1+\xi_2^1\phi_2^{12})-(\Theta-\widehat{\Theta})\,\dot{\widehat{\Theta}} \tag{2.44}$$

with $\xi_2^1 = x_2^1 - \alpha_1^1$ and for $\xi_2^1\phi_2^{11}$, it follows that

$$\begin{aligned}
\left|\xi_2^1\phi_2^{11}\right| &\leq \left|\xi_2^1\right|(\left|x_1^1\right| + \left|x_2^1\right|)\gamma_2^{11}\Theta \\
&\leq \left|\xi_2^1\right|(\left|x_1^1\right| + \left|\xi_2^1\right| + \left|\alpha_1^1\right|)\gamma_2^{11}\Theta \\
&\leq \left|\xi_2^1\right|(\left|x_1^1\right| + \left|\xi_2^1\right|)(1 + \frac{1}{2}c_1)\gamma_2^{11}\Theta \\
&\leq \frac{1}{2}(x_1^1)^2 + (\xi_2^1)^2\rho_1(x_1^1, \widehat{\Theta}) + A_1(x_1^1, x_2^1)(\Theta - \widehat{\Theta})
\end{aligned} \tag{2.45}$$

where $\rho_1(x_1^1, \widehat{\Theta}) = \frac{1}{2} + (w_1)^2\widehat{\Theta}^2$ and $A_1(x_1^1, x_2^1) = \frac{1}{2}[(x_1^1)^2 + (\xi_1^1)^2] + (\xi_1^1)^2(w_1)^2$ with $w_1(x_1^1) = (1 + \frac{1}{2}c_1)\gamma_2^{11}$.

Thus, (2.44) produces

$$\dot{V}_2^1 = -(c_1^1 - \frac{1}{2})(x_1^1)^2 - c_2^1(\xi_2^1)^2 + x_1^2(3x_1^1 + \xi_2^1\phi_2^{12}) + (A_1 - \dot{\widehat{\Theta}})(\Theta - \widehat{\Theta}) \tag{2.46}$$

with

$$u_1(x_1^1, x_2^1, \widehat{\Theta}) = -c_2^1\xi_2^1 - 2x_1^1 - \xi_2^1\rho_1 + \frac{\partial\alpha_1}{\partial x_1^1}(2x_2^1 + 3x_1^2)$$

Step 2.1*:* Consider the following Lyapunov function candidate

$$V_1^2 = V_2^1 + \frac{1}{2}(x_1^2)^2$$

Differentiating V_1^2 with respect to time yields

$$\dot{V}_1^2 = -(c_1^1 - \frac{1}{2})(x_1^1)^2 - c_2^1(\xi_2^1)^2 + x_1^2(3x_1^1 + \xi_2^1\phi_2^{12} + x_2^2 + \phi_1^{22}) + (A_1 - \dot{\widehat{\Theta}})(\Theta - \widehat{\Theta}) \tag{2.47}$$

and for $x_1^2(\xi_2^1\phi_2^{12} + \phi_1^{22})$, it follows that

$$\begin{aligned}
\left|x_1^2(\xi_2^1\phi_2^{12} + \phi_1^{22})\right| &\leq \left|x_1^2\right|\left[\left|\xi_2^1\right|\left|x_1^2\right| + (\left|x_1^1\right| + \left|x_2^1\right| + \left|x_1^2\right|)\gamma_1^{22}\right]\Theta \\
&\leq \left|x_1^2\right|\left[\left|x_1^2\right|\sqrt{1 + (\xi_2^1)^2} + (\left|x_1^1\right| + \left|x_2^1\right| + \left|x_1^2\right|)\gamma_1^{22}\right]\Theta \\
&\leq \left|x_1^2\right|\left[\left|x_1^2\right|\sqrt{1 + (\xi_2^1)^2} + (\left|x_1^1\right| + \left|\xi_2^1\right| + \left|x_1^2\right|)(1 + \frac{1}{2}c_1)\gamma_1^{22}\right]\Theta \\
&\leq \left|x_1^2\right|(\left|x_1^1\right| + \left|\xi_2^1\right| + \left|x_1^2\right|)w_2(x_1^1, x_2^1, x_1^2)\Theta
\end{aligned}$$

$$\begin{aligned} &\le \frac{1}{2}(x_1^1)^2 + \frac{1}{2}(\xi_2^1)^2 + (x_1^2)^2\rho_2(x_1^1, x_2^1, x_1^2, \widehat{\Theta}) \\ &+A_2(x_1^1, x_2^1, x_1^2)(\Theta - \widehat{\Theta}) \end{aligned} \tag{2.48}$$

where $w_2(x_1^1, x_2^1, x_1^2) = \sqrt{1+(\xi_2^1)^2} + (1+\frac{1}{2}c_1^1)\gamma_1^{22}$, $\rho_2(x_1^1, x_2^1, x_1^2, \widehat{\Theta}) = \frac{1}{2} + \frac{3}{2}(w_2)^2\widehat{\Theta}^2$ and $A_2(x_1^1, x_2^1, x_1^2) = \frac{1}{2}[(x_1^1)^2 + (\xi_2^1)^2 + (x_1^2)^2] + \frac{3}{2}(w_2)^2(x_1^2)^2$.

Thus, it follows that

$$\dot{V}_1^2 = -(c_1^1-1)(x_1^1)^2 - (c_2^1 - \frac{1}{2})(\xi_2^1)^2 - c_1^2(x_1^2)^2 + x_1^2(x_2^2 - \alpha_1^2) + (A_1 + A_2 - \dot{\widehat{\Theta}})(\Theta - \widehat{\Theta}) \tag{2.49}$$

with

$$\alpha_1^2(x_1^1, x_2^1, x_1^2, \widehat{\Theta}) = -c_1^2 x_1^2 - 3x_1^1 - x_1^2\rho_2$$

Step 2.2: Consider the following Lyapunov function candidate

$$V_2^2 = V_1^2 + \frac{1}{2}(x_2^2 - \alpha_1^2)^2$$

Differentiating V_2^2 with respect to time yields

$$\begin{aligned} \dot{V}_2^2 &= -(c_1^1-1)(x_1^1)^2 - (c_2^1 - \frac{1}{2})(\xi_2^1)^2 - c_1^2(x_1^2)^2 + \xi_2^2\Bigg[x_1^2 + u_2 - \frac{\partial \alpha_1^2}{\partial x_1^1}(2x_2^1 + 3x_1^2) \\ &-\frac{\partial \alpha_1^2}{\partial x_2^1}(u_1 + \phi_2^{11} + x_1^2\phi_2^{12}) - \frac{\partial \alpha_1^2}{\partial x_1^2}(x_2^2 + \phi_1^{22}) - \frac{\partial \alpha_1^2}{\partial \widehat{\Theta}}\dot{\widehat{\Theta}}\Bigg] + (A_1 + A_2 - \dot{\widehat{\Theta}})(\Theta - \widehat{\Theta}) \end{aligned} \tag{2.50}$$

where

$$\begin{aligned} \xi_2^2 &= x_2^2 - \alpha_1^2 \\ \frac{\partial \alpha_1^2}{\partial x_1^1} &= -3 - 3x_1^2 w_2\left[\frac{1}{2}c_1^1\frac{\xi_2^1}{\sqrt{1+(\xi_2^1)^2}} + (1+\frac{1}{2}c_1^1)\frac{4}{3}x_1^1\right]\widehat{\Theta}^2 \\ \frac{\partial \alpha_1^2}{\partial x_2^1} &= -3x_1^2 w_2\left[\frac{\xi_2^1}{\sqrt{1+(\xi_2^1)^2}} + (1+\frac{1}{2}c_1^1)\frac{4}{3}x_2^1\right]\widehat{\Theta}^2 \\ \frac{\partial \alpha_1^2}{\partial x_1^2} &= -c_1^2 - \rho_2 - 6w_2(1+\frac{1}{2}c_1^1)e^{\frac{1}{2}\ln^2[1+(x_1^2)^2]}\ln[1+(x_1^2)^2]\frac{(x_1^2)^2}{1+(x_1^2)^2}\widehat{\Theta}^2 \\ \frac{\partial \alpha_1^2}{\partial \widehat{\Theta}} &= -3x_1^2(w_2)^2\widehat{\Theta} \end{aligned}$$

Let

$$
\begin{aligned}
q_2^1 &= 3\sqrt{1+(x_1^2)^2}w_2\left[1+(1+\frac{1}{2}c_1^1)\frac{4}{3}\sqrt{1+(x_1^2)^2}\right]\widehat{\Theta}^2 \\
q_1^2 &= c_1^2+\rho_2+6w_2(1+\frac{1}{2}c_1^1)e^{\frac{1}{2}\ln^2[1+(x_1^2)^2]}\ln[1+(x_1^2)^2]\widehat{\Theta}^2
\end{aligned}
$$

Then $\left|\frac{\partial\alpha_1^2}{\partial x_2^1}\right| \le q_2^1$ and $\left|\frac{\partial\alpha_1^2}{\partial x_1^2}\right| \le q_1^2$. Therefore, for $\xi_2^2\left[\frac{\partial\alpha_1^2}{\partial x_2^1}(\phi_2^{11}+x_1^2\phi_2^{12})+\frac{\partial\alpha_1^2}{\partial x_1^2}\phi_1^{22}\right]$, it follows that

$$
\begin{aligned}
&\left|\xi_2^2[\frac{\partial\alpha_1^2}{\partial x_2^1}(\phi_2^{11}+x_1^2\phi_2^{12})+\frac{\partial\alpha_1^2}{\partial x_1^2}\phi_1^{22}]\right| \qquad (2.51)\\
\le\ & \left|\xi_2^2\right|\left[\left|\frac{\partial\alpha_1^2}{\partial x_2^1}\right|[(\left|x_1^1\right|+\left|x_2^1\right|)\gamma_2^{11}+(x_1^2)^2]+\left|\frac{\partial\alpha_1^2}{\partial x_2^1}\right|(\left|x_1^1\right|+\left|x_2^1\right|+\left|x_1^2\right|)\gamma_1^{22}\right]\Theta \\
\le\ & \left|\xi_2^2\right|\left[\left|\frac{\partial\alpha_1^2}{\partial x_2^1}\right|(\left|x_1^1\right|+\left|x_2^1\right|+\left|x_1^2\right|)(\sqrt{1+(x_1^2)^2}+\gamma_2^{11})+\left|\frac{\partial\alpha_1^2}{\partial x_2^1}\right|(\left|x_1^1\right|+\left|x_2^1\right|+\left|x_1^2\right|)\gamma_1^{22}\right]\Theta \\
\le\ & \left|\xi_2^2\right|(\left|x_1^1\right|+\left|\xi_2^1\right|+\left|x_1^2\right|)w_3\Theta \\
\le\ & \frac{1}{2}(x_1^1)^2+\frac{1}{2}(\xi_2^1)^2+\frac{1}{2}(x_1^2)^2+\frac{3}{2}(\xi_1^2)^2(w_3)^2\widehat{\Theta}^2+A_3(x_1^1,x_2^1,x_1^2,x_2^2)(\Theta-\widehat{\Theta})
\end{aligned}
$$

where $w_3(x_1^1,x_2^1,x_1^2,x_2^2)=(1+\frac{1}{2}c_1^1)\left[q_2^1\left(\sqrt{1+(x_1^2)^2}+\gamma_2^{11}\right)+q_1^2\gamma_1^{22}\right]$, $A_3(x_1^1,x_2^1,x_1^2,x_2^2)=\frac{1}{2}[(x_1^1)^2+(\xi_2^1)^2+(x_1^2)^2]+\frac{3}{2}(w_3)^2(\xi_2^2)^2$.

Thus, it follows from (2.50) that

$$
\dot{V}_2^2=-(c_1^1-\frac{3}{2})(x_1^1)^2-(c_2^1-1)(\xi_2^1)^2-(c_1^2-\frac{1}{2})(x_1^2)^2-c_2^2(\xi_2^2)^2+(\tau_3-\dot{\widehat{\Theta}})(\Theta-\widehat{\Theta}+\xi_2^2\frac{\partial\alpha_3}{\partial\widehat{\Theta}}) \quad (2.52)
$$

with

$$
\begin{aligned}
u_2(x_1^1,x_2^1,x_1^2,x_2^2,\widehat{\Theta}) &= -c_2^2\xi_2^2-x_1^2+\frac{\partial\alpha_1^2}{\partial x_1^1}(2x_2^1+3x_1^2)+\frac{\partial\alpha_1^2}{\partial x_2^1}u_1+\frac{\partial\alpha_1^2}{\partial x_1^2}x_2^2+\frac{\partial\alpha_1^2}{\partial\widehat{\Theta}}\tau_3-\frac{3}{2}\xi_2^2(w_3)^2\widehat{\Theta}^2 \\
\tau_3(x_1^1,x_2^1,x_1^2,x_2^2) &= A_1+A_2+A_3
\end{aligned}
$$

It is obvious that the feedback controller $u_1(x_1^1,x_2^1,\widehat{\Theta})$, $u_2(x_1^1,x_2^1,x_1^2,x_2^2,\widehat{\Theta})$ and the estimator $\dot{\widehat{\Theta}}=\tau_3(x_1^1,x_2^1,x_1^2,x_2^2)$ make $\dot{V}_2^2\le 0$, which implies the global stability of the closed-loop system.

The simulation results in Figure 2-1 show the state responses and parameter estimation of the closed-loop system (2.40) with the initial conditions $x_1(0)=x_2(0)=x_3(0)=x_4(0)=0.1$ and $\widehat{\Theta}(0)=0$. It demonstrates that the adaptive regulation for the system with nonlinear pa-

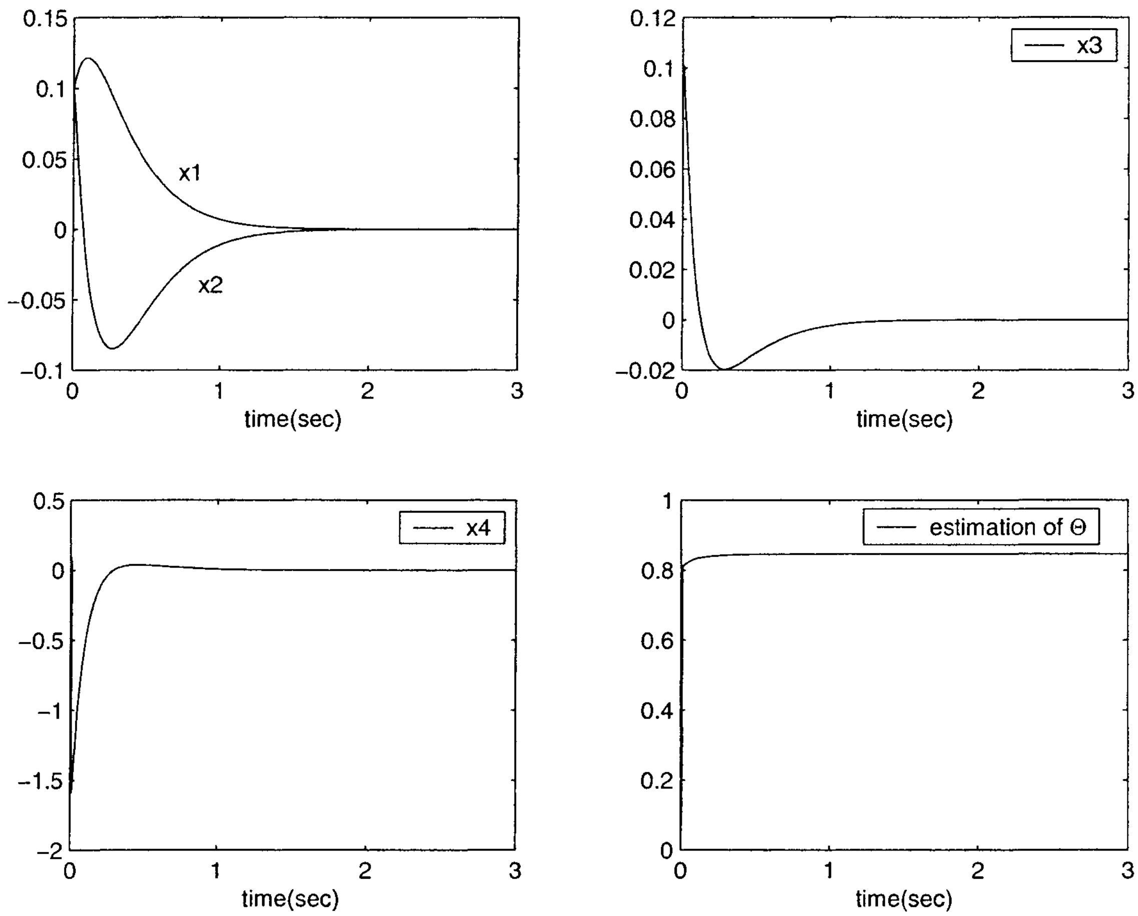

Figure 2-1: Responses of state variables

rameterization and nested triangular structure can be achieved via the methodology developed in this chapter.

2.5 Conclusion

In this chapter, the problem of adaptive control is studied for a class of MIMO nonlinearly parameterized systems with nested triangular structure. The methodology is developed based on adaptive backstepping technique. Our method benefits from both [18] and [23]. The results in [18] is extended to a class of MIMO systems with nested triangular form and adaptive controllers are constructed. It is shown that the resulting adaptive controller guarantees the global asymptotic stability of the closed-loop system.

Chapter 3

Adaptive Control of Nonlinear Differential-Algebraic Equation Systems

3.1 Introduction

Consider a MIMO nonlinear DAE system

$$\dot{x} = f_1(x) + p_1(x)z + g_1(x)u + \alpha_1(x)\theta \qquad (3.1)$$

$$0 = f_2(x) + p_2(x)z + g_2(x)u + \alpha_2(x)\theta \qquad (3.2)$$

$$y = h(x) \qquad (3.3)$$

where $x \in R^n$ is the vector of differential variables, $z \in R^s$ is the vector of algebraic variables, $u \in R^m$ is the vector of inputs, $y \in R^m$ is the vector of outputs, $\theta \in R^t$ is the vector of unknown parameters, $f_1(x)$, $f_2(x)$, $p_1(x)$, $p_2(x)$, $g_1(x)$, $g_2(x)$, $\alpha_1(x)$, $\alpha_2(x)$, $h(x)$ are matrix-valued smooth functions with dimensions of $n \times 1$, $s \times 1$, $n \times s$, $s \times s$, $n \times m$, $s \times m$, $n \times t$, $s \times t$, $m \times 1$, respectively. Assume that the origin is an isolated equilibrium point, i.e. $f_1(0) = 0$, $f_2(0) = 0$ and $h(0) = 0$.

In this chapter, one methodology will be developed to design a stabilizing feedback controller for the multi-input multi-output (MIMO) DAE systems with unknown parameters. Our aim is to find a change of coordinates to transform the DAE systems into an equivalent ordinary differential equation (ODE) systems with lower triangular structure. As a result, adaptive backstepping is applied to design an adaptive controller for the resulting ODE system.

One algorithm, called *Standardization*, is developed to construct the change of coordinates. The application of *Standardization* is under the full row rank condition that the block matrix $\begin{bmatrix} p_2(x) & g_2(x) \end{bmatrix}$ in (3.2) has full row rank. The other two algorithms are developed to guarantee the full row rank condition. The first algorithm is to calculate the generalized characteristic numbers. The second one is proposed to identify the constraints hidden behind the algebraic equations. These two algorithms can be considered as extensions of the first and second algorithm in [26] to adaptive control problem. In comparison with the algorithms in [26], one more term will be considered at each step of our algorithms in this chapter, which is caused by the term $\alpha_2(x)$.

The key step of the change of coordinates also involves the design of a static feedback $u = \gamma(x)z + v$. Different from the dynamic state compensator proposed in [15], with the new input v, our static feedback control scheme is much simpler to handle and guarantees not only bounded-input bounded-output (BIBO) stability, but asymptotic stability. The adaptive controller guarantees the global asymptotic stability of the closed-loop systems if the change of coordinates is defined globally. At last, following the approach proposed in this chapter, one adaptive controller is designed for a constrained manipulator with flexible joints in Section 4. The simulation results show the effectiveness of this approach.

3.2 Problem Formulation and Main Results

Considering the DAE system (3.1)-(3.3), the adaptive control problem is to find a static feedback $u = \gamma(x)z + \alpha(x, \widehat{\theta})$ and adaptive law $\dot{\widehat{\theta}} = \widehat{\theta}(x, \widehat{\theta})$, with $\gamma(x)$ and $\alpha(x, \widehat{\theta})$ smooth functions defined in a neighborhood U of the origin and $\alpha(0, \widehat{\theta}) = 0$, such that the corresponding closed-loop system

$$\dot{x} = f_1(x) + g_1(x)\alpha(x,\widehat{\theta}) + [p_1(x) + g_1(x)\gamma(x)]z + \alpha_1(x)\theta \quad (3.4)$$

$$0 = f_2(x) + g_2(x)\alpha(x,\widehat{\theta}) + [p_2(x) + g_2(x)\gamma(x)]z + \alpha_2(x)\theta \quad (3.5)$$

$$y = h(x) \quad (3.6)$$

has the following properties:

1. for any consistent initial condition $x_0 \in U$, it has a unique differentiable solution $(x(t), z(t))$ with $x(0) = x_0$;

2. $\lim_{t\to\infty} x(t) = 0$ for any consistent initial condition $x_0 \in U$.

In the following, three algorithms are given to transform the system (3.1)-(3.3) to a lower triangular form so that the backstepping technique can be applied. Algorithm 3.4 is applied to transform the DAE system into an equivalent ODE system. Algorithm 3.1 and 3.3 are used iteratively in Algorithm 3.4.

The first algorithm is to calculate the generalized characteristic number defined in [26], which is an extension of Algorithm 3.3 proposed in [24] and Algorithm 3.1 in [26]. The algorithm is developed under the assumption that the matrix $\begin{bmatrix} p_2(x) & g_2(x) \end{bmatrix}$ has full row rank.

Algorithm 3.1: Calculation of the Generalized Characteristic Number

Step 1. *Assign* $\phi_0(x) := \phi(x)$ *and set* $k = 0$. *Calculate* $L_{f_1}\phi_0(x)$, $L_{p_1}\phi_0(x)$, $L_{g_1}\phi_0(x)$ *and* $L_{\alpha_1}\phi_0(x)$.*, where*

$$\begin{aligned}
L_{f_1}\phi_0(x) &= \frac{\partial\phi_0(x)}{\partial x}\cdot f(x) \\
L_{p_1}\phi_0(x) &= \frac{\partial\phi_0(x)}{\partial x}\cdot p_1(x) = \begin{bmatrix} L_{p_1^1}\phi_0(x) & \dots & L_{p_1^j}\phi_0(x) & \dots & L_{p_1^s}\phi_0(x) \end{bmatrix} \\
L_{g_1}\phi_0(x) &= \frac{\partial\phi_0(x)}{\partial x}\cdot g_1(x) = \begin{bmatrix} L_{g_1^1}\phi_0(x) & \dots & L_{g_1^j}\phi_0(x) & \dots & L_{g_1^m}\phi_0(x) \end{bmatrix} \\
L_{\alpha_1}\phi_0(x) &= \frac{\partial\phi_0(x)}{\partial x}\cdot \alpha_1(x) = \begin{bmatrix} L_{\alpha_1^1}\phi_0(x) & \dots & L_{\alpha_1^j}\phi_0(x) & \dots & L_{\alpha_1^t}\phi_0(x) \end{bmatrix}
\end{aligned}$$

with p_1^j, g_1^j *and* α_1^j *are the j-th column of* p_1, g_1 *and* α_1*, respectively.*

If the matrix $\begin{bmatrix} p_2(x) & g_2(x) \\ L_{p_1}\phi_0(x) & L_{g_1}\phi_0(x) \end{bmatrix}$ *has a constant rank s, then there exists a unique vector-valued smooth function* $e_0(x)$ *of dimension s such that*

$$\begin{bmatrix} L_{p_1}\phi_0(x) & L_{g_1}\phi_0(x) \end{bmatrix} = e_0(x)\begin{bmatrix} p_2(x) & g_2(x) \end{bmatrix}$$

Define that $\phi_1(x) = L_{f_1}\phi_0(x) - e_0(x)f_2(x)$ *and* $w_1(x) = L_{\alpha_1}\phi_0(x) - e_0(x)\alpha_2(x)$. *Otherwise, set* $r = 1$ *and terminate the algorithm.*

Step $k+1$. *Suppose we have already defined a sequence of* $\phi_0(x)$, $\phi_1(x)\cdots\phi_k(x)$. *Now calculate* $L_{f_1}\phi_k(x)$, $L_{p_1}\phi_k(x)$, $L_{g_1}\phi_k(x)$ *and* $L_{\alpha_1}\phi_k(x)$. *If the matrix* $\begin{bmatrix} p_2(x) & g_2(x) \\ L_{p_1}\phi_k(x) & L_{g_1}\phi_k(x) \end{bmatrix}$ *has a constant rank s, then there exists a unique vector-valued smooth function* $e_k(x)$ *of dimension s such that*

$$\begin{bmatrix} L_{p_1}\phi_k(x) & L_{g_1}\phi_k(x) \end{bmatrix} = e_k(x)\begin{bmatrix} p_2(x) & g_2(x) \end{bmatrix}$$

Define that $\phi_{k+1}(x) = L_{f_1}\phi_k(x) - e_k(x)f_2(x)$ *and* $w_{k+1}(x) = L_{\alpha_1}\phi_k(x) - e_k(x)\alpha_2(x)$. *Otherwise, set* $r = k+1$ *and terminate the algorithm.*

The algorithm terminates at Step r. Such an integer is defined to be the generalized characteristic number of the function $\phi(x)$ under the constraint (3.2). Differentiating $\phi_k(x)$ with respect to time, it follows that for $k = 0, 1, \cdots, r-2$

$$\begin{aligned} \frac{d\phi_k(x)}{dt} &= L_{f_1}\phi_k(x) + L_{p_1}\phi_k(x)z + L_{g_1}\phi_k(x)u + L_{\alpha_1}\phi_k(x)\theta \\ &= \phi_{k+1}(x) + w_{k+1}(x)\theta + e_k(x)[f_2(x) + p_2(x)z + g_2(x)u + \alpha_2(x)\theta] \end{aligned} \tag{3.7}$$

and

$$\frac{d\phi_{r-1}(x)}{dt} = L_{f_1}\phi_{r-1}(x) + L_{p_1}\phi_{r-1}(x)z + L_{g_1}\phi_{r-1}(x)u + L_{\alpha_1}\phi_{r-1}(x)\theta \tag{3.8}$$

Remark 3.2 *Algorithm 3.1 will be used in Algorithms 3.3 and 3.4. Different from the algorithms in [24] and [26], besides* ϕ_k, *Algorithm 3.1 is also involved in calculating* w_k *due to the* α_2 *term, which the unknown parameter* θ *enters.*

Algorithm 3.3 is used to identify all the hidden constraints behind the algebraic equation (3.2). It begins with decomposing the algebraic equation (3.2) into the form of (3.9) and (3.10). Each step of Algorithm 3.3 involves calculating the generalized characteristic number r^i of the function $\phi^i(x)$ under the constraint produced in the previous step. The algorithm is developed under the assumption that the hidden constraints on x is independent of θ.

Algorithm 3.3: Regularization

Step 0. *Consider the constraint (3.2) and suppose the matrix* $\begin{bmatrix} p_2(x) & g_2(x) \end{bmatrix}$ *has a constant rank* s_0. *Without loss of generality, assume that its first* s_0 *rows, denoted by* $\begin{bmatrix} b_0(x) & c_0(x) \end{bmatrix}$, *has full row rank* s_0. *Let* $p = s - s_0$, *then for each* $i = 1, 2, \cdots, p$, *there exists a unique vector* $S^i(x)$ *such that*

$$\begin{bmatrix} p_2^{s_0+i}(x) & g_2^{s_0+i}(x) \end{bmatrix} = S^i(x) \begin{bmatrix} b_0(x) & c_0(x) \end{bmatrix}$$

where $p_2^{s_0+i}(x)$ *and* $g_2^{s_0+i}(x)$ *are the* (s_0+i)*th row of* $p_2(x)$ *and* $g_2(x)$ *respectively.*

Set $\phi^i(x) = f_2^{s_0+i}(x) - S^i(x)a_0(x)$ *and* $w^i(x) = \alpha_2^{s_0+i}(x) - S^i(x)d_0(x)$ *with* $a_0(x)$, $d_0(x)$ *being the first* s_0 *rows of* $f_2(x)$, $\alpha_2(x)$ *respectively and* $f_2^{s_0+i}(x)$, $\alpha_2^{s_0+i}(x)$ *being the first* (s_0+i)*th rows of* $f_2(x)$, $\alpha_2(x)$ *respectively. Then the algebraic equation (3.2) becomes*

$$\begin{aligned} 0 &= a_0(x) + b_0(x)z + c_0(x)u + d_0(x)\theta && (3.9) \\ 0 &= \phi^i(x) + w^i(x)\theta + S^i(x)[a_0(x) + b_0(x)z + c_0(x)u + d_0(x)\theta] && (3.10) \end{aligned}$$

for $i = 1, 2, \cdots, p$.

Substituting (3.9) into (3.10) leads to

$$0 = \phi^i(x) + w^i(x)\theta \tag{3.11}$$

Considering that the hidden constraints on x *is independent of* θ, *we have* $w^i(x) \equiv 0$ *for* $i = 1, 2, \cdots, p$. *So the hidden constraint (3.11) becomes*

$$0 = \phi^i(x) \tag{3.12}$$

Step 1. *Assign* $\phi(x) := \phi^1(x)$ *and carry out Algorithm 3.1 to calculate its generalized characteristic number under the constraint (3.9). Then* r^1, $\phi_0^1(x)$, $\phi_1^1(x)$, $\cdots$, $\phi_{r^1-1}^1(x)$, $w_1^1(x)$, $\cdots$, $w_{r^1-1}^1(x)$ *and* $e_0^1(x)$, $e_1^1(x)$, $\cdots$, $e_{r^1-2}^1(x)$ *are produced. Now define* $a_1(x) = L_{f_1}\phi_{r^1-1}^1(x)$, $b_1(x) = L_{p_1}\phi_{r^1-1}^1(x)$, $c_1(x) = L_{g_1}\phi_{r^1-1}^1(x)$ *and* $d_1(x) = L_{\alpha_1}\phi_{r^1-1}^1(x)$.

Differentiating $\phi_i^1(x)$ *with respect to time leads to for* $i = 0, 1, \cdots, r^1 - 2$

$$\frac{d\phi_i^1(x)}{dt} = \phi_{i+1}^1(x) + w_{i+1}^1(x)\theta + e_i^1(x)[a_0(x) + b_0(x)z + c_0(x)u + d_0(x)\theta] \tag{3.13}$$

and

$$\frac{d\phi_{r^1-1}^1(x)}{dt} = a_1(x) + b_1(x)z + c_1(x)u + d_1(x)\theta \tag{3.14}$$

It follows from (3.12) that $\frac{d\phi_0^1(x)}{dt} = 0$, *from which, together with (3.13), we obtain the hidden constraint* $0 = \phi_1^1(x) + w_1^1(x)\theta$.

Considering that the hidden constraints on x *is independent of* θ, *we have* $w_1^1(x) = 0$ *and the new hidden constraint is* $0 = \phi_1^1(x)$. *By the same token, it can be derived that* $w_i^1(x) \equiv 0$ *and*

$$\phi_i^1(x) = 0 \tag{3.15}$$

for $i = 1, 2, \cdots, r^1 - 1$. *It follows from (3.15) with* $i = r^1 - 1$ *and (3.14) that*

$$0 = a_1(x) + b_1(x)z + c_1(x)u + d_1(x)\theta \tag{3.16}$$

Combining (3.9) and (3.16) yields the following algebraic equation

$$0 = a^1(x) + b^1(x)z + c^1(x)u + d^1(x)\theta \tag{3.17}$$

where $a^1(x) = \begin{bmatrix} a_0(x) \\ a_1(x) \end{bmatrix}$, $b^1(x) = \begin{bmatrix} b_0(x) \\ b_1(x) \end{bmatrix}$, $c^1(x) = \begin{bmatrix} c_0(x) \\ c_1(x) \end{bmatrix}$, $d^1(x) = \begin{bmatrix} d_0(x) \\ d_1(x) \end{bmatrix}$.

If the matrix $\begin{bmatrix} b^1(x) & c^1(c) \end{bmatrix}$ *has full row rank* $s_0 + 1$, *then set* $k = 2$ *and go to next step. Otherwise, terminate the algorithm.*

Step k. *Suppose the algebraic equation produced at Step* $\mathsf{k-1}$ *is given by*

$$0 = a^{k-1}(x) + b^{k-1}(x)z + c^{k-1}(x)u + d^{k-1}(x)\theta \tag{3.18}$$

Assign $\phi(x) := \phi^k(x)$ *and carry out Algorithm 3.1 to calculate its generalized characteristic number under the constraint (3.18). Then* r^k, $\phi_0^k(x)$, $\phi_1^k(x)$, $\cdots$, $\phi_{r^k-1}^k(x)$, $w_1^k(x)$, $\cdots$, $w_{r^k-1}^k(x)$ *and* $e_0^k(x)$, $e_1^k(x)$, $\cdots$, $e_{r^k-2}^k(x)$ *are produced. Now define* $a_k(x) = L_{f_1}\phi_{r^k-1}^k(x)$, $b_k(x) = L_{p_1}\phi_{r^k-1}^k(x)$, $c_k(x) = L_{g_1}\phi_{r^k-1}^k(x)$ *and* $d_k(x) = L_{\alpha_1}\phi_{r^k-1}^k(x)$.

Differentiating $\phi_i^k(x)$ *with respect to time produces*

$$\frac{d\phi_i^k(x)}{dt} = \phi_{i+1}^k(x) + w_{i+1}^k(x)\theta + e_i^k(x)[a^{k-1}(x) + b^{k-1}(x)z + c^{k-1}(x)u + d^{k-1}(x)\theta] \tag{3.19}$$

for $i = 0,\ 1,\ \cdots, r^k - 2$ *and*

$$\frac{d\phi_{r^k-1}^k(x)}{dt} = a_k(x) + b_k(x)z + c_k(x)u + d_k(x)\theta \tag{3.20}$$

It follows from (3.12) that $\frac{d\phi_0^k(x)}{dt} = 0$, *from which, together with (3.19), we obtain the hidden constraint* $0 = \phi_1^k(x) + w_1^k(x)\theta$. *Considering that the hidden constraints on* x *is independent of* θ, *we have* $w_1^k(x) = 0$ *and the new hidden constraint is* $0 = \phi_1^k(x)$. *By the same token, it can be derived that* $w_i^k(x) \equiv 0$ *and*

$$\phi_i^k(x) = 0 \tag{3.21}$$

for $i = 1,\ 2,\ \cdots,\ r^k - 1$. *It follows from (3.21) with* $i = r^k - 1$ *and (3.20) that*

$$0 = a_k(x) + b_k(x)z + c_k(x)u + d_k(x)\theta \tag{3.22}$$

Combining (3.18) and (3.22) yields the following algebraic equation

$$0 = a^k(x) + b^k(x)z + c^k(x)u + d^k(x)\theta \tag{3.23}$$

where $a^k(x) = \begin{bmatrix} a^{k-1}(x) \\ a_k(x) \end{bmatrix}$, $b^k(x) = \begin{bmatrix} b^{k-1}(x) \\ b_k(x) \end{bmatrix}$, $c^k(x) = \begin{bmatrix} c^{k-1}(x) \\ c_k(x) \end{bmatrix}$, $d^k(x) = \begin{bmatrix} d^{k-1}(x) \\ d_k(x) \end{bmatrix}$.

If the matrix $\begin{bmatrix} b^k(x) & c^k(c) \end{bmatrix}$ *has full row rank* $s_0 + k$, *then set* $k = k + 1$ *and go to next step. Otherwise, terminate the algorithm.*

Algorithm 3.3 is said to be feasible if it terminates at Step $k = p$ and the matrix $\begin{bmatrix} b^p(x) & c^p(c) \end{bmatrix}$ has full row rank $s = s_0 + p$. It follows from Algorithm 3.3 that in order for solutions to the DAE system (3.1)-(3.2) to be impulse-free, the initial condition $x(0)$ must satisfy $x(0) \in M$ with

$$M = \{x \in R^n \mid \phi_0^i(x) = 0,\ \phi_j^i(x) = 0,\ \text{and } w_j^i = 0, \text{ for } j = 1, 2, \cdots, r^{i-1}, i = 1, 2, \cdots, p\}$$

If Algorithm 3.3 is feasible, the DAE system (3.1)-(3.2) is equivalent to the following DAE system

$$\dot{x} = f_1(x) + p_1(x)z + g_1(x)u + \alpha_1(x)\theta \tag{3.24}$$

$$0 = a(x) + b(x)z + c(x)u + d(x)\theta \tag{3.25}$$

where $x \in M$ and $\begin{bmatrix} b(x) & c(x) \end{bmatrix}$ has full row rank s.

With the assumption that Algorithm 3.3 is feasible, the DAE system (3.24)-(3.25) can be changed to lower triangular form by a feedback $u = \gamma(x)z + v$ and the following algorithm.

Algorithm 3.4: Standardization

Step 1. *Set* $\psi^1(x) := h^1(x)$ *and calculate the generalized characteristic number* q^1 *of* $\psi^1(x)$ *under the constraint (3.25). Then* q^1, $\psi_0^1(x)$, $\psi_1^1(x)$, $\cdots$, $\psi_{q^1-1}^1(x)$, $\varphi_1^1(x)$, $\cdots$, $\varphi_{q^1-1}^1(x)$ *and* $E_0^1(x)$, $E_1^1(x)$, $\cdots$, $E_{q^1-2}^1(x)$ *are produced. Now define* $A_1(x) = L_{f_1}\psi_{q^1-1}^1(x)$, $B_1(x) = L_{p_1}\psi_{q^1-1}^1(x)$, $C_1(x) = L_{g_1}\psi_{q^1-1}^1(x)$ *and* $D_1(x) = L_{\alpha_1}\phi_{q^1-1}^1(x)$.

Differentiating $\psi_i^1(x)$ *with respect to time yields*

$$\frac{d\psi_i^1(x)}{dt} = \psi_{i+1}^1(x) + \varphi_{i+1}^1(x)\theta + E_i^1(x)[a(x) + b(x)z + c(x)u + d(x)\theta] \tag{3.26}$$

for $i = 0,\ 1,\ \cdots, q^1 - 2$ *and*

$$\frac{d\psi^1_{q^1-1}(x)}{dt} = A_1(x) + B_1(x)z + C_1(x)u + D_1(x)\theta \tag{3.27}$$

Step k. *Set* $\psi^k(x) := h^k(x)$ *and calculate the generalized characteristic number* q^k *of* $\psi^k(x)$ *under the constraint (3.25). Then* q^k, $\psi^k_0(x)$, $\psi^k_1(x)$, $\cdots$, $\psi^k_{q^k-1}(x)$, $\varphi^k_1(x)$, $\cdots$, $\varphi^k_{q^k-1}(x)$ *and* $E^k_0(x)$, $E^k_1(x)$, $\cdots$, $E^k_{q^k-2}(x)$ *are produced. Now define* $A_k(x) = L_{f_1}\psi^1_{q^k-1}(x)$, $B_k(x) = L_{p_1}\psi^1_{q^k-1}(x)$, $C_1(x) = L_{g_1}\psi^1_{q^k-1}(x)$ *and* $D_1(x) = L_{\alpha_1}\phi^1_{q^k-1}(x)$.

Differentiating $\psi^k_i(x)$ *with respect to time leads to for* $i = 0,\ 1,\ \cdots, q^k - 2$

$$\frac{d\psi^k_i(x)}{dt} = \psi^k_{i+1}(x) + \varphi^k_{i+1}(x)\theta + E^k_i(x)[a(x) + b(x)z + c(x)u + d(x)\theta] \tag{3.28}$$

and

$$\frac{d\psi^k_{q^k-1}(x)}{dt} = A_k(x) + B_k(x)z + C_k(x)u + D_k(x)\theta \tag{3.29}$$

Algorithm 3.4 terminates at Step $k = m$. Now the following assumption is made.

Assumption 3.5 The matrix $\begin{bmatrix} b(x) & c(x) \\ B_1(x) & C_1(x) \\ \vdots & \vdots \\ B_m(x) & C_m(x) \end{bmatrix}$ is nonsingular in U.

The functions $\phi^i_j(x)$ for $j = 0,\ 1,\ \cdots, r^i - 1$ and $i = 0,\ 1,\ \cdots, p$, and $\psi^i_j(x)$ for $j = 0,\ 1,\ \cdots, q^i - 1$ and $i = 0,\ 1,\ \cdots, m$, form a set of new coordinates, which is guaranteed by Lemma 1, for proof, see [26].

Lemma 3.6 *Suppose that Algorithms 3.3 and 3.4 are feasible and Assumption 3.5 is satisfied. Then, the vectors*

$$\begin{aligned}
&d\phi^1_0(x), d\phi^1_1(x), \cdots, d\phi^1_{r^1-1}(x) \\
&d\phi^2_0(x), d\phi^2_1(x), \cdots, d\phi^2_{r^2-1}(x) \\
&\vdots \\
&d\phi^p_0(x), d\phi^p_1(x), \cdots, d\phi^p_{r^p-1}(x)
\end{aligned}$$

$$d\psi_0^1(x), d\psi_1^1(x), \cdots, d\psi_{q^1-1}^1(x)$$
$$d\psi_0^2(x), d\psi_1^2(x), \cdots, d\psi_{q^2-1}^2(x)$$
$$\vdots$$
$$d\psi_0^m(x), d\psi_1^m(x), \cdots, d\psi_{q^m-1}^m(x)$$

are linearly independent in U.

Please see [26] for the proof of Lemma 3.6.

Assumption 3.7 $n = r + q$ with $r = r^0 + r^1 + \cdots + r^p$ and $q = q^0 + q^1 + \cdots + q^m$.

Assumption 3.7 is introduced to avoid the appearance of zero dynamics. With Assumption 3.7, it follows from Lemma 3.6 that the function $\Phi(x) = \begin{bmatrix} \phi(x) \\ \psi(x) \end{bmatrix}$ constitutes a change of coordinates, where

$$\phi(x) = \begin{bmatrix} \phi^1(x)^T & \phi^2(x)^T & \cdots & \phi^p(x)^T \end{bmatrix}^T \quad \psi(x) = \begin{bmatrix} \psi^1(x)^T & \psi^2(x)^T & \cdots & \psi^m(x)^T \end{bmatrix}^T$$

with $\phi^i(x) = \begin{bmatrix} \phi_0^i(x) & \cdots & \phi_{r^i-1}^i(x) \end{bmatrix}^T$ for $i = 1, \cdots, p$ and $\psi^j(x) = \begin{bmatrix} \psi_0^j(x) & \cdots & \psi_{q^j-1}^j(x) \end{bmatrix}^T$ for $j = 1, \cdots, m$.

Set $\varepsilon_i^k = \phi_i^k(x)$ for $i = 0, 1, \cdots, r^k - 1$, $k = 1, \cdots, p$ and $\xi_i^k = \psi_i^k(x)$ for $i = 0, 1, \cdots, q^k - 1$, $k = 1, \cdots, m$. Let $\varepsilon = \begin{bmatrix} \varepsilon_0^1 & \cdots & \varepsilon_{r^1-1}^1 & \cdots & \varepsilon_0^p & \cdots & \varepsilon_{r^p-1}^p \end{bmatrix}^T$ and $\xi = \begin{bmatrix} \xi_0^1 & \cdots & \xi_{q^1-1}^1 & \cdots & \xi_0^m & \cdots & \varepsilon_{q^m-1}^m \end{bmatrix}^T$.

By differentiating ε_i^k and ξ_i^k with respect to time, in the new coordinates the DAE system (3.1)-(3.3) can be expressed as follows

$$\varepsilon = 0 \tag{3.30}$$

$$
\begin{aligned}
y^1 &= \xi_0^1 \\
\dot{\xi}_0^1 &= \xi_1^1 + \varphi_1^1(x)\theta \\
\dot{\xi}_1^1 &= \xi_2^1 + \varphi_2^1(x)\theta \\
&\vdots \\
\dot{\xi}_{q^1-2}^1 &= \xi_{q^1-1}^1 + \varphi_{q^1-1}^1(x)\theta \\
\dot{\xi}_{q^1-1}^1 &= A_1(x) + B_1(x)z + C_1(x)u + D_1(x)\theta
\end{aligned}
\tag{3.31}
$$

$$\vdots$$

$$
\begin{aligned}
y^m &= \xi_0^m \\
\dot{\xi}_0^m &= \xi_1^m + \varphi_1^m(x)\theta \\
\dot{\xi}_1^m &= \xi_2^m + \varphi_2^m(x)\theta \\
&\vdots \\
\dot{\xi}_{q^m-2}^m &= \xi_{q^m-1}^m + \varphi_{q^m-1}^m(x)\theta \\
\dot{\xi}_{q^m-1}^m &= A_m(x) + B_m(x)z + C_m(x)u + D_m(x)\theta
\end{aligned}
\tag{3.32}
$$

In order to apply the adaptive backstepping technique, the following assumption is needed to put the system (3.30)-(3.32) into lower triangular form.

Assumption 3.8 *The matrix*

$$
\begin{bmatrix} \frac{\partial \phi(x)}{\partial x} & \frac{\partial \psi_0^1(x)}{\partial x} & \cdots & \frac{\partial \psi_{q^1-1}^1(x)}{\partial x} & \cdots & \frac{\partial \psi_0^k(x)}{\partial x} & \cdots & \frac{\partial \psi_i^k(x)}{\partial x} & \frac{\partial \varphi_i^k(x)}{\partial x} \end{bmatrix}^T \tag{3.33}
$$

has constant row rank $r + \sum_{j=1}^{k-1} q^j + i + 1$ for $i = 0,\ 1,\ \cdots,\ q^k - 1$ and $k = 1,\ \cdots,\ m$.

Lemma 3.9 *Suppose that Algorithms 3.3 and 3.4 are feasible and Assumptions 3.5, 3.7 and 3.8 are satisfied. Then, in* ξ *coordinates, the system (3.30)-(3.32) takes the form of*

$$
\varepsilon = 0 \tag{3.34}
$$

$$
\begin{aligned}
y^1 &= \xi_0^1 \\
\dot{\xi}_0^1 &= \xi_1^1 + \varphi_1^1(\varepsilon, \xi_0^1)\theta \\
\dot{\xi}_1^1 &= \xi_2^1 + \varphi_2^1(\varepsilon, \xi_0^1, \xi_1^1)\theta \\
&\vdots \\
\dot{\xi}_{q^1-2}^1 &= \xi_{q^1-1}^1 + \varphi_{q^1-1}^1(\varepsilon, \xi_0^1, \xi_1^1, \cdots, \xi_{q^1-2}^1)\theta \\
\dot{\xi}_{q^1-1}^1 &= A_1(x) + B_1(x)z + C_1(x)u + D_1(x)\theta
\end{aligned}
\tag{3.35}
$$

$$
\begin{aligned}
&\vdots\\
&y^m = \xi_0^m\\
&\dot{\xi}_0^m = \xi_1^m + \varphi_1^m(\varepsilon, \xi^1, \cdots, \xi^{m-1}, \xi_0^m)\theta\\
&\dot{\xi}_1^m = \xi_2^m + \varphi_2^m(\varepsilon, \xi^1, \cdots, \xi^{m-1}, \xi_0^m, \xi_1^m)\theta\\
&\vdots\\
&\dot{\xi}_{q^m-2}^m = \xi_{q^m-1}^m + \varphi_{q^m-1}^m(\varepsilon, \xi^1, \cdots, \xi^{m-1}, \xi_0^m, \cdots, \xi_{q^m-2}^m)\theta\\
&\dot{\xi}_{q^m-1}^m = A_m(x) + B_m(x)z + C_m(x)u + D_m(x)\theta
\end{aligned}
\tag{3.36}
$$

Proof: By carrying out Algorithms 3.3 and 3.4, the system (3.1)-(3.3) is changed to the equivalent system (3.30)-(3.32). The matrix (3.33) takes the form of

$$
\begin{bmatrix}
\frac{\partial\phi(x)}{\partial\varepsilon} & \frac{\partial\psi_0^1(x)}{\partial\varepsilon} & \cdots & \frac{\partial\psi_{q^1-1}^1(x)}{\partial\varepsilon} & \cdots & \frac{\partial\psi_0^k(x)}{\partial\varepsilon} & \cdots & \frac{\partial\psi_i^k(x)}{\partial\varepsilon} & \frac{\partial\varphi_i^k(x)}{\partial\varepsilon}\\
\frac{\partial\phi(x)}{\partial\xi} & \frac{\partial\psi_0^1(x)}{\partial\xi} & \cdots & \frac{\partial\psi_{q^1-1}^1(x)}{\partial\xi} & \cdots & \frac{\partial\psi_0^k(x)}{\partial\xi} & \cdots & \frac{\partial\psi_i^k(x)}{\partial\xi} & \frac{\partial\varphi_i^k(x)}{\partial\xi}
\end{bmatrix}^T
\begin{bmatrix}
\frac{\partial\varepsilon(x)}{\partial x}\\
\frac{\partial\xi(x)}{\partial x}
\end{bmatrix}
\tag{3.37}
$$

for $i = 0, 1, \cdots, q^k - 1$ and $k = 1, \cdots, m$. Since the matrix $\begin{bmatrix} \frac{\partial\varepsilon}{\partial x} & \frac{\partial\xi}{\partial x} \end{bmatrix}^T$ is nonsingular, therefore the rank of the matrix

$$
\begin{bmatrix}
\frac{\partial\phi(x)}{\partial\varepsilon} & \frac{\partial\psi_0^1(x)}{\partial\varepsilon} & \cdots & \frac{\partial\psi_{q^1-1}^1(x)}{\partial\varepsilon} & \cdots & \frac{\partial\psi_0^k(x)}{\partial\varepsilon} & \cdots & \frac{\partial\psi_i^k(x)}{\partial\varepsilon} & \frac{\partial\varphi_i^k(x)}{\partial\varepsilon}\\
\frac{\partial\phi(x)}{\partial\xi} & \frac{\partial\psi_0^1(x)}{\partial\xi} & \cdots & \frac{\partial\psi_{q^1-1}^1(x)}{\partial\xi} & \cdots & \frac{\partial\psi_0^k(x)}{\partial\xi} & \cdots & \frac{\partial\psi_i^k(x)}{\partial\xi} & \frac{\partial\varphi_i^k(x)}{\partial\xi}
\end{bmatrix}^T
\tag{3.38}
$$

is the same as that of the matrix (3.33) for $i = 0, 1, \cdots, q^k - 1$ and $k = 1, \cdots, m$. In the matrix (3.38), $\frac{\partial\phi(x)}{\partial\varepsilon} = I$, $\frac{\partial\phi(x)}{\partial\xi} = 0$,

$$
\frac{\partial\psi_i^k(x)}{\partial\xi_j^l} = \begin{cases} 1, \ k = l \text{ and } i = j\\ 0, \ k \neq l \text{ or } i \neq j \end{cases}
$$

Since the rank of the matrix (3.33) or (3.38) is $r + \sum_{j=1}^{k-1} q^j + i + 1$, $\frac{\partial\varphi_i^k(x)}{\partial\xi_j^l} = 0$ for $l > k$ and $i > j$ if $l = k$, which implies that φ_i^k is the function of ε, $\xi^1, \cdots, \xi^{k-1}, \xi_0^k, \cdots, \xi_i^k$. Therefore, the system (3.30)-(3.32) can be expressed by (3.34)-(3.36).

Theorem 3.10 *Consider the system (3.1)-(3.3). Suppose that Algorithms 3.3 and 3.4 are feasible and Assumptions 3.5, 3.7 and 3.8 are satisfied. There exist a feedback controller*

$u = \gamma(x)z + \alpha(x, \widehat{\theta})$ *and an estimator* $\dot{\widehat{\theta}} = \widehat{\theta}(x, \widehat{\theta})$ *such that the corresponding closed-loop systems are asymptotically stable in a neighborhood* U *of the origin.*

A constructive proof of this theorem is given in Section 3.3.

3.3 Design of Adaptive Controllers

From Algorithm 3.3, we know that the matrix $\begin{bmatrix} b(x) & c(x) \end{bmatrix}$ has full row rank, therefore there exists a smooth matrix-valued function $\gamma(x)$ such that $b(x) + c(x)\gamma(x)$ is nonsingular. By introducing a feedback $u = \gamma(x)z + v$, the algebraic equation (3.25) admits

$$a(x) + [b(x) + c(x)\gamma(x)]z + c(x)v + d(x)\theta = 0 \tag{3.39}$$

Solving (3.39) for z gives

$$z = -[b(x) + c(x)\gamma(x)]^{-1}[a(x) + c(x)v + d(x)\theta] \tag{3.40}$$

As a result, u can be expressed as

$$u = -\gamma(x)[b(x) + c(x)\gamma(x)]^{-1}[a(x) + c(x)v + d(x)\theta] + v \tag{3.41}$$

Substituting u and z (3.30)-(3.32) leads to

$$\varepsilon = 0 \tag{3.42}$$

$$\begin{aligned}
&y^1 = \xi_0^1 \\
&\dot{\xi}_0^1 = \xi_1^1 + \varphi_1^1(0, \xi_0^1)\theta \\
&\dot{\xi}_1^1 = \xi_2^1 + \varphi_2^1(0, \xi_0^1, \xi_1^1)\theta \\
&\vdots \\
&\dot{\xi}_{q^1-2}^1 = \xi_{q^1-1}^1 + \varphi_{q^1-1}^1(0, \xi_0^1, \xi_1^1, \cdots, \xi_{q^1-2}^1)\theta \\
&\dot{\xi}_{q^1-1}^1 = \hat{V}_1 + \varphi_{q^1}^1(0, \xi^1, \cdots, \xi^{m-1}, \xi^m)\theta
\end{aligned} \tag{3.43}$$

$\vdots$

$$
\begin{aligned}
y^m &= \xi_0^m \\
\dot{\xi}_0^m &= \xi_1^m + \varphi_1^m(0,\xi^1,\cdots,\xi^{m-1},\xi_0^m)\theta \\
\dot{\xi}_1^m &= \xi_2^m + \varphi_2^m(0,\xi^1,\cdots,\xi^{m-1},\xi_0^m,\xi_1^m)\theta \\
&\vdots \\
\dot{\xi}_{q^m-2}^m &= \xi_{q^m-1}^m + \varphi_{q^m-1}^m(0,\xi^1,\cdots,\xi^{m-1},\xi_0^m,\cdots,\xi_{q^m-2}^m)\theta \\
\dot{\xi}_{q^m-1}^m &= \hat{V}_m + \varphi_{q^m}^m(0,\xi^1,\cdots,\xi^{m-1},\xi^m)\theta
\end{aligned}
\tag{3.44}
$$

with $\hat{V}_k = A_k-[B_k+C_k\gamma(x)][b(x)+c(x)\gamma(x)]^{-1}a(x)+\{C_k-[B_k+C_k\gamma(x)][b(x)+c(x)\gamma(x)]^{-1}c(x)\}v$ and $\varphi_{q^k}^k(x) = [B_k + C_k\gamma(x)][b(x) + c(x)\gamma(x)]^{-1}d(x) + D_k$, for $k = 1, \cdots, m$.

By applying adaptive backstepping technique to design an adaptive controller, the system (3.42)-(3.44) is guaranteed to be asymptotically stable for any unknown parameters. The brief design procedure is as follows.

Step 1.1: Consider the following Lyapunov function

$$V_1^1 = \frac{1}{2}(\xi_0^1)^2 + \frac{1}{2\Gamma}(\theta - \widehat{\theta})^T(\theta - \widehat{\theta})$$

Differentiating V_1 with respect to time leads to

$$\dot{V}_1^1 = \xi_0^1(\xi_1^1 + \varphi_1^1\theta) - \frac{1}{\Gamma}(\theta - \widehat{\theta})^T \dot{\widehat{\theta}} \tag{3.45}$$

Now define

$$
\begin{aligned}
\alpha_1^1(\xi_0^1) &= -c_0^1\xi_0^1 - \varphi_1^1\widehat{\theta} \\
\tau_1^1 &= \xi_0^1(\varpi_1^1)^T \\
\varpi_1^1 &= \varphi_1^1
\end{aligned}
$$

Then, (3.45) becomes

$$\dot{V}_1^1 = -c_0^1(\xi_0^1)^2 + \xi_0^1(\xi_1^1 - \alpha_1^1) + (\theta - \widehat{\theta})^T\left(\tau_1^1 - \frac{1}{\Gamma}\dot{\widehat{\theta}}\right) \tag{3.46}$$

Step $1.k+1$: Consider the following Lyapunov function

$$V_{k+1} = V_k + \frac{1}{2}(\xi_k^1 - \alpha_k^1)^2$$

Differentiating V_{i+1}^1 with respect to time yields

$$\begin{aligned} \dot{V}_{k+1}^1 &= -\sum_{j=0}^{k} c_j^1(\xi_j^1 - \alpha_j^1)^2 + (\xi_k^1 - \alpha_k^1)(\xi_{k+1}^1 - \alpha_{k+1}^1) \\ &\quad + \left[\frac{1}{\Gamma}(\theta - \widehat{\theta})^T + \lambda_{k+1}^1\right]\left(\Gamma\tau_{k+1}^1 - \dot{\widehat{\theta}}\right) \end{aligned} \tag{3.47}$$

where we define

$$\begin{aligned} \alpha_{k+1}^1(\xi_0^1, \xi_1^1, \cdots, \xi_i^1) &= -c_k^1(\xi_k^1 - \alpha_k^1) - (\xi_{k-1}^1 - \alpha_{k-1}^1) \\ &\quad -\upsilon_{k+1}^1 + \Gamma\frac{\partial\alpha_k^1}{\partial\widehat{\theta}}\tau_k^1 + \Gamma\lambda_{k+1}^1(\varpi_{k+1}^1)^T \\ \upsilon_{k+1}^1 &= \varphi_{k+1}^1\hat{\theta} - \sum_{j=0}^{k-1}\frac{\partial\alpha_k^1}{\partial\xi_j^1}(\xi_{j+1}^1 + \varphi_{j+1}^1\widehat{\theta}) \\ \tau_{k+1}^1 &= \tau_k^1 + (\xi_k^1 - \alpha_k^1)(\varpi_{k+1}^1)^T \\ \varpi_{k+1}^1 &= \varphi_{k+1}^1 - \sum_{j=0}^{k-1}\frac{\partial\alpha_k^1}{\partial\xi_j^1}\varphi_{j+1}^1 \\ \lambda_{k+1}^1 &= \lambda_k^1 + (\xi_k^1 - \alpha_k^1)\frac{\partial\alpha_k^1}{\partial\hat{\theta}} \end{aligned} \tag{3.48}$$

with $\alpha_0^1 = 0$.

Step $i.k+1$: Consider the following Lyapunov function

$$V_{k+1}^i = V_k^i + \frac{1}{2}(\xi_k^i - \alpha_k^i)^2 \tag{3.49}$$

Differentiating V_{k+1}^i with respect to time yields

$$\begin{aligned}\dot{V}^i_{k+1} &= -\sum_{l=1}^{i-1}\sum_{j=0}^{q^l-1} c^l_j(\xi^l_j-\alpha^l_j)^2+\sum_{l=1}^{i-1}(\hat{V}_l-\alpha^l_{q^l})(\xi^l_{q^l-1}-\alpha^l_{q^l-1})+\sum_{j=0}^{k} c^i_j(\xi^i_j-\alpha^i_j)^2 \\ &+(\xi^i_k-\alpha^i_k)(\xi^i_{k+1}-\alpha^i_{k+1})+[\frac{1}{\Gamma}(\theta-\widehat{\theta})^T+\lambda^i_{k+1}](\Gamma\tau^i_{k+1}-\dot{\widehat{\theta}})\end{aligned} \tag{3.50}$$

where we define

$$\begin{aligned}\alpha^m_{q^m}(\xi^1,\cdots,\xi^{i-1},\xi^i_1,\cdots,\xi^i_k) &= -c^i_k(\xi^i_k-\alpha^i_k)-(\xi^i_{k-1}-\alpha^i_{k-1}) \\ &\quad -\upsilon^i_{k+1}+\Gamma\frac{\partial\alpha^i_k}{\partial\hat{\theta}}\tau^i_k+\Gamma\lambda^i_{k+1}(\varpi^i_{k+1})^T \\ \upsilon^i_{k+1} &= \varphi^i_{k+1}\widehat{\theta}-\sum_{l=1}^{i-1}\sum_{j=0}^{q^l-1}\frac{\partial\alpha^i_k}{\partial\xi^l_j}(\xi^l_{j+1}+\varphi^l_{j+1}\widehat{\theta})-\sum_{j=0}^{k-1}\frac{\partial\alpha^i_k}{\partial\xi^i_j}(\xi^i_{j+1}+\varphi^i_{j+1}\widehat{\theta}) \\ \tau^i_{k+1} &= \tau^i_k+(\xi^i_k-\alpha^i_k)(\varpi^i_{k+1})^T \\ \varpi^i_{k+1} &= \varphi^i_{k+1}-\sum_{l=1}^{i-1}\sum_{j=0}^{q^l-1}\frac{\partial\alpha^i_k}{\partial\xi^l_j}\varphi^l_{j+1}-\sum_{j=0}^{k-1}\frac{\partial\alpha^i_k}{\partial\xi^i_j}\varphi^i_{j+1} \\ \lambda^i_{k+1} &= \lambda^i_k+(\xi^i_k-\alpha^i_k)\frac{\partial\alpha^i_k}{\partial\hat{\theta}}\end{aligned}$$

with $\alpha^l_0=0$ and $\xi^l_{q^l}=\hat{V}_l$ for $l=1,\cdots i$.

The iteration holds on at Step $m.q^m$. It is obvious that the feedback controller

$$\begin{cases}\hat{V}_1=\alpha_{q^1}(\xi^1,\cdots,\xi^m,\widehat{\theta}) \\ \hat{V}_2=\alpha^2_{q^2}(\xi^1,\xi^2,\cdots,\xi^m,\widehat{\theta}) \\ \vdots \\ \hat{V}_m=\alpha^m_{q^m}(\xi^1,\xi^2,\cdots,\xi^m,\widehat{\theta})\end{cases} \tag{3.51}$$

and the parameter estimator

$$\dot{\widehat{\theta}}=\Gamma\tau^m_{q^m}(\xi^1,\xi^2,\cdots,\xi^m) \tag{3.52}$$

make $\dot{V}^m_{q^m}$ negative definite, therefore guarantee the asymptotic stability of the closed-loop system in a neighborhood U of the origin.

With the controller (3.51) and the parameter estimator (3.52) in ξ coordinates, we will express them in terms of the original coordinates x. For convenience, let $\hat{A}(x) = \begin{bmatrix} \hat{A}_1^T & \cdots & \hat{A}_m^T \end{bmatrix}^T$ with $\hat{A}_k(x) = A_k - [B_k + C_k\gamma(x)][b(x) + c(x)\gamma(x)]^{-1}a(x)$, $\hat{C}(x) = \begin{bmatrix} \hat{C}_1^T & \cdots & \hat{C}_m^T \end{bmatrix}^T$ with $\hat{C}_k = C_k - [B_k + C_k\gamma(x)][b(x) + c(x)\gamma(x)]^{-1}c(x)$, $\hat{V} = \begin{bmatrix} \hat{V}_1^T & \cdots & \hat{V}_m^T \end{bmatrix}^T$ for $k = 1, \cdots, m$. Therefore, the controller u in the original coordinates is uniquely determined as

$$u = \gamma(x)z + \hat{C}^{-1}(x)[\hat{V} - \hat{A}(x)] \tag{3.53}$$

where the nonsingularity of $\hat{C}(x)$ is guaranteed by the nonsingularity of $b(x) + c(x)\gamma(x)$, Assumption 3.5 and the following equation

$$\begin{bmatrix} b(x) + c(x)\gamma(x) & 0 \\ B_1(x) + C_1(x)\gamma(x) & \hat{C}_1 \\ \vdots & \vdots \\ B_m(x) + C_m(x)\gamma(x) & \hat{C}_m \end{bmatrix} = \begin{bmatrix} b(x) & c(x) \\ B_1(x) & C_1(x) \\ \vdots & \vdots \\ B_m(x) & C_m(x) \end{bmatrix} \begin{bmatrix} I & 0 \\ \gamma(x) & I \end{bmatrix} \begin{bmatrix} I & -[b(x) + c(x)\gamma(x)]^{-1}c(x) \\ 0 & I \end{bmatrix}$$

3.4 Simulation Study: Constrained Manipulator with Flexible Joints

In this section, a constrained robotic system will be studied as an example to illustrate the methodology proposed in this chapter. Consider a constrained two-link robotic manipulator with two flexible joints [39]. Suppose that the end-effector is in contact with a straight line constraint. Its dynamic model is expressed as

$$\begin{aligned} M(q)\ddot{q} + \beta(q,\dot{q}) + g(q) + Kq - K\theta &= b(q)\lambda & (3.54) \\ R\ddot{\theta} + D\dot{\theta} - Kq + K\theta &= u & (3.55) \\ \phi(q) &= 0 & (3.56) \end{aligned}$$

where $q = (q_1, q_2)^T$ contains the link angles, $\theta = (\theta_1, \theta_2)^T$ the rotor angles, and $u = (u_1, u_2)^T$ the two inputs to the joint motors. R, K and D are the inertia matrix of joint motors, the matrix of spring stiffness and the joint friction coefficient matrix respectively, with the value $R = diag(1, 1)$, $K = diag(100, 100)$ and $D = diag(d, d)$. We assume d as an unknown parameter with the true value 0.1.

The manipulator inertia matrix is

$$M(q) = \begin{bmatrix} (l_2)^2 m_2 + 2l_1 l_2 m_2 \cos(q_2) + (l_1)^2(m_1 + m_2) & (l_2)^2 m_2 + l_1 l_2 m_2 \cos(q_2) \\ (l_2)^2 m_2 + l_1 l_2 m_2 \cos(q_2) & (l_2)^2 m_2 \end{bmatrix}$$

and the Coriolis, centrifugal and gravity terms are combined as

$$\beta(q, \dot{q}) + g(q) = \begin{bmatrix} -m_2 l_1 l_2 \dot{q}_2(2\dot{q}_1 + \dot{q}_2)\sin(q_2) + g m_2 l_2 \cos(q_1 + q_2) + g l_1(m_1 + m_2)\cos(q_1) \\ m_2 l_1 l_2 (\dot{q}_1)^2 \sin(q_2) + g m_2 l_2 \cos(q_1 + q_2) \end{bmatrix}$$

where the parameter values are $l_1 = l_2 = 0.3$ m, $m_1 = m_2 = 1$ kg, and $g = 9.8$ m/s^2.

Its Jacobian matrix is given by

$$b(q) = \begin{bmatrix} l_1(\cos(q_1) + A\sin(q_1)) + l_2(\cos(q_1 + q_2) + A\sin(q_1 + q_2) \\ l_2(\cos(q_1 + q_2) + A\sin(q_1 + q_2) \end{bmatrix}$$

The constraint is assumed to be a straight line described by

$$\phi(q) = y - Ax - B$$

where $A = -1$ and $B = 0.28$. Rewrite the straight line constraint in joint angles

$$\phi(q) = l_1(\sin(q_1) - A\cos(q_1)) + l_2(\sin(q_1 + q_2) - A\cos(q_1 + q_2)) - B = 0$$

It is easily seen that the system (3.54)-(3.56) is a DAE system. In the following, Algorithms 3.3 and 3.4 will be applied to convert this DAE system into an ODE system and then an adaptive stabilizing controller will be designed.

Let $x_1 = q_1, x_2 = q_2, x_3 = \dot{q}_1, x_4 = \dot{q}_1, x_5 = \theta_1, x_6 = \theta_2, x_7 = \dot{\theta}_1,\ x_8 = \dot{\theta}_2$ and $z = \lambda$. As a result, the system (3.54)- (3.56) can be put into the following form

$$
\begin{aligned}
\dot{x}_1 &= x_3 \\
\dot{x}_2 &= x_4 \\
\begin{bmatrix} \dot{x}_3 & \dot{x}_4 \end{bmatrix}^T &= M^{-1}(x_1,x_2)N(x_1,x_2,x_3,x_4,x_5,x_6) + M^{-1}(x_1,x_2)b(x_1,x_2)z \\
\dot{x}_5 &= x_7 \\
\dot{x}_6 &= x_8 \\
\dot{x}_7 &= u_1 + 100x_1 - 100x_5 - dx_7 \\
\dot{x}_8 &= u_2 + 100x_2 - 100x_6 - dx_8 \\
0 &= \phi(x_1,x_2)
\end{aligned}
$$

where $N(x_1,x_2,x_3,x_4,x_5,x_6) = -\beta(x_1,x_2,x_3,x_4) - g(x_1,x_2) - \begin{bmatrix} 100x_1 \\ 100x_2 \end{bmatrix} + \begin{bmatrix} 100x_5 \\ 100x_6 \end{bmatrix}$.

Performing Algorithm 3.3 on $0 = \phi(x_1,x_2)$ gives $r^1 = 2$, $\phi_0(x_1,x_2) = l_1(\sin(q_1) - A\cos(q_1)) + l_2(\sin(q_1+q_2) - A\cos(q_1+q_2)) - B$, $\phi_1(x_1,x_2) = l_1(x_3\cos(x_1) + Ax_3\sin(x_1)) + l_2((x_3+x_4)\cos(x_1 + x_2) + A(x_3+x_4)\sin(x_1+x_2))$, $a(x) = \Phi_1 + \Phi_2 M^{-1}(x_1,x_2)N(x_1,x_2,x_3,x_4,x_5,x_6)$, $b(x) = \Phi_2 M^{-1}(x_1,x_2)b(x_1,x_2)$, $c = 0$, and $d = 0$ where

$$
\begin{aligned}
\Phi_1 &= -l_1(x_3)^2\sin(x_1) + Al_1(x_3)^2\cos(x_1) - l_2(x_3+x_4)^2\sin(x_1+x_2) + Al_2(x_3+x_4)^2\cos(x_1+x_2) \\
\Phi_2 &= [l_1\cos(x_1) + Al_2\sin(x_1) + l_2\cos(x_1+x_2) + Al_2\sin(x_1+x_2),\ l_2\cos(x_1+x_2) + Al_2\sin(x_1+x_2)]
\end{aligned}
$$

Solving algebraic equations $0 = \phi_0(x_1,x_2)$ and $0 = \phi_1(x_1,x_2)$ for x_1 and x_3 gives $x_1 = P(x_2)$ and $x_3 = Q(x_1,x_2,x_4)$ where

$$
\begin{aligned}
P(x_2) &= \arcsin\left(\left(a'B - b'\sqrt{(a')^2 + (b')^2 - B^2}\right)/((a')^2 + (b')^2)\right) \\
Q(x_1,x_2,x_4) &= -x_4(l_2\cos(x_1+x_2) + Al_2\sin(x_1+x_2))/(l_1\cos(x_1) \\
&\quad + Al_1\sin(x_1) + l_2\cos(x_1+x_2) + Al_2\sin(x_1+x_2))
\end{aligned}
$$

with $a^{'} = l_1 + l_2 \cos(x_2) + Al_2 \sin(x_2)$ and $b^{'} = -Al_1 + l_2 \sin(x_2) - Al_2 \cos(x_2)$.

Solving the algebraic equation $0 = a(x) + b(x)z$ yields $z = -b^{-1}(x)a(x)$. By substituting z, x_1 and x_3 into the original system, the following ODE system is obtained, which is in the lower triangular form.

$$
\begin{aligned}
\dot{x}_2 &= x_4 \\
\dot{x}_4 &= \chi_{21}(x_1, x_2)x_5 + \chi_{22}(x_1, x_2)x_6 + \kappa_2(x_1, x_2, x_3, x_4) \\
\dot{x}_5 &= x_7 \\
\dot{x}_6 &= x_8 \\
\dot{x}_7 &= u_1 + 100x_1 - 100x_5 - dx_7 \\
\dot{x}_8 &= u_2 + 100x_2 - 100x_6 - dx_8
\end{aligned}
$$

where

$$
\begin{aligned}
\begin{bmatrix} \chi_{11} & \chi_{12} \\ \chi_{21} & \chi_{22} \end{bmatrix} &= M^{-1}(x_1, x_2)\left(\begin{bmatrix} 100 & \\ & 100 \end{bmatrix}\right. \\
&\quad \left. -b(x_1, x_2)\left[\Phi_2(x_1, x_2)M^{-1}(x_1, x_2)b(x_1, x_2)\right]^{-1}\Phi_2(x_1, x_2)M^{-1}(x_1, x_2)\begin{bmatrix} 100 & \\ & 100 \end{bmatrix}\right) \\
\begin{bmatrix} \kappa_1 & \kappa_2 \end{bmatrix}^T &= M^{-1}(x_1, x_2)\left[-\beta(x_1, x_2, x_3, x_4) - g(x_1, x_2) - \begin{bmatrix} 100x_1 & 100x_2 \end{bmatrix}^T\right]
\end{aligned}
$$

The desired link angles (q_1, q_2) and rotor angles (θ_1, θ_2) are assumed as $(115.73^0, 0)$ and $(115^0, 0)$ respectively. The desired contact force λ is set to be 3.187 N. Introducing the change of coordinates of $\bar{x}_1 = x_1 - 2.0199, \bar{x}_2 = x_2, \bar{x}_3 = x_3, \bar{x}_4 = x_4, \bar{x}_5 = x_5 - 2.0071, \bar{x}_6 = x_6, \bar{x}_7 = x_7$, $\bar{x}_8 = x_8$ yields the following system with the origin as the equilibrium point

$$
\begin{aligned}
\dot{\bar{x}}_2 &= \bar{x}_4 \\
\dot{\bar{x}}_4 &= \chi_{21}(\bar{x}_1, \bar{x}_2)(\bar{x}_5 + 2.0071) + \chi_{22}(\bar{x}_1, \bar{x}_2)\bar{x}_6 + \kappa_2(\bar{x}_1, \bar{x}_2, \bar{x}_3, \bar{x}_4)
\end{aligned}
$$

$$
\begin{aligned}
\dot{\bar{x}}_5 &= \bar{x}_7 \\
\dot{\bar{x}}_6 &= \bar{x}_8 \\
\dot{\bar{x}}_7 &= u_1 + 100\bar{x}_1 - 100(\bar{x}_5 + 2.0071) - d\bar{x}_7 \\
\dot{\bar{x}}_8 &= u_2 + 100\bar{x}_2 - 100\bar{x}_6 - d\bar{x}_8
\end{aligned}
$$

The controllers u_1 and u_2 are given as

$$
\begin{aligned}
u_1 &= -c_6(\bar{x}_7 - \alpha_4) - (\bar{x}_5 - \alpha_2) - 100\bar{x}_1 + 100\bar{x}_5 + 200.71 + \hat{d}\bar{x}_7 + \frac{\partial \alpha_4}{\partial \bar{x}_2}\dot{\bar{x}}_2 + \frac{\partial \alpha_4}{\partial \bar{x}_4}\dot{\bar{x}}_4 + \frac{\partial \alpha_4}{\partial \bar{x}_5}\dot{\bar{x}}_5 \\
u_2 &= -c_7(\bar{x}_8 - \alpha_5) - (\bar{x}_6 - \alpha_3) - 100\bar{x}_2 + 100\bar{x}_6 + \hat{d}\bar{x}_8 + \frac{\partial \alpha_5}{\partial \bar{x}_2}\dot{\bar{x}}_2 + \frac{\partial \alpha_4}{\partial \bar{x}_4}\dot{\bar{x}}_4 + \frac{\partial \alpha_4}{\partial \bar{x}_6}\dot{\bar{x}}_6
\end{aligned}
$$

and the parameter estimator is given as

$$
\dot{\hat{d}} = -\gamma[(\bar{x}_7 - \alpha_4)\bar{x}_7 + (\bar{x}_8 - \alpha_5)\bar{x}_8]
$$

where

$$
\begin{aligned}
\alpha_1 &= -c_1\bar{x}_2 \\
\alpha_2 &= -c_2\chi_{21}^{-1}(\bar{x}_4 - \alpha_1) - \chi_{21}^{-1}(\bar{x}_2 + 2.0071\chi_{21} + \kappa_2) \\
\alpha_3 &= -c_3\chi_{22}^{-1}(\bar{x}_4 - \alpha_1) - c_1\chi_{22}^{-1}\bar{x}_4 \\
\alpha_4 &= -c_4(\bar{x}_5 - \alpha_2) - \chi_{21}(\bar{x}_4 - \alpha_1) + \frac{\partial \alpha_2}{\partial \bar{x}_2}\dot{\bar{x}}_2 + \frac{\partial \alpha_2}{\partial \bar{x}_4}\dot{\bar{x}}_4 \\
\alpha_5 &= -c_5(\bar{x}_6 - \alpha_3) - \chi_{22}(\bar{x}_4 - \alpha_1) + \frac{\partial \alpha_3}{\partial \bar{x}_2}\dot{\bar{x}}_2 + \frac{\partial \alpha_3}{\partial \bar{x}_4}\dot{\bar{x}}_4
\end{aligned}
$$

Suppose the initial conditions are $q(0) = \theta(0) = (110.66^0, 10^0)$. The simulation results are shown in Figure 3-1. It is clear that the link angles converge to the equilibrium point $(115.73^0, 0)$ on the constraint line. The rotor angles converge to $(115^0, 0)$ to offset the gravity effects. The estimation of the unknown parameter d converges to its true value 0.1 very quickly. The contact force converges to the desired value 3.187 N. In the transient performance, the contact force is always positive, which implies that the contact is maintained throughout the motion.

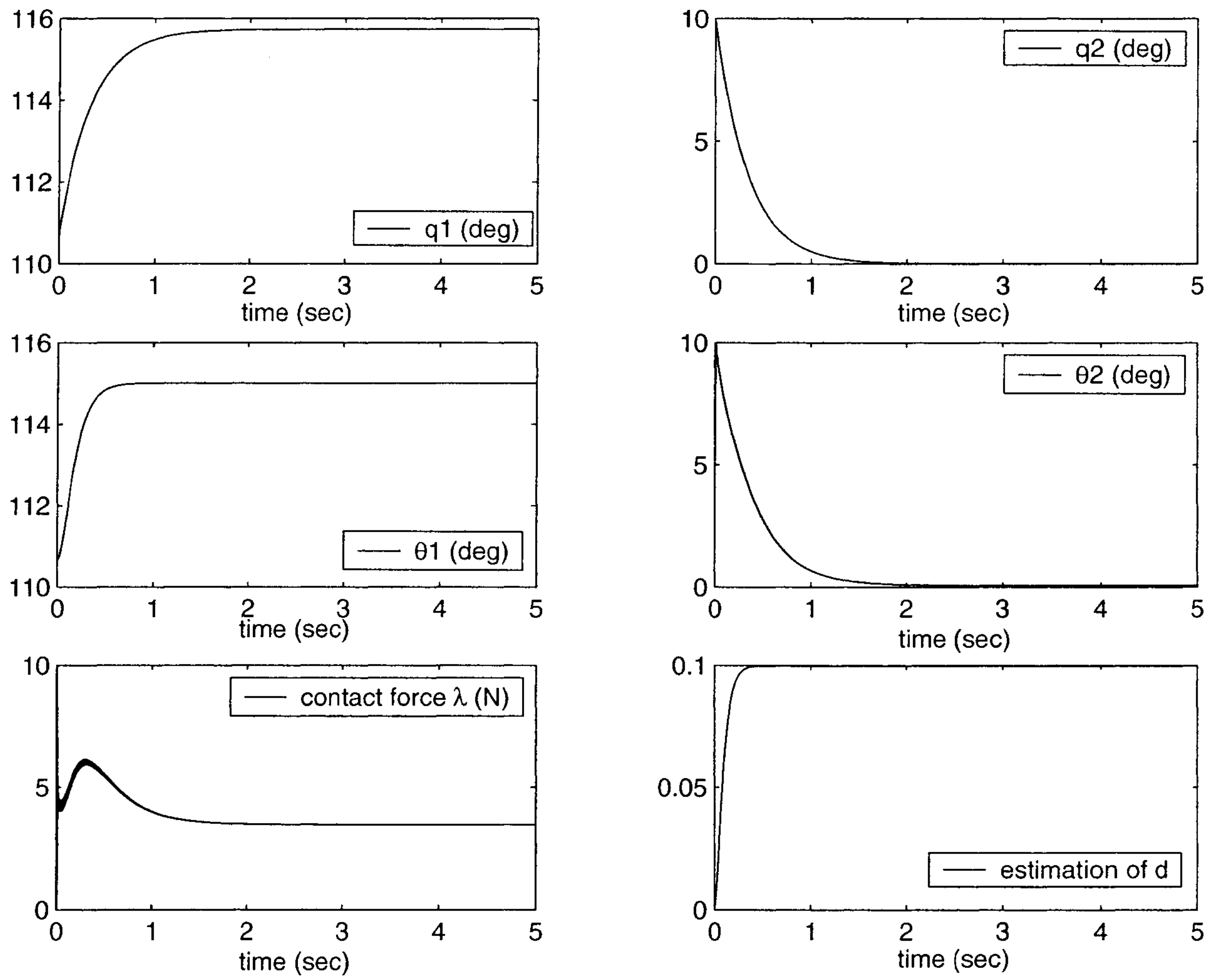

Figure 3-1: Responses of the constrained manipulator with flexible joints

3.5 Conclusion

The adaptive control problem has been solved for nonlinear DAE systems. Following the approach we developed in this chapter, we can design an adaptive controller for a nonlinear DAE system with unknown parameters appearing linearly in both differential and algebraic equations. Our methodology consists of three algorithms, by which the original system has been transformed to one equivalent system with lower triangular form. Adaptive backstepping is applied to design the adaptive controller, which guarantees the global asymptotic stability if the change of coordinates is defined globally. The example is given to illustrate the methodology proposed in this chapter.

Chapter 4

Experimental Study: Set Point Control of Parallel Robot

4.1 Introduction

In this chapter, one parallel robotic manipulator will be studied as an example of differential algebraic equation (DAE) systems. The nonlinear controller is designed by the backstepping technique and also implemented on the experimental system. The experimental system is shown in Figure 4-1.

The mechanisms of parallel robots are also known as closed kinematic chains [5]. Figure 4-2 shows planar examples of a parallel robot and a serial robot. Different from serial robots, the links of parallel robots are connected in series as well as in parallel combinations forming one or more closed-link loops. Typically, not all the joints of parallel robots are actuated. Generally, for parallel robots, the actuators are placed lower in the link chain. This makes the moving parts lighter which leads to greater efficiency and faster acceleration at the end-effector. Parallel robots also offer greater rigidity to weight ratio, which makes greater payload handling capability for the same number of actuators. Parallel robots are more suitable for fast assembly lines, flight simulators and robotics machining, etc.

A parallel robot can be considered as a DAE system. For a parallel robot with n dof and n' joints, we can obtain a DAE system with n' differential equations and $n'-n$ algebraic equations. Through solving the DAE system, we can get an ODE system with n independent differential

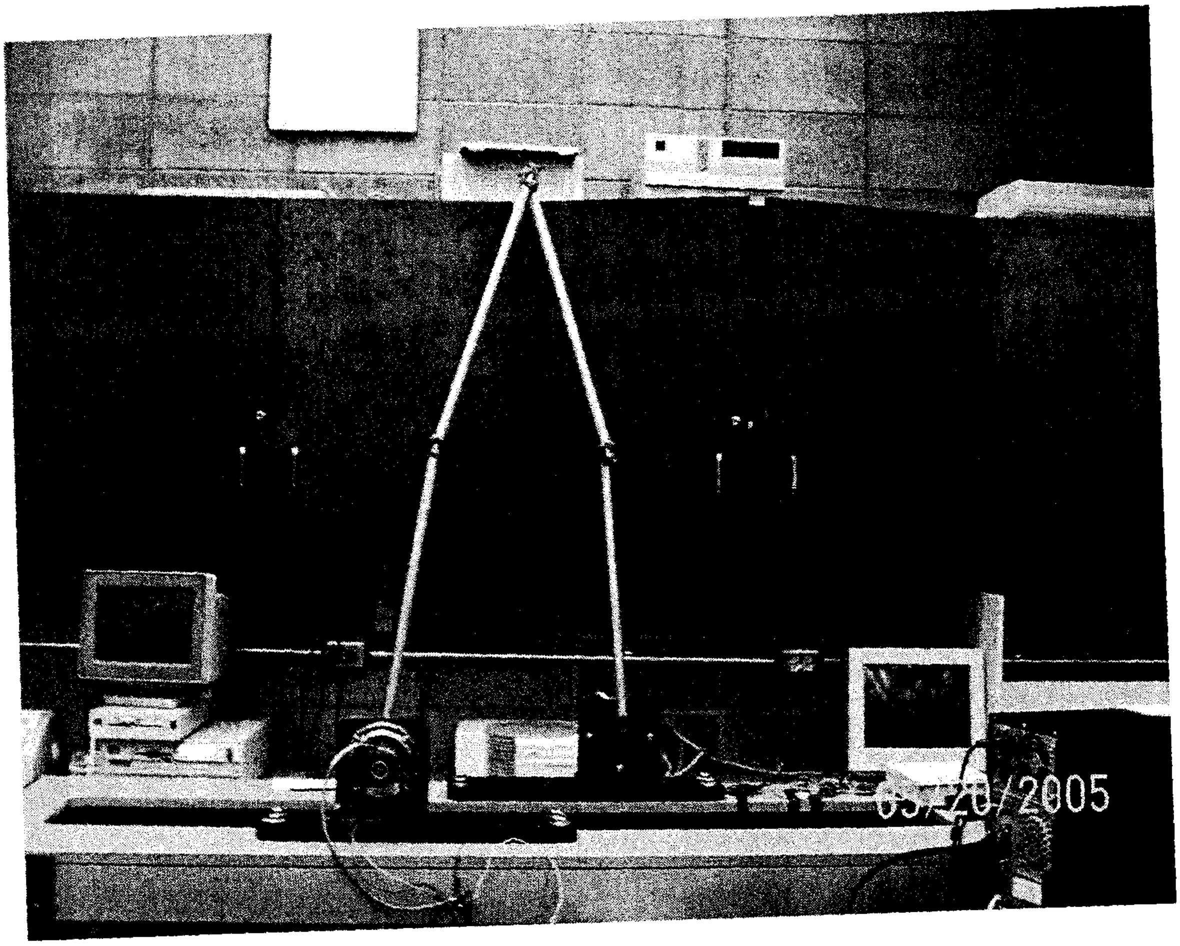

Figure 4-1: Parallel Robot

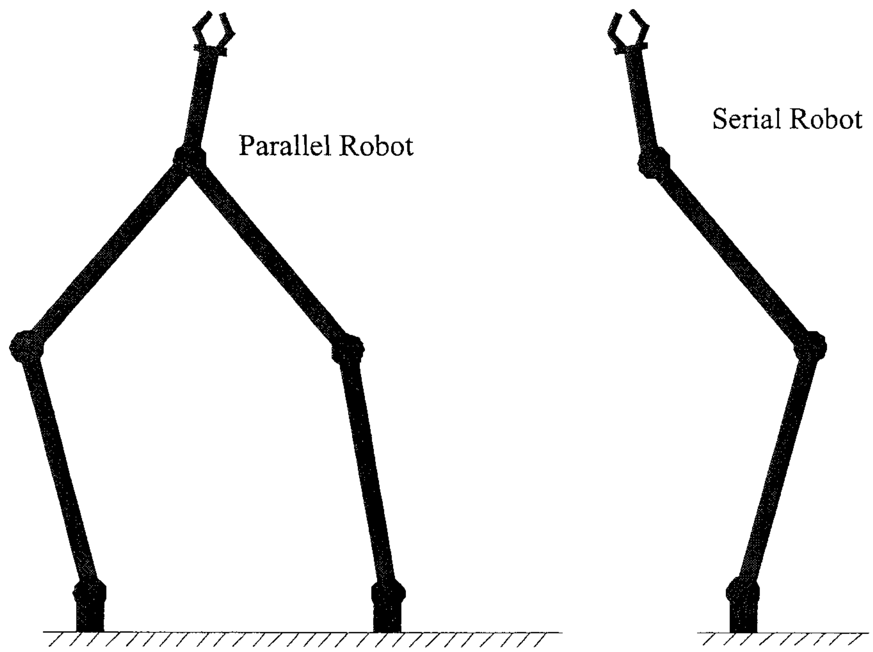

Figure 4-2: Parallel Robot and Serial Robot

equations and n independent state variables. In our case, $n = 2$ and $n' = 4$, so we can get a DAE system with four differential equations and two algebraic equations.

In this chapter, we are going to design a nonlinear controller by the backstepping technique for the parallel robot based on the dynamical model derived in [5]. The designed controller will also be implemented and both of the simulation and experiment results will be shown. For comparison, the simulation and experimental results of the PD controller will be also shown.

4.2 Experimental Setup

The experimental system, parallel robot, is shown in Figure 4-1. It has four links connected through revolute joints. Two of the links, Link 1 and Link 2 are actuated with DC motors while the other two are passive. The motors are driven by two H-Bridge circuits, which are controlled by PWM signals from the computer. The robot is controlled by PC with two DAQ (data acquisition) boards (PCI-6024E and PCI-MIO-16E), which are plugged in PCI slots inside the computer. These two DAQ boards are connected with two SCB-68 connector blocks through

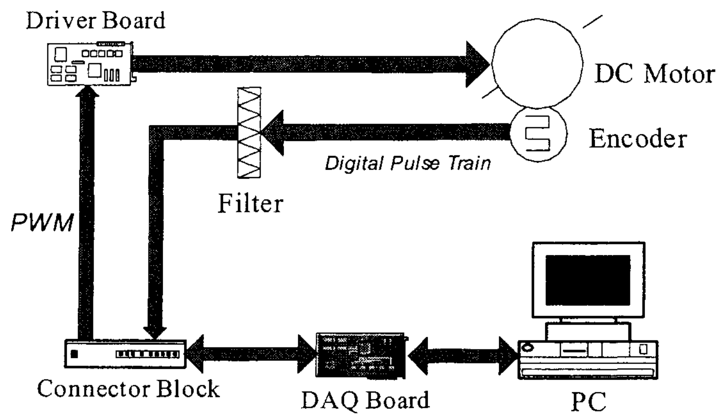

Figure 4-3: Diagram of the control system

shielded cables. The boards and connector blocks with cables are from National Instruments.

The motors are from Kollmorgen Motion Technologies Group with the gear ratio 99 to 1 and the peak torque 17.1 N-m. The optical encoders are built in the motors with the resolution of 1000 pulses per revolution.

Position feedbacks of Link 1 and Link 2 are provided by the optical encoders. Two analog low pass filters are used for filtering the signals from the encoders.

The controller is implemented by using Visual C++. The sampling period is controlled by a timer in Visual C++. The controller is designed with feedback of link positions and link velocities. Velocity feedbacks are calculated digitally based on the position measurements. Two digital low pass filters are introduced for velocity calculation, which are given by

$$v_{k+1} = (p_{k+1} - p_k + \tau v_k)/(\tau + T) \tag{4.1}$$

where v_k and v_{k+1} are the angular velocity at the sampling instants k and $k+1$, p_k and p_{k+1} are the position measurements of the links at the sampling instants k and $k+1$, respectively. T is the sampling period, and τ is the time constant set as 0.1.

In each sampling period, the computer obtains the current positions and velocities of Link 1 and Link 2, calculates the control input in terms of duty cycles of the PWM signals, and sends the PWM signals to driver boards to control the DC motors. The whole control system works in the way illustrated in Figure 4-3.

In the experiment, we set the sampling period as 10 milliseconds, which is fast enough to follow the movement of the robot and also adequate for the computer to finish reading signals from the boards, calculating and sending signals back to the boards. The driver board's voltage is 15 volts, which is the maximum voltage that the driver board can provide for the DC motors.

4.3 Mathematical Model

The dynamical model of the robot, presented in [5], is described as follows

$$\begin{aligned} D'(q')\,\ddot{q} + C'(q',\dot{q}')\dot{q} + g'(q') &= \phi_{q'}^{T}(q')\lambda + u' && (4.2) \\ 0 &= \phi(q') && (4.3) \end{aligned}$$

where $q' = \begin{bmatrix} q_1 & q_2 & q_3 & q_4 \end{bmatrix}^T$ is the vector of the generalized coordinates, $u' = \begin{bmatrix} u^T & 0 & 0 \end{bmatrix}^T$ with u the torque vector of the motors , $D'(q') \in R^{4\times4}$ is the inertia matrix, $C'\ (q', \dot{q}') \in R^4$ represents the centrifugal and Coriolis term, and $g'(q') \in R^4$ is the gravity vector, $\phi(q')$ represents the constraints by $n' - n = 2$ independent algebraic equations, $\phi_{q'}^{T}(q')$ is the Jacobian matrix of $\phi(q')$, λ is the vector of Lagrangian multipliers and $\phi^T(q')\lambda$ represents the constraint generalized force vector. $\phi(q')$ is at least twice continuously differentiable. The matrices $D'(q')$, $C'(q', \dot{q}')$, $g(q')$, $\phi(q')$ and $\phi_{q'}(q')$ are given as follows

$$D'(q') = \begin{bmatrix} d_{11} & 0 & d_{13} & 0 \\ 0 & d_{22} & 0 & d_{24} \\ d_{31} & 0 & d_{33} & 0 \\ 0 & d_{42} & 0 & d_{44} \end{bmatrix} \tag{4.4}$$

$$C'(q', \dot{q}') = \begin{bmatrix} h_1\dot{q}_3 & 0 & h_1(\dot{q}_1 + \dot{q}_3) & 0 \\ 0 & h_2\dot{q}_4 & 0 & h_2(\dot{q}_2 + \dot{q}_4) \\ -h_1\dot{q}_1 & 0 & 0 & 0 \\ 0 & -h_2\dot{q}_2 & 0 & 0 \end{bmatrix} \tag{4.5}$$

$$g(q') = \begin{bmatrix} (m_1 l_1 + m_3 a_1)\cos(q_1) + m_3 l_3 \cos(q_1 + q_3) \\ (m_2 l_2 + m_4 a_2)\cos(q_2) + m_4 l_4 \cos(q_2 + q_4) \\ m_3 l_3 \cos(q_1 + q_3) \\ m_4 l_4 \cos(q_2 + q_4) \end{bmatrix} \mathsf{g} \tag{4.6}$$

$$\phi(q') = \begin{bmatrix} a_1 \cos(q_1) + a_3 \cos(q_1 + q_3) - c - a_2 \cos(q_2) - a_4 \cos(q_2 + q_4) \\ a_1 \sin(q_1) + a_3 \sin(q_1 + q_3) - a_2 \sin(q_2) - a_4 \sin(q_2 + q_4) \end{bmatrix} = 0 \tag{4.7}$$

where

$$\begin{aligned}
d_{11} &= m_1 (l_1)^2 + m_3 \left((a_1)^2 + (l_3)^2 + 2 a_1 l_3 \cos(q_3)\right) + I_1 + I_3 \\
d_{13} &= m_3 \left((l_3)^2 + a_1 l_3 \cos(q_3)\right) + I_3 \\
d_{22} &= m_2 (l_2)^2 + m_4 \left((a_2)^2 + (l_4)^2 + 2 a_2 l_4 \cos(q_4)\right) + I_2 + I_4 \\
d_{24} &= m_4 \left((l_4)^2 + a_2 l_4 \cos(q_4)\right) + I_4 \\
d_{31} &= d_{13} \\
d_{33} &= m_3 (l_3)^2 + I_3 \\
d_{42} &= d_{24} \\
d_{44} &= m_4 (l_4)^2 + I_4 \\
h_1 &= -m_3 a_1 l_3 \sin(q_3) \\
h_2 &= -m_4 a_2 l_4 \sin(q_4)
\end{aligned}$$

As defined in Figure 4-4, m_i, a_i, and l_i are the mass, length of link i and distance to the center of mass from the previous joint, respectively. The inertia of link i about the line through the center of mass parallel to the axis of rotation is defined as I_i. The parameters corresponding to Link 2 and Link 4 are similar to the parameters of Link 1 and Link 3. The value of those parameters are measured and given in Table 4-1. The distance between the shafts of the two motors is given by $c = 0.4240$ m and the gravity constant is $\mathsf{g} = 9.81$ m/s^2.

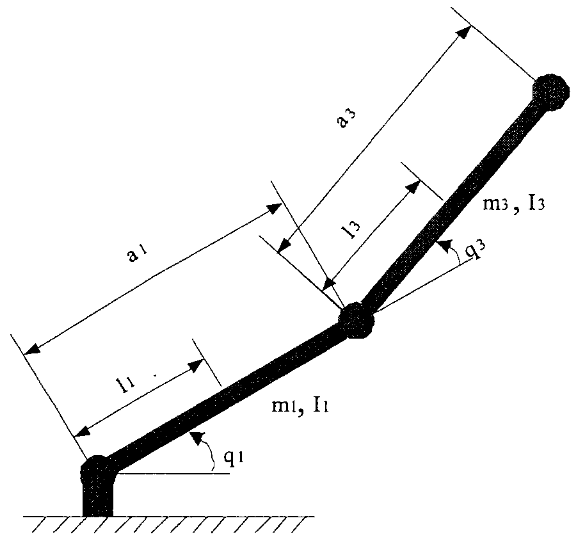

Figure 4-4: Link 1 and Link 3 of the parallel robot

Link i	m_i (kg)	a_i (m)	l_i (m)	I_i (kg·m^2)
1	0.1950	0.4600	0.3367	4.567×10^{-3}
2	0.1950	0.4600	0.3367	4.567×10^{-3}
3	0.2538	0.4600	0.2400	8.626×10^{-3}
4	0.2538	0.4600	0.2400	8.626×10^{-3}

Tab 4-1 Link Parameters

Obviously, the system (4.2)-(4.3) is a DAE system. In the next section, both PD controller and nonlinear stabilizing controller will be given for the system (4.2)-(4.3).

4.4 Controller Design

Considering the DAE system (4.2)-(4.3), we can convert this DAE system into the following ODE system by the method introduced in Chapter 3. The equivalent ODE system (4.8) has two independent controlled variables, q_1 and q_2.

$$D(q')\,\ddot{q} + C(q', \dot{q}')\dot{q} + g(q') = u \tag{4.8}$$

where

$$
\begin{aligned}
D(q') &= \rho(q')^T D^{'}(q')\rho(q') && (4.9)\\
C(q',\dot{q}') &= \rho(q')^T C^{'}(q',\dot{q}')\rho(q') + \rho(q')^T D^{'}(q')\dot{\rho}(q') && (4.10)\\
g(q') &= \rho(q')^T g^{'}(q') && (4.11)
\end{aligned}
$$

with $q' = \begin{bmatrix} q_1 & q_2 & q_3 & q_4 \end{bmatrix}^T$ and $q = \begin{bmatrix} q_1 & q_2 \end{bmatrix}^T$ and $\rho(q')$ and $\dot{\rho}(q')$ are given in (4.12) and (4.14) below. At this point, in order to convert the original DAE system into the ODE system (4.8), we also need $\rho(q^{'})$ and the expression of q_3 and q_4 in terms of q_1 and q_2, which are given as follows.

$$
\rho(q^{'}) = \psi_{q'}^{-1}(q^{'}) \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{4.12}
$$

where

$$
\psi_{q'}(q^{'}) = \begin{bmatrix} \psi_{q'}(1,1) & \psi_{q'}(1,2) & -a_3\sin(q_1+q_3) & a_4\sin(q_2+q_4) \\ \psi_{q'}(2,1) & \psi_{q'}(2,2) & a_4\cos(q_1+q_3) & -a_4\cos(q_2+q_4) \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{4.13}
$$

with $\psi_{q'}(1,1) = -a_1\sin(q_1) - a_3\sin(q_1+q_3)$, $\psi_{q'}(1,2) = a_2\sin(q_2) + a_4\sin(q_2+q_4)$, $\psi_{q'}(2,1) = a_1\cos(q_1) + a_3\cos(q_1+q_3)$, $\psi_{q'}(2,2) = -a_2\cos(q_2) - a_4\cos(q_2+q_4)$.

$$
\dot{\rho}(q') = -\psi_{q'}^{-1}(q^{'})\,\dot{\psi}_{q'}\,(q^{'},\dot{q}^{'})\rho(q') \tag{4.14}
$$

and q_4, q_3 are calculated as

$$
q_4 = \tan^{-1}\left(\frac{B(q_1,q_2)}{A(q_1,q_2)}\right) + \tan^{-1}\left(\frac{\pm\sqrt{A(q_1,q_2)^2 + B(q_1,q_2)^2 - C(q_1,q_2)^2}}{C(q_1,q_2)}\right) - q_2 \tag{4.15}
$$

$$q_3 = \tan^{-1}\left(\frac{\mu(q_1,q_2)+a_4\sin(q_2+q_4)}{\lambda(q_1,q_2)+a_4\cos(q_2+q_4)}\right) - q_1 \tag{4.16}$$

where

$$\begin{aligned}
A(q_1,q_2) &= 2a_4\lambda(q_1,q_2)\\
B(q_1,q_2) &= 2a_4\mu(q_1,q_2)\\
C(q_1,q_2) &= (a_3)^2-(a_4)^2-\lambda(q_1,q_2)^2-\mu(q_1,q_2)^2\\
\lambda(q_1,q_2) &= a_2\cos(q_2)-a_1\cos(q_1)+c\\
\mu(q_1,q_2) &= a_2\sin(q_2)-a_1\sin(q_1)
\end{aligned}$$

Assign $x_1 = q_1 - q_1^d$, $x_2 = q_2 - q_2^d$, $x_3 = \dot{q}_1$, $x_4 = \dot{q}_2$ with q_1^d and q_2^d being the desired angle of q_1 and q_2 . Therefore, the system (4.8) is rewritten as

$$\dot{x}_1 = x_3 \tag{4.17}$$

$$\dot{x}_2 = x_4 \tag{4.18}$$

$$\begin{pmatrix}\dot{x}_3\\ \dot{x}_4\end{pmatrix} = H(x_1,x_2,x_3,x_4)+v \tag{4.19}$$

where $H(x_1,x_2,x_3,x_4) = \begin{bmatrix} H_1(x_1,x_2,x_3,x_4) & H_2(x_1,x_2,x_3,x_4)\end{bmatrix}^T = -D^{-1}(q')C(q',\dot{q}')\dot{q} - D^{-1}(q')g(q')$ and v is the new control input with $v = \begin{bmatrix} v_1 & v_2 \end{bmatrix}^T = D^{-1}(q')u$. By applying the backstepping technique to design an controller, the system (4.17)-(4.19) is guaranteed to be globally asymptotically stable. This design is based on a recursive procedure. In each step, a Lyapunov function candidate is constructed and by choosing a controller α, the derivative of the Lyapunov function candidate is made negative definite. The design procedure is as follows:

Step 1: Choose the Lyapunov function candidate

$$V_1 = \frac{1}{2}(x_1)^2 + \frac{1}{2}(x_2)^2$$

Differentiating V_1 with respect to time yields

$$\dot{V_1} = -c_1(x_1)^2 - c_2(x_2)^2 + x_1(x_3 - \alpha_1) + x_2(x_4 - \alpha_2)$$

with c_1, c_2 are positive numbers and

$$\begin{aligned} \alpha_1 &= -c_1 x_1 \\ \alpha_2 &= -c_2 x_2 \end{aligned}$$

Step 2: Choose the Lyapunov function candidate

$$V_2 = V_1 + \frac{1}{2}(x_3 - \alpha_1)^2 + \frac{1}{2}(x_4 - \alpha_2)^2$$

Differentiating V_2 with respect to time yields

$$\begin{aligned} \dot{V_2} &= -c_1(x_1)^2 - c_2(x_2)^2 + x_1(x_3 - \alpha_1) + x_2(x_4 - \alpha_2) \\ &\quad +(x_3 - \alpha_1)(H_1 + v_1 - \dot{\alpha_1}) + (x_4 - \alpha_2)(H_2 + v_2 - \dot{\alpha_2}) \\ &= -c_1(x_1)^2 - c_2(x_2)^2 - c_3(x_3 - \alpha_1)^2 - c_4(x_4 - \alpha_2)^2 \end{aligned}$$

with c_3, c_4 are positive numbers and the controller v

$$v = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \begin{pmatrix} -c_3(x_3 - \alpha_1) - x_1 - H_1 + \dot{\alpha_1} \\ -c_4(x_4 - \alpha_2) - x_2 - H_2 + \dot{\alpha_2} \end{pmatrix} \tag{4.20}$$

The control (4.20) makes the derivative of V_2 negative definite, which means the corresponding closed-loop system (4.17)-(4.19) and (4.20) is stable. The control input u is described by v as

$$u = D(q')v \tag{4.21}$$

Therefore, the system (4.8) is stabilized by the control input u.

For comparison, we also give the PD control for the system (4.8) by

$$u = g(q^d) + K_p(q^d - q) - K_v\dot{q}$$

where K_p and K_v are selected to be diagonal matrix, $q^d = \begin{bmatrix} q_1^d & q_2^d \end{bmatrix}^T$ is the desired configuration and $g(q^d)$ is the gravity vector, which is calculated off-line.

As for experiments, the control input is not u, the torques applied to the joints. The direct control input is the armature voltage of the DC motor. Therefore, in order to implement our designed controller in terms of the torque of the motor, we need to convert the torque into the armature voltage of the DC motor. The conversion formula is given as follows

$$u = \frac{GK_t}{R}(V_a - K_e G\omega)$$

where u is the torque applied at the joints, $G = 99$ is the gear ratio of the motor, K_t is the torque constant with the value 2.28 N·cm/Amp, K_e is the back EMF constant with the value 2.39 volts/kRPM, R is the armature resistance with the value 0.640 Ohms, ω is the angular velocity of the motor shaft and V_a is the armature voltage of the motor we apply. Through the armature voltage V_a, we can control the real system from point to point.

4.5 Simulation and Experimental Results

Simulations and experiments are carried out for both PD controllers and nonlinear stabilizing controllers on the parallel robot. Three sets of simulations and experiments are performed for three configurations shown in Figure 4-5, Figure 4-6 and Figure 4-7 .

For the first set, the control objective is to achieve position control with PD controllers of $K_p = diag(11, 11)$ and $K_v = diag(2, 2)$ and nonlinear stabilizing controllers of $c_1 = 10$, $c_2 = 10$, $c_3 = 12$ and $c_4 = 13$. The initial configuration is set as $q_1 = 150^0$, $q_2 = 160^0$ as Figure 4-6 and the desired configuration is $q_1^d = 90^0$, $q_2^d = 100^0$ as Figure 4-5. The gravitational term $g(q^d)$ is calculated to be $\begin{bmatrix} -0.2404 & -0.801 \end{bmatrix}^T$. The simulation results are shown in Figure 4-8 and Figure 4-9. The experimental results are given in Figure 4-10 and Figure 4-11.

The simulations and experiments are also performed to achieve position control with the

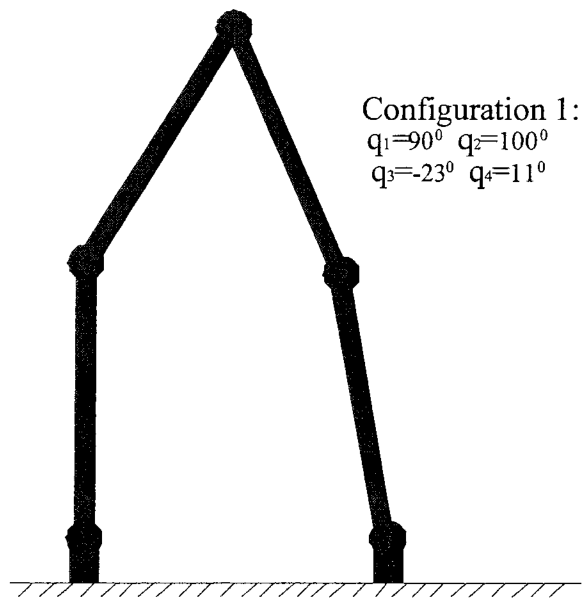

Figure 4-5: Configuration 1 of Parallel Robot

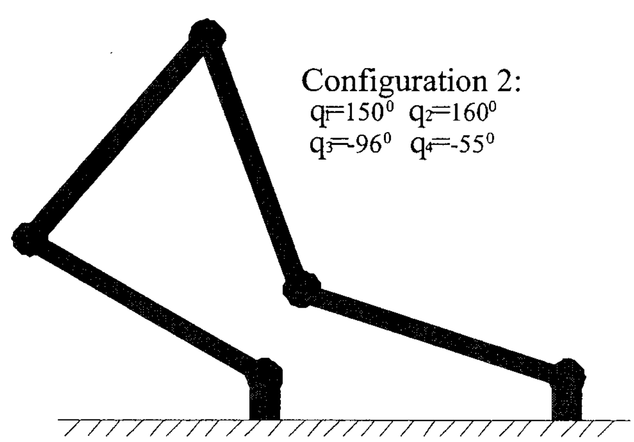

Figure 4-6: Configuration 2 of Parallel Robot

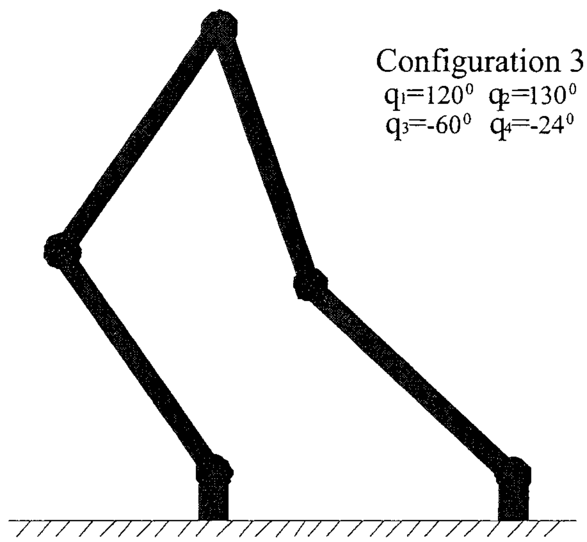

Figure 4-7: Configuration 3 of Parallel Robot

initial configuration of $q_1 = 90^0$, $q_2 = 100^0$, the desired configuration of $q_1^d = 150^0$, $q_2^d = 160^0$, and $g(q^d) = \begin{bmatrix} -1.4273 & -1.8612 \end{bmatrix}^T$. The control gains for PD controllers are kept the same with $K_p = diag(11, 11)$, $K_v = diag(2, 2)$, but for the nonlinear stabilizing controllers, $c_1 = 14$, $c_2 = 14$, $c_3 = 19$ and $c_4 = 20$. The simulation results are provided in Figure 4-12 and Figure 4-13. The experimental results are shown in Figure 4-14 and Figure 4-15.

For the third set, the simulations and experiments are performed to achieve position control with the initial configuration of $q_1 = 90^0$, $q_2 = 100^0$, the desired configuration of $q_1^d = 120^0$, $q_2^d = 130^0$ as Figure 4-7, and $g(q^d) = \begin{bmatrix} -0.9880 & -1.0916 \end{bmatrix}^T$. This time, the control gain is set as $K_p = diag(11, 11)$, $K_v = diag(2, 2)$ for PD controllers and $c_1 = 14$, $c_2 = 14$, $c_3 = 15$ and $c_4 = 20$ for the nonlinear stabilizing controllers. The simulation results are shown in Figure 4-16, Figure 4-17. Figure 4-18 and Figure 4-19 provide the experimental results.

From the given results, we can see that the simulation and experimental results show that both the nonlinear stabilizing controller designed by backstepping and the PD control work well on this parallel robot. The experimental results fit well with the simulation results and the transient responds, setting times are satisfactory.

The responses of the experiments are little slower than the simulations, which are due to the fact that, at the beginning, the expected control input voltages according to the simulations are around 40 volts or higher, while the maximum voltage we can provide from the driver board to the DC motor is only 15 volts. There are also some voltage drops in the circuits and the actual output voltage of the driver boards are around 14.4 volts. Such phenomena exist for both backstepping and PD control scheme.

The steady state control efforts in experiments are smaller than those in the corresponding simulations, which is more obvious in the second and third sets of experiments due to different steady state configurations. The existing friction that has not been considered in our model is the main reason. Due to the existence of friction, there is no need of that high control to balance the link's gravitational torque.

Since the maximum voltage of the driver board is much smaller than the expected voltage, it is reasonable that the control efforts last for around 0.3 seconds in the maximum voltage for the first and second set of experiments. The lasting time is much smaller for the third set of experiments, which is reasonable, considering that the initial error is only 30^0 for the third set of experiments instead of 60^0 for the first and second set of experiments.

There exist the steady state errors about three to four degrees in the experimental results for the second and third sets of experiments. This is caused by the inaccurate measurement for some system parameters and another reason is the friction that has not been taken into account. For the PD control scheme, the errors are caused directly by the gravitational compensation term $g(q^d)$. The required gravitational compensations of the Configuration 2, 3 are much larger than that of the first set. This is why there are no or quite small errors in the first set of experiments. For backstepping control scheme, we can increase the control gains c_1, c_2, c_3 and c_4 to decrease the steady state errors. But considering large control gains will lead to high armature current, the control gains can not be too large with our power supply's maximum current being 3.5 Amp. There must be a trade-off between the error and the armature current.

4.6 Conclusion

In this chapter, we design and implement a nonlinear stabilizing controller by the backstepping technique on the parallel robotic system. The parallel robotic system is considered as a DAE

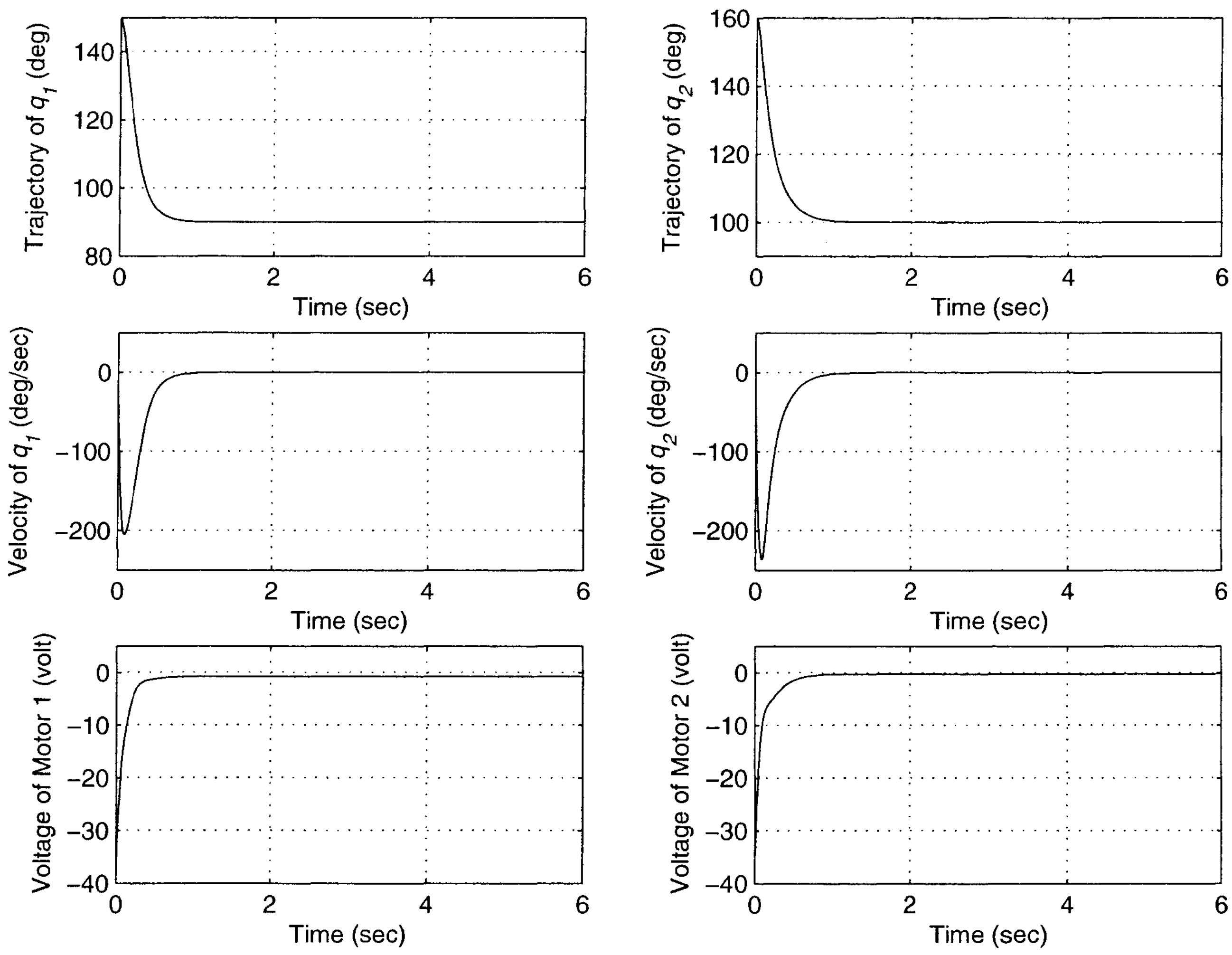

Figure 4-8: Point to Point Control, Simulation Results: PD Scheme, from $q_1 = 150^0$, $q_2 = 160^0$ to $q_1^d = 90^0$, $q_2^d = 100^0$

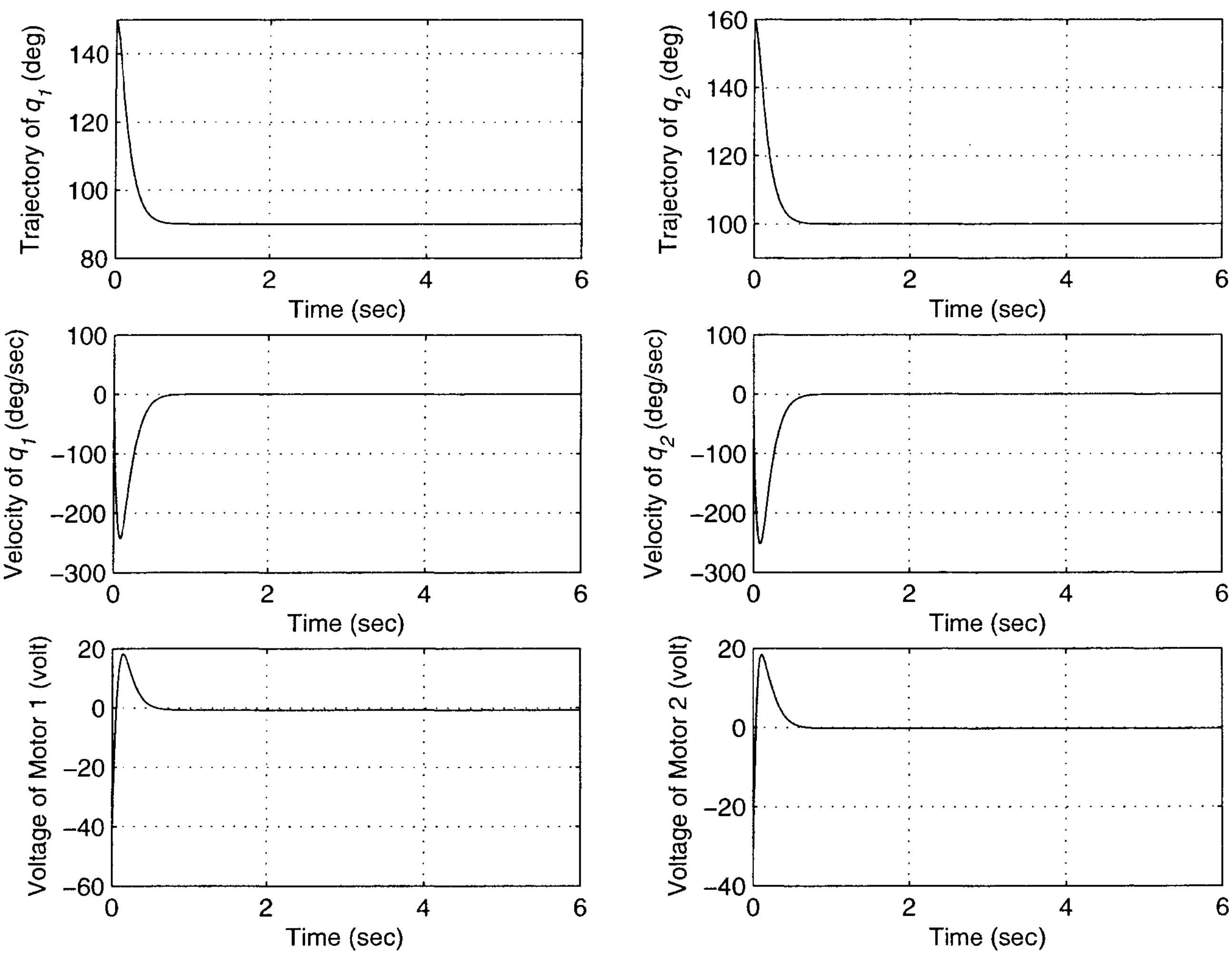

Figure 4-9: Point to Point Control, Simulation Results: Backstepping Scheme, from $q_1 = 150^0$, $q_2 = 160^0$ to $q_1^d = 90^0$, $q_2^d = 100^0$

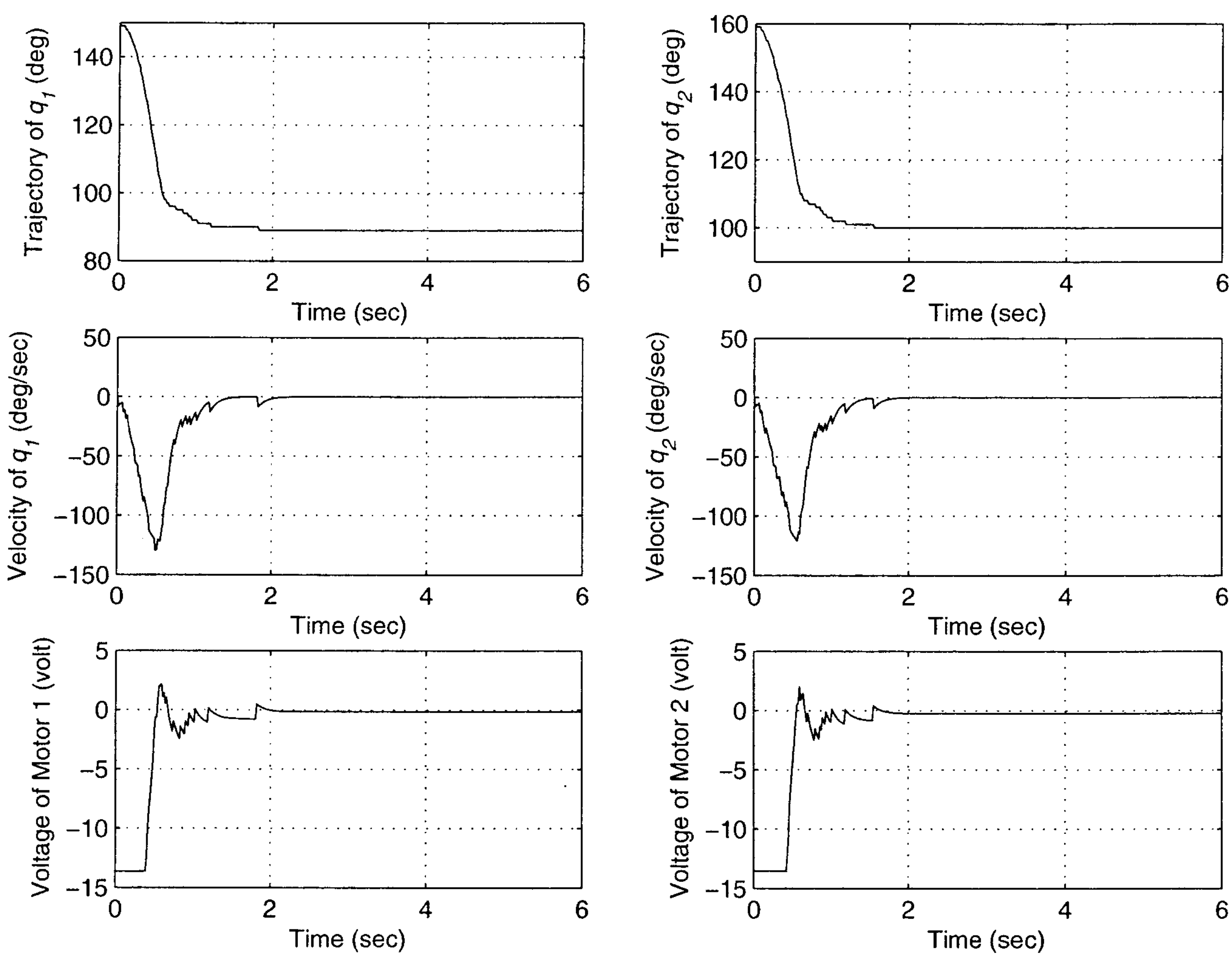

Figure 4-10: Point to Point Control, Experimental Results: PD Scheme, from $q_1 = 150^0$, $q_2 = 160^0$ to $q_1^d = 90^0$, $q_2^d = 100^0$

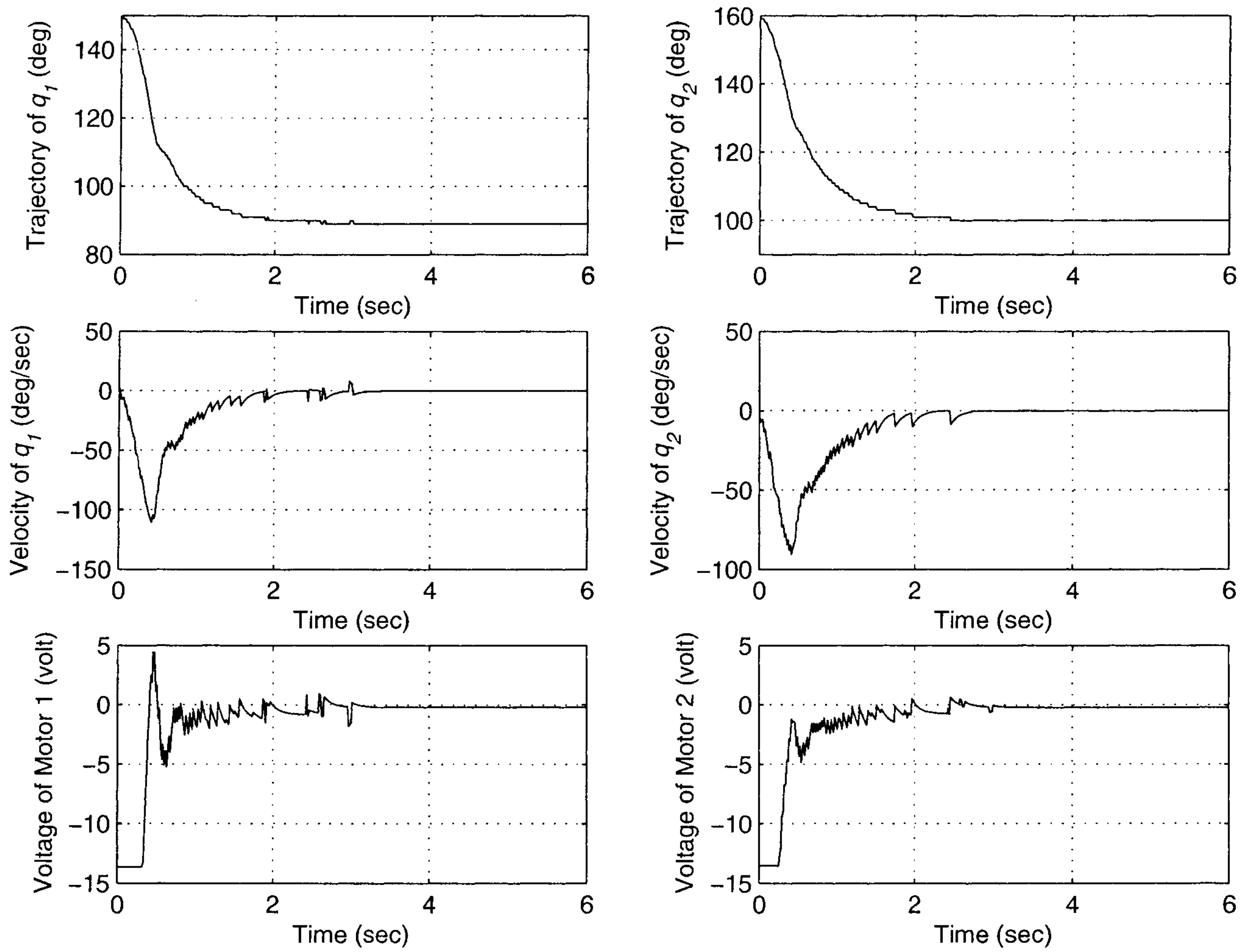

Figure 4-11: Point to Point Control, Experimental Results: Backstepping Scheme, from $q_1 = 150^0$, $q_2 = 160^0$ to $q_1^d = 90^0$, $q_2^d = 100^0$

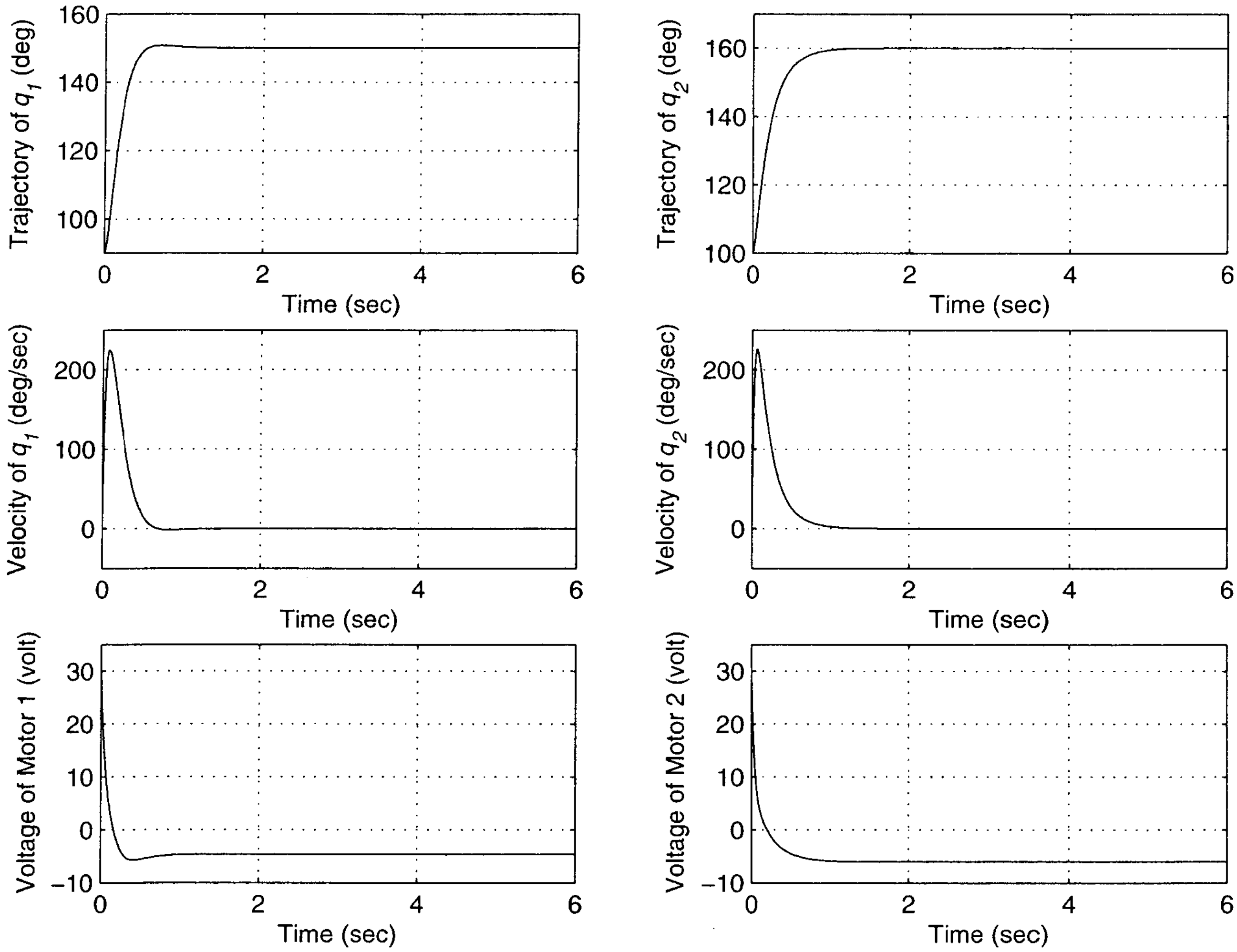

Figure 4-12: Point to Point Control, Simulation Results: PD Scheme, from $q_1 = 90^0$, $q_2 = 100^0$ to $q_1^d = 150^0$, $q_2^d = 160^0$

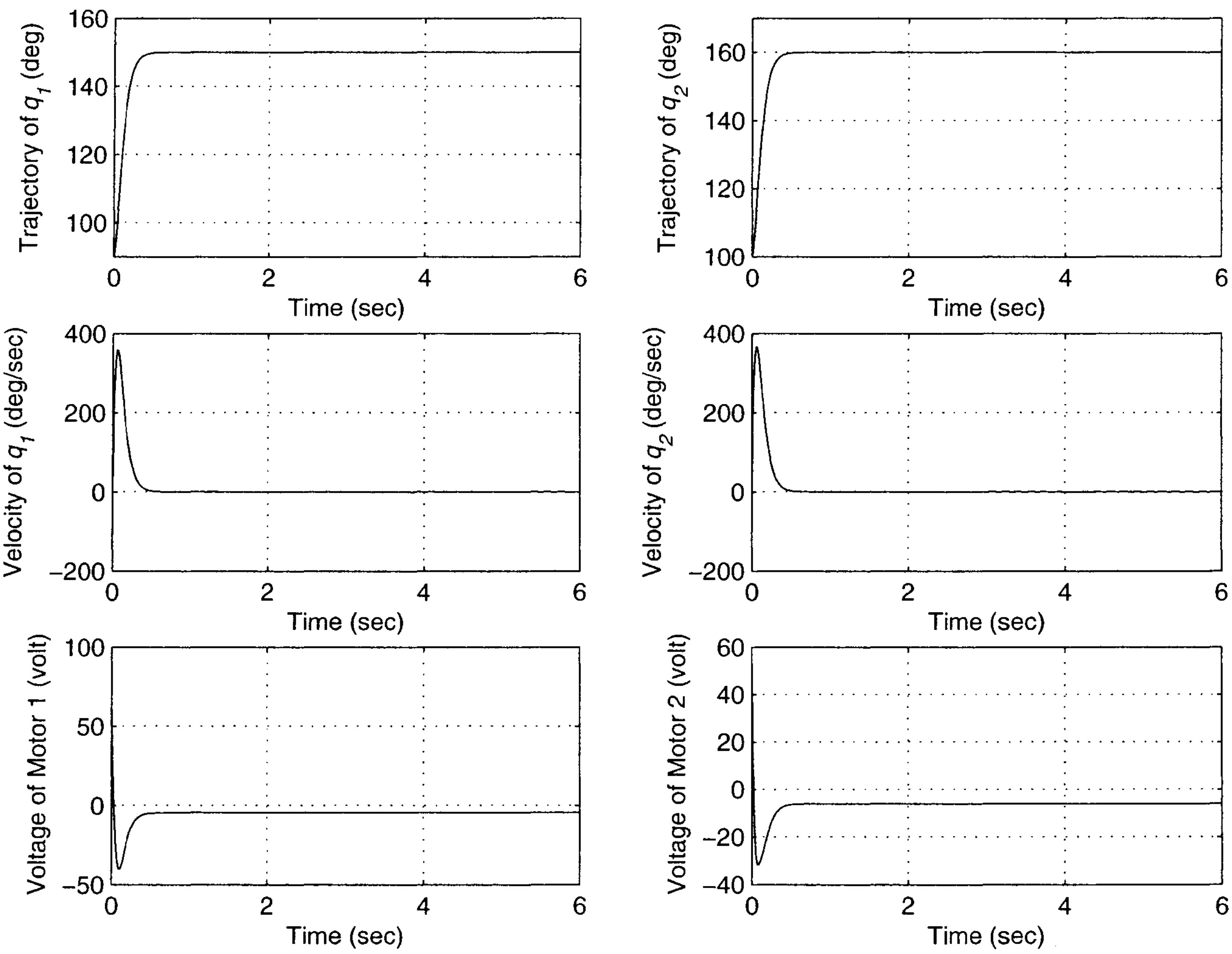

Figure 4-13: Point to Point Control, Simulation Results: Backstepping Scheme, from $q_1 = 90^0$, $q_2 = 100^0$ to $q_1^d = 150^0$, $q_2^d = 160^0$

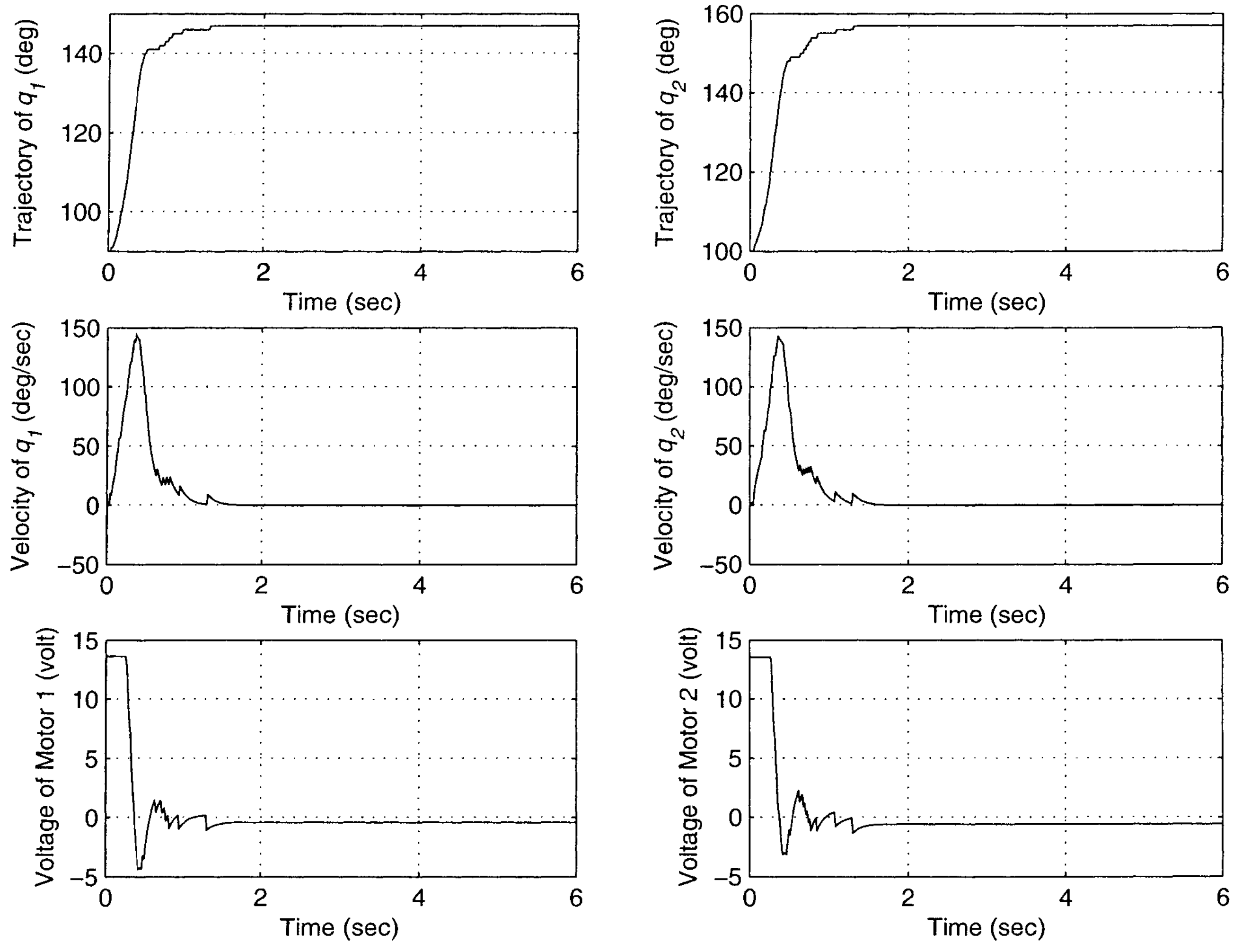

Figure 4-14: Point to Point Control, Experimental Results: PD Scheme, from $q_1 = 90^0$, $q_2 = 100^0$ to $q_1^d = 150^0$, $q_2^d = 160^0$

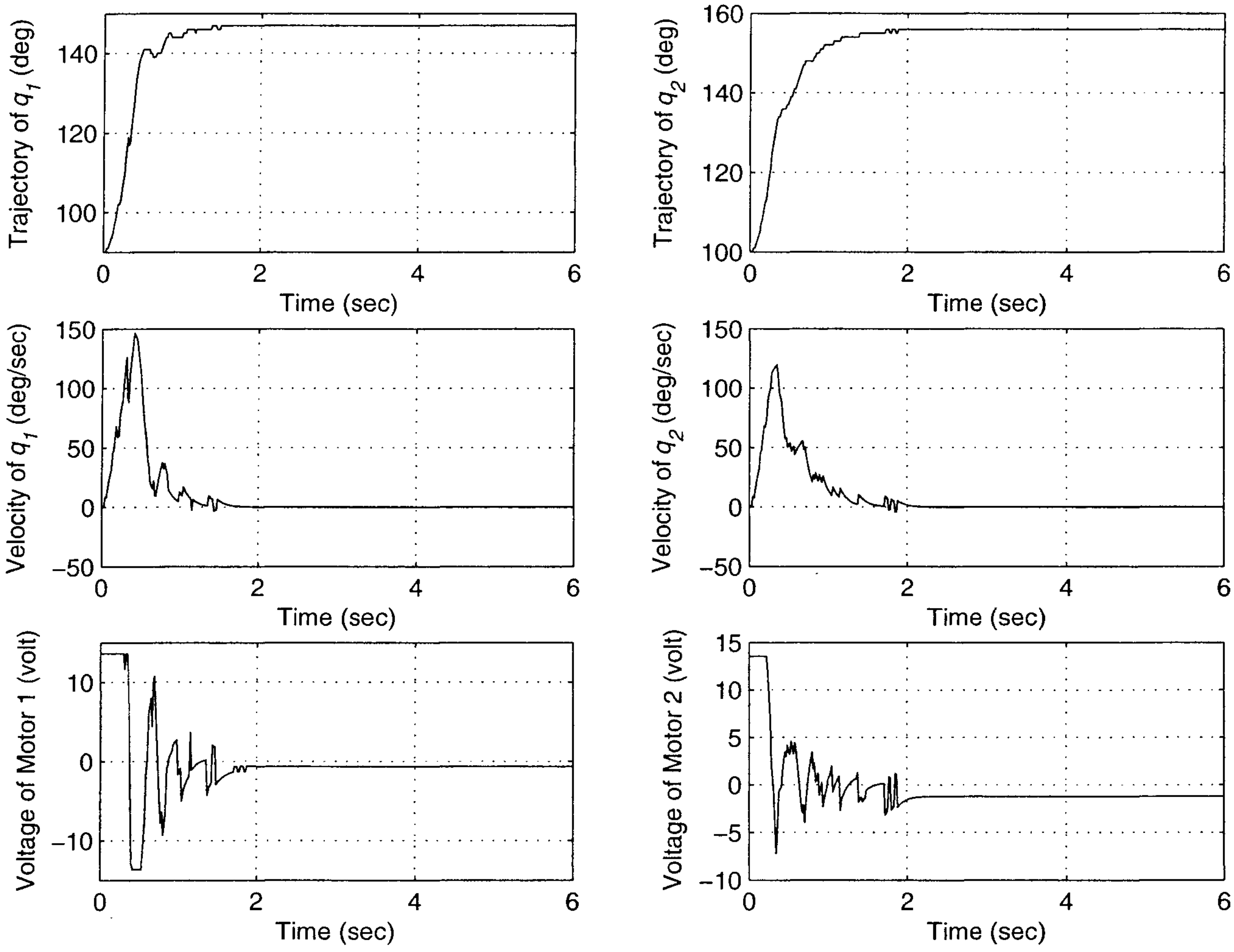

Figure 4-15: Point to Point Control, Experimental Results: Backstepping Scheme, from $q_1 = 90^0$, $q_2 = 100^0$ to $q_1^d = 150^0$, $q_2^d = 160^0$

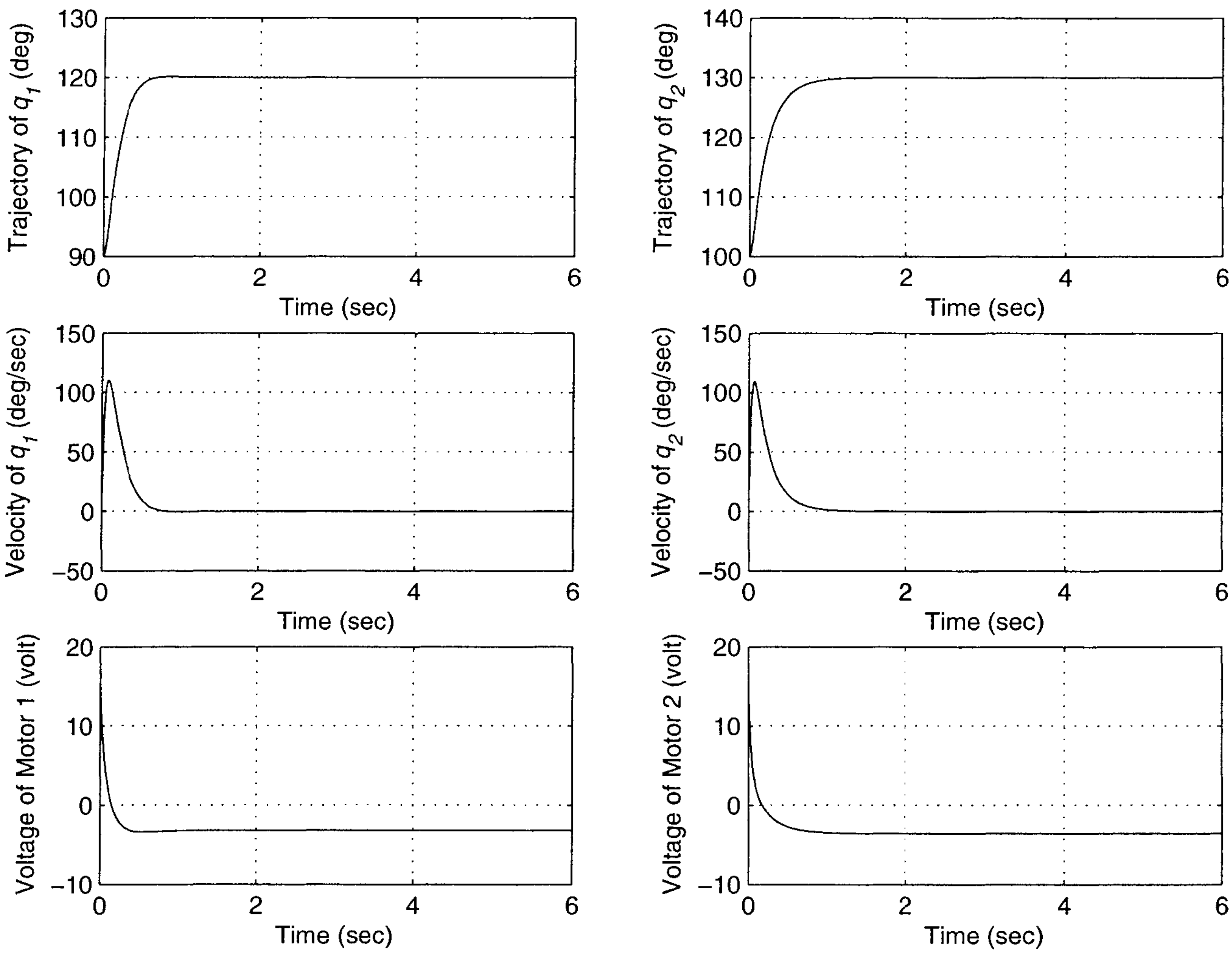

Figure 4-16: Point to Point Control, Simulation Results: PD Scheme, from $q_1 = 90^0$, $q_2 = 100^0$ to $q_1^d = 120^0$, $q_2^d = 130^0$

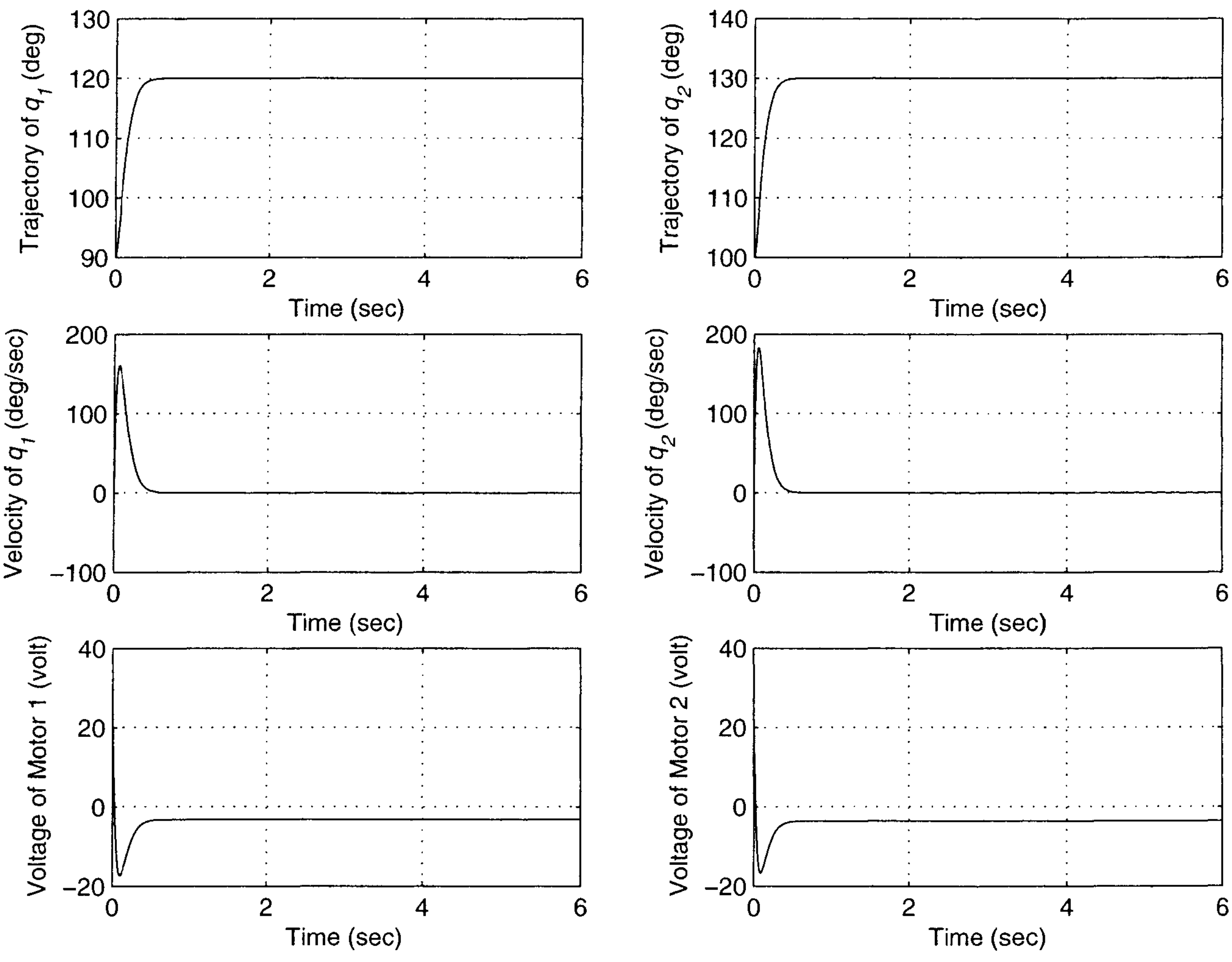

Figure 4-17: Point to Point Control, Simulation Results: Backstepping Scheme, from $q_1 = 90^0$, $q_2 = 100^0$ to $q_1^d = 120^0$, $q_2^d = 130^0$

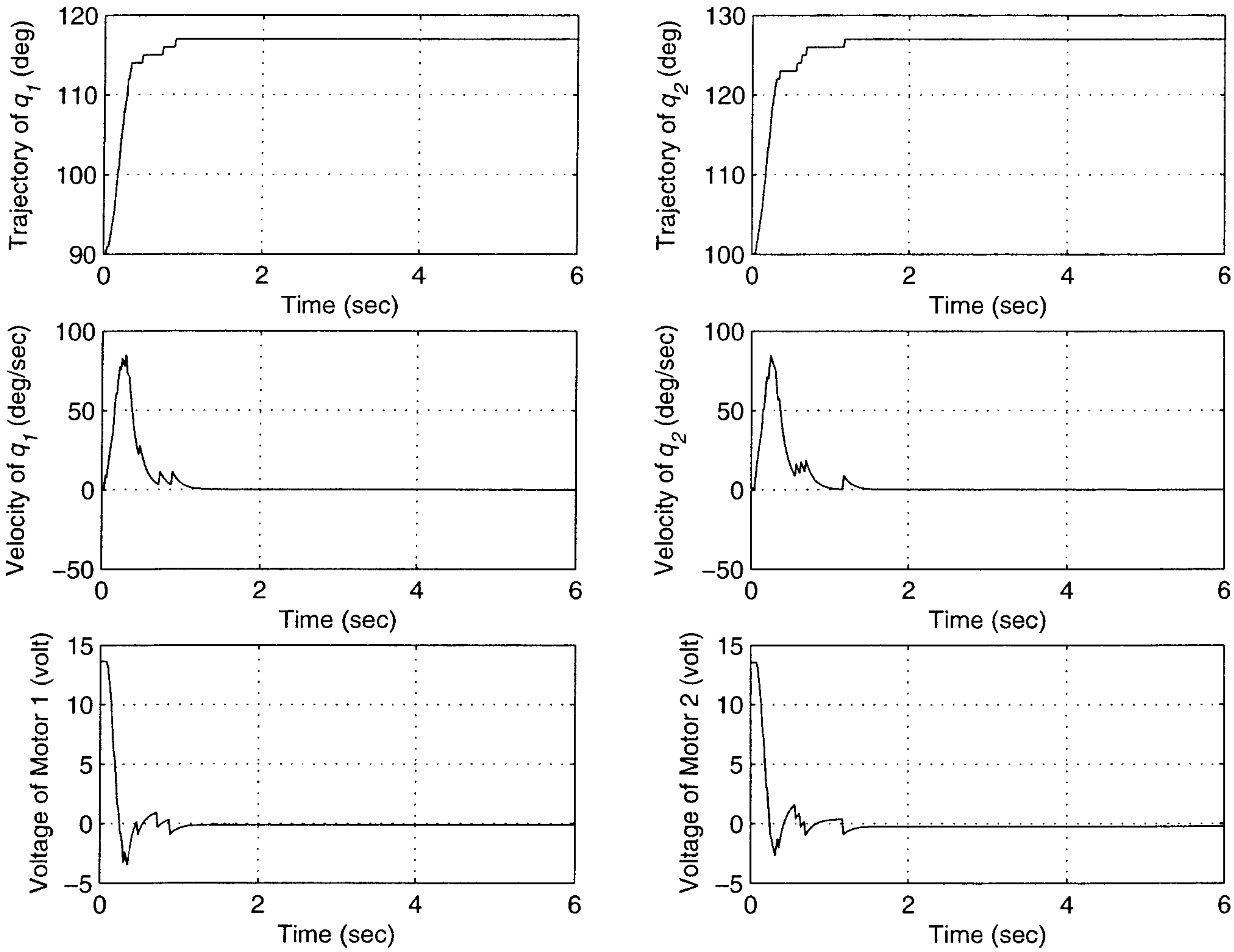

Figure 4-18: Point to Point Control, Experimental Results: PD Scheme, from $q_1 = 90^0$, $q_2 = 100^0$ to $q_1^d = 120^0$, $q_2^d = 130^0$

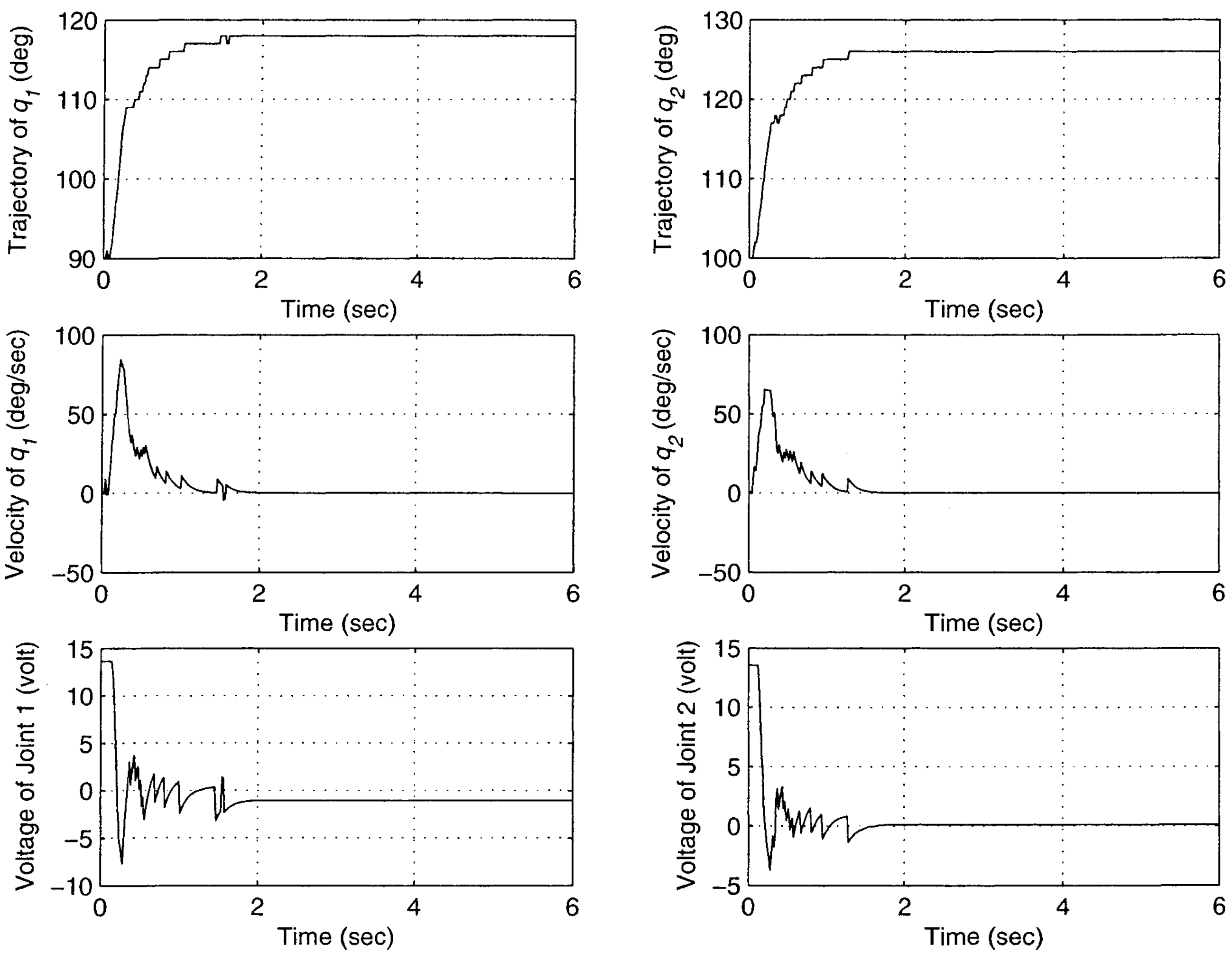

Figure 4-19: Point to Point Control, Experimental Results: Backstepping Scheme, from $q_1 = 90^0$, $q_2 = 100^0$ to $q_1^d = 120^0$, $q_2^d = 130^0$

system. We convert this DAE system into an equivalent ODE system so that the backstepping design method is applied. Both simulation and experimental results are performed for the backstepping control scheme. For comparison, the PD control scheme is also performed for both simulation and experiment. The experimental results for both backstepping and PD control scheme are reasonable and almost agree with the simulations results. More sophisticated model including friction and more accurate measurements of the system's parameters shall improve the performances. In the future work, we will take the adaptive control scheme on this parallel robotic system, by which the system parameters can be estimated by the controller instead of being measure physically.

Chapter 5

Conclusion

In this thesis, two theoretical topics are studied and an practical application is performed. The adaptive control for both MIMO nonlinearly parameterized systems with nested triangular structure and a class of nonlinear DAE system with unknown parameters are studied based on adaptive backstepping. The developed methodologies can guarantee the global stability of the corresponding systems. As a practical application, the nonlinear stabilizing controller designed by backstepping is implemented on a parallel robot. The simulation and experimental results are given and analyzed. The performances of the controlled system are satisfactory.

Besides the theoretical derivation, physical examples maybe need to be found to illustrate the developed methodology in Chapter 2 in the future work. In Chapter 3, we made several assumptions that seemed a little bit strict. The release of those assumptions is challenging. As for the set point control of the parallel robot, more sophisticated model including friction and more accurate measurement will be taken into account. Adaptive control scheme will also be taken to estimate the system parameters, which is expected to give better performances.

Acknowledgement

A journey is easier when you travel together. Interdependence is certainly more valuable than independence. This thesis is the result of two years of work whereby I have been accompanied and supported by many people. It is a pleasant aspect that I have now the opportunity to express my gratitude for all of them.

I am deeply indebted to my supervisor, Dr. Xiaoping Liu, whose help, stimulating suggestions and encouragement helped me in all the time of research during these two years. Not only in the academic research, has Dr. Liu also helped me a lot in my life, especially when I came to Canada two years ago, I couldn't adapt myself to a different life quickly without his help.

I am very grateful for my co-supervisor Dr. Abdelhamid Tayebi, who gave me two great lectures and helpful suggestions in my research. I am also very grateful for Dr. Kefu Liu, who helped me a lot and gave me valuable suggestions in the experiments, and Dr. Krishnamoorthy Natarajan for his review of this thesis and valuable suggestions.

I would like to thank for Manfred Klein, the technologist of Department of Electrical Engineering, who gave me a lot help for building the PCB circuits.

I also want to thank for all the graduate students of our program, who gave me a lot help in academics and life. I really enjoyed and shared the joyful two years' life with all the graduate students.

Bibliography

[1] C. Abdallah, D. Dawson, and M Jamshidi. Survey of robust control for rigid robots. *IEEE Control Systems Magazine*, 22:24–30, 1991.

[2] N.M. Ascher and L.R. Petzold. *Computer Methods for Ordinary Equations and Differential-Algebraic Equations.* SIAM Classics in Applied Mathematics Series, New York, 1998.

[3] K.E. Brenan, S.L. Campbell, and L.R. Petzold. *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations.* SIAM Classics in Applied Mathematics Series, New York, 1996.

[4] G.D. Byrne and P.R. Ponzi. Differential-algebraic systems, their applications and solutions. *Comput. Chem. Engng.*, 12:377–382, 1988.

[5] F.H. Ghorbel, O. Chetelat, R. Gunawardana, and Roland Longchamp. Modeling and set point control of closed-chain mechanisms: Theory and experiment. *IEEE Transaction on Control Systems Technology*, 8:801–815, 2000.

[6] P. B. Goldsmith, B. A. Francis, and A. A. Goldenberg. Stability of hybrid position/force control applied to manipulators with flexible joints. *International Journal of Robotics and Automation*, 14:146–160, 1999.

[7] D.T. Greenwood. *Classical Dynamics.* Prentice Hall, Englewood, NJ, 1995.

[8] D.J. Hill and I.M.Y. Mareels. Stability theory for differential-algebraic systems with applications to power systems. *IEEE Trans. Circuits and Systems*, 37:1461–1423, 1990.

[9] J. Huang and J.F. Zhang. Impulse-free output regulation of singular nonlinear systems. *Int. J. of Control*, 71:789–806, 1998.

[10] I. Kanellakopoulos, P. Kokotovic, and A. S Morse. Systematic design of adaptive controllers for feedback linearizable systems. *IEEE Transactions on Automatic Control*, 36:1241–1253, 1991.

[11] S. Kawaji and E.Z. Taha. Feedback linearization of a llass of nonlinear descriptor systems. In *Proceedings of the IEEE Conference on Decision and Control*, pages 4035–4042, Lake Buena Vista, FL, USA, 1994.

[12] A. Kojic and A.M. Annaswamy. Adaptive control of nonlinearly parameterized systems with a triangular structure. *Automatica*, 38:115–123, 2002.

[13] H Krishnan and N.H. McClamroch. Tracking in nonlinear differential-algebraic control systems with applications to contrained robot systems. *Automatica*, 30:885–897, 1994.

[14] M. Krstic, I. Kanellakopoulos, and P. Kokotovic. *Nonlinear and Adaptive Control Design*. Wiley, New York, 1995.

[15] A. Kumar and P. Daoutidis. *Control of nonlinear differential algebraic equations system with applications to chemical process*. CHAPMAN & HALL/CRC, New York, 1999.

[16] P. Kunkel and V. Mehrmann. Analysis of over- and underdetermined nonlinear differential-algebraic systems with application to nonlinear control problems. *Mathematics of Control, Signals, and Systems (MCSS)*, 14:233–256, 2001.

[17] W. Lin and C. Qian. Adaptive control of nonlinearly parameterized systems: A nonsmooth feedback framework. *IEEE Trans. Automatic Control*, 5:757–774, 2002.

[18] W. Lin and C Qian. Adaptive control of nonlinearly parameterized systems: the smooth feedback case. *IEEE Transactions on Automatic Control*, 47:1249–1266, 2002.

[19] W. Lin and C. Qian. Adaptive regulation of cascade systems with nonlinear parameterization. *Int. J. of Robust and Nonlinear Control*, 12:1093–1108, 2002.

[20] X.P. Liu. Robust stabilization of nonlinear singular systems. In *Proceedings of the IEEE Conference on Decision and Control*, pages 2375–2376, New Orleans, LA, USA, 1995.

[21] X.P. Liu. Asymptotic output tracking of nonlinear differential-algebraic control systems. *Automatica*, 34:393–397, 1998.

[22] X.P. Liu. Local disturbance decoupling of nonlinear differential-algebraic control systems. *Int. J of Control*, 70:685–702, 1998.

[23] X.P. Liu. Adaptive control of mimo nonlinear systems with nested triangular structure. In *Proceedings of the 5th World Congress on Intellgient Control and Automation*, pages 838–842, Beijing, China, 2004.

[24] X.P. Liu and C.S. Celikovsky. Feedback control of affine nonlinear singular control systems. *Int. Journal of Control*, 68:753–774, 1997.

[25] X.P. Liu, G.X. Gu, and K.M. Zhou. Robust stabilization of mimo nonlinear systems by backstepping. *Automatica*, 35:987–995, 1999.

[26] X.P. Liu and D. W. C. Ho. Stabilization of nonlinear differential-algebraic equation systems. *Int. Journal of Control*, 77:671–684, 2004.

[27] R. Marino and P. Tomei. Global adaptive output feedback control nonlinear systems, part ii: Nonlinear parameterization. *IEEE Transactions on Automatic Control*, 38:33–48, 1993.

[28] R. Marino and P. Tomei. *Nonlinear Control Design: Geometric, Adaptive and Robust*. Prentice Hall, UK, 1995.

[29] N.H. McClamroch. Singular system of differential equations as dynamic models for constrained robot systems. In *Proc. of the IEEE International Conf. on Robotics and Automation*, pages 21–28, San Francisco, CA, USA, 1986.

[30] N.H. McClamroch. Feedback stabilization of control systems described by a class of nonlinear differential-algebraic equations. *Systems& Control Letters*, 15(53-60), 1990.

[31] N.H. McClamroch and D.W. Wang. Feedback stabilization and tracking of constrained robots. *IEEE Trans. Auto. Control*, 33:419–426, 1998.

[32] R.W. Newcomb. The semistate description of nonlinear time-viarable circuits. *IEEE Tans. Circuits & Systems*, 28:61–71, 1981.

[33] R. Ortega and M.W. Spong. Apative motion control of rigid robots: A tutorial. In *Proceedings in the 27th Conference of Desicion Control*, pages 1575–1584, Austin, TX, USA, 1988.

[34] R. Sepulchre, M. Jankovic, and P. Kokotovic. *Constructive Nonlinear Control.* Springer, London, 1997.

[35] M.W. Spong. *Robot Dynamics and Control.* Wiley, New York, 1989.

[36] Qiao Sun. Control of flexible-link multiple manipulators. *Journal of Dynamic Systems, Measurement, and Control*, 124:67–75, 2002.

[37] L. Tian, J. Wang, and Z. Mao. Constrained motion control of flexible robot manipulators based on recurrent neural networks. *IEEE trans. Systems, Man, and CyberneticsqłPart B: Cybernetics*, 34:1541–1552, 2004.

[38] P. Tomei. A simple pd controller for robots with elastic joints. *IEEE trans. Auto. Contr.*, 36:1208–1213, 1991.

[39] D. W. Wang and Y. C. Soh. Control of constrained manipulator with flexible joints. *Dynamics and Control*, 6:33–48, 1996.

[40] K. Yokoi, N.V.Q. Huang, H.D. Tuan, and S. Hosoe. Adaptive control design for n-th order nonlinear multiplicative parameterized systems with triangular structure and application. In *Proceedings of the IEEE Conference on Decision and Control*, pages 3268–3273, Hawaii, USA, 2003.

[41] L.S. You and B.S. Chen. Tracking control designs for both holonomic and non-holonomic constrained menchanical systems: a unified viewpoint. *Int. Journal Control*, 58:587–612, 1993.

[42] G. Zimmer and J. Meier. On observing nonlinear descriptor systems. *Systems & Control Letters*, 32:43–48, 1997.