Comparative Experimental Study of Control Theoretic and Connectionist Controllers for Nonlinear Systems

by

He Huang

B. S., University of Science and Technology of China (1990)

S. M., Massachusetts Institute of Technology/
Woods Hole Oceanographic Institution (1994)

Submitted in partial fulfillment of the
requirements for the degree of

OCEAN ENGINEER

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

and the

WOODS HOLE OCEANOGRAPHIC INSTITUTION

September 1995

Signature of Author .....

Department of Ocean Engineering, MIT and the
MIT-WHOI Joint Program in Oceanographic Engineering

Certified by .....

Dr. Dana R. Yoerger
Associate Scientist, Woods Hole Oceanographic Institution
Thesis Supervisor

Accepted by .....

Professor Arthur B. Baggeroer
Chairman, Joint Committee for Oceanographic Engineering, Massachusetts
Institute of Technology and the Woods Hole Oceanographic Institution

Comparative Experimental Study of Control Theoretic and Connectionist Controllers for Nonlinear Systems

by

He Huang

Submitted to the Massachusetts Institute of Technology/
Woods Hole Oceanographic Institution
Joint Program in Oceanographic Engineering
in September, 1995 in partial fulfillment of the
requirements for the degree of

OCEAN ENGINEER

Abstract

This thesis compares classical nonlinear control theoretic techniques with recently developed neural network control methods based on the real time experimental control performance on a simple electro-mechanical system. The system has configuration-dependent inertia and velocity squared terms, which contribute a substantial nonlinearity. The controllers being studied include PID control, sliding control, adaptive control, and Gaussian network control.

Experimental results are given on the tracking performance and computation time for each controller. To establish a fair comparison among different controllers, the feedback bandwidth of each control method is chosen to be the same. The performance data for final comparison is collected using the same control frequency, i.e., the slowest pace of the four methods. These controllers are evaluated based on the amount of a priori knowledge required, tracking performance, stability guarantees, and computational requirements. The comparative study shows that for different control applications, appropriate control techniques should be selected depending on how much we know about the nonlinearity of the systems to be controlled and available computational resources.

Thesis Supervisor: Dr. Dana R. Yoerger

Associate Scientist
Woods Hole Oceanographic Institution

Acknowledgments

I greatly appreciate the efforts of my thesis advisor Dr. Dana Yoerger, who gave me the opportunity to study the interesting world of underwater robotics, control theory and neural networks. I thank Prof. Jean-Jacques Slotine for giving me insights in the Gaussian network control.

My thanks also go to everyone at the Deep Submergence Laboratory of the Woods Hole Oceanographic Institution, for all the help and support, especially from Dr. Louis Whitcomb, Ted Snow, Frank Weyer, Will Sellers, Tom Crook, Marty Marra, and Dr. Hanu Singh.

I would like to thank Cindy Sullivan, Larry Flick, Prof. Arthur Baggeroer and other MIT/WHOI Joint Program staff, who assisted me in many ways during my studies here.

Thank you, my friends in MIT Office 5-007, for all the helpful discussions.

I thank my dear parents for their love. I thank my husband Xiaoou Tang, my brother Shan, and my good friends Dan Li and Changle Fang. Without their support and encouragement, I would not have been able to complete this thesis.

Contents

List of Figures

Chapter 1

Introduction

1.1 Motivation

Underwater vehicles are important tools for exploration of the oceans. They are being applied to a wide range of tasks including hazardous waste clean-up, dam and bridge inspections, ocean bottom geological surveys, historical ship wreck documentation and salvages, and oceanographic researches. The Deep Submergence Lab at Woods Hole Oceanographic Institution has developed several underwater vehicles: ARGO, JASON, ABE, etc., and deployed them in dozens of ocean science expeditions [12] [24] [27].

Most of the current underwater vehicles are remotely operated through cables. Human pilots are heavily involved during the operations. Precise, repeatable computer control of the vehicles will significantly reduce the operator's workload and provide better performance. For the fast growing populations of autonomous underwater vehicles, automatic control will be the only choice. However, due to the nonlinearity and uncertainties introduced by hydrodynamic drag and effective mass properties, precise control of an underwater vehicle is very difficult to realize. Traditional well developed linear control techniques can only be applied to this highly nonlinear and uncertain scenario by compromising performance.

The same is true with the control of a manipulator on an underwater vehicle [28], which is different from its counterpart on the land vehicle. The hydrodynamic force

and other factors add nonlinearities into the dynamics [11], so choosing a good nonlinear control method on the vehicle and manipulator becomes necessary and important.

1.2 Overview of Traditional Control Theory and Neural Network Control Methods

Many control techniques have been used to control underwater vehicles and manipulators. There are traditional linear control methods for operations within the linearized region, sliding control and adaptive control methods for nonlinear operations, and most recently, the neural network approaches for nonlinear controls. This section summarizes these control techniques and their performance on underwater vehicles, and points out the importance to compare these different controllers and make better choices for our specific control applications.

1.2.1 Traditional Control Theory

Traditional linear control, which is a well developed control technique [13], performs poorly on nonlinear dynamic systems like underwater vehicles because the dynamics model must be linearized within a small range of operation and consequently loses its accuracy in representing the whole physical plant. The dynamics of the underwater vehicles needs to be understood and modeled thoroughly [9] [2] [23] [6] [7]. While good performance and stability can be reached [8] [10] within the linearized region, e.g., when the range of operation is small as in the case of constant speed heading control, they can not be achieved when the operation is out of the linearized region. Hard nonlinearities and model uncertainties often make linear control unable to perform well for nonlinear systems.

Nonlinear control methodologies are thus developed for better control of nonlinear dynamic systems. There is no general method for nonlinear control designs, instead, there is a rich collection of techniques each suitable for particular class of nonlinear control problems [18]. The most used techniques are feedback linearization, robust

control, adaptive control and gain-scheduling. The first and the last control techniques are more closely related to linear control methodology and their stability and robustness are not guaranteed. So in this thesis, I choose robust control and adaptive control for the purpose of study and comparison.

A simple approach to robust control is the sliding control methodology [18]. It provides a systematic approach to the problem of maintaining stability and consistent performance in the face of modeling imprecision. It quantifies trade-offs between modeling and performance, greatly simplifies the design process by accepting reduced information about the system. Sliding control has been successfully applied on the underwater vehicles [4] [3] [14] and other nonlinear dynamic systems. It eliminates lengthy system identification efforts and reduces the required tuning of the control system. The operational system can be made robust to unanticipated changes in the vehicle's dynamic parameters. However, the upper bounds on the nonlinearities and the unknown constant or slow varying parameters in the dynamic system have to be estimated for sliding control to be successful.

Adaptive control techniques [17] have also been successfully used to deal with the uncertain and nonlinear dynamics of underwater vehicles [29] [1]. It further provides an adaptation mechanism for the unknown parameters in the dynamic system, thus it achieves better performance if the initial parameter estimates are imprecise. It can be regarded as a control system with on-line parameter estimation. The form of the nonlinearities must be known, along with bounds on the parameter uncertainty. The unknown parameters it adapts to have to be constant or slowly-varying.

1.2.2 Neural Network Control Methods

Neural network control [15] [26] is a fast growing new candidate for nonlinear controls. These controllers have ability to learn the unknown dynamics of the controlled system. The parallel signal processing, computational inexpensive, and adaptive properties of the neural networks also make them appealing to the real time control of underwater vehicles. J. Yuh has developed a multi-layered neural network controller with the error estimated by a critic equation [32] [30] [31] . The only required information

about the system dynamics is an estimate of the inertia terms. K. P. Venugopal *et al.* described a direct neural network control scheme with the aid of a gain layer, which is proportional to the inverse of the system Jacobian [22].

While the back-propagation method used by the above researches is theoretically proven to be convergent, there is no theory regarding its stability when implemented into real time control problems. Moreover, there is no standard criteria to choose the number of layers and nodes within the networks.

A better solution to these problems is a network of Gaussian radial basis functions, called Gaussian networks [16]. It uses a network of Gaussain radial basis functions to adaptively compensate for the plant nonlinearities. The a priori information about the nonlinear function of the plant is its degree of smoothness. The weight adjustment is determined using Lyapunov theory, so the algorithm is proven to be globally stable, and the tracking errors converge to a neighborhood of zero. Its another feature is that the number of nodes within the network can be decided by the desired performance and the a priori frequency content information about the nonlinear function to be approximated.

The Gaussian network is a controller which combines theoretic control and connectionists approach. It uses the network to its full advantage of learning ability while in the mean time guarantees the stability of the whole system using traditional theoretical approach. The resulting controller is thus robust and retains its high performance.

1.2.3 Comparison of Traditional and Connnectionist Control

With the fast development of connectionist control methodology, there are growing interests in applying it to every nonlinear control application. Instead of embracing the new and promising control technique blindly, this thesis tries to evaluate it on solid theoretic base and real time control performance. By comparing with traditional control methods, discussing their strength and weakness, we can make a better choice

of controllers to our specific control need.

To establish an experiment comparison, the Gaussian neural network control is used to compare with the traditional theoretic control techniques. They are evaluated based on the amount of a priori knowledge required, tracking performance, stability guarantees, and computational requirements. A simple electro-mechanical system which has rich nonlinear dynamics is designed and built to take place of a real underwater vehicle to save the operational cost. Characteristics of these controllers are discussed based on the experimental results and suggestions for choosing appropriate control techniques for different nonlinear dynamic systems are given based on these comparison results.

To establish a fair comparison among different controllers, the feedback bandwidth of each control method is kept the same. The performance data for final comparison is collected using the same control frequency, i.e., the slowest pace of the four methods.

1.3 Outline of Thesis

The outline of the thesis is:

Chapter 2 describes details of the nonlinear electro-mechanical system design. The dynamics of the system is modeled using Lagrange equation.

Chapter 3 contains controller designs and experimental results of three control theoretic methods: PID control, sliding control, and adaptive control.

Chapter 4 presents the controller design and experimental results of the Gaussian network control on the nonlinear system.

Chapter 5 discusses the experimental results of the four controllers and compares their performances.

Chapter 6 summarizes and offers suggestions for choosing appropriate nonlinear controllers. Recommendations for future work are presented.

Chapter 2

A Nonlinear Dynamic System

In order to evaluate the performance of different controllers on nonlinear dynamic systems, an electro-mechanical system is set up to represent the nonlinear dynamics. The system has a configuration-dependent inertia, and a configuration-dependent velocity squared term, both contribute substantial nonlinearity to the system dynamics, thus provide an excellent base for comparing the tracking performance of control theoretic and connectionist controllers.

The experimental setup of the electro-mechanical system is detailed in Section 2.1. Then the dynamics of the nonlinear system is modeled using Lagrange equation in Section 2.2. Based on the design and model of the system, a desired tracking trajectory is selected in Section 2.3 so that different controllers will be used to control the system following the same tracking motion.

2.1 Experimental Setup for a Nonlinear Electro-mechanical System

Figure 2-1 illustrates the schematic of the experimental setup of a nonlinear electro-mechanical system. The linkage-weight mechanism is designed to provide complex nonlinearities so that different control methods can be compared from their tracking performance on this system. The dynamics of the mechanism is described in

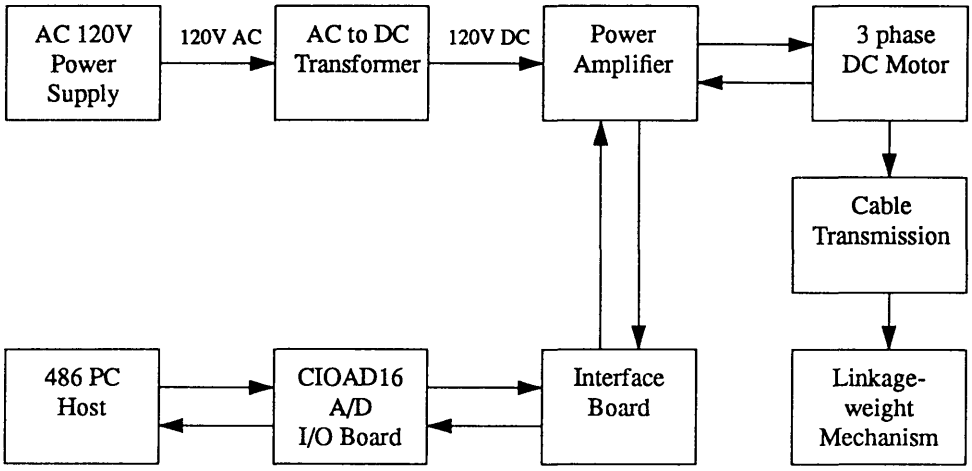

Figure 2-1: Configuration of the electro-mechanical control system

Section 2.2.

A 3–phase brushless DC motor designed for the control of the thrusters [25] on the 6000 meter JASON underwater vehicle is employed to control the movement of the linkage-weight mechanism. Since the motor operates in higher speeds than the mechanism, a cable drive [19] is used to reduce the speed of the motor by a factor of 6. The motor incorporates MOOG model 304-140A frameless windings and magnets and resolver feedback in a custom oil-compensated housing.

An ELMO model EBAF15/160 3-phase commutating 20KHz PWM current amplifier is used to drive the DC motor. Under the inductive load of the motor, the current amplifier was configured to track current commands with a 2ms time constant. The current limit was set at 9 Amps at a supply voltage of 120V for a maximum power of about 1KW. The resolver feedback from the motor carries the rotor angle information and is converted to quadrature by an AD2S82 on-board the EBAF15/160 to a resolution of 4096 counts per shaft mechanical revolution.

Using current amplifiers reduces the need to model motor electrical parameters such as winding resistance, winding inductance, and back-emf. Together with the high-resolution resolver measuring the shaft position, this experiment setup is perfect for closed-loop tracking control.

The host computer is a 486 class PC equipped with a quadrature interface, 12 bit Analog I/O, and a 100 KHz hardware clock. The amplifier, sensors, and host computer are extensively shielded and opto-isolated to minimize electromagnetic interference. The four kinds of control algorithms are written in C language and compiled using Quick C. The controllers are executed at 250Hz and the data is sampled and logged at 50Hz. Torque command and shaft position data are plotted *unfiltered* using Matlab — no numerical smoothing or post-processing has been employed for these signals. Angular velocity is obtained by numerically differentiating the raw angle position data.

2.2 Nonlinear Model of the Dynamic System

The linkage-weight mechanism is shown in Figure 2-2. A weight cart is attached to a linkage which connects with a crank on the output shaft of the cable transmission. The left side of the cable transmission is the input shaft connected through a Helical coupling with the DC brushless motor shaft.

The parameters for the mechanism are chosen to maximize the nonlinearity of the system. Since the mechanism is connected with the motor through the speed reducer, the weight M put on the cart has to be heavy enough to offset the constant motor inertia J magnified from its original value by a factor of 36. The parameters in the system are thus designed to be:

$$J = .03\ kg \cdot m^2;\quad M = 12.27\ kg;\quad R = .0587\ m;\quad L = .1714\ m;\quad E = .0619\ m \tag{2.1}$$

The equation of motion for this nonlinear dynamic system can be derived by using the Lagrange's equation:

$$\mathcal{L} = \mathcal{T} - \mathcal{U} = \text{Lagrangian} = \text{Kinetic Energy} - \text{Potential Energy} \tag{2.2}$$

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\theta}}\right) - \frac{\partial \mathcal{L}}{\partial \theta} = \sum \mathcal{F} \tag{2.3}$$

where θ is a generalized coordinate which uniquely specifies the location of the object

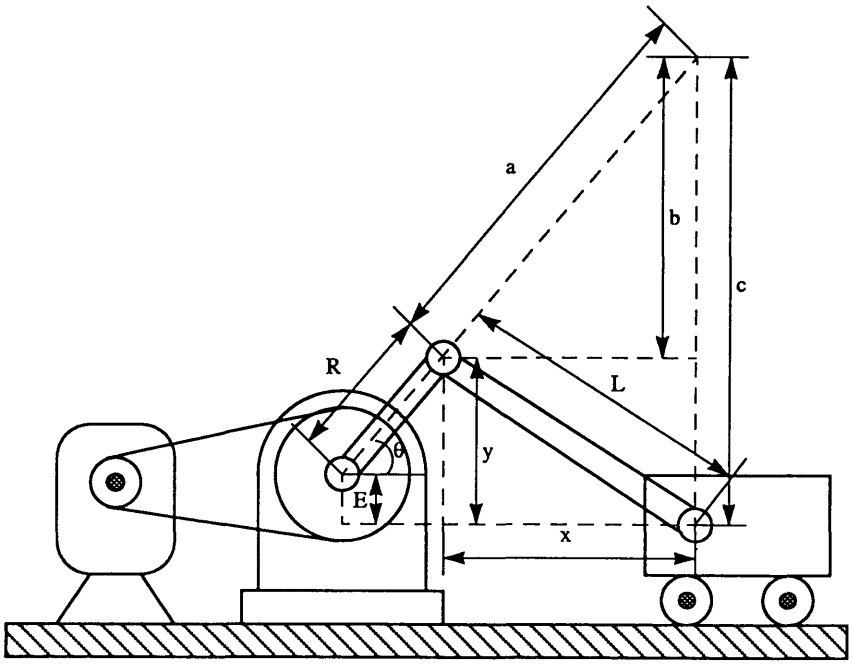

Figure 2-2: Structure of the nonlinear dynamic mechanical system

whose motion is being analyzed. In our case, θ is the angle of rotation of the crank. $\mathcal{F}$ is the nonconservative generalized force corresponding to the generalized coordinate θ. In our case, it's the motor torque u applied to the crank through the transmission.

The kinetic energy and potential energy of the nonlinear system are

$$\begin{aligned} \mathcal{T} &= \frac{1}{2}J\dot{\theta}^2 + \frac{1}{2}MV_M^2 \\ \mathcal{U} &= 0 \end{aligned}$$

The relation between the cart velocity V_M and θ is derived from the following equations.

$$\begin{aligned} y &= R\sin\theta + E \\ x &= \sqrt{L^2 - y^2} \\ a &= x/\cos\theta \end{aligned}$$

$$
\begin{aligned}
b &= x\tan\theta \\
c &= y + b \\
V_M &= \frac{R\dot{\theta}}{a}c
\end{aligned}
$$

so we have

$$
\begin{aligned}
\mathcal{L} &= \frac{1}{2}J\dot{\theta}^2 + \frac{1}{2}MV_M^2 \\
&= \frac{1}{2}J\dot{\theta}^2 + \frac{1}{2}M\frac{R^2c^2}{a^2}\dot{\theta}^2
\end{aligned}
$$

which leads to

$$
\frac{\partial \mathcal{L}}{\partial \dot{\theta}} = J\dot{\theta} + M\frac{R^2c^2}{a^2}\dot{\theta} \tag{2.4}
$$

$$
\begin{aligned}
\frac{d}{dt}(\frac{\partial \mathcal{L}}{\partial \dot{\theta}}) &= (J + M\frac{R^2c^2}{a^2})\ddot{\theta} + M\dot{\theta}^2\frac{d}{d\theta}(\frac{R^2c^2}{a^2}) \\
\frac{\partial \mathcal{L}}{\partial \theta} &= \frac{1}{2}M\dot{\theta}^2\frac{d}{d\theta}(\frac{R^2c^2}{a^2})
\end{aligned}
$$

Substituting the above expressions into the Lagrangian equation (2.3) yields

$$
(J + M\frac{R^2c^2}{a^2})\ddot{\theta} + \frac{1}{2}M\dot{\theta}^2\frac{d}{d\theta}(\frac{R^2c^2}{a^2}) = u \tag{2.5}
$$

Let's define

$$
\begin{aligned}
J_M(\theta) &= M\frac{R^2c^2}{a^2} \\
C(\theta) &= \frac{1}{2}M\frac{d}{d\theta}(\frac{R^2c^2}{a^2})
\end{aligned}
$$

then the equation of the nonlinear dynamics can be expressed as

$$
(J + J_M(\theta))\ddot{\theta} + C(\theta)\dot{\theta}^2 = u \tag{2.6}
$$

The definition shows both $J_M(\theta)$ and $C(\theta)$ are complex trigonometric functions of the rotating angle θ, as illustrated by Figure 2-3 and Figure 2-4. Figure 2-3 clearly

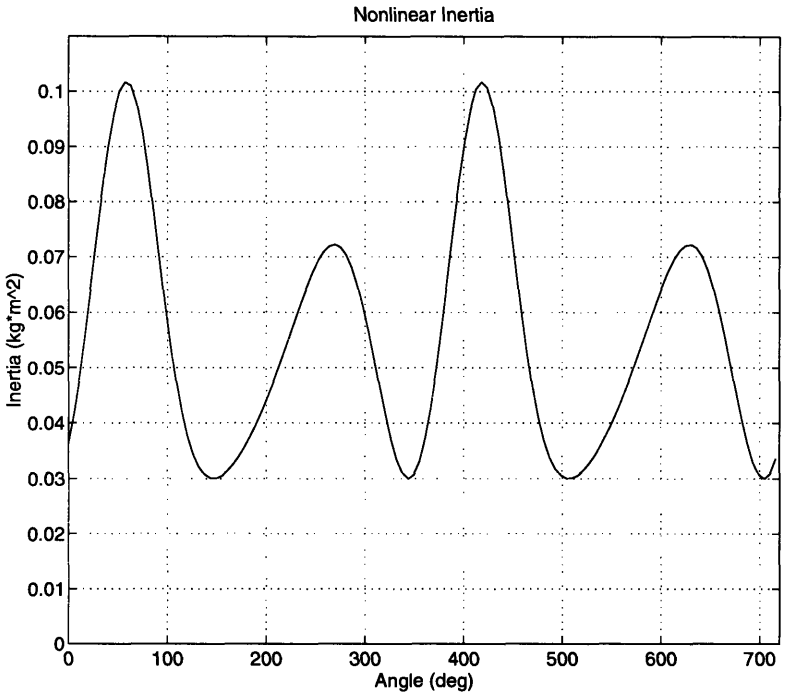

Figure 2-3: Nonlinear inertia which varies about 3 times the constant inertia

shows that the chosen weight on the cart is heavy enough to create a large nonlinear moment of inertia about 3 times the constant value.

2.3 Reference Trajectory

The different controllers are applied to the nonlinear dynamic system (2.6) and their results are compared in Chapter 3 and Chapter 4. To establish a fair comparison, all the controllers are designed to track the same trajectory shown in Figure 2-5, where the speed profile is a half sinusoidal wave of 5 second duration and the crank rotates a little over 2 turns.

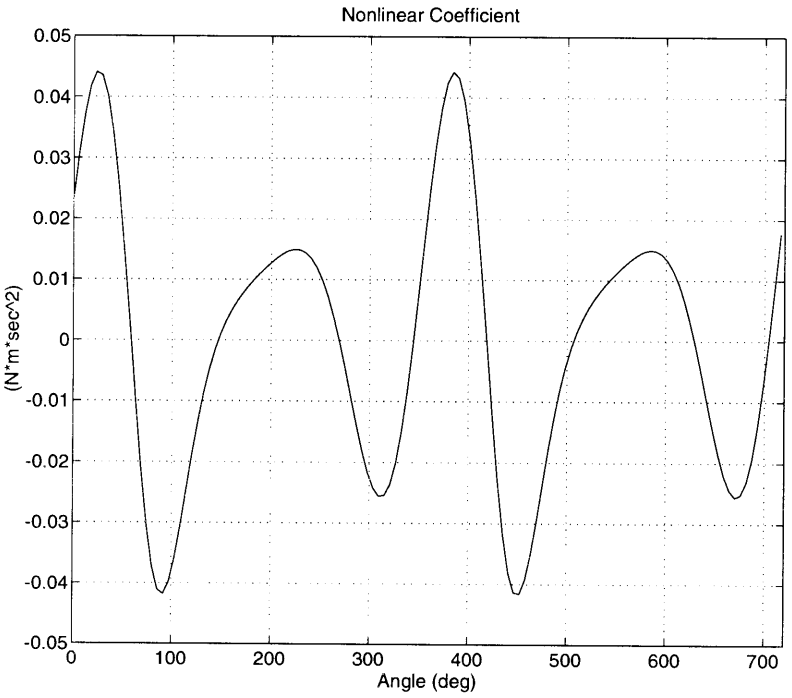

Figure 2-4: Nonlinear coefficient of the velocity squared term

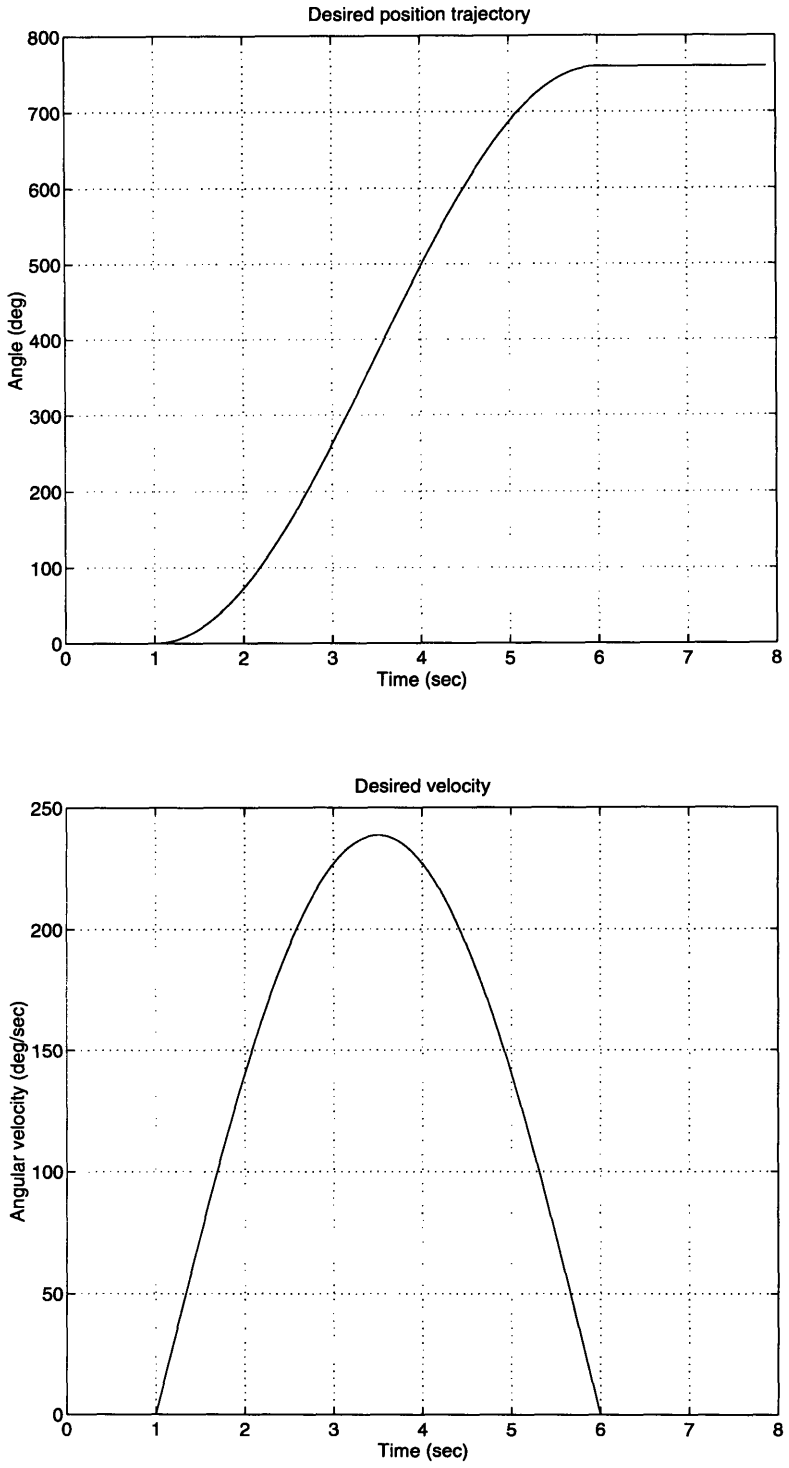

Figure 2-5: Desired trajectory and speed profile of the motion

Chapter 3

Control Theoretic Controllers for Nonlinear Systems

There is no general method for designing nonlinear controllers, instead, there is a rich collection of nonlinear control techniques, each dealing with different classes of nonlinear problems. The techniques studied in this thesis comprised of two groups: one of traditional theoretic nonlinear control techniques, the other of recently developed neural network control approaches. The first group applies to systems with known or partially known dynamic structure, but unknown constant or slowly-varying parameters, and can deal with model uncertainties. The second group is for systems without much a priori information about their dynamic structures and parameters. This chapter focuses on the first group, i.e., the control theoretic controllers for nonlinear systems.

The general procedure for theoretic controller design is presented in Section 3.1. Then three different theoretic controllers are designed and applied to the nonlinear dynamic system detailed in Chapter 2. The experimental results and plots are presented after the description of each controller design. The PID controller is discussed in Section 3.2, the sliding controller in Section 3.3, and the adaptive controller in Section 3.4.

3.1 Procedure for General Control Design

The objective of control design can be stated as follows: given a physical system to be controlled and the specifications of its desired behavior, construct a feedback control law to make the closed-loop system display the desired behavior. The procedure of constructing the control goes through the following steps, possibly with a few iterations:

1. specify the desired behavior, and select actuators and sensors;
2. model the physical plant by a set of differential equations;
3. design a control law for the system;
4. analyze and simulate the resulting control system;
5. implement the control system in hardware.

Generally, the tasks of control systems, i.e. the desired behavior, can be divided into two categories: stabilization and tracking. The focus of this thesis is on tracking problems. The design objective in tracking control problems is to construct a controller, called a tracker, so that the system output tracks a given time-varying trajectory.

The task of asymptotic tracking can be defined as follows:

Asymptotic Tracking Problem: Given a nonlinear dynamics system described by

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, t) \tag{3.1}$$

$$\mathbf{y} = \mathbf{h}(\mathbf{x}) \tag{3.2}$$

and a desired output trajectory $\mathbf{y}_d$, find a control law for the input $\mathbf{u}$ such that, starting from any initial state in a region Ω, the tracking errors $\mathbf{y}(t) - \mathbf{y}_d(t)$ go to zero, while the whole state $\mathbf{x}$ remains bounded.

This chapter focuses on different theoretic control law designs. The desired behavior and dynamic model of the physical plant are assumed to be known and are the same for all different control techniques.

Let the dynamic model of the nonlinear system represented by:

$$x^{(n)}(t) = f(\mathbf{X};t) + b(\mathbf{X};t)u(t) + d(t) \tag{3.3}$$

where

$u(t)$ is the control input

x is the output of interest

$\mathbf{X} = [\ x \quad \dot{x} \quad \cdots \quad x^{(n-1)}\ ]^T$ is the state.

$d(t)$ is the disturbance

$f(\mathbf{X};t)$ is the nonlinear function describing the system's dynamics

$b(\mathbf{X};t)$ is the control gain

The desired behavior, i.e., the control problem is to get the state $\mathbf{X}$ to track a specific state

$$\mathbf{X}_d = [\ x_d \quad \dot{x_d} \quad \cdots \quad x_d^{(n-1)}\ ]^T \tag{3.4}$$

in the presence of model imprecision on $f(\mathbf{X};t)$ and $b(\mathbf{X};t)$, and of disturbances $d(t)$.

If we define the tracking error vector as:

$$\tilde{\mathbf{X}} = \mathbf{X} - \mathbf{X}_d = [\ \tilde{x} \quad \dot{\tilde{x}} \quad \cdots \quad \tilde{x}^{(n-1)}\ ]^T \tag{3.5}$$

the control problem of tracking $\mathbf{X} \equiv \mathbf{X}_d$ is equivalent to reaching $\tilde{\mathbf{X}} \equiv 0$ in finite time.

3.2 PID Control

3.2.1 PID Control Theory

The combination of proportional control, integral control, and derivative control is called proportional-plus-integral-plus-derivative control, also called PID control. This combined control has the advantages of each of the three individual control actions.

The equation of a PID controller is given by

$$u(t) = K_p\tilde{x}(t) + K_d\dot{\tilde{x}}(t) + K_i\int \tilde{x}(t)dt \tag{3.6}$$

where K_p, K_d and K_i represents the proportional, derivative, and integral gains.

In the proportional control of a plant whose transfer function does not possess a free integrator, there is a steady-state error, or offset, in the response to steady disturbance. Such an offset can be eliminated if the integral control action is included in the controller. On the other hand, while removing offset or steady-state error, the integral control action may lead to oscillatory response of slowly decreasing amplitude or even increasing amplitude, both of which are usually undesirable.

Derivative control action, when added to a proportional controller, provides a means of obtaining a controller with high sensitivity. It responds to the rate of change of the actuating error and can produce a significant correction before the magnitude of the actuating error becomes too large. Derivative control thus anticipates the actuating error, initiates an early corrective action, and tends to increase the stability of the system.

Although derivative control does not affect the steady-state error directly, it adds damping to the system and thus permits the use of a larger value of the gain K, which will result in an improvement in the steady-state accuracy. Since derivative control operates on the rate of change of the actuating error and not the actuating error itself, this mode is never used alone. It is always used in combination with proportional or proportional-plus-integral action.

The selection of PID parameters K_p, K_d and K_i is based on the knowledge about the dynamic systems and the desired closed-loop bandwidth.

3.2.2 PID Controller Design

The control gains of the PID controller are selected according to the bandwidth λ, time constant τ and damping constant ζ of the nonlinear system.

$$K_i = J\lambda^2/\tau \tag{3.7}$$

$$K_p = J(\lambda^2\tau + 2\zeta\lambda)/\tau \tag{3.8}$$

$$K_d = J(1 + 2\lambda\zeta\tau)/\tau \tag{3.9}$$

with

$$\tau = 1.0/(\lambda\tau_f) \tag{3.10}$$

where τ_f is the time constant factor.

For the experiment, we assume the prior knowledge of the inertia term $J = 0.03$. The constant values are chosen as $\tau_f = 0.1$, $\lambda = 10$, and $\zeta = 0.707$.

Thus the control law for the PID controller is

$$u = K_i \int \tilde{\theta} dt + K_p\tilde{\theta} + K_d\dot{\tilde{\theta}} \tag{3.11}$$

The PID controller is applied to the electro-mechanical system described in Chapter 2 to follow the reference trajectory in Figure 2-5. The performance is shown from Figure 3-1 to Figure 3-2.

The results show that with the prior knowledge of the system parameters, the PID controller can achieve the tracking control, although with some large tracking errors.

3.3 Sliding Control

Given the perfect measurement of a linear system's dynamic state and a perfect model, the PID controller can achieve perfect performance. But it may quickly fail in the presence of model uncertainty, measurement noise, computational delays and disturbances. Analysis of the effects of these non-idealities are further complicated by nonlinear dynamics. The issue becomes one of ensuring a nonlinear dynamic system

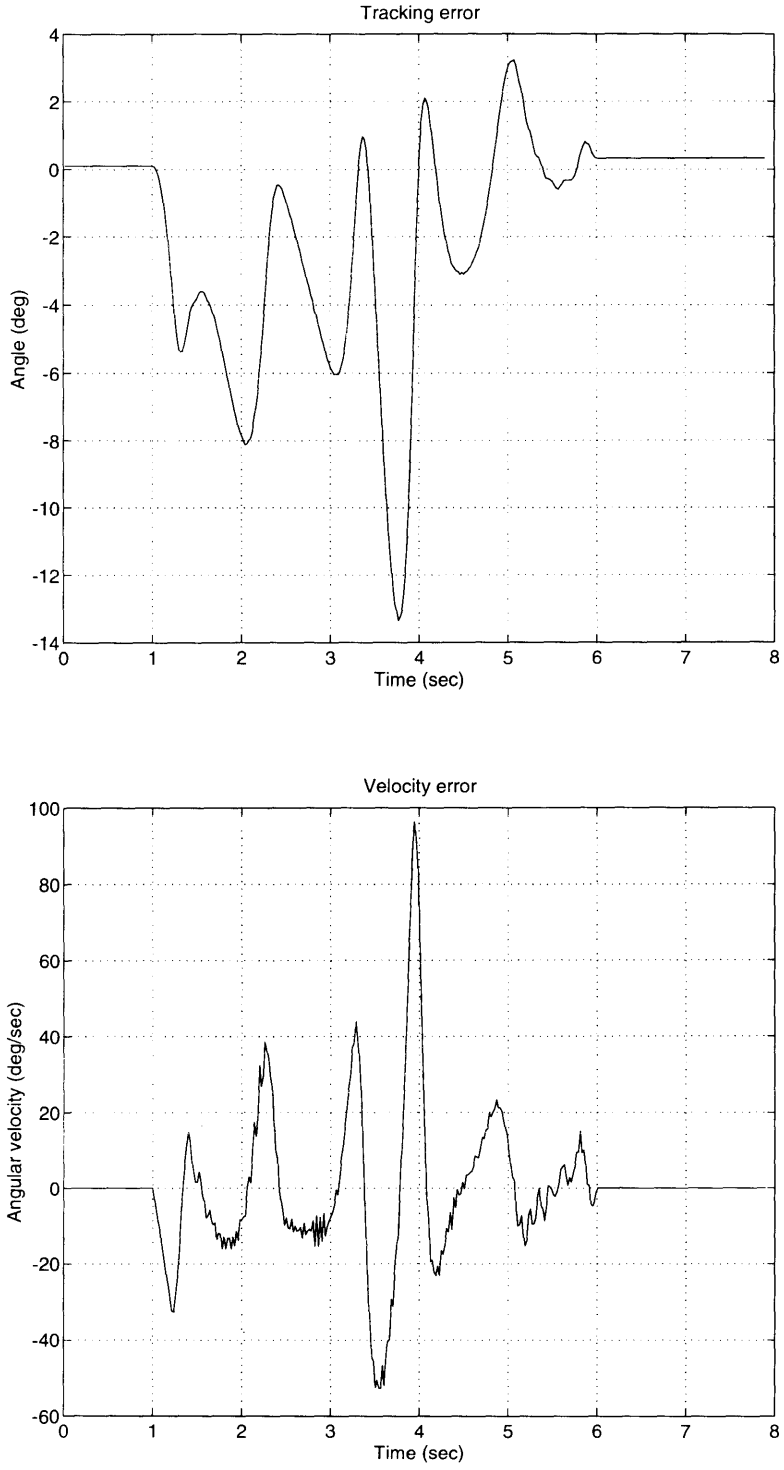

Figure 3-1: Tracking error and velocity error for PID control

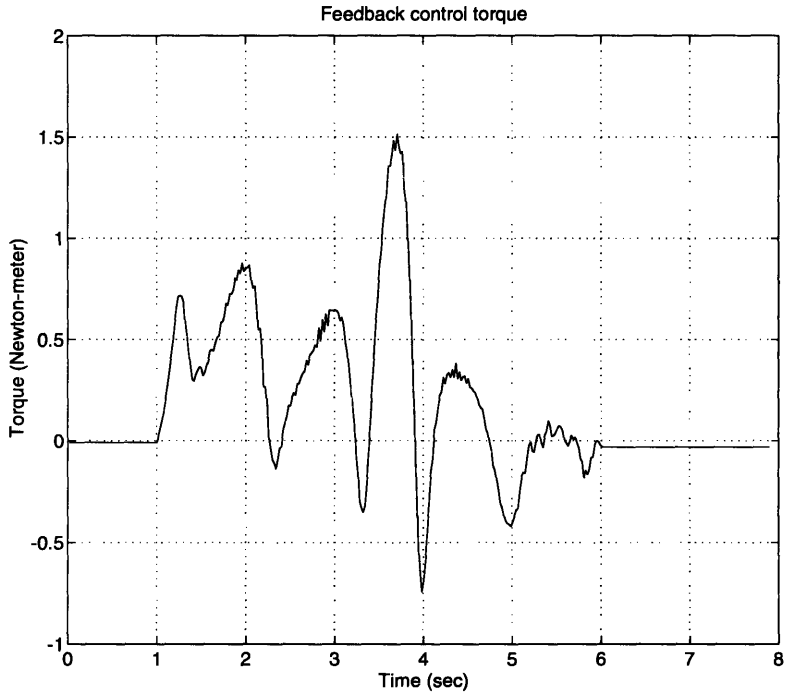

Figure 3-2: Control torque for PID control

remains robust to non-idealities while minimizing tracking error.

Two major and complementary approaches to dealing with model uncertainty are *robust control* and *adaptive control.* Sliding control methodology is a method of robust control. It provides a systematic approach to the problem of maintaining stability and consistent performance in the face of modeling imprecision.

Sliding Modes are defined as a special kind of motion in the phase space of a dynamic system along a sliding surface for which the control action has discontinuities. This special motion will exist if the state trajectories in the vicinity of the control discontinuity are directed toward the sliding surface. If a sliding mode is properly introduced in a system's dynamics through active control, system behavior will be governed by the selected dynamics on the sliding surface, despite disturbances, nonlinearities, time-variant behavior and modeling uncertainties.

3.3.1 Sliding Control Theory

Let's consider the dynamic system described by equation (3.3). A time-varying sliding surface $S(t)$ in the state-space R^n is defined as

$$s(\mathbf{X};t) = 0 \tag{3.12}$$

with

$$s(\mathbf{X};t) = (\frac{d}{dt} + \lambda)^{n-1}\tilde{x}, \quad \lambda > 0 \tag{3.13}$$

where λ is a positive constant related to the desired control bandwidth.

The tracking problem is now transformed to remaining the system state on the sliding surface $S(t)$ for all $t > 0$. This can be seen by considering $s \equiv 0$ as a linear differential equation whose unique solution is $\tilde{x} \equiv 0$, given initial condition:

$$\tilde{\mathbf{X}}|_{t=0} = 0 \tag{3.14}$$

This positive invariance of $S(t)$ can be reached by choosing the control law u of system (3.3) such that outside $S(t)$

$$\frac{1}{2}\frac{d}{dt}s^2(x;t) \leq -\eta|s| \tag{3.15}$$

where η is a positive constant. (3.15) is called the sliding condition. It constrains the trajectories of the system to point towards the surface $S(t)$.

The idea behind (3.13) and (3.15) is to pick a well behaved function of the tracking error, s, according to (3.13) and then select the feedback control law u such that s^2 remains a Lyapunov function of the closed-loop system despite the presence of model imprecision and disturbances. This guarantees the robustness and stability of the closed-loop system. Even when the initial condition (3.14) is not met, the surface $S(t)$ will still be reached in a finite time smaller than $s(\mathbf{X}(0);0)/\eta$, given the sliding condition (3.15) is verified.

The detailed controller design procedure is described in the following two sec-

tions. Section 3.3.1 shows how to select a feedback control law u to verify sliding condition (3.15) and account for the modeling imprecision. This control law leads to control chattering. Section 3.3.1 describes how to eliminate the chattering to achieve an optimal trade-off between control bandwidth and tracking precision.

Perfect Tracking Using Switched Control Laws

This section illustrates how to construct a control law to verify sliding condition (3.15) given bounds on uncertainties on $f(\mathbf{X};t)$ and $b(\mathbf{X};t)$.

Consider a second-order dynamic system

$$\ddot{x} = f + bu \tag{3.16}$$

The nonlinear dynamics f is not known exactly, but estimated as $\hat{f}$. The estimation error on f is assumed to be bounded by some known function F:

$$|\hat{f} - f| \leq F \tag{3.17}$$

Similarly, the control gain b is unknown but of known bounds:

$$0 < b_{min} \leq b \leq b_{max} \tag{3.18}$$

The estimate $\hat{b}$ of gain b is the geometric mean of the above bounds:

$$\hat{b} = \sqrt{b_{min} b_{max}} \tag{3.19}$$

Bounds (3.18) can then be written in the form

$$\beta^{-1} \leq \frac{\hat{b}}{b} \leq \beta \tag{3.20}$$

where

$$\beta = \sqrt{b_{max}/b_{min}} \tag{3.21}$$

In order to have the system track $x(t) \equiv x_d(t)$, we define a sliding surface $s = 0$ according to (3.13), namely:

$$s = (\frac{d}{dt} + \lambda)\tilde{x} \tag{3.22}$$

We then have:

$$\dot{s} = \ddot{x} - \ddot{x}_d + \lambda\dot{\tilde{x}} = f + bu - \ddot{x}_d + \lambda\dot{\tilde{x}} \tag{3.23}$$

The best approximation $\hat{u}$ of a continuous control law that would achieve $\dot{s} = 0$ is thus:

$$\hat{u} = \hat{bu} = -\hat{f} + \ddot{x}_d - \lambda\dot{\tilde{x}} \tag{3.24}$$

In order to satisfy the sliding condition (3.15) despite uncertainties on the dynamics f and the control gain b, we add to $\hat{u}$ a term discontinuous across the surface $s = 0$:

$$\begin{aligned} u &= \hat{b}^{-1}[\hat{u} - k\text{sgn}(s)] && (3.25) \\ &= \hat{b}^{-1}[-\hat{f} + \ddot{x}_d - \lambda\dot{\tilde{x}} - k\text{sgn}(s)] && (3.26) \end{aligned}$$

By choosing k in (3.25) to be large enough,

$$k \geq \beta(F + \eta) + (\beta - 1)|\hat{u}| \tag{3.27}$$

we can guarantee that (3.15) is verified. Indeed, we have from (3.23) to (3.26)

$$\dot{s} = (f - b\hat{b}^{-1}\hat{f}) + (1 - b\hat{b}^{-1})(-\ddot{x}_d + \lambda\dot{\tilde{x}}) - b\hat{b}^{-1}k\text{sgn}(s) \tag{3.28}$$

In order to let

$$\begin{aligned} \frac{1}{2}\frac{d}{dt}s^2 &= \dot{s}s \\ &= [(f - b\hat{b}^{-1}\hat{f}) + (1 - b\hat{b}^{-1})(-\ddot{x}_d + \lambda\dot{\tilde{x}})]s - b\hat{b}^{-1}k|s| \\ &\leq -\eta|s| \end{aligned}$$

k must verify

$$k \geq |\hat{b}b^{-1}f - \hat{f} + (\hat{b}b^{-1} - 1)(-\ddot{x}_d + \lambda\dot{\tilde{x}})| + \hat{b}b^{-1}\eta \tag{3.29}$$

Since $f = \hat{f} + (f - \hat{f})$, where $|f - \hat{f}| \leq F$, this leads to

$$k \geq \hat{b}b^{-1}F + |\hat{b}b^{-1} - 1| \cdot |\hat{f} - \ddot{x}_d + \lambda\dot{\tilde{x}}| + \hat{b}b^{-1}\eta \tag{3.30}$$

and using $\hat{b}b^{-1} \leq \beta$ leads to (3.27).

Continuous Control Laws to Approximate Switched Control

The control law derived from the above section is discontinuous across the surface $S(t)$ and leads to control chattering, which is usually undesirable in practice since it involves high control activity and may excite high-frequency dynamics neglected in the course of modeling. In this section, continuous control laws are used to eliminate the chattering.

The control discontinuity is smoothed out by introducing a thin *boundary layer* neighboring the switching surface:

$$B(t) = \{\mathbf{X}, |s(\mathbf{X}; t)| \leq \Phi\}; \quad \Phi > 0 \tag{3.31}$$

where Φ is the boundary layer *thickness*, and is made to be *time varying* in order to exploit the maximum control bandwidth available. Control smoothing is achieved by choosing control law u *outside* $B(t)$ as before, which guarantees boundary layer attractiveness and hence positive invariance—all trajectories starting inside $B(t = 0)$ remain inside $B(t)$ for all $t \geq 0$—and then interpolation u inside $B(t)$, replacing the term $\text{sgn}(s)$ in the expression of u by s/Φ. As proved by Slotine(1983), this leads to *tracking to within a guaranteed precision* $\varepsilon = \Phi/\lambda^{n-1}$, and more generally guarantees that for all trajectories starting inside $B(t = 0)$

$$|\tilde{x}^{(i)}(t)| \leq (2\lambda)^i\varepsilon; \quad i = 0, \cdots, n-1 \tag{3.32}$$

The sliding condition (3.15) is now modified to maintain attractiveness of the boundary layer when Φ is allowed to vary with time.

$$|s| \geq \Phi \Rightarrow \frac{1}{2}\frac{d}{dt}s^2 \leq (\dot{\Phi} - \eta)|s| \tag{3.33}$$

The term $k(\mathbf{X};t)\text{sgn}(s)$ obtained from switched control law u is also replaced by $\bar{k}(\mathbf{X};t)sat(s/\Phi)$, where:

$$\bar{k}(\mathbf{X}_d;t) = k(\mathbf{X};t) - k(\mathbf{X}_d;t) + \frac{\lambda\Phi}{\beta_d} \tag{3.34}$$

with $\beta_d = \beta(\mathbf{X}_d;t)$.

Accordingly, control law u becomes:

$$u = \hat{b}^{-1}[\hat{u} - \bar{k}\text{sat}(s/\Phi)] \tag{3.35}$$

The desired time-history of boundary layer thickness Φ is called *balance condition* and is defined according to the value of $k(\mathbf{X}_d;t)$:

$$k(\mathbf{X}_d;t) \geq \frac{\lambda\Phi}{\beta_d} \quad \Rightarrow \quad \dot{\Phi} + \lambda\Phi = \beta_d k(\mathbf{X}_d;t) \tag{3.36}$$

$$k(\mathbf{X}_d;t) \leq \frac{\lambda\Phi}{\beta_d} \quad \Rightarrow \quad \dot{\Phi} + \frac{\lambda\Phi}{\beta_d^2} = \frac{k(\mathbf{X}_d;t)}{\beta_d} \tag{3.37}$$

with initial condition $\Phi(0)$ defined as:

$$\Phi(0) = \beta_d k(\mathbf{X}_d(0);(0))/\lambda \tag{3.38}$$

The balance conditions have practical implications in terms of design / modeling / performance trade-offs. Neglecting time-constants of order $1/\lambda$, conditions (3.36) and (3.37) can be written

$$\lambda^n\varepsilon \approx \beta_d k(\mathbf{X}_d;t) \tag{3.39}$$

that is

$$\begin{aligned} (\text{bandwidth})^n \quad &\times \quad (\text{tracking precision}) \\ &\approx \quad (\text{parametric uncertainty measured along the desired trajectory}) \end{aligned}$$

It shows that the balance conditions specify the best tracking performance attainable, given the desired control bandwidth and the extent of parameter uncertainty.

3.3.2 Sliding Controller Design

To use the sliding control, the dynamic function (2.6) can be written as

$$\ddot{\theta} = f + bu \tag{3.40}$$

where

$$b = (J + J_M(\theta))^{-1} \tag{3.41}$$

$$f = -(J + J_M(\theta))^{-1} C(\theta)\dot{\theta}^2 \tag{3.42}$$

Assuming the exact values of J, M, R, E and L are not known, thus the exact values of the nonlinear inertia b and the nonlinear function f are unknown. But the estimations of the inertia and the nonlinear dynamics are available, and their upper and lower limits can also be found.

The following a priori information is about the nonlinear inertia b:

$$9 < b < 35 \tag{3.43}$$

$$\beta = \sqrt{b_{max}/b_{min}} = 1.9 \tag{3.44}$$

$$\hat{b} = \sqrt{b_{max} b_{min}} = 17 \tag{3.45}$$

and for the nonlinear dynamics f, the available knowledge is:

$$\hat{f} = -0.05 cos(2\theta)\dot{\theta}^2 \tag{3.46}$$

$$|f - \hat{f}| \leq F = 0.055\dot{\theta}^2 \tag{3.47}$$

Defining s as $s = \dot{\tilde{\theta}} + \lambda\tilde{\theta}$, computing $\dot{s}$ explicitly, and proceeding as described in Section 3.3.1, a control law satisfying the sliding condition can be derived as

$$\hat{u} = 0.05cos(2\theta)\dot{\theta}^2 + \ddot{\theta}_d - \lambda\dot{\tilde{\theta}} \tag{3.48}$$

$$u = \hat{b}^{-1}[\hat{u} - \bar{k}\text{sat}(s/\Phi)] \tag{3.49}$$

where

$$\bar{k} = k(\Theta; t) - k(\Theta_d; t) + \frac{\lambda\Phi}{\beta} \tag{3.50}$$

$$k(\Theta; t) = \beta(F(\Theta; t) + \eta) + (\beta - 1)|\hat{u}(\Theta; t)| \tag{3.51}$$

and the boundary layer thickness Φ is derived from the balance conditions (3.36) and (3.37), with the initial condition (3.38).

The constant values are chosen as $\eta = 0.5$ and $\lambda = 10$ so that the maximum value of $\hat{b}^{-1}\bar{k}/\Phi\lambda$ is approximately equal to the value of K_p used in the PID controller, thus establishes a fair comparison among different control methods.

The experiment results of the sliding controller are shown from Figure 3-3 to Figure 3-5. The performance in Figure 3-3 is much better than PID results, with the tracking error about 60 percent of the PID tracking error. Figure 3-4 shows the s trajectory stays within the boundary layers, thus confirming the parameter estimates are consistent with the real system parameters. The control torque in Figure 3-5 is noisier than PID control, we conclude it is caused by the feedforward estimate control value $\hat{u}$ which has a high gain for $\dot{\tilde{\theta}}$. Since the feedback control gains are kept the same as in PID controller, the comparison stays to be fair.

3.4 Adaptive Control

In order to improve the system performance when large parametric uncertainties are present, an *adaptive* controller is introduced, where uncertain parameters in the

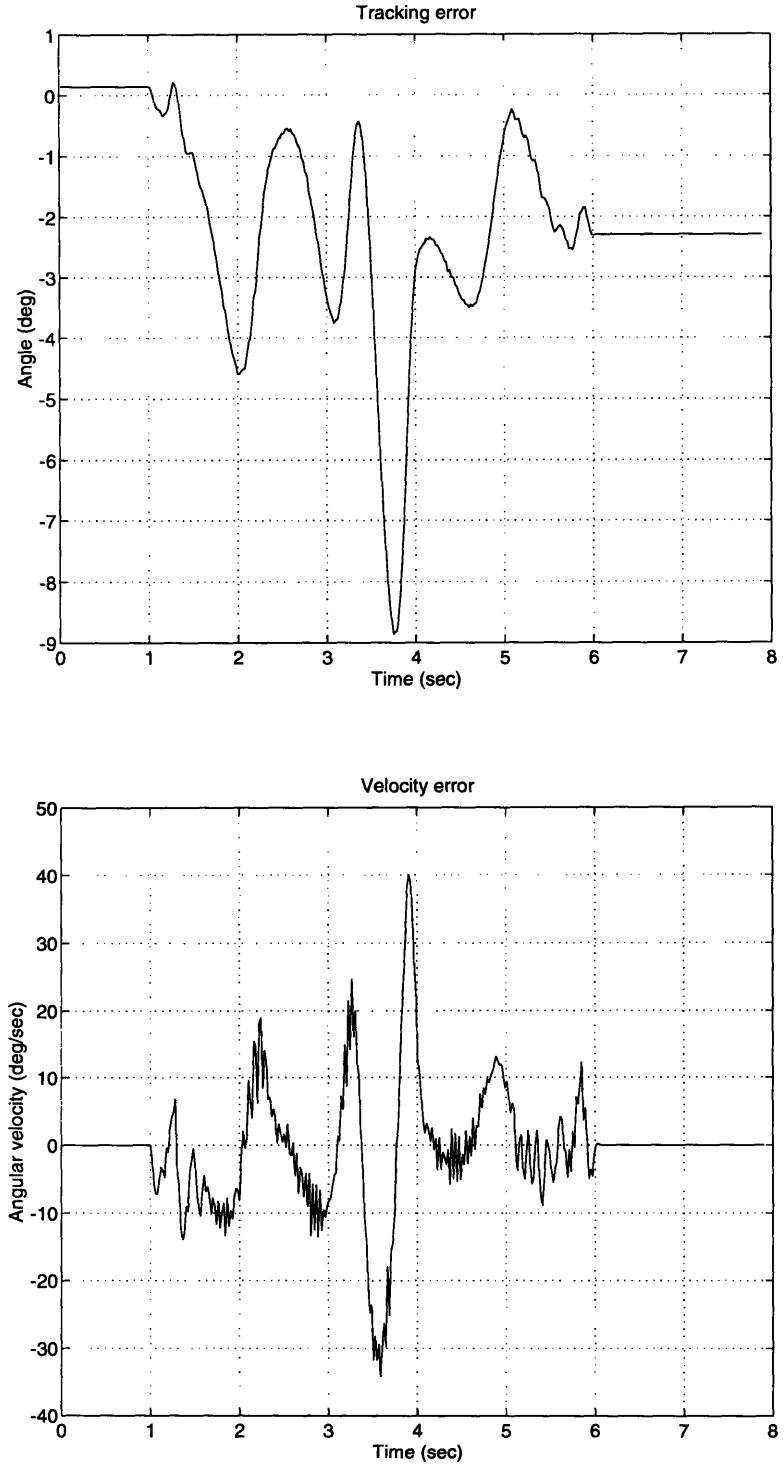

Figure 3-3: Tracking error and velocity error for sliding control, tracking error is about 60 percent of PID tracking error

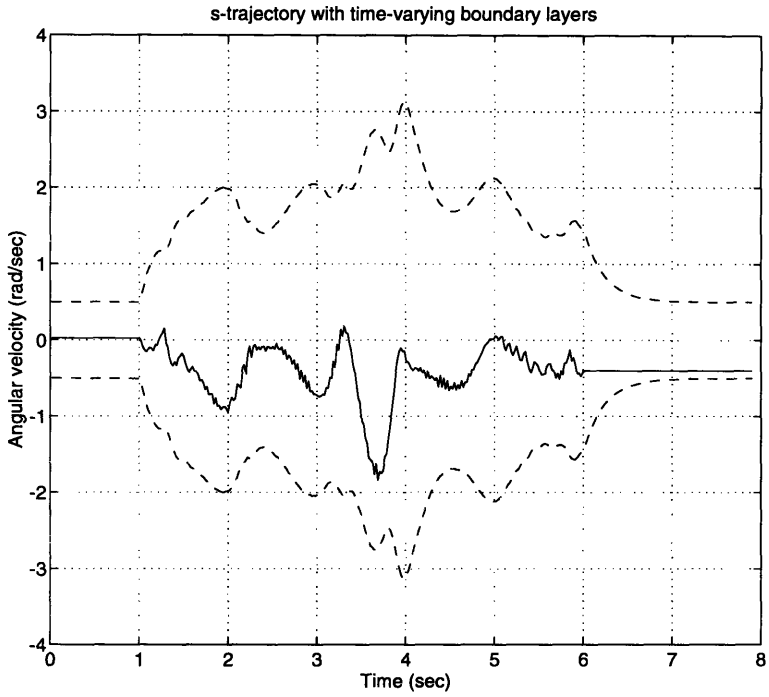

Figure 3-4: Boundary layers (dashed lines) and s trajectory (solid line) for sliding control, which confirms parameter estimates are consistent with the real system

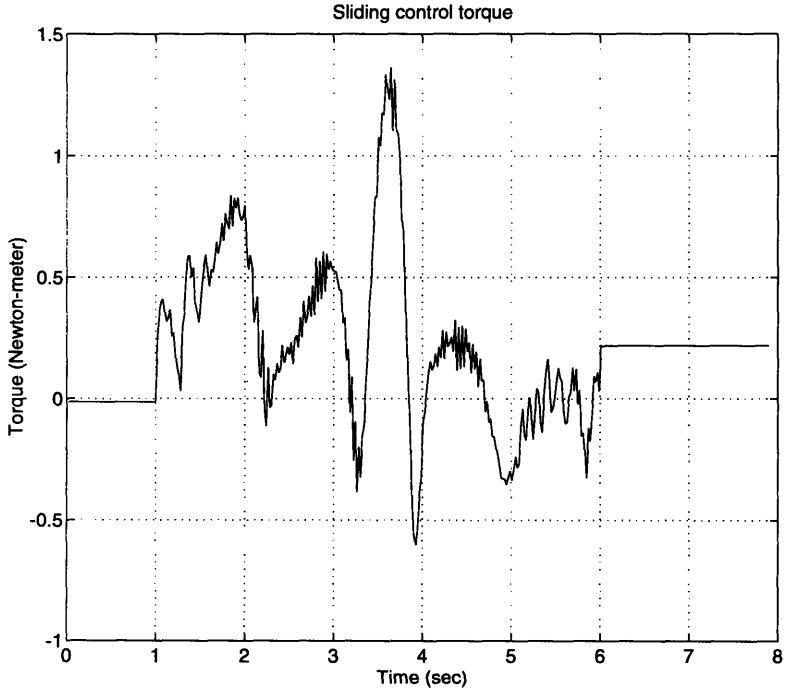

Figure 3-5: Control torque for sliding control

nonlinear dynamics are estimated on-line, so that the system can closely tracks the desired trajectory. The adaptation law is derived again from the Lyapunov function so the system maintains its stability.

3.4.1 Adaptive Control Theory

The adaptive controller is illustrated by applying it to a manipulator system as in [18], since our nonlinear mechanical set-up can be regarded as a simplified manipulator system. The general dynamics can be written as

$$H(\theta)\ddot{\theta} + C(\theta, \dot{\theta})\dot{\theta} + g(\theta) = u \tag{3.52}$$

where $H(\theta)$ is the manipulator inertia function, $C(\theta, \dot{\theta})\dot{\theta}$ is the centripetal and Coriolis torque, and $g(\theta)$ is the gravitational torque. Given a proper definition of the unknown parameter vector $\mathbf{a}$ describing the manipulator's mass properties, the terms $H(\theta)$, $C(\theta, \dot{\theta})\dot{\theta}$, and $g(\theta)$ all depend *linearly* on $\mathbf{a}$. This physical property of the manipulator system allows us to define a *known* matrix $\mathbf{Y} = \mathbf{Y}(\theta, \dot{\theta}, \dot{\theta}_r, \ddot{\theta}_r)$ such that

$$H(\theta)\ddot{\theta}_r + C(\theta, \dot{\theta})\dot{\theta}_r + g(\theta) = \mathbf{Y}(\theta, \dot{\theta}, \dot{\theta}_r, \ddot{\theta}_r)\mathbf{a} \tag{3.53}$$

where

$$\begin{aligned} \dot{\theta}_r &= \dot{\theta}_d - \lambda\tilde{\theta} \\ \tilde{\theta} &= \theta - \theta_d \\ \ddot{\theta}_r &= \ddot{\theta}_d - \lambda(\dot{\theta} - \dot{\theta}_d) \end{aligned}$$

Let us define $\tilde{\mathbf{a}} = \hat{\mathbf{a}} - \mathbf{a}$ as the parameter estimation error, with $\mathbf{a}$ being the constant vector of the unknown parameters in the manipulator system, and $\hat{\mathbf{a}}$ its estimate. Consider the Lyapunov function

$$V(t) = \frac{1}{2}[s^T H s + \tilde{\mathbf{a}}^T \mathbf{\Gamma}^{-1} \tilde{\mathbf{a}}] \tag{3.54}$$

where $\mathbf{\Gamma}$ is a symmetric positive definite matrix, and $s = \dot{\tilde{\theta}} + \lambda\tilde{\theta} = \dot{\theta} - \dot{\theta}_r$. Differentiating equation (3.54) yields

$$\dot{V}(t) = s^T(u - H\ddot{\theta}_r - C\dot{\theta}_r - g) + \dot{\hat{\mathbf{a}}}^T\mathbf{\Gamma}^{-1}\tilde{\mathbf{a}} \tag{3.55}$$

Taking advantage of the linear dependency of the unknown parameters shown in equation (3.53), we can choose the control law to be

$$u = \mathbf{Y}\hat{\mathbf{a}} - K_D s \tag{3.56}$$

which includes a feedforward term $\mathbf{Y}\hat{\mathbf{a}}$ equivalent to the $\hat{u}$ of the sliding controller, and a simple PD term $K_D s$. So the derivative of the Lyapunov function becomes

$$\dot{V}(t) = s^T\mathbf{Y}\tilde{\mathbf{a}} - s^T K_D s + \dot{\hat{\mathbf{a}}}^T\mathbf{\Gamma}^{-1}\tilde{\mathbf{a}} \tag{3.57}$$

choosing the adaptation law to be

$$\dot{\hat{\mathbf{a}}} = -\mathbf{\Gamma}\mathbf{Y}^T s \tag{3.58}$$

then yields

$$\dot{V}(t) = -s^T K_D s \leq 0 \tag{3.59}$$

This implies the output error converges to the surface $s = 0$, which in turn shows that $\tilde{\theta}$ and $\dot{\tilde{\theta}}$ tend to 0 as t approaches infinity. So the global stability of the system and convergence of the tracking error are both guaranteed by this adaptive controller.

3.4.2 Adaptive Controller Design

To design an adaptive controller for our nonlinear mechanical system, we assume the general form of the nonlinear dynamics of the system is known, but the constant parameters J and M are unknown. Now we can rewrite the system dynamics in a form similar to equation (3.53) where the dynamics linearly depend on the constant

vector of the unknown parameters $\mathbf{a} = [\ J \quad M\ ]^T$:

$$(J + M\frac{R^2c^2}{a^2})\ddot{\theta}_r + \frac{1}{2}M\dot{\theta}\frac{d}{d\theta}(\frac{R^2c^2}{a^2})\dot{\theta}_r = \mathbf{Y}(\theta, \dot{\theta}, \dot{\theta}_r, \ddot{\theta}_r)\mathbf{a} \tag{3.60}$$

with

$$\mathbf{Y}(\theta, \dot{\theta}, \dot{\theta}_r, \ddot{\theta}_r) = [\ \ddot{\theta}_r \quad \frac{R^2c^2}{a^2}\ddot{\theta}_r + \frac{1}{2}\dot{\theta}\frac{d}{d\theta}(\frac{R^2c^2}{a^2})\dot{\theta}_r\ ] \tag{3.61}$$

then the control law and adaptation law would be:

$$u = \mathbf{Y}\hat{\mathbf{a}} - K_D s \tag{3.62}$$

$$\dot{\hat{\mathbf{a}}} = -\mathbf{\Gamma}\mathbf{Y}^T s \tag{3.63}$$

The adaptation rates are chosen to be $\mathbf{\Gamma} = [\ .005 \quad 500\ ]$, starting without any *a priori* information about the parameters ($\hat{\mathbf{a}}(0) = \mathbf{0}$). Assume the same desired trajectory and velocity in Figure 2-5, and with $\lambda = 10$ and $K_D = 0.56$ (leading to the same PD gains as in the PID control), the corresponding tracking errors, control torque, and parameter estimates are plotted from Figure 3-6 to Figure 3-9.

Figure 3-6 shows that the tracking performance of the adaptive control is much better than both PID and sliding control, having only half the sliding control error. The performance improves along with the parameter adaptation, which is very evident at the last 2 seconds of the 5 second run.

The parameter estimates in Figure 3-7 and Figure 3-8 are both approaching to their true values during the control, but they never reach the final real values since there is still unmodeled dynamics left in the real nonlinear system.

Figure 3-9 shows the total control torque from both the feedforward adaptive control and the feedback portion of the control. The relatively high gains in the feedforward term again causes noisier torque than the PID control.

Figure 3-10 is a comparison of the feedforward adaptive control in the experiments and desired total control in the simulation. The adaptive controller is truly approaching the desired control value as the parameters are adapting closer to their real values.

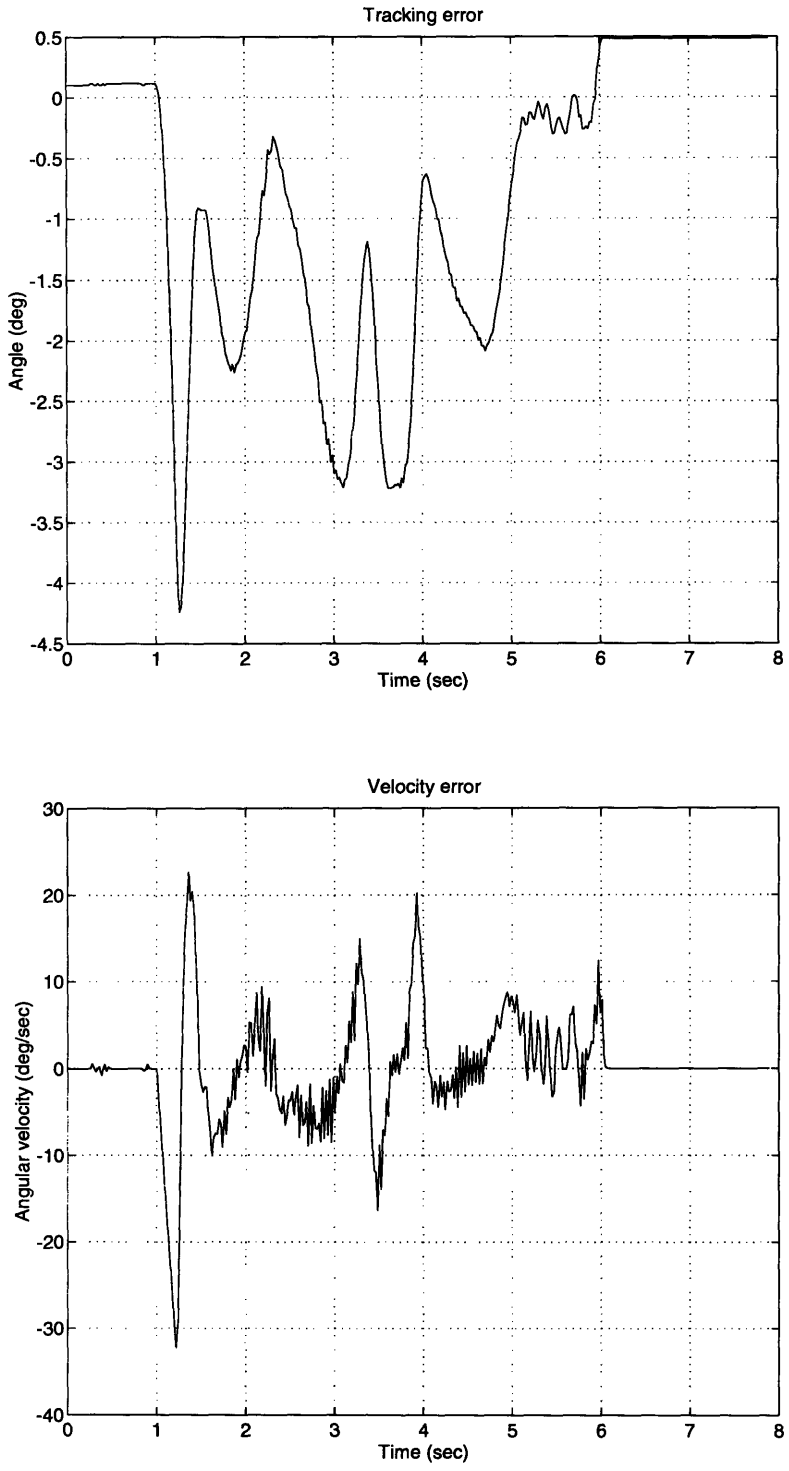

Figure 3-6: Tracking error and velocity error for adaptive control, only half as much as in sliding control

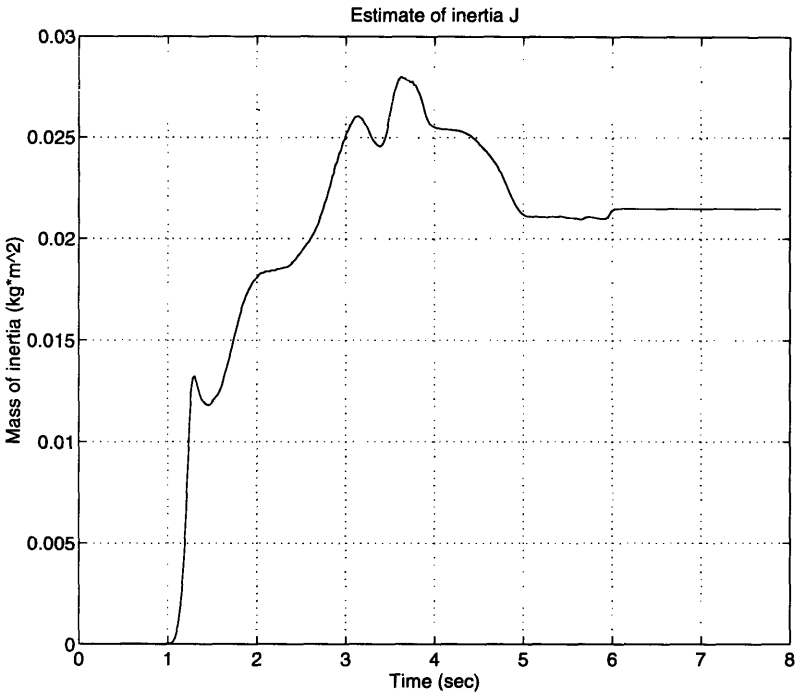

Figure 3-7: Inertia estimation for adaptive control

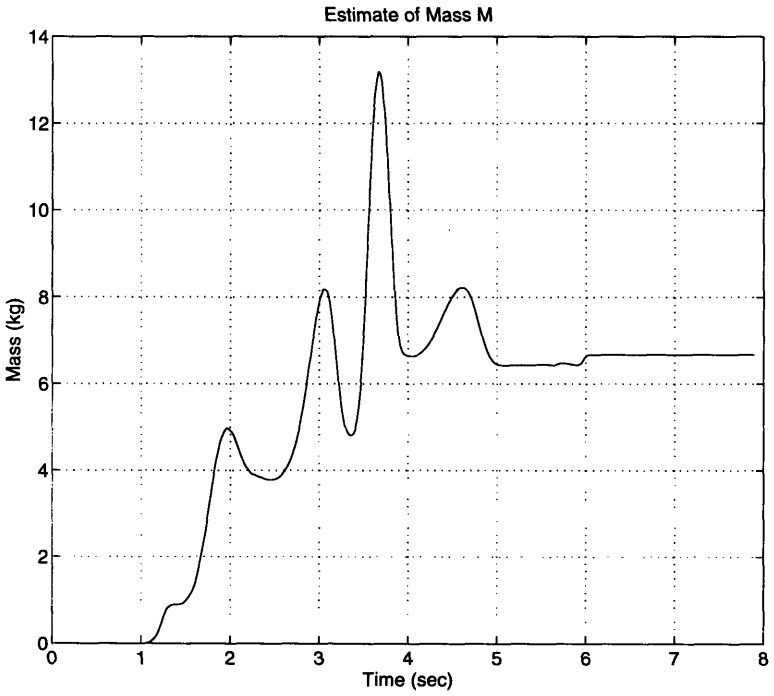

Figure 3-8: Mass estimation for adaptive control

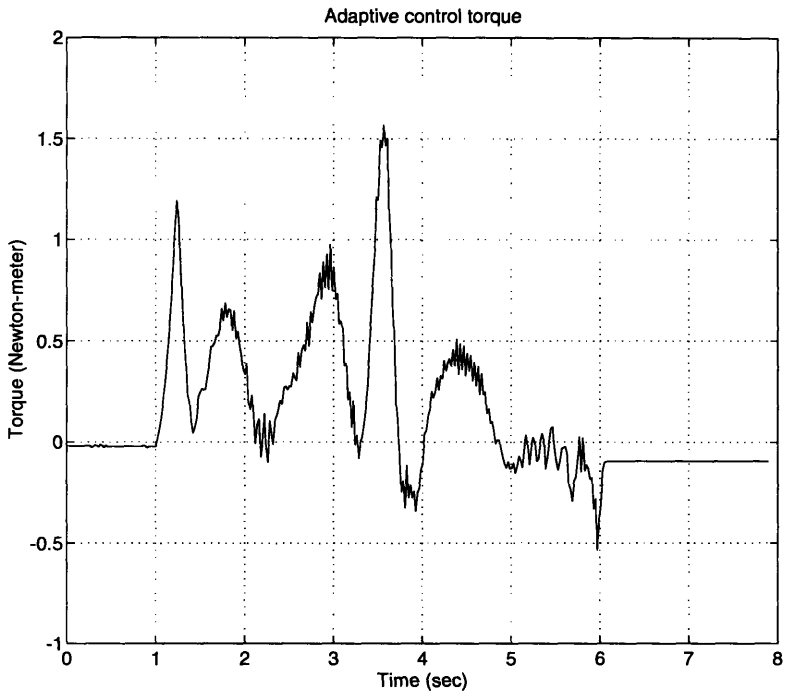

Figure 3-9: Total control torque for adaptive control

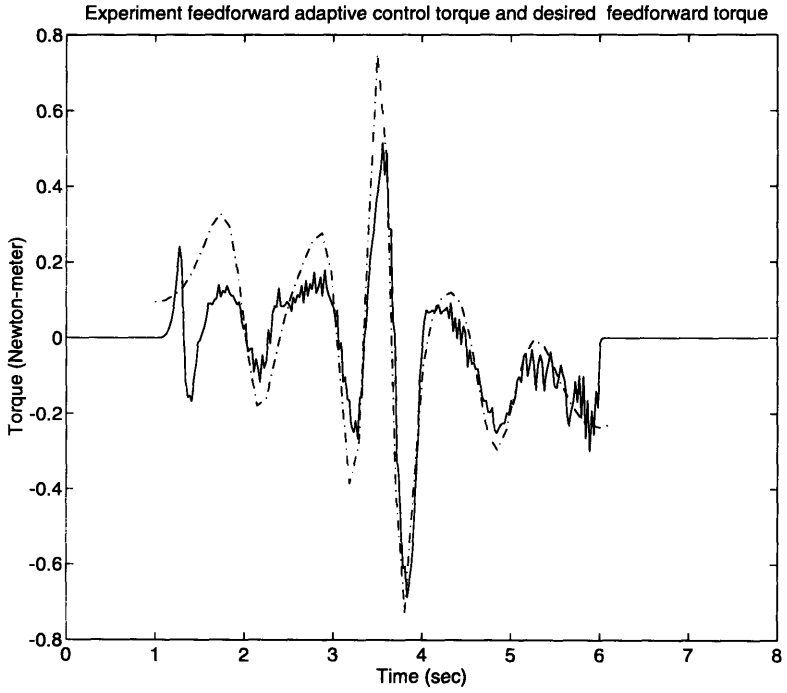

Figure 3-10: Experiment adaptive control (solid line) and simulated desired total control (dashed line)

Chapter 4

Connectionist Controllers for Nonlinear Systems

Nonlinear control design techniques like sliding control and adaptive control have been successfully used on some nonlinear systems. Yet, the system dynamic structure has to be understood beforehand and the uncertain parameters have to be estimated. If the system dynamics is hard to model and the parameters can not be easily estimated, connectionist controllers may be the choice.

This chapter introduces integration of a neural networks scheme called *Gaussian network control* into the trajectory control of nonlinear systems. No detailed a priori information about the system dynamics is required, the network will quickly learn the nonlinear dynamics and simultaneously controls the motion of the system.

4.1 Gaussian Network Control Theory

Gaussian network control uses a network of Gaussian radial basis functions to adaptively compensate for the plant nonlinearities [16]. The a priori information about the nonlinear function of the plant is its degree of smoothness. The weight adjustment is determined using Lyapunov theory, so the algorithm is proven to be globally stable, and the tracking errors converge to a neighborhood of zero.

Sampling theory shows that bandlimited functions can be exactly represented

at a countable set of points using an appropriately chosen interpolating function. When approximation is allowed, the bandlimited restriction can be relaxed and the interpolating function gets a wider selection. Specifically, the nonlinear function $f(\mathbf{x})$ can be approximated by smoothly truncated outside a compact set $\mathbf{A}$ so that the function can be Fourier transformed. Then by truncating the spectrum, the function can be bandlimited and uniformly approximated by

$$f(\mathbf{x}) = \sum_{I \in I_o} c_I g_\sigma(\mathbf{x} - \xi_I) \tag{4.1}$$

where c_I are the weighting coefficients and ξ_I form a regular lattice covering the subset $\mathbf{A}_T$, which is larger than the compact set $\mathbf{A}$ by an n-ball of radius ρ surrounding each point of $\mathbf{x} \in \mathbf{A}$. The index set is $I_o = \{I \mid \xi_I \in \mathbf{A}_T\}$.

$g_\sigma(\mathbf{x} - \xi)$ is the Gaussian radial basis function given by:

$$g_\sigma(\mathbf{x} - \xi) = \exp(-\pi\sigma_\nu^2||\mathbf{x} - \xi||^2) = \exp[-\pi\sigma_\nu^2(\mathbf{x} - \xi)^T(\mathbf{x} - \xi)] \tag{4.2}$$

Here ξ is the center of the radial Gaussian, and σ_ν^2 is a measure of its essential width. Gaussian functions are well suited for the role of interpolating function because they are bounded, strictly positive and absolutely integrable, and they are their own Fourier transforms.

Expansion (4.1) maps onto a network with a single hidden layer. Each node represents one term in the series with the weight ξ_I connecting between the input $\mathbf{x}$ and the node. It then calculates the activation energy $r_I^2 = ||\mathbf{x} - \xi_I||^2$ and outputs a Gaussian function of the activation, $\exp(-\pi r_I^2 \sigma_\nu^2)$. The output of the network is weighted summation of the output of each node, with each weight equals to c_I. Figure 4-1 shows the structure of the network described above.

The next step is the construction of the controller. Consider the nonlinear dynamics system

$$x^{(n)}(t) + f(\mathbf{x}(t)) = b(\mathbf{x}(t))u(t) \tag{4.3}$$

define the unknown nonlinear function $h = b^{-1}f$, and let h_A and b_A^{-1} be the ra-

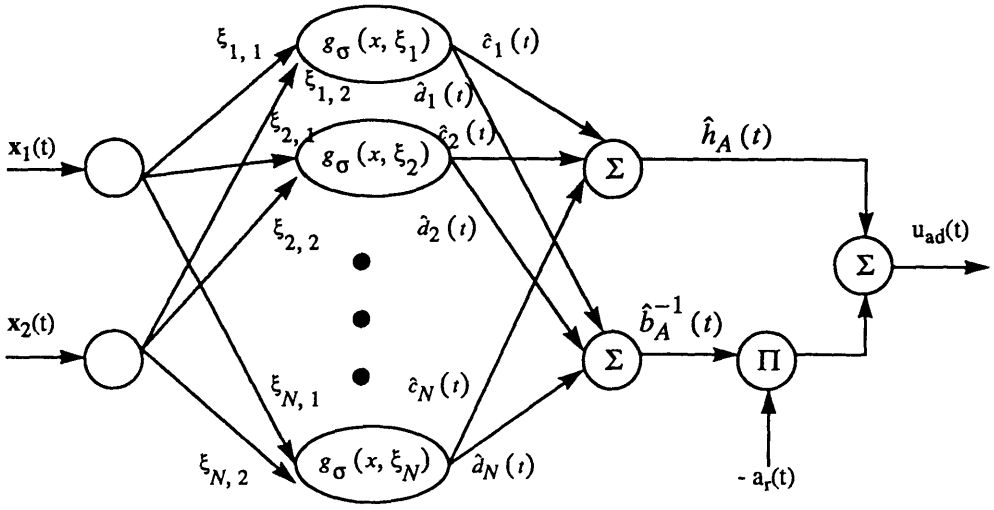

Figure 4-1: Structure of the Gaussian network

dial Gaussian network approximations to the functions h and b^{-1} respectively with approximation error ϵ_h and ϵ_b as small as desired on the chosen set $\mathbf{A}$.

Figure 4-2 shows the structure of the control law. To guarantee the stability of the whole dynamic system, the control law is constructed as a combination of three components: a linear PD control, a sliding control and an adaptive control represented by the Gaussian network.

$$\begin{aligned} u(t) &= -k_D s(t) - \frac{1}{2} M_2(\mathbf{x}(t)) ||\mathbf{x}(t)|| s_\Delta(t) + m(t) u_{sl}(t) \\ &\quad + (1 - m(t))[\hat{h}_A(t, \mathbf{x}(t)) - \widehat{b_A^{-1}}(t, \mathbf{x}(t)) a_r(t)] \end{aligned} \tag{4.4}$$

$m(t)$ is a modulation allowing the controller to smoothly transition between sliding and adaptive controls.

$$m(t) = \max(0, \text{sat}(\frac{r(t) - 1}{\Psi})) \tag{4.5}$$

where $r(t) = ||\mathbf{x}(t) - \mathbf{x}_0||$.

$$a_r(t) = \lambda_v^T \tilde{\mathbf{x}}(t) - x_d^{(n)}(t) \tag{4.6}$$

with $\lambda_v^T = [\ 0, \quad \lambda^{n-1}, \quad (n-1)\lambda^{n-2}, \quad \cdots, \quad (n-1)\lambda\ ]$ and $x_d^{(n)}(t)$ is the nth derivative of the desired trajectory.

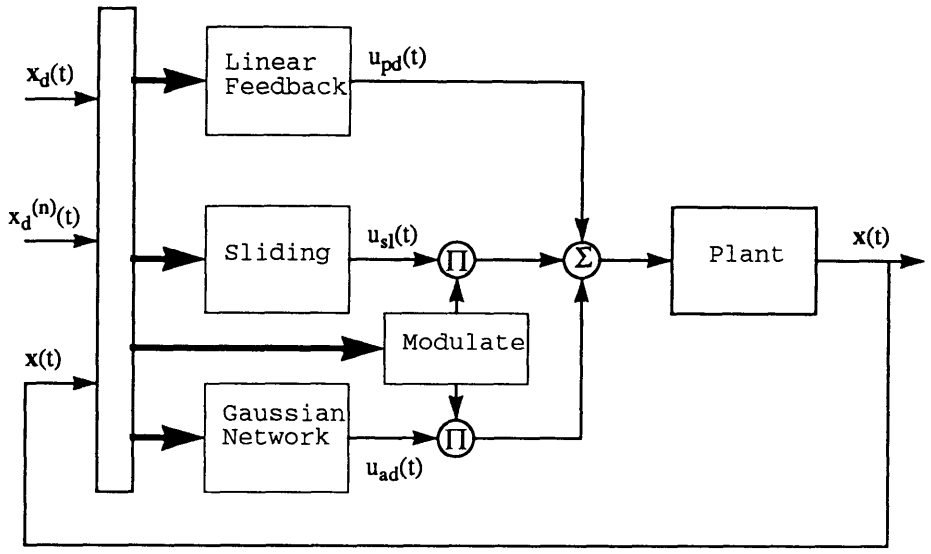

Figure 4-2: Structure of the Gaussian network controller

The adaptive components $\hat{h}_A$ and $\widehat{b_A^{-1}}$ are realized as the outputs of a single Gaussian network, with two sets of output weights: $\hat{c}_I(t)$ and $\hat{d}_I(t)$ for each node in the hidden layer.

$$\begin{aligned} \hat{h}_A(t, \mathbf{x}(t)) &= \sum_{I \in I_o} \hat{c}_I(t) g_\sigma(\mathbf{x}(t) - \xi_I) \\ \widehat{b_A^{-1}}(t, \mathbf{x}(t)) &= \sum_{I \in I_o} \hat{d}_I(t) g_\sigma(\mathbf{x}(t) - \xi_I) \end{aligned} \tag{4.7}$$

The output weights are adjusted according to the following adaptation law:

$$\dot{\hat{c}}_I(t) = -k_{a1}[(1 - m(t)) s_\Delta(t) g_\sigma(\mathbf{x}(t) - \xi_I)] \tag{4.8}$$

$$\dot{\hat{d}}_I(t) = k_{a2}\, a_r(t)[(1 - m(t)) s_\Delta(t) g_\sigma(\mathbf{x}(t) - \xi_I)] \tag{4.9}$$

where positive constants k_{a1} and k_{a2} are adaptation rates. See Figure 4-1 for the detailed structure of the adaptive control law.

As proved by Sanner and Slotine in [16], when the parameters in the controller are chosen appropriately according to the a priori knowledge about the smoothness and upper bounds of the nonlinear dynamic functions, the controller thus constructed will be stable and convergent. All the states in the adaptive system will remain bounded and the tracking errors will asymptotically converge to a neighborhood of zero.

4.2 Gaussian Network Controller Design

To apply the Gaussian network control to the nonlinear mechanical system, the dynamic function (2.6) can be expressed according to equation (4.3):

$$\ddot{\theta}(t) + f(\theta(t), \dot{\theta}(t)) = b(t, \theta(t))u(t) \tag{4.10}$$

where

$$b^{-1}(t, \theta(t)) = J + J_M(\theta(t)) \tag{4.11}$$

$$b^{-1}(t, \theta(t))f(\theta(t), \dot{\theta}(t)) = C(\theta(t))\dot{\theta}^2(t) \tag{4.12}$$

Since $h = b^{-1}f$ is a function of both $\theta(t)$ and $\dot{\theta}(t)$, if a Gaussian network is used to approximate the function, it has to be a two-dimensional network. This would increase the network size and increase the computation demand when it is implemented into the real time control. In light of this computation constraint, we choose to use a one dimensional Gaussian network C_A to approximate $C(\theta(t))$, with the prior knowledge that the nonlinearity is a multiplication of $\dot{\theta}^2(t)$ and the unknown function $C(\theta(t))$. Thus the control law in equation (4.4) would be modified as

$$\begin{aligned} u(t) = & -k_D s(t) - \frac{1}{2}M_2(\Theta(t))||\Theta(t)||s_\Delta(t) + m(t)u_{sl}(t) \\ & +(1 - m(t))[\hat{C}_A(t, \theta(t))\dot{\theta}^2(t) - \widehat{b_A^{-1}}(t, \theta(t))a_r(t)] \end{aligned} \tag{4.13}$$

Yet the adaptation laws for the output weights will remain unchanged in equation (4.8) and (4.9).

Now we come to the details of constructing the Gaussian network controller. Figure 4-3 shows the phase space plot of the desired position and velocity trajectory in Figure 2-5, which is the same as used in other controllers for the fair comparison purpose. The set $\mathbf{A}_d$ is thus chosen to be:

$$||\mathbf{x}||_\mathbf{w} = \max(\frac{|x|}{6.6317}, \frac{|\dot{x}|}{2.1}) \tag{4.14}$$

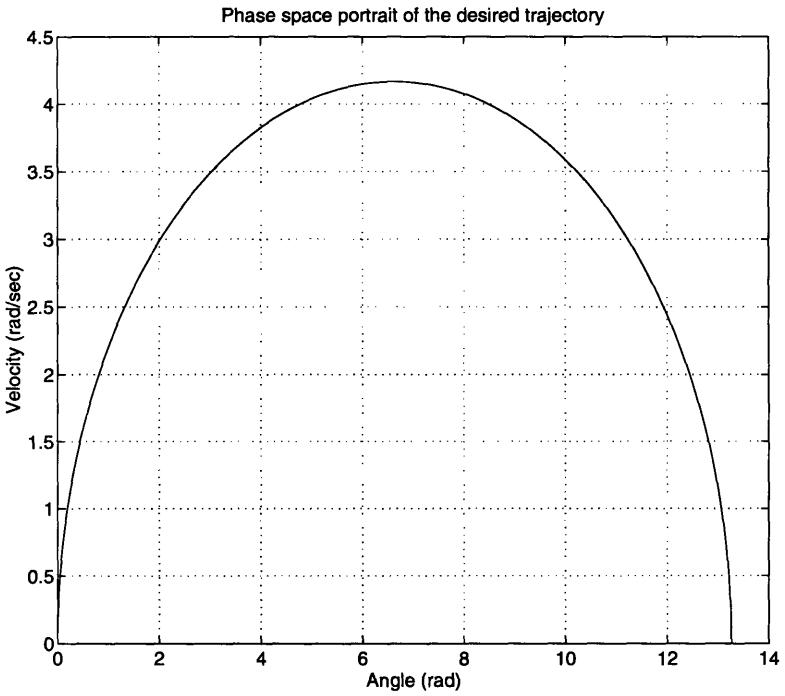

Figure 4-3: Phase space portrait of the desired trajectory

with its center at $\mathbf{x}_0 = [6.6317,\ 2.1]^T$. This represents a rectangular subset of the state space, $\mathbf{A}_d = [0,\ 13.27] \times [0,\ 4.2]$. The transition region between adaptive and sliding control modes is $\Psi = 0.1$ as the value in equation (4.5). So we have

$$\mathbf{A} = \{\mathbf{x} \mid ||\mathbf{x} - \mathbf{x}_0||_{\mathbf{w}} \leq 1.1\} \tag{4.15}$$

The maximum desired acceleration is $|\ddot{x}_d|_{max} \leq 2.62$ as shown in the desired trajectory plot in Figure 2-5 , which sets $|a_r| \leq 115$ for all $\mathbf{x} \in \mathbf{A}$ and the error bound to be $\epsilon_r = \epsilon_h + 115\epsilon_b$.

To achieve asymptotic tracking accuracy of 2.5 degrees, taking $k_D = 0.56$ and $\lambda = 10$ requires that $\epsilon_r \leq 0.25$. Assuming the frequency contents of h and b^{-1} on $\mathbf{A}$ are $\beta_x = 0.2$ with uniform error no worse than 0.05 and an upper bound of 0.2 for the transform of h and 0.8 for the transform of b^{-1}, a close following to the parameter selection procedure in [16] shows that the choices $\theta_x = 2$ and $l_x = l_{\dot{x}} = 1$ should be sufficient to ensure the required bound on ϵ_r. This leads to a network of Gaussian

nodes with variances $\sigma_x^2 = 0.13$ and mesh sizes $\Delta_x = 1.3$. So the network includes all the nodes contained in the set $\mathbf{A}_T = [-\Delta_x, 11\Delta_x]$, for a total of 13 nodes for each network.

The control laws are given by equation (4.4) and (4.7), using a value of $\Phi = 0.45$ as the sliding controller boundary layer, with adaptation rates $k_{a1} = k_{a2} = 0.5$. Assuming uniform upper bounds of $M_0(\mathbf{x}) = 0.8$, $M_1(\mathbf{x}) = 0.12$ and $M_2(\mathbf{x}) = 0.1$, the sliding controller gains are taken as $k_{sl} = 0.8 + 0.12|a_r|$.

Using Fourier transformation, it's easy to find that the above spectra assumptions are justified for both h and b^{-1}. The experiment results are shown from Figure 4-4 to Figure 4-7.

The performance in Figure 4-4 is considerably better than all the controllers in Chapter 3. Figure 4-6 indicates that the Gaussian network is adapting very fast to the nonlinearity of the system and takes most part of the total control action.

Figure 4-7 shows that the simulated control value doesn't include the unmodeled dynamics in the system, yet the unknown part of the dynamics is learned by the Gaussian network. So learning ability of the Gaussian network is much better than the adaptive control.

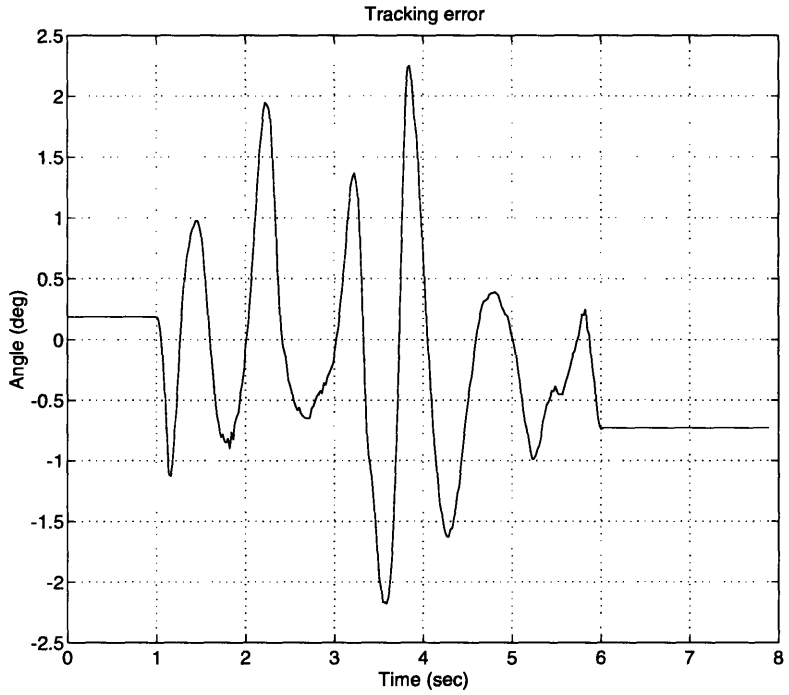

Figure 4-4: Tracking error for Gaussian network control, smaller than all previous controllers

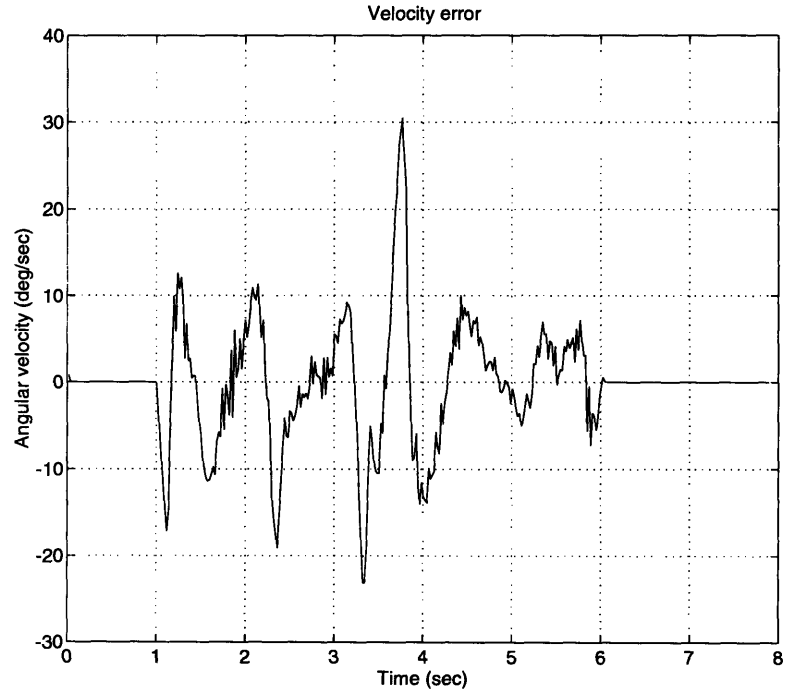

Figure 4-5: Velocity error for Gaussian network control

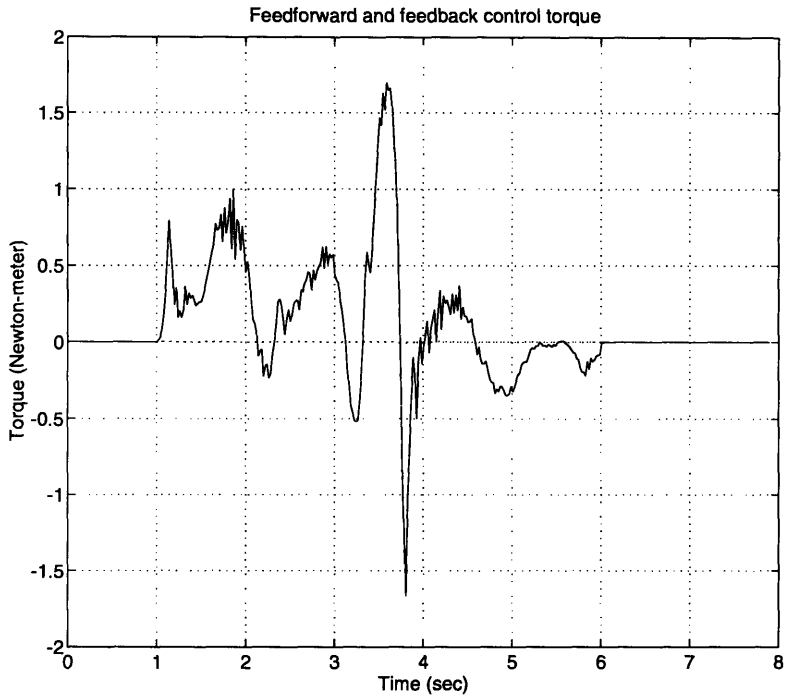

Figure 4-6: Control torque from PD (dashed line) and Gaussian network (solid line) for Gaussian network control, the PD term is invisibly small

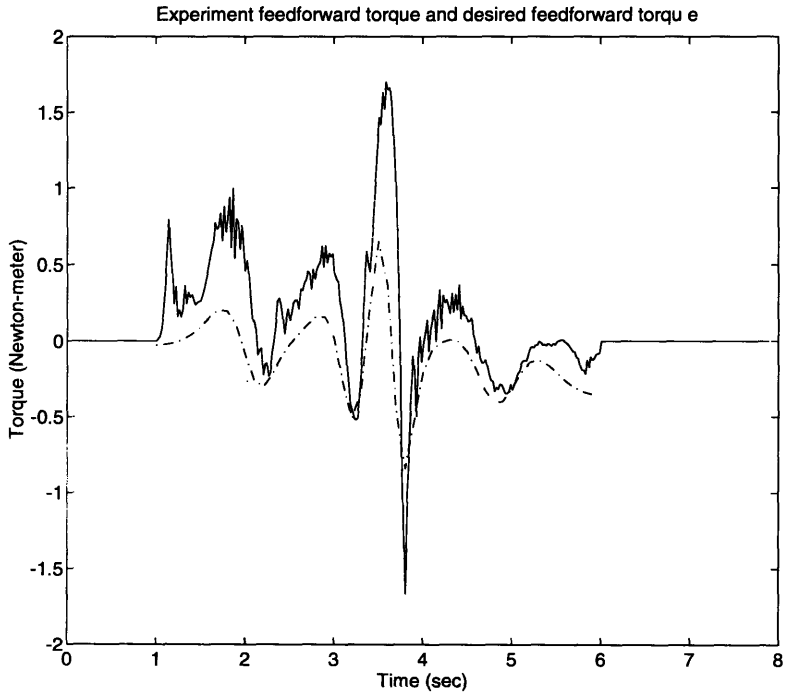

Figure 4-7: Control torque from experiment (solid line) and from simulation (dashed line) for Gaussian network control, which shows the Gaussian network has learned unmodeled dynamics

Chapter 5

Experimental Comparison of Control Theoretic and Connectionist Controllers

This chapter compares the control theoretic and connectionist controllers based on their experimental performance on the nonlinear electro-mechanical system in Chapter 3 and Chapter 4. Comparison of the four controllers are made about a priori information requirements, tracking performance, stability guarantees, and computational requirements.

To establish a fair comparison among different controllers, the feedback bandwidth of each control method is chosen to be the same. The performance data for final comparison is collected using the same control frequency. Section 5.1 describes how these fair comparison standards are constructed. In Section 5.2, detailed analysis of the performance of each control method is presented. Finally, Section 5.3 summarizes the comparison results, and suggests how to choose appropriate control techniques to one's specific control applications based on the comparative study.

5.1 How to Achieve a Fair Comparison

The four control methods differ from one another greatly, making it critical to establish a set of fair comparison standards. In the experiments, we first make certain that the feedback control gains are comparable for each method, then we run the experiments at the same control frequency despite one controller may need less computation time than another.

For the PID controller, the feedback control gains are simply the K_d, K_p and K_i terms. For the sliding controller, the gains are not so obvious. Let us look at equation (3.49), $\hat{u}$ is a feedforward term aiming to compensate the system nonlinearity from the given estimate. The second term $\bar{k}\text{sat}(s/\Phi)$ is a feedback term to keep the control surface to stay within the varying boundaries. Thus we can treat the coefficient of $s = \dot{\tilde{\theta}} + \lambda\tilde{\theta}$ as the feedback gains. In the experiments, the constant values are chosen as $\eta = 0.5$ and $\lambda = 10$ so that the maximum value of $\hat{b}^{-1}\bar{k}/\Phi\lambda$ is approximately equal to the value of K_p used in the PID controller.

As for the adaptive controller, equation (3.63) shows that $\mathbf{Y}\hat{\mathbf{a}}$ is a feedforward term adapting to the system nonlinearity, while $K_D s$ is the feedback control term. So we choose $K_D = 0.56$ and $\lambda = 10$ leading to the same PD gains as in the PID control. The same is true with the Gaussian network control, the PD gains in $k_D s(t)$ is once again set to the same values as other controllers.

Another issue regarding to the fair comparison is the control frequency. The fast computing ability of digital computers allows us to execute the controllers as continuous-time systems. Yet the higher the control frequency, the closer the controllers would be to the real-time continuous system. The control frequency is determined by the computation need of each controller, the Quick C compiler running time on the 486 PC and other hardware constraints such as the D/A and A/D conversion speed. Section 5.2 would give a detailed description of the computation each controller requires. Yet for the purpose of fair comparison, the final performance data is collected using the same control frequency, i.e., the lowest of the four methods.

5.2 Comparison of the Control Theoretic and Connectionist Controllers

The PID controller requires the prior knowledge of the constant inertia value of the system. It then selects the control gains based on the closed-loop bandwidth and this inertia value. The tracking performance in Figure 3-1 shows that the PID control can follow the trajectory, although with some large position and velocity errors. The computation time for each control step is 0.0006 seconds, which shows the computation load is small for the PID controller.

The sliding controller is provided with the estimated structure of the nonlinear dynamics and the error bounds. The tracking errors in Figure 3-3 are about 60 percent of those in the PID controller. The reasons for the improved performance are that the controller is provided with the desired acceleration term $\ddot{\theta}_d$, and that the nonlinear structure of the system is known and incorporated into the control action. Some tracking errors still remain during the control because of the parameter uncertainties in the estimated dynamics model. Unlike the PID controller, the performance of sliding controller can be further improved given better nonlinearity estimation. There is a quantified relationship between the performance and uncertainty trade off shown in equation (3.39). The sliding controller needs a slightly more computation time than PID: 0.00078 seconds.

The adaptive controller has the capability to quickly learn and adapt to parameter uncertainties. Like the sliding control, the explicit model of the nonlinear structure is provided to the controller, with unknown constants J and M both set to zero at the start. Figure 3-7 and Figure 3-8 show that both J and M adapt quickly during the run. The tracking performance in Figure 3-6 improves along with the parameter adaptation. During the first second of the trajectory, both position and velocity errors are large due to the poor starting parameter values. Then they decrease while the parameters are approaching to their true values. At the last two seconds of the run, both errors reduce to the minimum since the parameters are at their closest to the actual values.

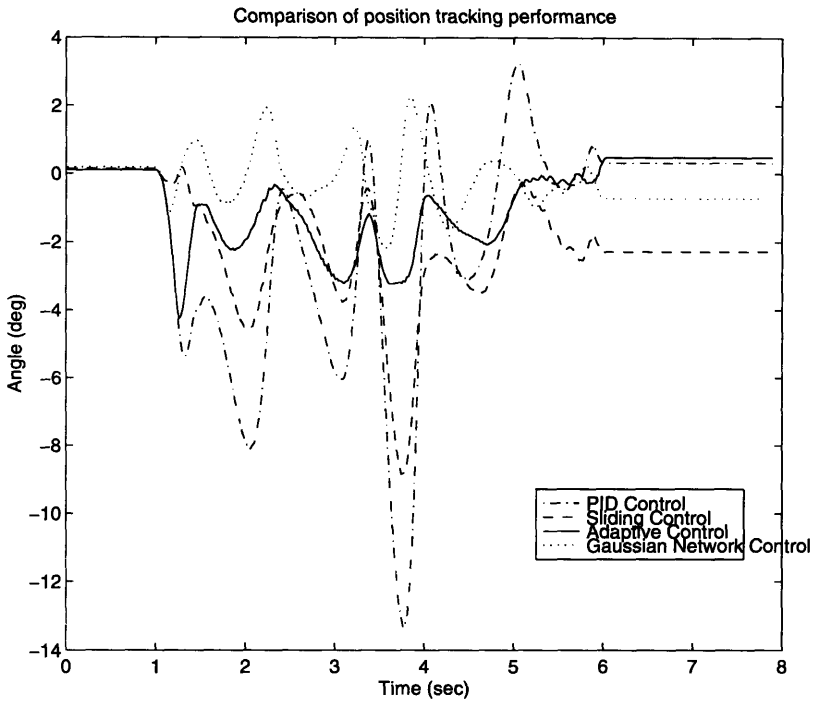

Figure 5-1: Comparison of tracking errors of the four controllers

These parameters never finally reach to their real values since there is still uncertain dynamics caused by friction in the system that are difficult to model. Nevertheless, the adaptation greatly reduced the tracking error to half the sliding control error. The computation requires 0.0011 seconds for each control step, longer than PID and sliding control. Both sliding control and adaptive sliding control are theoretically proven to be stable. So the overall system stability is assured.

While previous control techniques may fail to achieve the desired performance in the presence of unmodeled dynamics, the advantage of using a connectionist controller for the nonlinear control is that it doesn't require much information about the nonlinear dynamics which is difficult to obtain. Gaussian network control has a systematic approach to choose the structure of the network while assuring the overall system stability. The knowledge required for the controller is estimation of the frequency contents of the nonlinear function to be approximated. The structure of the Gaussian network is chosen based on this knowledge and the required performance.

Its performance is much better than the above three controllers, as seen from Figure 4-4 and Figure 5-1. The computation time step is 0.0036 seconds, indicating a large computation requirement and a large memory size. So this controller is best for lower order nonlinear systems which do not require high dimension networks. Also, when any information regarding to the dynamics is known, it is better to take advantage of the information and help to reduce the network size and achieve fast computation and make the real time control possible.

5.3 How to Choose a Controller Best for the Control Application

Figure 5-1, Figure 5-2 and Figure 5-3 summarize the experimental comparison results of the four control techniques. With the increasing complexity of the controller designs, the performance improves and the computation requirement increases. Combining with the analysis from section 5.2, we give the following guideline for choosing an appropriate control technique best for one's application need.

Depending on how much we know about the nonlinear dynamic system and what computation resources are available, we make choices of the control methods that are best for our applications.

If the system's nonlinear structure can be understood and modeled, we should choose the traditional control methodologies. If the nonlinearity is hard to analyse and difficult to model, and the fast computation resource is available, then we can choose the Gaussian network control to adapt to the unknown dynamics.

If the prior knowledge of the system nonlinearity can be obtained, there is no need to use the more costly connectionist approach. Instead, we can choose the PID control if the range of the control is small enough so that the nonlinearity within the system can be linearized.

If the range of the control motion is outside the linearizing limit, then robust control methods can be called upon. Sliding control is best for systems where the

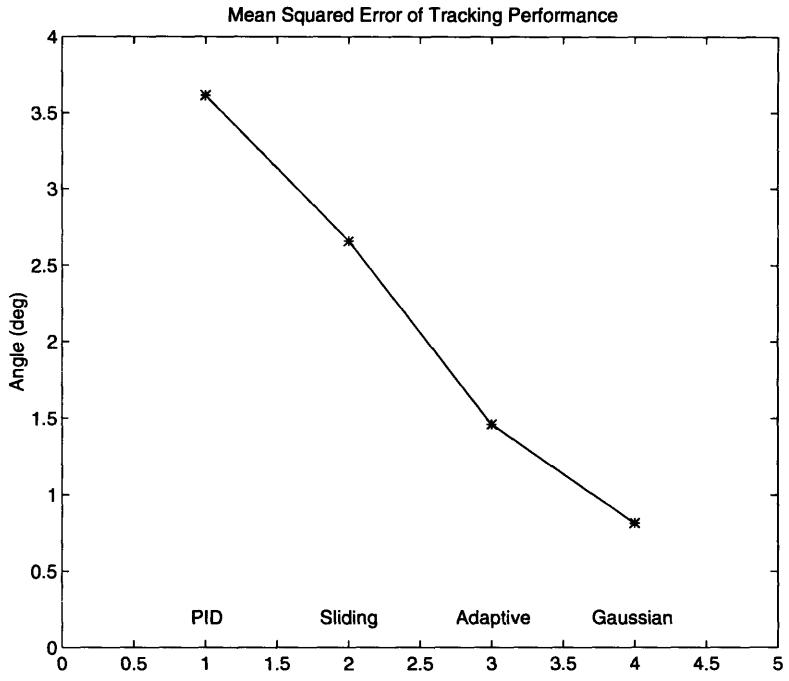

Figure 5-2: Mean squared error of the tracking performance

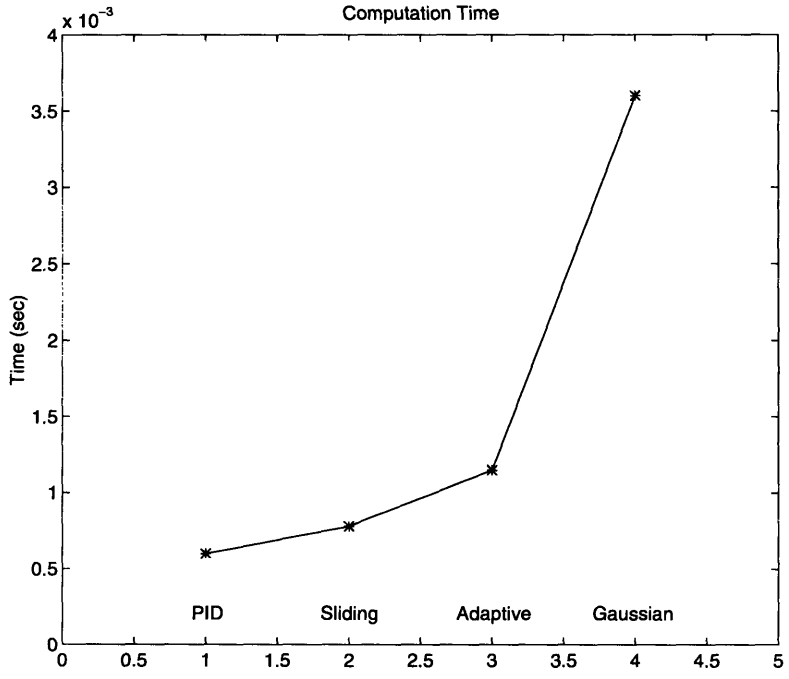

Figure 5-3: Computation time comparison

dynamics can be approximated and the estimation bounds available.

If the parameters involved in the nonlinear structure are hard to estimate, then adaptive control techniques can be used to adapt to the true parameter values and increase the performance tremendously.

Chapter 6

Summary and Recommendations for Future Work

6.1 Summary

The past decade has seen a dramatic increase in interest in neural networks systems. They are actively explored in psychology, signal processing, pattern classification, pattern recognition and optimization problems. There are also lots of interests growing in the control community to apply neural networks to nonlinear dynamic systems. Since control theory is a well-developed field with a large literature and solid mathematical foundation, instead of approaching neural networks as a blanket solution to control problems, we should make connections to existing control theory, illustrate their relationships to conventional and adaptive control techniques, and directly compare the new neural network approach to more traditional control system designs.

This thesis tries to study the connectionist controllers with the conventional nonlinear control methodologies, compare their performances on a nonlinear dynamic system, discuss their strength and weakness, and suggest choosing appropriate control techniques for different nonlinear control applications. Depending on how much we know about the nonlinear dynamic system and what computation resources are available, we can make choices of the control methods that are best for our applications.

The PID control is a linear control method with gains chosen based on the information of the plant and the desired closed-loop bandwidth. The results from simulation and experiment show that it needs just a rough estimate of the inertia term, with little knowledge about the nonlinearity. Depending on the significance of the nonlinearity and the bandwidth constraints, it can be implemented on real time control with little amount of computation and memory requirements. The tracking performance is not very good, and cannot be improved very much even if additional information about the nonlinearity is available. It's a good controller for simple control tasks on simple dynamic systems, and stability can be guaranteed for linear systems. When used on underwater vehicles, it can perform good only if the system dynamics is well known and linearized in a small region of operation, as in the case of constant speed heading control. Yet for complicated nonlinear dynamic system, performance is limited and no stability is guaranteed for closed-loop bandwidth.

Sliding control is a systematic nonlinear approach to nonlinear control problems. It requires information about the structure of the nonlinearity, but need only an estimation of the parameters involved and their estimation bounds. The tracking performance is improved and the stability of the overall system is maintained. There is a quantified trade off between the performance and the model precision. This approach can also be implemented into real time control, computation time is small. Overall, it's best for nonlinear systems with a good knowledge of the nonlinearity structure, but with just estimations of the parameters. When the underwater vehicles dynamics structure is known, sliding control can be effectively used for nonlinear controls like speed control during transient stages of vehicle motion.

The adaptive control technique used here is an extension to sliding control. It dynamically adapts to the unknown parameters of the nonlinear systems. Even if initial estimates are poor, the performance improves greatly after parameter adaptation. If the structure of the system is known, but the coefficients are hard to obtain, like the case with underwater vehicle's hydrodynamic forces, then adaptive sliding control can be a good choice. Like sliding control, the adaptive control can be called upon for nonlinear operation regions and its performance will be much better than

sliding control after the parameters have adapted to their real values.

Gaussian network control serves as a link between the theoretic control and the connectionist control. It requires very little information about the nonlinearity, namely its frequency contents and some upper bounds. Since the weight adjustment is determined using traditional Lyapunov theory, the overall system stability is assured and the tracking errors converge to a neighborhood of zero. Unlike other connectionist approach, the number of nodes within the Gaussian network can be decided by the desired performance and the a priori frequency content information about the nonlinear function to be approximated. It indeed needs more nodes inside the network in order to approximate the nonlinear function better, and consequently increases the computation time. Yet for low order nonlinear dynamic system, it still is the best choice when little information is available about the system dynamics. This control technique can be potentially used for autonomous underwater vehicles where unmodeled dynamics and disturbances are present in the operation.

The current results of the comparison is that when we have the ability to understand the nonlinear dynamic system's structure, the sliding controller and adaptive sliding controller are well established for such systems with good performance and assured overall system stability. When less information is available about the system nonlinearities, connectionist approaches can be applied to learn and adapt to these unknown structures.

6.2 Recommendations for Future Work

This thesis offers a preliminary comparison of some of the mostly used traditional nonlinear controllers and one connectionist approach. Experimental results suggest when and which controllers we should use depending on our prior knowledge of the nonlinearity and the computational resources available.

The current work is only a comparative study on a simple electro-mechanical system. It demonstrates the feasibility of the Gaussian network control, in addition to many other theoretical control methods. Real application of this promising controller

needs to be carried out so that its great advantages can be utilized for complex nonlinear systems like underwater vehicles, and help to enhance the autonomous control ability for AUVs.

Bibliography

[1] R. Cristi, F. A. Papoulias, and A. J. Healey. Adaptive sliding mode control of autonomous underwater vehicles in the dive plane. *IEEE Journal of Oceanic Engineering*, 15(3):152–160, July 1990.

[2] J. Feldman. DTNSRDC revised standard submarine equations of motion. Technical Report SPD-0393-09, DTNSRDC, June 1979.

[3] A. J. Healey and D. Lienard. Multivariable sliding mode control for autonomous diving and steering of unmanned underwater vehicles. *IEEE Journal of Oceanic Engineering*, 18(3):327–339, July 1993.

[4] S. J. Hills and D. R. Yoerger. A nonlinear sliding mode autopilot for unmanned undersea vehicles. In *Proceedings of OCEANS 94*, volume 3, pages 93–98, Brest, France, 13–16 September 1994. IEEE, New York, NY, USA.

[5] He Huang. Comparison of neural and control theoretic techniques for nonlinear dynamic systems. Master's thesis, Massachusetts Institute of Technology, Department of Mechanical Engineering and Department of Ocean Engineering, 1994.

[6] D. E. Humphreys. Hydrodynamic design and six-degree-of-freedom maneuvering evaluation of the naval underwater systems center unmanned underwater vehicle. A.R.A.P. Report 636, California Research and Technology, Inc, March 1989.

[7] A. T. Morrison III and D. R. Yoerger. Determination of the hydrodynamic parameters of an underwater vehicle during small scale, nonuniform, 1-dimensional

translation. In *Proceedings of OCEANS 93*, volume 2, pages 277–282, Victoria, BC, Canada, 18–21 October 1993. IEEE, New York, NY, USA.

[8] James Allen Mette Jr. Multivariable control of a submarine using the LQG/LTR method. Master's thesis, Massachusetts Institute of Technology, Dept. of Mechanical Engineering, 1985.

[9] J. D. Van Manen and P. Van Ossanen. *Principles of Naval Architecture.* Society of Naval Architects and Marine Engineers, Jersey City, N.J., second edition, 1988. Edward V. Lewis, editor.

[10] Richard J. Martin, Lena Valavani, and Michael Athans. Multivariable control of a submersible using the LQG/LTR design methodology. Technical Report 1548, Massachusetts Institute of Technology, Cambridge, Mass. : Laboratory for Information and Decision Systems, M.I.T., 1986.

[11] S. McMillan, D. E. Orin, and R. B. McGhee. Efficient dynamic simulation of an underwater vehicle with a robotic manipulator. *IEEE Transactions on Systems, Man and Cybernetics*, 25(8):1194–206, August 1995.

[12] D. A. Mindell, D. R. Yoerger, L. E. Freitag, L. L. Whitcomb, and R. L. Eastwood. JasonTalk: a standard ROV vehicle control system. In *Proceedings of OCEANS 93*, volume 3, pages 253–258, Victoria, BC, Canada, 18–21 October 1993. IEEE, New York, NY, USA.

[13] Katsuhiko Ogata. *Modern Control Engineering.* Prentice-Hall Electrical Engineering Series. Prentice-Hall, Englewood Cliffs, NJ, 1970.

[14] F. A. Papoulias, R. Cristi, D. Marco, and A. J. Healey. Modeling, sliding mode control design, and visual simulation of auv dive plane dynamic response. In *Proceedings of the 6th International Symposium on Unmanned Untethered Submersible Technology*, pages 536–547, Durham, NH, USA, 12–14 June 1989. IEEE, University of New Hampshire.

[15] D. E. Rumelhart, J. L. McCelland, and the PDP Research Group. *Parallel Distributed Processing*, volume 1. The MIT Press, Cambridge, MA, 1986.

[16] Robert M. Sanner and Jean-Jacques E. Slotine. Gaussian networks for direct adaptive control. *IEEE Transaction of Neural Networks*, November 1992.

[17] Jean-Jacques E. Slotine and J. A. Coetsee. Adaptive sliding controller synthesis for non-linear systems. *International Journal of Control*, 43(6):1631–1651, 1986.

[18] Jean-Jacques E. Slotine and Weiping Li. *Applied Nonlinear Control.* Prentice-Hall, Englewood Cliffs, NJ, 1991.

[19] Edward Ramsey Snow. The load/deflection behavior of pretensioned cable-pulley transmission mechanisms. Master's thesis, Massachusetts Institute of Technology, Department of Mechanical Engineering, 1994.

[20] John G. Cooke V. Incorporating thruster dynamics in the control of an underwater vehicle. Master's thesis, Massachusetts Institute of Technology, Department of Ocean Engineering, September 1989.

[21] Gregory M. Vaughn. Hybrid state estimators for the control of remotely operated underwater vehicles. Master's thesis, Massachusetts Institute of Technology, September 1988.

[22] K. P. Venugopal, R. Sudhakar, and A. S. Pandya. On-line learning control of autonomous underwater vehicles using feedforward neural networks. *IEEE Journal of Oceanic Engineering*, 17(4):308–319, October 1992.

[23] K. W. Watkinson, N. S. Smith, and D. S. Henn. User's manual for TRJUUV: A six-degree-of-freedom simulation program for unmanned underwater vehicles. A.R.A.P. Report 635, California Research and Technology, Inc, June 1989.

[24] L. L. Whitcomb and D. R. Yoerger. A new distributed real-time control system for the Jason underwater robot. In *Proceedings of IEEE International conference on intelligent robots and systems*, Yokohama, Japan, July 1993. IEEE.

[25] Louis L. Whitcomb and Dana R. Yoerger. Comparative experiments in the dynamics and model-based control of marine thrusters. In *Proceedings of OCEANS 95*, San Diego, CA, 9–12 October 1995. IEEE.

[26] David A. White and Donald A. Sofge, editors. *Handbook of Intelligent Control.* Van Nostrand Reinhold, New York, NY, 1992.

[27] D. R. Yoerger, A. M. Bradley, and B. B. Walden. The autonomous benthic explorer. *Unmanned Systems*, 9(2):17–23, 1991.

[28] D. R. Yoerger and J.-J.E. Slotine. Task resolved motion control of vehicle-manipulator systems. *International Journal of Robotics and Automation*, 2(3):144–151, 1987.

[29] Dana R. Yoerger and Jean-Jacques E. Slotine. Adaptive sliding control of an experimental underwater vehicle. In *Proceedings of 1991 IEEE International Conference on Robotics and Automation*, volume 3, pages 2746–2751, Sacramento, CA, USA, 9–11 April 1991. IEEE Comput. Soc. Press, Los Alamitos, CA, USA.

[30] J. Yuh. A neural net controller for underwater robotic vehicles. *IEEE Journal of Oceanic Engineering*, 15(3):161–166, July 1990.

[31] J. Yuh and R. Lakshmi. Design of an intelligent control system for remotely operated vehicles. In *Proceedings of IEEE Conference on Neural Networks for Ocean Engineering*, pages 151–160, 15–17 August 1991.

[32] J. Yuh and R. Lakshmi. An intelligent control system for remotely operated vehicles. *IEEE Journal of Oceanic Engineering*, 18(1):55–62, January 1993.