ADAPTIVE APPROXIMATION BASED CONTROL
Unifying Neural, Fuzzy and Traditional Adaptive Approximation Approaches

Jay A. Farrell
University of California Riverside

Marios M. Polycarpou
University of Cyprus and University of Cincinnati
ADAPTIVE APPROXIMATION BASED CONTROL
This Page Intentionally Left Blank
ADAPTIVE APPROXIMATION BASED CONTROL
Unifying Neural, Fuzzy and Traditional Adaptive Approximation Approaches

Jay A. Farrell
University of California Riverside

Marios M. Polycarpou
University of Cyprus and University of Cincinnati

WILEY-INTERSCIENCE
A JOHN WILEY & SONS, INC., PUBLICATION
To our families and friends.
This Page Intentionally Left Blank
CONTENTS

2.3 Function Approximation
   2.3.1 Offline (Batch) Function Approximation 31
   2.3.2 Adaptive Function Approximation 33

2.4 Approximator Properties
   2.4.1 Parameter (Non)Linearity 39
   2.4.2 Classical Approximation Results 43
   2.4.3 Network Approximators 46
   2.4.4 Nodal Processors 48
   2.4.5 Universal Approximator 50
   2.4.6 Best Approximator Property 52
   2.4.7 Generalization 54
   2.4.8 Extent of Influence Function Support 56
   2.4.9 Approximator Transparency 65
   2.4.10 Haar Conditions 66
   2.4.11 Multivariable Approximation by Tensor Products 67

2.5 Summary 68
2.6 Exercises and Design Problems 69

3 Approximation Structures 71

3.1 Model Types
   3.1.1 Physically Based Models 72
   3.1.2 Structure (Model) Free Approximation 73
   3.1.3 Function Approximation Structures 74

3.2 Polynomials
   3.2.1 Description 75
   3.2.2 Properties 77

3.3 Splines
   3.3.1 Description 78
   3.3.2 Properties 83

3.4 Radial Basis Functions
   3.4.1 Description 84
   3.4.2 Properties 86

3.5 Cerebellar Model Articulation Controller
   3.5.1 Description 88
   3.5.2 Properties 89

3.6 Multilayer Perceptron
   3.6.1 Description 93
   3.6.2 Properties 95

3.7 Fuzzy Approximation
   3.7.1 Description 96
   3.7.2 Takagi–Sugeno Fuzzy Systems 104
   3.7.3 Properties 105
3.8 Wavelets
  3.8.1 Multiresolution Analysis (MRA)
  3.8.2 MRA Properties
3.9 Further Reading
3.10 Exercises and Design Problems

4 Parameter Estimation Methods

4.1 Formulation for Adaptive Approximation
  4.1.1 Illustrative Example
  4.1.2 Motivating Simulation Examples
  4.1.3 Problem Statement
  4.1.4 Discussion of Issues in Parametric Estimation
4.2 Derivation of Parametric Models
  4.2.1 Problem Formulation for Full-State Measurement
  4.2.2 Filtering Techniques
  4.2.3 SPR Filtering
  4.2.4 Linearly Parameterized Approximators
  4.2.5 Parametric Models in State Space Form
  4.2.6 Parametric Models of Discrete-Time Systems
  4.2.7 Parametric Models of Input-Output Systems
4.3 Design of Online Learning Schemes
  4.3.1 Error Filtering Online Learning (EFOL) Scheme
  4.3.2 Regressor Filtering Online Learning (RFOL) Scheme
4.4 Continuous-Time Parameter Estimation
  4.4.1 Lyapunov-Based Algorithms
  4.4.2 Optimization Methods
  4.4.3 Summary
4.5 Online Learning: Analysis
  4.5.1 Analysis of LIP EFOL Scheme with Lyapunov Synthesis Method
  4.5.2 Analysis of LIP RFOL Scheme with the Gradient Algorithm
  4.5.3 Analysis of LIP RFOL Scheme with RLS Algorithm
  4.5.4 Persistency of Excitation and Parameter Convergence
4.6 Robust Learning Algorithms
  4.6.1 Projection Modification
  4.6.2 $\sigma$-Modification
  4.6.3 $\epsilon$-Modification
  4.6.4 Dead-Zone Modification
  4.6.5 Discussion and Comparison
4.7 Concluding Summary
4.8 Exercises and Design Problems
5 Nonlinear Control Architectures

5.1 Small-Signal Linearization
   5.1.1 Linearizing Around an Equilibrium Point
   5.1.2 Linearizing Around a Trajectory
   5.1.3 Gain Scheduling

5.2 Feedback Linearization
   5.2.1 Scalar Input-State Linearization
   5.2.2 Higher-Order Input-State Linearization
   5.2.3 Coordinate Transformations and Diffeomorphisms
   5.2.4 Input-Output Feedback Linearization

5.3 Backstepping
   5.3.1 Second Order System
   5.3.2 Higher Order Systems
   5.3.3 Command Filtering Formulation

5.4 Robust Nonlinear Control Design Methods
   5.4.1 Bounding Control
   5.4.2 Sliding Mode Control
   5.4.3 Lyapunov Redesign Method
   5.4.4 Nonlinear Damping
   5.4.5 Adaptive Bounding Control

5.5 Adaptive Nonlinear Control

5.6 Concluding Summary

5.7 Exercises and Design Problems

6 Adaptive Approximation: Motivation and Issues

6.1 Perspective for Adaptive Approximation Based Control

6.2 Stabilization of a Scalar System
   6.2.1 Feedback Linearization
   6.2.2 Small-Signal Linearization
   6.2.3 Unknown Nonlinearity with Known Bounds
   6.2.4 Adaptive Bounding Methods
   6.2.5 Approximating the Unknown Nonlinearity
   6.2.6 Combining Approximation with Bounding Methods
   6.2.7 Combining Approximation with Adaptive Bounding Methods
   6.2.8 Summary

6.3 Adaptive Approximation Based Tracking
   6.3.1 Feedback Linearization
   6.3.2 Tracking via Small-Signal Linearization
   6.3.3 Unknown Nonlinearities with Known Bounds
   6.3.4 Adaptive Bounding Design
   6.3.5 Adaptive Approximation of the Unknown Nonlinearities
8.3.6 Simulation Analysis 367
8.3.7 Conclusions 371
8.4 Aircraft Notation 371

Appendix A: Systems and Stability Concepts 377
A.1 Systems Concepts 377
A.2 Stability Concepts 379
   A.2.1 Stability Definitions 379
   A.2.2 Stability Analysis Tools 381
   A.2.3 Strictly Positive Real Transfer Functions 391
A.3 General Results 392
A.4 Trajectory Generation Filters 394
A.5 A Useful Inequality 397
A.6 Exercises and Design Problems 398

Appendix B: Recommended Implementation and Debugging Approach 399

References 401

Index 417
During the last few years there have been significant developments in the control of highly uncertain, nonlinear dynamical systems. For systems with parametric uncertainty, adaptive nonlinear control has evolved as a powerful methodology leading to global stability and tracking results for a class of nonlinear systems. Advances in geometric nonlinear control theory, in conjunction with the development and refinement of new techniques, such as the backstepping procedure and tuning functions, have brought about the design of control systems with proven stability properties. In addition, there has been a lot of research activity on robust nonlinear control design methods, such as sliding mode control, Lyapunov redesign method, nonlinear damping, and adaptive bounding control. These techniques are based on the assumption that the uncertainty in the nonlinear functions is within some known, or partially known, bounding functions.

In parallel with developments in adaptive nonlinear control, there has been a tremendous amount of activity in neural control and adaptive fuzzy approaches. In these studies, neural networks or fuzzy approximators are used to approximate unknown nonlinearities. The input/output response of the approximator is modified by adjusting the values of certain parameters, usually referred to as weights. From a mathematical control perspective, neural networks and fuzzy approximators represent just two classes of function approximators. Polynomials, splines, radial basis functions, and wavelets are examples of other function approximators that can be used—and have been used—in a similar setting. We refer to such approximation models with adaptivity features as adaptive approximators, and control methodologies that are based on them as adaptive approximation based control.

Adaptive approximation based control encompasses a variety of methods that appear in the literature: intelligent control, neural control, adaptive fuzzy control, memory-based control, knowledge-based control, adaptive nonlinear control, and adaptive linear control.
Researchers in these fields have diverse backgrounds: mathematicians, engineers, and computer scientists. Therefore, the perspective of the various papers in this area is also varied. However, the objective of the various practitioners is typically similar: to design a controller that can be guaranteed to be stable and achieve a high level of control performance for systems that contain poorly modeled nonlinear effects, or the dynamics of the system change during operation (for example, due to system faults). This objective is achieved by adaptively developing an approximating function to compensate the nonlinear effects during the operation of the system.

Many of the original papers on neural or adaptive fuzzy control were motivated by such concepts as ease of use, universal approximation, and fault tolerance. Often, ease of use meant that researchers without a control or systems background could experiment with and often succeed at controlling certain dynamics systems, at least in simulation. The rise of interest in the neural and adaptive fuzzy control approaches occurred at a time when desktop computers and dynamic simulation tools were becoming sufficiently cheap at reasonable levels of performance to support such research on a wide basis.

However, prior to application on systems of high economic value, the control system designer must carefully consider any new approach within a sound analytical framework that allows rigorous analysis of conditions for stability and robustness. This approach opens a variety of questions that have been of interest to various researchers: What properties should the function approximator have? Are certain families of approximators superior to others? How should the parameters of the approximator be estimated? What can be guaranteed about the properties of the signals within the control system? Can the stability of the approximator parameters be guaranteed? Can the convergence of the approximator parameters be guaranteed? Can such control systems be designed to be robust to noise, disturbances, and unmodeled effects. Can this approach handle significant changes in the dynamics due to, for example, a system failure. What types of nonlinear dynamic systems are amenable to the approach? What are the limitations? The objective of this textbook is to provide readers with a framework for rigorously considering such questions.

Adaptive approximation based control can be viewed as one of the available tools that a control designer should have in her/his control toolbox. Therefore, it is desirable for the reader not only to be able to apply, for example, neural network techniques to a certain class of systems, but more importantly to gain enough intuition and understanding about adaptive approximation so that she/he knows when it is a useful tool to be used and how to make necessary modifications or how to combine it with other control tools, so that it can be applied to a system that has not be encountered before.

The book has been written at the level of a first-year graduate student in any engineering field that includes an introduction to basic dynamic systems concepts such as state variables and Laplace transforms. We hope that this book has appeal to a wide audience. For use as a graduate text, we have included exercises, examples, and simulations. Sufficient detail is included in examples and exercises to allow students to replicate and extend results. Simulation implementation of the methods developed herein is a virtually necessary component of understanding implications of the approach. The book extensively uses ideas from stability theory. The advantage of this approach is that the adaptive law is derived based on the Lyapunov synthesis method and therefore the stability properties of the closed-loop system are more readily determined. Therefore, an appendix has been included as an aid to readers who are not familiar with the ideas of Lyapunov stability analysis. For theoretically oriented readers, the book includes complete stability analysis of the methods that are presented.
Organization. To understand and effectively implement adaptive approximation based control systems that have guaranteed stability properties, the designer must become familiar with concepts of dynamic systems, stability theory, function approximation, parameter estimation, nonlinear control methods, and the mechanisms to apply these various tools in a unified methodology.

Chapter 1 introduces the idea of adaptive approximation for addressing unknown nonlinear effects. This chapter includes a simple example comparing various control approaches and concludes with a discussion of components of an adaptive approximation based control system with pointers to the locations in the text where each topic is discussed.

Function approximation and data interpolation have long histories and are important fields in their own right. Many of the concepts and results from these fields are important relative to adaptive approximation based control. Chapter 2 discuss various properties of function approximators as they relate to adaptive function approximation for control purposes. Chapter 3 presents various function approximation structures that have been considered for implementation of adaptive approximation based controllers. All of the approximators of this chapter are presented using a single unifying notation. The presentation includes a comparative discussion of the approximators relative to the properties presented in Chapter 2.

Chapter 4 focuses on issues related to parameter estimation. First we study the formulation of parametric models for the approximation problem. Then we present the design of online learning schemes; and finally, we derive parameter estimation algorithms with certain stability and robustness properties. The parameter estimation problem is formulated in a continuous-time framework. The chapter includes a discussion of robust parameter estimation algorithms, which will prove to be critical to the design of stable adaptive approximation based control systems.

Chapter 5 reviews various nonlinear control system design methodologies. The objective of this chapter is to introduce the methods, analysis tools, and key issues of nonlinear control design. The chapter begins with a discussion of small-signal linearization and gain scheduling. Then we focus on feedback linearization and backstepping, which are two of the key design methods for nonlinear control design. The chapter presents a set of robust nonlinear control design techniques. These methods include bounding control, sliding mode control, Lyapunov redesign method, nonlinear damping, and adaptive bounding. Finally, we briefly study the adaptive nonlinear control methodology. For each approach we present the basic method, discuss necessary theoretical ideas related to each approach, and discuss the effect (and accommodation) of modeling error.

Chapters 6 and 7 bring together the ideas of Chapters 1-5 to design and analyze control systems using adaptive approximation to compensate for poorly modeled nonlinear effects. Chapter 6 considers scalar dynamic systems. The intent of this chapter is to allow a detailed discussion of important issues without the complications of working with higher numbers of state variables. The ideas, intuition, and methods developed in Chapter 6 are important to successful applications to higher order systems. Chapter 7 will augment feedback linearization and backstepping with adaptive approximation capabilities to achieve high-performance tracking for systems with significant unmodeled nonlinearities. The presentation of each approach includes a rigorous Lyapunov analysis.

Chapter 8 presents detailed design and analysis of adaptive approximation based controllers applied to fixed-wing aircraft. We study two control situations. First, an angular rate controller is designed and analyzed. This controller is applicable in piloted aircraft applications where the stick motion of the pilot is processed into body-frame angular rate commands. Then we develop a full vehicle controller suitable for uninhabited air vehicles
PREFACE

(UAVs). The control design is based on the approximation based backstepping methodology.

Acknowledgments. The authors would like to thank the various sponsors that have supported the research that has resulted in this book: the National Science Foundation (Paul Werbos), Air Force Wright-Patterson Laboratory (Mark Mears), Naval Air Development Center (Marc Steinberg), and the Research Promotion Foundation of Cyprus. We would like to thank our current and past employers who have directly and indirectly enabled this research: University of California, Riverside; University of Cyprus; University of Cincinnati; and Draper Laboratory. In addition, we wish to acknowledge the many colleagues, collaborators, and students who have contributed to the ideas presented herein, especially: P. Antsaklis, W. L. Baker, J.-Y. Choi, M. Demetriou, S. Ge, J. Harrison, P. A. Ioannou, H. K. Khalil, P. Kokotovic, F. L. Lewis, D. Liu, M. Mears, A. N. Michel, A. Minai, J. Nakanishi, K. Narendra, C. Panayiotou, T. Parisini, K. M. Passino, T. Samad, S. Schaal, M. Sharma, J.-J. Slotine, E. Sontag, G. Tao, A. Vemuri, H. Wang, S. Weaver, Y. Yang, X. Zhang, Y. Zhao, and P. Zufiria. Finally, we would like to thank our families for their constant support and encouragement throughout the long period that it took for this book to be completed.

Jay A. Farrell
Marios M. Polycarpou

Riverside, California and Nicosia, Cyprus
(10 hours time difference)
July 2005
CHAPTER I

INTRODUCTION

This book presents adaptive function estimation and feedback control methodologies that develop and use approximations to portions of the nonlinear functions describing the system dynamics while the system is in online operation. Such methodologies have been proposed and analyzed under a variety of titles: neural control, adaptive fuzzy control, learning control, and approximation-based control. A primary objective of this text is to present the methods systematically in a unifying framework that will facilitate discussion of underlying properties and comparison of alternative techniques.

This introductory chapter discusses some fundamental issues such as: (i) motivations for using adaptive approximation-based control; (ii) when adaptive approximation-based control methods are appropriate; (iii) how the problem can be formulated; and (iv) what design decisions are required. These issues are illustrated through the use of a simple simulation example.

1.1 SYSTEMS AND CONTROL TERMINOLOGY

Researchers interested in this area come from a diverse set of backgrounds other than control; therefore, we start with a brief review of terminology standard to the field of control systems, as depicted in Figure 1.1. The plant is the system to be controlled. The plant will by modeled herein by a typically nonlinear set of ordinary differential equations. The plant model is assumed to include the actuator and sensor models. The control system is designed to achieve certain control objectives. As indicated in Figure 1.1, the inputs to the control system include the reference input $y_c(t)$ (which is possibly passed through
a prefilter to yield a smoother function \( y_d(t) \) and its first \( r \) time derivatives \( y_d^{(i)}(t) \) for \( i = 1, \ldots, r \) and a set of measurable plant outputs \( y(t) \). The control system processes its inputs to produce the control system output \( u(t) \) that is applied to the plant actuators to affect the desired change in the plant output. The control system output \( u(t) \) is sometimes referred to as control signal or plant input. Figure 1.1 depicts as a block diagram a standard closed-loop control system configuration.

The control system determines the stability of the closed-loop system and the response to disturbances \( d(t) \) and initial condition errors. A disturbance is any unmodeled physical effect on the plant state, usually caused by the environment. A disturbance is distinct from measurement noise. The former directly and physically affects the system to be controlled. The latter affects the measurement of the physical quantity without directly affecting the physical quantity. The physical quantity may be indirectly affected by the noise through the feedback control process.

Control design typically distinguishes regulation from tracking objectives. Regulation is concerned with designing a control system to achieve convergence of the system state, with a desirable transient response, from any initial condition within a desired domain of attraction, to a single operating point. In this case, the signal \( y_c(t) \) is constant. Tracking is concerned with the design of a control system to cause the system output \( y(t) \) to converge to and accurately follow the signal \( y_d(t) \). Although the input signal \( y_c(t) \) to a tracking controller could be a constant, it typically is time-varying in a manner that is not known at the time that the control system is designed. Therefore, the designer of a tracking controller must anticipate that the plant state may vary significantly on a persistent basis. It is reasonable to expect that the designer of the open-loop physical system and the designer of the feedback control system will agree on an allowable range of variation of the state of the system. Herein, we will denote this operating envelope by \( D \). The designer of the physical system ensures safe operation when the state of the system is in \( D \). The designer of the controller must ensure that the state the system remains in \( D \). Implicitly it is assumed that the state required to track \( y_d \) lies entirely in \( D \).

To illustrate the control terminology let us consider the example of a simple cruise control system for automobiles. In this case, the control objective is to make the vehicle follow a desired speed profile \( y_c(t) \), which is set by the driver. The measured output \( y(t) \) is the sensed vehicle speed and the control system output \( u(t) \) is the throttle angle and/or fuel injection rate. The disturbance \( d(t) \) may arise due to the wind or road incline. In addition to disturbances, which are external factors influencing the state, there may also be modeling errors. In the cruise control example, the plant model describes the effect of changing the throttle angle on the actual vehicle speed. Hence, modeling errors may arise from simplifications or inaccuracies in characterizing the effect of changing the throttle angle on the vehicle speed. Modeling errors (especially nonlinearities), whether they arise due
to inaccuracies or intentional model simplifications, constitute one of the key motivations for employing adaptive approximation-based control, and thus are crucial to the techniques developed in this book.

In general, the objectives of a control system design are:

1. to stabilize the closed-loop system;
2. to achieve satisfactory reference input tracking in transient and at steady state;
3. to reduce the effect of disturbances;
4. to achieve the above in spite of modeling error;
5. to achieve the above in spite of noise introduced by sensors required to implement the feedback mechanism.

Introductory textbooks in control systems provide linear-based design and analysis techniques for achieving the above objectives and discuss some basic robustness and implementation issues [61, 66, 86, 140]. The theoretical foundations of linear systems analysis and design are presented in more advanced textbooks (see, for example, [10, 19, 39, 130]), where issues such as controllability, observability, and model reduction are examined.

1.2 NONLINEAR SYSTEMS

Most dynamic systems encountered in practice are inherently nonlinear. The control system design process builds on the concept of a model. Linear control design methods can sometimes be applied to nonlinear systems over limited operating regions (i.e., $\mathcal{D}$ is sufficiently small), through the process of small-signal linearization. However, the desired level of performance or tracking problems with a sufficiently large operating region $\mathcal{D}$ may require in which the nonlinearities be directly addressed in the control system design. Depending on the type of nonlinearity and the manner that the nonlinearity affects the system, various nonlinear control design methods are available [121, 134, 159, 234, 249, 279]. Some of these methods are reviewed in Chapter 5.

Nonlinearity and model accuracy directly affect the achievable control system performance. Nonlinearity can impose hard constraints on achievable performance. The challenge of addressing nonlinearities during the control design process is further complicated when the description of the nonlinearities involves significant uncertainty. When portions of the plant model are unknown or inaccurately defined, or they change during operation, the control performance may need to be severely limited to ensure safe operation. Therefore there is often an interest to improve the model accuracy. Especially in tracking applications this will typically necessitate the use of nonlinear models. The focus of this text is on adaptively improving models of nonlinear effects during online operation.

In such applications the level of achievable performance may be enhanced by using adaptive function approximation techniques to increase the accuracy of the model of the nonlinearities. Such adaptive approximation-based control methods include the popular areas of adaptive fuzzy and neural control. This chapter introduces various issues related to adaptive approximation-based control. This introductory discussion will direct the reader to the appropriate sections of the text where more detailed discussion of each issue can be found.
1.3 FEEDBACK CONTROL APPROACHES

To introduce the concept of adaptive approximation-based control, consider the following example, where the objective is to control the dynamic system

\[ \dot{y}(t) = f(y(t)) + g(y(t))u(t) \quad (1.1) \]

in a manner such that \( y(t) \) accurately tracks an externally generated reference input signal \( y_d(t) \). Therefore, the control objective is achieved if the tracking error \( \dot{y}(t) = y(t) - y_d(t) \) is forced to zero. The performance specification is for the closed-loop system to have a rate of convergence corresponding to a linear system with a dominant time constant \( \tau \) of about 5.0 s. With this time constant, tracking errors due to disturbances or initial conditions should decay to zero in approximately \( 15 \) s (\( \approx 3\tau \)). The system is expected to normally operate within \( y \in [20, 60] \), but may safely operate on the region \( D = \{ y \in [0, 100] \} \). Of course, all signals in the controller and plant must remain bounded during operation.

However, the plant model is not completely accurate. The best model available to the control system designer is given by

\[ \dot{y} = f_o(y) + g_o(y)u, \quad (1.2) \]

where \( f_o(y) = -y \) and \( g_o(y) = 1.0 + 0.3y \). The actual system dynamics are not known or available to the designer. For implementation of the following simulation results, the actual dynamics will be

\[ f(y) = -1 - 0.01y^2 \]
\[ g(y) = \frac{40(y + 1)}{40 + y}. \]

Therefore, there exists significant error between the design model and the actual dynamics over the desired domain of operation.

This section will consider four alternative control system design approaches. The example will allow a concrete, comparative discussion, but none of the designs have been optimized. The objective is to highlight the similarities, distinctions, complexity, and complicating factors of each approach. The details of each design have been removed from this discussion so as not to distract from the main focus. The details are included in the problem section of this chapter to allow further exploration. These methodologies and various others will be analyzed in substantially greater detail throughout the remainder of the book.

1.3.1 Linear Design

Given the design model and performance specification, the objective in this subsection is to design a linear controller for the system

\[ \dot{y}(t) = h(y(t), u(t)) = -y(t) + (1.0 + 0.3y(t))u(t) \quad (1.3) \]

so that the linearized closed-loop system is stable (stability concepts are reviewed in Appendix A) and has the desired tracking error convergence rate. This controller is designed based on the idea of small-signal linearization and is approximate, even relative to the model. Section 1.3.3 will consider feedback linearization, which is a nonlinear design approach that exactly linearizes the model using the feedback control signal.
For the scalar system \( \dot{y} = h(y, u) \), an operating point is a pair of real numbers \((y^*, u^*)\) such that \( h(y^*, u^*) = 0 \). If \( y = y^* \) and \( u = u^* \), then \( \dot{y} = 0 \). In a more general setting, the designer may need to linearize around a time-varying nominal trajectory \((y^*(t), u^*(t))\). Note that operating points may be stable or unstable (see the discussion in Appendix A). An operating point analysis only indicates the values of \( y \) at which it is possible, by appropriate choice of \( u \), for the system to be in steady state. For our example, the set of operating points is defined by \((y^*, u^*)\) such that

\[
u^* = \frac{y^*}{1 + 0.3y^*}.
\]

Therefore, the design model indicates that the system can operate at any \( y \in \mathcal{D} \).

The operating point analysis does not indicate how \( u(t) \) should be selected to get convergence to any particular operating point. Convergence to a desired operating point is an objective for the control system design. In a linear control design, the best available model is linearized around an operating point and a linear controller is designed for that linearized model. If we choose the operating point \((y^*, u^*) = (40, \frac{40}{13})\) as the design point, then the linearized dynamics are (see Exercise 1.1)

\[
\delta \dot{y} = -\frac{1}{13} \delta y + 13 \delta u,
\]

where \( \delta y = y - 40 \) and \( \delta u = u - \frac{40}{13} \). The linear controller

\[
U(s) = \frac{40}{13} - \frac{0.2(s + \frac{1}{13})}{13s}
\]

used with the design model results in a stable system that achieves the specification at \( y^* = 40 \). In the above, \( s \) is the Laplace variable, \( U(s) \) denotes the Laplace transform of \( u(t) \), \( \bar{y}(t) = y(t) - y_d(t) \), and \( y_d(t) \) is the reference input. Of course, \( \mathcal{D} \) is large enough that a linear controller designed to achieve the specification at one operating point will probably not achieve the specification at all operating points in \( \mathcal{D} \) or for \( y_d(t) \) varying with time over the region \( \mathcal{D} \).

Figure 1.2 shows the performance using the linear controller of eqn. (1.4) for a series of amplitude step inputs changing between \( y_d = 20 \) and \( y_d = 60 \). Note that the response exhibits two different convergence rates indicated by \( \tau_1 \) and \( \tau_2 \). One is significantly slower than the desired 5 s. Therefore, the linear controller does not operate as designed. There are two reasons for this. First, there is significant error between the design model and the actual dynamics of the system. Second, an inherent assumption of linear design is that the linear controller will only be used in a reasonably small neighborhood of the operating point for which the controller was designed. The degree of reasonableness depends on the nonlinear system of interest. For these two reasons, the actual linearized dynamics at the two points \( y^* = 20 \) and \( y^* = 60 \) are distinct from the linearized dynamics of the design model at the design point \( y^* = 40 \). The design methodology to determine eqn. (1.4) relied on cancelling the pole of the linearized dynamics. With modeling error, even for a linear system, the pole is not cancelled; instead, there are two poles. One near the desired pole and one near the origin. The second pole is dominant and yields the slowly converging error dynamics.

Improved performance using linear methods could be achieved by various methods. First, additional modeling efforts could decrease the error between the actual dynamics and the design model, but may be expensive and will not solve the problem of operating far from the linearization point. Second, high gain control will decrease the sensitivity to
modeling error, but will result in a higher bandwidth closed-loop system as well as a large control effort. Third, gain scheduling methods (although not truly linear) address the issue of limiting the use of a linear controller to a region about its design point by switching between a set of linear controllers as a function of the system state. Each linear controller is designed to meet the performance specification (for the design model) on a small region of operation $D_i$. The regions $D_i$ are defined such that they cover the region of operation $D$ (i.e., $D \subseteq \bigcup_{i=1}^{N} D_i$). Gain scheduling a set of linear controllers does not address the issue of error between the actual system and the design model.

1.3.2 Adaptive Linear Design

Through linearization, the dynamics near a fixed operating point $(y^*, u^*)$ are approximated by

$$\dot{y}(t) = a^* + b^* y(t) + c^* u(t),$$

(1.5)

where $a^*$, $b^*$, and $c^*$ are parameters that depend on $(y^*, u^*)$. In one possible adaptive control approach, the control law is

$$u = \frac{1}{c} (-a - b y + \dot{y}_d + 0.2(y_d - y)), \quad (1.6)$$

where $y_d \in C^1(D)$ (i.e., the first derivative of $y_d$ exists and is continuous within the region $D$), and $a$, $b$, $c$ are parameter estimates of $a^*$, $b^*$, and $c^*$, respectively. Note that if $(a, b, c) = (a^*, b^*, c^*)$, then exact cancellation occurs and the resulting error dynamics are

$$\dot{\tilde{y}} = -0.2\tilde{y},$$
where \( \tilde{y} = y - y_d \). Therefore, the closed-loop error dynamics (with perfect modeling) achieve the performance specification. This closed-loop system has a time constant for rejecting disturbances and initial condition errors of 5.0 s, even though the feedforward term in eqn. (1.6) (i.e., \( \frac{1}{2} \tilde{y}_d \)) will allow the system to track faster changes in the commanded input.

The differentiability constraint on \( y_d(t) \) will be enforced by passing the reference input \( y_c(t) \) through the first-order low pass prefilter

\[
\frac{Y_d(s)}{Y_c(s)} = \frac{5}{s + 5},
\]

(1.7)

where \( Y_d(s) \), \( Y_c(s) \) denote the Laplace transforms of the time signals \( y_d(t) \) and \( y_c(t) \) respectively. Therefore,

\[
\dot{y}_d = -5(y_d - y_c),
\]

which has the same bounded and continuous properties as \( y_c \); whereas, the signal \( y_d \) will be bounded, continuous, and differentiable as long as \( y_c \) is bounded.

If \((a^*, b^*, c^*)\) are assumed to be unknown constant parameters, then the corresponding parameter estimates \((a, b, c)\) are derived from the following update laws

\[
\dot{a} = \gamma_1 \tilde{y},
\]

(1.8)

\[
\dot{b} = \gamma_2 \tilde{y} y,
\]

(1.9)

\[
\dot{c} = \gamma_3 \tilde{y} u,
\]

(1.10)

where \( \gamma_i > 0 \) are design constants representing the adaptive gain of each parameter estimate. For the following simulation we select \( \gamma_1 = \gamma_2 = \gamma_3 = 0.01 \). In practice, the update law for \( c(t) \) needs to be slightly modified in order to guarantee that \( c(t) \) does not approach zero, which would cause \( u(t) \) to become very large, or even infinite. The resulting error dynamic equations are

\[
\begin{align*}
\dot{\tilde{y}} &= -0.2 \tilde{y} + \bar{a} y + \bar{b} y + \bar{c} u, \quad (1.11) \\
\dot{\bar{a}} &= -\gamma_1 \tilde{y}, \quad (1.12) \\
\dot{\bar{b}} &= -\gamma_2 \tilde{y} y, \quad (1.13) \\
\dot{\bar{c}} &= -\gamma_3 \tilde{y} u, \quad (1.14)
\end{align*}
\]

where \( \bar{a} = a^* - a, \bar{b} = b^* - b, \bar{c} = c^* - c \). The adaptive control law is defined by eqns. (1.6) and (1.8–1.10). Note that this controller is not linear and that the controller implementation does not require knowledge of \( a^*, b^*, \) or \( c^* \) (other than the sign of \( c^* \)). If the above adaptive scheme is applied to the system model (1.5) (without noise, disturbances, and unmodeled states), it can be shown that the closed-loop system is stable, after some small modification to ensure that the parameter estimate \( c \) does not approach zero. It is noted that robustness issues are neglected at this point to simplify the presentation, but are addressed in Chapter 4.

Relative to (1.5), even if the tracking error \( \tilde{y}(t) \) goes to zero, the adaptive parameters \((a, b, c)\) may never converge to the "actual" parameters \((a^*, b^*, c^*)\). Convergence (or not) of the parameter estimation error to zero depends on the nature of the signal \( y_d(t) \). From eqn. (1.11), if \( \bar{a} + \bar{b} y + \bar{c} u = 0 \), then \( \tilde{y} \) will approach zero and parameter adaptation will stop. Since for any fixed values of \( y \) and \( u \), the equation \( \bar{a} + \bar{b} y + \bar{c} u = 0 \) defines a hyperplane of \((\bar{a}, \bar{b}, \bar{c})\) values, there are many values of the parameter estimates that can result in \( \tilde{y} = 0 \). The hyperplane is distinct for different \((y, u)\) and the only parameter estimates on all such
hyperplanes satisfy \((\tilde{a}, \tilde{b}, \tilde{c}) = (0, 0, 0)\). Therefore, convergence of the parameter estimates would require that \((y, u)\) change sufficiently in an appropriate sense, leading to the concept of *persistence of excitation* (see Chapter 4). An important fact to remember in the design of adaptive control systems is that convergence of the tracking error does not necessarily imply convergence (or even boundedness) of the parameter estimates.

Relative to (1.1), the parameters of (1.5) will be a function of the operating point (see Exercise 1.2). Each time that the operating point changes, the parameter estimates will adapt. If the operating point changed slowly, then \(a^*, b^*,\) and \(c^*\) could be considered as slowly time-varying. In such an approach, depending on the magnitude of the adaptive gains \(\gamma_i,\) the corresponding estimates may be able to change the adaptive parameters fast enough to maintain high performance. However, in this case the operating point would be restricted to vary slowly so that the control approach would behave properly. It is also important to note that increasing \(\gamma_i\) may create stability problems of the closed-loop system in the presence of measurement noise.

![Performance of the adaptive linear control system of eqn. (1.6) with the dynamic system of eqn. (1.1). The solid curve is \(y(t)\). The dashed curve is \(y_d(t)\).](image)

Figure 1.3: Performance of the adaptive linear control system of eqn. (1.6) with the dynamic system of eqn. (1.1). The solid curve is \(y(t)\). The dashed curve is \(y_d(t)\).

Figure 1.3 displays the performance of this adaptive control law (applied to the actual plant dynamics) for a reference input \(y_c(t)\) consisting of several step commands changing between 20 and 60. The average tracking error is significantly improved relative to the linear control system. However, immediately following each significant change in \(y_c(t)\), the tracking error is still large and oscillatory. Also, the estimated parameters that result in good performance at one operating point do not yield good performance at the other. Therefore, for this example, as the operating point is stepped back and forth, the estimated parameters step between the manifold of parameters (i.e., hyperplane) that yield good performance for \(y \approx 20\) and the manifold of parameters that yield good performance for \(y \approx 60,\) see Figure 1.4. This is obviously inefficient. It would be convenient if the designer could devise a method to, in some sense, store the model (e.g., estimated parameters) as
a function of the operating condition (e.g., \(y\)). Such ideas are the motivation for adaptive approximation-based control methods.

![Figure 1.4: Time evolution of the estimated parameters \(a(t), b(t), c(t)\) for the adaptive control system of eqn. (1.6) applied to the dynamic system of eqn. (1.1).](image)

### 1.3.3 Nonlinear Design

Given the design model of eqn. (1.2), the feedback linearizing control law is

\[
u(t) = \frac{1}{g_o(y(t))} \left( -f_o(y(t)) + \dot{y}_d(t) + K(y_d(t) - y(t)) \right).
\]

Combining the feedback linearizing control law with the design model and selecting \(K = 0.2\), yields the following nominal closed-loop dynamics

\[
\dot{\tilde{y}} = -0.2 \tilde{y},
\]

where \(\tilde{y} = y - y_d\). In contrast to the small signal linearization approach discussed in Section 1.3.1, the feedback linearizing controller is exact (for the design model). Therefore, the closed-loop tracking error dynamics based on the design model are asymptotically stable with the desired error convergence rate. Note also that (for the design model) the tracking is perfect in the sense that the initial condition \(\tilde{y}(0)\) decays to zero with the linear dynamics of eqn. (1.16) and is completely unaffected by changes in \(y_d(t)\).

However, since the design model is different from the actual plant dynamics, the performance of the actual closed-loop system will be affected by the modeling error. The dynamic model for the actual closed-loop system is

\[
\dot{\tilde{y}} = -0.2 \tilde{y} + (f(y) - f_o(y)) + (g(y) - g_o(y)) u.
\]

Accurate tracking will therefore depend on the accuracy of the design model.
Figure 1.5: Performance of the nonlinear feedback linearizing control system of eqn. (1.15) with the dynamic system of eqn. (1.1). The dotted curve is the commanded response. The solid curve is the actual response.

Figure 1.5 displays the performance of the actual system compensated by the nonlinear feedback linearizing control law of eqn. (1.15) as a solid line. Again, the commanded state $y_d$ (shown as a dashed line) and its derivative are generated by prefiltering $y_c$ (a sequence of step changes) using the filter of eqn. (1.7). The actual response moves in the appropriate direction at the start of each step command, but the modeling error is significant enough that the steady state tracking error for each step is quite large. Since the feedback linearizing controller attempts to cancel the plant dynamics and insert the desired tracking error dynamics, the approach is very sensitive to model error. As shown in eqn. (1.17), the tracking error is directly affected by the error in the design model. An objective of adaptive approximation-based control methods is to adaptively decrease the amount of model error by using online data.

In addition to improving the model accuracy, either offline or online, the performance of the control law of eqn. (1.15) could be improved in a variety of other ways. The control gains could be increased, but this would change the rate of the error convergence relative to the specification, increase the magnitude of the control signal, and increase the effect of noise on the control signal. The linear portion of the controller, currently $K(y_d(t) - y(t))$ could be modified. Also, additional robustifying terms could be added to the nonlinear control law to dominate the model error. These approaches will be described in Chapter 5.

The difference in performance exhibited in Figs. 1.2 and 1.5 is worthy of comment, because the performance of the linear control is better even though both are based on the same design model. The major reason for the difference in performance is that the nonlinear controller is static whereas the linear controller is dynamic in the sense that it includes an integrator. The role of an integrator in a stable controller is to drive the steady state error to zero (see Exercise 1.3).
1.3.4 Adaptive Approximation Based Design

The performance of the feedback linearizing control law was significantly affected by the error between the design model and the actual dynamics. It is therefore of interest to consider whether the data accumulated online, in the process of controlling the system, can be used to decrease the modeling error and improve the control performance. This subsection discusses one such approach. The goal is to motivate various design issues relevant to generic adaptive approximation-based approaches. The remainder of this chapter will expand on these design issues and point the reader to the sections of the book that provide an in-depth discussion of both the issues and alternative design approaches.

In one method to implement such an approach, the designer assumes that the actual system dynamics can be represented as

\[ \dot{y}(t) = f(y(t)) + g(y(t))u(t), \quad (1.18) \]

where \( f(y) = (\theta_f^*)^T \phi(y) \) and \( g(v) = (\theta_g^*)^T \phi(y) \) and \( \phi(y) \) is a vector of basis functions selected by the designer during the offline design phase. Since \( f \) and \( g \) are unknown, the parameters \( \theta_f \) and \( \theta_g \) are also unknown and will be estimated online. Therefore, we define the approximated functions \( \hat{f}(y) = \theta_f^T \phi(y) \) and \( \hat{g}(y) = \theta_g^T \phi(y) \), where \( \theta_f \) and \( \theta_g \) are parameter vectors that will be estimated using the online data. One approach to using the design model (i.e., \( f_o \) and \( g_o \) of (1.2)) is to initialize the parameter vector estimates.

The adaptive feedback linearizing control law

\[ u = \frac{1}{\tilde{g}(y)} \left( -\tilde{f}(y) + \dot{y} + 0.2 \,(y_d - y) \right) \quad (1.19) \]

\[ \dot{\theta}_f = \gamma_1 \tilde{g} \phi(y) \quad (1.20) \]

\[ \dot{\theta}_g = \gamma_1 u \tilde{g} \phi(y) \quad (1.21) \]

results in the actual closed-loop system having error dynamics described by

\[ \dot{\tilde{y}} = -0.2 \tilde{y} + \tilde{\theta}_f \phi(y) + \tilde{\theta}_g \phi(y)u + e_{\phi}(y, u) \quad (1.22) \]

\[ \dot{\tilde{\theta}}_f = -\gamma_1 \tilde{g} \phi(y) \quad (1.23) \]

\[ \dot{\tilde{\theta}}_g = -\gamma_1 u \tilde{g} \phi(y) \quad (1.24) \]

where \( \tilde{\theta}_f = \theta_f^* - \theta_f, \tilde{\theta}_g = \theta_g^* - \theta_g \), and \( e_{\phi}(y, u) \) denotes the residual approximation error (i.e., the approximation error that may still exist even if the parameters of the adaptive approximators were set to their optimal values). The \( \tilde{y} \) error dynamics are very similar for the adaptive and nonadaptive feedback linearizing approaches. Relative to the nonadaptive feedback linearizing approach, the error dynamics are more complicated due to the presence of the dynamic equations for \( \tilde{\theta}_f \) and \( \tilde{\theta}_g \). The expected payoff for this added complexity is higher performance (i.e., decreased tracking error). The designer must be careful to analyze the stability of the state of the adaptive feedback linearizing system (i.e., \( \tilde{y}, \tilde{\theta}_f \) and \( \tilde{\theta}_g \)) and to analyze the effect of \( e_{\phi}(y, u) \). This term is rarely zero and the upper bound on its magnitude is a function of the designer's choice of approximation method (i.e., \( \phi \)).

Figure 1.6 displays the performance of the approximation-based feedback linearizing control law using the basis functions defined by

\[ \mu_i(v) = \exp \left( \frac{(v - c_i)^2}{25} \right) \]

\(^{2}\)Rigorous definitions of the optimal parameters and residual approximation error will be given in Section 1.4.2.
Figure 1.6: Performance of the approximation-based control system of eqn. (1.19)-(1.21) with the dynamic system of eqn. (1.1).

\[
\phi_i(v) = \frac{\mu_i(v)}{\sum_{i=1}^{21} \mu_i(v)}
\]

\[
c_i = (i - 1)\delta, \quad \text{for } i = 1, \ldots, 21.
\]

This simulation uses the actual plant dynamics. Initially, the tracking error is large, but as the online data is used to estimate the approximator parameters, the tracking performance improves significantly.

It is important that the designer understands the relationship between the tracking error and the function approximation error. It is possible for the tracking error to approach zero without the approximation error approaching zero. To see this, consider (1.22). If the last three terms sum to zero, then \( \tilde{v} \) will converge to zero. The last three terms sum to zero across a manifold of parameter values, most of which do not necessarily represent accurate approximations over the region \( D \). If the designer is only interested in accurate tracking, then inaccurate function approximation over the entire region \( D \) may be unimportant. If the designer is interested in obtaining accurate function approximations, then conditions for function approximation error convergence must be considered.

Figure 1.7 displays the approximations at the initiation (dotted) and conclusion (solid) of the simulation evaluation, along with the actual functions (dashed). The simulation was concluded after 3000 s of simulated operation. The first 100 s of operation involved the filtered step commands displayed in Figure 1.6. The last 2900 s of operation involved filtered step commands, each with a 10-s duration, randomly distributed in a uniform manner with \( y_c \in [20, 60] \). The initial conditions for the function approximation parameter vectors were defined to closely match the functions \( f_o \) and \( g_o \) of the design model. The bottom graph of Figure 1.8 displays the histogram of \( y_d \) at 0.1-s intervals. The top two graphs show the approximation error at the initial and final conditions. By 3000 s, both \( \tilde{f} \) and \( \tilde{g} \) have converged over the portion of \( D \) that contains a large amount of training data. Nothing can