SELF-TUNING FEEDBACK LINEARIZATION

By

CHARLES H. GEBO, B.S.E.E, M.S.E.E

A Thesis

Submitted to the School of Graduate Studies

in Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy

McMaster University

SELF-TUNING FEEDBACK LINEARIZATION

To

My Family

Marge, John, Carole, and Tom

DOCTOR OF PHILOSOPHY (2002) (Electrical Engineering)		McMASTER UNIVERSITY Hamilton, Ontario
TITLE: Self-tuning Feedback Linearization		
AUTHOR: Charles H. Gebo,	B.S. (Elec. Eng.)	(Tri-State College)
	M.S. (Elec. Eng.)	(Rochester Institute of Technology)
SUPERVISOR: Naresh K. Sinha		Professor, Electrical and Computer Engineering Department
		B.Sc.(Eng) (Banaras) Ph.D. (Manchester)
NUMBER OF PAGES: xv, 253		

ABSTRACT

This research investigates the development of a methodology for designing self-tuning feedback linearizing control laws. If the conventional architecture for linear plant self-tuning systems is applied to the feedback linearization case, it is shown that the estimation algorithm gradient vanishes as the parameter estimates approach the true parameter values. Vanishing of the gradient causes the covariance matrix to increase without bound and consequently system failure.

A new architecture is presented that eliminates the covariance problem but does not yield a direct estimation of the nonlinear plant parameters. The parameters estimated in the new architecture are composites of the true parameter values of the nonlinear plant and their estimated values. An adaptive law is designed to interpret an error equation formed from the composite parameters and asymptotically converge to the true nonlinear plant parameter values. A stability proof and convergence properties for the adaptive law are given. Sufficient conditions for a nonlinear plant to be capable of self-tuning in the new architecture are specified.

The new method is demonstrated with simulations of an arbitrarily chosen nonlinear plant and two plants of practical interest. One plant is a chemical reactor running an exothermic process where reactor temperature is the controlled variable. The other plant is a bioreactor where control of the substrate concentration in an anaerobic digestion process is the objective. In both cases the method developed in this thesis offers performance improvements as compared with previously published results on control of these processes by other methods.

ACKNOWLEDGEMENT

The author wishes to extend a sincere thank you to all his advisors. Each have contributed in different ways to the success of this undertaking: Dr. N.K. Sinha for accepting a non-traditional student into the graduate program and the sharing of his knowledge. Dr. J.F. MacGregor for never losing interest and his guidance and suggestions throughout the course of this project. Dr. T. Davidson who became a committee member very late in the project but nonetheless contributed many thought provoking questions that translated into over-all improvements. And last but not least, Dr. R.E. Swanson who has had a sympathetic ear and offered encouragement over the long course of this undertaking. My thanks to you all.

I would be remiss if I did not acknowledge some of my peers who have already gone on to other endeavors. I thank them for their help as well.

TABLE OF CONTENTS

PAGE

LIST OF FIGURES

FIGURE PAGE

CHAPTER 1

Introduction

1.1 Background

The decade of the 60's was a period of intense research in the control of linear systems. Among other things, there was a shift in emphasis from the standard input / output plant representation in the frequency domain to internal or state-space representation in the time domain. Numerous control techniques were developed and much insight into system behavior was gained using time domain methods (Rosenbrock (1970), Kalman (1963), Kailath (1980), Chen (1970)). Similar progress was made in system identification theory with many new algorithms proposed. The rapid development and increasing availability of digital computers was a very significant contributing factor not only aiding in the development of the theory but also in the practical implementation on real systems.

Research in linear systems has matured to the point where it no longer represents the leading edge of systems technology. The major emphasis now in linear systems is on applications and robustness (Grimble (1994), Zhou and Doyle (1998)). The new challenge is in developing controller design methodologies for nonlinear plants.

Interest in all aspects of control and analysis of nonlinear plants remains high as indicated by the number of recent journal articles. A representative number of these are (Jain et al. (1994), Boskovic (1995), Annaswamy et al. (1998), Jiang and Hill (1999), Meleiro and Filho (2000), Ferrara and Giacomini (2001), Kojic and Annaswamy (2002)). Control of uncertain nonlinear systems continues to be an area of active research. Nearly all of the literature addresses control of nonlinear plants with unknown parameters. Proposals for control of these types of systems are being pursued mostly by development of robust and adaptive techniques. The adaptive methods utilize both Lyapunov (Zhang et al. (2000)) and certainty equivalence (Meleiro and Filho (2000)) approaches.

In many practical cases the parameters of a plant model are not accurately known or may change with plant operating conditions or the grade of the product being made. However in model based controller design techniques, accurate knowledge of the plant parameters is necessary in order to achieve the design objective. In view of parameter uncertainty and since there are well known on-line parameter estimation methods available, control systems have been synthesized by combining an estimator with a means for translating the estimated parameters into controller settings. Systems having the above three elements, that is, a parameter estimator, a design criterion utilizing the estimated parameters, and a controller tuned by the output from the design function were designated as self-tuning or adaptive controllers. Controllers of this type are theoretically capable of tuning themselves to optimal performance and maintaining optimal settings should the plant dynamics change.

The interest in systems with self-tuning properties dates back to the 1950s (Kalman (1958), Gregory (1959)). However neither an adequate theory nor computing capability were available at the time to support successful applications. Major developments in system identification and control theory during the 1960s helped to sustain the interest in adaptive control. Development of adaptive control theory was continued throughout the 1970s and resulted in a methodology for the design of adaptive controllers. Advances in computing hardware and software were made during this same time period and when combined with the advanced theory, led to successful adaptive control applications (Harris and Billings (1981), Narendra and Monopoli (1980), Narendra et al. (1991)).

Self-tuning controllers were developed primarily for linear plants although the theory was extended to a certain class of nonlinear plant. For nonlinear plants, feedback linearization has evolved into one of the main methods of control. Development of this technique has attracted a great deal of research effort over the past decade. Feedback linearization is a model based method which can be used to obtain a linear input/output response from a nonlinear plant. Once linearized, the large body of well established linear controller design methods are available for application to the control of the nonlinear plant.

For the class of nonlinear systems that are feedback linearizable, it is well known that given precise knowledge of system parameters and states that a nonlinear state feedback and a set of coordinates can be chosen which will result in linear dynamic behavior.

There are two components comprising the feedback linearization design. The first is a control law which cancels the nonlinear plant terms that appear in $y^{(r)}$ where y represents the plant output and r is the number of differentiations needed to get an equation in which the plant input appears. For linear systems r is equal to the relative order of the plant. The second component is a nonlinear coordinate transformation which yields the states of the linearized plant model. Both the control law and the coordinate transformation are functions of the nonlinear plant parameters and states. It is necessary to have accurate parameter and state data in order to implement and realize the benefits of a linearizing controller. The problem now becomes one of accurately obtaining the parameters and states in practice in order to implement the above procedure. Assuming that accurate model parameters can be asymptotically obtained, one may consider a self-tuning scheme to optimize the linearizing control law.

1.2 An Overview Of The Problem

To build a self-tuning system for a nonlinear plant it would seem logical to try to adapt the linear plant self-tuning regulator configuration. The parameter estimator in this configuration is connected in parallel with the plant and consequently is driven by the same input signal. In this case the plant and estimator input signal is the output from the feedback linearizing control law. The plant states could be measurable or estimated if a bootstrap estimator is used. For example, the Recursive Prediction Error Method (RPEM) is designed to generate both state and parameter estimates.

Prediction error in the RPEM algorithm is defined as the difference between the plant output and the predicted plant output that is generated by the estimation algorithm. The negative gradient of this error with respect to the estimated parameters provides a direction in which the parameter estimates should be adjusted in order to bring them closer to the true values. In the usual case of parameter estimation, when the plant is driven by an arbitrary input, the gradient depends only on the estimated plant output. When a linearizing control law is used the real plant output also becomes a function of the estimated parameters. Therefore the gradient will now be a function of both the real plant output and the predicted plant output. As the estimated parameters converge to the true values, the gradient of the plant output and the predicted plant output become equal. Since these gradients appear in the prediction error gradient calculation with opposite signs, the prediction error gradient reduces to zero.

Inspection of the covariance matrix update equation shows that if the prediction error gradient vanishes then the update of this matrix is inversely proportional to the forgetting factor, λ. The forgetting factor has a value in the range of $0 < \lambda < 1$. Consequently the covariance will increase without bound causing the system to fail.

1.3 The Research Objective

Several researchers have proposed design techniques for adaptive control of nonlinear plants. Meleiro and Filho (2000) use an artificial neural network to estimate plant parameters. The estimated parameters are used in a certainty equivalence mode to tune a controller. Boskovic (1998) modifies an adaptive algorithm designed for linear plants described by a linear in the parameters model. His modification results in an adaptive algorithm designed to handle a certain class of nonlinear plants having nonlinear parameterization. In general these and other similar proposed techniques are complex. The complexity of these schemes has ramifications in practical implementation.

The objective of this research is to develop a self-tuning feedback linearizing controller of minimum complexity for SISO $n'th$ order relative order

$r=n$ plants. (Relative order r for a linear plant is defined as the difference between the order of the denominator and numerator polynomials of the plant transfer function. For a nonlinear plant it is defined as the number of differentiations of the output expression before the plant input term appears in the equation.) The complete state vector of the nonlinear plant will be assumed to be measurable but parameters are assumed to be unknown. It is a further objective to use one of the well known recursive parameter estimation routines to supply estimates of the parameters. The RPEM will be used to handle this aspect.

The discussion in the last section indicates that a new system architecture will be needed. Although not evident at this point, in addition to a new architecture, an adaptive law for obtaining the nonlinear plant parameters from RPEM estimates of the parameters of a reparameterized model is also needed. The adaptive law will be designed to estimate the nonlinear plant parameters that are needed in the linearizing control law and in the coordinate transformation equations. The new adaptation procedure will avoid the observability problem with parameter estimation in self-tuning feedback linearizing systems. This research addresses these main issues and others which are encountered during the course of developing a complete self-tuning feedback linearization system.

1.4 Thesis Organization

Chapter 2 presents a brief survey of the most useful methods of classical nonlinear control theory. An introduction to feedback linearization control theory and the mathematics and notation used with this method are given. The main theorem (Su (1982)) for determining whether or not a particular plant can be linearized by the feedback linearization method is presented. References where mathematical proofs can be found are cited.

Several approaches to implementation of non-adaptive feedback linearizing controllers have been reported in the journals. Some of these are reviewed and tested on a simulation of a bilinear plant. Adaptive methods for control of nonlinear plants are reviewed and the chapter is concluded with a discussion of adaptive control methods for linear plants.

Chapter 3 evaluates the self-tuning system configuration used for linear plants for use in the self-tuning feedback linearization application and establishes the motivation for a new design. The problem of observability and hence identifiability of the model parameters by conventional estimation approaches when working with linearizing controllers, as discussed in section 1.2, is illustrated by simulations. Several alternatives are tried in an attempt to avoid the difficulty. A new architecture is proposed that solves the problem encountered with the standard configuration.

Chapter 4 continues the development of the self-tuning feedback linearization system. The basic configuration proposed in Chapter 3 does not allow direct estimation of the nonlinear plant parameters. Instead it is convenient to estimate the parameters of a reparameterized model of the feedback linearized plant. A parameter error equation is formulated in such a way that the relative magnitude of the true nonlinear plant parameter and its estimated value can be determined from the sign of the error. An adaptive law is then designed to utilize the sign information to further reduce the error in the parameter estimate. Convergence properties of the adaptive law are investigated and stability is proven.

The complete system consisting of the new architecture and adaptive law is demonstrated on a simulation of an arbitrarily chosen plant. The ability of this system to track an input signal is also demonstrated.

Chapter 5 presents the results for the new method when applied to simulations of two real industrial processes. Further refinements of the method which are peculiar to the particular processes chosen are discussed. The systems investigated in this chapter are also used to demonstrate, with nonlinear plants, the problem with conventional architecture first shown in Chapter 3.

Chapter 6 summarizes the results obtained from this investigation and concludes that the research was successful in providing a new approach to self-tuning feedback linearizing design. Topics for further research are suggested.

1.5 Contributions

This research has resulted in a new method of design for self-tuning feedback linearized systems. The special problems associated with parameter estimation when linearizing control laws are used have been addressed and solved.

The effect that parameter errors in the control law has on the linearized plant model is analyzed. It is shown that these errors can result in a nonlinear parameterization of the linearized model. A parameter error equation has been derived based on the various forms taken by the combinations of true nonlinear plant parameters and the estimated values. An adaptive law has been designed to estimate the nonlinear plant parameters from information contained in the parameter error equation. The adaptive law is proven to be asymptotically stable and shown to be capable of one step convergence under ideal conditions. In order for the adaptive law to apply, the plant must meet certain sufficiency conditions. These conditions are given in section 4.2.2.

The new self-tuning method for feedback linearizing control was applied to simulations of two real industrial processes. These two processes, reactor temperature control for an exothermic process and bioreactor control, are used literally hundreds of times in industry. Controller design for these two processes has been investigated by many researchers, (Dochain (1986), Cott and Macchietto (1989), Annaswamy et al. (1998), Boscovic (1998), Viel et al. (1997)). Control and parameter estimation using the self-tuning method developed in this thesis is compared with results reported in the literature by Dochain (1986) and Cott and Macchietto (1989). For both the exothermic reactor temperature control and the bioreactor control the new method shows improved performance.

CHAPTER 2

A Survey of Classical and Modern Nonlinear Control Techniques

2.1 Introduction

System analysis and design is extremely difficult when the principle of superposition is not valid. The normal mathematical tools and procedures used for working with linear systems cannot be applied in these cases. However nonlinearities exist in many systems of practical interest making it essential that some form of analysis be undertaken. For example coulomb friction, stiction, binding, backlash, saturation, hysteresis, etc. are commonly found in components and systems. Early attempts at analyzing nonlinear systems were made during the 1940's and 50's. Those which proved to be the most useful were the phase plane, describing functions, linear approximation, and analog computer simulation studies. These techniques were used primarily on static type nonlinearities such as those mentioned above (Thaler and Pastel (1962), Truxal (1955), Leigh (1983)). More recently the focus has been on dynamic nonlinearities. The major techniques emerging in this area are feedback linearization and a method known as backstepping (Kristic et.al. (1995)).

2.2 A Review Of Some Classical Techniques

Over the years a number of methods have been developed for the analysis and design of nonlinear systems. Those developed prior to 1970 are arbitrarily referred to in this thesis as the classical techniques. The following sections give a brief review of some of these methods.

2.2.1 The Phase Plane

One of the most useful of the various classical methods is the phase plane (Graham and McRuer (1961), Slotine and Li (1991)). It is a graphical technique which in theory is applicable to n'th order systems but for all practical purposes can only be applied to first and second order systems. The phase trajectory is a plot of $\dot{x}$ versus x. It is constructed by first carrying out a series of slope calculations at several points in the phase plane and plotting short line segments having the calculated slope. This is a very tedious process if done by hand but with the widespread availability of computers today this is no longer a drawback. A curve tangent to each of the slope segments is then sketched yielding the phase trajectory. This curve depicts the transient response for one set of initial conditions. The process has to be repeated for each set of initial conditions of interest. A plot of several phase trajectories is called a phase portrait.

The system can be driven only by initial conditions, steps, or ramps for all of the various methods which can be used to construct the phase plane with one exception. The delta method can be used with forcing functions that are functions of time (Cunningham (1958)). Unlike other methods of construction, this method does not place restrictions on the plant input. The second order differential equation describing the plant is manipulated into the equation for a circle in the $x,\dot{x}$ plane. The center of the circle is at $(\delta,0)$ where $\delta = f(\dot{x},x,t)$ lies on the x axis and changes position as small incremental changes in the delta arguments are made. Starting from initial conditions, short arc segments of a circle with center at $(\delta,0)$ are pieced together to form the phase trajectory.

The major advantage offered by the phase plane is that the system response can be seen without requiring an analytical solution of the nonlinear equations. The phase plane method can be applied to systems having nonlinearities of any magnitude and they may be either continuous or discontinuous. The phase plane is useful for studying system transient response, stability, overshoot, existence of limit cycles, static accuracy and designing compensators for these properties. The main disadvantage is the limitation on system order.

Although we are only interested in nonlinear systems in this investigation, it should be noted that the phase plane is also applicable to linear systems. For example it is very useful for designing bang-bang predictive type controllers for linear plants.

2.2.2 Describing Functions

A describing function is a frequency domain description of a nonlinearity (Slotine and Li (1991)). The objective is to extend some of the frequency response concepts used with linear systems to the analysis of nonlinear systems. The method can be applied to systems which can be put into the configuration shown in Figure 2.1.

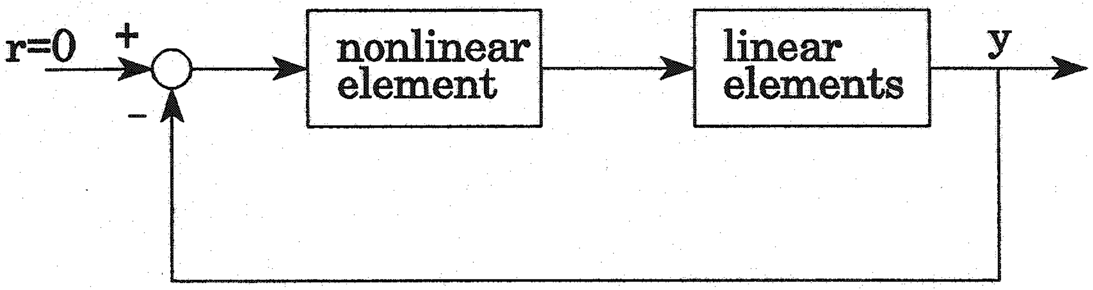

Figure 2.1 System configuration for describing function analysis.

The basic version of the describing function applies when there is only a single time invariant nonlinear element, only the fundamental component of the output from the nonlinearity need be considered, and the nonlinearity is an odd function.

The describing function is calculated by assuming that the input to the nonlinear device is a pure sinusoid[1] (i.e. $R\sin(\omega t)$) and expanding the output in a Fourier series. All terms in the output series except the fundamental frequency term, say $C(R,\omega)\sin(\omega t+\phi(R,\omega))$, are discarded. The describing function is defined as the ratio of the output to the input, i.e.

[1] A sinusoidal input is the general assumption although describing functions for Gaussian random signals and systems with d.c. bias can also be derived.

$$N(R,\omega) \doteq \left|\frac{C(R,\omega)}{R}\right| \angle \phi(R,\omega) \qquad (2.2.1)$$

for all permissible amplitudes and frequencies of the input. This in effect provides a linearization in the frequency domain. The justification for dropping higher frequency output terms is twofold. The first is that the magnitudes of harmonics are usually smaller than the fundamental and therefore have less effect. Secondly the linear part of the plant is assumed to act as a low pass filter which will attenuate higher frequency components thus reducing their significance even more. Describing functions for the most commonly found nonlinearities such as saturation, hysteresis, etc. are functions of the input amplitude only and not of ω. An example of a frequency dependent describing function is given in (Truxal (1955)). The plant is a D.C. motor operating in the velocity saturation mode.

Describing functions may be calculated by numerical integration if the input-output relationship is given graphically or as a table. (i.e. a numerical evaluation of the Fourier series integrals) Still another technique is to excite the nonlinear element with a sinusoid of known amplitude and frequency and evaluate the output with a harmonic analyzer.

After obtaining the describing function, all of the usual frequency response methods can be applied to the analysis and design of the control system. The predominant use of describing function analysis is for the prediction of limit cycles via the Nyquist criterion.

Many refinements and extensions have been made to the basic describing function approach. See Gelb and Vander Velde (1968) for a comprehensive treatment.

2.2.3 Linear Approximation

Linear approximation is in general only valid in a small neighborhood about an operating point. The approximation is usually obtained by retaining the first two terms of a Taylor series expansion of a nonlinearity for which an analytical expression is available. For example, given f, a function of a single variable,

$$f(a+h) \approx f(a) + h\frac{df}{dh}\Big|_a \tag{2.2.2}$$

where a is the fixed point about which the expansion is made and h is a small perturbation from a. The obvious disadvantage is the limited range over which the approximation is accurate enough to yield useful results. Linearization is useful for investigating the stability of a nonlinear system around an equilibrium point.

2.2.4 Computer Simulation

Computer simulation offers the advantage of including more of the nonlinearities than could be taken into account using any of the other methods. The effect of component tolerance on system performance is easily evaluated and it is possible to include actual pieces of hardware in the simulation. Hybrid systems using combinations of analog and digital computers have also been used to take advantage of the best characteristics of both machines. Of the classical techniques, simulation provides the most accurate means of investigating nonlinear systems. It offers the additional advantage of easily changing from one set of conditions to another making it practical to evaluate system performance for several different situations (Bekey and Karplus (1968)).

In the past computer simulations required large amounts of programming labor, hardware interfacing, etc. Nowadays computer systems and software intended specifically for control system design and evaluation are commercially available. Two of the most popular software packages are MATLAB with SIMULINK and VISSIM. Once a controller design has been proven to be satisfactory by simulation, the designer can then have the simulation system automatically generate the code necessary to implement the controller. The source code can be generated in ADA, C, or FORTRAN (from one particular vendor) and downloaded to a prototype computer control system for evaluation with the real process. The prototyping system has all the necessary drivers and I/O hardware for communicating with the process.

2.2.5 Lyapunov Theory

The Lyapunov function is the workhorse used to check the stability of nonlinear plants (LaSalle and Lefschetz (1961), Khalil (2002)). The definition of stability that is referred to in the following is asymptotic stability which means that for some $r > 0$ such that $\|x(0)\| < r$, $x(t) \to x_e$ as $t \to \infty$ where x_e is a stable equilibrium (Narendra and Annaswamy (1989)). Given the plant represented by

$$\dot{x} = f(x), \qquad f(0) = 0 \tag{2.2.3}$$

where $x \in R^n$ and $f(x)$ is a nonlinear function, stability can be investigated with the following theorem.

Theorem 2.2.1 Lyapunov stability theorem

If there exists a scalar function $V(x)$, with continuous first partial derivatives with respect to x such that:

1. $V(x) = 0, \qquad x = x_e$
2. $V(x) > 0, \qquad x \neq x_e$
3. $\dot{V}(x) < 0, \qquad x \neq x_e$
4. $V(x) \to \infty, \qquad \|x\| \to \infty$

then (2.2.3) is globally asymptotically stable.

That is, it is stable for any set of initial conditions.

The Lyapunov function is defined as V(x). Theorem 2.2.1 is an embodiment of Lyapunov's second method. The advantage of the second method is that the solution of (2.2.3) is not needed to determine stability. The difficulty in application is that there is no general method for finding Lyapunov functions except in certain special cases. The theorem gives a sufficient condition for stability. If a Lyapunov function candidate does not satisfy the conditions of the theorem, instability cannot be concluded. The only recourse is to choose a new candidate function and recheck the theorem conditions.

While Lyapunov theory was intended primarily to determine system stability, it also plays a large role in system design. The conditions for stability generated by Lyapunov functions can frequently be interpreted as algorithms for controller parameter adjustment in adaptive systems. An example of this is given in section 2.8.

2.2.6 Comments On Classical Methods

All of the above methods constitute the major techniques of the classical approach of nonlinear system analysis and design. In addition to the graphical methods, Lyapunov functions provided an additional means for evaluating system stability directly from the system equations. For the linear parts of the system, the primary mathematical tools are the Laplace and Z transforms. A large body of literature exists on the theory and application of these techniques.

The objective of the classical approach was to manipulate the nonlinear problem into a form which could be handled by the extension of existing linear theory. With the exception of simulation, which could actually model nonlinearities, and possibly the phase plane, the existing well developed theory of linear systems was applied to linear approximations of nonlinear systems.

Classical techniques yielded approximate but useful results and are still in use. For example Luh et.al. (1983) uses a describing function to analyze the effects of backlash on the stability of an industrial robot and then to design a stabilizing compensator for it. However for many important nonlinear plants, the classical theory was only marginally adequate. Nonlinear plants of practical interest can be found in many different fields including aerospace, robotics, chemical and biochemical processes, mechanical systems, electric motor drives, electronic circuits, and many others. The importance of these many areas of application has provided the motivation for research into new and improved nonlinear control techniques.

2.3 A Review Of Some Modern Techniques

Research on new approaches for design and analysis of nonlinear systems began in the early 1970's. Some of the earliest published accounts are due to Brockett (1976,1978), Su (1982,1986), and Hunt et.al.(1983). One of the main feature of the modern approach is again linearization. However there are several important differences from the classical approach. For a large class of plants the linearization is exact and global rather than approximate and local. Also there are no limitations on plant order as there is, for example, for practical application of the phase plane. Plant inputs are arbitrary and of much less concern than they were with the classical techniques. The modern technique applies to dynamic nonlinearities in contrast to many of the classical techniques which primarily handled static type nonlinearities. This is a great advantage since nonlinear differential equations arise naturally from the physical laws governing the behavior of many processes. Even though there is some calculation involved in the classical approaches, it is mainly for the purpose of generating either time or frequency domain plots and the analysis or design is then carried out as a graphical procedure from these plots. The modern approach is entirely analytic. The major advantage is that once linearized, a substantial simplification in analysis and in design of controllers is realized since the full range of linear theory can be applied. However the modern approach is not without cost. Incumbent with these new powerful techniques is an increase in the level of mathematical abstraction (Brockett (1976), Banks (1988)). The main mathematical tool in the modern approach to nonlinear systems is differential geometry. The most complete references at this time that are devoted to control system application are Isidori (1995) and Nijmeijer and van der Schaft (1990). A brief accessible discussion of some of the fundamentals can be found in Spong and Vidyasagar (1989). There are numerous papers applying the theory to various control problems: DeLuca and Ulivi (1987), Ilic'-Spong et.al. (1987), Soroush and Kravaris (1991), Marino et. al. (1993), Ma and Tao (2000).

2.3.1 Feedback Linearization

Given a nonlinear plant, the basic idea of feedback linearization is to find a control law and a nonlinear coordinate transformation such that the equivalent system in the new coordinates is linear. This method can be applied to plants having more than one nonlinearity and the nonlinear terms can appear in any of the state equations of the plant model although not all plants can be linearized by the above method. The next section describes a method which applies to a more restricted class of plants. The generalization of this method to arbitrary plants leads to the feedback linearization technique mentioned above.

2.3.2 Inverse Dynamics

In some cases the nonlinear terms appear only in the state equations where components of the input signal are also present. If this is true then it is possible to eliminate the nonlinear terms without the coordinate change. The method of linearizing this class of plant is known as inverse dynamics in the robotics literature. The objective of inverse dynamics is to find a nonlinear feedback control law that, when substituted into the original system equations, will result in a linear closed loop system. A simple illustration the of inverse dynamics principle can be given by considering the first order bilinear plant

$$\begin{aligned} \dot{x} &= -ax - xu + bu \\ y &= x \end{aligned} \tag{2.3.1}$$

where x is the state, u is the input, and y is the output. For this plant it is easily seen that by choosing the control law as either

$$u = \frac{v}{b-x} \tag{2.3.2}$$

or

$$u = \frac{v+ax}{b-x}, \tag{2.3.3}$$

where v denotes an arbitrary input, the nonlinear term will be cancelled and the resulting model will be linear in ν. According to the theory to be developed later, input (2.3.3) would be chosen. The input/output response would then be that of a pure integral plant. Generally feedback is added to a plant of this type to change the response to first order. If the first order response due to time constant $\tau = 1/a$ is acceptable then input (2.3.2) can be chosen.

Another example is the nonlinear second order plant

$$\ddot{e} + b\dot{e} + b_1\dot{e}^3 = u . \tag{2.3.4}$$

Choosing the plant input as

$$u = b_1\dot{e}^3 - ke + v \tag{2.3.5}$$

where v is an arbitrary input converts (2.3.4) into the linear plant

$$\ddot{e} + b\dot{e} + ke = v . \tag{2.3.6}$$

2.3.3 Generalized Inverse Dynamics

The generalization of the inverse dynamics concept is known as feedback linearization. A much broader class of plants can be linearized by this technique as compared with the inverse dynamics approach. The major difference is that nonlinearities can be present in more than one plant state equation even though a component of the input does not appear in that equation.

The architecture of a control system for a feedback linearized plant is shown in Figure 2.2. The function of the inner loop is to supply the plant states needed for linearization to the control law and the nonlinear coordinate

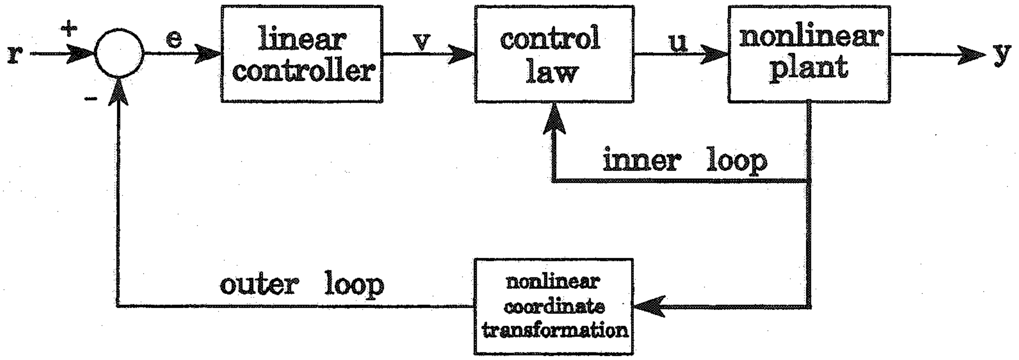

Figure 2.2 Feedback linearized control system.

transformation. It should be noted that plant parameters are also needed for linearization. The arbitrary input v is converted by the control law to the plant input *u* which is needed to produce a linear I/O relationship in the transformed coordinates.

The outer loop contains the nonlinear coordinate transformation which converts the nonlinear plant states and parameters into the state values of the linearized system. The transformed coordinates are fed back and compared with a reference signal, *r*, to generate an error, *e*, which is fed into a linear controller to complete the loop.

2.3.4 Backstepping

Backstepping is a recursive design procedure introduced by Kanellakopoulos (1991). It has as a goal the design of a control law for $\dot{x} = f(x,u)$ to meet a control objective and at the same time establish the equilibrium point,

$x = 0$, of the closed loop system to be globally asymptotically stable. The emphasis in backstepping controller design is on system stabilization and tracking rather than plant linearization. As such, useful nonlinearities in the plant need not be cancelled as in feedback linearization designs.

Adaptive backstepping can be used to design controllers for nonlinear plants with unknown parameters (Krstic et. al. (1995)). The parameter update law is obtained from the system Lyapunov function. In order to guarantee stability, the parameter update law is chosen from terms in the Lyapunov function which have to be cancelled in order to make $\dot{V}$ negative definite.

The backstepping method has removed a major obstacle in the Lyapunov based design of adaptive controllers. This was the relative order ≤ 2 limitation present with this design method. For relative order 1 plants $\frac{d\hat{\theta}}{dt}$ appearing in the control law can be substituted for from the update law. The control law for relative order 2 plants involves $\frac{d^2\hat{\theta}}{dt^2}$. Since there is no expression from which this second derivative can be evaluated, the design of a stabalizing controller for these plants could not be carried out using Lyapunov methods prior to the introduction of backstepping.

2.4 Mathematical Preliminaries And Notation

In this section we will review the notation and some of the mathematical tools that are in common usage in the feedback linearization literature and will be used throughout the rest of this thesis. The SISO nonlinear plants that we will be concerned with in this section can be represented by

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= h(x) \end{aligned} \qquad (2.4.1)$$

where $x \in R^n$, $u \in R$ and $f(x)$, $g(x)$ are C^∞ vector fields on R^n. The output expression, $h(x)$, is a C^∞ scalar field on R^n. (C^∞ designates a function having continuous partial derivatives of any order, R^n signifies an n dimensional space) Taking the time derivative of y yields

$$\dot{y} = \frac{\partial h}{\partial x} f(x) + \frac{\partial h}{\partial x} g(x) u\,. \qquad (2.4.2)$$

The Lie derivatives of h wrt f and g are defined as

$$L_f h(x) = \frac{\partial h}{\partial x} f = \langle dh, f \rangle \qquad (2.4.3)$$

and

$$L_g h(x) = \frac{\partial h}{\partial x} g(x) = \langle dh, g \rangle \qquad (2.4.4)$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product. The gradient of h wrt the vector x is defined as

$$dh = \left[\frac{\partial h}{\partial x_1}, \;\cdots\cdots\; , \frac{\partial h}{\partial x_n} \right]. \qquad (2.4.5)$$

Using (2.4.3) and (2.4.4) $\dot{y}$ can be written as

$$\dot{y} = L_f h(x) + L_g h(x) u\,. \qquad (2.4.6)$$

Higher order Lie derivatives are defined inductively as follows:

$$L_f^k h(x) = L_f \left(L_f^{k-1} h(x) \right), \qquad k = 2, 3, \ldots\,. \qquad (2.4.7)$$

The Lie bracket $[f,g]$, where f and g are C^∞ vector fields on R^n, is defined by

$$[f,g] = \frac{\partial g}{\partial x}f - \frac{\partial f}{\partial x}g \tag{2.4.8}$$

where $\frac{\partial g}{\partial x}$ and $\frac{\partial f}{\partial x}$ are n x n Jacobian matrices. Successive Lie brackets are defined as $[f,[f,g]]$, $[f,[f,[f,g]]]$, etc. The standard shorthand notation for Lie brackets is

$$\begin{aligned} ad_f^0(g) &= g \\ ad_f^1(g) &= [f,g] \\ ad_f^2(g) &= [f,[f,g]] \\ &= \vdots \\ ad_f^k(g) &= [f,ad_f^{k-1}(g)]\,. \end{aligned} \tag{2.4.9}$$

Relative order is a concept that will arise frequently in what follows. It is applicable to both linear and nonlinear plants. For a linear plant it is numerically equal to the difference between the order of the denominator and numerator polynomials of the transfer function. In the linear case it is easy to show that the relative order of a plant is also equal to the number of times the output expression has to be differentiated before the plant input appears explicitly. Relative order for a nonlinear plant is defined similarly with regard to output differentiation.

<u>Definition 2.4.1</u>

Relative order, r, is the smallest integer γ such that

$$L_g h(x) = L_g L_f h(x) = \cdots L_g L_f^{\gamma-2} h(x) = 0$$

and

$$L_g L_f^{\gamma-1} h(x) \neq 0, \qquad \forall\, x \in R^n .$$

2.4.1 Test For Linearizability

Not all systems can be linearized by nonlinear state feedback and a coordinate transformation. An existence theorem due to Su (1982) can be used to test a nonlinear system for feedback linearizability. We will need the following definition.

Definition 2.4.2

A linearly independent set of vector fields $\{X_1, X_2, \dots, X_m\}$ is said to be involutive if and only if there are scalar functions $\alpha_{ijk}: R^n \rightarrow R$ such that

$$\left[X_i, X_j\right] = \sum_{k=1}^{m} \alpha_{ijk} X_k \qquad \forall\, i,j,k . \tag{2.4.11}$$

Theorem 2.4.1 (Su (1982))

The nonlinear system

$$\dot{x} = f(x) + g(x)u \tag{2.4.10}$$

with *f(x), g(x)* smooth vector fields, and *f(0) = 0* is feedback linearizable if and only if there exists a region *U* containing the origin in R^n in which the following conditions hold:

1. The vector fields $\{g, ad_f(g), \cdots, ad_f^{n-1}(g)\}$ are linearly independent in U.
2. The set $\{g, ad_f(g), \cdots, ad_f^{n-2}(g)\}$ is involutive in U.

Theorem 2.4.1 guarantees that a plant is input-state linearizable. That is, with a coordinate transformation, a control law can be found such that the state space representation in the new coordinates contains no nonlinear terms. We are interested in input-output linearization but for the classification of plants we are considering, where the relative order is equal to the plant order, the input-output linearization procedure results in an input-state linearized plant as well. Therefore the theorem also guarantees input-output linearizability when $r = n$. In the next section it will be shown that conditions 1 and 2 of the theorem are both necessary and sufficient.

2.4.2 Coordinate Transformation And Control Law Calculation

The feedback linearization of a nonlinear system involves finding the linearizing control law and a nonlinear coordinate transformation that results in a linear I/O relationship. To develop this procedure, consider the nonlinear system given by (2.4.1). The control law will have the form

$$u = \alpha(x) + \beta(x)v \qquad (2.4.12)$$

where v is a new arbitrarily chosen input. Equation (2.4.12) is called a static control law since neither α nor β are functions of the states of another dynamic system such as a reference model (Isidori (1995)).

The state space representation of the output linearized system is given by

$$\dot{y} = Ay + bv \,. \tag{2.4.13}$$

Equation (2.4.13) is in Brunovsky canonical form, i.e.

$$A = \begin{bmatrix} 0 & 1 & 0 & . & . & . & 0 \\ 0 & 0 & 1 & 0 & . & . & 0 \\ . & 0 & 0 & 1 & & & \\ . & & & & & & \\ . & & & & & & \\ 0 & 0 & . & . & & & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \qquad b = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \tag{2.4.14}$$

$$y = T(x) \,. \tag{2.4.15}$$

Definition 2.4.3

A diffeomorphism is a differentiable function that has a differentiable inverse.

The first task is to find the diffeomorphism, *T(x),* which defines the new state coordinates. *T(x)* and *y* are column vectors. The time derivative of (2.4.15) is

$$\dot{y} = \frac{\partial T}{\partial x}\dot{x} \tag{2.4.16}$$

where $\frac{\partial T}{\partial x}$ is the Jacobian of *T(x)*. Using (2.4.10) and (2.4.13) in (2.4.16) yields

$$\frac{\partial T}{\partial x}\Big[f(x) + g(x)u\Big] = AT(x) + bv \,. \tag{2.4.17}$$

Equation (2.4.17) is a system of partial differential equations the first of which is

$$\frac{\partial T_1}{\partial x_1}\dot{x}_1 + \frac{\partial T_1}{\partial x_2}\dot{x}_2 + \ldots + \frac{\partial T_1}{\partial x_n}\dot{x}_n = T_2(x) . \tag{2.4.18}$$

This can be compactly written as

$$\left\langle dT_1, f(x) + g(x)u \right\rangle = \left\langle dT_1, f \right\rangle + \left\langle dT_1, g \right\rangle u = T_2 . \tag{2.4.19}$$

Similarly for the other components of *T(x)*,

$$\begin{aligned} \left\langle dT_2, f \right\rangle + \left\langle dT_2, g \right\rangle u &= T_3 \\ &\vdots \\ \left\langle dT_n, f \right\rangle + \left\langle dT_n, g \right\rangle u &= v . \end{aligned} \tag{2.4.20}$$

The choice of the Brunovsky form for the linearized system dictates that transformation *T(x)* must result in a system of relative order *n*. Therefore

$$\left\langle dT_i, g \right\rangle = 0, \qquad i = 1,2, \ldots ,n-1 \tag{2.4.21}$$

and the first *n-1* equations in (2.4.20) reduce to

$$\left\langle dT_i, f \right\rangle = T_{i+1}, \qquad i = 1,2, \ldots ,n-1 . \tag{2.4.22}$$

If T_1 can be found then evidently from (2.4.22) the complete *T* vector can be determined. The following lemma is useful in finding *T*.

Lemma 2.4.1

If $h{:}R^n \rightarrow R$ is a scalar function and *f* and *g* are vector fields on R^n, then the following is an identity.

$$\left\langle dh, [f,g] \right\rangle = \left\langle d\left\langle dh, g \right\rangle, f \right\rangle - \left\langle d\left\langle dh, f \right\rangle, g \right\rangle . \tag{2.4.23}$$

By imposing the conditions due to equations (2.4.21) and (2.4.22) in this identity and substituting for *dh* the equivalent dT_1, it can be shown that

$$\left\langle dT_1, ad_f^k(g)\right\rangle = 0\,. \tag{2.4.24}$$

By induction it can be shown that

$$\left\langle dT_1, ad_f^k(g)\right\rangle = 0, \qquad k = 0, 1, \ \dots\ , n-2 \tag{2.4.25}$$

$$\left\langle dT_1, ad_f^{n-1}(g)\right\rangle \neq 0\,. \tag{2.4.26}$$

The existence of a solution for the system of partial differential equations (2.4.25) can be guaranteed by Frobenius' theorem.

<u>Theorem 2.4.2</u> (Frobenius)

Given a set of vector fields $\{X_1, X_2, \ \dots\ , X_m\}$ that are linearly independent at each point, the set is completely integrable if and only if it is involutive.

In this case the vector fields are the Lie brackets in (2.4.25). The reasons for conditions *1* and *2* in Theorem 2.4.1 can be seen from the foregoing discussion. Referring to condition *1*, if $ad_f^{n-1}(g)$ could be expressed as a linear combination of the remaining Lie brackets then (2.4.26) would be zero by virtue of (2.4.25). Therefore T_1 could not be found and consequently none of the other components of *T*. Condition *2* is necessary to meet the requirements of Theorem 2.4.2 and insure the existence of a solution for *T*.

The preceding discussion has outlined the method for linearizing a system. First of all, use of Theorem 2.4.1 determines whether or not the system is linearizable. If the theorem conditions are satisfied, then expansion of (2.4.25) yields a system of partial differential equations in T_1 which is guaranteed to have a solution. With T_1 known, $T_2, T_3, \ldots, T_n$ can be found from (2.4.22). The remaining problem is to find the control law u. From (2.4.20)

$$u = \frac{1}{\left\langle dT_n, g \right\rangle} \left(v - \left\langle dT_n, f \right\rangle \right). \tag{2.4.27}$$

2.4.3 MIMO System Linearization

Multi-input multi-output systems can be linearized in a straight forward manner using the above techniques. Consider the square system

$$\dot{x} = f(x) + \sum_{k=1}^{p} g_k(x) u_k \tag{2.4.28a}$$

$$y_i = h_i(x), \qquad i = 1, 2, \ldots, p \tag{2.4.28b}$$

where $x \in R^n$, $u \in R^p$, $y \in R^p$ and f, g_k, h_i are smooth functions. For each y_i in (2.4.28b) time derivatives can be calculated until at least one of the$L_{g_i} h_i(x)$ coefficients of a system input is different from zero. For example, the derivative of the *j'th* output is

$$y_j^{(\gamma_j)} = L_f^{(\gamma_j)} h_j + \sum_{i=1}^{p} L_{g_i}\left(L_f^{(\gamma_j - 1)} h_j \right) u_i \tag{2.4.29}$$

where γ_j is the relative order. Define the matrix

$$A(x) = \begin{bmatrix} L_{g_1}L_f^{(\gamma_1-1)}h_1 & \cdots & L_{g_p}L_f^{(\gamma_1-1)}h_1 \\ \vdots & & \\ L_{g_1}L_f^{(\gamma_p-1)}h_p & \cdots & L_{g_p}L_f^{(\gamma_p-1)}h_p \end{bmatrix} \tag{2.4.30}$$

Using (2.4.30), the output derivatives can be written as

$$\begin{bmatrix} y_1^{(\gamma_1)} \\ \vdots \\ y_p^{(\gamma_p)} \end{bmatrix} = \begin{bmatrix} L_f^{(\gamma_1)}h_1 \\ \vdots \\ L_f^{(\gamma_p)}h_p \end{bmatrix} + A(x) \begin{bmatrix} u_1 \\ \vdots \\ u_p \end{bmatrix} . \tag{2.4.31}$$

Assuming that *A(x)* is nonsingular, choosing the state feedback control law as

$$u = -A^{-1}(x) \begin{bmatrix} L_f^{(\gamma_1)}h_1 \\ \vdots \\ L_f^{(\gamma_p)}h_p \end{bmatrix} + A^{-1}(x)v \tag{2.4.32}$$

results in the linear system

$$\begin{bmatrix} y_1^{(\gamma_1)} \\ \vdots \\ y_p^{(\gamma_p)} \end{bmatrix} = \begin{bmatrix} v_1 \\ \vdots \\ v_p \end{bmatrix} . \tag{2.4.33}$$

A nice feature that results from linearizing (2.4.28a,b) is that the system (2.4.33) is also decoupled. Individual controllers can now be applied to each I/O pair.

The case that has just been considered is analogous to the SISO inverse dynamics case. That is a coordinate transformation is not needed because at least one component of the input appears in each state equation. However multivariable plants that do not meet a matching condition can still be linearized. These plants will require a coordinate transformation. For an example see Spong and Vidyasagar (1989), pgs 274-276, or Isidori (1995).

2.5 Feedback Linearization Implementation Considerations

Feedback linearization is an important advancement in the control of nonlinear systems. Linearized systems can be made to track an input signal in a straight forward manner, dynamic response is easily controlled by feedback of the weighted transformed state variables, etc. In the MIMO case, the plant is decoupled as a by-product of linearization. However these desirable properties depend on exact cancellation of nonlinear terms.

The control law needed for cancellation is a function of both the states and parameters of the nonlinear plant. In terms of controller design there is a similarity at this point with linear plants in that the same information is needed. For example, a pole placement controller for a linear plant requires knowledge of the plant states and parameters (indirect design). But states are not always measurable and parameters may not be accurately known. In linear systems these conditions are handled by an appropriate type of estimator.

Similarly in the nonlinear case, if parameters needed in the control law and nonlinear coordinate transformation are unknown, then on-line estimation of these parameters is necessary in self-tuning or adaptive systems. However the linearizing control law has the effect of rendering some of the states and parameters of the nonlinear plant unobservable. As will be shown in Chapter 3, this can result in problems in the parameter estimation algorithm.

When linearized, the plant can be represented by a string of cascaded integrators. This is a flexible structure to work with but it is necessary to have the transformed state variables available for control purposes. If the nonlinear plant states and parameters are known then the linear plant states can be calculated from the transformation equations. In the following chapters it will be assumed that the nonlinear plant states are measurable but some of the plant parameters are unknown. The investigation then focuses on establishing an accurate means for estimating these parameters since they are crucial for producing complete linearization.

2.6 A Survey Of Some Current Nonlinear Control Strategies

In recent years a number of strategies for nonlinear systems control have been proposed. The underlying principle in the more recent of these proposals is feedback linearization. This technique depends on the availability of accurate plant models. Consequently the better performing controllers incorporate a means for coping with uncertain plants.

In this section we will review some of these methods. We will also review some aspects of adaptive control of linear systems. Of particular interest is the self-tuning regulator since it is the starting point for investigation of a new technique of self-tuning feedback linearization which will be developed in the following chapters.

2.6.1 Non-adaptive Methods

Several non-adaptive control systems for nonlinear plants have appeared recently in the literature. A review of three of these systems is given in the following subsections. All three systems are variations of the feedback linearization method. Before discussing the specific details of each system a summary of the feedback linearization technique is presented in the next section. This review will help to make the similarities between the systems and the relationship of each one to the underlying feedback linearization theory more transparent.

2.6.2 A Summary Of The Feedback Linearization Method

The linearizing control law is always of the form

$$u = \frac{V - L_f^r h(x)}{L_g L_f^{r-1} h(x)} \tag{2.6.1}$$

where V is an arbitrary input signal and $L_g L_f^{r-1} h(x) \neq 0$. Applying (2.6.1) to the nonlinear plant and specifying a coordinate transformation

$$y = T(x) \tag{2.6.2}$$

such that the equivalent linear system is in Brunovsky canonical form, the linearized plant responds as a cascaded string of integrators. The block diagram is shown in Figure 2.3 where r indicates the relative order of the plant.

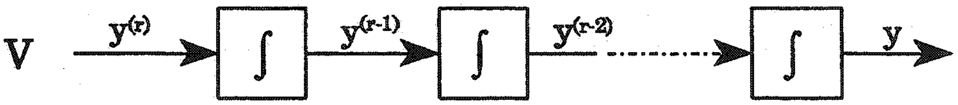

Figure 2.3 Equivalent feedback linearized plant.

This configuration is particularly well suited to pole placement control. The output from each integrator is simply fed back to the input through a gain block as shown in Figure 2.4. The arbitrary input V can be chosen to make the system output respond in some desired way. If the objective is to have the system output track a reference input, y_d, then the input to each integrator must be the appropriate

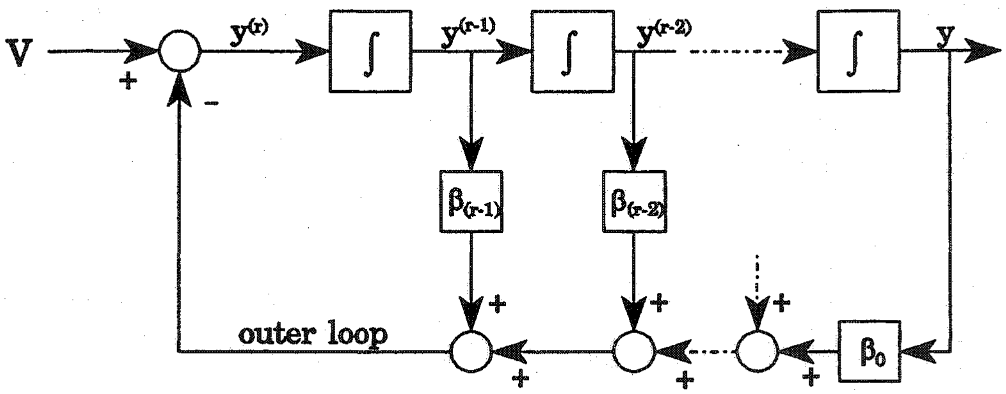

Figure 2.4 Feedback linearized plant with a pole placement controller.

derivative of the desired output. Therefore

$$\begin{aligned} V &= y_d^{(r)} + \beta_{r-1} y_d^{(r-1)} + \cdots + \beta_0 y_d \\ &= \sum_{k=0}^{r} \beta_k y_d^{(k)} . \end{aligned} \tag{2.6.3}$$

Referring to Figure 2.4 and with V given by (2.6.3)

$$y^{(r)} = \sum_{k=0}^{r} \beta_k y_d^{(r)} - \sum_{k=0}^{r-1} \beta_k y^{(k)} \tag{2.6.4}$$

which can be written as

$$\sum_{k=0}^{r} \beta_k y_d^{(k)} - \sum_{k=0}^{r} \beta_k y^{(k)} = 0 . \tag{2.6.5}$$

If we define the error as $e \doteq y_d - y$ then (2.6.5) reduces to

$$\sum_{k=0}^{r} \beta_k e^{(k)} = 0 . \tag{2.6.6}$$

Thus the error response of the feedback linearized system is readily controlled by choosing the feedback gains, β_k.

Figure 2.4 depicts the basic feedback linearized system with a pole placement controller. The pole placement loop is called the outer loop. The inner loop, which does not show in this diagram, transmits the plant states to the control law and coordinate transformation. (see Figure 2.2) The system of Figure 2.4 can be imbedded in still another feedback loop as shown in Figure 2.5. When this external loop is used and y_{sp} is set equal to y_d, the error response (2.6.6) is modified by $f(e)$, the output from the controller, such that the error equation becomes

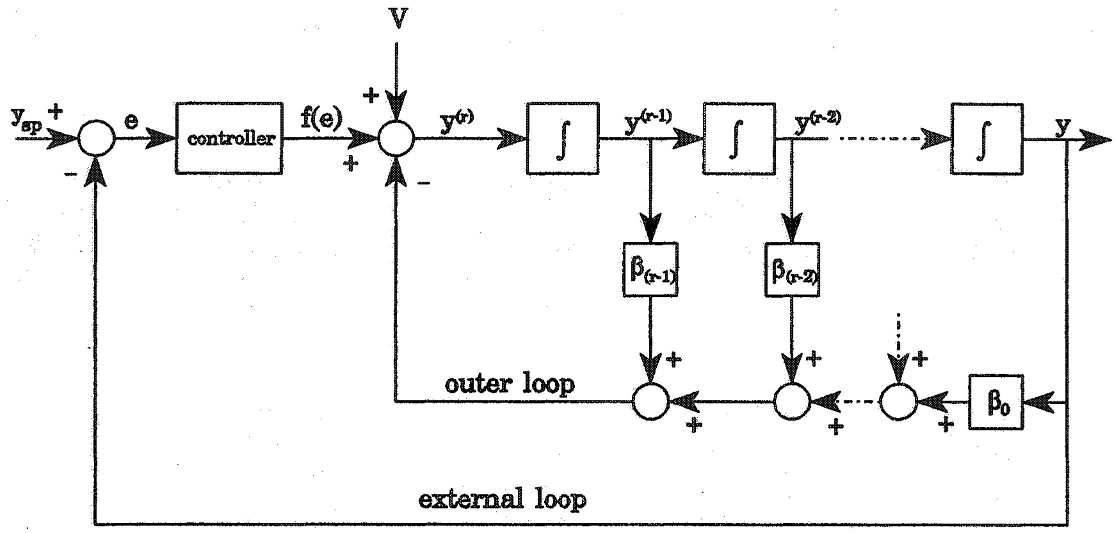

Figure 2.5 System of Figure 2.4 imbedded in an external control loop.

$$\sum_{k=0}^{r} \beta_k e^{(k)} + f(e) = 0 \qquad (2.6.7)$$

where $f(e)$ depends on the response modes of the selected controller.

McLellan et.al. (1990) has observed that the GMC (Generic Model Control), RSS (Reference System Synthesis), and GLC (Global Linearizing Control) systems, that will be reviewed in the next three subsections, are all derived from the configuration shown in Figure 2.5. These three systems are variations of the architecture of Figure 2.5 resulting from the designers choice of V, β_k, and $f(e)$.

Frequently $f(e)$ is chosen as the output from a PI or PID controller. The effect of the external loop with these controllers is to introduce redundant adjustments in the coefficients of error equation (2.6.6) and to increase the order to $r+1$. For example if a PI controller is used with a plant having relative order

r, (2.6.7) is

$$e^{(r)} + \beta_{r-1}e^{(r-1)} + \dots + \beta_0 e + k_1 e + k_2 \int_0^t e dz = 0. \qquad (2.6.8)$$

where k_1 and k_2 are the proportional and integral gain constants respectively. Differentiating to remove the integral and combining terms yields

$$e^{(r+1)} + \beta_{r-1}e^{(r)} + \dots + (\beta_0 + k_1)\dot{e} + k_2 e = 0. \qquad (2.6.9)$$

For a PID controller the result is

$$e^{(r+1)} + \beta_{r-2}e^{(r)} + \dots + (\beta_1 + k_3)\ddot{e} + (\beta_0 + k_1)\dot{e} + k_2 e = 0. \qquad (2.6.10)$$

where k_3 is the derivative gain constant.

The purpose of the external controller is to increase robustness to parameter variations and to aid in the rejection of process disturbances.

2.6.3 Generic Model Control

Generic model control (GMC) is a method which was proposed by Lee and Sullivan (1988). The basis for their procedure is to design the control law so that a particular closed loop response is realized. The plant, which can be multivariable, is described by the following equations:

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= h(x) \end{aligned} \qquad (2.6.11)$$

where x is an n dimensional state vector, u is the input vector, y is the output vector and f, g, and h are smooth functions. The functions f, g, and h can be nonlinear.

To derive a control law Lee and Sullivan choose as the desired output a second order response for the closed loop plant;

$$\ddot{y} + \frac{2\zeta}{\tau}\dot{y} + \frac{1}{\tau^2}y = \alpha(y_d,\dot{y}_d) \tag{2.6.12}$$

where α is a function of the desired output, y_d. The α function is chosen such that when (2.6.12) is integrated once then $\dot{y}$ will be a function of the tracking error, $e = y_d - y$. Integrating (2.6.12) yields

$$\dot{y} + \frac{2\zeta}{\tau}y + \frac{1}{\tau^2}\int_{t_0}^{t} y\,d\gamma = \int_{t_0}^{t}\alpha\,d\mu\,. \tag{2.6.13}$$

If α is chosen as

$$\alpha(y_d,\dot{y}_d) = \frac{2\zeta}{\tau}\dot{y}_d + \frac{1}{\tau^2}y_d \tag{2.6.14}$$

and (2.6.14) is substituted in the RHS of (2.6.13) the result is

$$\dot{y} = \frac{2\zeta}{\tau}(y_d - y) + \frac{1}{\tau^2}\int_{t_0}^{t}(y_d - y)\,d\mu\,. \tag{2.6.15}$$

For relative order one plants the time derivative of the plant output in (2.6.11) written in terms of the Lie derivative yields

$$\dot{y} = L_f h(x) + L_g h(x) u\,. \tag{2.6.16}$$

Setting (2.6.16) equal to (2.6.15) and solving for u the control law is

$$u = \frac{\frac{2\zeta}{\tau}(y_d - y) + \frac{1}{\tau^2}\int_{t_0}^{t}(y_d - y)\,d\mu - L_f h(x)}{L_g h(x)}\,. \tag{2.6.17}$$

The first two terms in the numerator of (2.6.17) can be recognized as a PI controller operating on the trajectory error. Output response can be altered by adjustment of tuning parameters ζ and τ.

Note that GMC is derived from Figure 2.5 when $r=1$ by choosing $V=0$, $\beta_0=0$, and $f(e)$ as the output from a PI controller. Some of the desirable features claimed for GMC are the inherent incorporation of the process model in the control law, disturbance compensation for measurable disturbances when they are included in the plant model (2.6.11), and the presence of integration in the control law. It should be noted that the first two characteristics are the result of the feedback linearization design procedure and are not unique to GMC. The last factor is due to the choice of a second order output response which has to be integrated once in order to obtain the control law. The simplicity of this method depends on having a plant model in control affine form and on the relative order of the plant. Affine form means that the control u appears linearly in the equation even though the coefficient multiplying it may be nonlinear (e.g. $g(x_1,x_2)u$ where $g(x_1,x_2)$ is a nonlinear function of x). Implicit in (2.6.16) is the assumption that the plant has relative order one. This insures that $L_g h(x) \neq 0$ so that the first derivative of the output results in an equation involving u. For plants having relative order greater than one, higher order derivatives of (2.6.16) are needed to find the control law. Then selection of the α function is not obvious and the simplicity of the method is lost.

Since the class of plants for which this technique is viable are of relative order one, the I/O relationship for any arbitrary input V is

$$\dot{y} = V \tag{2.6.18}$$

when the control law is derived from (2.6.16). Thus the plant could be easily controlled by a pole placement controller which in this case involves choosing only one β parameter.

Figure 2.6 shows the GMC system. GMC cannot track a time-varying setpoint with asymptotically zero tracking error because the input to the control law does not contain the derivative of the desired output. The derivative can be included by defining the α function as

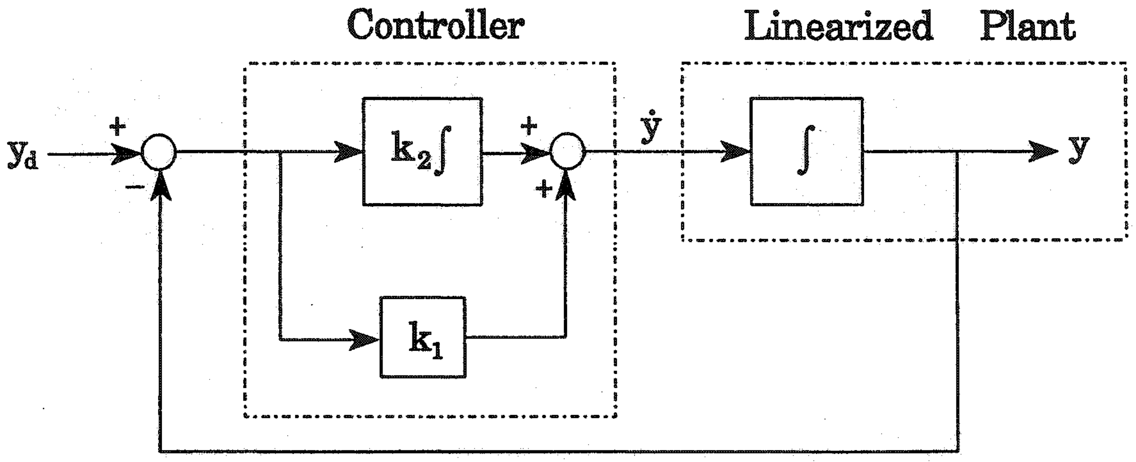

Figure 2.6 GMC block diagram.

$$\alpha(y_d,\dot{y}_d,\ddot{y}_d) = \ddot{y}_d + \frac{2\zeta}{\tau}\dot{y}_d + \frac{1}{\tau^2}y_d\,. \tag{2.6.19}$$

This α produces the control law

$$u = \frac{\dot{y}_d + \frac{2\zeta}{\tau}(y_d - y) + \frac{1}{\tau^2}\int_{t_0}^{t}(y_d - y)\,d\mu - L_f h(x)}{L_g h(x)} \tag{2.6.20}$$

that contains the $\dot{y}_d$ needed for signal tracking. In contrast with GMC, signal tracking in a feedback linearized system with the configuration shown in Figure 2.4 can be achieved simply by proper specification of the arbitrary input signal, V.

2.6.4 Reference System Synthesis

Reference system synthesis (RSS) is a state space model based method for designing feedforward/feedback controllers proposed by Bartusiak et al. (1989). The control objective is to produce a desired closed loop behavior. Either linear or nonlinear controllers can be designed and the method is applicable to both linear and nonlinear plants.

Assuming that the plant model is known, controller design is a two step procedure. First the desired closed loop servo response is chosen. This amounts to selecting the type of control action since the control law will cancel the plant dynamics. For relative order one plants the controller output is equal to the first derivative of the plant output. The controller time response is defined to be the time derivative of the so called reference system trajectory. The controller response is set equal to the derivative of the plant output and the resulting equation is solved for the plant input. Note that this procedure assumes a plant of relative order one. Incorporating disturbance rejection capability into the controller design is possible if the disturbance is measurable. A simple example will illustrate the basic design procedure for a first order linear SISO plant represented by

$$\dot{x}_p = -ax_p + bu + cd \tag{2.6.21}$$

where u is the plant input and d is a disturbance. If the closed loop response is chosen to be a first order lag with adjustable time constant τ and unity gain then

$$\dot{x}_r = g(V - x_p) \tag{2.6.22}$$

where $\dot{x}_r$ denotes the derivative of the reference trajectory, $g = 1/\tau$, V is the system input, and x_p is the plant output. The control law is found by solving

$$\dot{x}_p - \dot{x}_r = 0 \tag{2.6.23}$$

for u and is given by

$$u = \frac{1}{b}\left[ax_p + g(V - x_p) - cd\right]. \tag{2.6.24}$$

Using this control law in (2.6.21) the closed loop response is

$$X_p(s) = \frac{g}{s+g}V(s) \tag{2.6.25}$$

as desired.

Proportional, integral, and derivative action can be included in the closed loop response by specifying these terms in the reference system. For example, choose

$$\dot{x}_r = k_1(x_{sp} - x_p) + k_2\int_0^t (x_{sp} - x_p)dt + k_3\frac{d}{dt}(x_{sp} - x_p). \tag{2.6.26}$$

RSS is also depicted by Figure 2.5 for $r = 1$ by choosing $V = 0$, $\beta_0 = 0$, and $f(e)$ as the output from a particular controller type. The same procedure as given above can be used to design controllers for nonlinear plants. Also the reference system may be chosen to be nonlinear. For example it could involve coefficients that are functions of the absolute value of the trajectory error. This will result in a control law that responds with strong corrective action for large trajectory errors and more moderately for small errors.

The key step in this technique is the designers choice of reference trajectory. Solving for the control law is straight forward for plants of relative order one. RSS can be applied to higher relative order plants but with considerably more tedious calculation. In common with GMC design, RSS will lead to a control law that incorporates the plant model. It also has the same signal tracking problem as GMC.

2.6.5 Global Linearizing Control

Global linearizing control (GLC) was proposed by Kravaris and Chung (1987). Their stated purpose is to design a feedback controller for trajectory tracking for single input/single output nonlinear systems. The class of plants considered can be represented by

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= h(x)\,. \end{aligned} \tag{2.6.27}$$

The steps in development of the method consist of:

1. deriving the I/O linearizing plant input
2. adding feedback from the output of each integrator in the linearized plant (i.e. close the outer loop)
3. choosing the driving function for the feedback linearized plant as the output from a PI controller
4. closing the external loop through the PI controller with the setpoint equal to the desired output signal.

For the general case, assuming a relative order *r* plant, the control law is found as follows,

$$y^{(r)} = L_f^r h(x) + L_g L_f^{r-1} h(x) u \tag{2.6.28}$$

where $y^{(r)}$ is the *r'th* derivative of the linearized plant output. This derivative is also equal to some arbitrary input signal plus the negative feedback from the output of each integrator.

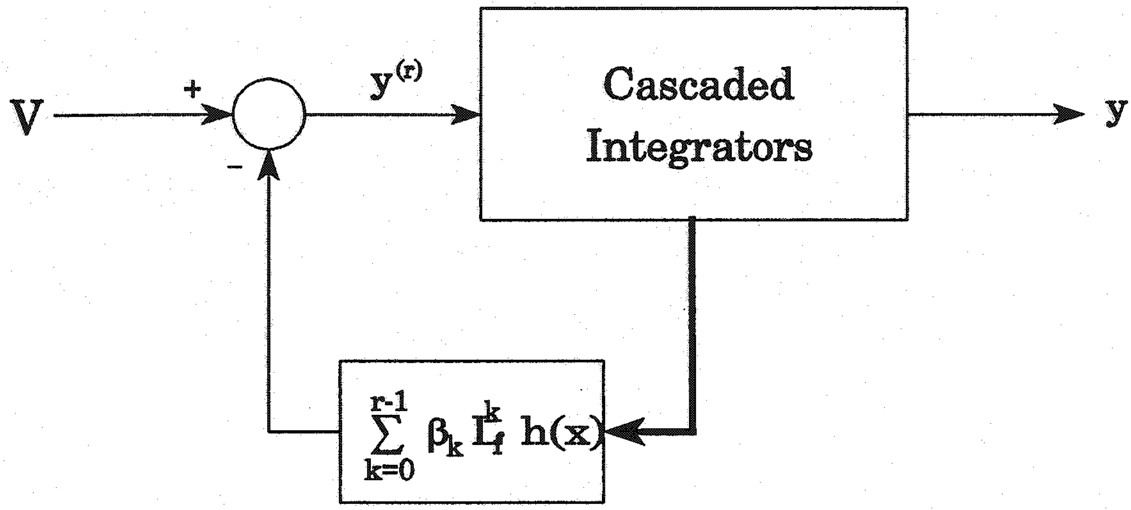

Figure 2.7 GLC equivalent outer loop block diagram.

Referring to Figure 2.7,

$$y^{(r)} = V - \sum_{k=0}^{r-1} \beta_k L_f^k h(x) . \tag{2.6.29}$$

From (2.6.28) and (2.6.29),

$$V = \sum_{k=0}^{r-1} \beta_k L_f^k h(x) + L_f^r h(x) + L_g L_f^{r-1} h(x) u . \tag{2.6.30}$$

Combining terms in (2.6.30) yields

$$V = \sum_{k=0}^{r} \beta_k L_f^k h(x) + L_g L_f^{r-1} h(x) u \tag{2.6.31}$$

where $\beta_r = 1$. From (2.6.31) the control law is

$$u = \frac{V - \sum_{k=0}^{r} \beta_k L_f^k h(x)}{L_g L_f^{r-1} h(x)} . \tag{2.6.32}$$

The input V can be chosen to be the output from any controller type. In the GLC method it is chosen to be the output from a PI controller. With the external feedback loop closed through a PI controller the control law is

$$u = \frac{k_1(y_d - y) + k_2 \int_0^t (y_d - y)dz - \sum_{k=0}^{r} \beta_k L_f^k h(x)}{L_g L_F^{r-1} h(x)} . \tag{2.6.33}$$

The external loop setpoint is y_d. The system is tuned by first specifying the$\beta_{k's}$ and then tuning the PI controller. GLC as originally presented by Kravaris and Chung (1987) has the same signal tracking problem as GMC and RSS. This difficulty was corrected in a later revision of the GLC method (Soroush and Kravaris (1992)). The modification, which was called a bias, is actually the input V shown in Figure 2.4. This signal contains $y_d^{(r)}$ which is needed for tracking.

The original version of GLC is shown by Figure 2.5 if V is chosen to be zero. The later modified version is identical to Figure 2.5. As indicated in the figure, it can be used with relative order r plants.

2.6.6 GMC, RSS, and GLC Control Of A Bilinear Plant

The three methods reviewed above produce identical control systems since for relative order one plants they are just different statements of the same objective. Consider the first order bilinear plant represented by

$$\begin{aligned} \dot{x} &= -ax + (b-x)u \\ y &= x , \end{aligned} \tag{2.6.34}$$

Assume that the control objective is to track a sinusoidal input signal, $y_d = \sin(\omega t)$. Designing the control system for this plant by any of the methods with a PI controller in the external loop leads to the control law

$$u = \frac{1}{b-x}\left[ax + k_1 e(t) + k_2 \int_0^t e(z)dz\right] \tag{2.6.35}$$

where $e = y_d - y$. The GLC method normally has the outer loop closed. Control law (2.6.35) assumes $\beta_0 = 0$ for the GLC method. However there is no loss in generality since closing the outer loop for GLC will not improve the tracking response.

Applying simple feedback linearization to this plant yields the control law

$$u = \frac{1}{b-x}\left[V + ax\right]. \tag{2.6.36}$$

For tracking in this system the input is set equal to the desired output plus its derivative

$$V = \beta_0 \sin(\omega t) + \omega\cos(\omega t). \tag{2.6.37}$$

Figures 2.8, 2.9, and 2.10 show the tracking capabilities of the GMC/RSS/GLC method versus the simpler feedback linearization scheme with a pole placement controller with $\beta_0 = 1$. Plant parameters were set at $a = 0.5$ and $b = 1.34$. The tracking signal frequency was chosen as $\omega = \pi/4$ and the initial plant output was set at $y(0) = -0.25$. Controller tuning was set at $k_1 = 5.0$, and $k_2 = 0.5$.

Response for the feedback linearized system, shown in Figure 2.8, shows excellent tracking after an initial transient error. Figure 2.9 shows the constant tracking error which is characteristic of the GMC/RSS method and the GLC method as originally proposed.

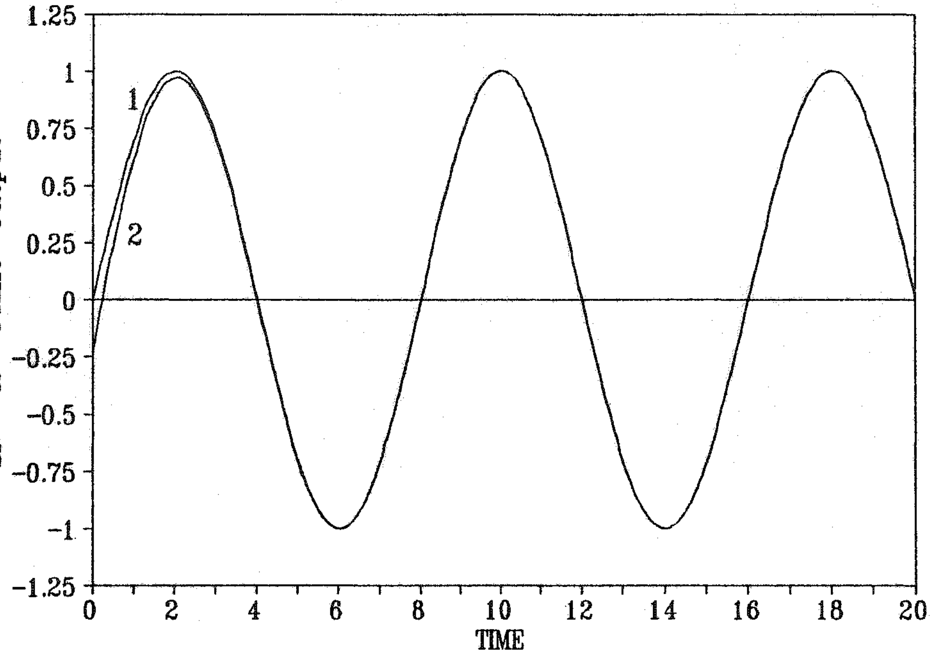

Figure 2.8 Tracking response for feedback linearized plant: curve 1, setpoint; curve 2, plant output.

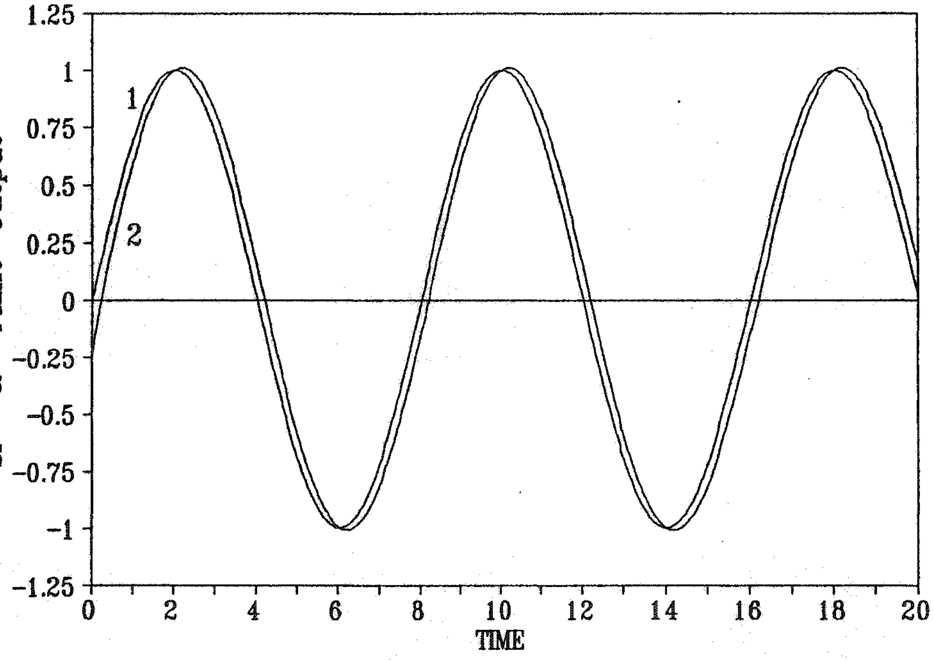

Figure 2.9 Tracking response using either GMC, RSS or GLC controller: curve 1, setpoint; curve 2, plant output.

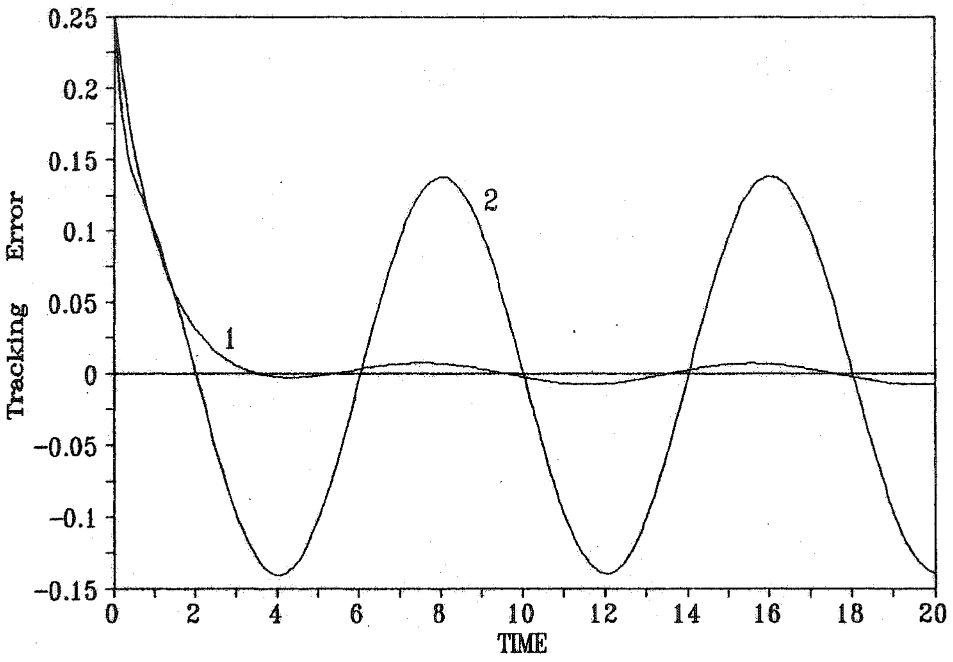

Figure 2.10 Tracking error comparison: curve 1, feedback linearized plant, curve 2, GMC, RSS or GLC controller.

Figure 2.10 is a comparison of the tracking errors in Figures 2.8 and 2.9.

2.7 Adaptive Control Of Nonlinear Plants

Many of the recent proposals for control of nonlinear plants are based on the feedback linearization technique. For this method to be effective the plant parameters have to be accurately known and hence the motivation for an adaptive design.

When plant parameters vary in a predictable way then adaptation of a different type can be effective. An example of this is the gain scheduling controller. In this section several different types of adaptive schemes are reviewed.

2.7.1 Self-Tuning Nonlinear System Controller

The nonlinear controller designs discussed so far have been based on state space plant models. Agarwal and Seborg (1987) have proposed a self-tuning controller based on a modified ARMAX model. The modification is replacement of the input term $q^{-d}B(q^{-1})u(t)$ on the RHS of the standard ARMAX model with a summation which can involve products of powers of the input and output with many types of nonlinearities.

The nonlinear discrete time plant is modeled as

$$A(q^{-1})y(t) = \sum_{i=1}^{N} b_i u^{r_i}(t-k)Y_i(t-1) + d + c(q^{-1})\xi(t) \qquad (2.7.1)$$

where t is the sampling instant, k is a known time delay including the sampler delay, $y(t)$ is the measured output, $u(t)$ is the manipulated variable, d is an unknown disturbance, and ξ(t) is zero mean random noise. $A(q^{-1})$ and $c(q^{-1})$ represent polynomials in the backward shift operator q^{-1} (i.e. $q^{-1}y(t) = y(t-1)$). The term $Y_i(t\text{-}1)$ represents nonlinearities involving single valued time invariant functions with known parameters. Powers of the input are handled through the choice of the r_i parameters.

The control law is derived by minimizing the performance index

$$I = \mathrm{E}\Big\{\big[P(q^{-1})y(t+k) - R(q^{-1})W(t)\big]^2 + \big[Q'(q-1)u(t)\big]^2\Big\} \qquad (2.7.2)$$

where E is the expectation operator, $W(t)$ is the setpoint, and $P(q^{-1})$, $R(q^{-1})$, and

$Q'(q^{-1})$ are rational transfer functions. Minimization of (2.7.2) leads to a polynomial in u(t) of order

$$\gamma = 2[\max_i(r_i)] - 1 . \tag{2.7.3}$$

The plant input has to be chosen from the γ roots of this polynomial. Real roots that do not violate input constraints are feasible inputs. If more than one root remains as a feasible input the value closest to the last input can be chosen. If $r_i \leq 1,\ i=1,2,.....N$ in (2.7.1) then u(t) can be found explicitly.

This method requires converting the natural and convenient differential equation description of a plant into a discrete transfer function form for which a self-tuning controller is then designed. The justification is the inclusion of the plant nonlinearities in the control law. However this can be done when working directly with the plant differential equation so that the conversion process seems to be unjustified. On the other hand, this method does not require the plant to be in control affine form.

2.7.2 Gain Scheduling

One of the first adaptive schemes introduced was gain scheduling (Ioannou and Sun (1996)). The name derives from the fact that it was originally used to compensate only changes in plant gain. In this approach auxiliary variables are found which are related to plant dynamics in a known way. As conditions in the plant change beyond prespecified limits, as indicated by the auxiliary variables, the controller parameters are set to new predetermined values which are optimal over the current range of plant dynamics.

Gain scheduling is an effective method for controlling plants with known nonlinearities. It also has the advantage that controller parameters can be changed quickly in response to plant conditions. One practical drawback is the determination of the large number of schedules that are needed for some plants. For example the flight control system for the CH-47 helicopter was implemented using ninety schedules (*i.e.* different controller settings) (Sastry and Bodson (1989)). It is also an open loop system in the sense that plant output does not influence schedule selection. If for some reason the plant dynamics originally associated with an auxiliary variable in a certain operating range should change, the controller settings are not automatically updated.

2.7.3 High Gain Adaptation

This scheme uses an adjustable high gain in cascade with the controlled plant (Sastry and Bodson (1989)). The idea here is to keep the open loop gain as high as possible under all operating conditions so that the closed loop gain from input to output is equal to one and perfect tracking of the input signal is achieved. The input signal is generated by the output from a reference model.

The controller gain is automatically adjusted by detecting the existence of a limit cycle and reducing or increasing the gain to keep the amplitude of the limit cycle within specified bounds. The constant presence of oscillation, even though it may be of low amplitude, is obviously an undesirable feature of this system.

2.7.4 Adaptive Tracking

An adaptive tracking scheme has been developed by Han (1992) for a class of pure feedback nonlinear systems (Su and Hunt (1986)). In this method the state variables of the nonlinear plant are assumed to be measurable. The plant parameters are assumed unknown and are estimated on-line. For a SISO system the plant model is

$$\begin{aligned} \dot{x} &= f(x,p) + g(x,p)u \\ y &= x_1 \end{aligned} \qquad (2.7.4)$$

where p indicates the parameter vector.

The design methodology is derived by first defining the differences in the vector fields $f(x,p)$ and $f(x,\hat{p})$, $g(x,p)$ and $g(x,\hat{p})$, where $\hat{p}$ represent the estimated parameter vector. The difference equations are defined as

$$\begin{aligned} \Delta f(x,p,\tilde{p}) &\doteq f(x,p) - f(x,\hat{p}) \doteq \psi(x)\tilde{p} \\ \Delta g(x,p,\tilde{p}) &\doteq g(x,p) - g(x,\hat{p}) \doteq \psi_0(x)\tilde{p} \\ \tilde{p} &\doteq p - \hat{p} \,. \end{aligned} \qquad (2.7.5)$$

Applying the control $u(x,\hat{p})$ to the plant (2.7.4) results in the closed loop linear system where the transformed states are $\hat{y} = \alpha(x,\hat{p})$. Therefore

$$\frac{d\hat{y}}{dt} = \frac{\partial\hat{y}}{\partial x}\dot{x} + \frac{\partial\hat{y}}{\partial\hat{p}}\frac{d\hat{p}}{dt} \,. \qquad (2.7.6)$$

Expanding equation (2.7.6) yields

$$\frac{d\hat{y}}{dt} = \left(A + bk^T\right)\hat{y} + br + Y_x\tilde{p} + Y_p\frac{d\tilde{p}}{dt}. \tag{2.7.7}$$

where

$$Y_x = \frac{\partial\hat{y}(x,\hat{p})}{\partial x}\left(\Psi(x) + \Psi_0(x)\right)$$

and

$$Y_p = -\frac{\partial\hat{y}(x,\hat{p})}{\partial p}.$$

The feedback gain vector is denoted by k. The first two terms on the RHS of (2.7.7) are the vector matrix representation of the linear closed loop system with input r. The last two terms are due to parameter estimate errors. Setting r equal to the tracking signal

$$r = y_d^{(n)} + d_1 y_d^{(n-1)} + \dots + d_n y_d \tag{2.7.8}$$

and substituting into (2.7.7) yields the error equation

$$\dot{e} = A_1 e + Y_x\tilde{p} + Y_p\frac{d\tilde{p}}{dt}. \tag{2.7.9}$$

This is the key equation in adaptive strategies because it relates measureable system errors to errors in parameter estimates. From it a parameter updating law can be deduced and, with the use of Lyapunov theory, the overall stability of the adaptive system can be established.

The advantages claimed for the method are a smaller computational load, and neither overparameterization nor matching conditions are required as they are with some other methods. However preliminary setup for applying this method is a fairly complex process.

2.8 Adaptive Control Of Linear Plants

Adaptive control for linear plants was motivated by the desire to improve on the performance of fixed parameter controllers when process dynamics are uncertain or varying. Historically flight control systems for aircraft were among the first applications of adaptive control. Due to many technical advances over the years, in both theory and computing hardware, adaptive control of linear plants is used in a broad spectrum of applications today. This research is concerned with the control of nonlinear plants but since linear plant adaptive control systems have many of the features that are needed in adaptive or self-tuning control of nonlinear systems, we look at them first. For example, when parameters are unknown and estimation is done on-line then the feedback linearization problem is very similar to that of standard adaptive control methods.

In general, adaptive controllers consist of:

1. a control law having adjustable parameters
2. a performance index or control strategy

   (*e.g.* output from a reference model or a design criteria)
3. an on-line parameter updating mechanism.

A considerable amount of research effort has gone into developing strategies for implementing the above steps and many variations have been reported in the literature.

Two different approaches have emerged for implementing adaptive control strategies. The indirect approach is shown in Figure 2.11. The controller is designed such that the controller parameters $\hat{\theta}_c$ are functions of the plant parameters, $\hat{\theta}_p$. The scheme is made adaptive by replacing the plant parameters $\hat{\theta}_p$ in the controller design by their current estimates at time *t*. (*i.e.* certainty equivalence principle) The updated plant parameter estimates are supplied by a recursive estimator.

Several different combinations of controller types and parameter estimators can be considered. Although there is no guarantee that all combinations will produce equally good results. The inner loop consisting of the controller and plant is a standard feedback loop.

The direct approach does not identify plant parameters. An example of the direct approach is shown in Figure 2.12. The controller function in this scheme is to modify the plant transfer function so that the response of the combination of controller and plant to the reference signal matches that of the reference model. The logical way to adjust controller parameters in this case is to make them a function of the output error. As with the indirect scheme, the inner loop is a conventional feedback control loop.

There are two basic types of adaptive control systems, model reference adaptive control (MRAC) and self-tuning regulators (STR). Either of these can be implemented in the direct or indirect mode but the most common embodiments

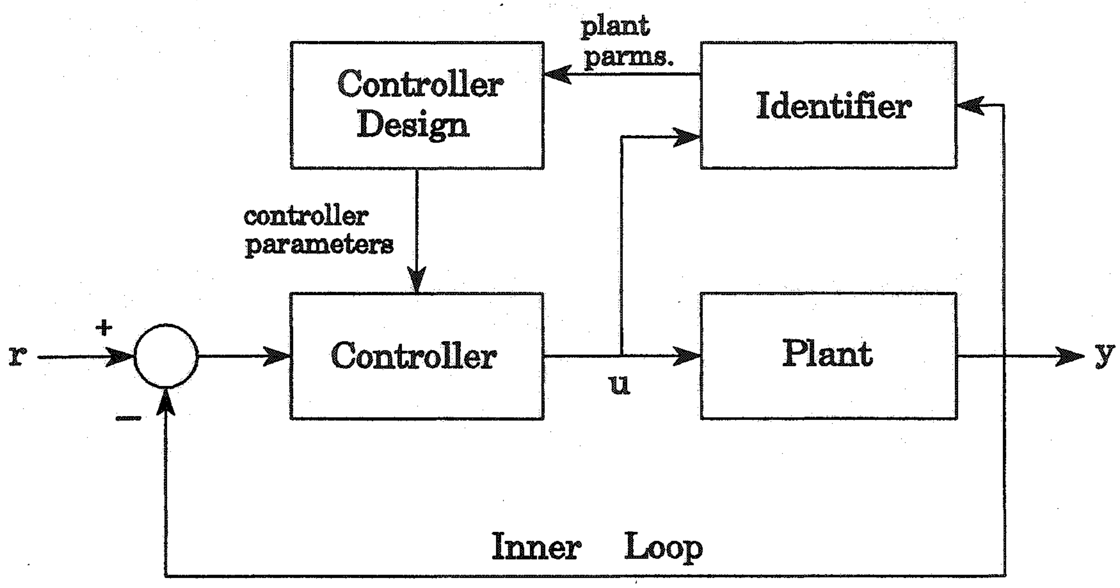

Figure 2.11 Indirect adaptive control, (STR).

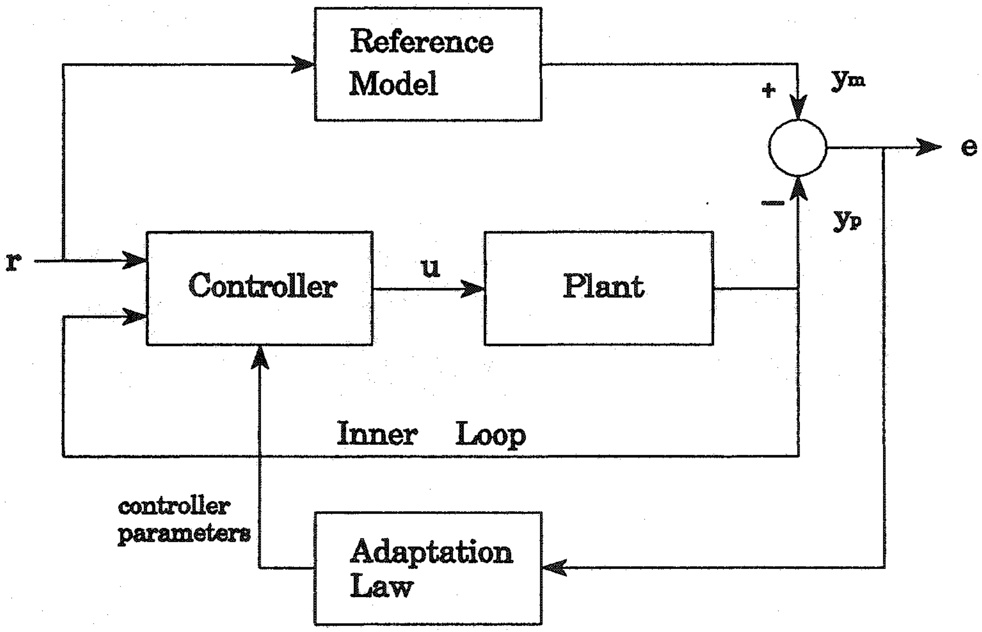

Figure 2.12 Direct adaptive control, (MRAC).

are indirect for STR's and direct for MRAC. The block diagrams are shown in Figures 2.11 and 2.12 respectively.

The performance criteria in MRAC is to asymptotically reduce the output tracking error to zero. The design method seeks to accomplish this objective by first ensuring stable dynamics for the error response. This perspective was first proposed by Grayson (1963) and is the cardinal rule in the design methods he presents. A simple demonstration of how first ensuring system stability leads to stable adaptive laws is the following.

Assume that the plant with unknown parameters is given by

$$\dot{x}_p = a_p x_p + k_p u \tag{2.8.1}$$

and the reference model is represented by

$$\dot{x}_m = a_m x_m + k_m r \tag{2.8.2}$$

where r is the reference input. From (2.8.1) if we were to apply a control input

$$u = \frac{a_m - a_p}{k_p} x_p + \frac{k_m}{k_p} r \tag{2.8.3}$$

the plant output would indeed match the reference model output. Therefore we may express (2.8.3) as

$$u = \hat{\theta} x_p + \hat{k} r \tag{2.8.4}$$

where $\hat{\theta}$ and $\hat{k}$ are defined as the controller parameters to be adapted. We now define the output error and the parameter errors as

$$e \doteq x_p - x_m, \quad \alpha \doteq \hat{\theta} - \theta^*, \quad \rho \doteq \hat{k} - k^* \tag{2.8.5}$$

where θ^* and k^* denote the optimum values. Substituting the plant input (2.8.4) into (2.8.1) and using the above definitions we derive the output error differential equation

$$\dot{e} = -a_m e + k_p \alpha x_p + k_p \rho r \tag{2.8.6}$$

where the fact that coefficient a_m in the reference model is negative has been utilized.

To get stable adaptive laws and at the same time ensure system stability we choose the Lyapunov function candidate

$$V(e,\alpha,\rho) = \frac{1}{2}\left[e^2 + |k_p|\left(\alpha^2 + \rho^2\right)\right]. \tag{2.8.7}$$

This leads to

$$\dot{V} = -a_m e^2 + k_p \alpha e x_p + k_p \rho e r + |k_p| \alpha \dot{\alpha} + |k_p| \rho \dot{\rho}\,. \tag{2.8.8}$$

To ensure that $\dot{V}$ is negative definite all the RHS terms of (2.8.8) except the first have to be eliminated. In so doing the global stability of the system is assured and adaptive laws for the controller parameters will automatically be defined. Therefore let

$$\begin{aligned} |k_p| \alpha \dot{\alpha} &= -k_p \alpha e x_p \\ \therefore \quad \dot{\alpha} &= -sgn(k_p) e x_p \end{aligned} \tag{2.8.9}$$

and

$$\begin{aligned} |k_p| \rho \dot{\rho} &= -k_p \rho e r \\ \therefore \quad \dot{\rho} &= -sgn(k_p) e r\,. \end{aligned} \tag{2.8.10}$$

Note that the adaptive laws are expressed in terms of measurable system signals.

However it is necessary to know the sign of the plant gain.

Parameter estimates may not converge to their theoretical optimum values. Recall that the performance criterion only requires that the tracking error be reduced to zero. Exact convergence depends on the plant input signal being persistently exciting.

While this simple example demonstrates the concept of designing a stable adaptive system, unfortunately it can only be directly applied to relative order one plants. For higher relative order plants different techniques are used (Slotine and Li (1991), Narendra and Annaswamy (1989), Sastry and Bodson (1989), Krstic et. al.(1995).

Self-tuning regulators generally fall under the classification of indirect adaptive controllers. The self-tuning strategy can be stated as follows: combine an on-line parameter estimator with a control system design technique to produce a control law which is capable of self-optimization. Parameter estimation can be done by any of the standard recursive algorithms such as least squares, instrumental variables, recursive maximum likelihood, etc. In contrast with MRAC, in order for the controller design to be realized, accurate parameter estimates are needed in STR systems. For example pole-placement is often used in the design of STR's. One drawback with STR systems is the difficulty of proving stability. In these systems the parameter estimation algorithm is simply chosen and does not guarantee a stable system as the adaptive laws did in MRAC.

2.9 Summary

Since there has been no general method for designing controllers for nonlinear plants, with the possible exception of the recently developed backstepping method, what has evolved over a period of time is a number of complimentary techniques. Several of these have been reviewed in this chapter. The classical techniques such as the phase plane, describing function, etc. are used to guide a controller design which is later evaluated by simulations.

The modern technique of feedback linearization allows the designer to first of all determine the linearizability of the given plant. If linearizable, then a control law and coordinate transformation can be found which results in, for example, a linear input/output response in the new coordinate system. The basic mathematical tools needed to test for linearizability and carrying out a feedback linearizing design have been reviewed in section 2.4.

Feedback linearization *per se* results in an equivalent system which is represented by a cascaded string of integrators. The next step is the design of a controller for the linearized plant. This can be done in a straight forward manner by using a pole placement controller. If the conditions require it, then an additional control loop (i.e. the external loop) to improve robustness to parameter errors and disturbances can be added. (c.f. section 2.6.2)

System designs which closely follow the above design procedure are reviewed in sections 2.6.3, 2.6.4, and 2.6.5. These systems are known as Generic Model Control, Reference System Synthesis, and Global Linearizing Control.

These three techniques are nearly identical for relative order one plants. However GLC is slightly different from the other two methods since it makes use of an outer feedback loop. (c.f. Figure 2.5) The basic procedure is to first linearize the plant. Then a particular type of control action is chosen to drive the linearized plant and an external feedback loop is closed through this controller. The controller setpoint becomes the desired tracking signal. These methods have poorer tracking capability than a feedback linearized system with an outer loop pole placement controller and properly specified input signal. However if plant parameters are uncertain they are more robust than a feedback linearized system with regard to eliminating steady state error. The tracking error problem in these strategies can be remedied by adding $\dot{y}_d$ to the control law. This addition was incorporated in later versions of GLC. The most versatile of the three methods is GLC since it is not limited for practical purposes to just relative order one plants.

Han (1992) has presented one of the better schemes for adaptive control. His method parallels that of MRAC for linear plants. Han presents a Lyapunov based stability proof for his method.

Agarwal and Seborg (1987) have developed a complex adaptive scheme around a modified ARMAX model. For most cases it seems that the relative simplicity of methods based on state-space models are to be preferred over this approach.

This chapter is concluded with a short discussion of MRAC and STR's for linear plants since the STR will be the starting point for development of a self-tuning feedback linearizing controller in subsequent chapters.

CHAPTER 3

Adaptive Feedback Linearization System Configuration

3.1 Introduction

Chapter 2 describes the techniques which can be used to design a nonlinear feedback law and how a nonlinear coordinate change can be made to produce a linear input-output response from a nonlinear system. Several nonlinear control schemes which are representative of the research in nonlinear control theory were also reviewed. The majority of these schemes attacked the nonlinear control problem via feedback linearization in some form. In order to implement feedback linearization it is necessary to have the parameters and states of the nonlinear plant available. Uncertainties in the model parameters result in incomplete cancellation of nonlinear terms and the desired linear input-output response is not achieved. This is the main drawback in practical implementation of linearizing control laws.

In some important applications the plant parameters are either time-varying or a nonlinear term in the plant model can be treated as a time-varying parameter. An example of the latter is given in Chapter 5 where a batch reactor with a nonlinear term is modeled as a plant with a time-varying parameter. To effectively deal with these cases some authors have suggested adaptive control strategies for feedback linearizable systems (Sastry and Isidori (1989), Nam and Arapostathis (1988)). These strategies closely parallel those developed for adaptive control of linear systems. In both the linear and nonlinear cases the design is based on an output error model. An adaptive law for updating parameters is derived from the system output error equation.

In this chapter it is shown that the classical adaptive scheme may not work well with feedback linearizing controllers. However a new technique to remedy the difficulties with estimating feedback linearizing parameters will be developed. The new method does not require an error equation like those mentioned above. Instead use is made of the *a-priori* knowledge of the linearized plant structure and how it is modified when errors exist in the nonlinear plant parameter estimates used in the control law. For example, when parameter estimates are in error, the partially linearized model may contain coefficients such as $a_i - \hat{a}_i$ where a_i represents the true parameter value and $\hat{a}_i$ is the estimated parameter value. In order to find the true value of a_i, the model is reparameterized in terms of the parameter difference. The new parameter is defined as

$$\theta_i \doteq a_i - \hat{a}_i \,. \tag{3.1.1}$$

A recursive estimation routine is then used to estimate θ_i. The ideal value for θ_i in (3.1.1) is zero. Therefore, deviation from the ideal value of θ_i provides information about the error in the parameter estimate, $\hat{a}_i$.

Other combinations of the true and estimated parameters may be present. These combinations are treated in a manner similar to the above. That is, they are manipulated into a form where the value of θ_i is known when $\hat{a}_i = a_i$.

Of course θ_i is not the variable needed in the control law but it represents the error in the estimate of the variable, a_i, which is needed. In some cases θ_i is a direct representation of the error between a_i and $\hat{a}_i$ as shown by (3.1.1). In other cases it is only an indirect measure of the error. For either case, given the error estimate, an adaptive law can be devised which uses the error information contained in θ_i to provide an improved estimate of a_i. This results in a simpler scheme than those proposed in the references cited above for estimation of the control law parameters. Only a standard algorithm is needed to do the parameter estimation.

In section 3.3 the classical adaptive control system architecture is evaluated for use in feedback linearization applications. The results of this evaluation indicate that under certain conditions the classical architecture can have some serious drawbacks when used with linearizing control laws. Section 3.4 suggests a modified architecture which eliminates the shortcomings of the classical configuration for feedback linearizing applications.

3.2 Parameter Estimation For Adaptive Feedback Linearized Systems

An adaptive system has been defined as one that monitors its own performance and adjusts its parameters to achieve better performance (Drenick and

Shawbender (1957)). Feedback linearized systems achieve better performance by neutralizing the effects of unwanted nonlinear response terms in the plant. As mentioned in Chapter 1, the control law which linearizes a plant is a function of both plant parameters and states. In many cases parameters are unknown or may be changing with time or plant operating conditions. The next sections review parameter estimation in a closed loop and a well known algorithm for combined parameter and state estimation.

3.2.1 Closed Loop Parameter Estimation

Estimation of plant parameters in an open loop mode for linear plants is fairly straight forward. Parameter estimation in closed loops presents some unusual difficulties. It is well known that unless certain necessary and sufficient conditions are met it is impossible to estimate the open loop plant parameters in a closed loop (Isermann (1981)). In adaptive systems, parameter estimation is done under closed loop conditions.

A problem arising in closed loop parameter estimation can be easily demonstrated with the aid of an ARX (Auto Regressive with eXogeneous variable) model (Gustavsson et.al. (1977)). Consider the discrete time system

$$y(t) + ay(t-1) = bu(t-1) + e(t) \tag{3.2.1}$$

controlled by a known proportional regulator

$$u(t) = gy(t)\,. \tag{3.2.2}$$

Substituting (3.2.2) into (3.2.1) yields

$$y(t) + (a - bg)y(t-1) = e(t)\,. \tag{3.2.3}$$

If a and b are unknown parameters it is obvious from (3.2.3) that neither can be uniquely determined. Only the composite quantity, $(a\text{-}bg)$, can be estimated. Furthermore any estimates of a and b satisfying

$$\begin{aligned} \hat{a} &= a + \gamma g \\ \hat{b} &= b + \gamma \end{aligned} \tag{3.2.4}$$

where γ is arbitrary will yield the same value for $(a - bg)$. Replacing a and$b$ in $(a - bg)$ by $\hat{a}$ and $\hat{b}$ from (3.2.4) yields the same result as the true values, e.g.

$$\hat{a} - \hat{b}g = a + \gamma g - (b + \gamma)g = a - bg\,. \tag{3.2.5}$$

The basic problem is that the regulator (3.2.2) may convert some of the moving average terms of the plant model into terms identical to those in the autoregressive part of the plant model. This results in parameter combinations such as illustrated above and makes identification of individual parameters impossible.

Necessary and sufficient conditions for which the open loop plant model parameters can be identified from closed loop data have been developed by Soderstrom and Stoica (1989) and many others. Consider the general ARMAX (Auto Regressive Moving Average with eXogeneous variable) plant represented by

$$A(q^{-1})y(t) = q^{-k}B(q^{-1})u(t) + C(q^{-1})e(t) \tag{3.2.6}$$

and controlled by the proportional regulator

$$R(q^{-1})u(t) = -S(q^{-1})y(t). \tag{3.2.7}$$

In this model the moving average term is the white noise contribution represented by $C(q^{-1})e(t)$ and $u(t)$ is the exogeneous variable. Polynomials $A(q^{-1})$, $C(q^{-1})$, and $R(q^{-1})$ are monic and of order na, nc, and nr respectively. Polynomials $B(q^{-1})$ and $S(q^{-1})$ are not monic and have order nb and ns. The structure of the estimated model is chosen to be similar to that of the plant, i.e.

$$\hat{A}(q^{-1})y(t) = q^{-k}\hat{B}(q^{-1})u(t) + \hat{C}(q^{-1})\epsilon(t) \tag{3.2.8}$$

where $\epsilon(t)$ represents the innovation (i.e. the prediction error $y(t)-\hat{y}(t)$). If (3.2.8) identically represents (3.2.6) then

$$\epsilon(t) = e(t). \tag{3.2.9}$$

For the ARMAX plant (3.2.6) and the model given by (3.2.8) the necessary and sufficient identifiability condition in terms of the polynomial orders is

$$\max(nr-nb,\ k+ns-na) - 1 \geq nh \tag{3.2.10}$$

where nh is the order of any common factors in the polynomials $C(q^{-1})$ and

$$A(q^{-1})R(q^{-1}) - q^{-k}B(q^{-1})S(q^{-1}).$$

In closed loop systems, three basic approaches to parameter estimation have been suggested. They are direct identification, indirect identification, and joint identification. Direct identification prediction error methods ignores the closed loop and fits the model (3.2.8) to the plant I/O data. Indirect identification fits a model to the data $y(t)$ versus $d(t)$ where $d(t)$ is an auxiliary input to the plant. The open loop plant z transfer function is then calculated by using the known controller transfer function and solving the closed loop equation for the unknown plant transfer function, e.g.

$$\frac{Y(z)}{D(z)} \doteq G_{cl}(z) = \frac{G_p(z)}{1 + G_p(z)\,G_c(z)} \tag{3.2.11}$$

where $G_{cl}(z)$ is the closed loop transfer function from $D(z)$ to $Y(z)$, $G_c(z)$ is the controller transfer function, and $G_p(z)$ is the unknown plant transfer function. From (3.2.11)

$$G_p(z) = \frac{G_{cl}(z)}{1 - G_{cl}(z)\,G_c(z)} \tag{3.2.12}$$

The third method, joint identification, fits a canonical multivariate time series model to the output $y(t)$ and the input $u(t)$ jointly. The open loop model is then computed by rearrangement of the canonical time series model.

Inspection of the identifiability condition (3.2.10) shows that one way to insure identifiability is to use a sufficiently high order regulator. (i.e. to obtain a large nr and ns) Even though (3.2.10) guarantees identifiability, the estimates obtained from pure feedback data are quite poor (Box and MacGregor 1974).

Condition (3.2.10) applies when other special experimental conditions such as additional system inputs and use of more than one regulator during the course of the experiment are not present. By proper choice of experimental conditions it is still possible to estimate the open loop parameters of a closed loop system without satisfying condition (3.2.10). For example if two or more different regulators are used to control the plant during an experiment then the open loop plant can be identified even though condition (3.2.10) is not met. An explanation of why use of multiple regulators makes closed loop parameter estimation possible is given in Soderstrom and Stoica (1989), pg. 390.

According to Ljung (1987), the recursive prediction error method is the best method to use in closed loop cases. If it fails, so will all other methods.

3.2.2 Recursive State And Parameter Estimation

Accurate on-line determination of model parameters is essential for good performance in adaptive control systems. There are a great number of estimation algorithms described in the literature (Ljung and Soderstrom (1983), Sinha and Kuszta (1983), Soderstrom and Stoica (1989)). One of the most versatile is the recursive prediction error method (RPEM) for joint state and parameter estimation. This algorithm was originally proposed by Ljung (1979) and also by Moore and Weiss (1979). Using the RPEM, the effects of noise can conveniently be accounted for when the plant is represented by a state space innovations model. In this model the Kalman gain vector is explicitly parameterized and can be estimated on-line with the other parameters. This avoids the need to know noise variances necessary to calculate these gains in the standard or extended Kalman filters (Goodwin and Sin (1984)). The RPEM can be applied to any model for which the gradient of the prediction error can be computed. This is an important feature because we are ultimately interested in estimating the parameters for nonlinear plants.

Before reviewing the calculations in this algorithm it is convenient to choose a model for representing the plant. For illustration we will consider a state-space innovations model in controller canonical form,

$$\hat{x}_{t+1} = A(\theta)\hat{x}_t + B(\theta)u_t + K(\theta)\varepsilon_t \tag{3.2.13a}$$

$$\hat{y}_t = C(\theta)\hat{x}_t + \varepsilon_t \tag{3.2.13b}$$

where ε_t is the prediction error. The $A(\theta)$ matrix and $B(\theta)$ vector are given by

$$A(\theta) = \left[\begin{array}{ccccc|c} -a_1 & -a_2 & . & . & . & -a_n \\ - & - & - & - & - & - \\ & & & & & 0 \\ & & I_{n-1} & & & \vdots \\ & & & & & 0 \end{array}\right], \qquad B(\theta) = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{3.2.14a}$$

where I_{n-1} is an $(n-1)$ x $(n-1)$ unit matrix. C(θ) and K(θ) are given by

$$C(\theta) = \left[c_1, c_2, \ldots, c_n\right], \qquad K(\theta) = \begin{bmatrix} k_1 \\ k_2 \\ k_3 \\ \vdots \\ k_n \end{bmatrix}. \tag{3.2.14b}$$

Note that in this model the Kalman vector is directly parameterized for on-line estimation.

The RPEM is based on minimizing a function of the prediction error,

$$V_t(\theta) = \frac{1}{2}\sum_{s=1}^{t} \lambda^{t-s}\, \varepsilon_t^2(s,\theta) \tag{3.2.15}$$

where λ is the forgetting factor, ε_t is the prediction error and s is an index relating the powers of λ to the corresponding prediction error. In (3.2.15), λ is chosen to be less than one. This diminishes the influence that previous values of the prediction errors have on the performance index, $V_t(\theta)$. The prediction error is defined as

$$\varepsilon_t \doteq y_t - \hat{y}_t \ . \tag{3.2.16}$$

Equations (3.2.14a,b) contain a maximum of *3n* parameters,

$$\theta^T = \left[a_1 \ \dots \ a_n,\ c_1 \ \dots \ c_n,\ k_1 \ \dots \ k_n\right], \tag{3.2.17}$$

where a_i, c_i, and k_i, are elements of $A(\theta)$, $C(\theta)$, and $K(\theta)$ respectively. Define

$$\psi_t \doteq -\left[\frac{d\varepsilon_t}{d\theta}\right]^T = \left[\frac{d\hat{y}_t}{d\theta}\right]^T \qquad (a\ 3n\ vector) \tag{3.2.18}$$

where ψ_t is the negative gradient of the prediction error and hence provides a descent direction for the recursive minimization of $V_t(\theta)$. In order to compute ψ_t the following quantities are needed.

$$W_t \doteq \frac{d}{d\theta}\left[\hat{x}_t(\theta)\right], \qquad (\ an\ nx3n\ matrix) \tag{3.2.19}$$

$$D_t \doteq \frac{\partial}{\partial\theta}\left[C(\theta)\hat{x}_t\right]\Big|_{\theta=\theta_t}, \qquad (a\ 3n\ row\ vector) \tag{3.2.20}$$

$$M_t \doteq \frac{\partial}{\partial\theta}\left[A(\theta)\hat{x}_t + B(\theta)u_t + K(\theta)\,\varepsilon_t\right]\Big|_{\theta=\theta_t}, \tag{3.2.21}$$

(an nx3n matrix)

Using the above, ψ_t may be computed from

$$\psi_t = W_t^T C^T(\theta) + D_t^T \tag{3.2.22}$$

where the dynamics of W_t are

$$W_{t+1} = \Big[A(\theta) - K(\theta)\,C(\theta)\Big]W_t + M_t - K(\theta)D_t\,. \tag{3.2.23}$$

The parameter vector is given by (3.2.17) and the other two quantities needed by the algorithm are

$$D_t = \Big[0\ \ 0\ \ \ldots\ 0,\ \hat{x}_{1,t}\ \ \ldots\ \ \hat{x}_{n,t}\,,\ 0\ \ 0\ \ \ldots\ \ 0\Big] \tag{3.2.24}$$

and

$$M_t = \begin{bmatrix} \hat{x}_{1,t} & \ldots & \hat{x}_{n,t}\,, & 0 & \ldots & 0, & \varepsilon_t & 0 & \ldots & 0 \\ 0 & \ldots & 0, & 0 & \ldots & 0, & 0 & \varepsilon_t & \ldots & 0 \\ \vdots & \ldots & \vdots, & \vdots & \ldots & \vdots, & 0 & 0 & \ddots & 0 \\ 0 & \ldots & 0, & 0 & \ldots & 0, & 0 & \ldots & \ldots & \varepsilon_t \end{bmatrix}. \tag{3.2.25}$$

The RPE algorithm computations are the following:

1. Compute the prediction error

$$\varepsilon_t = y_t - \hat{y}_t \tag{3.2.26a}$$

2. Compute the adaptation gain

$$L_t = \frac{P_{t-1}\psi_t}{\lambda + \psi_t^T P_{t-1}\psi_t} \tag{3.2.26b}$$

3. Update the parameter estimates

$$\theta_t = \theta_{t-1} + L_t\varepsilon_t \tag{3.2.26c}$$

4. Update the covariance matrix

$$P_t = \frac{1}{\lambda}\left[P_{t-1} - \frac{P_{t-1}\psi_t\psi_t^T P_{t-1}}{\lambda + \psi_t^T P_{t-1}\psi_t}\right] \tag{3.2.26d}$$

5. Calculate the state predictions

$$\hat{x}_{t+1} = A_t\hat{x}_t + B_t u_t + K_t\varepsilon_t \tag{3.2.26e}$$

6. Predict the next output

$$\hat{y}_{t+1} = C_t\hat{x}_{t+1} \tag{3.2.26f}$$

7. Compute the gradient of $\hat{x}_{t+1}$

$$W_{t+1} = \left[A_t - K_tC_t\right]W_t + M_t - K_tD_t \tag{3.2.26g}$$

8. Compute the prediction error gradient

$$\psi_{t+1} = W_{t+1}^T C_t^T + D_{t+1}^T \tag{3.2.26h}$$

where A_t, B_t, C_t, and K_t are all functions of the latest parameter estimates θ_t calculated in step 3. The gradient transition equation, (3.2.26g), in step 7 represents a dynamic system which can become unstable during the course of parameter estimation. Accordingly the roots of

$$\det\left[zI - A_t + K_tC_t\right] \tag{3.2.27}$$

have to be checked on each iteration. A simple subroutine for determining if roots are strictly inside the unit circle is given in Ljung and Soderstrom (1983), pg. 486. If any root is found to be outside the unit circle then a simple method to avoid instability is to ignore the current estimates and let

$$\theta_t = \theta_{t-1} \,. \qquad (3.2.28)$$

A more sophisticated procedure is to successively halve the correction term $L_t\varepsilon_t$ in (3.2.26c) until θ_t produces stable roots as determined by equation (3.2.27).

The RPE algorithm will be used later in this chapter to implement simulations. It will be applied to a Runge-Kutta solution of the state-space representation of a continuous plant.[1]

3.3 Investigation of Standard System Architecture for Adaptive Feedback Linearization

Figure 3.1 shows the classical architecture which is commonly used in adaptive control of linear systems. The figure depicts the indirect adaptive case where the plant parameters are first determined by the identifier and then the controller design block calculates the controller parameters to achieve the particular type of control desired. The controller generates the manipulated variable u based on the error signal and controller parameters.

If the blocks in Figure 3.1 are rearranged and one new block is added, the architecture shown in Figure 3.1 could be considered for implementing a self-tuning feedback linearization scheme. The main difference is in the location and function of the controller design block. This block would be connected in series

[1] Use of RPEM with a Runge-Kutta model has not been found in any of the references checked and appears to be a new application with this type of model. For examples involving input-output models of bilinear plants see Faniech and Ljung (1987).

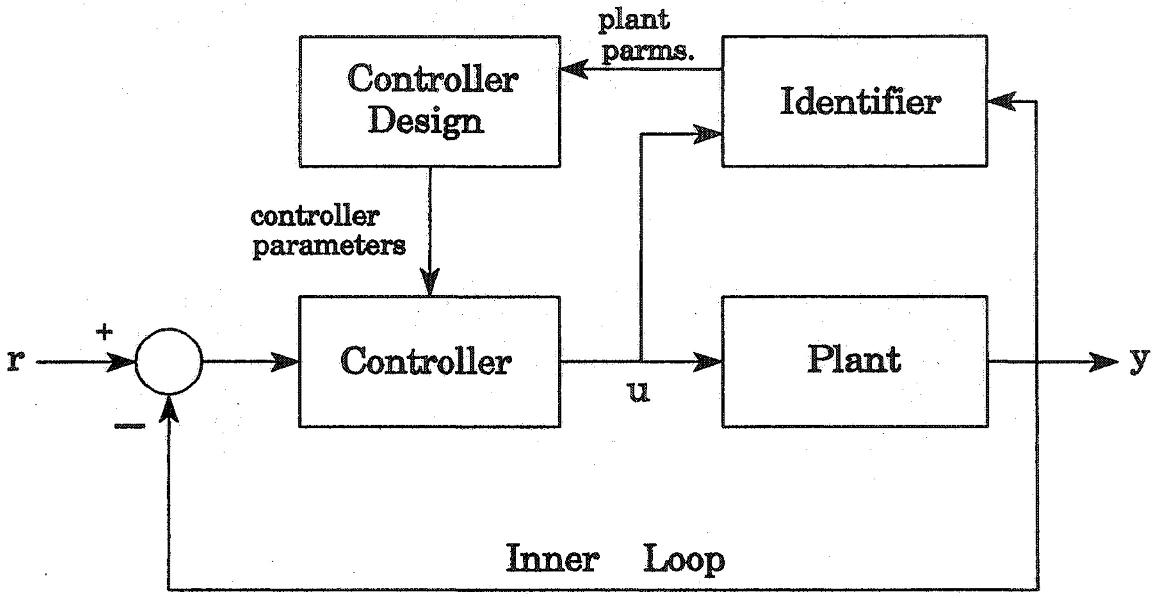

Figure 3.1 Indirect adaptive control system architecture for a linear plant.

between the controller and plant. Its function is to synthesize the control law. It would have as inputs the controller output and those plant states and estimated plant parameters that are needed in the control law. The output signal from this block is the feedback linearizing input which drives the plant and identifier.

The new additional block performs a nonlinear coordinate transformation. The new coordinates are the states of the linearized system and are fed back for control purposes. Figure 3.2 is the block diagram for the rearranged system.

Before using the architecture in Figure 3.2 for a feedback linearization scheme, we will investigate the performance characteristics in greater detail for this application. One desireable feature is the use of well known parameter

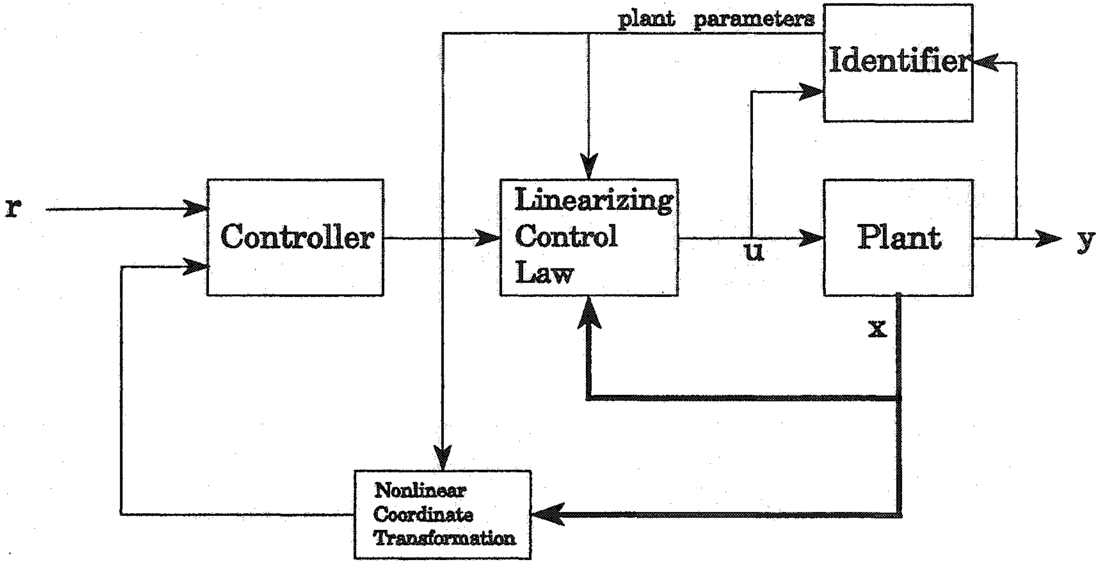

Figure 3.2 Rearrangement of Figure 3.1 for use in a feedback linearizing control system.

estimation algorithms. As pointed out in many references on adaptive control, any of the standard estimation routines can be used in this scheme. This is a simplification when compared with the majority of the techniques which have recently been proposed in the literature and some of which were reviewed in Chapter 2. Recall that most of them relied on an output or state error equation which also in some way involved parameter error terms. Adaptive laws for parameter estimation were then derived based on Lyapunov function stability considerations.

When feedback linearization is used, the ideal control law completely cancels the effect of some terms in the plant response. The control law will have a similar effect on the plant model imbedded in the parameter estimator. In view of the cancellation properties of feedback linearization, it is questionable whether or not the unknown parameters can be accurately estimated using the standard parallel connection of plant and identifier as shown in Figure 3.2.

In the interest of simplicity we will initially investigate this question using a linear plant since there is no loss in generality of the result. The control law will be calculated by the same method as used to determine a feedback linearization control law.

Consider the linear SISO n'th order plant

$$\dot{x} = Ax + bu \tag{3.3.1a}$$

$$y = h(x) = c^T x \tag{3.3.1b}$$

which has relative order one. The control law is found from

$$\dot{y} = c^T Ax + c^T bu = V \tag{3.3.2}$$

by solving for u,

$$u = \frac{1}{c^T b}\left(-c^T Ax + V\right) \tag{3.3.3}$$

where it is assumed that $c^T b \neq 0.$

Using (3.3.3) in (3.3.1a) results in the closed loop system

$$\dot{x} = \left[I - \frac{bc^T}{c^T b}\right] Ax + \frac{b}{c^T b}V \tag{3.3.4a}$$

$$y = c^T x. \tag{3.3.4b}$$

Since the control law (3.3.3) yields the transfer function for (3.3.4a,b) as (see Chapter 2)

$$G(s) = \frac{1}{s} \tag{3.3.5}$$

it follows that *n-1* states of (3.3.1a) have become unobservable.

To better illustrate this point we will consider a specific example. Suppose the plant is represented by

$$G(s) = \frac{s+b}{s^2 + a_1 s + a_2} \tag{3.3.6}$$

and that it is driven by a control law that is derived by the same method that was used to find (3.3.3). There are a myriad of state space realizations possible for representing (3.3.6). Kailath (1980) defines four different canonical forms. However, only two of these, the controller and observer forms, have an *A* matrix and *b* or *c* vectors in which the transfer function numerator and denominator coefficients appear explicitly. Obviously it is important to have these parameters appear explicitly from an estimation standpoint. Of these two representations, only the observer form has a *c* vector which defines the plant output in terms of a single state variable. If the output is a function of more than one state variable then it is not clear from the state space equations or block diagram that the plant is equivalent to a string of integrators between input and output. However, this can be shown to be the case by deriving the transfer function from whatever state-space representation was chosen. For the present purpose the observer form is the most convenient. The observer form representation for G(s) in (3.3.6) is

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -a_1 & 1 \\ -a_2 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 \\ b \end{bmatrix} u \tag{3.3.7a}$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} x \doteq h(x). \tag{3.3.7b}$$

Following the same procedure as for calculating the linearizing input for a nonlinear plant we find

$$\dot{y} = \frac{\partial h}{\partial x}\dot{x} = -a_1 x_1 + x_2 + u. \tag{3.3.8}$$

Choosing a new system input V and setting it equal to (3.3.8) the plant input is

$$u = V + a_1 x_1 - x_2. \tag{3.3.9}$$

Substituting this u in (3.3.7a) the state-space representation is

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ -(a_2 - ba_1) & -b \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 \\ b \end{bmatrix} V \tag{3.3.10a}$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} x. \tag{3.3.10b}$$

Figure 3.3 is the block diagram representation of equations (3.3.10a,b). Equation (3.3.6) represents a plant having relative order one and consequently control law (3.3.9) should make $n\text{-}r=1$ states unobservable where r is the relative order. Inspection of Figure 3.3 shows that x_2 is the unobservable state and further that the I/O relationship is a pure integration as expected. To verify the observability condition we can check the rank of the observability matrix,

$$W_0 = \begin{bmatrix} c \\ cA \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}. \tag{3.3.11}$$

Since $\rho(W_0) = 1 < 2$, the plant is not completely observable. The observable states can be found from an equivalency transformation, $\bar{X} = PX$ (Chen (1984), Zhou et.al. (1996)). P is a nonsingular constant matrix. The first $n-n_1$ columns of P are chosen from the first $n-n_1$ independent columns of the observability matrix. (n_1 is the rank of the observability matrix) The last n_1 columns are arbitrary. For this example, P can be taken to be the identity matrix. Consequently the observable state is

$$y = cP^{-1}\overline{X} = \overline{x}_1 = x_1 \tag{3.3.12}$$

which agrees with Figure 3.3.

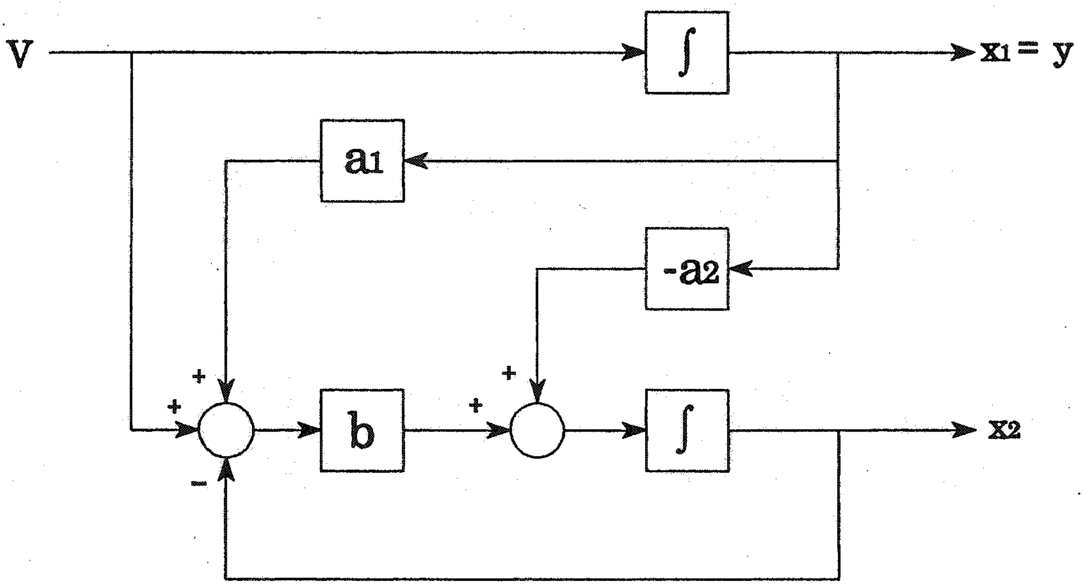

Figure 3.3 Block diagram representation of equations (3.3.10a,b) showing unobservable state, x_2.

The above discussion assumes that the true values of states and parameters are available for use in the control law. In the next section we will investigate on-line estimation of unknown control law parameters.

3.4 Gradient Calculation for Feedback Linearized Plants

The gradient in the RPEM algorithm is defined by equation (3.2.18) for other than linearizing control laws. When the plant is driven from a linearizing control law, the plant output becomes a function of the control law parameters (i.e. estimated plant parameters) and therefore contributes to the gradient. To calculate the gradient in this case

$$\psi_t = -\left[\frac{d\varepsilon_t}{d\hat{\theta}}\right]^T = \left[\frac{d\hat{y}}{d\hat{\theta}} - \frac{dy}{d\hat{\theta}}\right]^T \doteq \psi_m - \psi_p\,. \tag{3.4.1}$$

Equation (3.4.1) illustrates the potential problem with the gradient. It may vanish when the estimated parameters reach their true values. If this happens, the covariance matrix will increase without bound and eventually cause a failure of the estimation algorithm.

If the gradient vanishes the covariance update from (3.2.26d) is

$$P_t = P_{t-1}\left(\frac{1}{\lambda}\right) \tag{3.4.2}$$

where λ is the forgetting factor. Recursively updating the covariance with (3.4.2) generates a geometric progression with common ratio $1/\lambda$. Since λ is typically less than unity, the updated covariance increases with each succeeding iteration.

3.4.1 Parameter Estimation Using Standard System Architecture

The control law for the plant represented by (3.3.7a) is given by (3.3.9). Let us assume that parameter a_1 is unknown and attempt to estimate it using the configuration shown in Figure 3.4. This architecture is basically the same as that shown in Figure 3.2. Since the plant is linear, and has relative order one, the coordinate transformation is simply $y_1 = x_1$. The loop has been closed through a feedback gain, d, to insure stability if parameter $\hat{a}_1$ converges and the plant responds as a pure integrator (the time constant is $1/d$ for the linearized system). The Kalman gains in a deterministic system have only a transient effect since the prediction error vanishes if the parameter estimate converges. Therefore, in the interest of simplicity, we do not use an innovations model in the estimation algorithm in the following.

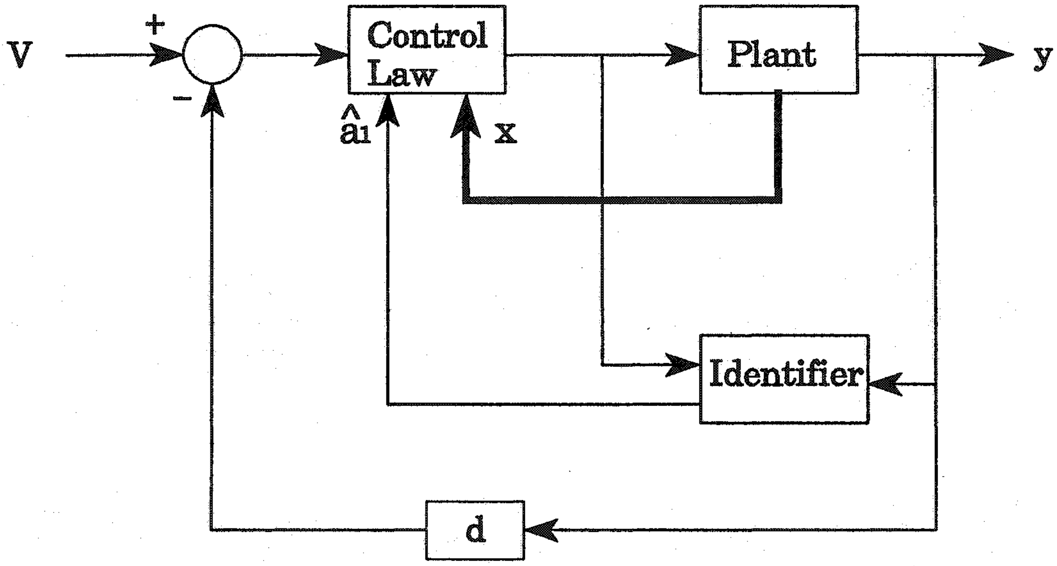

Figure 3.4 Standard parameter estimation configuration with a linearizing control law.

With reference to (3.3.7a), the plant parameters were chosen as $a_1 = 4,\ a_2 = 3,\ b = 2$ and initial conditions were set at zero. The feedback gain was set at unity and the reference input to the system was a step of magnitude 5. The parameter estimate $\hat{a}_1(0)$ was chosen as 2.0.

As pointed out in section 3.2.1, it may not be possible to estimate parameters in a closed loop. Before proceeding with a linearizing control, we will try estimating the parameter with the control law replaced by a proportional only controller with a gain of one. This will demonstrate for the case at hand that the parameter can be estimated under closed loop conditions when the linearizing control law is not present. Figure 3.5 shows that under these conditions the closed loop does not affect the parameter estimate.

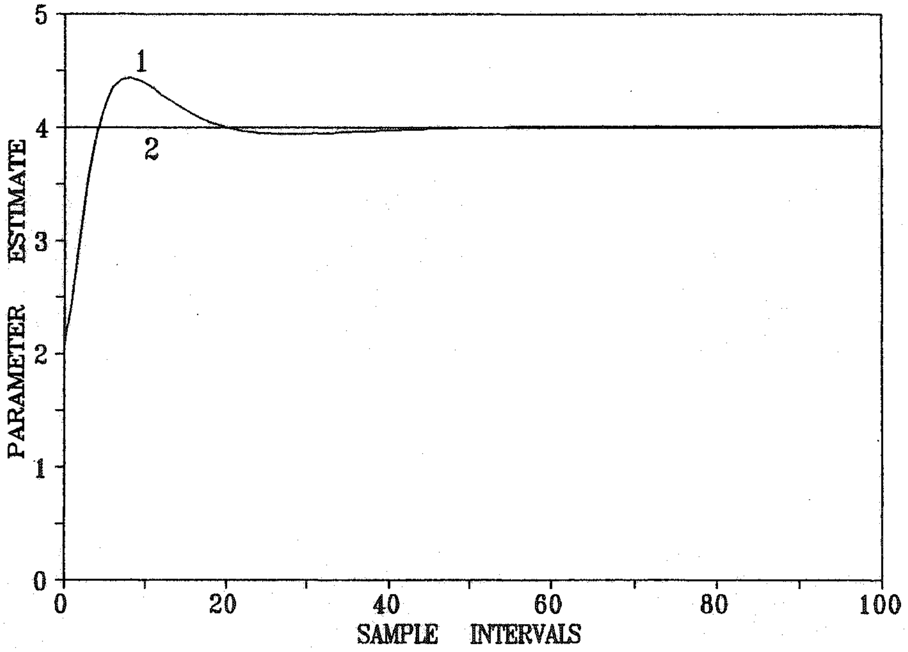

Figure 3.5 Closed loop parameter estimate with proportional only controller. Curve 1, parameter estimate, curve 2, parameter true value.

The following results were obtained with a linearizing control law in place of the proportional only controller. (i.e. architecture as shown in Figure 3.4) The feedback gain d was set at one.

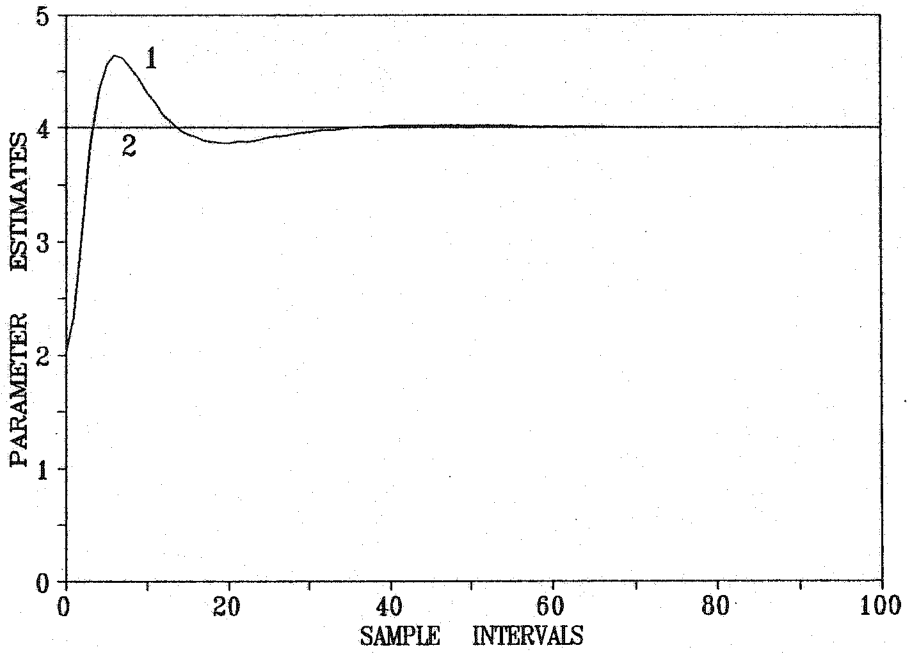

Figure 3.6 Parameter estimate in a closed loop with a linearizing control law. Curve 1, parameter estimate, curve 2, parameter true value.

Figure 3.6 shows that the parameter estimate converges to the true value. Figure 3.7 is a plot of the gradients. This plot shows that the plant gradient, ψ_p, and plant model gradient, ψ_m, equalize after $\hat{a}_1$ converges to a_1. Consequently the algorithm gradient, ψ_t, (cf. (3.4.1)) vanishes.

Figure 3.8 is a plot of the covariance update denominator repeated here for reference, (cf. (3.2.26d))

$$\lambda + \psi_t^T P \psi_t .$$

The forgetting factor was set at 0.97. This plot shows that the covariance update denominator reduces in value to just the forgetting factor. As stated above, this causes the covariance to blow up. Figure 3.9 shows the increase in the covariance. Figure 3.10 depicts the tracking between the plant simulation states and the plant model states. The parameter convergence is fast enough so that there is no detectible difference between these curves.

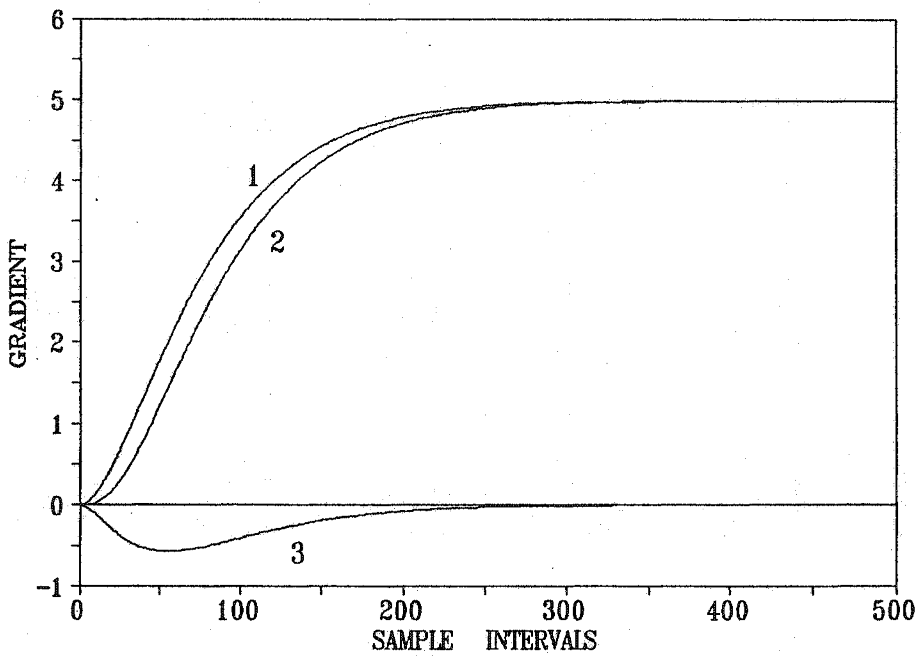

Figure 3.7 Plant and model gradients using standard system architecture. Curve 1, plant gradient, curve 2, plant model gradient, curve 3, algorithm gradient.

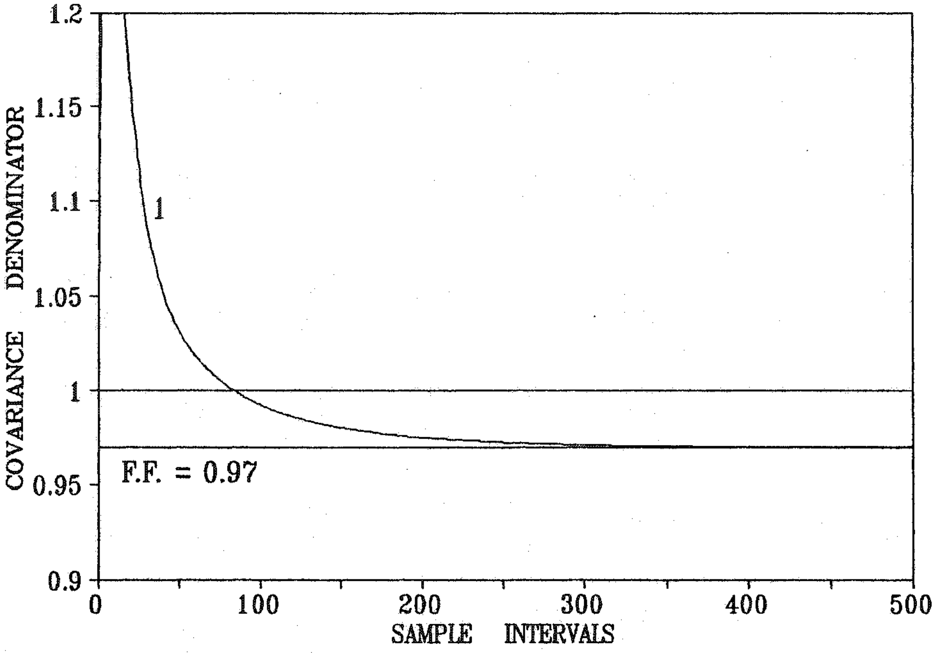

Figure 3.8 Curve 1, covariance update denominator.

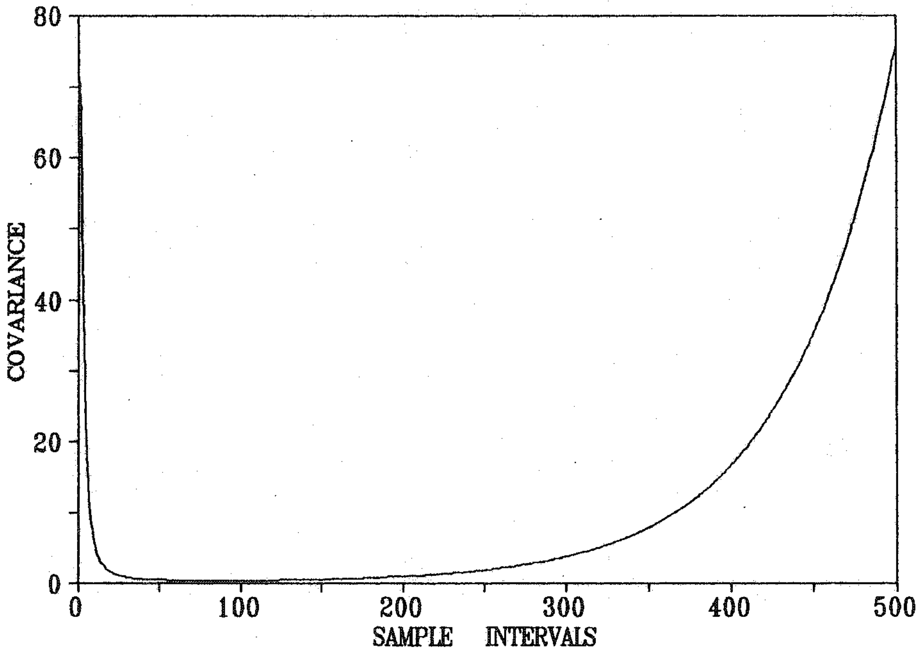

Figure 3.9 Plot of square root of covariance, P.

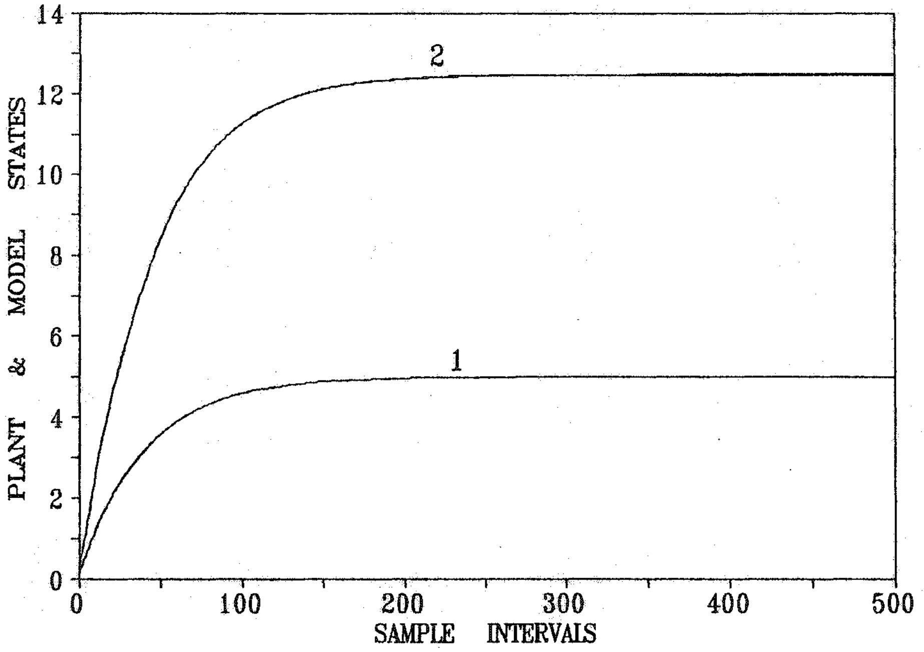

Figure 3.10 Plant and model state tracking. Curve 1, plant output, curve 2, unobservable state.

Even though the plant output is a function of the estimated parameter, we can choose to ignore this and try to estimate the parameter. The gradient in this case is

$$\psi_t = \left[\frac{d\hat{y}}{d\hat{\theta}}\right]^T \doteq \psi_m . \tag{3.4.3}$$

Since the algorithm gradient now depends only on the model gradient, the cancellation which was present in the above test will be eliminated.

All test conditions remain the same as previously stated. Figure 3.11 is a plot of the parameter estimate. The estimate converges even though the gradient, ψ_t, is theoretically in error.

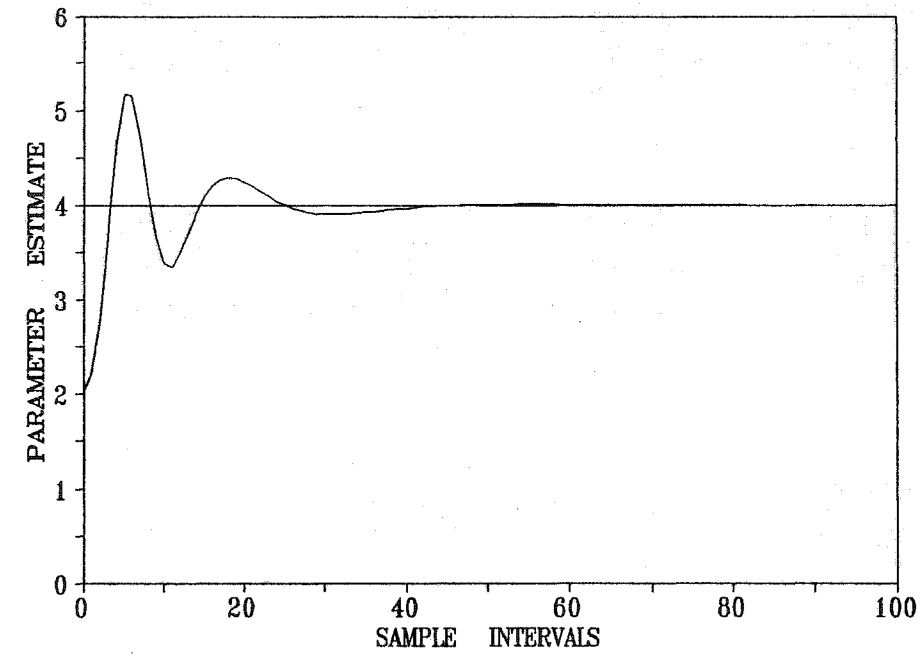

Figure 3.11 Parameter estimate using only model gradient.

Figure 3.12 compares the error in the parameter estimate when the true gradient (3.4.1) and only the model gradient (3.4.3) is used in the estimation algorithm. This figure indicates that substantially better transient performance is obtained when using the true gradient.

Figures 3.13 and 3.14 can be compared with Figures 3.8 and 3.9 respectively. In the present case, with model gradient (3.4.3), the covariance update denominator converges to unity and the covariance remains bounded.

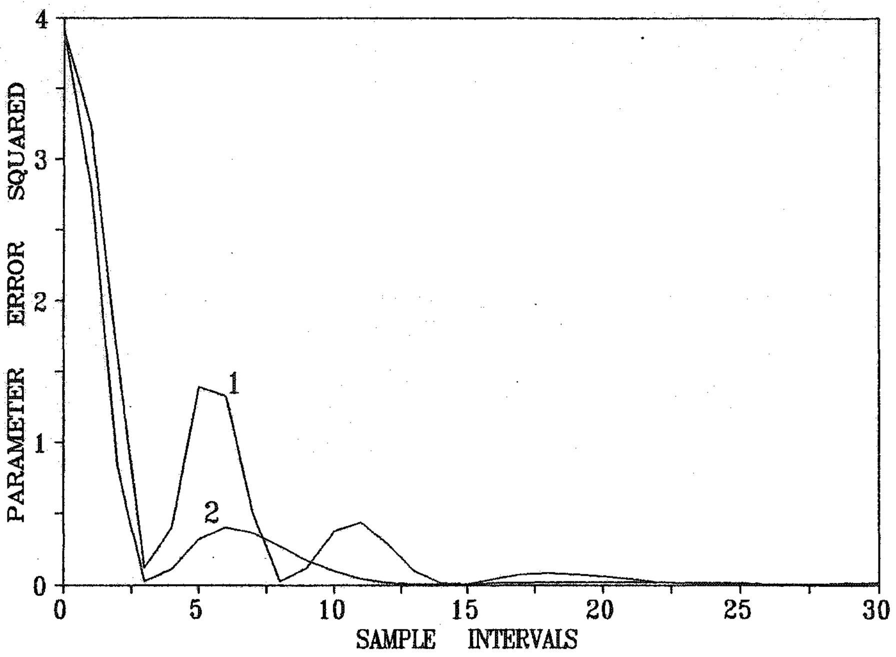

Figure 3.12 Parameter estimate error comparison, curve 1, only plant model gradient, curve 2, true gradient.

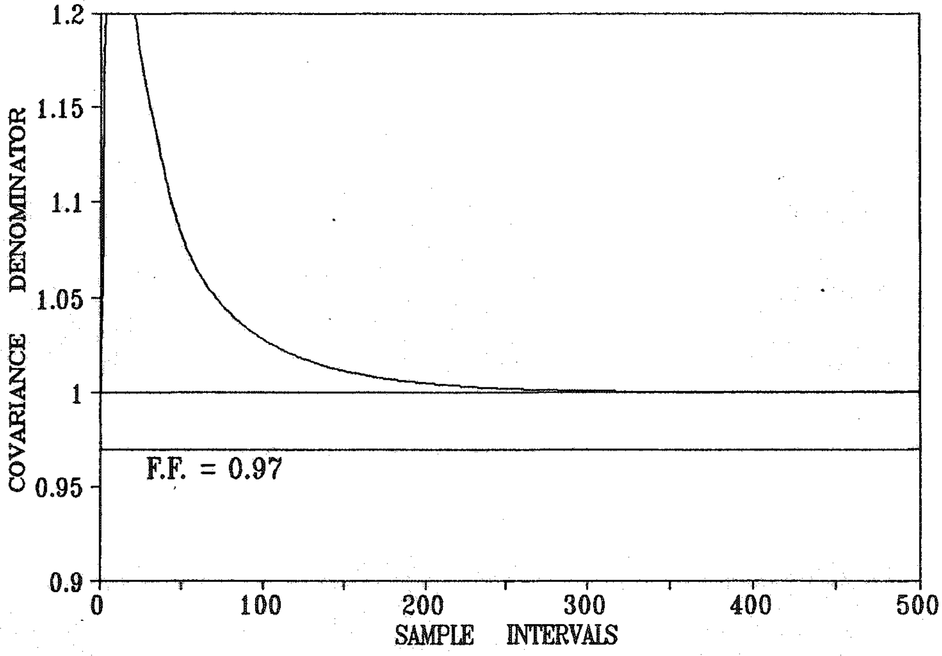

Figure 3.13 Covariance update denominator, gradient dependent only on plant model.

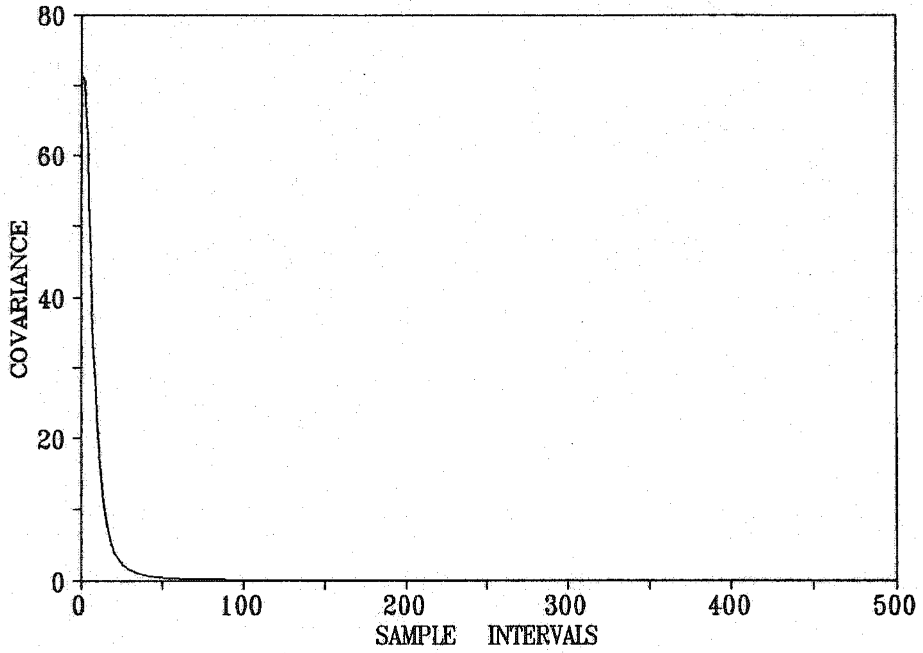

Figure 3.14 Plot of square root of covariance when only plant model gradient is used.

So far we have assumed that the states are measureable and considered only the effects that the gradient has on the parameter estimate and algorithm performance. To catagorize these tests we will refer to the test using the plant model gradient (3.4.3) as case I and the test using the true gradient (3.4.1) as case II.

The RPEM algorithm is a bootstrap estimator. In cases where some states may not be measureable, difficult to accurately measure or noisy, the estimated state values generated by the algorithm could be used. Estimated states in the control law will result in gradients which are different from those used in case I and case II. Case III will parallel case I but with the gradient rederived with estimated

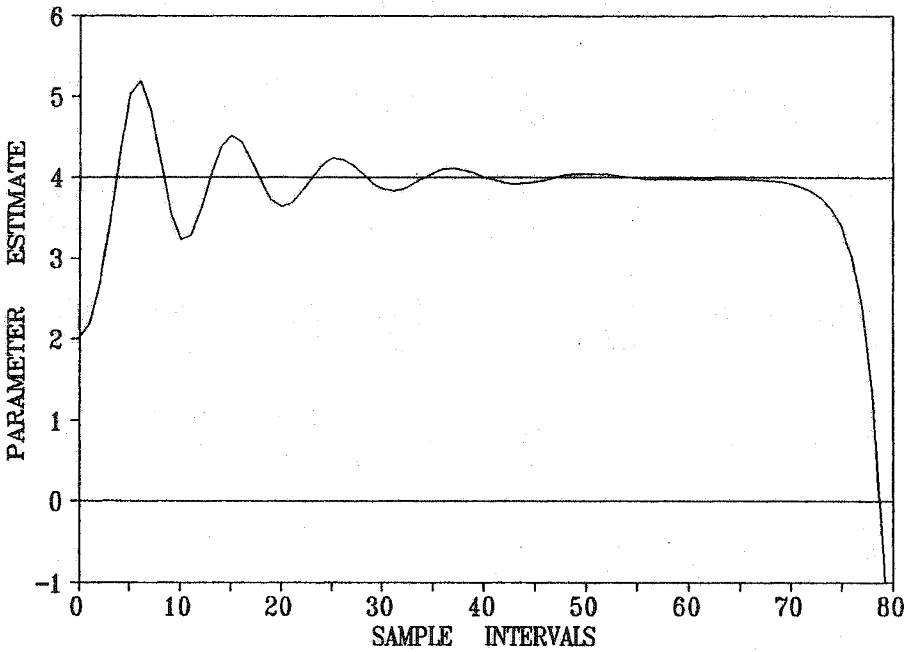

Figure 3.15 Parameter estimate with estimated states in the control law.

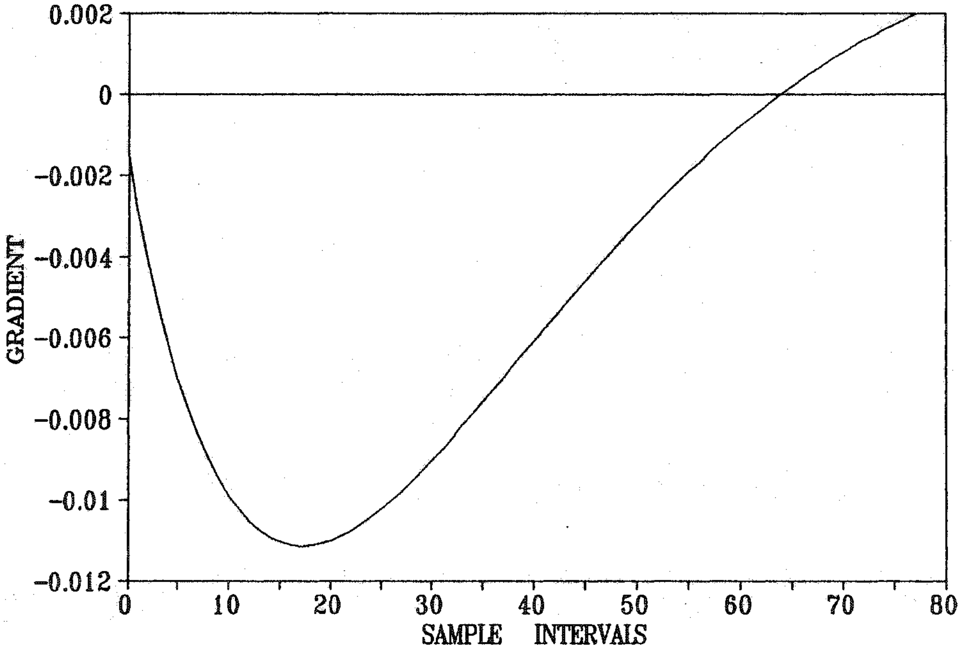

Figure 3.16 Gradient with estimated states in the control law.

states in the control law. Case IV will utilize the true gradient derived as it was in case II but again with estimated states.

Figures 3.15 and 3.16 are plots of the parameter estimate and gradient respectively for case III. Investigation of the plant simulation and algorithm plant model with estimated states in the control law shows why the performance has deteriorated.

Using estimated parameters and states in the control law, the plant input is

$$u = \hat{a}_1\hat{x}_1 - \hat{x}_2 - d\hat{x}_1 + V, \tag{3.4.4}$$

and the state-space representation (3.3.7a) changes to

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -a_1x_1 + (\hat{a}_1 - d)\hat{x}_1 + (x_2 - \hat{x}_2) \\ -a_2x_1 + b(\hat{a}_1 - d)\hat{x}_1 - b\hat{x}_2 \end{bmatrix} + \begin{bmatrix} 1 \\ b \end{bmatrix} V. \tag{3.4.5}$$

Only the a_1 estimate appears in (3.4.5) since it is the only parameter in the control law in this case.

The plant model in the estimation algorithm is the same as that of the real plant which is given by (3.3.7a,b). However, in the plant model, some parameters are estimates and all of the states are estimated values, i.e.

$$\frac{d}{dt}\begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix} = \begin{bmatrix} -\hat{a}_1 & 1 \\ -a_2 & 0 \end{bmatrix}\begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix} + \begin{bmatrix} 1 \\ b \end{bmatrix} u \tag{3.4.6a}$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix}\hat{x}. \tag{3.4.6b}$$

Parameters a_2 and b are assumed known. The same control law, containing estimated parameters,is applied to both plant and estimator. (Refer to Figure 3.2) For the plant simulation the model is then given by (3.4.5). However, since the estimator is determining the control law parameters, the estimator model becomes from (3.4.6a)

$$\frac{d}{dt}\begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix} = \begin{bmatrix} -\hat{a}_1 & 1 \\ -a_2 & 0 \end{bmatrix}\begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix} + \begin{bmatrix} \hat{a}_1 - d & -1 \\ b(\hat{a}_1 - d) & -b \end{bmatrix}\begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix} + \begin{bmatrix} 1 \\ b \end{bmatrix} V$$

$$= \begin{bmatrix} -d & 0 \\ [b(\hat{a}_1 - d) - a_2] & -b \end{bmatrix}\begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix} + \begin{bmatrix} 1 \\ b \end{bmatrix} V. \tag{3.4.7}$$

Note that the parameter a_1 needed for the control law does not appear explicitly in the estimator model because the term in which it originally appeared has been cancelled by the control law. Therefore variations in this parameter are unobservable in the estimator model output, $\hat{x}_1$. Since the gradient for the case we are investigating is calculated based on $\hat{x}_1$ only, it has no meaning. The control law is by design intended to cancel or modify certain terms in the plant. Therefore it follows that in general the gradient will not be valid for the conditions we are presently considering.

For a period of time in Figure 3.16, the gradient has the correct sign but not the correct magnitude. Figure 3.17 is a plot of the parameter estimate when the gradient is held at a constant negative value. This plot shows that as long as the gradient has the correct sign and is small in magnitude the parameter estimate converges but estimate quality is poorer under these conditions. As would be expected, the estimate diverges if the sign of the gradient is reversed. This test explains why the parameter converged for a time in Figure 3.15 with the gradient shown in Figure 3.16.

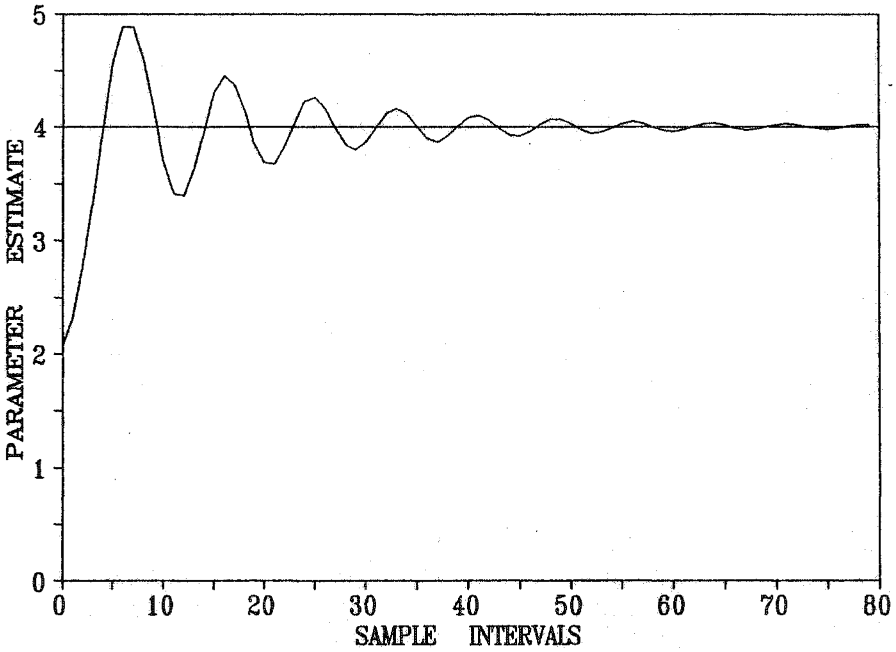

Figure 3.17 Parameter estimate with a constant gradient of -0.005.

Inspection of (3.4.7) shows that the $\hat{a}_1$ parameter influences $\hat{x}_2$. For estimating purposes, x_2 can be considered as the plant output and the gradient calculated accordingly from

$$\psi_t = \frac{d\hat{x}_2}{d\hat{a}_1} \doteq \psi_m . \qquad (3.4.8)$$

Of course the noisy or inaccurately measured value for x_2 must be used in calculating the prediction error. (cf. (3.2.26a))

Figure 3.18 is a plot of the $\hat{a}_1$ parameter estimate under deterministic conditions and an accurate measurement of x_2. By comparison with Figure 3.6, the time required to settle within 1% of the true value is approximately ten times as long. Figure 3.19 compares the squared error in the parameter estimate between Figures 3.6 and 3.18.

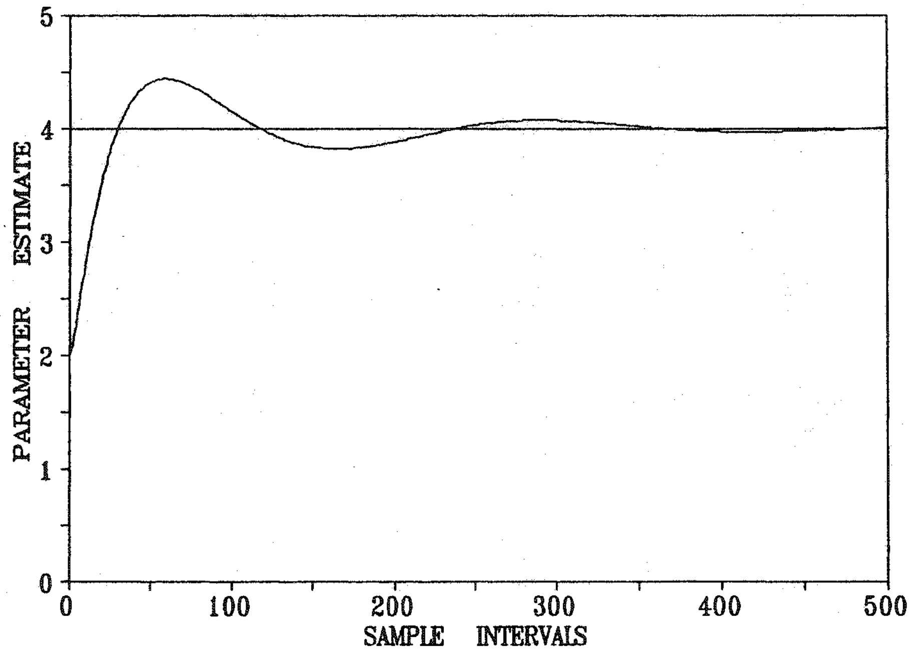

Figure 3.18 Parameter estimate when x_2 is taken as plant output.

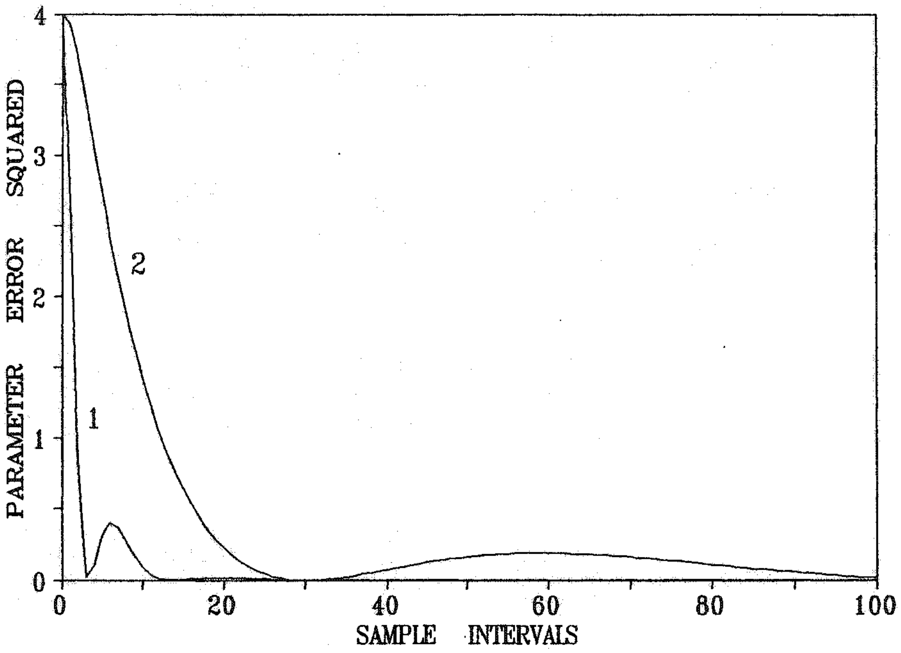

Figure 3.19 Parameter estimate error comparison, curve 1 when using x_1 as output, curve 2 when using x_2 as output.

Aside from the larger error and longer settling time, the estimation algorithm performs normally. The covariance reduces to a small value and the covariance denominator converges to unity.

Case IV calculates the gradient according to (3.4.1) and uses estimated states in the control law. The parameter estimate is nearly identical to that in case II. Compare Figure 3.20 with Figure 3.6.

As was mentioned previously, when estimated parameters are used, the gradient calculated from the estimation model is nearly zero and has no meaning. Therefore the algorithm gradient is equivalent to just the plant gradient. The plant gradient in this case converges to zero and, as a result, the covariance denominator reduces in value to just the forgetting factor. This causes the covariance to

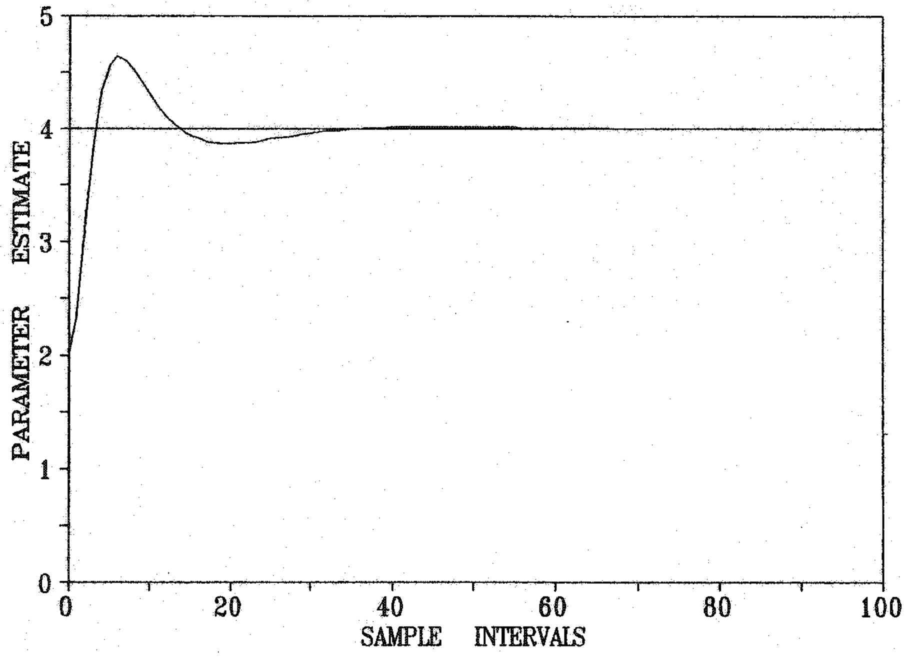

Figure 3.20 Parameter estimate for case IV.

increase identically as shown in Figure 3.9.

3.5 Modified System Architecture For Adaptive Feedback Linearization

In the last section it was shown that the system architecture which is used for adaptive control of linear plants can malfunction when used for adaptive linearization. The primary problem is with the gradient which can vanish under certain conditions. This happens because the true gradient is the difference between the plant gradient and the estimation algorithm plant model gradient (cf. (3.4.1)). These two gradients become equal as parameters converge. This difficulty can be avoided if instead of attempting to estimate the parameters of the nonlinear plant, the parameters of the linearized system are estimated. In order

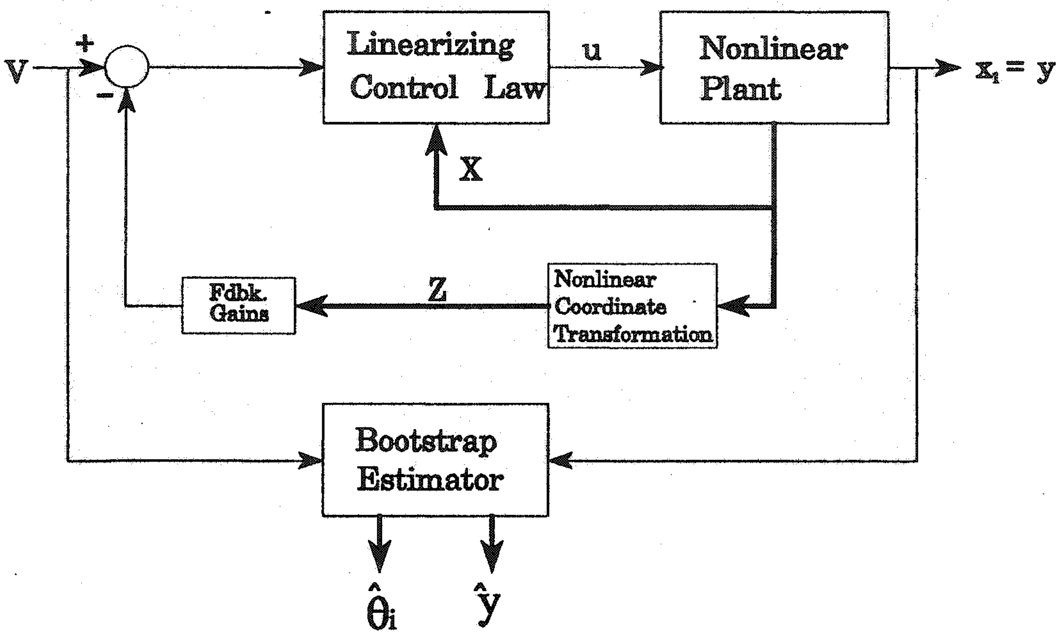

Figure 3.21 Modified system architecture.

to do this the estimator is driven from the system input and plant output signals. Between these points the system model will have a linear structure. Knowing this we choose an appropriate linear model for the estimator. The structure for this model will be described in detail in the next chapter. The modified system architecture is shown in Figure 3.21.

The parameters that have to be estimated for linearization are still those of the nonlinear plant. Therefore the nonlinear plant parameters have to be estimated by some means from the linearized plant parameter estimates, $\hat{\theta}$. We can deduce *a-priori* from the linear plant model what the converged value of the parameters $\hat{\theta}$ will be when perfect linearization of the nonlinear plant has been achieved. This is the key observation. As outlined in the introduction to this chapter, by using this *a-priori* knowledge it is possible to develop an adaptive law for estimating the nonlinear plant parameters from estimates of the linearized model parameters. This technique will be developed in the next chapter.

3.6 Summary and Conclusions

When a linearizing control law is used, the estimation algorithm gradient is the difference between the plant gradient and the plant model gradient which is part of the estimation algorithm. (cf.(3.4.1)) As time evolves these two gradients become equal causing the algorithm gradient to vanish and the covariance to increase without bound.

Several different conditions were investigated using the classical system architecture for parameter estimation in adaptive control loops. The tests can be broadly catagorized as either using the true gradient or only the plant model gradient in the estimation algorithm. Using only the plant model gradient is an approximation and will be referred to in what follows as the approximate gradient. The tests were further catagorized by the use of either measured plant states or estimated states in the control law with each of the gradients. The results obtained are summarized as follows.

A. Measured plant states

1. Case I: approximate gradient.

   The parameter estimate converges to the true value but with a larger error and slower convergence rate than when the true gradient is used. The covariance remains bounded. See Figures 3.11, 3.12, and 3.14.

2. Case II: true gradient.

   The parameter estimate converges to the true value but the covariance blows up. See Figures 3.6 and 3.9.

B. Estimated plant states

3. Case III: approximate gradient.

   In this case the unknown parameter is unobservable via the plant output and the estimate never converges. See Figures 3.15 and 3.16.

4. Case IV: true gradient.

   The parameter estimate converges but the covariance blows up. See Figure 3.20 for a plot of the parameter estimate. The covariance response is identical to that shown in Figure 3.9.

Case I is the only one of the four cases tested which produced somewhat satisfactory but poorer results as compared with the parameter estimate shown in Figure 3.6. However these results are based on an approximate gradient.

The unknown parameter may influence a state which is unobservable. However if this state is controllable and can be measured it can be used for parameter estimating purposes. Figure 3.18 is a plot of the parameter estimate when x_2 was used rather than x_1 since the unknown parameter did not affect x_1

in this test. Again the results are not as good as when x_1 is used. Refer to Figure 3.19 for a comparison of the errors when x_1 and x_2 are used in estimating the parameter.

In section 3.5 an improved method for estimating parameters in feedback linearized systems is suggested. The new method is based on a modified system architecture which connects the estimator between the system input and plant output. The true gradient between these points can be based solely on the estimator plant model and therefore the cancellation which was present with the true gradient in the classical architecture is eliminated.

The parameters which are estimated in this scheme are those of the linear system model. It remains to obtain estimates of the nonlinear plant parameters from estimates of the linearized system parameters since it is the nonlinear plant parameters that are needed in the control law. A method for obtaining these parameters with the new configuration is developed in Chapter 4.

CHAPTER 4

Nonlinear Plant Parameter Estimation

4.1 Introduction

In the last chapter a new architecture for a self-tuning feedback linearization system was proposed. While this configuration solves the identifiability problem found with the standard configuration used for linear plant self-tuning systems, the nonlinear plant parameters cannot be estimated directly from it. What can be estimated directly are the parameters of the model that represents the linear system between input V and output y. These parameters are a composite of the true value of the unknown nonlinear plant parameters and the current estimate of the unknown parameters. The parameter estimator block shown in Figure 3.21 provides these composite estimates.

If exact plant states and parameters were used in the control law then the I/O model can be represented by a string of cascaded integrators. However when control law parameters are inaccurate this model is modified.

This chapter has two objectives. One is to develop the structure for the I/O model when imperfect linearization exists due to parameter errors in the control law. The other is to develop an adaptive law which will provide estimates of the nonlinear plant parameters from the composite parameter estimates of the quasi-linearized plant. The estimated nonlinear plant parameters supplied by the adaptive law are used in a certainty equivalence (i.e. as if they are the true parameter values) manner in the control law and nonlinear coordinate transformation. The complete self-tuning system architecture is shown in Figure 4.1.

The chapter concludes with simulations demonstrating the new self-tuning feedback linearization strategy. The linear plant used for demonstration purposes in Chapter 3 is revisited using the new architecture and control of a nonlinear plant with the proposed self-tuning method is also demonstrated.

4.2 Adaptive Law Development

Assume for the moment that an adaptive law exists which provides estimates of the nonlinear plant parameters from estimates of the linearized plant parameters, $\hat{\theta}_i$, in Figure 3.21. Using the nonlinear plant parameter estimates in the control law, linearization would then proceed asymptotically as the estimates of the nonlinear plant parameters converge towards their true values. During this period of time parameters estimated for the linear plant will be in error since the I/O relationship of the nonlinear plant is, to a varying degree, still nonlinear due to incomplete cancellation of some nonlinear terms. The estimated parameters in the partially linearized model actually represents the error between the true nonlinear plant parameter and its estimated value. If it can be established *a-priori* how the true parameter and its estimate combine to produce the linearized plant parameter, this information can be used to reduce the error in the nonlinearized plant parameter estimate. Several different parameter combinations are possible. One form is given in the introduction to Chapter 3 by equation (3.1.1) and additional forms will be derived in the following sections. The adaptive law will be designed to interpret the error information contained in the linearized plant model parameter estimates and use this information to refine the estimates of the nonlinear plant parameters.

In general the parameters appearing in the linear plant model will be functions of the true nonlinear plant parameters and their estimates. To investigate the nature of this relationship, consider the single input nonlinear plant

$$\dot{x} = f(x,p) + g(x,p)u \qquad (4.2.1)$$

where p represents a vector of unknown parameters and the plant output is

$$y = x_1. \qquad (4.2.2)$$

The linearized plant model is assumed to be in Brunovsky canonical form. States in this model are defined by the nonlinear transformation vector

$$z = T(x,p) = \left[T_1, \langle dT_1\,, f(x,p)\rangle, \dots, \langle dT_{n-1}, f(x,p)\rangle\right]^T. \qquad (4.2.3)$$

Taking the time derivative of (4.2.3) yields

$$\begin{aligned}
\dot{z}_1 &= \langle dT_1, f(x,p)\rangle = z_2 \\
\dot{z}_2 &= \langle dT_2, f(x,p)\rangle = z_3 \\
&\vdots \\
\dot{z}_n &= \langle dT_n, f(x,p)\rangle + \langle dT_n, g(x,p)\rangle u(x,p)\,.
\end{aligned} \tag{4.2.4}$$

From equation (2.4.27) the certainty equivalence linearizing control law is

$$u(x,\hat{p}) = \frac{V - \langle dT_n, f(x,\hat{p})\rangle}{\langle dT_n, g(x,\hat{p})\rangle} \tag{4.2.5}$$

where $\hat{p}$ is the estimate of parameter vector p. Substituting control law $u(x,\hat{p})$ for $u(x,p)$ in (4.2.4) yields

$$\dot{z}_n = \langle dT_n, f(x,p)\rangle - \langle dT_n, f(x,\hat{p})\rangle \frac{\langle dT_n, g(x,p)\rangle}{\langle dT_n, g(x,\hat{p})\rangle} + \frac{\langle dT_n, g(x,p)\rangle}{\langle dT_n, g(x,\hat{p})\rangle} V. \tag{4.2.6}$$

It is obvious from (4.2.6) that when $\hat{p}=p$

$$\dot{z}_n = V \tag{4.2.7}$$

as expected. The first two terms in (4.2.6) can be viewed as additional inputs to the linear plant model until parameter estimates converge and these terms cancel.

4.2.1 Analysis of Equation (4.2.6)

Several variations of (4.2.6) are possible depending on the structure of $f(x,p)$, $g(x,p)$, and the location of unknown parameters in the elements of these vectors. Some of these variations will now be investigated by considering a specific plant. The objective is to find an adaptive law, ideally the same for all cases, which can be used to estimate the unknown control law parameters.

Given the SISO relative order 2 linearizable plant

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} a\sin(x_2) \\ -bx_1^2 \end{bmatrix} + \begin{bmatrix} 0 \\ c \end{bmatrix} u \tag{4.2.8a}$$

$$y = x_1 \tag{4.2.8b}$$

it can be shown that the nonlinear coordinate transformation is

$$\begin{aligned} z_1 &= x_1 \doteq T_1 \\ z_2 &= a\sin(x_2) \doteq T_2 \end{aligned} \tag{4.2.9}$$

and that

$$\begin{aligned} \dot{z}_1 &= z_2 \\ \dot{z}_2 &= \left(a\cos(x_2)\right)\left(-bx_1^2 + cu\right). \end{aligned} \tag{4.2.10}$$

The plant (4.2.8a) has three parameters, *a, b,* and *c,* which could be unknown. We will calculate the control law and then assume that certain of these parameters are unknown either individually or in combinations. These unknown parameters will be replaced by estimates wherever they appear in the control law. This will give some insight into how the theoretical linearized plant model is modified by an approximate control law. It is this modified model that has to be used in the parameter estimation algorithm. States of the nonlinear plant are assumed to be measurable.

Case I: a is the only unknown parameter (cf. (4.2.5))

$$\langle dT_2, f(x,p)\rangle = [0,\ a\cos(x_2)]\begin{bmatrix} a\sin(x_2) \\ -bx_1^2 \end{bmatrix} \tag{4.2.11}$$

$$= -bx_1^2(a\cos(x_2))$$

$$\langle dT_2, g(x,p)\rangle = [0\ ,\ a\cos(x_2)]\begin{bmatrix} 0 \\ c \end{bmatrix} = c(a\cos(x_2)) \tag{4.2.12}$$

Substituting (4.2.11) and (4.2.12) into (4.2.5) the control law is

$$u(x,\hat{p}) = \frac{V + bx_1^2(\hat{a}\cos(x_2))}{c(\hat{a}\cos(x_2))}. \tag{4.2.13}$$

Using this approximate control law in (4.2.10) yields

$$\dot{z}_2 = -bx_1^2(a\cos(x_2)) + bx_1^2(\hat{a}\cos(x_2))\left[\frac{c(a\cos(x_2))}{c(\hat{a}\cos(x_2))}\right]$$

$$+\frac{ca\cos(x_2)}{c\hat{a}\cos(x_2)}V = \frac{a}{\hat{a}}V.$$

(4.2.14)

Therefore the state space representation for the linearized plant with unknown parameter a is

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \begin{bmatrix} 0 \\ a/\hat{a} \end{bmatrix} V. \tag{4.2.15}$$

In this case error in the nonlinear plant parameter estimates modifies the coefficient of the arbitrary input, V. This coefficient in the Brunovsky canonical form should always be unity.

Case II: c is the only unknown parameter.

Following the same steps as in Case I,

$$\langle dT_2, f(x,p)\rangle = -bx_1^2\left(a\cos(x_2)\right) \tag{4.2.16}$$

$$\langle dT_2, g(x,p)\rangle = c\left(a\cos(x_2)\right) \tag{4.2.17}$$

$$u(x,\hat{p}) = \frac{V + bx_1^2\left(a\cos(x_2)\right)}{\hat{c}a\cos(x_2)} \; . \tag{4.2.18}$$

From equation (4.2.10) this control law yields

$$\dot{z}_2 = bx_1^2\left(a\cos(x_2)\right)\left[\frac{c}{\hat{c}} - 1\right] + \frac{c}{\hat{c}}V. \tag{4.2.19}$$

Replacing $\dot{z}_2$ in (4.2.10) with (4.2.19) the linearized plant model is

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \left(c/\hat{c} - 1\right) \end{bmatrix} abx_1^2\cos(x_2) + \begin{bmatrix} 0 \\ c/\hat{c} \end{bmatrix} V. \tag{4.2.20}$$

Equation (4.2.20) illustrates the earlier observation that the first two terms in (4.2.6) can be viewed as generating additional inputs to the linear plant model until parameter estimates converge. In this case the additional input derives from the component

$$\begin{bmatrix} 0 \\ \left(\frac{c}{\hat{c}} - 1\right) \end{bmatrix} abx_1^2\cos(x_2)$$

and obviously its effect vanishes when $\hat{c}=c$. In more complex nonlinear plants several inputs of this type may be present initially but all will eventually reduce to zero assuming parameter convergence.

Case III: a and b are unknown parameters

Again following the procedure in the previous cases,

$$\langle dT_2, f(x,p)\rangle = -bx_1^2\left(a\cos(x_2)\right) \qquad (4.2.21)$$

$$\langle dT_2, g(x,p)\rangle = ca\cos(x_2) \qquad (4.2.22)$$

$$u(x,\hat{p}) = \frac{V + \hat{b}x_1^2\left(\hat{a}\cos(x_2)\right)}{c\hat{a}\cos(x_2)} \qquad (4.2.23)$$

$$\dot{z}_2 = \left(\hat{b} - b\right)x_1^2\left(a\cos(x_2)\right) + \frac{a}{\hat{a}}V \qquad (4.2.24)$$

The linearized plant model in this case is

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} - \begin{bmatrix} 0 \\ b-\hat{b} \end{bmatrix} ax_1^2\cos(x_2) + \begin{bmatrix} 0 \\ a/\hat{a} \end{bmatrix} V. \qquad (4.2.25)$$

Note that the unknown parameter a appears explicitly in the first term of (4.2.24). In order to implement the estimator model, (4.2.25), this parameter has to be replaced by its estimate.

The basic form for the estimator models for the three cases considered are given by equations (4.2.15), (4.2.20), and (4.2.25). Inspection shows that these models converge to the Brunovsky canonical form as estimated parameters approach their true values. These models will be reparameterized and state feedback added before using them in the RPEM algorithm. This aspect will be discussed in greater detail when cases I and III are used to illustrate the development of the adaptive law.

The characteristic of interest in all three cases is that errors in the nonlinear plant parameter estimates produce terms in the linearized plant model which involve both the true value of the unknown parameter and its estimate. The appearance of the true values of the nonlinear plant parameters in the partially linearized plant model offers the possibility of identifying these parameters from this model. A key element in what follows in the next section is that the true values, at convergence, of these plant parameter and estimated parameter combinations can be deduced in advance.

4.2.2 Estimating the Nonlinear Plant Parameters

So far we have shown how parameters of the nonlinear plant enter into the linearized plant model, via the control law, for three specific cases. In this section we will develop a methodology for estimating these parameters. The state space model (4.2.25) will be used to illustrate development of the adaptive law.

The two terms in (4.2.25) containing the unknown parameter estimates also involve the unknown parameters explicitly and so a direct estimation is impossible. However, the linearized model can be reparameterized such that each of the original combinations of unknown parameter and its estimate is represented by a single new parameter. The estimation routine is then set up to estimate these new parameters. Since the true values of the new parameters are known in advance from knowledge of the linearized plant structure, an error equation can be written and from it the adaptive law deduced.

To illustrate this procedure let β and δ be the new parameters defined as

$$\hat{\beta}(k) \doteq b - \hat{b}(k-1) \tag{4.2.26}$$

and

$$\hat{\delta}(k) \doteq \frac{a}{\hat{a}(k-1)}\,. \tag{4.2.27}$$

To define a metric for the parameter error in the reparameterized model we will adopt the convention that:

[1]parameter error = true value - estimated value

where true value is defined as the value of the linearized plant parameters after the estimates of the nonlinear plant parameters have converged. Considering (4.2.26),

[1] In the subsequent generalization of the adaptive law given by equation (4.2.36a), the expressions for the estimated quantities are manipulated into an equivalent form such that the true value is always unity.

the true value of $\hat{\beta}(k)$ is obviously zero. Therefore, according to the above definition, the error in $\hat{\beta}(k)$ is

$$e_{\hat{\beta}}(k) = 0 - \hat{\beta}(k) = \hat{b}(k-1) - b\,. \tag{4.2.28}$$

Inspection of (4.2.28) shows that only the sign of $e_{\hat{\beta}}(k)$ is needed to determine whether the value of the previous estimate should be increased or decreased to reduce the error. Table 4.1 tabulates the correction to be applied to the estimate based on the sign of the error.

Table 4.1 $\hat{b}(k-1)$ adjustment.

sgn[e_β(k)]	[2]correction applied to $\hat{b}(k-1)$
+	decrease
-	increase

This suggests a sign algorithm for adjustment of this parameter (Johnson (1988)).

A typical recursive sign algorithm is:

new estimate = (old estimate) + (adaptation gain) x (change direction)

x (function of prediction error).

Implementing this yields the adaptation law,

[2] This table assumes that b is known to be a positive number and that estimate $\hat{b}(k)$ is constrained to be positive.

$$\hat{b}(k) = \hat{b}(k-1) - \mu_b sgn\left(e_{\hat{\beta}}(k)\right) f\left(e_{\hat{\beta}}(k)\right). \tag{4.2.29}$$

In the adaptation law there is a choice for the error function, $f\left(e_{\hat{\beta}}(k)\right)$. Usually algorithms of this type use $e^2(k)$ (e.g. LMS, (least mean squares)). However, squaring the error eliminates the sign information so that it would be necessary to keep the $sgn\left(e_{\hat{\beta}}(k)\right)$ term in the algorithm and evaluate it on each iteration. If instead the $|e(k)|$ is chosen then

$$sgn\left(e_{\hat{\beta}}(k)\right)\left|e_{\hat{\beta}}(k)\right| = e_{\hat{\beta}}(k)$$

and so no special tracking of the error sign is needed. In addition to this, the $|e(k)|$ has a larger value than $e^2(k)$ for $|e(k)| < 1$ and will tend to eliminate small errors faster. Using $\left|e_{\hat{\beta}}(k)\right|$ for the function of the prediction error, $f\left(e_{\hat{\beta}}(k)\right)$, the adaptation law becomes

$$\hat{b}(k) = \hat{b}(k-1) - \mu_b e_{\hat{\beta}}(k) . \tag{4.2.30}$$

The sign of the correction term in (4.2.29) is determined from Table 4.1. For example, if $sgn\left(e_{\hat{\beta}}(k)\right)$ is positive then the sign of the correction term should be negative in order to decrease the previous estimate. If the $sgn\left(e_{\hat{\beta}}(k)\right)$ is negative then a positive correction is needed so again the sign of the correction term has to be negative.

In self-tuning applications the parameter error signal depends on other estimated quantities that contain errors. Excessively large parameter errors generate large corrections that tend to be over-corrections and can lead to instability. On the other hand, very small corrections can lead to long convergence times. Therefore a modified adaptive gain term is used to hedge against the effect of a large error that is not representative of the true error. Analogous to the normalization technique of adaptive control we make the adaptation gain an inverse function of the error magnitude. That is set

$$\mu_b(k) = \frac{\mu_b(0)}{|e_{\hat{\beta}}(k)| + \alpha}. \tag{4.2.31}$$

When the adaptive gain is defined in this way, large errors will reduce the gain and the likelihood of over correction while small errors will allow the gain to increase with a positive effect on convergence time. Parameter α determines the level at which the error magnitude begins to have increased or diminished influence in determining the gain. It is also needed to prevent the denominator from becoming zero if the error vanishes. Once α is chosen, $\mu_b(0)$ can be chosen depending on the maximum desired gain. The effect these parameters have on plant performance is demonstrated in an application example given in Chapter 5.

For the other unknown parameter, $\hat{\delta}(k)$, the converged value is one. Therefore following the error convention which we have established,

$$e_{\hat{\delta}}(k) = 1 - \hat{\delta}(k-1). \tag{4.2.32}$$

As before, it is sufficient to know only the sign of $e_{\hat{\delta}}(k)$ in order to determine the direction of the correction to be made in $\hat{a}(k-1)$. Table 4.2 is a tabulation for this parameter similar to Table 4.1.

Table 4.2 $\hat{a}(k-1)$ adjustment.

$\mathrm{sgn}[e_{\hat{\delta}}(k)]$	$\hat{\delta}(k-1)$	correction applied to $\hat{a}(k-1)$
+	<1	decrease
-	>1	increase

Footnote 2 with b and $\hat{b}(k)$ replaced by a and $\hat{a}(k)$ respectively applies to Table 4.2 also. Following the same procedure as above leads to the adaptive law

$$\hat{a}(k) = \hat{a}(k-1) - \mu_a e_{\hat{\delta}}(k) \tag{4.2.33}$$

and

$$\mu_a(k) = \frac{\mu_a(0)}{|e_{\hat{\delta}}(k)| + \rho} . \tag{4.2.34}$$

Along with a recursive estimation algorithm of the type discussed in Chapter 3, section 3.2.2, adaptive laws (4.2.30) and (4.2.33) provide the means to implement adaptive feedback linearization for plant (4.2.8a,b) utilizing the new system architecture shown in Figure 3.21.

It is also interesting to note that the true nonlinear plant parameters can be found even though the estimated values may appear in nonlinear combinations with the true values. In this example we have the nonlinear combination

$$\hat{\delta} = \frac{a}{\hat{a}} \tag{4.2.35}$$

where a is the parameter of interest.

Figure 4.1 shows the architecture for a self-tuning feedback linearizing system. This system is derived from Figure 3.21 by adding the adaptive law. Referring to Figure 4.1 and the plant used in the above development of the adaptive law, operation of the self-tuning feature is as follows. First the parameters β and δ appearing in the linearized model are estimated by the RPEM. Then the nonlinear plant parameters a and b are estimated from $\hat{\beta}$ and $\hat{\delta}$ via the adaptive law.

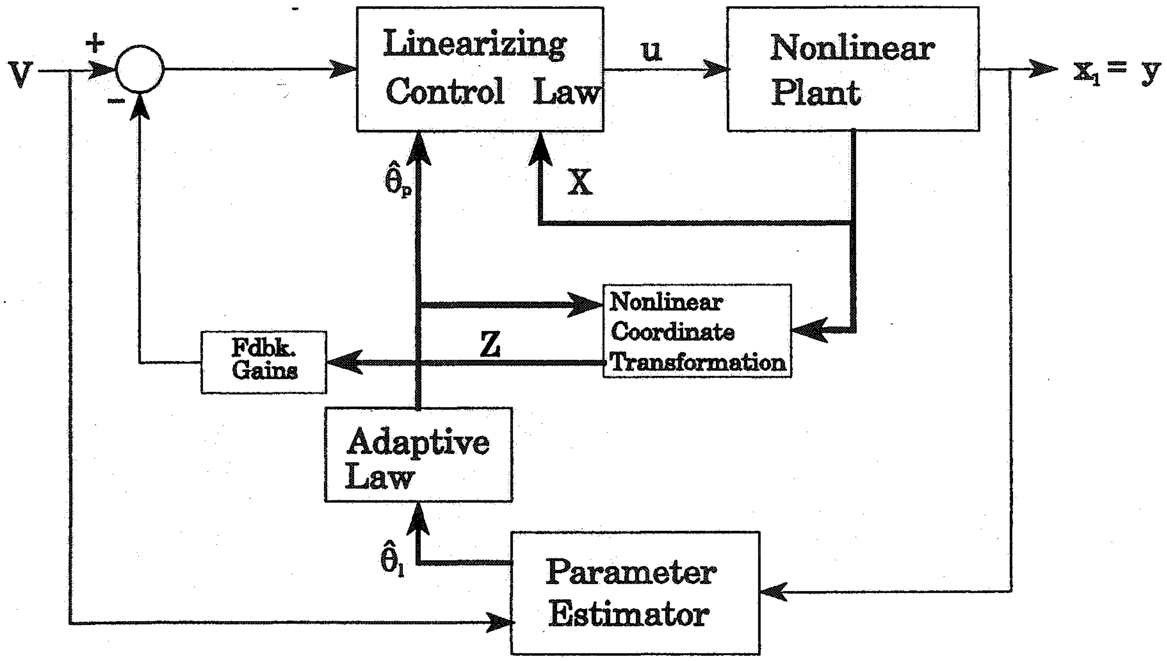

Figure 4.1 System architecture for self-tuning feedback linearization.

The adaptive law just developed was based on the plant model given by (4.2.8a,b). However (4.2.6) is completely general and will be used to show that the adaptive law developed above applies in a more general case. The following assumption will be made.

Assumption 4.2.1 Parameter estimates always have the same sign as the true parameter.

Assumption 4.2.1 is needed in interpreting the parameter error equation in the following analysis. If the estimate does not have the same sign as the true parameter then the adaptive law will be unstable. It is not a very restrictive requirement since logic to ensure this condition can always be included in the estimation algorithm. The following conditions are sufficient for adaptive law (4.2.36a) to be applicable for calculating the nonlinear plant parameters.

Given an n'th order feedback linearizable plant

$$\dot{x} = f(x,p) + g(x,p)u$$

with the following conditions:

1. *the relative order is n*
2. *elements* $g_i(x,p) = 0, \quad i = 1, \dots, n-1,$ *and* $g_n(x,p)$ *is a monomial*
3. *the vector* dT_n *does not contain unknowns which cannot be estimated from other terms in* $\dot{z}_n$
4. *unknown control law parameters appear in* $\dot{z}_n$ *only as coefficients*
5. *the parameter error is defined as*

$$e_i(k) = 1 - \frac{p_i}{\hat{p}_i(k-1)}$$

where p_i *is the true value of the unknown parameter*

then the adaptive law,

$$\hat{p}_i(k) = \hat{p}_i(k-1) - sgn(p_i)\mu_{\hat{p}_i}e_i(k)\,, \tag{4.2.36a}$$

where $\hat{p}_i(k)$ is the estimate of the i'th unknown nonlinear plant parameter and $\mu_{\hat{p}_i}$ is a gain term, will provide estimates of the nonlinear plant parameters in the self-tuning configuration of Figure 4.1.

The general form for the adaptive law was developed in this section for a specific case. It will now be shown that it applies to the broader class of plants described by the above conditions.

In the Brunovsky canonical form the only state equation affected by the control law and hence the unknown parameters is $\dot{z}_n$. (refer to (4.2.4,5,6)) Equation (4.2.6), repeated below, represents $\dot{z}_n$ after the approximate control law has been substituted for u. This equation therefore will contain coefficients which are functions of both the true plant parameters and estimated plant parameters. The starting point is the examination of the terms in (4.2.6) to establish the relationship between the unknown parameters and their estimates.

$$\dot{z}_n = \langle dT_n, f(x,p)\rangle - \langle dT_n, f(x,\hat{p})\rangle \frac{\langle dT_n, g(x,p)\rangle}{\langle dT_n, g(x,\hat{p})\rangle} + \frac{\langle dT_n, g(x,p)\rangle}{\langle dT_n, g(x,\hat{p})\rangle} V. \tag{4.2.6}$$

Case I: If $g_n(x,p)$ contains the only unknown parameter, i.e.

$$g(x,p) = \begin{bmatrix} 0 \\ \vdots \\ \hat{c}x_i \end{bmatrix} \tag{4.2.36}$$

then from (4.2.6)

$$\dot{z}_n = \left[1 - \frac{c}{\hat{c}}\right] \langle dT_n, f(x,p)\rangle + \frac{c}{\hat{c}}V$$
$$= \left[1 - \frac{c}{\hat{c}}\right] \alpha(x,p) + \frac{c}{\hat{c}}V. \qquad (4.2.37)$$

where $\alpha(x,p)$ is an arbitrary function of the plant states and known plant parameters.

Case II: If $f(x,p)$ contains the only unknown parameters, then

$$\dot{z}_n = \langle dT_n, f(x,p)\rangle - \langle dT_n, f(x,\hat{p})\rangle + V$$
$$= \frac{\partial T_n}{\partial x_1}\left[f_1(x,p) - f_1(x,\hat{p})\right] + \cdots + \frac{\partial T_n}{\partial x_n}\left[f_n(x,p) - f_n(x,\hat{p})\right] + V$$
$$= \sum_{i=1}^{n} \frac{\partial T_n}{\partial x_i}(f_i - \hat{f}_i) + V \qquad (4.2.38)$$

According to condition 4, the f_i and $\hat{f}_i$ terms are identical except for the coefficients. One is the true parameter value and the other is the estimated value. Therefore the $f_i - \hat{f}_i$ term in (4.2.38) can be factored as

$$f_i - \hat{f}_i = (p_1 - \hat{p}_1)_i\alpha_{i1}(x) + \cdots + (p_j - \hat{p}_j)_i\alpha_{ij}(x)\,. \qquad (4.2.39)$$

Within the constraint of condition 3, the vector dT_n may contain unknown parameters which will appear in the $\alpha_{iq}(x)$ terms. We have seen an example of this in the last section. Referring to (4.2.9),

$$dT_2 = \left[0,\ a\cos(x_2)\right]. \tag{4.2.40}$$

From (4.2.25)

$$\alpha_{11}(x) = a x_1^2 \cos(x_2) \tag{4.2.41}$$

where a is an unknown parameter. However it can be replaced by $\hat{a}$ obtained from the coefficient of V in this same equation (i.e. (4.2.25)) and so does not present a problem. Also refer to the simulation results for (4.3.17) which demonstrates this case (see Figure 4.10a).

<u>Case III:</u> If $f(x,p)$ and $g_n(x,p)$ both contain unknown parameters, then

$$\begin{aligned} \dot{z}_n &= \langle dT_n, f(x,p)\rangle - \frac{c}{\hat{c}}\langle dT_n, f(x,\hat{p})\rangle + \frac{c}{\hat{c}}V \\ &= \langle dT_n, f(x,p) - \frac{c}{\hat{c}}f(x,\hat{p})\rangle + \frac{c}{\hat{c}}V \end{aligned} \tag{4.2.42}$$

$$f_i - \frac{c}{\hat{c}}\hat{f}_i = \left(p_1 - \frac{c}{\hat{c}}\hat{p}_1\right)_i \alpha_{i1}(x) + \cdots + \left(p_j - \frac{c}{\hat{c}}\hat{p}_j\right)_i \alpha_{ij}(x). \tag{4.2.43}$$

The comments concerning the factorization of $f_i - \hat{f}_i$ and unknown parameters in dT_n for case II apply here as well to $f_i - \frac{c}{\hat{c}}\hat{f}_i$ and $\alpha_{iq}(k)$.

We now show that the parameter error equation given in condition 5 can be used for all three of the above cases. To simplify the notation the following definitions will be used. In the reparameterized linear model, let $\hat{\theta}$ denote the estimated parameters arising from the first two terms in (4.2.6), i.e.

$$\left\langle dT_n, f(x,p)\right\rangle - \left\langle dT_n, f(x,\hat{p})\right\rangle \frac{\left\langle dT_n, g(x,p)\right\rangle}{\left\langle dT_n, g(x,\hat{p})\right\rangle} . \tag{4.2.44}$$

Cases II and III above result from parameter estimate errors in these terms.

Let $\hat{\delta}$ denote the estimate arising from the last term in (4.2.6),

$$\frac{\left\langle dT_n, g(x,p)\right\rangle}{\left\langle dT_n, g(x,\hat{p})\right\rangle} . \tag{4.2.45}$$

Case I and Case III contain parameter errors in this term. Then, for example, rewriting (4.2.38) in terms of these definitions we would have

$$\dot{z}_n = \hat{\theta}_1\alpha_1(x) + \cdots + \hat{\theta}_v\alpha_v(x) + \hat{\delta}V \tag{4.2.46}$$

where

$$\hat{\theta}_j = p_j - \hat{p}_j$$

and

$$\hat{\delta} = \frac{c}{\hat{c}} .$$

The influence of the parameter error equation (see condition 5) on the adaptive law is two-fold. First of all it affects the magnitude of the correction to be made in the parameter estimate. Secondly it determines the direction in which the correction is to be made. (i.e. should the previous estimate be increased or decreased in value). This last effect depends on the sign of the error. However the sign information has to be interpreted based on whether the unknown parameter is positive or negative. For example if we were to compute the error for a negative parameter whose estimate is larger in magnitude than the true value we would find that the sign of the error is positive. But so is the sign of the error for a positive parameter whose estimate is larger than the true value. However the sign of the correction term needed to decrease the estimated value is not the same for both the positive and negative parameters. Obviously the correction must be positive to decrease the value if a negative parameter is being estimated but negative to decrease the value if a positive parameter is being estimated. This error sign ambiguity is resolved by including $sgn(p_i)$ as a multiplying factor in the correction term.

Consider the situation discussed above and the adaptive law given by equation (4.2.36a).

For $p_i > 0$ and $\hat{p}_i(k-1) > p_i$,

$$\begin{aligned}\hat{p}_i(k) &= \hat{p}_i(k-1) - sgn(p_i)\mu_{\hat{p}_i}e_i(k) \\ &= \hat{p}_i(k-1) - \mu_{\hat{p}_i}e_i(k)\end{aligned} \tag{4.2.47}$$

For $p_i < 0$ and $|\hat{p}_i(k-1)| > |p_i|$,

$$\begin{aligned}\hat{p}_i(k) &= -\hat{p}_i(k-1) - sgn(p_i)\mu_{\hat{p}_i}e_i(k) \\ &= -\hat{p}_i(k-1) + \mu_{\hat{p}_i}e_i(k)\,.\end{aligned} \tag{4.2.48}$$

Equations (4.2.47) and (4.2.48) demonstrate that the $sgn(p_i)$ factor ensures the correct sign for the correction term.

To complete the analysis we have to verify that the negative sign in front of the correction term is correct for both positive and negative parameter estimates which can be either larger or smaller than their true values. This has to be done for the parameter combinations which occur in Cases I through III discussed above. (cf. (4.2.37,39,43)). Also, to make the adaptive law universal, error equations of the same form as given in condition 5 have to be derived for all three cases of the parameter combinations.

In the following we define p and $\hat{p}$ to be the generic representation of the true and estimated values respectively of the nonlinear plant parameters. Using the error definition from condition 5 where the iteration index has been dropped to simplify the equations we have:

For case I:

$$\frac{p}{\hat{p}} \approx \hat{\delta} \tag{4.2.49}$$

$$\therefore \quad e \approx 1 - \hat{\delta} \tag{4.2.50}$$

For Case II:

$$p_i > 0: \quad \hat{\theta}_i \approx p_i - \hat{p}_i \tag{4.2.51}$$

$$\frac{p_i}{\hat{p}_i} \approx 1 + \frac{\hat{\theta}_i}{\hat{p}_i} \tag{4.2.52}$$

$$e_i \approx 1 - \left[1 + \frac{\hat{\theta}_i}{\hat{p}_i}\right] = -\frac{\hat{\theta}_i}{\hat{p}_i} \tag{4.2.53}$$

$$p_i < 0: \quad \hat{\theta}_i \approx \hat{p}_i - p_i \tag{4.2.54}$$

$$\frac{p_i}{\hat{p}_i} \approx 1 - \frac{\hat{\theta}_i}{\hat{p}_i} \tag{4.2.55}$$

$$e_i \approx 1 - \left[1 - \frac{\hat{\theta}_i}{\hat{p}_i}\right] = \frac{\hat{\theta}_i}{\hat{p}_i} \tag{4.2.56}$$

<u>For Case III:</u>

$$p_i > 0: \quad \hat{\theta}_i \approx p_i - \hat{\delta}\hat{p}_i \tag{4.2.57}$$

$$\frac{p_i}{\hat{p}_i} \approx \hat{\delta} + \frac{\hat{\theta}_i}{\hat{p}_i} \tag{4.2.58}$$

$$e_i = 1 - \left[\hat{\delta} + \frac{\hat{\theta}_i}{\hat{p}_i}\right] \tag{4.2.59}$$

$$p_i < 0: \quad \hat{\theta}_i \approx \hat{\delta}\hat{p}_i - p_i \tag{4.2.60}$$

$$\frac{p_i}{\hat{p}_i} \approx \hat{\delta} - \frac{\hat{\theta}_i}{\hat{p}_i} \tag{4.2.61}$$

$$e_i = 1 - \left[\hat{\delta} - \frac{\hat{\theta}_i}{\hat{p}_i}\right] \tag{4.2.62}$$

As pointed out earlier, the sign of the error conveys information about the relative magnitude of the parameter estimate as compared with the true value. To show this let

$$\hat{p}_i = \rho p_i \tag{4.2.63}$$

where $0 < \rho < \rho_{\max}$ and $\rho_{\max} > 1$. From (4.2.63)

$$\frac{p_i}{\hat{p}_i} = \frac{1}{\rho} . \tag{4.2.64}$$

Substituting $\frac{1}{\rho}$ for the parameter ratio in (4.2.49,52,55,58,61) shows that the error equations are all equivalent to

$$e_i = 1 - \frac{1}{\rho} . \tag{4.2.65}$$

Using (4.2.65), Table 4.3 is a tabulation of the sign errors produced by estimates which are either too large or too small for both positive and negative parameters. The last column lists the sign of the correction term needed to reduce the error in the estimate.

Table 4.3 correction term sign versus error sign

parameter	ρ	$sgn(e_i)$	correction term sign
$p_i > 0$	>1	+	-
	<1	-	+
$p_i < 0$	>1	+	+
	<1	-	-

The table shows that for $p_i > 0$ the sign of the correction term should be the negative of the sign of the error. For $p_i < 0$ the correction term sign should be the same as the sign of the error. Hence the adaptive law with a negative sign for the correction term and *sgn(p$_i$)* included as a factor to ensure compatibility with negative parameter estimates is universal.

Equation (4.2.65) shows algebraically that the sign of the error is determined only by $\frac{p_i}{\hat{p}_i}$. However this is not obvious from (4.2.59,62) for Case III. The dependence of e$_i$ on $\hat{\delta}$ obscures this fact. Figure 4.2 which is a plot of (4.2.59) graphically demonstrates that the *sgn(e$_i$)* is independent of $\hat{\delta}$.

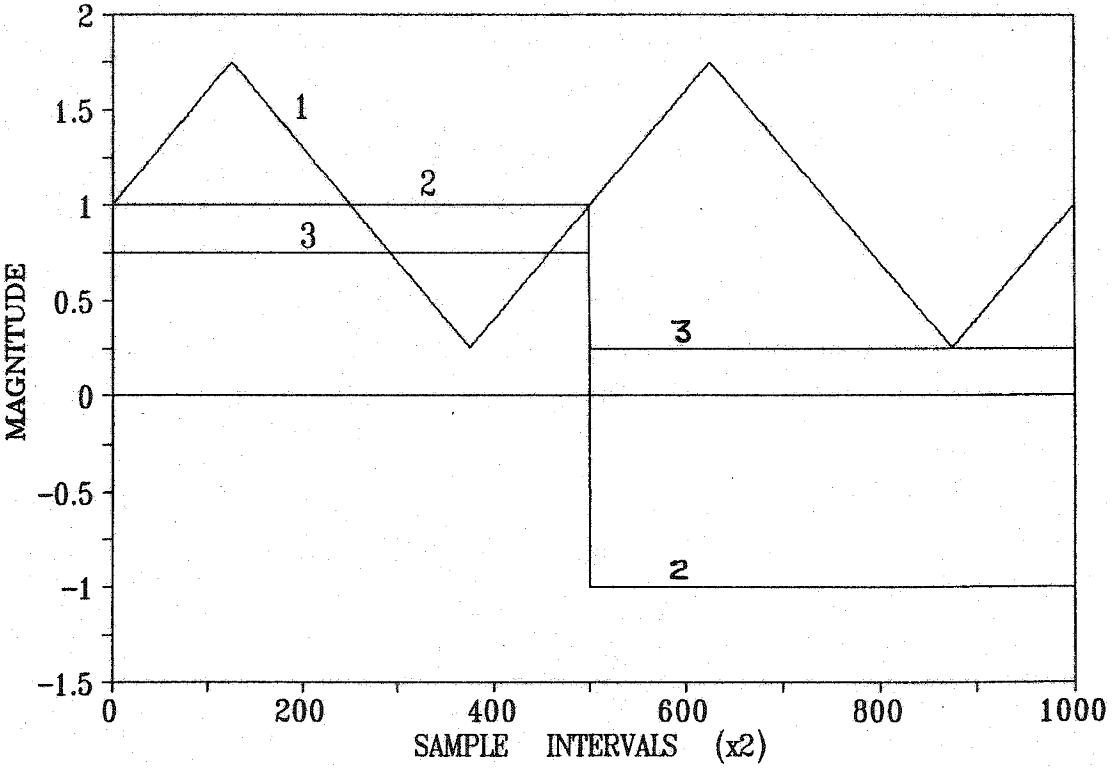

Figure 4.2 Effect of $\hat{\delta}$ variation on *sgn(e$_i$)*. curve 1, $\hat{\delta}$; curve 2, *sgn(e$_i$)*; curve 3, $\hat{p}_i$; $p_i = 0.5$.

The estimate $\hat{p}_i$ is initially set to a value greater than the true parameter. As long as $\hat{p}_i$ is held at its initial value, the $sgn(e_i)$ doesn't change even though $\hat{\delta}$ is varying through a range of values around its true value. At $k = 1000$, $\hat{p}_i$ is set to a value which is less than p_i and the $sgn(e_i)$ changes from plus to minus as expected. Again $\hat{\delta}$ is varied through the same range of values as before with no effect on the $sgn(e_i)$. The sign independence of the error from $\hat{\delta}$ can also be shown analytically by substituting for $\hat{\theta}$ in (4.2.59) and (4.2.62).

Remark 1. Note that the error expressions are approximations rather than equalities. The fact that the true parameter error is not accurately known on each iteration is compensated for by making the adaptive gain an inverse function of the error magnitude. This results in an adaptive law that performs cautiously.

Remark 2. The adaptive law given by (4.2.36a) makes adjustments in the parameter estimate by integral action. Integral action alone can produce a slow response. By adding proportional gain, the estimate from the adaptive law will respond faster. Proportional action can be added to this algorithm as follows.

$$I_i(k) = I_i(k-1) - sgn(p_i)\mu_{p_i}e_i(k)$$

$$\hat{p}_i(k) = I_i(k) - sgn(p_i)\rho_{p_i}e_i(k)$$

where $I_i(k)$ is the original adaptive law and $\hat{p}_i(k)$ is the new adaptive law having both integral and proportional response. The most expedient way to adjust the integral and proportional gains is to run a simulation and determine a suitable range of values for each. Final tuning is then done on the real process.

4.2.3 Adaptive Law Stability and Convergence

In the last section it was shown that estimates of the nonlinear plant parameters can be calculated by using the adaptive law (4.2.36a) . In this section the stability and convergence of the algorithm will be investigated.

The adaptive law can be written as

$$\hat{p}(k) = \hat{p}(k-1) - sgn(p)\,\mu \left[1 - \frac{p}{\hat{p}(k-1)} \right] . \tag{4.2.66}$$

The equilibrium point at convergence is the value of the unknown parameter rather than the origin. The unknown parameter value will be shifted to the origin and stability investigated using Lyapunov theory. In the shifted coordinate system

$$\alpha(k) = \hat{p}(k) - p \tag{4.2.67}$$

where α is the new coordinate, p is the true parameter value and $\hat{p}(k)$ is the parameter estimate. From (4.2.67)

$$\hat{p}(k) = \alpha(k) + p \tag{4.2.68}$$

and

$$\hat{p}(k-1) = \alpha(k-1) + p. \tag{4.2.69}$$

Substituting equation (4.2.68) and (4.2.69) into (4.2.66) and rearranging terms yields

$$\alpha(k) = \alpha(k-1) - \mu \left[1 - \frac{p}{\alpha(k-1) + p} \right] \tag{4.2.70}$$

for a positive unknown parameter, p.

From equation (4.2.70), $\alpha(k)=0$ when $\alpha(k-1)$ is either 0 or $\mu - p$.

Then equation (4.2.68) reduces to

$$\hat{p}(k) = p\,.$$

From (4.2.69) and (4.2.66) it can be shown that

$$\hat{p}(k-1) = p\,.$$

Consequently if the origin is a stable equilibrium in the $\alpha(k)$ plane then the adaptive law (4.2.66) is stable.

We will choose $\alpha^2(k)$ for the Lyapunov function candidate. Then

$$V[\alpha(k)] = \alpha^2(k) \tag{4.2.71}$$

and

$$V[\alpha(k-1)] = \alpha^2(k-1)\,. \tag{4.2.72}$$

The incremental change can be shown to be

$$\Delta V = -\left[\frac{\alpha(k-1)}{\alpha(k-1)+p}\right]^2 \left[2\mu[\alpha(k-1)+p]-\mu^2\right]. \tag{4.2.73}$$

A negative definite ΔV will insure stability of the origin in the $\alpha(k)$ plane. By Assumption 4.2.1, $\hat{p}(k-1) > 0 \quad \forall k \geq 1$ because $p > 0$. Therefore from (4.2.69),

$$\hat{p}(k-1) > 0 \Rightarrow \alpha(k-1) + p > 0\,. \tag{4.2.74}$$

In order to meet the stability requirement for ΔV

$$2\mu[\alpha(k-1)+p] - \mu^2 > 0\,. \tag{4.2.75}$$

Therefore

$$\mu_{\max} < 2\min_k[\alpha(k-1)+p] = 2p\,. \tag{4.2.76}$$

A similar analysis for the case of a negative unknown parameter leads to the same $\mu_{\max}$.

The adaptive law, equation (4.2.66), is capable of one step convergence with the proper choice of μ. Let $\hat{p}(1)=p$. Then from equation (4.2.66)

$$p = \hat{p}(0) - \mu\left[1 - \frac{p}{\hat{p}(0)}\right]. \tag{4.2.78}$$

Solving (4.2.78) for μ yields

$$\mu_{opt} = \hat{p}(0) \tag{4.2.79}$$

for a positive parameter. For a negative parameter

$$\mu_{opt} = -\hat{p}(0)\,. \tag{4.2.80}$$

Using μ_{opt} in (4.2.66)

$$\hat{p}(1) = \hat{p}(0) - \hat{p}(0)\left[1 - \frac{p}{\hat{p}(0)}\right] = p\,. \tag{4.2.81}$$

If the unknown, p, is negative,

$$\hat{p}(1) = -\hat{p}(0) + \hat{p}(0)\left[1 - \frac{p}{\hat{p}(0)}\right] = -p\,. \tag{4.2.82}$$

Figure 4.3 shows the parameter convergence for several values of the adaptive gain including μ_{opt}. Figure 4.4 shows the parameter estimate for a μ value close to but less than the stability limit. Figure 4.5 depicts the estimate when μ is set to a value slightly greater than $\mu_{\max}$. Figure 4.6 shows the response for a positive parameter when $\hat{p}(0)$ is either greater than or less than the true parameter. The lack of symmetry in the curves is due to the larger correction made when $\hat{p}(0)$ has a smaller value. Figure 4.7 is similar to Figure 4.6 except the unknown parameter is negative.

Asymptotic stability means that a response trajectory, in this case a parameter estimate, comes closer to the equilibrium value on each successive iteration. The above analysis shows that the adaptive law is locally asymptotically stable depending on the gain. However it is difficult to select the adaptive law gain to insure stability unless a conservative a-priori estimate of the unknown parameter is available. This is because the maximum gain depends on the unknown parameter. A better approach, in an ideal setting, is to set the adaptive law gain according to equation (4.2.79) or (4.2.80). This would guarantee both stability and one step convergence.

The stability and convergence characteristics shown above are present if the adaptive law is operating closed loop on itself. When other dynamic elements are included in the loop then the error signal deviates from the ideal case and μ has to be adjusted away from the ideal value to compensate.

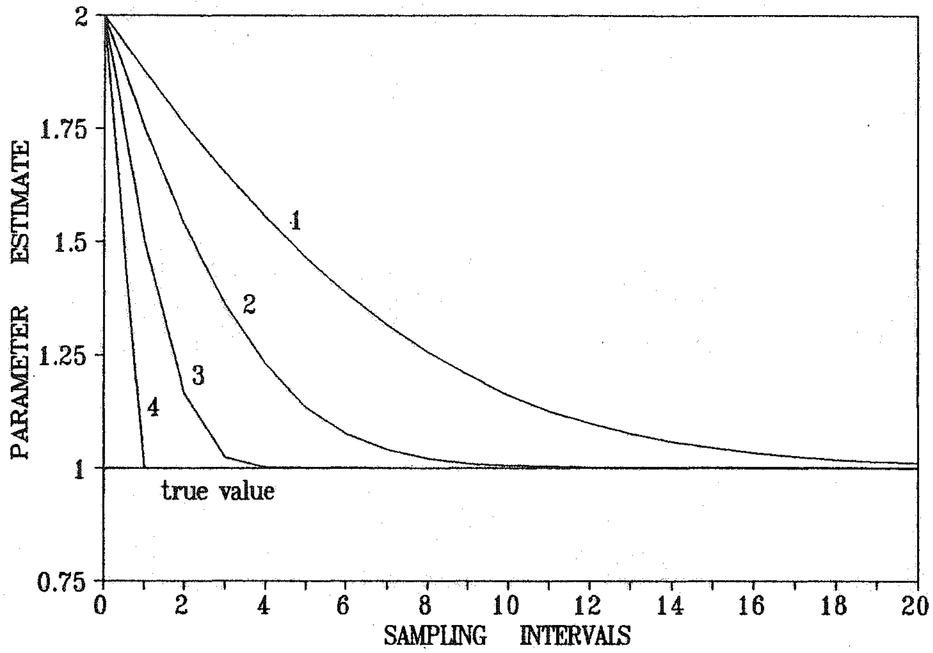

Figure 4.3 Adaptive law convergence. Curve 1, $\mu=0.25$; Curve 2, $\mu=0.5$; Curve 3, $\mu=1.0$; Curve 4, $\mu=\mu_{\text{opt}}=2.0$.

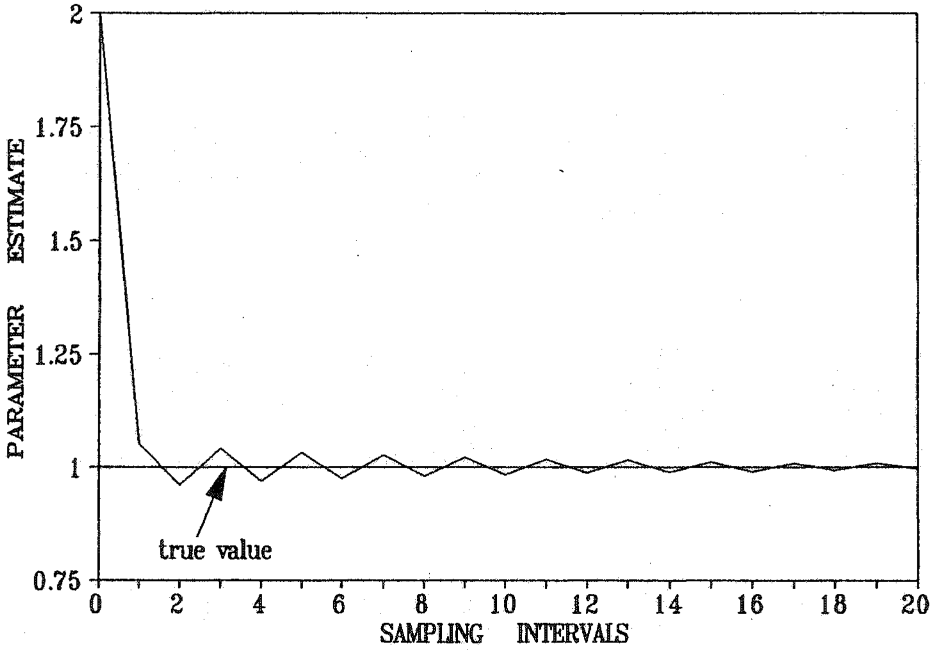

Figure 4.4 Adaptive law convergence for μ close to $\mu_{\max}$. $\mu_{\max}=2.0$, $\mu=1.9$.

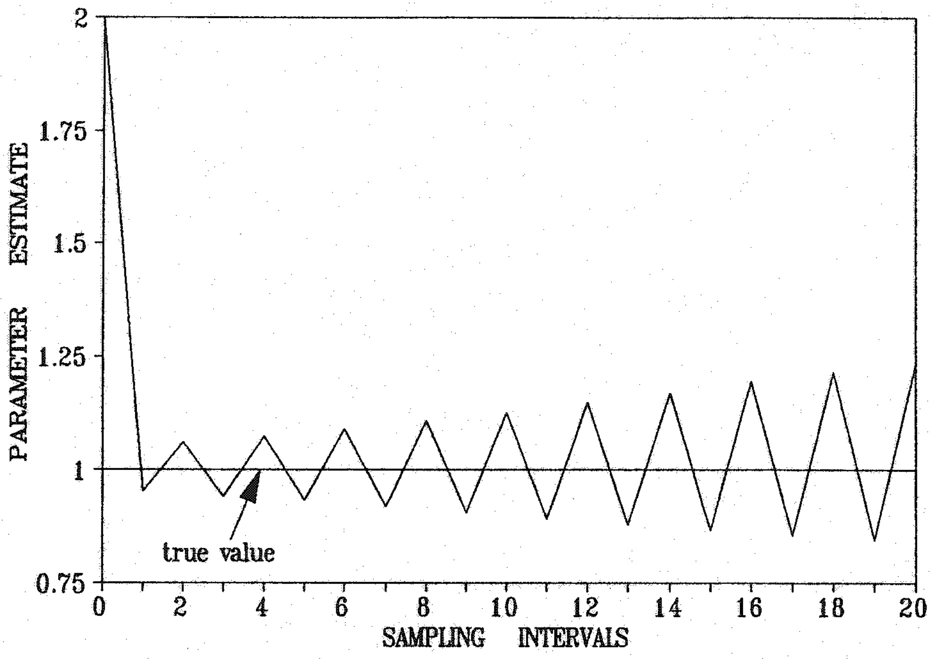

Figure 4.5 Adaptive law convergence for $\mu > \mu_{\max}$.
$\mu_{\max}=2.0$ $\mu=2.1$.

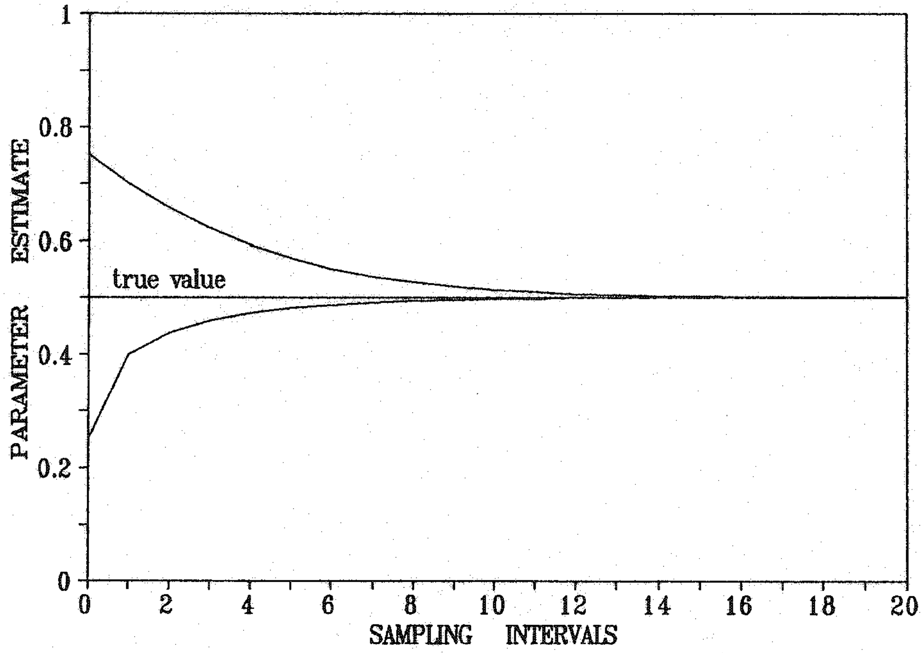

Figure 4.6 Positive parameter convergence for two different initial values.
$\hat{p}(0)=0.25$ and 0.75. $\mu=0.15$.

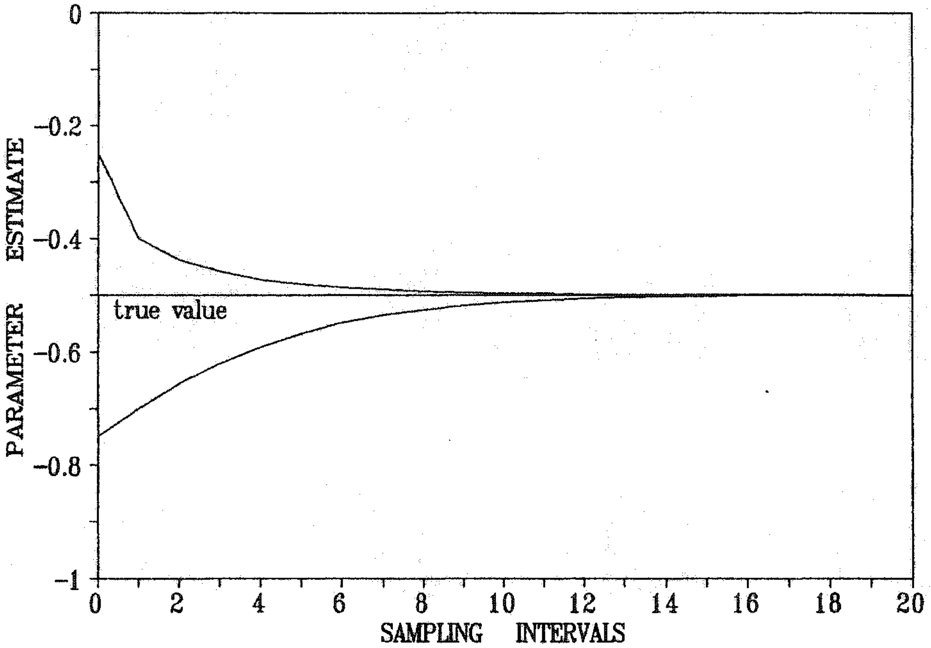

Figure 4.7 Negative parameter convergence for two different initial values. $\hat{p}(0)=-0.25$ and -0.75. $\mu=0.15$.

4.3 Simulation Evaluation of Self-Tuning Feedback Linearization

In Section 4.2.2, Figure 4.1, a new architecture for an adaptive or self-tuning feedback linearization system was proposed. In this section the viability of this new scheme will be demonstrated using simulations of nonlinear plants. The input signal tracking capability will also be demonstrated.

4.3.1 Plant Modelling For Simulation Studies

Simulation studies are commonly carried out before implementing a new control strategy on a real plant. In self-tuning feedback linearization applications the real plant may appear to change in structure asymptotically as the parameter estimates converge toward their true values. This happens because the control law becomes more effective in neutralizing the effect of nonlinear terms in the real plant as time progresses. Therefore the plant model chosen to represent the real plant in a simulation must be capable of reproducing any apparent structural changes that the real plant exhibits. In the following when structural change is referred to it should be interpreted to mean an apparent structural change. In other words it does not mean that the real physical plant changes in structure but only appears to do so in response to the linearizing control law.

The Runge-Kutta method is a well known solution technique which is applicable to both linear and nonlinear differential equations (Lastman & Sinha (1989)). It can also be applied to systems of first order ordinary differential equations. The method uses the exact differential equations which represent the plant at all times and will therefore respond like the real plant for any input including linearizing control laws. Runge-Kutta models therefore have the necessary structural change capability needed to track the response of the real plant.

A four stage Runge-Kutta method consists of the following for a system of *n* first order differential equations given by

$$\begin{bmatrix} \dot{x}_1 \\ \vdots \\ \dot{x}_n \end{bmatrix} = \begin{bmatrix} f_1(x,u,t) \\ \vdots \\ f_n(x,u,t) \end{bmatrix} \tag{4.3.12}$$

where x is the state vector and u is the plant input. The functions f_i may be nonlinear. Each of the state equations in (4.3.12) can be approximated by

$$K_{1i} = f_i(x_k, u(t_k), t_k)h \tag{4.3.13}$$

$$K_{2i} = f_i\left[(x_k + MK_{1i}), u(t_k + Mh), (t_k + Mh)\right]h \tag{4.3.14}$$

$$K_{3i} = f_i\left[(x_k + MK_{2i}), u(t_k + Mh), (t_k + Mh)\right]h \tag{4.3.15}$$

$$K_{4i} = f_i\left[(x_k + K_{3i}), u(t_k + h), (t_k + h)\right]h \tag{4.3.16}$$

$$x_{i,k+1} = x_{i,k} + (K_{1i} + 2K_{2i} + 2K_{3i} + K_{4i}) / 6 \tag{4.3.17}$$

where $M = 0.5$, h is the increment in the independent variable and $x_{i,k}$ denotes the approximate value of the *i'th* component of the state vector at $t = t_k$. Evaluation of (4.3.13) through (4.3.17) over a period of time yields the state trajectories. Error in a four stage R-K solution is $O(h^5)$.

The gradient WRT $\hat{\theta}_i$ for an R-K model can be calculated as

$$-\psi_{\hat{\theta}_i,k+1} = \frac{dx_{i,k}}{d\hat{\theta}_i} + \frac{1}{6}\left[\frac{dK_{1i}}{d\hat{\theta}_i} + 2\frac{dK_{2i}}{d\hat{\theta}_i} + 2\frac{dK_{3i}}{d\hat{\theta}_i} + \frac{dK_{4i}}{d\hat{\theta}_i}\right] \tag{4.3.18}$$

where

$$\frac{dx_{i,k}}{d\hat{\theta}_i} = -\psi_{\hat{\theta}_i,k} .$$

The gradient equation involves four derivatives for each unknown parameter for each of the n differential equations in the system.

Another solution method which has the needed structural change property is the Euler representation,

$$x_{i,k+1} = x_{i,k} + f_i(x_{i,k}, u(t_k), t_k)h \tag{4.3.19}$$

having an accuracy $O(h)$. This method is also applicable to a system of first order equations. While less accurate than Runge-Kutta, calculation of the gradient is much simplier than with R-K models. Both representations will be used for the simulation studies in the following section.

4.3.2 Parameter Estimator Imbedded Model

In section 4.3.3 simulations will be carried out using the system configuration shown in Figure 4.1. For the estimation algorithm,. the imbedded plant model is the discrete representation of the linearized state space equations. Here the discrete representation can be used to model the linearized plant because it is not required to undergo any structural changes. The only change is in the number of inputs as time evolves. The number of inputs decrease as parameter estimates converge to the true values.

4.3.3 Simulation Results

A number of simulations have been run to demonstrate the performance of this new technique. The next section deals with adaptive feedback linearization of plants whose states are measurable. In section 4.3.5 adaptive tracking of a feedback linearizable plant is demonstrated.

Before proceeding with the evaluation of the new system architecture and adaptive law with nonlinear plants, we will revisit the linear plant of Chapter 3, (cf.(3.3.7a,b)). This plant was used to demonstrate the problem with the standard system when feedback linearization is used. In this chapter we use the new system architecture and adaptive law to demonstrate that the problem experienced with the standard system and this plant in Chapter 3 has been solved.

The system architecture is as shown in Figure 4.1. The linearizing control law is given by (3.3.9) with a_1 an unknown parameter, i.e.

$$u = V + \hat{a}_1 x_1 - dx_1 - x_2 .$$

The $-dx_1$ term is contributed by the outer feedback loop. In this relative order one case, the model for the linearized plant could be written down directly. However, as a further illustration of the theory in this chapter, we will derive the model from (4.2.6). For this plant $z_1 = x_1$. Evaluating the terms in (4.2.6) we get

$$\begin{aligned} \dot{z}_1 &= (-a_1 x_1 + x_2) - (-\hat{a}_1 x_1 + x_2) + (V - dx_1) \\ &= \hat{\alpha} x_1 - d x_1 + V \end{aligned} \tag{4.3.20}$$

where $\hat{\alpha} \doteq \hat{a}_1 - a_1$.

In Chapter 3 and again in this chapter we are still considering deterministic systems. In Chapter 3 it was pointed out that the output error signal vanishes if the estimated parameters converge. Vanishing of the innovation will make the Kalman gain unobservable and cause certain elements of the covariance matrix to increase without bound. To demonstrate this we will initially use an innovations model.

The continuous linearized plant model will be discretized using a first order Euler expansion,

$$z_1(k+1) = z_1(k) + \left[\hat{\alpha} z_1(k) - dz_1(k) + V + \hat{K}\varepsilon(k)\right] h \tag{4.3.21}$$

where

$$\varepsilon(k) = y(k) - z_1(k) \, .$$

The gradient is calculated as follows, (cf. section 3.2.2).

$$\hat{\theta}^T \doteq \left[\hat{\alpha}, \hat{K}\right] \tag{4.3.22}$$

$$\psi(k) = W^T(k)\, C^T(\hat{\theta}) + D^T(k) \tag{4.3.23}$$

$$W(k+1) = W(k) + \left[(\hat{\alpha} - d - \hat{K}) W(k) + \left[z_1(k) \; \varepsilon(k)\right]\right] h \tag{4.3.24}$$

Expanding (4.3.23) and substituting $W_{11}(k+1)$ and $W_{12}(k+1)$ from (4.3.24) we find the gradients for $\hat{\alpha}$ and $\hat{K}$ to be

$$\psi_{\hat{\alpha}}(k+1) = W_{11}(k) + \left[(\hat{\alpha} - d - \hat{K}) W_{11}(k) + z_1(k)\right] h \tag{4.3.25}$$

and

$$\psi_{\hat{K}}(k+1) = W_{12}(k) + \left[(\hat{\alpha} - d - \hat{K}) W_{12}(k) + \varepsilon(k)\right] h \, . \tag{4.3.26}$$

The adaptive law is given in Theorem 4.2.1. Inspection of the linearized plant model equation (4.3.20) shows that the error term in the adaptive law should be defined by (4.2.56) where

$$e(k) = \frac{\hat{\theta}_i(k)}{\hat{p}_i(k)} = \frac{\hat{\alpha}(k)}{\hat{a}_1(k)} . \tag{4.3.27}$$

The sign of $\hat{a}_1$ as used in the control law is positive. Using both integral and proportional responses, the adaptive law is given by the equations in Remark 2, page 134. Set up for the present case, these equations are

$$I(k) = I(k-1) - \mu_{\hat{a}_1} e(k) \tag{4.3.28}$$

and

$$\hat{a}_1 = I(k) - .05 e(k) \tag{4.3.29}$$

where

$$\mu_{\hat{a}_1} = \frac{.07}{|e(k)| + .75} .$$

Integral and proportional gain constants were empirically determined. The range of values for these parameters is application dependent. In the present case $.04 \leq \mu(0) \leq .09$ gave reasonable results with $\alpha = 0.75$. Proportional gain shortens the convergence time and can be adjusted as needed to get a desired response in the parameter estimate. The Kalman gain estimate is obtained directly from the RPEM.

Figure 4.8 is a plot of $\hat{a}_1$ and $\hat{\alpha}$. The initial parameter estimate was 0.5 and the true value of the parameter is 4.0. Since $\hat{\alpha} = \hat{a}_1 - a$, in this example, $\hat{\alpha}$ is negative as would be expected. The $\hat{\alpha}$ response is somewhat distorted from what might be expected due to the effect of the Kalman gain and innovations term in the plant model. This is evident by comparing Figure 4.8 and Figure 4.17 for the non-innovations model. However both $\hat{\alpha}$ and $\hat{a}_1$ converge to the expected values. Figure 4.10 is a plot of the Kalman gain estimate and the innovations. The Kalman gain has been scaled down by a factor of fifty for plotting compatibility with the much smaller innovations signal.

Figures 4.9 and 4.11 are plots of the $\hat{\alpha}$ and $\hat{K}$ gradients respectively. From (4.3.24),

$$W_{11}(k+1) = \left[1 + (\hat{\alpha} - d - \hat{K})h\right]W_{11}(k) + z_1(k)h \tag{4.3.30}$$

$$W_{12}(k+1) = \left[1 + (\hat{\alpha} - d - \hat{K})h\right]W_{12}(k) + \varepsilon(k)h\,. \tag{4.3.31}$$

The solutions of (4.3.30) and (4.3.31) are

$$W_{11}(k) = \phi^k W_{11}(0) + h\sum_{i=0}^{k-1}\phi^{(k-1)-i}z_1(i) \tag{4.3.32}$$

and

$$W_{12}(k) = \phi^k W_{12}(0) + h\sum_{i=0}^{k-1}\phi^{(k-1)-i}\varepsilon(i) \tag{4.3.33}$$

respectively where

$$\phi \doteq 1 + (\hat{\alpha} - d - \hat{K})h\,. \tag{4.3.34}$$

As we have seen, $\hat{\alpha}$ converges to zero. If $0 < |d+\hat{K}|\,h < 2$ the first terms on the right of the equals sign in (4.3.32) and (4.3.33) will vanish. The sampling rate is 0.025 so $|d+\hat{K}|$ would have to be an unreasonably large value (i.e. 80 or greater) for the above condition not to be met. In this simulation the condition

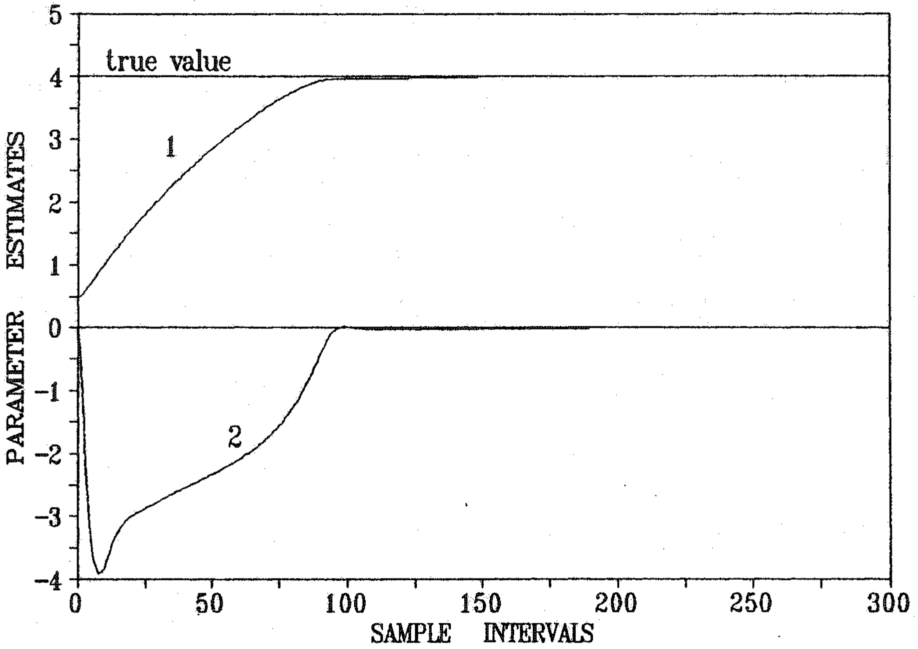

Figure 4.8 Plant and model parameter estimates, innovations model. Curve 1, plant parameter, $\hat{a}_1$ Curve 2, model parameter, $\hat{\alpha}$.

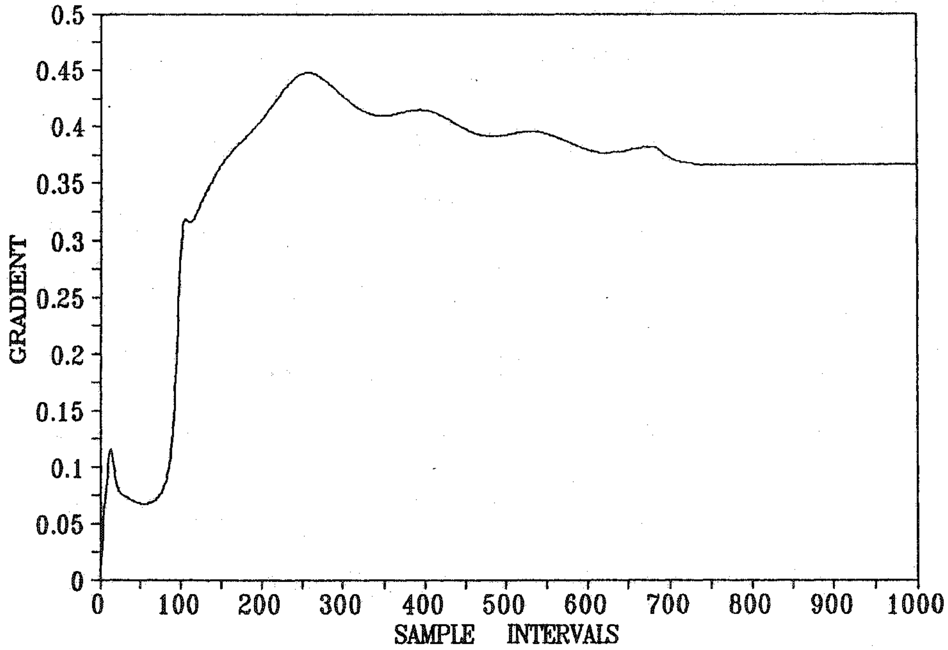

Figure 4.9 $\hat{\alpha}$ gradient, innovations model.

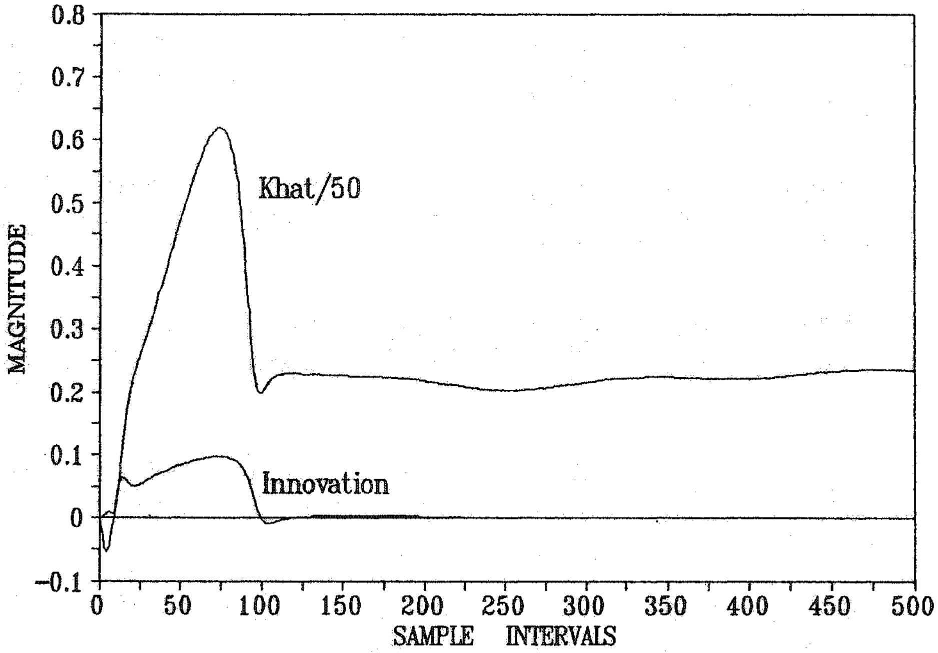

Figure 4.10 Kalman gain estimate and innovations signal.

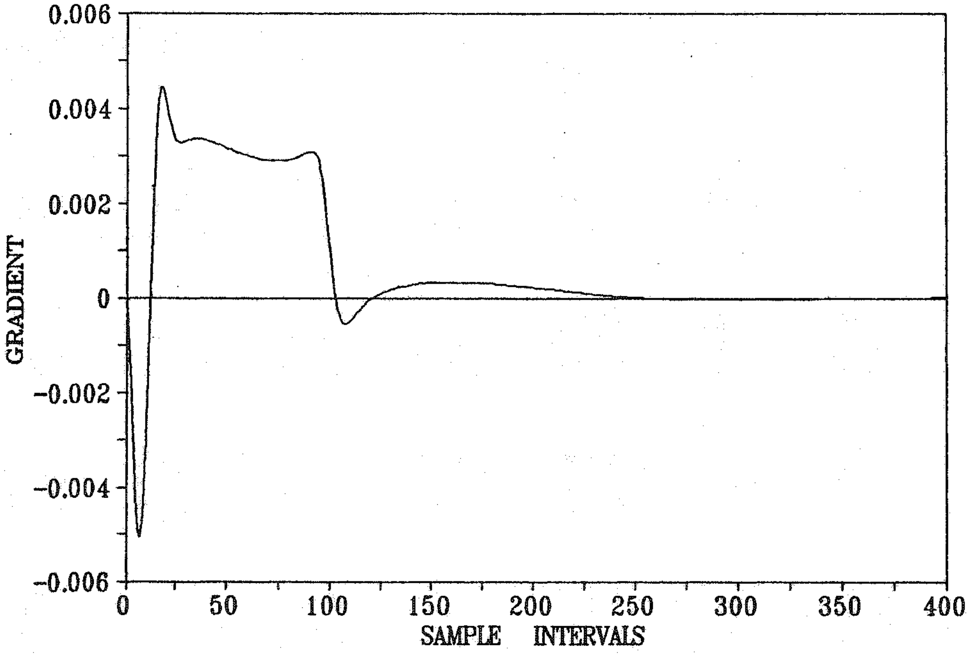

Figure 4.11 Kalman gain gradient, for $\hat{K}$.

is met, $\phi < 1$, and so these terms do vanish. The second term on the right in (4.3.33) will vanish since $\varepsilon(k)$ converges to zero, (cf. Figure 4.10). The second rightmost term in (4.3.32) will always have a value because in general the plant output, x_1, $(z_1 = x_1)$ is not zero and is unipolar. Figure 4.12 is a plot of the covariance update denominator. Plots of the covariance matrix elements are shown in Figure 4.13, 4.14 and 4.15. Figure 4.16 shows the tracking between the plant output and the model output during the self-tuning linearization cycle.

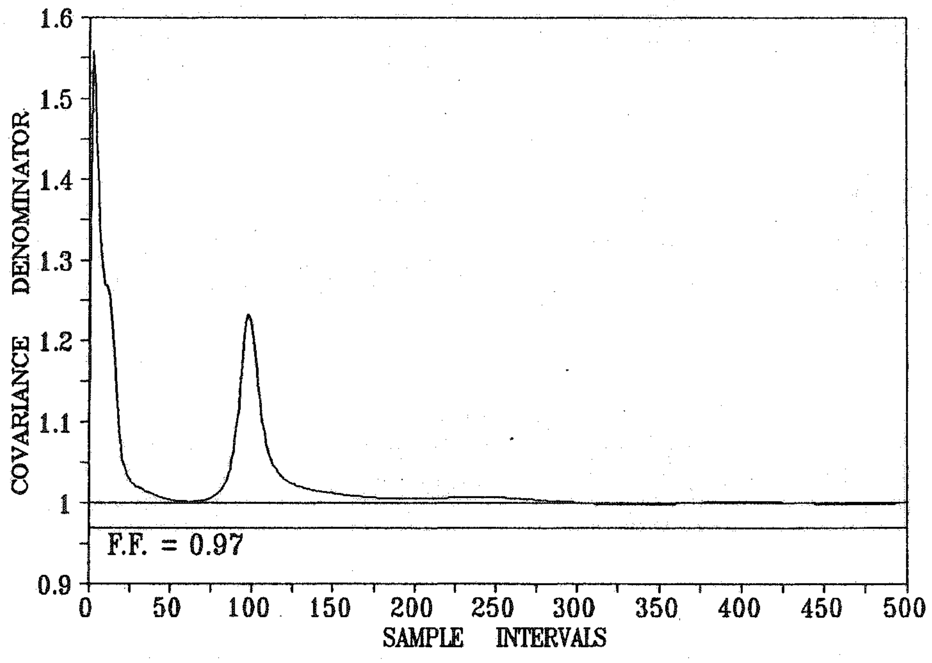

Figure 4.12 Plot of covariance denominator.

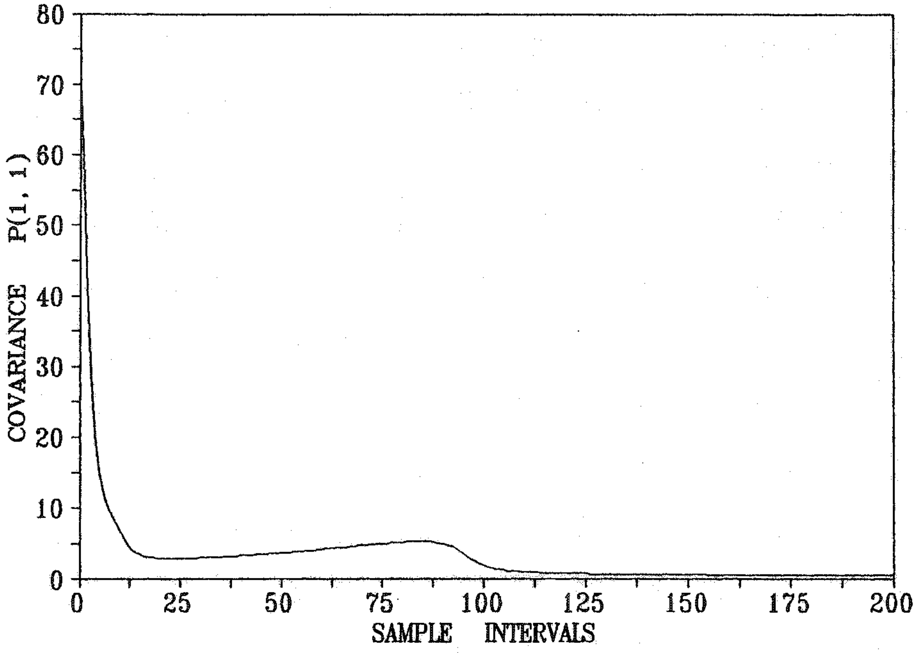

Figure 4.13 Plot of the square root of covariance element p(1,1).

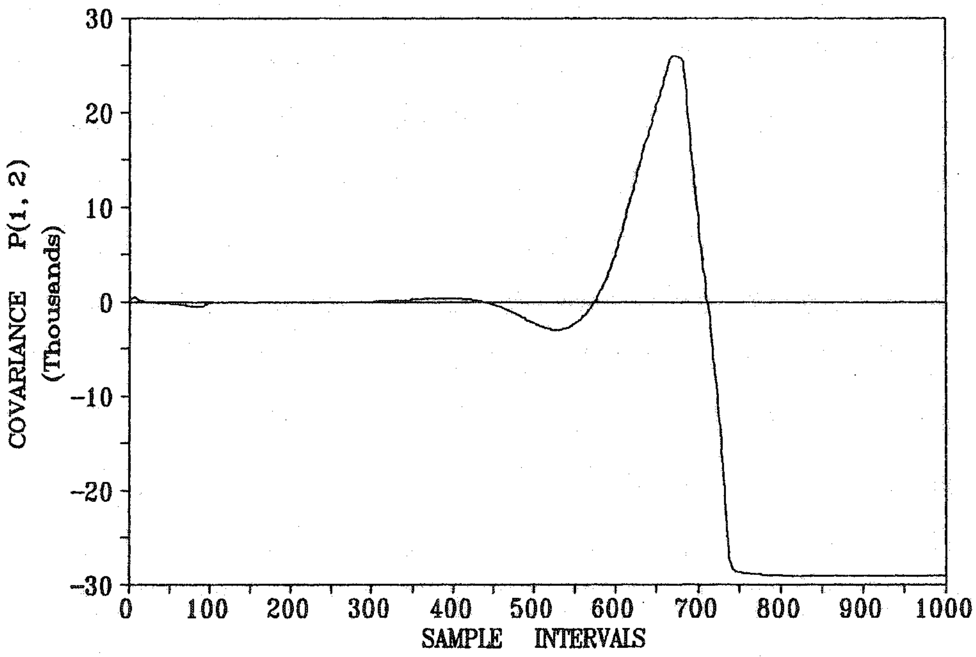

Figure 4.14 Plot of covariance element p(1,2).

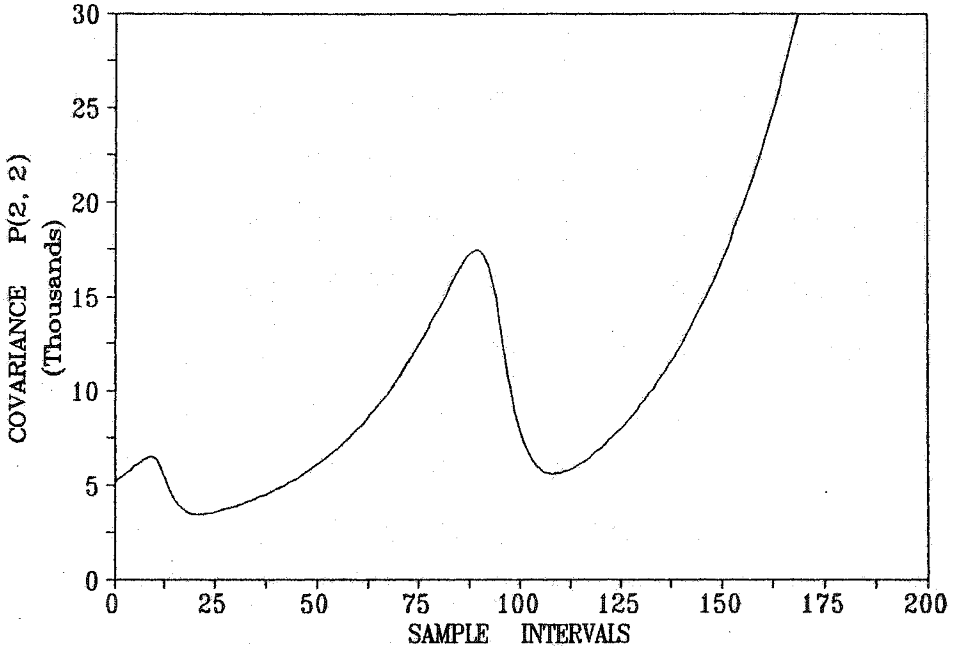

Figure 4.15 Plot of covariance element p(2,2).

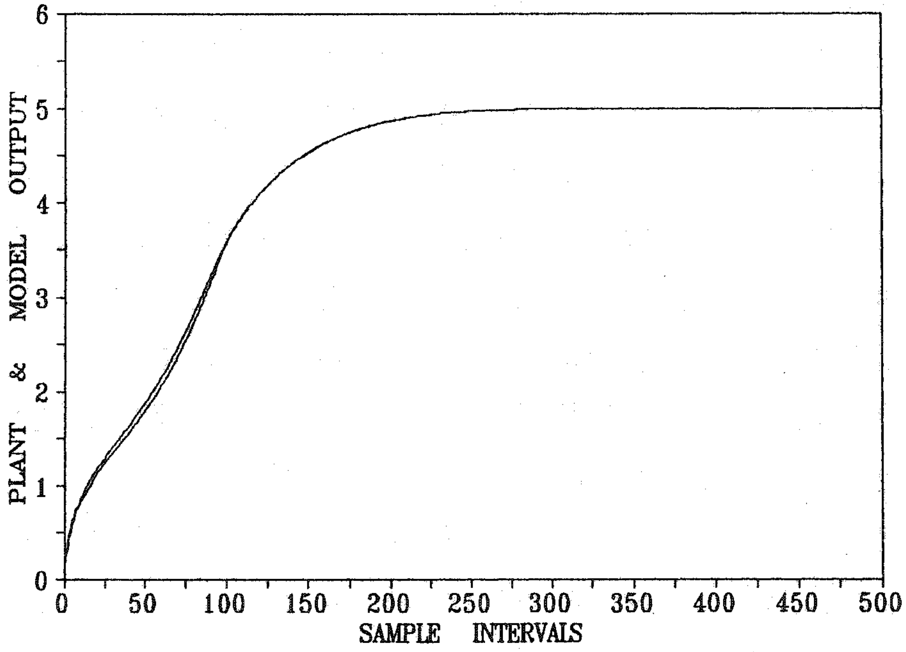

Figure 4.16 Plant and model output tracking during control law tuning

For the deterministic case, an innovations model is not needed and indeed cannot be used due to the unbounded response of elements in the covariance matrix. Returning to (4.3.21) the plant model without the innovations term is

$$z_1(k+1) = z_1(k) + \left[\hat{\alpha} z_1(k) - d z_1(k) + V\right] h . \tag{4.3.35}$$

Following the same steps as for the innovations model, we arrive at

$$W_{11}(k+1) = \left[1 + (\hat{\alpha} - d)h\right] W_{11}(k) + z_1(k)h \tag{4.3.36}$$

$$\begin{aligned} \psi_{\hat{\alpha}}(k+1) &= W_{11}(k) + \left[(\hat{\alpha} - d) W_{11}(k) + z_1(k)\right] h \\ &= \left[1 + (\hat{\alpha} - d)h\right] W_{11}(k) + z_1(k)h \end{aligned} \tag{4.3.37}$$

and

$$W_{11}(k) = \phi^k W_{11}(0) + h \sum_{i=0}^{k-1} \phi^{(k-1)-i} z_1(i) \tag{4.3.38}$$

where

$$\phi = 1 + (\hat{\alpha} - d)h . \tag{4.3.39}$$

A discussion of ϕ analogous to that following (4.3.34) also applies to (4.3.39). The conclusion that the gradient does not vanish can be reached by analyzing (4.3.38) similar to the analysis given for (4.3.32).

Figure 4.17 is a plot of $\hat{\alpha}$ and $\hat{a}_1$ for the plant model (4.3.35). Convergence of these two parameters is a little slower than it was for the innovations model. (cf. Figure 4.8) The main feature is that the covariance remains bounded as shown in Figure 4.20.

Figures 4.18, 4.19 and 4.21 for the standard model are the counterparts of Figures 4.9, 4.12 and 4.16 respectively for the innovations model.

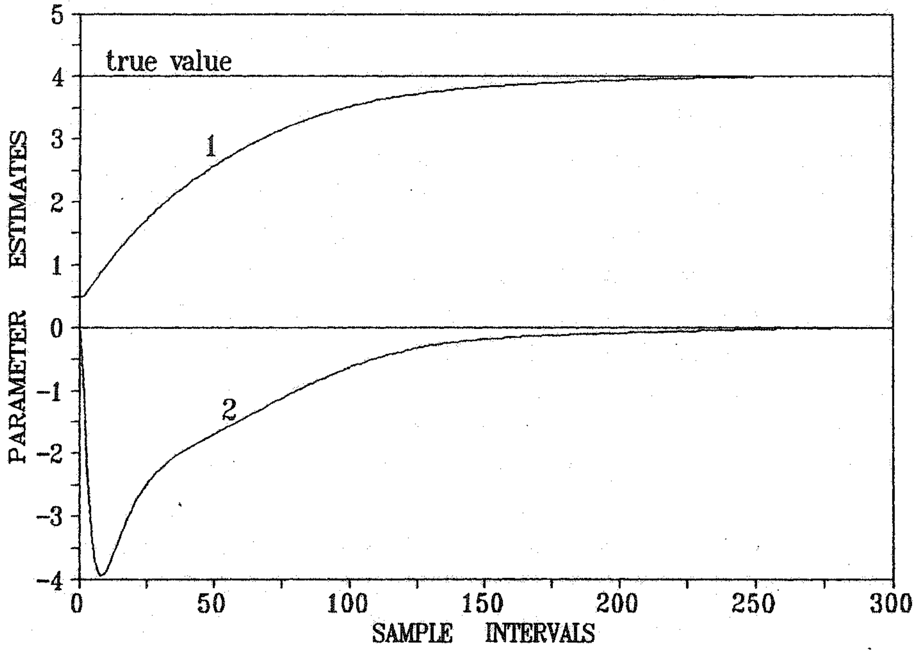

Figure 4.17 Plant and model parameter estimates, non-innovations model. Curve 1, plant parameter, $\hat{a}_1$ Curve 2, model parameter, $\hat{\alpha}$.

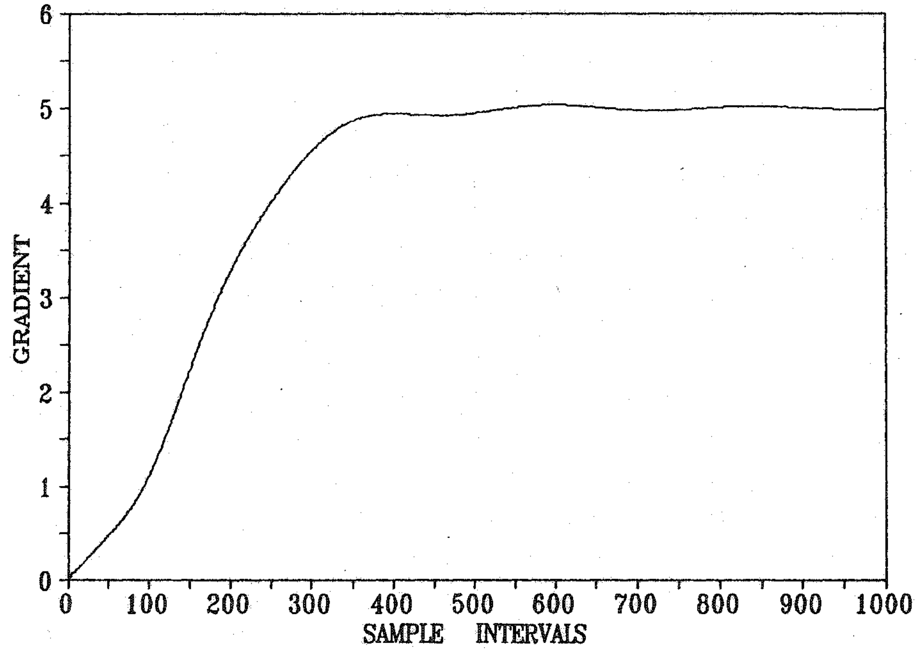

Figure 4.18 $\hat{\alpha}$ gradient, non-innovations model.

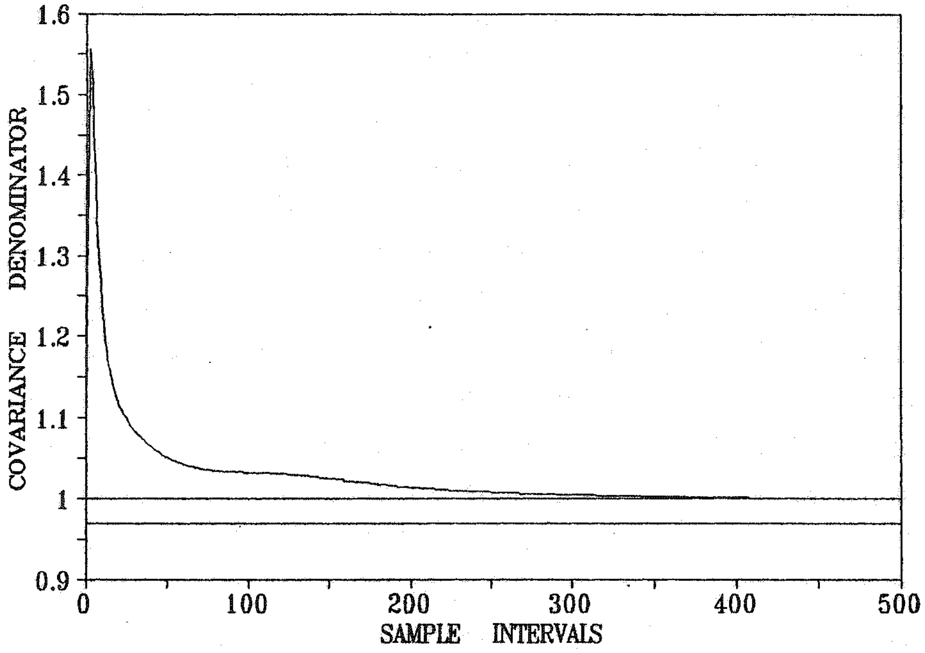

Figure 4.19 Plot of the covariance denominator.

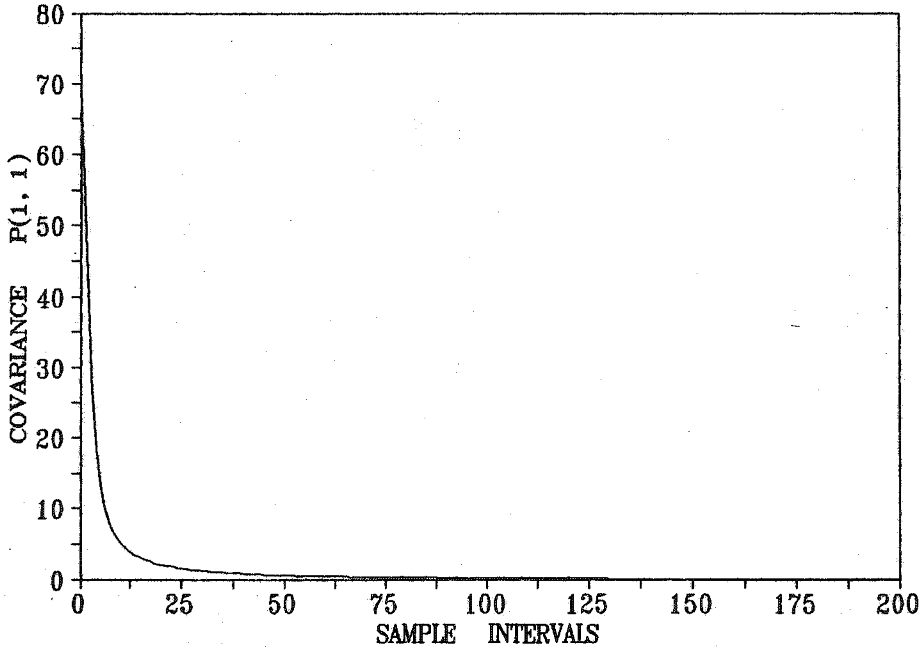

Figure 4.20 Plot of the square root of covariance element p(1,1).

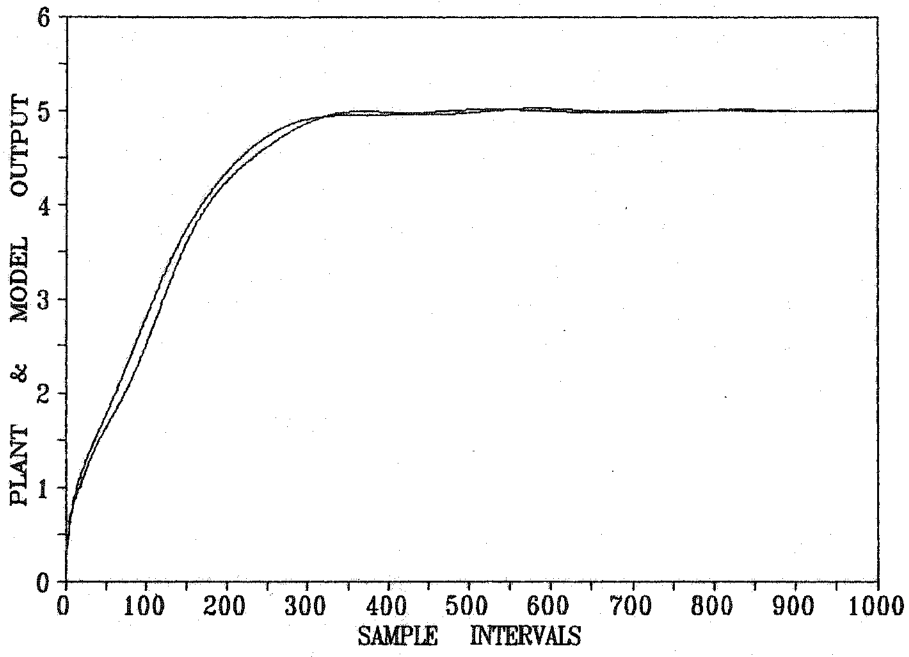

Figure 4.21 Plant and model output tracking during control law tuning.

4.3.4 Nonlinear Plant Self-tuning Feedback Linearization

Again consider the feedback linearizable plant,

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} a\sin(x_2) \\ -bx_1^2 \end{bmatrix} + \begin{bmatrix} 0 \\ c \end{bmatrix} u \tag{4.3.40a}$$

$$y = x_1, \tag{4.3.40b}$$

which was discussed in Section 4.2.1. The true values of the parameters are *a=0.785*, *b=5*, and *c=1*. We will initially consider *a* to be the only unknown parameter. Also recall from the discussion in Chapter 2, section 2.6.2 that state feedback is needed to stabilize and control the linearized plant. Following the procedure developed in section 4.2.1, the linear model for the plant is

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -d_1 & -d_2 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \hat{\delta} \end{bmatrix} V \qquad (4.3.41)$$

where d_1 and d_2 are feedback gains. These gains are determined when pole locations for the closed loop linearized plant are chosen. For this simulation the poles were set at $s_1 = -5$ and $s_2 = -10$. With these poles the characteristic equation is

$$s^2 + 15s + 50 = 0$$

and therefore $d_1 = 50$ and $d_2 = 15$. Equation (4.3.41) is the modified version of (4.2.15) (i.e. feedback has been added). The linearizing control law for (4.3.40a,b) with a unknown is

$$u(x,\hat{a}) = \frac{1}{c}\left[bx_1^2 + \frac{V}{\hat{a}\cos(x_2)}\right]. \qquad (4.3.42)$$

The complete architecture of the adaptive feedback linearizing system for this example is shown in Figure 4.22. Figure 4.23 is a plot of $\hat{\delta}$ and $\hat{a}$. As can be seen, both parameters converge to their true values of, $\hat{\delta} = 1$, and $\hat{a} = .785$. The filtering effect of integral action in the adaptive law produces a smoother response in the $\hat{a}$ estimate. The sample interval is 0.025 time units.

When more than one parameter estimate is needed the system architecture is basically the same as shown in Figure 4.22 but adaptive laws for the additional parameters are added to the system. Errors in the second parameter estimate produce another input to the linearized plant model. Suppose a and b are both unknown parameters in control law (4.3.42). Then the estimator model is

(4.2.25) with feedback added,

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -d_1 & -d_2 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \hat{\delta} \end{bmatrix} V - \begin{bmatrix} 0 \\ \hat{\beta} \end{bmatrix} a(\hat{\delta}) x_1^2 \cos(x_2) \quad (4.3.43)$$

where a($\hat{\delta}$) denotes the dependence of a on $\hat{\delta}$. The unknown parameter a in (4.3.43) is replaced by an estimate calculated from $\hat{a}_{k-1}$ and $\hat{\delta}_{k-1}$,

$$\hat{a}_k = \hat{a}_{k-1} \hat{\delta}_{k-1} \, . \quad (4.3.44)$$

Figures 4.24a and 4.24b are plots of the estimates for two unknown parameters, a and b. The true values for $\hat{\delta}$ and $\hat{a}$ are the same as before. For the other pair of parameters the true values are $\hat{\beta} = 0$ and $\hat{b} = 5$. Again all estimates converge to their true values. The convergence rate for $\hat{\delta}$ and $\hat{a}$ is slower in this case than it was in the first example. Note the factor of four difference in the time scales of Figures 4.23 and 4.24a. The slower convergence is probably due to the interaction caused by the simultaneous adjustment of two parameters.

This is a case where performance was improved by adding proportional response to the adaptive law for $\hat{b}$. (refer to Remark 2, pg 26). The proportional term significantly reduced the convergence time for this parameter. Figure 4.25 is a plot of $\hat{\beta}$ and $\hat{b}$ without proportional response. Note that the time scale is five times longer in Figure 4.25 than it is in Figure 4.24b. A slight additional improvement in response time was realized by passing the$\hat{\beta}$ estimate through a low pass filter to reduce the initial transient peaks and using the filtered value in the calculations. However this result was not plotted.

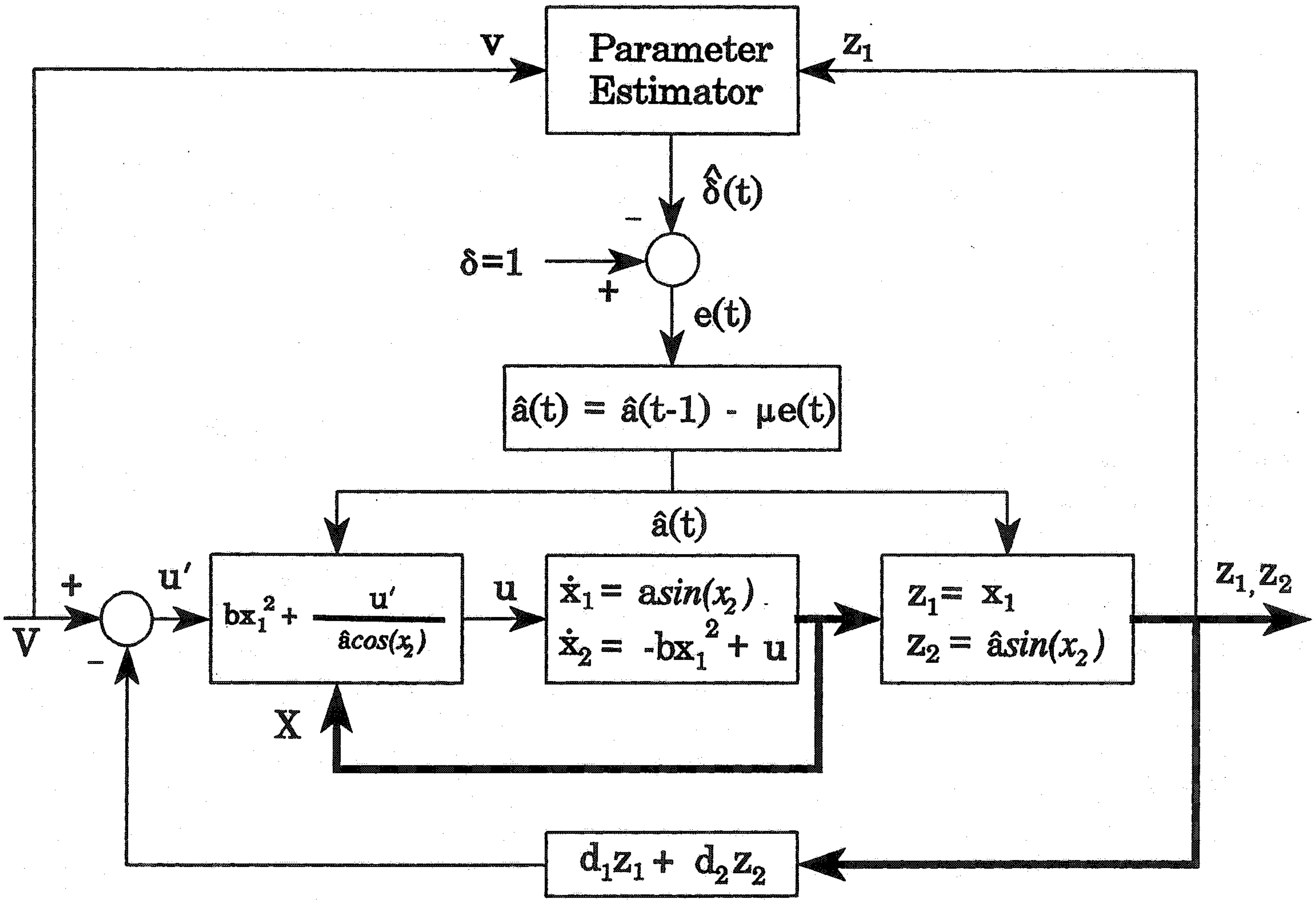

Figure 4.22 Architecture of self-tuning feedback linearizing controller

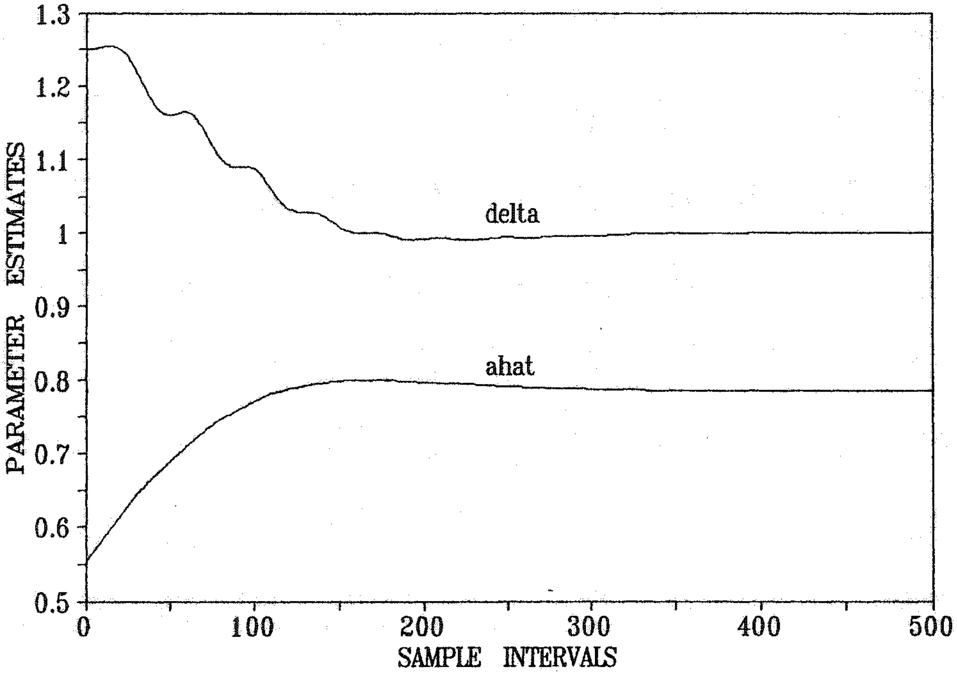

Figure 4.23 Linear model, $\hat{\delta}$, and nonlinear plant, â, parameter estimates.

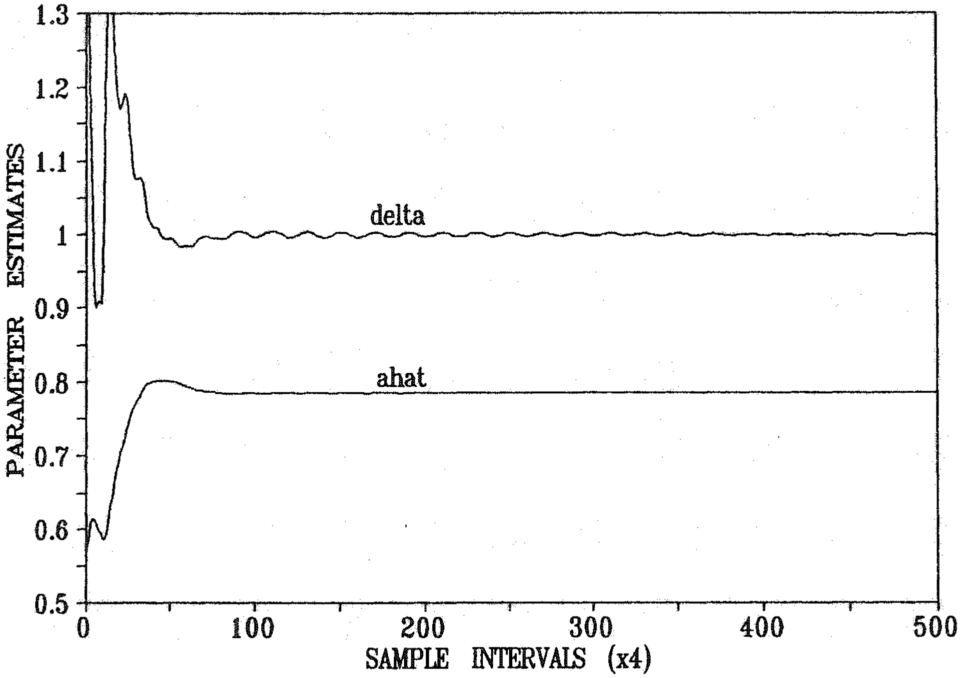

Figure 4.24a Linear model, $\hat{\delta}$, and nonlinear plant, â, estimates; two unknown parameters.

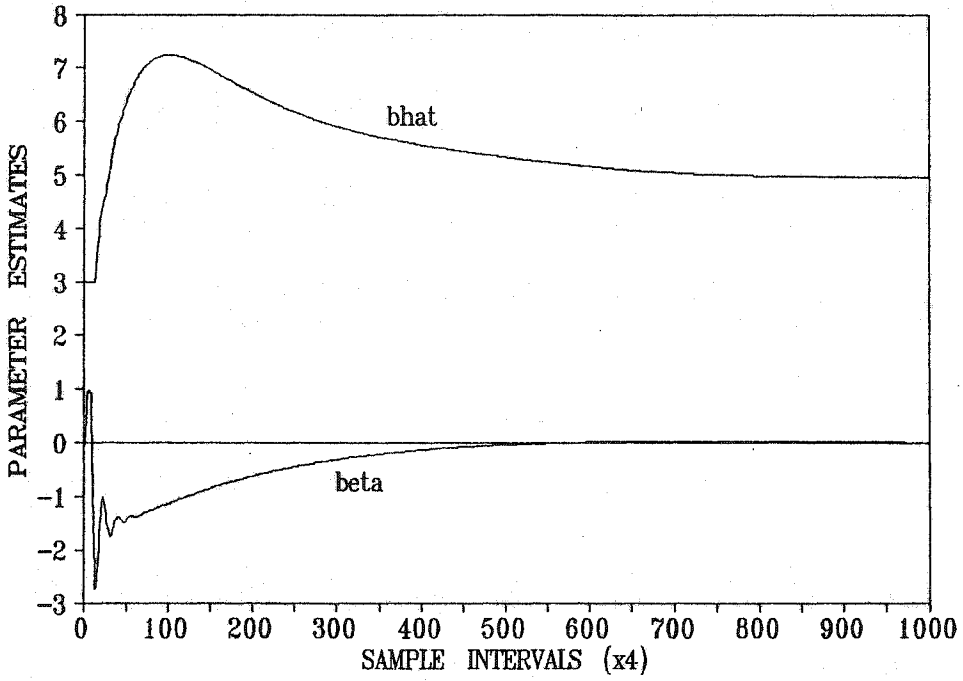

Figure 4.24b Linear model, $\hat{\beta}$, and nonlinear plant, $\hat{b}$, estimates two unknown parameters.

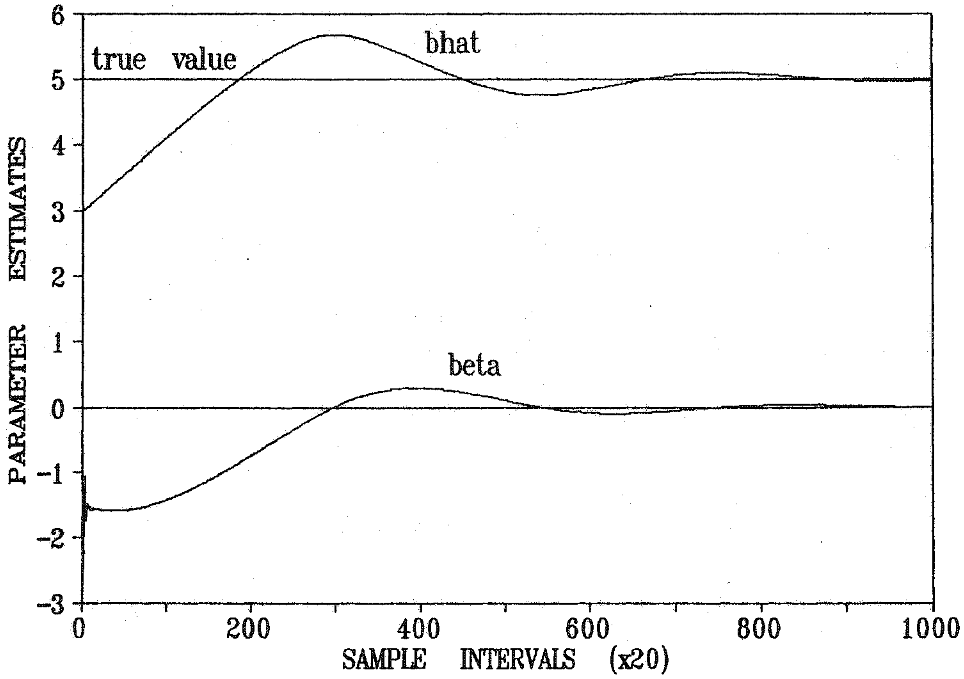

Figure 4.25 Same as Figure 4.24b but without proportional gain.

4.3.5 Tracking of Self-tuning Feedback Linearizable Plants

In many applications the control objective is to track an input signal. This can easily be done with a feedback linearizable system. The system input signal is set equal to the desired plant output and its derivatives up to the relative order of the plant. Each derivative is multiplied by the appropriate feedback gain. For the plant given by (4.3.40a,b) assume that we want to track the signal

$$v = 0.25\sin(\pi t). \tag{4.3.45}$$

Therefore the system input is set to

$$V = \ddot{y}_d + d_2\dot{y}_d + d_1 y_d \tag{4.3.46}$$

where y_d is the desired output. (i.e. equation (4.3.45)

The control law for this case is

$$u(x,\hat{a}) = \frac{1}{c}\left[bx_1^2 + \frac{\ddot{y}_d - d_2(\hat{a}\sin(x_2) - \dot{y}_d) - d_1(x_1 - y_d)}{\hat{a}\cos(x_2)}\right]. \tag{4.3.47}$$

For the non-tracking case the closed loop poles may be chosen according to the desired time response. In the signal tracking case, the poles determine the tracking error response. With V given by (4.3.46),

$$\ddot{y} = \ddot{y}_d + d_2\dot{y}_d + d_1 y_d - d_2\dot{y} - d_1 y \tag{4.3.48}$$

Rewriting (4.3.48) as

$$(\ddot{y}_d - \ddot{y}) + d_2(\dot{y}_d - \dot{y}) + d_1(y_d - y) = 0 \qquad (4.3.49)$$

yields the tracking error

$$\ddot{e} + d_2\dot{e} + d_1 e = 0 \qquad (4.3.50)$$

where d_1 and d_2 are chosen to provide the desired tracking error response.

Figure 4.26 depicts the tracking error when parameter a is the only unknown. It shows that the error quickly reduces to a negligible value. Figure 4.27 shows the plant output compared with the desired output y_d and Figure 4.28 shows the convergence of the parameter estimates to their true values. Figures 4.29, 4.30, 4.31a, and 4.31b are similar to the above set of tracking curves except that two unknown parameters, a and b, are being estimated. (i.e. b is replaced by $\hat{b}$ in (4.3.47)).

Remark 3 In both tracking examples the parameters converge to their true values. However this is not necessary if the only requirement is for the plant output to track the input signal. This can sometimes occur when the parameters do not converge to their true values.

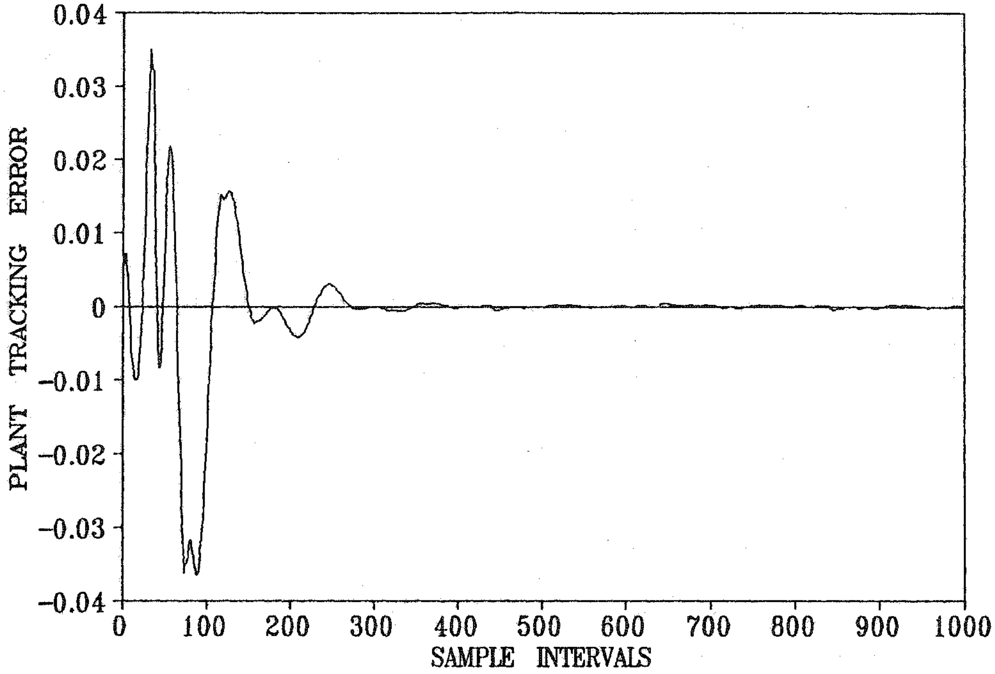

Figure 4.26 Plant tracking error. One unknown, parameter a, estimated.

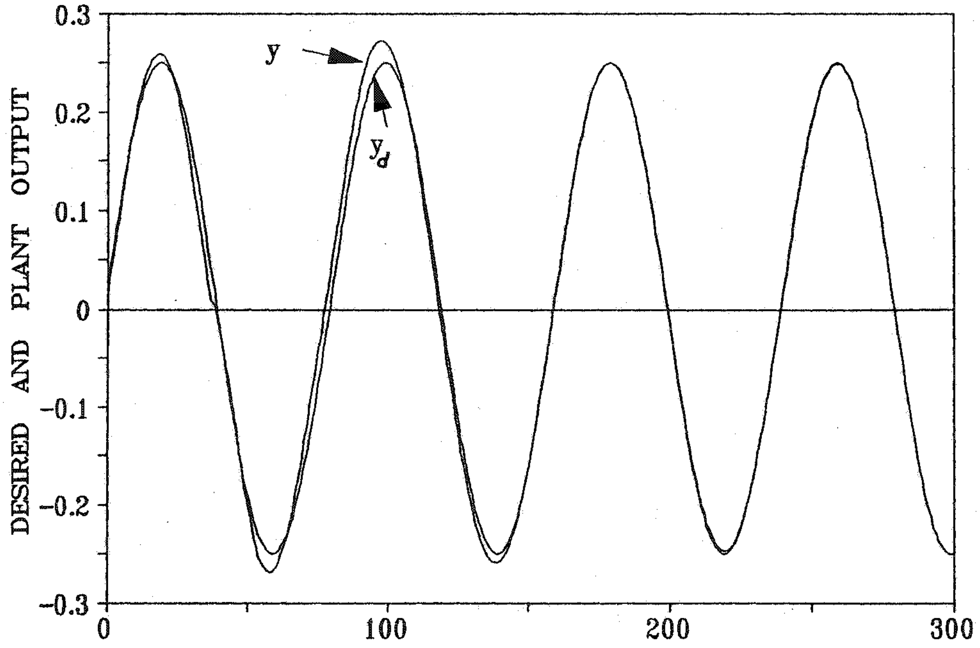

Figure 4.27 Comparison of plant output with desired output.

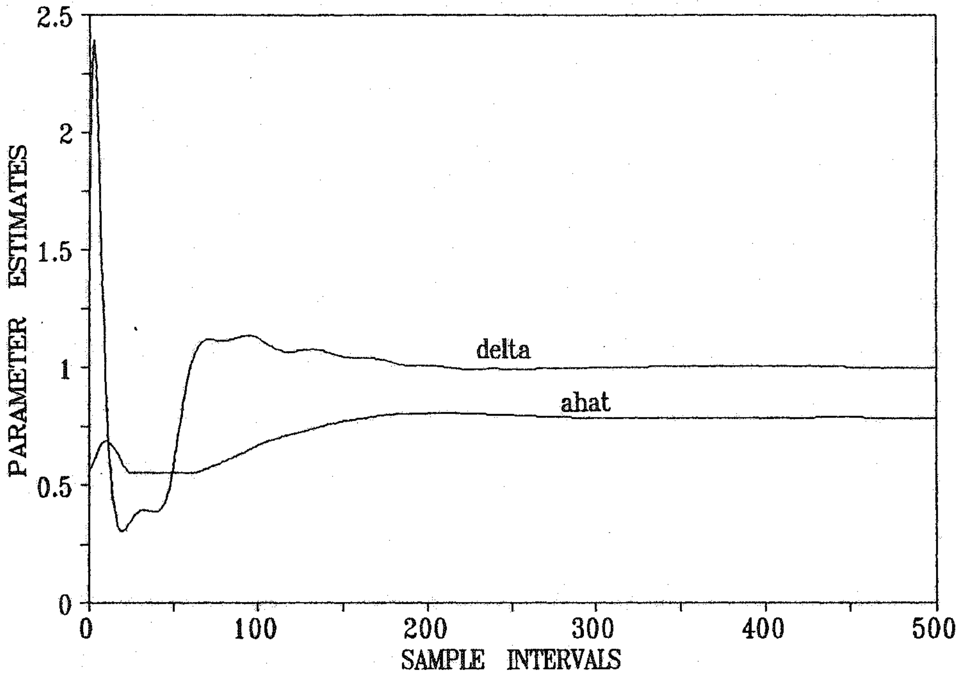

Figure 4.28 Parameter estimates $\hat{\delta}$ and $\hat{a}$ for input tracking case.

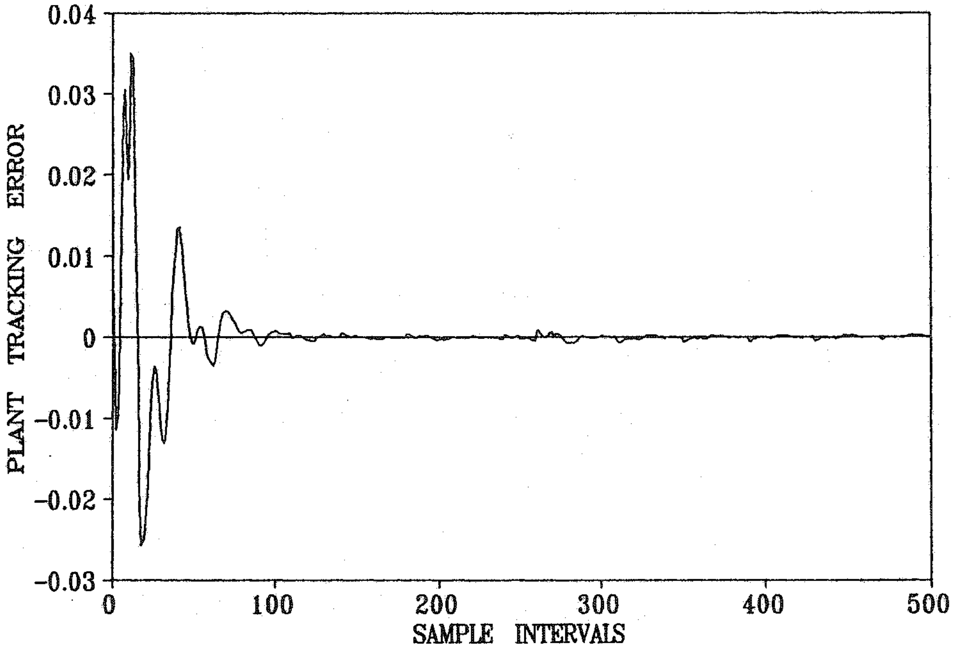

Figure 4.29 Plant tracking error. Two unknown parameters, **a** and **b**, estimated.

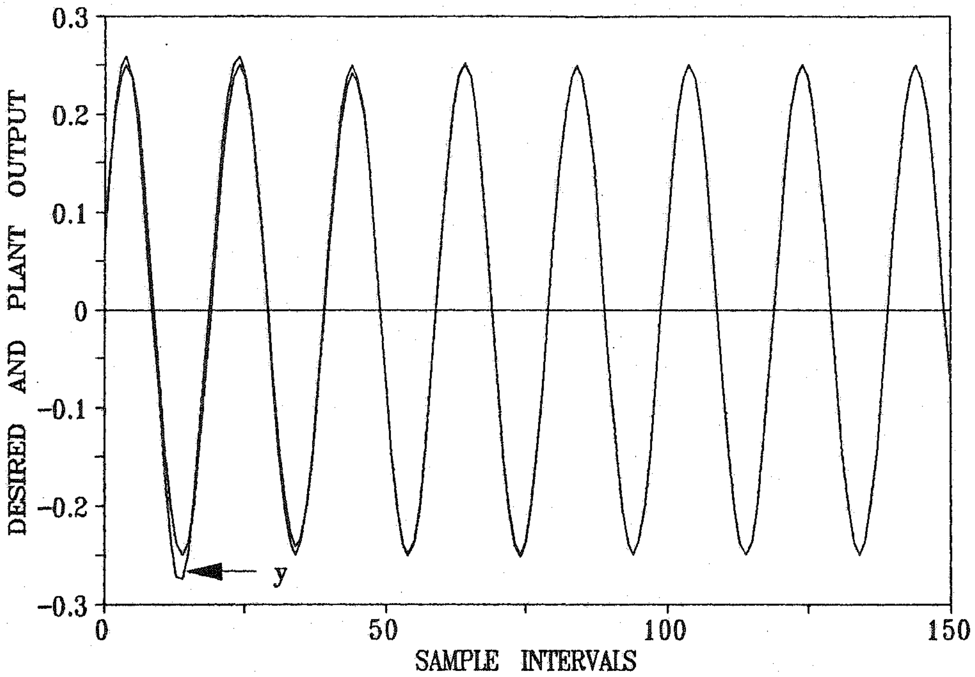

Figure 4.30 Comparison of plant output, y, with desired output; two unknown parameters estimated.

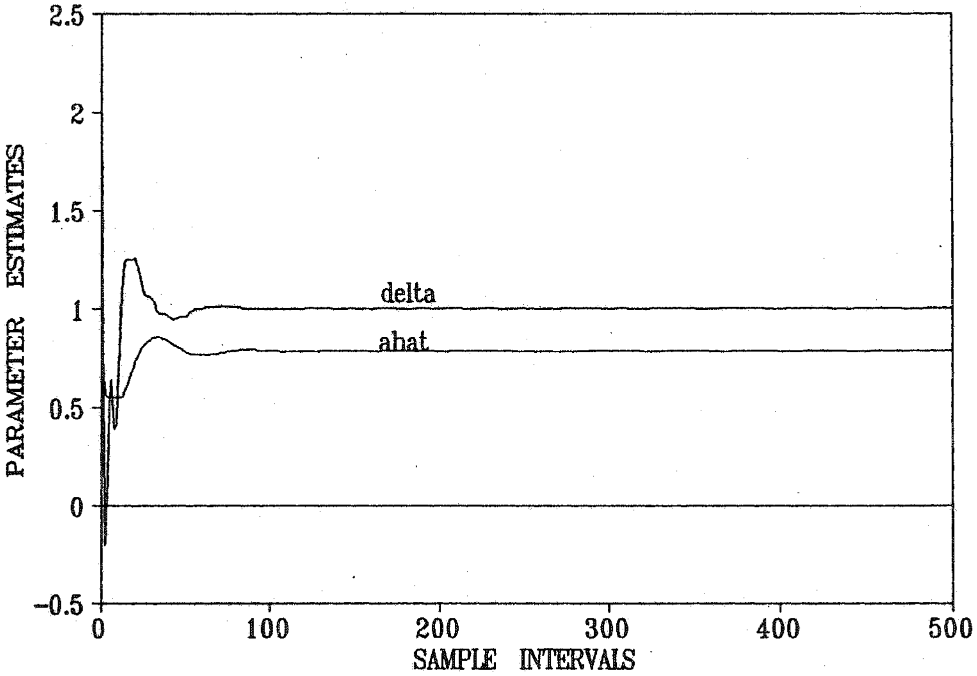

Figure 4.31a Parameter estimates $\hat{\delta}$ and $\hat{a}$ for input tracking case, two unknowns.

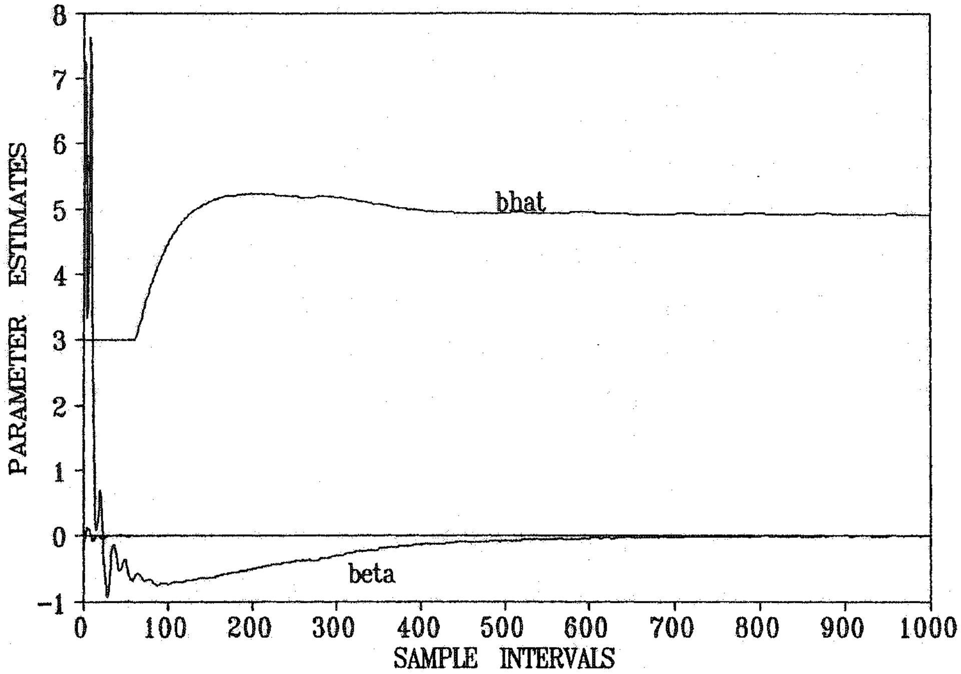

Figure 4.31b Parameter estimates $\hat{\beta}$ and $\hat{b}$ for input tracking case, two unknowns.

4.4 Summary and Conclusions

In this chapter an adaptive law was developed for use with the system architecture proposed in Chapter 3. The complete system is shown in Figure 4.1. In this configuration errors in those plant parameter estimates which are used in the control law result in additional inputs to the linear model. These inputs involve coefficients that are functions of the true plant parameters as well as the estimates, possibly in nonlinear combinations. Since the true nonlinear plant parameters appear in the partially linearized model, this provides an opportunity to estimate these parameters from the linear model. However the unknown parameters cannot be estimated directly. In order to solve this problem the linear model is reparameterized in terms of the parameter combinations which are then estimated by the RPEM. Since the final values (i.e. the values after perfect estimation has been achieved) of the parameters in the reparameterized linear model are known *a-priori*, an error equation for each can be written. The sign and magnitude of the errors are used to devise an adaptive law for updating estimates of the nonlinear plant parameters. Sufficient conditions defining the class of plants for which this method applies are given in section 4.2.2. The adaptive law is given by (4.2.36a). In section 4.2.3 the stability and convergence properties of the adaptive law were investigated. It is shown to be locally asymptotically stable and capable of one step convergence under ideal conditions.

Some important modelling considerations in connection with simulation studies of feedback linearizable systems are discussed. The most important requirement is to choose a plant modelling method that can track the apparent change in structure that the real plant exhibits as linearization is asymptotically approached. Both Runge-Kutta and Euler models are suggested for linearization studies because these solution methods utilize the real plant equations and will respond to the control law in the same manner as the real plant.

Several simulations were run to demonstrate the performance of the proposed architecture and adaptive law for both arbitrary inputs and signal tracking applications. The first simulation was a revisitation of the linear plant from Chapter 3 which had been used to point out problems with the combination of feedback linearization and standard self-tuning system architecture. The simulation demonstrated that the problems encountered in Chapter 3 had been eliminated using the new method. Following this, additional simulations using a nonlinear plant were carried out. The results from all simulation studies conducted confirm the efficacy of the combination of the new architecture and adaptive law for implementation of a self-tuning feedback linearization system.

CHAPTER 5
Self-tuning Feedback Linearization Applications

5.1 Introduction

It can be said with some justification that every process is nonlinear. Some are inherently nonlinear while others are nonlinear due to imperfections in components etc. From this hypothesis those which are considered to be linear are those that exhibit only slight nonlinear behavior and therefore can be successfully controlled using linear controllers. Controller design for these plants benefits from a large body of well developed and well known design techniques. Feedback linearization has attracted a great deal of research effort in recent years because it offers the possibility of making linear controller design techniques applicable to inherently nonlinear processes.

When complete knowledge of the parameters of an accurate plant model are available, design of the linearizing control law and nonlinear coordinate transformation is straight forward. However this is generally not the case and the designer is confronted with a structured uncertainty problem. In Chapters 3 and 4 the theory and a methodology for dealing with this problem was developed. A familiar technique for parameter estimation and the concept of self-tuning were combined in a certainty equivalence manner and the problems unique to applying these methodologies in a feedback linearization setting were solved. With the addition of a parameter adaptation law, a new method for designing self-tuning feedback linearized systems was proposed. The method was demonstrated using simulations of arbitrarily chosen linear and nonlinear plants. In this chapter self-tuning feedback linearization will be applied to simulations of real nonlinear plants.

5.2 Process Control Applications

Nonlinear plants are very common in the chemical processing industry. Frequently encountered nonlinearities are the ubiquitous Arrenhius dependency of reaction rates on temperature and the product of the plant input with the controlled variable such as flow rate times concentration etc. (i.e. bilinear plant).

A typical controlled variable in chemical processes is temperature. Temperature in a reactor is in general related to the manipulated variable through nonlinear terms in energy balance equations (Aoufoussi, et.al. (1992)). In this section we will study the temperature control of an exothermic batch reactor when controlled by a self-tuning feedback linearizing algorithm.

5.2.1 Self-tuning Feedback Linearization Control of An Exothermic Batch Reactor

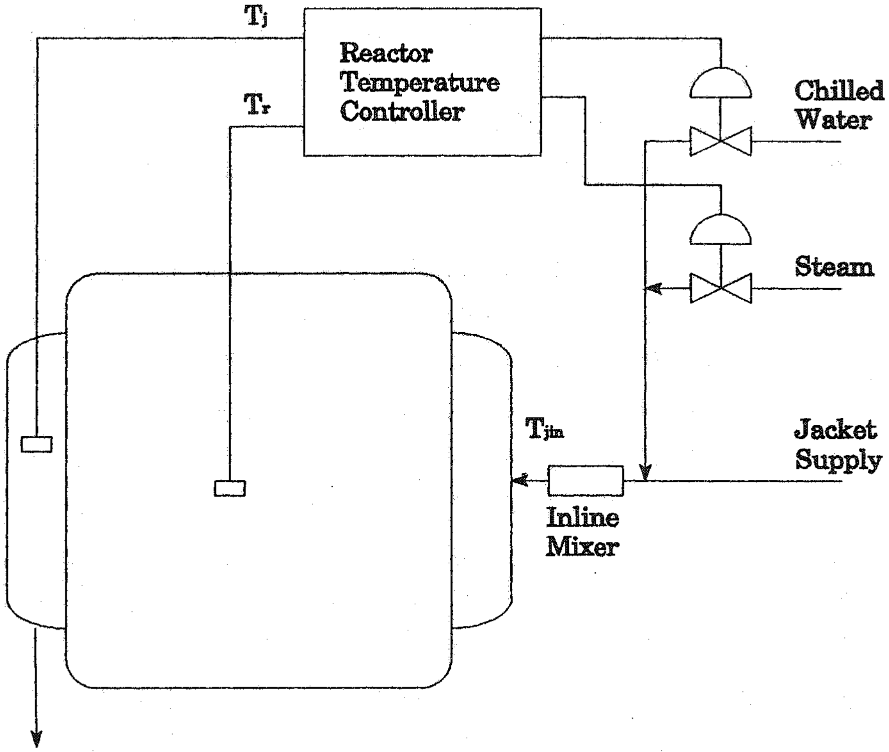

Figure 5.1 Batch reactor schematic diagram.

Figure 5.1 is a schematic diagram of a jacketed batch reactor. Temperature in the reactor is controlled indirectly by controlling the temperature of the fluid circulated through the jacket. Both the reactor and jacket temperatures are measured and brought into the reactor temperature controller. Output from the controller is directed to either a steam or chilled water valve. The steam or chilled water is injected directly into the jacket supply line ahead of an in-line mixer.

The reaction process used in this simulation is based on a dynamic model developed by Pulley (1986). Two well mixed liquid phase reactions are modelled:

$$\begin{aligned} A + B &\rightarrow C \\ A + C &\rightarrow D \end{aligned} \tag{5.2.1}$$

where component C is the desired product and D is an undesired by-product. The process is assumed to be reaction rate limited and strongly exothermic. Optimal conditions for reactor operation are an initial equimolar charge of reactants A and B and a reaction temperature in the range of 90 to 100 degrees centigrade. Physical limitations of the jacket system are assumed to restrict the jacket temperature range to from 20 to 120 degrees centigrade. The initial charge is assumed to be at 20 degrees C. The complete reactor model, process data, and initial conditions are given in Appendix A.

The main control objective is to raise the reactor temperature to the optimum value for conversion of the reactants to the desired product C while minimizing the production of by-product D. Obviously the optimum conditions have to be reached while staying within the physical constraints imposed by the processing equipment. Ideally the optimum reactor temperature is reached in a fairly short time and maintained at this temperature until the reaction reaches completion.

Exothermic reactions are positive feedback processes which have the potential for running away and possibly endangering personnel or damaging equipment. Therefore it is highly desirable that the control scheme be one which can estimate in real time the nonlinear heat of reaction being generated by the process. This information can then be used to make the appropriate compensating changes in the jacket temperature. Previous approaches to the control of this type of process are given in Shinskey and Weinstein (1965) and Cott and Macchietto (1989).

Shinskey and Weinstein proposed a dual mode scheme for heat up and maintenance of the temperature at the desired operating level. Heat up is done in a bang-bang mode where maximum heat is applied until the reactor temperature reaches a certain level below the operating setpoint. At this time a switch to maximum cooling is made to slow down the temperature rise and hopefully decrease the rate of rise to zero at the instant the temperature reaches the setpoint. Jacket temperature is then set at an intermediate value for a short time followed by a switch to linear control. PID control is then used to maintain the setpoint temperature. There are a total of seven tuning constants to be determined in this scheme for the reactor temperature plus some additional ones for the jacket temperature controller.

The drawback to this method is that the time durations of maximum cooling and intermediate heating are determined *a-priori* and the process is operating open loop until the final switch to PID is made. Any changes in process dynamics or disturbances entering the system will result in deviations from the optimum temperature profile.[1]

Cott and Macchietto propose a control scheme based on the Generic Model Control approach. (refer to Chapter 2). GMC eliminates the undesirable open loop heat up mode of the dual mode scheme and also requires only a single controller for heat up and control at the operating temperature. The GMC controller requires the on-line estimation of the heat of reaction. The authors derive this estimate by rearranging an energy balance equation written around the reactor contents. However this calculation involves the derivative of the reactor temperature which must be obtained by a numerical procedure from filtered measurements of the jacket and reactor temperatures.

The GMC method leads to a PI control structure and hence only two tuning parameters. These are calculated from performance figures given by Lee and Sullivan (1988).

In the following sections we will study the performance of the batch reactor with self-tuning feedback linearizing control applied. This method has none of the undesirable features of either the dual mode or GMC controllers.

[1] A solution to this problem was proposed by Gebo (1983). The process was the heat up of a polyethylene extruder barrel. Barrel temperature was periodically measured to ascertain that it was still on the desired trajectory. If not, then switching times from heating to cooling and vice-versa were recalculated on-line to insure that the temperature error trajectory would pass through the origin of the phase plane.

5.2.2 Control Law Development

Dynamic models for the reactor and jacket temperatures are derived from energy balances around the reactor contents and between the jacket fluid and reactor contents. These balances yield for the reactor temperature

$$\frac{dT_r}{dt} = -\frac{UA}{M_t C_{p_t}} T_r + \frac{UA}{M_t C_{p_t}} T_j + \frac{Q_r}{M_t C_{p_t}} \tag{5.2.2}$$

and for the jacket temperature

$$\frac{dT_j}{dt} = -\frac{F_j}{V_j} T_j - \frac{Q_j}{V_j \rho_j C_{p_j}} + \frac{F_j}{V_j} u \tag{5.2.3}$$

where u represents the jacket inlet temperature T_{jin}. Q_r is a nonlinear term in equation (5.2.2). The complete nomenclature is given in Appendix A. It is assumed that the steam and chilled water valve positioning time constants are negligible in comparison with the jacket temperature time constant. As a matter of convenience, equations (5.2.2) and (5.2.3) will be rewritten as

$$\begin{aligned} \dot{x}_1 &= -ax_1 + ax_2 + c \\ \dot{x}_2 &= -bx_2 - n + bu. \end{aligned} \tag{5.2.4}$$

The terms in (5.2.4) are defined as follows:

$$x_1 \doteq T_r, \quad x_2 \doteq T_j, \quad a \doteq \frac{UA}{M_t C_{p_t}}, \quad c \doteq \frac{Q_r}{M_t C_{p_t}}, \quad b \doteq \frac{F_j}{V_j}, \quad \mathit{and} \quad n \doteq \frac{Q_j}{V_j \rho_j C_{p_j}}.$$

In the following there is an economy of notation if (5.2.4) is expressed as

$$\dot{x} = f(x) + g(x)u \tag{5.2.5}$$

where

$$\dot{x} \doteq \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix}, \quad f(x) \doteq \begin{bmatrix} f_1(x) \\ f_2(x) \end{bmatrix}, \quad \textit{and} \quad g(x) \doteq \begin{bmatrix} 0 \\ b \end{bmatrix}.$$

Definitions of the vector elements are obvious by comparison with (5.2.4). Also the argument x will be dropped since the meaning will be clear in the sequel.

Application of theorem 2.4.1 indicates that the plant is feedback linearizable. The new coordinates

$$z_i = T_i \qquad i = 1,2$$

are found by evaluating

$$\begin{aligned} \left\langle dT_1, ad_f^0 g \right\rangle &= 0 \\ \left\langle dT_1, ad_f^1 g \right\rangle &\neq 0 . \end{aligned} \tag{5.2.6}$$

Carrying out the calculations indicated in (5.2.6) yields:

$$\begin{bmatrix} \dfrac{\partial T_1}{\partial x_1} & \dfrac{\partial T_1}{\partial x_2} \end{bmatrix} \begin{bmatrix} 0 \\ b \end{bmatrix} = 0$$

$$\therefore \frac{\partial T_1}{\partial x_2} = 0$$

$$\left[ad_f^1 g\right] = \nabla g f - \nabla f g = \begin{bmatrix} -ab \\ b^2 \end{bmatrix}$$

$$\begin{bmatrix} \frac{\partial T_1}{\partial x_1} & \frac{\partial T_1}{\partial x_2} \end{bmatrix} \begin{bmatrix} -ab \\ b^2 \end{bmatrix} = -ab\frac{\partial T_1}{\partial x_1} + b^2\frac{\partial T_1}{\partial x_2} \neq 0$$

$$\therefore \frac{\partial T_1}{\partial x_1} \neq 0. \tag{5.2.7}$$

To satisfy (5.2.7) we choose

$$T_1 = x_1 \doteq z_1 \tag{5.2.8}$$

which defines z_1 as the plant output. The second coordinate is found by evaluating

$$T_2 = \left\langle dT_1, f \right\rangle = \begin{bmatrix} \frac{\partial T_1}{\partial x_1} & \frac{\partial T_1}{\partial x_2} \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} = \frac{\partial T_1}{\partial x_1} f_1,$$

$$\therefore z_2 = -ax_1 + ax_2 + c. \tag{5.2.9}$$

Equation (5.2.8) and (5.2.9) represent the plant states in the new coordinate system. The feedback control law is calculated from

$$u = \frac{v - \langle dT_2, f\rangle}{\langle dT_2, g\rangle}$$

$$= \frac{v}{ab} - \frac{\dot{c}}{ab} + \frac{1}{b}(f_1 - f_2). \tag{5.2.10}$$

The unknown nonlinear heat of reaction term which has to be estimated on-line in order to achieve accurate feedback linearization is contained in f_1.

Usually the control law does not contain derivatives of the unknown parameters. In this case the unknown is the heat of reaction term which is a function of time. Differentiating the output, y_1, to obtain an equation involving the plant input introduces $\dot{c}$ into the linearizing control law.

5.2.3 Linearized Plant Model Development

From (5.2.8) and (5.2.9) the state-space model for the linearized plant is

$$\begin{aligned} \dot{z}_1 &= z_2 \\ \dot{z}_2 &= -af_1 + af_2 + \dot{c} + abu\,. \end{aligned} \qquad (5.2.11)$$

Substituting the control law (5.2.10) into $\dot{z}_2$, (5.2.11) becomes

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \hat{c}-c \end{bmatrix} a - \frac{d}{dt} \begin{bmatrix} 0 \\ \hat{c}-c \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} v. \qquad (5.2.12)$$

Equation (5.2.12) represents the open loop linearized plant model with three inputs. The first two are due to error in the estimate of the unknown parameter c and its rate of change, $\dot{c}$. As time evolves, the adaptive law would refine the estimates of c and $\dot{c}$ until these inputs eventually reduce to zero. The third input, v, is arbitrary and will be chosen to produce the desired closed loop response from the linearized system.

Each unknown in the control law increases the dimension of the parameter vector which in turn adds complexity to the RPE algorithm. Besides this, individual adaptive laws are needed for each estimated parameter. Therefore unless an unknown parameter will have a significant effect on the control objective it is desireable to leave it out of the control law.

Figure 5.2 is a plot showing the relative magnitude of c and $\dot{c}$. These curves were obtained when the system was operating closed loop but neither c nor $\dot{c}$ was included in the control law. The setpoint was tracing the desired reactor temperature profile.

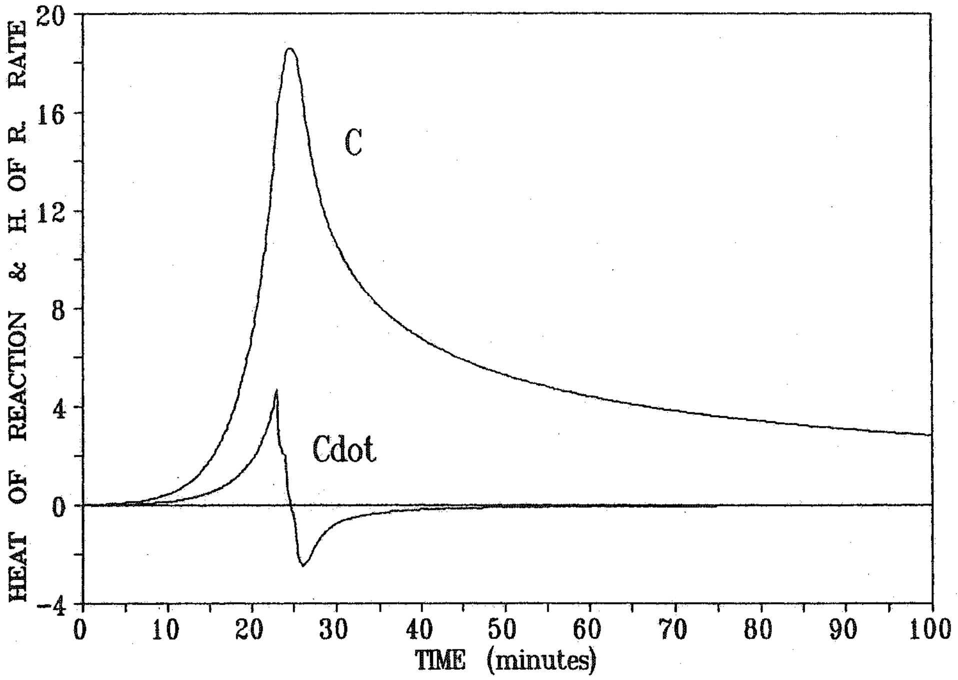

Figure 5.2 Heat of reaction response when compensation is not included in the control law.

The control objective is to track a temperature profile and it may be possible to do this satisfactorily even though the control law is sub-optimal. For these reasons it was decided that $\dot{c}$ would be ignored in the control law and linearized plant model. Subsequent testing and evaluation of the tracking error shows that ignoring $\dot{c}$ had substantially no effect on system performance.

The linearization procedure, by design, results in an open loop which is a cascaded string of integrators. The loop is closed by feeding back the output from each integrator. The closed loop system can be represented as

$$\sum_{i=1}^{3} d_i \frac{d^{i-1} z_1}{dt^{i-1}} = v_{ref}$$

where d_i are the feedback gains. With feedback added the reference input needed is

$$v_{ref} = \ddot{z}_1^d + d_2 \dot{z}_1^d + d_1 z_1^d \tag{5.2.13}$$

where z_1^d is the desired reactor temperature profile. In this example the objective is to ramp the reactor temperature from an initial value of 20 degrees to a final value of 95 degrees C in 25 minutes and hold at 95 degrees. Based on the above conditions and (5.2.13),

$$v_{ref} = 0 + 3d_2 + 3d_1 t + 20d_1, \qquad 0 < t < 25 \text{ min} \tag{5.2.14}$$

$$v_{ref} = 95d_1, \qquad 25 \le t < t_f \tag{5.2.15}$$

where $20d_1$ is due to the initial temperature. The input signal to the control law is

$$\begin{aligned} v &= v_{ref} - d_1 z_1 - d_2 z_2 \\ &= d_1(z_1^d - z_1) + d_2\left(\dot{z}_1^d - \dot{z}_1\right). \end{aligned} \tag{5.2.16}$$

The error equation for the closed loop system can be written as

$$\ddot{e} + d_2\dot{e} + d_1 e = 0 \tag{5.2.17}$$

where $e \doteq z_1^d - z_1$. The error equation provides a basis for choosing the feedback gains. Parameters d_1 and d_2 determine the rate at which the tracking error approaches zero. Of course the input due to error in the parameter estimate $\hat{c}$ will also have a transient effect on tracking error. The feedback gains were set at $d_1 = 16$ and $d_2 = 8$. This produces a dominant decaying exponential factor with a 0.25 minute time constant in the tracking error response.

Updating the model given by (5.2.12) to include feedback, reparameterizing such that $\hat{\theta} = \hat{c} - c$, and converting to discrete form yields for a sampling time of $\Delta T = .025$ min.

$$\begin{bmatrix} z_1(k+1) \\ z_2(k+1) \end{bmatrix} = \begin{bmatrix} .99532 & .02262 \\ -.36193 & .81435 \end{bmatrix} \begin{bmatrix} z_1(k) \\ z_2(k) \end{bmatrix} - \begin{bmatrix} .0000876 \\ .0067863 \end{bmatrix} m\hat{\theta}$$

$$+ \begin{bmatrix} .000292 \\ .022621 \end{bmatrix} v_{ref} \tag{5.2.18}$$

where parameter $a=0.3$ has been included in the coefficient matrix of $\hat{\theta}$. The reason for the multiplying factor m will be discussed in the next section. Equation (5.2.18) is the plant model used in the RPEM algorithm.

5.2.4 Estimating the Time-varying Heat of Reaction Term

The input signal v_{ref}, calculated from (5.2.14,15), has a minimum value of 344 at $t = 0$ and increases to 1520 when $t \geq 25$ minutes. The true value of parameter c starts at zero and remains small never exceeding *12* in magnitude. It is therefore reasonable to assume that the upper bound on $\hat{\theta}$ is also about *12*. The relative magnitude of v_{ref} and $\hat{\theta}$, coupled with the fact that their coefficient matrices are roughly in the ratio of 3.3 : 1 with the v_{ref} coefficient being the larger, means that the $\hat{\theta}$ term in (5.2.18) has very little effect on the output of the linearized model. Therefore insufficient information exists to track this rapidly changing parameter.

This ill-conditioning can be corrected by choosing an arbitrary multiplier, m, for the $\hat{\theta}$ coefficient matrix. This approach can be justified on the basis that as $\hat{c}$ converges to c, $\hat{\theta}$ approaches zero thereby eliminating this term from the plant model. If the multiplier chosen is too small $\hat{c}$ estimates will still be poor or impossible to obtain. On the other hand beyond a certain value no improvement in the performance of the estimator is realized. A few trials allows one to quickly zero in on a suitable multiplier value. In the present case a value of *300* gave satisfactory results.

The situation just described is probably not unusual. It comes about because of the a parameter being small and because the true value of the unknown parameter c is initially zero and remains small for a significant period of time.

(see Figure 5.2) It is conceivable that in some cases the reverse situation could occur. That is a plant parameter which multiplies the $\hat{\theta}$ coefficient matrix is large and the unknown c parameter, implicit in $\hat{\theta}$, is also large. This could cause the v_{ref} plant input term to be overpowered with unpredictable results. In this case choosing a suitably small multiplier would correct the problem.

The procedure for obtaining the gradient transition equation is given in Chapter 3, equations (3.2.19) through (3.2.25). Applying this procedure to (5.2.18), taking into account that (5.2.18) is not an innovations model, we obtain the following:

$$M_k = \frac{\partial}{\partial\hat{\theta}}\left[\phi z - \alpha m\hat{\theta} + \gamma V_{ref}\right] = -\alpha m \tag{5.2.19}$$

where ϕ is the discrete state transition matrix and α and γ are coefficient vectors in (5.2.18). From (3.2.23)

$$W(k+1) = \phi W(k) + M_k . \tag{5.2.20}$$

Expanding (5.2.20) yields

$$W_{11}(k+1) = .99532W_{11}(k) + .02262W_{21}(k) - .02628 \tag{5.2.21}$$

$$W_{21}(k+1) = -.36193W_{11}(k) + .81435W_{21}(k) - 2.0359 . \tag{5.2.22}$$

Applying (3.2.22) the prediction error gradient is

$$\psi(k+1) = \left[W_{11}(k+1),\, W_{21}(k+1)\right] \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad (5.2.23)$$

$$= W_{11}(k+1)\,.$$

The adaptive law for this example is

$$\hat{c}(k) = \hat{c}(k-1) - \mu_I\, e_{\hat{c}}(k)\,. \qquad (5.2.24)$$

5.2.5 Optional Modifications For Enhanced Performance

We will consider two modifications which provide slightly better control of the reactor temperature. Neither are necessary but if a small overshoot or some low amplitude ringing around the switching point from ramp to hold is important, then these modifications will help to eliminate those conditions.

5.2.5.1 Adaptive Law Proportional And Derivative Response

There are two types of estimation problems which can occur when applying feedback linearization. One is structured uncertainty in the plant which leads to unknown but time invariant parameters in the control law. (e.g. see (4.3.22)). For estimation of constant parameters the adaptive law initial gain and denominator constant can be chosen so that the estimate converges asymptotically to the true value. (see Figure 4.17).

The second case involves the estimation of nonlinear terms. In the batch reactor control law we have the nonlinear heat of reaction in combination with two other process variables. This combination was lumped together in the model and considered as a single parameter, c. However the difference in this case is that c is time-varying. (see Figure 5.9). The rise is exponential until the hold temperature is reached and then the slope reverses and the curve descends but at a much slower rate. In order for the estimate to track the high rate of change over the first part of the curve a fast integration rate (i.e. high initial integrator gain) is needed in the adaptive law. However this rate is too fast for the descending portion of the curve and causes some oscillation in the estimated value which shows up, considerably damped, in the reactor temperature. Reducing the integration rate prevents the oscillation but at the expense of accuracy over the rising portion of the response. This is where the heat of reaction has the greatest effect on reactor temperature and therefore where it is desirable to have the highest accuracy in the estimate.

Adding derivative action to the adaptive law helps to alleviate this situation and allows the use of a higher integration rate. When c is rapidly increasing the derivative action adds an incremental amount to the adaptive law. More importantly it subtracts an incremental amount when the slope of the curve reverses. In other words the response rate is in effect increased or reduced over the portion of the curve where those two actions are beneficial. However the accuracy of the estimate can be improved even more if a proportional response is also added.

Figure 5.3 is a plot of c and $\hat{c}$ without proportional and derivative response in the adaptive law. The integrator gain was set to the highest value possible which would not cause large overshoots and considerable ringing on the decreasing portion of the curve. Comparing Figure 5.3 with Figure 5.9 shows the improvement in estimation accuracy with proportional and derivative responses added to the adaptive law.

The heat of reaction term is difficult to track because of the rapid slope reversal near the hold temperature. An alternative representation of the Q_r term that might have been easier to track is investigated in Appendix B.

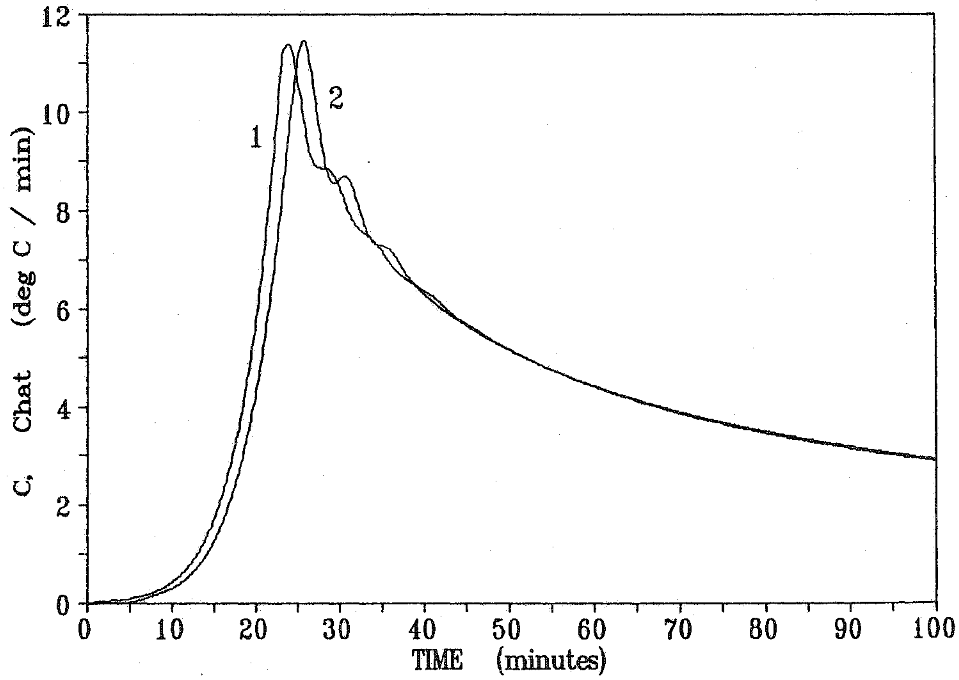

Figure 5.3 Heat of reaction estimate. Integral only adaptive law. curve 1, c; curve 2, ĉ.

Figure 5.4 is a plot of the temperature tracking error both with and without derivative response in the adaptive law. Derivative action has the greatest effect after the setpoint ramp has reached the hold temperature. Ringing around this point is considerably damped as compared with no derivative action and there is virtually no overshoot. The adaptive law with the two additional response terms is

$$\begin{aligned} I(k) &= I(k-1) - \mu_I e_{\hat{c}}(k) \\ \hat{c}(k) &= I(k) - \mu_P e_{\hat{c}}(k) - \mu_D\big(e_{\hat{c}}(k) - e_{\hat{c}}(k-1)\big) \end{aligned} \qquad (5.2.25)$$

where the $\mu_{i's}$ represent the integral, proportional, and derivative gains respectively. The sign of the derivative and proportional terms are determined by the same

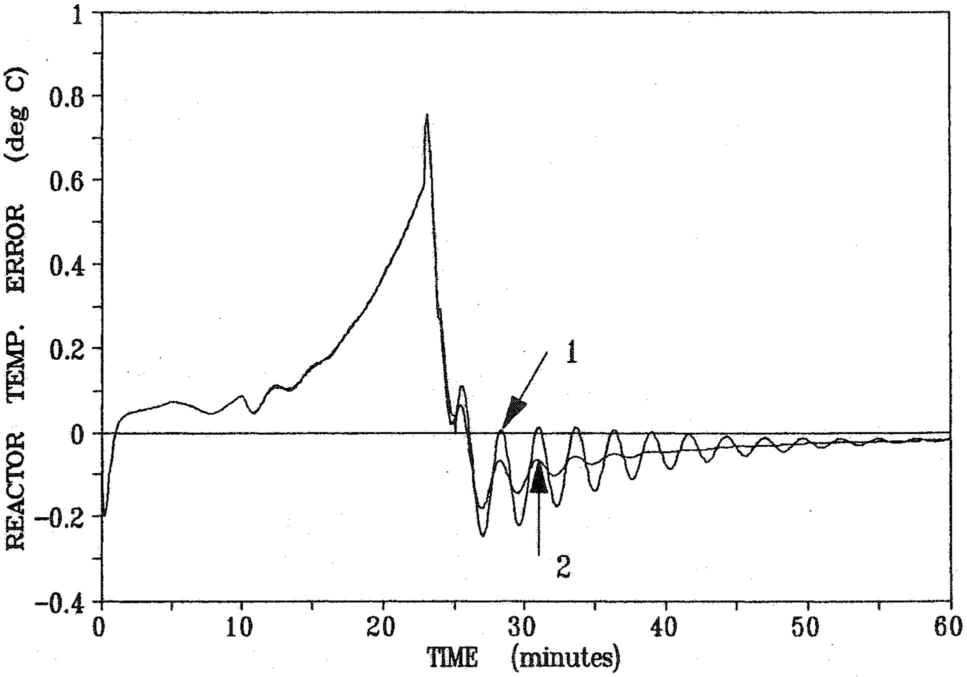

Figure 5.4 Reactor temperature error. curve 1, without derivative action; curve 2, with derivative action.

procedure that was used in Chapter 4 to determine the sign of the integral term.

5.2.5.1 Ramp Rate Reduction

If the reference signal is ramped at a constant rate until the 95 degree hold level is reached the temperature in the reactor may overshoot by some small amount. Since this could be undesirable in some applications, the ramp rate can be reduced before reaching the hold temperature to prevent this from happening.

For the present application the ramp up time was held constant at 25 minutes. The ramp rate was reduced by a factor of 2 in each of two steps starting 2.5 degrees below the hold temperature. The time duration of each of the steps was chosen to be equal. Figure 5.5 is a sketch illustrating the modified reactor temperature profile in the vicinity of the hold temperature. Since the ramp rate is reduced over a portion of the ramp period, the initial ramp rate has to be increased slightly in order to stay within the 25 minute ramp up period.

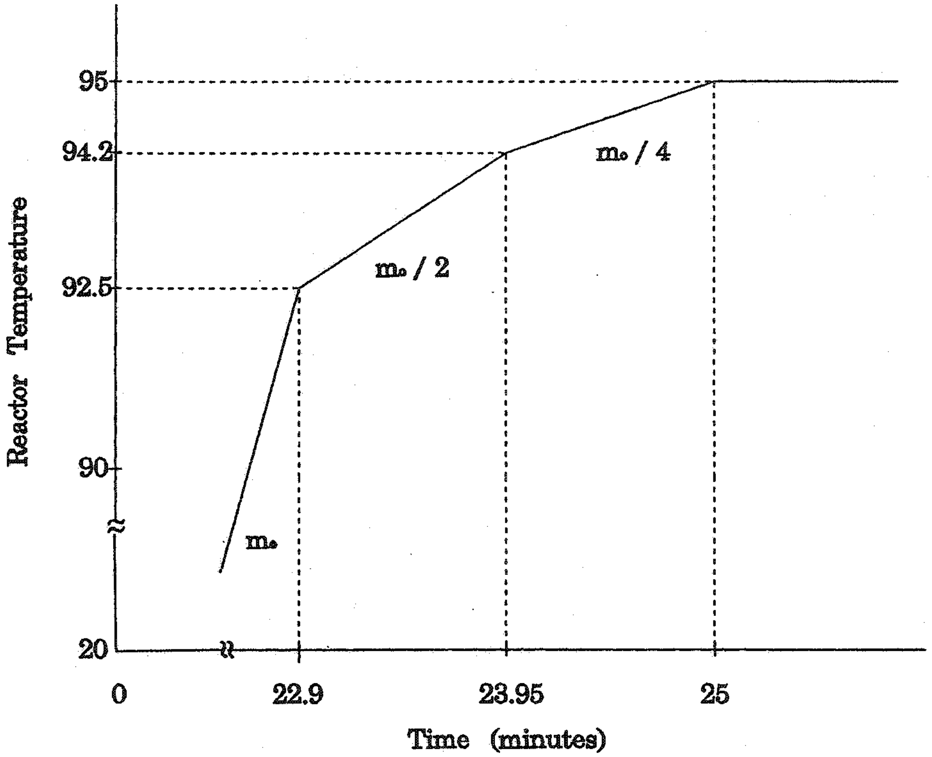

Figure 5.5 Sketch of the modified reactor temperature profile.

Let: $m_0 \doteq$ the initial ramp rate

T = ramp up time (i.e. 25 minutes)

$\Delta t \doteq$ the elapsed time at rates $\frac{m_0}{2}$ and $\frac{m_0}{4}$

then

$$m_0(T - 2\Delta t) = 72.5$$

$$\frac{m_0}{2}\Delta t + \frac{m_0}{4}\Delta t = 2.50$$

from which

$$m_0 = 3.1667 \quad °C/\text{min.}$$

$$\frac{m_0}{2} = 1.5833 \quad °C/\text{min.}$$

$$\frac{m_0}{4} = .7917 \quad °C/\text{min.}$$

$$\Delta t = 1.053 \quad \text{min.}$$

Therefore the reference signal given in (5.2.14) is modified as follows:

$$V_{ref} = 3.1667d_2 + 3.1667d_1t + 20d_1 \qquad 0 < t < 22.89 \text{ min}$$

$$V_{ref} = 1.5833d_2 + 1.5833d_1t + 92.5d_1 \qquad 22.89 \le t < 23.94 \text{ min}$$

$$V_{ref} = 0.7917d_2 + 0.7917d_1t + 94.17d_1 \qquad 23.94 \le t < 25 \text{ min}$$

For $t \geq 25$ minutes V_{ref} is given by (5.2.15).

5.2.6 Simulation Results

Figures 5.6 through 5.13 illustrate various aspects of the reactor performance under self-tuning feedback linearizing control. A comparison is made between the cases when the nonlinear heat of reaction term is ignored in the control law and coordinate transformation and when it is estimated on-line and included in both. It should be noted that the reactor is still under closed loop control even when the heat of reaction term is not taken into account. To differentiate between the two cases the self-tuning mode will be referred to as compensated meaning that the heat of reaction term was estimated on-line and included in the control law and transformed state equations. Uncompensated will indicate that it has been set to zero in both. The optional modifications described in section 5.2.5 were included in the simulation.

Figures 5.6 and 5.7 shows the desired temperature profile, actual reactor temperature, and jacket temperature for the uncompensated and compensated cases respectively. Without compensating the heat of reaction term the reactor temperature overshoots and remains above the setpoint for the entire length of the run. In the compensated case shown in Figure 5.7 the desired profile is followed with high fidelity throughout and the variation in the jacket temperature is considerably reduced.

Figure 5.8 is a plot of the reactor temperature tracking error. (i.e. $e = T_r - T_d$). In the compensated case the rise in tracking error prior to the transition from ramp to hold is due in part to error in the heat of reaction estimate and to a small degree in the reduction of the ramp rate just prior to the transition from ramp to hold. The error is obviously much larger over the entire operating range for the uncompensated case.

Figure 5.9 shows the comparison between the true and estimated heat of reaction term. Since this term can never be negative, a lower bound of *0.001* was set on the estimate. Figure 5.10 depicts the heat of reaction term for the uncompensated case.

Figures 5.11 is a plot of the heat supplied by the jacket, the heat of reaction, and the sum of the two for the compensated case. Since heat loss to the surroundings has been assumed negligible, the sum of the heats is constant while the reactor temperature is increasing at a constant rate. When the hold temperature is reached the net heat supplied drops to zero consistent with the negligible loss assumption. Figure 5.12 is a plot of the same quantities as shown in Figure 5.11 but for the uncompensated case.

Figures 5.13 is a plot of the reactant variations as the reaction evolves for the compensated case.

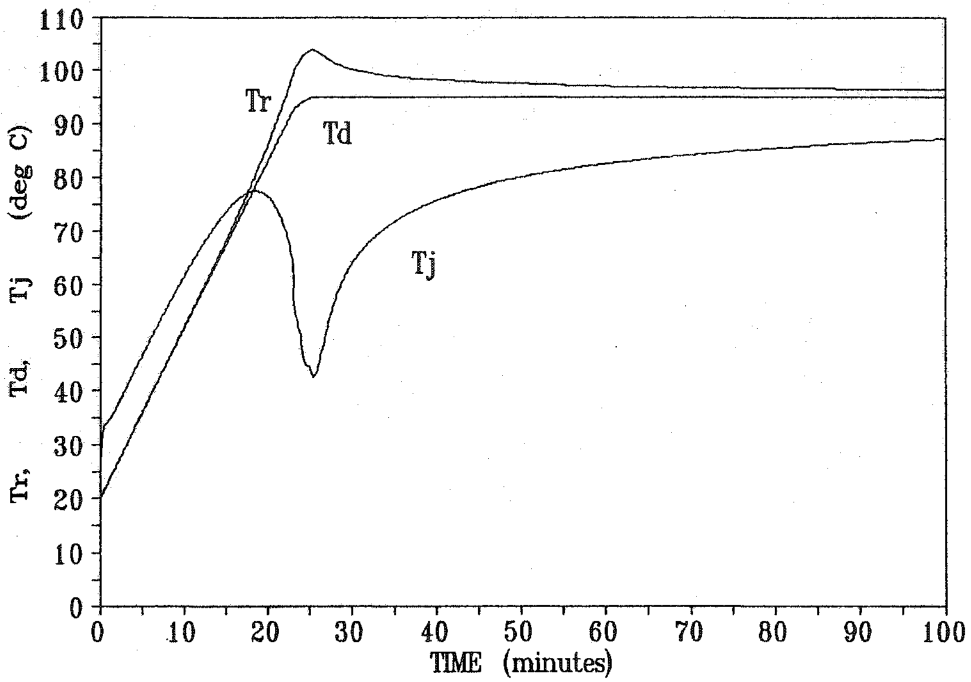

Figure 5.6 Reactor and jacket temperature response, uncompensated case.

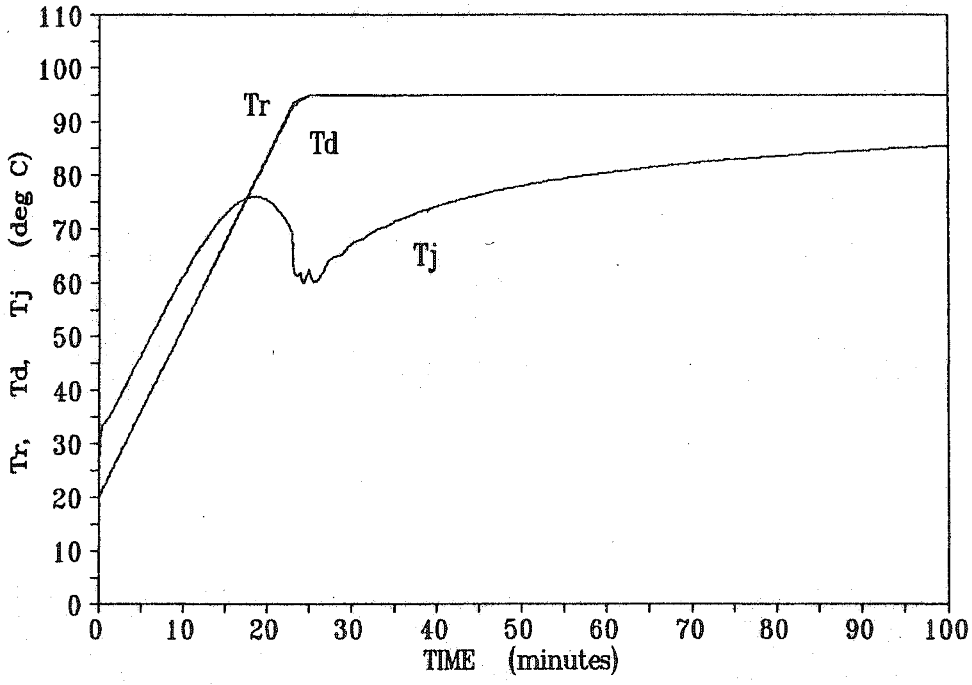

Figure 5.7 Reactor and jacket temperature response, compensated case.

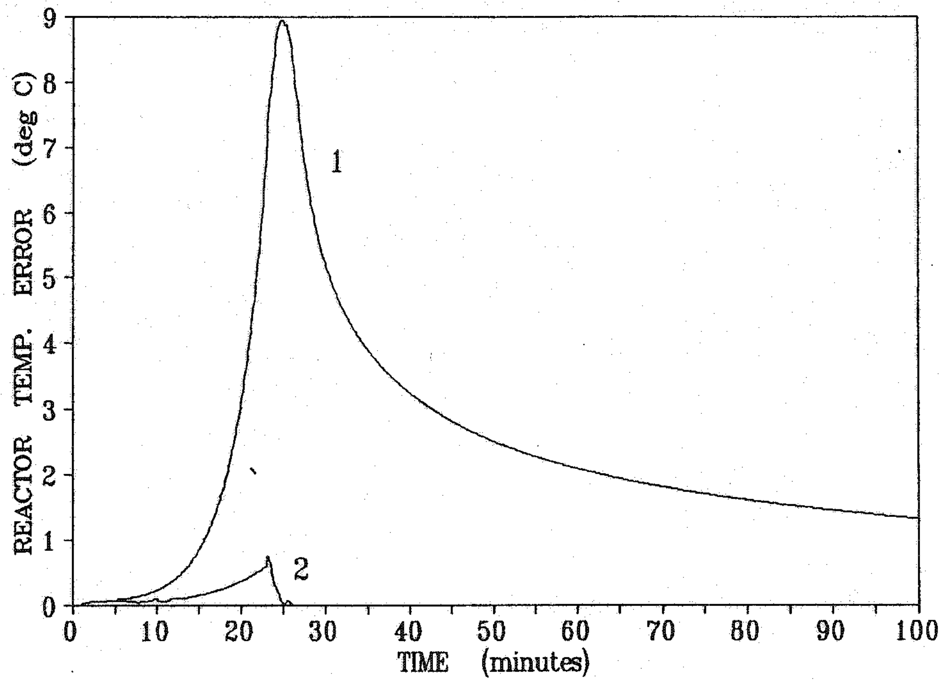

Figure 5.8 Reactor temperature error: 1=uncompensated, 2=compensated.

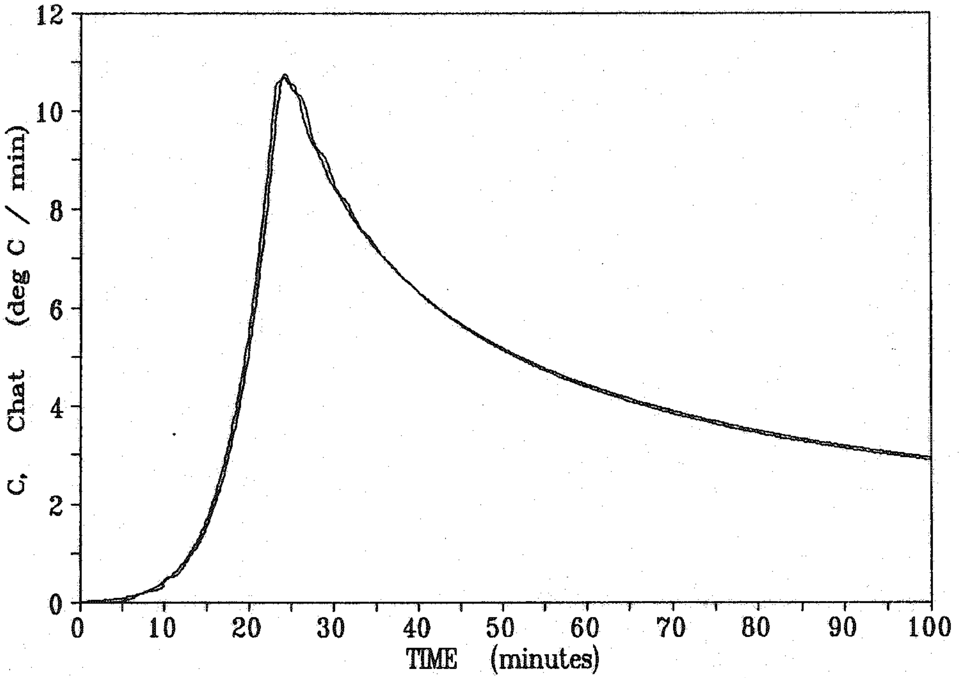

Figure 5.9 Heat of reaction term c and estimate $\hat{c}$.

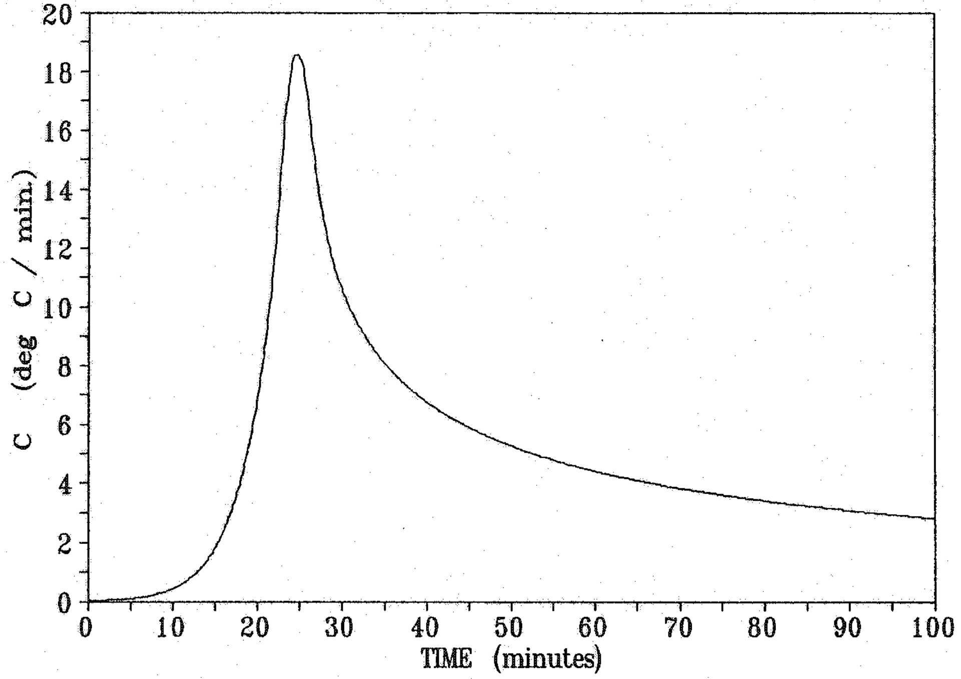

Figure 5.10 Heat of reaction for uncompensated case.

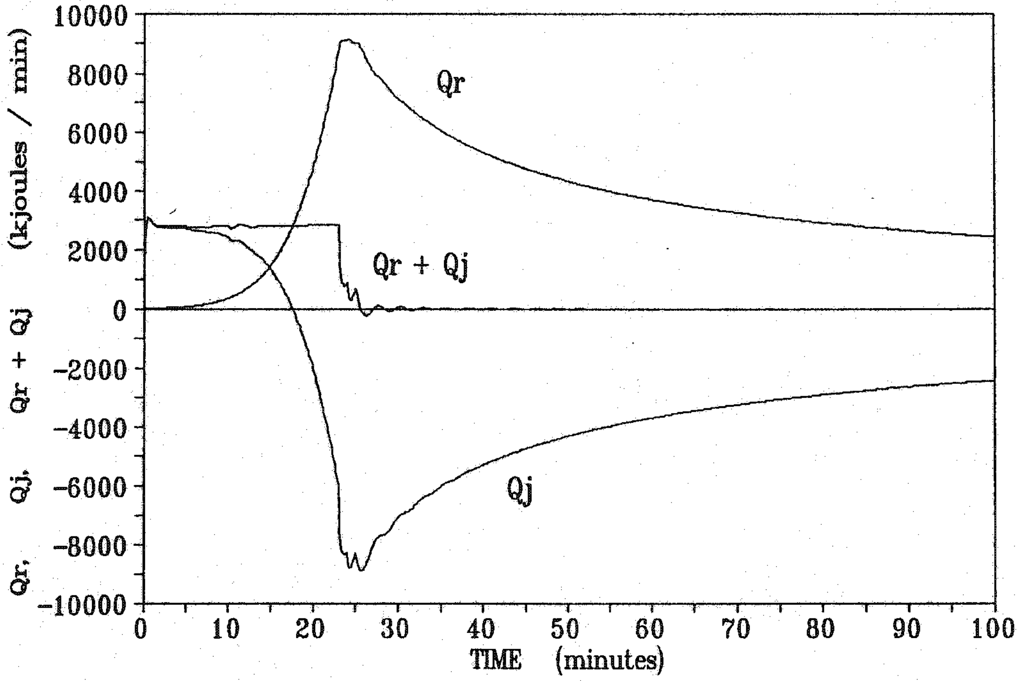

Figure 5.11 Heat supplied to the reactor: Q_j from jacket, Q_r from heat of reaction. Compensated case.

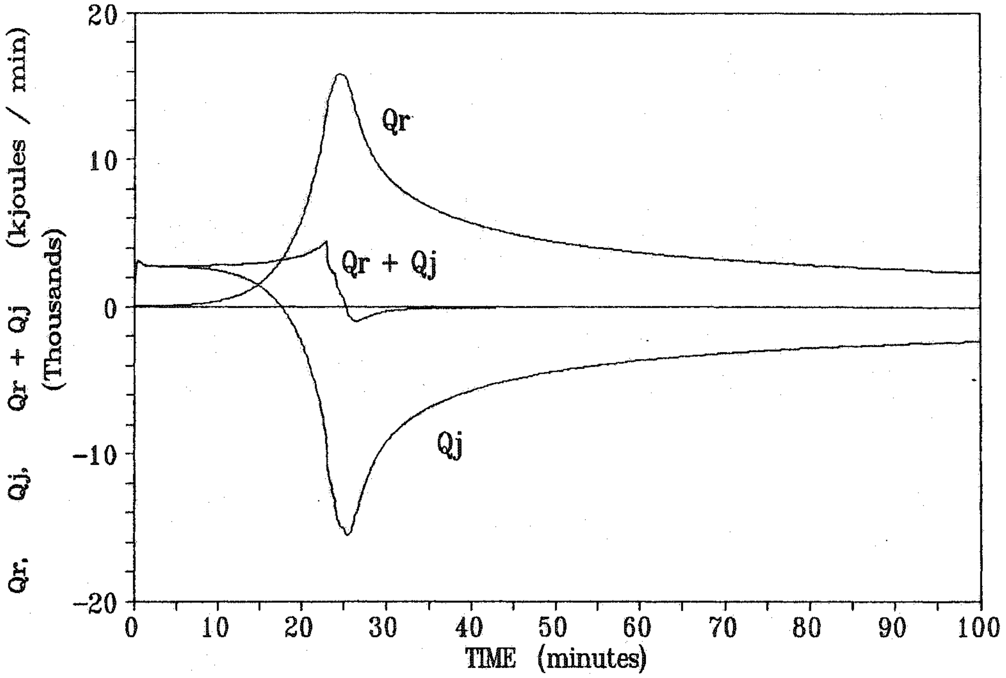

Figure 5.12 Heat supplied to the reactor: Q_j from jacket, Q_r from heat of reaction. Uncompensated case.

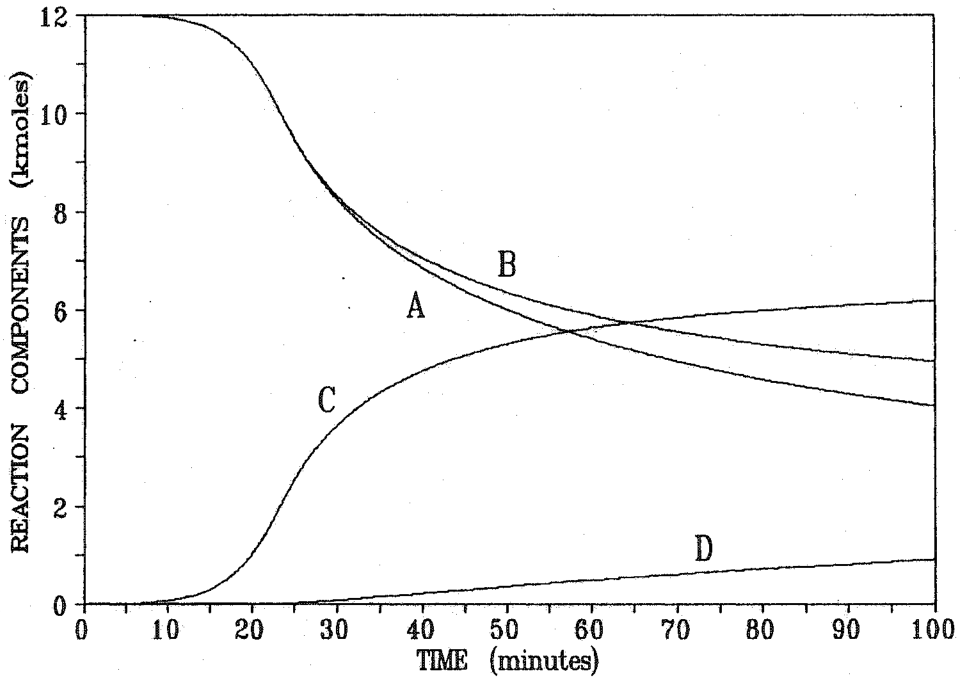

Figure 5.13 Reaction response curves. A and B are reactants, C is the product, and D is a by-product.

5.2.7 Bioreactor Control

Fermentations are an important class of nonlinear processes (Bailey and Ollis (1986)). They have numerous applications in food processing, waste water treatment, synthesis of products such as antibiotics and pesticides, etc. The presence of living organisms in these processes makes the dynamics strongly nonlinear and nonstationary (Bastin and Dochain (1990), Zhang et. al. (2000)). In this section we will apply self-tuning feedback linearization to control the substrate concentration in a bioreactor.

5.2.8 Fermentation Process Model

There are three basic methods of bioreactor operation, batch, fed-batch, and continuous flow. The batch reactor is initially charged and no addition or removal of material occurs until a predetermined quantity of substrate has been consumed thereby signaling the end of the fermentation. In fed-batch mode the reactor is initially charged with a small amount of biomass and substrate after which substrate continues to be added until the reactor is filled. There is no outflow from the reactor during this time. The third mode, and the one which we will study, is continuous feed. In this mode the outflow equals the inflow after the reactor is filled and the process is ongoing.

Dochain (1986) gives a basic model for the continuous (CSTR) mode of operation as follows.

$$\frac{dx}{dt} = \mu x - Dx \tag{5.2.26a}$$

$$\frac{ds}{dt} = -k_1\mu x + D(s_{in} - s) \tag{5.2.26b}$$

$$Q = k_2\mu x. \tag{5.2.26c}$$

In this model:

x represents the biomass concentration in the reactor

μ is the specific growth rate

$D \doteq F_{in}/V$ is the dilution rate

F_{in} is the influent flow rate

V is the reactor volume

s represents the substrate concentration in the reactor

s_{in} represents the influent substrate concentration

Q represents a production rate per unit volume (e.g. methane gas)

k_1, k_2 are yield coefficients.

There are many different models in use for the specific growth rate. One of the most common is the Monod model

$$\mu = \frac{\mu^* s}{K_m + s} \tag{5.2.27}$$

which shows the dependence of the growth rate on the substrate concentration. In this model μ^* is the maximum growth rate and K_m is the Michaelis-Menten parameter. Specific growth rate is also influenced by biomass concentration and many other chemical and environmental factors in the reactor. In some cases the substrate concentration has an inhibiting effect on the growth rate. This effect is taken into account by the Haldane model

$$\mu = \frac{\mu_0 s}{\frac{s^2}{K_i} + s + K_m} \tag{5.2.28}$$

where K_i is the inhibition parameter and

$$\mu_0 = \frac{\mu^*}{1 + 2\sqrt{\frac{K_m}{K_i}}} \; . \tag{5.2.29}$$

Microorganism growth in the reactor may result in a product which is soluble in the reactor contents or given off as a gas. Equation (5.2.26c) represents product formation when it is assumed to be proportional to the biomass growth rate.

5.2.9 Substrate Control In An Anaerobic Digestion Process

In this application the objective is to control the effluent substrate concentration of a waste water treatment process. The anaerobic digestion processing of organic waste produces methane gas. Equations (5.2.26b and c) can be used to model this process. In this model s_{in} and s represents the influent substrate concentration and the effluent substrate concentration respectively while

Q is the methane production rate. Variations in s_{in} are disturbances coming into the process and the objective is to eliminate these disturbances in the effluent by manipulating the dilution rate, D. The variables s_{in}, s, and Q are assumed to be measurable. The yield coefficients k_1, and k_2 are unknown.

By taking advantage of the fact that Q is a measured quantity the model can be reduced to a single nonlinear differential equation. Equation (5.2.26c) is solved for μx and this is substituted into (5.2.26b) to yield

$$\dot{s} = -KQ + D(s_{in} - s) \qquad (5.2.30)$$

where $K = k_1 / k_2$.

5.2.10 Control Law Development

The plant model (5.2.30) obviously represents a feedback linearizable process. The control law is easily found from (5.2.30) to be

$$D = \frac{V + \hat{K}Q}{s_{in} - s} \qquad (5.2.31)$$

where V is an arbitrary input. V is defined by choosing a first order closed loop response for the plant having time constant τ and setting the effluent substrate concentration setpoint at some desired level, say s^*. Then

$$V = s^* - ds \qquad (5.2.32)$$

where $d = 1/\tau$. $\hat{K}$ will be estimated on-line.

5.2.11 Linearized Plant Model, Parameter Estimation, And Control Configuration

As in the previous example, the control scheme developed in Chapters 3 and 4 will be applied to this plant. The system configuration is shown in Figure 5.14.

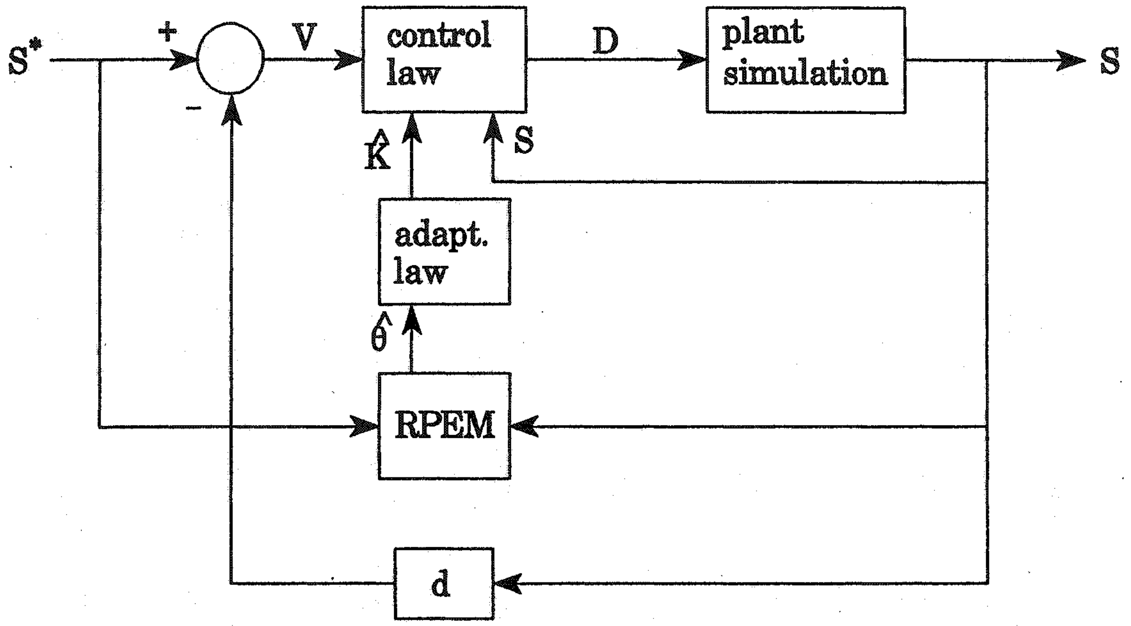

Figure 5.14 Anaerobic digestion process substrate control.

The linearized plant model needed for the RPEM algorithm is derived by substituting the control law (5.2.31) into (5.2.30) and replacing input *V* by (5.2.32). This yields

$$\dot{z} = -dz + \hat{\theta}Q + s^{*} \tag{5.2.33}$$

where $\hat{\theta} = \hat{K} - K$.

The discrete solution of (5.2.33) is

$$z(t+1) = \phi z(t) + \Gamma Q \hat{\theta} + \Gamma s^{*} \tag{5.2.34}$$

where $\phi = e^{-dh}$, h is the sampling interval, and

$$\Gamma = \frac{1}{d}\left(1 - e^{-dh}\right).$$

The negative gradient of the prediction error is from (5.2.34)

$$\psi(t+1) = \phi\psi(t) + \Gamma Q. \tag{5.2.35}$$

Equations (5.2.34) and (5.2.35) are the two model related equations needed for the RPEM algorithm. From (4.2.36a) the adaptive law is

$$\hat{K}(t) = \hat{K}(t-1) - \mu_{\hat{K}} e_{\hat{K}}(t). \tag{5.2.36}$$

Choice of the adaptive law gain parameters will be discussed later when simulation results are reviewed.

5.2.12 Simulation Results

The real plant is assumed to be represented by (5.2.26a,b,c) and the Haldane growth model given by (5.2.28). Following the simulation given in Dochain (1986), the model parameters were taken as:

$$k_1 = 27.27, \ k_2 = 75, \ K_m = 0.4, \ K_i = 2.5, \ \mu^* = 0.4$$

The initial conditions are:

$$x(0) = 0.049, \ s(0) = 0.5, \ D(0) = 0.2, \ s_{in}(0) = 3.0$$

$$s_d = 1.66, \quad \hat{K} = 0.2, \quad D_{mx} = 0.5$$

Feedback gain was set to one. (i.e. $d = 1$ in Figure 5.14).

For any real process there are finite limits on the manipulated variable. In this case the manipulated variable is a flow and so the obvious lower bound is zero. The upper bound is set at *0.5*. The disturbance in influent substrate concentration is varied from *3* to *5* in step fashion with a period of *48* hours.

Figure 5.15 shows that with the self-tuning nonlinear controller the output concentration is well maintained at the setpoint value of 1.66 in spite of the strong disturbances in the influent concentration. Control action stays well within the desired limits of $0 \leq u \leq 0.5$ except for instantaneous spikes coinciding with points of maximum change in the input disturbance Figure 5.16 shows the rapid convergence of the estimated parameter, $\hat{K}$, to the true value.

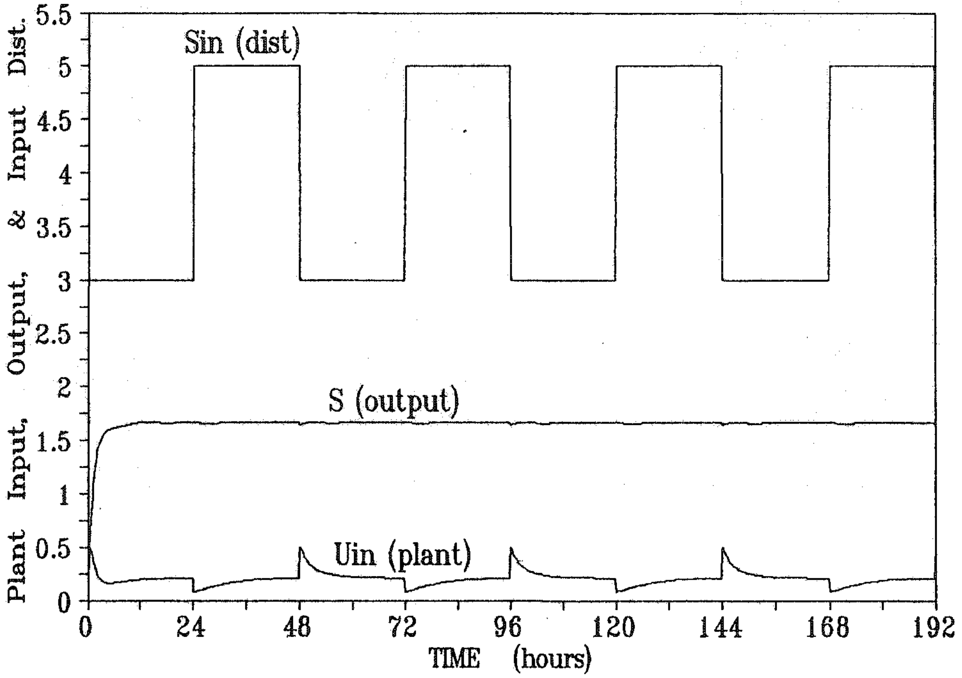

Figure 5.15 Output concentration control.

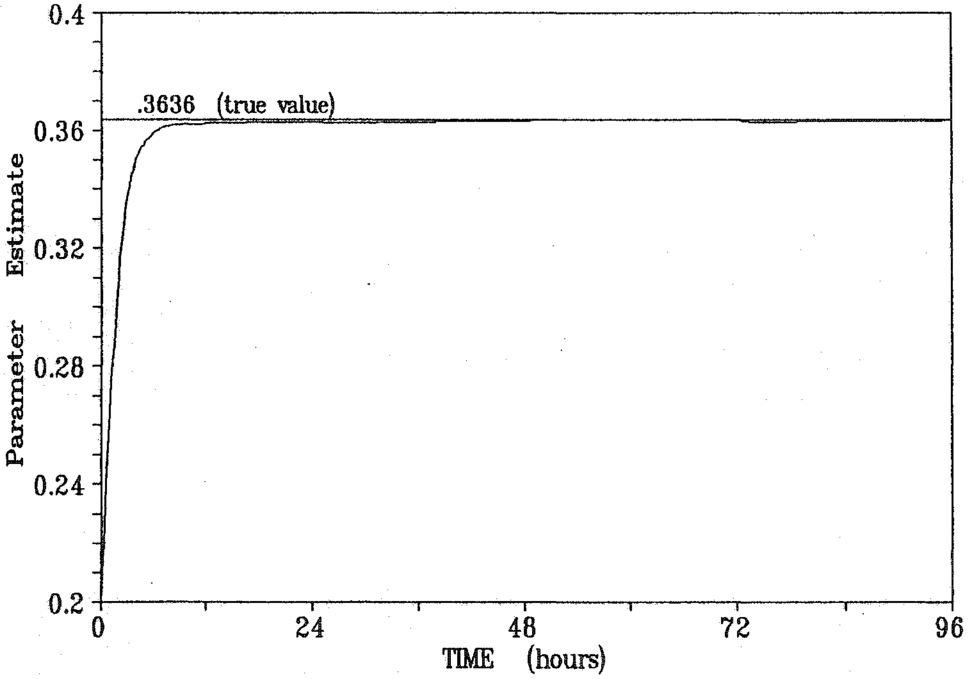

Figure 5.16 Control law parameter estimate, $\hat{K}$. $g_0 = 0.0039$, $\alpha = 100$.

The adaptive gain expression for this example is

$$\mu_{\hat{g}}(k) = \frac{g_0}{\left|\hat{\theta}(k)\right| + \alpha} \tag{5.2.37}$$

where g_0 and α are chosen by the designer. Refer to the discussion in Chapter 4 concerning the effect that each of these parameters has on performance. To demonstrate this effect, tests were run using two different sets of values for g_0 and α.

The initial transient in the linearized plant parameter estimate, $\hat{\theta}$, has a peak value of approximately *-100*. This is shown in Figure 5.17.

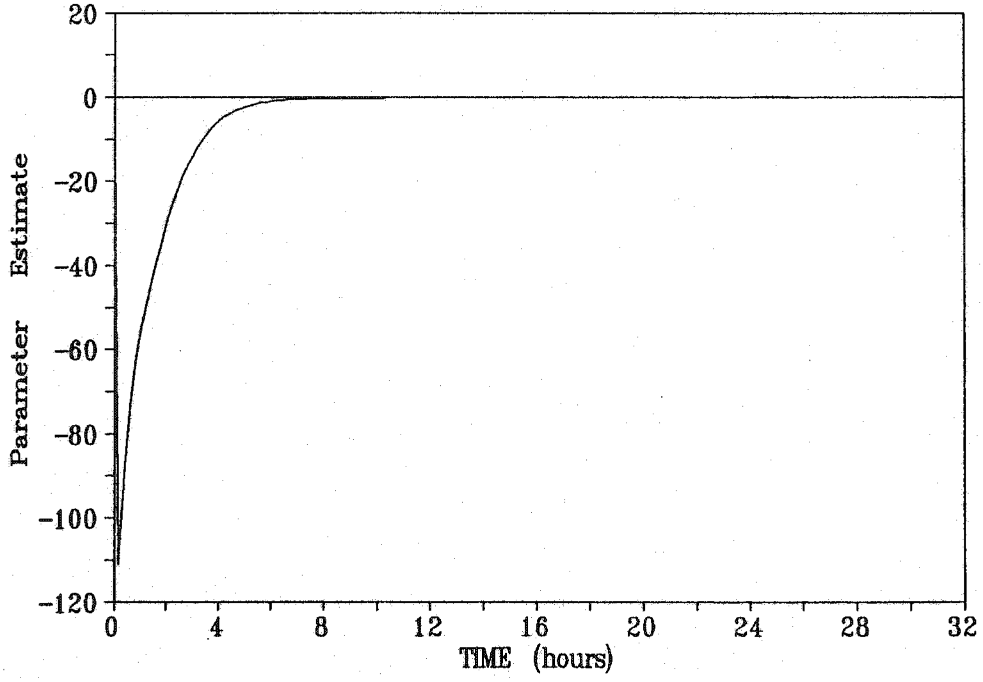

Figure 5.17 Linearized plant parameter estimate, $\hat{\theta}$.

If α has a small value (e.g. 1) and g_0 is set to a value which provides good steady state performance, then the initial gain is quite low due to the large $\hat{\theta}$. Consequently parameter convergence is slow. On the other hand if g_0 is chosen large enough to provide a good initial convergence rate then the parameter estimate is more sensitive to disturbances entering the system after the estimate has converged. This condition is demonstrated in Figures 5.18 through 5.21. Curves 1 and 2 in these figures are plots for gains of

$$\mu_1(k) = \frac{5x10^{-4}}{\left|\hat{\theta}(k)\right| + 1} \tag{5.2.38}$$

and

$$\mu_2(k) = \frac{3.9x10^{-3}}{\left|\hat{\theta}(k)\right| + 100} \tag{5.2.39}$$

respectively. By choosing α larger and approximately equal to $\left|\hat{\theta}(0)\right|$, g_0 can be chosen larger. With the values chosen the ratio of $\mu_2(0)/\mu_1(0)$ is approximately 4.

As Figures 5.20 and 5.21 show, the higher initial gain reduces the convergence time. Curves 1 with lower g_0 and α exhibit transients for every change in influent concentration. The higher α value of μ_2 completely desensitizes the system to these disturbances. However the substrate concentration control shown in Figure 5.19 is acceptable in both cases indicating that while the performance is better with μ_2 versus μ_1, the system is fairly robust with regard to these two parameters.

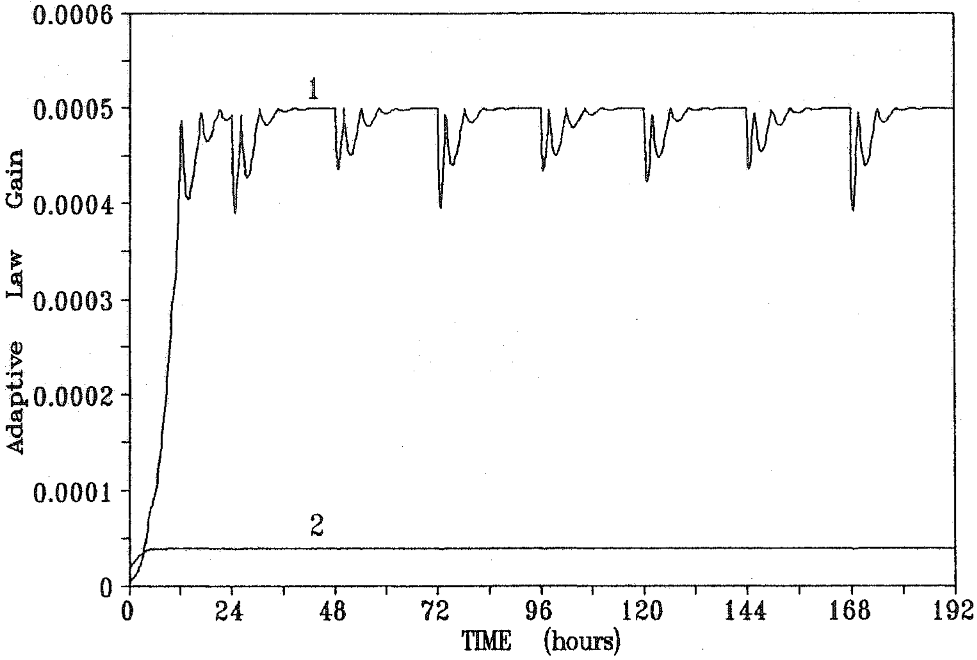

Figure 5.18 Adaptive law gain. Curve 1, μ_1; Curve 2: μ_2.

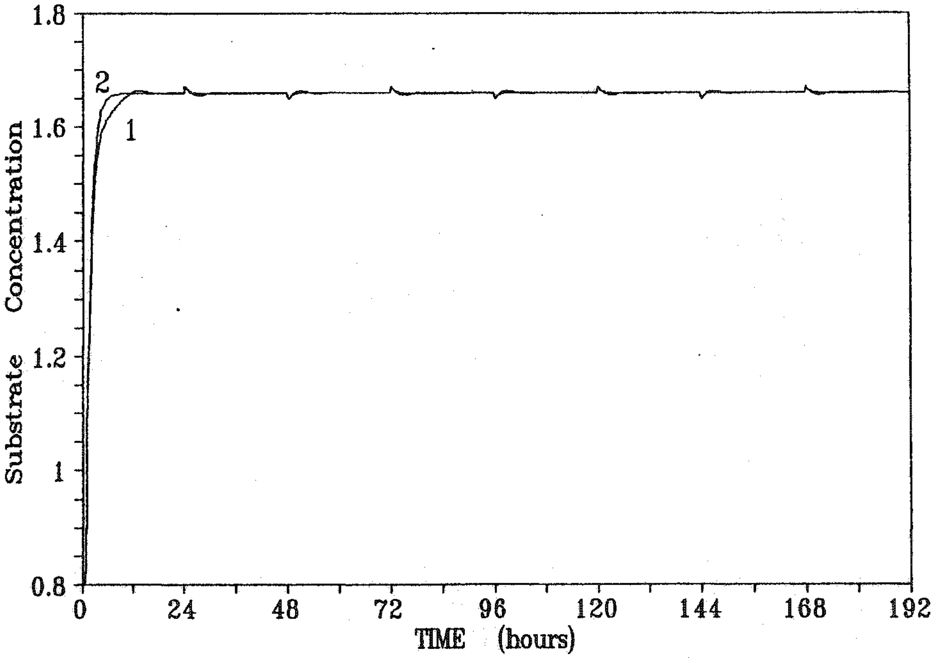

Figure 5.19 Adaptive law gain effect on plant output. Curve 1, μ_1; Curve 2, μ_2.

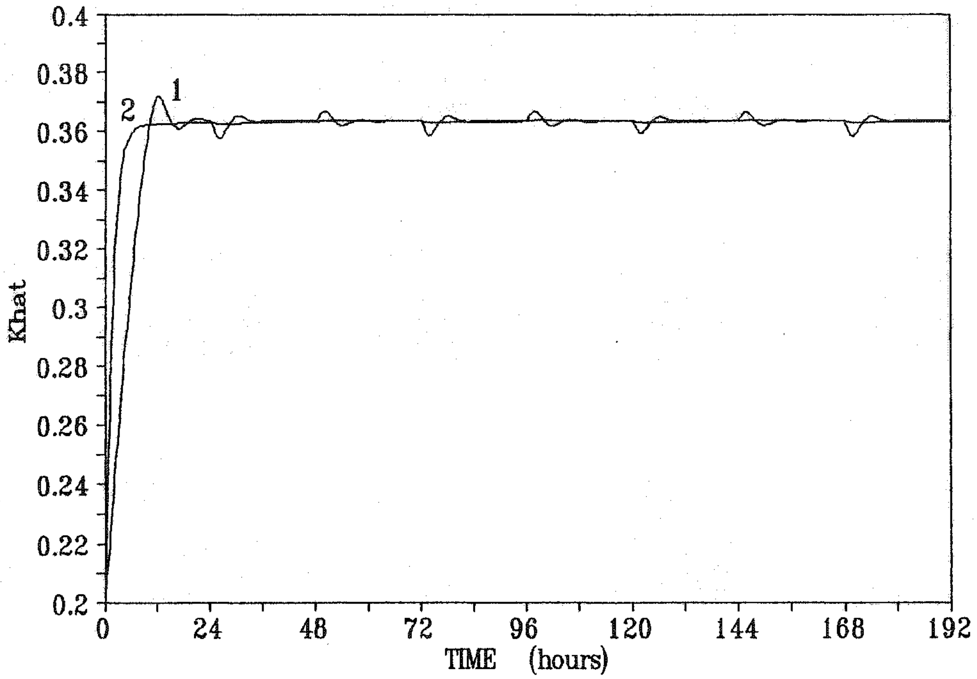

Figure 5.20 Adaptive law gain effect on parameter estimate. Curve 1, μ_1; Curve 2, μ_2.

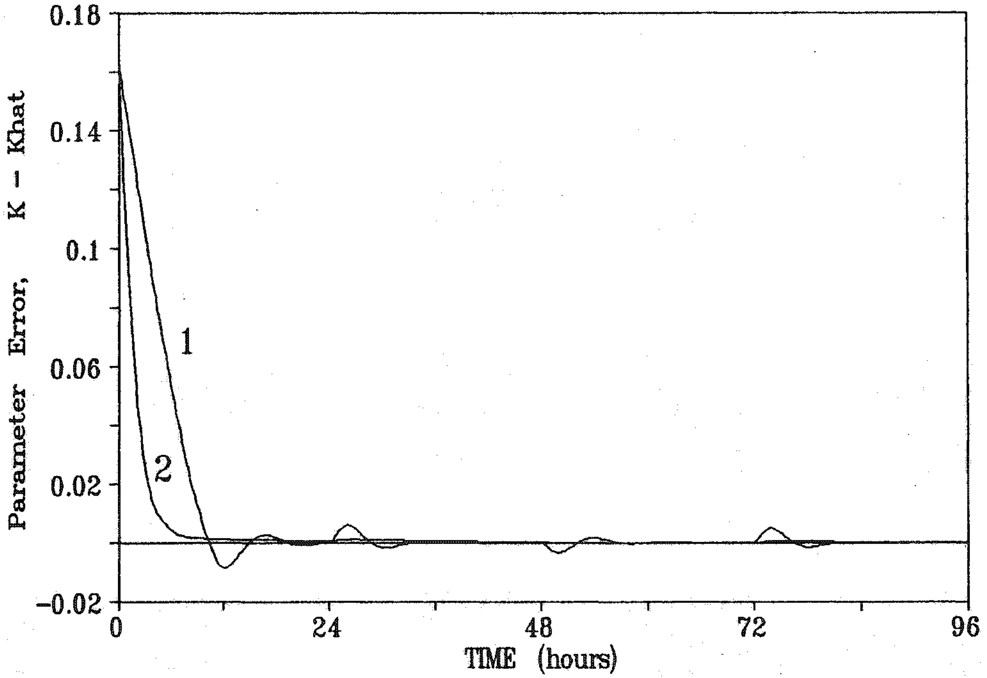

Figure 5.21 Adaptive law gain effect on the $\hat{K}$ parameter estimate convergence rate. Curve 1, μ_1; Curve 2, μ_2.

5.2.13 Standard System Configuration Performance

In Chapter 3, section 3.2.2, the problems encountered when using the standard estimator/plant configuration were demonstrated using a linear plant for simplicity. We will now attempt to control the nonlinear bioreactor using the standard system configuration and show that a similar problem exists.

The bioreactor is operating under closed loop control. Since the estimated parameter appears explicitly in the control law the plant output is an explicit function of the estimated parameter. This in turn makes the input signal to the control law a function of the estimated parameter due to the closed loop operation. The dependence of the plant output on the estimated parameter has a considerable impact on the gradient calculation.

Figure 5.22 is a sketch of the standard system configuration showing the two possibilities for using either the measured plant output or the estimated plant output in the control law and for closing the feedback loop.

In a standard configuration, with other than linearizing control laws, the gradient is calculated based on the RPEM model alone. We will attempt to follow this same method with a linearizing control law. Next we will investigate the system performance taking into account the effect the linearizing control law has on the plant output and consequently on the gradient. The two cases mentioned above will be investigated when the control law and feedback signal source is the measured plant output and when it is the estimated plant output. Refer to Appendix C for the details of the gradient calculation for each case.

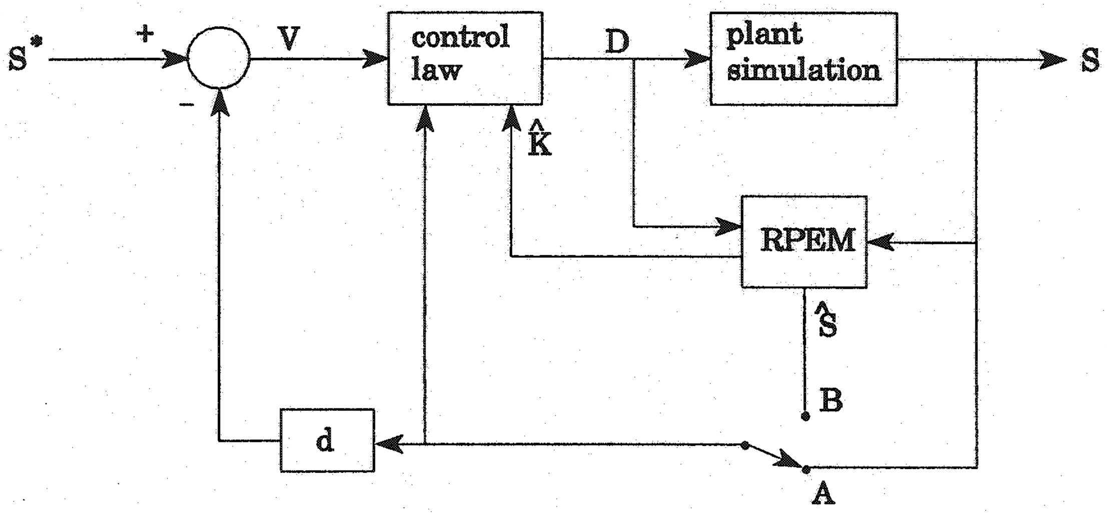

Figure 5.22 Standard configuration for plant and estimator.

A. For the switch in position A in Figure 5.22 . (measured plant output)

Case 1: The dependence of the plant responses (i.e. s and Q) on $\hat{K}$ neglected. Then (refer to C.1.8)

$$\begin{aligned}\psi(t+1) &= \psi_s(t+1) \\ &= (1-hu)\psi_s(t) + \frac{e(t)}{s_{in}-s}hQ\,. \end{aligned} \tag{5.2.40}$$

where $e(t) = s(t) - \hat{s}(t)$. Since $e(t)$ would vanish if the parameter estimate converges, $\psi(t+1)$ would reduce to

$$\psi(t+1) = (1-hu)\psi_s(t)\,. \tag{5.2.41}$$

Inspection of (5.2.41) shows that if $0 < hu < 2$ then the gradient vanishes. If $hu < 0$ or if $hu > 2$ the gradient is unbounded. For this plant $0 < hu \leq .0125$ so if the parameter converges to the true value, the gradient would vanish causing the covariance to blow up.

Figures 5.23 through 5.25 show the simulation results with the gradient given by (5.2.40). The unknown parameter does not converge to the true value and the system performance is obviously unacceptable. Figures 5.23 and 5.25 can be compared with Figures 5.16 and 5.15 which were generated from the approach suggested in this thesis.

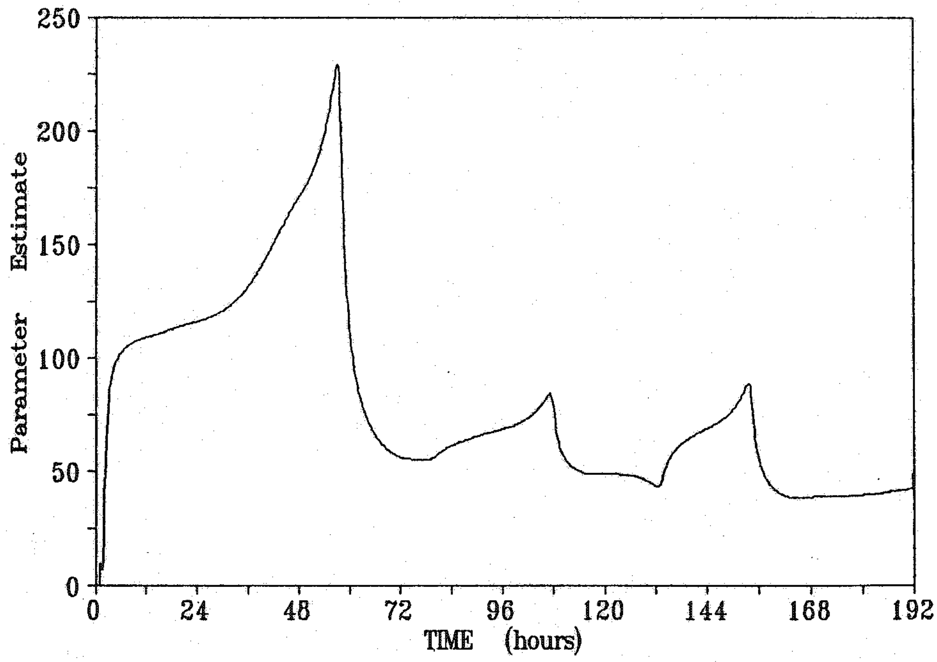

Figure 5.23 Parameter estimate for Case 1.

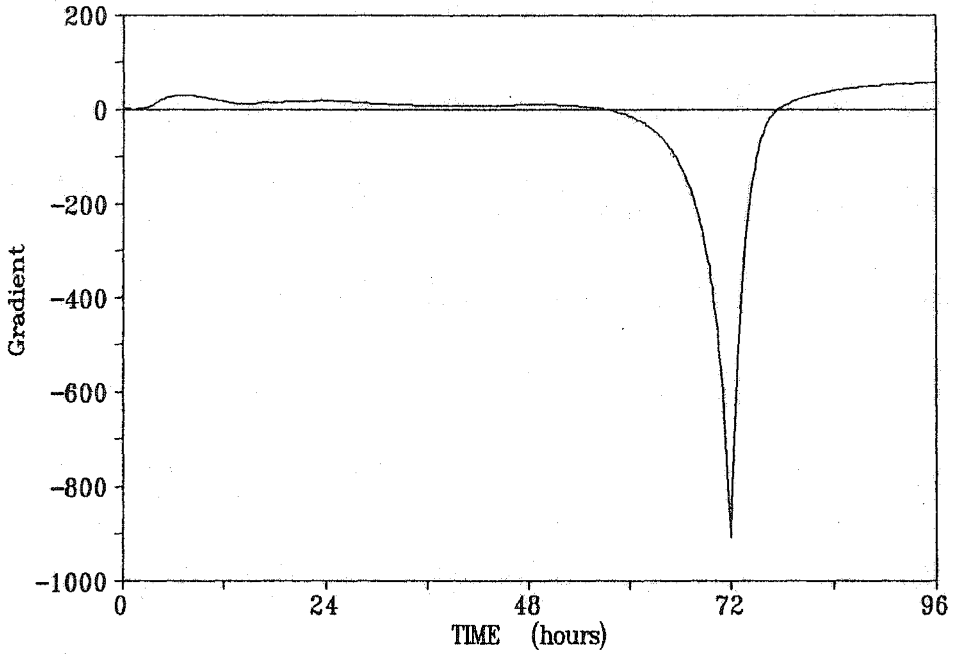

Figure 5.24 Gradient for Case 1.

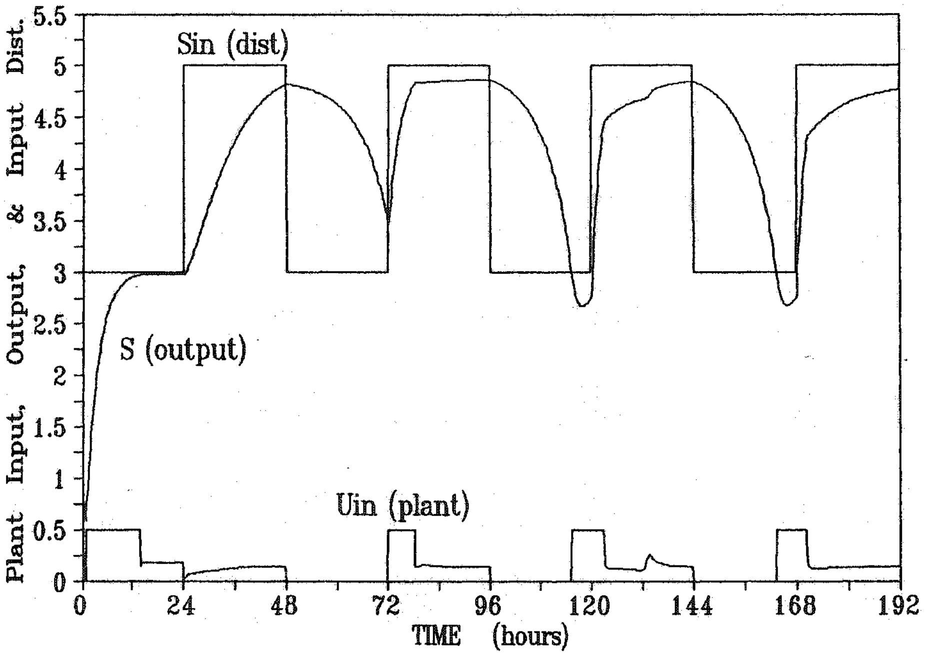

Figure 5.25 Output concentration control, Case 1.

Case 2: The dependence of the plant responses on $\hat{K}$ taken into account. Then (refer to C.1.11)

$$\begin{aligned}\psi(t+1) &= \psi_{\hat{s}}(t+1) - \psi_s(t+1)\\ &= \psi(t) + (u-d)h\psi_s(t)\left[\frac{s_{in}-\hat{s}}{s_{in}-s}\right] - hu\psi_{\hat{s}}(t) + hd\psi_s(t)\\ &\quad + \left[\frac{e(t)}{s_{in}-s}\hat{K} + (K-\hat{K})\right] h\frac{dQ}{d\hat{K}}\\ &\quad + \left[\frac{e(t)}{s_{in}-s} - 1\right] hQ.\end{aligned} \tag{5.2.42}$$

The terms where $\psi_s(t)$, $\psi_{\hat{s}}(t)$, K and $\dfrac{dQ}{d\hat{K}}$ appear in (5.2.42) can not be fully implemented because they contain another unknown in addition to K. If the parameter estimate converges, then the gradient in this case would reduce to

$$\psi(t+1) = \psi(t) - hQ. \tag{5.2.43}$$

Equation (5.2.43) represents an integrator with an input of $-Q$.

A simulation for this case was run using an approximate gradient derived from (5.2.42) with the terms containing unknowns left out. Therefore from (5.2.42),

$$\psi(t+1) = \psi(t) + \left[\frac{e(t)}{s_{in}-s} - 1\right] hQ. \tag{5.2.44}$$

Figure 5.26 and 5.27 show the parameter estimate and gradient respectively.

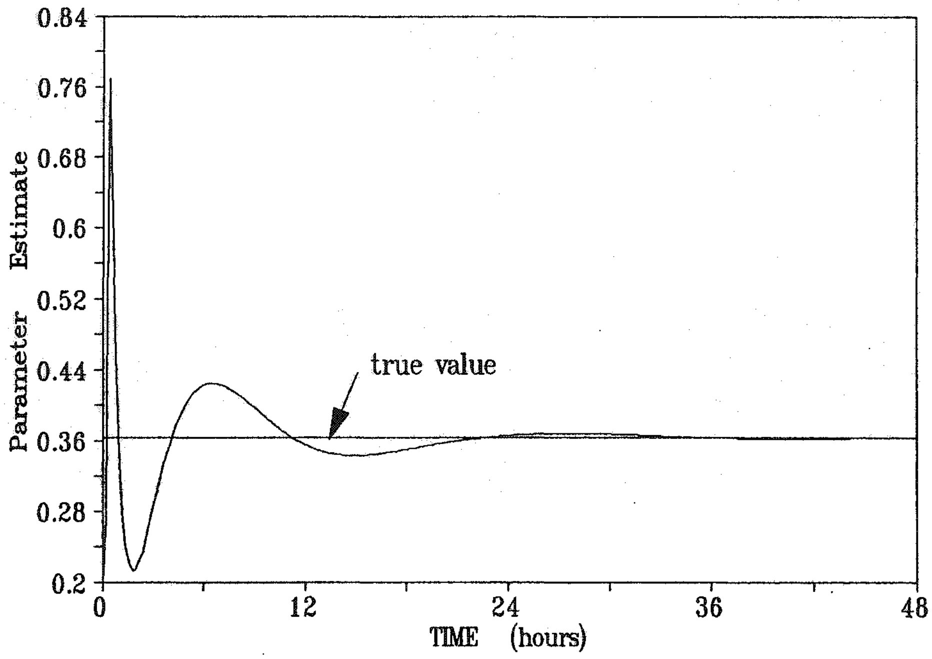

Figure 5.26 Parameter estimate, Case 2.

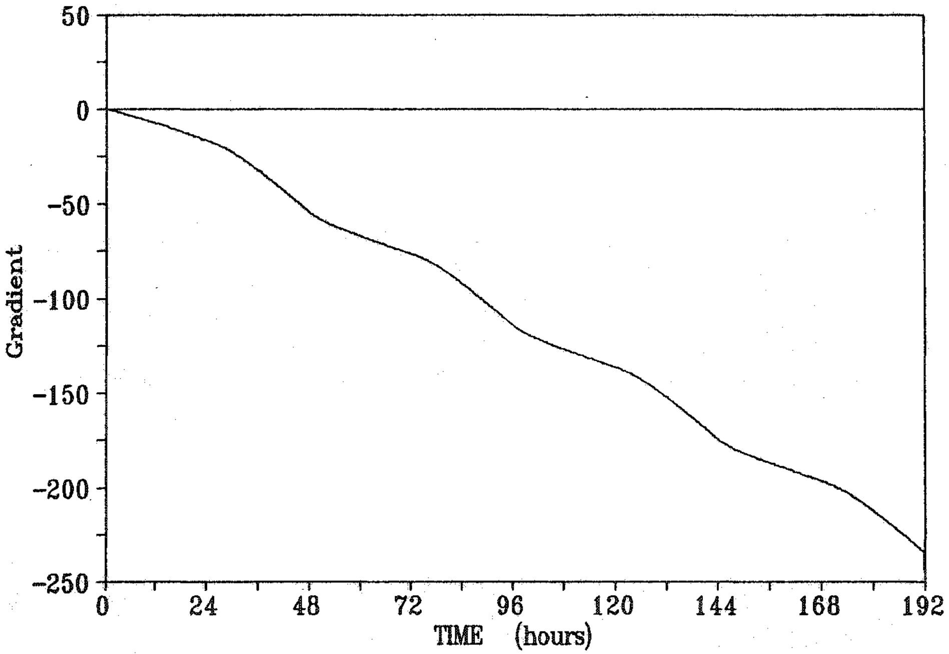

Figure 5.27 Gradient for Case 2.

The parameter estimate converges to the true value even though the gradient is unrealistic. It was shown in Chapter 3 that the parameter will converge even when the true gradient is replaced by a constant. This worked as long as the sign of the constant was chosen correctly and the range of arbitrary constant gradient values is sufficiently limited. There is a similarity between the present case and the constant gradient case. In the present case the gradient has the correct sign and so the parameter converges. Since the parameter converges, $e(t)$ vanishes and the gradient is given by (5.2.43). However a gradient which responds as an integrator with a continuous input is not practical. If the estimation routine is discontinued after the parameter converges, in order to limit the magnitude of the gradient, then future changes in the unknown parameter cannot be followed. (i.e. the system cannot be adaptive)

B. For the switch in position B in Figure 5.22. (estimated plant output)

Case 3: The dependence of the plant responses (s and Q) on $\hat{K}$ neglected. Then (refer to C.1.14)

$$\psi_s(t+1) = (1-dh)\,\psi_s(t)\,. \tag{5.2.45}$$

No simulation results could be obtained using (5.2.45). This is not surprising since this gradient is totally independent of the estimated parameter. It depends only on $\psi_s(0)$. The term dh, where h is the sampling rate, will generally be less than unity. If by chance the parameter should happen to converge for a particular

$\psi(0)$, as time evolves this gradient will vanish and the covariance will blow up.

Case 4: The dependence of the plant responses on $\hat{K}$ taken into account. Then (refer to C.1.17)

$$\begin{aligned}\psi(t+1) &= \psi_{\hat{s}}(t+1) - \psi_s(t+1) \\ &= \psi(t) - (u-d)h\psi_{\hat{s}}(t)\left[\frac{s_{in}-s}{s_{in}-\hat{s}}\right] + hu\psi_s(t) - hd\psi_{\hat{s}}(t) \\ &\quad - \left[\frac{s_{in}-s}{s_{in}-\hat{s}}\hat{K} - K\right]h\frac{dQ}{d\hat{K}} - \left[\frac{s_{in}-s}{s_{in}-\hat{s}}\right]hQ.\end{aligned} \tag{5.2.46}$$

This gradient is very similar to the gradient given by (5.2.42) in Case 2. When terms which cannot be implemented are dropped it reduces to

$$\begin{aligned}\psi(t+1) &= \psi(t) - \left[\frac{e(t)}{s_{in}-\hat{s}}hd + \frac{s_{in}-s}{s_{in}-\hat{s}}hu\right]\psi_{\hat{s}}(t) \\ &\quad - \frac{s_{in}-s}{s_{in}-\hat{s}}hQ.\end{aligned} \tag{5.2.47}$$

Using this approximation, the parameter converges but the gradient is unbounded. The simulation results are nearly identical to Figures 5.26 and 5.27 for Case 2.

In this section we have demonstrated with a nonlinear plant that self-tuning of the control law using the standard configuration for plant and estimator will fail. None of the four cases considered produced satisfactory results. Neglecting the dependence of the plant effluent substrate concentration and methane gas production on the unknown parameter is analogous to the way in which parameter estimation in a system without a linearization control law is done. This was attempted in Case 1 and 3 utilizing the measured and the estimated plant output respectively. This method fails because the gradient is strongly dependent on the neglected terms.

Cases 2 and 4 includes in the gradient all variables that are functions of $\hat{K}$. Several of the terms in these gradients contain unknowns and therefore they can only be approximately implemented. With these approximate gradients the parameter converges to the true value. However, at convergence many of the gradient terms vanish and the remaining terms represent the integral with respect to time of the methane gas production. This of course leads to an unbounded gradient for a continuous process.

5.3 Summary

In this chapter we have applied self-tuning feedback linearization to simulations of two real plants of industrial interest. Both plants have appeared in the literature where control schemes, different from the one proposed here, have been investigated (Cott and Macchietto (1989), Dochain (1986)). Unfortunately, due to lack of certain data and other details in the publications, it was not possible to exactly reproduce the previous authors results and include a comparison in this chapter. However self-tuning feedback linearization as developed in this thesis equals or exceeds the published control performance of the other schemes for both applications.

Another feature of the new method was brought out by the batch reactor example. Due to the way in which the unknown parameters enter the linearized model, a multiplying factor can be used to increase the influence of small parameters on model performance. These parameters may not be identifiable otherwise. Conversely, the effect of a very large unknown parameter could be reduced by the same means to improve the overall conditioning of the estimation problem. At convergence the terms containing these arbitrary multiplying factors vanish and have no effect on the feedback linearized plant performance.

Generally the unknown coefficients in the nonlinear plant model are constant. In the batch reactor application the unknown coefficient of the heat of reaction term is time-varying. The estimation problem was formulated considering the composite of the time-varying coefficient and nonlinear Arrhenius relationship as a single time-varying parameter. In Appendix B the problem was reformulated such that the Arrhenius relationship was a calculable quantity and only the time-varying coefficient was estimated. Comparison of results did not reveal any significant relative advantage to either method.

The second application studied was the control of the effluent substrate concentration of a bioreactor subjected to strong substrate concentration disturbances in the feed stream. The parameter estimation and plant output control was substantially improved in comparison to the method found in Dochain (1986). Using the bioreactor as an example, failure of the standard estimator/plant configuration for a self-tuning feedback linearized system was again demonstrated.

CHAPTER 6

Summary, Conclusions, And Recommendations For Further Research

6.1 Summary

The objective of this research was to develop a simple self-tuning feedback linearizing control system. Chapter 1 introduces this topic and previews the subsequent chapters. An overview of the problem that is investigated is given and the approach taken is outlined.

Chapter 2 reviews some classical methods of nonlinear system control and more importantly discusses the feedback linearization concepts and mathematics that are used in the development of the self-tuning system. Also in this chapter there are brief reviews of several of the more recent proposals for control of nonlinear systems.

Chapter 3 explores the possibility of using the well known linear plant self-tuning system configuration for a self-tuning feedback linearizing application. Because of its versatility, the recursive prediction error method was chosen for parameter estimation. The performance of the standard system architecture was evaluated when a linearizing control law was used. Several different test conditions were tried with both measured and estimated plant states. For each method of obtaining the states either the true gradient or an approximate gradient was used. When the true gradient was used the covariance always increased without bound. When using an approximate gradient the parameter estimate did not converge in one case or converged slowly with a large transient error as compared with the true gradient case.

A new architecture is proposed to eliminate the problem of covariance blow up when the true gradient is used. In the new architecture the estimator is connected between the system input and plant output. (i.e. in parallel with the linearized plant) The gradient between these two points is not dependent on both the real plant and plant model gradients. Therefore the gradient cancellation that is present with the standard architecture is eliminated and the covariance remains bounded.

The parameters that can be directly estimated from the new architecture are not those of the nonlinear plant. These parameters are combinations of the true nonlinear plant parameters and their estimates. It was pointed out that subsequently a method for obtaining the nonlinear plant parameters from the composite estimates would be developed.

In Chapter 4 the problem of obtaining the nonlinear plant parameters from the feedback linearized plant parameters is studied. The various functional relationships that can occur between these two sets of parameters is investigated. An adaptive law is developed by first reparameterizing the linearized model and then formulating a parameter error equation from which the relative magnitude of the true parameters and their estimates can be made.

Interpretation of the linearized plant parameter error equation leads directly to the adaptive law. That is, the sign of the error indicates whether the magnitude of the previous estimate should be increased or decreased. This leads naturally to a sign algorithm for the adaptive law. The adaptive law and the class of plants for which this adaptive law is applicable are specified in section 4.2.2. The adaptive law is shown to be asymptotically stable and capable of one step convergence under ideal conditions.

Several simulations were run to evaluate the self-tuning system developed in Chapters 3 and 4. Prior to running the simulations the need for using a proper numerical solution routine for the plant model equations is pointed out. For the simulation to accurately represent the plant response the numerical routine has to be capable of following the apparent structural changes which will occur in the real plant. Both Runge-Kutta and Euler methods have this property and are used for the simulations. The plots show excellent parameter convergence and therefore successful self-tuning. Adaptive signal tracking is also demonstrated.

Chapter 5 applies the new method to simulations of two real industrial processes. The first is temperature profile tracking control for an exothermic batch reactor. The time-varying nonlinear heat of reaction term is the unknown parameter which is estimated and used in the control law. This example also demonstrates another aspect of estimating parameters which appear in the linearized model. In this particular case the heat of reaction term is multiplied by a small constant. Consequently it had very little influence on the output of the linear model and it could not be accurately estimated. A multiplying factor was added to this term which increased its effect on the output and made accurate estimation possible. This approach is justified since terms of this type asymptotically reduce to zero and eventually have no effect on system performance. Comparison of this technique with Generic Model Control applied to this process (Cott and Macchietto (1989)) shows significant reduction in jacket temperature swings needed to keep the reactor temperature tracking the profile.

The second application is substrate concentration control in an anaerobic digestion process. This process has a bilinear nonlinearity and an unknown parameter which is needed in the control law. As in the batch reactor case, the self-tuning feedback linearization scheme worked very well exhibiting both accurate parameter estimation and good control of the substrate concentration in spite of strong concentration disturbances entering the system.

This process was also used to demonstrate the failure of the classical self-tuning configuration when utilized for self-tuning feedback linearization. This problem was demonstrated previously in Chapter 3 but with a linear plant.

6.2 Suggested Future Research Directions

1. The linearizing control law is always a function of the nonlinear plant states. Therefore, implementation of feedback linearization is restricted to plants with measurable states.

   The nonlinear coordinate transformation defines the states of the linearized plant as a function of the nonlinear plant states. With the new system configuration, estimates of the linearized plant states are available. Possibly these estimates could be substituted into the nonlinear coordinate transformations and these equations solved for estimates of the nonlinear plant states. This would remove the state measurability restriction on the plant.

2. At present, the new system relies on computer simulations to establish stability. Investigate the possibility of analytically proving stability.

3. Only the deterministic case was considered in this research. However, the stochastic performance is of interest. The effect of noise on the linearized

plant parameter estimates can readily be investigated by using the RPEM algorithm with an innovations model for the linearized plant.

4. Throughout the development the plant structure was assumed to be known. Since this is not always the case in practice, it would be worthwhile to investigate the robustness of the system to structural uncertainty.

5. Three different nonlinear plants were involved in the development and evaluations of the self-tuning feedback linearization system. The application to simulations of real plants brought out some aspects of performance which had not been anticipated. This suggests that investigation of the method applied to simulations of a broader spectrum of real plants would be advantageous in further refinement of the method.

6. In section 4.2.2 sufficient conditions on the plant are given for adaptive law (4.2.36a) to apply. Conduct an investigation to find the necessary and sufficient conditions.

7. Investigate applying this method to multivariable systems.

6.3 Conclusions

Rapid progress is being made in both the theoretical and practical application of feedback linearization technology. This powerful technique is the first major departure from the myriad of methods which have been the only means for designing controllers for nonlinear plants for many years.

Self-tuning systems for linear plants are well known. This research has focused on solving the special problems associated with the design of a self-tuning controller for a nonlinear plant based on feedback linearization theory. A new method for designing a self-tuning controller for a nonlinear plant was successfully developed. This system was demonstrated on simulations of two real processes and should prove to be useful in many other industrial applications.

APPENDIX A

A.1 Nomenclature

C_p mass heat capacity of reactor contents, kj/(kg °C)

C_{pi} molar heat capacity of component i, kj/(kmol °C)

ΔH_i heat of reaction for reaction i, kj/kmol

k_i rate constant for reaction i, $kmol^{-1} s^{-1}$

k_i^1 rate constant 1 for reaction i

k_i^2 rate constant 2 for reaction i

M_i number of moles of component i, kmol

MW_i molecular weight of component i, kg/kmol

Q heat realeased in reactor, kW

ρ density of reactor contents, kg/m^3

r radius of reactor, m

R_i reaction rate of reaction i, kmol/s

T temperature, °C

U heat transfer coefficient of reactor, kW/(m^2 °C)

V volume, m^3

W reactor weight, kg

Subscripts

1 reaction 1, (A + B → C)

2 reaction 2, (A + C → D)

A component A

B component B

C conponent C

D component D

j jacket

r reactor

Superscript

sp setpoint

A.2 Batch reactor model equations

$$
\begin{aligned}
\dot{M}_A &= -R_1 - R_2 \\
\dot{M}_B &= -R_1 \\
\dot{M}_C &= +R_1 - R_2 \\
\dot{M}_D &= +R_2 \\
R_1 &= k_1 M_A M_B \\
R_2 &= k_2 M_A M_B
\end{aligned}
$$

$$k_1 = e^{k_1^1 - \frac{k_1^2}{T_r + 273.15}}$$

$$k_2 = e^{k_2^1 - \frac{k_2^2}{T_r + 273.15}}$$

$$W = MW_A M_A + MW_B M_B + MW_C M_C + MW_D M_D$$

$$M_t = M_A + M_B + M_C + M_D$$

$$C_{p_t} = (C_{p_A} M_A + C_{p_B} M_B + C_{p_C} M_C + C_{p_D} M_D) / M_t$$

$$V = \frac{W}{\rho}$$

$$A = \frac{2V}{r}$$

$$Q_j = UA(T_j - T_r)$$

$$Q_r = -\Delta H_1 R_1 - \Delta H_2 R_2$$

$$\dot{T}_r = \frac{Q_r + Q_j}{M_t C_{p_t}}$$

$$\dot{T}_j = \frac{F_j \rho_j C_{p_j} (T_j^{sp} - T_j) - Q_j}{V_j \rho_j C_{p_j}}$$

A.3 Physical properties and process data

$MW_A = 30$ kg/kmol
$MW_B = 100$ kg/kmol
$MW_C = 130$ kg/kmol
$MW_D = 160$ kg/kmol

$C_{pA} = 75.310$ kJ/(kmol °C)
$C_{pB} = 167.36$ kJ/(kmol °C)
$C_{pC} = 217.57$ kJ/(kmol °C)
$C_{pD} = 334.73$ kJ/(kmol °C)

$k_1^1 = 20.9057$
$k_1^2 = 10{,}000$
$k_2^1 = 38.9057$
$k_2^2 = 17{,}000$

$\Delta H_1 = -41840$ kJ/kmol
$\Delta H_2 = -25105$ kJ/kmol

$\rho = 1000$ kg/m^3
$r = 0.5$ m
$u = 0.6807$ kW/(m^2 °C)
$\rho_j = 1000$ kg/m^3
$C_{pj} = 1.8828$ kJ/(kg °C)
$F_j = .348$ m^3/min.
$V_j = 0.6912$ m$_3$

A.4 Initial conditions

$M_A = 12$ kmol
$M_B = 12$ kmol
$M_C = 0$ kmol
$M_D = 0$ kmol

$T_r = 20$ °C
$T_j = 20$ °C

APPENDIX B

B.1 Evaluation of an alternate representation of Q_r with regard to ease of estimation

The heat of reaction term can be factored into two components. The objective is to separate, as much as possible, the highly nonlinear terms which are known functions of the reactor temperature from the remaining unknown terms. If this can be done then the curve representing the unknown terms should not contain the sharp peak and slope reversal (e.g. Figure 5.2) which are difficult for the estimator to track.

The heat of reaction term can be factored as

$$Q_r = - \left[\Delta H_1 M_A M_B + \Delta H_2 M_A M_C e^{(k_2^1 - k_1^1)} \; e^{\frac{(k_1^2 - k_2^2)}{T_E}} \right] e^{k_1^1} e^{-\frac{k_1^2}{T_E}} \tag{B.1.1}$$

$$\doteq \alpha E$$

where α represents the bracketed term and

$$E \doteq e^{k_1^1} e^{-\frac{k_1^2}{T_E}} \tag{B.1.2}$$

where T_E is in degrees Kelvin. The heat of reaction appears in the plant model divided by another unknown varying quantity, $M_t C_{p_t}$. The complete term is therefore

$$c = \frac{\alpha}{M_t C_{p_t}} E \doteq c_0 E. \tag{B.1.3}$$

All of the unknowns are contained in c_0. (i.e. M_A, M_B, M_C, M_t, and C_{p_t}) Figures B1, B2, and B3 are plots of c_0, E, and c respectively. Figure B3 was obtained from the product of the data plotted in Figures B1 and B2.

Factoring Q_r as given by (B.1.1) produces a smooth unknown term, c_0, for estimation. Figure B4 compares $\hat{c}_0$ with c_0 The initial error when t is less than approximately 20 minutes is due to the start up transient in the estimation algorithm. Output from the algorithm during this period of time is in general incorrect in magnitude and sign. This in turn causes the adaptive law to respond incorrectly.[1] After the transient has died out and the output from the algorithm is more accurate, the rate of change of c_0 is too fast for the system to track. Increasing the adaptive law gain improves the tracking accuracy but accentuates the initial error and the ringing present on the lower portion of the curve.

From Figure B1, c_0 has a dynamic range of approximately 6800. Equation (B.1.3) can be written as

$$c = \frac{c_0}{\Delta}(\Delta E) \doteq c_0' E' \tag{B.1.4}$$

where Δ is some arbitrary constant. The dynamic range and hence the rate of

[1] The magnitude of the initial response from the estimation algorithm is usually large. However by making the adaptive law gain an inverse function of the absolute value of the algorithm output, the magnitude of the incorrect response from the adaptive law is limited. See chapter 4, equation (4.2.31).

change of c_0 is reduced by a factor of Δ. Figure B5 compares $\hat{c}_0'$ with c_0' for $\Delta = 100$. The plot shows substantial improvement in tracking when compared with Figure B4. The initial error which is due to the estimation algorithm transient is still present as would be expected. Choosing a $\Delta > 100$ did make any significant improvement in the tracking error.

Figure B6 compares the heat of reaction term estimate

$$\hat{c} = \hat{c}_0' E' \tag{B.1.5}$$

with the true value. Figure B7 compares the error in the heat of reaction estimates when the whole term is estimated as was done in chapter 5 and when only the unknown coefficient is estimated as in (B.1.5). Figure B8 compares the error in reactor temperature, T_r, under the same conditions as for Figure B7.

Figure B7 shows similar errors for both methods of estimating the heat of reaction term. However in this case the overshoot in the estimate shown in Figure B6 occurs at a time which has a beneficial effect on reactor temperature error. But this is not a controllable or predictable characteristic of the estimation process and therefore cannot be relied upon.

For the batch reactor considered in chapter 5 there does not appear to be any relative advantage to either method of estimating the heat of reaction term. However the method outlined in this appendix would probably perform best for even more highly nonlinear cases.

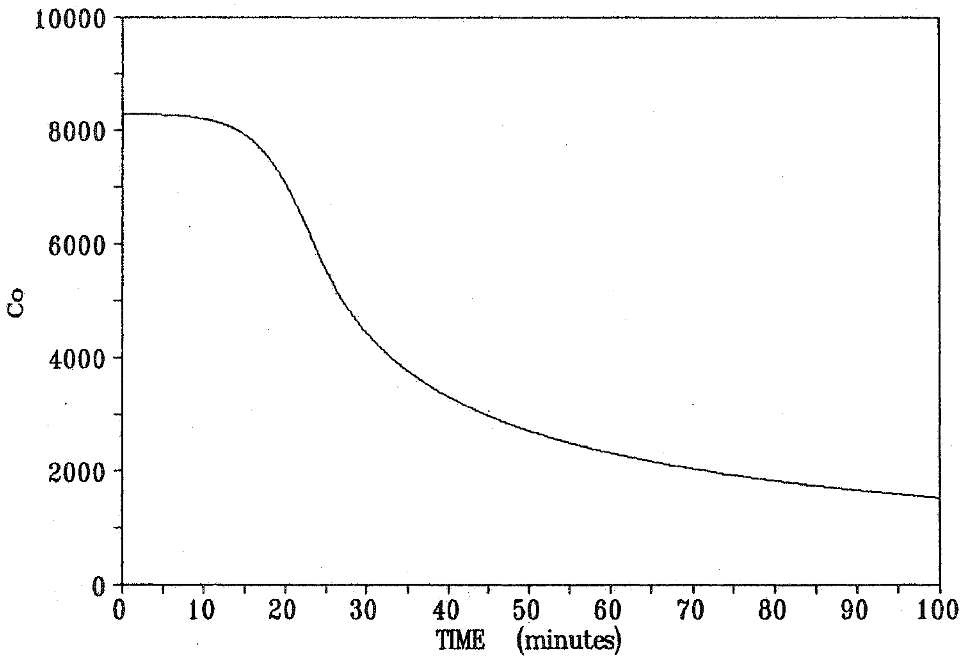

Figure B1 Plot of c_0 factor from Eq. (B.1.3).

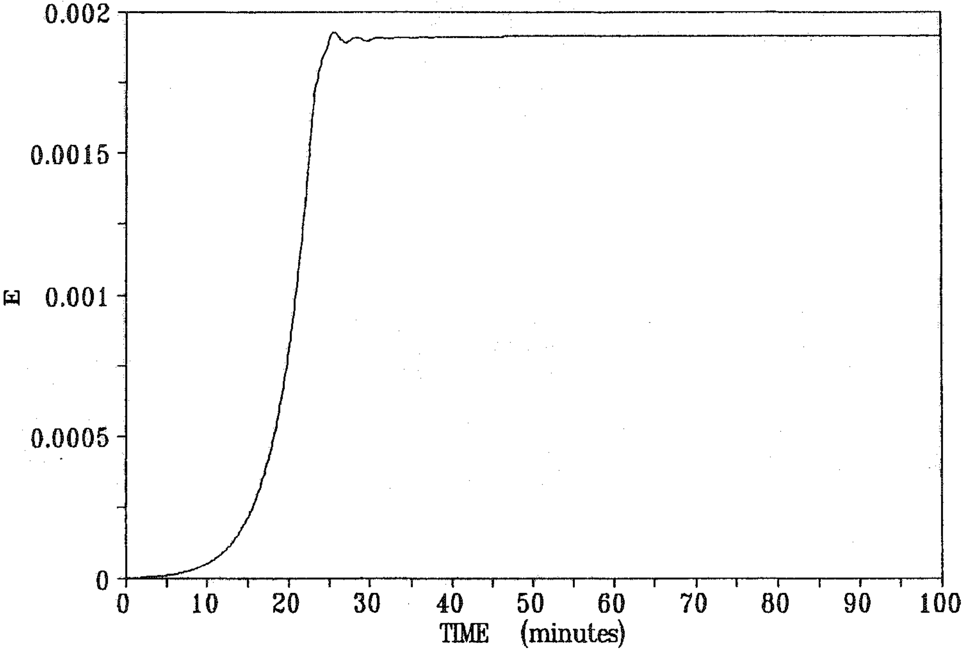

Figure B2 Plot of E factor from Eq. (B.1.3)

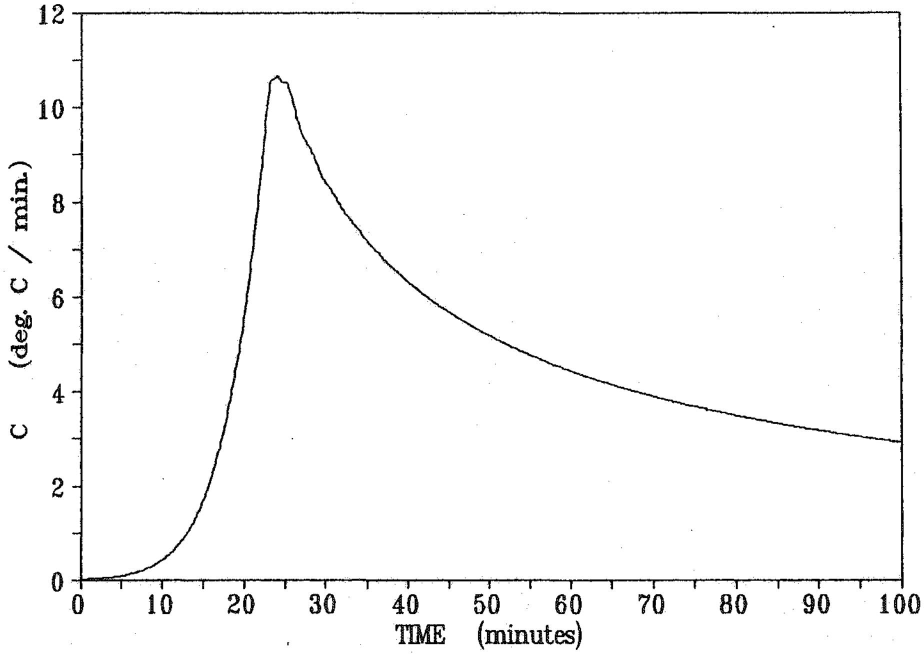

Figure B3 Plot of Eq. (B.1.3), $c = c_0E$.

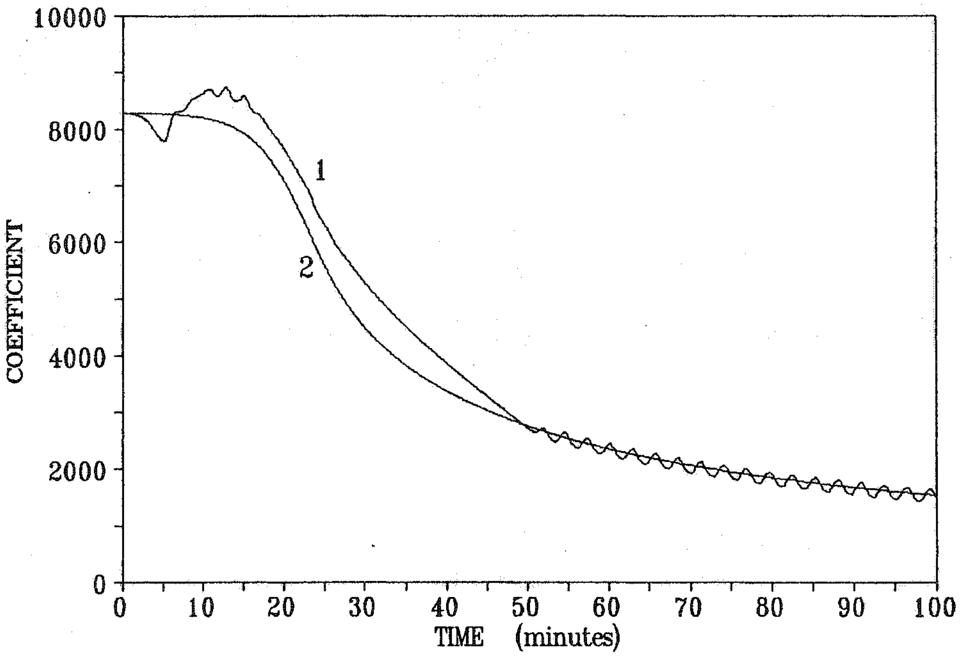

Figure B4 Comparison of coefficient estimate $\hat{c}_0$ with true value. Curve 1: estimate, Curve 2: true value.

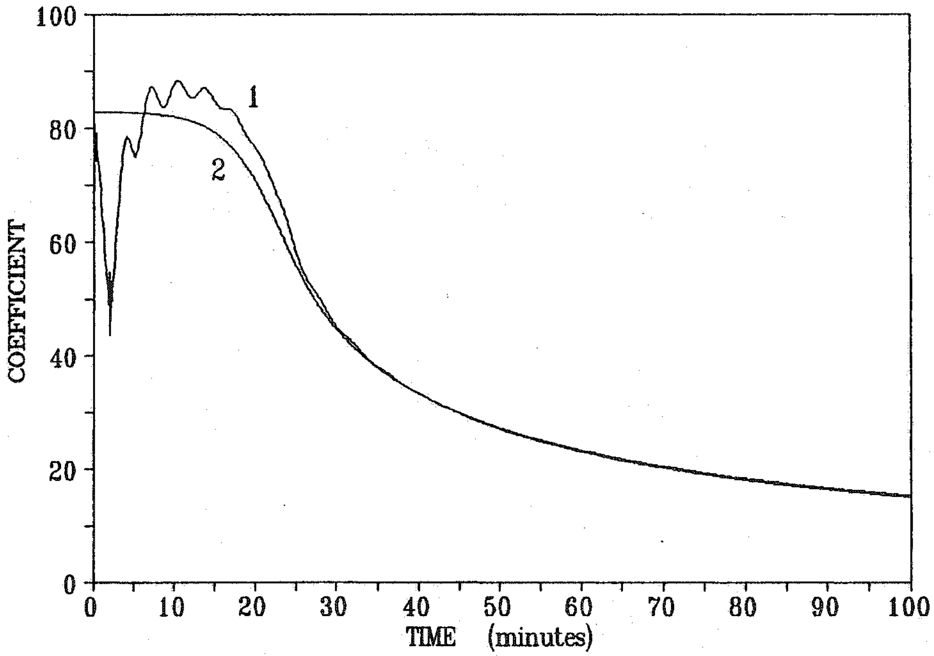

Figure B5 Comparison of coefficient estimate $\hat{c}_0'$ with true value for $\Delta = 100$. Curve 1: $\hat{c}_0'$, curve 2: c_0'.

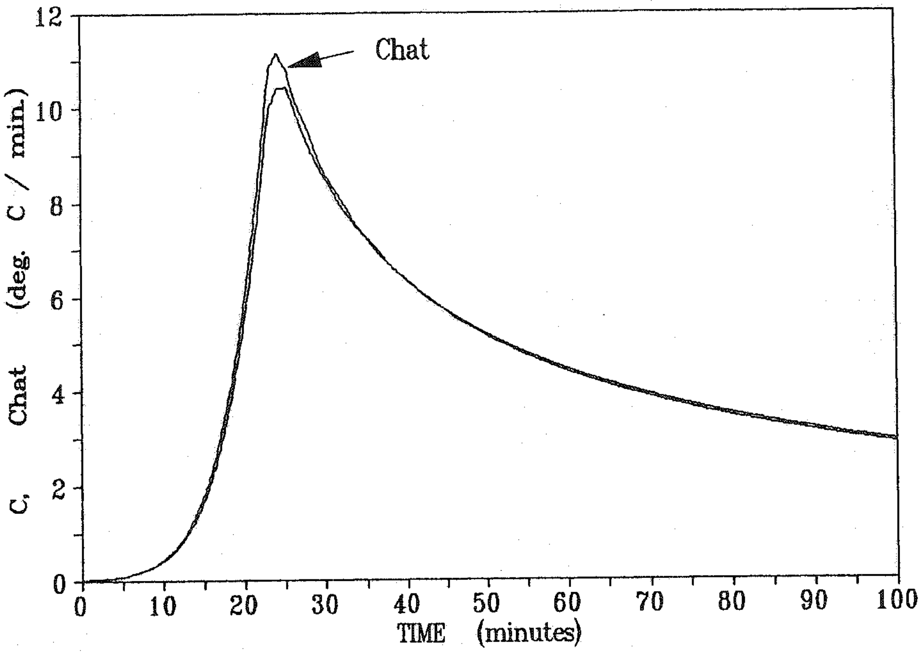

Figure B6 Comparison of heat of reaction estimate with the true value.

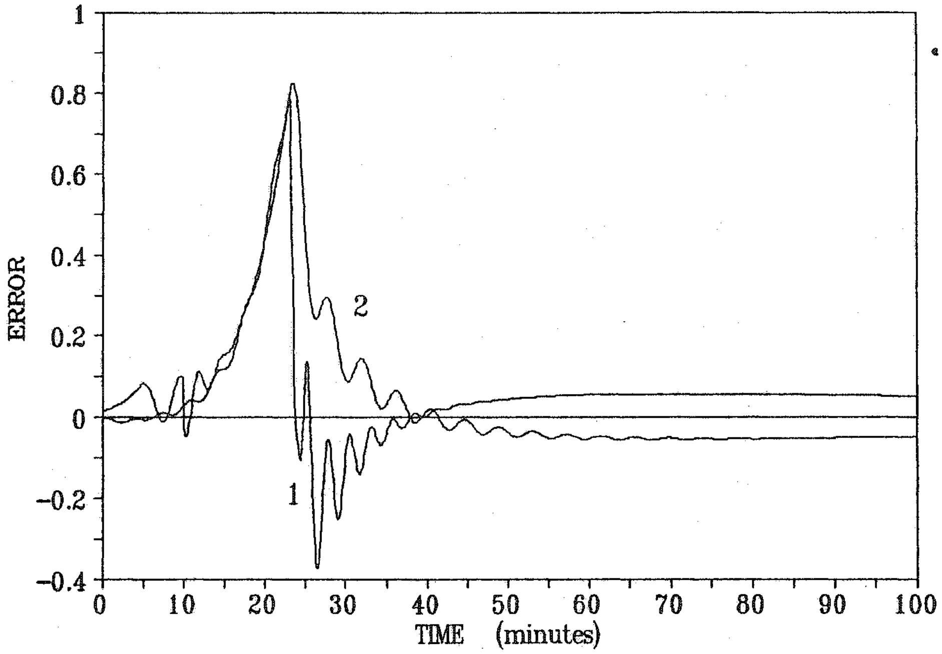

Figure B7 Error in heat of reaction estimate. Curve 1: single term representation, Curve 2: composite representation.

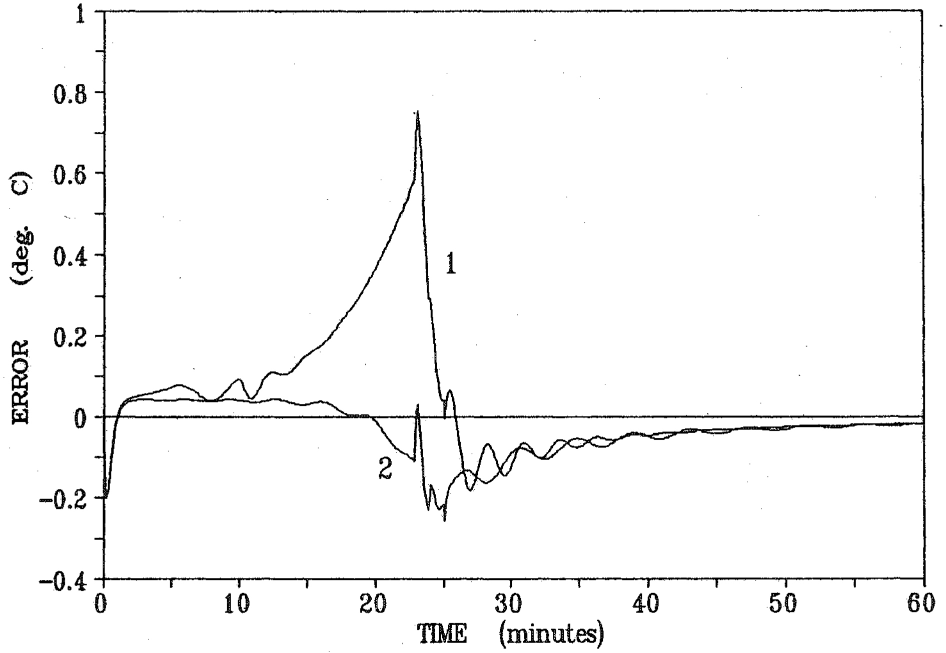

Figure B8 Error in reactor temperature. Curve 1: single term representation of Q_r, Curve 2: composite term representation.

APPENDIX C

C.1 Bioreactor plant gradients

The linearizing control law for the bioreactor is

$$u = \frac{s^{*} - d\alpha + \hat{K}Q}{s_{in} - \alpha} \tag{C.1.1}$$

where α is either s, the measured plant output, or $\hat{s}$ the estimated plant output. (refer to Figure 5.22) The Euler representation for the plant and RPEM plant model is

$$s(t+1) = s(t) + \left[(s_{in} - s)u - KQ\right]h \tag{C.1.2}$$

and

$$\hat{s}(t+1) = \hat{s}(t) + \left[(s_{in} - \hat{s})u - \hat{K}Q\right]h \tag{C.1.3}$$

respectively.

Consider the case where $\alpha = s$ and u is applied to (C.1.2) and (C.1.3). The plant response is

$$s(t+1) = s(t) + \left[s^{*} - ds + (\hat{K} - K)Q\right]h\,. \tag{C.1.4}$$

For the RPEM plant model the response is

$$\hat{s}(t+1) = \hat{s}(t) + \left[\frac{s_{in}-\hat{s}}{s_{in}-s}\left(s^* - ds + \hat{K}Q\right) - \hat{K}Q\right]h. \tag{C.1.5}$$

In Chapter 5, Case 1, a gradient was used which ignored the fact that both s and Q are functions of $\hat{K}$. The negative of the true gradient for the system architecture shown in Figure 5.22 is

$$\psi(t+1) = \psi_{\hat{s}}(t+1) - \psi_{s}(t+1) \tag{C.1.6}$$

but since the functional relationship between s, Q, and $\hat{K}$ is not being considered,

$$\psi(t+1) = \psi_{\hat{s}}(t+1). \tag{C.1.7}$$

Therefore from (C.1.5)

$$\begin{aligned}\psi_{\hat{s}}(t+1) = \psi_{\hat{s}}(t) + \Bigg[&\left(\frac{s_{in}-\hat{s}}{s_{in}-s}\right)\frac{d}{d\hat{K}}(s^* - ds + \hat{K}Q) \\ &+ (s^* - ds + \hat{K}Q)\frac{d}{d\hat{K}}\left(\frac{s_{in}-\hat{s}}{s_{in}-s}\right) - Q\Bigg]h \\ = {}&(1-hu)\psi_{\hat{s}}(t) + \frac{e(t)}{(s_{in}-s)}hQ.\end{aligned} \tag{C.1.8}$$

where $e(t) = s(t) - \hat{s}(t)$. The system gradient can also be derived taking into account the functional dependence of s and Q on $\hat{K}$. For the plant model from (C.1.5),

$$\psi_s(t+1) = \psi_s(t) + \left[\left[\frac{s_{in} - \hat{s}}{s_{in} - s} \right] \frac{d}{d\hat{K}} (s^* - ds + \hat{K}Q) \right.$$

$$\left. + (s^* - ds + \hat{K}Q) \frac{d}{d\hat{K}} \left[\frac{s_{in} - \hat{s}}{s_{in} - s} \right] - \hat{K}\frac{dQ}{d\hat{K}} - Q \right] h \tag{C.1.9}$$

$$= (1 - hu)\psi_s(t) + \frac{e(t)}{(s_{in} - s)} hQ$$

$$+ (u - d)h\psi_s(t) \left[\frac{s_{in} - \hat{s}}{s_{in} - s} \right] + \frac{e(t)}{s_{in} - s} h\hat{K} \frac{dQ}{d\hat{K}} \, .$$

For the plant from (C.1.4),

$$\psi_s(t+1) = \left(1 - hd\right)\psi_s(t) + \left[\hat{K}\frac{dQ}{d\hat{K}} + Q \right] h - hK\frac{dQ}{d\hat{k}} \, . \tag{C.1.10}$$

The system gradient is calculated by combining (C.1.9) and (C.1.10) according to (C.1.6).

$$\psi(t+1) = \psi(t) + (u - d)h\psi_s(t) \left[\frac{s_{in} - \hat{s}}{s_{in} - s} \right] - hu\psi_s(t) + hd\psi_s(t)$$

$$+ \left[\frac{e(t)}{s_{in} - s} \hat{K} + (K - \hat{K}) \right] h \frac{dQ}{d\hat{K}} \tag{C.1.11}$$

$$+ \left[\frac{e(t)}{s_{in} - s} - 1 \right] hQ \, .$$

This gradient contains unknowns and could not be fully implemented. Consider the $\frac{dQ}{d\hat{K}}$ term. The methane gas produced is given by

$$Q = k_2 \mu x \tag{C.1.12}$$

where k_2 is a yield coefficient, μ is the specific growth rate, and x is the biomass concentration in the reactor. Q is a measured variable and does not present a problem. However the expression for $\frac{dQ}{d\hat{K}}$ depends on the unknown k_2. Only portions of the $\psi_s(t)$ and $\psi_s(t)$ terms in (C.1.11) could be implemented since they also involve $\frac{dQ}{d\hat{K}}$. (refer to (C.1.9) and (C.1.10)) The gradient given by (C.1.11) is used in studying Case 2 in Chapter 5.

So far the gradients have been derived using the measured plant output in the control law and as the feedback signal. An alternative is to use the estimated plant output in place of the measured plant output. We will now rederive the gradients using $\hat{s}$ in place of s. Therefore $\alpha = \hat{s}$ in (C.1.1).

Case 3 in Chapter 5 is similar to Case 1 in that neither s nor Q are considered to be functions of $\hat{K}$. Consequently the system gradient depends only on the RPEM plant model, ((C.1.3) driven by (C.1.1))

$$\hat{s}(t+1) = \hat{s}(t) + \left[s^{*} - d\hat{s}\right]h \,. \tag{C.1.13}$$

The gradient calculated from (C.1.13) is

$$\psi_{\hat{s}}(t+1) = (1 - hd)\psi_{\hat{s}}(t) \,. \tag{C.1.14}$$

For Case 4 the dependence of s and Q on $\hat{K}$ is taken into account. This does not change the gradient for the RPEM plant model when $\alpha = \hat{s}$. This gradient is the same as for Case 3. (i.e. (C.1.14)) For the plant,

$$s(t+1) = s(t) + \left[\frac{s_{in} - s}{s_{in} - \hat{s}}\left(s^{*} - d\hat{s} + \hat{K}Q\right) - KQ\right]h \,. \tag{C.1.15}$$

The plant gradient is from (C.1.15)

$$\psi_s(t+1) = \psi_s(t) + \Bigg[\left[\frac{s_{in}-s}{s_{in}-\hat{s}}\right] \frac{d}{d\hat{K}}(s^{\star}-d\hat{s}+\hat{K}Q)$$

$$+ (s^{\star}-d\hat{s}+\hat{K}Q)\frac{d}{d\hat{K}}\left[\frac{s_{in}-s}{s_{in}-\hat{s}}\right] - K\frac{dQ}{d\hat{K}}\Bigg] h \qquad \text{(C.1.16)}$$

$$= \psi_s(t) + (u-d)\left[\frac{s_{in}-s}{s_{in}-\hat{s}}\right] h\psi_{\hat{s}}(t) - hu\psi_s(t)$$

$$+ \left[\frac{s_{in}-s}{s_{in}-\hat{s}}\hat{K} - K\right] h\frac{dQ}{d\hat{K}} + \frac{s_{in}-s}{s_{in}-\hat{s}} hQ$$

Combining (C.1.14) and (C.1.16) the system gradient is

$$\psi(t+1) = \psi(t) - (u-d)h\psi_{\hat{s}}(t)\left[\frac{s_{in}-s}{s_{in}-\hat{s}}\right] + hu\psi_s(t) - hd\psi_{\hat{s}}(t) \qquad \text{(C.1.17)}$$

$$- \left[\frac{s_{in}-s}{s_{in}-\hat{s}}\hat{K} - K\right] h\frac{dQ}{d\hat{K}} - \left[\frac{s_{in}-s}{s_{in}-\hat{s}}\right] hQ\,.$$

This is similar to the gradient derived for Case 2, (C.1.11). While it cannot be implemented exactly as given due to unknowns, more terms in (C.1.17) can be implemented than for (C.1.11). The main difference is that $\psi_{\hat{s}}(t)$ from (C.1.14) does not depend on unknowns whereas in Case 2, (C.1.9), it did.

Further analysis of each of these gradients is given in Chapter 5.

REFERENCES

Agarwal, M. and D.E. Seborg (1987). "Self-tuning controllers for nonlinear systems", ***Automatica***, vol. 23, no. 2, pp. 209-214.

Alvarez, Je., J. Alvarez and E. Gonzalez (1989). "Global nonlinear control of a continuous stirred tank reactor", ***Chem. Engng. Sci.***, vol. 44, pp. 1147-1160.

Annaswamy, A.M., C. Thanomsat, N. Mehta, and A.P. Loh (1998). "Applications of adaptive controllers to systems with nonlinear parameterizations", ***ASME J. DYN. Syst. Meas. Control***, vol. 120, pp.477-487.

Aoufoussi, H., M. Perrier, J. Chaouki, C. Chavarie, and D. Dochain (1992). "Feedback linearizing control of a fluidized bed reactor", ***Canadian Journal of Chem. Eng.***, vol. 70, pp. 356-367.

Aseltine, J.A., A.R. Mancini, and C.W. Sarture (1958). "A survey of adaptive control systems", ***IRE Trans. Automat. Contr.***, vol. 3, pp. 102-108.

Astrom, K.J., and B. Wittenmark (1989). ***Adaptive Control***, Addison-Wesley, Reading, Massachusetts.

Bailey, J.E. and D.F. Ollis (1986). ***Biochemical Engineering Fundamentals***, McGraw-Hill, New York, N.Y.

Banks, S.P. (1988). ***Mathematical Theories of Nonlinear Systems***, Prentice-Hall, Englewood Cliffs, N.J.

Bartusiak, D.R., C. Georgakis and M.J. Reilly (1989). "Nonlinear feedforward/feedback control structures designed by reference system synthesis", ***Chem. Engng. Sci.***, vol. 44, no. 9, pp. 1837-1851.

Bastin, G. and D. Dochain (1990). ***On-line Estimation and Adaptive Control of Bioreactors***, Elsevier, New York, N.Y.

Bekey, G.A. and W.J. Karplus (1968). ***Hybrid Computation***, John Wiley & Sons, New York, N.Y.

Bell, D. (1988). in review of ***Algebraic and Geometric Methods in Nonlinear Control***, M. Fliess and M. Hazewinkel, eds., ***Automatica***, vol. 24, no. 4, pp. 586-587.

Boskovic, J.D. (1995). "Stable adaptive control of a class of first-order nonlinearly parameterized plants", ***IEEE Trans. Automat. Contr.***, vol. AC-40, no. 2, pp. 347-350.

Boskovic, J.D. (1998). "Adaptive control of a class of nonlinearly parameterized plants", ***IEEE Trans. Automat. Contr.***, vol. AC-43, no. 7, pp. 930-934.

Box, G.E.P. and J.F. MacGregor (1974). "The analysis of closed-loop dynamic stochastic systems", ***Technometrics***, vol. 15, pp. 380-391.

Brockett, R.W. (1978). "Feedback invariants for nonlinear systems", ***Preprint 7th IFAC World Congress***, pp 1115-1120, Helsinki, Finland.

Brockett, R.W. (1976). "Nonlinear systems and differential geometry", ***Proc. of the IEEE***, vol. 64, no. 1.

Chen, C-T. (1970). ***Introduction to Linear System Theory***, Holt, Rinehart, and Winston, New York, N.Y.

Chen, C-T. (1984). ***Linear System Theory and Design***, CBS College Publishing, New York, N.Y.

Cott, B.J. and S. Macchietto (1989). "Temperature control of exothermic batch reactors using generic model control", ***Ind. Eng. Chem. Res.***, vol. 28, no. 8, pp. 1177-1184.

Cunningham, W.J. (1958). ***Introduction to Nonlinear Analysis***, McGraw-Hill, New York, N.Y.

DeLuca, A. and G. Ulivi (1987). "Full linearization of induction motors via nonlinear state feedback", ***Proc. Of The 26'th Conf. Decision and Contr.***, Los Angeles, Ca., pp. 1765-1770.

Dochain, D. (1986) ***On-line Parameter Estimation, Adaptive State Estimation, And Adaptive Control Of Fermentation Processes***, Ph.D. thesis, Universite Catholique de Louvain, 1348 Louvain-la-Neuve, Belgium.

Drenick, R.F. and R.A. Shahbender (1957). " Adaptive servo-mechanisms", ***AIEE Transactions***, vol. 76, pp 286-292.

Ferrara, A. and L. Giacomini (2001). "Output feedback second-order sliding mode control for a class of nonlinear systems with non-matched uncertainties", ***ASME J. Dyn. Syst. Meas. Control***, vol. 123, pp.317-323.

Fnaiech, F., and L. Ljung (1987). "Recursive identification of bilinear systems", ***Int. J. Control***, vol. 45, no. 2, pp. 453-470.

Gebo, C.H. (1983). "Extruder barrel temperature control", internal report, Eastman Kodak Co., Rochester, N.Y.

Gelb, A. and W.E. Vander Velde (1968). ***Multiple-Input Describing Funtions And Nonlinear System Design***, McGraw-Hill, New York, N.Y.

Goodwin, G.C., and K.S. Sin (1984). ***Adaptive Filtering Prediction and Control***, Prentice-Hall, Englewood Cliffs, N.J.

Graham, D. and D. McRuer (1961). ***Analysis of Nonlinear Control Systems***, John Wiley & Sons, New York, N.Y.

Grayson L.P. (1963). "Design via Lyapunov's second method", ***Proceedings of The 4th Joint Automat. Contr. Conf.***, Minneapolis, Minnesota.

Gregory, P.C., ed. (1959). ***Proceedings of The Self-adaptive Flight Control Symposium***, WADC Tech. Rep. 59-49, Wright-Patterson Air Force Base, Dayton, Ohio.

Grimble, M. (1994). ***Robust Industrial Control***, Prentice-Hall, Englewood Cliffs, N.J.

Gustavsson, I., L. Ljung, and T. Soderstrom (1977). Survey Paper, "Identification of processes in closed loop: identifiability and accuracy aspects", ***Automatica***, vol. 13, pp. 59-75.

Han, Y. (1992). ***Adaptive Tracking In Feedback Linearizable Systems***, Ph.D. thesis, Department of Electrical and Computer Engineering, McMaster University, Hamilton, Ontario, Canada.

Harris, C.J. and S.A. Billings eds. (1981). ***Self-tuning and Adaptive Control: Theory and Applications***, Peter Peregrinus, London.

Hunt, L.R., R. Su and G. Meyer (1983). "Global transformation of nonlinear systems", ***IEEE Trans. Automat. Contr.***, vol. AC-28, pp. 24-34.

Ilic'-Spong, M., R. Marino, S.M. Peresada, D.G. Taylor (1987). "Feedback linearizing control of switched reluctance motors", ***IEEE Trans. Automat. Contr.***, vol. AC-32, no. 5.

Ioannou, P.A. and J. Sun (1996). ***Robust and Adaptive Control***, Prentice Hall, Upper Saddle River, N.J.

Isermann, R. (1981). ***Digital Control Systems***, Springer-Verlag, New York, N.Y.

Isidori, A. (1995). ***Nonlinear Control Systems, An Introduction***, 2nd edition, Springer-Verlag, New York, N.Y.

Jain, S., F. Khorrami, B. Fardanesh (1994). "Adaptive nonlinear excitation control of power systems with unknown interconnections", ***IEEE Trans. Contr. Syst. Tech.***, vol. 2, no. 4, pp. 436-446.

Jiang, Z.P. and D.J. Hill (1999). "A robust adaptive backstepping scheme for nonlinear systems with unmodeled dynamics", ***IEEE Trans. Automat. Contr.***, vol. AC-44, no. 9, pp. 1705-1711.

Johnson, C. R. (1988). ***Lectures On Adaptive Parameter Estimation***, Prentice-Hall, Englewood Cliffs, N. J.,

Kailath, T. (1980). ***Linear Systems***, Prentice-Hall, Englewood Cliffs, N. J.

Kalman, R.E. (1958). "Design of a self-optimizing control system", ***Trans. ASME***, vol. 80, pp. 468-478.

Kalman, R.E. (1963). "Mathematical description of linear dynamical systems", ***SIAM J. Control***, 1, pp. 152-192.

Kanellakopoulos, I. (1991). ***Adaptive Control of Nonlinear Systems***, Ph.D thesis, University of Illinois, Urbana, IL.

Kanellakopoulos, I., P.V. Kokotovic and A.S. Morse (1991). "Systematic design of adaptive controllers for feedback linearizable systems", ***IEEE Trans. Automat. Contr.***, vol. AC-36, no.11, pp.1241-1253.

Khalil, H.K. (2002). ***Nonlinear Systems***, 3rd ed., Prentice-Hall, Upper Saddle River, N.J.

Kojic, A. and A.M. Annaswamy (2002). "Adaptive control of nonlinearly parameterized systems with a triangular structure", ***Automatica***, vol. 38, pp. 115-123.

Kravaris, C. and C-B. Chung (1987). "Nonlinear state feedback synthesis by global input/output linearization", ***AIChE Journal***, vol. 33, no. 4, pp. 592-603.

Krener, A.J. (1973). "On the equivalence of control systems and the linearization of nonlinear systems", ***SIAM J. Control***, 11:670.

Krstic, M., I. Kanellakopoulos and P.V. Kokotovic (1994). "Nonlinear design of adaptive controllers for linear systems", ***IEEE Trans. Automat. Contr.***, vol. AC-39, no. 4, pp.738-751.

Krstic, M., I. Kanellakopoulos, and P. Kokotovic (1995). ***Nonlinear and Adaptive Control Design***, John Wiley and Sons, Inc., New York, N.Y.

LaSalle, J. and S. Lefschetz (1961). ***Stability By Liapunov's Direct Method***, Academic Press

Lastman, G.J., and N.K. Sinha (1989). ***Microcomputer-Based Numerical Methods for Science and Engineering***, Saunders College Publishing, A Division of Holt Rinehart and Winston, Inc., New York, N. Y.

Lee, P.L., and G.R. Sullivan (1988). "Generic model control (GMC)", ***Computers and Chemical Engineering***, vol. 12, no. 6, pp. 573-580.

Leigh, J.R. (1983). ***Essentials of Nonlinear Control Theory***, Peter Peregrinus Ltd., London, UK.

Ljung, L. (1979). "Asymptotic behavior of the extended Kalman filter as a parameter estimator for linear systems" ***IEEE Trans. Auto. Control***, vol. AC-24, no. 1, pp. 36- 50.

Ljung, L. and T. Soderstrom (1983). ***Theory and Practice of Recursive Identification***, The MIT Press, Cambridge, Mass.

Ljung, L. (1987). ***System Identification: Theory for The User***, Prentice-Hall, Englewood Cliffs, N.J.

Luh, J.Y.S., W.D. Fisher, and R.P.C. Paul (1983). "Joint torque control by a direct feedback for industrial robots", ***IEEE Trans. Automat. Contr.***, vol. AC-28, no. 2, pp. 153-161.

Ma, X. and G. Tao (2000). "Adaptive actuator compensation control with feedback linearization", ***IEEE Trans. Automat. Contr.***, vol. AC-45, pp. 1705-1710.

Marino, R., S. Peresada and P. Valigi (1993). "Adaptive input-output linearizing control of induction motors", ***IEEE Trans. Automat. Control***, vol. 38, pp. 208-221.

McLellan, P.J., T.J. Harris and D.W. Bacon (1990). "Error trajectory descriptions of nonlinear controller designs", Department of Chemical Engineering., Queen's University, Kingston, Ontario, Canada.

Melerio, L.A.C. and R. Filho (2000). "A self-tuning adaptive control applied to an industrial large scale ethanol production", ***Computers and Chemical Engineering***, vol. 24, pp. 925-930.

Meyer, G. and L. Cicolani (1980). "Application of nonlinear system inverses to automatic flight control design-systems concepts and flight evaluations", AGARDograph 251 on ***Theory and Applications of Optimal Control in Aerospace Systems***, P. Kent, Ed., reprinted by NATO.

Moore, J.B., and H. Weiss (1979). "Recursive prediction error methods for adaptive estimation",***IEEE Trans. on Systems Man and Cybernetics***, vol. SMC-9, no. 4, pp. 197-205.

Nam, K. and A. Arapostathis (1988). "A model reference adaptive control scheme for pure-feedback nonlinear systems", ***IEEE Trans. Automat. Contr.***, vol. AC-33, no. 9, pp.803-811.

Narendra, K.S., and A.M. Annaswamy (1989). ***Stable Adaptive Systems***, Prentice-Hall, Englewood Cliffs, New Jersey.

Narendra, K.S. and R.V. Monopoli eds. (1980). ***Applications of Adaptive Control***, Academic Press, New York, N.Y.

Narendra, K.S., R. Ortega, and P. Dorato eds. (1991). ***Advances In Adaptive Control***, IEEE Press, New York, N.Y.

Nijmeijer, H., and A.J. van der Schaft (1990). ***Nonlinear Dynamical Control Systems***, Springer-Verlag, New York,

Parks, P.C. (1966). "Lyapunov redesign of model reference adaptive control systems", ***IEEE Trans. Automat. Contr.***, vol. AC-11, pp. 362-367.

Pulley R.A. (1986). "Batch process modelling club report CR2828(CON)", Warren Spring Laboratory, Stevenage, Herts, U.K.

Rosenbrock, H.H. (1970). ***State-Space and Multivariable Theory***, Wiley, New York, N.Y.

Sastry, S. (1999). ***Nonlinear Systems Analysis, Stability and Control***, Springer-Verlag, New York, N.Y.

Sastry, S. and A. Isidori (1989). "Adaptive control of linearizable systems", ***IEEE Trans. Automat. Contr.***, vol. AC-34, no. 11, pp. 1123-1131.

Sastry, S. and M. Bodson (1989). ***Adaptive Control: Stability, Convergence, and Robustness***, Prentice-Hall, Englewood Cliffs, New Jersey.

Shinskey, F.G. and J.L. Weinstein (1965) "A dual-mode control system for a batch exothermic reactor", ***Twentieth Annual Inst. Soc. Amer. Conference***, Los Angeles, Ca.

Sinha, N.K., and B. Kuszta (1983). ***Modeling and Identification of Dynamic Systems***, Van Nostrand Reinhold, New York, N.Y.

Slotine, J.E. and W. Li (1991). ***Applied Nonlinear Control***, Prentice-Hall, Englewood Cliffs, N.J.

Soderstrom, T., and P. Stoica (1989). ***System Identification***, Prentice-Hall, Englewood Cliffs, N.J.

Soroush, M. and C. Kravaris (1991). "Nonlinear control of a batch polymerization reactor", ***Proc. Amer. Contr. Conf.*** ,Boston, p. 2879.

Spong, M.W., and M. Vidyasager (1989). ***Robot Dynamics and Control***, John Wiley & Sons, New York, N.Y.

Stromer, P. R. (1959). "Adaptive or self-optimizing control systems - a bibliography", ***IRE Trans. Automat. Contr.***, vol. 4, pp. 65-68.

Su, R. (1982). "On the linear equivalents of nonlinear systems", ***Syst. Contr. Lett.***, vol. 2, pp. 48-52.

Su, R. and L.R. Hunt (1986). "A canonical expansion for nonlinear systems", ***IEEE Trans. Automat. Contr.***, vol. AC-31, no. 7, pp. 670-673.

Thaler, G.J., and M.P. Pastel (1962). ***Analysis and Design of Nonlinear Feedback Control Systems***, McGraw-Hill, New York, N.Y.

Truxal, J.G. (1955). ***Automatic Feedback Control System Synthesis***, McGraw-Hill, New York, N.Y.

Viel, F., F. Jadot and G. Bastin (1997). "Robust feedback stabilization of chemical reactors", ***IEEE Trans. Automat. Contr.***, vol. 42, no. 4, pp. 473-481.

Zhang, T., S.S. GE, C.C. Hang and T.Y. Chai (2000). "Adaptive control of first order systems with nonlinear parameterizations", ***IEEE Trans. Automat. Cont.***, vol. AC-45, no. 8, pp. 1512-1516.

Zhou, K., J.C. Doyle and K. Glover (1996). ***Robust and Optimal Control***, Prentice-Hall, Englewood Cliffs, N.J.

Zhou, K., and J.C. Doyle (1998). ***Essentials of Robust Control***, Prentice-Hall, Englewood Cliffs, N.J.