Dual adaptive control for linear systems

He, Yusheng, Ph.D.

City University of New York, 1989

U·M·I
300 N. Zeeb Rd.
Ann Arbor, MI 48106

H

DUAL ADAPTIVE CONTROL FOR LINEAR SYSTEMS

by

YUSHENG HE

A dissertation submitted to the Graduate Faculty in Engineering in partial fulfillment of the requirement for the degree of Doctor of Philosophy, The City University of New York.

1989

This manuscript has been read and accepted for the Graduate Faculty in Engineering in satisfaction of the dissertation requirement for the degree of Doctor of Philosophy.

8-24-89 — F E Thau

Date — Chair of Examining Committee

8/24/89 — Jacques E. Benveniste

Date — Executive Officer

Prof. G. Kranc

Prof. J. Barba

Prof. K. Sobel

Prof. F. E. Thau (Chairman)

Supervisory Committee

The City University of New York

ACKNOWLEDGEMENTS

I wish to thank Professor Frederick E. Thau for providing me with the opportunity to explore this topic and for his invaluable assistance in obtaining the desired results. Special thanks to my parents and my wife for their continual support and encouragement during this endeavor.

Y. S. H.

TABLE OF CONTENTS

List of Figures

Figure	Title	Page
Fig. 4.1	The root loci of (4.15) for case 1	... 80
Fig. 4.2	The root loci of (4.15) for case 2	... 81
Fig. 5.1	A comparison of the system outputs between different dual control techniques	... 101
Fig. 5.2	A comparison of the system parameter estimates between different dual control techniques	... 102
Fig. 6.1	The root locus for case one in (6.16)	... 116
Fig. 6.2	The root locus for case two in (6.19)	... 116
Fig. 6.3	The comparison of the system outputs between the dual adaptive control and the conventional pole-placement control for a nonminimum phase system	... 128
Fig. 6.4	The comparison of the estimated parameters between the dual adaptive control and the conventional pole-placement control for a nonminimum phase system	... 129
Fig. 6.5	A comparison of the system outputs between the dual adaptive control and the conventional pole-placement control for the same system as above with the system gain change at k=230	... 130

Figure	Title	Page

Chapter One

Introduction

1.1) Introduction

Many different proposals for adaptive control have been made in the past. However, their application, mostly based on analog realization, was not very successful and was unconvincing until the early 1970s. The development of cheaper and reliable digital computers has reactivated interest in the field of adaptive control. Because good results have been reported in some applications, [1, 2, 3, 4, 5 and 30], various adaptive control algorithms have received much attention in the last few years.

Many results in modern control theory require a description of the system in terms of differential or difference equations. If the system has unknown parameters, the parameters of the system may have to be estimated.

On-line parameter estimation combined with on-line control leads to the indirect adaptive controller shown in Fig. 1.1.

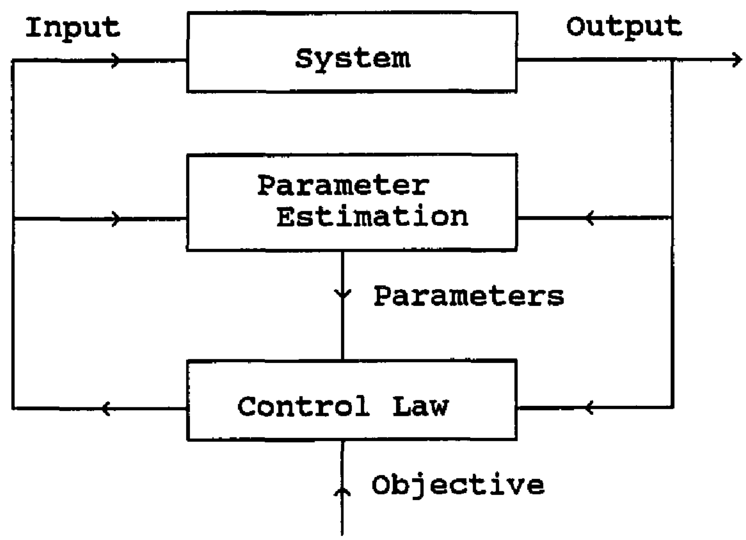


Fig.1.1 Basic structure of an adaptive controller.

From the above discussion, it can be seen that parameter estimation forms an integral part of adaptive control. It is natural to ask: does there exist an input signal that plays a dual role, ie. on the one hand, driving the system toward a desired state; on the other hand, perturbing the system enough to allow good parameter identification? The answer is positive, such an input is called a dual control. To provide a setting for the following discussion of dual control, some background on parameter estimation and adaptive control and some of dual control techniques are summarized below.

1.2) Previous research

- General parameter estimation techniques, summarized in chapter 2, include the projection algorithm and least squares algorithm. Both algorithms are recursive. The first one is simple but with slow convergent speed. The last one has faster convergent speed compared with the projection algorithm. The estimates will converge to the true parameters for both algorithms only provided persistent excitation. For an adaptive control system, the condition of persistent excitation is difficult to verify.

- Some popular adaptive control schemes are presented in chapter two. All those schemes, eg. self tuning, model reference [5] and pole placement adaptive controls, treat the estimated parameters as if they were true ones. So there is no consideration to improve the parameter estimation for the above control schemes.

- Some dual control schemes are discussed in chapter 5 [18, 20 and 21]. All those dual adaptive control are limited to work with the least squares parameter estimation algorithm. Goodwin's technique [20] may cause the system to become unstable. No stability proof is available for all of the above dual control schemes.

1.3) Major contributions

The major contributions in this thesis are as follows:

- A combined performance index (in chapter 3), with time varying scalars α and λ, provides some flexibility to suit different parameter estimation schemes. And it also provides a compromise between control and estimation by properly selecting α and λ.

- The stability proof of the new designed dual control scheme is presented in chapter 4 (for minimum phase systems) and chapter 6 (for nonminimum phase systems). No stability proof has been presented in the literature yet. Especially, the dual control for nonminimum phase systems is also the first attempt to apply dual control for this class of systems in the literature.

- The approximately 'error free' parameter estimation algorithm for constant unknown parameters is presented in chapter 7. The parameter error will approach zero after n+m steps. This multistep parameter estimation scheme may provide a new way for adaptive control in that the estimated parameters will converge to the true ones in a finite number of steps.

1.4) Organization of the thesis

Chapter 2 is a summary of the background on parameter estimation and adaptive control. Conditions of convergence of the parameter estimation algorithms are also presented in chapter 2.

In chapter three, a comprehensive example of dual control is presented in order to demonstrate properties of dual control.

Following the example, a dual control approach is presented which is based on the minimization of a performance function V(k), that is a sum of a component $V_c(k)$ whose minimization with respect to the plant input would result in a one-step-ahead self tuning controller, and a component $V_1(k)$ whose minimization with respect to the plant input would result in a reduction of the a posteriori output modeling error. The optimal control law can be found from the solution of a high-order polynomial equation in the unknown control u(k). Some extensions and simulations show how dual control improves the future behavior of the system. A stability proof for minimum phase systems is presented in chapter four. Chapter five presents a comprehensive comparison of the designed dual control algorithm and other current dual control algorithms.

Chapter six extends this algorithm to more general cases by using a pole-placement technique. Thus the extended algorithm is applicable to nonminimum phase systems. Since the designed dual control law is a suboptimum control law (one-step optimum control) and the parameter estimation scheme is one of the conventional schemes, the designed dual control algorithm still can not guarantee that the parameter estimates converge to the true parameters of the system. In the last chapter, a new parameter estimation scheme is introduced, so that the parameter estimation error will converge to zero in a finite number of steps. This scheme is named an 'error free' parameter estimation algorithm.

Chapter Two

Background on Adaptive Control

2.1) Introduction

A brief summary of linear adaptive control strategies along with different parameter estimation techniques is presented in this chapter. A linear deterministic dynamical system in which parameters are unknown is considered throughout this thesis. It can be described by the following simple expression:

$$y(k) = h^T(k-1)\ \theta \qquad \ldots (2.1)$$

where $y(k)$ denotes the system output at time k.

$h(k-1)$ denotes a regression vector that is

$$h^T(k-1) = [\ y(k-1),\ldots,y(k-n),u(k-1),\ldots,u(k-m)\ ]$$

θ denotes an unknown parameter vector.

Based on the model (2.1), we introduce some parameter estimation schemes which are very popularly used by control engineers. Also they are useful in deriving a new adaptive control algorithm and in making comparison with existing adaptive control algorithms.

2.2) Parameter Estimation Algorithms

- **Projection Algorithm**

The projection algorithm

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \frac{h(k-1)}{h^T(k-1)h(k-1)}[y(k) - h^T(k-1)\hat{\theta}(k-1)] \quad \ldots (2.2)$$

results from the following optimization problem. Given $\hat{\theta}(k-1)$ and $y(k)$, determine $\hat{\theta}(k)$ so that

$$J = \frac{1}{2} ||\hat{\theta}(k) - \hat{\theta}(k-1)||^2 \quad \ldots (2.3)$$

is minimized subject to

$$y(k) = h^T(k-1)\ \hat{\theta}(k) \quad \ldots (2.4)$$

The proof can be found in [5].

An alternative scheme for avoiding division by zero is given as following:

$$\hat{\theta}(k)=\hat{\theta}(k-1) + \frac{a\ h(k-1)}{c + h^T(k-1)h(k-1)} [y(k)-h^T(k-1)\hat{\theta}(k-1)]$$

with $c \geq 0$ and $0 < a < 2$. ... (2.5)

- Least-Squares Algorithm [5, 6 and 7]

For the least-squares algorithm, the criterion

$$J = \frac{1}{2} \sum_{k=1}^{N} [\ y(k) - h^T(k-1)\theta\ ]^2$$

... (2.6)

is minimized with respect to θ to give the estimate $\hat{\theta}(k)$.

A recursive scheme for calculating $\hat{\theta}(k)$ is

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \frac{P(k-1)h(k-1)}{1 + h^T(k-1)P(k-1)h(k-1)}[y(k)-h^T(k-1)\hat{\theta}(k-1)]$$

$$P(k) = P(k-1) + \frac{P(k-1)h(k-1)h^T(k-1)P(k-1)}{1 + h^T(k-1)P(k-1)h(k-1)}$$ ... (2.6')

where $\hat{\theta}(0)$ is given and $P(0)$ is any positive definite matrix.

There are some different modified versions of the least-squares algorithm [5]. We will not introduce them here because of space limitations.

The least-squares algorithm generally has much faster convergence than the projection algorithm. Also the least-squares algorithm can be used essentially unaltered with noisy signals. The advantage of the projection algorithm is that it is computationally simpler than the least-squares algorithm.

■ Parameter Convergence

i) The projection algorithm

The projection algorithm is globally exponentially convergent to the true parameters provided that the following condition is satisfied:

$$\sum_{i=0}^{j-1} \frac{h(k+i)h^T(k+i)}{h^T(k+i)h(k+i)} \geqslant aI \qquad a > 0 \qquad \ldots (2.7)$$

for all k and some fixed $j > 0$, I is the identity matrix (see [5]).

An additional condition for convergence of the modified projection algorithm (2.5) is $c \geq 0$ and $0<a<2$.

ii) The Least-Squares Algorithm

A basic property of the least-squares algorithm [5, 30] is that the error function V(k), defined below, is positive and nonincreasing, and hence converges where

$$V(k) = \tilde{\theta}^T(k) P^{-1}(k) \tilde{\theta}(k)$$

$$\tilde{\theta}(k) = \theta - \hat{\theta}(k)$$

Also, from the matrix inversion lemma [4] we have that

$$P^{-1}(k) = P^{-1}(0) + \sum_{i=1}^{k} h(i) h^T(i) \qquad \ldots (2.8)$$

It is clear that

$$V(k) \geq [\lambda_{min} P^{-1}(k)] \tilde{\theta}^T(k) \tilde{\theta}(k) \qquad \ldots (2.9)$$

From equation (2.9) we can see that $\tilde{\theta}(k)$ converges to zero provided that

$$\lim_{k \to \infty} [\lambda_{min} P^{-1}(k)] = \infty \qquad \ldots (2.10)$$

where $\lambda_{min} P^{-1}(k)$ is the minimum eigenvalue of $P^{-1}(k)$. From (2.8), (2.10) is guaranteed by

$$\lim_{k\to\infty} \lambda_{min}\{ \sum_{i=1}^{k} h(i-1)h^T(i-1) \} = \infty \qquad \ldots (2.11)$$

■ Persistent Excitation [8]

From equations (2.7) - (2.11) we can see that the convergence conditions are closely related to conditions on the input sequence $\{u(k)\}$. This condition, called "persistent excitation" is defined as follows:

Definition 2.1) An input sequence $\{u(k)\}$ is said to satisfy a persistency of excitation condition if for all k, there exists an integer L such that

$$\rho_1 I > \sum_{k}^{k+L} \begin{pmatrix} u(k+n) \\ u(k+n-1) \\ \vdots \\ \vdots \\ u(k+1) \end{pmatrix} [u(k+n) \ \ldots \ u(k+1)] > \rho_2 I \qquad \ldots (2.12)$$

where ρ_1 , $\rho_2 > 0$ and I is the identity matrix.

For deterministic systems, (1) The estimates generated by the projection algorithm are exponentially convergent provided that the system input is persistently exciting. (2) The estimates generated by the least-squares algorithm converge provided that the system input is persistently exciting [8].

In addition to the algorithms discussed above, there are many other techniques for parameter identification. We will not list them here; readers who are interested in this subject may refer to [3-9, and 30].

2.3) Adaptive Control Algorithms

The general block diagram of an indirect adaptive control system is shown in Fig. 1.1. A very natural approach to the design of an adaptive control system is to combine a particular parameter estimation technique with any asymptotically stable or quasi-optimum control law. This approach of using the estimates as if they were the true parameters for the purpose of design is called certainty equivalence adaptive or self-tuning control. The simplest conceptual scheme is when the system is parameterized in a natural way and the design calculations are carried out based on the estimated system model. This class of

algorithms is commonly called indirect since the evaluation of the control law is indirectly achieved via the system model. It is possible to parameterize the system directly in terms of the control law parameters. This class of algorithms is commonly called direct since the control law is directly estimated.

Three schemes for adaptive control -- model reference control, self-tuning adaptive control and pole-placement adaptive control-- are described briefly in this section.

- Self-Tuning Control

Instead of discussing general self-tuning control, we use a particular case (one-step-ahead control) to illustrate the self-tuning control. As we mentioned before, we shall assume that the system is described by a deterministic autoregressive moving average [DARMA] model of the form

$$y(k) = A'(q^{-1})\ y(k-1) + B'(q^{-1})\ u(k-1) \qquad \ldots (2.13)$$

where $A'(q^{-1})$ and $B'(q^{-1})$ are given by

$$A'(q^{-1}) = a_1 + a_2 q^{-1} + \ldots + a_n q^{-n+1}$$

$$B'(q^{-1}) = b_1 + b_2 q^{-1} + \ldots + b_m q^{-m+1}; \qquad b_1 \neq 0$$

q^{-1} is the delay operator.

We are given a desired output sequence $\{y^*(k)\}$ and our objective is to design an adaptive control law to achieve closed loop stability and to asymptotically achieve zero tracking error, that is

$$\lim_{k\to\infty} [\ y(k) - y^*(k)\ ]^2 = 0$$

The time-varying feedback control law is calculated from minimizing the prediction error, ie.

$$V(k) = [\ y(k) - y^*(k)\ ]^2 \qquad \ldots (2.14)$$

is minimized with respect to u(k-1) and yields

$$u(k-1) = -\frac{1}{b_1}[a_1 y(k-1) + \ldots + a_n y(k-n) + b_2 u(k-2) +$$

$$+ \dots + b_m u(k-m) - y^*(k)] \qquad \dots (2.15)$$

The modified projection algorithm (2.5) is used for parameter estimation. Goodwin [5] established the global convergence result for this one-step-ahead algorithm using the projection estimator, under assumptions i) Upper bounds for the orders of the polynomials in (2.13) are known; ii) All modes of the inverse of the model (2.13) (i.e., the zeros of the polynomial $B'(q^{-1})$) lie inside or on the closed unit circle.

The general self-tuning control was studied by several authors [2, 10-12]. In this chapter, we will not discuss it further.

▪ Model Reference Adaptive Control

A block diagram of model reference adaptive control is shown in Fig. 2.1. The model reference adaptive control was developed by different authors [13-16 and 25]. Consider the SISO discrete system described by (2.13).

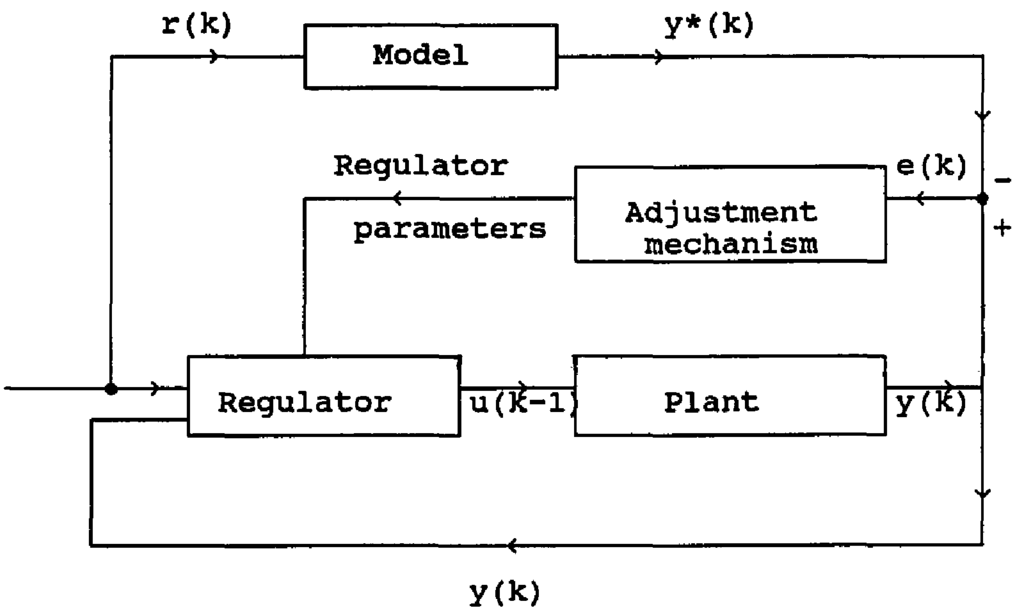


Fig. 2.1 Block diagram of model reference adaptive control

The desired output y*(k) satisfies the following reference model:

$$E(q^{-1})\ y^*(k) = q^{-1}H(q^{-1})\ r(k) \qquad \ldots (2.16)$$

where $E(q^{-1}) = e_0 + e_1 q^{-1} + \ldots + e_j q^{-j}$

$$\ldots (2.17)$$

$H(q^{-1}) = h_0 + h_1 q^{-1} + \ldots + h_j q^{-j}$

and $E(q^{-1})$ is assumed stable. The system (2.13) can be expressed in predictor form [since the objective was $y(k+1)=y*(k+1)$]

$$E(q^{-1})y(k+1) = \alpha(q^{-1})y(k) + \beta(q^{-1})u(k) \quad \ldots (2.18)$$

where

$$\alpha(q^{-1}) = G(q^{-1}) = q^{-1}H(q^{-1})/E(q^{-1})$$

$$\beta(q^{-1}) = F(q^{-1})B(q^{-1})$$

and $F(q^{-1})$ and $G(q^{-1})$ are the unique polynomials, satisfying

$$E(q^{-1}) = F(q^{-1})[1-A(q^{-1})] + q^{-1} G(q^{-1}) \quad \ldots (2.19)$$

Based on the result above [in particular, (2.18)] it is clear that the design objective can be obtained by setting u(k) according to

$$\alpha(q^{-1})\ y(k) + \beta(q^{-1})u(k) = H(q^{-1})r(k) \quad \ldots (2.20)$$

The feedback control law (2.20) depends on the knowledge of the system parameters. When this knowledge is unavailable a priori, any adaptive control algorithm can be employed, for example, one - step - ahead adaptive controller (see [5]).

Global convergence of the above model reference adaptive control by using projection estimation algorithm can be obtained in the same way as for a one-step-ahead controller with an additional constraint on parameter estimation, ie. to ensure that the estimate $1/b_1 \neq 0$ [5].

- Pole-placement Adaptive Control:

Another adaptive control scheme: pole-placement adaptive control, which may be used for nonminimum phase systems [25] is discussed briefly in this section. For simplicity we still use the SISO DARMA (2.1) model to discuss the pole placement adaptive control algorithm. Consider a controlled system in the following form

$$A(q^{-1})y(k) = B(q^{-1})u(k) = q^{-1}B'(q^{-1})u(k) \qquad \ldots (2.21)$$

where $$A(q^{-1}) = 1-q^{-1}A'(q^{-1})$$

and $A'(q^{-1})$ and $B'(q^{-1})$ are defined in (2.13).

The following assumptions were used to derive the pole-placement algorithm [5]:

1) The order of the system (2.21) n is known.

2) $A(q^{-1})$ and $B(q^{-1})$ are relative prime (but having unknown coefficients).

The assumption 2) is to ensure that there exists unique solution of polynomials $P(q^{-1})$ and $Q(q^{-1})$ such that

$$P(q^{-1})B'(q^{-1})+Q(q^{-1})A(q^{-1})=A* \qquad \ldots (2.22)$$

where $A*$ is an arbitrary stable polynomial of order 2n-1. The proof can be found in [5].

The implication of the above equation is that the input u(k-1) may be generated by the following control law

$$Q(q^{-1})u(k-1) = P(q^{-1})[y(k)-y*(k)] \qquad \ldots (2.23)$$

where $y*$ is an arbitrary but bounded reference signal and the resulting closed-loop system has the characteristic polynomial $A*$. Unfortunately, to calculate u(k-1) from (2.23) the system parameters must be known. In order to implement, a parameter estimation algorithm has to be used each step. That leads to an adaptive pole placement control.

Adaptive pole placement controller:

The projection parameter estimation is described by (2.5) and $\hat{\theta}$ has the following form:

$$\hat{\theta}(k) = [\hat{a}_1, \ldots, \hat{a}_n, \hat{b}_1, \ldots, \hat{b}_m] \qquad \ldots (2.24)$$

$\hat{A}$ and $\hat{B}$ are the polynomials with the estimated parameters. The input u(k-1) is then determined by solving the pole placement equation at each time instant

$$\hat{P}(q^{-1})\hat{B}(q^{-1})+\hat{Q}(q^{-1})\hat{A}(q^{-1}) = A^* \qquad \ldots (2.25)$$

and then $\hat{P}$ and $\hat{Q}$ are used to calculate the following control law:

$$\hat{Q}(q^{-1})u(k-1) = \hat{P}(q^{-1})[h(k-1)^T\hat{\theta}(k-1)-y^*] \qquad \ldots (2.26)$$

The solvability of the pole placement equation (2.25) is a key problem. Difficulties arise when $\hat{A}$ and $\hat{B}$ may not be relative prime at some time instant. However, stable pole-zero cancellation does not present a problem since one can simply include the common roots in the closed loop polynomial, A*. The possibility of an unstable pole-zero cancellation must be avoided. If the system parameters are known sufficiently so that the initial estimation lies in a convex region in parameter space surrounding θ such that, within this region, $\hat{A}$ and $\hat{B}$ are relative prime, the estimated parameters $\hat{A}$ and $\hat{B}$ can be ensured that they will remain inside the region as k>0. Another way of avoiding having such difficulty is shown in [27].

The pole-placement adaptive control algorithm has the following properties:

1. {u(k)} and {y(k)} are bounded

2. The closed-loop characteristic polynomial tends to A* in the sense that

$$\lim_{k\to\infty} [A^*y(k)-\hat{P}(q^{-1})\hat{B}(q^{-1})y^*] = 0 \qquad \ldots (2.27)$$

The proof can be found in [5].

2.4) Applications

There have been a number of applications of adaptive control since the mid-fifties. Experiments with adaptive flight control system were done early in 1960. Industrial experiments with self-tuners were performed in 1972. Full scale experiments with adaptive autopilots for ship steering were made in 1977. Some process control loops have been running continuously since 1974. It is, however, only in the eighties that the adaptive techniques are starting to have real impact on industry. There are several commercial autopilots for ship steering, and motor drives and adaptive systems for industrial robots [5 and 30]. A rough estimate indicates that according to [30] in May of 1988 there were at least 50,000 loops where adaptive techniques are used [30].

Adaptive techniques are, however, not widely used. This means that the technology is still not mature nor is cost-effective. One reason is that simple robust methods for automatic tuning have not become available.

2.5) Summary

Before ending this section we have to emphasize the advantages of the adaptive control schemes mentioned above in particular, the one-step-ahead self-tuning regulator, are: i) These controls can achieve a good output tracking for some linear systems. ii) Analytic convergence and stability proof is available. iii) With the weighted one-step-ahead control, a compromise is made between output tracking and the size of the control effort. iv) The pole - placement control can be applied to nonminimum phase systems.

Chapter three

DUAL ADAPTIVE CONTROL

3.1) Introduction

In a control problem concerning an unknown process, the input signal plays two conflicting roles. On the one hand it drives the system toward some desired state (regulation); on the other hand, it must perturb the system enough to allow good parameter identification (learning). Feldbaum [17] was one of the first to point out this dual nature of the control. It is often the case that the two roles of the control signal are conflicting and thus the controller must achieve an optimal compromise between learning (which may require large perturbation) and regulation (which may only need a relatively small signal).

Since dual control takes account of control and identification at the same time, the approach has the potential to improve the control system performance over regular certainty equivalence adaptive controls. For example, a self-tuning control law, which is a certainty equivalence control, treats the estimated parameters as if they were the true values, even though the estimated parameters may be completely different from the true values. Secondly, certainty equivalence control ignores the caution and learning aspects of the optimal controller. Thus some unexpected phenomena, for example, control "turn off", may occur during the adaptive control operation. Dual control may give a better trade-off between control and estimation.

Since the work of Feldbaum, many research efforts have been dedicated to the dual control problem. There are two classes of dual control suggested in the literature. In the first class, the dynamic programming equations are formulated [23] and the analysis results in a very complicated and intractable computational problem.

However obtaining a useful numerical result in the available computational time is usually beyond our capability. In the second class, the optimization problem is reformulated to include an explicit penalty for the identification error of the controller. In this case, the compromise between estimation and control is enforced by a reformulated optimization problem. The various formulations of this second approach include the following: Wittenmark [18] considers a one - step - ahead control objective, appending a term which is a function of the covariance of the parameter error vector. Belanger [19] also uses a one-step-ahead control subject to a constraint imposed on the trace of the covariance matrix of the parameter error vector. Goodwin [20] considers the ratio of the determinants of the parameter error covariance at two consecutive instants. Padilla [21] introduces an innovation variance to the control performance to compromise between control and estimation.

The second class of dual control is considered to be suboptimal because the optimum control is obtained for one step only. The suboptimal solutions are much easier to calculate and to implement than those based on dynamic programming.

In the following section, a demonstration of a dual adaptive control and a comparison with a self-tuning regulator (certainty equivalence adaptive control) will be presented.

Section 3.3) will introduce a new combined performance index based in part on minimizing the ratio of the system's a posteriori output error to the system's a priori output error (see section 3.3). The optimal control law can be found from the solution to a fifth order polynomial equation in the unknown control u(k). This leads to a numerical root-searching problem. Another way to build up a dual-role performance index is to maximize a scalar function of the correction gain of the parameter estimation vector. These two formulations are quite similar. In section 3.4) some modifications are made based on these two formulations in order to i) fit some other parameter estimation algorithms; ii) simplify the calculations required in control law implementation.

Sections 3.4) and 3.5) present some analysis and simulations to show how these formulations work and how they improve the future behavior of the system. Simulation results compare the performance of the new dual-adaptive system with other control algorithms.

In order to show that dual control results in better performance than a regular self-tuning regulator, the following section uses a comprehensive example to demonstrate the properties of dual control stated above.

3.2) Comparison of Self-tuning Control and Dual Control

As an example of dual control, let's consider an adaptive control system with a diagram as follows

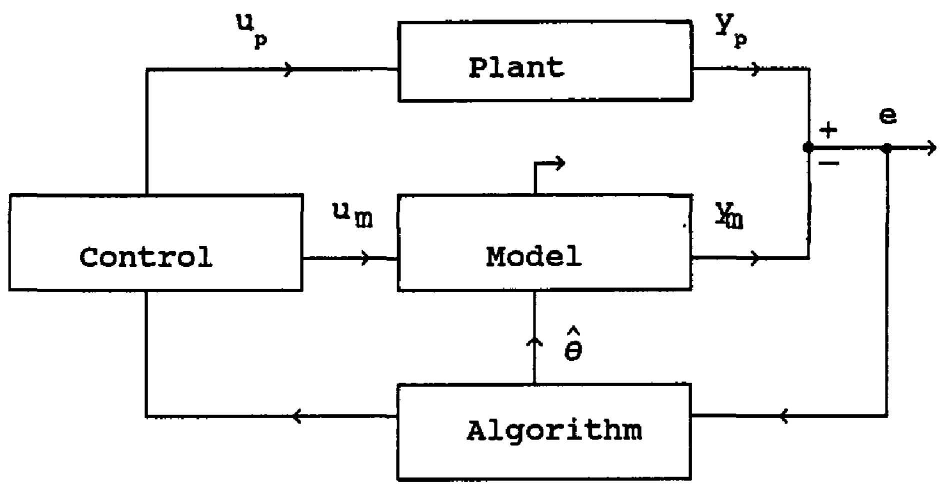


Fig. 3.1 The block diagram of a dual control system.

Let's assume that the model has the same dimension as the plant. Though this assumption is not satisfied in practice, it is a common one and it provides a starting point for a more realistic formulation. The plant has unknown parameters and the model's parameters are updated by a parameter identification algorithm. In order to build up a dual control law, a combined performance index is considered, ie.

$$V(k) = \lambda V_c(k) + (1-\lambda)V_i(k) + \alpha[u_p(k-1)^2 + u_m(k-1)^2] \quad \dots (3.1)$$

where $V(k)$ is the overall performance index;

$V_i(k)$ is the identification performance;

$V_c(k)$ is the control performance;

$u_p(k-1)$ is the plant input at time k-1;

$u_m(k-1)$ is the model input at time k-1;

λ is the weighting scalar, $0 \le \lambda \le 1$.

The weighting scalar λ in eqn. (3.1) can be varying with time. It is natural to set $\lambda=0$, at the beginning of the adaptive control operation. This means that at the beginning, without any knowledge of the parameters of the system, the input will be used for parameter identification.

Consider that a plant and its model are both described by a SISO difference equations: for the plant,

$$y_p(k)=a_1 y_p(k-1)+\ldots+a_n y_p(k-n)+b_1 u_p(k-1)+\ldots+b_m u_p(k-m)$$

$$\ldots \quad (3.2)$$

The model is of the form of

$$y_m(k)=\hat{a}_1 y_m(k-1)+\ldots+\hat{a}_n y_m(k-n)+\hat{b}_1 u_m(k-1)+\ldots+\hat{b}_m u_m(k-m)$$

$$\ldots \quad (3.3)$$

Using equation (3.1) we construct a performance criterion for dual control. Let

$$V_c(k)= y_p(k)^2 + y_m(k)^2$$

and

$$V_i(k)= [y_p(k)-y_m(k)]^2$$

The overall performance index is

$$V(k)= \lambda[y_p(k)^2 + y_m(k)^2] +(1-\lambda) [y_p(k)-y_m(k)]^2+$$

$$\alpha[u_p(k-1)^2+ u_m(k-1)^2] \qquad \ldots \quad (3.4)$$

Weighting scalar α is used to impose a penalty on the magnitude of the input.

Take the derivatives of (3.4) with respect to u (k-1) and u (k-1) and set them equal to zero to yield

$$\begin{pmatrix} u_p(k-1) \\ \\ u_m(k-1) \end{pmatrix} = \frac{\begin{pmatrix} -b_1 C^T z_p(\alpha+\hat{b}_1^2(1+(1-\lambda)^2))+b_1\hat{C}^T z_m(2(1-\lambda)\hat{b}_1^2+(1-\lambda)\alpha) \\ -\hat{b}_1\hat{C}^T z_m(\alpha+b_1^2(1+(1-\lambda)^2))+\hat{b}_1 C^T z_p(2(1-\lambda)\hat{b}_1^2+(1-\lambda)\alpha) \end{pmatrix}}{\alpha^2 + \lambda b_1^2\hat{b}_1^2 + \alpha(b_1^2 + \hat{b}_1^2)}$$

$$\dots \quad (3.5)$$

where

$$C^T = [a_1, \dots, a_n, b_2, \dots, b_m]$$

$$\hat{C}^T = [\hat{a}_1, \dots, \hat{a}_n, \hat{b}_2, \dots, \hat{b}_m]$$

$$z_p(k)^T = [y_p(k), \dots, y_p(k-n+1), u_p(k-1), \dots, u_p(k-m+1)]$$

$$z_m(k)^T = [y_m(k), \dots, y_m(k-n+1), u_m(k-1), \dots, u_m(k-m+1)]$$

Because the control law includes the true unknown parameters of the system, the estimated parameters should be used in implementing (3.5), ie,

$$\begin{pmatrix} u_p(k-1) \\ \\ u_m(k-1) \end{pmatrix} = \frac{\begin{pmatrix} -\hat{b}_1\hat{C}^T[z_p(\alpha+\hat{b}_1(1+(1-\lambda)^2)-z_m(2(1-\lambda)\hat{b}_1+(1-\lambda)\alpha)] \\ -\hat{b}_1\hat{C}^T[z_m(\alpha+\hat{b}_1(1+(1-\lambda)^2)-z_p(2(1-\lambda)\hat{b}_1+(1-\lambda)\alpha)] \end{pmatrix}}{\alpha^2 + 2\alpha\hat{b}_1^2 + \lambda\hat{b}_1^4}$$

$$\dots \quad (3.6)$$

The well-known self-tuning regulator results from minimizing the performance index [1]

$$V(k) = \hat{y}(k)^2 + \alpha u_c(k-1)^2$$

and is given by

$$u_c(k-1) = - \frac{\hat{b}_1 \hat{C}^T z(k-1)}{\alpha + \hat{b}_1^2} \qquad \ldots (3.7)$$

where $\hat{y}(k)$ is the predicted output at time k-1, ie.

$$\hat{y}(k) = h(k-1)^T \hat{\theta}(k-1)$$

$$z(k-1)^T = [y(k-1),\ldots,y(k-n),u_c(k-2),\ldots,u_c(k-m)]$$

$$h^T(k-1) = [y(k-1),\ldots,y(k-n),u_c(k-1),\ldots,u_c(k-m)]$$ and

$$\hat{\theta}^T(k-1) = [\hat{a}_1,\ldots,\hat{a}_n,\hat{b}_1,\ldots,\hat{b}_m]$$ at time k-1.

The following numerical studies apply (3.6) and (3.7) to a system

$$y_p(k) = 1.9662y_p(k-1) - 1.0348y_p(k-2) + 0.271[u_p(k-1) - u_p(k-2)] \qquad \ldots (3.8)$$

which was used by Johnson and his co-workers [24] for aircraft wing/store flutter control. The least-squares parameter estimation is used for both controllers, and with the same initial conditions: $\hat{\theta}^T(0)=[0.0,\ 0.0,\ 0.1,\ 0.0]$. $P(0)=\text{diag}(10{,}000)$, a large diagonal matrix to improve the convergence speed of parameter estimation. $\alpha= 0.1$ for this computer simulation. The simulation of the system response resulting from the dual control algorithm follows the flow chart Fig. 3.2. Fig. 3.2 includes the following logical steps:

i) Initialization: $z(0)$, $\hat{\theta}(0)$ and naturally setting $\lambda=0$ because at the beginning we have no knowledge of the parameters, parameter estimation is the most important objective;

ii) Input signal calculations based on equation (3.6);

iii) Apply the inputs to the plant and the model, and measure their outputs;

iv) If $[y_p(k)-y_m(k)]^2 < \varepsilon$ or $y_p(k)^2 > 20$, then set $\lambda=1$, otherwise set $\lambda = 0$;

v) Parameter estimation, obtain new estimated parameters of the model;

vi) k increases by 1;

vii) Calculate the predicted output $\hat{y}_p(k)$ for some computer runs ($\hat{y}_p(k) = h^T(k-1)\ \hat{\theta}(k-1)$), see Fig. 3.2.;

viii) If $y_p(k)^2 < \delta$, end the process, otherwise return to step ii).

Step iv) is called on-line test, ε was 10^{-7} and δ was 10^{-7} in these computer runs. Different on-line tests were selected in the computer simulation for switching between control and identification. They were $[y_p(k)-y_m(k)]^2<10^{-7}$ and $\| [\ \hat{\theta}(k)-\hat{\theta}(k-1)] \|^2<10^{-7}$. Figs. 3.3 and 3.4 show the response with corresponding value of $\hat{\theta}$. Fig. 3.5 shows a comparison between dual control and the self-tuning regulator. In the first 10 steps the transient response of dual control is larger than that of the self-tuning regulator, but the total cost is less than that of the self-tuning regulator. This simulation shows that the dual controller may sacrifice some performance at the beginning steps in order to reduce the parameter uncertainty for remaining steps.

In summary, the simple example above serves to illustrate a number of ideas:

1. When the parameter estimates have large uncertainty the dual control law emphasizes the goal of parameter identification, so as to improve the parameter estimation first. The self-tuning controller, a sort of certainty equivalence control, treats the updating parameters as if they were true ones, and does not select the input to improve parameter identification.

2. From the above example, we see that at the beginning of the control interval the dual control loses some regulator performance, but it gains some parameter accuracy.

3. From a long term consideration, the dual control

system is a better regulator than is the self-tuning control system.

4. Self-tuning control is used by many control engineers, because it is thought to be simpler to implement than is dual control.

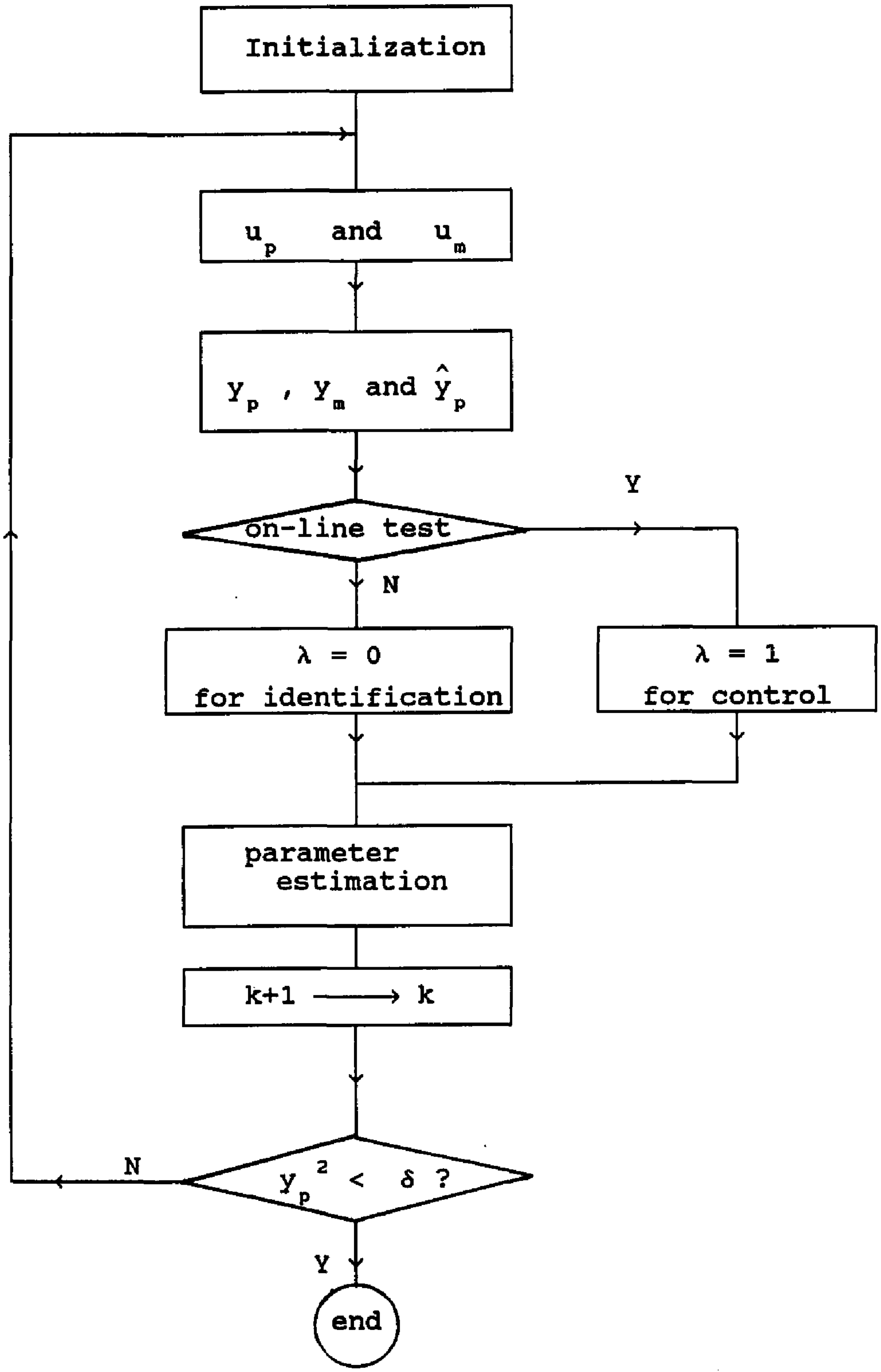
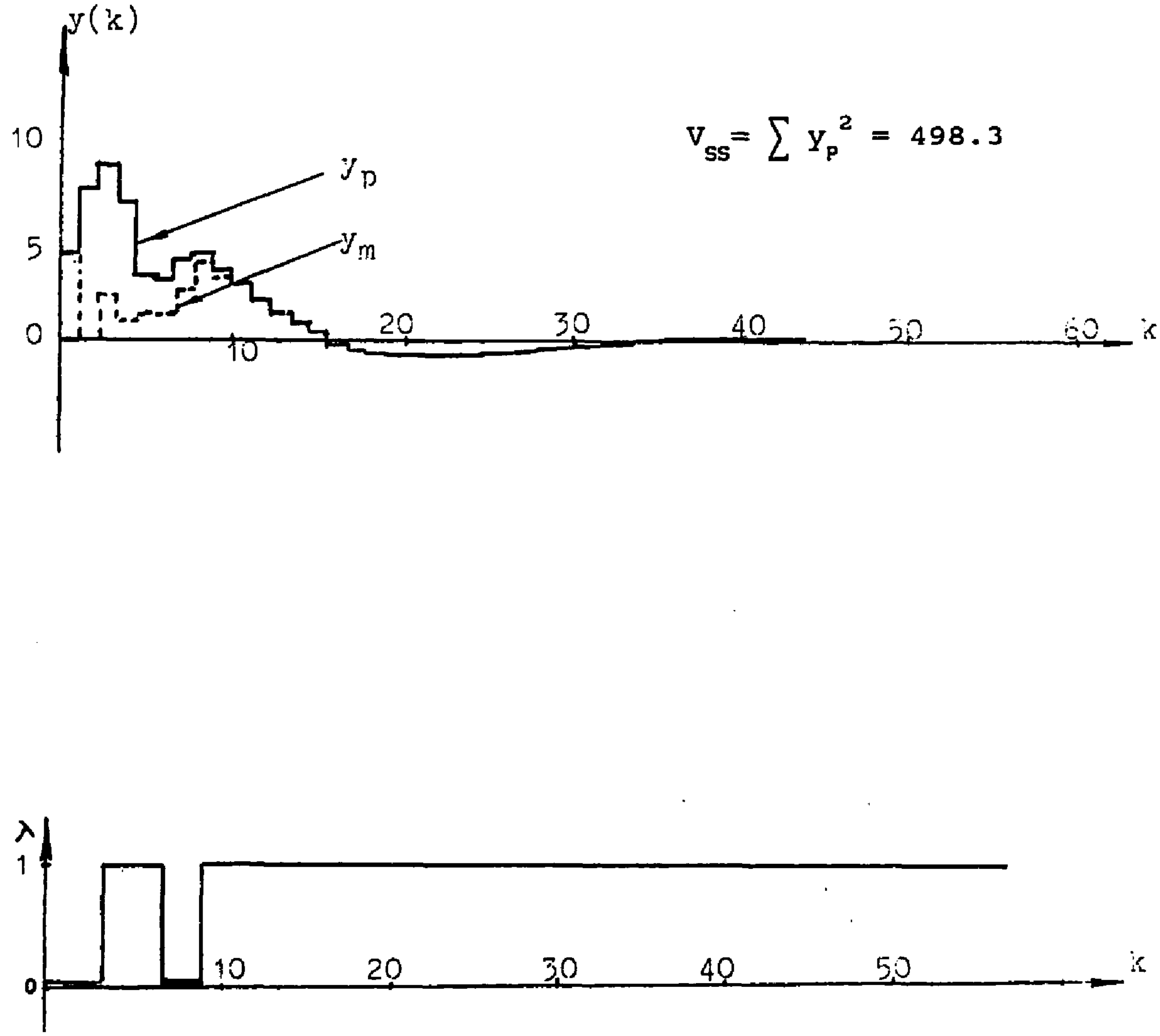
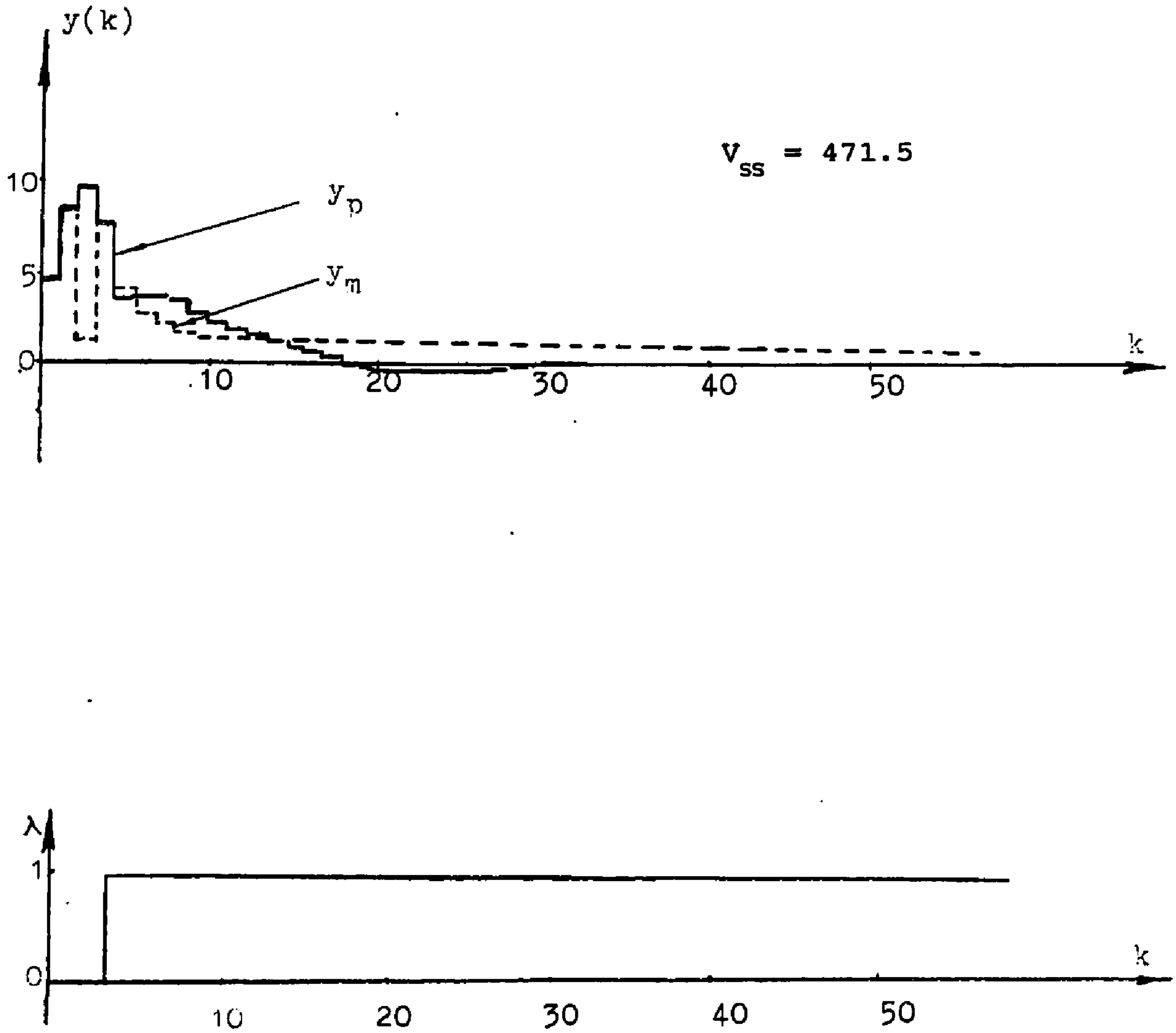


Fig. 3.2 Flow chart of the simulation of dual control, the system is described by (3.8) and the control law is (3.6).



λ=1 for control; λ=0 for identification.

Fig. 3.3 Comparison of the plant output and model output of the dual adaptive control, by using the on-line test, $|y_p(k)-\hat{y}_p(k)|^2<10^{-7}$ to turn on control.



λ=1 for control; λ=0 for identification.

Fig. 3.4 Comparison of the plant output and the model output of the dual adaptive control, by using $\|\hat{\theta}(k+1)-\hat{\theta}(k)\|^2<10^{-7}$ to turn on control.

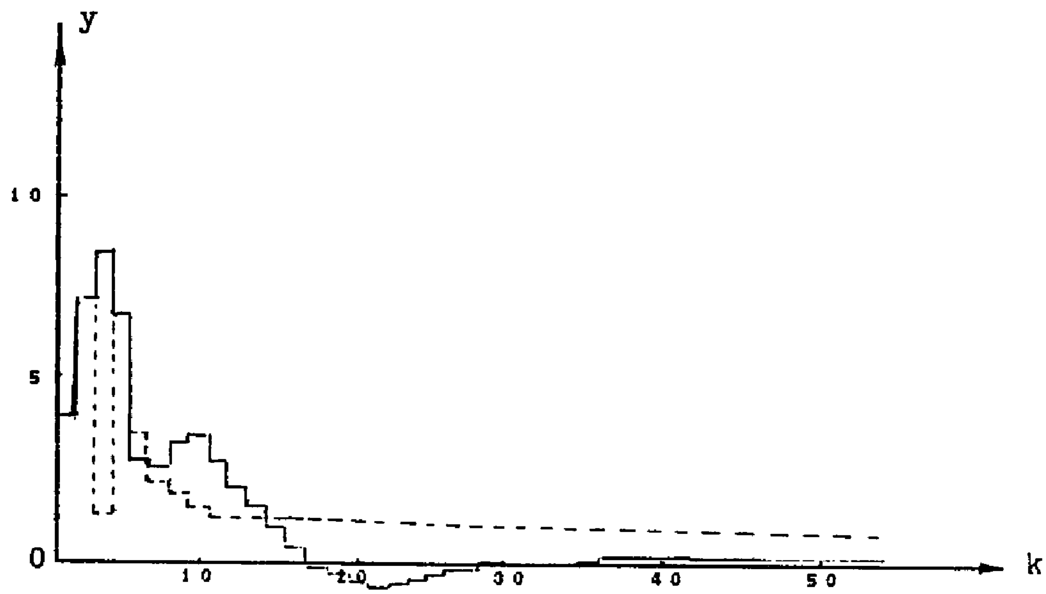


Fig. 3.5 The comparison between the dual control and self-tuning regulator used by Johnson and his co-workers, the solid line represents the dual control and dot line stands for the STR (deterministic).

Accumulated performance index ($\sum y_p(k)^2$)

dual control: 471.5

self-tuning regulator: 758.1

3.3) A New Developed Dual Control Algorithm

The above section discussed some advantages of dual control over certainty equivalence control. In this section a single-input, single-output DARMA system is considered. It is desired to design a controller that minimizes the deviation of the output of the system from a given reference sequence. The dual role of the input may be exploited by minimizing a performance index which has a term related to the parameter error. It is clear that the dual role has only a transient significance, if the parameter estimation process converges.

The system is described by

$$A\,(q^{-1})\,y(k) = B\,(q^{-1})\,u(k) \qquad \ldots (3.9)$$

where y and u denote the output and the input respectively, and $A(q^{-1})$ and $B(q^{-1})$ are scalar polynomials in the delay operator q^{-1}

$$A\ (q^{-1}) = 1 - a_1 q^{-1} - \ldots - a_n q^{-n}$$

and $$B\ (q^{-1}) = b_1 q^{-1} + \ldots + b_m q^{-m}$$

It is assumed that the coefficients of A and B polynomials are unknown and that only the input sequence $\{u(k)\}$ and output sequence $\{y(k)\}$ are directly available up to time k.

If polynomials A and B were known, the control input signal could be calculated from the system (3.9) directly by setting output y(k) equal to the reference sequence y*(k),

$$u(k-1) = - \frac{1}{b_1} [\ a_1 y(k-1) + \ldots + a_n y(k-n) + b_2 u(k-2)$$

$$+ \ldots + b_m u(k-m) - y^*(k)\ ] \qquad \ldots (3.10)$$

The control law (3.10) is called a one-step-ahead control law. Define the following vectors:

$$z^T(k-1) = [\ y(k-1), \ldots ,y(k-n),u(k-2), \ldots ,u(k-m)\ ]$$

$$c^T = [\ a_1, \ldots ,a_n,b_2, \ldots ,b_m\ ]$$

$$h^T(k-1) = [\ y(k-1), \ldots ,y(k-n),u(k-1), \ldots ,u(k-m)\ ]$$

$$\theta^T = [\ a_1,\ \dots\ ,a_n,b_1,\ \dots\ ,b_m\ ]$$

where the regression vector z(k-1) is a measurable vector at time k-1 before the control input u(k-1) is calculated and the regression vector h(k-1) is a measurable vector at time k-1 after u(k-1) is calculated. In vector notation, the open loop system is described by

$$y(k) = h^T(k-1)\ \theta \qquad \dots\ (3.11)$$

and the control (3.10) is given as

$$u^{\circ}(k-1) = -\frac{1}{b_1}\ [\ C^T z(k-1) - y*(k)\ ] \qquad \dots\ (3.12)$$

The projection estimator [5] is used throughout this section, ie.

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \frac{h(k-1)\ [\ y(k)-h^T(k-1)\hat{\theta}(k-1)]}{1 + h^T(k-1)h(k-1)} \qquad \dots\ (3.13)$$

The reason that the projection algorithm is selected, is that it is simple and converges if input signals are persistently exciting, see (2.7).

The performance index $V(k)$ to be used comprises a sum of three terms,

$$V(k) = V_c(k) + \lambda V_i(k) + \alpha u(k-1)^2 \qquad \dots (3.14)$$

where

$$V_c(k) = [\hat{y}(k)-y*(k)]^2$$

$$= \hat{b}_1 u(k-1)^2 + 2\hat{b}_1[\hat{C}^T z(k-1)-y*(k)]u(k-1)$$

$$+ [\hat{C}^T z(k-1)-y*(k)]^2 \qquad \dots (3.15)$$

$$V_i(k) = [1 + h^T(k-1)h(k-1)]^{-1}$$

$$= [u(k-1)^2+z^T(k-1)z(k-1)+1]^{-1} \qquad \dots (3.16)$$

The first term on the right hand of (3.14) represents the desire of the controller to drive the system towards the reference y*. The second term provides a penalty on parameter estimation error. The last term of (3.14) is an input constraint since the input signal may be very large without it.

How does the second term of (3.14) relate to parameter accuracy? Let's start from the eqn. (3.13). Premultiply both sides of equation (3.13) by vector $h^T(k-1)$. Rearrange the result to yield

$$h^T(k-1)\hat{\theta}(k)=h^T(k-1)\hat{\theta}(k-1)+\frac{h^T(k-1)h(k-1)[y(k)-h^T(k-1)\hat{\theta}(k-1)]}{1+h^T(k-1)h(k-1)}$$

Define the a priori output error:

$$e(k) = y(k) - h^T(k-1)\ \hat{\theta}(k-1) \qquad \text{... (3.17)}$$

and the a posteriori output error:

$$\eta(k) = y(k) - h^T(k-1)\ \hat{\theta}(k) \qquad \text{... (3.18)}$$

Then we have the following

$$\eta(k) = e(k) - \frac{h^T(k-1)h(k-1)e(k)}{1 + h^T(k-1)h(k-1)} \qquad \text{... (3.19)}$$

From the point of view of parameter identification, it is natural to select the input $u(k-1)$ to try to minimize a measure of the a posteriori output error $\eta(k)$ at time k. From (3.19) the ratio of the a posteriori output error and the a priori output error e(k), is

$$\eta(k)/e(k) = \frac{1}{1 + h^T(k-1)h(k-1)} \qquad \ldots (3.20)$$

Hence an input that enhances parameter identification is one that minimizes (3.20).

In general, to minimize (3.14) with respect to u(k-1) needs numerical calculation. Using root locus technique may help to find the optimal input signal for minimizing (3.14). Here a graphical technique is introduced (the details are given on pages 66-68). And it will give us the upper bound and lower bound on u(k-1). The boundedness of u(k-1) is needed for the proof of system output convergence and system stability.

Take the derivative (3.14) with respect to u(k-1) and equate to zero to yield

$$[\hat{b}_1^{\,2}+\alpha]u(k-1)+ \hat{b}_1[\hat{C}^T z(k-1)-y^*(k-1)] - \frac{\lambda u(k-1)}{[1 + h^T(k-1)h(k-1)]^2} = 0 \qquad \ldots (3.21)$$

which is the same as

$$[\hat{b}_1^2+\alpha]u(k-1)^5+\hat{b}_1[\hat{C}^T z(k-1)-y*(k)]u(k-1)^4+2[\hat{b}_1^2+\alpha]M$$

$$u(k-1)^3+2\hat{b}_1[\hat{C}^T z(k-1)-y*(k)]Mu(k-1)^2+[M^2(\hat{b}_1^2+\alpha)-\lambda]$$

$$u(k-1)+M^2\hat{b}_1[\hat{C}^T z(k-1)-y*(k)]=0 \qquad \ldots (3.22)$$

where $M=z^T(k-1)z(k-1)+1$

Equation (3.22) involves a fifth order polynomial in u(k-1). The numerical solution to (3.22) can be found through on-line calculation using the diagram of Fig. 3.6 to find the optimal control input. The function F in Fig. 3.6 is the fifth order polynomial of (3.22). Fig. 3.6 is a flow chart for finding the root of (3.22) by knowing the upper and the lower bound of u(k-1). In this figure 'Right' represents the lower bound and 'Left' represents the upper bound of u(k-1). And u(k-1) lies between the interval ('Intvl') of 'Left' and 'Right'. The method continually divides in half the interval until the interval is less than the value of the convergence criterion. Given the left and right ends of the interval, comparing the signs of F(Right), F(Left) and F(Mid), we can decide whether u(k-1) is located close to the right or left end. We accordingly cut the original interval in half by moving the right end (if the sign of F(Mid) is the same as the sign of F(Right)) of the interval to the midpoint. Then the new interval is between the old midpoint and left end. Now we repeat the process, until the width of the interval is less than the convergence criterion.

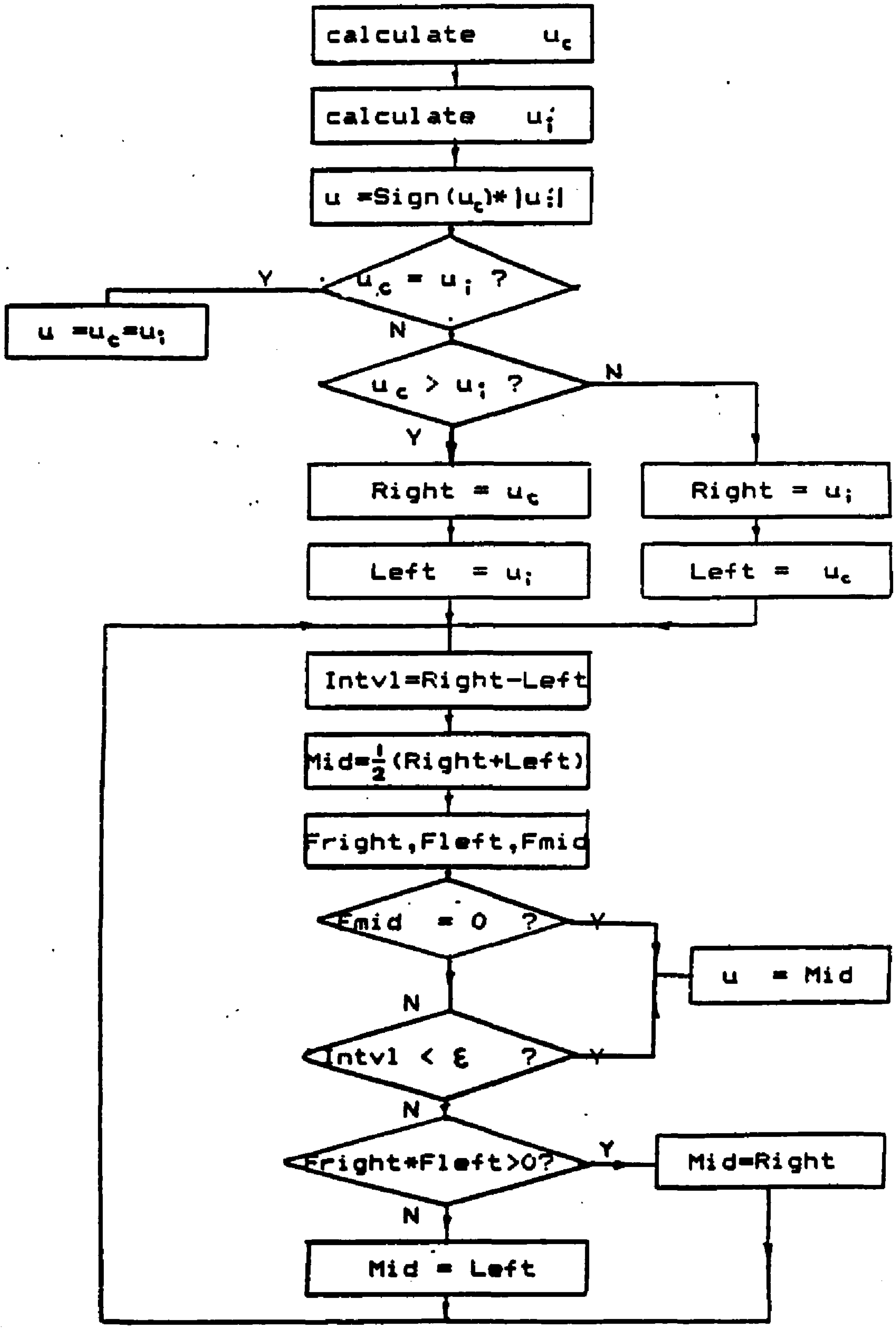


Fig. 3.6 The flow chart of root locus searching algorithm

3.4) Extensions of the New Developed Dual Control Algorithm

The discussion of the new developed dual control law in section 3.3) was limited to the special case of projection estimation algorithm (3.13) only. The projection algorithm is simple but it is poor in the sense of parameter accuracy and convergence speed. Also the dual control law [(3.13), (3.14), (3.20) and (3.22)] can also be simplified in order to obtain a performance index which is a quadratic function of u(k-1). Hence some modifications and extensions are easily made based on the above considerations.

a) Taylor expansion

Using Taylor expansion on (3.20) at $u(k-1)=u_c$, see (3.7), we have

$$\frac{1}{1+h^T(k-1)h(k-1)} \approx D + 2E[u(k-1)-u_c] + F[u(k-1)-u_c]^2$$

where

$$D = \left. \frac{1}{1+h^T(k-1)h(k-1)} \right|_{u(k-1)=u_c}$$

$$E = \left. \frac{1}{2[1+h^T(k-1)h(k-1)]} \right|'_{u(k-1)=u_c}$$

$$F = \left. \frac{1}{2[1+h^T(k-1)h(k-1)]} \right|''_{u(k-1)=u_c} \qquad \dots (3.23)$$

where ' means first derivative and '' means second derivative. Then the total performance is approximately,

$$V(k) \approx [\hat{b}_1^2+\alpha+F]u(k-1)^2+2[\hat{b}_1^2(\hat{C}^T z(k-1)-y^*)+E-Fu_c]u(k-1)+ [\hat{C}^T z(k-1)-y^*]^2+D \qquad \dots (3.24)$$

Equation (3.24) is a quadratic function of u(k-1). An approximate dual control law can be easily calculated by taking the derivative of (3.24) with respect to u(k-1)

$$u(k-1) = - \frac{\hat{b}_1[\hat{C}^T z(k-1)-y^*]+E-Fu_c}{\hat{b}_1^2 + \alpha + F} \qquad \dots (3.25)$$

(3.25) is quite similar to the weighted one-step-ahead controller (3.7), and is also similar to the one in [20 and 21]. The difference between them will be discussed in the next chapter. Its advantage, obviously, is that it is easy to calculate and to implement. But the disadvantages may be as follows: i) (3.24) is only true as u(k-1) approaches u_c (3.7), ii) adding term E to the first-order term of u(k-1) may increase the tracking error, and iii) the negative value of F may cause u(k-1) to become very large. Hence some special care should be added to the use of this formula.

b) Minmax principle

Instead of using optimization on (3.14), a minmax principle was introduced by Thau [22]. The minmax principle applied to adaptive control is based on making the best (optimal) decision out of the worst (pessimistic) conditions. In other words, the cost function is minimized with respect to input signal under the worst condition. Rewrite eqn. (3.14) into the form of

$$V(k) = \lambda V_c(k) + (1-\lambda)V_i(k) \qquad \ldots (3.26)$$

where V_i is the same as in (3.14) but $\alpha u(k-1)^2$ is included in V_i. Rewrite equation (3.26) as

$$V(k) = \lambda\ [V_c(k)-V_i(k)] + V_i(k) \qquad \ldots (3.27)$$

Note that the right side of (3.27) is a linear function of λ and is also a function of u(k-1). Maximizing (3.27) with respect to λ yields

$$\begin{aligned} &\lambda = 1 && \text{for } V_c(k)-V_i(k) > 0 \\ &\lambda = 0 && \text{for } V_c(k)-V_i(k) < 0 \\ &\lambda = * \text{ (doesn't care)} && \text{for } V_c(k)-V_i(k)=0 \qquad \ldots (3.28) \end{aligned}$$

To decide whether λ is equal to 1, equal to 0, or *, one needs to know the value of $V_c(k)-V_i(k)$. In order to find the input signal, a graphical method may be introduced.

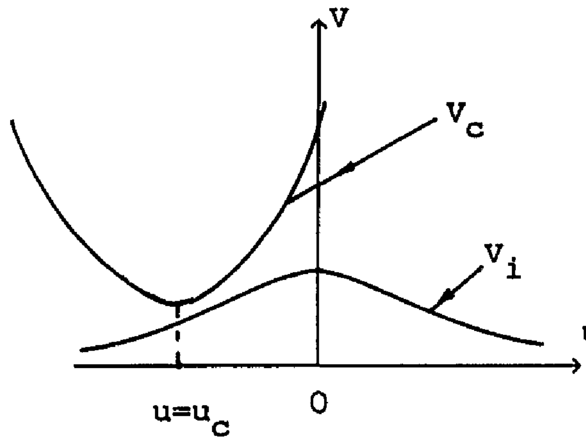


Fig. 3.7a

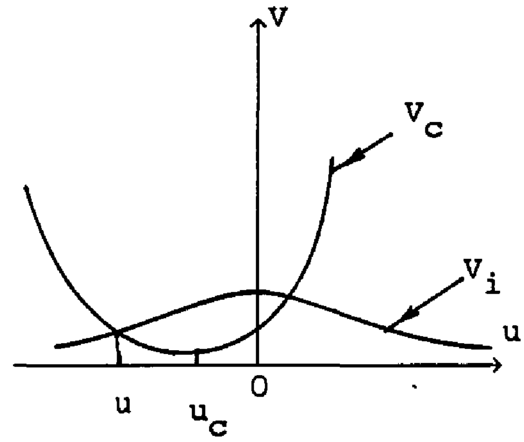


Fig. 3.7b

Fig. 3.7a is the case of $V_c(k)-V_i(k) > 0$, u_c is the best choice; Fig. 3.7b is the case of $V_c(k)-V_i(k)=0$, $u(k-1)$ is selected from the two intersections of $V_c(k)$ and $V_i(k)$. The one which is further from the origin is the best choice.

From the above graphics we can decide the $u(k-1)$ for cases of $V_c(k)-V_i(k)>0$ and $V_c(k)-V_i(k)=0$. And also we can conclude that the case of unconditional $V_c(k)-V_i(k)<0$ never occurs. From the above graph we can see that a) using the minmax principle, the control law is finite; b) to find the control law numerically the root finding method has to be used, ie, to solve the equation

$$V_c(k) = V_l(k) \qquad \ldots (3.29)$$

by using Fig. 3.6.

Equation (3.25) and the minmax principle leading to (3.27) are two different modifications to find the control law using performance (3.14). The following two extensions can be made as parameter identifier changes.

c) General projection estimation algorithm

The general projection algorithm has the following form

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \frac{ch(k-1)e(k)}{a + h^T(k-1)h(k-1)} \qquad \ldots (3.30)$$

The best choice of a and c is a=0 and c=1 in the sense of minimizing the parameter error. If a=0 and c=1, the algorithm is the same as that specified by the following problem: given $\hat{\theta}(k-1)$ and $y(k)$, determine $\hat{\theta}(k)$ so that

$$J = \frac{1}{2} \| \hat{\theta}(k) - \hat{\theta}(k-1) \|^2 \qquad \ldots (3.31)$$

is minimized subject to $\eta(k)=0$. See [5]. Because of the constraint $\eta(k)=0$, the dual control algorithm based on (3.30) is independent of the selection of u(k-1) so as to minimize the ratio of the a posteriori output and the a priori output.

If $1<c<2$, the ratio of the a posteriori output error to the a priori output error may be negative for some value of u(k-1). For the general projection algorithm, in order to guarantee the convergence of the algorithm, $a \geq 0$ and $0 < c \leq 1$ is suggested.

d) The Least-squares estimation algorithm

It is the well known that the least-squares algorithm has the form of eqn. (2.6). As in the derivation above, we use the ratio of the a posteriori output to the a priori output as the criterion of parameter identification. $V_1(k)$ then has the following form

$$V_1(k) = \frac{1}{1 + h^T(k-1)P(k-1)h(k-1)} \qquad \ldots \quad (3.32)$$

With the least-squares algorithm, a comparison can be made between dual control and the self-tuning control.

e) A modified dual control

In general, any parameter estimation scheme has the following common structure:

$$\hat{\theta}(k) = \hat{\theta}(k-1) + K(k)[y(k)-h^T(k-1)\ \hat{\theta}(k-1)] \qquad \ldots \quad (3.33)$$

Comparing (3.33) with algorithm (3.13) yields

$$K(k) = \frac{h(k-1)}{1+h^T(k-1)h(k-1)} \qquad \ldots \quad (3.34)$$

K in (3.34) is called the correction gain of the parameter identifier (3.13). If it is zero for some k then parameter estimates are not updated at that time and if K is zero for all k then the estimated parameters may converge to constant values which may not be the true parameters. For good parameter identification, this term should be relatively "large" so that the estimates of (3.13) may approach the true value as soon as possible. To maximize the square of the norm of (3.34) is equivalent to minimizing its reciprocal

$$\|K\ (k)\|^{-2} = \frac{[1+h^T(k-1)h(k-1)]}{h^T(k-1)h(k-1)}$$

This equivalent to minimizing

$$V_1(k)=\lambda\|K(k)\|^{-2} = \frac{\lambda}{h^T(k-1)h(k-1)} +\lambda(M+1) \ +\lambda u(k-1)^2 \quad \ldots (3.35)$$

where M is equal to $z^T(k-1)z(k-1)+1$ (vector z is defined in (3.10)).

The first term of (3.35) is similar to (3.20), the second term is a constant which does not affect the minimization, the third term is nothing but a penalty on the input signal. It indicates that the input signal can not be too big nor too small because the first term (3.35) makes the input signal large and the third term of (3.35) is the penalty on the input.

Hence the dual control strategy used in the simulations of the next section are based on the above performance indices (3.14, 3.32 and 3.35) and identification algorithms.

3.5) Numerical Simulations

Some simulation examples will now be shown to illustrate the properties of some of the dual control algorithms discussed above. The simulations also show the transient response of different algorithms which are difficult to investigate by analysis [2].

The system to be controlled is given by

$$y(k)=a_1y(k-1)+a_2y(k-2)+b_1u(k-1)+b_2u(k-2) \quad \ldots (3.37)$$

where the numerical values of the parameters are

$$a_1= 2.0, \quad a_2= -1.0, \quad b_1= 1.0 \quad \text{and} \quad b_2=1.0.$$

The system has two poles $p_1=p_2=1$, one zero $z=-1$. This system is the discrete version of double integral system, which is not invertibly stable.

The performance criteria considered for the simulation are as follows

$$V(k) = V_c(k) + \lambda V_i(k) + \alpha u^2(k-1) \qquad \dots (3.38)$$

the control performance

$$V_c(k) = [y(k)-y^*(k)]^2 \qquad \dots (3.39)$$

the trajectory $y^*(k)$

$$y^*(k) = (-1)^n \quad \text{where } n=\mathrm{Int}(k/50) \qquad \dots (3.40)$$

Function Int(.) represents the integer portion of (.). V_i's are selected from equations (3.16), (3.32) and (3.35). The numerical solutions for control laws that minimize (3.38) can be found by using the root finding algorithm Fig. 3.6. There are two weighting scalars α and λ in these performance indices. All simulations are made for $\lambda=1$, and some simulations are made by changing α, ie. $\alpha=0.1$, 0.01 or 0.001 in order to compare its effects. All initial conditions are the same for each run, ie. $y(0)=0$. $\hat{\theta}(0)^T=(0.1,\ 0.1,\ 0.1,\ 0.1)$. Some simulations are made for comparisons with the self-tuning regulator, ie.

$$u_s(k-1) = -\frac{\hat{b}_1[\hat{C}^T z(k-1) - y^*(k)]}{\hat{b}_1^2 + \alpha} \qquad \ldots (3.41)$$

where vectors $\hat{C}$ and $z(k-1)$ are defined as the same as before.

Fig. 3.8 and Fig. 3.9 show the comparisons between the system responses of dual control based on (3.22) and self-tuning control (3.41). Both figures contain two curves: one is the response for self-tuning control, the other one is the response for the dual control. Fig. 3.8 is for α=0.1 and Fig. 3.9 is for α=0.01. Fig. 3.10 and Fig. 3.11 show the estimated parameters. Fig. 3.10 shows the estimated parameters with different control laws, one is for self-tuning control (3.41) and another one is for the dual control based on (3.16) with α=0.1 and λ=1. There are three curves in Fig 3.11 which show the estimated parameters for system (3.37) by using the dual control based on (3.16) with $\alpha=0.1$, 0.01 and 0.001. Fig. 3.12 shows the comparison of the system responses by using different control laws, the dotted line is for self-tuning control (3.41), the solid line is for dual control based on equation (3.35) with $\alpha=0.1$ and $\lambda=1$. Fig. 3.13 is the response of the dual control which is based on equation (3.32) with $\alpha=0.1$, $\lambda=1$ and P(0) is a diagonal matrix with all entries 10.

From the above graphs we can conclude:

a) From the graphs of the estimated parameters, Fig. 3.10, dual control can obtain better results than certainty equivalence control by selecting suitable values of α and λ.

b) Transient response changes as α does. The suitable value of the weighting scalars α and λ may improve either parameter estimation or transient response.

c) From parameter accuracy and convergence speed point of view the least-square algorithm works better than the projection algorithm (see Fig.3.11). But the projection algorithm is simple and some modification of it, for example (3.24), is easier to implement.

d) Input signals of dual control are always bounded, that is a need of parameter convergence and system stability.

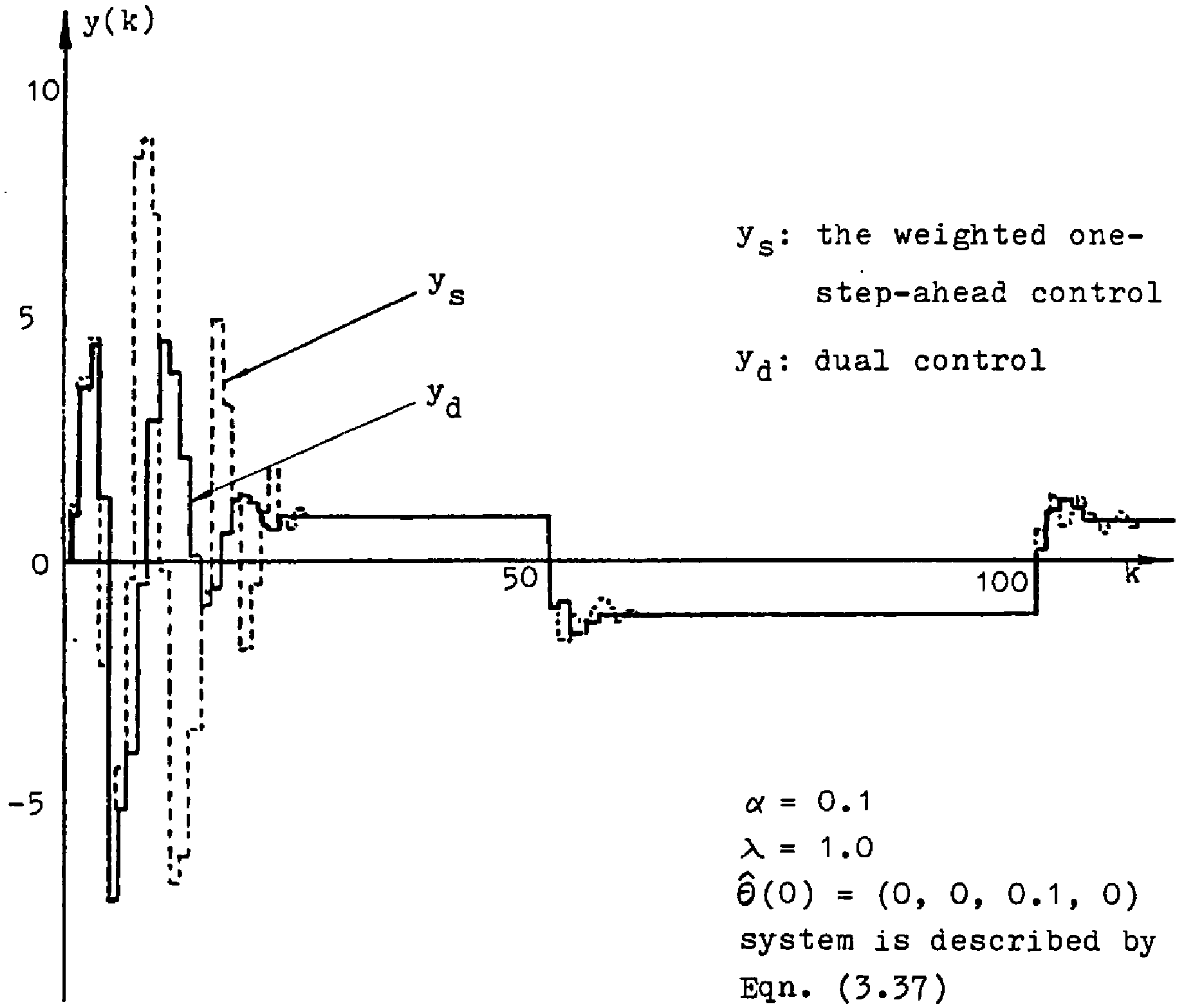


Fig. 3.8 A comparison of the system output between dual control and the weighted one-step-ahead control with the same energy constraint (eqn. 3.22).

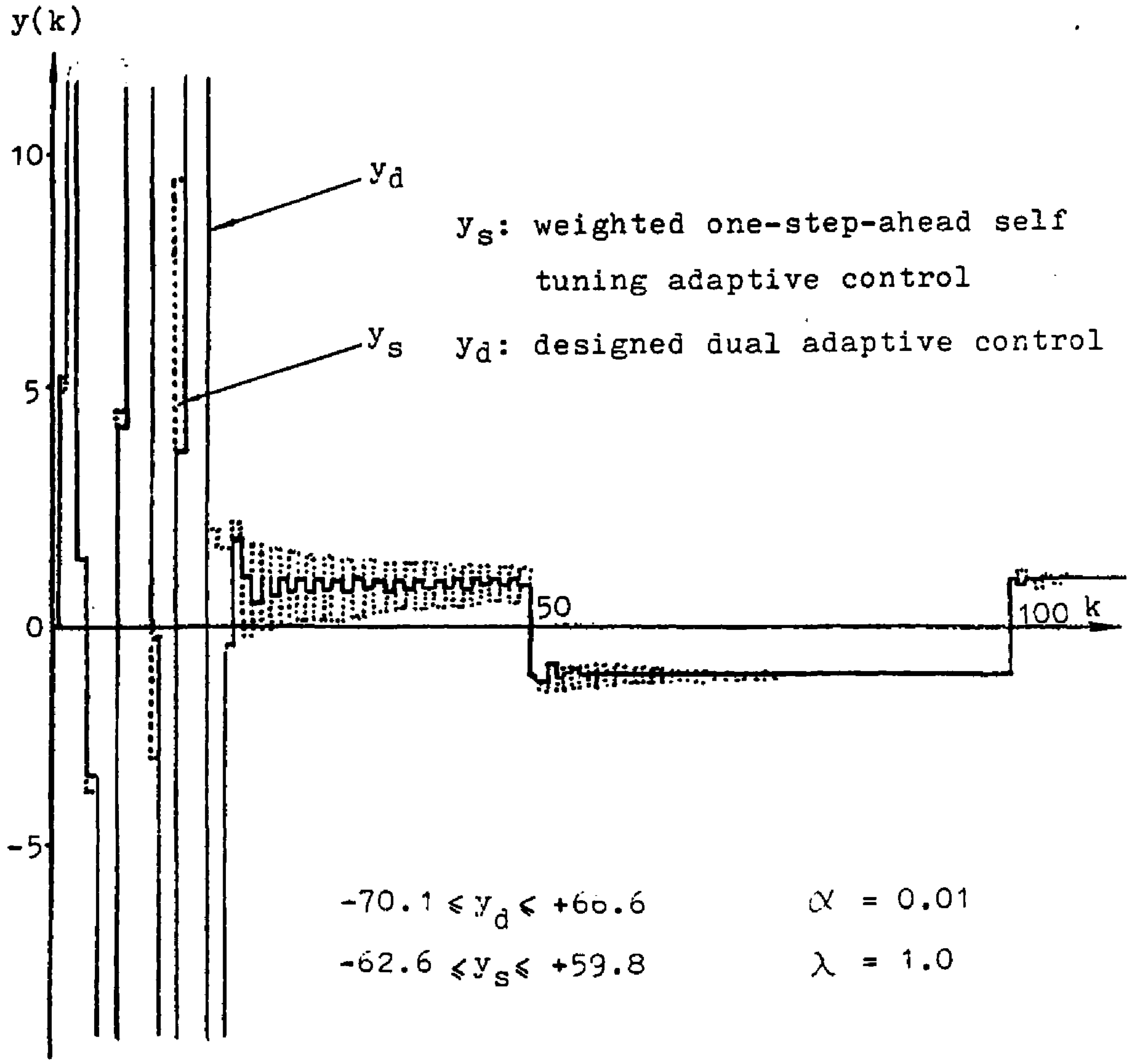


Fig. 3.9 Comparison of system responses for different control laws with the projection parameter estimation algorithm.

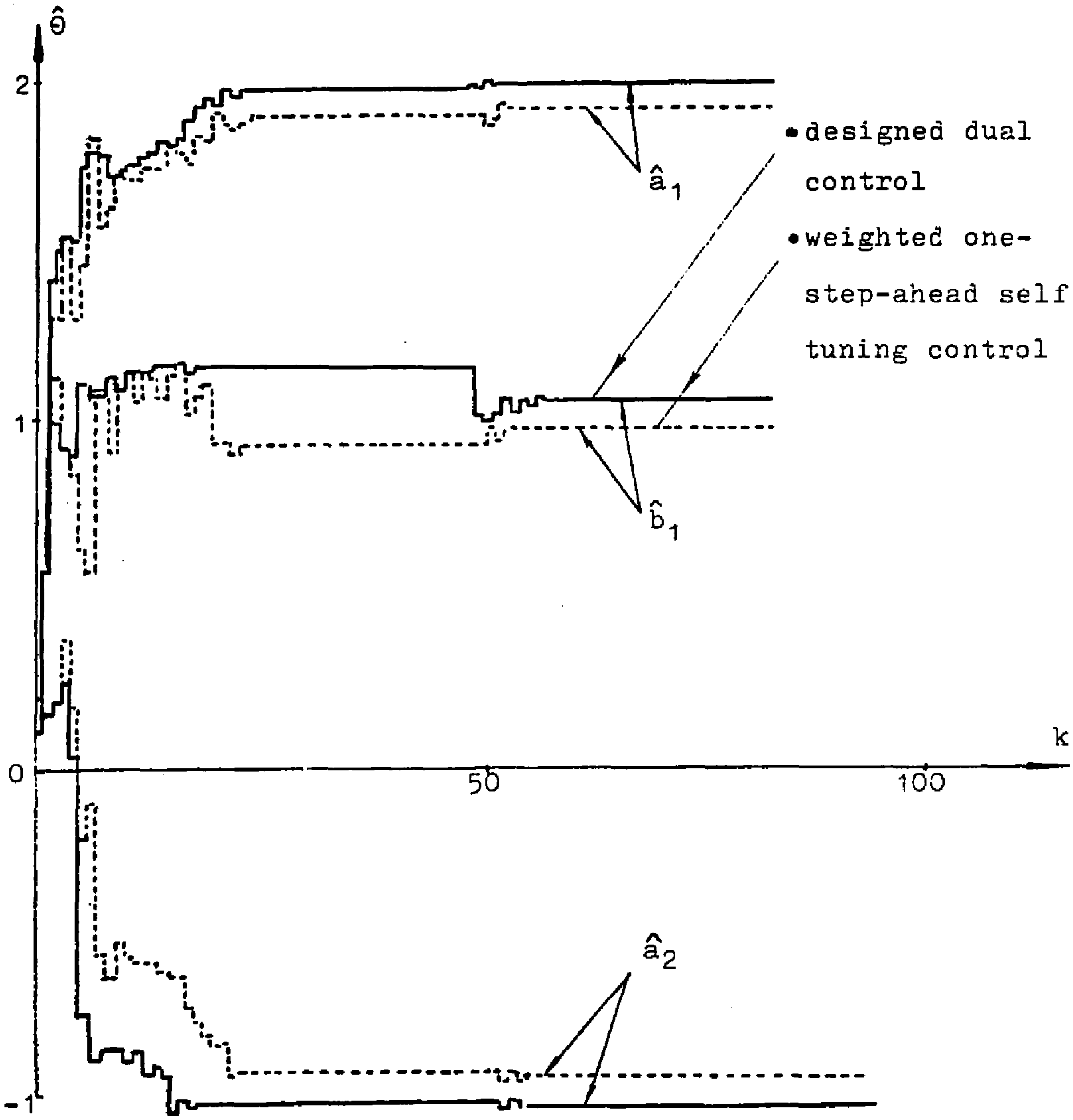


Fig. 3.10 Comparison of the estimated parameter for different adaptive control laws.

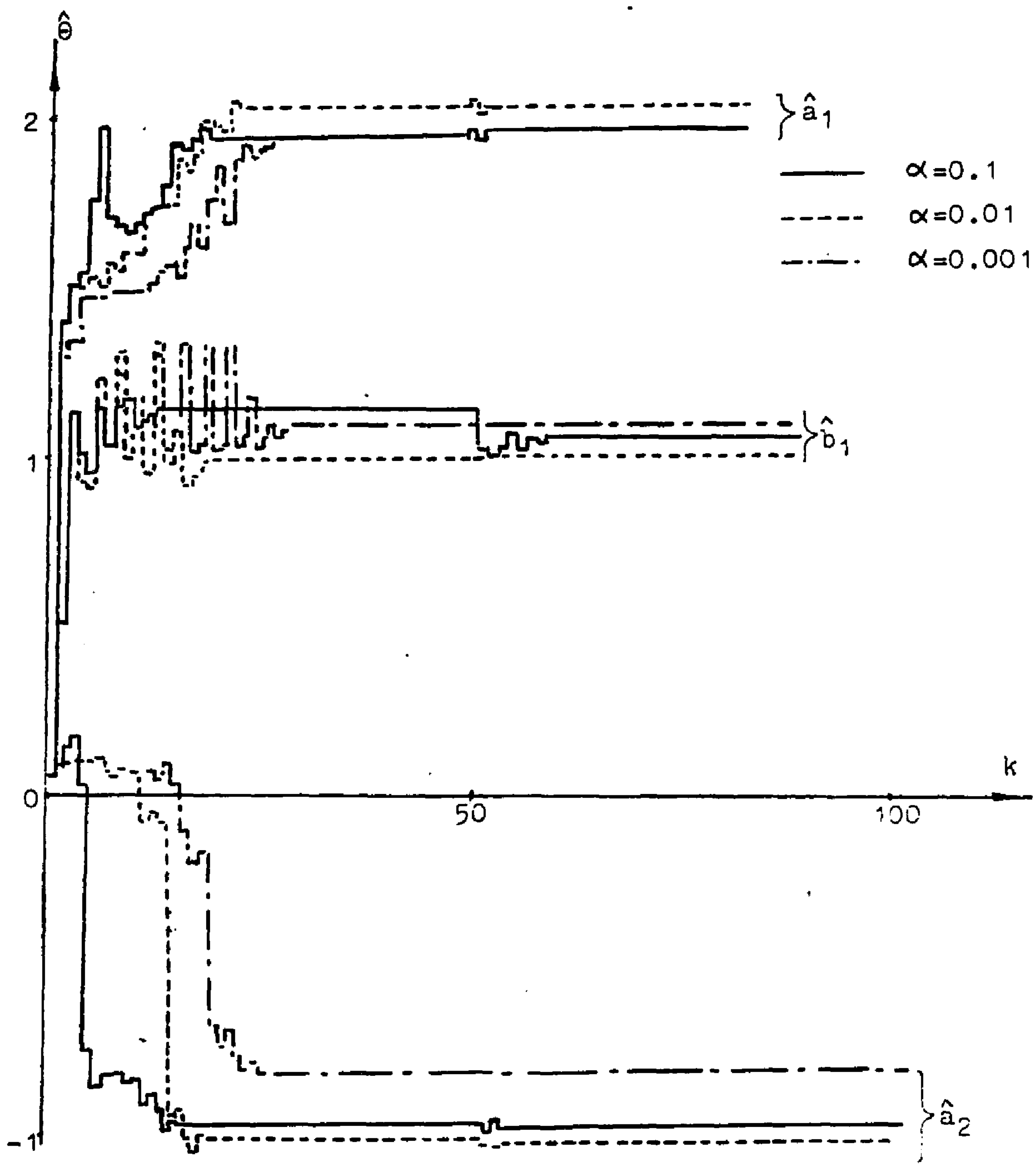


Fig. 3.11 A comparison of the estimated parameters for the system (3.37) and the dual control law based on (3.16) with different α's.

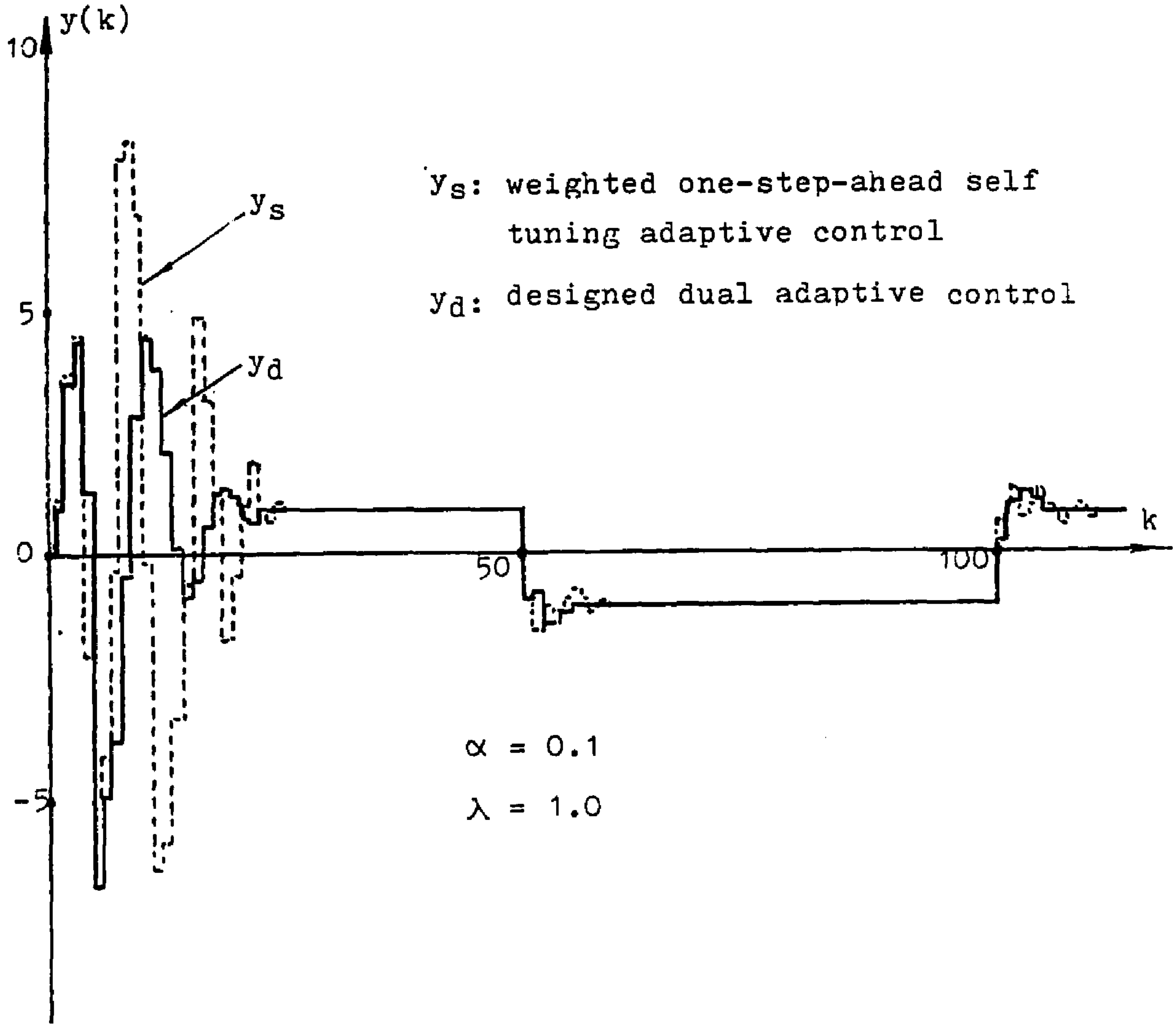


Fig. 3.12 The comparison of the modified dual control and weighted one-step-ahead control with same input energy constraint (eqn. 3.35).

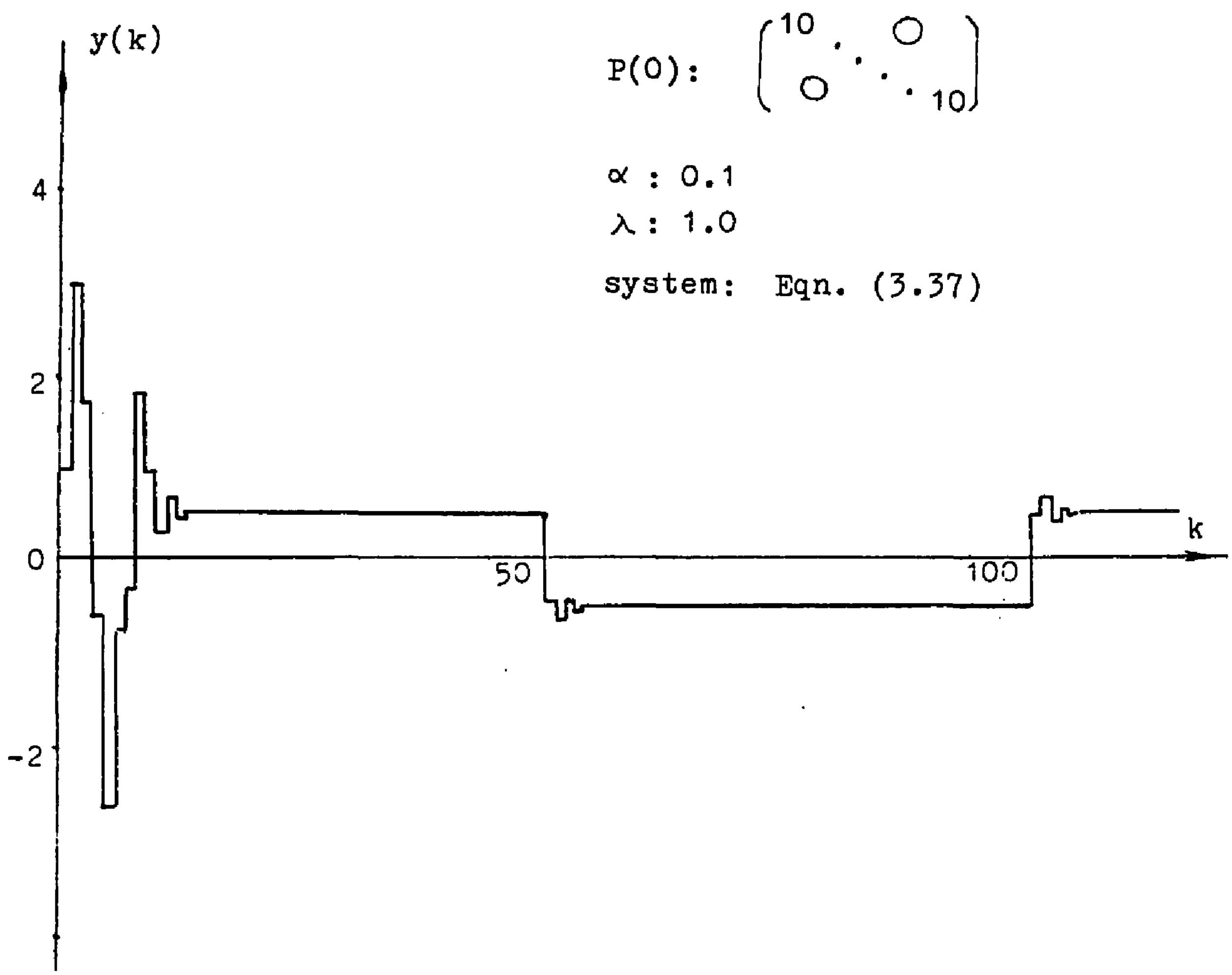


Fig. 3.13 System output of dual control with least-squares estimation algorithm based on (3.32).

Chapter Four

Stability Proof of the Dual Control Algorithm for Minimum Phase Systems

4.1) Introduction

In this chapter the newly developed dual control algorithm in the last chapter is reviewed and an analytical stability proof is presented. It appears that no stability proof for any other dual control algorithm has been presented in the literature. The proof uses the Goodwin stability lemma to yield bounded-input and bounded-output stability.

4.2) Brief Review of the Design Procedure

The system to be controlled is defined by

$$y(k) = h^T(k-1)\ \theta$$

$$= [1-A(q^{-1})]\ y(k)+B(q^{-1})u(k) \qquad \ldots (4.1)$$

where $\qquad [1-A(q^{-1})] = a_1q^{-1}+ \ldots+a_nq^{-n}$

$$B(q^{-1}) = b_1q^{-1}+ \ldots+b_mq^{-m}$$

$$\theta^T = (a_1, \ldots ,a_n,b_1, \ldots b_m)$$

$$h^T(k-1) = [y(k-1), \ldots,y(k-n),u(k-1),\ldots,u(k-m)]$$

It is assumed that:

A1) The order of the system n is known, and $n \geq m$.

A2) The system is minimum phase, ie. all roots of the polynomial B are in the unit circle.

A3) The sign of the leading coefficient of the polynomial B is known. A.3) assures that $\hat{b}_1(k)$ is not zero for all k.

We define parameter estimates

$$\hat{\theta}^T(k) = [\hat{a}_1(k),\ldots,\hat{a}_n(k),\hat{b}_1(k),\ldots,\hat{b}_m(k)] \quad \ldots (4.2)$$

that are updated by

$$\hat{\theta}^T(k) = \hat{\theta}^T(k-1) + \frac{h^T(k-1)[y(k)-h^T(k-1)\hat{\theta}(k-1)]}{1 + h^T(k-1)h(k-1)} \quad \ldots (4.3)$$

The initial condition $\hat{\theta}(0)$ are arbitrary.

The adaptive controller design algorithm is based on minimizing the performance cost V(k)

$$V(k)=[\hat{y}(k)-y*(k)]^2+\alpha u^2(k-1) + \frac{\lambda}{1 + h^2(k-1)h(k-1)} \quad \ldots (4.4)$$

where $\hat{y}(k)$ is the predicted output at time k-1,

$$\hat{y}(k) = h^T(k-1)\hat{\theta}(k-1) \quad \ldots (4.5)$$

and $y*(k)$ is a known desired reference for all k. The weighting scalar α and λ are of the form

$$\alpha(k-1) = k_1|e(k-1)|$$

$$\lambda(k-1) = k_2|e(k-1)| \qquad k_1 \, , \, k_2 > 0 \quad \ldots (4.6)$$

where e(k) is the prediction error,

$$e(k) = y(k) - \hat{y}(k) \quad \ldots (4.7)$$

The purpose of introducing the second and third terms of (4.4) was described in Chapter 3. u(k-1) can be found from minimizing (4.4), ie. solving the following equation

$$\hat{b}_1[\hat{y}(k)-y*(k)]+\alpha u(k-1)- \frac{\lambda \, u(k-1)}{[1+h^T(k-1)h(k-1)]^2} = 0 \quad \ldots (4.8)$$

Rearrange (4.8) to yield the following fifth order polynomial in u(k-1)

$$[\hat{b}_1^2+\alpha]u^5+\hat{b}_1[\hat{C}^Tz-y^*]u^4+2[\hat{b}_1^2+\alpha]Mu^3+2\hat{b}_1[\hat{C}^Tz-y^*]Mu^2+M^2[(\hat{b}_1^2+\alpha)-\lambda]u+M^2\hat{b}_1[\hat{C}^Tz-y^*]=0 \qquad \ldots (4.9)$$

where vectors $\hat{C}(k-1)$ and $z(k-1)$ satisfy

$$\hat{C}^T(k-1) = [\hat{a}_1, \ \ldots \ ,\hat{a}_n,\hat{b}_2, \ \ldots,\hat{b}_m]$$

$$z^T(k-1) = [y(k-1),\ldots,y(k-n),u(k-2),\ldots,u(k-m)]$$

and the scalar $M(k-1)=1+z^T(k-1)z(k-1)$.

The designed dual control scheme comprises the following computational steps:

1) At time k-1, given $\hat{\theta}(k-1)$, $\alpha(k-1)$, $\lambda(k-1)$ and $y^*(k)$, calculate $u(k-1)$ from (4.9) and $y(k)$.

2) Update the parameter estimates $\hat{\theta}(k)$ from (3.3).

3) If necessary, change the weighting scalar α and λ according to (4.6).

4) k increases by 1, return to step 1).

4.3) Outline of the Proof

It will be shown that the designed dual control scheme which is described above stabilizes the system (4.1) under the assumptions A1) through A3). The parameter estimation converges and the system output will converge to the desired reference y* as $k \to \infty$.

The proof depends on the following properties:

1) The adaptive control law u(k-1) which is solved numerically from (4.9) is bounded by $u_c(k-1)$ and $u_l(k-1)$ (the definition of u_c and u_l will be given later). Between these bounds there exists one and only one solution of u(k-1).

2) A stability lemma (key technical lemma) based on the properties of the parameter estimation scheme will be used to establish the linear boundedness of y(k) and u(k) (linearly bounded by e(k)).

3) The input u(k) and output y(k) are shown to be linearly bounded for all k.

4) Weighting scalars α and λ are selected such that $y(k) \longrightarrow y^*(k)$ as time approaches infinity.

4.4) The Stability Proof of the Designed Dual Adaptive

Control

We now investigate the stability and convergence properties of the closed-loop system for minimum phase systems.

i) Existence and boundedness of u(k-1):

Define

$$u_c(k-1) = - \frac{[\hat{C}^T(k-1)z(k-1)-y^*(k)]}{\hat{b}_1(k-1)} \qquad \ldots (4.10)$$

and

$$u_1(k-1) = \begin{cases} 0 & \text{if } \sqrt{\lambda/\alpha} - M<0 \\ [\text{sign}(u_c)]\sqrt{\sqrt{\dfrac{\lambda}{\alpha}} - M} & \text{otherwise} \end{cases} \qquad \ldots (4.11)$$

Now we show that there exists one and only one solution of u(k-1) which is located between $u_c(k-1)$ and $u_1(k-1)$.

Lemma 1: Given the system (4.1) and the dual control law solved from (4.9) which minimizes the performance (4.4). There exists one and only one solution u(k-1) of (4.4) which is located between $u_c(k-1)$ and $u_i(k-1)$.

Proof: $u_c(k-1)$ (see (4.10)) can be found from minimizing the following performance, which is the typical minimum variance adaptive control

$$V_c(k) = [\hat{y}(k)-y^*(k)]^2 \qquad \dots (4.12)$$

and $u_i(k-1)$ can be obtained from minimizing the following performance

$$V_i(k) = \alpha u^2(k-1) + \frac{\lambda}{1 + h^T(k-1)h(k-1)} \qquad \dots (4.13)$$

It is clear that

$$V(k) = V_c(k) + V_i(k) \qquad \dots (4.14)$$

In order to show that $u(k-1)$ is bounded by $u_c(k-1)$ and $u_i(k-1)$, we note that $\hat{y}(k)-y^*(k)=\hat{b}_1u(k-1)+\hat{C}^T(k-1)z(k-1)-y^*(k)$ and we convert (4.8) into the following form,

$$1 + \frac{\alpha u(k-1)[1+h^Th]^2-\lambda u(k-1)}{[1+h^Th]^2[\hat{b}_1^2u(k-1)+\hat{b}_1(\hat{C}^Tz-y^*)]} = 0 \qquad \dots (4.15)$$

The typical root locus technique will be used in order to locate u(k-1). There is only one real pole of (4.15) which is $u_c(k-1)$.

And there are two different cases according to zeros of the second term of the left side of (4.15). First, $\sqrt{\lambda/\alpha}-M<0$, the real zero of this case is 0, ie. z=0; and second, $\sqrt{\lambda/\alpha}-M>0$, the real zeros are $z_1=0$, $z_{2,3}=\pm u\,(k-1)$. The following figures correspond to the above two cases

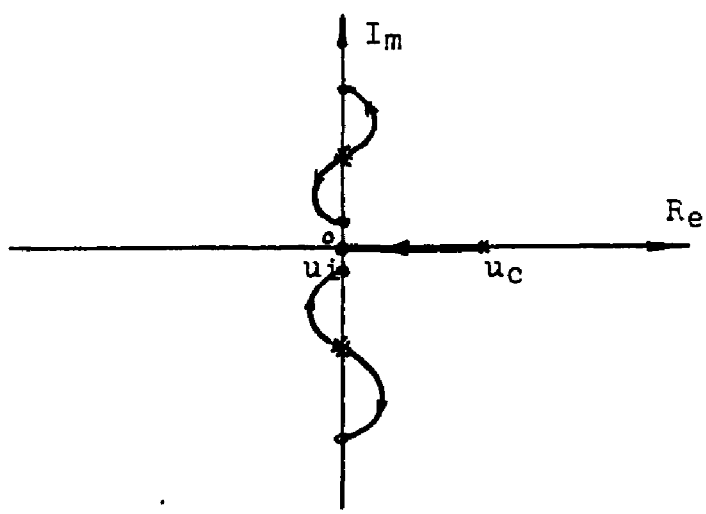


Fig. 4.1 The root loci of (4.15) for case 1.

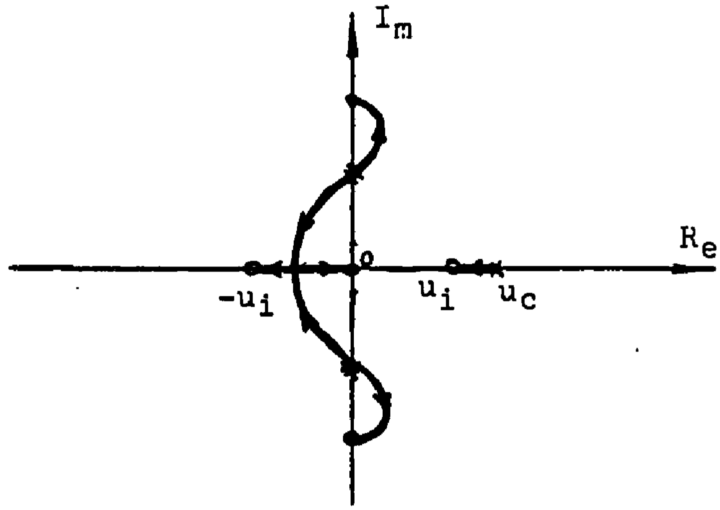


Fig. 4.2 The root loci of (4.15) for case 2.

The above root loci include all possible locations for $u(k-1)$. If we note that $u(k-1)$ must be a real number and require that its sign be the same as u_c, then there is only one solution for all the above cases so that $u(k-1)$ is located between $u_c(k-1)$ and $u_i(k-1)$.

From the above root loci, we can see that 1) the optimal input $u(k-1)$ lies between $u_c(k-1)$ and $u_i(k-1)$ and only one solution exists between them. 2) If $|u_i(k-1)|$ is less than $|u_c(k-1)|$, this means that $|u(k-1)| < |u_c(k-1)|$. This corresponds to cautious control. If u_c is small and/or M is small, then u_1 may be larger than u_c. An extra term, which is usually called a learning component, is added to u_c, in order to improve the parameter estimation as the input signal becomes smaller. 3) we will use the following expression

$$u(k-1) = u_c(k-1) + \Delta(k-1) \qquad \ldots (4.16)$$

to analysis the system stability.

The second term of the right side of (4.16) is a compensation component which is chosen by suitably selecting α and λ. Almost all other dual control algorithms make corrections in the direction of increasing the magnitude of the input signal-learning component, which may, in general, cause the system to be unstable.

Before introducing the stability key technical lemma, we list some important properties of the parameter estimation scheme (4.3) without proof. Readers who are interested in the proof of these properties should see [5].

p1) $$\lim_{k\to\infty} \frac{e(k)}{[1 + h^T(k-1)h(k-1)]^{1/2}} = 0 \qquad \ldots(4.17)$$

p2) $$\lim_{k\to\infty} \| \hat{\theta}(k)-\hat{\theta}(k-1)\| = 0 \qquad \ldots (4.18)$$

p3) $$\|\tilde{\theta}(k)\| < \|\tilde{\theta}(k-1)\| \quad \text{for all } k \qquad \ldots (4.19)$$

(4.17) will be used for the proof of the stability lemma, and (4.18) and (4.19) show the uniform boundedness of the parameter error and parameter estimates.

Lemma 2: The key technical lemma.

For some positive numbers K_3 and K_4, if

$$\|h(k-1)\| < K_3|e(k)| + K_4 \qquad \ldots (4.20)$$

for all time k, then $e(k) \longrightarrow 0$, and u(k), y(k) are uniformly bounded.

The proof has been shown in various references, for example [5], by using the recursive parameter estimation scheme (3.3). Simply we can see that (4.17) implies

$$\lim_{k\to\infty} \frac{e(k)}{[1+(K_3|e(k)|+K_4)^2]^{1/2}} = 0 \qquad \ldots (4.21)$$

If $\lim_{k\to\infty} \sup|e(k)|$ evaluated as $k \longrightarrow \infty$ were nonzero, (4.21) would be immediately contradicted.

Next we will show that the input and output sequences are linearly bounded, ie. (4.20) is satisfied, under the minimum phase assumption A2).

ii) Linear boundedness of u(k) and y(k).

With the assumptions A2), and A3) and the dual control law u(k-1) calculated from (4.9), the input and output sequences are linearly bounded, ie. (4.20) satisfies for all k.

Proof: Use the definition of (4.16) and substitute into (4.2) and (4.1), we have

$$\hat{y}(k) = y^*(k) + \hat{b}_1(k-1)\Delta(k-1) \qquad \ldots (4.22)$$

and

$$y(k) = \hat{y}(k) + e(k)$$

$$= y^*(k) + \hat{b}_1\ \Delta(k-1) + e(k) \qquad \ldots (4.23)$$

If $\Delta(k-1)$ is linearly bounded by $e(k)$ for all k, then $y(k)$ from (4.23) is linearly bounded by $e(k)$. Now we demonstrate that $\Delta(k-1)$ is linearly bounded and then that input and output sequences are linearly bounded sequences.

Since the system is minimum phase,

$$|u(k-1)| < n_1 + n_2 \max|y(k-d)| \qquad \ldots(4.24a)$$

Notice that equation (4.10) shows

$$|u_c(k-1)| < n_3 + n_4 \max|y(k-d)| \qquad \ldots (4.24b)$$

for n_1, n_2, n_3 and $n_4>0$ and $d<n$. And from (4.16)

$$\Delta(k-1) = u(k) - u_c(k) \qquad \ldots (4.25)$$

Consider (4.23), (4.24) and (4.25), we can conclude that

$$|y(k-1)| < K_5 + K_6|e(k)|$$

$$|u(k-1)| < K_7 + K_8|e(k)| \qquad \ldots (4.26)$$

Using the stability lemma, we can obtain

$$e(k) \longrightarrow 0 \quad \text{as } k \longrightarrow \infty \qquad \ldots (4.27)$$

Next we are ready to show that by suitably selecting α and λ, we will have that $y(k) \longrightarrow y^*(k)$ as $k \longrightarrow \infty$.

iii) Stability theorem

With the assumptions A1) - A3), the adaptive control law (4.3) and (4.9) stabilize the system (4.1), ie. $y(k)$ and $u(k)$ are uniformly bounded. Further more if the weighting scalars

$$\alpha(k-1) = K_1|e(k-1)|$$

$$\lambda(k-1) = K_2|e(k-1)| \qquad \dots (4.28)$$

then $\qquad y(k) \longrightarrow y^*(k) \qquad$ as $k \longrightarrow \infty \qquad \dots (4.29)$

Proof: From the above result (4.26), the input and output sequences are linear bounded by $e(k)$. Using (4.26) one can conclude that

$$\| h(k-1) \| \le (n+m) \max_{\substack{1\le i\le n+m \\ 1\le j\le n+m}} [|y(k-i)|+|u(k-j)|]$$

The Goodwin key technical lemma can be used to show that $u(k-1)$ and $y(k-1)$ are uniformly bounded for all k, ie. $\ni$ $N > 0$

$$|u(k)| < N$$

$$|y(k)| < N \qquad \text{for all } k \qquad \dots (4.30)$$

With the selection (4.28), the weighting scalars will vanish as $k \longrightarrow \infty$. Hence as $k \longrightarrow \infty$, $u(k-1)$ approaches to $u_c(k-1)$, since $|e(k)| \longrightarrow 0$. The proof of $y(k) \longrightarrow y^*(k)$ is the same as typical self-tuning control for $k \longrightarrow \infty$ [5].

4.5) Conclusion

It has been proven that the designed dual control algorithm results in a stable closed-loop system. This method can be extended to nonminimum phase systems by using a pole-placement control scheme as shown in chapter 6.

Chapter Five

Comparison of Some Dual Control Algorithms with the New Designed Dual Control Algorithm

5.1) Introduction

Because most current dual control algorithms use the least squares estimation algorithm to identify the system parameters, V_i is chosen in the form of (3.32) in order that the comparison can be made with the least squares parameter estimation algorithm. Consider that V_i has the following form, ie.

$$V_i(k) = \frac{1}{1 + h^T(k-1)P(k-1)h(k-1)} \quad \ldots (5.1)$$

The least squares estimation algorithm is

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \frac{P(k-1)h(K-1)e(k)}{1 + h^T(k-1)P(k-1)h(k-1)}$$

$$P(k) = P(k-1) - \frac{P(k-1)h(k-1)h^T(k-1)P(k-1)}{1 + h^T(k-1)P(k-1)h(k-1)}$$

One can write P(k) in the another form:

$$P(k)^{-1} = P(k-1)^{-1} + h(k-1)h^T(k-1) \qquad \dots (5.2)$$

and $h(k-1)^T=[u(k-1),\dots,u(k-m),y(k-1),\dots,y(k-n)]$,

$$\hat{\theta}(k)^T=[\hat{b}_1(k),\dots,\hat{b}_m(k),\hat{a}_1(k),\dots,\hat{a}_n(k)].$$

The system considered is the same as (2.1), but the sequences of the above two vectors are slightly changed. The designed dual control will be compared with (i) Goodwin's active adaptive control (AAC) [20]; (ii) Wittenmark's active suboptimal dual control [18] and (iii) Padilla's innovation dual control [21].

Before making the comparison between the above dual control laws we summarize some general features of the dual control which are induced from the result of multi-stage optimal control [5].

a) Use 'caution' only when the parameter estimates have large uncertainty, which is called cautious control [18, 21], ie.

$$u_{cau}(k-1) = \frac{-(\hat{b}_1 \hat{C}^T + P_{bc}{}^T) z(k-1)}{\hat{b}_1^2(k-1) + P_{bb}(k-1)} \qquad \dots \ (5.3)$$

where

$$P(k) = \begin{array}{cc} \begin{array}{cc} \scriptstyle 1 & \scriptstyle n+m-1 \end{array} & \\ \left(\begin{array}{c|c} P_{bb}(k) & P_{bc}{}^T(k) \\ \hline P_{bc}(k) & P_{cc}(k) \end{array}\right) & \begin{array}{c} \scriptstyle 1 \\ \scriptstyle n+m-1 \end{array} \end{array}$$

For simplification, we use the following expressions $z^T = z^T(k-1)$, $C^T = C^T(k-1)$ and $P = P(k-1)$, $P^{-1} = P^{-1}(k-1)$, etc. It is clear that $P_{bb}(k-1)$ in (5.3) reduces the size of $u_{cau}(k-1)$ if the size of the uncertainty of $b_1(k-1)$ is too big. In turn, the small size of $u_{cau}(k-1)$ then will slow down the convergence speed of parameter estimation (5.2).

b) A good dual control provides learning and regulation at the same time. Usually probing signals of sufficient size and with sufficiently rich components are used to persistently excite the system to achieve exponential convergence of the parameter identification.

c) Use the certainty equivalence approximation , ie.

$$u_{cer}(k-1) = \frac{-\hat{C}^T(k-1)\,z(k-1)}{\hat{b}_1(k-1)} \qquad \ldots (5.4)$$

adaptively to control the system when the estimated parameters are close to the true parameters.

5.2) Comparison with Goodwin's Dual Control Law [20].

Goodwin used

$$V_1(k) = -\frac{\det[P(k-1)]}{\det[P(k)]} \qquad \ldots (5.5)$$

as a subperformance to derive a dual adaptive control law. We assume that $V_c(k)$ is the same as the one which was used in chapter three. We keep using the same $V_c(k)$ throughout this chapter for comparison. Equation (5.5) can be written in the following way

$$V_1(k) = -\frac{\det[P^{-1}(k)]}{\det[P^{-1}(k-1)]}$$

$$= -[\ 1 + h^T\ (k-1)P(k-1)h(k-1)\ ] \qquad \ldots (5.6a)$$

$$= -P_{bb}u(k-1)^2 - 2P_{bc}{}^T z\ u(k-1) - z^T P_{cc}\ z \qquad \ldots (5.6b)$$

Equations (5.6b) and (5.1), are equivalent in the sense of maximizing $[1+h^T(k-1)P(k-1)h(k-1)]$ with respect to $u(k-1)$. Equation (5.6b) is a quadratic function of $u(k-1)$, so it is easy to minimize. And substituting (5.6b) into $V(k)$, taking the derivative with respect to $u(k-1)$, the input signal is found to be

$$u_G(k-1) = -\frac{[\hat{b}_1 \hat{C}^T - P_{bc}{}^T(k-1)]\ z(k-1)}{\hat{b}_1(k-1)^2 - P_{bb}(k-1)} \qquad \ldots (5.7)$$

The advantage of using (5.7) is that numerical calculation is then not needed to find $u(k-1)$. But (5.7) has some disadvantages: There are three cases for the equation (5.7). First, if the denominator of (5.7) is positive, ie. $\hat{b}_1(k-1)^2 > P_{bb}(k-1)$, it may increase the magnitude of $u_G(k-1)$. The increment of input u_G can be considered as a learning component. But when $\hat{b}_1(k-1)^2 = P_{bb}(k-1)$, $u_G(k-1) \longrightarrow \infty$. This is not what we want for the regulation and parameter estimation. Furthermore as $\hat{b}_1(k-1)^2 <$ $P_{bb}(k-1)$, the control signal obtains an opposite sign. That may cause the closed-loop system to become unstable. From the above three cases, we can see that the formula (5.7) is quite limited, because it may be only suitable when $P_{bb}(k-1) < \hat{b}_1(k-1)^2$.

5.3) Compare with Wittenmark's Technique [18].

Wittenmark proposed, in [18], to design a dual control law by minimizing a total performance cost which consists of the subperformance $V_1 = P_{bb}(k)$. Using (5.2) we calculate

$P_{bb}(k)$:

$$P_{bb}(k) = P_{bb}(k-1) - \frac{P^2_{bb}u^2 \; +2P_{bb}P^T_{bc}zu \; + \; P^T_{bc}zz^TP_{bc}}{1 + h^T(k-1)P(k-1)h(k-1)}$$

$$= \frac{P_{bb}(k-1)[1+z^TP_{cc}(k-1)z]-z^TP_{bc}P_{bc}{}^Tz}{1 + h^T(k-1)P(k-1)h(k-1)} \qquad \ldots (5.8)$$

The numerator of (5.8) is always positive, because the P matrix is a positive definite matrix and $P_{bb}(k)$ is positive. Note that all three terms of the numerator are constant in terms of u(k-1). To find the optimal control law, this algorithm requires numerical calculation because there is no explicit expression for finding $u_w(k-1)$ based on eqn. (5.8). In the sense of minimization, equations (3.32) and (5.8) are equivalent, but to calculate $P_{bb}(k)$ in (5.8) is an extra step compared with (3.32), and the numerical calculation of (5.8) may be a little complicated too (this will be seen in the simulation section of this chapter).

5.4) Comparison with Padilla's Technique [21]

Padilla considered the following cost function

$$V_p[u(k-1),\ z(k-1)] = E\{[y(k)-y^*]^2 - \lambda(k)[y(k)-h^T(k-1)\hat{\theta}(k-1)]^2 | z(k-1)\} \quad \ldots (5.9)$$

where $z^T(k-1) = \{\ y(k-1),\ldots,y(k-n),u(k-2),\ldots,u(k-m)\ \}$ and $0<\lambda(k)<1$. From minimizing (5.9) the innovation dual controller is given by

$$u_p(k-1) = -\frac{(1-\lambda)P_{bc}^T z(k-1) + \hat{b}_1[\ \hat{C}^T z(k-1)-y^*]}{(\ 1-\lambda)\ P_{bb}(k-1) + \hat{b}_1^2} \quad \ldots (5.10)$$

This formula is derived from the stochastic circumstance [21], so it is only true for certain stochastic systems, because it requires the parameter error covariance P. As λ changes, (5.10) changes from certainty equivalence control (λ=1) to cautious control (λ=0). Since λ is selected between 0 and 1 (claimed by Padilla), it means that $u_p(k-1)$

always reduces the magnitude of the input signal. It is not clear to see the dual function of the innovation control $u_p(k)$ from (5.10).

From the above analysis we can conclude that:

i) The designed dual control is closely related to all the above suboptimal dual controllers. The certainty equivalence control can be obtained by setting α and λ to zero in (3.14). ii) The dual control law used by Wittenmark is similar in philosophy to the one proposed here since both of them require numerical calculation.

iii) Equations (5.7) and (5.10) are explicit expressions. Of course, this is an advantage of these two expressions over the designed dual control algorithm. But they have several disadvantages as pointed out above.

5.5) Numerical Simulations

In order to simulate the above algorithms with the designed dual control algorithm by using the least-squares algorithm, we rewrite the performance index

$$V(k)=[\hat{y}(k)-y*(k)]^2+\alpha u(k-1)^2+\frac{\lambda}{1+h^T(k-1)P(k-1)h(k-1)} \quad \ldots (5.11)$$

taking derivative (5.11) with respect to u(k-1), it yields

$$\hat{b}_1[\hat{y}(k)-y*(k)]+\alpha u(k-1)-\frac{\lambda[P_{bb}u(k-1)+P_{bc}^Tz(k-1)]}{[1+h^T(k-1)P(k-1)h(k-1)]^2}=0 \quad \ldots (5.12)$$

To find u(k-1) from (5.12), it is equivalent to solve the following fifth order polynomial in u(k-1)

$$F_5u(k-1)^5+F_4u(k-1)^4+F_3u(k-1)^3+F_2u(k-1)^2+F_1u(k-1)+F_0=0$$

where $F_5=(\hat{b}_5{}^2+\alpha)P_{bb}^2$

$$F_4=4(\hat{b}_1^2+\alpha)P_{bb}P_{bc}^TZ(k-1)+\beta P_{bb}^2$$

$$F_3=(\hat{b}_1^2+\alpha)[4(P_{bc}^TZ)^2+2P_{bb}(Z^TP_{cc}Z+1)]+4\beta P_{bb}P_{bc}^TZ$$

$$F_2=4P_{bc}^TZ(Z^TP_{cc}Z+1)(\hat{b}_1^2+\alpha)+\beta[4(P_{bc}^TZ)^2+2P_{bb}(Z^TP_{cc}Z+1)]$$

$$F_1=(\hat{b}_1^2+\alpha)[1+Z^TP_{cc}Z]^2+4\beta[P_{bc}^TZ(Z^TP_{cc}Z)+1]-\lambda P_{bb}$$

$$F_0=\beta(1+Z^TP_{cc}Z)^2-\lambda P_{bc}^TZ$$

$$\beta=\hat{b}_1(\hat{C}^TZ-y*) \quad \ldots (5.13)$$

Using the root locus argument, one can find that u(k-1) is located in the unique region to guarantee that there is one and only one real solution of (5.13).

Bounds on the input signal are needed for this algorithm.

To find the bounds from (5.12), we define

$$\gamma = P_{bb}u+P_{bc}^{T}z(k-1) \qquad \ldots (5.14)$$

$$u_{pol}= \frac{-\hat{b}_1\hat{C}^T z+\hat{b}_1 y^{*}+\alpha P_{bc}^{T}z/P_{bb}}{\hat{b}_1^{2}} \qquad \ldots (5.15)$$

Then (5.12) can be rewritten as

$$[\hat{b}_1^{2}u+\hat{b}_1\hat{C}^T z-\hat{b}_1 y^{*}-\alpha P_{bc}^{T}z/P_{bb}]+ \alpha\gamma/P_{bb} - \frac{\lambda\gamma}{(1+h^T Ph)^2} = 0 \qquad \ldots (5.16)$$

Notice that $(1+h^T Ph)=1+z^T P_{cc}z+(\gamma^2-z^T P_{bc}P_{bc}^{T}z)/P_{bb}$

(5.16) is the following

$$1 + \frac{\alpha\gamma(1+h^T Ph)^2-\lambda P_{bb}\gamma}{(1+h^T Ph)^2(u-u_{pol})P_{bb}} = 0 \qquad \ldots (5.17)$$

Then the bounds are u_{pol} and u_{zero}:

where

$$u_{zero} = \begin{cases} \dfrac{-P_{bc}^{T} z}{P_{bb}} & \text{if } M > \sqrt{\lambda P_{bb/\alpha}} + z^T P_{bc} P_{bc}^{T} z / P_{bb} \\ \text{Sign}(u_{pol}) \sqrt{P_{bb}(\sqrt{\lambda P_{bb}/\alpha} - M) + z^T P_{bc} P_{bc}^{T} z}\ ] & \\ & \text{otherwise} \end{cases} \quad \ldots (5.18)$$

and $M=1+z^T P_{cc} z$.

The simulation is the following

The system selected is

$$y(k) = 2.0y(k-1)-y(k-2)+u(k-1)+u(k-2) \quad \ldots (5.19)$$

but with unknown parameters. The initial conditions of the system parameters are (0.1,0.1,0.1,0.1). The parameter estimation algorithm which is used for the comparison is the typical least square estimation algorithm. P(0) is a diagonal matrix, ie. P(0)=100.0*I where I is the identity matrix. The comparison is made to the above system (5.19) with the same initial conditions by using different control laws: (5.7), (5.10) are compared with the designed dual control law which is calculated from (5.13). Fig. 5.1 and 5.2 show the comparisons of output tracking and estimated system parameters. The solid line in these two figures represents the dual control by using (5.13), the dotted line is Goodwin's control law (5.7), the point-dash line is for Padilla's control law (5.10) for $\lambda = 0.8$. We can see that the dual control law provides improved performance.

5.6) Conclusion

The above comparison and simulation results showed the following conclusions:

(1) The new designed dual control algorithm provides a better performance than any of the competing dual control algorithms mentioned above.

(2) The new designed dual control algorithm provides a good compromise between tracking and parameter estimation by proper selection of the weighting scalars α and λ.

(3) The time varying parameters α and λ provide more flexibility so that the feedback system will obtain a good dual control.

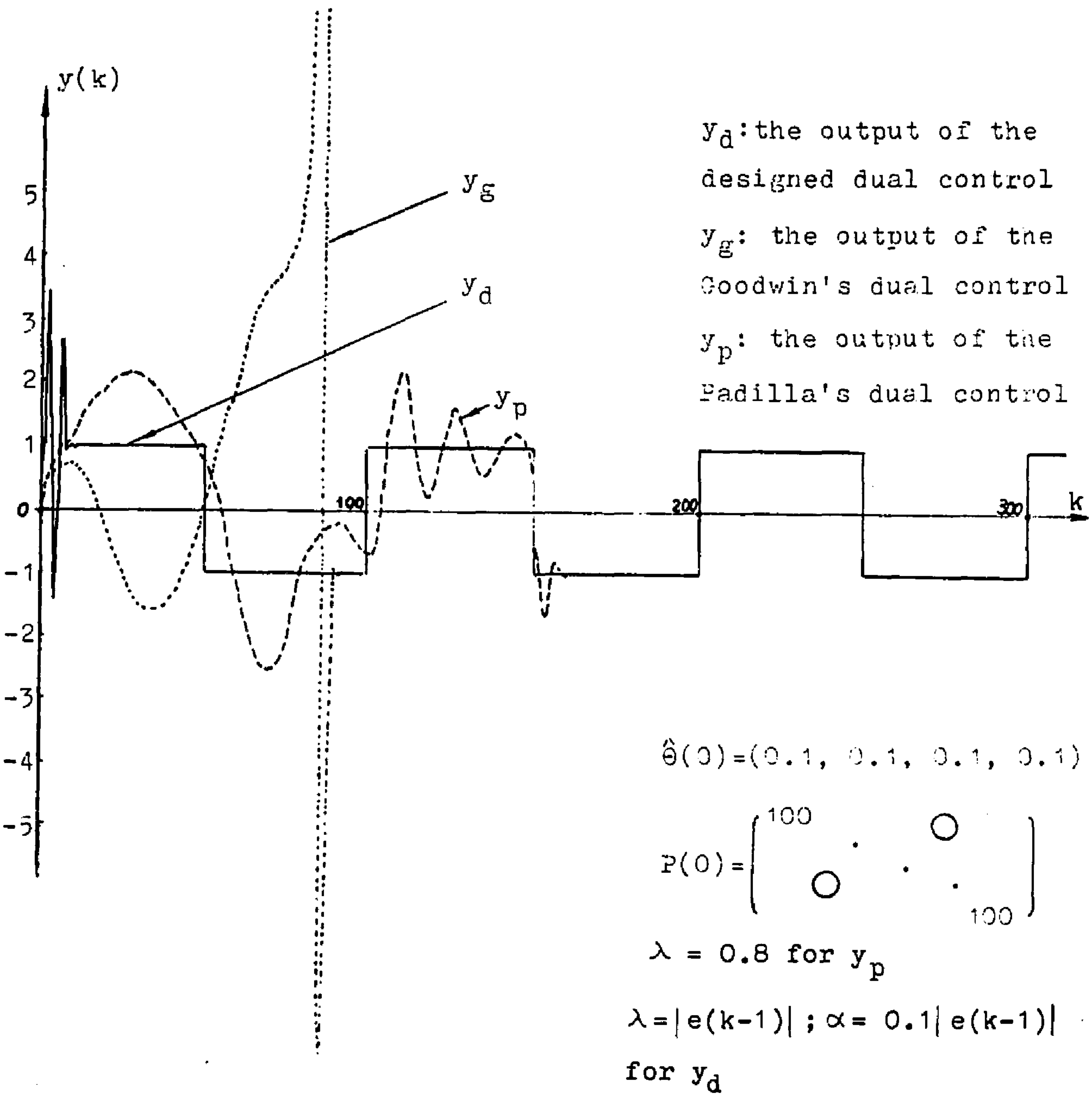


Fig. 5.1 The comparison of the system outputs with different dual adaptive control techniques.

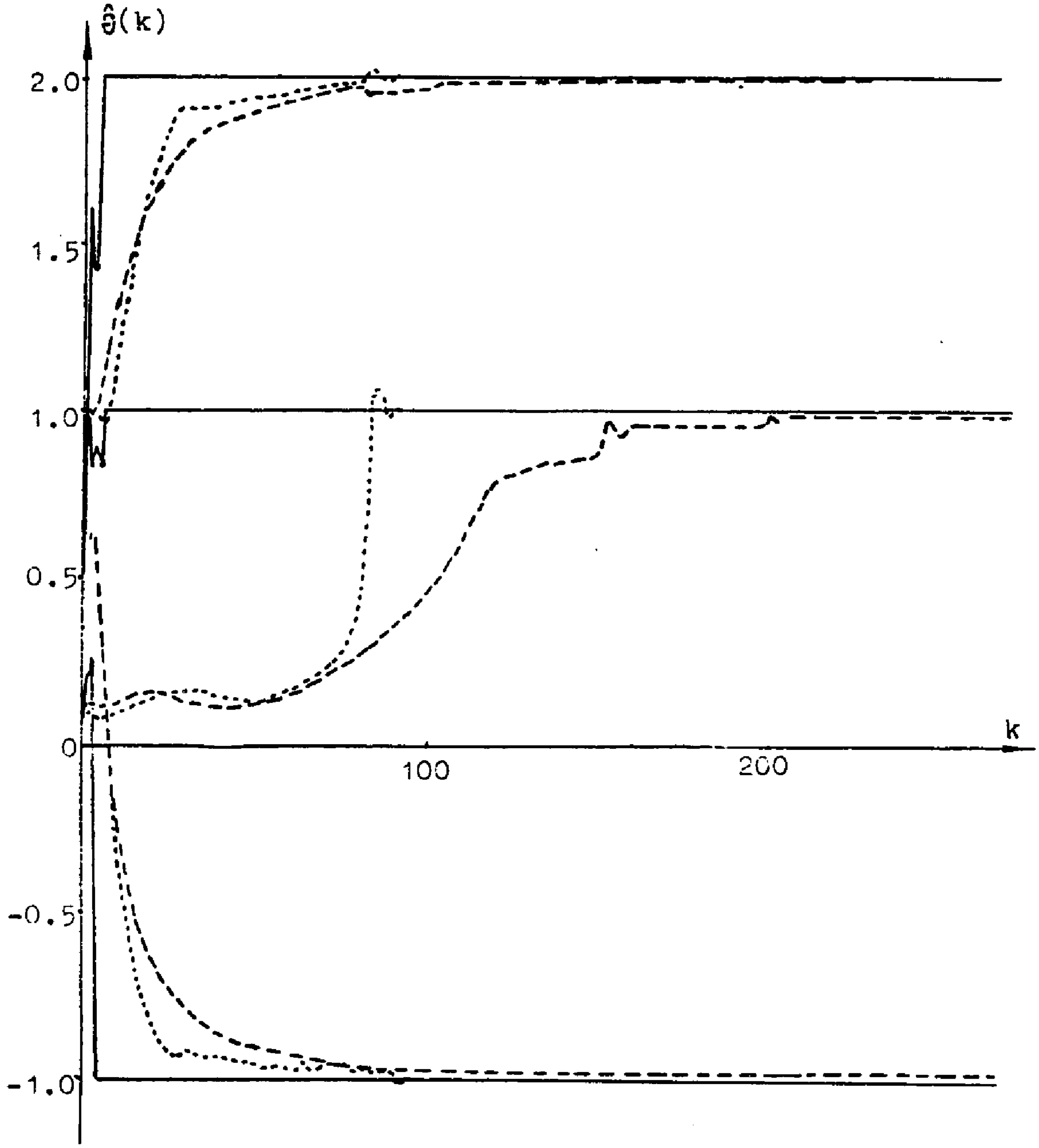


Fig. 5.2 Comparison of the system parameter estimations between different dual adaptive control techniques. The solid line is for the designed dual control, the dotted line represents Goodwin's dual control and the dash line is for Padilla's dual control.

Chapter Six

A Dual Adaptive Controller Design For Nonminimum Phase Systems by Using Pole-placement technique

6.1 Introduction

This chapter is the extension of chapter 4) to nonminimum phase systems. The synthesis of stable adaptive schemes for the control of nonminimum phase systems has received a great deal of attention in recent years [5, 25-29]. Of particular interest has been the issue of establishing closed-loop stability without a requirement that an external probing signal be used to guarantee that plant signals are persistently exciting. One approach is based on using a pole - placement control approach and a least squares parameter identification algorithm with a specific forgetting factor [26]. A second approach [27] uses a nonlinear feedback signal, a function of an identification mismatch error, to provide a signal of "self-excitation" capability, avoiding the need for an external probing signal.

In this chapter we use a dual control technique to treat the problem of achieving close-loop stable tracking for a single-input and single-output linear time invariant plant of known order n with arbitrary but unknown constant parameters. We design a feedback controller which will (1) adaptively stabilize the plant (2) result in an output that will asymptotically converge to a desired trajectory and (3) identify parameters in an ARMA plant model. The initial conditions on parameter estimation are only limited such that the Sylvester matrix [5] (for more detail see page 96 below) for the plant's DARMA model is not singular at time $k=0$.

The dual control approach is based on the minimization of a one-step-ahead performance function $V(k)$ that is a sum of a component $V_c(k)$, whose minimization with respect to the plant input would result in a pole-placement controller if the plant parameters were known, and a component $V_1(k)$ whose minimization with respect to the plant input would result in a reduction of the a posteriori output modeling error. This dual control algorithm is presented in section 6.2). The stability proof of the algorithm is presented in section 6.3). The key parts of the proof are the stability lemma which was introduced by Goodwin [5] and the boundedness of input signal u(k-1). Finally, some simulation results are presented to show the improvement in parameter identification and in output tracking compared with a conventional pole-placement adaptive control system. The simulation example also shows that the selection of performance parameters α and λ will balance the regulation and parameter estimation performance of the designed system.

6.2) The Dual Adaptive Control Algorithm for Nonminimum Phase Systems:

Consider a SISO ARMA model which may be written as

$$A(q^{-1})y(k) = q^{-1}B'(q^{-1})u(k) = B(q^{-1})u(k) \qquad \ldots (6.1)$$

Furthermore we assume that:

(A.1) The orders of the polynomials A and B are known;

(A.2) A and B are relatively coprime;

(A.3) $y^*(k)$ is known and $|y^*(k)|<N$ for all k.

Equation (6.1) may be written as:

$$y(k) = h^T(k-1)\ \theta \qquad \ldots (6.2)$$

where $\quad h^T(k-1) = [y(k-1),\ldots,y(k-n),u(k-1),\ldots,u(k-m)]$

and $\quad \theta^T = [-a_1,\ \ldots,\ -a_n,\ b_1,\ \ldots\ ,\ b_m]$

The controller is designed by minimizing with respect to $u(k-1)$ the following performance cost function

$$V(k) = V_c(k) + V_1(k) \qquad \ldots (6.3a)$$

$$=\{P(q^{-1})[y(k)-y^*(k)]\}^2+[Q'(q^{-1})u(k-1)]^2+\alpha'u(k-1)^2$$

$$+\ \frac{\lambda'}{1 + h^T(k-1)h(k-1)} \qquad \ldots (6.3b)$$

Then $V_c(k)$ is the sum of the first two terms in (6.3b) and

$V_i(k)$ is the sum of the last two terms in (6.3b). V_c is similar to the performance cost that leads to a conventional pole placement adaptive control system. V_i is the ratio of the a posteriori output error of the system to the a priori output error of the system [see the detail in chapter 3] plus an input constraint term. Polynomials P and Q' are polynomials in q^{-1}, ie.

$$P(q^{-1})=p_0+p_1q^{-1}+...+P_nq^{-n}$$

$$Q'(q^{-1})=q_0'+q_1'q^{-1}+...+q_m'q^{-m} \qquad ... \ (6.3c)$$

α' and λ' are positive numbers (to be discussed later).

Taking the derivative of (6.3b) with respect to u(k-1) yields

$$P[y(k)-y^*]+Qu(k-1)+\alpha u(k-1)-\frac{\lambda u(k-1)}{[1+h^T(k-1)h(k-1)]^2}=0 \qquad ... \ (6.4a)$$

where $$Q(q^{-1})=\frac{q_0}{p_0b_1}Q'(q^{-1})$$

and $$\alpha=(1/p_0b_1)\alpha',$$

$$\lambda=(1/p_0b_1)\lambda' \qquad \dots (6.4b)$$

p_0 and q_0 are the leading coefficients of P and Q. And P and Q are calculated at each step k so that the following equation is satisfied

$$P\ B + Q\ A = A^* \qquad \dots (6.5)$$

A* is a known desired stable polynomial. Unfortunately, since the parameters of A and B are unknown, (6.5) can not be used directly. To calculate Q' is not important during the calculation of the control law. During the calculation, it is required to find P and Q directly (see the design procedure described below). Because equation (6.4) includes the unknown system parameters, the control law is obtained from an equation similar to (6.4), ie. by using the estimated parameters as if they were true ones in (6.4). The projection algorithm is used for the parameter estimation,

ie.

$$\hat{\theta}(k) = \hat{\theta}(k-1) + \frac{h(k-1)e(k)}{1 + h^T(k-1)h(k-1)} \qquad \dots (6.6)$$

where the equation error e(k) is

$$e(k) = y(k) - \hat{y}(k) = y(k) - h(k-1)^T\hat{\theta}(k-1) \qquad \dots (6.7)$$

Rewrite (6.4) as

$$\hat{P}[h(k-1)^T\hat{\theta}(k-1) - y^*] + \hat{Q}u(k-1) + \alpha u(k-1) - \frac{\lambda u(k-1)}{[1+h(k-1)^T h(k-1)]^2} = 0 \qquad \dots (6.4')$$

Calculating u(k-1) to satisfy the above equation is the same as solving the following fifth order equation for u(k-1)

$$[1+h(k-1)^T h(k-1)]^2 \{\hat{P}[\hat{y}(k) - y^*] + \hat{Q}u(k-1) + \alpha u(k-1)\} - \lambda u(k-1) = 0 \qquad \dots (6.8)$$

where the polynomials $\hat{P}$ and $\hat{Q}$ are calculated from a modified equation (6.5), ie.

$$\hat{P}(q^{-1})\hat{B}(q^{-1}) + \hat{Q}(q^{-1})\hat{A}(q^{-1}) = A^* \qquad \dots (6.9)$$

Computation of $u(k-1)$ from (6.8) is aided by requiring $u(k-1)$ to be bounded. The bounds on $u(k-1)$ arise from the following considerations: (1) to guarantee that the root finding algorithm converges, (2) to limit the computation time such that it meets the need for on-line computation of the input control signal, and (3) to guarantee that there is only one real solution to (6.8) that guarantees that $u(k-1)$ is linearly bounded by $e(k)$. The last property will be used in the stability proof. How to find appropriate bounds for $u(k-1)$ will be discussed in the next section.

The calculation procedures of the new dual control algorithm is as follows:

(1) At time k-1, known $\hat{A}$ and $\hat{B}$ (the estimated polynomials corresponding to A and B) are used to calculate $\hat{P}$ and $\hat{Q}$ from (6.9) using Sylvester's theorem.®

(2) Bounds for $u(k-1)$ (to be discussed in the next section) are calculated.

(3) Known $\hat{P}$, $\hat{Q}$, $\hat{A}$ and $\hat{B}$, are used in (6.8) to calculate $u(k-1)$, the input to the controlled system.

(4) At the next stage $y(k)$ is measured and new parameter estimates $\hat{\theta}(k)$ are generated from (6.6). Then the procedure starting with step (1) above is repeated.

6.3) The Stability Proof:

A proof that the feedback system using this algorithm is

® The Sylvester's theorem [5]:

If the polynomial A and B are relatively coprime, there exists a unique solution of P and Q for the following polynomial equation

$$PB+QA=A^*$$

The coefficients of P and Q can be found from the following equation

$$\begin{pmatrix}
a_0 & & & b_0 & & \\
a_1 & a_0 & & b_1 & b_0 & \\
\vdots & a_1 & \ddots & \vdots & b_1 & \ddots \\
a_n & \vdots & a_0 & b_n & \vdots & b_0 \\
 & a_n & a_1 & & b_n & b_1 \\
 & & \vdots & & & \vdots \\
 & & a_n & & & b_n
\end{pmatrix} \times \begin{pmatrix} q_0 \\ \vdots \\ q_{n-1} \\ p_0 \\ \vdots \\ p_{n-1} \end{pmatrix} = \begin{pmatrix} a^*_0 \\ \vdots \\ \vdots \\ a^*_{2n-1} \end{pmatrix}$$

The matrix in the above equation is called Sylvester's matrix. If A and B are relatively coprime, this matrix is not singular.

stable is quite similar to the proof presented for minimum phase systems in chapter 4. The proof depends on (1) a stability lemma, (2) the boundedness of u(k-1), (3) the properties that as k becomes sufficiently large, $e(k) \longrightarrow 0$, (4) the selection of α and λ as functions of e(k) to ensure that the control system will approach the conventional pole placement adaptive control system.

Lemma 1: With the assumptions A.1 - A.3, the optimal control law which is calculated from (6.8), is always bounded by u_c and u_1, and there is only one real solution which is bounded by u_c and u_1. u_c is calculated from the following equation

$$\hat{P}(q^{-1})[\hat{y}(k)-y^*(k)]+\hat{Q}(q^{-1})u_c(k-1) = 0 \qquad \dots (6.10)$$

and u_1 is a real solution which is calculated from the following equation

$$\alpha\ u_1(k-1) - \frac{\lambda\ u_1(k-1)}{[\ 1 + h^T(k-1)h(k-1)]^2} = 0 \qquad \ldots(6.11)$$

Proof: In order to prove this lemma, we use a root locus argument to show all possible cases. For each case there always exists a pair of real bounds such that u(k-1) is located between them. Note that equation (6.8) may be rewritten in the following form

$$1 + \frac{\alpha u(k-1)[1+h^T(k-1)h(k-1)]^2 - \lambda u(k-1)}{[1+h^T(k-1)h(k-1)]^2[\hat{P}(\hat{y}(k)-y*)+\hat{Q}u(k-1)]} = 0 \qquad \ldots\ (6.12)$$

Then the solution of (6.8) is located between the real pole and zero of the second term of the left side of (6.12). Note that there is only one real pole of the second term of the left side of (6.12), ie.

$$\hat{P}(q^{-1})[\hat{y}(k)-y*] + \hat{Q}(q^{-1})u_c(k-1) = 0 \qquad \ldots\ (6.13)$$

To solve (6.13) yields

$$u_c(k-1) = u_{p1}(k-1) = \frac{\hat{P}y^* - (\hat{P}-\hat{p}_0)\hat{y}(k) - \hat{p}_0\hat{C}^T Z(k-1)}{\hat{p}_0\hat{b}_1 + \hat{q}_0} \quad \ldots (6.14)$$

where $\hat{p}_0$ and $\hat{q}_0$ are the leading coefficients of $\hat{P}$ and $\hat{Q}$, and

$$\hat{C}^T = [-\hat{a}_1, \ \ldots, \ -\hat{a}_n, \ \hat{b}_2, \ \ldots, \ \hat{b}_m \ ]$$

$$Z(k-1)^T = [y(k-1), \ldots, y(k-n), u(k-2), \ldots, u(k-m)]$$

There are two double imaginary poles, that are

$$u_{p2\text{-}5} = \pm j \sqrt{1+Z^T(k-1)Z(k-1)} \quad \ldots (6.15)$$

There are two different cases to be considered in calculating the zeros of the second term in (6.12).

For case one, ie, $(1+Z^TZ) > \sqrt{\lambda/\alpha}$, then

$$u_{z1}=0$$

$$u_{z2\text{-}5} = \pm j\left(\sqrt{(1+Z^TZ)\pm\sqrt{\lambda/\alpha}} \right)$$

for real solution $\quad u_1 = 0 \quad \ldots (6.16)$

For case two, ie. $(1+Z^TZ)<\sqrt{\lambda/\alpha}$, then

$$u_{z1}=0$$

$$u_{z2\text{-}3} = \pm \sqrt{\sqrt{\lambda/\alpha} - (1+Z^T Z)} \qquad \ldots (6.17)$$

$$u_{\_} = \pm j \sqrt{\sqrt{\lambda/\alpha} + (1+Z^T Z)} \qquad \ldots (6.18)$$

then $$u_i(k-1) = \mathrm{sign}(u_C)\,|u_{z2}| \qquad \ldots (6.19)$$

To select u_i whose sign is the same as u_C is based on the fact that u_i in (6.19) will give a smaller value of V than $-u_i$.

We are applying root locus technique according the above two different cases, and assume u_C is bounded, then we have

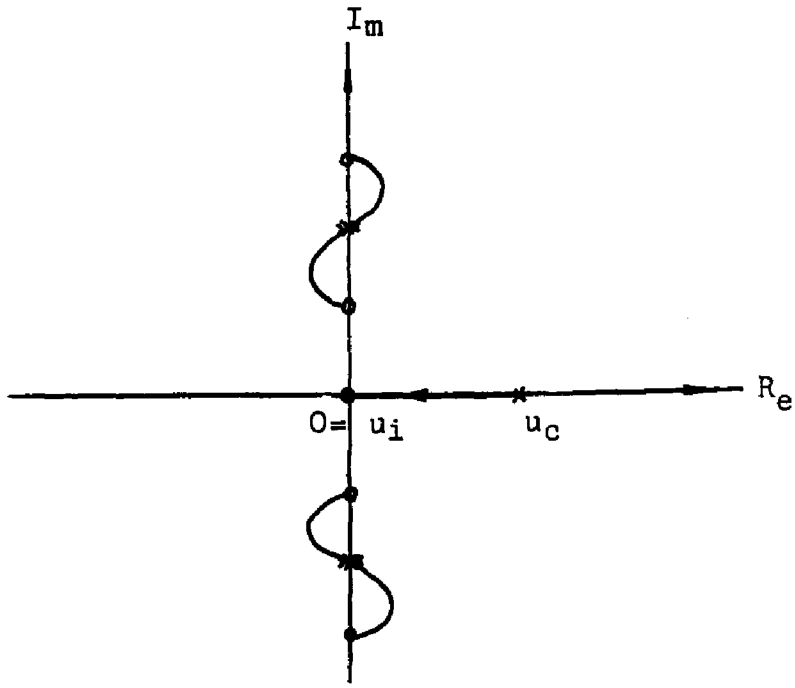


Figure 6.1. The root locus for case one in (6.16)

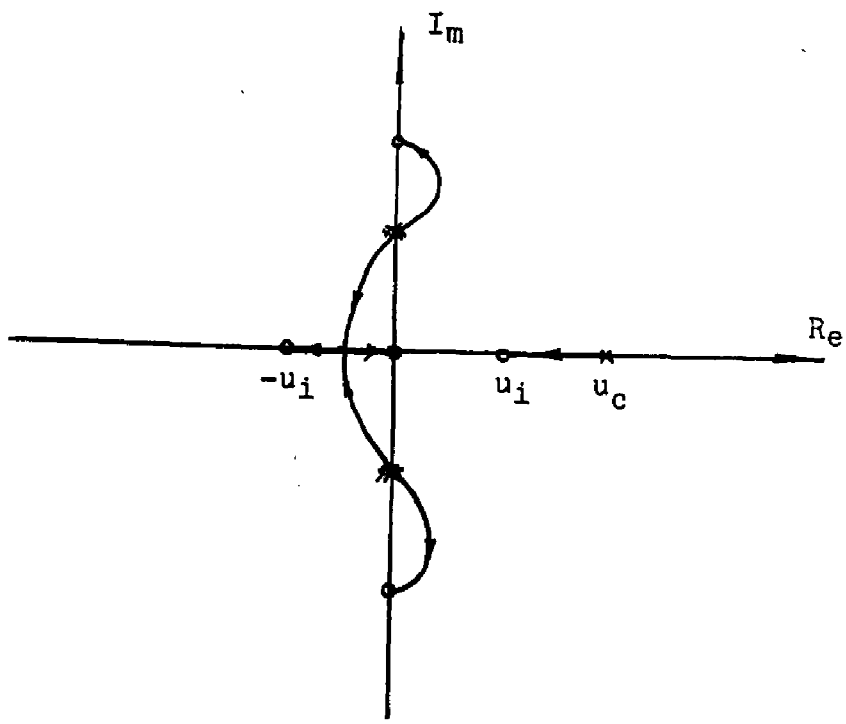


Figure 6.2. The root locus for case two in (6.19)

From all the above cases, we can see that: (1) There is only one solution which is located between u_c and u_i. For case two, there may be another possibility, ie. u(k-1) could be located on the other side of the real axis, but values of u(k-1) closer to u_c will give smaller values of the total performance index V(k). For all above possible cases, we conclude that there is one and only one real solution located between u_c and u_i. (2) The Figs. (6.1)-(6.3) are based on Eqn. (6.12). The weighting scalar α functions as an open loop gain. As α is very small, u(k-1) is close to u_c, otherwise it is close to u_i. As we noticed above, the value of α cannot be too large, otherwise the system may lose control (see discussion later). The weighting scalar λ, from Eqns. (6.16)-(6.19), moves the ending point of the root locus, ie. u_i, toward or away from u_c. (A method for selecting values for α and λ will be given below.)

Next we briefly describe the stability lemma for later use.

Lemma 2: Stability lemma (Goodwin) [5]:

This lemma was introduced by Goodwin. We list it here without proof. Readers who are interested in that proof should see [5].

With the projection parameter estimation algorithm of the system (6.1), and for some positive numbers β_1 and β_2, if

$$\| h^T(k-1) \| < \beta_1 |e(k)| + \beta_2 \qquad \ldots (6.20)$$

for all k, then $e(k) \longrightarrow 0$, as $k \longrightarrow \infty$, and u(k) and y(k) are uniformly bounded [a simple explanation was given in chapter 4].

Now we establish that sequences $\{u(k)\}$ and $\{u_c(k)\}$ are linearly bounded by e(k).

Lemma 3: With the same assumptions of lemma 1) and if the weighting scalars are selected as follows

$$\lambda(k-1) < \beta|e(k-1)|,$$

and $$\alpha(k-1) < \rho|e(k-1)|(u_c^2+z^Tz+1)^{-2} \qquad \ldots (6.21)$$

for some positive numbers β and ρ, then $y(k)$ and $u(k-1)$ are linearly bounded by $e(k)$, ie.

$$|y(k)| < \beta_3 + \beta_4|e(k)|$$

and $$|u(k-1)| < \beta_5 + \beta_6|e(k)|$$

for some β_3, β_4, β_5 and $\beta_6>0$ ... (6.22)

Proof: Let us assume that the weighting scalars α and λ are selected as in (6.21); also assume that the feedback system is not stable, ie. there exists a time instant t, so that for (k-1)>t, inputs u(k-1) and output y(k-1) are out of the bounds

$$|y(k-1)|>N_y \qquad \text{for some } N>0$$

$$|u(k-1)|>N_u \qquad \text{for some } N>0 \qquad \ldots (6.23)$$

Then the last two terms in (6.3b)

$$\frac{\lambda(k-1)'}{1+h^T(k-1)h(k-1)} < \varepsilon_1 \text{ for some finite number } \varepsilon_1>0$$

$$\text{and} \quad \alpha' u(k-1)^2 < \varepsilon_2 \qquad \text{for some finite number } \varepsilon_2>0 \qquad \ldots (6.24)$$

(6.24) is based on the following facts: (i)

$$|e(k-1)| \le \| h(k-2) \| . \| \tilde{\theta}(k-2) \| \le \| h(k-2) \| . \| \tilde{\theta}(0) \|$$

$$\text{then} \quad \lambda \le \beta \ |e(k-1)| \le \beta' \| h(k-2) \| \quad \text{for some } \beta'>0 \qquad \ldots (6.25)$$

and (ii) Relations between α and α', λ and λ' can be found in (6.4b). (iii) The denominator in (6.24) has the property

$$(1+h^T(k-1)h(k-1)) \ge \| h(k-1) \|^2 \qquad \ldots (6.26)$$

(6.24) shows that if (6.23) occurs with the selection (6.21), the feedback system will become the conventional pole-placement control system because the overall performance index becomes only the first two terms in (6.3b). Then the feedback system will be stabilized by the conventional pole-placement controller. This result will conflict with the equation (6.23). That proves the lemma 3).

The above discussion is a special consideration for nonminimum phase systems, due to the unstable inverse model B. In fact if we define

$$u(k-1) = u_c(k-1) + \Delta(k-1) \qquad \ldots (6.27)$$

Recall equations (6.13), (6.14) and (6.7), and rewrite $\hat{y}(k)$ as follows

$$\hat{y}(k) = h^T(k-1)\ \hat{\theta}(k-1)$$

$$= [1-\hat{A}(q^{-1})]y(k) + \hat{B}(q^{-1})[u_c(k)+\Delta(k)]$$

Substituting (6.7) into the above relation yields

$$\hat{A}(q^{-1})\hat{y}(k) = [1-\hat{A}(q^{-1})]e(k)+\hat{B}(q^{-1})[u_c(k)+\Delta(k)] \qquad \ldots (6.28)$$

Substituting (6.13) into (6.28) yields

$$(\hat{P}\hat{B}+\hat{Q}\hat{A})u_c(k) = \hat{P}\hat{A}y^*-\hat{P}[1-\hat{A}]e(k)-\hat{P}\hat{B}\Delta(k)$$

$$(\hat{P}\hat{B}+\hat{Q}\hat{A})\hat{y}(k) = \hat{P}\hat{B}y^*+\hat{Q}(1-\hat{A})e(k)+\hat{Q}\hat{B}\Delta(k) \qquad \ldots (6.29)$$

Because $\hat{P}\hat{B}+\hat{Q}\hat{A}$ is a known stable polynomial A*, then

$$|u_c(k-1)| < \rho_1 + \rho_2|e(k)| + \rho_3|\Delta(k-1)|$$

$$|\hat{y}(k)| < \rho_4 + \rho_5|e(k)| + \rho_6|\Delta(k-1)| \qquad \ldots (6.30)$$

for some positive numbers ρ_1, ρ_2 , ρ_3, ρ_4, ρ_5 and ρ_6.

From lemma 3 and (6.14), we concluded that u(k-1) and y(k) are linearly bounded by e(k) for all k. Also we conclude that $\hat{y}(k)$ is linearly bounded by e(k). From equation (6.30), $\Delta(k-1)$ is linear bounded by e(k). Furthermore, the following inequality holds

$$|u_c(k)| < \beta_3 + \beta_4|e(k)| \qquad \text{for some } \beta_3,\ \beta_4 > 0 \qquad \ldots (6.31)$$

Theorem: With the same assumptions as lemma 1), the adaptive control law (6.6), (6.8) with the weighting scalars α and λ satisfying (6.21), stabilizes the system (6.1), ie. y(k) and u(k) are uniformly bounded. Furthermore

$$A^* \, y(k) - \hat{P}\hat{B}y^* = 0 \qquad \text{as } k\longrightarrow\infty. \qquad \ldots (6.32)$$

Proof: We have established that the sequences {y(k)} and

$\{u(k)\}$ are linearly bounded by $e(k)$. It is clear that the regression vector $h(k-1)$ must be linearly bounded by $e(k)$, ie.

$$||h(k-1)|| < \rho_4 + \rho_5 \max_{1<\tau<k-1} |y(\tau)| + \rho_6 \max_{1<\tau<k-1} |u(\tau)|$$

$$< \rho_7 + \rho_8 \max_{1<\tau<k-1} |e(\tau)| \qquad \ldots (6.33)$$

Hence, from the stability lemma, the sequences $\{u(k)\}$ and $\{y(k)\}$ are uniformly bounded, and $e(k) \longrightarrow 0$ as $k \longrightarrow \infty$.

Furthermore, if the weighting scalars satisfy (6.21), clearly α and λ will approach zero when $e(k) \longrightarrow 0$, as $k \longrightarrow \infty$. Then the total performance (6.3b) reduces to V_c only. And $u(k) \longrightarrow u_c(k)$ as $k \longrightarrow \infty$. Equation (6.32) will be satisfied as $k \longrightarrow \infty$. (See [5] for the proof of (6.32).)

Remark 1): If $\hat{A}$ and $\hat{B}'$ have a common factor, equation (6.9) is not solvable for general A^*. A special treatment, for example, set $\hat{\theta}(k) = \hat{\theta}(k-1)$, has to be applied such that the continuation of the calculation can be done.

Remark 2): The initial conditions of θ are decided such that the Sylvester matrix is not singular at $k=0$. This restriction is not critical.

6.4) Numerical Simulations

This simulation example shows that, first, the designed dual control law yields better performance than that of the conventional pole-placement adaptive control law due to the contribution of the parameter estimation component of the dual control input. Second, the example demonstrates how the selection of α and λ affects the parameter estimation accuracy and the system tracking response.

The system selected for the numerical simulation is

$$y(k) = 2.0y(k-1) - 0.99y(k-2) + 0.5u(k-1) + 1.0\,u(k-2) \qquad \text{... (6.36)}$$

Obviously (6.36) is a nonminimum phase system. The system poles are 1.1 and 0.9, and the system zeros are 0.0 and

-2.0. The initial conditions of the unknown coefficients are listed in the following table:

Table 1

Parameters	a_1	a_2	b_1	b_2
True value	2.0	-0.99	0.5	1.0
Initial value	1.0	-0.5	0.1	0.2

y^* is a unit squared pulse for each 50 steps, $A^*=1$.

$$\alpha(k-1) = |e(k-1)|/[1+z^T(k-1)z(k-1)]^{-2}$$

$$\lambda(k-1) = K\ |e(k-1)| \qquad \ldots (6.37)$$

Figures 6.3) and 6.4) show the comparisons of the system outputs and the estimated parameters between the dual adaptive control algorithm and a conventional pole-placement adaptive control, with the projection parameter estimation algorithm. Figures 6.5) and 6.6) show the comparison of the system outputs and the estimated parameters with different selection of K.

From these simulation plots we can see that:

i) For a proper selection of α and λ (Fig. 6.3 and 6.4), at the beginning of the simulation, the dual control algorithm yields a bigger transient response than does the conventional algorithm, but it attains a better parameter estimate. After the initial transient both the tracking response and parameter estimation are improved.

ii) The dual control with a very large value of K generates a larger transient response and also takes a longer time to settle down (Fig. 6.5). Later it achieves good parameter estimates (Fig. 6.6).

iii) With a large value of α, the dual control reduces the size of u(k-1). For an unstable controlled system, the selection of α is an important matter.

6.5) Conclusion

This chapter presented a stable dual adaptive control algorithm which for nonminimum phase systems. The above stability proof showed that if the weighting parameters in the performance index, α and λ, are properly selected, then the designed control system can compromize very well between tracking and parameter estimation performance objectives. Finally, simulation results were presented to show the improvement in tracking and parameter estimation.

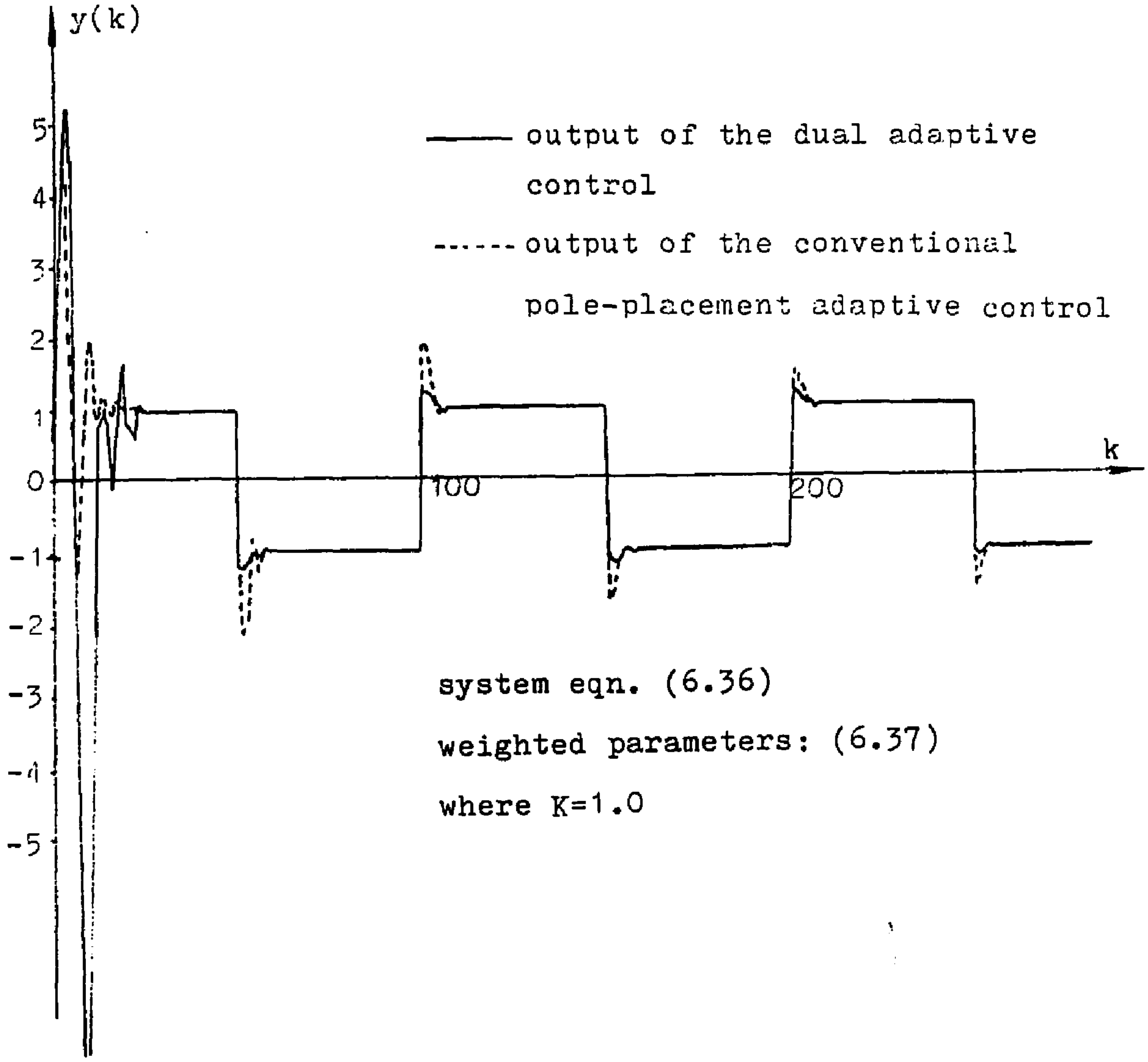


Fig. 6.3 The comparison of the system outputs between the dual adaptive control and the conventional pole-placement control for a nonminimum system.

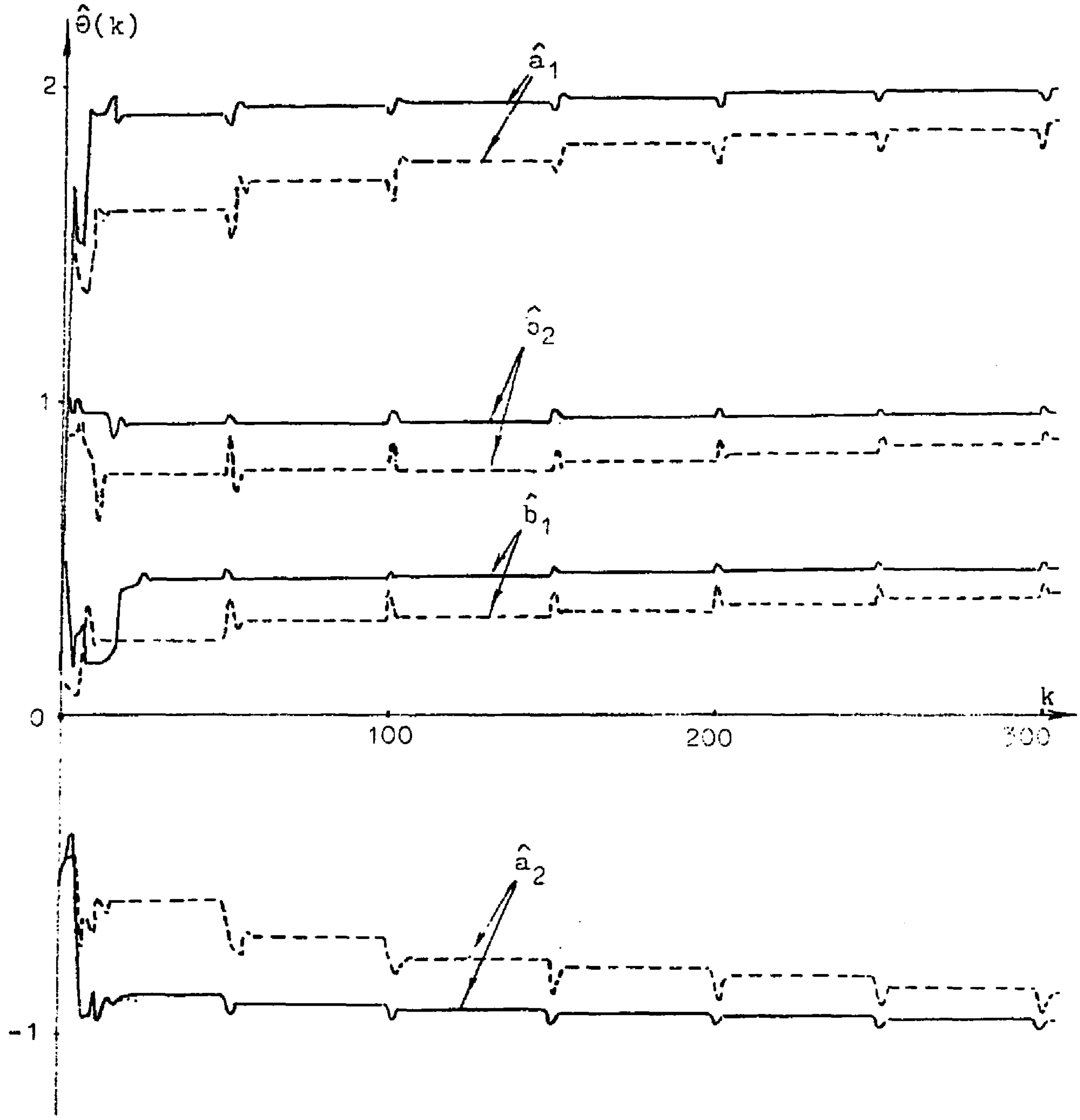
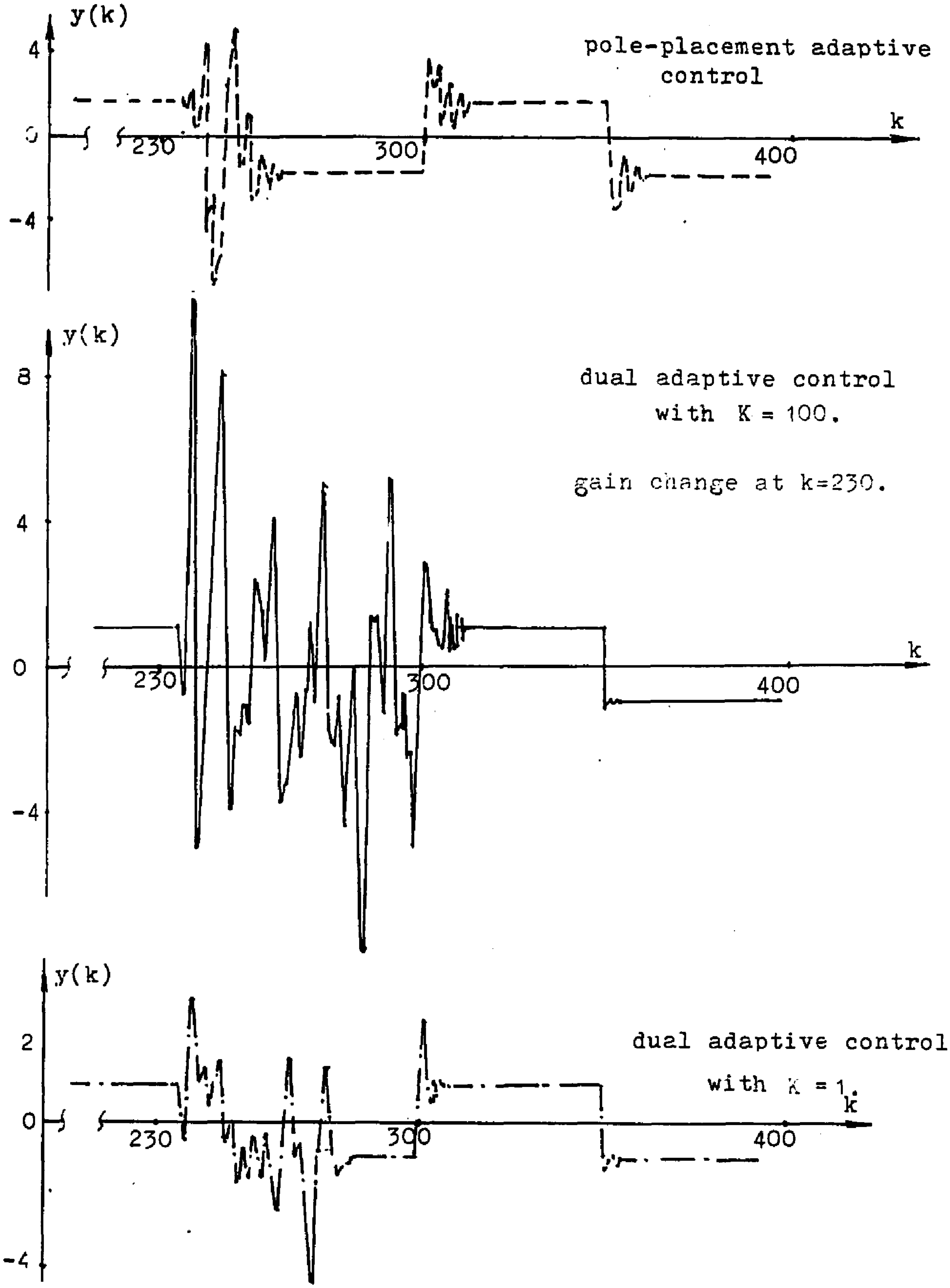


Fig. 6.4 The comparison of the parameter estimation between the dual adaptive control and the conventional pole-placement control for a nonminimum system.

Fig. 6.5 The comparison of the system outputs between the dual adaptive control and the conventional pole-placement control for the same system as above with the system gain change at k=230.


y(k)
4
0
-4
230
300
400
k
pole-placement adaptive control
y(k)
8
4
0
-4
230
300
400
k
dual adaptive control with K = 100.
gain change at k=230.
y(k)
2
0
-4
230
300
400
k
dual adaptive control with K = 1.

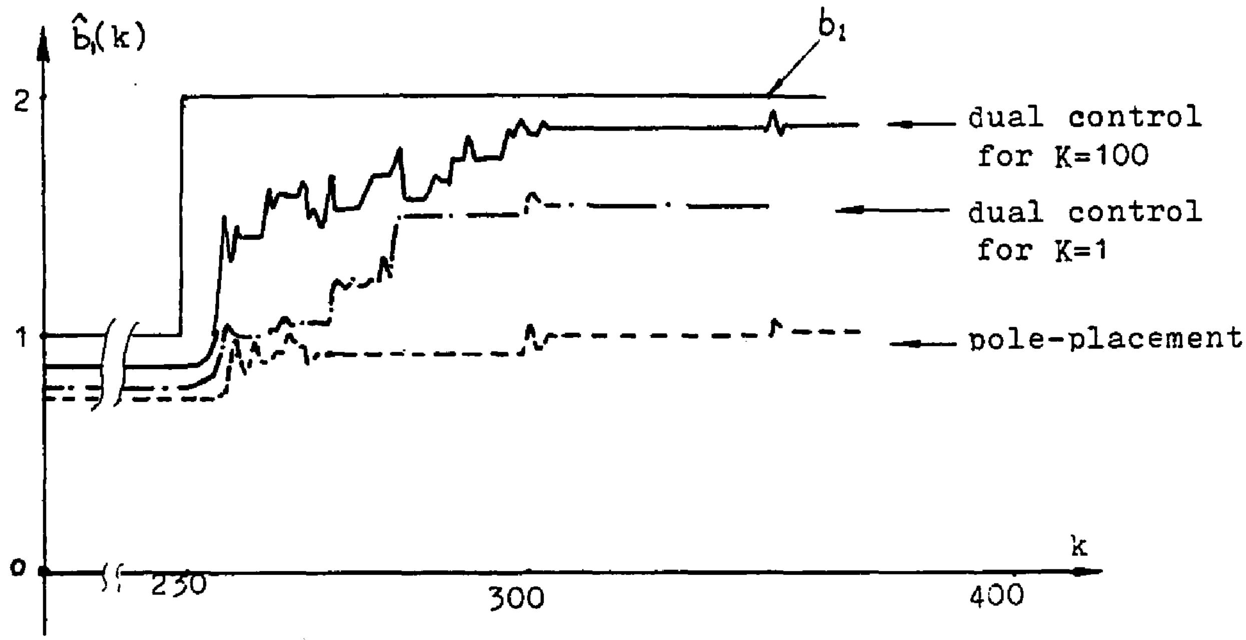


Fig. 6.6 The comparison of the estimated parameters between the dual adaptive control and the conventional pole-placement control for the same system as above with the system gain change at k=230.

Chapter 7

An approximately 'Error-Free' Parameter Estimation

7.1) Introduction:

In the above chapters we concentrated our efforts on designing an input that will compromize between control or tracking and estimation performance objectives. From the simulation results and the convergence proof we can see that one step dual control improves the performance between control and parameter estimation. But it still can not guarantee that the parameter error converges to zero.

In this chapter, we investigate a parameter estimation algorithm by which the parameter error goes to zero in some finite number of steps.

As we mentioned in the previous chapters, all of the conventional parameter estimation algorithms have the following form:

$$\hat{\theta}(k)=\hat{\theta}(k-1)+K(k-1)e(k)$$

where $$e(k)=y(k)-\hat{y}(k)$$

and $$\hat{y}(k)=h^T(k-1)\hat{\theta}(k-1) \qquad \ldots (7.1)$$

Vectors $h(k-1)$ and $\hat{\theta}(k)$ are defined as in (2.1).

The gain $K^T(k)=[\ k_1(k),\ k_2(k),\ldots,k_{n+m}(k)\ ]$ is determined recursively (see the projection algorithm or least-square algorithm). The equation error $e(k)$ may be written as

$$e(k)=h^T(k-1)\theta-h^T(k-1)\hat{\theta}(k-1)$$

$$=h^T(k-1)\tilde{\theta}(k-1) \qquad \ldots (7.2)$$

where $\tilde{\theta}(k) = \theta - \hat{\theta}(k)$. Then from (7.2) and (7.1),

$$\tilde{\theta}(k)=\tilde{\theta}(k-1)-K(k-1)h^T(k-1)\tilde{\theta}(k-1)$$

$$=[I-K(k-1)h^T(k-1)]\tilde{\theta}(k-1)$$

$$=B(k-1)\tilde{\theta}(k-1) \qquad \ldots(7.3)$$

The matrix $B(k-1)$ always has some eigenvalues on the unit circle, no matter what values of the elements of $K(k-1)$ are.

Then $\tilde{\theta}(k)$ may not converge to zero due to the unit valued eigenvalues. The following section presents a design technique for a parameter estimation algorithm based on achieving zero eigenvalues of a multistep parameter error system (will be seen later). Some simulation results are presented in the following sections to show the improved performance in parameter estimation and in adaptive control using this new parameter estimation algorithm.

7.2) The Main Result:

Consider a first-order linear discrete time system[®]

$$y(k)=ay(k-1)+bu(k-1)$$

$$=[y(k-1),u(k-1)]\begin{pmatrix} a \\ b \end{pmatrix} \qquad \ldots (7.4)$$

with a and b are unknown and the parameter estimate after two steps is

---

® We start with a first order system first for simplification of the derivation. Later, the more general case will be developed.

$$\hat{\theta}(k+1)=\hat{\theta}(k)+K(k)e(k+1)$$

$$=\hat{\theta}(k-1)+K(k-1)e(k)+K(k)e(k+1) \qquad \ldots (7.5)$$

The parameter error system can be written as

$$\tilde{\theta}(k+1)=\tilde{\theta}(k-1)-K(k-1)h^T(k-1)\tilde{\theta}(k-1)-K(k)h^T(k)\tilde{\theta}(k)$$

Consider $\tilde{\theta}(k)=\tilde{\theta}(k-1)-K(k-1)h^T(k-1)\tilde{\theta}(k-1)$

then $\tilde{\theta}(k+1)$ will be

$$\tilde{\theta}(k+1)=[I-K(k)h^T(k)][I-K(k-1)h^T(k-1)]\tilde{\theta}(k-1)$$

$$=B(k)B(k-1)\tilde{\theta}(k-1)$$

$$=A(k,k-1)\tilde{\theta}(k-1) \qquad \ldots (7.6)$$

If one can select K(k-1) and K(k), such that the nullity of A(k,k-1), the dimension of its null space is the same as the dimension of A(k,K-1)), see (7.14), then $\tilde{\theta}(k+1)$ will be zero. It means that $\hat{\theta}(k+1)=\theta$ [28].

Note that

$$A(k,k-1)=[I-K(k)h^T(k)-K(k-1)h^T(k-1)$$

$$+K(k)h^T(k)K(k-1)h^T(k-1)] \qquad \ldots (7.7)$$

The characteristic equation of A(k,k-1) is

$$|zI-A(k,k-1)| = z^2 - [(1-K^T(k)h(k))+(1-K^T(k-1)h(k-1))+ K^T(k-1)h(k)K^T(k)h(k-1)]z + (1-K^T(k)h(k)) (1-K^T(k-1)h(k-1)) = 0 \qquad ...(7.8)$$

The necessary condition for zero eigenvalues of (7.8) is

$$1-K^T(k)h(k)=0 \qquad ...\ (7.9a)$$

$$1-K^T(k-1)h(k-1)=0 \qquad ...\ (7.9b)$$

$$K^T(k)h(k-1)=0 \qquad ...\ (7.9c)$$

$$K^T(k-1)h(k)=0 \qquad ...\ (7.9d)$$

In the another form, equations (7.9) can be written as

$$H_2(k,k-1)K(k-1) = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad ...\ (7.10a)$$

$$H_2(k,k-1)K(k) = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \qquad ...\ (7.10b)$$

where

$$H_2 = \begin{pmatrix} h^T(k-1) \\ \\ h^T(k) \end{pmatrix} \qquad \ldots (7.11)$$

To solve (7.10) for K(k) and K(k-1) requires that

$$\mathrm{Det}\ (H_2) \neq 0 \qquad \ldots (7.12)$$

ie.

$$u(k) \neq \frac{y(k)u(k-1)}{y(k-1)} \qquad \ldots (7.13)$$

To verify the nullity of matrix A in (7.7), one can substitute the gain vectors that satisfy (7.10) into (7.7) to yield

$$A(k,K-1) = [I-H_2^{-1}\begin{pmatrix} 0 \\ 1 \end{pmatrix}h^T(k) - H_2^{-1}\begin{pmatrix} 1 \\ 0 \end{pmatrix}h^T(k-1) +$$

$$H_2^{-1}\begin{pmatrix} 0 \\ 1 \end{pmatrix}h^T(k)K(k-1)h^T(k-1) \qquad \ldots (7.14)$$

The sum of of the second and third terms is the negative identity matrix. In the last term, the factor $h^T(k)K(k-1)$ is zero from (7.9d). Then $A(k,k-1)=0$. Clearly, $\tilde{\theta}(k+1)=0$ from (7.6), ie. $\theta=\hat{\theta}(k+1)$. After two steps from time k-1, the unknown system parameters a and b in (7.4) will be identified at time k+1.

To extend the derivation of this algorithm to any nth order system, let us consider the three-parameter and three-step case now. The system with three parameters is as follows

$$y(k) = a_1y(k-1)+a_2y(k-2)+bu(k-1)$$

$$=h^T(k-1)\theta \qquad \ldots \ (7.15)$$

and the parameter estimation for three steps is

$$\hat{\theta}(k+2)=\hat{\theta}(k-1)+K(k+1)e(k+2)+K(k)e(k+1)+K(k-1)e(k) \qquad \ldots \ (7.16)$$

(7.16) is the algorithm that is implemented for the ARMA model with three unknown parameters.

The parameter estimation error system is

$$\tilde{\theta}(k+2)=[I-K(k+1)h^T(k+1))(I-K(k)h^T(k))(I-K(k-1)h^T(k-1))\tilde{\theta}(k-1)$$

$$=(I-K(k+1)h^T(k+1)-K(k)h^T(k)-K(k-1)h^T(k-1)+K(k+1)h^T(k+1)$$

$$K(k)h^T(k)+K(k+1)h^T(k+1)K(k-1)h^T(k-1)+K(k)h^T(k)K(k-1)$$

$$h^T(k-1)-K(k+1)h^T(k+1)K(k)h^T(k)K(k-1)h^T(k-1)]\tilde{\theta}(k-1)$$

$$=A(k+1,k,k-1)\tilde{\theta}(k-1) \qquad \ldots (7.17)$$

And

$$\mathrm{Det}(zI-A)=z^3-\{\sum_{i=-1}^{1}(1-K^T(k-i)h(k-i))+\sum_{\substack{i=-1\\j=-1}}^{1}{}_{i\neq j}[K^T(k-i)h(k-j)$$

$$K^T(k-j)h(k-i)]\}z^2+\{\sum_{\substack{i=-1\\j=-1}}^{1}{}_{i\neq j}(1-K^T(k-i)h(k-i))(1-K^T(k-j)h(k-j))$$

$$+K^T(k-1)h(k)K^T(k)h(k+1)K^T(k+1)h(k-1)\}z-\{(1-K^T(k+1)h(k+1))$$

$$(1-K^T(k)h(k))(1-K^T(k-1)h(k-1))\}=0$$

Similarly, the gain K can be found from the following equations

$$H_3(k+1,k,k-1)K(k-1)=\begin{pmatrix}1\\0\\0\end{pmatrix} \qquad \ldots (7.18a)$$

$$H_3(k+1,k,k-1)K(k)=\begin{pmatrix}0\\1\\0\end{pmatrix} \qquad \ldots (7.18b)$$

$$H_3(k+1,k,k-1)K(k+1)=\begin{pmatrix}0\\0\\1\end{pmatrix} \qquad \ldots (7.18c)$$

where

$$H_3(k+1,k,k-1)=\begin{pmatrix} h^T(k-1) \\ h^T(k) \\ h^T(k+1) \end{pmatrix}$$

To find unique gain vectors K(k-1), K(k) and K(k+1) requires that H_3 is not singular.

As we did in (7.14) to verify that matrix A in (7.17) is a zero matrix, one just simply substitutes the gain vectors that satisfy (7.18) into (7.17) to yield

$$A(k+1,k,k-1)=[I-H_3^{-1}\begin{pmatrix}0\\0\\1\end{pmatrix}h^T(k+1)-H_3^{-1}\begin{pmatrix}0\\1\\0\end{pmatrix}h^T(k)-H_3^{-1}\begin{pmatrix}1\\0\\0\end{pmatrix}h^T(k-1)$$

$$+K(k+1)h^T(k+1)K(k)h^T(k)+K(k+1)h^T(k+1)K(k-1)h^T(k-1)+$$

$$K(k)h^T(k)K(k-1)h^T(k-1)-K(k+1)h^T(k+1)K(k)h^T(k)K(k-1)h^T(k-1)]$$

With the same argument as in (7.14), one can conclude $A(k+1,k,k-1)=0$. Then from (7.17) and (7.16) we obtain $\hat{\theta}(k+2)=\theta$.

It can be seen that this development can be extended to any nth order system. The gains K(i) for (n+m)-step parameter estimation of an nth order linear system

$$y(k)=a_1 y(k-1)+\ldots+a_n y(k-n)+b_1 u(k-1)+\ldots+b_m u(k-m) \quad \ldots (7.19)$$

satisfy

$$K(k-1)=H_{n+m}^{-1}\begin{pmatrix}1\\0\\\vdots\\0\end{pmatrix}$$

$$\vdots$$

$$K(k+n+m-2)=H_{n+m}^{-1}\begin{pmatrix}0\\\vdots\\0\\1\end{pmatrix} \quad \ldots (7.20)$$

The gain vectors obtained through the solution to (7.20) are used for the following (n+m)-step parameter estimation algorithm

$$\hat{\theta}(k+n+m-1)=\hat{\theta}(k-1)+K(k-1)e(k)+\ldots+K(k+n+m-2)e(k+n+m-1) \quad \ldots (7.21)$$

7.3) On-line Test for Obtaining the Nonsingular Matrix H

Equation (7.20) shows that the calculation of the gains K(i) in the multistep parameter estimation (7.21) requires the non-singularity of matrix H_{n+m}. The one thing that has not been discussed is that how to decide that matrix H_{n+m} is not singular before calculating the gains K(i). In order to develop a general procedure for obtaining a non-singular matrix H_{n+m} let us examine the following example first.

Example 1: Consider the following three parameter case, let

h_1^T:	y(k-1)	y(k-2)	u(k-1)
h_2^T:	y(k)	y(k-1)	u(k)

be a subset of two 3-vectors in H_3. u(k) is the only quantity that can be changed at time k. There are two possibilities: (1) if $y(k-1)/y(k) \neq y(k-2)/y(k-1)$, u(k) can be any value for linear independence. (2) If $y(k-1)/y(k) = y(k-2)/y(k-1)$, then the input $u(k) \neq y(k)u(k-1)/y(k-1)$ and also y(k+1) which is created by u(k) should be $y(k+1)/y(k) \neq y(k)/y(k-1)$ so that H_3 is not singular. If one always considers the worst case-the case (2) - the new input u(k) is constrained by

$$u(k) \neq \frac{y(k)u(k-1)}{y(k-1)} \text{ or } \frac{y(k-1)u(k-1)}{y(k-2)} \quad \ldots (7.22)$$

And y(k+1), which is generated by u(k), is not available at time k. One can use the predicted output $\hat{y}(k+1)$, ie

$$\hat{y}(k+1) \neq y(k)^2/y(k-1) \quad \ldots (7.23)$$

(7.22) and (7.23) are the general constraints for u(k). With the constraints (7.22) and (7.23), the non-singular matrix H_3 can be easily obtained by selecting u(k) and u(k+1).

$$\text{Det}(H_3)=u(k-1)[y(k)^2-y(k+1)y(k-1)]-u(k)[y(k)y(k-1)-y(k+1)y(k-2)]+u(k+1)[y(k-1)^2-y(k)y(k-2)] \quad \ldots (7.24)$$

With the constraints (7.22) and (7.23), the first two terms of the right side in (7.24) are not zero if u(k-1) and u(k) are not zero. The last term is zero for case (2). If one always considers the worst case, then the conditions (7.22) and (7.23) are necessary for $Det(H_3) \neq 0$ (if the parameter estimates are close enough to the true parameters so that $\hat{y}(k+1) \approx y(k+1)$).

To extend this particular example to any nth order system, the procedure for the on-line identification procedure is the following: At time k-1, a non-zero input u(k-1) is selected upon any feedback control law with the knowledge of $\hat{\theta}(k-1)$ to form vector $h^T(k-1)$. At the next sampling time k, y(k) is measured and e(k) is calculated. u(k) is carefully selected using a procedure to be specified below so that it has to meet the basic requirements (7.22) and (7.23) (for the worst case) to form $h^T(k)$. In the mean time, the linear independence of the all of the row vectors $h^T(k+i-1)$ $(i=1,..,j,$ for $j \leq n+m-1)$ should be tested. If the rank of these row vectors is not full, then change u(k).

As the rank is full, then go to next step. Repeat the above step until time is equal to k+n+m-2, ie. $h^T(k+n+m-2)$ is obtained. After the all of n+m vectors form a square matrix H_{n+m}, one can use (7.20) to find the gains K(i) and (7.21) to update the parameter estimates.

The key of the above procedure is that the input u(k) (for each time instant k) is decided so that the row vectors $h^T(i)$ must be linearly independent. Then at time k+m+n-2, the square matrix H_{n+m} is nonsingular. Before we discuss the on-line test for linear independence of these row vectors, we introduce the following definitions:

Def. 7.1: The number of degree of freedom, F (freedom number for short):

F is defined as a number of elements in an n-vector v(k) that can be selected freely at time k.

Example 2: If all elements of vector $v^T=(x_1(k),....x_n(k))$

can be changed at time k arbitrarily, then the freedom number of v is n.

Example 3: As we know in the vector

$$h^T(k)=[y(k),...y(k-n+1),\ u(k),...,u(k-m+1)]$$

there is only one element, ie. u(k), that can be selected at time k, so its freedom number F = 1.

Def. 2) Condition number N of a linearly independent set S of m n-row-vectors (condition number for short):

Let S be a set of m n-row-vectors (m≤n). The condition number N is defined as the largest number of mxm submatrices formed by adjacent columns in S whose determinants are not zero.

Example 4: Consider that S is a set of two 4-row-vectors:

$$S: \left\{ \begin{array}{cccc} a & a & b & b \\ c & a & x & b \end{array} \right.$$

There are three 2x2 submatrices formed from adjacent columns in S:

$$\begin{pmatrix} a & a \\ c & a \end{pmatrix}, \quad \begin{pmatrix} a & b \\ a & x \end{pmatrix} \text{ and } \begin{pmatrix} b & b \\ x & b \end{pmatrix}$$

If $a \neq c$ and $x \neq b$, then the determinants of all of these three 2x2 submatrices are not zero. The condition number=3. Corresponding to vector $h^T(k)=[y(k), y(k-1), u(k), u(k-1)]$, $c=y(k)$ and $x=u(k)$. If $a=c$, it corresponds to the case (2) in example 1). x must be selected $x \neq b$ such that the two row vectors in S are linearly independent. In this case the maximum condition number N=2. From the above example, we can see that the condition number N is a measure of linear independence of a set of row vectors.

Example 5: Furthermore consider that S is a set of 3 4-vectors

$$S: \left\{ \begin{matrix} a & a & b & b \\ a & a & x & b \\ a & a & y & x \end{matrix} \right.$$

There are two 3x3 submatrices to be considered in S, ie.

$$S_1: \begin{pmatrix} a & a & b \\ a & a & x \\ a & a & y \end{pmatrix} \qquad S_2: \begin{pmatrix} a & b & b \\ a & x & b \\ a & y & x \end{pmatrix}$$

$Det(S_1)=0$, and $Det(S_2)=a(x-b)^2$. If $a\neq 0$ and $x\neq b$, then $N=1$, otherwise $N=0$. Examples 5 shows that if selection of the value of the current input x (in the second row vector of S) does not consider the future output a (the first element in the third row vector), then it will be more difficult to select u(k) for linear independence of S.

Note 1: The maximum condition number of S_{mxn} is n-m+1.

Note 2: The larger condition number N is, the better linearly independent condition of a set of row vectors is. 'Better' means easier to select the current input u(k) so as to obtain a linearly independent set S (see examples 4 and 5).

Theorem 7.1: If the condition number N of an mxn ($m\leq n$) matrix is not zero, then the row vectors (or column vectors) in matrix S are linearly independent. Otherwise they are linearly dependent.

Proof: It is clear that if there is one or more $m \times m$ matrices in S whose determinant is not zero, that means rank(S)=m. Then m row vectors must be linearly independent. If the determinants of all $m \times m$ matrices are zero, which means that the rank of S is less than m. In this case the row vectors of S are linearly dependent.

Note 3: Let S be an independent subset of a vector space H, dim H=n and dim S=m. Suppose that β is a vector in H. If the set obtained by adjoining β to S is linearly dependent, then the determinant of each $(m+1)\times(m+1)$ matrix of this new set must be zero. If there is (are) one (more) non-zero determinant(s) of the $(m+1)\times(m+1)$ square matrices, then this set $(S \cap \beta)$ is linearly independent.

The significance of this note is that if S is considered as a set of row vectors $h^T(i)$, and β is a new added row vector $h^T(k)$, then one can test one of the $(m+1)x(m+1)$ matrices that includes the input $u(k)$, so that the determinant of this $(m+1)x(m+1)$ matrix is not zero. The non-zero determinant of any $(m+1)x(m+1)$ matrix guarantees that the new vector β is linearly independent to S. We can ignore to calculate the determinants of other matrices. In this way one can simplify the test procedure for obtaining a non-singular H.

Note 4: For a linearly independent subset S, it is required only finite number of steps to find the condition of the free-selected element in β (F=1), such that the new subset $S \cap \beta$ is linearly independent. This statement is based on the following fact, the new generated subset contains only one element that can be selected (u(k)). And one constraint, $\det(M) \neq 0$, for each matrix M. M is any $(m+1)x(m+1)$ matrix which contains u(k). For non-zero determinants of some $(m+1)x(m+1)$ matrices in $S \cap \beta$, there are a finite number of u(k)'s that make $\det(M)=0$. If also consider the future output (see example 1), it just add one more constraint to u(k). This note tells that in a finite number of searching steps one always can find an input so that $Det(H) \neq 0$.

Note 3 tells us that it may need more than one step to find the current input u(k) so that it satisfies (7.22), (7.23) and the full ranked subset of the row vectors $h^T(i)$. Note 4 tells that if it needs more than one step to find u(k), then it only needs a finite number of steps to find u(k).

Fig. 7.1 is the flow chart of the simulation procedure for an adaptive control (self-tuning) combining with the error-free parameter estimation. The basic idea of this diagram is: carefully select an input signal (see (7.25 and 7.26)), test the rank of the new combined row vector subset. If the rank is full, then the input is acceptable, otherwise the input should be modified until the rank is full. The technique for selecting u(k) is based on writing

$$u(k) = u_c(k)+\delta(k) \qquad \ldots (7.25)$$

where $$u_c(k) = \frac{y^* - \hat{C}^T(k)z(k)}{\hat{b}_1(k)}$$

y^* is the reference signal and the vectors $\hat{C}$ and z are defined in chapter 3. $\delta(k)$ is a modification part. And the future output $y(k+1)$ will be affected by different δ's, ie.

$$\hat{y}(k+1) = \hat{b}_1 u(k) + \hat{C}^T z(k)$$

Substituting (7.25) into the above equation yields

$$\hat{y}(k+1) = y^* + \hat{b}_1 \delta(k) \qquad \ldots (7.26)$$

As we mentioned in example 1, $\hat{y}(k+1)$ is constrained by

$$\hat{y}(k+1) \neq y(k)^2/y(k-1) \qquad \ldots (7.27)$$

The modification part $\delta(k)$ in (7.25) causes the estimated output $\hat{y}(k+1)$ to move away from the reference signal y^*, see (7.26). So by properly selecting $\delta(k)$ (7.22) and (7.23) can be satisfied. A practical technique for selecting $\delta(k)$ is

$$|\hat{b}_1 \delta(k)| \leq |y^*| \; 5\% \quad \text{if } y^* \neq 0$$

$$|\hat{b}_1 \delta(k)| \leq 0.05 \qquad \text{if } y^*=0 \qquad \ldots (7.28)$$

If we consider that (1) $\delta(k)$ should be different for different time k, (2) the time average of $\delta(k)$ should be zero, then we have

$$\frac{1}{n+m} \sum_{k=1}^{n+m} \delta(k) = 0 \qquad \ldots (7.29)$$

7.4) Simulations (self-tuning control plus 'error-free' parameter estimation):

In this section, some simulation results are presented by using the 'error-free' parameter estimation algorithm along with a self-tuning adaptive control law. There are two sets of simulation results (1) one set is for a first order system, ie. two parameter case. (2) One set for a second order system, ie. four parameter case.

Fig. 7.2 and Fig. 7.3 are related the following first order linear system

$$y(k) = ay(k-1)+bu(k-1)$$

$$\text{with} \qquad a = \begin{cases} 0.7 & K \le 230 \\ 1.1 & k > 230 \end{cases}$$

$$\text{and} \qquad b = \begin{cases} 0.5 & k \le 230 \\ 0.8 & k > 230 \end{cases} \qquad \ldots (7.30)$$

Fig. 7.2 plots the inputs and outputs of the system (7.30) with different parameter estimation algorithms. After two steps, the system output with the 'error-free' estimation algorithm achieves an almost perfect tracking. At k=230 parameters a and b change, y(k) with the 'error free' parameter estimation algorithm still obtains a perfect tracking after two-step delay. Fig.7.3 shows the comparison of parameter estimates between these two parameter estimation algorithms. The 'error-free' algorithm is quite superior to the conventional projection algorithm.

The second set shows the same result as the first set does for the system

$$y(k)=a_1y(k-1)+a_2y(k-2)+b_1u(k-1)+b_2(k-2)$$

$$a_1=1.9662, \qquad a_2=-1.0348 \qquad (7.31)$$

$$b_1=0.27 \quad \text{for } k\le 230 \quad \text{otherwise } 0.8$$

$$b_2=-0.27 \quad \text{for } k\le 230 \quad \text{otherwise } -0.8$$

The Fig. 7.4 is the comparison of the outputs with two different parameter estimation techniques. The result with 'error-free' is very superior to that with the projection algorithm. So is the parameter estimation, see Fig. 7.5.

7.5) Conclusion:

In this chapter we present a new parameter estimation algorithm which makes the parameter estimates converge to the true parameters in a finite number of steps. This algorithm may be combined with some feedback control laws to form a stable adaptive control algorithm, and the simulation results show that this algorithm is very effective. It may obtain an exponential stable parameter error system by setting the eigenvalues $0 < \lambda < 1$. The disadvantages that occurred during the simulations are 1) It is required to calculate determinants of matrices during on-line test for determination of nonsingular H. 2) The 'error-free' algorithm takes more computation time than the conventional self-tuning control. But it is still worth trying because of the convergence and accuracy of the parameter estimation. 3) Because this algorithm needs n+m steps to estimate the system parameters, the time delay restricts that this algorithm is used for unknown constant parameters.

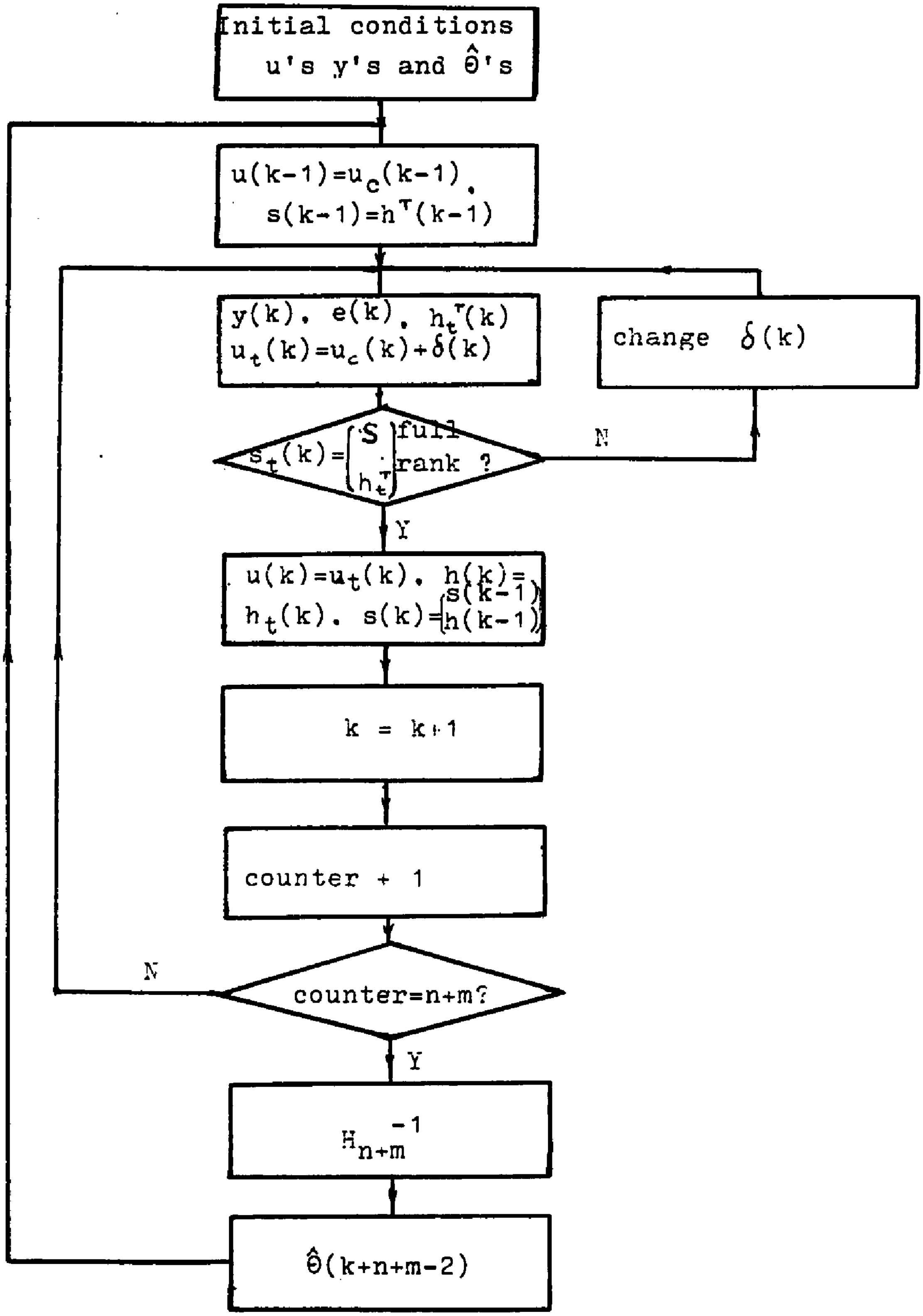
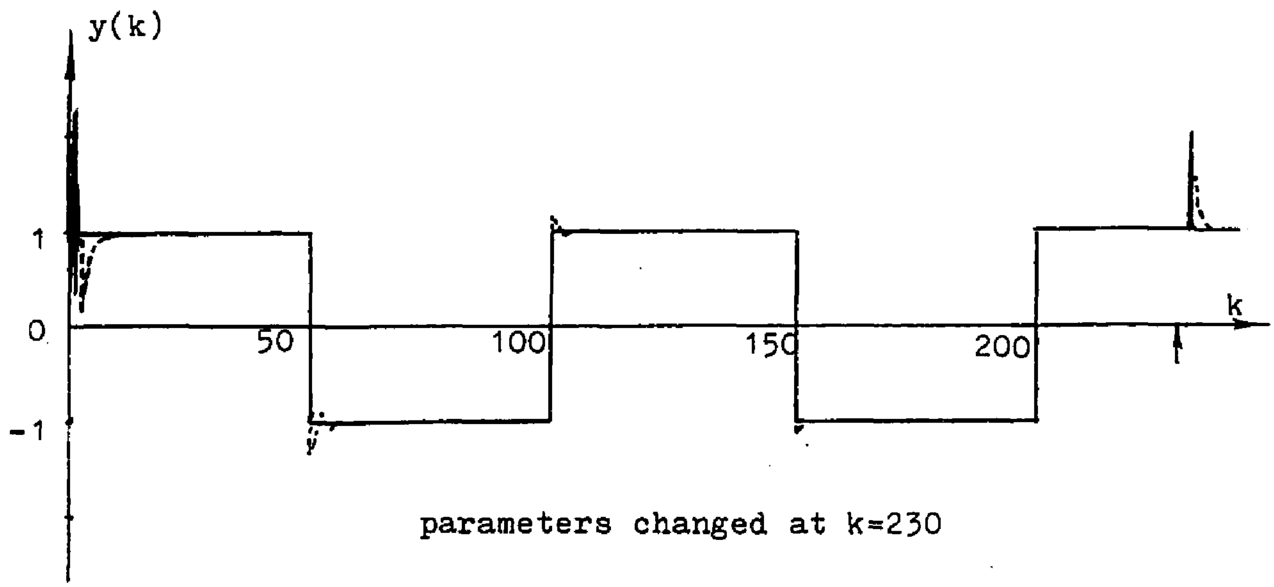


Fig. 7.1 Flow chart of the 'error free' parameter estimation.

$\hat{\theta}(0)=\begin{bmatrix}1.0\\0.2\end{bmatrix}, \qquad h(0)=\begin{bmatrix}0.5\\0.5\end{bmatrix}$



parameters changed at k=230

Fig. 7.2a The outputs of the self-tuning adaptive control with different parameter estimation techniques for a first order system (7.30) (the solid line is for 'error free' technique and dash line is for projection algorithm).

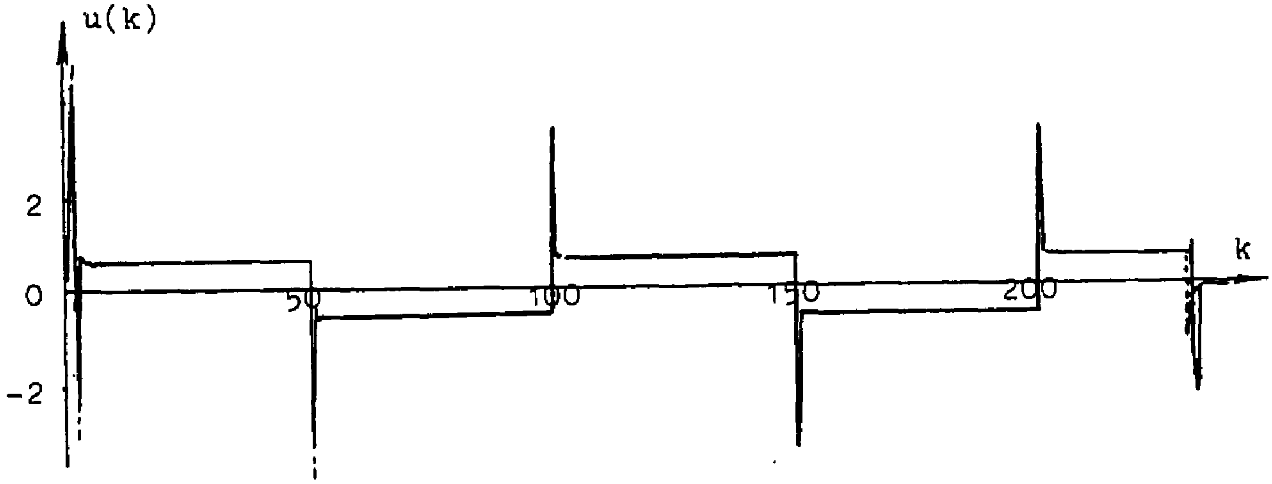


Fig. 7.2b The inputs of the self-tuning adaptive control with different parameter estimations for a first order system (7.30) (the solid line is for 'error free' technique and the dash line is for projection algorithm).

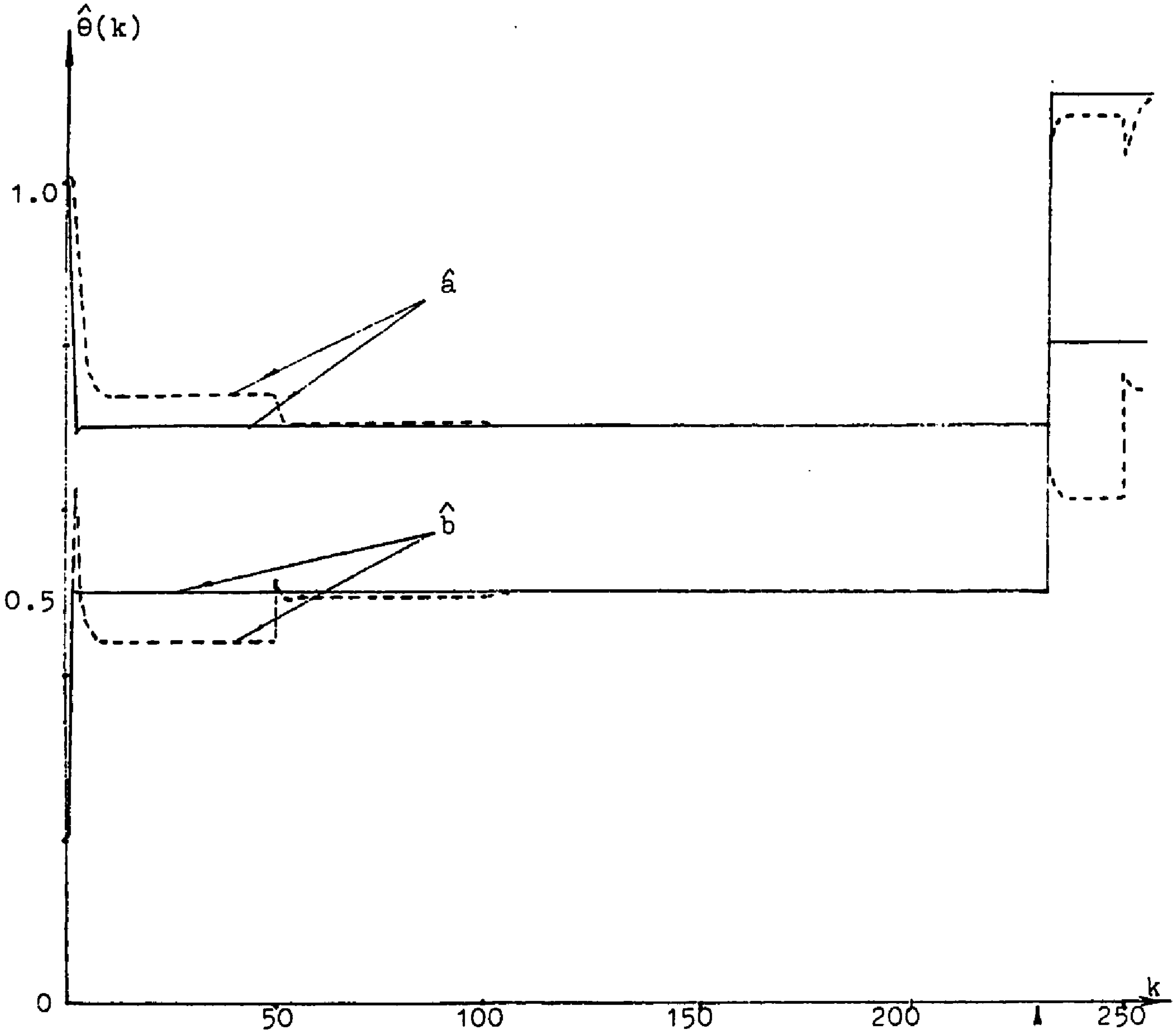


Fig. 7.3 The parameter estimation comparison of self-tuning adaptive control with different parameter estimation techniques for a first order system (7.30) (solid line represents 'error free' technique and dash line represents projection algorithm).

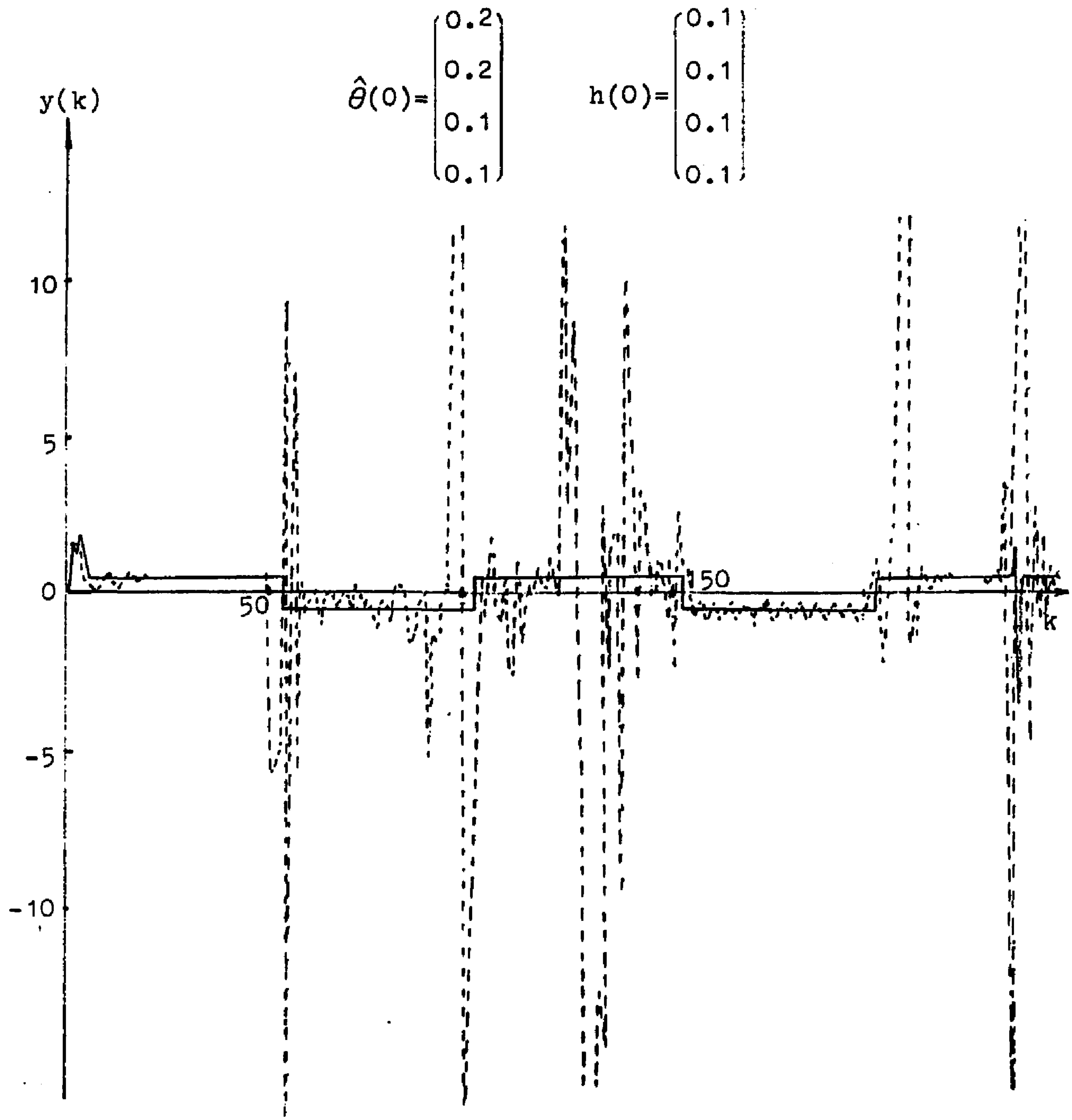


Fig. 7.4 The output comparison of self-tuning adaptive control with different parameter estimation techniques for 2nd order system (7.31) (solid line is for 'error free' technique and dash line represents projection algorithm).

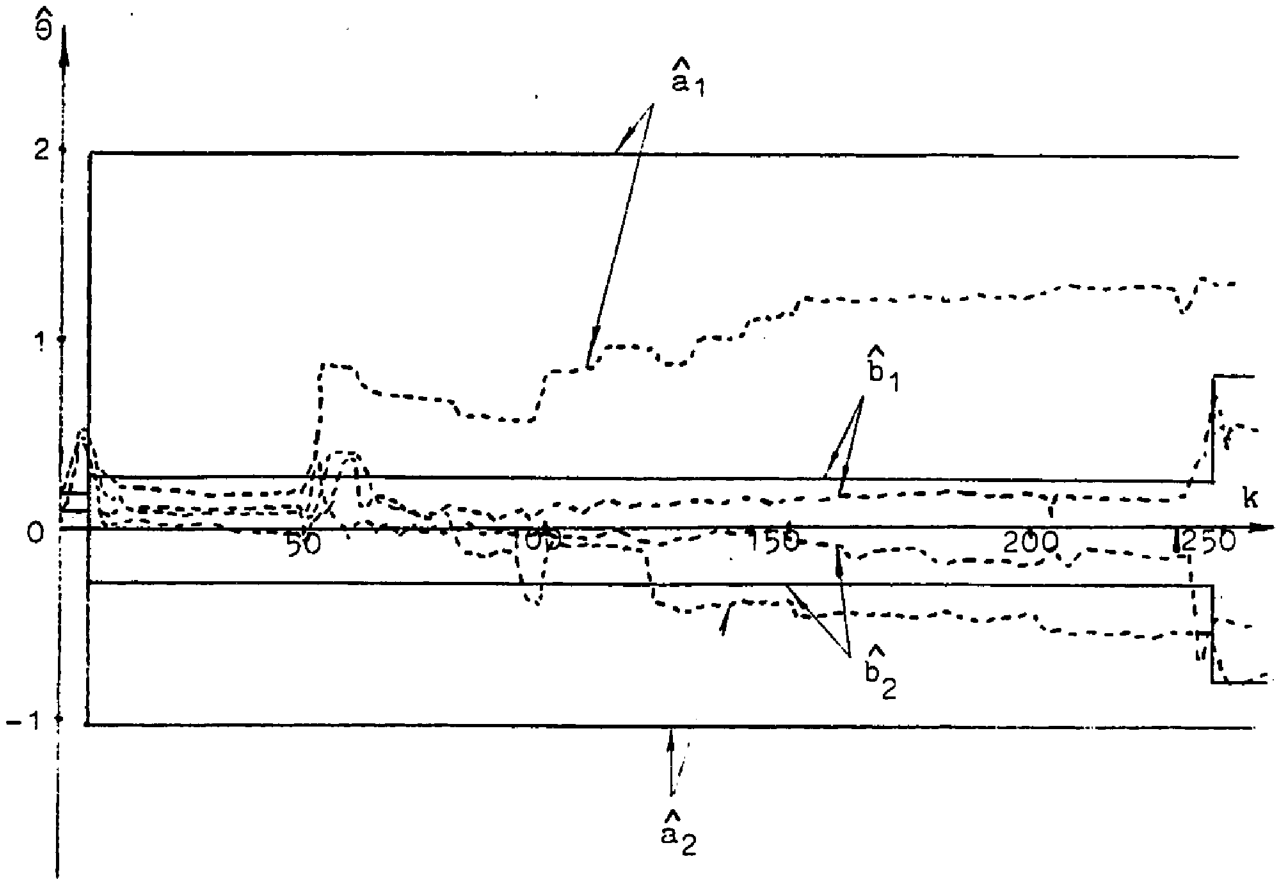


Fig. 7.5 The parameter estimation comparison of self-tuning adaptive control with different parameter estimation techniques for 2nd order system (7.31) (solid line represents 'error free' technique and dash line is for projection algorithm).

Chapter 8

Summary and Conclusion

8.1) Summary of Contributions

In the above chapters a suboptimum dual adaptive control algorithm for minimum phase and nonminimum phase systems was presented. Some simulation results and comparisons showed that the designed dual control has improved the performance in control and parameter estimation by having a dual functioning input. The major contribution which relates to the designed dual adaptive control is stability proof for minimum and nonminimum phase systems. As far as we know no stability proof for dual control systems has been presented yet in the adaptive control literature. The proof uses the key lemma to show the linear boundedness of the input and output by properly selecting the weighting parameters α and λ. From the derivation we also established a pair of bounds (u_c and u_1) for the input signal u(k). The bounds of u(k) are useful for applying the key lemma in the stability proof of the designed dual control algorithm. The time varying weighting scalars α and λ give the designer some flexibility to balance the regulation and parameter estimation. The disadvantage of this algorithm is that the designed procedure is based on the measurement of e(k). So the system output converges only if the weighting scalars α and λ are functions of e(k). But the parameter errors can not be guaranteed to be zero if the designed control algorithm uses one of the current parameter estimation algorithms. In chapter seven a multistep approximately 'error free' parameter estimation algorithm was presented which may be one of the ways to obtain approximately zero parameter error in a finite number of steps. The result of this algorithm is quite similar to the batch processing approach to parameter estimation. A major difference is that in the new algorithm presented in chapter 7 is that each step of the approximately 'error free' parameter estimation procedure consists of an on-line test so that each row vector $h^T(k)$ is independent of each other, thus verifying persistent excitation through an on-line procedure. A limitation of this algorithm is that it is suitable only for systems whose parameters change very slowly or whose parameters are constant. Another limitation is that the parameter estimation scheme assumes that the system order is known exactly so that the parameter error is zero in a finite number of steps.

8.2) Possible Extensions

We have thus far demonstrated the ability of the new adaptive control system to recover from abrupt parameter changes and to follow a desired reference signal. Future work on extending the methods developed here should include the following:

■ A new technique should involve the dual-adaptive control of processes with external disturbance inputs and measurement noise. The simulation results of chapters 5 and 7 are encouraging in this regard.

■ The approximately 'error free' parameter estimation algorithm [in chapter 7] should be converted from multistep into recursive form with self-exciting input signals to drive the parameter error to zero exponentially fast. Many attempts which are related to the above purpose have been tried recently by different authors. The recent Kreisselmeier paper [31] is an example of these attempts.

■ The combined performance index may be extended from one step to multistep. That may approach the case of optimal adaptive control.

■ Some dual control techniques have been developed in Chapter 3, eg. Taylor expansion, minmax principle, modified dual control. From the simulation results in Chapter 3, it shows that those techniques are effective, but there is a lack of analysis. Future analysis of the techniques of Chapter 3.4 should be developed.

References

[1] Astrom, K. J. and P. Eykhoff "System Identification-a Survey" <u>Automatica</u> 7, pp. 123-162 (1972)

[2] K. J. Astrom "Theory and Applications of Adaptive Control - a Survey" <u>Automatica</u> 19 pp. 471-486 (1983)

[3] R. Isermann "Parameter Adaptive Control Algorithm-a tutorial" <u>Automatica</u> 18, pp. 513-528 (1982)

[4] J. Mendel <u>Discrete Techniques of Parameter Estimation</u> New York, Marcel Dekker, 1973

[5] Goodwin, G. C. and K. C. Sin <u>Adaptive Filtering Prediction and Control</u> New Jersey, Prentice-Hall, (1984)

[6] Astrom, K. J., U. Borisson, L. Ljung and B. Wittenmark "Theory and Application of Self-tuning Regulator" <u>Automatica</u> 13, pp. 457-476 (1977)

[7] Goodwin, G. C. and H. Elliott " Deterministic Convergence of a Self-tuning Regulator with Covariance Resetting" <u>Proc. IEE</u> 130, pp. 6-8 (1983)

[8] Johnston, R. M., C. R. Johnson Jr., R. B. Bitmead and D. O. Anderson "Exponential Convergence of Recursive Least Squares with Exponential Forgetting Factor" <u>21st IEEE CDC</u> Orlando FL, pp. 994-997 Dec. (1983)

[9] P. Eykhoff <u>System Identification</u> Wiley, (1974)

[10] Astrom, K. J. and B. Wittenmark "On Self-tuning Regulators" Automatica 9, pp. 185-199 (1973)

[11] Clarke, D. W. and P. J. Gawthrop " Self-tuning Controller" Proc. IEE 122, pp. 929-934 (1975)

[12] Clarke, D. W. and P. J. Gawthrop "Self-tuning Control" Proc. IEE 126, pp. 633-640. (1979)

[13] I. D. Landau Adaptive Control - the Model Reference Approach New York, Marcel Dekker. (1979)

[14] Landau, I. D. and H. M. Silveira "A Stability Theorem with Applications to Adaptive Control" IEEE Trans. Automatic Control, AC-24 pp. 305-311 (1979)

[15] Narendra, K. S. and Y. H. Lin "Stable Discrete Adaptive Control" IEEE Trans. Automatic Control. AC-25 pp.456-461 (1980)

[16] Narendra, K. S., Y. H. Lin and L. S. Valavani "Stable Adaptive Controller Design, Part II: Proof of Stability" IEEE Trans. Automatic Control, AC-25 (1980)

[17] A. A. Feldbaum "Dual Control Theory I-IV" Automatic Remote Control 21 pp.1033 (1960)

[18] B. Wittenmark "Stochastic Adaptive Control Methods: a Survey" International Journal of Control, 21 pp. 705-730, (1970)

[19] Alster, J. and P. R. Belanger "A Technique for Dual Adaptive Control" Automatica 10 pp.627-634 (1974)

[20] Goodwin, G. C. and R. Payne " Dynamic System Identification: Experiment Design and Data Analysis" Mathematics in Science and Engineering 136, New York Academic (1977)

[21] Milito, A., C. S. Padilla, R. A. Padilla and D. Cadorin "An Innovations Approach to Dual Control" IEEE Trans. Automatic Control. AC-27 PP. 132-137 (1982)

[22] F. E. Thau "A Nonlinear Dual-adaptive Control Strategy for Identification and Control of Flexible Structures" Proceedings of the Workshop on Identification and Control of Flexible Space Structures III pp. 117-125 (1985)

[23] Bar-shalom, Y. and E. Tse "Dual Effect, Certainty Equivalence and Separation in Stochastic Control" IEEE Trans. Automatic Control. AC-19 pp. 494-500 (1974)

[24] Johnson, T. L., C. A. Harvey and G. Stein "Self-tuning Regulator Design for Adaptive Control of Aircraft Wing / Store Flutter" IEEE Trans. Automatic Control AC-27 pp. 1014-1023, (1982)

[25] D. W. Clarke "Self-tuning Control of Nonminimum-phase Systems" Automatica 20, pp. 501-517, (1984)

[26] Lozano-leal, R. and G. C. Goodwin "A Globally Convergent Adaptive Pole Placement Without a Persistency of Excitation Requirement" IEEE Trans. on Automatic Control AC-30, pp. 795-817 (1985)

[27] Kreisselmeier, G. and M. C. Smith "Stable Adaptive Regulation of Arbitrary nth-Order Plants" IEEE Trans. on Automatic Control AC-31 pp. 299-305 (1986)

[28] Hoffman, K. and R. Kunze Linear Algebra New Jersey 2nd Edition, Prentice Hall, (1971)

[29] Anderson, P. D. O. and R. M. Johnstone "Global Adaptive Pole Positioning" IEEE Trans. on Automatic Control, AC-30 pp. 11-22 (1985)

[30] Astrom, K. J. and B. Wittenmark Adaptive Control Addison-Wesley, (1988)

[31] Kreisselmeier, G. "An Indirect Adaptive Controller with a Self-excitation Capability" IEEE Trans. on Automatic Control, AC-34, (1989)