DOCTOR OF PHILOSOPHY

Self-Tuning Control for Bilinear Systems

Burnham, Keith

Award date:
1991

Awarding institution:
Coventry Polytechnic

Self-tuning Control

for

Bilinear Systems

K. J. Burnham B.Sc. M.Sc.

A Thesis submitted in partial fulfilment of the requirements of the Council for National Academic Awards for the degree of Doctor of Philosophy.

May 1991

Control Theory & Applications Group
Coventry Polytechnic

Dedication

This Thesis is dedicated to my brother

Guy Charles Burnham

who died tragically on 22nd May 1987

aged 18 years.

Summary

Prompted by the desire to increase the industrial applicability range of self-tuning control, the objective of this work has been to extend the standard linear self-tuning framework to facilitate the design of self-tuning controllers for bilinear systems. Bilinear systems form a well structured class of non-linear systems within which linear systems coexist as a special subclass. They are, therefore, appropriate for modelling a wider range of processes and plant than the restrictive, yet convenient, linear model structures since such models are valid both within the linear subregion and beyond. In addition to extending the self-tuning framework for bilinear systems another significant contribution of the Thesis is the introduction of a cautious least squares estimation procedure which also enhances the existing linear self-tuning schemes.

In recognition of the inevitable plant/model mismatch problems that accompany the standard linear self-tuning approach, it is pertinent to consider extending the linear self-tuning framework to accommodate the wider class of bilinear systems. Such an extended framework should alleviate the problems of plant/model mismatch whilst at the same time increasing the range of applicability of self-tuning control. An extended form of the linear pole-placement control strategy is investigated and attention is restricted to the class of single-input single-output and multiple-input single-output bilinear systems, noting that the more general class of multiple-input multiple-output systems can be represented by a series of interconnected multiple-input single-output subsystems.

In the development of an appropriate bilinear self-tuning controller, a number of enhancements to the standard estimation algorithms used for linear self-tuning control have been necessitated; this being due mainly to the increased sensitivity of the bilinear approach . Enhancements include; a hybrid form of the variable forgetting factor to facilitate the tracking of slowly varying model parameters; a two-tier adaptive mechanism

involving variable forgetting factor reset coupled with covariance matrix reset for both rapid and slow parameter variation; and a cautious least squares parameter estimation scheme for increased robustness.

The bilinear self-tuning controller and its successive variants are assessed using both simulation studies and real-time laboratory based trials. It is shown that when the bilinear self-tuner is applied to systems exhibiting bilinear characteristics that significant improvements in performance are possible over the use of standard linear schemes incorporating enhanced parameter estimation procedures. Finally, since the resulting self-tuning controller is potentially applicable for a wider range of applications than the linear self-tuning scheme, it is pertinent to consider, as one does for the linear case, the applicability of the bilinear self-tuner to other forms of non-linear systems for which local bilinearity may be assumed.

Acknowledgements

I wish to formally acknowledge the following persons:

My colleague and mentor Professor D. J. G. James for his unbounded enthusiasm and encouragement, Dr. D. N. Shields for his patience, particularly in the early years, and to my former student, colleague and good friend Mr. S. G. Goodhart to whom I extend a special thanks for his diligent support and invaluable assistance in preparing this Thesis.

K.J. Burnham

May 1991

Contents

1. Introduction and outline of approach

Introduction

A self-tuning controller (STC), of whatever generic type, is required to perform the dual role of on-line parameter estimation followed by control law implementation and it is this sequential rationale that distinctively characterises STC from any other form of adaptive control. In principle once an appropriate model structure has been identified, any analytical control law design procedure can be combined with any recursive parameter estimation procedure. The choice of appropriate algorithms is, therefore, wide and will be influenced, to some extent, by the particular application under consideration. In adopting the STC rationale it is assumed that variations in system behaviour, including effects due to inherent plant non-linearities, may be adequately absorbed within the time varying nature of the parameters of an adopted linear model structure.

At first, not surprisingly, industry was slow to respond to the new technology, however, STC is now rapidly maturing to provide a realistic option for the control of an increasingly widening range of industrial applications; this being only recently made possible by the rapid parallel developments in microcomputer technology. Significant interest has already been shown by the process industries where large system time constants and assumptions on slowly varying plant dynamics has permitted the widespread use of STC techniques. However, the number of reported successful implementations of STC to systems exhibiting fast dynamics and/or where assumptions on local linearity may be invalid remain rather few and it is in these directions that there is currently much interest and on-going research.

Traditionally, manually tuned three term proportional-integral-derivative

(PID) controllers have found wide usage in industry; being tuned empirically to provide an acceptable engineering trade-off between optimal performance and operational robustness, thus ensuring 'satisfactory' operation over a 'limited' operating range of the plant. An advantage of the conventional PID controller is that it may be effectively tuned without the need for a mathematical model of the plant. This is particularly relevant since all but for the simplest of plant exhibit non-linear characteristics, suggesting that even if a model were to be obtained for the purpose of tuning it would only be an approximation so that there could well be no guaranteed advantages. The disadvantage of the PID controller, however, is that it is required to be re-tuned / de-tuned for each new operating point / range of the plant; a time consuming task which requires the skill of an experienced operator and yet neither guarantees optimality or repeatability.

In an attempt to overcome the latter shortfalls, auto-tuning controllers have been introduced, thereby removing the need for manual tuning and, as such, have provided for an element of repeatability. There are basically three generic types of auto-tuner currently commercially available; one type being based on a mathematical model of the plant is classified as a 'one-shot' STC. The other types make use of pattern recognition techniques in the time domain and limit cycling techniques in the frequency domain. Once tuned, however, the auto-tuning PID controller suffers the same shortfalls as the manually tuned scheme in that, as controller gains are fixed, it is unable to adapt to variations in the plant or, as experienced by the controller, variations in the operating range.

Whilst the auto-tuner has found favour in the process industries, it has not been widely adopted as a replacement for the conventional PID controller for applications which exhibit fast dynamics and/or severe non-linear characteristics. Distinct from the auto-tuner is the STC which continually updates a discrete linear mathematical model of the plant. A key feature of the standard STC approach is that variations in plant behaviour are assumed to be absorbed into the time varying nature of the updated model

parameters. Shortfalls arising in the implementation of standard linear STC techniques have been recognised and, in an attempt to alleviate such shortfalls, various adaptations and enhancements to the existing schemes have been proposed. It is found in practice, however, that when applied to systems for which assumptions on local linearity may be invalid, the performance of even the enhanced linear STC often falls short of that achieved using the conventional PID schemes.

Prompted by the above, STC schemes are required to be developed application specific and, in order to provide for an effective implementation, it is believed to be advantageous to incorporate the knowledge of plant non-linearities into either or both of the estimation and control algorithms. Since STC is implemented digitally, the incorporation of such knowledge is accomplished using time-step quasi-linearisation techniques.

In this work, the standard linear self-tuning framework is extended to accommodate a class of discrete single-input single-output (SISO) and multiple-input single-output (MISO) bilinear model structures, such non-linear structures being representative of a wide range of engineering, biomedical and socioeconomic systems. Bilinear systems are an appealing class of non-linear systems for which the linear control theory may be readily extended and applied. It is shown that, by taking into account the non-linearity, or bilinearity, at the design stage, potential significant improvements in overall system performance are possible. Further, it is believed that the resulting bilinear STC may be applicable to a wider range of systems, with the applicability range of the linear STC being a subset of this. Additionally, the bilinear STC may well be appropriate for other forms of non-linear system for which local bilinearity may be assumed. ■

Outline of approach

The principles of self-tuning control are introduced in Chapter 2 together with a brief historical development of the subject. A derivation of the recursive least squares algorithm is presented and this serves to form the basis of a number of extended parameter estimation schemes which have been developed specifically to overcome the problems encountered when dealing with bilinear systems. Particular emphasis is directed towards the pole-placement control strategy since this forms the basis of the bilinear STC developed in Chapter 3. The noise rejection capabilities of the state-space pole-placement STC is highlighted via comparative simulation studies involving both the state-space and polynomial techniques. The shortfalls arising in the implementation of standard linear STC to non-linear systems is also discussed.

A formal definition of a bilinear system is given in Chapter 3 and examples of non-linear systems which may be more appropriately represented by bilinear model structures are briefly discussed. The concept of a bilinear STC is introduced and, based on the state-space pole-placement approach, a number of interesting alternative STC algorithms are investigated. The approach, which is adopted for all investigative studies in this work, makes use of a time-step quasi-linearisation procedure in which a 'boot-strapping' technique is employed in a tandem state/parameter estimation scheme. Early investigations involving the bilinear STC highlighted the need for a more robust parameter estimation scheme when dealing with such non-linear systems and served to provide the stimulus for much of the work in Chapter 4. Finally, the approach is extended to accommodate a MISO bilinear model structure and preliminary investigations undertaken.

In Chapter 4, a number of enhanced/extended parameter estimation schemes, which have been developed specifically for improving the integrity of the bilinear STC, are presented. These include: a hybrid form of a variable forgetting factor for improved

adaptivity in the presence of slowly varying system parameters; a combined variable forgetting factor and covariance matrix resetting technique for improved adaptivity and alertness in the presence of both slow and sudden parameter variations; a recursive instrumental variables technique for improved accuracy in the presence of coloured noise output disturbance; Kalman filtering and extended Kalman filtering for improved parameter and joint state/parameter tracking ability in the presence of *a priori* engineering knowledge of the plant; and a cautious least squares estimation procedure, applied either sequentially, cyclically or on reset, in an attempt to improve the robustness of the bilinear STC as well as provide for increased numerical stability in the absence of a sufficiently exciting input signal to the plant. Whilst all of these techniques have been developed principally for bilinear systems, the results are equally applicable to linear systems and may be readily extended to other forms of non-linear system in which the non-linearity may be represented in polynomial form.

Real-time trials involving two laboratory based non-linear systems are presented in Chapter 5. In the first application, use is made of a standard linear STC which incorporates the various enhanced parameter estimation techniques developed in Chapter 4. The cautious least squares procedure when used in conjunction with a switched model linearisation scheme is found to be particularly appropriate. The second application, which is known from physical considerations to exhibit bilinear characteristics, forms an ideal test environment for evaluating the effectiveness of the bilinear STC. A bilinear model structure appropriate for the system is derived from first principles and the self-tuning framework is extended to accommodate this structure as outlined in Chapter 3. The effectiveness of the bilinear STC is demonstrated when use is made of extended forms of the linear pole-placement and PID self-tuning schemes. The results are particularly encouraging and indicate that a markedly improved performance is possible when use is made of the bilinear STC.

Conclusions and discussions for further work are presented in Chapter 6.

2. Principles of self-tuning control

2.1 Resume

Self-tuning control was originaly proposed under the title of *self-optimisation* by Kalman in 1958 [1] but it did not receive much attention because of difficulties in implementation due mainly to the state of the art of computer technology at that time. Around the same time an important symposium on self-adaptive flight control systems was held at what is now Wright-Patterson U.S. Air Force base at Dayton, Ohio in 1959. At this symposium many interesting ideas were put forward which were later to form the basis for the early work on *model reference adaptive control* (MRAC). The sequential rationale of identification followed by adaptive control was also proposed and this provided the impetus which ultimately led to the concept of *self-tuning control* (STC). Although Peterka revived the concept in 1970 [2] it was the pioneering paper of Astrom and Wittenmark in 1973 [3], in which the convergence properties were proved, that triggered off intense interest in STC which is still very much in evidence today. In parallel with the rapid developments in computer technology, STC is rapidly maturing to provide a viable control design approach appropriate for a wide range of complex industrial applications. This is reflected in the amount of interest currently being shown by industry and the fact that STC's are now available commercially.

The algorithm proposed by Astrom and Wittenmark was the *minimum variance* STC in which the sole objective of the control strategy was to minimise the variance of the system output. Recognising that the *minimum variance* approach could lead to a widely varying and possibly unrealisable control input, Clarke and Gawthrop in 1975 [4] extended this approach and proposed the *generalised minimum variance* STC in which the objective was to minimise the output variance subject to constraints on the input variance. Unfortunately both approaches are unable to cope with systems which

exhibit variable dead-time and non-minimum phase characteristics; the latter condition being prevalent in sampled data systems. In an attempt to overcome these problems Wellstead and his colleagues in 1979 [5] proposed the *pole-placement* STC in which the objective of the control strategy was to re-locate the system closed-loop poles, defining some pre-specified transient response, whilst the system zeros remain in their open-loop positions. This approach has proved to be particularly attractive to practising engineers, probably due to its close links with the familiar classical control design techniques. The approach also has the advantage in that the desired control objective can be achieved using a more realistic control action. However, all three approaches have their limitations when applied in practice and much of the recent research in STC has been directed towards refining the algorithms and, in many cases, tailoring them to particular applications. One such variant was introduced by Warwick in 1981 [6] when a *state-space approach* to the existing pole-placement STC was proposed. This approach, which provides a significantly smoother control effort due to its inherent filtering action, forms the basic framework of the STC which is proposed in this Thesis for bilinear systems and, as such, is given detailed consideration in this Chapter.

2.2 Problem formulation for self-tuning control

2.2.1 Preliminaries

In principle STC is a conceptually simple and straightforward approach which may be characterised by two coupled sub-algorithms; one for on-line parameter estimation and the other for control law implementation. These sub-algorithms can be linked via a simple route known as *certainty equivalence*, in which the estimated parameters are always assumed to be correct, or via a more complex route using probing or superimposed test signals. These approaches lead to the so-called *non-dual* and *dual*

STC respectively; a *dual* STC being one in which the input serves as both an ideal control and test signal, whereas in the *non-dual* STC the input serves as a control signal only. A further categorisation is made possible by either merging or separating the parameter estimation and control law design procedures; the resulting STC's being termed *implicit* or *explicit* respectively [7]. A schematic representation of an *explicit non-dual certainty equivalence* STC is illustrated in Figure 2.1.

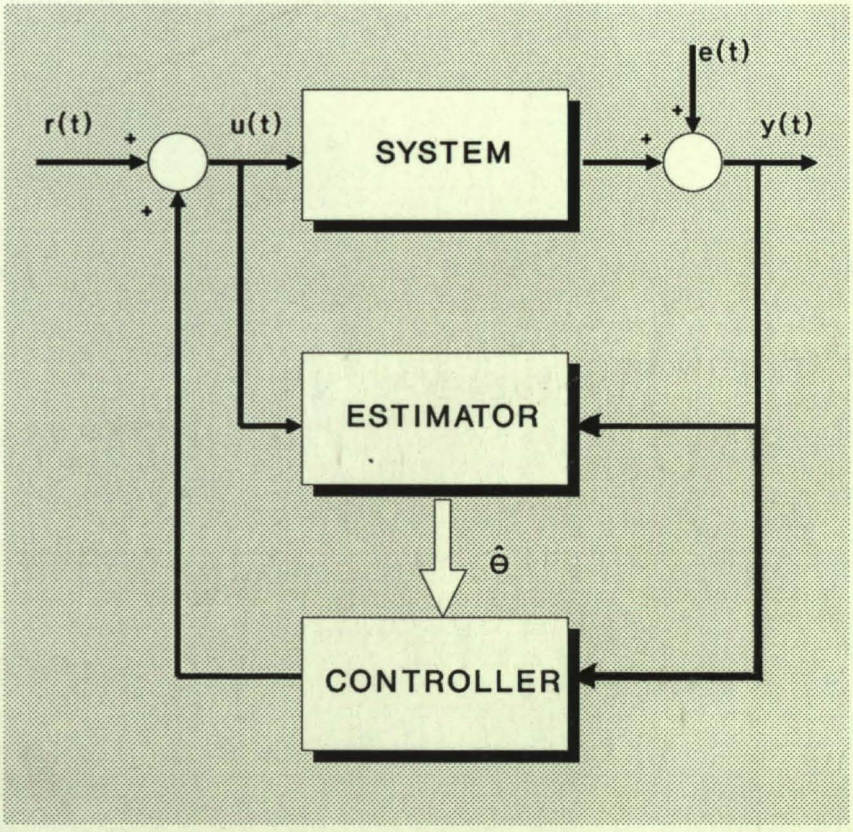


Figure 2.1. Schematic representation of an explicit non-dual certainty equivalence STC.

Adopting the rationale of on-line parameter estimation followed by control implementation, the STC or *self-tuner* is able to adjust or 'self-tune' its controller coefficients, in order to satisfy some particular control law objective, as the system under control itself evolves. The approach assumes that the system to be controlled can be adequately represented by a linear discrete-time stochastic model, commonly a single-input single-output (SISO) difference equation. The parameters of the assumed model are recursively updated using past histories of the input u(t) and output y(t) time series data; this being commonly achieved using a *recursive least squares* (RLS) technique of the form originally proposed by Plackett [8] (See section 2.3). Whilst it is recognised that almost all practical systems are to some extent non-linear it is assumed that by adopting a linear model structure the non-linear characteristics of the system are absorbed into the time varying nature of the model parameters.

A distinctive feature of STC is that any analytical control law design procedure can, in principle, be combined with any on-line recursive parameter estimation scheme; with the particular control law being repeatedly applied in order to achieve some pre-specified system performance criterion as updated parameter estimates are recursively supplied. Many design procedures for STC have been proposed in the literature and an excellent coverage of the relevant material together with exhaustive bibliographies on the subject may be found in recent texts [9,10,11].

2.2.2 **Model formulation**

It is generally assumed that the continuous-time system to be controlled may be modelled by the discrete-time linear SISO difference equation or ARMAX representation

$$A(q^{-1})y(t) = q^{-k}B(q^{-1})u(t) + C(q^{-1})e(t) \qquad (2.1)$$

where the polynomials $A(q^{-1})$, $B(q^{-1})$ and $C(q^{-1})$ are defined as

$$A(q^{-1}) = 1 + a_1q^{-1} + a_2q^{-2} + \ldots + a_{na}q^{-na},$$

$$B(q^{-1}) = b_0 + b_1q^{-1} + b_2q^{-2} + \ldots + b_{nb}q^{-nb},\ b_0 \neq 0,$$

$$C(q^{-1}) = 1 + c_1q^{-1} + c_2q^{-2} + \ldots + c_{nc}q^{-nc},$$

with u(t), y(t) and e(t) being the input, output and white noise sequences respectively, $k \geq 1$ represents the system time delay and is expressed as an integer multiple of the sampling interval, q^{-1} is the backward shift operator defined as $q^{-i}y(t) \equiv y(t-i)$ and the zeros of the noise colouring polynomial $C(q^{-1})$ are assumed to lie inside the unit circle. Representation (2.1) is refered to as the polynomial model structure.

Due to non-linearities and time varying characteristics inherent in all physical systems, some model mismatch will inevitably occur. Consequently the validity of the model and the subsequent performance of the STC is dependent on the ability of the designer to minimise this mismatch. Also, it is normal to adopt a model structure which presents the least computational overhead yet adequately models the system over a particular operating range of interest. The selection of the most appropriate form of equation (2.1) for a particular system is thus an important first step in model formulation and is summarised as follows:

Selection of sampling interval:

The selection of the sampling interval is a key factor, since this will determine the value of the integer k (which must always satisfy $k \geq 1$). The sampling interval is ideally chosen as one tenth of the fastest time constant to be identified. This latter requirement suggests that some *a priori* knowledge of the system is desirable. In the absence of such knowledge, good trial and error procedures must be adopted.

■

Selection of polynomial orders:

The selection of polynomial orders n_a and n_b will also ideally require some physical knowledge of the system. For a practical system $n_a \geq n_b$ and the minimum values of

n_a and n_b are selected which adequately model the system. This latter feature prevents problems arising due to overparameterisation which may lead to numerical deficiencies in the algorithms. Consideration must also be given to ensure that n_b can accommodate the possibility of an increase in system time delay. Finally, it is usual to adopt a white noise model by setting the noise colouring polynomial $C(q^{-1})=1$ in equation (2.1) so that $n_c=0$.

■

There are many alternative state-space representations of the ARMAX model (2.1), each being characterised by the definition of the respective state vectors. The representation adopted in this work is that proposed by Warwick [6]. It is known as the innovations or prediction error model and takes the implicit delay observable canonical form

$$\mathbf{x}(t+1) = \mathbf{P}\mathbf{x}(t) + \mathbf{Q}u(t) + \mathbf{R}e(t) \tag{2.2a}$$

$$y(t) = \mathbf{H}\mathbf{x}(t) + e(t) \tag{2.2b}$$

where $\mathbf{x}(t) \in \mathbb{R}^{n_i}$, $n_i=n_j+k$ where $n_j=\max\{ n_a, n_b, n_c \}$, is the vector of state variables and the matrices **P**, **Q**, **R** and **H** are given initially as

$$\mathbf{P} = \left[\begin{array}{ccccc|c} 0 & & & & & 0 \\ 1 & 0 & & & & \cdot \\ & \cdot & & & & 0 \\ & & \cdot & & & -a_{n_j} \\ & & & \cdot & & \cdot \\ & & & & 1 & -a_1 \end{array}\right] \qquad \mathbf{Q} = \left[\begin{array}{c} b_{n_j} \\ \cdot \\ b_0 \\ 0 \\ \cdot \\ \cdot \\ 0 \end{array}\right] \tag{2.3}$$

$$\mathbf{R} = \left[\begin{array}{c} 0 \\ \cdot \\ 0 \\ (c_{n_j}-a_{n_j}) \\ \cdot \\ \cdot \\ (c_1-a_1) \end{array}\right] \qquad \mathbf{H}^T = \left[\begin{array}{c} 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \\ 1 \end{array}\right]$$

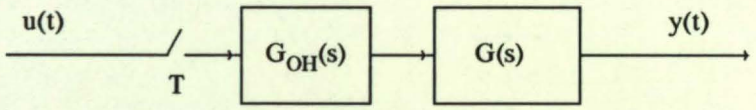
The state-space framework gives rise to a greater understanding and insight

into the intrinsic properties of the system, such as stability, controllability, observability and closed-loop dynamic behaviour. For example, representation (2.2) may be representative of an uncontrollable system when the Kalman controllability test matrix

$$\mathbf{K} = [\, \mathbf{P}^{n_i-1}\mathbf{Q} : \ldots : \mathbf{PQ} : \mathbf{Q} \,]$$

has the property rank{**K**}$<n_i$. This feature illustrates the problem of overparameterisation which may not be so apparent from representation (2.1). Once identified, this apparent problem may be readily overcome by reducing the dimension of the state-space model.

Illustrative example



zero order hold

Figure 2.2

As an illustration, consider the sampled system of Figure 2.2, in which

$$G(s) = \frac{K_1}{s(s+\alpha)} \quad \text{and} \quad G_{OH}(s) = \frac{(1-e^{-sT})}{s}$$

The corresponding z-form transfer function G(z) is given by

$$G(z) = \frac{b_0 z + b_1}{z^2 + a_1 z + a_2}$$

where $b_0 = \dfrac{K_1(\alpha T + e^{-\alpha T} - 1)}{\alpha^2}$, $b_1 = \dfrac{K_1(1 - e^{-\alpha T}(1+\alpha T))}{\alpha^2}$,

$a_1 = -(1 - e^{-\alpha T})$ and $a_2 = e^{-\alpha T}$.

This leads to the ARMAX representation (replacing z^{-1} by q^{-1})

$$A(q^{-1})y(t) = q^{-1}B(q^{-1})u(t) + e(t)$$

with $n_a=2$, $n_b=1$ and $k=1$. From (2.3) the matrices **P** and **Q** are determined as

$$\mathbf{P} = \left[\begin{array}{c|cc} 0 & 0 & 0 \\ \hline 1 & 0 & -a_2 \\ 0 & 1 & -a_1 \end{array}\right] \qquad \mathbf{Q} = \left[\begin{array}{c} b_2 \\ \hline b_1 \\ b_0 \end{array}\right]$$

and because $b_2=0$, it is evident that the pair **PQ** does not satisfy the Kalman controllability criterion. To overcome this apparent problem a reduction in dimension of the state-space model, indicated by the partition lines, is necessary.

■

A sufficient, but not necessary, condition which ensures both controllability and no loss of information in the reduction process is that $n_a \leq n_b+k$ [A1]*. Allowing for possible reduction, the dimension of the resulting state-space representation is denoted as dim{ state-space } = $n \leq n_i$.

2.3 **Parameter estimation**

Having established an appropriate model structure, the next step is to estimate the values of the model parameters. This is commonly achieved using the technique of *recursive least squares* (RLS).

---

* Note that references prefixed by the letter 'A' correspond to the authors publications which are listed separately.

Historical note

Estimation based on *least squares* techniques is today a well established and versatile approach in the field of adaptive control. The method of least squares estimation was proposed in 1795 by Gauss, perhaps the most original and prolific mathematician of his era [12]. Gauss was just 18 years old when he first used the method in the analysis of astronomical observations of planet and comet motions. The motion of such bodies can be completely characterised by six parameters (the Laws of Kepler), and the estimation problem perceived by Gauss was that of infering the values of the parameters from the measured data [13].

It is interesting to note that over 150 years had elapsed before least squares theory advanced significantly beyond the work of Gauss. In 1950 a very important development to least squares theory was made when Plackett [8] extended the results of Gauss to provide a *recursive least squares* (RLS) scheme. This work was perhaps inspired by the unfulfilled promise of Gauss whereby he *"reserved for another occassion the explanation of the devices by which the estimates of parameters can be adjusted with a minimum of fresh calculation due to the appearance of additional observations."* Today, recursive least squares estimation techniques are widely used in a diversity of practical applications and, over the last three decades or so, has provided the stimulus for a wealth of theoretical research.

■

To provide a basis for developing and extending improved estimation algorithms for bilinear systems, the development of the RLS algorithm is breifly reviewed here. It is convenient to follow the route taken by Gauss and Plackett which involves the progression from the least squares algorithm to the recursive least squares algorithm.

2.3.1 Least squares algorithm

Re-arranging equation (2.1)

$$y(t) = \boldsymbol{x}^T(t)\boldsymbol{\theta}(t) + \xi(t) \qquad (2.4)$$

where $\boldsymbol{x}^T(t)$ is the observation vector

$$\boldsymbol{x}^T(t) = [y(t-1)\ y(t-2) \ldots y(t-n_a)\ ;\ u(t-k)\ u(t-1-k) \ldots u(t-n_b-k)\ ]$$

consisting of previous measured values of system input and output; and $\boldsymbol{\theta}(t)$ is the parameter vector

$$\boldsymbol{\theta}^T(t) = [-a_1\ -a_2 \ldots -a_{n_a}\ ;\ b_0\ b_1 \ldots b_{n_b}]$$

with $\xi(t)$ representing a sequence of fitting errors which includes both measurement errors and estimation errors. The standard linear regression approach is then used to estimate the p, $p=n_a+n_b+1$, elements of the parameter vector $\boldsymbol{\theta}(t)$.

$$\begin{aligned} y(1) &= \boldsymbol{x}^T(1)\boldsymbol{\theta} + \xi(1) \\ y(2) &= \boldsymbol{x}^T(2)\boldsymbol{\theta} + \xi(2) \\ &\cdot \quad \cdot \quad \cdot \\ &\cdot \quad \cdot \quad \cdot \\ y(M) &= \boldsymbol{x}^T(M)\boldsymbol{\theta} + \xi(M) \end{aligned}$$

This assumes that M, $M \gg p$, data sets are available. In matrix form this may be expressed as

$$y(M) = \boldsymbol{X}(M)\boldsymbol{\theta} + \xi(M) \qquad (2.5)$$

where

$$y(M) = \begin{bmatrix} y(1) \\ y(2) \\ \cdot \\ \cdot \\ y(M) \end{bmatrix} \qquad \boldsymbol{X}(M) = \begin{bmatrix} \boldsymbol{x}^T(1) \\ \boldsymbol{x}^T(2) \\ \cdot \\ \cdot \\ \boldsymbol{x}^T(M) \end{bmatrix} \qquad \xi(M) = \begin{bmatrix} \xi(1) \\ \xi(2) \\ \cdot \\ \cdot \\ \xi(M) \end{bmatrix}.$$

The least squares problem percieved by Gauss was that of estimating the parameter vector denoted $\hat{\theta}(M)$ subject to minimising the least squares cost function

$$J_M(\theta) = \sum_{i=1}^{M} \xi(i)^2 = \xi^T(M)\xi(M) \tag{2.6}$$

After M observations it follows from (2.5) that

$$J_M(\theta) = (y(M) - X(M)\theta(M))^T (y(M) - X(M)\theta(M)) \tag{2.7}$$

In order to minimise the cost function $J_M(\theta)$ of (2.6) it is required that the p partial derivatives of $J_M(\theta)$, with respect to the elements of $\theta(M)$, should be simultaneously zero.

$$\text{i.e.} \quad \left.\frac{\partial J_M(\theta)}{\partial \theta}\right|_{\theta = \hat{\theta}} = 0 \tag{2.8}$$

Expanding equation (2.7) gives

$$J_M(\theta) = y^T(M)y(M) - \theta^T(M)X^T(M)y(M) - y^T(M)X(M)\theta(M) + \theta^T(M)X^T(M)X(M)\theta(M)$$

Using the vector differentiation [14] given in Appendix 1 leads to

$$\left.\frac{\partial J_M(\theta)}{\partial \theta}\right| = -X^T(M)y(M) - X^T(M)y(M) + (X^T(M)X(M))^T\theta(M) + (X^T(M)X(M))\theta(M)$$

and since $X^T(M)X(M)$ is symmetric it follows that

$$X^T(M)X(M) = (X^T(M)X(M))^T,$$

giving

$$\left.\frac{\partial J_M(\theta)}{\partial \theta}\right|_{\theta = \hat{\theta}} = -2X^T(M)y(M) + 2X^T(M)X(M)\hat{\theta}(M),$$

so that

$$X^T(M)y(M) = X^T(M)X(M)\hat{\theta}(M) \tag{2.9}$$

which gives

$$\hat{\theta}(M) = [\, X^T(M)X(M)]^{-1}\, X^T(M)y(M) \qquad (2.10)$$

which is the optimal estimate in the sense of least squares based on M observed data sets. The matrix $[X^T(M)X(M)]$ is the information matrix and its inverse is known as the error covariance matrix denoted $\Phi(M)$. Making this substitution (2.10) becomes

$$\hat{\theta}(M) = \Phi(M)X^T(M)y(M) \qquad (2.11)$$

Remarks

i) The resulting least squares estimator minimises the residual error $\xi(M)=y(M)-X^T(M)\theta(M)$, and does not automatically minimise the error $\theta(M)-\hat{\theta}(M)$.

ii) The least squares approach given in equation (2.11) is only suitable for 'batch' processing off-line. In its present form, $\hat{\theta}(M)$ is estimated from M observations and there is no means by which an extra observation can be used to update the estimate $\hat{\theta}(M+1)$ without repeating the whole procedure.

iii) It is worth noting that the least squares cost criterion is by no means absolute; with other cost functions being equally appropriate such as

$$J_M(\theta)= \sum_{i=1}^{M} \left| e(i) \right|$$

or $$J_M(\theta)= \max \left| e(i) \right|, \quad 1 \leq i \leq M$$

or any other meaningful cost function. The least squares technique is algebraically tractable and, as such, has found wide acceptance.

■

2.3.2 **Recursive least squares algorithm**

The proposed extension by Plackett [8] was to make equation (2.11) recursive; this being achieved by expressing the estimate $\hat{\theta}(M+1)$, obtained after (M+1) samples, in terms of the estimate $\hat{\theta}(M)$ obtained after M samples, plus a corrective action based on the new information available at sample (M+1). In qualitative terms

$\hat{\theta}(M+1) = \hat{\theta}(M) +$ corrective action based on y(M+1) and $\boldsymbol{x}(M+1)$.

This extension reflected the increasing interest in the use of the least squares technique at the time. Whereas it had previously been used for retrospective off-line batch analysis, its potential as an on-line tool for generating updated parameter estimates as new information became available was becoming increasingly apparent.

Extending (2.5) after (M+1) observations an extra row $\boldsymbol{x}^T(M+1)$ appears in $\boldsymbol{X}$ and an extra element y(M+1) appears in $\mathbf{y}$,

$$\text{i.e.} \quad \mathbf{y}(M+1) = \begin{bmatrix} y(1) \\ y(2) \\ \cdot \\ \cdot \\ y(M) \\ \hline y(M+1) \end{bmatrix} \qquad \boldsymbol{X}(M+1) = \begin{bmatrix} \boldsymbol{x}^T(1) \\ \boldsymbol{x}^T(2) \\ \cdot \\ \cdot \\ \boldsymbol{x}^T(M) \\ \hline \boldsymbol{x}^T(M+1) \end{bmatrix} \tag{2.12}$$

which may be expressed in the partitioned form

$$\begin{bmatrix} \mathbf{y}(M) \\ \hline y(M+1) \end{bmatrix} = \begin{bmatrix} \boldsymbol{X}(M) \\ \hline \boldsymbol{x}^T(M+1) \end{bmatrix} \boldsymbol{\theta}(M+1) + \begin{bmatrix} \boldsymbol{\xi}(M) \\ \hline \xi(M+1) \end{bmatrix} \tag{2.13}$$

It follows from (2.11) that after (M+1) observations

$$\hat{\boldsymbol{\theta}}(M+1) = \boldsymbol{\Phi}(M+1)\boldsymbol{X}^T(M+1)\mathbf{y}(M+1) \tag{2.14}$$

where

$$\boldsymbol{\Phi}(M+1) = [\, \boldsymbol{X}^T(M+1)\boldsymbol{X}(M+1) \,]^{-1} \tag{2.15}$$

Re-writing (2.15) in partitioned form leads to

$$\Phi(M+1) = \left[\begin{bmatrix} X(M) \\ \hline x^T(M+1) \end{bmatrix}^T \begin{bmatrix} X(M) \\ \hline x^T(M+1) \end{bmatrix} \right]^{-1} \tag{2.16}$$

which on expanding gives

$$\Phi(M+1) = [\, X^T(M)X(M) + x(M+1)x^T(M+1) \,]^{-1} \tag{2.17}$$

and because $\Phi = [\, X^T X \,]^{-1}$,

$$\Phi(M+1) = [\, \Phi^{-1}(M) + x(M+1)x^T(M+1) \,]^{-1} \tag{2.18}$$

It is apparent that matrix inversion is necessary to obtain the updated covariance matrix. However, by making use of the matrix inversion lemma [15] (see Appendix 2) equation (2.18) simplifies to

$$\Phi(M+1) = \Phi(M) - \frac{\Phi(M)x(M+1)x^T(M+1)\Phi(M)}{1 + x^T(M+1)\Phi(M)x(M+1)} \tag{2.19}$$

Note that the quantity $[1 + x^T(M+1)\Phi(M)x(M+1)]$ is a scalar, so that no matrix inversion actually takes place.

Similarly, re-writing equation (2.14) in the partitioned form leads to

$$\hat{\theta}(M+1) = \Phi(M+1) \left[\, X^T(M) : x(M+1) \,\right] \begin{bmatrix} y(M) \\ \hline y(M+1) \end{bmatrix} \tag{2.20}$$

which on expanding gives

$$\hat{\theta}(M+1) = \Phi(M+1) \,[\, X^T(M)y(M) + x(M+1)y(M+1) \,] \tag{2.21}$$

and from equation (2.9) it follows that

$$X^T(M)y(M) = \Phi^{-1}(M)\hat{\theta}(M) \tag{2.22}$$

Substituting equation (2.22) into equation (2.21) gives

$$\hat{\theta}(M+1) = \Phi(M+1) \,[\, \Phi^{-1}(M)\hat{\theta}(M) + x(M+1)y(M+1) \,] \tag{2.23}$$

and adding the quantity

$$\hat{\theta}(M) - \Phi(M+1)\Phi^{-1}(M+1)\hat{\theta}(M) = 0$$

yields a simplification, such that

$$\hat{\theta}(M+1) = \Phi(M+1)[\Phi^{-1}(M)\hat{\theta}(M) + x(M+1)y(M+1)]$$
$$+ [\hat{\theta}(M) - \Phi(M+1)\Phi^{-1}(M+1)\hat{\theta}(M)]$$

$$= \Phi(M+1)[\Phi^{-1}(M)\hat{\theta}(M) + x(M+1)y(M+1)$$
$$- \Phi^{-1}(M+1)\hat{\theta}(M)] + \hat{\theta}(M)$$
$$= \hat{\theta}(M) + [[\Phi^{-1}(M) - \Phi^{-1}(M+1)]\hat{\theta}(M)$$
$$+ x(M+1)y(M+1)] \qquad (2.24)$$

From equation (2.18)

$$\Phi^{-1}(M) - \Phi^{-1}(M+1) = \Phi^{-1}(M) - [\Phi^{-1}(M) + x(M+1)x^{T}(M+1)]$$
$$= - x(M+1)x^{T}(M+1) \qquad (2.25)$$

Substituting equation (2.25) into equation (2.24) and re-arranging gives

$$\hat{\theta}(M+1) = \hat{\theta}(M) + \Phi(M+1)x(M+1)[y(M+1) - x^{T}(M+1)\hat{\theta}(M)] \qquad (2.26)$$

in which $\Phi(M+1)x(M+1)$ is known as the gain vector, y(M+1) is the current observation of the output and $x^{T}(M+1)\hat{\theta}(M)$ is the least squares prediction of y(M+1); the quantity $[y(M+1)-x^{T}(M+1)\hat{\theta}(M)]$ being known as the estimation prediction error.

The recursive algorithm

On-line use of the recursive scheme makes use of data pairs u(i), y(i), for i=1,2,..M where M is now allowed to increase to infinity with the recursive form making use of new data as it appears on-line. It is convenient to adopt a new notation for the recursive procedure, since only the quantities $\hat{\theta}$ and Φ are retained between iterations. This is achieved by replacing $\hat{\theta}(M+1)$ and $\Phi(M+1)$ by $\hat{\theta}(t)$ and $\Phi(t)$ respectively, leading to the algorithm

$$\Phi(t) = \Phi(t-1) - \frac{\Phi(t-1)x(t)x^{T}(t)\Phi(t-1)}{1+ x^{T}(t)\Phi(t-1)x(t)} \qquad (2.27)$$

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \Phi(t)x(t)[y(t) - x^{T}(t)\hat{\theta}(t-1)] \qquad (2.28)$$

Such a recursive scheme generates the current estimate $\hat{\theta}(t)$ at time t based upon the previous estimate $\hat{\theta}(t-1)$ generated at time t-1, the previous observations $x(t)$ available at time t, the updated covariance matrix $\Phi(t)$ at time t and the current measurement of the system output y(t) taken at time t.

■

Remarks

i) The RLS algorithm is computationally efficient, requiring only matrix multiplication and no matrix inversion. Between samples it is only necessary to store the quantities $\hat{\theta}(t)$ and $\Phi(t)$.

ii) The matrix $\Phi(t)$ is directly proportional to the estimation error covariance matrix

$$\text{cov}\,(\hat{\theta}-\theta) = E\,\{(\hat{\theta}-\theta)(\hat{\theta}-\theta)^T\}$$

Consider the presence of measurement noise e on the output y, then

$$\hat{\theta} = (X^TX)^{-1}X^Ty$$
$$= (X^TX)^{-1}X^T(X\theta+e)$$
$$= \theta + (X^TX)^{-1}X^Te$$

so that the estimation error covariance matrix becomes

$$\text{cov}\,(\hat{\theta}-\theta) = E\,\{(X^TX)^{-1}X^Tee^TX(X^TX)^{-1}\}$$
$$= (X^TX)^{-1}X^T\ E\,\{ee^T\}\ X(X^TX)^{-1}$$

assuming that the elements of X are uncorrelated with the elements of $\mathbf{e}$ (which may not be the case in practice). If it is further assummed that the elements of $\mathbf{e}$ are serially uncorrelated with variance σ^2 (i.e. that the output sequence y(t) is contaminated with white noise of variance σ^2), then

$$\text{cov}\,(\hat{\theta}-\theta) = (X^TX)^{-1}X^T\ \sigma^2\mathbf{I}\ X(X^TX)^{-1}$$
$$= \sigma^2(X^TX)^{-1}$$
$$= \sigma^2\Phi(t) \qquad (2.29)$$

Then $\Phi(t)$ is directly proportional to the estimation error covariance matrix and conveniently provides an indication of the accuracy of the estimated vector $\hat{\theta}$, so

that the algorithm provides its own error analysis. Thus one implication of equation (2.29) is that a little noise can be advantageous in keeping the algorithm alert.

iii) Note that equation (2.28) has an intuitive 'feel' for when the estimate $\hat{\theta}$ is close to its true value, the term $[y(t)-x^T(t)\hat{\theta}(t-1)]$ will be small and hence the corrective action will be small. Also, as the elements of $\Phi(t)$ decrease, indicating increasing accuracy in $\hat{\theta}(t)$, the corrections to $\hat{\theta}(t)$ will tend to decrease.

iv) Initialisation of the algorithm: Perhaps the simplest method is to set $\hat{\theta}(0)$ according to any *a priori* information and then to set $\Phi(0)$ according to the confidence one has in such information (noting the interpretation of $\Phi(t)$ as a covariance matrix). A typical choice in the absence of such *a priori* knowledge is $\hat{\theta}(0)=0$ and $\Phi(t)=\mu I$ where I is the identity matrix and μ is some large positive scalar.

v) The RLS algorithm can be arranged into the slightly different form [15]

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \phi(t)[\, y(t) - x^T(t)\hat{\theta}(t-1)\,] \tag{2.30}$$

where

$$\phi(t)= \Phi(t-1)x(t)[\, 1+ x^T(t)\Phi(t-1)x(t)]^{-1} \tag{2.31}$$

and then

$$\Phi(t)= [\, I- \phi(t)x^T(t)\,]\Phi(t-1). \tag{2.32}$$

In this form ϕ can be considered as a gain vector which acts on the estimation prediction error $(y-x^T\theta)$.

vi) The RLS algorithm considered so far is unable to adequately adapt to variations in plant behaviour since all measured data is given equal weighting in the least squares cost function (2.7). In order to facilitate the tracking of slowly varying parameters, a forgetting factor $\lambda \leq 1$ is introduced into the algorithm. The forgetting factor effectively provides an exponential window length, or fading memory, given

approximately by $M=(1-\lambda)^{-1}$ and ensures that greatest emphasis is given to the most recent observations. The forgetting factor is incorporated into the RLS algorithm via the least squares cost function

$$J = (y - X\theta)\, \Lambda\, (y - X\theta)$$

where Λ is the diagonal matrix

$$\Lambda = \begin{bmatrix} \lambda^M & & & & \\ & \cdot & & & \\ & & \cdot & & \\ & & & \lambda^2 & \\ & & & & \lambda^1 \end{bmatrix}$$

The resulting adaptive least squares scheme is realised by a simple modification to the covariance update equation (2.32), such that

$$\Phi(t) = [\, I - \phi(t)x^T(t)\,]\Phi(t-1) \,/\, \lambda.$$

Factors affecting the choice of forgetting factor are discussed in [A2]. In an attempt to overcome the attendant problems of covariance blow-up, reported in [16], a hybrid form of the variable forgetting factors proposed in [17,18] is developed in Chapter 4 for bilinear systems.

vii) Equations (2.31) and (2.32) may be further converted to a 'true' covariance matrix form (See Kalman filter, Chapter 4). Since the error covariance matrix $\mathrm{cov}(\hat{\theta}-\theta)=\sigma^2\Phi(t)$, where σ^2 is the assumed variance of the noise on y, then

$$\phi(t) = \Phi'(t-1)x(t)[\sigma^2 + x^T(t)\Phi'(t-1)x(t)]^{-1} \qquad (2.33)$$

and

$$\Phi'(t) = [\, I - \phi(t)x^T(t)\,]\Phi'(t-1) \qquad (2.34)$$

where $\Phi'(t)=\mathrm{cov}(\hat{\theta}-\theta)$.

■

2.4 Control law design

As indicated in the introduction, the pole-placement control strategy is adopted here. The corresponding control law procedures for both the polynomial and state-space approaches are briefly outlined.

The self-tuning pole-placement controller makes use of the non-dual certainty equivalence approach in which the parameter estimation and control law implementation stages are considered to be separated (i.e. an explicit STC scheme).

Following the STC rationale, updated parameter estimates generated from within the estimation algorithm are progressed to the control law algorithm where controller parameters are subsequently updated.

2.4.1 Polynomial pole-placement control law

The original self-tuning pole-placement control algorithm proposed in [5] makes use of the control law

$$D(q^{-1})u(t) = G(q^{-1})y(t) \tag{2.35}$$

where the controller polynomials $D(q^{-1})$ and $G(q^{-1})$ are given by

$$D(q^{-1}) = 1 + d_1q^{-1} + d_2q^{-2} + \ldots + d_{nd}q^{-nd},$$

$$G(q^{-1}) = g_0 + g_1q^{-1} + g_2q^{-2} + \ldots + g_{ng}q^{-ng}.$$

The controller coefficients are obtained from the polynomial identity, or Diophantine equation,

$$A(q^{-1})D(q^{-1}) - q^{-k}B(q^{-1})G(q^{-1}) = \Gamma(q^{-1}) \tag{2.36}$$

where $\Gamma(q^{-1})$ is the user specified closed-loop pole polynomial

$$\Gamma(q^{-1}) = 1 + \gamma_1 q^{-1} + \gamma_2 q^{-2} + \ldots + \gamma_{n_\gamma} q^{-n_\gamma}, \quad n_\gamma = n_a, \tag{2.37}$$

the zeros of which are the desired closed-loop poles. In order for a unique solution to exist for the controller coefficients, it is suggested [5,19] that the orders of the controller polynomials $D(q^{-1})$ and $G(q^{-1})$ be constrained such that $n_d = n_b + k - 1$ and $n_g = n_a - 1$. A derivation of the polynomial identity together with a discussion on the orders of the controller polynomials is given in Appendix 3.

Note that (2.36) may be re-formulated in the matrix form

$$\begin{bmatrix} 1 & & 0 & 0 \\ & & \vdots\ (k-1\ \text{zeros}) & \vdots \\ a_1 & 1 & 0 & \\ a_2 & a_1 & -b_0 & \\ \vdots & a_2 & -b_1 & 0 \\ \vdots & \vdots & \vdots & -b_0 \\ a_{n_a} & \vdots & -b_{n_b} & \\ 0 & a_{n_a} & & \\ \vdots & & & \\ 0 & & & -b_{n_b} \end{bmatrix} \begin{bmatrix} d_1 \\ \vdots \\ d_{n_b+k-1} \\ g_0 \\ \vdots \\ g_{n_a-1} \end{bmatrix} = \begin{bmatrix} \gamma_1 - a_1 \\ \vdots \\ \gamma_{n_a} - a_{n_a} \\ 0 \\ \vdots \\ 0 \end{bmatrix} \begin{matrix} \\ \\ \\ \\ \\ \\ \\ \\ \\ \end{matrix} \begin{matrix} \\ \\ \\ \\ \\ \\ \\ \\ n_b+k-1 \\ \text{zeros} \end{matrix}$$

(n_b+k-2 zeros; n_b+k-1; n_a)

which is readily solved at each time step to provide the updated controller coefficients.

Remarks

In order to justify use of a recursive least squares estimation procedure the noise colouring polynomial $C(q^{-1})$ in equation (2.1) must be taken as unity, implying that white noise is assumed. Although this assumption may lead to biased estimates if the noise is non-white, the self-tuning property [5,19] ensures that overall control is still achieved provided the estimated system parameters converge to some steady-state values.

■

2.4.2 State-space pole-placement control law

The state-space pole-placement control strategy makes use of the familiar state variable feedback control law

$$u(t) = \mathbf{F}\hat{\mathbf{x}}(t) \tag{2.38}$$

where $\hat{\mathbf{x}}(t)$ is the estimated state vector obtained from a steady-state Kalman filter [20] and $\mathbf{F} = [\, f_1, f_2, \ldots f_n \,]$ is a feedback vector chosen such that the resulting closed-loop system matrix [P+QF], obtained from substituting (2.38) in to the state equation (2.2a), has the desired eigenvalues, or equivalently

$$\det[\, \mathbf{I} - q^{-1}[\mathbf{P}+\mathbf{QF}] \,] = 1 + \sum_{i=1}^{n_\gamma} \gamma_i q^{-i} = \Gamma(q^{-1}),$$

where $\Gamma(q^{-1})$ is the desired closed-loop pole polynomial given in (2.37).

Steady-state Kalman filter

Subsitituting the output equation (2.2b) into the state equation (2.2a) and eliminating the noise term e(t) leads to

$$\mathbf{x}(t+1) = [\mathbf{P} - \mathbf{RH}]\mathbf{x}(t) + \mathbf{Q}u(t) + \mathbf{R}y(t) \tag{2.39}$$

which is effectively equivalent to an identity observer.

Re-arranging (2.39), replacing x(t) by its estimate $\hat{\mathbf{x}}(t)$, substituting $\mathbf{P}_1=[\mathbf{P}-\mathbf{RH}]$ and making use of the backward shift operator q^{-1} leads to

$$\hat{\mathbf{x}}(t) = [\, \mathbf{I} - q^{-1}\mathbf{P}_1]^{-1}[\, \mathbf{Q}u(t-1) + \mathbf{R}y(t-1) \,] \tag{2.40}$$

which is known as the steady-state Kalman filter (SKF) used previously by Warwick [6,19] and Caines [20].

It is interesting to note that the implied matrix inversion in (2.40) is really only a transparent problem with $[\mathbf{I}-q^{-1}\mathbf{P}_1]^{-1}$ being readily generalised as a lower

triangular matrix consisting of backward shift operators. It takes the general form

$$[\mathbf{I} - q^{-1}\mathbf{P}_1]^{-1} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ q^{-1} & 1 & & \\ q^{-2} & q^{-1} & 1 & \\ \vdots & & \ddots & \vdots \\ q^{-(n-1)} & & q^{-1} & 1 \end{bmatrix} \qquad (2.41)$$

The resulting state estimator (2.40) is optimal in the sense that if **P**, **Q** and **R** are known and the output disturbance is white then the error dynamics, determined by the eigenvalues of $\mathbf{P}_1$=[**P**—**RH**], ensure that the estimated state vector converges to the true state vector in n discrete steps. (In the case of coloured noise, the rate of convergence will depend on the locations of the zeros of $C(q^{-1})$ which are assumed to lie inside the unit circle.) However, within the self-tuning framework the elements of **P**, **Q** and **R** are themselves recursively estimated and, as a consequence, convergence of the state estimator is dependent upon convergence of the parameter estimator. This does not present an unreasonable situation since variations in system parameters are normally assumed to be much slower than variations in system states.

■

Dyadic form of feedback vector F

There are many methods for determining the feedback vector **F** and a good summary of these may be found in [21]. The approach adopted here makes use of the dyadic form [22] and differs from that originally proposed in [6]. The dyadic approach, which is based on functional relationships between the open and closed loop systems [23,24], provides a straightforward method for obtaining the feedback **F** and a detailed derivation is given in Appendix 4.

Essentially, the feedback vector is obtained from

$$\mathbf{F}^T = \mathbf{W}^{-1}\mathbf{S} \tag{2.42}$$

where $\mathbf{S}^T = [\,0 \ldots 0 \quad (\gamma_{n_a} - a_{n_a}) \ldots (\gamma_1 - a_1)\,]$

and $\mathbf{W}$ is the symmetric matrix given by

$$\mathbf{W} = \mathbf{K}\,\mathbf{L}$$

in which $\mathbf{K}$ is the Kalman controllability test matrix

$$\mathbf{K} = [\,\mathbf{P}^{n-1}\mathbf{Q} : \cdots : \mathbf{PQ} : \mathbf{Q}\,]$$

and $\mathbf{L}$ the lower triangular matrix

$$\mathbf{L} = -\begin{bmatrix} 1 & 0 & \cdot & \cdot & \cdot & \cdot & \cdot & 0 \\ a_1 & & \cdot & & & & & \\ \cdot & \cdot & & & & & & \\ \cdot & & \cdot & & \cdot & & & \\ \cdot & & & & & & & 0 \\ & & & & \cdot & & \cdot & \\ a_{n-1} & & & & & & a_1 & 1 \end{bmatrix}.$$

It is interesting to note that for the feedback vector $\mathbf{F}$ to exist, the symmetric matrix $\mathbf{W}$ must be non-singular. Being lower triangular, $\mathbf{L}$ will always satisfy this so the only condition for the existence of $\mathbf{F}$ is that $\mathbf{K}$ be of full rank (i.e. the Kalman controllability test must be satisfied). This feature highlights the need for model reduction in order to avoid potential problems of overparameterisation. The symmetry of the matrix $\mathbf{W}$ and its representation in terms of the Kalman controllability test matrix are considered in Appendices 5 and 6 respectively.

■

2.5 Analysis of the polynomial and state-space approaches

A simple analysis of the polynomial and state-space control algorithms presented here reveals that whilst the two control laws are theoretically identical, having equivalent input/output maps, their implementation in the presence of output or measurement noise gives rise to distinctively different yet equable control action. The results of this section serve to substantiate the findings reported in [6,A1,A3]. The results suggest that the state-space approach, with its inherent filtering action, should produce the more 'smooth' performance.

2.5.1 Analysis of the control algorithms

Whilst the following results may be readily generalised, for illustrative purposes consideration is given here to the system

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

in which $n_a=2$, $n_b=1$, $n_c=0$ and $k=1$, in order to highlight the fundamental difference between the two approaches.

The corresponding polynomial control law given by equation (2.35) may be expressed as

$$(1+d_1q^{-1})u(t) = (g_0+g_1q^{-1})y(t)$$

which on rearranging leads to

$$u(t) = g_0y(t)+g_1y(t-1)-d_1u(t-1). \qquad (2.43)$$

Similarly, the corresponding control action generated by the state variable feedback control law of equation (2.38) may be expressed as

$$u(t) = f_1\hat{x}_1(t) + f_2\hat{x}_2(t). \qquad (2.44)$$

In order to demonstrate the equivalence of the two approaches in the absence of noise,

consider the direct expansion of the state estimates obtained from the SKF, equation (2.40),

$$\hat{x}_1(t) = -a_2y(t-1) + b_1u(t-1)$$

$$\hat{x}_2(t) = -a_1y(t-1) - a_2y(t-2) + b_0u(t-1) + b_1u(t-2).$$

Note that due to the form of **H** in the state-space formulation (2.3) it follows that the n^{th} state variable $\hat{x}_n(t)$, in this case $\hat{x}_2(t)$, corresponds to the noise free system output (y(t)—e(t)). Thus in the absence of noise (i.e. e(t)=0) the state estimates may be expressed as

$$\hat{x}_1(t) = -a_2y(t-1) + b_1u(t-1) \qquad (2.45a)$$

$$\hat{x}_2(t) = y(t). \qquad (2.45b)$$

Substituting equations (2.45) into equation (2.44) leads to

$$u(t) = f_1[-a_1y(t-1) + b_1u(t-1)] + f_2y(t).$$

Re-arranging

$$u(t) = f_2y(t) - f_1a_1y(t-1) + f_1b_1u(t-1). \qquad (2.46)$$

By comparing coefficients on the right hand sides of equations (2.43) and (2.46) it is clear that there is a direct relationship, such that

$$f_2 \equiv g_0, \qquad f_1a_1 \equiv -g_1 \quad \text{and} \quad f_1b_1 \equiv -d_1 \qquad (2.47)$$

which may be verified by direct expansion of the respective controller coefficient equations (2.36) and (2.42). Hence in the absence of output noise the state-space and polynomial control algorithms are equivalent.

The fundamental difference between the two control algorithms is that the polynomial approach makes direct use of the current output whereas the state-space approach, which incorporates the SKF equation (2.40), makes use of a noise free prediction of the current system output. Consequently in the presence of output noise the control algorithms differ with $\hat{x}_n(t) \neq y(t)$ and the resulting control actions give rise to differing input/output sequences. It is noted that with differing input/output sequences, due to the presence of noise, the convergence behaviour of the estimated model parameters could also differ so that the relationships (2.47) may no longer be strictly

correct.

Remarks

i) A simple analysis of the polynomial and state space model structures has revealed that whilst they are commonly regarded as being equivalent, in practice (in the presence of output noise) it is found that they may well give rise to distinctively different controller action.

ii) The above observation stimulated the search for a state space structure which is truly identical to the polynomial approach in terms of its practical implementation as well as from theoretical considerations. It is interesting to find that this may be achieved via a simple re-configuration of the system state vector. The resulting structure differs from the normal innovations approach, which essentially gives rise to a prediction error model of the form (2.3), in that a 'filtered' state-space model is produced [A4].

■

2.6 Simulation studies

This section highlights typical results of simulation studies undertaken in [A1,A3] involving both the polynomial and state-space pole-placement techniques. The results reinforce the findings of section 2.5 and generally support the claim in [6] that the state-space approach produces the more favourable response in terms of a 'lighter' control action.

In [A1] the basic form of the system under investigation is that previously adopted by Astrom and Eykoff [25], in which

$$(1 - 1.5q^{-1} + 0.7q^{-2})y(t) = q^{-1}(1 + 0.5q^{-1})u(t) + e(t)$$

and the objective is to regulate the output y(t) about a level given by $B(1)/\Gamma(1)$ with r(t)=1.0 and closed-loop poles effectively specified at -3 and -10 in the s-plane when the sampling interval T is taken as 0.1 second. Each test is taken over 100 iterations.

The system parameters are varied from their nominal values in order to produce a number of interesting test conditions. The test conditions investigated in [A1] include

(i) various levels of white output noise (fixed parameters),

(ii) various levels of coloured output noise (fixed parameters),

(iii) non-minimum phase system (parameter b_1 differs from its nominal value of 0.5 to become 1.1),

(iv) open-loop unstable system (parameter a_1 differs from its nominal value of -1.5 to become -2.1),

Results of test (i) are presented in Figures 2.3 and 2.4 which correspond to the use of the polynomial and state-space approach respectively. In each case the upper trace shows the system response y(t) and the lower trace the control actuation signal u(t). It is readily observed that the variance of the control signal is significantly reduced when use is made of the state-space approach. This feature is highlighted in Figures 2.5 and 2.6 which

illustrate the same control actuation signals when increased by a factor of five. It is found [A1] that with the exception of the open-loop unstable system, test (iv), that in terms of reduced control input variance, the state-space approach generally produces the more favourable results.

In [A3] the system under investigation is that of an industrial hydraulic test rig [26] which has been developed specifically for the purpose of evaluating alternative STC techniques. The model adopted for simulation studies takes the form

$$(1 - 0.48927q^{-1} + 0.36327q^{-2})y(t) = 4.78426u(t\text{-}1);$$

this model being identified on-line for a given region of operation. A full description of the test rig including a discussion on the inherent non-linearities is given in [27]. In [A3], the performance of the existing polynomial STC is compared to the state-space STC when the system model is subjected to various levels of output noise disturbance. The results presented here again serve to highlight the difference between the two approaches in the presence of noise. Figures 2.7 and 2.8 correspond respectively to use of the polynomial and state-space STC when the feed flow is varied in the absence of noise, indicating little or no difference in performance. However, it is clear from the results given in Figures 2.9 and 2.10, which also correspond to the polynomial and state-space STC, that in the presence of noise use of the state-space approach, with its inherent noise rejection cabapilities, leads to a markedly superior performance.

Remarks

The state-space innovations model, or so called prediction error model, is found to provide the more favourable results in terms of its noise rejection capabilities and, as such, forms the framework for the bilinear self-tuning controller which is developed in Chapter 3.

■

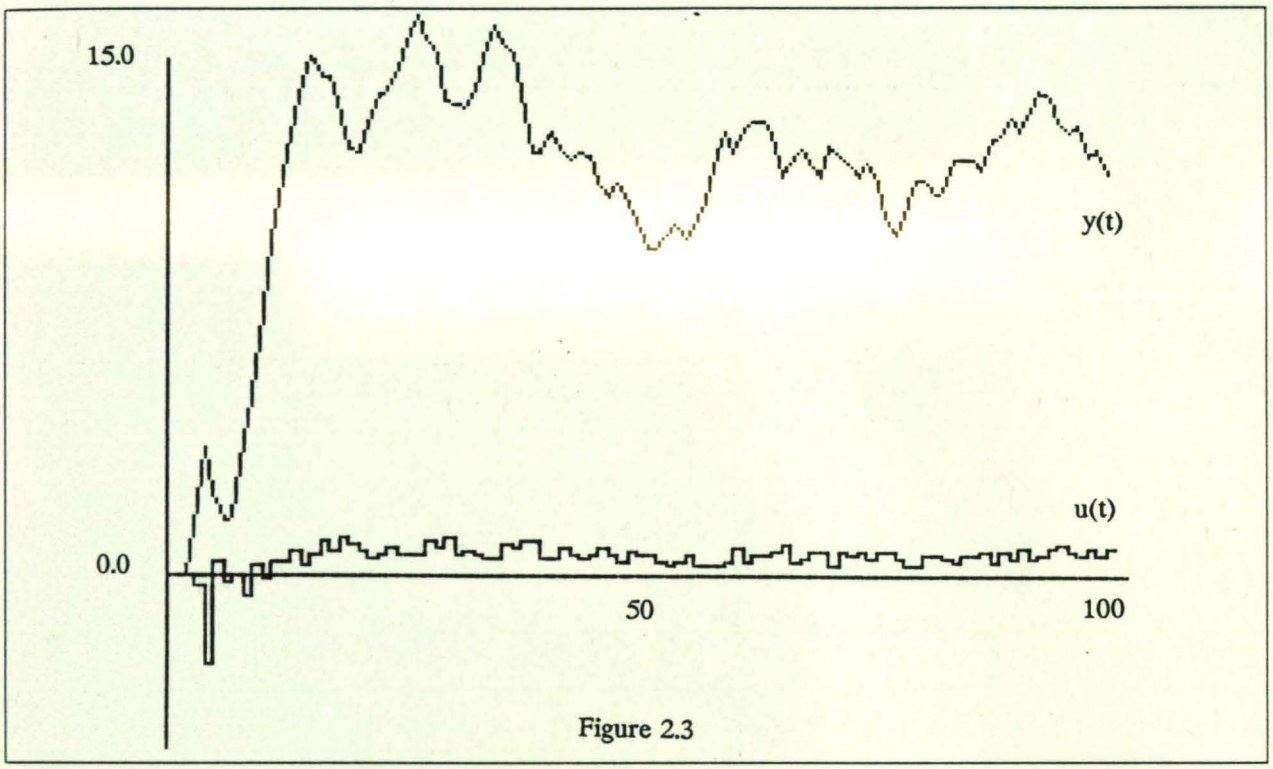


Figure 2.3

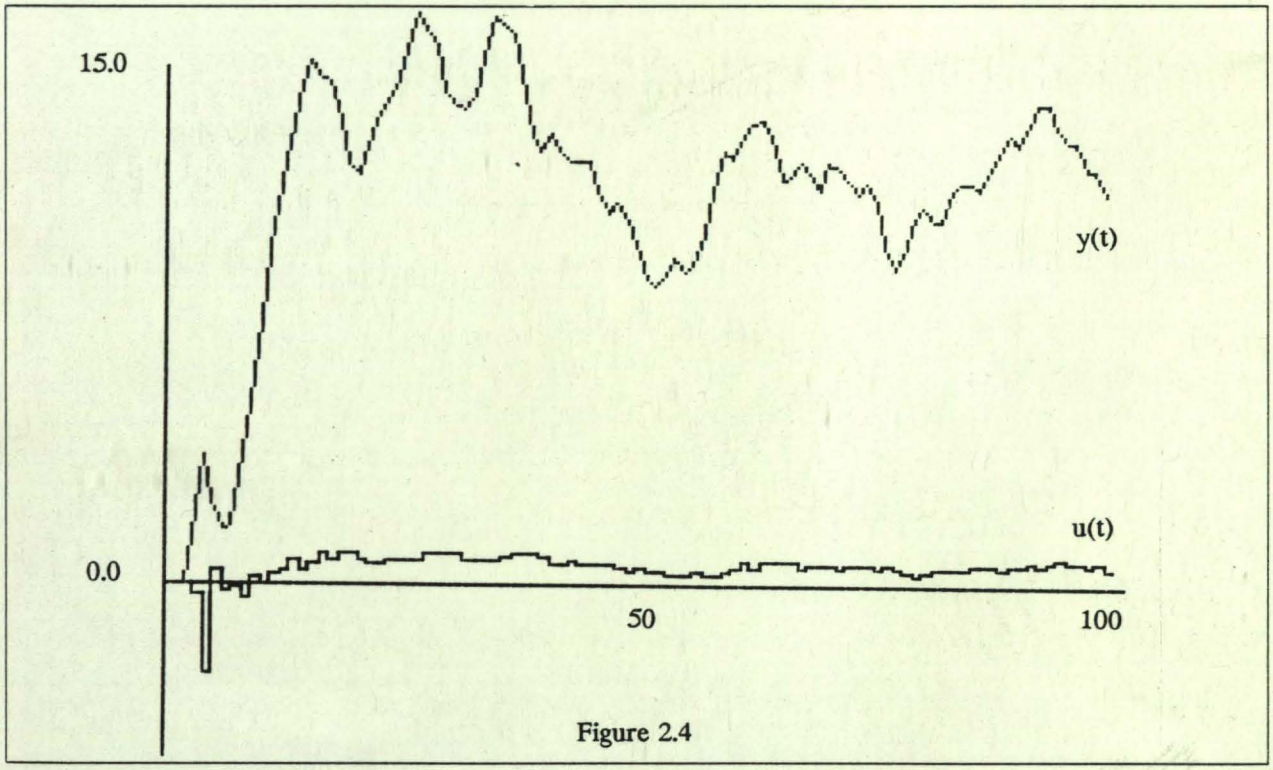


Figure 2.4

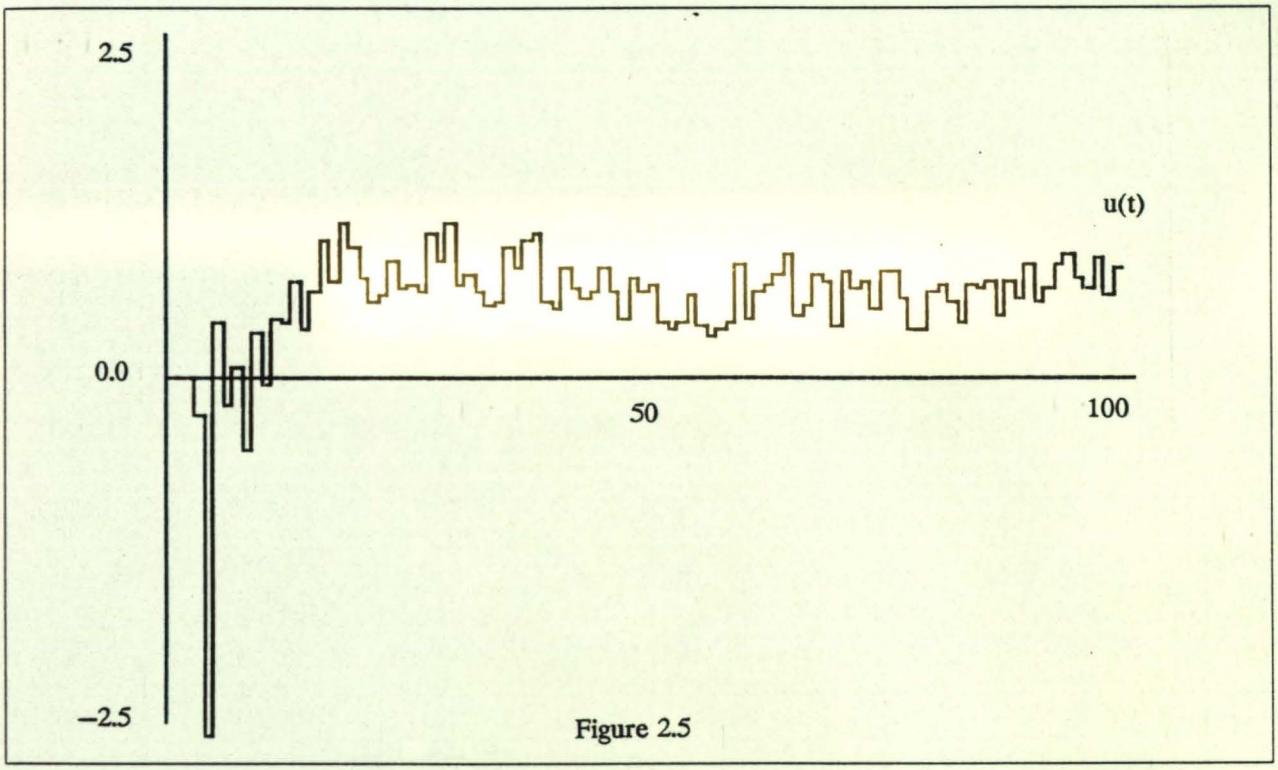


Figure 2.5

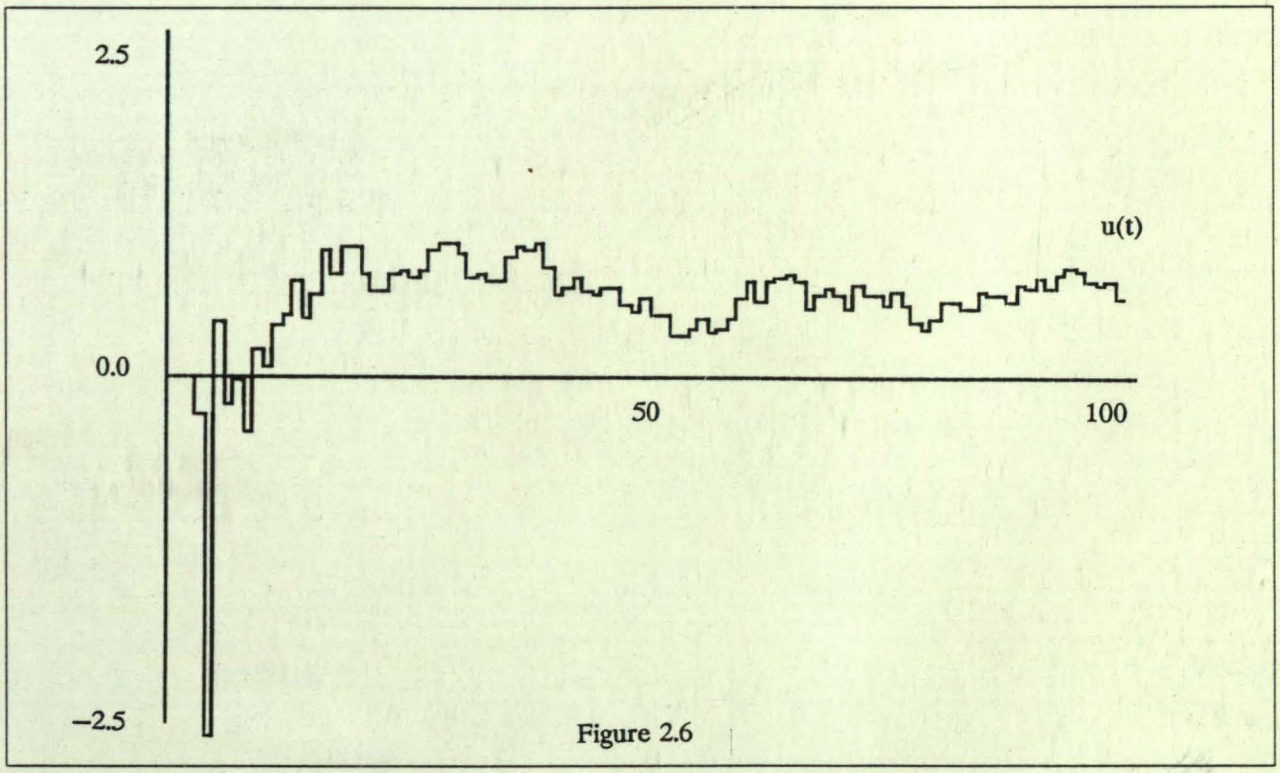


Figure 2.6

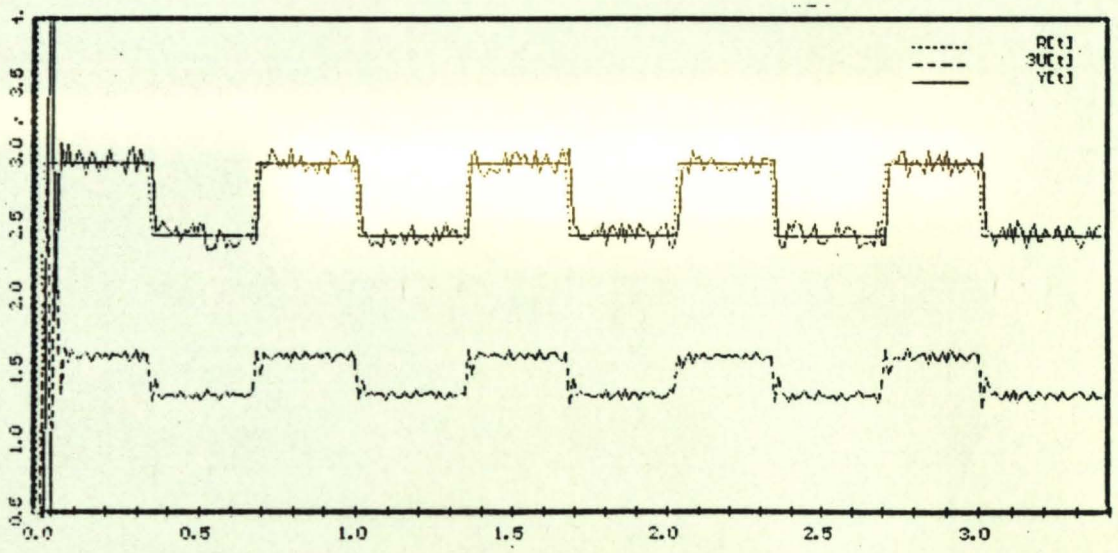


Figure 2.7

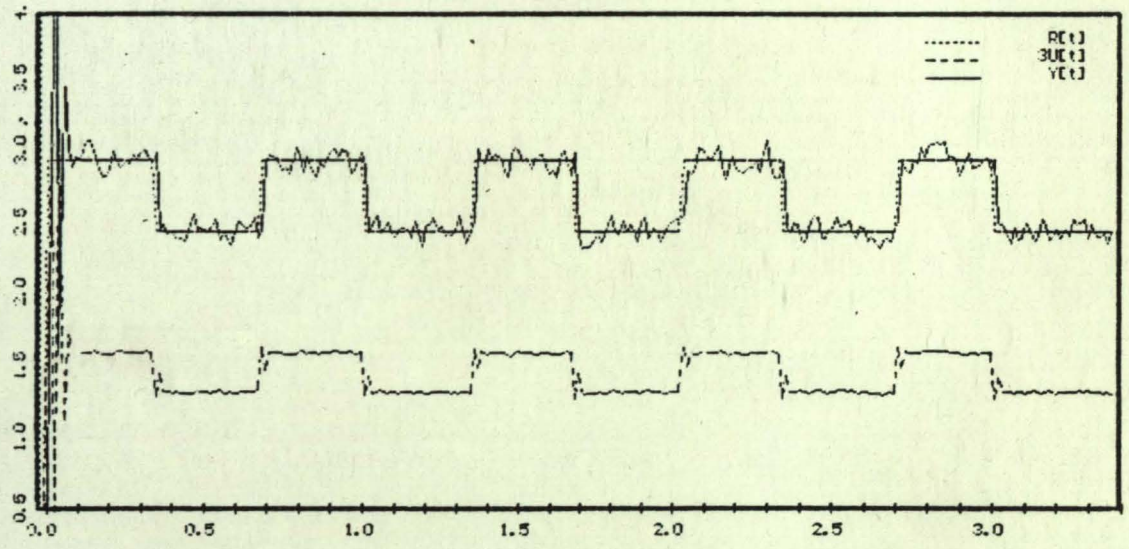


Figure 2.8

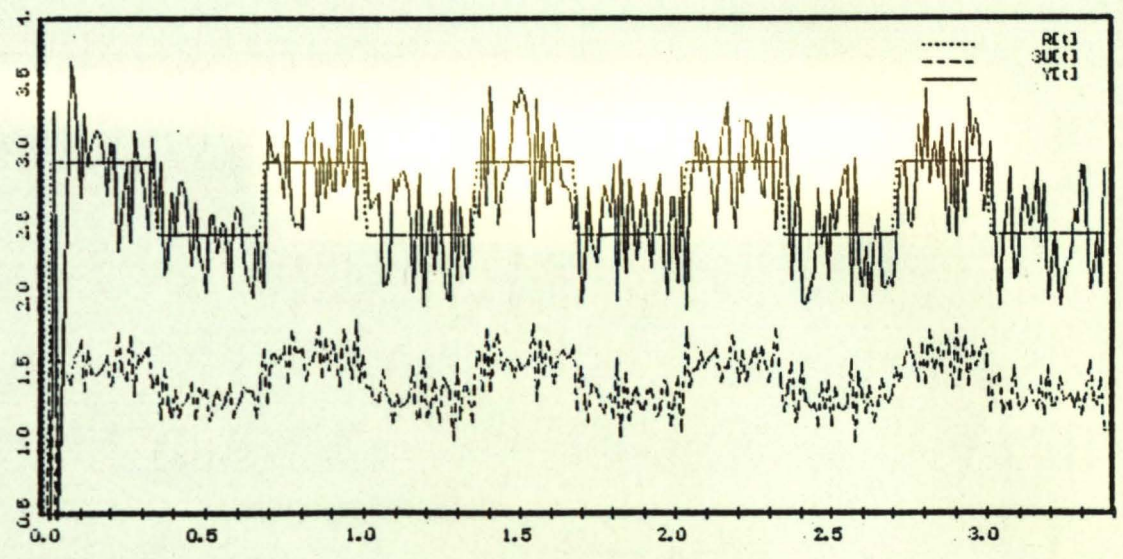


Figure 2.9

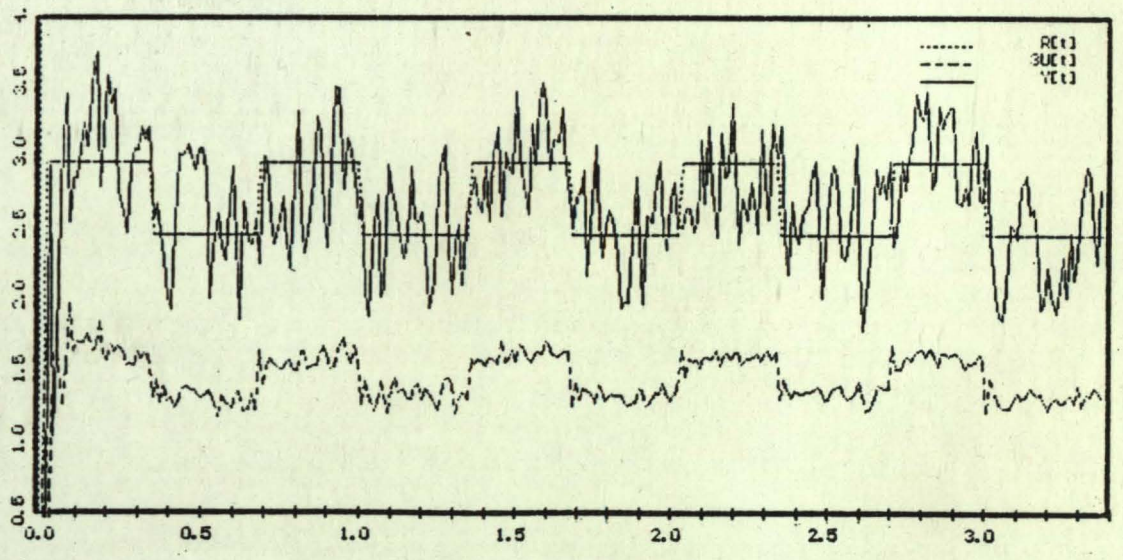


Figure 2.10

2.7 Concluding remarks

Whilst there are significant potential advantages to be gained by adopting STC techniques, it must be noted that the number of reported successful implementations, on all but the simplest of plant, remain few.

Historically, the main shortfall has been two fold: Firstly, early implementations of STC were, by necessity, based on restrictive low order model structures and subsequent controller performance was often, not surprisingly, found to be inadequate; secondly, attempts to capture complex plant dynamics, including effects due to non-linearities, necessitates high order linear models, thus leading to excessively large computational overheads and the possibility of numerical instability due to overparameterisation.

Due to the recent advances in microcomputer technology the above arguments no longer hold and STC potentially offers a realistic option for the control of an increasingly widening range of applications. It is believed, however, that if the advantages of the rapidly developing technology are to be fully realised then the self-tuning framework should be extended to accommodate for known/identified plant non-linearities at the design stage. Thus significantly reducing the major problem of plant/model mismatch which is unavoidable when adopting linear STC.

In an attempt to increase the applicability range of STC, the self-tuning framework is extended in Chapter 3 to accommodate a class of discrete-time bilinear systems; such systems encompassing a wide range of practical applications.

3. Self-tuning control for bilinear systems

3.1 Bilinear systems

Self-tuning control has generated significant interest in the process industries where large system time constants and assumptions on slowly varying model parameters permits the widespread use of such adaptive schemes. However, the reported successful implementations of STC to systems which exhibit fast dynamics and/or where assumptions on local linearity may be invalid remain rather few and it is in these directions that there is currently much interest and on-going research.

Prompted by the shortfalls arising in the implementation of standard STC schemes to non-linear plant, the need for an alternative approach, in which plant non-linearities are taken into account at the design stage, has been identified. In this Chapter, the standard linear STC framework is extended to accommodate a well structured class of non-linear systems for which extensions of the linear control theory may be readily developed and applied; such systems being defined originally by Mohler [28] are known as *bilinear systems.*

Bilinear systems are defined to be linear in terms of both state and control when considered independently, with the non-linearity or bilinearity arising from coupled terms [28]. Such systems form an important class of 'near linear' systems and are representative of a wide range of biological, economical and engineering applications [29-35]. These include fermentation processes, effects due to cancer drugs, population growth, distillation columns, disc braking systems, AC/DC motors and heating/cooling processes. When considering such applications the use of a bilinear model structure can often provide a greater insight into the underlying physical phenomenon, leading to a better understanding of system behaviour than the convenient, yet often inadequate,

linear model structures.

Whilst identification of bilinear systems has received much attention in the literature [36-41] it is found that in general, the resulting algorithms are not readily implementable and are thus inappropriate for use within a STC. Although aspects of stability, controllability and optimal control of bilinear systems have also received much attention, often leading to complex strategies [42-49], the integration of both identification and control strategies in the design of STC appears to be limited to the work of the author [A5-A13].

In references [A5-A13] self-tuning principles have been developed for a class of SISO discrete time bilinear systems and extended forms of the linear pole-placement algorithm have been investigated. Essentially, due to its noise rejection capabilities, the state-space innovations framework has been adopted and, by making use of quasi-linear state variable feedback, the objective of this extended STC scheme is to relocate the closed-loop poles of the linear part of the overall bilinear system. In developing the bilinear STC algorithms a number of approaches for obtaining the feedback vector and the estimated state vector have been considered [A5] and these are described in section 3.3. The resulting algorithms are assessed in terms of their ability to achieve the control objective whilst at the same time producing an acceptable level of variance in both control input and system response. The approach adopted on this basis is further developed to include an alternative structure for the observation vector [A6] which is used in an extended form of the basic RLS algorithm. The resulting scheme, which is described in [A7], makes use of a 'boot-strapping' technique in which the parameters and states are estimated in tandem. In [A8,A9] an enhanced estimation scheme involving a combined variable forgetting factor and covariance matrix resetting procedure is investigated. The resulting two-tier adaptive mechanism is able to handle both slow and sudden parameter variations.

A significant step in the development of the bilinear STC has been the introduction of a cautious supervisory procedure [A10]. The resulting cautious approach differs from that discussed in [9,16] in that caution is applied within the estimation algorithm rather than at the control implementation stage. In this way the designer has the ability to influence the STC through a practical knowledge of the system. A summary of the developments in the design of the bilinear STC is given in [A11,A12]. Finally in [A13], the STC is extended to accommodate the class of multiple-input single-output (MISO) bilinear systems.

In this Chapter attention is focused towards the development of the bilinear STC framework, with the enhanced estimation techniques being considered separately in Chapter 4.

3.2 Problem formulation

Consider the SISO state-space bilinear system SBL(n,m)

$$\mathbf{x}(t+1) = \mathbf{P}\mathbf{x}(t) + \mathbf{Q}u(t) + \mathbf{R}e(t) + \sum_{i=1}^{m} u(t-i+1)\mathbf{N}_i\mathbf{x}(t) \tag{3.1a}$$

$$y(t) = \mathbf{H}\mathbf{x}(t) + e(t) \tag{3.1b}$$

where $\mathbf{x}(t)\epsilon\mathbb{R}^{n_i}$, $n_i=n_j+k$ with $n_j=\max\{n_a,n_b,n_c\}$, is the vector of state variables and $u(t)$, $y(t)$ and $e(t)\epsilon\mathbb{R}$ are the input, output and white noise sequences respectively, $k\geq 1$ is the system time delay expressed as an integer multiple of the sampling interval and n_a, n_b and n_c are the orders of the polynomials $A(q^{-1})$, $B(q^{-1})$ and $C(q^{-1})$ as previously defined in Chapter 2. A diagrammatic representation of equation (3.1) is illustrated in Figure 3.1

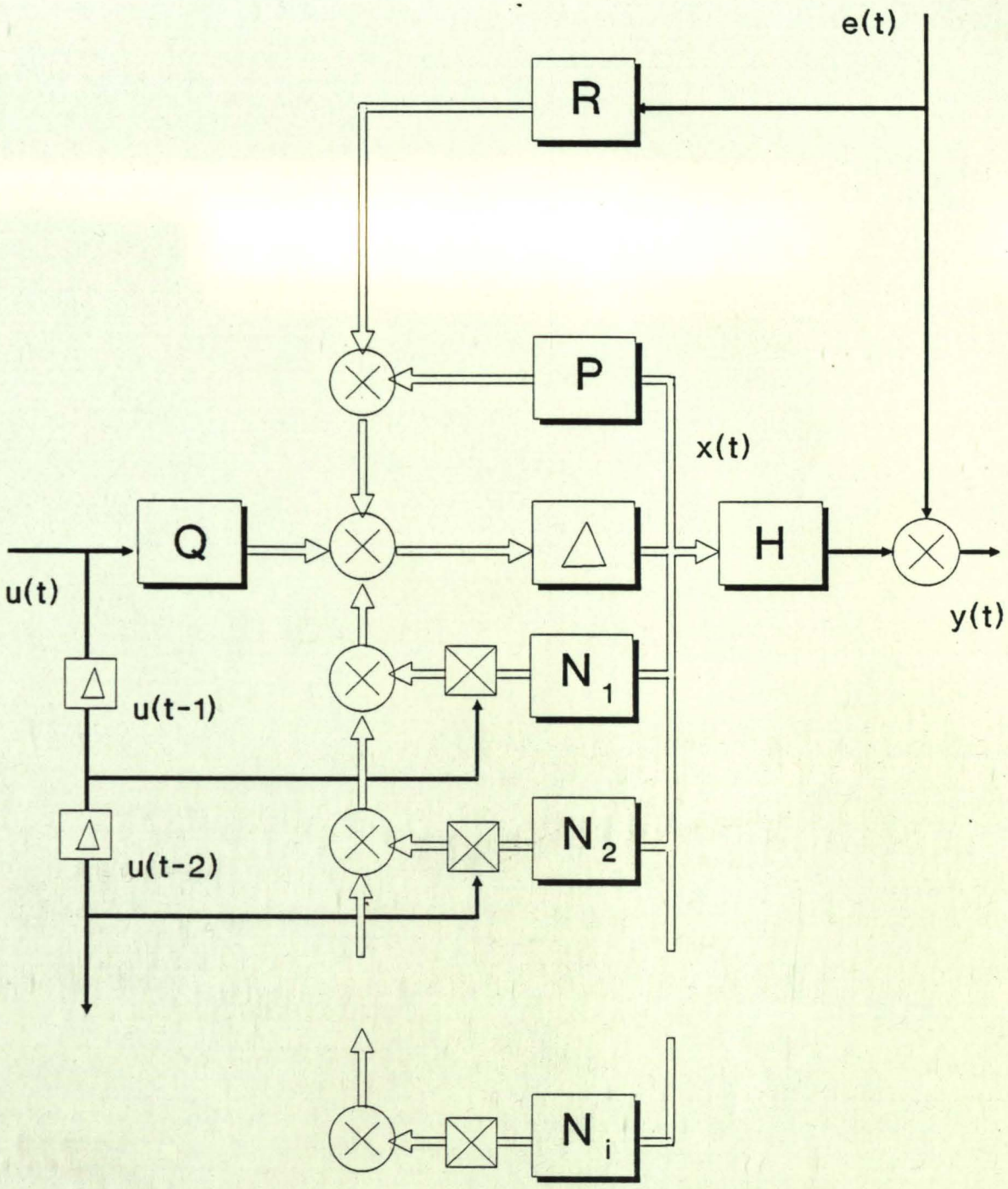


Figure 3.1. Illustrating SISO bilinear system SBL(n,m).

In adopting the state-space self-tuning framework for the bilinear STC, two model structures are required; a polynomial structure for parameter estimation and a state-space structure for control law implementation. Since all forms of equation (3.1) will not be a realisation of a polynomial bilinear system PBL(n,m), the following equivalent structures are adopted.

Structure 1

State-space innovations model SBL(n,m):

$$\mathbf{P} = \begin{bmatrix} 0 & \cdots & & 0 \\ 1 & & 0 & \vdots \\ & \ddots & & 0 \\ & & \ddots & -a_{n_j} \\ & & & \vdots \\ & & & \vdots \\ 0 & & 1 & -a_1 \end{bmatrix} \quad \mathbf{Q} = \begin{bmatrix} b_{n_j} \\ \vdots \\ \vdots \\ b_0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

$$\mathbf{R} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ (c_{n_j}-a_{n_j}) \\ \vdots \\ \vdots \\ (c_1-a_1) \end{bmatrix} \quad \mathbf{H}^T = \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ \vdots \\ 0 \\ 1 \end{bmatrix} \quad \mathbf{N}_i = \begin{bmatrix} 0 & \cdots & 0 & \eta_{n_j i} \\ \vdots & & \vdots & \vdots \\ \vdots & & \vdots & \vdots \\ \vdots & & \vdots & \eta_{0i} \\ 0 & & 0 & 0 \\ \vdots & & \vdots & \vdots \\ 0 & & 0 & 0 \end{bmatrix} \qquad (3.2)$$

Note that the form of **P**, **Q**, **R** and **H** is the same as those given in equation (2.3). Any system of the form (3.1) may be represented in the form (3.2) by a similarity transformation provided the linearised part of (3.1) is observable and that rank$\{\mathbf{N}_i^T\mathbf{H}^T\}$=1 for all i [42].

■

Polynomial bilinear model PBL(n,m):

$$A(q^{-1})x(t) = q^{-k}\, B(q^{-1})u(t) + [\, C(q^{-1}) - A(q^{-1}) \,]\, e(t)$$

$$+ q^{-k} \sum_{i=0}^{n_b} \sum_{j=1}^{m} x(t-i)u(t-i-j+1)\eta_{ij} \qquad (3.3a)$$

$$y(t) = x(t) + e(t) \qquad (3.3b)$$

where $x(t) \in \mathbb{R}$ is the unmeasurable output and, due to the form of **H** in (3.2), $x(t) = x_n(t)$ the n^{th} component of $\mathbf{x}(t)$.

Allowing for possible reduction in dimension, as indicated in Chapter 2, the class of systems is restricted such that $n_a \leq n_b + k$.* Following the reduction process the matrices **P** and **R** may not have the k leading zeros as indicated in (3.2) and the dimension of the state-space model is denoted $n \leq n_i$.

■

3.3 Control algorithms considered

In the linear self-tuning framework pole-placement via state variable feedback provides a straightforward procedure, however, in the bilinear self-tuning framework a number of interesting options are made possible. In this section four such options are outlined.

*This restriction is only necessary if reduction in the state-space dimension is required. It is still necessary for practical purposes that $n_a \geq n_b$.

3.3.1 Outline of approach

The control algorithms considered all make use of the general state variable feedback form

$$u(t) = \mathbf{F}\hat{\mathbf{x}}(t) \tag{3.4}$$

where, as outlined in Chapter 2, $\hat{\mathbf{x}}(t)$ is the estimated state vector and $\mathbf{F}$ is the feedback vector chosen such that the resulting closed-loop system behaves in some pre-specified manner. In developing the self-tuning controller for bilinear systems four potential self-tuning algorithms were investigated. With the exception of the first algorithm, all schemes considered make use of an extended form of the linear state variable feedback control law given by equation (3.4):

i) As an initial starting point it is pertinent to consider the application of a linear STC scheme (indeed this would be a natural first choice in reality, with linear STC strategies being applied to non-linear systems generally). This approach, which forms the basis of *Algorithm 1*, continually invokes the assumption that inherent plant non-linearities are adequately absorbed within the time varying nature of the parameters of an assumed linear model.

ii & iii) An alternative approach is to estimate the parameters of the bilinear system, a step in itself which can be achieved in a number of different ways, and then to separate the model into its constituent linear and non-linear (bilinear) parts. The feedback vector is again calculated as a function of the estimated parameters of the linear part which, in adopting this approach, should be more accurately estimated due to the non-absorbtion of the elements of $\mathbf{N}_i$. Having obtained the feedback vector, the next step is to reconstruct the state vector, again a step which can be achieved in a number of different ways. In this respect, *Algorithm 2* makes use of a time-step quasi-linearisation procedure when an extended SKF, termed an iterated steady-state observer (ISO), is adopted. *Algorithm 3*, on the other hand,

makes use of an equivalent full order state observer (FLO). In both cases, direct use is made of the estimated bilinear parameters in the state reconstruction stage.

iv) A further possibility, which forms the basis of *Algorithm 4*, is to estimate the parameters of the bilinear system, as in (ii) and (iii), and then to combine the linear and non-linear parts to form a 'lumped' quasi-linear model. A quasi-linear feedback vector is then calculated as a function of the 'lumped' model and the state vector obtained from a quasi-linear SKF in the form of the ISO used in *Algorithm 2*.

3.3.2 **Parameter estimation**

The recursive least squares (RLS) technique has found wide acceptance for parameter estimation when dealing with linear systems and forms the basis of the approach adopted here. Despite convergence and bias problems it is believed that, due to ease of implementation and computational efficiency, the RLS algorithm is preferable to the maximum likelihood approach [41] when dealing with bilinear systems.

Substituting (3.3b) into (3.3a) leads to the non-linear ARMAX (or NARMAX) representation

$$A(q^{-1})y(t) = q^{-k}\, B(q^{-1})u(t) + C(q^{-1})e(t) + q^{-k} \sum_{i=0}^{n_b} \sum_{j=1}^{m} x(t-i)u(t-i-j+1)\eta_{ij} \qquad (3.5)$$

from which it is clear that by setting $\eta_{ij}=0$, for all i, j, is equivalent to the linear ARMAX representation (2.1). By setting the noise colouring polynomial $C(q^{-1})=1$, equation (3.5) is readily rearranged into the form of equation (2.4) which is then suitable for application of RLS.

$$y(t) = \boldsymbol{x}^{T}(t)\boldsymbol{\theta}(t) + \xi(t) \qquad (3.6)$$

where $\xi(t)$ is a sequence of fitting errors which becomes equal to the noise sequence upon

convergence, θ is the extended parameter vector,

$$\theta^T(t) = [\ -a_1 \ \ldots\ -a_{na}\ ;\ b_0 \ldots b_{nb}\ \ ;\ \eta_{01}\ \ \ldots\eta_{nb1}\ ;\ \ldots\ ;\ \eta_{0m}\ \ \ldots\eta_{nbm}\ ] \tag{3.7}$$

and $x(t)$ is the corresponding observation vector,

$$x^T(t) = [\ y(t-1) \ldots y(t-n_a)\ ;\ \ u(t-k)\ \ldots u(t-k-n_b\ );\ x(t-k)u(t-k)\ldots\ x(t-k-n_b)u(t-k-n_b);$$

$$\ldots\ ;\ x(t-k)u(t-k-m+1)\ \ldots\ x(t-k-n_b)u(t-k-n_b-m+1)] \tag{3.8}$$

Note that only the input and output sequences u(t) and y(t) are available so that the sequence x(t), which is the noise free system output, (x(t)=y(t)–e(t) from equation (3.3b)), is unmeasurable. In order to form an unbiased basis from which to assess the proposed control algorithms outlined in section 3.3.1 it is necessary to replace x(t) by y(t) (this is not unreasonable given the assumptions on the noise sequence e(t)), so that the observation vector, equation (3.8), becomes

$$x^T(t) = [\ y(t-1)\ \ldots\ y(t-n_a)\ ;\ u(t-k)\ \ldots\ u(t-k-n_b\ )\ ;\ y(t-k)u(t-k)\ \ldots\ y(t-k-n_b)u(t-k-n_b);$$

$$\ldots\ ;\ \ y(t-k)u(t-k-m+1)\ldots y(t-k-n_b)u(t-k-n_b-m+1)]. \tag{3.9}$$

Having established the most appropriate control algorithm with use being made of observation vector (3.9) further investigations were undertaken with y(t) replaced by $\hat{x}_n(t)$, which is the estimate of $\dot{x}(t)$ (i.e. the noise free system output) obtained from the state reconstruction stage. When use is made of state reconstruction, a more appropriate observation vector takes the form

$$x^T(t) = [\ y(t-1)\ldots\ y(t-n_a)\ ;\ u(t-k)\ldots\ u(t-k-n_b\ );\ \hat{x}_n(t-k)u(t-k)\ldots\hat{x}_n(t-k-n_b)u(t-k-n_b);$$

$$\ldots\ ;\ \hat{x}_n(t-k)u(t-k-m+1)\ldots\hat{x}_n(t-k-n_b)u(t-k-n_b-m+1)]. \tag{3.10}$$

The next phase in the development of an appropriate bilinear STC algorithm is then to compare the overall performance resulting from the use of equations (3.9) and (3.10) in the parameter estimation stage.

The basic form of the RLS algorithm adopted is given by equations (2.30)-(2.32)

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \phi(t)[y(t) - x^T(t)\hat{\theta}(t-1)] \tag{3.11}$$

$$\phi(t) = \Phi(t-1)x(t)[1 + x^T(t)\Phi(t-1)x(t)]^{-1} \tag{3.12}$$

$$\Phi(t) = [I - \phi(t)x^T(t)]\ \Phi(t-1)\ /\lambda \tag{3.13}$$

Note, that in adopting the observation vector given by (3.10), the parameter estimation scheme makes use of the state estimates and the state estimation scheme makes use of the parameter estimates. The resulting overall tandem state/parameter estimation scheme may be illustrated schematically as in Figure 3.2.

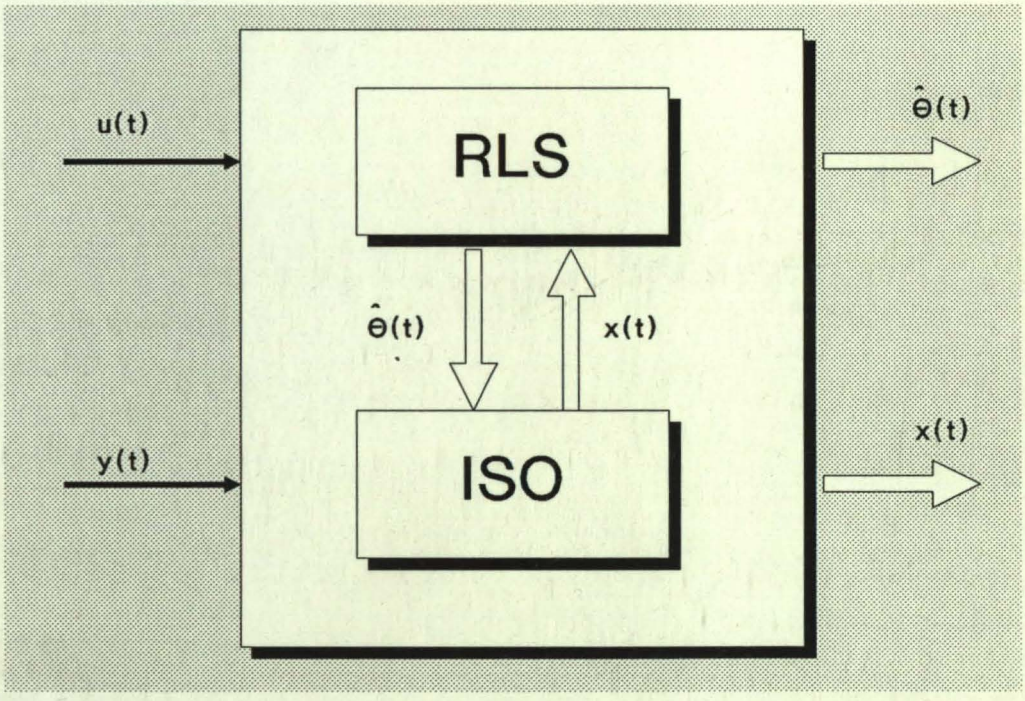


Figure 3.2. Illustrating tandem operation of state/parameter estimation.

3.3.3 Quasi-linearisation

State estimation

Define the quasi-linear state-space representation $\bar{S}BL(n,m)$ of Structure 1 by

$$\mathbf{x}(t+1) = \bar{\mathbf{P}}(\tilde{\mathbf{u}}(t))\mathbf{x}(t) + \bar{\mathbf{Q}}(\mathbf{x}(t))\mathbf{u}(t) + \mathbf{R}\mathbf{e}(t) \tag{3.14a}$$

$$\mathbf{y}(t) = \mathbf{H}\mathbf{x}(t) + \mathbf{e}(t) \tag{3.14b}$$

where

$$\tilde{\mathbf{u}}(t) = [u(t-1)...u(t-m+1)]^T, \quad m \geq 2 \tag{3.15}$$

and

$$\bar{\mathbf{P}}(\tilde{\mathbf{u}}(t)) = \mathbf{P}, \qquad m = 1, \tag{3.16}$$

$$= \mathbf{P} + \sum_{i=2}^{m} u(t-i+1)\mathbf{N}_i, \qquad m \geq 2,$$

with

$$\bar{\mathbf{Q}}(\mathbf{x}(t)) = \mathbf{Q} + \mathbf{N}_1\mathbf{x}(t). \tag{3.17}$$

Substituting the output equation (3.14b) into the state equation (3.14a) yields the quasi-linear full order state observer (FLO)

$$\hat{\mathbf{x}}(t+1) = \bar{\mathbf{P}}_1(\tilde{\mathbf{u}}(t))\hat{\mathbf{x}}(t) + \bar{\mathbf{Q}}(\hat{\mathbf{x}}(t))\mathbf{u}(t) + \mathbf{R}\mathbf{y}(t) \tag{3.18}$$

where $\bar{\mathbf{P}}_1(\tilde{\mathbf{u}}(t)) = \bar{\mathbf{P}}(\tilde{\mathbf{u}}(t)) - \mathbf{RH}$; which on rearranging yields the equivalent steady-state Kalman filter (SKF)

$$\hat{\mathbf{x}}(t) = [\mathbf{I} - q^{-1}\bar{\mathbf{P}}_1(\tilde{\mathbf{u}}(t))]^{-1} \; [\bar{\mathbf{Q}}(\hat{\mathbf{x}}(t))\mathbf{u}(t-1) + \mathbf{R}\mathbf{y}(t-1)] \tag{3.19}$$

which is referred to as the iterated steady-state observer (ISO).

Note that, unlike the SKF for linear systems, convergence is no longer guaranteed. The error dynamics associated with equation (3.19) are dependent on the eigenvalues of the matrix $\bar{\mathbf{P}}_1(\tilde{\mathbf{u}}(t))$,

$$\bar{P}_1(\tilde{u}(t)) = P - RH + N_2u(t-1) + N_3u(t-2) + \ldots + N_mu(t-m+1)$$

$$= P - RH + \sum_{i=2}^{m} N_iu(t-i+1) \qquad (3.20)$$

which are themselves dependent on past values of the control input sequence $\tilde{u}(t)$.

Whilst some initial work on the stability and convergence behaviour of observers for bilinear systems has been undertaken [50], further work on the convergence behaviour of the ISO remains an outstanding issue. It may be possible, however, under the assumption of convergence of RLS, to compute the eigenvalues of $\bar{P}_1(\tilde{u}(t))$ on-line as the process evolves. This would allow bounds on future control action to be placed, thereby ensuring convergence of the ISO. Such a procedure could well be realised as a parallel tasking procedure in the STC jacketing software.

■

Feedback vector

With the exception of the *Algorithm 4*, all proposed approaches make use of the linear form of the feedback vector given by equation (2.42). In *Algorithm 4*, the corresponding quasi-linear feedback vector is given by

$$\bar{F}^T = \bar{W}^{-1}S \qquad (3.21)$$

where $\bar{W}=\bar{K}L$ in which $\bar{K}$ is the *quasi-linear* Kalman controllability test matrix

$$\bar{K} = [\, \bar{P}(\tilde{u}(t))^{n-1}\bar{Q}(\tilde{x}(t)) : \ldots : \bar{P}(\tilde{u}(t))\bar{Q}(\tilde{x}(t)) : \bar{Q}(\tilde{x}(t)) \,] \qquad (3.22)$$

and the matrices L and S are defined as in section 2.4.2.

■

3.3.4 Extended pole-placement algorithms

The four algorithms are simply stated as

Algorithm 1

i) Estimate parameters of an assumed linear model,

ii) Reconstruct $\hat{\mathbf{x}}(t)$ from SKF (equation (2.40))

iii) Compute feedback vector (equation (2.42))

Algorithm 2

i) Estimate parameters of an assumed bilinear model

ii) Reconstruct $\hat{\mathbf{x}}(t)$ from ISO (equation (3.19))

iii) Compute feedback vector (equation (2.42))

Algorithm 3

i) Estimate parameters of an assumed bilinear model

ii) Reconstruct $\hat{\mathbf{x}}(t)$ from FLO (equation (3.18))

iii) Compute feedback vector (equation (2.42))

Algorithm 4

i) Estimate parameters of an assumed bilinear model

ii) Reconstruct $\hat{\mathbf{x}}(t)$ from ISO (equation (3.19))

iii) Compute feedback vector (equation (3.21))

Note that in the first phase of the investigation use is made of the observation vector given by equation (3.9). Having established the most appropriate algorithm, the second phase involved comparative studies with use also being made of the observation vector given by equation (3.10).

The algorithms are summarised in Table 3.1.

Algorithm	Assumed Model Linear (L) Bilinear (BL)	State Reconstruction Equation No.	Compute Feedback Equation No.
1	L	(2.40)	(2.42)
2	BL	(3.19)	(2.42)
3	BL	(3.18)	(2.42)
4	BL	(3.19)	(3.21)

Table 3.1

3.4 Simulation studies

In order to evaluate the four initial algorithms, consideration is given to the following bilinear systems

System 1 PBL(2,1)

$$y(t) = 1.5y(t-1) - 0.7y(t-2) + u(t-1) + 0.5u(t-2) + 0.25x(t-1)u(t-1) + e(t)$$

System 2 PBL(2,2)

$$y(t) = 1.5y(t-1) - 0.7y(t-2) + u(t-1) + 0.5u(t-2) + 0.2x(t-1)u(t-1) - x(t-2)u(t-2) + e(t)$$

For both systems a reduction of the implicit delay state-space representation is required to ensure system controllability. Following reduction, *System 1* may be expressed in the state-space SBL(2,1) form

$$\mathbf{x}(t+1) = \begin{bmatrix} 0 & -0.7 \\ 1 & 1.5 \end{bmatrix}\mathbf{x}(t) + \begin{bmatrix} 0.5 \\ 1.0 \end{bmatrix}u(t) + \begin{bmatrix} -0.7 \\ 1.5 \end{bmatrix}e(t) + u(t)\begin{bmatrix} 0 & 0 \\ 0 & 0.25 \end{bmatrix}\mathbf{x}(t)$$

$$y(t) = [\,0 \;\; 1\,]\,\mathbf{x}(t) + e(t).$$

Similarly for *System 2* the matrix $\mathbf{N}_1$ becomes

$$\mathbf{N}_1 = \begin{bmatrix} 0 & -1.0 \\ 0 & 0.2 \end{bmatrix}.$$

3.4.1 Comparison of four algorithms

The two systems were simulated using the proposed control algorithms of section 3.3.1; the objective of the controller being to regulate the system output about a zero reference in the presence of noise. To give a sufficiently fast response to any disturbance the closed-loop poles, or eigenvalues, of the linear part were specified as repeated poles at -10 in the s-plane, which for a sampling interval T=1 second, gives rise to a virtual dead beat response, i.e. $z=e^{sT}\approx 0$. In each case the simulation is run over 100 iterations with a fixed forgetting factor of $\lambda=0.98$. The results are shown in Tables 3.2(a)(b)(c) and 3.3(a)(b) where σ_y, $\bar{y}$, σ_u and $\bar{u}$ denote the variance and mean of the system output y(t) and the control input u(t) respectively and a_i, b_i and η_{i1} denote the estimated parameter values after each simulation run. (A blank entry in these Tables indicates that instability has occurred.)

The results given in Table 3.2(a), (b) and (c) compare algorithms 1, 2 and 3 and relate to *System 1*. Columns 1 to 4 correspond to simulation under various levels of white noise output disturbance and constant parameters. In columns 5 to 11 the bilinear

parameter η_{01} is varied linearly from its nominal value of 0.25 to 0.5 over the duration of the simulation, with columns 5 to 8 and 9 to 11 corresponding to various levels of white and coloured noise respectively. In the case of coloured noise, the noise colouring polynomial

$$C(q^{-1}) = 1 - 0.75q^{-1} + 0.25q^{-2}$$

is adopted.

The results given in Table 3.3(a) and (b) compare algorithms 2 and 4 and relate to *System 2*. In the case of varying parameters, columns 5 and 6 correspond to varying the parameter a_2 linearly from its nominal value of 0.7 to 1.4 over the duration of the simulation whilst column 7 corresponds to the simultaneous variation of a_2 and η_{11}, with η_{11} being varied linearly from its nominal value of -1.0 to -2.0. Columns 8 to 10 correspond to bounds being placed on the control effort, $\left| u(t) \right| \leq 1.0$; noting that for the noise levels considered both algorithms gave rise to instability when the control was unbounded.

The results of Table 3.2 indicate that the variance of the control effort is always less when use is made of *Algorithm 2*. However, for noise levels lower than those tabulated *Algorithm 1* produces comparable results. As the noise level is increased, *Algorithm 3* is the first to give rise to instability, followed by *Algorithm 1* with *Algorithm 2* continuing to maintain stability for higher noise levels. Figures 3.3 and 3.4 illustrate control input (a) and system response (b) for *Algorithms 1* and *2* respectively and correspond to the test conditions of column 7 in Tables 3.2(a) and 3.2(b).

The results of Table 3.3 indicate that in the case of unbounded control, *Algorithm 2* again produces the more favourable results with a significantly lower control input variance than *Algorithm 4*. Furthermore, in the case of bounded control, whilst the control input variances are of the same order, *Algorithm 2* produces a system response with significantly lower variance. Figures 3.5 and 3.6 illustrate control input and system

response for *Algorithms 2* and *4* respectively and correspond to the test conditions of column 3 of Tables 3.3(a) and 3.3(b). Similarly Figures 3.7 and 3.8 correspond to the test conditions of column 8.

It is noted for both systems, that when use is made of unbounded control, good initial estimates are required for the linear parameters otherwise instability can arise during the initial iterations. This requirement is not essential when a bounded control is used. Provided either good initial estimates of the linear parameters are available, or that the control is initially bounded, it is found that the initial estimates of the bilinear coefficients may be arbitrarily set at zero.

Remarks

i) For the simulation studies undertaken, *Algorithm 2* gives rise to a superior performance over the other proposals in the sense that the closed-loop system remain stable for a wider range of conditions in terms of measurement noise and parameter variations. Also, the variance of the control effort and system response is significantly smaller than the other algorithms operating under convergence conditions; a desirable feature of any control system.

ii) The simulation studies also indicate that for convergence of parameters and stability of the closed-loop system it is necessary to have good initial estimates of the parameters of the linear part of the system if an unbounded control is to be used; thus highlighting the vulnerability of bilinear systems to numerical instability particularly during initialisation, or 'start-up' stages.

iii) The results of the initial investigations, albeit limited, indicate that the self-tuning principle may be successfully applied to bilinear systems. In the development of the bilinear STC throughout this work, *Algorithm 2* is taken to provide the basic framework.

■

Noise levelN peak to peak		CONSTANT PARAMETERS				TIME VARYING PARAMETERS (BILINEAR TERM)						
		WHITE NOISE				WHITE NOISE				COLOURED NOISE		
		0.75	0.85	1.00	1.25	0.50	0.75	0.85	1.00	0.50	0.75	0.85
COLUMN		1	2	3	4	5	6	7	8	9	10	11
	b_0	0.85	0.78	0.60		1.00	0.78	0.82		0.40	1.00	
	b_1	0.53	0.49	0.49		0.60	0.52	0.53		0.91	1.88	
	a_1	-1.53	-1.50	-1.33		-1.52	-1.49	-1.45		-0.65	-0.79	
	a_2	0.79	0.73	0.44		0.74	0.73	0.73		-7.4E-2	-0.33	
	σ_y	0.52	0.80	0.87		0.10	0.62	0.52		0.20	0.49	
	$\bar{y}$	-0.28	-0.46	-0.20		-5.2E-2	-0.36	-0.39		-2.6E-2	-7.5E-2	
	σ_u	0.54	0.79	1.30		0.15	0.65	0.72		7.4E-2	0.56	
	$\bar{u}$	-2.4E-2	-4.7E-2	8.3E-2		-1.5E-2	-2.3E-2	-3.5E-2		-1.7E-5	-1.7E-2	

Table 3.2(a). Algorithm 1.

Noise level N peak to peak		CONSTANT PARAMETERS				TIME VARYING PARAMETERS (BILINEAR TERM)						
		WHITE NOISE				WHITE NOISE				COLOURED NOISE		
		0.75	0.85	1.00	1.25	0.50	0.75	0.85	1.00	0.50	0.75	0.85
COLUMN		1	2	3	4	5	6	7	8	9	10	11
	b_0	1.01	1.03	1.02	0.98	0.98	0.99	1.00	1.01	1.02	0.86	0.46
	b_1	0.55	0.53	0.49	0.49	0.58	0.57	0.54	0.49	1.64	1.45	0.85
	a_1	-1.49	-1.49	-1.45	-1.47	-1.44	-1.44	-1.43	-1.44	-0.90	-0.91	-0.72
	a_2	0.66	0.69	0.71	0.72	0.61	0.63	0.65	0.69	-0.15	-0.11	-1.9E-2
	η_{01}	0.27	0.27	0.27	0.25	0.28	0.29	0.29	0.28	3.1E-2	0.18	0.31
	σ_y	0.24	0.34	0.73	1.74	9.3E-2	0.25	0.38	1.16	0.30	0.70	0.92
	$\bar{y}$	-1.7E-2	-2.4E-2	-0.12	-0.29	-1.2E-2	-3.3E-2	-4.3E-2	-0.26	-4.9E-2	-8.9E-2	-0.25
	σ_u	0.38	0.55	1.00	1.84	0.14	0.38	0.58	1.32	5.5E-2	0.15	0.39
	$\bar{u}$	2.2E-2	3.4E-2	6.5E-2	0.13	7.2E-3	2.6E-2	4.1E-2	9.3E-2	4.4E-3	2.4E-2	2.5E-2

Table 3.2(b). Algorithm 2.

Noise level N peak to peak		CONSTANT PARAMETERS				TIME VARYING PARAMETERS (BILINEAR TERM)						
		WHITE NOISE				WHITE NOISE				COLOURED NOISE		
		0.75	0.85	1.00	1.25	0.50	0.75	0.85	1.00	0.50	0.75	0.85
COLUMN		1	2	3	4	5	6	7	8	9	10	11
	b_0	1.08	1.09			1.00	1.05			0.94	0.70	
	b_1	0.55	0.54			0.57	0.55			1.58	0.71	
	a_1	-1.56	-1.60			-1.46	-1.50			-1.05	-0.88	
	a_2	0.72	0.75			0.63	0.68			-4.2E-2	4.4E-2	
	η_{01}	0.27	0.26			0.32	0.28			0.29	0.31	
	σ_y	0.24	0.37			9.3E-2	0.26			0.43	0.92	
	$\bar{y}$	-1.5E-2	-2.7E-2			-8.9E-3	-3.4E-2			-7.2E-2	0.23	
	σ_u	0.58	1.19			0.16	0.68			8.4E-2	0.59	
	$\bar{u}$	3.1E-2	5.5E-2			8.4E-3	3.8E-2			-1.5E-5	8.5E-2	

Table 3.2(c). Algorithm 3.

		CONTROL : UNBOUNDED							BOUNDED			
White noise N peak - peak		PARAMETERS: CONSTANT				TIME VARYING			CONSTANT			
		0.10	0.25	0.30	0.35	0.25	0.30	0.30	0.4	0.5	0.6	0.4
COLUMN		1	2	3	4	5	6	7	8	9	10	11
	b_0	1.09	1.06	1.05	1.06	0.79	0.79	0.82	1.01	1.12	1.14	1.00
	b_1	0.58	0.57	0.53	0.51	0.39	0.38	0.36	0.57	0.66	0.61	0.51
	a_1	-1.55	-1.56	-1.57	-1.55	-1.23	-1.20	-1.20	-1.54	-1.54	-1.56	-1.48
	a_2	0.68	0.64	0.66	0.68	0.95	0.95	0.98	0.75	0.69	0.71	0.67
	η_{01}	9.5E-2	5.1E-2	0.10	0.10	0.12	0.12	0.11	-0.12	-0.10	0.15	0.21
	η_{11}	-0.37	-0.71	-0.81	-0.90	-0.9	-0.98	-1.10	-0.93	-0.93	-0.91	-0.98
	σ_y	3.1E-3	2.5E-2	4.3E-2	8.0E-2	2.8E-2	4.6E-2	5.3E-2	5.1E-2	7.3E-2	0.16	1.58
	$\bar{y}$	5.4E-3	2.3E-2	3.9E-2	7.6E-2	1.9E-2	3.0E-2	5.3E-2	4.3E-2	6.0E-2	0.13	1.52
	σ_u	4.9E-3	3.9E-2	6.8E-2	0.13	3.6E-2	5.9E-2	8.1E-2	5.2E-2	5.4E-2	9.4E-2	0.58
	$\bar{u}$	-1.7E-3	-1.1E-2	-1.8E-2	-2.8E-2	-9.2E-2	-1.5E-2	-1.9E-2	-1.1E-2	-1.3E-2	-2.3E-2	-5.1E-2

Table 3.3(a). Algorithm 2.

		CONTROL : UNBOUNDED							BOUNDED			
White noise N peak - peak		PARAMETERS: CONSTANT				TIME VARYING			CONSTANT			
		0.10	0.25	0.30	0.35	0.25	0.30	0.30	0.4	0.5	0.6	0.4
COLUMN		1	2	3	4	5	6	7	8	9	10	11
	b_0	1.07	1.00	0.99		0.79	0.79	0.81	1.10	1.10	1.03	
	b_1	0.52	0.54	0.47		0.55	0.39	0.45	0.69	0.65	0.71	
	a_1	-1.56	-1.49	-1.51		-1.43	-1.20	-1.30	-1.54	-1.54	-1.51	
	a_2	0.69	0.67	0.72		0.97	0.93	0.86	0.75	0.69	0.65	
	η_{01}	0.41	0.21	0.22		9.9E-2	0.19	0.21	-0.12	-0.10	0.24	
	η_{11}	-1.08	-0.96	-0.95		-1.00	-0.92	-1.00	-0.93	-0.93	-0.99	
	σ_y	3.9E-3	4.5E-2	8.2E-2		6.2E-2	0.10	0.23	5.1E-2	7.3E-2	0.27	
	$\bar{y}$	2.1E-3	-1.8E-2	2.2E-2		2.3E-2	2.9E-2	4.6E-2	4.3E-2	6.0E-2	0.14	
	σ_u	6.2E-3	6.9E-2	0.12		8.8E-2	0.14	0.25	5.2E-2	5.4E-2	9.8E-2	
	$\bar{u}$	-2.4E-3	-1.8E-2	-4.0E-2		-2.2E-2	-4.4E-2	-7.2E-2	-1.1E-2	-1.3E-2	-3.0E-2	

Table 3.3(b). Algorithm 4.

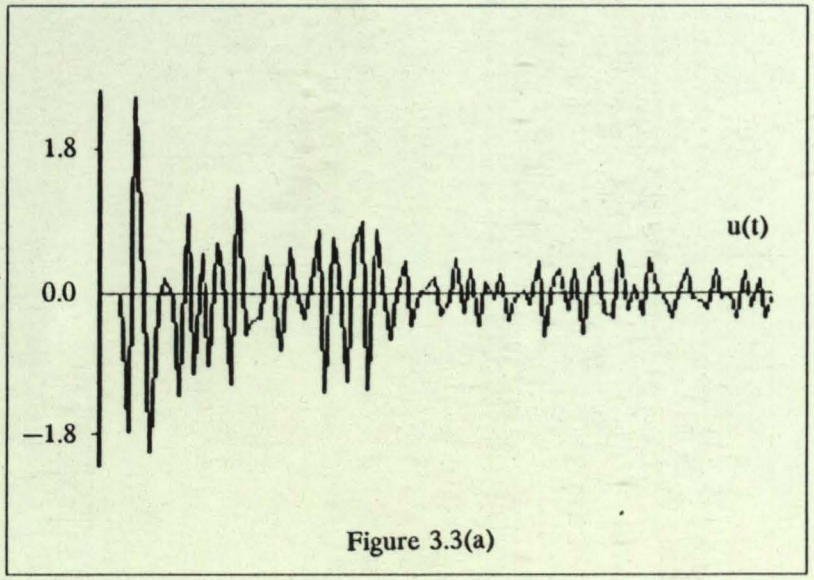


Figure 3.3(a)

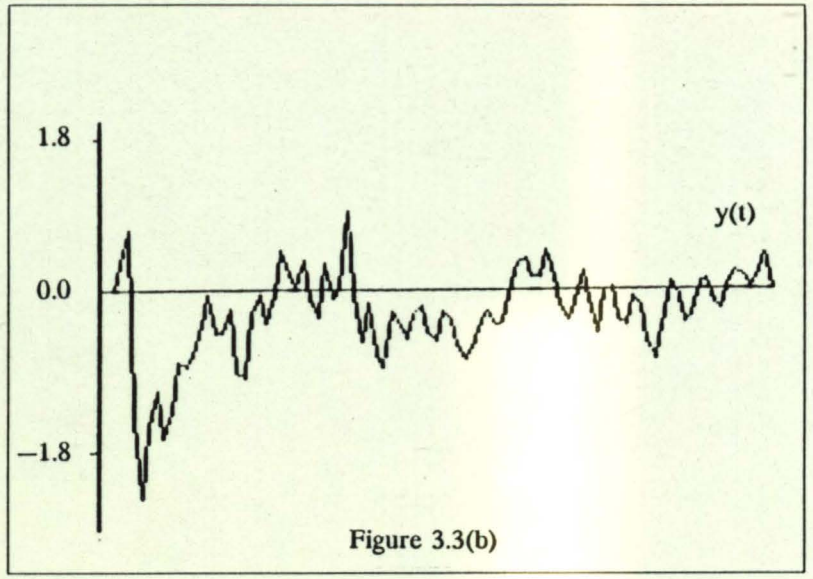


Figure 3.3(b)

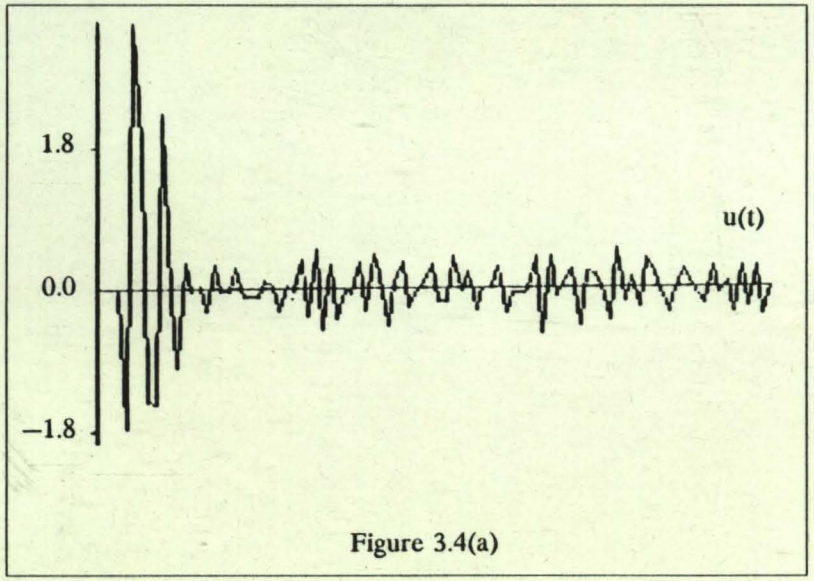


Figure 3.4(a)

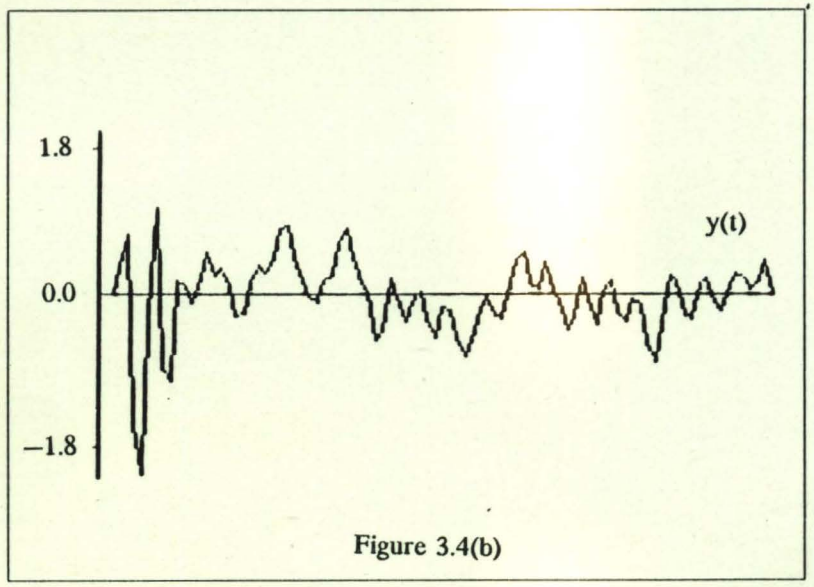


Figure 3.4(b)

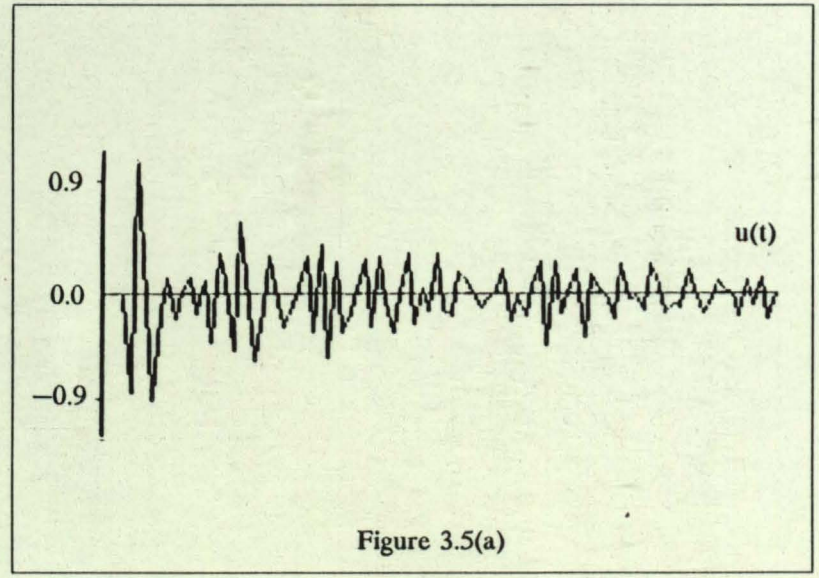


Figure 3.5(a)

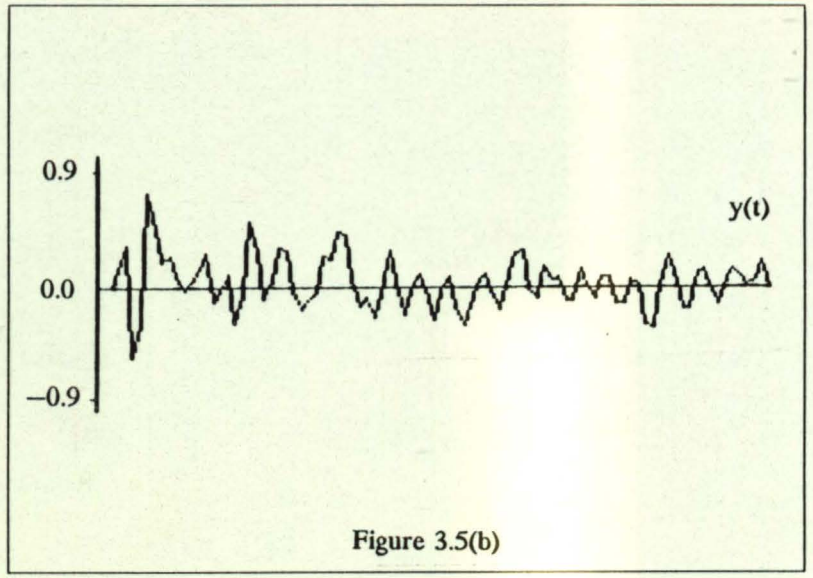


Figure 3.5(b)

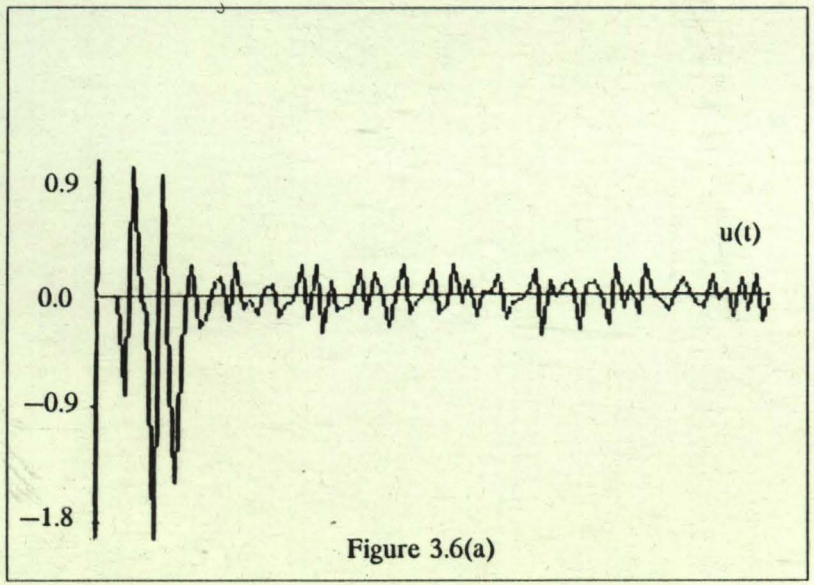


Figure 3.6(a)

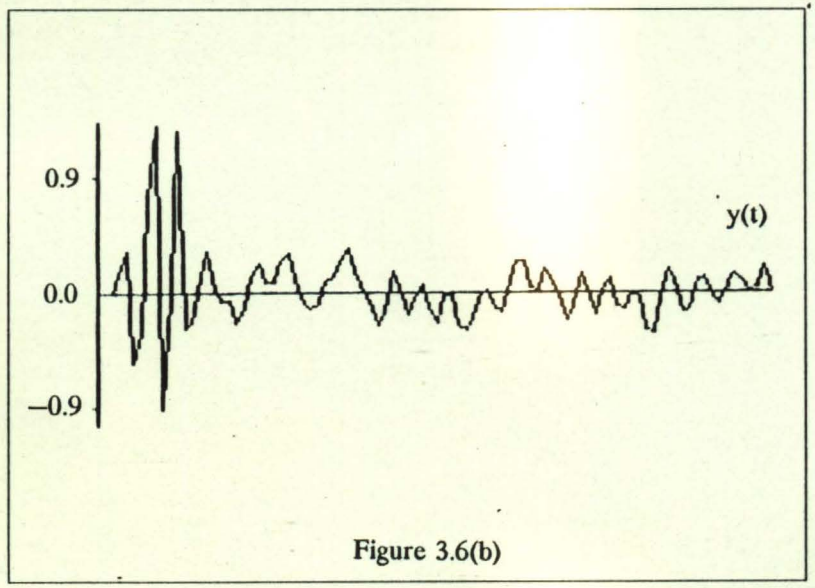


Figure 3.6(b)

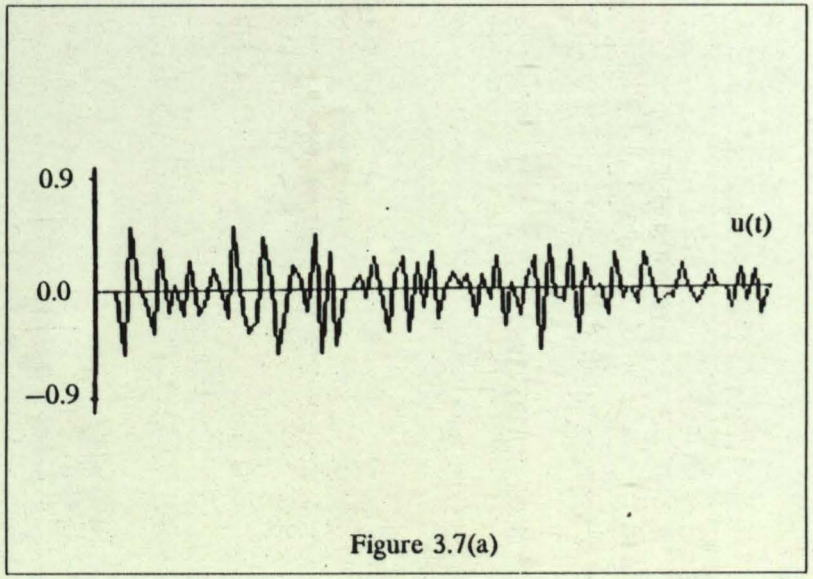


Figure 3.7(a)

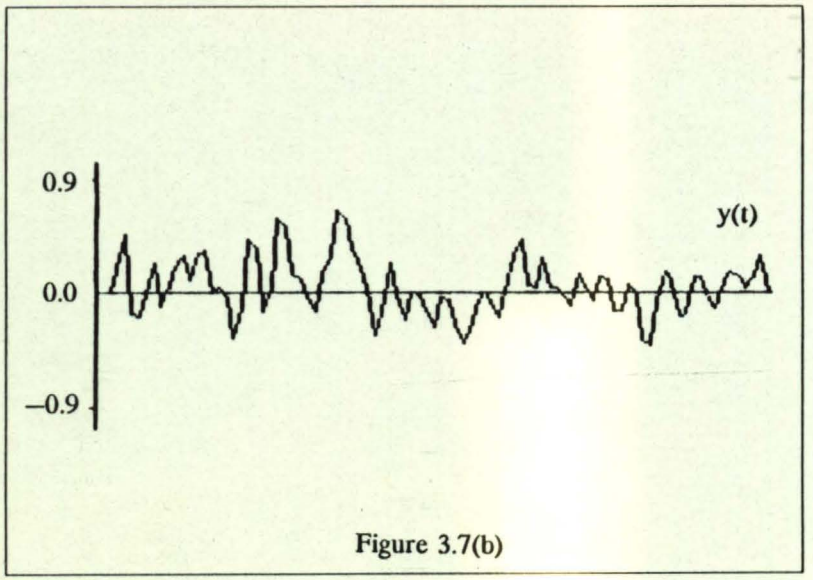


Figure 3.7(b)

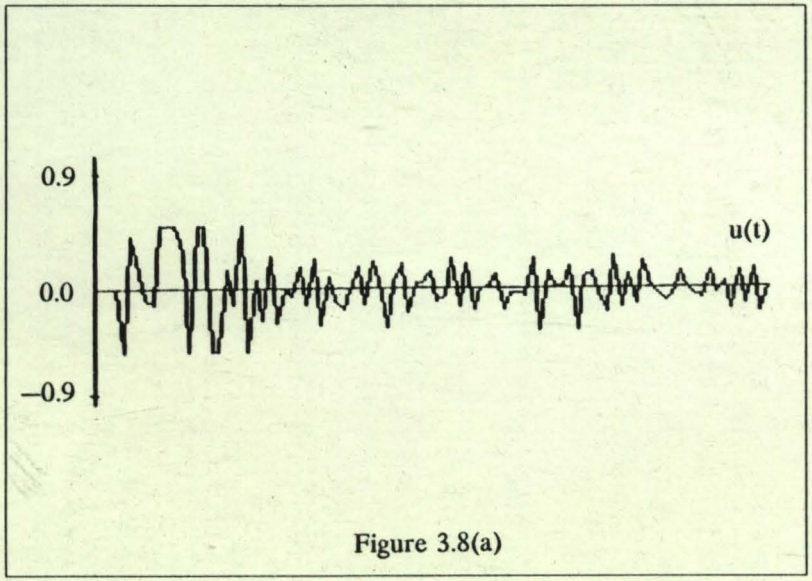


Figure 3.8(a)

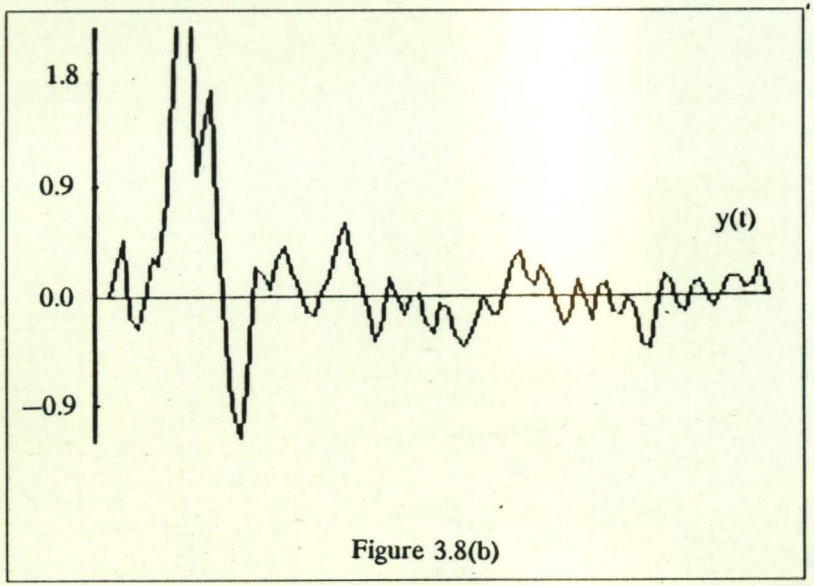


Figure 3.8(b)

3.4.2 Alternative form of the observation vector

Having established the basis of the bilinear STC, the second phase involved further comparative simulation studies in order to investigate the effectiveness of the alternative observation vector given by equation (3.10). This observation vector differs from that given by equation (3.9) in that the measured output y(t-1) is replaced by $\hat{x}_n(t-1)$ in the product terms corresponding to the bilinear coefficients.

System 1 is adopted for simulation trials when the objective of the controller is to place poles at -3 and -5 in the s-plane with an effective sampling interval of T=0.1 second. The system is subjected to a range of output noise levels N and, in each case, various values of the fixed forgetting factor are investigated.

Results given in Tables 3.4(a) and 3.4(b) correspond to the use of the measured output y(t-1) and the estimated noise free output $\hat{x}_n(t-1)$ in the product term of the observation vectors respectively. In each Table, σ_y, $\bar{y}$, σ_u and $\bar{u}$ denote the variance and mean values of the system response and control input respectively.

The results show that use of $\hat{x}_n(t-1)$ in the observation vector, particularly in the case of high noise levels, leads to an improved performance in terms of reduced variance of both control input and system response. This is to be expected due to the inherent filtering effects of the iterated steady-state observer (ISO) of equation (3.19). In the presence of noise levels, however, there is little difference in performance with the use of the measured output y(t-1) perhaps introducing a marginal improvement. This observation once again indicates that a little noise on the system is often advantageous from an improved estimation point of view.

N	λ	1.00	0.98	0.95	0.93	0.90
0.10	σ_y	4.8E-2	3.3E-2	3.2E-2	3.1E-2	3.8E-2
	$\bar{y}$	-3.6E-2	-4.7E-2	-4.5E-2	-3.8E-2	-5.1E-2
	σ_u	7.6E-4	6.6E-4	6.6E-4	8.5E-4	8.4E-4
	$\bar{u}$	-1.1E-3	-2.7E-3	-2.3E-3	-1.4E-3	-3.1E-3
0.35	σ_y	0.710	0.634	0.553	0.692	0.557
	$\bar{y}$	-5.7E-2	-4.8E-2	-2.4E-2	-9.3E-2	2.7E-2
	σ_u	1.3E-2	1.2E-2	2.4E-2	1.5E-2	1.3E-2
	$\bar{u}$	-2.5E-3	3.6E-4	5.1E-3	-6.7E-3	1.2E-2
0.40	σ_y	1.541	1.190	1.042	1.051	1.104
	$\bar{y}$	0.222	0.159	0.194	0.207	0.125
	σ_u	2.3E-2	2.0E-2	2.9E-2	2.3E-2	2.5E-2
	$\bar{u}$	2.5E-2	2.1E-2	2.9E-2	3.0E-2	1.8E-2

Table 3.4(a). Use of measured output y(t-1).

It is further noted that use of a forgetting factor less than unity produces the more favourable results. However, use of such a forgetting factor in the presence of prolonged periods of steady-state operation can lead to the problem of covariance blow-up. The latter conflicting observations prompt the need for a variable forgetting factor and this is investigated in Chapter 4.

N	λ	1.00	0.98	0.95	0.93	0.90
0.10	σ_y	4.6E-2	3.4E-2	3.2E-2	3.0E-2	7.2E-2
	$\bar{y}$	-3.7E-2	-4.7E-2	-4.3E-2	-3.5E-2	-8.0E-2
	σ_u	7.4E-4	6.7E-4	6.7E-4	7.7E-4	1.8E-3
	$\bar{u}$	-1.4E-3	-2.6E-3	-2.1E-3	-9.4E-4	-7.8E-3
0.35	σ_y	0.732	0.624	0.867	0.558	0.595
	$\bar{y}$	-4.6E-2	-4.9E-2	-0.132	-5.9E-2	-0.150
	σ_u	1.4E-2	1.2E-2	6.6E-2	1.3E-2	1.4E-2
	$\bar{u}$	-1.5E-3	4.3E-4	-1.0E-2	1.7E-2	-1.2E-2
0.40	σ_y	1.530	1.090	0.890	0.819	0.863
	$\bar{y}$	0.296	0.108	3.1E-2	-3.7E-2	-7.6E-2
	σ_u	2.4E-2	1.9E-2	0.126	1.9E-2	3.6E-2
	$\bar{u}$	3.3E-2	1.6E-2	1.8E-2	1.4E-3	-9.9E-4

Table 3.4(b). Use of state estimate $\hat{x}_n(t\text{-}1)$.

3.5 Extension of the bilinear STC framework for MISO systems

In this section, the bilinear approach is extended to accommodate multiple-input single-output (MISO) bilinear model structures. Such systems are found to be representative of interconnected non-linear sub-systems such as, for example, computer architecture structures appropriate for parallel processing [51].

3.5.1 MISO model structure

The MISO state-space bilinear model structure SBL(n,m,r) is an extension of the SISO SBL(n,m) structure of equation (3.1). It takes the form

$$\mathbf{x}(t+1) = \mathbf{P}\mathbf{x}(t) + \mathbf{Q}\mathbf{u}(t) + \mathbf{R}e(t) + \sum_{j=1}^{r}\sum_{i=1}^{m} u_j(t-i+1)\mathbf{N}_i^j\,\mathbf{x}(t) \tag{3.23a}$$

$$y(t) = \mathbf{H}\mathbf{x}(t) + e(t) \tag{3.23b}$$

where $\mathbf{x}(t) \in \mathbb{R}^n$, $\mathbf{u}(t) \in \mathbb{R}^r$, $y(t)$, $e(t) \in \mathbb{R}$, are, respectively, the system state vector, input vector, output and white noise sequences. The form of the matrices **P**, **R** and **H** are as previously defined in (3.2) with the partitioned input matrix **Q** and bilinear coefficient matrices $\mathbf{N}_i^j$ taking the form

$$\mathbf{Q} = \begin{bmatrix} b_{n_i 1} & \cdots & b_{n_i r} \\ \vdots & & \vdots \\ b_{01} & \cdots & b_{0r} \\ 0 & & 0 \\ \vdots & & \vdots \\ 0 & & 0 \end{bmatrix} \qquad \mathbf{N}_i^j = \begin{bmatrix} 0 & \cdots & 0 & \eta_{n_i i}^j \\ \vdots & & \vdots & \vdots \\ \vdots & & \vdots & \vdots \\ \vdots & & \vdots & \eta_{0i}^j \\ 0 & & 0 & 0 \\ \vdots & & \vdots & \vdots \\ 0 & & 0 & 0 \end{bmatrix} \tag{3.24}$$

where n_i is the initial dimension of the state-space.

The equivalent polynomial bilinear model structure PBL(n,m,r) which is adopted in the parameter estimation stage of the MISO bilinear STC takes the form

$$A(q^{-1})x(t) = q^{-k}\sum_{j=1}^{r} B_j(q^{-1})u_j(t) + [C(q^{-1})-A(q^{-1})]\, e(t)$$

$$+ q^{-k}\sum_{\ell=0}^{n_i}\sum_{i=1}^{m}\sum_{j=1}^{r} x(t-\ell)\, u_j(t-i-\ell+1)\eta_{\ell i}^{j} \qquad (3.25a)$$

$$y(t) = x(t) + e(t) \qquad (3.25b)$$

where $x(t)\in\mathbb{R}$ is the unmeasurable noise free system output and, due to the form of **H** in (3.2), $x(t)=x_n(t)$ the n^{th} component of $\mathbf{x}(t)$. The polynomials A(q-1) and C(q-1) are as defined in Chapter 2 with the polynomials $B_j(q^{-1})$, j=1,2, . . r, defined as

$$B_1(q^{-1}) = b_0^1 + b_1^1 q^{-1} + b_2^1 q^{-2} + \ldots + b_{n_b}^1 q^{-n_b}$$

$$B_2(q^{-1}) = b_0^2 + b_1^2 q^{-1} + b_2^2 q^{-2} + \ldots + b_{n_b}^2 q^{-n_b}$$

. . . . .

. . . . .

. . . . .

$$B_r(q^{-1}) = b_0^r + b_1^r q^{-1} + b_2^r q^{-2} + \ldots + b_{n_b}^r q^{-n_b} \qquad (3.26)$$

As in the SISO formulation of section 3.2, in order to ensure no loss of information and system controllability, the class of systems is restricted such that $n_a \leq n_b^j + k$, j=1,2, . . r. In practice this may require a reduction in the dimension of the state-space model so that dim{state-space}=n_b+k=n.

3.5.2 **Parameter estimation**

As in the case of SISO bilinear systems, since only the known inputs, measured outputs and estimated state variables are available, it is necessary to

re-formulate the PBL(n,m,r) model (3.25). Substituting (3.25b) into (3.25a) and setting the noise colouring polynomial $C(q^{-1})$ to unity leads to

$$A(q^{-1})y(t) = q^{-k}\sum_{j=1}^{r} B_j(q^{-1})u_j(t) + q^{-k}\sum_{\ell=0}^{n_i}\sum_{i=1}^{m}\sum_{j=1}^{r}\hat{x}(t-\ell)\,u_j(t-i-\ell+1)\eta_{\ell i}^{j} + e(t) \qquad (3.27)$$

which is now in the form of equation (3.5) and suitable for RLS. Rearranging yields

$$y(t) = \boldsymbol{x}^{T}(t)\boldsymbol{\theta}(t) + \xi(t)$$

where $\boldsymbol{x}(t)$ and $\boldsymbol{\theta}(t)$ are respectively of the form (3.7) and (3.8), extended appropriately to accommodate the additional (r-1) inputs. That is

$$\begin{aligned}\boldsymbol{\theta}^{T}(t) = [\ &-a_1 \ \ldots\ -a_{n_a}\ ;\ b_0^1 \ldots b_{n_b}^1\ ;\ b_0^2 \ldots b_{n_b}^2\ ;\ \ldots\ ;\ b_0^r \ldots b_{n_b}^r\ ;\\ &\eta_{01}^1 \ldots \eta_{n_b1}^1\ ;\ \ldots\ ;\ \eta_{0m}^1 \ldots \eta_{n_bm}^1\ ;\ \ldots\ ;\\ &\ldots\ ;\ \ldots\ ;\ \eta_{01}^r \ldots \eta_{n_b1}^r\ ;\ \ldots\ ;\ \eta_{0m}^r \ldots \eta_{n_bm}^r\ ;\ ]\end{aligned} \qquad (3.28)$$

and

$$\begin{aligned}\boldsymbol{x}^{T}(t) = [\ &y(t-1) \ldots y(t-n_a)\ ;\ u_1(t-k) \ldots u_1(t-k-n_b)\ ;\ \ldots\ ;\ u_r(t-k) \ldots u_r(t-k-n_b)\ ;\\ &\hat{x}(t-k)u_1(t-k) \ldots \hat{x}(t-k-n_b)u_1(t-k-n_b)\ ;\ \ldots\ ;\\ &\hat{x}(t-k)u_1(t-k-m+1) \ldots \hat{x}(t-k-n_b)u_1(t-k-n_b-m+1)\ ;\\ &\ldots\ ;\\ &\hat{x}(t-k)u_r(t-k) \ldots \hat{x}(t-k-n_b)u_r(t-k-n_b)\ ;\ \ldots\ ;\\ &\hat{x}(t-k)u_r(t-k-m+1) \ldots \hat{x}(t-k-n_b)u_r(t-k-n_b-m+1)\ ]\end{aligned} \qquad (3.29)$$

The standard RLS algorithm given by equation (3.11)-(3.13) is then adopted in order to estimate the $[n_a+r(m+1)(n_b+1)]$ model parameters.

3.5.3 State estimation

Extending the approach outlined in section 3.3.3, define the quasi-linear state-space representation $\overline{\text{SBL}}(n,m,r)$ of the state-space representation (3.23) by

$$\mathbf{x}(t+1) = \bar{\mathbf{P}}(\tilde{\mathbf{u}}(t))\mathbf{x}(t) + \bar{\mathbf{Q}}(\mathbf{x}(t))\mathbf{u}(t) + \mathbf{R}\mathbf{e}(t) \tag{3.30a}$$

$$\mathbf{y}(t) = \mathbf{H}\mathbf{x}(t) + \mathbf{e}(t) \tag{3.30b}$$

where

$$\tilde{\mathbf{u}}(t) = \left[u_1(t-1) \ldots u_1(t-m+1) \; ; \; u_2(t-1) \ldots u_2(t-m+1) \; ; \ldots ; \; u_r(t-1) \ldots u_r(t-m+1) \; ; \right]^T, \tag{3.31}$$

and

$$\bar{\mathbf{P}}(\tilde{\mathbf{u}}(t)) = \mathbf{P}, \qquad m = 1,$$

$$= \mathbf{P} + \sum_{i=2}^{m} \sum_{j=1}^{r} u_j(t-i+1)\mathbf{N}_i^j, \qquad m \geq 2, \tag{3.32}$$

with

$$\bar{\mathbf{Q}}(\mathbf{x}(t)) = \left[\mathbf{b}_1 + \mathbf{N}_1^1\mathbf{x}(t) : \mathbf{b}_2 + \mathbf{N}_1^2\mathbf{x}(t) : \ldots : \mathbf{b}_r + \mathbf{N}_1^r\mathbf{x}(t) \right] \tag{3.33}$$

where $\mathbf{b}_j$, j=1,2, . . r, denotes the j^{th} column of **Q**.

Substituting the output equation (3.30b) into the state equation (3.30a), rearranging and setting $\bar{\mathbf{P}}_1(\tilde{\mathbf{u}}(t))=\bar{\mathbf{P}}(\tilde{\mathbf{u}}(t))-\mathbf{RH}$ leads to

$$\hat{\mathbf{x}}(t) = [\, \mathbf{I} - q^{-1}\bar{\mathbf{P}}_1(\tilde{\mathbf{u}}(t)) \,]^{-1} \, [\bar{\mathbf{Q}}(\hat{\mathbf{x}}(t))\mathbf{u}(t-1) + \mathbf{R}\mathbf{y}(t-1)] \tag{3.34}$$

which is an extended form of the ISO given by equation (3.19).

Note that the error dynamics are now dependent on the eigenvalues of the matrix

$$\bar{\mathbf{P}}_1(\tilde{\mathbf{u}}(t)) = \mathbf{P} - \mathbf{RH} + \sum_{j=1}^{r} \mathbf{N}_2^j u_j(t-1) + \sum_{j=1}^{r} \mathbf{N}_3^j u_j(t-2) + \ldots + \sum_{j=1}^{r} \mathbf{N}_m^j u_j(t-m+1)$$

$$= \mathbf{P} - \mathbf{RH} + \sum_{i=2}^{m} \sum_{j=1}^{r} \mathbf{N}_i^j u_j(t-i+1). \tag{3.35}$$

Again, as in the case of the SISO system, the error dynamics are a function of both the

past values of the control input sequences and the estimated values of the bilinear coefficients. Since the ISO performs effectively for SISO bilinear systems it is pertinent to consider its effectiveness for MISO bilinear systems and is adopted here.

3.5.4 MISO control law synthesis

A realistic approach to pole-placement in the linear multivariable case is to make use of the dyadic control [22]. This effectively involves determining an (rxn) state variable feedback gain matrix **F** which is constrained to have unity rank by defining it in the dyadic form

$$\mathbf{F} = \mathbf{f}_1\mathbf{f}_2^{\mathrm{T}}. \tag{3.36}$$

The vector $\mathbf{f}_1 \in \mathbb{R}^r$ is arbitrarily specified by the designer and then $\mathbf{f}_2 \in \mathbb{R}^n$ is computed from

$$\mathbf{f}_2 = \mathbf{W}^{-1}\left[\, 0 \ldots 0 \; (\gamma_{n_a} - a_{n_a}) \ldots (\gamma_1 - a_1) \,\right]^{\mathrm{T}} \tag{3.37}$$

where γ_i, i=1,2, . . n_a, are the coefficients of the desired closed-loop pole-polynomial and **W** is determined by

$$\mathbf{W} = \mathbf{KL}$$

with **K** being the Kalman controllability test matrix for the effective single input problem

$$\mathbf{K} = \left[\mathbf{P}^{n-1}\mathbf{Q}\mathbf{f}_1 : \mathbf{P}^{n-2}\mathbf{Q}\mathbf{f}_1 : \ldots : \mathbf{P}\mathbf{Q}\mathbf{f}_1 : \mathbf{Q}\mathbf{f}_1 \right] \tag{3.38}$$

and **L** is the lower triangular matrix as previously defined in Chapter 2.

3.5.5 Preliminary investigations

In adopting *Algorithm 2* as the basis of the bilinear STC, use is made of the equivalent MISO quasi-linear ISO, equation (3.34) and the effective linear multivariable feedback matrix of equation (3.36).

In [A13], simulation studies are undertaken using a simple SBL(2,2,2) model structure. For convenience the vector $\mathbf{f}_1$ is chosen such that $\mathbf{f}_1^T$=[1 1] and the investigation compared the effectiveness of fixed forgetting factors, variable forgetting factors, variable forgetting factors combined with covariance reset and variable forgetting factors combined with cautious covariance reset . The results once again indicate the potential vulnerability of the bilinear approach to numerical instability and highlight the need for more robust parameter estimation procedures. In particular it must be noted that in reality, the problem could be further compounded by the accompanying effects of computational delay, particularly for MISO systems.

It is noted [22], for linear systems, that although the closed-loop system will exhibit the desired characteristics in terms of overall closed-loop pole locations, different choices of $\mathbf{f}_1$ will give rise to distinctive controller action. In practice the elements of $\mathbf{f}_1$ may be regarded as tuning knobs allowing an element of trade-off in terms of proportioning the controller action; the latter feature being advantageous when attempting to avoid saturating input signals.

The MISO approach has been successfully applied to a laboratory scale heater-cooling system (introduced in Chapter 5) which is known from physical considerations to exhibit bilinear characteristics. It has been shown that by carefully tailoring the elements of $\mathbf{f}_1$ that improvements can be achieved over the use of the SISO approach, with the MISO STC providing the designer with the facility for weighting or proportioning the control effort in each input channel so as to produce an overall improved performance.

3.6 Concluding remarks

By taking into account the non-linearity or bilinearity at the design stage, the standard linear self-tuning framework has been extended to accommodate a class of discrete SISO and MISO bilinear systems.

Investigative studies involving four initial candidates for extended pole-placement algorithms have been undertaken and, based on extensive simulation trials involving a number of bilinear system models subjected to a range of different test conditions, one such algorithm has been proposed and this forms the basis for further studies.

Whilst preliminary results obtained when applying the proposed bilinear STC to both SISO and MISO systems have been encouraging, it is believed that in order to provide for a more effective self-tuning scheme, enhanced estimation techniques are required. These are to be robust enough to minimise the susceptibility of the algorithm to numerical instability without compromising the desirable features of adaptivity and without increasing significantly the complexity.

4. Enhanced parameter estimation techniques for bilinear STC

4.1 Preliminaries

The need for increased integrity within STC schemes is of paramount importance, particularly when dealing with non-linear systems. In an attempt to improve the integrity of the bilinear STC introduced in Chapter 3 a number of enhanced parameter estimation techniques have been specifically investigated/developed for a class of SISO discrete bilinear systems. The results may readily be extended to the MISO case. It should be noted, however, that all enhanced techniques are accompanied by an increase in computational overhead such that the full advantages of adopting the enhanced techniques may never in practice be realised. As such, a compromise situation exists in which a trade-off must be made between increased integrity arising from the use of enhanced techniques and reduced overall closed-loop performance due to the effects of increased computational complexity and it is against this background that investigative studies have been based.

The enhanced estimation techniques considered here include: fixed and variable forgetting factors [17,18,A6] for increased adaptivity whilst reducing the possibility of numerical instability arising from covariance blow-up [16]; covariance matrix resetting techniques [52,53,54,A9] for improved alertness and numerical stability; instrumental variable techniques [55,56] for improved accuracy in the presence of coloured noise; Kalman filtering techniques [57,58,59,A14,A15] for increased tracking ability; and cautious least squares [60,A10,A11,A12,A16] for increased robustness. Whilst the techniques are shown to be advantageous when applied to bilinear systems, they are equally appropriate for linear systems. Indeed the techniques are considered to be highly appropriate when dealing with systems exhibiting non-linear characteristics and linear/bilinear self-tuning techniques are to be employed.

4.2 Forgetting factors

4.2.1 Fixed forgetting factor

When parameters to be estimated are varying with time, the recursive parameter estimation procedure may be modified to produce an 'adaptive' least squares scheme; this being achieved by introducing a forgetting factor [15] as indicated in Chapter 2. The resulting recursive form of the adaptive least squares algorithm is then given by

$$\hat{\theta}(t) = \hat{\theta}(t\text{-}1) + \phi(t)[\, y(t) - x^T(t)\hat{\theta}(t\text{-}1)\,] \tag{4.1}$$

$$\phi(t) = \Phi(t\text{-}1)x(t)[\, 1 + x^T(t)\Phi(t\text{-}1)x(t)]^{-1} \tag{4.2}$$

$$\Phi(t) = [\, I - \phi(t)x^T(t)\,]\Phi(t\text{-}1) / \lambda \tag{4.3}$$

in which the scalar $\lambda \leq 1.0$ is the forgetting factor. (See Appendix 7.)

Use of a forgetting factor enables greater emphasis to be placed on the more recent observations and gives rise to a fading memory length which is given approximately by

$$M \simeq (1-\lambda)^{-1}. \tag{4.4}$$

Clearly, for a least squares solution the value of M must be such that $M \gg p$ where $p = n_a + n_b + 1$ is the number of parameters to be identified and the value of the forgetting factor must be bounded from below such that $\lambda_\ell < \lambda \leq 1.0$; typically $\lambda_\ell = 0.9 - 0.95$. Note that when $\lambda = 1$, all observations are given equal weighting and the algorithm is unable to adequately adapt.

A problem for the user lies in the choice for a suitable value for λ. The lower its value, the more adaptive in nature the procedure becomes, unfortunately however, this also renders the algorithm highly sensitive to external disturbances such as measurement noise. Consequently, the margin for error is small with poor choice of λ

leading possibly to a correspondingly poor control. A summary of the features which affect the choice of forgetting factor are given in [A2].

A major problem arising in the use of a fixed forgetting factor is that of covariance blow-up and possible instability [16]. This phenomenon occurs when the system is operating in steady-state: The STC has driven the system to its destination and there is now little or no new information coming in from the plant; the forgetting factor λ is less than unity and the information content within the algorithm is therefore reducing. A reduction in the information content of the algorithm leads to the information matrix (inverse of the covariance matrix) becoming 'near singular', giving rise to a corresponding growth in the elements of the covariance matrix. This condition may well result in instability, however, it may often be self-correcting giving a 'near blow-up' or 'near instability' situation and this can generate sufficient control input to the system to remedy the lack of information content from the plant. In either case when dealing with bilinear systems, which are known to be particularly sensitive, the latter is a serious problem which can be alleviated by using a variable forgetting factor.

4.2.2 Variable forgetting factors

Use of a variable or regulated forgetting factor $\lambda(t)$ will give rise to increased adaptivity of the STC scheme whilst also preventing the possibility of instability or near instability arising due to covariance blow-up. This is effectively achieved by regulating the information content of the STC algorithm. Whilst various variable forgetting factors (VFF) have been proposed for linear systems [17,18] it is pertinent to consider their use when dealing with bilinear systems. As a result, a hybrid of those proposed in [17,18] has been found to be advantageous for bilinear systems [A6] and has also found favour when dealing with other non-linear systems [A16,A17]. Essentially, the mechanism for varying the value of the forgetting factor is based on the estimation prediction error within the

estimator; this being regarded as an indicator of the 'accuracy' of the algorithm in the presence of parameter variation.

Method 1: Fortescue et al [17].

This approach is based on the early theoretical work of Albert and Sittler, 1966 [61]. As a means of ensuring a consistent information content within the algorithm, the VFF is regulated such that a weighted sum of past estimation prediction errors is minimised. It takes the form

$$\lambda(t) = 1 - [\, 1 - \boldsymbol{x}^T(t\text{-}k\text{-}1)\boldsymbol{\Phi}(t\text{-}k)\boldsymbol{x}(t\text{-}k\text{-}1)\,]\, \Sigma^2(t\text{-}1) / \Sigma_0 \qquad (4.5)$$

where

$$\Sigma(t) = y(t) - \boldsymbol{x}^T(t)\hat{\boldsymbol{\theta}}(t\text{-}1) \qquad (4.6)$$

is the estimation prediction error and

$$\Sigma_0 = \sigma_0^2 M_o, \qquad (4.7)$$

in which σ_0^2 is the expected measurement noise variance and M_o is the nominal memory length of the estimator.

■

Method 2: Wellstead and Sanoff [18].

This is essentially a heuristic approach and includes an exponential delay term in its formulation. It takes the form

$$\lambda(t) = \lambda_1(t).\lambda_2(t) \qquad (4.8)$$

where

$$\lambda_1(t) = \lambda_0 + (1-\lambda_0)(1-e^{-t/M_0}) \qquad (4.9)$$

$$\lambda_2(t) = 1 - \Sigma^2(t\text{-}1) / s(t) \qquad (4.10)$$

$$s(t) = (M_0-1)s(t\text{-}1) + \Sigma^2(t\text{-}1) \qquad (4.11)$$

where λ_0 is the initial value of the forgetting factor which from equation (4.4), $\lambda_0=(M_0-1)/M_0$.

■

Hybrid method 3: [A6]

This method attempts to exploit the desirable features of the two previously existing methods and takes the form

$$\lambda(t) = \lambda_a(t).\lambda_b(t) \qquad (4.12)$$

where $\lambda_a(t) = \lambda(t)$ from Method 1,

$\lambda_b(t) = \lambda_1(t)$ from Method 2.

■

4.2.3 Simulation studies

With consideration being given to the bilinear system

$$y(t) = u(t-1) + 0.5u(t-2) - a_1(t)y(t-1) - 0.7y(t-2) + 0.25x_n(t-1)u(t-1) + e(t),$$

in which the parameter a_1 is allowed to be time varying, the three forms of VFF are compared. In each case, the simulation is run over 200 iterations with the control objective being to regulate the system output about a zero reference when closed-loop poles are effectively specified at -3 and -5 in the s-plane with use made of a sampling interval of T=0.1 second.

The three methods for obtaining the VFF defined by equations (4.5), (4.8) and (4.12) are investigated when the parameter a_1 is varied as follows: Fixed at -1.5 for $0<t\leq 40$, decreasing linearly to a value of -2.0 over $40<t\leq 120$, followed by a step change back to -1.5 when t=120 then held constant. In each case an initial value of λ_0=0.925 is used and the methods are compared when subject to a range of noise levels and nominal estimator memory lengths.

Figures 4.1(a)(b)(c), 4.2(a)(b)(c) and 4.3(a)(b)(c), illustrate (a) the value of the variable forgetting factor, (b) the system response and (c) the control input corresponding to use of the bilinear STC when use is made of Method 1, Method 2 and

the hybrid method of obtaining the variable forgetting factor respectively, with noise level e(t) bounded such that $\left| e(t) \right| \leq 0.05$ and the nominal memory length M_0=60.

Whilst detailed discussion is given in [A6], the results indicate that, when applied to the simulated model of a time varying bilinear system, use of the hybrid VFF is found to give rise to a superior system performance in terms of its ability to maintain system stability, providing for a more robust algorithm in term of user choice of estimator memory length.

4.3 Resetting techniques

4.3.1 Covariance matrix reset

Whilst variable forgetting factors are found to provide effective algorithms for the tracking of slowly varying parameters, their use in tracking sudden changes is limited by the need to retain a sufficient memory length. An alternative, albeit ad hoc, approach which facilitates fast adaptation (or re-adaption) of the estimation algorithm is that of covariance matrix reset. Covariance reset has the effect of removing all *a priori* knowledge from within the algorithm, and is achieved by resetting the covariance matrix $\Phi(t)$ in equation (4.3) to $\mu\mathbf{I}$, where **I** is the identity matrix and μ is a user defined large positive scalar. The larger the value of μ the 'harder' the reset with more of the older information being discarded. (Typically μ is taken to be of the order 100.) Reset action can be enforced at regular intervals [52,53], or triggered on fault detection [54,A8,A9]; with the fault condition being an indicator of the accuracy of the estimator.

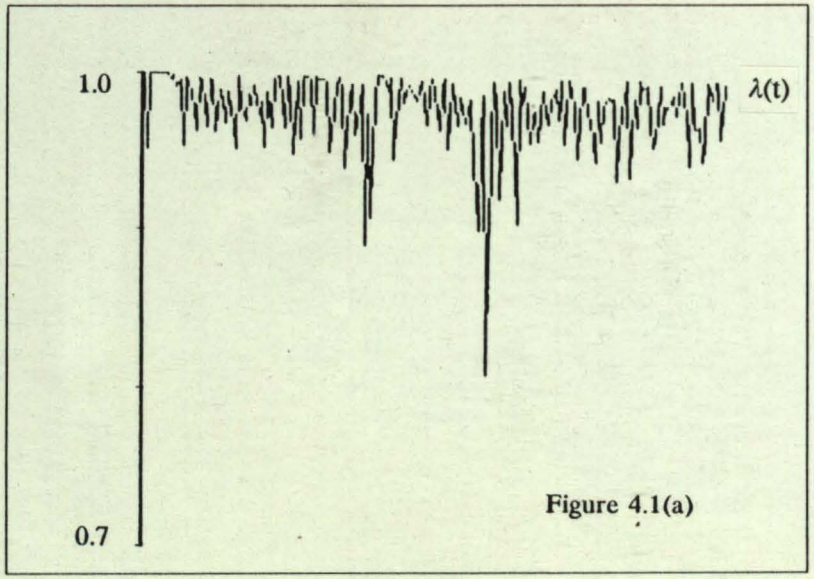


Figure 4.1(a)

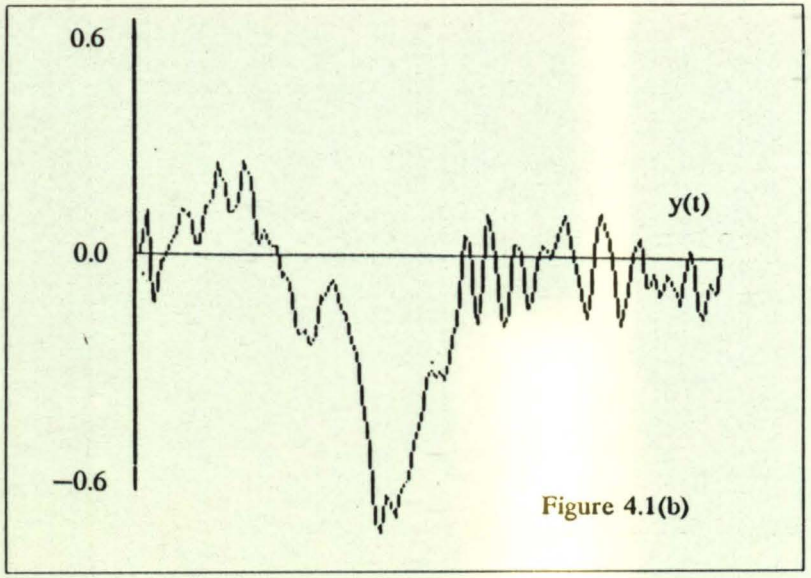


Figure 4.1(b)

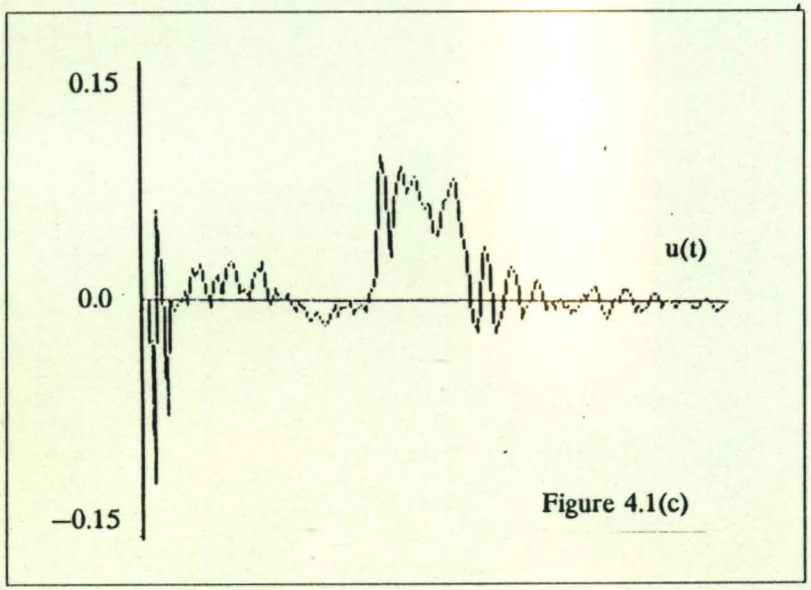


Figure 4.1(c)

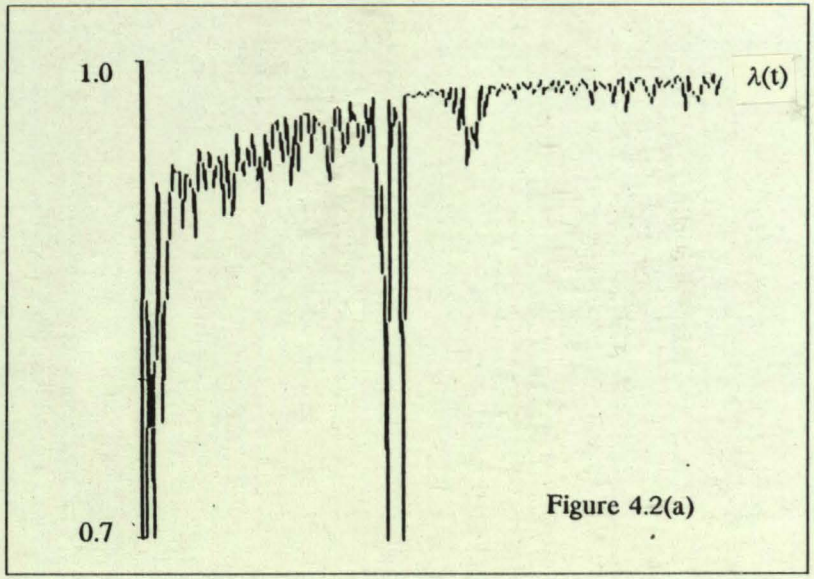


Figure 4.2(a)

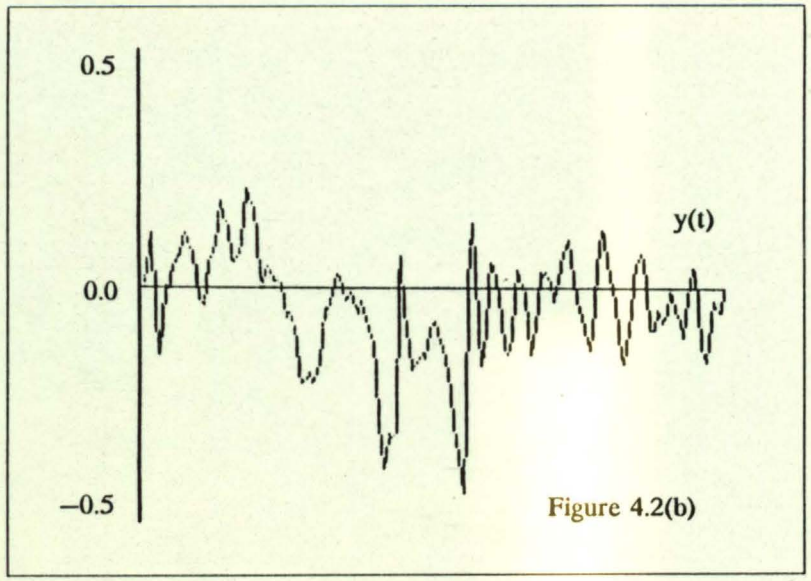


Figure 4.2(b)

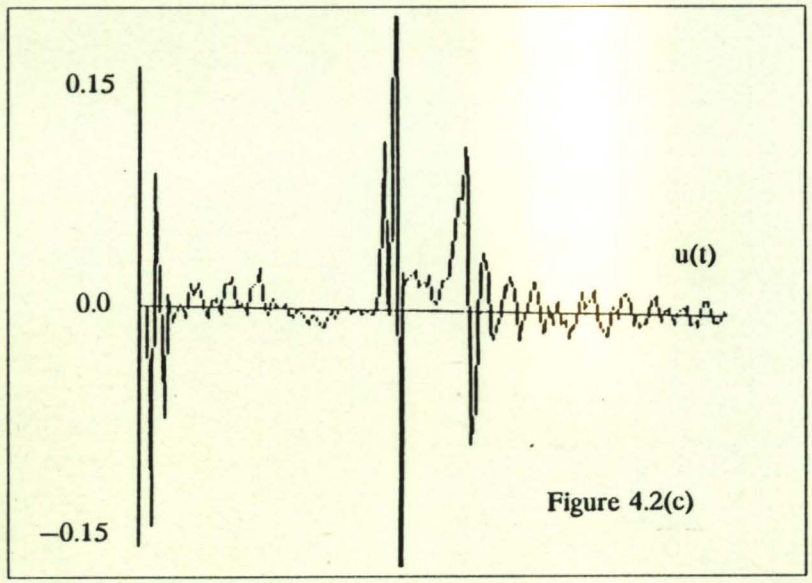


Figure 4.2(c)

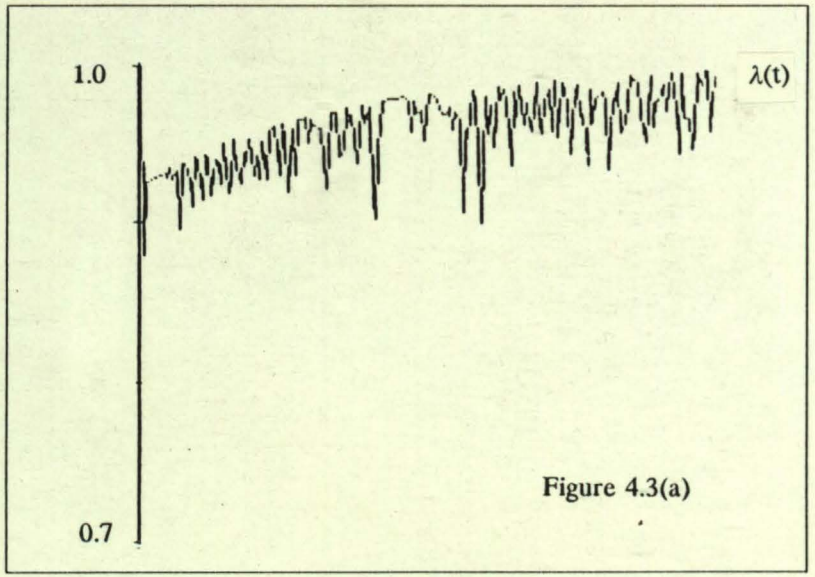


Figure 4.3(a)

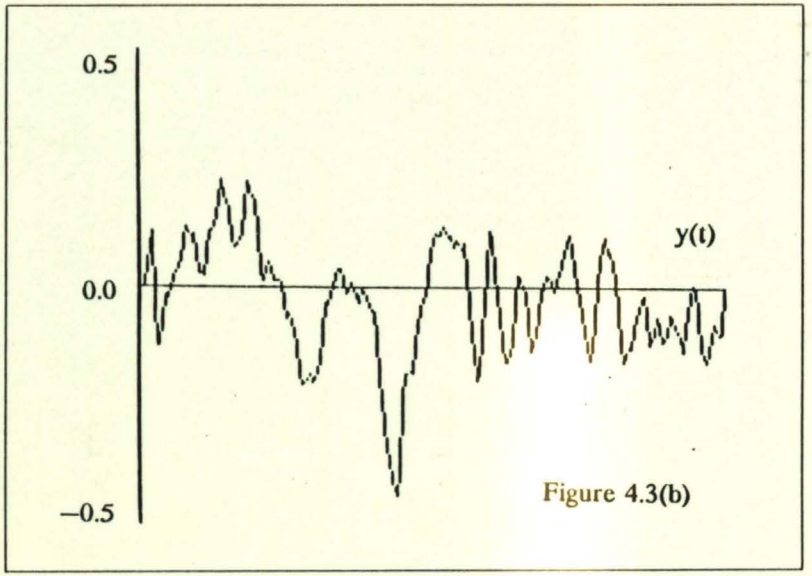


Figure 4.3(b)

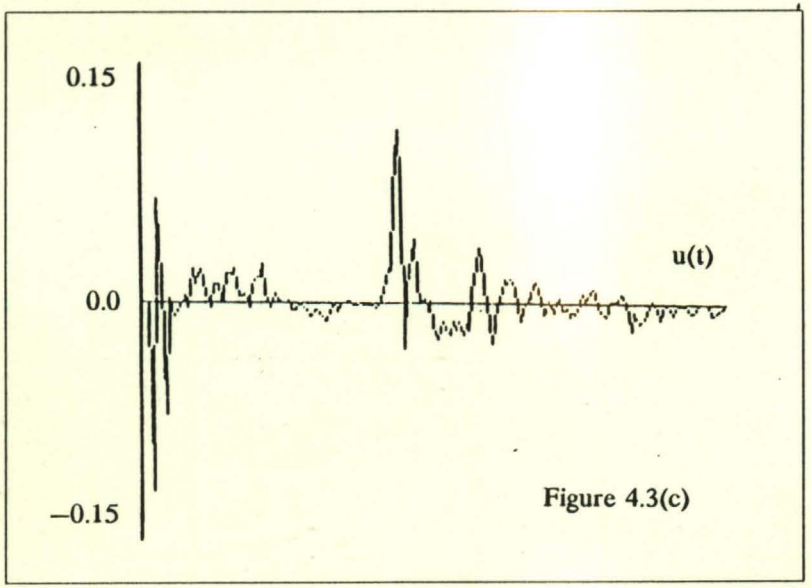


Figure 4.3(c)

4.3.2 Combined covariance matrix and VFF reset techniques

By combining the desirable features of covariance matrix reset together with reset applied to the exponential decay term of the hybrid VFF of equation (4.12) an algorithm has been developed which can handle both slow and sudden parameter variation. Investigative studies involving the application of such a two-tier approach [A8,A9] to the simulated model of a bilinear system have indicated that a markedly improved performance is possible over that achieved when use is made of VFF and covariance reset in isolation. In [A8,A9] reset action is triggered using a fault detection mechanism which is indicative of large estimation errors. The mechanism, which is similar to that proposed in [54] for linear systems, triggers the reset action whenever the demand for a VFF falls below some pre-specified threshold level λ_ρ. (Note that the actual value of the VFF is bounded from below such that $\lambda(t) > \lambda_\ell > \lambda_\rho$.)

4.3.3 Simulation studies

In order to compare the effectiveness of the combined two-tier algorithm, the bilinear system which had previously been investigated by Gabr and Subba Rao [41], in a different context, is adopted. This takes the form

$$y(t) = -a_1(t)y(t-1) - 0.7y(t-2) + u(t-1) + 0.5u(t-2) + 0.12x(t-1)u(t-1) + e(t)$$

As in section 4.2.3 the objective in this case is to place closed-loop poles at -3 and -5 in the s-plane and with a sample interval of T=0.1 second the system was subject to a step input of r(t)=0.05. There is no attempt to make adjustment to the closed-loop steady-state gain of the system. With the noise level e(t) chosen such that $\left| e(t) \right| \leq 0.01$ in each case, the techniques are compared both when combined and in isolation with each simulation being run over 750 iterations.

It is shown [A8, A9] that the use of the combined two-tier approach leads to

a marked improvement over that previously attained using VFF and covariance reset in isolation.

As a test condition, the system is subjected to a step input r(t)=0.05 and an output disturbance e(t) represented by discrete white noise, where $|e(t)| \leq 0.04$. The parameter $a_1(t)$ is varied according to

$$a_1(t) = \begin{cases} -1.5 & t \leq 50 \\ -1.5-(t-50)/1200 & 50 < t \leq 650 \\ -1.5 & t > 650 \end{cases}$$

with each simulation being run over 750 iterations.

Figures 4.4(a) and 4.4(b) illustrate, respectively, system response and control input when use is made of the combined VFF and covariance matrix reset scheme. By combining the desirable features of the hybrid VFF with the covariance resetting approach the resulting algorithm is able to handle both slow and sudden parameter variation and the resulting mechanism is believed to be applicable to a wide range of non-linear and/or time-varying systems which may be required to be controlled using STC schemes.

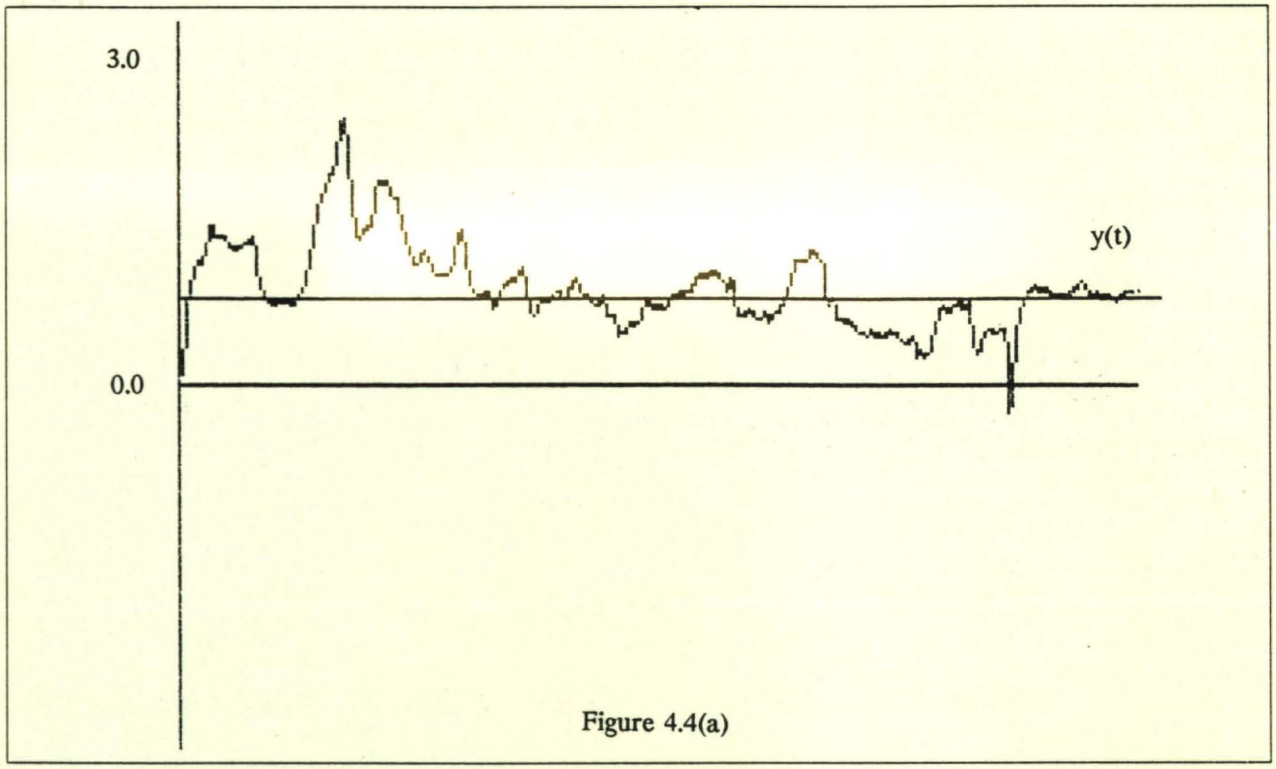


Figure 4.4(a)

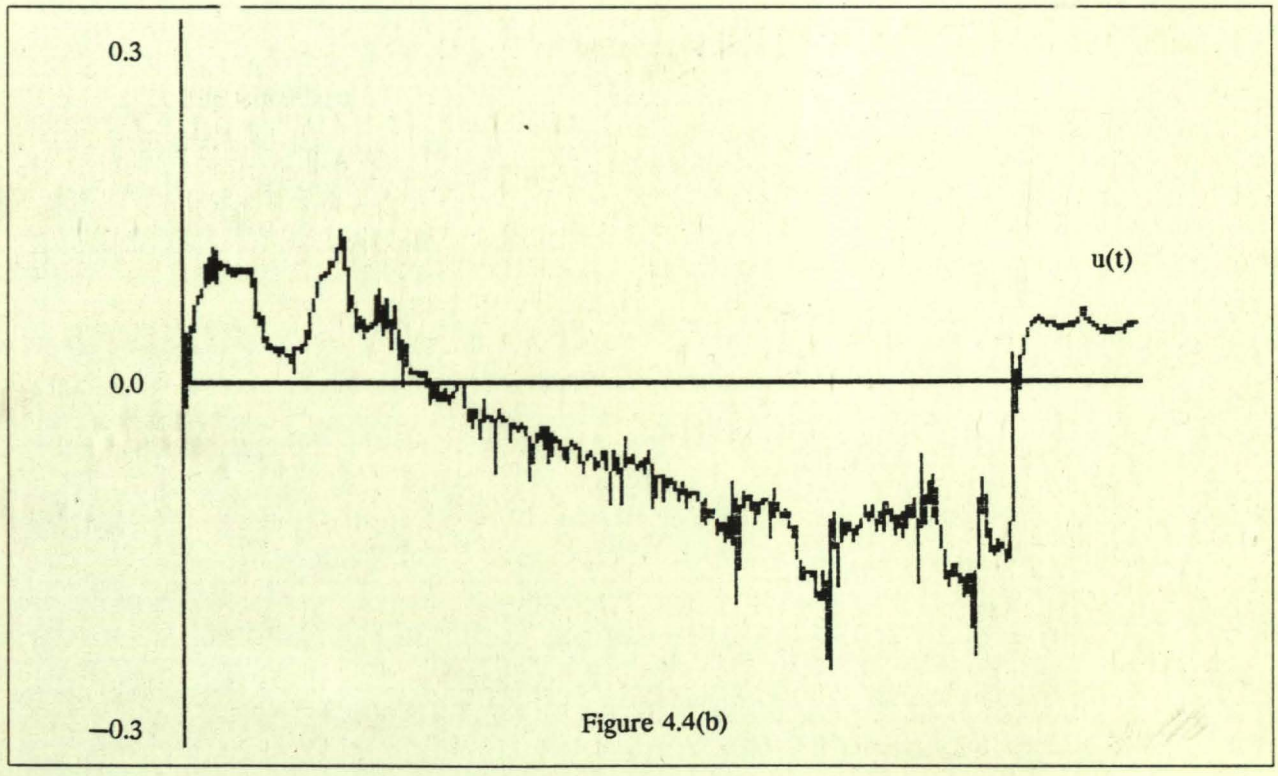


Figure 4.4(b)

4.4 The method of instrumental variables

4.4.1 Instrumental variables (linear case)

The method of instrumental variables (IV) is based on the concept of a multi-pass RLS algorithm and has been found to be advantageous for linear systems when the noise is non-white [55,56]. The instrumental variables are the (noise free) generated outputs z(t) obtained when the identified model is subjected to the same input excitation signal as the actual system.

Whilst there appears to be no unique approach for generating the instrumental variables [56], the simplest form of the recursive algorithm is to make use of the prediction of the system output obtained directly from the estimation prediction error of equation (4.1) of the RLS algorithm i.e. $z(t)=\boldsymbol{x}^T(t)\hat{\boldsymbol{\theta}}(t-1)$. The recursive instrumental variables (RIV) algorithm takes the form

$$\hat{\boldsymbol{\theta}}(t) = \hat{\boldsymbol{\theta}}(t-1) + \boldsymbol{\phi}(t)[\, y(t) - \boldsymbol{x}^T(t)\hat{\boldsymbol{\theta}}(t-1)\,] \tag{4.13}$$

where $$\boldsymbol{\phi}(t) = \boldsymbol{\Phi}(t-1)\bar{\boldsymbol{x}}(t)\,[\; 1 + \boldsymbol{x}^T(t)\boldsymbol{\Phi}(t-1)\bar{\boldsymbol{x}}(t)\,]^{-1} \tag{4.14}$$

and $$\boldsymbol{\Phi}(t) = [\, \mathbf{I} - \boldsymbol{\phi}(t)\boldsymbol{x}^T(t)\,]\,\boldsymbol{\Phi}(t-1) \,/\, \lambda(t) \tag{4.15}$$

where $\boldsymbol{x}(t)$ is the observation vector as defined in Chapter 2

$$\boldsymbol{x}^T(t) = [\, y(t-1)\; y(t-2)\; \ldots\; y(t-n_a);\; u(t-k)\; u(t-k-1)\; \ldots\; u(t-k-n_b)\,], \tag{4.16a}$$

$\bar{\boldsymbol{x}}(t)$ is the vector consisting of the instrumental variables

$$\bar{\boldsymbol{x}}^T(t) = [\, z(t-1)\; z(t-2)\; \ldots\; z(t-n_a);\; u(t-k)\; u(t-k-1)\; \ldots\; u(t-k-n_b)\,] \tag{4.16b}$$

and $\boldsymbol{\Phi}(t)$ is the corresponding error covariance matrix.

4.4.2 Instrumental variables (bilinear case)

When considering bilinear systems, the RIV algorithm is extended in a similar manner to the RLS algorithm as outlined in Chapter 3. However, a number of interesting possibilities arise. Recall that in the case of bilinear systems, the adopted form of the observation vector (3.10) is given by

$$\begin{aligned} \boldsymbol{x}^T(t) = [\ & y(t-1) \ldots y(t-n_a);\ u(t-k) \ldots u(t-k-n_b);\\ & \hat{x}_n(t-k)u(t-k) \ldots \hat{x}_n(t-k-n_b)u(t-k-n_b);\ldots\ ;\\ & \hat{x}_n(t-k)u(t-k-m+1) \ldots \hat{x}_n(t-k-n_b)u(t-k-n_b-m+1)\ ] \end{aligned} \tag{4.17a}$$

so that the equivalent vector of instrumental variables becomes

$$\begin{aligned} \bar{\boldsymbol{x}}^T(t) = [\ & z(t-1) \ldots z(t-n_a);\ u(t-k) \ldots u(t-k-n_b);\\ & \hat{x}_n(t-k)u(t-k) \ldots \hat{x}_n(t-k-n_b)u(t-k-n_b);\ldots\ ;\\ & \hat{x}_n(t-k)u(t-k-m+1) \ldots \hat{x}_n(t-k-n_b)u(t-k-n_b-m+1)\ ] \end{aligned} \tag{4.17b}$$

However, note that the instrumental variables are already available in the form of past values of the noise free filtered estimates $\hat{x}_n(t-i)$ obtained from the iterated steady-state observer (ISO), equation (3.19), so that an alternative form for the vector of instrumental variables becomes

$$\begin{aligned} \bar{\boldsymbol{x}}^T(t) = [\ & \hat{x}_n(t-1) \ldots \hat{x}_n(t-n_a);\ u(t-k) \ldots u(t-k-n_b);\\ & \hat{x}_n(t-k)u(t-k) \ldots \hat{x}_n(t-k-n_b)u(t-k-n_b);\ldots\ ;\\ & \hat{x}_n(t-k)u(t-k-m+1) \ldots \hat{x}_n(t-k-n_b)u(t-k-n_b-m+1)\ ] \end{aligned} \tag{4.17c}$$

The distinction between the instrumental variables z(t-i) obtained from RLS and the noise free filtered estimates $\hat{x}_n(t-i)$ obtained from the ISO is that whereas the z(t-i) are based on $\hat{\boldsymbol{\theta}}(t-i-1)$, the $\hat{x}_n(t-i)$ are based on the most recent estimates $\hat{\boldsymbol{\theta}}(t-i)$.

The extended RIV algorithm for bilinear systems takes the form of equations (4.13)-(4.15) with $\bar{\boldsymbol{x}}(t)$ being of the form (4.17b) or (4.17c).

4.4.3 Simulation studies

In order to assess the effectiveness of the extended forms of the RIV algorithm for bilinear systems, the performance arising, in terms of minimum estimation prediction error, when use is made of the vectors of instrumental variables, given by (4.17b) and (4.17c), are compared. The bilinear system described in section 4.3 is subjected to a range of white and non-white noise levels when driven in open-loop by a pseudo random input sequence. At each time step the total absolute error between actual parameter values and estimated parameter values is formed. It is found that use of the ISO, corresponding to the vector of instrumental variables given by equation (4.17c), leads to a faster convergence than when use is made of the vector given by equation (4.17b). However, in the steady-state, both methods are found to produce virtually identical estimation errors. Typical plots of estimation errors, indicating convergence behaviour, are given in Figures 4.5(a) and 4.5(b) which correspond to a step change in a single parameter (at the 100^{th} iteration) and a ramp change in a single parameter (commencing at the 100^{th} iteration) respectively. In each case, the solid line corresponds to the use of (4.17b) and the dashed line corresponds to the use of (4.17c). In both cases, a fixed forgetting factor λ=0.99 is employed and the simulation run over 200 iterations. It is evident from these Figures that both approaches are able to recover from these disturbances. However, due to the faster initial convergence when use is made of equation (4.17c), the approach in which the IV's are generated from the ISO is the more favourable. This approach is believed to be particularly appropriate when used in conjunction with the covariance resetting techniques outlined in section 4.3, where the estimation algorithm may be required to be repeatedly re-initialised.

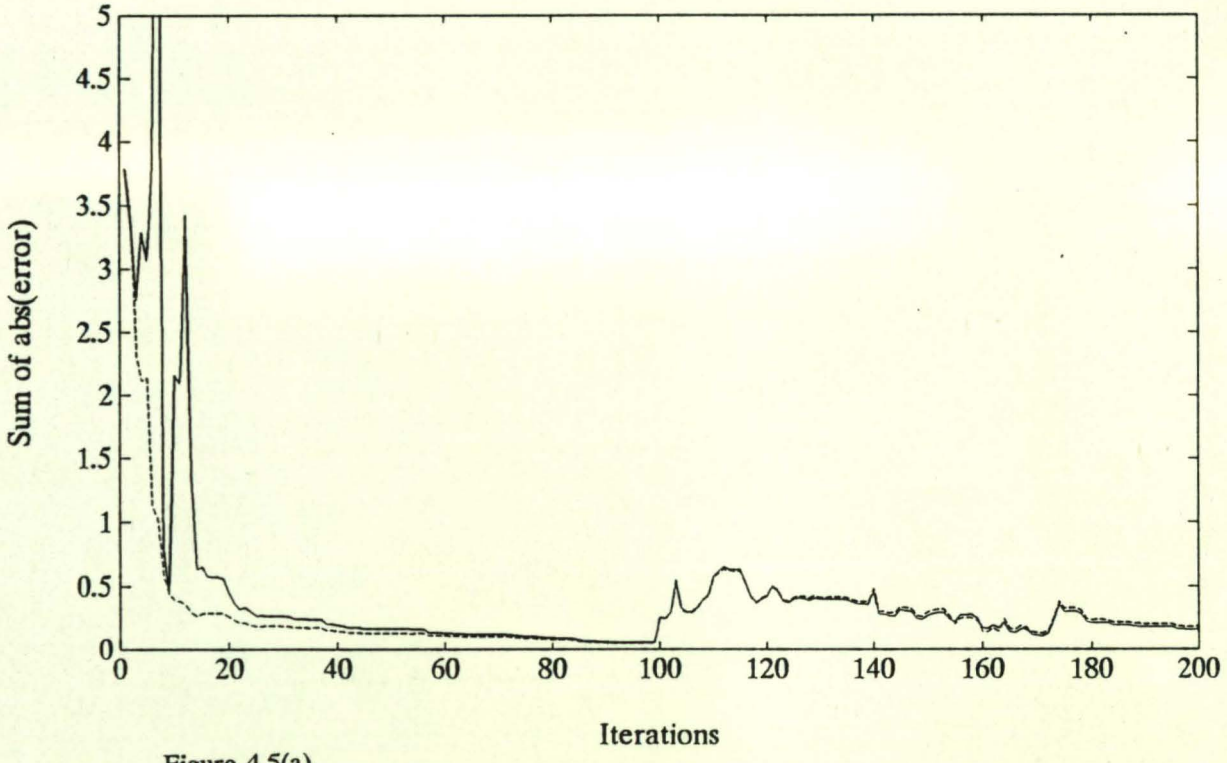


Figure 4.5(a)

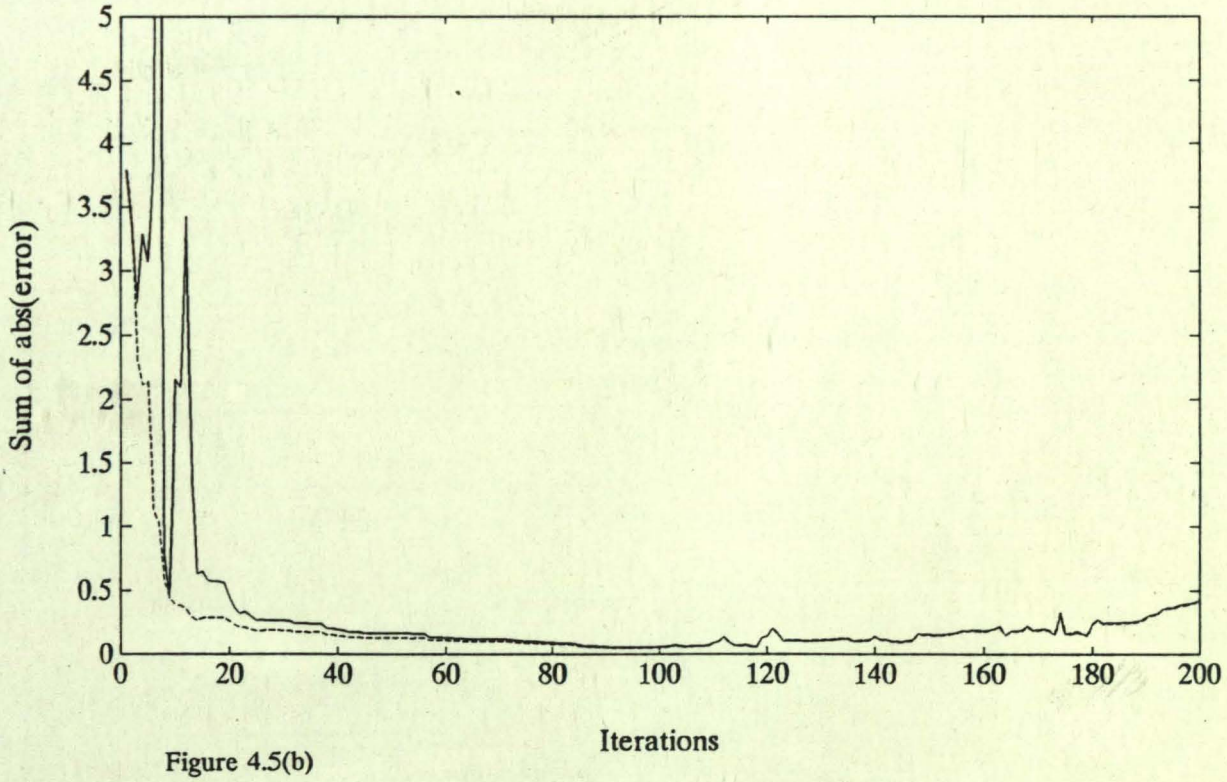


Figure 4.5(b)

4.5 Kalman filtering techniques

Based on the RLS algorithm developed in Chapter 2, the Kalman filter (KF) for linear state estimation and the extended Kalman filter (EKF) for non-linear state estimation are developed in Appendices 8 and 9 respectively. For brevity and to maintain continuity, consideration is focused here to the problem of joint state and parameter estimation for bilinear systems. Via a simple transformation the KF is re-formulated for parameter estimation and the EKF is re-formulated for both state and parameter estimation (see Appendices 8 and 9 respectively). The applicability of the EKF is compared to the tandem estimation schemes, introduced in Chapters 2 and 3, when applied to the simulated models of both a linear and a bilinear system.

4.5.1 Brief historical note

The KF is a well known and versatile tool in the field of estimation theory. Originally proposed for linear state estimation [57,62], the KF has been widely used in a diversity of practical applications and has also provided a wealth of theoretical research.

Essentially, Kalman [62] developed a general framework for the RLS algorithm with the important distinction being that the parameters to be identified were no longer constants (as in the work of Plackett [8]), but were time varying quantities governed by linear differential (or difference) equations. The KF can handle both measurement noise on the system output and process noise in the system dynamics and is the optimal estimator, in the sense of least squares, when such noises are Gaussian in nature.

The work of Kalman, and later Kalman and Bucy [63], has resulted in estimation techniques that are both elegant and powerful and can cater for both linear

and non-linear systems; the latter being achieved using the EKF. The filter in its various forms has been widely used for more than three decades and shows no sign of losing its prominence as a successful and reliable engineering tool.

4.5.2 Kalman filter for parameter estimation

In this approach, the RLS parameter estimation procedure, used in the tandem RLS/SKF scheme for the linear STC introduced in Chapter 2, is replaced by a linear KF to produce a tandem KF/SKF scheme. In principle this allows incorporation of *a priori* engineering knowledge without significantly increasing the computational overhead [64].

For the purpose of the KF, the evolution of the parameter vector $\boldsymbol{\theta}$ is described by

$$\boldsymbol{\theta}(t+1) = \boldsymbol{\theta}(t) + \boldsymbol{\omega}(t) \tag{4.18}$$

where $\boldsymbol{\omega}(t)$ is a noise sequence which takes into account any likely variation in the parameter values with time, i.e. $\boldsymbol{\theta}$ is principally time invariant, but includes a random component to reflect any possible variations.

The KF generates estimates $\hat{\boldsymbol{\theta}}(t|t)$ via a simple two stage prediction/correction process; prediction takes place during the sampling intervals, followed by correction at the sampling instants. (The notation $\hat{\boldsymbol{\theta}}(t|t)$ is read as the estimate of $\hat{\boldsymbol{\theta}}$ at time t (t|) based on information upto and including time t (|t).) When configured for parameter estimation, the KF equations become

Prediction: $\hat{\boldsymbol{\theta}}(t|t-1) = \hat{\boldsymbol{\theta}}(t-1|t-1)$ (4.19a)

$$\boldsymbol{\Phi}(t|t-1) = \boldsymbol{\Phi}(t-1|t-1) + \mathbf{R}_w \tag{4.19b}$$

Correction: $\hat{\theta}(t|t) = \hat{\theta}(t|t-1) + \phi(t)[\, y(t) - x^T(t)\hat{\theta}(t|t-1)\,]$ (4.20a)

where $\phi(t) = \Phi(t|t-1)x(t)\,[\, r_v + x^T(t)\Phi(t|t-1)x(t)\,]^{-1}$ (4.20b)

and $\Phi(t|t) = [\, I - \phi(t)x^T(t)\,]\,\Phi(t|t-1)$ (4.20c)

where $\mathbf{R}_w$ and r_v are the process noise covariance matrix and the output noise variance respectively.

Note the similarity to RLS, both in form and computational requirements. The primary difference is in the way in which elements of the covariance matrix are inflated to increase adaptivity in the presence of time varying parameters. In the case of RLS all elements are scaled by the forgetting factor λ, equation (4.3), whereas in the KF only selected elements are increased by the addition of the noise covariance matrix $\mathbf{R}_w$, equation (4.19b).

In adopting the KF for the parameter estimation stage of the bilinear STC, the observation vectors are extended as indicated in Chapter 3 to produce a tandem KF/ISO scheme.

4.5.3 Extended Kalman filter for joint state and parameter estimation

The tandem schemes in which parameters and states are estimated in separate stages may be replaced by the EKF scheme where parameters and states are estimated simultaneously. Define the augmented state vector

$$z^T(t) = [\, x^T(t) : \theta^T(t)\,] \tag{4.21}$$

consisting of both states and parameters. The EKF generates estimates $\hat{z}(t|t)$ of the augmented state vector via a two stage prediction/correction process in a similar manner to the KF. The important distinction between the KF for parameter estimation and the EKF for joint state and parameter estimation is that the prediction stage is no longer

trivial, requiring a knowledge of the state evolution between samples. In addition, even in the linear case the formulation of the EKF problem leads to non-linear state equations which are required to be linearised about each discrete operating region along the trajectory of $\mathbf{z}(t)$ (See Appendix 9). The non-linear function is replaced by the Jacobian matrix $\mathbf{J}(t)$ which is evaluated at each time step.

Linear STC

The EKF algorithm for the linear STC takes the form

Prediction: $$\hat{\mathbf{z}}(t|t-1) = \mathbf{P}^* \hat{\mathbf{z}}(t-1|t-1) + \mathbf{Q}^* \mathbf{u}(t-1) \tag{4.22a}$$

$$\boldsymbol{\Phi}(t|t-1) = \mathbf{J}(t-1)\boldsymbol{\Phi}(t-1|t-1)\mathbf{J}^T(t-1) + \mathbf{R}_w \tag{4.22b}$$

Correction: $$\hat{\mathbf{z}}(t|t) = \hat{\mathbf{z}}(t|t-1) + \boldsymbol{\phi}(t)[\, y(t) - \mathbf{H}^* \hat{\mathbf{z}}(t|t-1) \,] \tag{4.23a}$$

where $$\boldsymbol{\phi}(t) = \boldsymbol{\Phi}(t|t-1)\mathbf{H}^{*T}[\, r_v + \mathbf{H}^* \boldsymbol{\Phi}(t|t-1)\mathbf{H}^{*T} \,] \tag{4.23b}$$

and $$\boldsymbol{\Phi}(t|t) = [\, \mathbf{I} - \boldsymbol{\phi}(t)\mathbf{H}^* \,]\boldsymbol{\Phi}(t|t-1) \tag{4.23c}$$

■

Augmented linear system

The evolution of the augmented state vector for the linear system is described by the augmented state-space model

$$\mathbf{z}(t+1) = \mathbf{P}^* \mathbf{z}(t) + \mathbf{Q}^* \mathbf{u}(t) + \mathbf{R}^* \mathbf{e}(t) \tag{4.24a}$$

$$y(t) = \mathbf{H}^* \mathbf{z}(t) + e(t) \tag{4.24b}$$

in which

$$\mathbf{P}^* = \begin{bmatrix} \mathbf{P} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{n_a} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_{n_{b+1}} \end{bmatrix} \qquad \mathbf{Q}^* = \begin{bmatrix} \mathbf{Q} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}$$

$$\mathbf{R}^* = \begin{bmatrix} \mathbf{R} \\ - \\ \mathbf{0} \\ - \\ \mathbf{0} \end{bmatrix} \qquad \mathbf{H}^{*T} = \begin{bmatrix} \mathbf{H}^T \\ - \\ \mathbf{0} \\ - \\ \mathbf{0} \end{bmatrix} \tag{4.25}$$

and $\mathbf{I}$ is the identity matrix of appropriate dimension

i.e. $$\mathbf{z}(t+1) = \begin{bmatrix} \boldsymbol{x}(t+1) \\ - \; - \\ \boldsymbol{\theta}(t+1) \end{bmatrix} = \begin{bmatrix} \mathbf{P}\boldsymbol{x}(t) + \mathbf{Q}\mathbf{u}(t) \\ - \; - \; - \; - \\ \boldsymbol{\theta}(t) \end{bmatrix} + \begin{bmatrix} \mathbf{R}\mathbf{e}(t) \\ - \; - \\ \mathbf{0} \end{bmatrix}$$

$$\mathbf{y}(t) = \begin{bmatrix} \mathbf{H} & : & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}(t) \\ - \; - \\ \mathbf{0} \end{bmatrix} + \mathbf{e}(t)$$

so that the state equation (4.24a) may be expressed

$$\mathbf{z}(t+1) = f(\mathbf{z}(t),\mathbf{u}(t)) + \mathbf{R}^*\mathbf{e}(t) \tag{4.26}$$

where f is a non-linear function (strictly bilinear) of state and control. The Jacobian for this system is given by

$$\mathbf{J}_\ell(t) = \left. \frac{\partial f}{\partial \mathbf{z}(t)} \right|_{\mathbf{z}(t)=\hat{\mathbf{z}}(t|t)}$$

where suffix ℓ indicates that the original system is linear. This reduces to

$$\mathbf{J}_{\ell}(t) = \left[\begin{array}{ccc|c|c} 0 & & -a_n & & \\ 1 & & -a_{n-1} & \mathbf{0} \quad x_n(t) & \mathbf{0} \quad u(t) \\ \cdot & & \cdot & \cdot \quad \cdot & \cdot \quad \cdot \\ \cdot & \cdot & \cdot & \cdot \quad \mathbf{0} & \cdot \quad \mathbf{0} \\ 0 & 1 & -a_1 & x_n(t) & u(t) \\ \hline & & & & \\ \multicolumn{3}{c|}{\mathbf{0}} & \mathbf{I}_{n_a} & \mathbf{0} \\ & & & & \\ \hline & & & & \\ \multicolumn{3}{c|}{\mathbf{0}} & \mathbf{0} & \mathbf{I}_{n_b+1} \\ & & & & \end{array}\right] \tag{4.27}$$

i.e. the matrix whose ijth element is the partial derivative of the ith element of f with respect to the jth element of $\mathbf{z}(t)$.

■

Bilinear STC

In the case of the bilinear STC, equation (4.22a) in the EKF algorithm is modified to accommodate the bilinear terms, such that

$$\hat{\mathbf{z}}(t|t-1) = \mathbf{P}^{*}\hat{\mathbf{z}}(t-1|t-1) + \mathbf{Q}^{*}\mathbf{u}(t-1) + \sum_{i=1}^{m} u(t-i)\mathbf{N}_i^{*}\hat{\mathbf{z}}(t-1|t-1) \tag{4.28}$$

■

Augmented bilinear system

The state evolution for the augmented bilinear system is described by

$$\mathbf{z}(t+1) = \mathbf{P}^{*}\mathbf{z}(t) + \mathbf{Q}^{*}\mathbf{u}(t) + \mathbf{R}^{*}\mathbf{e}(t) + \sum_{i=1}^{m} u(t-i+1)\mathbf{N}_i^{*}\mathbf{z}(t) \tag{4.29a}$$

$$\mathbf{y}(t) = \mathbf{H}^{*}\mathbf{z}(t) + \mathbf{e}(t) \tag{4.29b}$$

Similarly, the matrices $\mathbf{P}^*$, $\mathbf{Q}^*$, $\mathbf{R}^*$, $\mathbf{H}^*$ and $\mathbf{N}_i^*$ are of the form (4.25) and are defined by

$$\mathbf{P}^* = \begin{bmatrix} \mathbf{P} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{n_a} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_{n_{b+1}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_{m.(n_b+1)} \end{bmatrix} \qquad \mathbf{Q}^* = \begin{bmatrix} \mathbf{Q} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}$$

$$\mathbf{R}^* = \begin{bmatrix} \mathbf{R} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \qquad \mathbf{H}^{*T} = \begin{bmatrix} \mathbf{H}^T \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}$$

and

$$\mathbf{N}_i^* = \begin{bmatrix} \mathbf{N}_i & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \qquad i = 1 \ldots m \qquad (4.30)$$

The Jacobian for this system is given by $\mathbf{J}_b(t) =$

$$
\left[\begin{array}{c|c|c|c|c}
0 \quad -a_n + \sum_{i=1}^{m} \eta_{(n-1)i} u(t-m+1) & x_n(t) & u(t) & u(t)x_n(t) & u(t-m+1)x_n(t) \\
1 \quad -a_{n-1} + \sum_{i=1}^{m} \eta_{(n-2)i} u(t-m+1) & \cdot & \cdot & \cdot & \cdot \\
\cdot \quad \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot \quad \cdot & \cdot & \cdot & \cdot & \cdot \\
0 \quad 1 \quad -a_1 + \sum_{i=1}^{m} \eta_{0i} u(t-m+1) & x_n(t) & u(t) & u(t)x_n(t) & u(t-m+1)x_n(t) \\
\hline
\mathbf{0} & \mathbf{I}_{n_a} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\hline
\mathbf{0} & \mathbf{0} & \mathbf{I}_{n_b+1} & \mathbf{0} & \mathbf{0} \\
\hline
\mathbf{0} & \mathbf{0} & \mathbf{0} & & \mathbf{I}_{m.(n_b+1)}
\end{array}\right]
$$

... (4.31)

where the suffix b indicates that the original system is bilinear.

■

4.5.4 Simulation studies

Simulation studies involving the application of Kalman filtering and extended Kalman filtering techniques are described in [A14] and [A15].

In [A14] the tandem RLS/SKF approach outlined in Chapter 2 is compared to both the KF/SKF approach and the EKF approach when a linear self-tuning pole-placement controller is applied to an identified linear model of a hydraulic servo system [27] in the presence of non-ideal identification signals. In order to replicate the observed non-linearities, in terms of slow variation in feed flow with operating temperature, parameters within this model are varied linearly. Results presented in Figures 4.6, 4.7 and 4.8, which illustrate (a) the control input u(t) and system response y(t) and (b) the convergence behaviour of the estimated parameters, correspond respectively to the use of RLS/SKF, EKF and KF/SKF schemes.

When a system is subject to poor input excitation signals, the estimation procedure attempts to determine the 'p' parameters from effectively 'one' equation. As a consequence, the estimated parameter values can become highly correlated, drifting in sympathy, such that this 'near singular' set of equations is satisfied. From Figure 4.6 it is evident that the RLS/SKF approach can lead to divergent estimates and possible instability. It is evident from Figure 4.7 that the EKF reduces problems of divergence but may produce biased estimates. The KF/SKF, on the other hand, is found to eliminate the problems of both divergence and bias; as is evident in Figure 4.8.

The major shortfall of the RLS approach would appear to be due to the fact that use of a forgetting factor tends to increase the cross correlations (as observed by the algorithm), causing the estimated parameter values to drift in sympathy. For both the KF and EKF approaches, the process noise covariance matrix $\mathbf{R}_w$ is added and, since $\mathbf{R}_w$ is usually diagonal (or at least diagonally dominant), the observed cross correlations within the algorithms are not increased, leading possibly to the better performance of the KF/SKF and EKF schemes. The reduced performance of the EKF, in this case, is due possibly to the cross correlations introduced at the covariance update stage involving the Jacobian.

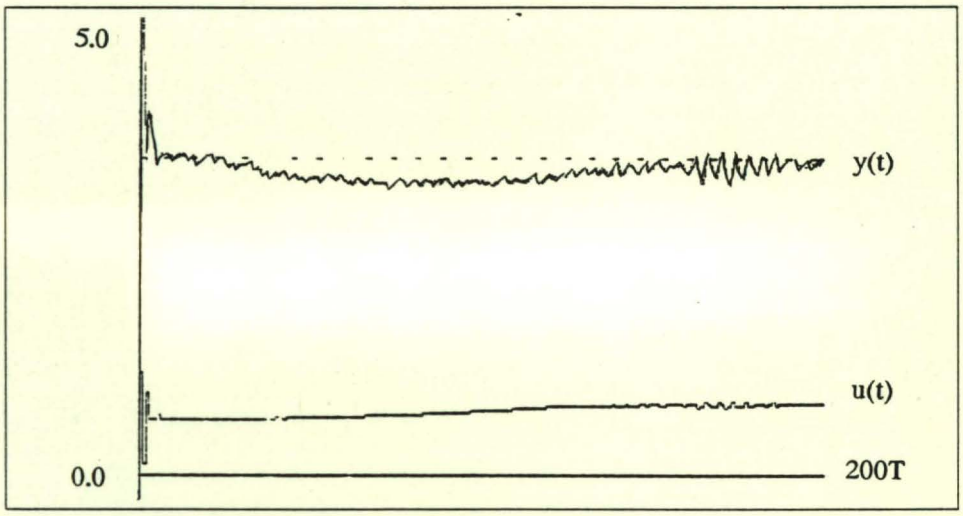


Figure 4.6(a)

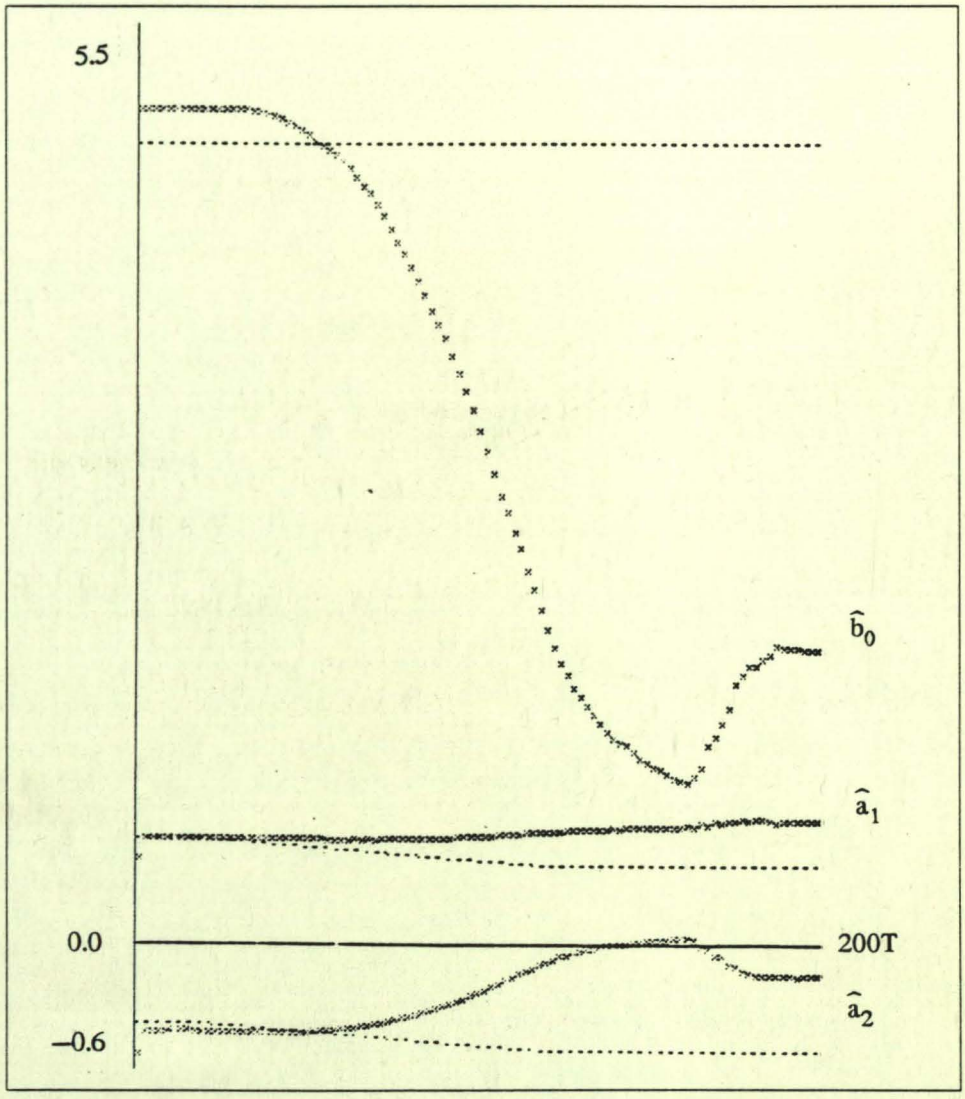


Figure 4.6(b)

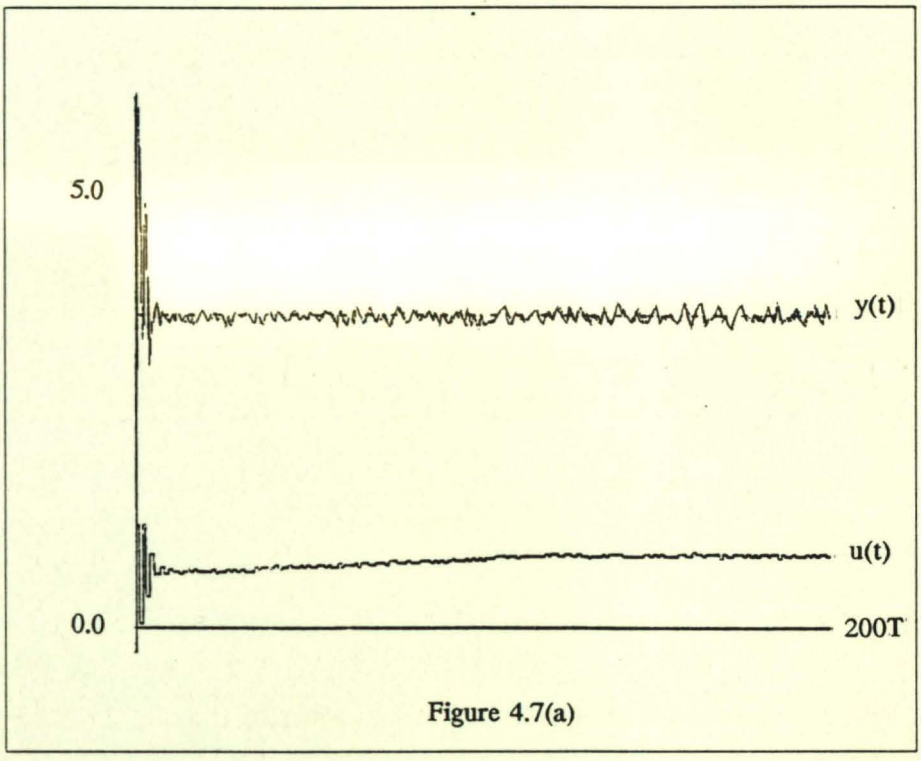


Figure 4.7(a)

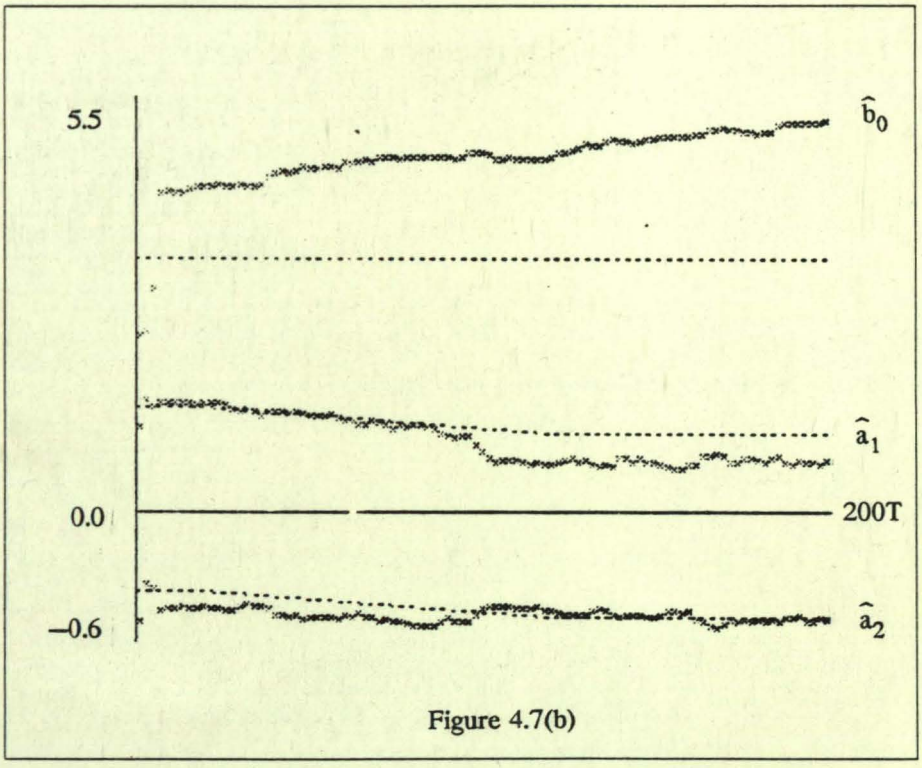


Figure 4.7(b)

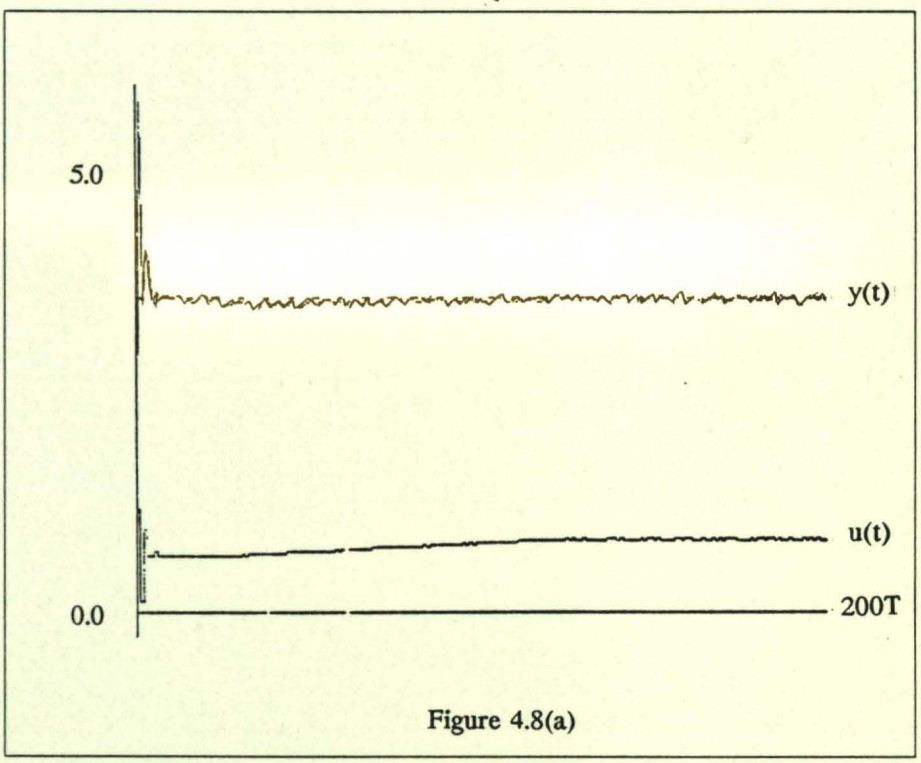


Figure 4.8(a)

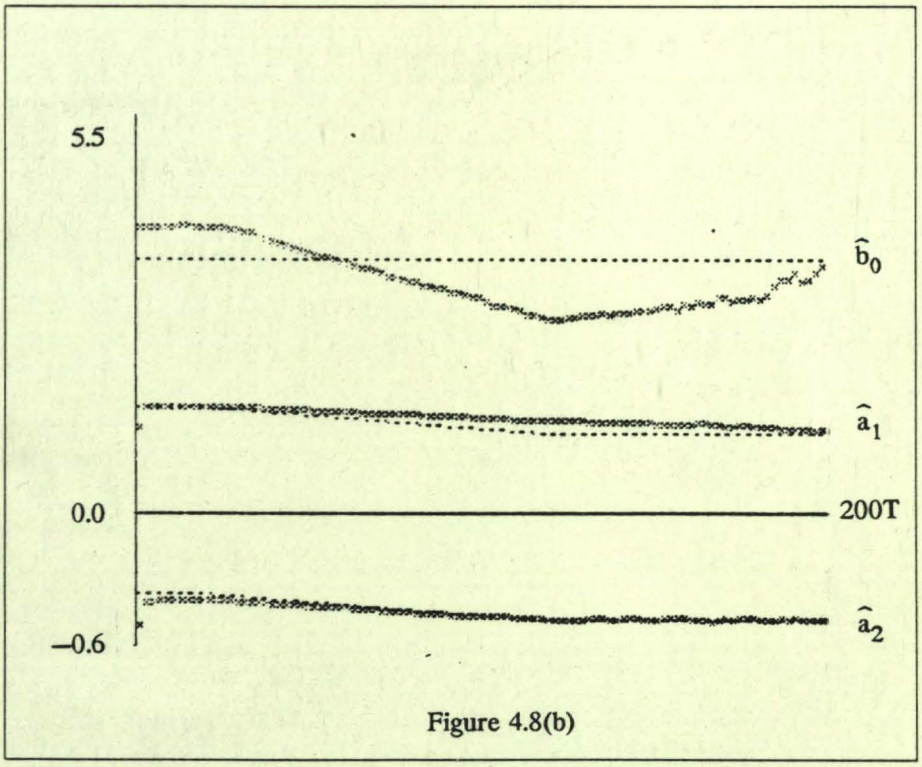


Figure 4.8(b)

In [A15] the tandem RLS/ISO approach outlined in Chapter 3 is compared to the EKF approach when applied to the simulated bilinear system of section 4.3.3. The system is operated in open-loop and driven by and 'ideal' pseudo random input signal. In order to evaluate the integrity of the two estimation procedures the system is subjected to a number of test conditions. It is noted that the EKF can make more meaningful use of available *a priori* knowledge than the RLS/ISO approach and, in order to draw sensible conclusions, it is useful to consider the implications of different initialisation procedures outlined in [A15].

The results presented in Figures 4.9, 4.10 and 4.11 illustrate the convergence behaviour of the estimated parameters for (a) the RLS/ISO scheme and (b) the EKF scheme and correspond, respectively, to Tests 1, 2 and 3 which are outlined as follows.

In Test 1, the model parameters are fixed and the system is subjected to a range of noise levels. Both techniques are initiated to reflect an absence of *a priori* knowledge. It is found that for low noise levels there is little difference in performance. However, the RLS/ISO scheme becomes increasingly superior for higher noise levels as is evident in Figure 4.9 which corresponds to $\left| e(t) \right| \leq 0.2$.

In Test 2, the model parameters are again fixed, but this time the estimation schemes are initialised to reflect the presence of *a priori* knowledge [A15]. Figure 4.10 illustrates the corresponding convergence behaviour of the estimated parameters when subjected to the noise sequence in which $\left| e(t) \right| \leq 0.2$. Although, as expected, steady-state performance is similar, use of the tuned EKF gives rise to an improved transient performance.

In Test 3, the parameter a_1 within the model is varied linearly over the duration of the simulation with the two schemes initiated as in Test 2 except that the appropriate diagonal element in $\mathbf{R}_w$, corresponding to the parameter a_1, is increased by a

factor of ten. It is evident from Figure 4.11 that, due to its versatility, the tuned EKF can give rise to superior performance in terms of parameter tracking ability. The tracking ability of the RLS/ISO approach can be improved by reducing the value of the forgetting factor. However, it is noted that even in the case of 'ideal' input excitation signals, this is achieved at the expense of reduced estimation accuracy of the fixed parameters.

Whilst it is shown that the tandem RLS/ISO scheme provides a robust approach requiring little or no *a priori* knowledge, it is noted that the EKF scheme, with its increased degrees of freedom, is able to make more meaningful use of such knowledge and, if tuned correctly, can give rise to a superior performance.

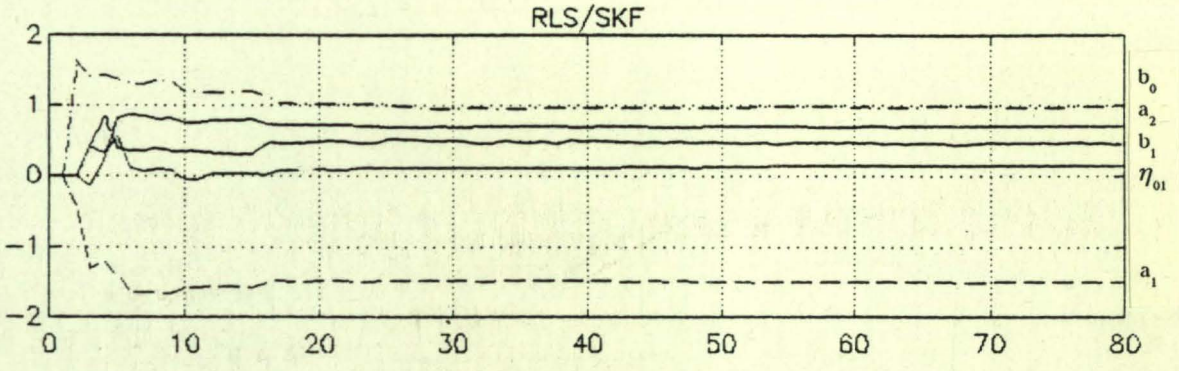


Figure 4.9(a)

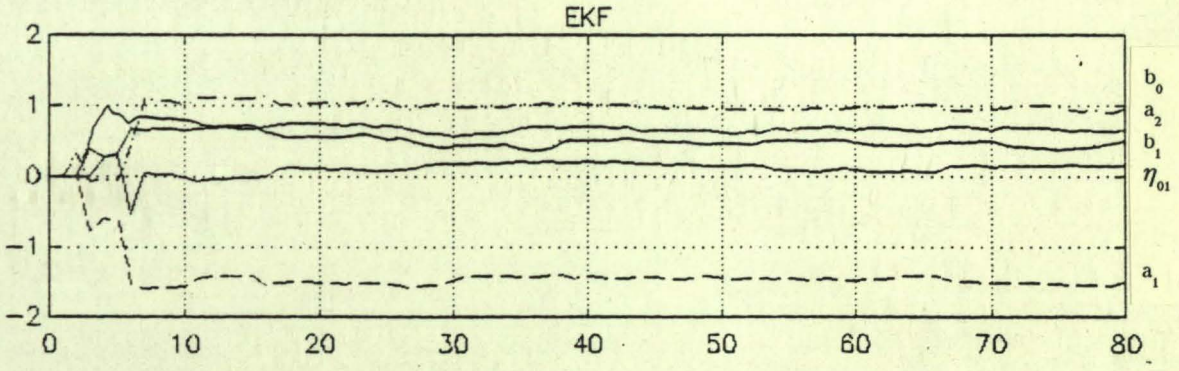


Figure 4.9(b)

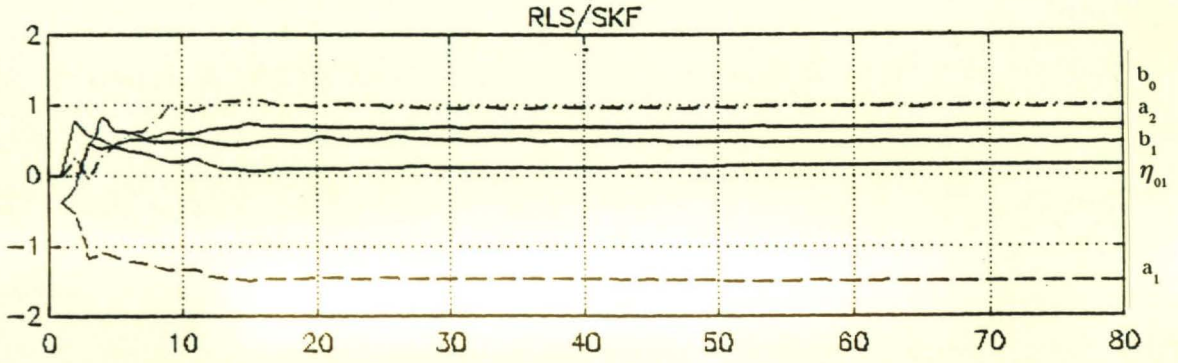


Figure 4.10(a)

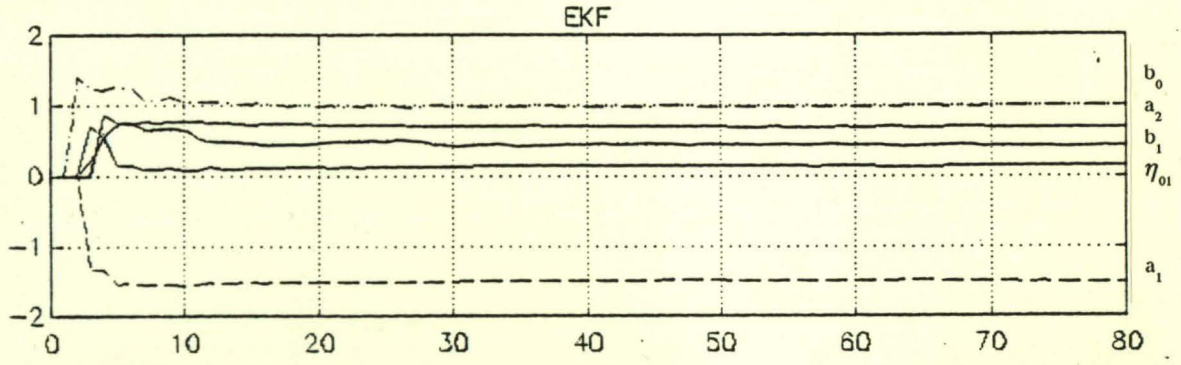


Figure 4.10(b)

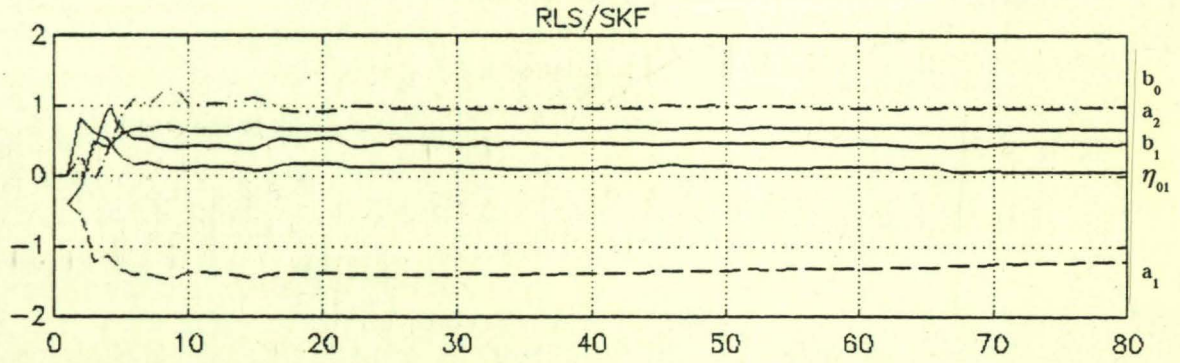


Figure 4.11(a)

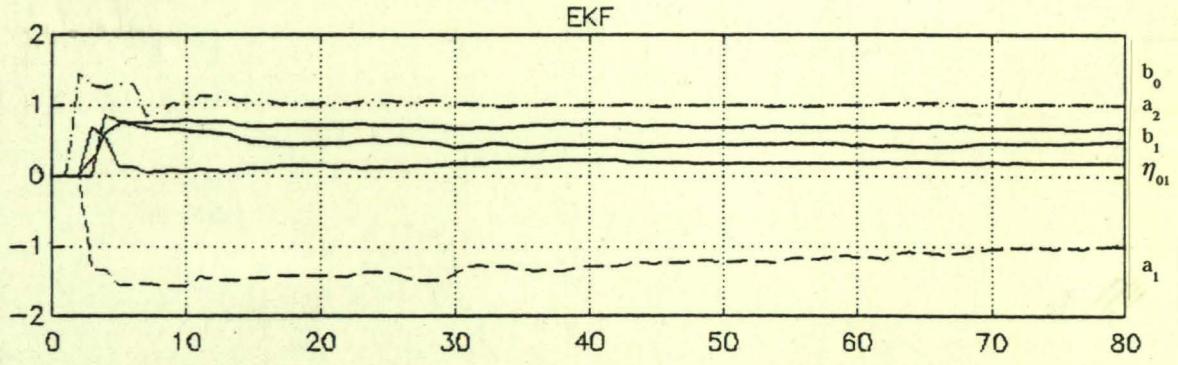


Figure 4.11(b)

4.6 Cautious least squares

4.6.1 The search for a simple yet effective approach

Proposals outlined so far for enhanced estimation schemes have, to some extent, been found to alleviate the problems encountered in the implementation of STC. However, the fundamental conflict within an STC lies in the requirement that it must simultaneously perform the dual roles of estimation and control. In practice the performance of an STC may deteriorate as a result of non-ideal input excitation signals; unfortunately a condition experienced by well regulated plant particularly over prolonged periods of steady-state operation. Whilst combinations of the various enhanced parameter estimation techniques have been employed in an albeit ad hoc manner, the problem of compromise between robustness and adaptivity remains an outstanding issue. This is further compounded by the increased computational complexity which often accompanies the enhanced techniques. This in turn leads to an increased computational delay and the possibility of an overall deterioration in performance of the closed-loop STC, such that the potential benefits of the enhanced techniques may never be fully realised.

Prompted by the need to overcome these problems, a computationally simple yet effective technique termed cautious least squares has been proposed [A10]. Cautious least squares is a conceptually simple approach providing a robust estimation scheme which is able to retain the desirable adaptive features of any enhanced technique, such as variable forgetting factors and covariance matrix reset, etc., which may readily be incorporated within its framework. Furthermore, it is able to alleviate problems arising during prolonged periods of closed-loop steady-state operation due to poor input excitation.

4.6.2 Derivation of cautious least squares

Essentially, cautious least squares (CLS) attempts to minimise the modified cost function

$$J_c(\theta) = (y-X\theta)^T \Lambda (y-X\theta) + (\theta-\theta_s)^T \Psi (\theta-\theta_s) \tag{4.32}$$

i.e. $J_c(\theta) = J_1 + J_2$

the first term of which corresponds to the normal RLS cost function in which $y-x^T\theta$ is the prediction error Σ (equation (4.6)) and Λ is the diagonal matrix

$$\Lambda = \text{diag}\,[\,\lambda^M \ldots . \lambda^2\ \lambda\,],$$

see Appendix 7. The second 'cautious' term in the cost function attempts to minimise the deviation of the estimated parameter vector $\hat{\theta}(t)$ from some pre-specified 'safe' set of parameter values denoted θ_s. The CLS algorithm is realised as an additional sub-algorithm which operates in tandem with the RLS algorithm. It has the effect of realigning the estimated parameter vector $\hat{\theta}(t)$ generated from within the normal RLS algorithm towards the safe set θ_s; the amount of realignment being associated with the user defined weighting matrix Ψ. The parameter vector which minimises the modified cost function (4.32) is defined as the cautious parameter vector $\tilde{\theta}(t)$.

Making use of the vector differentiation given in Appendix 1

$$\frac{\partial J_c(\theta)}{\partial \theta} = 2\left[\ -X^T\Lambda(y-X\theta) + \Psi(\theta-\theta_s)\ \right] \tag{4.33}$$

which equating to zero and setting $\theta=\tilde{\theta}$ gives

$$-X^T\Lambda y + X^T\Lambda X\tilde{\theta} + \Psi\tilde{\theta} - \Psi\theta_s = 0$$

$$(X^T\Lambda X + \Psi)\tilde{\theta} = X^T\Lambda y + \Psi\theta_s$$

so that

$$\tilde{\theta} = [\,X^T\Lambda X + \Psi\,]^{-1}\,(X^T\Lambda y + \Psi\theta_s) \tag{4.34}$$

Since the estimated parameter vector $\hat{\theta}$ generated from RLS is given by

$$\hat{\theta} = [\,X^T\Lambda X]^{-1}\,X^T\Lambda y \tag{4.35}$$

it follows that equation (4.34) may be expressed as

$$\tilde{\theta} = [\Phi^{-1} + \Psi]^{-1}[\Phi^{-1}\hat{\theta} + \theta_s] \tag{4.36}$$

where $\Phi = [X^T \Lambda X]^{-1}$ is the error covariance matrix.

4.6.3 Realisation of the CLS approach

Since a recursive solution for equation (4.36) is not immediate, the approach is realised as a tandem operation of two separate procedures of:

Obtain $\hat{\theta}$ from RLS by minimising J_1 only,

i.e. $$J_1(\theta)\Big|_{\theta=\hat{\theta}} = (y - X\theta)^T \Lambda (y - X\theta) \; ; \tag{4.37}$$

followed by, obtain $\tilde{\theta}$ from CLS by minimising J_2 only,

i.e. $$J_2(\theta)\Big|_{\hat{\theta}=\tilde{\theta}} = (\hat{\theta} - \theta_s)^T \Psi (\hat{\theta} - \theta_s) \tag{4.38}$$

The tandem RLS/CLS approach outlined by equations (4.37) and (4.38) forms the basis of the algorithms adopted. Note that if CLS is applied at each time step then a further p (p=dim{θ}) iterations would be required. Whilst such a sequential scheme would ensure that full weighting is given to the safe set the resulting algorithm would be computationally cumbersome. However, its operation is considered here since it forms the basis of a number of possible realisations.

Although it has not been considered in this Thesis, it is interesting to note that the CLS approach could also be realised in the information matrix filtering domain [65].

The cautious parameter vector θ_s is obtained from either knowledge of the plant or time series analysis of input/output data.

Sequential operation of CLS

It should be noted that in the tandem operation of RLS and CLS, the weighting matrix Ψ provides the user with the flexibility to assign confidence in the individual safe set values; the overall influence on the safe set being dependent upon how frequently the CLS sub-algorithm is applied.

In order to give a detailed description of the operation of sequential CLS it is useful to imagine that following each successive iteration of RLS time is 'frozen' until the cautious estimate has been generated. The approach may be illustrated schematically as in Figure 4.12.

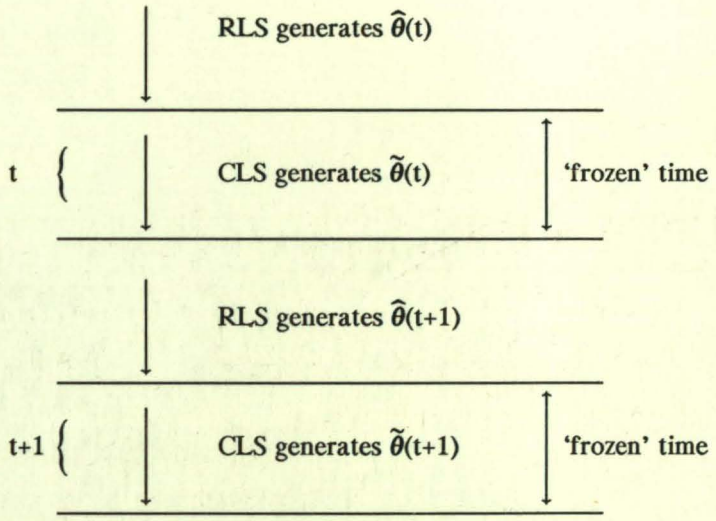


Figure 4.12.

The sequential operation of the CLS algorithm involves, at each time step, a further p iterations in order to minimise the cautious component J_2 of the cost function (4.32). For clarity, it is convenient to assume that whilst time is 'frozen' between sucessive iterations of RLS the discrete time index t is replaced by the index j, where j=1,2,..p. The sequential CLS algorithm then takes the form

$$\theta_j = \theta_{j-1} + \phi_j [\, e_j^T(\theta_s - \theta_{j-1}) \,] \tag{4.39}$$

$$\phi_j = \Phi_{j-1}\, e_j [\, 1 + e_j^T\, \Phi_{j-1}\, e_j \,]^{-1} \tag{4.40}$$

$$\Phi_j = [\, I - \phi_j\, e_j^T \,] \Phi_{j-1} \tag{4.41}$$

where the e_j are the orthogonal unit vectors defined as

$$e_j^T = [\, \delta_{1j}\, \delta_{2j}\, \delta_{3j} \ldots \delta_{pj} \,] \tag{4.42}$$

in which δ_{ij} is the Kronecker delta function

$$\delta_{ij} = \begin{cases} 1 & i=j \\ 0 & i \neq j \end{cases}$$

Note that by making use of artificial data in the form of the orthogonal unit vectors the CLS algorithm is able to influence the overall estimation algorithm, tending to re-align the estimated parameter vector towards the safe set θ_s. It is also noted that use of the orthogonal unit vectors effectively provides for an artificial excitation signal within the estimation algorithm, thus alleviating, to some extent, the problem of covariance blow-up during steady-state periods of operation without the need to disturb/perturb the plant.

CLS is initiated at each time step with $\theta_0 = \hat{\theta}(t)$ and $\Phi_0 = \Phi(t)$ where $\hat{\theta}(t)$ and $\Phi(t)$ are the estimated parameter vector and covariance matrix generated from RLS respectively. Following the p iterations of CLS, the cautious parameter estimate and covariance matrix become θ_p and Φ_p respectively, with the cautious estimate $\tilde{\theta}(t) = \theta_p$ used for control purposes. At the next time step $\tilde{\theta}(t)$ and Φ_p are fed back to the RLS algorithm such that $\hat{\theta}(t-1) = \tilde{\theta}(t)$ and $\Phi(t-1) = \Phi_p$. The overall procedure corresponding to the use of sequential CLS for a linear system, as viewed by the estimator, then becomes

$$
\begin{bmatrix} y(t-1) \\ \theta_{s_1} \\ \theta_{s_2} \\ \vdots \\ \theta_{s_j} \\ \vdots \\ \theta_{s_p} \\ \\ y(t) \\ \theta_{s_1} \\ \theta_{s_2} \\ \vdots \\ \theta_{s_j} \\ \vdots \\ \theta_{s_p} \\ \vdots \end{bmatrix} = \begin{bmatrix} y(t-2) & y(t-3) & \dots & y(t-1-n_a) & u(t-1-k) & u(t-2-k) & \dots & u(t-1-k-n_b) \\ 1 & 0 & \dots & & & & & 0 \\ 0 & 1 & \dots & & & & & 0 \\ & & & & & & & \\ 0 & 0 & \dots & & & & & 0 \\ & & & & & & & \\ 0 & 0 & \dots & & & & & 1 \\ \\ y(t-1) & y(t-2) & \dots & y(t-n_a) & u(t-k) & u(t-1-k) & \dots & u(t-k-n_b) \\ 1 & 0 & \dots & & & & & 0 \\ 0 & 1 & \dots & & & & & 0 \\ & & & & & & & \\ 0 & 0 & \dots & & & & & 0 \\ & & & & & & & \\ 0 & 0 & \dots & & & & & 1 \\ \vdots & & & & & & & \vdots \\ \vdots & & & & & & & \vdots \end{bmatrix} \begin{bmatrix} \hat{\theta}(t-1) \\ \tilde{\theta}_1 \\ \tilde{\theta}_2 \\ \vdots \\ \tilde{\theta}_j \\ \vdots \\ \tilde{\theta}_p = \tilde{\theta}(t-1) \\ \\ \hat{\theta}(t) \\ \tilde{\theta}_1 \\ \tilde{\theta}_2 \\ \vdots \\ \tilde{\theta}_j \\ \vdots \\ \tilde{\theta}_p = \tilde{\theta}(t) \\ \vdots \end{bmatrix}
$$

... (4.43),

which may be readily extended to the bilinear case. Note that the maximum influence of the safe set is obtained if CLS is applied at each time step. It is clear that many variants of the sequential approach are possible and two such variants are outlined here.

■

Cyclic operation of CLS

A natural extension of the sequential approach is the cyclic, or selective, operation of CLS. In this way, only one parameter is 'cautioned' after each iteration of RLS. In the cyclic approach all the parameters are cautioned 'cyclically' whereas in the selective approach only the problematic parameters are cautioned. In the latter approach, some form of fault detection mechanism would need to be incorporated. The cyclic operation of CLS may be viewed as

$$\begin{bmatrix} \cdot \\ \cdot \\ \theta_{s_p} \\ y(t) \\ \theta_{s_1} \\ y(t+1) \\ \theta_{s_2} \\ \cdot \\ \cdot \end{bmatrix} = \begin{bmatrix} \cdot & & & & & & & \cdot \\ \cdot & & & & & & & \cdot \\ 0 & 0 & \cdots & & & & & 1 \\ y(t-1) & y(t-2) & \cdots & y(t-n_a) & u(t-k) & u(t-1-k) & \cdots & u(t-k-n_b) \\ 1 & 0 & \cdots & & & & & 0 \\ y(t) & y(t-1) & \cdots & y(t-n_a+1) & u(t-k+1) & u(t-k) & \cdots & u(t-k-n_b+1) \\ 0 & 1 & \cdots & & & & & 0 \\ \cdot & & & & & & & \cdot \\ \cdot & & & & & & & \cdot \end{bmatrix} \begin{bmatrix} \cdot \\ \cdot \\ \tilde{\theta}(t-1) \\ \hat{\theta}(t) \\ \tilde{\theta}(t) \\ \hat{\theta}(t+1) \\ \tilde{\theta}(t+1) \\ \cdot \\ \cdot \end{bmatrix} \quad \ldots \quad (4.44)$$

■

CLS on reset

In this approach CLS is combined within a variable forgetting factor and covariance matrix reset procedure which is implemented whenever a fault condition, indicative of large estimation errors, is detected. The reset mechanism, being a function of the estimation prediction error, is triggered when the demand for the variable forgetting factor falls below some pre-specified threshold level λ_ρ (see section 4.3.2). Such a combined procedure facilitates, at reset stages, an efficient means of cautioning all parameters in a single iteration; this being made possible by exploiting the diagonal

properties of the newly updated covariance matrix. The algorithm takes the form:

If the demand for the forgetting factor $\lambda(t) \leq \lambda_\rho$ (Note that the actual value of the forgetting factor is bounded such that $\lambda_\ell < \lambda(t) \leq 1.0$ where $\lambda_\ell \gg \lambda_\rho$.) then the cautious covariance matrix $\boldsymbol{\Phi}_p$ becomes $\tilde{\mu}\mathbf{I}$ so that the cautious parameter estimate $\tilde{\boldsymbol{\theta}}(t)$ from (4.36) becomes

$$\tilde{\boldsymbol{\theta}}(t) = (1-\omega)\hat{\boldsymbol{\theta}}(t) + \omega\boldsymbol{\theta}_s \tag{4.45}$$

where $\omega = \tilde{\mu}(1+\tilde{\mu})^{-1}$.

The value of ω in (4.45) may be interpreted as a measure of the users confidence in the safe set $\boldsymbol{\theta}_s$, which then determines the value of $\tilde{\mu}$ used in the resetting. In order to retain the influence of $\boldsymbol{\theta}_s$ in subsequent iterations, RLS is re-initiated with the covariance matrix $\boldsymbol{\Phi}$ reset to the identity matrix **I**.

■

4.6.4 Simulation studies

The bilinear system used previously

$$y(t) = -a_1(t)y(t-1) - 0.7y(t-2) + u(t-1) + 0.5u(t-2) + 0.12x(t-1)u(t-1) + e(t)$$

is again adopted for investigative purposes where $a_1(t)$ is a time varying parameter. Closed-loop poles are specified at -3 and -5 in the s-plane with a sampling interval of T=0.1 second. The objectives of the simulation studies are to investigate the use of caution on reset combined with variable forgetting factor and covariance matrix reset and to compare the resulting performance to that obtained in the absense of caution.

The system is again subjected to a step input r(t)=0.05 and an output disturbance e(t) represented by discrete white noise, where $\left| e(t) \right| \leq 0.04$, with the parameter $a_1(t)$ varied according to

$$a_1(t) = \begin{cases} -1.5 & t \leq 50 \\ -1.5-(t-50)/1200 & 50 < t \leq 650 \\ -1.5 & t > 650 \end{cases}$$

with each simulation being run over 750 iterations. The algorithm is initiated with $\Phi(0)=10^5\mathbf{I}$ and $\theta_s^T=[1.7\ -0.7\ 1.0\ 0.5\ 0.12]$, a confidence factor $\omega=0.75$ and an initial value of forgetting factor being $\lambda_0=0.97$.

Figures 4.13(a) and 4.13(b) illustrate, respectively, the system response y(t) and control input u(t) when caution is employed whilst those of Figures 4.14(a) and 4.14(b) give corresponding responses in the absence of caution. It is clear that use of caution on reset gives rise to a markedly superior performance. Figures 4.15(a) and 4.15(b) illustrate, for the case of caution on reset, the convergence behaviour of the parameter estimates and the state space control law parameters respectively. These again show an improvement when compared to Figures 4.16(a) and 4.16(b) which correspond to the same conditions in the absence of caution.

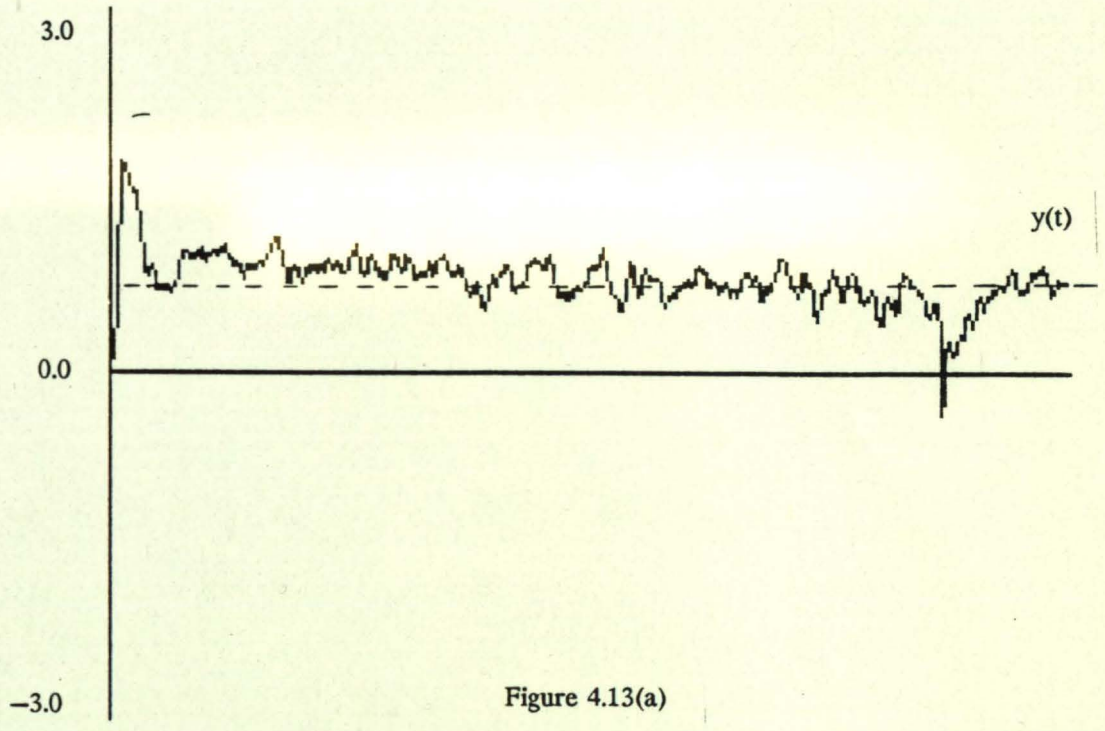


Figure 4.13(a)

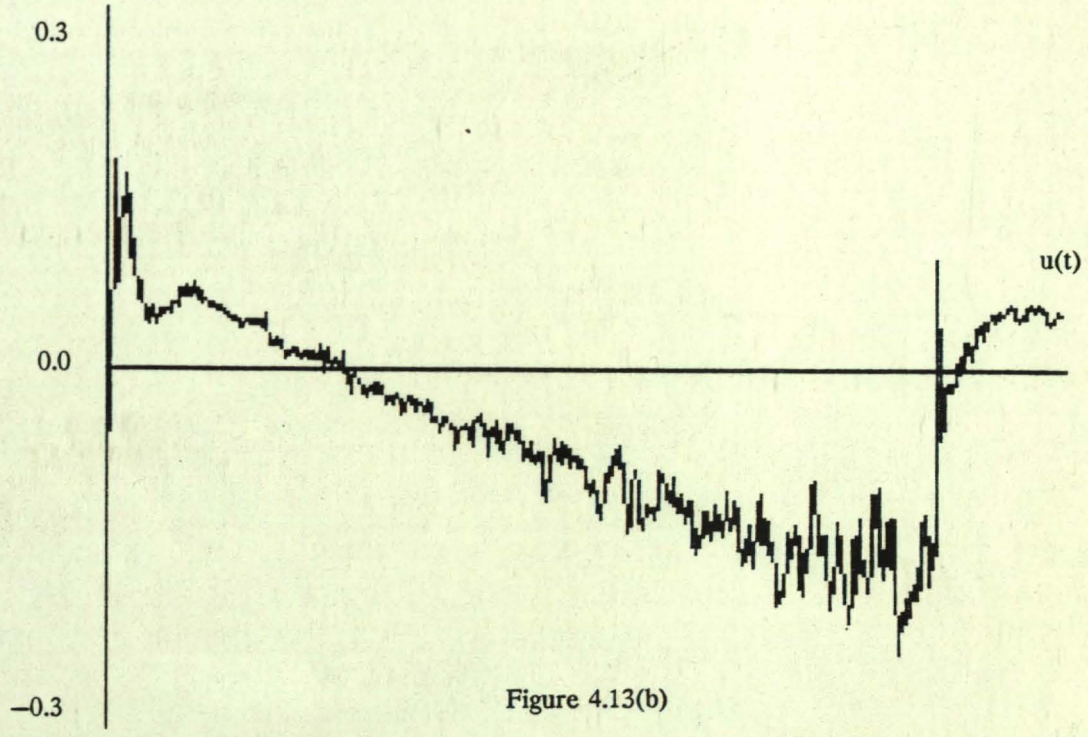


Figure 4.13(b)

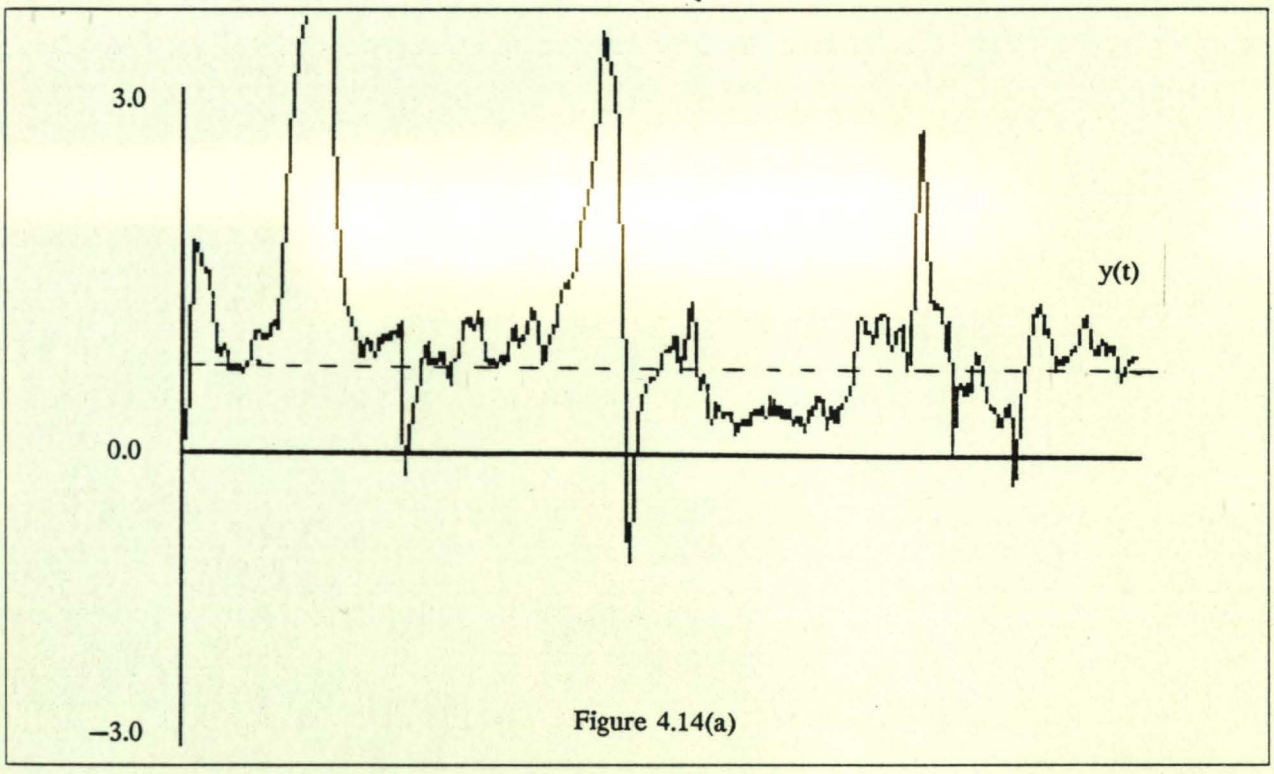


Figure 4.14(a)

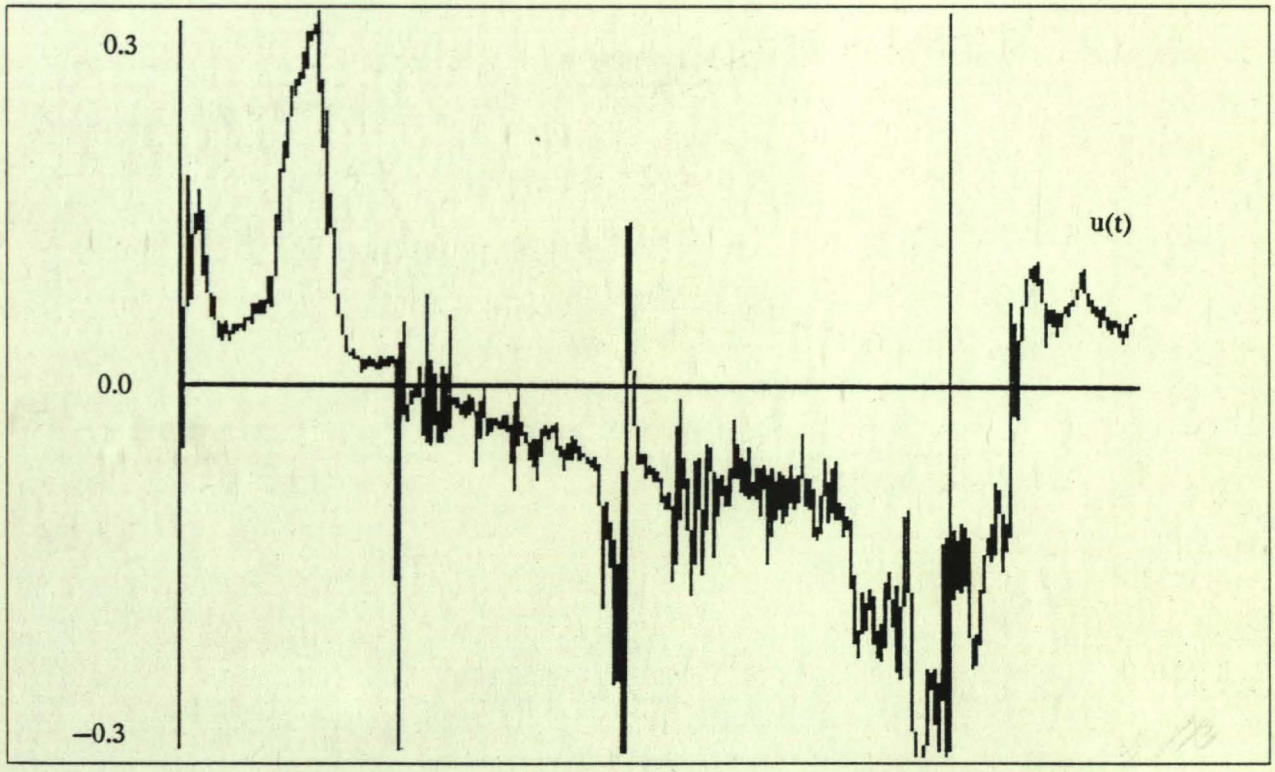


Figure 4.14(b)

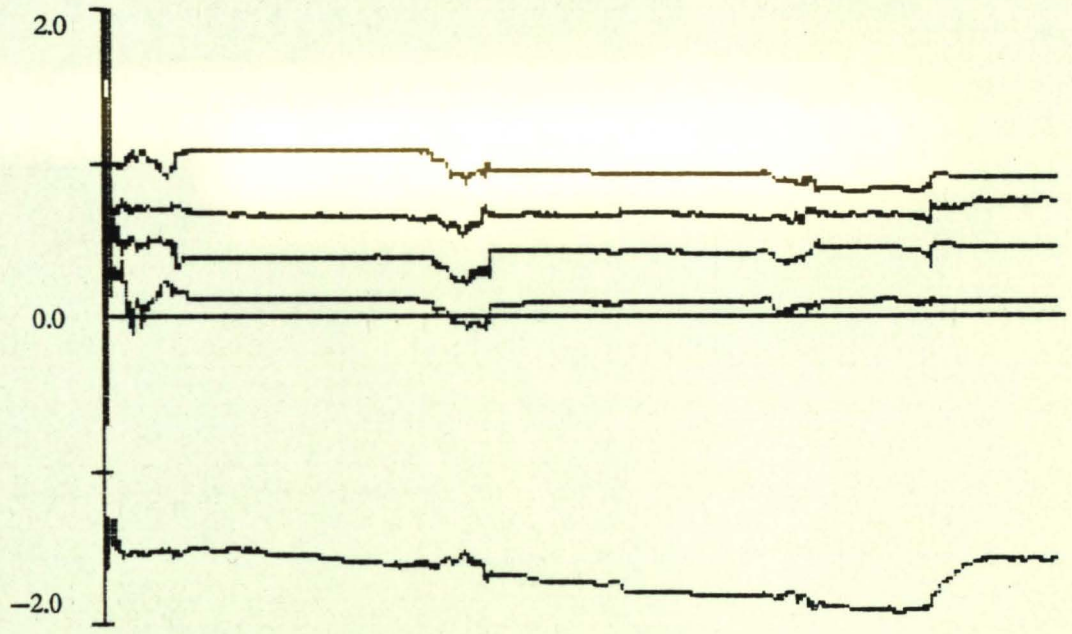


Figure 4.15(a)

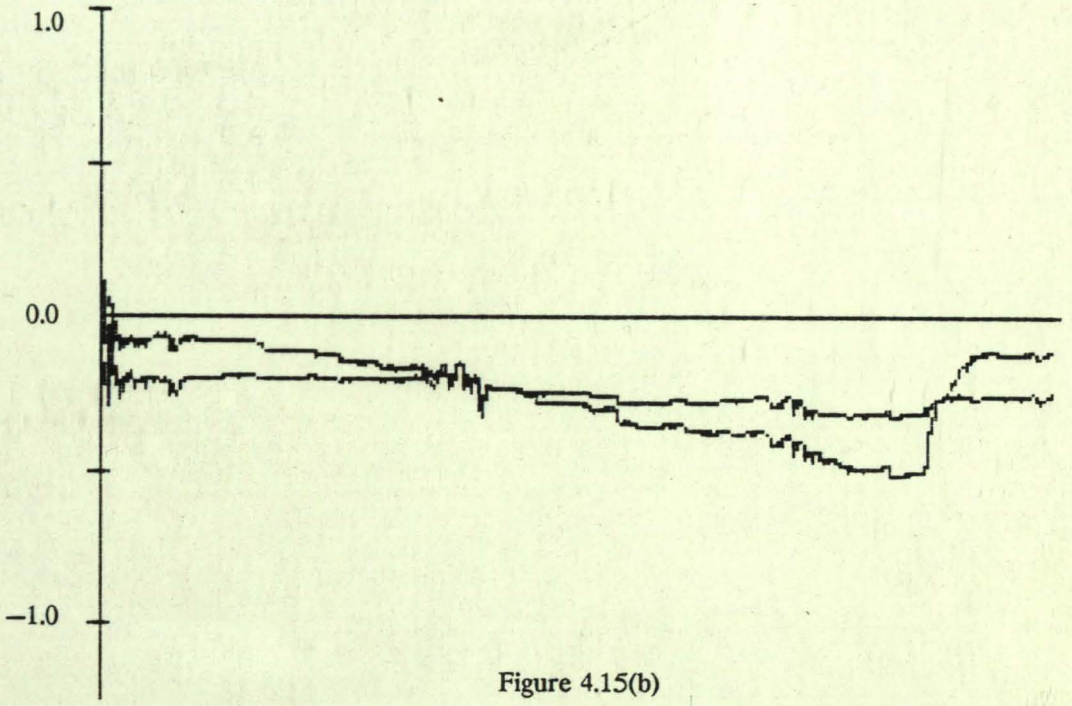


Figure 4.15(b)

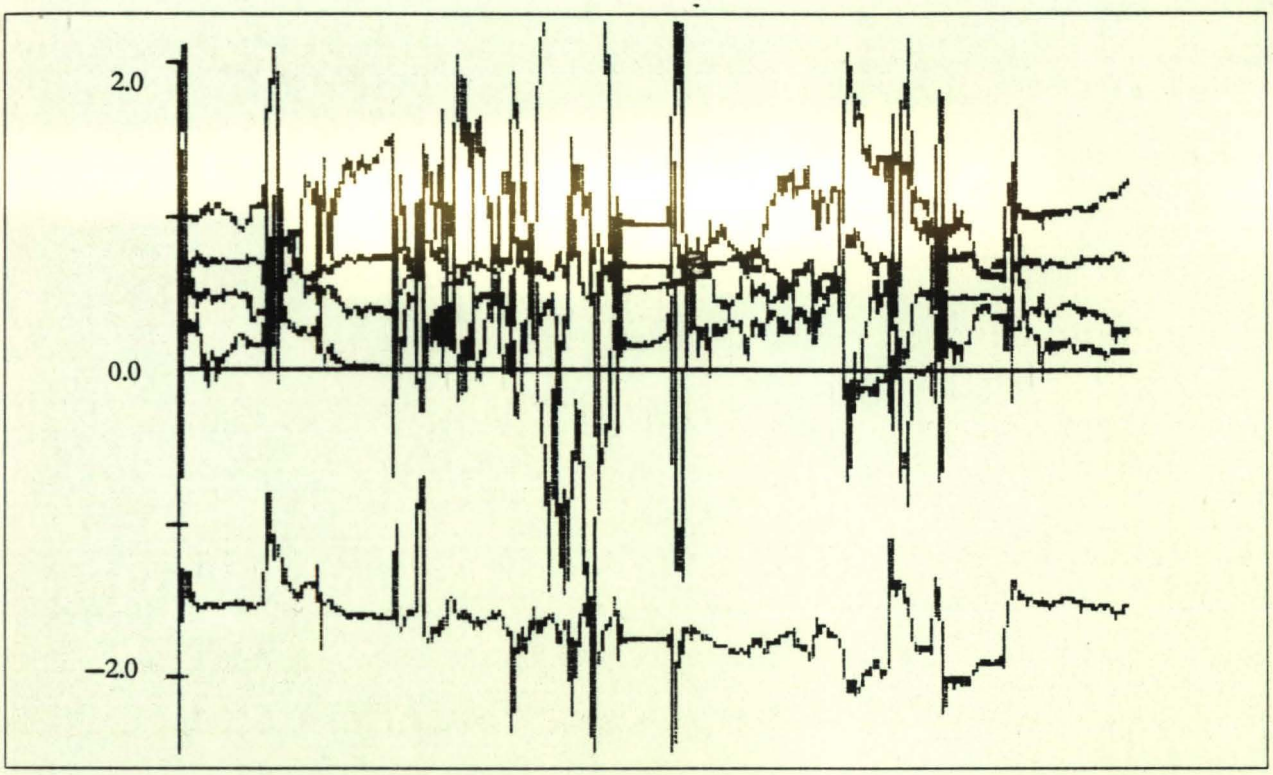


Figure 4.16(a)

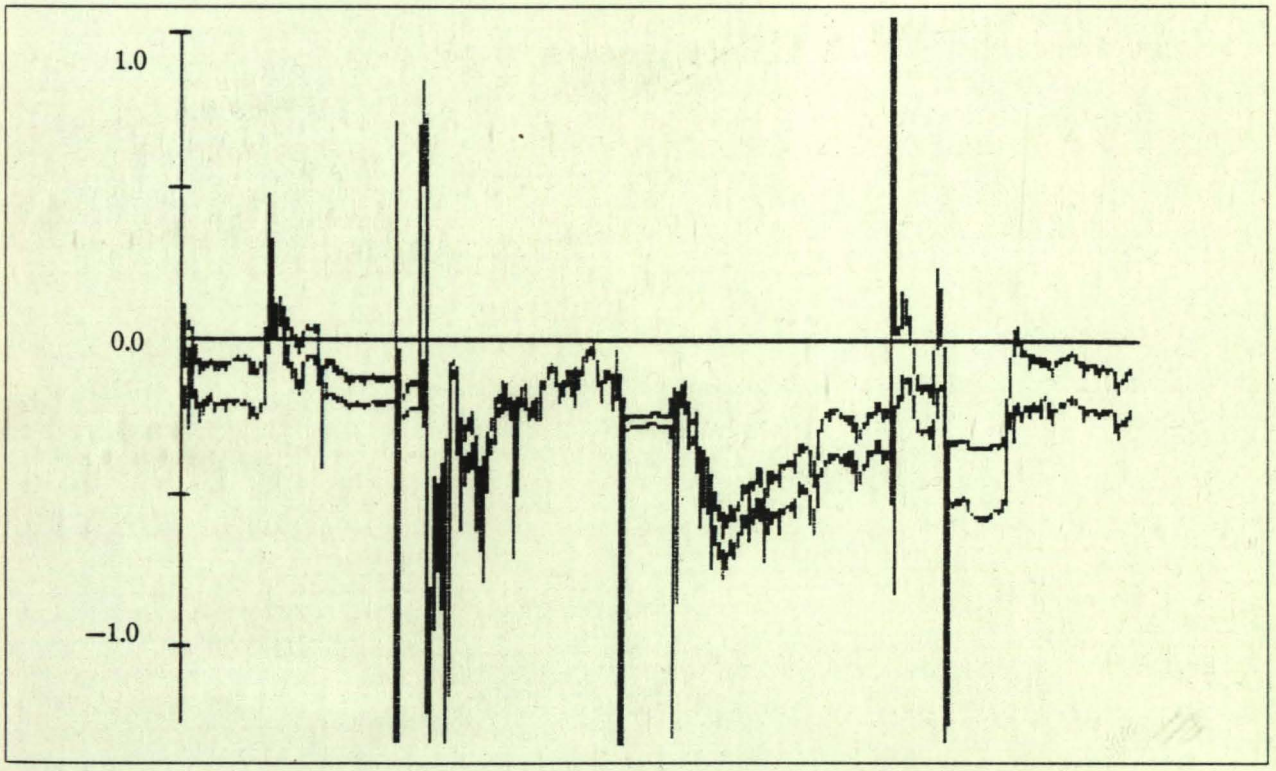


Figure 4.16(b)

4.7 Concluding remarks

A number of enhanced parameter estimation techniques have been investigated and developed with specific attention being directed towards their effectiveness when incorporated within the bilinear STC framework introduced in Chapter 3.

The proposed enhanced techniques are essentially extensions and adaptations of existing techniques for linear systems and include; a hybrid variable forgetting factor for slow parameter variation; a combined two-tier variable forgetting factor and covariance matrix resetting technique for both slow and sudden parameter variation; an alternative form of the recursive instrumental variables technique for improved convergence behaviour; Kalman and extended Kalman filtering techniques allowing the incorporation of engineering knowledge; and a novel cautious least squares procedure which may be applied sequentially, cyclically or on reset conditions to provide a simple yet robust approach capable of retaining the desirable adaptive properties of any other coexisting enhanced technique.

Whilst the enhancements have been aimed at improving the integrity of the bilinear STC, the results are applicable for linear systems although the level of improvement may not be as significant. When appliying STC to practical systems exhibiting bilinear characteristics, it is anticipated that any enhancement to the parameter estimation scheme will largely depend upon the application under consideration and may well involve a combination of the schemes which have been outlined here.

5. Real-time laboratory scale trials

5.1 Introduction

In order to evaluate the effectiveness of both the enhanced linear STC and extended bilinear STC algorithms, developed in Chapters 3 and 4, real-time laboratory scale trials are undertaken.

The first system considered is the coupled tanks apparatus, marketed by TecQuipment Ltd. [66], in which the control objective is to regulate the liquid level in the secondary tank by controlling the fluid flow into the primary tank; the tanks being coupled via an orifice. The system is known to exhibit non-linearities, due to the square law relationship between output flow and liquid head and, in order to accommodate for this, use of enhanced linear STC schemes are investigated. The second system is the heating-cooling system, marketed by Flight Electronics Ltd. [67], in which the control objective is to regulate the surface temperature of a heated bar onto which air is blown from an ambient source; the air flow being regarded as a disturbance on the system. The resulting overall system is known from physical considerations to exhibit bilinear characteristics and it is used as a basis for evaluating the bilinear STC algorithms developed in Chapter 3.

5.2 Coupled tanks hydraulic system

The system under investigation in this section is the coupled tanks hydraulic system which is illustrated diagrammatically in Figure 5.1.

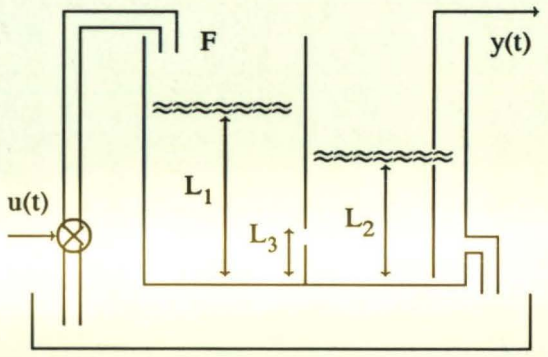


Figure 5.1. Coupled tanks apparatus.

Investigations are focused on developing STC algorithms to regulate the liquid level L2 in the secondary tank by controlling the input flow F to the primary tank. Whilst from physical considerations it is known that the system behaves in a non-linear manner, it is never the less useful to consider an approximate linear model structure. Define $L_1=\bar{L}_1+l_1$ and $L_2=\bar{L}_2+l_2$ where l_1 and l_2 are, respectively small variations in liquid level about mean operating levels $\bar{L}_1$ and $\bar{L}_2$. By applying the law of conservation to each tank an approximate linear model for determining l_1 and l_2 is readily obtained. Such a model may be expressed in the state-space form

$$\begin{bmatrix} \dot{l}_1 \\ \dot{l}_2 \end{bmatrix} = \begin{bmatrix} -k_1/A & k_1/A \\ k_1/A & -(k_1+k_2)/A \end{bmatrix} \begin{bmatrix} l_1 \\ l_2 \end{bmatrix} + \begin{bmatrix} 1/A \\ 0 \end{bmatrix} f \qquad (5.1)$$

where A is the cross sectional area of each tank, f is the variation in input flow rate about a steady-state flow $\bar{F}$ (i.e. $F=\bar{F}+f$) and k_1 and k_2 are constants of proportionality which are themselves dependent on the mean operating levels $\bar{L}_1$ and $\bar{L}_2$.

The implications of equation (5.1) are that for small variations about mean operating levels the system may be adequately modelled as a linear second order system. When considering the application of STC it is first necessary, as outlined in Chapter 2, to identify an appropriate linear model structure. For a second order system such a discrete representation takes the form

$$y(t) = -a_1 y(t-1) - a_2 y(t-2) + b_0 u(t-k) + b_1 u(t-k-1) + e(t) \quad (5.2)$$

where in the case of the coupled tanks system u(t), y(t) and e(t) are, respectively, the input flow rate, output liquid level in the secondary tank and an assumed white noise measurement disturbance. The integer $k \geq 1$ is the system time delay expressed as an integer multiple of the sampling interval. Adopting a sampling interval of T=8.75 second gives rise to a value of k=1. The corresponding state-space representation of equation (5.2), as outlined in Chapter 2, becomes

$$\mathbf{x}(t+1) = \mathbf{P}\mathbf{x}(t) + \mathbf{Q}u(t) + \mathbf{R}e(t) \quad (5.3a)$$

$$y(t) = \mathbf{H}\mathbf{x}(t) + e(t) \quad (5.3b)$$

where $\mathbf{x}(t) \in \mathbb{R}^2$ and the matrices **P**, **Q**, **R** and **H** following the necessary reduction in dimension are given by

$$\mathbf{P} = \begin{bmatrix} 0 & -a_2 \\ 1 & -a_1 \end{bmatrix} \quad \mathbf{Q} = \begin{bmatrix} b_1 \\ b_0 \end{bmatrix} \quad \mathbf{R} = \begin{bmatrix} -a_2 \\ -a_1 \end{bmatrix} \quad \mathbf{H}^T = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad (5.4)$$

Based on the assumption that variations in system behaviour, including the effects due to non-linearities, may be absorbed into the time varying nature of the adopted model parameters, equations (5.2) and (5.3) are taken to be an appropriate model structure for implementing linear STC.

5.2.1 Implementation of enhanced linear STC

For small variations in head height, standard linear STC is found to be more effective than conventional PID control [68]. However, for large variations in head height both standard linear STC and PID schemes are found to be inappropriate. In [A17] a switched model linearisation approach is introduced in an attempt to accommodate the system non-linearities and in [A16] the approach is extended to combine switched model linearisation with cautious least squares.

The performance of the various STC schemes are compared to that achieved when use is made of variable forgetting factors and covariance reset only. In each case the adopted STC makes use of the state-space pole-placement strategy [6] and in order to achieve a first order type response closed-loop poles are specified at -0.229 and -100 in the s-plane. Each test was taken over 300T second (T=8.75 second) and the system subjected to a set-point demand equivalent to a square wave reference input of period 150T second. The reference levels r(t) are specified as 10cm and 18cm of fluid head in the secondary tank, corresponding to transducer readings of 2 volt and 4 volt respectively.

Figures 5.2a and 5.2b illustrate the responses corresponding to the use of variable forgetting factors [A2,A6] and covariance reset action coincident with set-point change. It is evident from Figure 5.2a that reasonable set-point following is achieved only at the expense of a relatively high and possibly unacceptable input variance. Whilst use of reset action at set-point change should remove all *a priori* information from within the algorithm and facilitate rapid adaptation Figure 5.2b, which corresponds to this, shows that this improvement is not forthcoming, rather surprisingly leading to a deterioration in response.

Recognising the apparent shortfalls in the use of variable forgetting factors and covariance resetting techniques, when dealing with this non-linear system, prompted the need for an alternative approach and the use of switched model linearisation in conjunction with cautious least squares (CLS) is investigated [A16]. Simulation studies which make use of switched linear model structures given in [A17] are used as a basis to assess the approach. Figures 5.3a and 5.3b correspond to the use of sequential CLS and caution on reset respectively. Whilst these Figures would seem to imply comparable results in terms of set-point following and control input variance, it should be noted that the sequential approach is computationally more intensive. Real-time implementation trials corresponding to the above gives rise to the responses shown in Figures 5.4a and 5.4b. As expected, results given in Figures 5.3b and 5.4b are similar, whilst those of

Figures 5.3a and 5.4a indicate a deterioration in the real-time implementation; this being due possibly to the increased computational delay.

It is readily observed that in this application, the use of caution on reset gives rise to a superior performance in terms of increased accuracy in set-point following and reduced control input variance.

5.3 Heating-cooling system

This section describes investigative studies into the application of the bilinear STC techniques developed in Chapter 3. A laboratory scale non-linear heating-cooling system [67], which is known from physical considerations to exhibit bilinear characteristics, forms the test facility for evaluating the applicability of the bilinear approach. The system which has been used previously [69] for simulating the effects of a controlled glass house, is illustrated schematically in Figure 5.5.

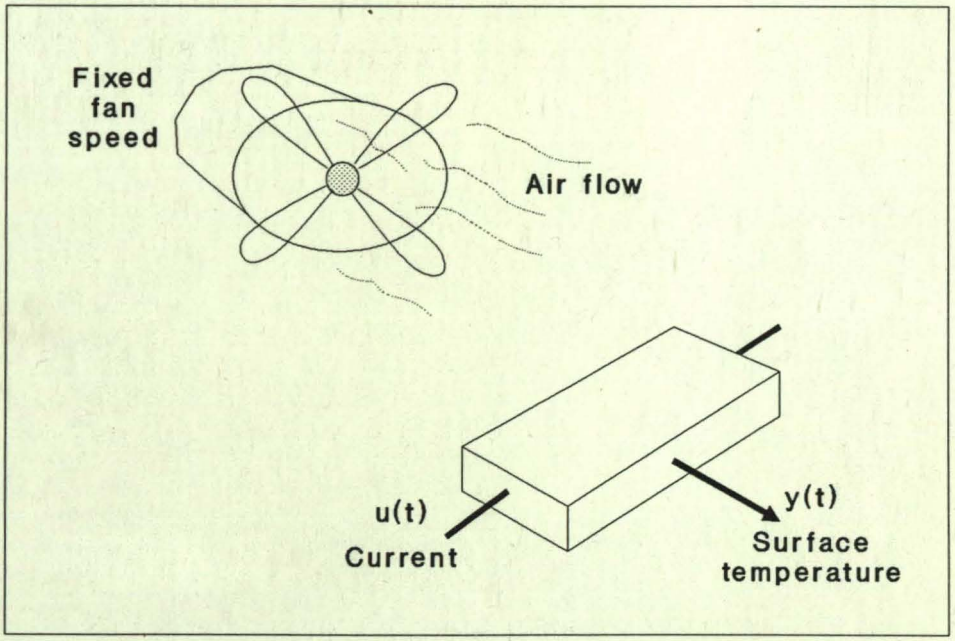


Figure 5.5. Schematic of heating-cooling system.

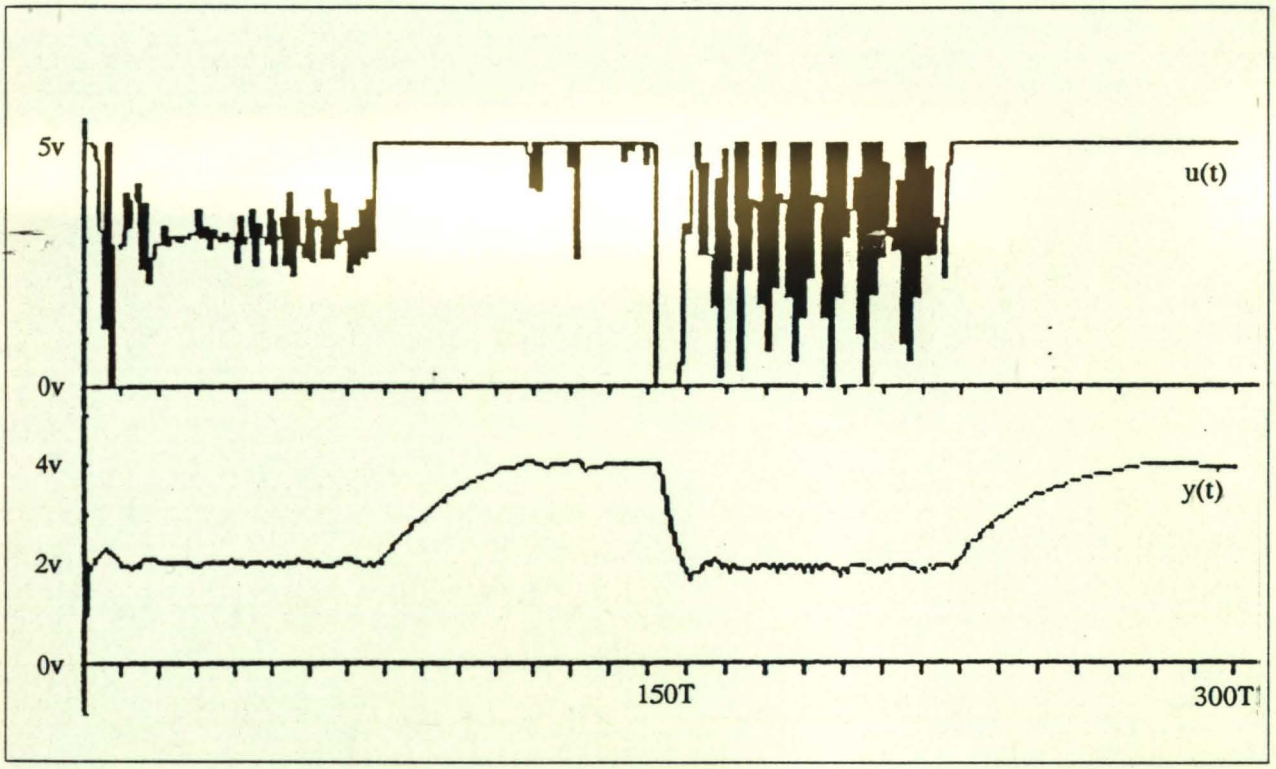


Figure 5.2(a)

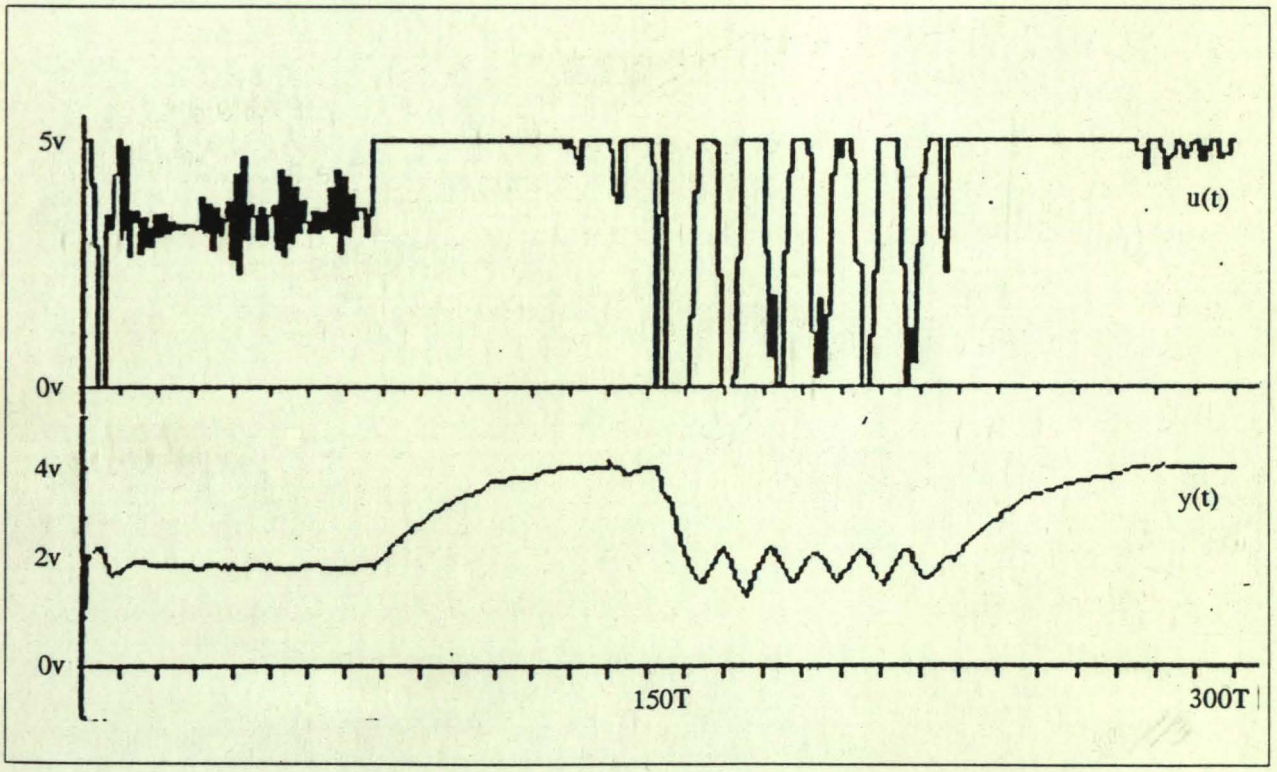


Figure 5.2(b)

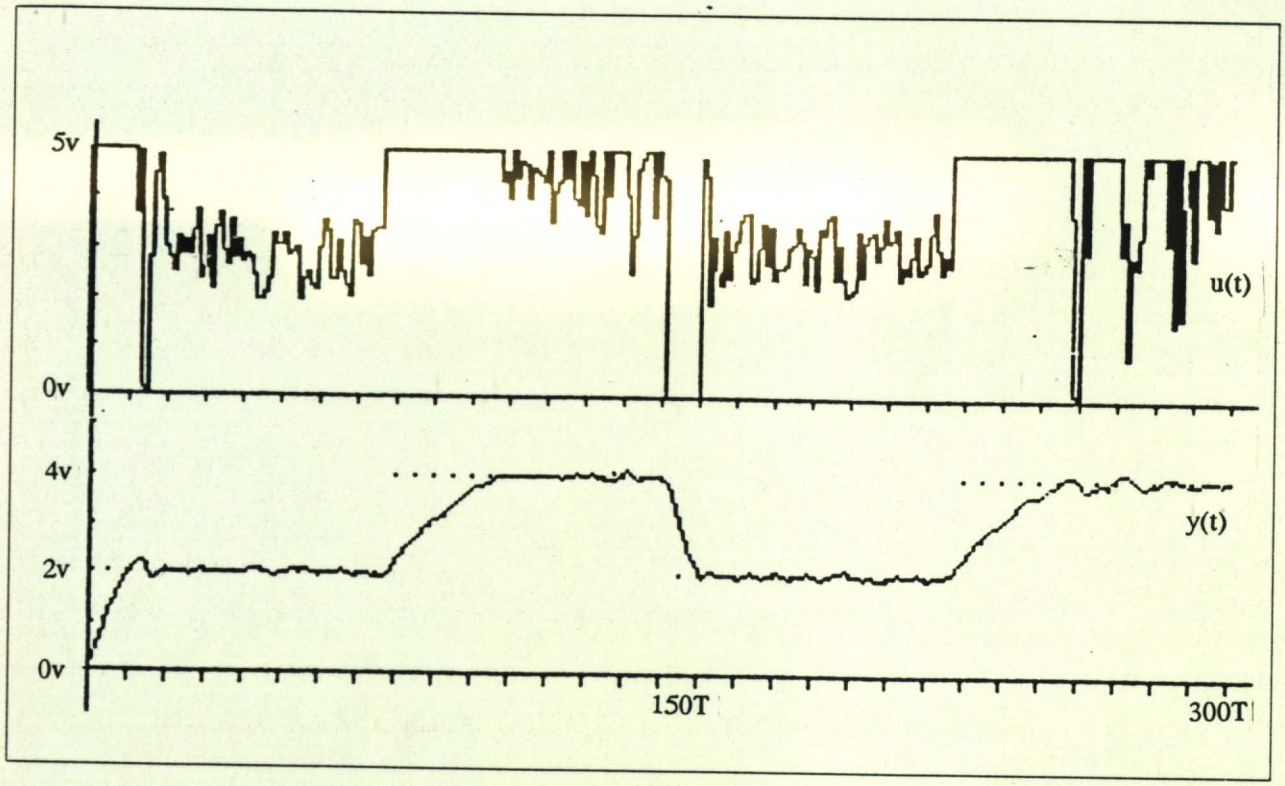


Figure 5.3(a)

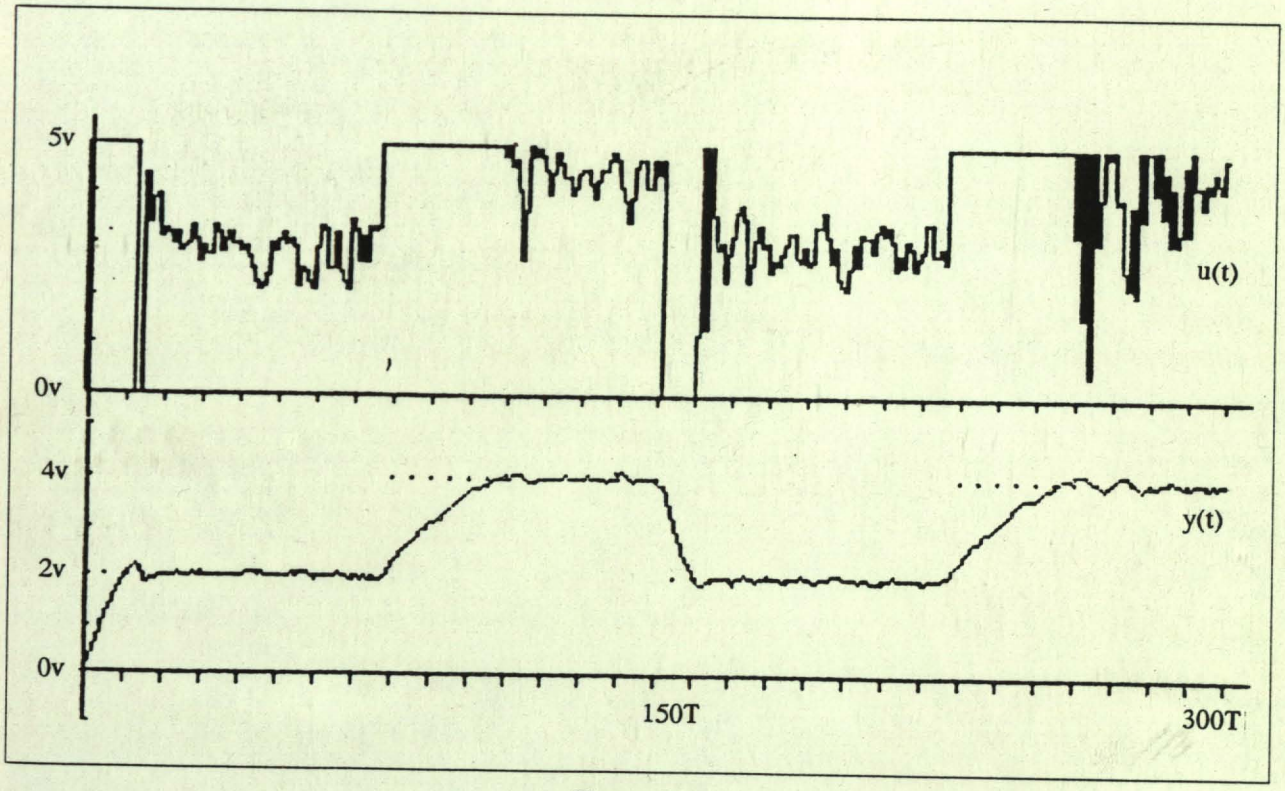


Figure 5.3(b)

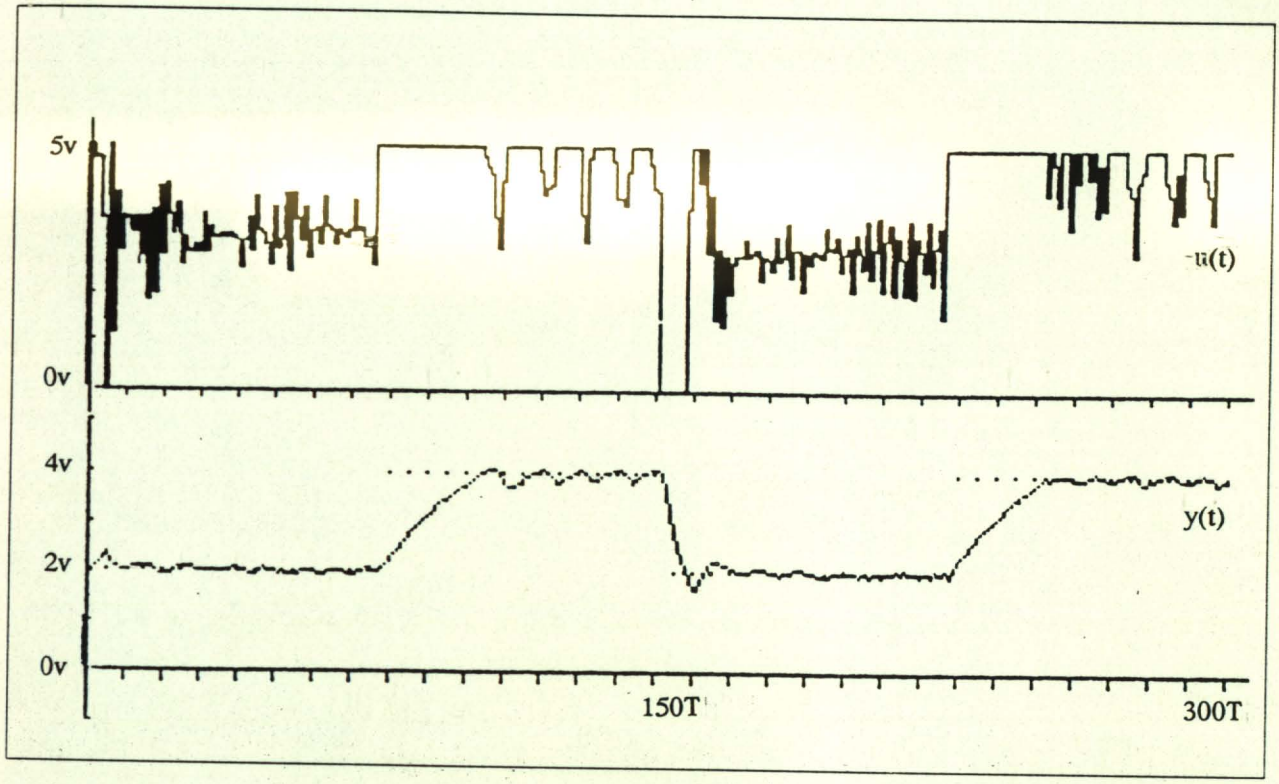


Figure 5.4(a)

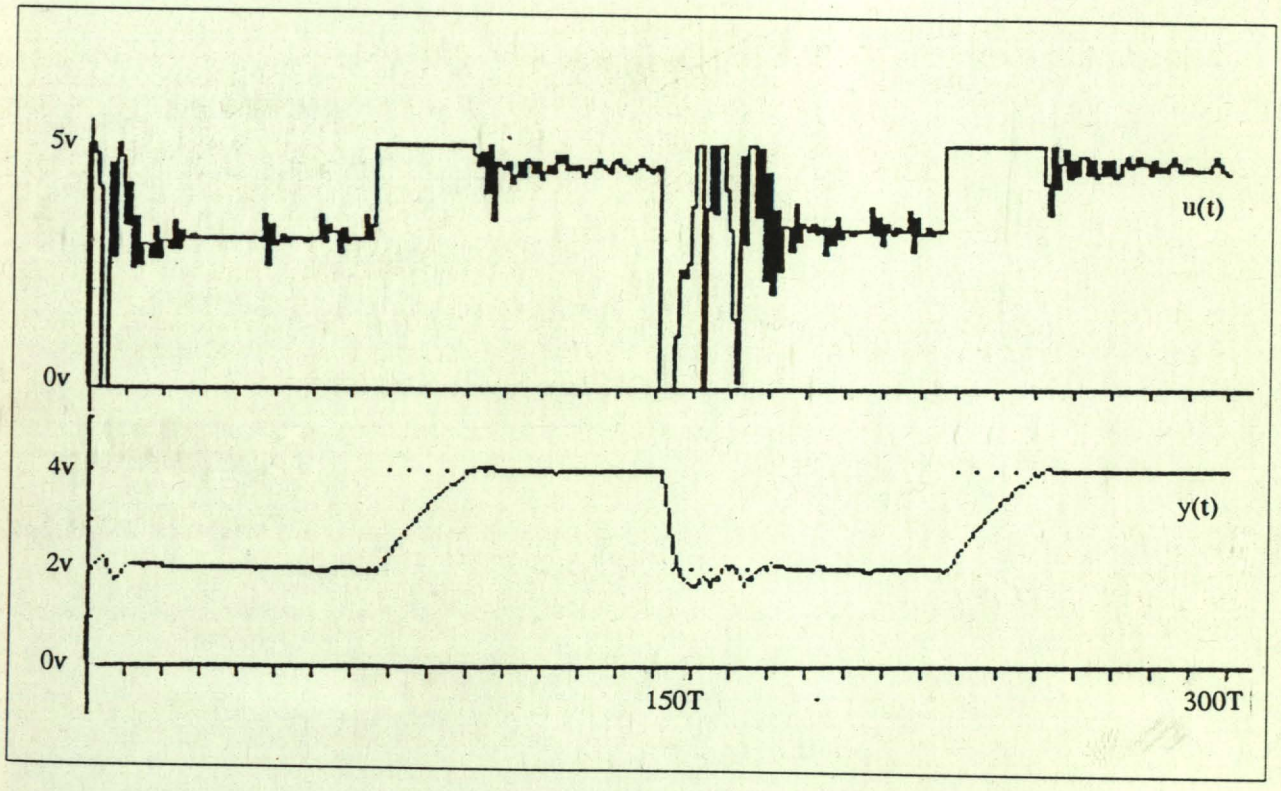


Figure 5.4(b)

The system consists of a heated bar onto which air is blown from an ambient source; the heat being largely transferred by convection. The objective of the controller is to regulate the surface temperature of the heated bar, by controlling the input current, so that it follows a prescribed temperature profile; the air flow being regarded as a load disturbance on the system.

In order to implement STC a discrete transfer function representation is required; a major step when dealing with a non-linear system is to form an appropriate model structure. The modelling stage is based on a set of assumptions which must be comprehensive enough to fully characterise the system whilst at the same time provide the framework for the simplest possible model structure. For the heating-cooling system which is known from physical considerations to exhibit bilinear characteristics, the following assumptions provide the basis for a simple first order bilinear model structure:

(i) The system obeys the first law of thermodynamics.

(ii) Heat is derived from conduction of current and heat loss is a function of temperature gradient.

(iii) Temperature gradient is a function of input current and fan speed.

(iv) For any given fan speed, input current and ambient temperature there will be some steady-state point of equilibrium.

Based on the qualitative assumptions (i)-(iv) a simple first order bilinear model is now derived.

Heat stored = Internal conduction – External convection

alternatively

$$cm\dot{y} = \bar{k}_1 u(y-\alpha) - \bar{k}_2 f(y-\alpha) \qquad (5.5)$$

where c = specific heat capacity of the resistive element/bar,

m = mass of the resistive element,

$\dot{y}$= rate of change of surface temperature of the resistive element with respect to time,

u= current input to resistor,

f = fan speed,

y= actual surface temperature of resistive element,

α= temperature of ambient surroundings

and $\bar{k}_1, \bar{k}_2$ = constants of proportionality.

Rearranging equation (5.5) leads to

$$\dot{y} = k_1 u(y-\alpha) - k_2 f(y-\alpha) \tag{5.6}$$

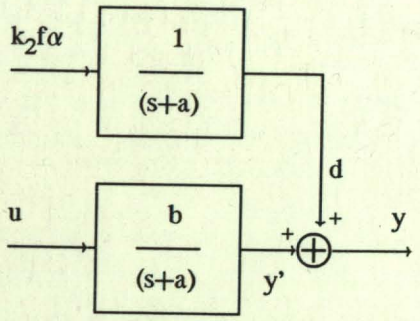
where k_1 and k_2 are $\bar{k}_1$/cm and $\bar{k}_2$/cm respectively, so that

$$\dot{y} + k_2 f y = k_1 u(y-\alpha) + k_2 f\alpha \tag{5.7}$$

Since the term involving the product of u and y cannot be separated it is convenient to introduce an element of time-step quasi-linearisation. Defining y on the right hand side of equation (5.7) to be the quantity Y, which is assumed to be constant at any given sampling instant, leads to

$$\dot{y} + k_2 f y = k_1 u(Y-\alpha) + k_2 f\alpha \tag{5.8}$$

which may be represented in the block diagram form



where $a=k_2 f$, $b=k_1(Y-\alpha)$, d is an offset and s is the Laplace variable.

The z-form transfer function from u to y' is readily shown to be

$$\frac{y'(t)}{u(t)} = \frac{z^{-1}(b/a)(1-e^{-aT})}{(1-e^{-aT}z^{-1})} \tag{5.9}$$

where T denotes the sampling interval. The corresponding difference equation takes the form

$$y'(t) = e^{-aT}y'(t-1) + (b/a)(1-e^{-aT})u(t-1) \tag{5.10}$$

substituting for a and b and including the offset term d gives

$$y(t) = e^{-k_2fT}y(t-1) + \frac{k_1(Y-\alpha)}{k_2f}(1-e^{-k_2fT})u(t-1) + d(t) \tag{5.11}$$

which on expanding leads to

$$y(t) = e^{-k_2fT}y(t-1) + \frac{k_1\alpha}{k_2f}(e^{-k_2fT}-1)u(t-1) - \frac{k_1}{k_2f}(e^{-k_2fT}-1)Yu(t-1) + d(t) \tag{5.12}$$

Substituting $a_1 = -e^{-k_2fT}$,

$$b_0 = \frac{k_1\alpha}{k_2f}(e^{-k_2fT}-1),$$

$$\eta = -\frac{k_1}{k_2f}(e^{-k_2fT}-1),$$

and replacing Y by y(t-1) yields

$$y(t) = -a_1y(t-1) + b_0u(t-1) + \eta y(t-1)u(t-1) + d(t) \tag{5.13}$$

which is readily observed to be of the required bilinear structure.

5.3.1 Implementation of bilinear STC

In order to provide a basis from which to assess the effectiveness of the proposed bilinear STC it is pertinent to first consider the application of linear and enhanced linear STC. In this section the effectiveness of self-tuning linear PID and linear pole-placement strategies are compared to self-tuning bilinear PID and bilinear pole-placement strategies.

Whilst the simple first order bilinear model structure, equation (5.13), has been derived from physical principles it is found in practice that use of the second order bilinear model

$$y(t) = -a_1 y(t-1) - a_2 y(t-2) + b_0 u(t-1) + \eta y(t-1)u(t-1) + d(t) \qquad (5.14)$$

leads to a more robust STC scheme and is adopted here for all real-time trials.

Tests on the heating-cooling system are run over 150 iterations with a sampling interval of T=6 second and the surface temperature is recorded as a percentage of full scale (70°C).

In the case of the PID controller, a discrete form of the ideal non-interacting algorithm

$$u(t) = k_p \left(z(t) + k_i \int z(t)dt + k_d \frac{dz(t)}{dt} \right)$$

is adopted [70], in which

$$k_p = 1/b_0, \qquad k_i = (Tb_0)/a_1, \qquad k_d = a_2(Tb_0)$$

with T being the sampling interval and z(t) the error (r(t)-y(t)). The philosophy of the bilinear PID approach is simply based on the fact that by estimating the bilinear coefficient the linear coefficients a_1, a_2, b_0 are obtained more accurately.

In the case of the pole-placement strategies, both the linear and bilinear

control laws attempt to place a dominant closed-loop pole at 0.1 in the z-plane. When using a second order model structure the second pole is specified at the origin.

To form a basis from which to compare the various STC strategies a fixed PID controller is tuned manually and the resulting 'satisfactory' performance is recorded. The system is run over a 1200 second period (200 iterations) and is subjected to a square wave temperature reference profile switching from 27.5% to 31.4% of full scale at a period of 600 second. The system response and control input corresponding to the fixed PID scheme is illustrated in Figure 5.6. It is evident from this Figure that the system response is fairly well regulated but at the expense of a very active control actuation signal.

The responses given in Figures 5.7 and 5.8 correspond respectively to typical results obtained when use is made of the first and second order linear pole-placement schemes. In both cases the linear STC is found to be rendered inadequate. Figures 5.9 and 5.10 correspond to the use of the first order and second order bilinear pole-placement schemes. In contrast to the results shown in Figures 5.7 and 5.8, the bilinear STC is able to maintain a consistent controller performance, the second order controller producing the smoother control action. The bilinear STC clearly achieves its objective and produces a markedly superior performance over the linear STC. Figure 5.11 illustrates the system response and control input corresponding to the adaptive bilinear PID scheme, this again producing a superior performance to the linear STC.

The results presented in this section are representative of typical results obtained in real-time and they serve to highlight the potential significant benefits to be gained by adopting the bilinear approach.

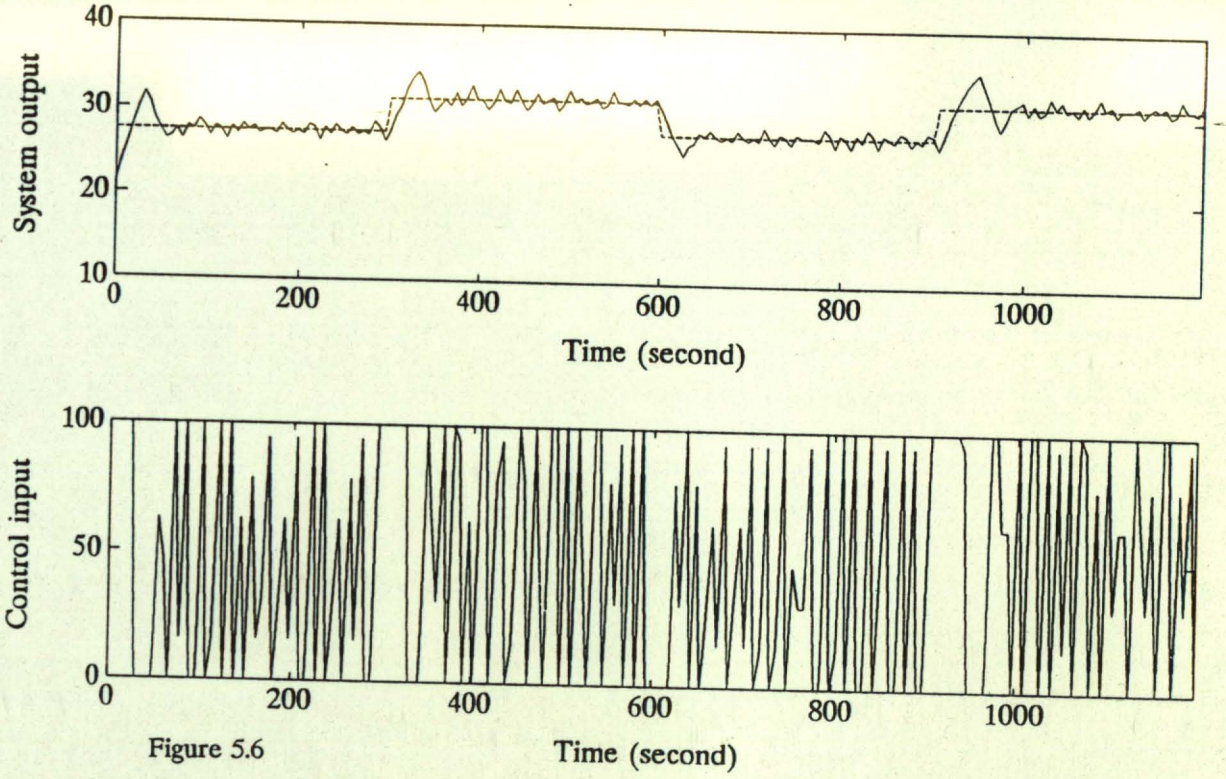
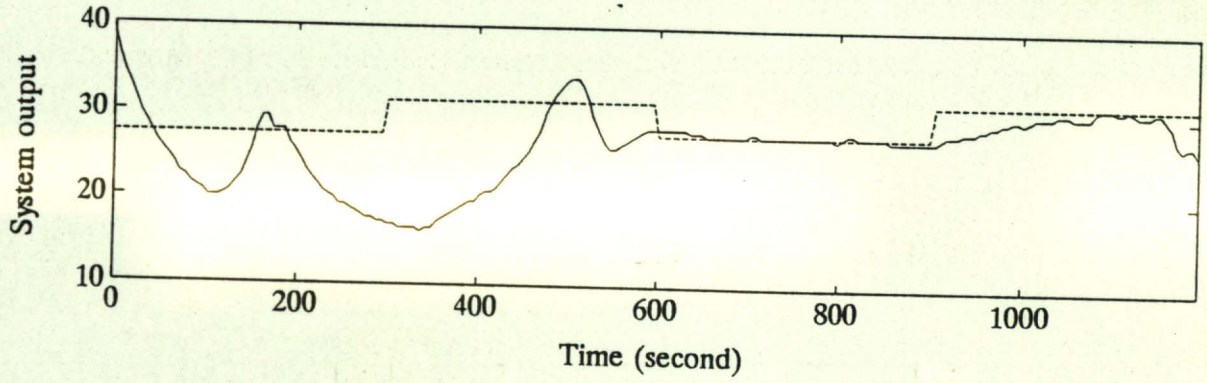

System output
40
30
20
10
0
200
400
600
800
1000
Time (second)
Control input
100
50
0
0
200
400
600
800
1000
Time (second)

Figure 5.6



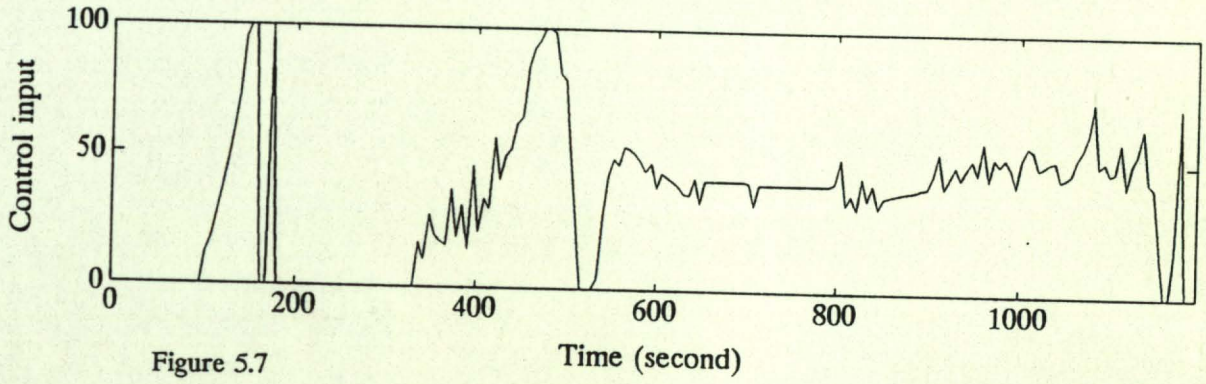
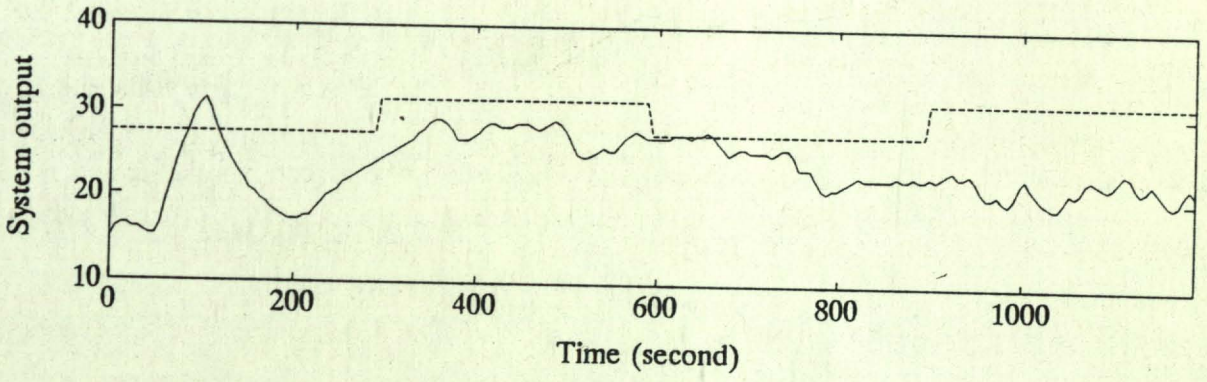


Figure 5.7



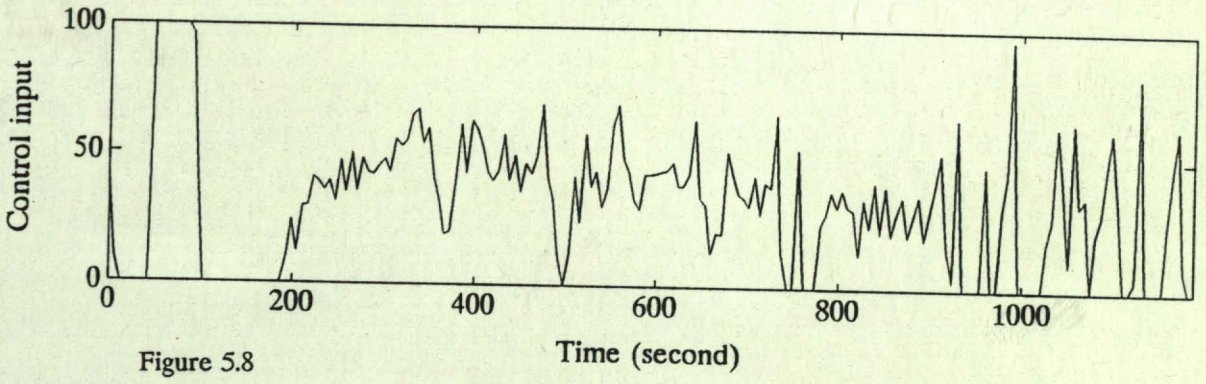
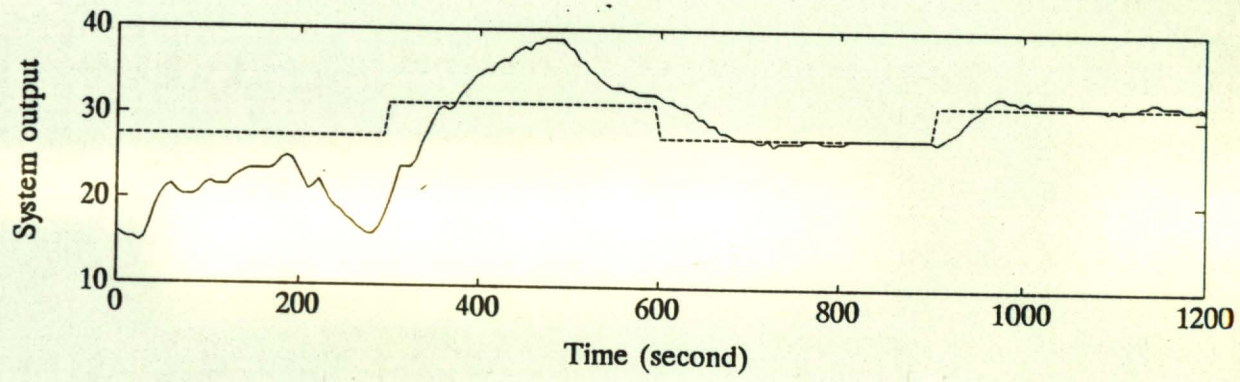


Figure 5.8



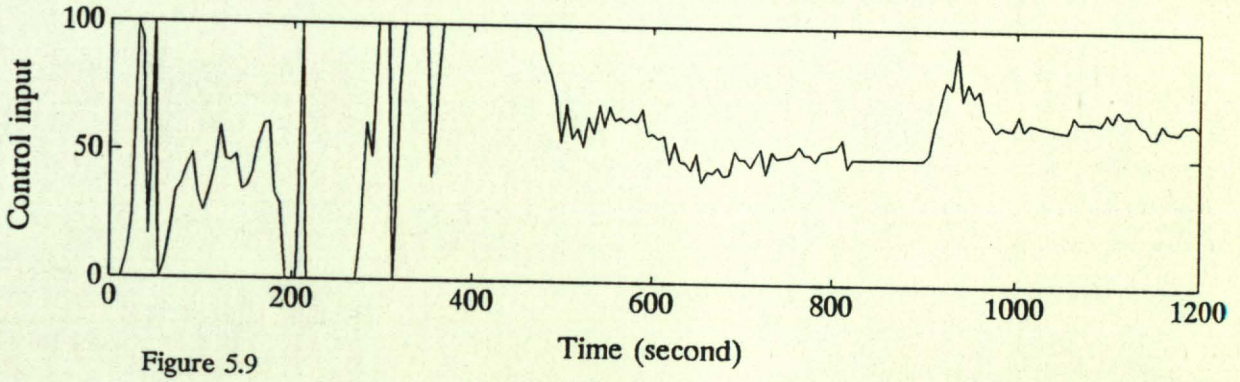
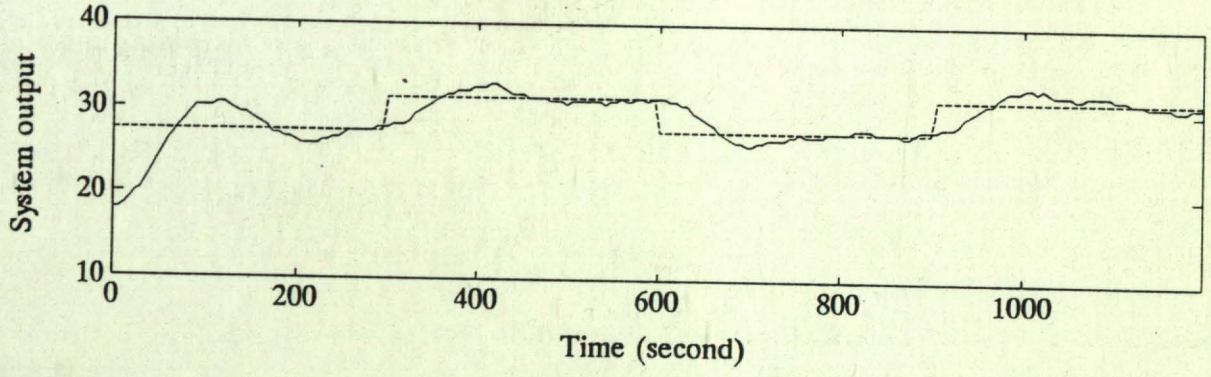


Figure 5.9



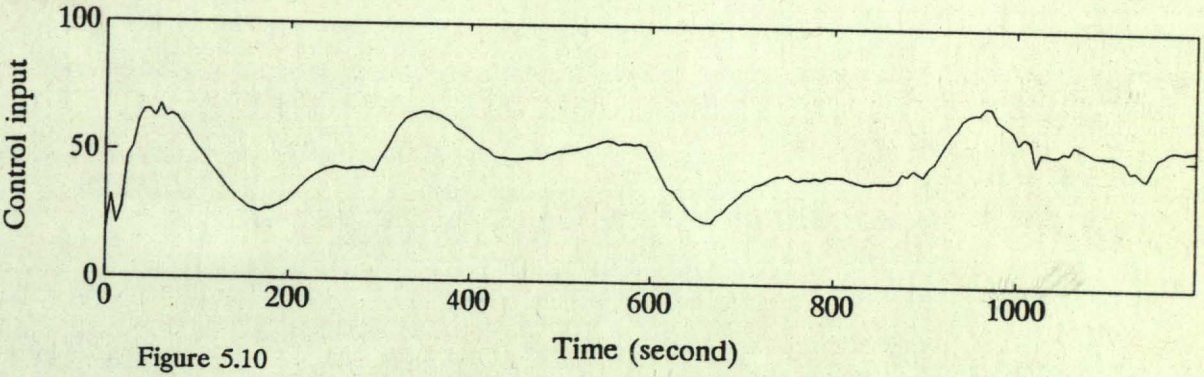


Figure 5.10

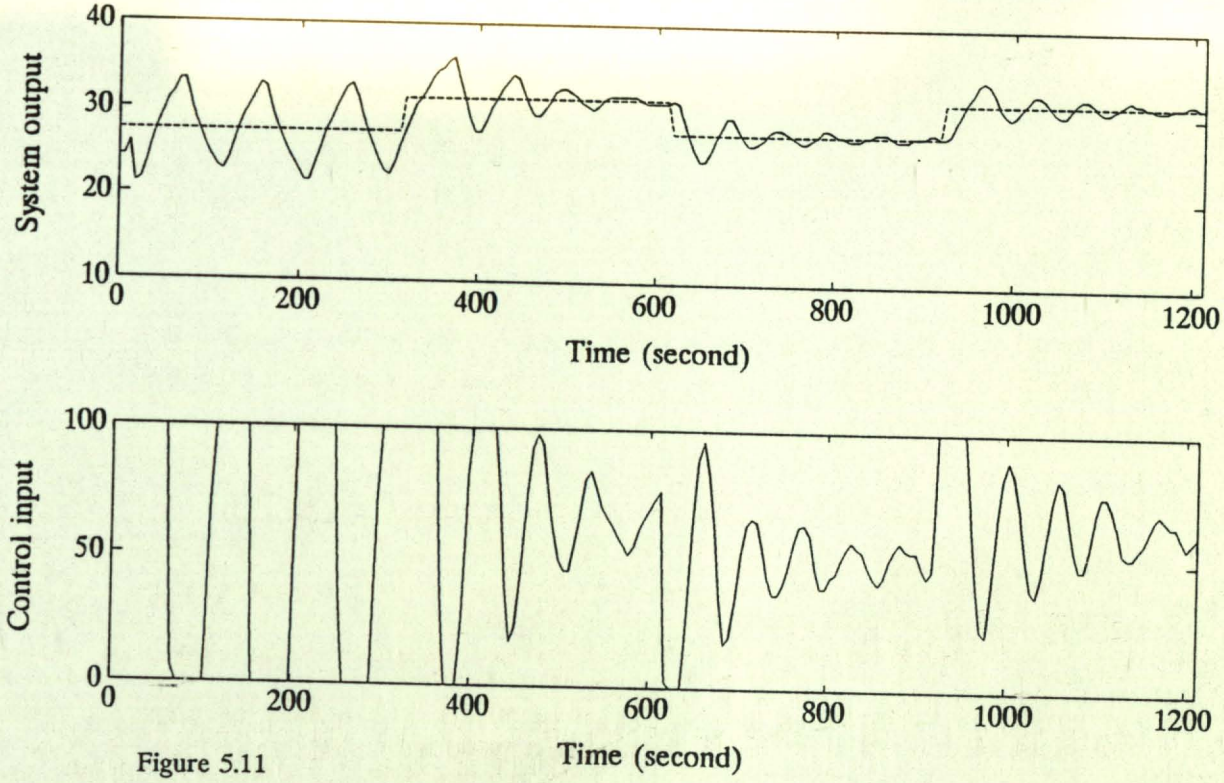

System output
40
30
20
10
0
200
400
600
800
1000
1200
Time (second)
Control input
100
50
0
0
200
400
600
800
1000
1200
Time (second)

Figure 5.11

5.4 Concluding remarks

The effectiveness of both the enhanced linear STC and the bilinear STC schemes have been investigated via real-time trials involving two laboratory based non-linear systems.

In the case of the coupled tank system, the performance of the standard linear STC is found to be ineffective when assumptions on local linearity are invalid (i.e. the performance deteriorates when there are large variations in head height). However, it is found that when use is made of enhanced linear STC schemes making use of switched model linearisation combined with a cautious least squares procedure, significant improvements are possible.

In the case of the heating-cooling system which is known from physical consideration to exhibit bilinear characteristics the enhanced linear STC is found to be ineffective and the bilinear STC scheme is found to produce a markedly superior performance. Further work has since been carried out and the MISO bilinear STC scheme has been applied to the heating-cooling system. It has been found that, by carefully tailoring the weighting on each control input channel further improvements in performance are possible [71].

The results obtained using the bilinear approach are very encouraging and indicate that the bilinear STC may well be applicable for a wider class of non-linear systems for which local bilinearity may be assumed. It is anticipated that the form of the bilinear STC will have to be tailored for each specific application with use being made, where appropriate, of the enhanced features outlined in Chapter 4.

6. Discussion and conclusions

Conclusions

The major problem of the standard linear self-tuning approach is believed to be that of plant/model mismatch. In adopting the standard approach the designer is forced to adopt a linear model structure which presents the 'best' available compromise albeit limited, due to the current state of the art in microcomputer technology, between open-loop model accuracy and closed-loop computational intensity. Consequently, the plant/model mismatch problem is inevitable and when attempting to apply linear self-tuning control to systems which exhibit time varying dynamic behaviour or to systems for which local linearity may not hold, the widely accepted assumption on absorption of these effects into the estimated parameters of an adopted linear model is rendered invalid thus effectively prohibiting use of the standard self-tuning approach.

Prompted by the need to alleviate the shortfalls encountered when attempting to apply the standard linear self-tuning approach to systems exhibiting non-linear and/or time varying dynamic behaviour, the linear self-tuning framework has been extended to accommodate a class of SISO and MISO bilinear model structures; such structures being appropriate for a wider range of systems than the often inadequate yet convenient linear model structures. Bilinear control systems represent a well structured class of nearly linear systems in which the non-linearity or bilinearity is characterised by additional multiplicative terms involving products of states and controls. Such systems occur naturally in a wide variety of engineering, biomedical and socioeconomic applications. Engineering applications include; fermentation processes, disc braking systems, AC and DC machines, distillation columns and heat transfer processes.

Attention has been focused towards the development of an extended form of

the linear self-tuning pole-placement strategy and, recognising that all self-tuners make use of discrete-time transfer function models, a number of alternative quasi-linear candidates for the bilinear self-tuning control algorithm have been explored. The resulting bilinear self-tuning pole-placement controller makes use of two model structures: a polynomial bilinear model structure for parameter estimation, with use being made of an extended form of the standard recursive least squares algorithm; and a quasi-linearised state space model structure for state estimation, with use being made of an iterated steady-state observer. Due to the presence of the bilinear product term(s), the two estimation algorithms are required to be operated serially in tandem. Essentially, with use being made of a boot strapping technique, the estimated states are progressed to the recursive least squares algorithm in order to estimate the bilinear model parameters and similarly the estimated parameters are progressed to the iterated steady-state observer in order to estimate the states.

Preliminary simulation studies revealed that the bilinear self-tuner offers potential advantages over the linear self-tuning schemes when applied to systems exhibiting bilinear characteristics; this being believed to be due possibly to the reduced plant/model mismatch problem. However it is found that the bilinear self-tuning controller is susceptible to noise and is vulnerable to numerical difficulties within the algorithms. This being particularly the case during transient periods when excessive control action has detrimental effects and during prolonged periods of steady-state operation when a well regulated plant can give rise to a near singular situation. This potential shortfall in the approach gave rise to the search for and development of enhanced parameter and joint parameter/state estimation algorithms; with investigations being specifically directed towards improving the robustness of the bilinear self-tuning controller. A number of enhancements have been investigated and whilst these have by necessity been considered for use within the bilinear self-tuning framework they are equally applicable, although will not generally produce the same degree of improvement, for linear systems. More importantly recognising the shortfalls in the standard approach, it is believed that the

enhanced procedures developed for bilinear systems will be appropriate when attempting to apply linear self-tuning techniques to practical non-linear systems when it is desirable to minimise the inevitable mismatch problems found in practice.

The most significant improvement in terms of improving the integrity of the bilinear self-tuning controller has been that of cautious least squares. Essentially by defining a safe set of parameter values over a particular operating range of interest, the designer is able to influence the estimation algorithm through a practical knowledge of the system. The cautious least squares procedure is realised as a separate sub-algorithm which is inserted between the extended recursive least squares parameter estimation algorithm and the pole-placement control law design algorithm. Caution may be applied either sequentially or cyclically at each time step or on fault detection coincident with covariance matrix reset; the effect being to realign the estimated parameter vector towards the appropriate safe set with the amount of realignment being dependent on the confidence in the safe set values. An additional feature which arises in the use of cautious least squares is that it provides an element of excitation within the estimation algorithm without physically disturbing the plant; this effectively being achieved by repeated use of the artificial observation data in the form of orthogonal unit vectors.

Real-time trials involving two laboratory scale non-linear systems have been undertaken with use being made of the bilinear self-tuning controller and a linear self-tuning controller incorporating the enhanced estimation procedures. It is found that when applied to the hydraulic liquid level system, for which local linearity cannot be assumed due to the large variations in operating head height, the standard linear self-tuning controller is rendered inadequate. However, when use is made of switched model linearisation combined with cautious least squares, significant improvements in overall performance are possible. In the case of the heating-cooling system, for which a simple bilinear model structure has been derived from first principles, the enhanced linear self-tuner is found to be inadequate. The bilinear self-tuning controller, however, is able

to satisfactorily control the system, with further improvements being made possible when use is made of the enhanced estimation procedures.

The overall results of the real-time trials serve to illustrate that significant improvements in performance are possible when known/identified non-linearities (bilinearities) are 'actively' accommodated at the design stage within an extended self-tuning framework rather than being assumed to be 'passively' absorbed into the parameters of a linear model structure during implementation. Furthermore, since the range of applicability of the linear self-tuner is a subset of that of the bilinear self-tuner, it may be conjectured that, in the same way the linear self-tuner is valid where local linearity holds, the bilinear self-tuner may well be applicable for a wider class of non-linear systems where local bilinearity holds.

It should be noted that use of the bilinear self-tuning controller introduces additional model parameters which are required to be recursively identified, thus increasing computational complexity of the algorithm. To allow the full advantage of potential improvements offered by such controllers a good *a priori* knowledge of the system to be controlled is required in order to ensure that only relevant bilinear terms are included in the algorithm, therefore reducing the computational burden to a minimum.

Current research

Whilst there is interest in theoretical aspects of identification and parameter estimation for bilinear systems, as is evident in the recent publications [72-77], the number of reported applications of control strategies to such systems remains limited. However, extended schemes, including those incorporating predictive forms [78,79], of the basic bilinear self-tuning framework proposed in this Thesis are currently being successfully applied to a gas fired industrial heat treatment furnace [80-83]. In terms of

efficient energy usage, the performance of these extended bilinear self-tuning schemes are markedly superior to that achieved using commercial auto-tuning PID controllers; such controllers currently being regarded as the accepted industrial standard.

A further area of research which has been highlighted is that of applying self-tuning control to other forms of non-linear systems. In [84-87] enhanced linear self-tuning strategies are being applied to the dynamometer torque loop of an engine test cell. Such systems are typical of a wide range of industrial plant, which exhibit fast dynamics and non-linear behaviour, and it is found that in order to capture the process dynamics it is necessary to make use of enhanced estimation techniques.

Another area of on-going work is that of the automobile industry where the reducing size and cost and increasing reliability of microprocessor hardware is continually posing new challenges. The industry is already facing a revolution in terms of drive by wire technology. In particular fundamental research on the microprocessor controlled advanced suspension systems is currently taking place [88], as well as the fault detection mechanism [89] in readiness for the suspension hardware which is yet to emerge.

Combining the on-going research activities [A18,A19,A20] with the potential benefits offered by the bilinear self-tuning framework it is clear that self-tuning control is already becoming a well established approach and promises to provide a powerful and versatile option for the control of an increasingly widening range of industrial applications.

Whilst initial investigations on the stability and convergence properties of the iterated steady-state observer for bilinear systems have already been undertaken, there remains much scope in this direction for further theoretical work. This is particularly relevant when considering the extension of the bilinear approach to accommodate interconnected multiple-input single-output bilinear subsystems.

In addition the cautious least squares aspect of the work requires further evaluation on industrial systems and in this context the information matrix filtering approach, with its increased facilities for greater selectivity, needs to be pursued.

It is envisaged that further research should also be directed towards the design and development of hybrid schemes involving self-tuning control in association with other schemes in an attempt to provide robustness over a wider operating range whilst maintaining optimum performance. Such hybrid schemes should, therefore, aim to incorporate either the robustness offered by H_∞ designs, the learning capabilities of a neural network or the inferential logic of a fuzzy control scheme. This together with the desirable features of a self-tuning controller should provide for an increased range of industrial applicability.

The continuing developments in computer technology coupled with the increasing demands for improved system performance will almost certainly continue to stimulate the search for new adaptive controller designs.

oooOooo

References

1. Kalman, R.E., Design of a self-optimising control system, Trans. ASME, **80**, 468-478, 1958
2. Peterka, V., Adaptive digital regulation of noisy systems, Proc. IFAC Symp. on Identification and Process Parameter Estimation, Prague, 1970
3. Astrom, K.J., and Wittenmark, B., On self-tuning regulators, Automatica, **9**, 185-199, 1973
4. Clarke, D.W., and Gawthrop, P.J., Self-tuning controller, Proc. IEE, **122**, 929-934, 1975
5. Wellstead, P.E., Prager, D.L., and Zanker, P.M., Pole-assignment self-tuning regulator, Proc. IEE, **126**, 781-787, 1979
6. Warwick, K., Self-tuning regulators - a state-space approach, Int. J. Control, **33**, 839-858, 1981
7. Lam, K.P., Implicit and explicit self-tuning controllers, D.Phil Thesis, Oxford Univ., 1980
8. Plackett, R.L., Some theorems in least squares, Biometrika, **37**, 149-157, 1950
9. Harris, C.J., and Billings, S.A., (Eds.) Self-tuning and adaptive control, IEE Control Engineering Series, **15**, (2nd Edn.) Peter Peregrinus, 1985
10. Chalam, V.V., Adaptive control systems, Marcel Dekker, New York, 1987
11. Warwick, K., (Ed.) Implementation of self-tuning controllers, IEE Control Engineering Series, **35**, Peter Peregrinus, 1988
12. Bell, E.T., Men of mathematics, Simon and Schuster, New York, 1937
13. Sorenson, H.W., Least squares estimation: From Gauss to Kalman, IEEE Spectrum, 63-68, 1970
14. Cadzow, J.A., and Martens, H.R., Discrete-time and computer control systems, Prentice Hall, 1970
15. Hsia, T.C., System Identification, Lexington Books, 1977

16. Astrom, K.J., Self-tuning regulators, design principles and applications, (in Narendra, K.S., and Monopoli, R.V., Eds.) Applications of adaptive control, Academic Press, 1980

17. Fortescue, T.R., Kershenbaum, L.S., and Ydstie, B.E., Implementation of self-tuning regulators with variable forgetting factors, Automatica, **17**, 831-835, 1981

18. Wellstead, P.E., and Sanoff, S.P., Extended self-tuning algorithm, Int. J. Control, **34**, 433-455, 1981

19. Warwick, K., Self-tuning property of state-space self-tuners, IEE Proc., **129**, Pt. D., 96-100, 1982

20. Caines, P.E., Relationship between Box-Jenkins-Astrom control and Kalman linear regulator, Proc. IEE, **119**, 615-620, 1972

21. Bell, D.J., Cook, P.A., and Munro, N., Design of modern control systems, IEE Control Engineering Series, **20**, Peter Peregrinus, 1982

22. Young, P.C., and Willems, J.C., An approach to the linear multivariable servomechanism problem, Int. J. Control, **15**, 961-979, 1972

23. MacFarlane, A.G.J., Return-difference and return-ratio matrices and their use in analysis and design of multivariable feedback control systems, Proc. IEE, **117**, 2037-2050, 1970

24. Barnett, S., Introduction to mathematical control theory, Clarendon Press, Oxford, 1975

25. Astrom, K.J., and Eykhoff, P., System Identification: A survey, Automatica, **7**, 123-162, 1971

26. Daley, S., Application of a fast self-tuning control algorithm to a hydraulic test rig, Proc. Instn. Mech. Engrs., **201**, 285-295, 1987

27. Goodhart, S.G., Burnham, K.J., and James, D.J.G., Self-tuning control of a non-linear industrial test rig, Proc. Systems Science X, Wroclaw, Poland, 19-22 September 1989

28. Mohler, R.R., Natural bilinear control processes, IEEE Trans. Systems Science and Cybernetics, **6**, 192-197, 1970

29. Mohler, R.R., Bilinear control processes, Mathematics in Science and Engineering, **106**, Academic Press, 1973

30. Bruni, C., Di Pillo, G., and Koch, G., Bilinear systems: An appealing class of 'nearly linear' systems in theory and applications, IEEE Trans. Aut. Control, **19**, 334-348, 1974

31. Williamson, D., Observations of bilinear systems with application to biological control, Automatica, **13**, 243-254, 1977

32. Espana, M., and Landau, I.D., Reduced order bilinear models for distillation columns, Automatica, **14**, 345-355, 1978

33. Biran, Y., and McInnis, B., Optimal control for bilinear systems: Time-varying effects of cancer drugs, Automatica, **15**, 325-329, 1979

34. Figalli, G., La Cava, M., and Tomasi, L., An optimal feedback control for a bilinear model of induction motor drives, Int. J. Control, **39**, 1007-1016, 1984

35. Banks, S.P., Mathematical theories of non-linear systems, Prentice Hall International, 1988

36. Mohler, R.R., Barton, C.F., and Karanam, V.R., BLS identification by orthogonal functions with applications to immunology, Proc. IEEE Conf. on Decision and Control, 143-147, 1976

37. Baheti, R.S., Mohler, R.R., and Spang, H.A., Second-order correlation method for bilinear system identification, Proc. IEEE Conf. on Decision and Control, 1119-1125, 1977

38. Frick, P.A., and Shadnian Valari, A., Estimation and identification of bilinear systems, Automatic Control Theory and Applications, **6**, 1-7, 1978

39. Baheti, R.S., Mohler, R.R., and Spang, H.A., A new cross-correlation algorithm for Volterra kernel estimation of bilinear systems, IEEE Trans. Aut. Control, **24**, 661-664, 1979

40. Baheti, R.S., Mohler, R.R., and Spang, H.A., A new approach to Volterra kernel estimation of bilinear systems, Proc. IEEE Conf. on Decision and Control, 546-550, 1979

41. Gabr, M.M., and Subba Rao, T., On the identification of bilinear systems from operating records, IFAC Symp. Theory and Applications of Digital Control, New Delhi, India, 375-380, 1982

42. Goka, T., Tarn, T.J., and Zaborszky, J., On the controllability of a class of discrete bilinear systems, Automatica, 9, 615-622, 1973

43. Slemrod, M., Stabilisation of bilinear control systems with applications to non-conservative problems in elasticity, SIAM J. Control and Optimisation, 131-141, 1978

44. Grasselli, O.M., Isidori, A., and Nicolo, F., Output regulation of a class of bilinear systems under constant disturbances, Automatica, **15**, 189-195, 1979

45. Murthy, D.N.P., Controllability of a discrete time bilinear system, IEEE Trans. Aut. Control, **24**, 974-975, 1979

46. Gutman, P.O., Stabilising controllers for bilinear systems, IEEE Trans. Aut. Control., **26**, 917-922, 1981

47. Soto, J.C., De La Sen, M., and Malaina, J.L., Simulation algorithm for optimal control of a class of composite bilinear systems, in Hamza M.H. (Ed.) Proc. IASTED Conf. Applied Informatica, Grindelwald, Switzerland, ACTA Press, Zurich, 95-98, 1985

48. Gounardis-Minaidis, C.H., and Kalouptsidis, N., Stability of discrete time bilinear systems with constant inputs, Int. J. Control, **43**, 663-669, 1986

49. Banks, S.P., and Yew, M.K., On the optimal control of bilinear systems and its relation to Lie algebras, Int. J. Control, **43**, 891-900, 1986

50. James, D.J.G., Maghsoodi, Y. and Shields, D.N., The design and numerical derivation of a stable state observer for multivariable bilinear systems, in Hamza, M.H., (Ed.) Proc. IASTED Int. Symp. Robotics and Automation, Lugano, Switzerland, ACTA Press, Zurich, 212-215, 1985

51. De La Sen, M., A model reference adaptive control system for discrete multivariable bilinear systems with interconnected subsystems, IEE Proc., **133**, Pt. D., 165-171, 1986

52. Shah, S.L., Recursive least squares estimation schemes, IEE Colloquium on Advances in Adaptive Control, April 1986

53. Balmer, L., Short term spectral estimation with applications, PhD Thesis, Warwick Univ., U.K., 1986

54. Hagglund, T., New estimation techniques for adaptive control, PhD Thesis, Lund Univ., Sweden, 1983

55. Goodwin, G.C., and Sin, K.S., Adaptive filtering, prediction and control, Prentice Hall, Englewood Cliffs, New Jersey, U.S.A., 1984

56. Hunt, K.J., A survey of recursive identification algorithms, Trans. Inst. Measurement and Control, **18**, 273-278, 1986

57. Young, P.C., Recursive approaches to time series analysis, IMA Bulletin, May/June, 209-224, 1974

58. Randall, A., The extended Kalman filter and its implementation in the steel rolling industry, M.Sc. Thesis, Coventry Polytechnic, U.K., 1988

59. King, P.J., Identification for bilinear systems, B.Eng. Thesis, Coventry Polytechnic, U.K., 1988

60. Reeve, P.J., Adaption methods, Internal Report No. PJRADAP, GEC Electrical Projects Ltd., Rugby, U.K., 1985

61. Albert, A., and Sittler, R.W., A method of computing least squares estimators that keep up with the data, SIAM J. Control, **3**, 384-417, 1966

62. Kalman, R.E., A new approach to linear filtering and prediction problems, Trans. ASME J. Basic Engineering, **82**, 35-45, 1960

63. Kalman, R.E., and Bucy, R.S., New results in linear filtering and prediction theory, Trans. ASME J. Basic Engineering, **83**, 95-108, 1961

64. Nelson, N.W., and Stear, E., The simultaneous on-line estimation of parameters and states in linear systems, IEEE Trans. Aut. Control, **21**, 94-98, 1976

65. Bierman, G.J., Factorisation methods for discrete sequential estimation, Academic Press, 1977

66. Wellstead, P.E., (Ed.) Coupled tanks apparatus, CE5, Manual, TecQuipment Ltd., 1981

67. Turner, D., (Ed.) Microelectronics applications, Users Manual, Flight Electronics Ltd., 1986

68. Goodhart, S.G., Self-tuning control applied to a non-linear hydraulic system, B.Eng. Thesis, Coventry Polytechnic, U.K., 1988

69. Young, P.C., Behzadi, M.A., Wang, C.L., and Chotai, A., Direct digital and adaptive control by input-output state variable feedback pole assignment, Int. J. Control, **47**, 1867-1881, 1987

70. Warwick, K., Adaptive deadbeat control of stochastic systems, Int. J. Control, **44**, 651-663, 1986

71. Emmerson, R.A., Self-tuning control for non-linear systems - a bilinear approach, B.Eng. Thesis, Coventry Polytechnic, U.K., 1990

72. Bielinska, E.M., Minimum variance bilinear prediction, Proc. 11th IFAC World Congress, Tallinn, Estonia, U.S.S.R., **3**, 35-40, 1990

73. Dorissen, H.T., A method for bilinear system identification, Proc. 11th IFAC World Congress, Tallinn, Estonia, U.S.S.R., **3**, 186-191, 1990

74. Lessi, O., Recursive and non-parametric methods for the identification of bilinear systems, Proc. 11th IFAC World Congress, Tallinn, Estonia, U.S.S.R., **3**, 233-238, 1990

75. Dorrisen, H.T., and Schwarz, H., Deterministic and stochastic methods for identification of bilinear systems, Proc. IMACS-IFAC Symp. Modelling and Control of Technological Systems, Lille, France, **2**, 488-493, 1991

76. Akhmonch, M., and Hmamed, A., Analysis and parameter identification of bilinear time delay systems via Fourier Series, Proc. IMACS-IFAC Symp. Modelling and Control of Technological Systems, Lille, France, **2**, 494-498, 1991

77. Hespel, Ch., and Jacob, G., Bilinear approximation of non-linear dynamic systems, Proc. IMACS-IFAC Symp. Modelling and Control of Technological Systems, Lille, France, **2**, 609-615, 1991

78. Clarke, D.W., Mohtadi, C., and Tuffs, P.S., Generalised predictive control, Parts 1 & 2, Automatica, **23**, 137-160, 1987

79. Day, R.J., and Burnham, K.J., Self-tuning minimum variance controller with adaptive weightings, Proc. 8th Int. Conf. on Systems Engineering, Coventry Polytechnic, U.K., 1991

80. Goodhart, S.G., Burnham, K.J., and James, D.J.G., Self-tuning control for industrial heating plant - a bilinear approach, Proc. 7th Int. Conf. on Systems Engineering, Univ. of Nevada, Las Vegas, U.S.A., 358-365, 1990

81. Goodhart, S.G., Burnham, K.J., and James, D.J.G., A bilinear self-tuning controller for industrial heating plant, Proc. IEE Conf. Control '91, Edinburgh, U.K., **2**, 779-783, 1991

82. Goodhart, S.G., Burnham, K.J., and James, D.J.G., Self-tuning control of non-linear plant - a bilinear approach, Inst. Measurement and Control Symp. on Developments in Self-tuning Control, 5th June 1991

83 Goodhart, S.G., Burnham, K.J., and James, D.J.G., A bilinear self-tuning controller for industrial heating plant, Proc. 8th Int. Conf. on Systems Engineering, Coventry Polytechnic, U.K., 1991

84. King, P.J., Burnham, K.J., James, D.J.G., Martyr, A.J., and Sharpe, S.R., Real-time implementation trials of a self-tuning controller applied to a dynamometer torque loop, Proc. 7th Int. Conf. on Systems Engineering, Univ. of Nevada, Las Vegas, U.S.A., 382-389, 1990

85. King, P.J., Burnham, K.J., James, D.J.G., Norton, J., and Sharpe, S.R., Implementation of a self-tuning controller to the dynamometer torque loop of an engine test cell, Proc. IEE Conf. Control '91, Edinburgh, U.K., **1**, 110-114, 1991

86. King, P.J., Burnham, K.J., James, D.J.G., Norton, J., and Sharpe, S.R., Self-tuning controller design for a non-linear engine testing dynamometer, IEE Colloquium on Real control problems - no solutions yet?, 30th May 1991

87. King, P.J., Burnham, K.J., and James, D.J.G., Adaptive Kalman filter for fast parameter variation, in Hamza, M.H., (Ed.) Proc. IASTED Int. Symp. Modelling, Identification and Control, Innsbruck, Austria, ACTA Press, Zurich, 386-389, 1990

88. Paterson, C.A., Burnham, K.J., James, D.J.G., and Williams, R.A., A study of computer controlled suspension systems, in Hamza, M.H., (Ed.) Proc. IASTED Int. Symp. Modelling, Identification and Control, Innsbruck, Austria, ACTA Press, Zurich, 1991

89. Walker, C.J., A cautious fault detection algorithm, B.Eng. Thesis, Coventry Polytechnic, U.K., 1991

Authors Publications

A1. Shields, D.N., Burnham, K.J., and James, D.J.G., State-space pole-assignment self-tuning controllers, in Hamza, M.H., (Ed.) Proc. IASTED Int. Conf. Computer Aided Design, Nice, France, ACTA Press, Zurich, 28-30, 1984

A2. Burnham, K.J., James, D.J.G. and Shields, D.N., Choice of forgetting factor for self-tuning control, J. Systems Science, **11**, 65-73, 1985

A3. Burnham, K.J., Chan T.W.K., Goodhart, S.G., and James, D.J.G., Design techniques for adaptive control of industrial plant, in Hamza, M.H., (Ed.) Proc. IASTED Int. Symp. Simulation and Modelling, Lugano, Switzerland, ACTA Press, Zurich, 48-51, 1989

A4. Warwick, K., and Burnham, K.J., Filtered and predicted states for discrete-time adaptive control, Proc. IFAC Symp. Adaptive Systems in Control and Signal Processing, Glasgow, U.K., **2**, 387-392, 1989

A5. James, D.J.G., Shields, D.N., and Burnham, K.J., Application of the self-tuning principle to bilinear systems, Proc. 3rd Int. Conf. on Systems Engineering, Wright State Univ., Dayton, Ohio, U.S.A., 26-36, 1984

A6. Burnham, K.J., James, D.J.G., and Shields, D.N., Self-tuning control algorithm for bilinear systems - role of variable forgetting factor, in Hamza, M.H., (Ed.) Proc. IASTED Int. Symp. Modelling, Identification and Control, Grindelwald, Switzerland, ACTA Press, Zurich, 101-105, 1985

A7. Burnham, K.J., James, D.J.G., and Shields, D.N., Self-tuning control principles applied to bilinear systems, Proc. V Polish-English Real-time Process Control Seminar, Tech. Univ. of Warsaw, 1986

A8. James, D.J.G., Burnham, K.J., and Shields, D.N., Extended self-tuning control algorithm for bilinear systems, in Hamza, M.H., (Ed.) Proc. IASTED Int. Symp. Modelling, Identification and Control, Innsbruck, Austria, ACTA Press, Zurich, 1-4, 1986

A9. Burnham, K.J., James, D.J.G., and Shields, D.N., Self-tuning control of bilinear systems, J. Optimal Control Applications and Methods, **8**, 147-157, 1987

A10. Burnham, K.J., and James, D.J.G., Use of cautious estimation in the self-tuning control of bilinear systems, Proc. RAI/IPAR, **1**, Toulouse, France, 419-432, 1986

A11. Burnham, K.J., James, D.J.G., and Shields, D.N., Self-tuning controller design for bilinear systems, Presented at 2nd IMA Symp. on Control Theory, Somerville College, Oxford Univ., September 1986

A12. Burnham, K.J., James, D.J.G., and Shields, D.N., Developments in self-tuning control for bilinear systems, J. Systems Science, **13**, 57-78, 1987

A13. Burnham, K.J., James, D.J.G., and Shields, D.N., Self-tuning control for a class of MISO bilinear systems, Proc. 5th Int. Conf. on Systems Engineering, Wright State Univ., Dayton, Ohio, U.S.A., 261-271, 1987

A14. Randall, A., Burnham, K.J., and James, D.J.G., A study of Kalman filtering techniques for joint state and parameter estimation in self-tuning control, J. Systems Science, **17**, 3, 1991

A15. King, P.J., Burnham, K.J., and James, D.J.G., Comparison of two identification techniques for bilinear systems, in Hamza, M.H., (Ed.) Proc. IASTED Int. Symp. Modelling, Identification and Control, Grindelwald, Switzerland, ACTA Press, Zurich, 71-74, 1989

A16. Goodhart, S.G., Burnham, K.J., and James, D.J.G., Self-tuning control applied to a coupled tank hydraulic system, Proc. 6th Int. Conf. on Systems Engineering, Coventry Polytechnic, U.K., 318-325, 1988

A17. Burnham, K.J., Goodhart, S.G., Gordon, C.M. and James, D.J.G., A laboratory based introduction to self-tuning control, in Hamza, M.H., (Ed.) Proc. IASTED Int. Symp. Modelling, Identification and Control, Grindelwald, Switzerland, ACTA Press, Zurich, 175-178, 1988

A18. Burnham, K.J., and James, D.J.G., Industrial applications of self-tuning control, Proc. 11th IFAC World Congress, Tallinn, Estonia, U.S.S.R., **8**, 32-37, 1990

A19. Burnham, K.J., and James, D.J.G., Implementation of self-tuning controllers on industrial plant, Proc. IMACS-IFAC Symp. Modelling and Control of Technological Systems, Lille, France, **1**, 50-56, 1991

A20. Burnham, K.J., Goodhart, S.G., James, D.J.G., and King, P.J., Design of self-tuning controllers for industrial plant, accepted for presentation at IFAC Symp. Design Methods of Control Systems, Zurich, Switzerland, September 1991

Appendix 1. **Vector Differentiation**

If the following scalars are differentiated with respect to the vectors $\mathbf{v}$ and $\mathbf{z}$, the following results hold [14].

If a) $J_1 = \mathbf{z}^T \mathbf{w}_1 \mathbf{v}$ (A1.1a)

b) $J_2 = \mathbf{v}^T \mathbf{w}_2 \mathbf{z}$ (A1.2a)

and c) $J_3 = \mathbf{z}^T \mathbf{w}_3 \mathbf{z}$ (A1.3a)

Then i) $\dfrac{\partial J_1}{\partial \mathbf{v}} = \mathbf{w}_1^T \mathbf{z}$ (A1.1b)

ii) $\dfrac{\partial J_2}{\partial \mathbf{v}} = \mathbf{w}_2 \mathbf{z}$ (A1.2b)

iii) $\dfrac{\partial J_3}{\partial \mathbf{z}} = \mathbf{w}_3^T \mathbf{z} + \mathbf{w}_3 \mathbf{z}$ (A1.3b)

Appendix 2. **Matrix inversion lemma**

In the derivation of the recursive least squares algorithm in Chapter 2, use is made of the matrix inversion lemma, attributed to Householder (1964). The simple proof presented here follows that given by Astrom in [15].

Lemma:

Let **A**, **C** and **A+BCD** be non-singular square matrices. Then the matrix identity

$$[\mathbf{A}+\mathbf{BCD}]^{-1}=\mathbf{A}^{-1}-\mathbf{A}^{-1}\mathbf{B}[\mathbf{C}^{-1}+\mathbf{DA}^{-1}\mathbf{B}]^{-1}\mathbf{DA}^{-1} \tag{A2.1}$$

holds.

Proof:

Pre-multiply both sides of equation (A2.1) by **[A+BCD]**

$$\mathbf{I}= [\mathbf{A}+\mathbf{BCD}][\mathbf{A}^{-1}-\mathbf{A}^{-1}\mathbf{B}[\mathbf{C}^{-1}+\mathbf{DA}^{-1}\mathbf{B}]^{-1}\mathbf{DA}^{-1}] \tag{A2.2}$$

To prove the lemma, it is suffice to show that the right hand side of equation (A2.2) reduces to the identity matrix. This may be demonstrated by direct manipulation.

$$[\mathbf{A}+\mathbf{BCD}][\mathbf{A}^{-1}-\mathbf{A}^{-1}\mathbf{B}[\mathbf{C}^{-1}+\mathbf{DA}^{-1}\mathbf{B}]^{-1}\mathbf{DA}^{-1}]$$

$$= \mathbf{I}+\mathbf{BCDA}^{-1}-\mathbf{B}[\mathbf{C}^{-1}+\mathbf{DA}^{-1}\mathbf{B}]^{-1}\mathbf{DA}^{-1}$$

$$-\mathbf{BCDA}^{-1}\mathbf{B}[\mathbf{C}^{-1}+\mathbf{DA}^{-1}\mathbf{B}]^{-1}\mathbf{DA}^{-1}$$

$$= \mathbf{I}+\mathbf{BCDA}^{-1}-\mathbf{B}[\mathbf{I}+\mathbf{CDA}^{-1}\mathbf{B}][\mathbf{C}^{-1}+\mathbf{DA}^{-1}\mathbf{B}]^{-1}\mathbf{DA}^{-1}$$

$$= \mathbf{I}+\mathbf{BCDA}^{-1}-\mathbf{BCDA}^{-1}$$

$$= \mathbf{I} \quad \text{as required.}$$

By inspection of equation (2.18) it is clear that $\Phi^{-1}(M)$, $X(M+1)$ and $X^{T}(M+1)$ correspond to the matrices **A**, **B** and **D** in the identity (A2.1) and **C=I** the identity matrix.

Appendix 3. **Derivation of polynomial identity**

The polynomial form of the pole-placement self-tuning controller operates directly on the ARMAX representation (2.1)

$$A(q^{-1})y(t) = q^{-k}B(q^{-1})u(t) + C(q^{-1})e(t) \tag{A3.1}$$

or

$$y(t) = q^{-k}\frac{B(q^{-1})}{A(q^{-1})}u(t) + \frac{C(q^{-1})}{A(q^{-1})}e(t) \tag{A3.2}$$

The polynomial control law which shifts the open-loop poles to their desired closed-loop locations is given by

$$D(q^{-1})u(t) = G(q^{-1})y(t) \tag{A3.3}$$

Defining all polynomials to be of the general form

$$L(q^{-1}) = l_0 + l_1q^{-1} + l_2q^{-2} \ldots + l_{nl}q^{-nl}, \tag{A3.4}$$

it is assumed initially that the orders of $D(q^{-1})$ and $G(q^{-1})$ are $n_d=n_b+k$ and $n_g=n_a$ respectively, with $d_0=1$.

Rearranging (A3.3) gives

$$u(t) = \frac{G(q^{-1})}{D(q^{-1})}y(t) \tag{A3.5}$$

Substituting (A3.5) into (A3.2) leads to

$$y(t) = q^{-k}\frac{B(q^{-1})}{A(q^{-1})}\cdot\frac{G(q^{-1})}{D(q^{-1})}y(t) + \frac{C(q^{-1})}{A(q^{-1})}e(t) \tag{A3.6}$$

$$\left(1 - q^{-k}\frac{B(q^{-1})}{A(q^{-1})}\cdot\frac{G(q^{-1})}{D(q^{-1})}\right)y(t) = \frac{C(q^{-1})}{A(q^{-1})}e(t) \tag{A3.7}$$

Multiplying through by $A(q^{-1})D(q^{-1})$ gives

$$\left(A(q^{-1})D(q^{-1}) - q^{-k}B(q^{-1})G(q^{-1})\right)y(t) = C(q^{-1})D(q^{-1})e(t) \tag{A3.8}$$

Equation (A3.8) is the closed-loop system equation and $\left(A(q^{-1})D(q^{-1}) - q^{-k}B(q^{-1})G(q^{-1})\right)$ is the closed-loop characteristic polynomial, the zeros of which are the eigenvalues or poles of the closed-loop system. Hence the polynomial identity becomes

$$\left(A(q^{-1})D(q^{-1}) - q^{-k}B(q^{-1})G(q^{-1})\right) = \Gamma(q^{-1}) \qquad \text{(A3.9)}$$

Investigation into the polynomial identity (A3.9) reveals that a unique solution for the controller polynomials $D(q^{-1})$ and $G(q^{-1})$ does not exist when their respective orders are taken to be $n_d=n_b+k$ and $n_g=n_a$. In general it is found that there is always 'N' equations and 'N+1' unknowns. It is therefore necessary to fix one of the controller coefficients to some arbitrary value.

It has been suggested [5,19] that d_{nd} should be set to zero. By the nature of the problem, this choice automatically forces the parameter g_{ng} to become zero. Hence for a unique solution the orders of $D(q^{-1})$ and $G(q^{-1})$ become $n_d=n_b+k-1$ and $n_g=n_a-1$ respectively.

It is noted [19] that arbitrary choice of any controller coefficient other than $d_{nd}=0$ will produce a control algorithm which will still perform pole-placement but will have its own distinct controller action.

Appendix 4. **Derivation of feedback vector F**

The derivation of the feedback vector **F** is the central feature of the state-space approach. Essentially, the feedback vector will be a function of the estimated model parameters and the parameters of the specified closed-loop characteristic polynomial. Its derivation is based on the return ratio and return difference matrices and the functional realtionships, given in [23], and these are simply stated here.

Return difference matrices

The open-loop return difference matrix is given by

$$\mathbf{J}_o(q^{-1}) = q^{-1}\,\mathbf{F}\left[\mathbf{I}-q^{-1}\mathbf{P}\right]^{-1}\mathbf{Q}. \tag{A4.1a}$$

The closed-loop return difference matrix is given by

$$\mathbf{J}_c(q^{-1}) = q^{-1}\,\mathbf{F}\left[\mathbf{I}-q^{-1}[\mathbf{P}+\mathbf{QF}]\right]^{-1}\mathbf{Q}. \tag{A4.1b}$$

The return ratio matrix is defined as

$$\mathbf{J}(q^{-1}) = \left[\mathbf{I}-\mathbf{J}_o(q^{-1})\right]. \tag{A4.1c}$$

Functional relationships

Denote the open-loop characteristic polynomial

$$A(q^{-1}) = \Delta_o(q^{-1}) \tag{A4.2a}$$

and the closed-loop characteristic polynomial

$$\Gamma(q^{-1}) = \Delta_c(q^{-1}). \tag{A4.2b}$$

Denote the determinant of the return ratio matrix

$$\det \mathbf{J}(q^{-1}) = \Delta(q^{-1}) \tag{A4.2c}$$

So that

$$\Delta_o(q^{-1}) = \det\left[\mathbf{I}-q^{-1}\mathbf{P}\right] \tag{A4.3a}$$

$$\Delta_c(q^{-1}) = \det\left[\mathbf{I}-q^{-1}[\mathbf{P}+\mathbf{QF}]\right] \tag{A4.3b}$$

$$\Delta(q^{-1}) = \det\left[\mathbf{I}-\mathbf{J}_o(q^{-1})\right] \tag{A4.3c}$$

The fundamental relationship between the open loop and close loop systems is now

derived

$$\left[\mathbf{I}-q^{-1}[\mathbf{P}+\mathbf{QF}]\right] = \left[\mathbf{I}-q^{-1}\mathbf{P}-q^{-1}\mathbf{QF}\right]$$
$$= \left[\mathbf{I}-q^{-1}\mathbf{P}\right]\left[\mathbf{I}-q^{-1}\left[\mathbf{I}-q^{-1}\mathbf{P}\right]^{-1}\mathbf{QF}\right] \qquad \text{(A4.4)}$$

Taking determinants on both sides of (A4.4) and making use of (A4.3) leads to

$$\Delta_c(q^{-1}) = \Delta_o(q^{-1})\left|\left[\mathbf{I}-q^{-1}\left[\mathbf{I}-q^{-1}\mathbf{P}\right]^{-1}\mathbf{QF}\right]\right| \qquad \text{(A4.5)}$$

Denoting $\mathbf{P}^*=q^{-1}[\mathbf{I}-q^{-1}\mathbf{P}]^{-1}$ leads to

$$\Delta_c(q^{-1}) = \Delta_o(q^{-1})\left|\left[\mathbf{I}-q^{-1}\mathbf{P}^*\mathbf{QF}\right]\right| \qquad \text{(A4.6)}$$

To proceed, use is made of the identity [24].

Matrix Identity

For any mXn matrix $\mathbf{M}_1$ and nXm matrix $\mathbf{M}_2$ the identity

$$\left|\mathbf{I}_m+\mathbf{M}_1\mathbf{M}_2\right| = \left|\mathbf{I}_n+\mathbf{M}_2\mathbf{M}_1\right|$$

holds.

In (A4.6) dim{$\mathbf{P}^*$}=nXn, dim{$\mathbf{Q}$}=nX1 and dim{$\mathbf{F}$}=1Xn. If $\mathbf{M}_1=\mathbf{P}^*\mathbf{Q}$ and $\mathbf{M}_2=\mathbf{F}$ then dim{$\mathbf{M}_1$}=nX1 and dim{$\mathbf{M}_2$}=1Xn. Applying the above identity gives

$$\left|\mathbf{I}_n+[\mathbf{P}^*\mathbf{Q}]\mathbf{F}\right| = \left|1-\mathbf{F}[\mathbf{P}^*\mathbf{Q}]\right| \qquad \text{(A4.7)}$$

Substituting (A4.7) into (A4.6) leads to

$$\Delta_c(q^{-1}) = \Delta_o(q^{-1})\left|1-\mathbf{FP}^*\mathbf{Q}\right| \qquad \text{(A4.8)}$$

equivalently

$$\Delta_c(q^{-1}) = \Delta_o(q^{-1})\left|1-q^{-1}\mathbf{F}[\mathbf{I}-q^{-1}\mathbf{P}]^{-1}\mathbf{Q}\right| \qquad \text{(A4.9)}$$

Thus from (A4.1) it follows that

$$\Delta_c(q^{-1}) = \Delta_o(q^{-1})\left|\mathbf{J}(q^{-1})\right| \qquad \text{(A4.10)}$$

so that

$$\Delta_c(q^{-1}) = \Delta_o(q^{-1})\,\Delta(q^{-1}) \qquad \text{(A4.11)}$$

or

$$\Delta(q^{-1}) = \frac{\Delta_c(q^{-1})}{\Delta_o(q^{-1})}. \qquad \text{(A4.12)}$$

Equation (A4.12) is a fundamental relationship between the open-loop and closed-loop systems and is called upon later in the derivation.

Substituting (A4.1c) into (A4.1a) gives

$$\mathbf{J}(q^{-1}) = \mathbf{I}-q^{-1}\mathbf{F}\left[\mathbf{I}-q^{-1}\mathbf{P}\right]^{-1}\mathbf{Q}. \qquad \text{(A4.13)}$$

which from the definition of an inverse

$$\left[\mathbf{I}-q^{-1}\mathbf{P}\right]^{-1} = \frac{\mathrm{Adj}\left[\mathbf{I}-q^{-1}\mathbf{P}\right]}{\mathrm{Det}\left[\mathbf{I}-q^{-1}\mathbf{P}\right]}$$

$$= \frac{\mathrm{Adj}\left[\mathbf{I}-q^{-1}\mathbf{P}\right]}{\Delta_o(q^{-1})} \qquad \text{(A4.14)}$$

Substituting (A4.14) into equation (A4.13) gives

$$\mathbf{J}(q^{-1}) = \mathbf{I}-q^{-1}\mathbf{F}\,\frac{\mathrm{Adj}\left[\mathbf{I}-q^{-1}\mathbf{P}\right]}{\Delta_o(q^{-1})}\,\mathbf{Q}. \qquad \text{(A4.15)}$$

Denoting $\mathscr{L}=\mathrm{Adj}\left[\mathbf{I}-q^{-1}\mathbf{P}\right]$, equation (A4.15) becomes

$$\mathbf{J}(q^{-1}) = \mathbf{I}-q^{-1}\frac{\mathbf{F}\mathscr{L}\mathbf{Q}}{\Delta_o(q^{-1})}. \qquad \text{(A4.16)}$$

Since dim{**F**}=(1X**n**), dim{$\mathscr{L}$}=(**n**X**n**) and dim{**Q**}=(**n**X1) equation (A4.16) is a scalar so that

$$\det \mathbf{J}(q^{-1}) = \mathbf{J}(q^{-1}) = \Delta(q^{-1}). \qquad \text{(A4.17)}$$

Hence

$$\Delta(q^{-1}) = 1 - q^{-1}\frac{\mathbf{F}\mathscr{L}\mathbf{Q}}{\Delta_o(q^{-1})} \qquad \text{(A4.18)}$$

i.e.
$$\Delta(q^{-1}) = \frac{\Delta_o(q^{-1}) - q^{-1}\,\mathbf{F}\mathscr{L}\mathbf{Q}}{\Delta_o(q^{-1})} \qquad \text{(A4.19)}$$

Using the fundamental relationship given by (A4.12)

$$\Delta(q^{-1})\Delta_o(q^{-1}) = \Delta_c(q^{-1}) = \Delta_o(q^{-1}) - q^{-1}\,\mathbf{F}\mathscr{L}\mathbf{Q} \qquad \text{(A4.20)}$$

i.e.
$$\Delta_c(q^{-1}) = -q^{-1}\,\mathbf{F}\mathscr{L}\mathbf{Q} + \Delta_o(q^{-1}) \qquad \text{(A4.21)}$$

Since (A4.21) is essentially a scalar equation, it follows that transposition gives

$$\Delta_c(q^{-1}) = -q^{-1}\,[\mathbf{F}\mathscr{L}\mathbf{Q}]^T + \Delta_o(q^{-1}) \qquad \text{(A4.22)}$$

i.e.
$$\Delta_c(q^{-1}) = -q^{-1}\,\mathbf{Q}^T\mathscr{L}^T\mathbf{F}^T + \Delta_o(q^{-1}) \qquad \text{(A4.23)}$$

$$\Gamma(q^{-1}) = -q^{-1}\,\mathbf{Q}^T\mathscr{L}^T\mathbf{F}^T + A(q^{-1}) \qquad \text{(A4.24)}$$

In order to rearrange (A4.24) into a form suitable for the evaluation of **F**, the term $-q^{-1}\mathbf{Q}^T\mathscr{L}^T\mathbf{F}^T$ is considered in more detail.

The solution of **F**, by necessity, involves equating coefficients of like powers of q^{-1} in (A4.24). Expanding (A4.24) leads to

$$\begin{bmatrix} q^{-n} \ldots . q^{-1}\ q^0 \end{bmatrix}\begin{bmatrix} 0 \\ 0 \\ \gamma_{n_a} \\ . \\ . \\ \gamma_1 \\ 1 \end{bmatrix} = -q^{-1}\,\mathbf{Q}^T\mathscr{L}^T\mathbf{F}^T + \begin{bmatrix} q^{-n} \ldots . q^{-1}\ q^0 \end{bmatrix}\begin{bmatrix} 0 \\ 0 \\ a_{n_a} \\ . \\ . \\ a_1 \\ 1 \end{bmatrix} \qquad \ldots \text{(A4.25)}$$

In order that (A4.25) be consistent, the term $q^{-1}\mathbf{Q}^T\mathscr{L}^T$ should reduce to the form

[row of shift operators] [nXn matrix],

so that the required scalar quantity is produced when operated on by $\mathbf{F}^T$. Recall

$$\mathscr{L}^T = \left[\text{Adj}[\mathbf{I}-q^{-1}\mathbf{P}]\right]^T \tag{A4.26}$$

Now since $\mathscr{L}^T$ is the matrix of minors of $[\mathbf{I}-q^{-1}\mathbf{P}]$ and since $[\mathbf{I}-q^{-1}\mathbf{P}]$ is of dimension nXn, it follows that the elements of $\mathscr{L}^T$ are polynomials in q^{-1} of maximum degree (n-1),

$$\text{i.e. } \mathscr{L}^T = \begin{bmatrix} \ell_{11}(q^{-1}) & \ell_{12}(q^{-1}) & \cdots & \ell_{1n}(q^{-1}) \\ \ell_{21}(q^{-1}) & \ell_{22}(q^{-1}) & \cdots & \ell_{2n}(q^{-1}) \\ \vdots & & & \\ \ell_{n1}(q^{-1}) & \ell_{n2}(q^{-1}) & \cdots & \ell_{nn}(q^{-1}) \end{bmatrix} \tag{A4.27}$$

$$\text{or } \mathscr{L}^T = \left[\boldsymbol{\ell}_1(q^{-1}) \;\; \boldsymbol{\ell}_2(q^{-1}) \;\; \cdots \;\; \boldsymbol{\ell}_n(q^{-1})\right] \tag{A4.28}$$

where $\boldsymbol{\ell}_i$ denotes the i^{th} column of $\mathscr{L}^T$. Pre-multiplying (A4.28) by $-q^{-1}\mathbf{Q}^T$ gives

$$-q^{-1}\mathbf{Q}^T\left[\boldsymbol{\ell}_1(q^{-1}) \;\; \boldsymbol{\ell}_2(q^{-1}) \;\; \cdots \;\; \boldsymbol{\ell}_n(q^{-1})\right] \tag{A4.29}$$

so that

$$-q^{-1}\mathbf{Q}^T\,\mathscr{L}^T = \left[-\mathbf{Q}^T\boldsymbol{\ell}_1(q^{-1}) \;\; -\mathbf{Q}^T\boldsymbol{\ell}_2(q^{-1}) \;\; \cdots \;\; -\mathbf{Q}^T\boldsymbol{\ell}_n(q^{-1})\right] \tag{A4.30}$$

Note that the polynomials in (A4.30) are now of maximim degree n. Detaching the coefficients, the term $-q^{-1}\mathbf{Q}^T\,\mathscr{L}^T$ may now be expressed in the reqired form of (A4.25)

$$-q^{-1}\mathbf{Q}^T\,\mathscr{L}^T = \left[q^{-n}\ldots q^{-1}\right]\begin{bmatrix} (-\mathbf{Q}^T\boldsymbol{\ell}_1)_n & (-\mathbf{Q}^T\boldsymbol{\ell}_2)_n & \cdots & (-\mathbf{Q}^T\boldsymbol{\ell}_n)_n \\ (-\mathbf{Q}^T\boldsymbol{\ell}_1)_{n-1} & (-\mathbf{Q}^T\boldsymbol{\ell}_2)_{n-1} & \cdots & (-\mathbf{Q}^T\boldsymbol{\ell}_n)_{n-1} \\ \vdots & & & \\ (-\mathbf{Q}^T\boldsymbol{\ell}_1)_1 & (-\mathbf{Q}^T\boldsymbol{\ell}_2)_1 & \cdots & (-\mathbf{Q}^T\boldsymbol{\ell}_n)_1 \end{bmatrix}$$

$$\ldots \tag{A4.31}$$

Denote the nxn matrix in (A4.31) as **W**,

$$\text{i.e. } \mathbf{W} = \begin{bmatrix} (-\mathbf{Q}^T\boldsymbol{\ell}_1)_n & (-\mathbf{Q}^T\boldsymbol{\ell}_2)_n & \cdots & (-\mathbf{Q}^T\boldsymbol{\ell}_n)_n \\ (-\mathbf{Q}^T\boldsymbol{\ell}_1)_{n-1} & (-\mathbf{Q}^T\boldsymbol{\ell}_2)_{n-1} & \cdots & (-\mathbf{Q}^T\boldsymbol{\ell}_n)_{n-1} \\ \vdots & & & \\ (-\mathbf{Q}^T\boldsymbol{\ell}_1)_1 & (-\mathbf{Q}^T\boldsymbol{\ell}_2)_1 & \cdots & (-\mathbf{Q}^T\boldsymbol{\ell}_n)_1 \end{bmatrix} \quad \ldots \quad \text{(A4.32)}$$

then

$$-q^{-1}\mathbf{Q}^T\,\mathscr{L}^T = [q^{-n} \ldots q^{-1}]\,\mathbf{W}. \qquad \text{(A4.33)}$$

It is interesting to note that the matrix $\mathbf{W}$ is symmetric.

Substituting (A4.33) into equatin (A4.25) and omitting the constant terms leads to

$$\begin{bmatrix} q^{-n} \ldots q^{-1} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \gamma_{n_a} \\ \vdots \\ \gamma_2 \\ \gamma_1 \end{bmatrix} = \begin{bmatrix} q^{-n} \ldots q^{-1} \end{bmatrix} \mathbf{W}\mathbf{F}^T + \begin{bmatrix} q^{-n} \ldots q^{-1} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ a_{n_a} \\ \vdots \\ a_2 \\ a_1 \end{bmatrix} \quad \ldots \quad \text{(A4.34)}$$

Finally, removing the shift operators

$$\mathbf{W}\mathbf{F}^T = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \gamma_{n_a} - a_{n_a} & \\ \vdots & \\ \gamma_1 - a_1 & \end{bmatrix} \qquad \text{(A4.35)}$$

Denoting $\left[0 \ldots 0\ (\gamma_{n_a} - a_{n_a}) \ldots (\gamma_1 - a_1)\right] = \mathbf{S}^T$ leads to

$$\mathbf{W}\mathbf{F}^{\mathrm{T}} = \mathbf{S} \tag{A4.36}$$

so that

$$\mathbf{F}^{\mathrm{T}} = \mathbf{W}^{-1}\mathbf{S} \tag{A4.37}$$

as required.

The key feature of the state-space approach lies in the setting up of the symmetric matrix W. The symmetry of W and its representation in terms of the Kalman controllability test matrix are considered in Appendices 5 and 6 respectively.

Appendix 5. **The symmetry of the matrix W**

The symmetry of **W** may be readily illustrated by example. Here, its general structure is explored and a pattern is detected which verifies that **W** is symmetic in the general case.

From Appendix 4,

$$\mathscr{L}^T = \left[\text{Adj}[\mathbf{I}-q^{-1}\mathbf{P}]\right]^T$$

or
$$\mathscr{L}^T = \text{Adj}[\mathbf{I}-q^{-1}\mathbf{P}^T] \qquad \text{(A5.1)}$$

Since, from Appendix 4, elements of the nXn matrix $\mathscr{L}^T$ are polynomials in q^{-1} of maximum order (n-1), it follows that $\mathscr{L}^T$ can be represented by a polynomial of order (n-1) with matrix coefficients of order nXn,

i.e.
$$\mathscr{L}^T = \left[\chi_0 + \chi_1 q^{-1} + \ldots + \chi_{(n-1)} q^{-(n-1)}\right] \qquad \text{(A5.2)}$$

so that

$$\text{Adj}[\mathbf{I}-q^{-1}\mathbf{P}^T] = \left[\chi_0 + \chi_1 q^{-1} + \ldots + \chi_{(n-1)} q^{-(n-1)}\right] \qquad \text{(A5.3)}$$

For any non-singular square matrix **Y** the following identities hold

(i)
$$\mathbf{Y}^{-1} = \frac{\text{Adj } \mathbf{Y}}{\det \mathbf{Y}} \qquad \text{(A5.4a)}$$

(ii)
$$\mathbf{Y}[\text{Adj } \mathbf{Y}] = (\det \mathbf{Y})\mathbf{I} \qquad \text{(A5.4b)}$$

(iii)
$$\det \mathbf{Y} = \det \mathbf{Y}^T \qquad \text{(A5.4c)}$$

Applying (A5.4b) and (A5.4c) to the square matrix $[\mathbf{I}-q^{-1}\mathbf{P}^T]$ leads to

$$[\mathbf{I}-q^{-1}\mathbf{P}^T]\left[\text{Adj}[\mathbf{I}-q^{-1}\mathbf{P}^T]\right] = \det [\mathbf{I}-q^{-1}\mathbf{P}^T]\,\mathbf{I}$$
$$= \det [\mathbf{I}-q^{-1}\mathbf{P}]\,\mathbf{I} \qquad \text{(A5.5)}$$

Now $\det [\mathbf{I}-q^{-1}\mathbf{P}]$ is the open-loop characteristic polynomial $A(q^{-1})$, so that

$$[\mathbf{I}-q^{-1}\mathbf{P}^T]\left[\text{Adj}[\mathbf{I}-q^{-1}\mathbf{P}^T]\right] = (\,1 + a_1 q^{-1} + a_2 q^{-2} + \ldots + a_{n_a} q^{-n_a}\,)\,\mathbf{I} \qquad \text{(A5.6)}$$

Substituting (A5.3) into (A5.6) gives

$$[\mathbf{I}-q^{-1}\mathbf{P}^T]\left[\chi_0 + \chi_1 q^{-1} + \ldots + \chi_{(n-1)} q^{-(n-1)}\right] = (\,1 + a_1 q^{-1} + a_2 q^{-2} + \ldots + a_{n_a} q^{-n_a}\,)\,\mathbf{I}$$
$$\ldots \text{(A5.7)}$$

which on equating coefficients leads to

$$\chi_0 = \mathbf{I}$$

$$\chi_1 = a_1\mathbf{I} + \mathbf{P}^T\chi_0 = a_1\mathbf{I} + \mathbf{P}^T$$

$$\chi_2 = a_2\mathbf{I} + \mathbf{P}^T\chi_1 = a_2\mathbf{I} + a_1\mathbf{P}^T + \mathbf{P}^{T2}$$

$$\chi_3 = a_3\mathbf{I} + \mathbf{P}^T\chi_2 = a_3\mathbf{I} + a_2\mathbf{P}^T + a_1\mathbf{P}^{T2} + \mathbf{P}^{T3}$$

.

.

$$\chi_{(n-1)} = a_{(n-1)}\mathbf{I} + \mathbf{P}^T\chi_{(n-2)} = a_{(n-1)}\mathbf{I} + a_{(n-2)}\mathbf{P}^T + \ldots . + \mathbf{P}^{T(n-1)} \qquad \text{(A5.8)}$$

Pre-multiplying (A5.2) by $-q^{-1}\mathbf{Q}^T$ leads to

$$-q^{-1}\mathbf{Q}^T\mathscr{L}^T = -q^{-1}\mathbf{Q}^T\left[\chi_0 + \chi_1 q^{-1} + \ldots . + \chi_{(n-1)}q^{-(n-1)}\right] \qquad \text{(A5.9)}$$

and since (A5.9) is essentially a scalar equation it can be written

$$\left[q^{-n} \ldots . q^{-1}\right] \begin{bmatrix} -\mathbf{Q}^T\chi_{(n-1)} \\ - - - - \\ \vdots \\ - - - - \\ -\mathbf{Q}^T\chi_1 \\ - - - - \\ -\mathbf{Q}^T\chi_0 \end{bmatrix} \qquad \text{(A5.10)}$$

.

Note that the partitioned (nXn) matrix in (A5.10) is **W**. If x_{ij} denotes the j^{th} column of χ_i, then **W** becomes

$$\mathbf{W} = \begin{bmatrix} -\mathbf{Q}^T x_{(n-1)1} & -\mathbf{Q}^T x_{(n-1)2} & \ldots . & -\mathbf{Q}^T x_{(n-1)n} \\ -\mathbf{Q}^T x_{(n-2)1} & -\mathbf{Q}^T x_{(n-2)2} & \ldots . & -\mathbf{Q}^T x_{(n-2)n} \\ \vdots & & & \\ -\mathbf{Q}^T x_{01} & -\mathbf{Q}^T x_{02} & \ldots . & -\mathbf{Q}^T x_{0n} \end{bmatrix} \qquad \text{(A5.11)}$$

To show the symmetry of **W**, the matrix which has as its elements the columns x_{ij} need only be considered. Denote this matrix by χ_W,

i.e.
$$\chi_W = \left[\begin{array}{c} \chi_{(n-1)} \\ \hline \vdots \\ \hline \chi_1 \\ \hline \chi_0 \end{array}\right] \qquad \text{(A5.12)}$$

It is clear from (A5.8) that the matrix χ_W and hence **W** will always be symmetric. In the general case, it may be readily shown that

$$\chi_W = \left[\begin{array}{cccccccccccc}
a_{(n-1)} & a_{(n-2)} & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & a_1 & 1 \\
0 & 0 & & & & & & & 0 & 0 \\
\vdots & \vdots & & & & & & & \vdots & \vdots \\
0 & 0 & & & & & & & 0 & 0 \\
\hline
a_{(n-2)} & a_{(n-1)} & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & 1 & 0 \\
0 & a_{(n-2)} & & & & & & & a_1 & 1 \\
0 & -a_{(n-1)} & & & & & & & & 0 \\
0 & 0 & & & & & & & \vdots & \vdots \\
\vdots & \vdots & & & & & & & \vdots & \vdots \\
0 & 0 & & & & & & & -a_{(n-1)} & 0 \\
\hline
\vdots & \vdots & & & & & & & \vdots & \vdots \\
\vdots & \vdots & & & & & & & \vdots & \vdots \\
\vdots & \vdots & & & & & & & \vdots & \vdots \\
\hline
a_1 & 1 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 & 0 \\
0 & a_1 & & & & & & & \vdots & \vdots \\
 & & & & & & & & 1 & 0 \\
\vdots & \vdots & & & & & & & a_1 & 1 \\
\vdots & \vdots & & & & & & & -a_2 & 0 \\
0 & -a_{(n-1)} & & & & & & & & \\
\hline
1 & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 & 0 \\
0 & 1 & & & & & & & \vdots & \vdots \\
 & 0 & & & & & & & \vdots & \vdots \\
\vdots & \vdots & & & & & & & & \vdots \\
\vdots & \vdots & & & & & & & 1 & \vdots \\
0 & 0 & & & & & & & 0 & 1
\end{array}\right] \qquad \text{(A5.13)}$$

Thus the matrix χ_W and hence **W** is shown to be symmetric.

Appendix 6. The symmetric matrix W in terms of the Kalman controllability test matrix

The matrix **W** has been shown to be symmetric,

i.e. $$\mathbf{W} = \mathbf{W}^T \tag{A6.1}$$

From Appendix 5,

$$\mathbf{W} = \begin{bmatrix} -\mathbf{Q}^T\chi_{(n-1)} \\ - - - - \\ \vdots \\ - - - - \\ -\mathbf{Q}^T\chi_1 \\ - - - - \\ -\mathbf{Q}^T\chi_0 \end{bmatrix} \tag{A6.2}$$

Applying the result of (A6.1) gives

$$\mathbf{W} = \left[-\chi^T_{(n-1)}\mathbf{Q} : \ldots : -\chi_1^T\mathbf{Q} : -\chi_0^T\mathbf{Q} \right] \tag{A6.3}$$

Transposition of equation (A5.8) of Appendix 5 gives

$$\chi_0^T = \mathbf{I}$$

$$\chi_1^T = a_1\mathbf{I} + \mathbf{P}$$

$$\chi_2^T = a_2\mathbf{I} + a_1\mathbf{P} + \mathbf{P}^2$$

$$\vdots$$

$$\chi^T_{(n-1)} = a_{(n-1)}\mathbf{I} + a_{(n-2)}\mathbf{P} + \ldots + \mathbf{P}^{(n-1)} \tag{A6.4}$$

Substituting (A6.4) into (A6.3) leads to

$$\mathbf{W} = \left[-\left[a_{(n-1)}\mathbf{I} + a_{(n-2)}\mathbf{P} + \ldots + \mathbf{P}^{(n-1)} \right]\mathbf{Q} : \ldots : -\left[a_1\mathbf{I} + \mathbf{P} \right]\mathbf{Q} : -\mathbf{IQ} \right] \tag{A6.5}$$

Hence

$$\mathbf{W} = \left[\mathbf{P}^{(n-1)}\mathbf{Q} : \ldots : \mathbf{PQ} : \mathbf{Q}\right] \begin{bmatrix} -1 & 0 & \cdots & 0 \\ -a_1 & -1 & & \\ & & \ddots & \\ -a_{(n-1)} & \cdots & -a_1 & -1 \end{bmatrix} \qquad \text{(A6.7)}$$

so that

$$\mathbf{W} = \mathbf{KL} \qquad \text{(A6.8)}$$

as required.

Appendix 7. Adaptive least squares - role of variable forgetting factor

In the standard linear regression problem all data is given equal weighting. However, in oder to facilitate the tracking of slowly varying model parameters an exponential weighting of data is required in which greater emphasis is attached to the more recent observations. This is achieved by introducing a scalar forgetting factor $\lambda \leq 1.0$ into the recursive least squares algorithm. Essentially the least squares cost function is modified such that

$$J = (y - X\theta)^T \Lambda (y - X\theta) \tag{A7.1}$$

where Λ is a diagonal matrix. In the standard approach Λ is taken to be

$$\Lambda = \begin{bmatrix} \lambda^{M-1} & & & \\ & \ddots & & \\ & & \lambda & \\ & & & 1 \end{bmatrix} \tag{A7.2}$$

so that the covariance matrix becomes

$$\Phi(M) = \left[X^T(M)\, \Lambda(M)\, X(M)\right]^{-1} \tag{A7.3}$$

and

$$\theta(M) = \Phi(M) X^T(M) \Lambda(M) y(M) \tag{A7.4}$$

so that at the next step

$$\Phi(M+1) = \left[\begin{bmatrix} X(M) \\ \hline x^T(M+1) \end{bmatrix}^T \begin{bmatrix} \lambda^M & & \\ & \ddots & \\ & & \lambda \end{bmatrix} \begin{bmatrix} X(M) \\ \hline x^T(M+1) \end{bmatrix} \right]^{-1} \tag{A7.5}$$

$$= \left[X^T(M) \begin{bmatrix} \lambda^M & & \\ & \ddots & \\ & & \lambda \end{bmatrix} X(M) + x(M+1) x^T(M+1) \right]^{-1}$$

$$= \left[\lambda \boldsymbol{X}^{T}(M)\boldsymbol{\Lambda}\boldsymbol{X}(M) + \boldsymbol{x}(M+1)\boldsymbol{x}^{T}(M+1) \right]^{-1}$$

$$= \left[\lambda \boldsymbol{\Phi}(M)^{-1} + \boldsymbol{x}(M+1)\boldsymbol{x}^{T}(M+1) \right]^{-1}$$

Applying the matrix inversion lemma (Appendix 2) leads to

$$\boldsymbol{\Phi}(M+1) = \left[\boldsymbol{\Phi}(M) - \frac{\boldsymbol{\Phi}(M)\boldsymbol{x}(M+1)\boldsymbol{x}^{T}(M+1)\boldsymbol{\Phi}(M)}{\lambda + \boldsymbol{x}^{T}(M+1)\boldsymbol{\Phi}(M)\boldsymbol{x}(M+1)} \right] / \lambda \qquad \text{(A7.6)}$$

Note that if λ=1.0 all data is given equal weighting.

The approach adopted here makes use of the slightly different cost function

$$J = (\mathbf{y} - \boldsymbol{X}\boldsymbol{\theta})^{T}\, \bar{\boldsymbol{\Lambda}}\, (\mathbf{y} - \boldsymbol{X}\boldsymbol{\theta}) \qquad \text{(A7.7)}$$

where $\bar{\boldsymbol{\Lambda}}$ is the diagonal matrix

$$\bar{\boldsymbol{\Lambda}} = \lambda \boldsymbol{\Lambda} \qquad \text{(A7.8)}$$

i.e.
$$\bar{\boldsymbol{\Lambda}} = \begin{bmatrix} \lambda^{M} & & & & \\ & \cdot & & & \\ & & \cdot & & \\ & & & \lambda^{2} & \\ & & & & \lambda \end{bmatrix}$$

so that at the next step

$$\boldsymbol{\Phi}(M+1) = \left[\begin{bmatrix} \boldsymbol{X}(M) \\ \hline \boldsymbol{x}^{T}(M+1) \end{bmatrix}^{T} \left[\begin{array}{c|c} \lambda^{M+1} & \\ \cdot & \\ \cdot & \\ \lambda^{2} & \\ \hline & \lambda \end{array} \right] \begin{bmatrix} \boldsymbol{X}(M) \\ \hline \boldsymbol{x}^{T}(M+1) \end{bmatrix} \right]^{-1} \qquad \text{(A7.9)}$$

$$= \left[\lambda \boldsymbol{X}^{T}(M)\, \bar{\boldsymbol{\Lambda}}\, \boldsymbol{X}(M) + \lambda \boldsymbol{x}(M+1)\boldsymbol{x}^{T}(M+1) \right]^{-1}$$

which leads to

$$\boldsymbol{\Phi}(M+1) = \left[\boldsymbol{\Phi}(M)^{-1} + \boldsymbol{x}(M+1)\boldsymbol{x}^{T}(M+1) \right]^{-1} \lambda^{-1}$$

Application of the matrix inversion lemma gives

$$\boldsymbol{\Phi}(M+1) = \left[\boldsymbol{\Phi}(M) - \frac{\boldsymbol{\Phi}(M)\boldsymbol{x}(M+1)\boldsymbol{x}^{T}(M+1)\boldsymbol{\Phi}(M)}{1 + \boldsymbol{x}^{T}(M+1)\boldsymbol{\Phi}(M)\boldsymbol{x}(M+1)} \right] / \lambda \qquad \text{(A7.10)}$$

Note that the only difference between (A7.6) and (A7.10) is in the scalar denominator terms. Whilst the difference between the two approaches is negligible, there is a slight advantage in the adopted approach in that the scalar denominator $[1+\boldsymbol{x}^T(M+1)\boldsymbol{\Phi}(M)\boldsymbol{x}(M+1)]$ in (A7.10) is readily available for use within the variable forgetting factor scheme.

The resulting adaptive least squares algorithm then becomes

$$\hat{\boldsymbol{\theta}}(t) = \hat{\boldsymbol{\theta}}(t\text{-}1) + \boldsymbol{\phi}(t)[\, y(t) - \boldsymbol{x}^T(t)\hat{\boldsymbol{\theta}}(t\text{-}1)\,]$$

$$\boldsymbol{\phi}(t) = \boldsymbol{\Phi}(t\text{-}1)\boldsymbol{x}(t)[\, 1 + \boldsymbol{x}^T(t)\boldsymbol{\Phi}(t\text{-}1)\boldsymbol{x}(t)]^{-1}$$

$$\boldsymbol{\Phi}(t) = [\, \mathbf{I} - \boldsymbol{\phi}(t)\boldsymbol{x}^T(t)\,]\boldsymbol{\Phi}(t\text{-}1) / \lambda.$$

In the case of a variable forgetting factor, the scalar λ is simply replaced by $\lambda(t)$.

Appendix 8. **The Kalman filter**

The Kalman filter (KF) problem is essentially that of estimating the state vector **x** of a linear system with noisy measurements and process noise in the dynamics. Such a system in its discrete form may be expressed

$$\mathbf{x}(t+1) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{D}\mathbf{w}(t) \tag{A8.1}$$

$$y(t) = \mathbf{c}^T\mathbf{x}(t) + v(t) \tag{A8.2}$$

where $\mathbf{x}\in\mathbb{R}^n$ is the state vector (to be estimated), $\mathbf{u}\in\mathbb{R}^m$ is the input vector, $\mathbf{w}\in\mathbb{R}^n$ is the process noise vector, $y\in\mathbb{R}$ is the system output and $v\in\mathbb{R}$ is the output measurement noise. The matrices **A**, **B**, **c**, and **D** are assumed to be known. It is further assumed that v and **w** are independent white noise sequences (i.e. serially independent random variables with zero mean) and have constant known covariance matrices,

i.e. $E\{\mathbf{w}\} = 0$, $E\{v\} = 0$, $E\{\mathbf{w}_i v_j\} = 0$, $E\{\mathbf{w}_i \mathbf{w}_j\} = \mathbf{R}_w\delta_{ij}$, $E\{v_i v_j\} = r_v\delta_{ij}$

where $E\{\cdot\}$ denotes the expected value and δ_{ij} is the Kronecker delta.

The problem is to produce a recursive estimate $\hat{\mathbf{x}}$ which minimises the estimation error $(\hat{\mathbf{x}}-\mathbf{x})$ based on the inputs **u** and the measured output y. The important distinction between the KF problem for estimating the state vector $\hat{\mathbf{x}}$ and that of recursive least squares (RLS) for estimating the principally time invariant parameter vector $\hat{\boldsymbol{\theta}}$, is that the state vector is now being forced to follow a trajectory governed by (A8.1). The Kalman filter is a two stage prediction/correction process, with the prediction stage allowing for modellable changes in **x** to be taken into account between samples and the correction stage, which is essentially the RLS algorithm, taking place at the sampling instants.

Given (A8.1) and the statistical assumptions on **w**, the best *a priori* estimate (or prediction) of **x**(t) given $\hat{\mathbf{x}}$(t-1) is

$$\hat{\mathbf{x}}(t|t-1) = \mathbf{A}\hat{\mathbf{x}}(t-1) + \mathbf{B}\mathbf{u}(t-1) \tag{A8.4}$$

with the notation $\hat{\mathbf{x}}(t|t\text{-}1)$ being read as the estimate of $\hat{\mathbf{x}}(t)$ based on information upto and including (t-1). The corresponding estimation error is

$$\mathbf{e} = \hat{\mathbf{x}}(t|t\text{-}1) - \mathbf{x}(t)$$

which, from (A8.1) and (A8.4), becomes

$$\mathbf{e} = \mathbf{A}\left(\hat{\mathbf{x}}(t\text{-}1) - \mathbf{x}(t\text{-}1)\right) - \mathbf{D}\mathbf{w}(t\text{-}1)$$

so that the error covariance matrix

$$\boldsymbol{\Phi}(t|t\text{-}1) = E[\mathbf{e}\mathbf{e}^T]. \tag{A8.5}$$

Expanding (A8.5) leads to

$$\begin{aligned}\boldsymbol{\Phi}(t|t\text{-}1) = E\Big[\Big[\ \mathbf{A}\left(\hat{\mathbf{x}}(t\text{-}1)-\mathbf{x}(t\text{-}1)\right) - \mathbf{D}\mathbf{w}(t\text{-}1)\ \Big]\Big[\ \left(\hat{\mathbf{x}}(t\text{-}1)-\mathbf{x}(t\text{-}1)\right)^T\mathbf{A}^T - \mathbf{w}^T(t\text{-}1)\mathbf{D}^T\ \Big]\Big] \\ = \mathbf{A}\,E\Big[\ \left(\hat{\mathbf{x}}(t\text{-}1)-\mathbf{x}(t\text{-}1)\right)\left(\hat{\mathbf{x}}(t\text{-}1)-\mathbf{x}(t\text{-}1)\right)^T\ \Big]\mathbf{A}^T \\ - \mathbf{A}\,E\Big[\ \left(\hat{\mathbf{x}}(t\text{-}1)-\mathbf{x}(t\text{-}1)\right)\mathbf{w}^T(t\text{-}1)\ \Big]\mathbf{D}^T \\ - \mathbf{D}\,E\Big[\ \mathbf{w}(t\text{-}1)\left(\hat{\mathbf{x}}(t\text{-}1)-\mathbf{x}(t\text{-}1)\right)^T\ \Big]\mathbf{A}^T \\ + \mathbf{D}\,E\Big[\ \mathbf{w}(t\text{-}1)\mathbf{w}^T(t\text{-}1)\ \Big]\mathbf{D}^T\end{aligned} \tag{A8.6}$$

Note that the second and third terms of (A8.6) are zero, so that

$$\boldsymbol{\Phi}(t|t\text{-}1) = \mathbf{A}\ \mathrm{cov}\Big(\ \hat{\mathbf{x}}(t\text{-}1)-\mathbf{x}(t\text{-}1)\ \Big)\mathbf{A}^T + \mathbf{D}\ \mathrm{cov}\Big(\ \mathbf{w}(t\text{-}1)\ \Big)\mathbf{D}^T$$

From (A8.3) and denoting $\boldsymbol{\Phi}(t\text{-}1|t\text{-}1) = \mathrm{cov}\Big(\ \hat{\mathbf{x}}(t\text{-}1)-\mathbf{x}(t\text{-}1)\ \Big)$ leads to

$$\boldsymbol{\Phi}(t|t\text{-}1) = \mathbf{A}\,\boldsymbol{\Phi}(t\text{-}1|t\text{-}1)\,\mathbf{A}^T + \mathbf{D}\,\mathbf{R}_w\,\mathbf{D}^T \tag{A8.7}$$

Equations (A8.4) and (A8.7) form the prediction stage of the KF and, by making the translation $\boldsymbol{\theta} \rightarrow \mathbf{x}$ and $\boldsymbol{x} \rightarrow \mathbf{c}$, the standard RLS algorithm essentially forms the correction stage. The KF algorithm then becomes

Prediction (between sampling instants)

$$\hat{\mathbf{x}}(t|t\text{-}1) = \mathbf{A}\hat{\mathbf{x}}(t\text{-}1|t\text{-}1) + \mathbf{B}\mathbf{u}(t\text{-}1) \tag{A8.8}$$

$$\boldsymbol{\Phi}(t|t\text{-}1) = \mathbf{A}\,\boldsymbol{\Phi}(t\text{-}1|t\text{-}1)\,\mathbf{A}^T + \mathbf{D}\,\mathbf{R}_w\,\mathbf{D}^T \tag{A8.9}$$

Correction (at sampling instants)

$$\hat{\mathbf{x}}(t|t) = \hat{\mathbf{x}}(t|t\text{-}1) + \boldsymbol{\phi}(t)\Big[y(t)-\mathbf{c}^T\hat{\mathbf{x}}(t|t\text{-}1)\Big] \tag{A8.10}$$

$$\boldsymbol{\phi}(t) = \boldsymbol{\Phi}(t|t\text{-}1)\mathbf{c}\Big[r_v + \mathbf{c}^T\boldsymbol{\Phi}(t|t\text{-}1)\mathbf{c}\Big]^{-1} \tag{A8.11}$$

$$\Phi(t|t) = \left[I - \phi(t)c^T\right]\Phi(t|t-1) \qquad \text{(A8.12)}$$

Remarks

(i) Like RLS, the KF produces its own error analysis via the covariance matrix Φ. However, it must be noted that its accuracy depends on the assumed system matrices **A**, **B**, **c**, **D**, and the noise covariances r_v and R_w.

(ii) Note the intuitive manner in which the KF deals with noise. An increase in r_v will decrease the gain ϕ thus decreasing the correction on $\hat{x}(t|t-1)$ (i.e. the noisier the measurement the less effect it is allowed to have on the estimate). An increase in the process noise R_w leads to an increase in the covariance $\Phi(t|t-1)$ so that confidence in the estimate is reduced and greater weighting is given to the measurement.

(iii) The KF equations can easily be extended to cope with multiple-output systems (replacing the scalar y by a vector **y**). The form of the equations is unchanged with the vector **c** being replaced by the matrix **C** and the noise variance r_v being replaced by a niose covariance matrix R_v. However, the scheme has the disadvantage of introducing the need for matrix inversion and it is simpler to treat the problem as a number of separate single-output subsystems.

■

The KF may also be used for parameter estimation, making the translation $\mathbf{x} \rightarrow \boldsymbol{\theta}$ and $\mathbf{c} \rightarrow \boldsymbol{x}$ leads to

Prediction (between sampling instants)

$$\hat{\boldsymbol{\theta}}(t|t-1) = \hat{\boldsymbol{\theta}}(t-1|t-1) \tag{A8.13}$$

$$\boldsymbol{\Phi}(t|t-1) = \boldsymbol{\Phi}(t-1|t-1) + \mathbf{R}_w \tag{A8.14}$$

Correction (at sampling instants)

$$\hat{\boldsymbol{\theta}}(t|t) = \hat{\boldsymbol{\theta}}(t|t-1) + \boldsymbol{\phi}(t)\left[y(t) - \boldsymbol{x}^T(t)\hat{\boldsymbol{\theta}}(t|t-1)\right] \tag{A8.15}$$

$$\boldsymbol{\phi}(t) = \boldsymbol{\Phi}(t|t-1)\boldsymbol{x}(t)\left[r_v + \boldsymbol{x}^T(t)\boldsymbol{\Phi}(t|t-1)\boldsymbol{x}(t)\right]^{-1} \tag{A8.16}$$

$$\boldsymbol{\Phi}(t|t) = \left[\mathbf{I} - \boldsymbol{\phi}(t)\boldsymbol{x}^T(t)\right]\boldsymbol{\Phi}(t|t-1) \tag{A8.17}$$

The important difference between RLS and the KF for parameter esimation is the way in which the algorithms are tuned for tracking variation in parameters. In the RLS algorithm the scalar forgetting factor λ is used to inflate all elements of the covariance matrix $\boldsymbol{\Phi}$ and in the KF the process noise covariance matrix $\mathbf{R}_w$ is used to inflate only selected elements in $\boldsymbol{\Phi}$. Thus the KF offers increased flexibility and selectivity and is advantageous in facilitating the use of *a priori* knowledge of the system.

■

Appendix 9. **The extended Kalman filter**

The extended Kalman filter (EKF) substantially increases the range of applicability of the Kalman filter, being able to handle a wider range of non-linear systems. Consideration is given to the non-linear system

$$\mathbf{x}(t+1) = f(\mathbf{x}(t), \mathbf{u}(t)) + \mathbf{D}\mathbf{w}(t) \tag{A9.1}$$

$$\mathbf{y}(t) = g\,\mathbf{x}(t) + \mathbf{v}(t) \tag{A9.2}$$

Essentially the EKF follows the same rationale as the KF and may be described as

(i) *a priori* extrapolation of the state vector using the non-linear model (A9.1),

(ii) linearisation of the state equation (A9.1) around the operating point to obtain the filter statistics,

(iii) linearisation of the output equation (A9.2) to carry out the Kalman correction.

Note that the output equation (A9.2) can usually be relaxed to the linear form of (A8.2), as is assumed to be the case here, so that the only real difference between the KF and EKF algorithms is linearisation of the state equation (A9.1).

The *a priori* estimate or prediction of $\mathbf{x}(t)$ given $\mathbf{x}(t-1)$ is given as

$$\hat{\mathbf{x}}(t|t-1) = f(\hat{\mathbf{x}}(t-1), \mathbf{u}(t-1)) + E\left[\mathbf{D}\mathbf{w}(t-1)\right]$$

and since $\mathbf{w}$ is assumed to be white,

$$\hat{\mathbf{x}}(t|t-1) = f(\hat{\mathbf{x}}(t-1), \mathbf{u}(t-1)) \tag{A9.3}$$

Based on the assumption that the system can be approximated by a linear system along the trajectory between $\hat{\mathbf{x}}(t-1|t-1)$ and $\hat{\mathbf{x}}(t|t)$, the non-linear function f may be regarded as an approximate linear function described by the Jacobian $\mathbf{J}$,

$$\mathbf{J}(t-1) = \left.\frac{\partial f}{\partial \mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}(t-1|t-1)} \tag{A9.4}$$

i.e. the matrix whose ij^{th} element is the partial derivative of the i^{th} element of f with respect to the j^{th} element of $\mathbf{x}$.

For small deviations of $\mathbf{x}$ around $\hat{\mathbf{x}}(t\text{-}1|t\text{-}1)$

$$f(\mathbf{x}(t\text{-}1|t\text{-}1), \mathbf{u}(t\text{-}1)) = f(\hat{\mathbf{x}}(t\text{-}1|t\text{-}1), \mathbf{u}(t\text{-}1)) + \mathbf{J}(t\text{-}1)\left(\mathbf{x}(t\text{-}1) - \hat{\mathbf{x}}(t\text{-}1|t\text{-}1)\right)$$

+ higher order terms

so that for a first oder approximation

$$\mathbf{x}(t) = f(\hat{\mathbf{x}}(t\text{-}1|t\text{-}1), \mathbf{u}(t\text{-}1)) + \mathbf{J}(t\text{-}1)\left(\mathbf{x}(t\text{-}1) - \hat{\mathbf{x}}(t\text{-}1|t\text{-}1)\right) + \mathbf{D}\mathbf{w}(t\text{-}1)$$

or alternatively

$$\mathbf{x}(t) = \mathbf{J}(t\text{-}1)\mathbf{x}(t\text{-}1) + \mathbf{D}\mathbf{w}(t\text{-}1) + \left(\text{terms in } \hat{\mathbf{x}}(t\text{-}1|t\text{-}1)\right) \tag{A9.5}$$

If (A9.5) approximately holds over the piecewise linear trajectory then the equivalent error covariance matrix for the non-linear system is

$$\Phi(t|t\text{-}1) = \mathbf{J}(t\text{-}1)\Phi(t\text{-}1|t\text{-}1)\mathbf{J}^{T}(t\text{-}1) + \mathbf{D}\mathbf{R}_{w}\mathbf{D}^{T} \tag{A9.6}$$

The correction stage for the system is given by

$$\hat{\mathbf{x}}(t|t) = \hat{\mathbf{x}}(t|t\text{-}1) + \phi(t)\left[y(t) - \mathbf{c}^{T}\mathbf{x}(t|t\text{-}1)\right] \tag{A9.7}$$

$$\phi(t) = \Phi(t|t\text{-}1)\mathbf{c}\left[r_{v} + \mathbf{c}^{T}\Phi(t|t\text{-}1)\mathbf{c}\right]^{-1} \tag{A9.8}$$

$$\Phi(t|t) = \left[\mathbf{I} - \phi(t)\mathbf{c}^{T}\right]\Phi(t|t\text{-}1) \tag{A9.9}$$

The EKF is then, prediction using (A9.3) and (A9.6) followed by correction using (A9.7)–(A9.9).

It should be noted that in the case of the EKF $\Phi(t|t)$ is only an approximation to the error covariance matrix. It can only be taken as a indication of the estimation accuracy and there is no guarantee that divergence will not occur.

Application of the EKF for joint parameter/state estimation

Recall that the KF is essentially a method for estimating the state vector $\mathbf{x}$ of a linear system in which the system matrices are assumed to be known. It is often the

case that little is known about the values of **A** and **B** and it becomes necessary to identify the unknown parameters by some scheme such as RLS. The EKF can be configured for joint parameter/state estimation whereby parameters and states are estimated simultaneously. Consider the system defined by the augmented state vector

$$\mathbf{z}^T = [\, \mathbf{x}^T \,:\, \mathbf{a}^T \,:\, \mathbf{b}^T \,] \tag{A9.10}$$

where $\mathbf{a}^T$ and $\mathbf{b}^T$ are the vectors corresponding to the estimated parameter vector $\boldsymbol{\theta}$ (i.e. $\mathbf{z}^T = [\, \mathbf{x}^T \,:\, \boldsymbol{\theta}^T \,]$). This leads to the following state equations

$$\mathbf{z}(t) = \begin{bmatrix} \mathbf{x}(t) \\ \hline \mathbf{a}(t) \\ \hline \mathbf{b}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A}\mathbf{x}(t-1) + \mathbf{B}\mathbf{u}(t-1) \\ \hline \mathbf{a}(t-1) \\ \hline \mathbf{b}(t-1) \end{bmatrix} + \begin{bmatrix} \mathbf{w}(t-1) \\ \hline 0 \\ \hline 0 \end{bmatrix} \tag{A9.11}$$

$$\mathbf{y}(t) = [\, \mathbf{c}^T \,:\, 0 \,:\, 0 \,]\, \mathbf{z}(t) + \mathbf{v}(t) \tag{A9.12}$$

Note that these equations, in contrast to the initial state equations, are non-linear (or strictly bilinear) since they now involve products of the elements of $\mathbf{z}(t)$, thus

$$\begin{aligned} \mathbf{z}(t) &= f(\, \mathbf{z}(t-1), \mathbf{u}(t-1) \,) + \mathbf{w}(t-1) \\ \mathbf{y}(t) &= \mathbf{c}^T\mathbf{z}(t) + \mathbf{v}(t) \end{aligned} \tag{A9.13}$$

where

$$\mathbf{w}^T(t-1) = [\, \mathbf{w}^T(t-1) \,:\, 0 \,:\, 0 \,]$$

and

$$\mathbf{c}^T(t-1) = [\, \mathbf{c}^T(t-1) \,:\, 0 \,:\, 0 \,]$$

It is readily observed that the problem of joint parameter/state estimation is one which the EKF is directly applicable. Note that the function f, being bilinear, is easily differentiable to form the Jacobian **J**.