SELF-TUNING CONTROLLERS VIA THE STATE SPACE

by

KEVIN WARWICK

January 1982

A thesis submitted for the degree of Doctor of Philosophy of the University of London and for the Diploma of Membership of the Imperial College.

Department of Electrical Engineering
Imperial College
London SW7 2BT

ABSTRACT

The field of self-tuning has developed due to the need for the control of systems, with unknown parameters, that are affected by stochastic disturbances. Most of the theory has been centred around polynomial controller descriptions based on a system considered as an autoregressive moving average process.

In this thesis the system model is retained to enable an estimation of the system parameters to be carried out. However, once these estimations are obtained, a state space approach is used to calculate the required control action. By use of this scheme a controller has been achieved which has different properties, some advantageous, to those previously encountered. The resultant recursive control can be calculated simply and is such that it deals with a variable system time delay by employing a pole placement technique. With regard to a simple extension from its primary form, the controller is able to deal with non-zero external inputs by means of steady state following.

The self-tuning property of the state space method is proven, and from the proof it is shown that both this approach, and certain schemes previously employed, are special cases of a more generalised format. Subsequently, by taking account of this format, a tuner is suggested which incorporates a control input dependent on state feedback used in combination with linear output feedback, the state feedback providing pole placement and the linear output feedback allowing the variance of the system output to be optimized. The addition to the controller is shown to affect neither the use of an external input nor the overall self-tuning property.

Observer theory connected with each of the control schemes is discussed with a view to the formation of optimal and non-optimal observers , and where possible simulations are employed to show the nature of the various conditions obtainable. The systems considered are, for the most part, single-input-single-output, although the extension to the multivariable case is looked at.

The use of the state space in self-tuning has given rise not only to alternative tuner operating techniques, but also to a deeper theoretical understanding of self-tuning in general, and this widens its field of applications to encompass areas in process control where state space theory predominates.

PREFACE

In the past decade two major factors have influenced the theory connected with control systems. The ever increasing use of computers and more especially minicomputers has pointed the way towards an expansion in the practical application of control algorithms which take account of this method of implementation. More complex theory has developed because of the wider handling capabilities thus achieved and from this much control system design is now based on a state space representation rather than the more classical frequency analysis or root locus techniques.

The field of self-tuning controllers, which has arisen due to the improvement in computing power, has nevertheless remained in a polynomial system description framework. This thesis is intended to extend the field of self-tuning to the state space, where it is hoped further developments and increased awareness will result from its new found cohesion with existing modern control procedures.

The first chapter serves as an explanation of how self-tuning has evolved in its own right from previous adaptive control schemes. Basic definitions, used throughout the rest of the text are included, and previous, polynomial based, controllers are reviewed.

The state space formulation is discussed in Chapter 2, and the resulting control action compared with that obtained from previous methods. Hence it is shown how, as well as arriving at an equivalent state space controller, by using other means of reconstructing the state, certain advantages over the polynomial forms can be achieved.

The properties of the state space controller are discussed in Chapter 3, where the proof of the general self-tuning property of pole placement type controllersis given as a starting point. Included in this chapter is the application of an external reference input signal, where it is explained how a state space description can be of benefit.

Chapter 4 is used to provide extensions to the basic state space scheme and to show how various plant conditions, e.g. non-linearities, multivariable design; affect the fundamental assumptions. Of significance in this chapter is the section in which linear output feedback is employed in combination with state feedback, providing, in an output optimization sense, an improved control action under the pole placement criterion.

The underlying filtering and observer theory is considered, as a slight diversification, in Chapter 5, where the techniques behind the numerous control schemes are discussed. This comparison is, of course, only possible by means of the state space representation. The thesis is set out in a way such that Chapter 5 may be read after Chapter 2 if so desired, with hopefully no loss of continuity, although in its present position it also serves as an explanation of certain results contained in Chapters 3 and 4.

In conclusion I would like to express my gratitude and sincere appreciation to all those who have helped me in the completion of this work. Firstly, and most importantly, I wish to thank my supervisor, Professor J.H. Westcott, but I must extend my thanks to include Professor P. Antsaklis, Professor K.J. Åström and Dr. D.W. Clarke, who all took time

for useful discussion with me in the earlier stages.

The helpful comments and suggestions from my fellow research students and departmental lecturers will never be forgotten, and to this end Dr. A. Voreadis must be singled out for a special mention.

Without the financial support of the U.K. Science and Engineering Research Council I would be heavily in debt, and therefore I acknowledge, gratefully, their assistance.

Finally I wish to thank Mrs. D. Abeysekera for her careful typing of the manuscript, which included the correction of my numerous spelling mistakes.

Glossary of Terms Used

Sec. i.j.	Represents Chapter i, Section j
Sec. i. j(k)	Represents Chapter i, Section j, subsection k.
(i.j.k)	Denotes equation number k in Chapter i, Section j.
Fig. i.j.	Denotes figure number j in Chapter i.
$E\{\cdot\}$	The expected value of the function contained within the parentheses.
z^{-1}	Discrete time backward shift operator.
$y(s), u(s)$	Output and Input signals respectively, in the frequency domain.
$G(s)$	Frequency domain transfer function.
τ_l	Transport delay (total)
T_l	Sampling period
$y(t), u(t), e(t)$	Output, Input and disturbance signals respectively, at time t in the discrete time domain.
k	Integer part of the system time delay.
τ	Fractional part of the system time delay.
$A(z^{-1}), B(z^{-1})$ etc.	Polynomials in the backward shift operator.
δ_{ij}	Kronecker delta; $i = j \rightarrow \delta_{ij} = 1$ $i \neq j \rightarrow \delta_{ij} = 0$
$\Phi(z)$	Spectral Density function.
Ω	Covariance function.
ARMA	Autoregressive Moving Average.
CARMA	Controlled ARMA.
k_m, k_{max}	Maximum value of k.
$\emptyset(t)$	Vector of regressors in parameter estimation (at time t), or otherwise a general cost function.

$\theta(t)$	Vector of parameter estimates. (at time t)
$\lambda(t)$	Variable forgetting factor.
n,m,p	Denote dimension or number of parameters, n often has a subscript, e.g. n_b.
$V(\bar{\theta})$	Maximum Likelihood function.
$\varepsilon(t)$	Prediction error estimate.
$\hat{A}(z^{-1}),\hat{B}(z^{-1})$,etc.	Polynomials in the backward shift operator containing estimated parameters.
$\underline{x}(t)$	State vector at time t.
$\hat{\underline{x}}(t)$	Estimate of state vector at time t.
$\tilde{\underline{x}}(t)$	State vector at time t formed using estimated parameters.
$\hat{\tilde{\underline{x}}}(t)$	Estimate of $\tilde{\underline{x}}(t)$.
S_1,S_2,S_3,S_2'	Weighting values in L.Q.G. design
$[\ ]^T$	Transpose of a matrix, vector or in the limiting case a scalar value.
T	Time domain or matrix containing specified pole polynomial parameters.
t	Present time instant.
F or F(t)	State feedback vector.
$\hat{\underline{x}}'(t)$	Modified state estimate.
W	Matrix used in pole placement design.
0	Null (zero) matrix
$\Delta(t)$	Reconstruction error at time t.

CONTENTS

CHAPTER 1

INTRODUCTION TO SELF TUNING CONTROLLERS

Adaptive control is a general technique of which self-tuning is a special, recently initiated, classification. Hence the purpose of the early sections of this chapter is to consider the developments in adaptive control that have led to the evolution of self-tuning. The need for adaptive control systems is stressed in Sec. 1.1, and in order to compare self-tuning with other methods, various adaptive control schemes are considered in Sec. 1.2, where the overlap between different approaches is also discussed.

When examining any adaptive system at length, basic definitions, concerning the system to be controlled and the structure of environmental disturbances affecting the plant, must be given. These are outlined in Sec. 1.3, in which system models, particularly relevant to the field of self-tuning, are formulated. However, these systems are considered to contain parameters which are unknown or slowly varying with respect to time, and hence in Sec. 1.4 some identification algorithms are mentioned, whereby the system parameters are continually estimated such that a controller may be formed from these estimations. This controller may be deduced with regard to a specific control objective, several of which are discussed in Sec. 1.5, where the various methods fundamental to self-tuning are introduced.

Throughout this chapter reference is given, where possible, to applications of the differing techniques, although primarily the tendency is more towards the theoretical aspects.

1.1 The Necessity for Adaptive Controllers

The fundamental principles of control theory were formulated because of the need to explain practical physical control processes. As improvements and refinements were made to the original theory, its field of usage diversified to include abstract reasoning and theoretical concepts in a mathematical sense. This broadening of the subject area paid dividends, as many results and findings from other sources, such as mathematics, could be redeployed for its development.

When separated from the real world, however, it becomes difficult to measure the success of one achievement in comparison to any other, the only reasonable solution being a return to the original practical control system. In this work, consideration has been taken of the practical implications when applying a particular theoretical control idea.

The driving force behind the development of control theory has been the need for improved performance control systems, where a satisfactory controller designed using intuitive reasoning is no longer deemed appropriate. However, this theory is not enough in practical design problems, as a certain amount of knowledge concerning the dynamics of the process to be controlled is assumed. This knowledge can be difficult to obtain from essentially physical characteristics or when variations, which may be unforeseen and of large magnitude, occur in the plant. Thus a need has arisen for controllers which deal with these phenomena : - a gap filled potentially by a class of control systems called Adaptive Controllers.

Many different Adaptive Controllers are in use at present, this is due primarily to the large number of possible measures of plant performance. In any specific design project the type of Adaptive Controller used is dependent on whichever measure of system performance is considered to be most important, although in many cases numerous performance indices are used at the same time.

As may be expected from their large number, Adaptive Controllers exhibit several similarities concerning their basic structure. Because of this it is possible to split them up into various groupings in order to analyse their behaviour, each group containing controllers with either a common performance objective or those formulated via identical building blocks. The most popular of these are described in Sec. 1.3.

Despite the differences that arise, not only with each other, but also with alternative control system forms, Adaptive Controllers have a problem in common with all other controllers, namely the determination of the parameters used to describe the plant it is wished to control, i.e. the problem of parameter identification. This assumes that we have at least some apriori knowledge about the structure of the mathematical description of the system in which the parameters are contained. If this assumption holds, which for the moment is considered to be the case, then, by means of the identification exercise, a controller may be formulated, making use of the identified parameters, which causes the system to react in the same way, or as near as the particular identification technique will allow, as the system would have reacted if the same controller had been obtained by using the actual system parameters, had they been known.

But, as indicated earlier, in practice the unpredictable nature in which the parameters, associated with the plant, vary with respect to time, means that the Adaptive Controller must repeat the identification process each time the control law is applied such that the most recently available information is used. The optimum solution obtained using these ideals is however, invariably impossible to apply computationally, as the level of performance required is not practically possible. Thus suboptimum schemes are used, whereby the plant parameters are only estimated, rather than being identified, and this leads again to many more adaptive techniques because of the type of parameter estimator, of which there are many, chosen. In Sec. 1.4, several of these estimators are considered in more detail.

The ultimate aim of the designer of an adaptive controller is that, despite the plant parameter variations that may occur, the desired performance of the overall control system is unimpaired. This may be carried out in a number of ways. Firstly, when variations in the dynamic parameters are of small magnitude and remain around their original nominal value, a feedback control law may be applied such that the performance of the overall system is relatively unaffected by the variations. This is the basis for the theory related to control system sensitivity. Secondly, the parameters forming the control law can be altered when consideration is taken of the on line measurements of system parameters. Finally, a comparison may be made between the actual index of performance and the index of performance which has been chosen as the basis for the design procedure. From the result of this comparison, the parameters in the feedback control law are altered in an attempt to cause the difference to tend to zero.

Of the approaches mentioned, the second and third are both adaptive controllers, where the problem becomes one of correct choice of adaption law, such that the desired performance index is followed, despite variations in the plant or its environment.

The complexity of these controllers is largely dependent on their mode of employment, whether it is to be applied in analog or digital form. The increasing acceptance of microprocessor based control systems has meant that modern control theory, due to its application in the field of direct digital control, has recently been developed with the possibility if it being applied via a microprocessor being considered. This has resulted in many control algorithms previously disgarded, or not initially taken up, now being reviewed due to the decreasing cost of computing facilities. The continual improvements on the hardware side, in respect of computing time and capabilities, also mean that algorithms considered too cumbersome now, may well be reintroduced due to their simple and efficient operation in a few years' time.

1.2 Adaptive Control Systems

In the design of high performance control systems it is often the case that an adaptive control scheme is required. Of the many approaches carried out, several are considered here.

1. Early Adaptive Controllers

In Mishkin and Braun (1961), many of the adaptive controllers developed by that time are considered. Most prevalent among these is the use of a deterministic system model, where the stability or sensitivity of the system in response to environmental disturbances or alterations in the values of the system parameters is studied. The

controller is designed such that despite the parameter variations, the performance of the overall system remains unimpaired. This is done by an increase in the feedback gain subject to a maximum possible value, above which the closed loop system becomes unstable. Despite the use of nonlinearities in the feedback path, which can increase the range of system operation, designs based on this principle are only useful when small variations occur in the plant parameters.

Rather than making use of the alterations in the feedback path, a manipulation of the forward system gain can be used to retain either the closed-loop poles or closed-loop zeros, or even both, in previously specified zones, i.e. to ensure stability. This method, where the parameters of the system are altered on line in response to a measured system performance, brings the techniques mentioned above more in line with most of the recent work. In Bryson (1977) the method of forward gain tuning is found to operate sufficiently, and is shown in certain cases to be a more suitable choice than other more recently developed approaches. Of the more recent adaptive controllers, there are perhaps two general classifications. In the first of these the controller parameters are adjusted in a way dependent on the estimated parameters or state variables of the unknown plant. These controllers have come to be known under the heading of Self-Tuners, Åström and Wittenmark (1973). In the second method the parameters of the controller are adjusted to cause the difference between the actual output of the plant and the output of the model of the plant, used as reference, to tend asymptotically to zero. No identification or estimation scheme is explicitly used in this method which is known as Model Following Adaptive Control, Landau (1979). This latter approach will be considered in the next subsection.

2. Model Reference Adaptive Control

There are several constituent parts to a Model Reference Adaptive control system, the first of these being a reference model, which is used to specify the required performance of the system. The plant, which is to be controlled, must be adjustable in that it can contain variable forward and feedback gain characteristics. It is desired that the performance of this plant will follow that of the reference model as closely as possible, and to this end the difference between the reference model and plant outputs or states needs to be measured. Finally an adaptation process is formed which reacts to the measured error by altering either the plant or a controller in accordance with a set pattern.

The first model reference adaptive systems were those based on a policy of local parameteric optimization, where a quadratic performance index, dependent on the state or output error, is formulated and used to adjust the controller parameters subject to an optimization procedure. Of the optimization methods available, those of the gradient method, the steepest descent method and the conjugate gradient method have been widely used, Landau (1979). The main drawback with this procedure in general, however, is the possible lack of stability in the overall system.

The stability problem caused Model Reference systems to be designed as though they were primarily and solely a stability problem themselves, and from the result a stable adaptive controller could be obtained. To meet these needs Lyapunov functions were applied in various configurations, Lindorff and Caroll (1973). However, it is difficult, in consideration of a particular application, to choose the

best adaptive algorithm, for that situation, from those available. In fact, the largest possible number of stable controllers is required, from which the choice can be made. Because of this the most recent model reference adaptive controllers have been designed using hyperstability and positivity ideas, Landau (1979). In Narendra and Valavani (1979) this latter method is compared with the following.

3. Stochastic Controllers

In several controllers the process is identified prior to system operation, and the parameters of the controller are then computed on-line, Kalman (1958) used a deterministic plant model. Although this provided the foundation for adaptive control, disturbances in the environment and the system parameters are now usually included in a stochastic system model, surveyed in Wittenmark (1975) and the varying approaches being compared in Jacobs and Saratchandran (1980). Stochastic theory in the form of dynamic programming, in which a probability measure is placed on an uncertain random model has given a deeper mathematical understanding, and it has been found that when a control input is active it will affect the uncertainty in the future as well as the present value of the state, this is called dual control.

In Wittenmark (1975) a hierarchical structure was given for stochastic adaptive controllers, where both suboptimal and optimal dual control schemes were included, the latter being more complex to apply. Non-dual schemes were given as certainty equivalence and cautious, once again the latter being more complex, although both being simpler than the dual schemes. The simplest approach, therefore,

is that of certainty equivalence, of which Linear-Quadratic-Gaussian controllers are an example, van de Water and Willems (1981). However, due to the system parameters remaining unknown in an adaptive controller, there is some question as to whether or not the certainty equivalence principle is applicable.

4. Self-Tuning Controllers

Self-Tuning Control has invoked much recent interest, the initial work being carried out by Peterka(1970) and Åström and Wittenmark (1973). They concentrated on the object of minimizing the variance of the output signal, and although the first self-tuners, based on an implicit minimum variance scheme, proved successful in various practical applications, Åström et al. (1977), many difficulties had to be overcome before a more generalised scheme could be adopted.

Clarke and Gawthrop (1975) proposed a suboptimum control method which considered not only the system output, but also the input and set point variations. This took the form of a cost function incorporating the above values, weighted to meet certain specifications. Despite further problems, which have been encountered when using this approach, the application of the strategy on microprocessors was reported in Clarke and Gawthrop (1979b).

As an attempt to make the field of self-tuning more acceptable to practising engineers, explicit schemes involving pole and pole-zero techniques have been developed, Wellstead et al. (1979), Åström and Wittenmark (1980). The corresponding multivariable version of the pole placement self-tuner being found in Prager and Wellstead (1981).

The convergence and stability of self-tuners is discussed in

Gawthrop (1979) with the relationships between self-tuning and Model Reference controllers being considered in Gawthrop (1977) and Egart (1978).

1.3 System Models

There are many different models used to describe a deterministic system. Of these , the transfer function representation is that which is most commonly used in industry. From this, by means of classical control theory stability margins, compensation techniques and response characteristics may be used to improve the performance of the system.

This work, however, is concerned with the application of digital control, and hence the design of Sampled Data systems, based on a z-transform system description.

1. Deterministic System Models

The transfer function of a system is a mathematical description of its input to output performance, and in the case of a continuous time system, may be given in the s-domain, obtained from the Laplace Transform. Consider

$$y(s) = G(s) \cdot u(s) \tag{1.3.1}$$

where G(s) is the time-invariant transfer function, y(s) the system output and u(s) the system input. The transfer function may also be described as

$$G(s) = e^{-s\tau_1} \frac{B_1(s)}{A_1(s)} \tag{1.3.2}$$

where $A_1(s)$ and $B_1(s)$ are polynomials in the s-domain, τ_1 being the overall transport delay.

For the purpose of digital control, this continuous time signal must be sampled every T_1 seconds, and to this end the zero-order hold is introduced. The sampled signal is now converted into a sequence of steps, such that each step assumes the signal level at the sampling instant and remains at that level for one complete period, until the continuous signal is sampled once more. The s-domain representation for the zero-order hold is given by

$$G_0(s) = \frac{1-e^{-sT_1}}{s} \tag{1.3.3}$$

From which the discrete-time transfer function between u(t), the value of the input signal at time t, and y(t), the system output at time t, may then be obtained by taking the z-transform of $G_0(s).G(s)$, to give ,

$$y(t) = z^{-k} \frac{B_1(z^{-1})}{A_1(z^{-1})} u(t) \tag{1.3.4}$$

where $$z \triangleq e^{+sT_1} \tag{1.3.5}$$

In (1.3.4) k is the system integer time delay, where the following hold : (1) $k \geq 1$, (2) $0 \leq \tau \leq T_1$ and (3) $\tau = \tau_1 + (1-k)T_1$.

The polynomials in the difference equation being defined by

$$\left.\begin{aligned} A_1(z^{-1}) &= a_0' + a_1'z^{-1} + a_2'z^{-2} + \ldots + a'_{n_{a_1}} z^{-n_{a_1}} \\ B_1(z^{-1}) &= b_0' + b_1'z^{-1} + b_2'z^{-2} + \ldots + b'_{n_{b_1}} z^{-n_{b_1}} \end{aligned}\right\} . \tag{1.3.6}$$

in which z^{-1} is the backward shift operator, such that $z^{-i}\,u(t) = u(t-i)$. However, due to the definition of k given earlier $b_0 \neq 0$ and $a_0 = 1$, causing $A_1(z^{-1})$ to be monic. At the present time no restriction is placed on the order of the polynomials $A_1(z^{-1})$ and $B_1(z^{-1})$, denoted by n_{a_1} and n_{b_1} respectively.

2. Stochastic Disturbance

The system is affected by a disturbance, which it is assumed can be represented by the following,

$$e_1(t) = \frac{C_1(z^{-1})}{C_2(z^{-1})}\, e(t) \tag{1.3.7}$$

where $\{e(t),\ t = 0,\ \pm1,\ \pm2,\ \ldots\}$ is a white noise sequence with zero mean and finite variance, such that

$$E\{e(i)e^{T}(j)\} = \delta_{ij}\Omega \tag{1.3.8}$$

and δ_{ij} is the Kronecker delta.

$C_1(z^{-1})$ and $C_2(z^{-1})$ are of a similar form to the polynomials defined in (1.3.6), and no generality is lost by scaling the disturbance such that both of these polynomials are monic.

An in depth view of stochastic processes can be found in many works, e.g. Melsa and Sage (1973), and it is not intended to provide such a coverage in this work, although some special cases of the Autoregressive Moving Average model (ARMA), given in (1.3.7), will be considered. The first case being when $C_2(z^{-1})$ is unity , which leads to

$$e_1(t) = C_1(z^{-1})e(t) \tag{1.3.9}$$

the model now being called a moving average process. Conversely, when $C_1(z^{-1})$ is unity

$$C_2(z^{-1})e_1(t) = e(t) \tag{1.3.10}$$

and the model is termed Autoregressive.

As the signal e(t) is a white noise sequence, by the Spectral factorization theorem, Åström (1970), it is always possible to describe the spectral density of the stationary process $e_1(t)$ as

$$\Phi(z) = \sum_{i=0}^{\infty} \Omega_i z^{-i} = \frac{C_1(z^{-1})C_1(z)}{C_2(z^{-1})C_2(z)} \Omega \tag{1.3.11}$$

where $\Omega_i = E\{e_1(t),e_1(t+i)\}$. The process $e_1(t)$ therefore has a rational spectral density.

3. The complete Model

Due to the linearity of the system the superposition principle can be employed such that all disturbances acting on the system may be portrayed as one single noise, whereby the deterministic model is reformulated to give

$$y(t) = z^{-k} \frac{B_1(z^{-1})}{A_1(z^{-1})} u(t) + e_1(t) \tag{1.3.12}$$

or by using (1.3.7),

$$y(t) = z^{-k} \frac{B_1(z^{-1})}{A_1(z^{-1})} u(t) + \frac{C_1(z^{-1})}{C_2(z^{-1})} e(t) \tag{1.3.13}$$

From this latter equation, it is readily observed, that the overall model may be reconsidered by dividing the numerator polynomials by their respective denominators. However, the order of either of the two resultant

polynomials will most likely be much higher than any of those in (1.3.13), and this is extremely undesireable. By multiplying both sides of equation (1.3.13) by $A_1(z^{-1})C_2(z^{-1})$, and cancelling through by any common factors, the following form is arrived at,

$$A(z^{-1})y(t) = z^{-k}B(z^{-1})u(t) + C(z^{-1})e(t) \tag{1.3.14}$$

where the polynomials are redefined as

$$\left.\begin{aligned} A(z^{-1}) &= a_0 + a_1z^{-1} + a_2z^{-2} + \ldots + a_{n_a}z^{-n_a} \\ B(z^{-1}) &= b_0 + b_1z^{-1} + b_2z^{-2} + \ldots + b_{n_b}z^{-n_b} \\ C(z^{-1}) &= c_0 + c_1z^{-1} + c_2z^{-2} + \ldots + c_{n_c}z^{-n_c} \end{aligned}\right\} \tag{1.3.15}$$

also $\{y(t) : t \in T\}$ and $\{u(t) : t \in T\}$ are sequences of endogenous and exogenous variables, respectively, measured at the sampling instants. Without loss of generality $a_0 = c_0 = 1$ and

$$E\{u(i)e^T(i)\} = 0 \tag{1.3.16}$$

By defining $n_1 = \max\{n_a, n_b, n_c\}$ (1.3.17)

the polynomials can be increased in order, if this is necessary, such that they contain coefficients up to and including the n_1th terms. If the original order of a particular polynomial was less than n_1, the higher order terms are zero; i.e.

if $A(z^{-1}) = 1 + a_1z^{-1} + a_2z^{-2}$; and $n_1 = 4$; then $a_3 = a_4 = 0$

The overall system model given by (1.3.14) is called a Controlled-Autoregressive-Moving-Average-model (CARMA), and this has been extensively used in control theory, to a great extent because of its simplicity.

4 Notes on the CARMA Model

Although the model (1.3.14) has been derived using a Single-Input-Single-Output (SISO) approach, its extension to a multivariable system is carried out simply if the constituent polynomials are merely converted to polynomial matrices. This multivariable form can, however, cause problems when certain control strategies are applied, Prager and Wellstead (1981), and this will be looked at in depth in Sec. 4.2.

The use of the SISO model in time varying systems is identical to that in the time-invariant case, however, due to the effect of the superposition theorem in its formulation, when non-linear systems are to be considered, measurement noise must be treated as a separate entity, as in Gawthrop (1977). The superposition theorem, though, allows for the assumption to be made that the $C_1(z^{-1})$ and $C(z^{-1})$ polynomials, obtained from the canonical forms (1.3.13) and (1.3.14), have all their zeros inside or on the unit circle in the z-plane, Åström (1970), and this is an important result in the stability analysis of closed-loop systems. If it were not the case that $C_1(z^{-1})$ had stable zeros†, then by driving the input to zero, any unstable modes of $A_1(z^{-1})$, which were not also unstable modes of $C_1(z^{-1})$, would not appear at the output.

The stability of the C polynomials is a necessary condition for the design of minimum variance controllers, which are based on a prediction of the output signal at time t+k, i.e. k-steps ahead of the present instant, as the errors inherent in the controller must be allowed to decay. The additional assumption, therefore, has to be made,

†The term 'stable zeros' is used to indicate that the roots of a polynomial lie inside the unit disc in the z-plane.

with this controller, that the polynomial concerned has no zeros on the unit circle, in order to remove the problem of critical stability.

By comparison of (1.3.13) and (1.3.14), it is apparent that iff $A_1(z^{-1}) = C_2(z^{-1})$, then $B_1(z^{-1}) = B(z^{-1})$ and $A_1(z^{-1}) = A(z^{-1})$. When $A_1(z^{-1}) \neq C_2(z^{-1})$ the assumption must be made that $A_1(z^{-1}) = A(z^{-1})$ $C_3^{-1}(z^{-1})$, where $C_3(z^{-1})$ is monic, and $A(z^{-1})$ and $C_3(z^{-1})$ are relatively prime. From this the following equalities are obtained : 1) $B_1(z^{-1}) = B(z^{-1})C_3^{-1}(z^{-1})$ and 2) $C_1(z^{-1}) = C(z^{-1}) \cdot C_2(z^{-1})A_1^{-1}(z^{-1})C_3^{-1}(z^{-1})$, where $B(z^{-1})$ and $C_3(z^{-1})$ must also be relatively prime.

If these conditions apply, certain properties, set out below, relating to $B_1(z^{-1})$ also apply to $B(z^{-1})$. The first of these is the fact that if the fractional time delay, τ, becomes equal to zero, such that the overall transport delay is equal to an integer number of sampling instants, then the order of the $B_1(z^{-1})$ polynomial will fall by unity. In self-tuning, though, because of the computing time requirements in formulating the necessary control input, a fractional time delay is almost always present, and hence $\tau \neq 0$ almost always.

The second property concerns the steady state gain obtained by setting $z^{-1} = 1$ in the $B(z^{-1})$ polynomial, denoted by $B(1)$, this is found to be independent of the fractional time delay, which is an important property when a costing is placed on the control in optimal controller design. The proof of both of these properties being given in Appendix 1.1.

5. State Space Models

Much theory has developed around the use of a state-space

description of linear time-invariant systems, Rosenbrock (1970). Where stochastic systems are concerned the state-space model is found to be extremely versatile in dealing with disturbances. Consider the equations

$$\underline{x}(t+1) = \bar{A}\underline{x}(t) + \bar{B}u(t) + w(t) \qquad (1.3.18)$$

$$y(t) = \bar{C}\,\underline{x}(t) + \bar{D}u(t) + v(t) \qquad (1.3.19)$$

where $\{w(t)\}$ and $\{v(t)\}$ are two uncorrelated, zero-mean, white noise sequences with covariances $\bar{W}$ and $\bar{V}$ respectively. If there are $\bar{n}$ states , $\bar{p}$ inputs and $\bar{m}$ outputs the matrices are of the following dimensions,

$$\bar{A} : \bar{n}\times\bar{n} \; ; \quad \bar{B} : \bar{n}\times\bar{p}; \quad \bar{C} : \bar{m}\times\bar{n} \; ; \quad \bar{D} : \bar{m}\times\bar{p}$$

where the input and output signals are considered as vectors and the constituent matrices are assumed, at this stage, to have no relationship with the polynomials discussed earlier in this section. The $\bar{D}$ matrix, which provides a direct link between the control input and the system output, is in many cases equivalent to the null matrix, as will be seen later in this work.

In much of the control theory carried out, the matrices above are considered to be known, resulting in problems such as the estimation of the state vector, where this is not directly measurable, and providing a control input which is in some sense optimal.

However, if it is the case that these matrices are unknown, and therefore an identification technique is required, because of the high state-space dimension, which may be present, this can prove to be computationally inefficient. The problem is exaggerated in an Adaptive Control system where an on-line estimation scheme coupled with the computation of an up to date control law is necessary.

Of the estimation schemes available, though, the Extended Kalman filter, in which both states and unknown parameters are estimated in a composite state vector, is the most widely used method, and this is discussed in section 4 of this chapter. However, this estimation technique is itself nonlinear and hence the resulting estimates are invariably inaccurate.

Another approach is to carry out the estimation making use of the CARMA model (1.3.14), and to then formulate from the parameter estimates obtained, the state-space description. This has the advantage of relieving the problems mentioned concerning estimation via a state-space model, and yet the state-space theory available can be made use of in the design of the controller. The state-space description obtained from the parameters in the CARMA model is discussed at length in Chapter 2, but is given here as

$$\underline{x}(t+1) = \bar{A}\underline{x}(t) + \bar{B}u(t) + \bar{E}e(t) \qquad (1.3.20)$$

$$y(t) = \bar{C}\underline{x}(t) + e(t) \qquad (1.3.21)$$

This general representation may also be obtained from the equations (1.3.18) and (1.3.19), where $\bar{D}$ is the null matrix and the system is considered to be linear, such that the superposition theorem holds, Goodwin and Payne (1977).

The use of this model in the multivariable case, which is discussed in Sec. 4.2 , usually provides a high dimension state-space, although a minimal representation can be obtained by making use of other models.

1.4 Recursive Estimation for Discrete Time Processes

1. Introduction

When carrying out dynamic process identification, the parameters of the mathematical model, used to describe the system, are determined. A complete identification procedure can prove to be extremely time consuming, and hence various recursive parameter identifier algorithms are considered. These recursive identifiers are termed real time algorithms, as the parameters are estimated during controller operation with regard to only a limited supply of the available input-output information. This reduces the amount of computation necessary and allows for estimates of parameters, whose value varies with time, to be continually updated.

It must be remembered, though, that for reasonable results to be achieved with the recursive estimators the nature, i.e. the dimensions, of the mathematical model must be well known apriori. If a good knowledge of the system, subjected to the identification procedure, is not held, the model must be obtained by means of a structural identification process based on previous input-output details. It will be shown in Sec. 4.4, however, that where a self-tuning algorithm is to be used, this model fit can be down-graded and yet the resultant controller action will remain efficient.

Many algorithms available for system identification purposes were reviewed in Åström and Eykhoff (1971), although a more thorough approach is given in Goodwin and Payne (1977). Of the varying possible techniques, it has been shown, Sôderström et al (1974), that

overall there is no estimation technique which can be said to be better than all others. However, several algorithms have been found suitable for use in self-tuning controllers, and for the purpose of this section only these will be considered in any detail. The theory is extended to include the effects of approximate models and the consistency and convergence of the algorithms discussed.

2. Recursive Least Squares

Let the process be modelled by

$$A(z^{-1})y(t) = B'(z^{-1})u(t) \tag{1.4.1}$$

where no error term is present, and

$$B'(z^{-1}) = \beta_0 z^{-1} + \beta_1 z^{-2} + \ldots + \beta_{n_b+k_m-1} z^{-(n_b+k_m)} \tag{1.4.2}$$

Here k_m is equal to the maximum value of system integer time delay. This means that only k_m need be prespecified, thus a varying integer time delay may be accomodated in this model in so far as k, the actual system time delay, does not exceed k_m. The model may also be described, by inclusion of an error term, $\varepsilon(t)$, as

$$y(t) = \phi^T(t)\theta(t-1) + \varepsilon(t) \tag{1.4.3}$$

$$\text{where } \theta^T(t) = [a_1,\ldots,a_{n_a};\beta_0,\ldots,\beta_{n_b+k_m-1}] \tag{1.4.4}$$

$$\text{and } \quad \phi^T(t) = [-y(t-1),\ldots,-y(t-n_a);\ u(t-1),\ldots,u(t-n_b-k_m)] \tag{1.4.5}$$

The least squares method is now based on the minimization of the function,

$$s(\theta) = \sum_{t=t-n}^{t} (y(t) - \phi^T(t)\theta(t))^2 \tag{1.4.6}$$

where $\bar{n} = \max\{n_a, n_b+k_m\}$. (1.4.7)

The least squares estimate of $\theta(t)$, designated as $\hat{\theta}(t)$, can now be calculated by means of the recursive least squares algorithm, Åström and Wittenmark (1980), where

$$\hat{\theta}(t) = \hat{\theta}(t-1) + J(t)\phi(t)\,\varepsilon(t) \tag{1.4.8}$$

and $$J(t) = \{[I-K(t)\phi^T(t-1)]\,J(t-1)\}/\lambda(t) \tag{1.4.9}$$

in which $$K(t) = J(t-1)\phi(t-1)[1 + \phi^T(t-1)J(t-1)\phi(t-1)]^{-1}. \tag{1.4.10}$$

It is known, however, that numerical problems associated with the J(t) matrix can lead to instability, and hence an updating procedure which contains better numerical stability, such as the UD factor update of Bierman (1977, 1981), should be used.

In the equations (1.4.8→10), $\varepsilon(t)$ is an estimate of the one step prediction error, and denotes the difference between the output at time t, and the estimated output from the multiplication of regressors at time t and parameter estimates at time t-1, the last sampling interval.

The assumption must be made that the initial parameter estimates, characterised by $\hat{\theta}(0)$, and the initial values contained in the error covariance matrix, J(0), are known. This assumption is not important in itself, as the least squares estimator will generally still operate effectively for any initial values, although setting all initial parameter estimates to zero can lead to problems on start up. If any apriori information, concerning the parameters, is held, though, a faster convergence of the estimates can be achieved when this is used.

The major disadvantage of the recursive least squares method

is that biased estimates will result where the noise, given as $\varepsilon(t)$ in (1.4.3) is not white. However, its use in self-tuning problems has resulted from its computational simplicity, which may become an overriding feature in the design of a particular on-line process controller.

The parameter vector, $\hat{\theta}(t)$, can be shown, under weak conditions, to converge to a unique point, around which it is locally stable, Goodwin and Payne (1977). It is apparent, however, that the parameter vector will converge globally with a probability of one.

3. Variable Forgetting Factor

The factor $\lambda(t)$ in (1.4.9) is named the variable forgetting factor and is used for exponential forgetting of past data during the least squares process. Although this factor was originally chosen to be a constant of value $0.95 \leq \lambda(t) \leq 0.99$ for all t, it has been found that where process and measurement noise is of similar magnitude to that of interface circuitry, a variable forgetting factor must be employed, Fortescue et al (1979). The effect of a constant forgetting factor on such a system is to cause the parameters in $J(t)$ to become excessive under certain conditions, Åström and Wittenmark (1980). This can lead to large and rapid variations in both the output and the parameter estimates.

There are numerous solutions to the problems associated with $J(t)$, Clarke and Gawthrop (1979a). Firstly, by use of a suitable signal $|J(t)|$ may be retained within the necessary boundary limits. Secondly, if $|J(t)|$ or $J(t)\emptyset(t)$ exceeds prespecified bounds, then the $J(t)$ update can be discontinued, and finally a specific function

of $J(t)$ may be held constant by appropriate variations in $\lambda(t)$. This latter method has been dealt with in detail, by keeping the sum of squares of the least squares errors constant by means of varying the forgetting factor $\lambda(t)$, Fortescue et al (1979). However, to ensure convergence of the algorithm used in this method, trace $J(t)$ and thus $\| J(t) \|$ must always be bounded, Cordero and Mayne (1980), and hence the three methods mentioned above are by no means independent.

A forgetting factor which is much simpler to apply in practice, and requires less computation, has been proposed by Wellstead and Sanoff (1981), where

$$\lambda(t) = \lambda_1(t)\ \lambda_2(t) \qquad (1.4.11)$$

Here $\lambda_1(t)$ is used as an exponential start up factor, to enable the estimator to converge rapidly in the initial tuning period. $\lambda_2(t)$, meanwhile, is used to consider a specified number of points, e.g. 100, of residual magnitude. When the estimations are a good fit $\lambda_2(t)$ approaches unity, but when the converse is true larger residuals result and the value of $\lambda_2(t)$ falls.

This method, which has been found to operate efficiently and economically, was studied in depth by Omani (1981), where it was not only shown to be similar to that of Cordero and Mayne (1980) when certain conditions hold, but also was found to be more stable under rapidly varying circumstances.

4. Recursive Extended Least Squares

When the disturbance is non white an extension can be made to the least squares process to take account of this.

By redefining (1.4.3) as

$$y(t) = \phi^T(t)\theta(t-1) + C(z^{-1})\varepsilon(t) \tag{1.4.12}$$

where $$C(z^{-1}) = 1 + c_1 z^{-1} + c_2 z^{-2} + \ldots + c_{\bar{n}} z^{-\bar{n}} \tag{1.4.13}$$

and the degree of $C(z^{-1})$ is assumed to be $n_c \leq \bar{n}$. The new model (1.4.12) may now be rearranged to give a similar method to that of least squares, such that by extending the parameter and regressor vectors to include the terms associated with $c_i : i = 1, \ldots, \bar{n}$, the least squares terminology remains adequate. The vectors affected become,

$$\bar{\theta}^T(t) = [a_1,\ldots,a_{n_a}; \beta_0,\ldots, \beta_{n_b+k_m-1}; c_1,\ldots,c_{\bar{n}}] \tag{1.4.14}$$

and $$\bar{\phi}^T(t) = [-y(t-1),\ldots,-y(t-n_a); u(t-1),\ldots,u(t-n_b-k_m); \varepsilon(t-1),\ldots,\varepsilon(t-\bar{n})] \tag{1.4.15}$$

Hence the one-step prediction error estimate becomes

$$\varepsilon(t) = y(t) - \bar{\phi}^T(t)\hat{\bar{\theta}}(t-1) \tag{1.4.16}$$

and equations (1.4.8→10) can be applied to obtain estimates of the parameters in the vector $\bar{\theta}(t)$.

For the convergence of this extended procedure, a unique convergence point is only found when :

1) the polynomial defined by $C(z^{-1})$ is positive real.

i.e. Re $C(e^{i\omega}) > 0$; $\pi \leq \omega \leq \pi$

or 2) the system is an ARMA process.

Global stability with a probability of one is true iff

$\{2-C(z^{-1})\}[2C(z^{-1})]^{-1}$ is positive real, and in this case it is generally true that local stability is not a feasible concept.

5. Recursive Maximum-Likelihood

In its recursive form the Maximum-Likelihood method is an approximation to its off-line equivalent. One of the first descriptions of this method being given by Söderström (1973), where the basis used was the original off-line approach of Åström and Bohlin (1966). The method is now summarized as follows.

From (1.4.1) and (1.4.12), the prediction error estimate is given by,

$$\varepsilon(t,\bar{\theta}) = C^{-1}(z^{-1})\ \{A(z^{-1})y(t) - B'(z^{-1})u(t)\} \qquad (1.4.17)$$

such that the calculation of the maximum-likelihood estimates is equivalent to minimizing the function $v(\bar{\theta})$, where

$$V(\bar{\theta}) = \tfrac{1}{2} \sum_{t=t-n}^{t} \varepsilon^2(t,\bar{\theta}) \qquad (1.4.18)$$

Of the many possible approaches to this problem, the most applicable involves approximating the function $v(\bar{\theta})$ by a quadratic equation via a Taylor series expansion, such that

$$\varepsilon(t,\bar{\theta}) \simeq \varepsilon(t,\hat{\bar{\theta}}) + \bar{\phi}_1^T(t)(\bar{\theta}-\hat{\bar{\theta}}) \qquad (1.4.19)$$

and

$$\bar{\phi}_1^T(t) = \frac{\partial\varepsilon}{\partial\theta}(t,\hat{\bar{\theta}}) = [\hat{c}(z^{-1})]^{-1}\,\bar{\phi}^T(t) \qquad (1.4.20)$$

The prediction error estimate may now be calculated from the equation

$$\varepsilon(t,\hat{\bar{\theta}}) = [\hat{C}(z^{-1})]^{-1}\ [\hat{A}(z^{-1}y(t) - \hat{B}'(z^{-1})u(t)] \qquad (1.4.21)$$

and the recursive algorithm (1.4.8→10) may be carried out by replacement of $\bar{\phi}(t)$ by $\bar{\phi}_1(t)$.

The necessary calculations can, however, be simplified by use of the expression

$$\varepsilon(t,\hat{\bar{\theta}}) \simeq \varepsilon(t) = y(t) - \bar{\phi}^T(t)\hat{\bar{\theta}}(t-1) \qquad (1.4.16a)$$

where $\hat{C}(z^{-1})$ is replaced by unity in equation (1.4.21) only, i.e. the remainder of the recursive algorithm is unaltered. As a follow on from this, if $\hat{C}(z^{-1})$ is replaced by unity in equations (1.4.8→10) also, the recursive Maximum-Likelihood method becomes equivalent to the recursive least squares approach, showing that the former is a general method of which the latter is merely a special case. However, Ljung (1978) discussed the convergence of the prediction error identification method, Caines (1976) , of which Maximum-Likelihood is in turn a special case, and by using this even more general classification results were obtained for the consistency problem.

The approximate Maximum-Likelihood scheme results in a convergence, with probability one, to a local minimum of the likelihood function. However, although the convergence point is unique for ARMA processes, this is not universally applicable, and as a unique convergence point is a necessary condition for global convergence, a general condition is not possible.

An important practical consideration with the recursive Maximum-Likelihood method is the nature of the $\hat{C}(z^{-1})$ polynomial in (1.4.21). As the parameters of this polynomial are estimations, there is a possibility of instability occurring, and hence the prediction error estimate may become excessive. If this happens there may be a divergence of the parameters in the estimate vector. Therefore, this must be avoided by either placing a bound on $J(t)\emptyset(t)\varepsilon(t)$ or by reducing the same factor until the $\hat{C}(z^{-1})$ polynomial is stable.

6. Kalman Filtering Techniques

The Kalman Filter approach to system estimation, models the variations of the parameters with respect to time, by means of stochastic difference equations. As a state-space description is used, a Maximum-Likelihood estimate of the state is found, where the noise is considered as a normal distribution.

The full advantage of this technique is found when the model parameters are nonlinear. Numerous methods then become available by considering the current estimate as the norm, and linearizing the model about this norm. The general heading for this philosophy is Extended Kalman Filters, Jazwinski (1970), in which the states and parameters of the model are both recursively estimated. This type of filter, based on first-order linearization, can be made to converge globally, if required, Ljung (1979), and it was shown in Panuska (1980) that the recursive Maximum-Likelihood, and hence the recursive Extended Least Squares, schemes can be derived from the filter equations.

The states of the system, however, are not necessarily required, and thus a simplification of the Kalman Filter results. By regarding the system parameters as the states a reduced filter is achieved, and in Åström and Eykhoff (1971), the connection between this reduced filter and the previous recursive schemes mentioned is discussed.

7. Consistency and Convergence

Although the convergence properties of the various estimators have been considered in the preceeding text, to an extent, an in depth view can be found in Sôderström et al (1978) and Tsypkin et al (1981).

It is found that with the recursive Least Squares and recursive Maximum-Likelihood methods, a global convergence of the estimates to the actual system parameters is not possible, although the actual parameters are locally stable, possible, points of convergence.

The previously mentioned prediction error method of Caines (1976) is more complicated than the other approaches, and for this reason has not been considered at length, although Ljung (1978) proved its convergence with probability one to the set of best approximate models of the system considered.

Finally, many of the algorithms considered in this section are covered in great theoretical depth in Holst (1977), although for easy reference, that of Sôderström et al (1978) is prefered.

1.5 Self-tuning

Self-tuning controllers may be categorised as either implicit or explicit. In the implicit scheme the parameters of the system are estimated and these estimations are used directly in the formation of the next control input, Fig. 1.1. With explicit self-tuners, however, two approaches are considered. The first of these uses the parameters estimated from the CARMA model in the calculation of a cost function, from which the controller parameters are obtained. The second explicit method uses a further assumption that the CARMA model disturbance is a white noise process, the parameters estimated from this subsidiary model then being used to calculate the final controller form, Fig. 1.2.

Implicit and explicit self-tuning controllers

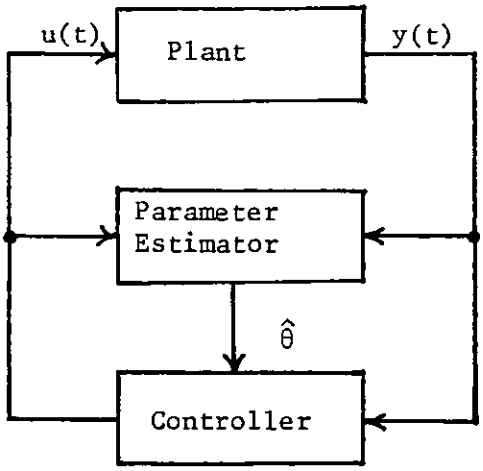


Fig. 1.1 Implicit self-tuning controller (no external input)

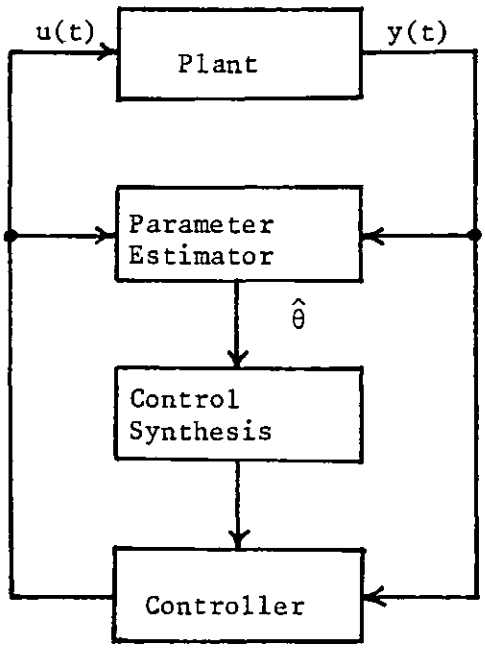


Fig. 1.2 Explicit self-tuning controller (no external input)

1. Minimum-Variance Self-Tuner

This implicit scheme uses the CARMA model (1.3.14) as its basis, this is restated as

$$A(z^{-1})y(t) = z^{-k}B(z^{-1})u(t) + C(z^{-1})e(t) \qquad (1.5.1)$$

where the polynomials are considered to be of order n_1.

It is now required that the cost function (1.5.2) is minimized.

$$\emptyset = E\{y^2(t)\} \qquad (1.5.2)$$

where $E\{\cdot\}$ denotes the expected value.

Before proceeding we make the assumption that the parameters contained in (1.5.1) are known, hence it is desired to derive a control law which minimizes $\emptyset$ by means of these known parameters. Rewriting (1.5.1) as,

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

it can be seen that the first output signal which will be affected by our choice of control input at time t, is that at time t+k. It therefore follows that by a correct selection of the present value, at time t, of the control input the variance of the output k-steps ahead can be reduced. However, the disturbance contains two distinctly separate sections. Firstly $e(t+1),e(t+2),\ldots,e(t+k)$; which, as they are future values, are unknown, and secondly $e(t), e(t-1),\ldots,e(t-n_1+k)$; which can be calculated from information obtained up to and including that at time t.

To separate these noise terms, the equality (1.5.3) is postulated,

$$C(z^{-1}) = A(z^{-1})E(z^{-1}) + z^{-k}G(z^{-1}) \qquad (1.5.3)$$

where $G(z^{-1})$ is of degree n_1-1 and $E(z^{-1})$ is monic and of degree $k-1$.

Substituting for (1.5.3) in the reformulated version of (1.5.1),

$$A(z^{-1})y(t+k) = B(z^{-1})u(t) + G(z^{-1})e(t) + A(z^{-1})E(z^{-1})e(t+k)$$

and substituting for e(t) from (1.5.1), using (1.5.3), gives

$$y(t+k) = \frac{B(z^{-1})E(z^{-1})}{C(z^{-1})}\,u(t) + E(z^{-1})e(t+k) + \frac{G(z^{-1})}{C(z^{-1})}\,y(t) \qquad \ldots \qquad (1.5.4)$$

By squaring both sides of (1.5.4) and taking expected values,

$$E\{y^2(t+k)\} = E\left\{\frac{B(z^{-1})E(z^{-1})}{C(z^{-1})}\,u(t) + \frac{G(z^{-1})}{C(z^{-1})}\,y(t)\right\}^2 + E\{E(z^{-1})e(t+k)\}^2$$

As the cost function (1.5.2) needs to be minimized, this is done simply, with regard to the above equation, by setting the control input to be,

$$u(t) = -\frac{G(z^{-1})}{B(z^{-1})E(z^{-1})}\,y(t) \qquad (1.5.5)$$

If expected values are now taken for both sides of the equation (1.5.4)

$$E\{y(t+k)\} = \left\{\frac{B(z^{-1})E(z^{-1})}{C(z^{-1})}\,u(t) + \frac{G(z^{-1})}{C(z^{-1})}\,y(t)\right\}$$

which is identically zero, iff the control input (1.5.5) is applied.

But the assumption was made earlier that the parameters contained in the CARMA model, (1.5.1), were known, this is now stated as not being the case. The parameters of the model must therefore be estimated and thence an estimate can be made of the required control input (1.5.5) from these parameter estimates.

For reasons of computational simplicity and problem matching, the recursive least squares estimation procedure, discussed in Sec. 1.4 is employed. This is used to estimate the parameters in the model,

$$y(t) = z^{-k}\{\beta(z^{-1})u(t) - A(z^{-1})y(t)\} + \varepsilon(t) \qquad (1.5.6)$$

where $\varepsilon(t)$ is a moving average of the noise $e(t)$. Also, the polynomials contained in (1.5.6) are defined as,

$$\left.\begin{aligned} A(z^{-1}) &= \alpha_0 + \alpha_1 z^{-1} + \ldots + \alpha_m z^{-m} \\ \beta(z^{-1}) &= \beta_0(1+\beta_1 z^{-1} + \ldots + \beta_\ell z^{-\ell}) \end{aligned}\right\} \qquad (1.5.7)$$

It is shown in Åström and Wittenmark (1973), that if the parameter estimates, contained in (1.5.7), converge, the minimum variance control scheme is provided by use of the control input,

$$u(t) = -\frac{A(z^{-1})}{\beta(z^{-1})}\, y(t) \qquad (1.5.8)$$

although, as all the parameters in the estimation model (1.5.6) cannot be uniquely obtained, β_0 must be either known or chosen prior to controller operation.

However, for the regulator to converge to the minimum-variance case, several factors must be accounted for. Firstly, and perhaps most importantly, the integral part of the system time delay, k, must be known exactly. This provides for a limiting usage of this type of tuner where varying delay systems are encountered. The second point concentrates on the dimension of the estimated model, and the respective property of self-tuning. If either $\ell = n_1+k-1$ and $m \geq n_1-1$ or $\ell \geq n_1+k-1$ and $m = n_1-1$ the self tuning property holds, where $A(z^{-1})$ and $\beta(z^{-1})$ are overparameterized, though, the condition of relative primeness is no longer assumed, and thus common factors will occur. In the converse case, however, when the model is of lower order than the plant, the property still holds, and moreover it has been found that experimentally the model may be of

much lower order than the actual process, unless large variations in the output signal occur, and a good control action is provided, Åström (1980).

A major drawback with the minimum-variance tuner is the fact that if the $B(z^{-1})$ polynomial, defined in (1.5.1), contains unstable zeros, instability will most likely occur due to the controllers' high sensitivity to variations in the parameters, Åström (1970). Even if the closed loop retains its stability, though, the control signal may well be excessive due to the objective of the controller being entirely centred on reducing the variance of the output signal from the plant.

Because of the problems mentioned above, various alternative methods have been considered, the first of which is looked at next.

2. Generalised Cost Function

In the previous method the cost function was entirely dependent on the system output signal, here a cost function is considered which incorporates inputs, outputs and set-points and is due to Clarke and Gawthrop (1975). This function is given as

$$\emptyset(t+k) = P(z^{-1})y(t+k) + Q(z^{-1})u(t) - R(z^{-1})w(t) \qquad (1.5.9)$$

where $P(z^{-1})$, $Q(z^{-1})$ and $R(z^{-1})$ are polynomial transfer functions, and $w(t)$ is a set-point external input.

In this instance it is desired to minimize $E\{\emptyset^2(t)\}$, and this is done, once again, by setting the predicted value of $\emptyset$ at time $t+k$ as being equal to zero, which can be interpreted as minimizing

$$E\{(P(z^{-1})y(t+k) - R(z^{-1})w(t))^2 + (Q'(z^{-1})u(t))^2\}$$

where the expected value takes into consideration all information up to and including time t.

By defining,

$$P(z^{-1}) = \frac{P_N(z^{-1})}{P_D(z^{-1})} \qquad (1.5.10)$$

From the combination of (1.5.1) and (1.5.9), we obtain

$$\bar{\phi}(t+k) = \bar{\phi}_y(t+k) + Q(z^{-1})u(t) - R(z^{-1})w(t) \qquad (1.5.11)$$

where $$\bar{\phi}_y(t+k) = \frac{G(z^{-1})}{C(z^{-1})P_D(z^{-1})}\, y(t) + \frac{E(z^{-1})B(z^{-1})}{C(z^{-1})}\, u(t) \qquad (1.5.12)$$

and $$E(z^{-1})A(z^{-1}) + z^{-k}\,\frac{G(z^{-1})}{P_D(z^{-1})} = C(z^{-1})P(z^{-1}) \qquad (1.5.13)$$

$\bar{\phi}$ denoting the least squares prediction of ϕ, and $P_N(z^{-1})$ and $P_D(z^{-1})$ are of order p_1 and p_2 respectively. Also, we have that $\phi_y(t+k) = P(z^{-1})y(t+k)$.

The actual cost function is thus dependent on its least squares prediction plus an error term.

$$\phi(t+k) = \bar{\phi}(t+k) + \varepsilon(t+k) \qquad (1.5.14)$$

where $\bar{\phi}(t+k)$ is the prediction of $\phi(t+k)$, made by taking into account all information available at time t, and may be written $\bar{\phi}(t+k/t)$.

Therefore, substituting for $\bar{\phi}(t+k)$ from (1.5.11), using (1.5.12)

$$\phi(t+k) = \frac{G(z^{-1})}{P_D(z^{-1})}\, y(t) + [C(z^{-1})Q(z^{-1}) + E(z^{-1})B(z^{-1})]u(t)$$

$$- \sum_{i=1}^{n_1} c_i \bar{\phi}(t+k-i) - C(z^{-1})R(z^{-1})w(t) + C(z^{-1})\varepsilon(t+k)$$

... (1.5.15)

The prediction model may then be obtained in one of several forms, different schemes being considered in Clarke and Gawthrop (1975) and (1979a).

Here, the following is used :

$$\phi(t) = \frac{\hat{G}(z^{-1})}{P_D(z^{-1})}\, y(t-k) + \hat{E}(z^{-1})u(t-k) - \hat{H}(z^{-1})w(t-k) + \xi(t) \qquad \ldots\ldots \quad (1.5.16)$$

where $\varepsilon(t+k) = E(z^{-1})\xi(t+k)$

The control signal is now chosen to set the k-step-ahead prediction to zero, and this can be seen from (1.5.15) to be,

$$u(t) = \left[C(z^{-1})Q(z^{-1}) + E(z^{-1})B(z^{-1})\right]^{-1} \left\{C(z^{-1})R(z^{-1})w(t) - \frac{G(z^{-1})}{P_D(z^{-1})}\, y(t)\right\} \qquad (1.5.17)$$

By the use of recursive least squares estimation on the model (1.5.16), we require

1) $\hat{G}(z^{-1}) \rightarrow G(z^{-1})$

2) $\hat{E}(z^{-1}) \rightarrow C(z^{-1})Q(z^{-1}) + E(z^{-1})B(z^{-1})$

3) $\hat{H}(z^{-1}) \rightarrow C(z^{-1})R(z^{-1})$

where $\hat{G}(z^{-1})$, $\hat{E}(z^{-1})$ and $\hat{H}(z^{-1})$ are the estimated polynomials of order n_G, n_E , and n_H respectively.

As with the minimum-variance self-tuner, one parameter must be chosen prior to controller operation. In this case, however, as $C(z^{-1})$ is monic, by causing the $\hat{H}(z^{-1})$ polynomial to be monic as well, this problem is removed.

The polynomials $P(z^{-1})$, $Q(z^{-1})$ and $R(z^{-1})$ are chosen by the

operator, which means that several control characteristics may be specified, thus causing this method to be a far more generalised approach then the previous case, however, as with the simple minimum--variance case, the estimated polynomials can be overspecified. For the self-tuning property to hold it is necessary that 1) $n_H \geq n_1 + 0\{R(z^{-1})\}$, where $0\{\cdot\}$ denotes the order of the polynomial, 2) $n_E \geq \max[0\{C(z^{-1})Q(z^{-1})\}, 0\{E(z^{-1})B(z^{-1})\}]$, 3) $n_G \geq n_1 + \max[p_1,p_2]- 1$. Although if all three estimated polynomials are of order greater than the dimensions given, then the property does not hold because of factors which are common to all three polynomials.

The closed loop characteristic equation in this generalised tuner is given as $Q(z^{-1})A(z^{-1}) + P(z^{-1})B(z^{-1}) = 0$, and this means that if $Q(z^{-1})$ is of sufficient magnitude, the problem of instability due to non-minimum phase systems is removed on condition that the open-loop system is stable.

However, a problem apparent with both types of minimum-variance controllers is that the system time delay, k, must be known, due to the k-step-ahead prediction procedure.

This difficulty is overcome by the consideration of the following explicit self-tuning algorithms.

3. Pole Placement

Due to the problem of the failure to deal with a system time delay which could be variable and/or unknown, and the difficulties encountered with non-minimum phase behaviour, other approaches have been developed. Of these, the pole placement method, Wellstead et al (1979b), not only operates efficiently under both of these circumstances,

but also provides a smoother control action thus discounting, to a large extent, the possibility of saturation effects. In this approach the parameters of a system model are estimated, these estimations being used in the calculation of the required control action.

The system itself is considered to be governed by the CARMA model (1.5.1), and assuming initially that the constituent parameters are known, the controller is developed as follows.

Define the control input as,

$$u(t) = \frac{G(z^{-1})}{D(z^{-1})}\, y(t) \qquad (1.5.18)$$

where $G(z^{-1})$ is of degree n_g and $D(z^{-1})$ is monic and of degree n_d.

The closed loop pole polynomial may then be obtained by substituting this control input into the CARMA model, giving,

$$\{A(z^{-1})D(z^{-1}) - z^{-k}B(z^{-1})G(z^{-1})\}y(t) = C(z^{-1})D(z^{-1})e(t) \qquad \ldots \qquad (1.5.19)$$

It is now required to choose the parameters incorporated in this closed-loop pole polynomial, and for this purpose the desired polynomial is specified as taking the following form

$$T(z^{-1}) = 1 + t_1 z^{-1} + t_2 z^{-2} + \ldots + t_{n_t} z^{-n_t} \qquad (1.5.20)$$

where t_1, t_2, etc. are chosen prior to operation of the controller. Hence, subject to limitations placed on the respective orders of the polynomials, $D(z^{-1})$ and $G(z^{-1})$ from (1.5.18) can be calculated from the identity

$$A(z^{-1})D(z^{-1}) - z^{-k}B(z^{-1})G(z^{-1}) = C(z^{-1})T(z^{-1}) \qquad (1.5.21)$$

to give a closed loop form with denominator $T(z^{-1})$.

We now consider the self-tuning scheme, where the parameters

present in the polynomials $A(z^{-1}), B(z^{-1})$, $C(z^{-1})$, and thus $D(z^{-1})$, $G(z^{-1})$ also, are unknown along with the time delay, k. These parameters must, therefore, be estimated, a process which is carried out using recursive least squares on the model,

$$\hat{A}(z^{-1})y(t) = \hat{B}'(z^{-1})u(t) + \varepsilon(t) \tag{1.5.22}$$

where
$$\left.\begin{aligned} \hat{A}(z^{-1}) &= 1 + \hat{a}_1 z^{-1} + \hat{a}_2 z^{-2} + \ldots + \hat{a}_{n_a} z^{-n_a} \\ \text{and} \quad \hat{B}'(z^{-1}) &= \hat{b}_0 z^{-1} + \hat{b}_1 z^{-2} + \ldots + \hat{b}_{n_b+k_m-1} z^{-(n_b+k_m)} \end{aligned}\right\} \tag{1.5.23}$$

also $k_m = \max\{k\}$.

By once again employing the control input (1.5.18), the closed-loop equation becomes,

$$\{\hat{A}(z^{-1})D(z^{-1}) - \hat{B}(z^{-1})G(z^{-1})\}y(t) = D(z^{-1})\varepsilon(t) \tag{1.5.24}$$

However, in this case, the parameters of the control polynomials, $D(z^{-1})$ and $G(z^{-1})$, must be calculated from,

$$\hat{A}(z^{-1})D(z^{-1}) - \hat{B}'(z^{-1})G(z^{-1}) = T(z^{-1}) \tag{1.5.25}$$

where $T(z^{-1})$ is defined in (1.5.20).

For the self-tuning property to hold we require that the parameters of $D(z^{-1})$ and $G(z^{-1})$ obtained recursively from (1.5.25) will have a possible convergence point given by the values obtained by solving (1.5.21) offline, had the parameters been known.

Subject to the parameter estimates, in the model (1.5.22), converging it is shown in Wellstead et al (1979b) that as long as the following apply, the controller parameters will converge.

1) $n_d = n_b + k_m - 1$

2) $n_g = n_a - 1$

3) $n_t < n_a + n_b + k_m - n_c$

On the above conditions holding, the error sequence, $\varepsilon(t)$, will also converge to the disturbance, $e(t)$, described in the CARMA model.

Unless the case occurs that $C(z^{-1})$ is equal to unity, the parameter estimates obtained from (1.5.22) will be biased. This, however, accounts for the calculation of the controller polynomials from (1.5.25) being the same as those obtained from (1.5.21), despite the disappearance of the $C(z^{-1})$ polynomial.

The time delay, k_m, given in the estimation model (1.5.22), is now defined as the maximum possible value of system integer time delay, and thus the robustness of this type of tuner allows it to operate as long as the actual time delay is less than or equal to its specified maximum value, although the regulation characteristics are found to deteriorate when the actual time delay is greater than the minimum value allowed for.

Several other methods have been developed on the basis of pole or pole-zero assignment, Wellstead et al (1979a), Åström and Wittenmark (1980), although this invariably results in an increase in the necessary computational effort.

4. Extended Algorithm

The model is now considered as,

$$\hat{A}(z^{-1})y(t) = \hat{B}'(z^{-1})u(t) + \hat{C}(z^{-1})\varepsilon(t) \qquad (1.5.26)$$

where the constituent parameters are estimated by means of the recursive Extended Least Squares procedure, although a recursive Maximum-Likelihood estimator is also allowable. The controller polynomials to be used in (1.5.18) can then be evaluated, at each sampling interval, from the identity

$$\hat{A}(z^{-1})D(z^{-1}) - \hat{B}'(z^{-1})G(z^{-1}) = \hat{C}(z^{-1})T(z^{-1}) \qquad (1.5.27)$$

Due to the larger number of parameters estimated, the computational effort is much increased, although it can be considered that the $\hat{C}(z^{-1})$ polynomial is specified arbitrarily by the operator, thus reducing the estimation required to that of the recursive least squares procedure, Wellstead and Sanoff (1981). This tuner then contains the original pole placement self-tuner as a special case, when $\hat{C}(z^{-1})$ is set equal to unity.

5. Explicit Method with Known Time Delay

The last two approaches operate most efficiently when the exact value of the integer part of the system time delay is known, the method described here, however, can only operate when this delay is known explicitly.

The estimation model is defined as in (1.5.6), where $m = n_1 - 1 = \ell$, and hence the recursive least squares procedure can be used to obtain the required estimates. With a pole polynomial specified by (1.5.20) and the control action by (1.5.18), the parameters of the polynomials $D(z^{-1})$ and $G(z^{-1})$ are obtained from the equation,

$$D(z^{-1}) + z^{-k}A(z^{-1})D(z^{-1}) - z^{-k}\beta(z^{-1})G(z^{-1}) = P'(z^{-1})T(z^{-1}) \qquad \ldots \qquad (1.5.28)$$

where $P'(z^{-1})$ is monic and of degree k-1, and $T(z^{-1})$ is of degree $n_t \leq n_1 + k-1$; in this case $P'(z^{-1})$ is calculated, along with the control polynomials, from the above equation, Wellstead et al (1979a). If the parameters of the estimation model (1.5.6) converge, the closed-loop pole polynomial will be given by $T(z^{-1})$.

Apart from the necessity of knowledge about the system time delay, this tuner requires extra computational effort to calculate

(1.5.28) at each sampling interval, although it has been shown, Allidina and Hughes (1980), that if the operator chooses certain polynomials in the generalised cost function method, a pole placement scheme similar to that described here can be achieved.

1.6 Concluding Remarks

In this chapter various adaptive control techniques have been introduced, and from these self-tuning has been selected for particular discussion. Therefore different methods in this specific field, dependent on the desired control objective, have been considered.

In all self-tuning controllers an estimation scheme is required, and as the emphasis has been placed on finding a simple recursive controller, implementable on a microprocessor, this necessitates an estimator with low computational effort. Hence the identification methods used in self-tuning concentrate on merely producing estimates of the parameters in a system model. Of these methods, that of recursive Least Squares is perhaps the simplest, and hence this has led to its widespread usage. However, where a greater accuracy or a more rigorous mathematical formulation is desired, other approaches such as Extended Least Squares or Maximum-Likelihood estimation are employed, with a resultant increase in the number of calculations and thus the time required.

The first control systems to be given the name Self-Tuners, used as their objective the minimization of the system output signal variance. They proved to operate efficiently in numerous industrial applications, however their scope became limited due to the failure

to deal with non-minimum phase systems and/or variable or unknown system time delays. The minimum-variance philosophy was, therefore, generalised to account for a costing on the system input as well as the output, and this removed, to an extent, the first of the problems.

The most recent self-tuners have concentrated on the idea of pole-placement, which not only deals with non-minimum phase behaviour, but can also cope with an unknown time delay. As this gives the final method a larger possible field of application, it is this latter type of tuner which predominates throughout the following work.

Finally, a point concerning the convergence of the self-tuning systems described in this chapter must be stressed, and this is the choice of dimensions for the estimation model, especially when a minimum-variance tuner is concerned. However, this will be dealt with further in Sec. 4.4.

CHAPTER 2

STATE SPACE CONTROLLERS

In recent years the analysis of control systems has seen a constant shift towards a state space approach. This is often considered to be due to an underlying increase in the complexity of a large number of systems, which it is required to control. Much theory has therefore been developed in terms of optimization and regulation techniques by means of a state space framework.

The majority of self-tuning theory is, however, centred around the use of backward shift polynomials in CARMA models. It therefore remains for self-tuning to be viewed with regard to the state space, not only to reconsider and possibly improve existing algorithms, but also to widen the scope of feasible applications to those areas dominated by state space methods.

Thus the problem arises, firstly to find a suitable state space form and then to consider the possibility of improvements and extensions to the existing work. Sec.2.1, therefore, introduces the field of Linear Quadratic Gaussian control and emphasises the development of the basic self-tuning techniques. Possible state space formulations are then considered in Sec.2.2, although the most useful for self-tuning controllers is dealt with more thoroughly. Once a state space model has been chosen an important handicap comes to light in that it will, in most cases, be impossible to obtain the state vector directly from the process under control. The state must, therefore, be reconstructed by making use of the measured

input and output variables actually at our disposal, and, hence, a state estimation procedure is constructed in Sec. 2.3.

There are essentially two basic classifications for self-tuning algorithms, single stage design and pole placement design. These are formulated in the state space in Secs. 2.4 and 2.5 respectively. In both cases the state space controller is compared with the original polynomial method and the stability of the final closed-loop form is analysed. Although, by use of the state space, controllers are arrived at which carry out the same control operation as in the polynomial case, further algorithms, developed solely via the state space, are considered and then compared with the original design.

2.1 Linear Quadratic Gaussian Control

The solution of the optimal Linear Quadratic Gaussian (LQG) control problem has been considered in both the frequency and time domains, Åström (1970). The time domain can, however, be considered in one of two ways, either continuous time or discrete time, the latter approach being the most appropriate and in fact the simplest when digital control is decided upon.

Using a time series analysis, Åström (1970), a control law is formulated by means of feedback control from the system output, $y(t)$, to the control input, $u(t)$. If there exists a k-step delay between input and output, then the control minimizes the predicted quadratic loss, k steps ahead of the present time. This theory formed the basis of the minimum variance self-tuning regulator, Åström and Wittenmark (1973), discussed in Sec.1.5. However, the minimization of the loss predicted other than k steps ahead cannot be accomplished with the same control law construction, hence, the need for an

extension to the original work.

When a state-space representation is employed for linear systems, Kalman (1958), the separation theorem can be used, which means, in this case, that the observer required for estimation of the state vector, obtained from the system parameters, can be made optimal in the sense that the reconstruction error is minimized, this is considered in detail in Chapter 5. From this state estimate a linear feedback control law can be established to satisfy the specific control objective, and in order to obtain the correct feedback it is considered that the state estimate is in fact the true state and that the system is devoid of any disturbances.

Relationships between the two approaches mentioned have been shown to exist, in particular, when considering the unit delay case the methods lead to the same control action, Caines (1972), in which the general delay case is also discussed. In order that a reasonable comparison between the methods may be made, though, it is required that the final state space closed loop equations are transformed into the time series transfer function form, and therefore the filter used to obtain the state estimate must be considered as having achieved a steady-state condition, this is the well known Kalman filter.

2.2 State Space Formulations

There are numerous possible state space formulations which may be obtained from a system described by the CARMA model (1.3.14). But as one of the main objectives in self-tuning is to obtain a computationally simple control law, only those state space representations bearing an essentially trivial relationship to the CARMA model need

be considered. With these representations, of which there are two of direct relevance here, it is possible to arrive at a form in which the parameters included in the state space model are obtained directly, or with very little inconvenience, from the CARMA model. The difference between the two approaches being the way in which they deal with the system time delay, k.

The delay, k, may be included in the dimension of the state, in which case the representation is termed an implicit delay model. Conversely, the delay may be taken into account via the control input, u(t), in which case the term explicit delay model is used. Due to the need for incorporating a variable time delay later in this work, the implicit delay model will be used throughout the text, it has, in any case, been shown to be the more efficient method of the two, Lam (1980). A brief summary of the explicit delay model is, however, given here for comparison.

To avoid confusion, it is felt worth noting that the names assigned to the types of state space model considered, implicit and explicit, bear no relationship, other than in the actual words used, to the implicit and explicit self-tuners discussed in Sec. 1.5.

1. The explicit state space model

A state space representation which is equivalent to the CARMA model (1.3.14) is,

$$\left.\begin{aligned} &(a) \quad \underline{x}(t+1) = P\underline{x}(t) + Qu(t-k+1) + R\, e(t) \\ &(b) \quad y(t) \quad = H\underline{x}(t) + e(t) \end{aligned}\right\} \qquad (2.2.1)$$

where $\underline{x}(t)$ gives the vector of state variables, and the matrices, P,Q,R and H are defined as follows

$$P = \begin{bmatrix} 0 & \cdots & 0 & 0 \\ 1 & \ddots & 0 & -a_{n_1} \\ 0 & \ddots & \vdots & \vdots \\ \vdots & \ddots & 0 & \vdots \\ 0 & 0 & 1 & -a_1 \end{bmatrix}, \quad Q = \begin{bmatrix} b_{n_1} \\ \vdots \\ b_0 \end{bmatrix}$$

$$R = \begin{bmatrix} 0 \\ c_{n_1} - a_{n_1} \\ \vdots \\ c_1 - a_1 \end{bmatrix}, \quad H^T = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \qquad (2.2.2)$$

The formulation (2.2.1) is singular in the sense that the system noise is a linear combination of the measurement noise.

The dimension of the state space is given as n_1+1, where n_1 is the maximum degree of the polynomials $A(z^{-1})$, $B(z^{-1})$, $C(z^{-1})$. However, the term b_{n_1} will only exist where a fractional part of the system time delay is apparent, the integral part being accounted for by k. If no fractional time delay is present, therefore, $b_{n_1} = 0$ and the representation may be reduced in dimension by unity.

2. General state-space representation

This formulation, termed the implicit state-space representation, takes the integral delay, k, into the dimension of the state vector. The state-space equivalent of (1.3.14) is in this case,

$$\left.\begin{aligned} &(a) \quad \underline{x}(t+1) = P\underline{x}(t) + Qu(t) + R\,e(t) \\ &(b) \quad y(t) \quad\;\; = H\underline{x}(t) + e(t) \end{aligned}\right\} \tag{2.2.3}$$

where $\underline{x}(t)$ is, once again, the vector of state variables, and

$$\left.\begin{aligned} P &= \begin{pmatrix} 0 & \cdots & \cdots & \cdots & 0 \\ 1 & \ddots & & & \vdots \\ 0 & \ddots & \ddots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & -a_{n_1} \\ \vdots & \ddots & \ddots & 0 & \vdots \\ 0 & \cdots & 0 & 1 & -a_1 \end{pmatrix}, \quad Q = \begin{pmatrix} b_{n_1} \\ \vdots \\ b_0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \\ R &= \begin{pmatrix} 0 \\ \vdots \\ 0 \\ c_{n_1} - a_{n_1} \\ \vdots \\ c_1 - a_1 \end{pmatrix}, \quad H^T = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix} \end{aligned}\right\} \tag{2.2.4}$$

The above matrices being similar in composition to those in the implicit delay state space model of Åström (1974). Thus the dimension of P is $(n_1+k)\times(n_1+k)$, and Q has $(k-1)$ zeros in its lower rows, whereas R has k zeros in its upper rows. Again, if $b_{n_1} = 0$, the representation may be reduced in dimension by one, as no fractional time delay is present. H is simply a row vector of dimension $1\times(n_1+k)$, and Q and R are column vectors of dimension $(n_1+k)\times 1$.

It follows from (2.2.3) that

$$y(t) = H(I-z^{-1}P)^{-1}Qu(t-1) + \{1+z^{-1}H(I-z^{-1}P)^{-1}R\}e(t)$$

or

$$y(t) = H(I-z^{-1}P)^{-1}\{Qu(t-1) + R\,e(t-1)\}+e(t)$$

where $H(I - z^{-1}P)^{-1}$ is then given as,

$$\frac{1}{(1+a_1 z^{-1}+\ldots+a_{n_a} z^{-n_a})}\,[z^{-(n_1+k-1)}, \ldots, z^{-1}, 1]$$

$$= \frac{1}{A(z^{-1})}\,[z^{-(n_1+k-1)}, \ldots, z^{-1}, 1]$$

Furthermore, where $b_{n_1} \neq 0$, it follows that

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

and hence equation (1.3.14) holds.

A special case of this general state space representation may be considered when the time delay, k, is equal to unity, under this condition the latter model is identical to the explicit delay state space model. It is this assumption that is made in order to explain the meaning of the states selected, and this is shown by a simple example, as follows.

Consider the case where $n_1 = 2$, $k = 1$, $c_2 = c_1 = 0$ and $b_2 = 0$, then the state space representation is

$$\underline{x}(t+1) = \begin{bmatrix} 0 & -a_2 \\ 1 & -a_1 \end{bmatrix} \underline{x}(t) + \begin{bmatrix} b_1 \\ b_0 \end{bmatrix} u(t) + \begin{bmatrix} -a_2 \\ -a_1 \end{bmatrix} e(t)$$

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

where $\underline{x}(t+1) = \begin{bmatrix} x_1(t+1) \\ x_2(t+1) \end{bmatrix}$

Therefore we have

$$\text{i)} \quad x_1(t+1) = -a_2 x_2(t) + b_1 u(t) - a_2 e(t)$$

$$\text{ii)} \quad x_2(t+1) = x_1(t) - a_1 x_2(t) + b_0 u(t) - a_1 e(t)$$

$$\text{iii)} \quad y(t) = x_2(t) + e(t)$$

One solution to these equations, which will be shown later to be the optimal solution, see Sec. 5.1, is to denote the state variables as

$$\text{i)} \quad x_1(t+1) = b_1 u(t) - a_2 y(t)$$

$$\text{ii)} \quad x_2(t+1) = b_0 u(t) + b_1 u(t-1) - a_1 y(t) - a_2 y(t-1)$$

The method of obtaining these solutions will now be considered in a general framework.

2.3 State Estimation

In most situations the system state is not a directly measurable quantity, therefore this needs to be estimated from the information available. The estimation scheme derived by means of a Kalman filter is given simply here, but will be considered in depth in Chapter 5. It is beneficial, though, to consider the final requirements for the estimator when in use, and it can be seen that by adopting a recursive least squares parameter estimation scheme, Sec.1.4, this can be regarded, initially, as assuming the $C(z^{-1})$ polynomial to be equal to unity. This assumption was, in fact, made in the example of the preceding section.

Thus, taking $c_1 = c_2 = \ldots = c_{n_c} = 0$,

$$R^T = [0,\ldots,0, -a_n,\ldots,-a_1] \qquad (2.3.1)$$

Also, in the matrices P,Q and H (2.2.4), n_1 becomes n where,

$$n = \max\{n_a, n_b\}$$

An estimate, $\underline{\hat{x}}(t)$, of the state at time t is now required, and this can be obtained by considering the formulation (2.2.3).

Rewriting (2.2.3b) as

$$e(t) = y(t) - H\underline{\hat{x}}(t)$$

where the state is replaced by its estimate. Then by substitution of this into (2.2.3a),

$$\underline{\hat{x}}(t+1) = (P - RH)\underline{\hat{x}}(t) + Q\,u(t) + R\,y(t)$$

and setting $\bar{P} = P - RH$ we have

$$(I - z^{-1}\bar{P})\underline{\hat{x}}(t) = z^{-1}Q\,u(t) + z^{-1}R\,y(t).$$

The estimated state therefore becomes

$$\underline{\hat{x}}(t) = z^{-1}[I - z^{-1}\bar{P}]^{-1}\{Q\,u(t) + R\,y(t)\} \qquad (2.3.2)$$

The majority of the computational effort involved in obtaining this estimate of the state thus appears to be in finding the inverse of $(I - z^{-1}\bar{P})$.

But $\bar{P} = P - RH$, where

$$RH = \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & & \vdots \\ 0 & \cdots & 0 \\ \vdots & & -a_n \\ \vdots & & \vdots \\ 0 & \cdots & -a_1 \end{bmatrix}$$

as $c_1 = c_2 = \ldots = c_{n_c} = 0$.

Therefore,

$$P\text{-}RH = \bar{P} = \begin{bmatrix} 0 & \cdots & 0 \\ 1 & \ddots & \vdots \\ 0 & \ddots & \vdots \\ \vdots & \ddots & \vdots \\ 0 & \cdots\ 0\ \ 1 & 0 \end{bmatrix} \qquad (2.3.3)$$

Hence ,

$$(I-z^{-1}\bar{P}) = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ -z^{-1} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & -z^{-1} & 1 \end{bmatrix} \tag{2.3.4}$$

which gives

$$(I-z^{-1}\bar{P})^{-1} = P^{*}z = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ z^{-1} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ z^{-(n+k-1)} & \cdots & z^{-1} & 1 \end{bmatrix} \tag{2.3.5}$$

A more detailed approach, where $C(z^{-1})$ has non-zero parameters, leads to a much greater computational requirement, although it may be easily constructed, Lam (1979).

Hence,

$$\underline{\hat{x}}(t) = P^{*}\{Qu(t) + Ry(t)\} \tag{2.3.6}$$

and this is the steady-state value of the expected state vector.

So, we have

$$P^{*}Q = \begin{bmatrix} b_n z^{-1} \\ b_{n-1}z^{-1} + b_n z^{-2} \\ \vdots \\ z^{-k}(b_0 + b_1 z^{-1} + \ldots + b_n z^{-n}) \end{bmatrix} \tag{2.3.7}$$

and also,

$$P^{*}R = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ -a_n z^{-1} \\ \vdots \\ -a_1 z^{-1} - a_2 z^{-2} - \ldots - a_n z^{-n} \end{bmatrix} \qquad (2.3.8)$$

Thus, this formulation gives an estimate of the entire state vector at time t, and it can now be seen how the states used in the example of Sec. 2.2 were obtained.

By use of (2.3.7) and (2.3.8) the $(n+k)^{th}$ state at time t is given by,

$$\hat{x}_{n+k}(t) = z^{-k}B(z^{-1})u(t) - A(z^{-1})y(t) + y(t) \qquad (2.3.9)$$

Also, from (2.2.3),

$$H\underline{\hat{x}}(t) = y(t) - e(t)$$

and as $H = [0,\ldots,0,1]$

$$\hat{x}_{n+k}(t) = y(t) - e(t) \qquad (2.3.10)$$

Equating (2.3.9) and (2.3.10) the following is obtained,

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

Hence

$$A(z^{-1})y(t) = z^{-k}B(z^{-1})u(t) + e(t) \qquad (2.3.11)$$

which is the original CARMA model, where $C(z^{-1}) = 1$.

2.4 Single Stage Cost Function Design

1. Using the state-space model

The control input, u(t), is to be chosen at each time instant such that the cost function $\phi(u)$ is minimized, where

$$\phi(u) = \sum_{t=t}^{k-1+t} \{\underline{x}^T(t+1)S_3(t+1)\underline{x}(t+1) + u^T(t)S_2(t)u(t)\} + \underline{x}^T(t+k)S_1\underline{x}(t+k) \quad \ldots \quad (2.4.1)$$

and $\underline{x}(t)$ is the solution of the linear stochastic equations given by (2.2.3).

This is known as the discrete-time Linear-Optimal Regulator problem, Kwakernaak and Sivan (1972), in which S_1 is non-negative definite and $S_2(t)$ and $S_3(t+1)$ are both positive definite during the time interval under consideration. Further, the assumption is made that $u^T(t)S_2(t)u(t)$ may be written as $S_2u^2(t)$, because of the scalar nature of u(t). The optimal control input, derived by state feedback, is defined as being,

$$u(t) = F(t)\underline{x}(t) \quad (2.4.2)$$

where $\underline{x}(t)$ is the state vector, $F = [f_1,\ldots,f_{n+k}]$, and thus $\{f_i : i=1,\ldots,n+k\}$ must be chosen such that

$$F(t) = -(S_2+Q^TV(t+1)Q)^{-1}Q^TV(t+1)P \quad (2.4.3)$$

where the matrices $\{V(i): i=t,t+1,\ldots,t+k-1\}$ satisfy the matrix Riccati equation,

$$V(t) = P^TV(t+1)[P+QF(t)] + S_3 \quad (2.4.4)$$

and $$V(t+k) = S_1 \quad (2.4.5)$$

From (2.4.3) it can be seen that F(t) is calculated by use of V(t+1) and from (2.4.4), V(t) by use of F(t) and V(t+1). Therefore F(t) and V(t) are both obtained from the k^{th} iteration of the Riccati equation,

working in a backwards fashion with (2.4.3) being evaluated prior to (2.4.4) at each step.

The assumption is now made that,

$$S_1 = S_3 = \begin{pmatrix} 0 & \cdots & 0 \\ \vdots & & \vdots \\ \vdots & & \vdots \\ \vdots & & \vdots \\ \cdots & 0 & 0 \\ \vdots & & \\ 0 \cdots & 0 & 1 \end{pmatrix} \qquad (2.4.6)$$

and using (2.4.5), V(t+k) is also equal to the above.

As $Q^T V(t+k) P = 0$, unless $k = 1$

then $F(t+k-1) = 0$, under the same condition.

But $V(t+k-1) = \begin{pmatrix} 0 & \cdots & 0 \\ \vdots & & \vdots \\ \vdots & & \vdots \\ \vdots & & \vdots \\ \cdots & 0 \; 0 & 0 \\ \vdots & & \\ \cdots & 0 \; 1 & -a_1 \\ 0 \cdots & 0 \; -a_1 & 1-a_1^2 \end{pmatrix}$

By continual back substitution it follows that,

$$F(t) = (b_0 + S_2 b_0^{-1})^{-1} [0,\ldots,0,1,f_{n+1},\ldots,f_{n+k}] \qquad (2.4.7)$$

which contains n-1 leading zeros, and where

$$Q^T V(t+1) P = b_0^2$$

The feedback factors f_{n+1}, ... , f_{n+k} are found to be such that

$$f_{n+i} = -a_i - a_{i-1} f_{n+1} - \ldots - a_1 f_{n+i-1} \qquad (2.4.8)$$

which means that they are identical to the parameters contained in the solution of the identity (1.5.3), when $C(z^{-1})$ = unity.

Rewriting this identity for completeness,

$$1 = A(z^{-1})E(z^{-1}) + z^{-k}G(z^{-1}) \qquad (2.4.9)$$

$$\left.\begin{array}{ll} \text{where} & \text{i)} \quad e_i = f_{n+i} \; ; \; i = 1,\ldots,k-1 \\ & \text{ii)} \quad g_0 = f_{n+k} \end{array}\right\} \tag{2.4.10}$$

where $E(z^{-1})$ is of degree k-1 and $G(z^{-1})$ is of degree n-1.

But the control input (2.4.2) employed the feedback parameter vector, F(t), with a known, or measurable state. When the latter must be estimated, however, the control input must be redefined as,

$$u(t) = F(t)\hat{\underline{x}}(t) \tag{2.4.11}$$

where $\hat{\underline{x}}(t)$ is an estimate of the state vector obtained in Sec.2.3, and F(t) is obtained as set out above.

2. Comparison with the polynomial model

The generalised cost function is discussed in Sec.1.5, and from this the following is obtained,

$$\emptyset(u) = E\{y^2(t+k) + S_2'u(t)\} \tag{2.4.12}$$

where this is the expected value at time t, w(t) being zero, $P(z^{-1}) = 1$ and $Q(z^{-1}) = S_2'$.

The control input is found by applying (1.5.17) to be,

$$u(t) = \frac{1}{S_2'C(z^{-1}) + E(z^{-1})B(z^{-1})} \{-G(z^{-1})y(t)\} \tag{2.4.13}$$

The case considered thus far, makes the assumption that $C(z^{-1}) = 1$, and therefore a comparison of the polynomial and state space controllers is given here with that assumption retained.

Lemma 2.1

With $C(z^{-1})$ = unity, the polynomial and state space generalised minimum-variance controllers are similar. The only difference

occurring because of the $(n+k)^{th}$ state at time t. However, where a deterministic rather than stochastic case is concerned, subject to suitable choice of S_2 and S_2', the controllers are identical.

Proof: by induction.

Example 2:1; n = 2, k = 2.

The polynomial controller is then given by (2.4.13) to be

$$u(t) = \frac{-G(z^{-1})}{S_2' + E(z^{-1})B(z^{-1})}\, y(t)$$

where $E(z^{-1}) = 1 + e_1 z^{-1}$; $B(z^{-1}) = b_0 + b_1 z^{-1} + b_2 z^{-2}$,

$G(z^{-1}) = g_0 + g_1 z^{-1}$ and $C(z^{-1}) = 1$.

The state space controller is obtained from (2.3.6), (2.4.7) and (2.4.10,11) as

$$u(t) = \frac{-b_0}{S_2+b_0^2} [0 \quad 1 \quad e_1 \quad g_0] \left\{ \begin{bmatrix} b_2 z^{-1} \\ b_1 z^{-1} + b_2 z^{-2} \\ z^{-1}(b_0 + b_1 z^{-1} + b_2 z^{-2}) \\ z^{-2}(b_0 + b_1 z^{-1} + b_2 z^{-2}) \end{bmatrix} u(t) + \begin{bmatrix} 0 \\ 0 \\ -a_2 z^{-1} \\ -a_1 z^{-1} - a_2 z^{-2} \end{bmatrix} y(t) \right.$$

and thus,

$$\{(S_2/b_0+b_0) + (b_1+e_1 b_0)z^{-1} + (b_2+e_1 b_1+g_0 b_0)z^{-2} + (e_1 b_2+g_0 b_1)z^{-3} + g_0 b_2 z^{-4}\}u(t)$$

$$= \{-g_1 z^{-1} + g_0(a_1 z^{-1} + a_2 z^{-2})\}y(t)$$

where $g_1 = -a_2 e_1$.

The u(t) coefficient can now be rearranged to give

$$\frac{S_2}{b_0} + E(z^{-1})B(z^{-1}) + g_0 z^{-k}B(z^{-1}).$$

By replacing S_2/b_0 by S_2', and adding and subtracting $g_0 y(t)$ to the right hand side

$$\{S_2' + E(z^{-1})B(z^{-1})\}u(t) = -G(z^{-1})y(t) + g_0\{A(z^{-1})y(t) - z^{-k}B(z^{-1})u(t)\}$$

From the CARMA model, $A(z^{-1})y(t) - z^{-k}B(z^{-1})u(t) = e(t)$, due to the definition of $C(z^{-1})$ in this example. Therefore, in the deterministic case the state space controller is equivalent to the polynomial controller iff $S_2' = S_2/b_0$.

Also the $(n+k)^{th}$ state, which in this example is the 4^{th} state, is given by,

$$x_{n+k}(t) = z^{-k}B(z^{-1})u(t) - A(z^{-1})y(t) + y(t).$$

or $$x_{n+k}(t) = y(t) - e(t)$$

and this was shown in (2.3.10).

Hence if a modification is carried out on the estimation of the state vector, such that only the $(n+k)^{th}$ state is altered, a new state estimator will be found such that the state space controller is identical to the polynomial controller for both stochastic and deterministic problems.

Defining the new state as

$$\hat{\underline{x}}'(t) = \hat{\underline{x}}(t) + H^T(y(t) - H\hat{\underline{x}}(t)) \qquad (2.4.14)$$

or $$\hat{\underline{x}}'(t) = \hat{\underline{x}}(t) + H^T e(t)$$

where $H = [0,..,0,1]$

which merely removes $e(t)$ from the $(n+k)^{th}$ state.

The form given for the state (2.4.14) is very important in self-tuning, as it forms the basis of many self-tuning techniques.

It is discussed to a much greater extent in Chapter 5, but here a few points relating to it are mentioned. Firstly, the state estimation initially formulated was dependent on values of input and output signal up to and including those at time t-1, for the state estimated at time t. In rearranging the state, the contents of the $(n+k)^{th}$ state have, in effect, been replaced by the present value of the output signal. Hence, we can write

$$\left.\begin{aligned} \underline{\hat{x}}(t) &= \underline{\hat{x}}(t/t-1) \\ \text{and} \quad \underline{\hat{x}}'(t) &= \underline{\hat{x}}(t/t) \end{aligned}\right\} \qquad (2.4.15)$$

This means that no equality can exist, in a stochastic control system, between the polynomial single stage controller and the state space version using the state $\underline{\hat{x}}(t)$, because in the polynomial case, unless $g_0 = 0$, the present value of control input, u(t), is dependent on the present value of output signal, y(t). This can be seen from (2.4.13) and the definition of the $G(z^{-1})$ polynomial.

Also, from the feedback equation (2.4.13), the value S_2' leads to the generalised minimum-variance controller. If $S_2' = S_2 = 0$, then the original minimum-variance controller is arrived at, with its respective feedback equation given by (1.5.5). Hence the relationships obtained between the polynomial and state space controllers based on a single stage cost function design, also apply to the special case of the minimum-variance self-tuner.

The comparison of polynomial and state-space controllers has been carried out with the assumption, $C(z^{-1}) = 1$. The case for a general $C(z^{-1})$ is far more cumbersome to analyse, although for continuity the following Lemma is given.

Lemma 2.2

For a generalised minimum-variance tuner the state space controller is equivalent to the polynomial controller, subject to the following conditions.

i) S_2 and S_2' are suitably chosen, as in Lemma 2.1.

ii) The estimated state is modified such that if $\underline{x}'(t)$ is the state vector employed,

$$\hat{\underline{x}}'(t) = \hat{\underline{x}}(t) + \bar{H}^T\{y(t) - H\hat{\underline{x}}(t)\} \qquad (2.4.16)$$

where $P\bar{H}^T = R$. i.e. $\bar{H} = [0,\ldots,0,c_{n_c},\ldots,c_1,1]$

Proof: by induction, example given in Appendix 2.1.

It can be seen from (2.4.16), that in the case when $C(z^{-1})$ is unity, this reduces to (2.4.14).

The analysis carried out in this section considers a single stage k step ahead cost function, requiring k iterations of the Matrix Riccati equation (2.4.4). This, however, can be generalised to a 1+N stage cost function, where N is the extra number of stages. The number of iterations of the Riccati equation required to deal with the multistage fixed horizon case increases, though, to N+k. No advantage is achieved by using the extra stages in the minimum-variance controller, as the feedback gain, F(t), remains unchanged whatever the number of stages considered. But in the generalised minimum variance case a difference is apparent, and therefore it remains with the system designer to choose that which is most appropriate.

3. Stability of the single stage controller

The stability of the single stage control law can be considered as follows.

When $C(z^{-1}) = 1 + c_1 z^{-1} + \ldots + c_{n_c} z^{-n_c}$, the control input is found from the equation,

$$u(t) = \frac{-G(z^{-1})}{S_2' C(z^{-1}) + E(z^{-1})B(z^{-1})}\, y(t)$$

where $S_2' = S_2/b_0$.

Substituting for u(t) from this equation into the general CARMA model (1.3.14) leads to

$$\{S_2' A(z^{-1})C(z^{-1}) + A(z^{-1})E(z^{-1})B(z^{-1}) + z^{-k}B(z^{-1})G(z^{-1})\}y(t)$$

$$= C(z^{-1})\{S_2' C(z^{-1}) + E(z^{-1})B(z^{-1})\}e(t) \qquad (2.4.17)$$

Using the identity (1.5.3) to substitute for $z^{-k}G(z^{-1})$ in the above, the closed loop poles are obtained from the polynomial,

$$C(z^{-1})\{S_2' A(z^{-1}) + B(z^{-1})\}$$

which is equated to zero to obtain the characteristic roots.

When $S_2' = 0$, we return to the original minimum-variance controller, whose closed loop poles are therefore given by $C(z^{-1})B(z^{-1})$, and thus even if $C(z^{-1})$ has all its roots inside the unit circle, if $B(z^{-1})$ is unstable, i.e. a non-minimum phase system, there will be unstable poles in the closed loop equation. In the generalised case, however, this can possibly be avoided due to the bias on closed loop poles attributed to the choice of S_2'.

There are, though, at least k poles which cannot be varied. As the state space is of dimension n_1+k, where $C(z^{-1})$ is non-unity,

there will be n_1+k closed loop poles given by the feedback control alone. The value k does not appear in the poles given by the control, i.e. $S_2'A(z^{-1}) + B(z^{-1})$, thus there are k poles at the origin of the z-plane.

From the state space viewpoint, the $C(z^{-1})$ term in the characteristic equation is provided by the state estimation procedure, see Appendix 2.1. Therefore, the closed loop poles of the overall system are, because of the separation theorem, those due to the state estimation coupled with those due to the feedback control law.

4. Self-tuning single stage controller

The formulations for the single stage control design have been obtained as though the system parameters were known. Now, however, the fact that they must be estimated by means of one of the recursive schemes, discussed in Sec. 1.4, is considered.

The state estimation carried out in Sec. 2.3, assumed $C(z^{-1}) = 1$, and so the recursive least squares procedure is most appropriate as a continuation of this. If a parameter estimation technique were employed, whereby the parameters of the $C(z^{-1})$ polynomial, or their equivalent, are also estimated, e.g. extended least squares, the calculation of the estimate of the state vector becomes far more complicated, as can be seen from Appendix 2.1.

By using the least squares method, though, only the maximum possible value of the integer time delay k, needs to be selected, and the state space model chosen allows for this.

Specifying the recursive least squares model as,

$$\hat{A}(z^{-1})y(t) = \hat{B}'(z^{-1})u(t) + \varepsilon(t) \tag{2.4.18}$$

where

$$\left.\begin{aligned} \hat{A}(z^{-1}) &= 1 + \hat{a}_1 z^{-1} + \ldots + \hat{a}_{n_a} z^{-n_a} \\ \hat{B}'(z^{-1}) &= \hat{b}_0 z^{-1} + \hat{b}_1 z^{-2} + \ldots + \hat{b}_{n_b+k_m-1} z^{-(n_b+k_m)} \end{aligned}\right\} \tag{2.4.19}$$

and the maximum value of system integer time delay, $k_m \geq 1$. Although where the actual delay $k < k_m$, several estimated coefficients; $\hat{b}_{n_b+k},\ldots,\hat{b}_{n_b+k_m-1}$; are equal to zero. Hence when $k = k_m$, none of these zero estimates occur.

Nevertheless, in a true state space model obtained from the estimate the aforementioned terms must be included to account for any k up to and including k_m. Therefore, considering (2.2.4), matrix Q becomes

$$\hat{Q}^T = [\hat{b}_{n_b+k_m-1},\ldots,\hat{b}_0] \tag{2.4.20}$$

such that, when $k = k_m$, and this is known, $\dim(\hat{Q}) = (n_b+k_m) \times 1$, so we have

$$\hat{Q}^T = [\hat{b}_{n_b+k_m-1},\ldots,\hat{b}_{k-1},0,\ldots,0] \tag{2.4.21}$$

where the latter $(k-1)$ terms are zero.

The $\hat{Q}$ matrix now becomes similar to Q in (2.2.4), except that the parameters above are estimates.

In the single stage cost function analysis considered via the state space in the previous subsections, the system integer time delay has been assumed known, and hence by applying recursive least squares estimation with a known k, and using $\hat{Q}$ as given by (2.4.21), a self-tuning operation can be performed which is identical to that carried out in the polynomial case, subject to the estimated state used being $\underline{\hat{x}}(t/t)$ and not $\underline{\hat{x}}(t/t-1)$

The overall algorithm via the state space may be summarized as

1) Estimate system model parameters by recursive least squares estimation.

2) Form state space representation from the estimated parameters.

3) Calculate state estimation, using information available up to the present time instant.

4) Calculate state feedback from the matrix Riccati equation.

5) Obtain the control input signal.

6) Repeat.

At first glance it appears that Step 2 need not be calculated during the actual implementation cycle, as to provide the required control input this need not be known. But in the method discussed in this section, the Riccati equation in Step 4 uses certain of the matrices contained in the state space model, and hence Step 2 must be included.

2.5 Pole Placement Design

Most recent self-tuning techniques have concentrated more on pole, Wellstead et al (1979b), or pole-zero, Åström and Wittenmark (1980), placement approaches. This has brought self-tuning more in line with model reference adaptive control, yet its separate identity remains.

In this section a pole placement design will be formulated, Warwick (1981a), which provides a different control action to that obtained via a polynomial approach, Wellstead et al (1979b). However, by reformulating the state estimation, it is shown how the two methods can be made to give identical controllers.

1 State space construction

It is considered, firstly, that the system parameters are known, and $C(z^{-1}) = 1$. The state estimation is then formed, as in Sec.2.3. We now require a control law such that the closed loop poles are

assigned to previously specified positions.

Using the state feedback,

$$u(t) = F\underline{\hat{x}}(t) \tag{2.5.1}$$

where $\underline{\hat{x}}(t)$ is the estimate of the state vector and $F = [f_1,\ldots,f_{n+k}]$, the parameters $\{f_i: i = 1,\ldots,n+k\}$ must be assigned such that the closed loop poles are given by a preselected polynomial,

$$T(z^{-1}) = 1 + t_1 z^{-1} + \ldots + t_{n_t} z^{-n_t} \tag{2.5.2}$$

Substituting (2.5.1) into (2.2.3(a)) gives

$$\underline{\hat{x}}(t+1) = (P+QF)\underline{\hat{x}}(t) + Re(t) \tag{2.5.3}$$

Let $\bar{F} = P + QF$, then

$$\underline{\hat{x}}(t) = z^{-1}(I-z^{-1}\bar{F})^{-1} R\, e(t)$$

and substituting this into (2.2.3(b)) means that,

$$y(t) = \{z^{-1}H(I-z^{-1}\bar{F})^{-1}R+1\}e(t) \tag{2.5.4}$$

Hence the closed loop poles are given in the state space by the determinant of $(I-z^{-1}\bar{F})$.

Thus $\{f_i: i = 1,\ldots,n+k\}$ must be such that

$$\det(I-z^{-1}\bar{F}) = T(z^{-1})$$

where $T(z^{-1})$ is defined in (2.5.2).

Now

$$(I-z^{-1}\bar{F}) = \begin{cases} \begin{bmatrix} 1-z^{-1}b_nf_1 & -z^{-1}b_nf_2 & \cdots & -z^{-1}b_nf_{n+k} \\ -z^{-1}(1+b_{n-1}f_1) & 1-z^{-1}b_{n-1}f_2 & & \vdots \\ -z^{-1}b_{n-2}f_1 & -z^{-1}(1+b_{n-2}f_2) & & \vdots \\ \vdots & -z^{-1}b_{n-3}f_2 & & \vdots \\ \vdots & \vdots & -z^{-1}(b_{n-k}f_{n+k}-a_n) & \\ \vdots & \vdots & -z^{-1}(b_jf_{n+k}-a_i) & \\ -z^{-1}b_0f_1 & -z^{-1}b_0f_2 & & \vdots \end{bmatrix} \\ \begin{bmatrix} 0 & 0 & & \vdots \\ \vdots & \vdots & & a_2z^{-1} \\ 0 & 0 \cdots 0 & & 1+a_1z^{-1} \end{bmatrix} \end{cases}$$

$(k-1)$ zeros

. . . . (2.5.5)

such that $1 \leq i \leq n$, $0 \leq j \leq n$ and $i-j = k$

By taking the determinant of the above matrix, we can write

$$T = WF^T - R' \qquad (2.5.6)$$

or $$F^T = W^{-1}(T+R')$$

where the following definitions are made. F is given in (2.5.1) and R' in (2.2.4) in which n_1 is replaced by n as $c_1 = c_2 = \ldots = c_n = 0$.

Thus $R' = \lim_{c_i \to 0} \{R\}$, for all i.

The matrix T in (2.5.6) is specified as,

$$T^T = [t_{n+k}, \ldots, t_1]$$

which means that up to (n+k) poles may be specified in this model, although if $n_t < n + k$,

$$t_{n+k} = \ldots = t_{n_t+2} = t_{n_t+1} = 0$$

This limitation on the degree of the closed loop pole polynomial is discussed further in Chapter 3, where a further restriction is placed on its maximimum value for the purpose of self-tuning. The integer n_t denotes the number of poles specified, although if a number of these poles are chosen to be at the origin of the z-plane, then that number of trailing coefficients of the $T(z^{-1})$ polynomial will be zero.

The W matrix relating the closed loop polynomial to the feedback terms is easily constructed as follows.

Referring back to (2.5.6)

$$\text{Let } W = W_1 + W_2 + W_3 \tag{2.5.7}$$

where $\{W_i : i = 1,2,3\}$ are matrices of dimension (n+k) × (n+k), when a fractional part is present in the system time delay, such that

$$W_1 = \begin{matrix} (n+k+\delta_f-2) \\ \text{zeros} \end{matrix} \left\{ \begin{matrix} 0 & \cdots & 0 & -b_n \\ \vdots & & & \vdots \\ & & & -b_0 \\ & & & 0 \\ 0 & & & \vdots \\ -b_n & \cdots & -b_0 & 0 \cdots 0 \end{matrix} \right\} \begin{matrix} \\ \\ \\ \\ \\ (k-1)\text{zeros} \\ \\ \\ \\ \\ (k-1)\text{zeros} \end{matrix} \tag{2.5.8}$$

In equation (2.5.8), k is the integer time delay and δ_f the Kronecker delta used to signify the presence of a fractional time delay, i.e.

$\delta_f = 1$ if a fractional delay is present, whereas

$\delta_f = 0$ if this is not the case.

The overall dimension is thus reduced to $(n+k-1)\times(n+k-1)$ where there is no fractional delay.

$$W_2 = \begin{array}{l} \left.\begin{array}{c} \\ \\ \end{array}\right\} (\delta_f+k-1) \text{ zeros} \\ \\ \\ \left.\begin{array}{c} \\ \\ \end{array}\right\} k \text{ zeros} \end{array} \overbrace{\left[\begin{array}{ccc} 0 & 0 & 0 \\ 0 & \bar{W}_2 & 0 \\ 0 & 0 & 0 \end{array}\right]}^{\substack{(\delta_f+k-1) \text{ zeros} \\ \qquad\qquad k \text{ zeros}}} \tag{2.5.9}$$

and $\bar{W}_2$ is defined as being

$$\bar{W}_2 = \begin{bmatrix} a_n b_{n-(k+1)} & \cdots & a_n b_1 & a_n b_0 \\ \vdots & & & \vdots \\ & & & \\ & & * & \\ & & a_i b_{i-(k+1)} \cdots & a_i b_0 \\ a_n b_1 & & & \vdots \\ & & & a_{k+2} b_0 \\ a_n b_0 & \cdots & a_i b_0 \cdots a_{k+2} b_0 & a_{k+1} b_0 \end{bmatrix} \tag{2.5.10}$$

All terms on the perimeter of $\bar{W}_2$ are repeated diagonally (bottom left to top right). But these terms must be added to an $a_i b_j$ term (*) at each elemental position, where $k+1 \leq i \leq n$ and $\ell \leq j \leq i-(k+1)$, ℓ being the coefficient of the b term in the diagonal repetition

However, where $j \neq \ell+1$, further terms must also be added to that position as follows,

$$\text{diagonal term} = a_{i+j} b_{\ell}$$

$$\text{secondary term} = a_i b_j$$

$$\text{terms to be added} = a_{i+1} b_j + \ldots + a_{i+j-1} b_{\ell+1}$$

Hence each element in the matrix $\bar{W}_2$, and thus W_2, is readily obtained.

To show a case of W_2 by example, consider $n = 5$, $k=1$, $\delta_f=0$. Then

$$W_2 = \begin{Bmatrix} a_5b_3 & a_5b_2 & a_5b_1 & a_5b_0 & 0 \\ a_5b_2 & a_4b_2+a_5b_1 & a_4b_1+a_5b_0 & a_4b_0 & 0 \\ a_5b_1 & a_4b_1+a_5b_0 & a_3b_1+a_4b_0 & a_3b_0 & 0 \\ a_5b_0 & a_4b_0 & a_3b_0 & a_2b_0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{Bmatrix} \qquad (2.5.11)$$

The final matrix required in the formulation of W is W_3, and this is built up of horizontal, vertical and diagonal elements, such that when these cross they must be added together. This can be seen from the definition of W_3, given by,

$$
W_3 = \left[\begin{array}{cccccccc}
0 & \cdots & 0, & -a_n b_{\tilde{n}}, & \cdots & , -a_1 b_{\tilde{n}} & & , 0 \\
\vdots & \ddots & & & \ddots & & -a_1 b_{\tilde{n}-1} & \vdots \\
0 & & & & & \ddots & \ddots & \vdots \\
-a_n b_{\tilde{n}} & , -a_n b_{\tilde{n}-1} & & & & & & \vdots \\
\vdots & & \ddots & & \ddots & & & \vdots \\
\vdots & & & & & & -a_1 b_{(1-\delta_{qk})} & 0 \\
-a_2 b_{\tilde{n}} & \cdots X & & & & & \ddots & \vdots \\
\vdots & \ddots & & & & \ddots & & \vdots \\
-a_1 b_{\tilde{n}}, & a_1 b_{\tilde{n}-1} & & & -a_1 b_{(1-\delta_{qk})} & & & \vdots \\
0 & \cdots & \cdots & \cdots & \cdots & 0 & \cdots & 0
\end{array}\right] \tag{2.5.12}
$$

(The first k' rows, marked "k' zeros", begin with zeros; the last q rows of the final column, marked "q zeros", and the last q columns of the final row, marked "q zeros", are zeros.)

where $\quad k' = q + \delta_f - 2 \qquad (2.5.13)$

and $\quad q = \max\{k, 2\} \qquad (2.5.14)$

also
$$
\delta_{qk} = \begin{cases} 0 & \text{when } q \neq k \\ 1 & \text{when } q = k \end{cases} \tag{2.5.15}
$$

and $\quad \tilde{n} = n + \delta_f - 1 \qquad (2.5.16)$

The highest coefficient of an 'a' term is n, as stated in (2,5.12), except for the case when $k + \delta_f < 2$, i.e. when $k = 1$, $\delta_f = 0$. Under these conditions the highest coefficient of an 'a' term is (n-1).

Again, all terms on the perimeter of W_3 are repeated diagonally (bottom left to top right) in a similar fashion to those of $\bar{W}_2$, such that W_3 is equal to its transpose.

To show a case of W_3 by example, consider $n = 2$, $k = 2$, $\delta_f = 1$. Then,

$$W_3 = \begin{Bmatrix} 0 & -a_2b_2 & -a_1b_2 & 0 \\ -a_2b_2 & -a_1b_2-a_2b_1 & -a_1b_1 & 0 \\ -a_1b_2 & -a_1b_1 & -a_1b_0 & 0 \\ 0 & 0 & 0 & 0 \end{Bmatrix} \qquad (2.5.17)$$

The overall W matrix may now be obtained by means of the expression (2.5.7), an example of which is set out below. However, to calculate the state feedback vector F, W must be inverted (2.5.6), and this presents the major computational requirement in the state space method.

For the example of obtaining W, the previous case of $n = 2$, $k = 2$, $\delta_f = 1$, will be used, such that with W_3 given by (2.5.17),

$$W_1 = \begin{Bmatrix} 0 & 0 & 0 & -b_2 \\ 0 & 0 & -b_2 & -b_1 \\ 0 & -b_2 & -b_1 & -b_0 \\ -b_2 & -b_1 & -b_0 & 0 \end{Bmatrix}$$

and W_2 = null matrix (4×4)

Hence,

$$W = \begin{bmatrix} 0 & -a_2b_2 & -a_1b_2 & -b_2 \\ -a_2b_2 & -a_1b_2-a_2b_1 & -b_2-a_1b_1 & -b_1 \\ -a_1b_2 & -b_2-a_1b_1 & -b_1-a_1b_0 & -b_0 \\ -b_2 & -b_1 & -b_0 & 0 \end{bmatrix}$$

2. Comparison with the polynomial case

The pole placement method, carried out using the polynomial form for controller design, was considered in Sec.1.5. We wish to compare this with the state space controller, which has now been developed.

The polynomial control law defines the control input as,

$$D(z^{-1})u(t) = G(z^{-1})y(t) \qquad (2.5.18)$$

where $D(z^{-1})$ and $G(z^{-1})$ are polynomials calculated from the identity,

$$A(z^{-1})D(z^{-1})-z^{-k}B(z^{-1})G(z^{-1}) = C(z^{-1})T(z^{-1}) \qquad (2.5.19)$$

The state space method, however, defines the control input, as in (2.5.1), where the state feedback parameters are obtained from,

$$\det(I-z^{-1}\bar{F}) = T(z^{-1}) \qquad (2.5.20)$$

and $\bar{F} = P + QF.$

The state estimate (2.3.6), though, was achieved with $C(z^{-1})$ assumed to be unity. Thus for an initial comparison of the controllers to be made, this assumption must be carried over to the identity (2.5.19). The following Lemma is then obtained.

Lemma 2.3

With $C(z^{-1}) = 1$, the polynomial and state space pole placement controllers are similar. The only difference occurs due to the $(n+k)^{th}$ state at time t. Where a deterministic system model is considered, though, the controllers are identical.

Proof: by induction, example given in Appendix 2.2.

The difference between the controllers therefore arises from the definition of the $(n+k)^{th}$ state, given in (2.3.10) as

$$x_{n+k}(t) = y(t) - e(t)$$

The system disturbance, $e(t)$, thus becomes an 'extra' term, as the $(n+k)^{th}$ state required for equality of the controllers is,

$$x_{n+k}(t) = y(t)$$

The state estimate can, therefore, be remodelled to obtain the estimation procedure necessary for an equivalent form to the polynomial pole placement scheme. Denoting this new state estimate as $\underline{\hat{x}}'(t)$, it may be found from the original estimate, $\underline{\hat{x}}(t)$, by

$$\underline{\hat{x}}'(t) = \underline{\hat{x}}(t) + H^T\{y(t) - H\underline{\hat{x}}(t)\}.$$

where $H = [0,\ldots,0,1]$.

This new state estimate, $\underline{\hat{x}}'(t)$ obtained for the pole placement controller, is identical to that formulated previously in the generalised minimum variance case (2.4.14), and thus it has been shown that in the case $C(z^{-1}) = 1$, the polynomial pole placement and generalised minimum variance controllers make use, effectively, of the same state estimator. The only difference between these controllers, therefore, arises from the calculation of the feedback control terms required to achieve the desired control objective.

The question now may be asked as to how the state space and polynomial pole placement controllers are related when a general $C(z^{-1})$ disturbance polynomial is considered, and whether or not this relationship is the same as that which was found in the generalised minimum-variance case. For this reason the following Lemma is introduced.

Lemma 2.4

The polynomial and state space pole placement controllers are equivalent iff the estimated state vector, used in the state space method, is considered to be,

$$\underline{\hat{x}}'(t) = \underline{\hat{x}}(t) + \bar{H}^T\{y(t) - H\underline{\hat{x}}(t)\}$$

where $\bar{H} = [0,\ldots,0,c_{n_c},\ldots,c_1,1]$.

Proof: by induction, example in Appendix 2.3.

As a result of Lemma 2.4 we have that, if the polynomial pole placement controller is to be considered from a state space viewpoint, then the estimated state employed must be considered as being $\underline{\hat{x}}'(t)$ rather than $\underline{\hat{x}}(t)$. The use of the estimate $\underline{\hat{x}}'(t)$ by both the pole placement and generalised minimum variance controllers gives an important underlying similarity between them, not evident when the state space is neglected. It is, therefore, possible to carry out the operations of either of the above controllers by means of a state space description, although this, naturally, only results in a control action identical to that obtained with the polynomial approach.

It has been shown, however, that from the state space description a controller can be formed which uses as its basis an estimate of the state vector given as $\underline{\hat{x}}(t)$. The required design procedure, e.g. pole placement, can, therefore, be calculated, and an essentially different

control action will result, if this type of state estimate is used.

3. Stability of the controller

The stability of the pole placement controller is viewed with the inclusion of a general $C(z^{-1})$ polynomial. In the pole-placement, as opposed to minimum variance, case it is the closed loop poles which are being operated upon directly, and hence an analysis becomes simpler. The closed loop pole polynomial is chosen such that the identity (2.5.19) is satisfied. This is repeated here as,

$$A(z^{-1})D(z^{-1}) - z^{-k}B(z^{-1})G(z^{-1}) = C(z^{-1})T(z^{-1}) \qquad (2.5.19)$$

where $D(z^{-1})$ and $G(z^{-1})$ are polynomials required for the control input, whose parameters are chosen such that the above equality holds. The closed loop poles are therefore selected such that the characteristic equation is described by,

$$C(z^{-1})T(z^{-1}) \qquad (2.5.21)$$

where $T(z^{-1})$ is the polynomial selected by the designer prior to controller operation. If the equality (2.5.19) holds therefore, this particular controller does not suffer from the effects on stability caused by a nonminimum phase process, as was the case with minimum variance type control laws. For stability of the overall system, though, the disturbance polynomial, $C(z^{-1})$, must still contain roots which lie, without exception, within the unit circle of the z-plane. If this is so, and the roots of $T(z^{-1})$ are chosen to lie within the same bounds, then the closed loop system will be stable.

The only problems associated with stability occur when the parameters of the system model are estimated. The required control law is then obtained from these estimations, as will be discussed

in the next subsection, and hence if the parameter estimates are not identical to the parameters themselves, the closed loop roots will not be given exactly by (2.5.21). This point is, in fact, of little consequence when (a) the estimates are reasonably close to their true values, and/or (b) when the roots of $C(z^{-1})$ and $T(z^{-1})$ lie well into the unit disc. On the contrary, though, the problem is emphasised to a greater extent when a model is chosen which allows for a variable integer time delay, k, as the estimates of the $\hat{B}'(z^{-1})$ polynomial, found in Sec. 1.5, which incorporates the time delay, will be purposefully biased away from the actual parameters in the system's $B(z^{-1})$ polynomial.

4. Self-tuning pole placement controllers

It was considered at the commencement of this section that the system parameters were known, and the pole placement control laws which followed were obtained with that in mind. Now, however, the practical case is looked at, whereby a system model is formulated and the parameters contained in this system model are estimated.

The simplest self-tuning algorithm is obtained by using the recursive least squares estimation procedure, where a model is given for the estimation process as,

$$\hat{A}(z^{-1})y(t) = \hat{B}'(z^{-1})u(t) + \varepsilon(t) \qquad (2.4.18)$$

The nature of the estimation procedure, and its effect on the state space description was discussed in Sec. 2.4(4), although a further comment must be made here concerning the $\hat{B}'(z^{-1})$ polynomial. This is redefined here as being,

$$\hat{B}'(z^{-1}) = \hat{b}_0 z^{-1} + \hat{b}_1 z^{-2} + \ldots + \hat{b}_{\bar{n}-1} z^{-\bar{n}} \qquad (2.5.22)$$

and $$\bar{n} = n_b + k_m \qquad (2.5.23)$$

It must, nevertheless, be remembered that $\hat{b}_i$ in no way means that this is an estimate of the actual system parameter, b_i, it is merely an estimate of the parameter in the system model used for recursive least squares estimation.

From the state space description (2.2.3) the matrix Q now becomes

$$\hat{Q}^T = [\hat{b}_{\bar{n}-1}, \ldots, \hat{b}_o] \qquad (2.5.24)$$

The maximum possible value of the system integer time delay, k, is given by k_m, and the self-tuner will operate as long as $k \leq k_m$. The preceding view of pole placement via the state space, however, has regarded k as being known, whereas now the delay is included as a further unknown. It may be the case, though, that despite the fact that the delay is unknown, it will remain within strict bounds, such that a minimum value is also apparent. Let k_{min} be the minimum possible value of k, then

$$1 \leq k_{min} \leq k \leq k_{max} = k_m. \qquad (2.5.25)$$

If we consider that the value of k is taken account of in the calculation of the W matrix, Sec. 2.5(1), then the matrix is directly applicable to the estimated parameter model, subject to the following :

1) $b_0 \rightarrow \hat{b}_{k_{min}-1}$

2) $b_n \rightarrow \hat{b}_{k_{max}-1+n}$

3) The overall state space dimension = $(n+k_{max})\times(n+k_{max})$

Hence $f_{n+k} \rightarrow \hat{f}_{n+k_{max}}$

and in general,

4) wherever k appears with n (i.e. n+k):$k \rightarrow k_{max}$

5) wherever k appears without n (i.e. k-1):$k \rightarrow k_{min}$.

Obviously, if a minimum value of time delay can be specified, better tuner performance will result, although no generality is lost by setting $k_{min} = 1$.

The overall scheme for the pole placement state space self-tuner is now given in steps, as a breakdown of its algorithmic construction.

1) Estimate model parameters using the recursive least squares estimation procedure.

2) Obtain an 'estimate' of the state estimate, $\hat{\underline{x}}(t)$.

3) Calculate the state feedback vector by means of the W matrix.

4) Evaluate the new control input.

5) Repeat.

Several points worth mentioning arise from the overall algorithm. Firstly, in step 2 the estimate of the state vector is now calculated by using the estimated model parameters, and hence this is called an estimate of the state estimate and will be denoted later by $\hat{\hat{\underline{x}}}(t)$. This state vector can be formed in step 2 without consideration of the actual state space model because the parameters included in the vector are obtained directly from the estimated ARMA model. This also applies to the calculation of the state feedback vector, and hence in the pole placement method the state space model itself need not be

constructed when an on line procedure is in operation. This is the converse of the generalised minimum variance case, where certain matrices included in the state space model were required due to the calculation of the matrix Riccati equation.

Secondly, it can occur that, due to the estimation process, the first column and row of the W matrix may contain almost zero values in every position, and hence problems would arise in the inversion of the matrix if this situation was allowable. These almost zero values occur because the first column and row can contain the estimated term $\hat{b}_{n-1}$ as a common factor, and thus if this tends to zero the dimension of both the W matrix and the state estimate must be reduced in value by unity. The first state being, in this case, dependent only on the term b_{n-1}, and hence will be almost zero if b_{n-1} is almost zero. This poses no problem, however, as in the Gaussian elimination method, used for the inversion of the W matrix, the maximum value in each column must be obtained. If, in the first column, the maximum value does not meet a previously specified lower limit, then this column is merely ignored with regard to the inversion process, and the elimination continues with the next column.

Finally, the algorithm given above can be used to carry out the equivalent of the polynomial pole placement self-tuner merely by altering the state estimate obtained in Step 2, by use of (2.4.14) given as,

$$\underline{\hat{x}}'(t) = \underline{\hat{x}}(t) + H^T(y(t) - H\underline{\hat{x}}(t)).$$

where $H = [0,\ldots,0,1]$.

The state estimate formed by using $\bar{H}^T$ instead of H^T is not

required here, as the parameters $c_1,\ldots,c_{n_c}$ included in $\bar{H}^T$ are considered to be zero in the model used for recursive least squares estimation.

5. Simulation study

The simulation considered employs a nonminimum phase discrete time system with white noise being of zero mean and variance = 0.11. Recursive Least Squares estimation was carried out using the Bierman UD filter, Bierman (1977), and the variable forgetting factor, due to Fortescue et al (1979), with a window of 500 samples.

The closed loop pole polynomial, $T(z^{-1})$, is chosen to be unity. Thus,

$$t_1 = t_2 = \ldots = t_{n_t} = 0$$

This choice of pole polynomial allows theoretical signal variances to be calculated with relative ease , such that they may be compared with the actual values obtained. Both the original polynomial, Wellstead et al (1979b), and the state space, Sec.2.5(1), tuning algorithms were applied, with the state estimation procedure being that discussed in Sec.2.3. The results obtained from the two types of self-tuner are considered in detail, where the system is given as

$$(1 - 1.2z^{-1} + 0.6z^{-2} + 0.2z^{-3})y(t)$$
$$= z^{-k}(1+1.4z^{-1})u(t) + (1-0.6z^{-1} +0.1z^{-2})e(t) \qquad (2.5.26)$$

e(t) being the white noise sequence.

The simulation was carried out over a total of 5000 time intervals, and until the 2500^{th} of these the time delay k remained at unity, for the remaining intervals it was increased to two.

The state space model formed, which contains the estimates obtained from the recursive least squares procedure is thus,

$$\left.\begin{aligned}\hat{\underline{\tilde{x}}}(t+1) &= \begin{bmatrix} 0 & 0 & -\hat{a}_3 \\ 1 & 0 & -\hat{a}_2 \\ 0 & 1 & -\hat{a}_1 \end{bmatrix}\hat{\underline{\tilde{x}}}(t) + \begin{bmatrix} \hat{b}_2 \\ \hat{b}_1 \\ \hat{b}_0 \end{bmatrix} u(t) + \begin{bmatrix} -\hat{a}_3 \\ -\hat{a}_2 \\ -\hat{a}_1 \end{bmatrix} e(t) \\ y(t) &= [\,0 \quad 0 \quad 1\,]\hat{\underline{\tilde{x}}}(t) + e(t)\end{aligned}\right\} \qquad (2.5.27)$$

where $\hat{a}_i$ and $\hat{b}_i$ are the model estimates

From the system description (2.5.26), $n_a = 3$, $n_b = 1$ and $k_m = 2$. Therefore $n = \max\{n_a, n_b\} = 3$, from which it follows that the dimension of the state vector is $(n+k_m)\times 1 = 5\times 1$. However as $b_n = b_3 = 0$, the dimension is reduced to 4×1 due to the first state being zero. This reduction, though, causes the second state to become equal to zero, as $b_{n-1} = b_2 = 0$ also. Hence , finally the dimension of the state vector is reduced to 3×1 in the model.

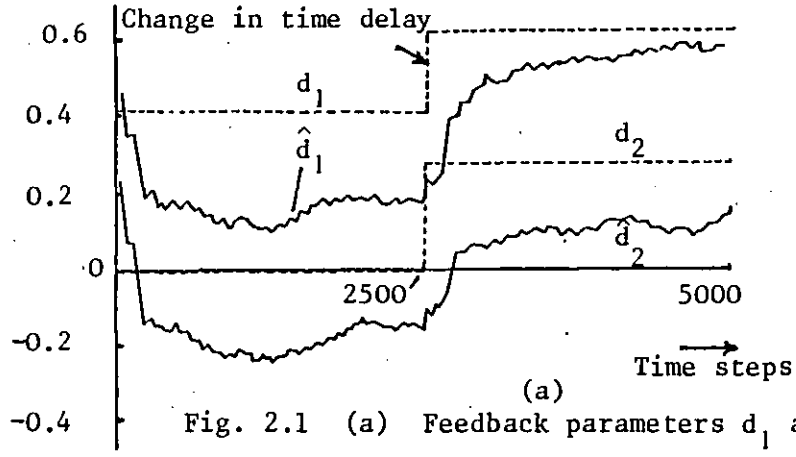
The feedback matrix, W , is then given as ,

$$W = \begin{bmatrix} -\hat{a}_2\hat{b}_2+\hat{a}_3\hat{b}_1 & -\hat{a}_1\hat{b}_2+\hat{a}_3\hat{b}_0 & -\hat{b}_2 \\ -\hat{a}_1\hat{b}_2+\hat{a}_3\hat{b}_0 & -\hat{b}_2-\hat{a}_1\hat{b}_1+\hat{a}_2\hat{b}_0 & -\hat{b}_1 \\ -\hat{b}_2 & -\hat{b}_1 & -\hat{b}_0 \end{bmatrix} \qquad (2.5.28)$$

giving rise to tne feedback terms $\hat{f}_1$, $\hat{f}_2$ and $\hat{f}_3$ acting upon the state vector obtained from,

$$\hat{\underline{\tilde{x}}}(t) = \begin{bmatrix} \hat{b}_2 z^{-1} \\ \hat{b}_1 z^{-1}+\hat{b}_2 z^{-2} \\ \hat{b}_0 z^{-1}+\hat{b}_1 z^{-2}+\hat{b}_2 z^{-3} \end{bmatrix} u(t) + \begin{bmatrix} -\hat{a}_3 z^{-1} \\ -\hat{a}_2 z^{-1}-\hat{a}_3 z^{-2} \\ -\hat{a}_1 z^{-1}-\hat{a}_2 z^{-2}-\hat{a}_3 z^{-3} \end{bmatrix} y(t)$$

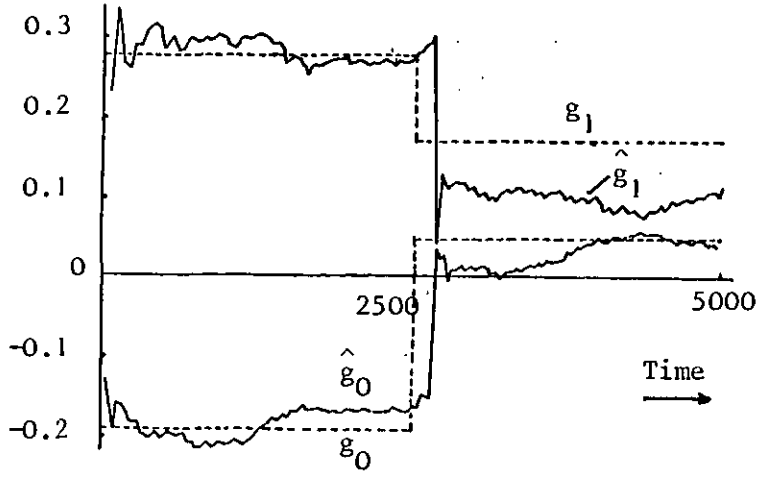
. . . (2.5.29)



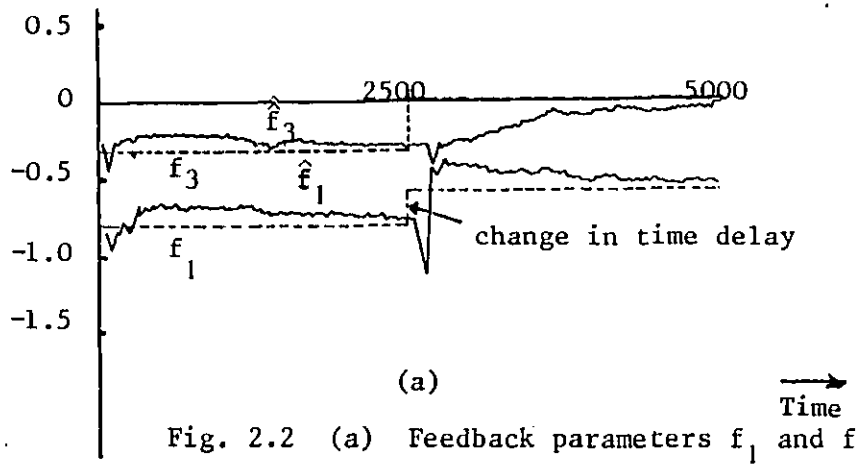
(a)

Fig. 2.1 (a) Feedback parameters d_1 and d_2 under polynomial controller.

(b) Feedback parameters g_0 and g_1 under polynomial controller (g_2 not shown)



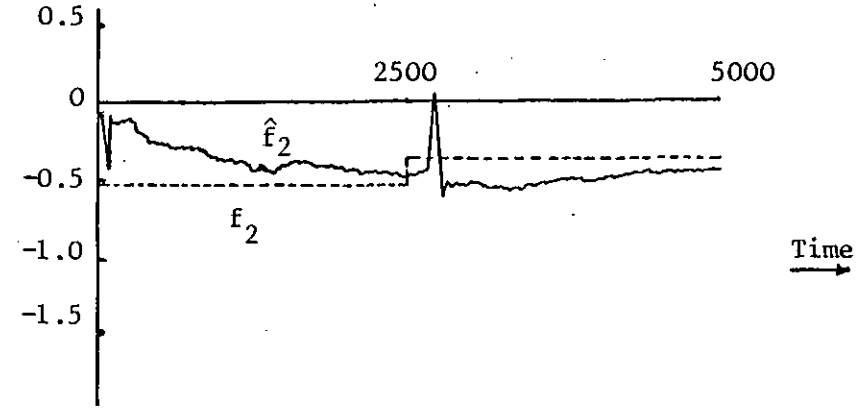
(b)



(a)

Fig. 2.2 (a) Feedback parameters f_1 and f_3 under state space controller

(b) Feedback parameter f_2 under state space controller.



(b)

In the polynomial controller, the control law is given by,

$$(1 + d_1 z^{-1} + d_2 z^{-2})u(t) = (g_0 + g_1 z^{-1} + g_2 z^{-2})y(t) \qquad (2.5.30)$$

where these parameters are obtained as the solution to the equation,

$$\hat{A}(z^{-1})D(z^{-1}) - \hat{B}'(z^{-1})G(z^{-1}) = T(z^{-1}) \qquad (2.5.31)$$

$$\left.\begin{aligned} &\text{in which, } \hat{A}(z^{-1}) = 1 + \hat{a}_1 z^{-1} + \hat{a}_2 z^{-2} + \hat{a}_3 z^{-3} \\ &\text{and} \qquad \hat{B}'(z^{-1}) = \hat{b}_0 z^{-1} + \hat{b}_1 z^{-2} + \hat{b}_2 z^{-3} \end{aligned}\right\} \qquad (2.5.32)$$

Figures 2.1(a) and (b) show the convergence of the polynomial feedback parameters, although in 2.1(b) g_2 is not shown. Figures 2.2(a) and (b) show the variation of the state feedback parameters during the simulation. In both cases the dotted lines designate the parameter values to which the estimates should converge. However, in the state space case these convergence points can only be obtained by taking account of the steady values of the $\hat{a}_1$, $\hat{b}_1$ etc. parameters. This point is considered further in chapter 3.

Figure 2.3 shows the disturbance, e(t), affecting the system and figures 2.4(a) and (b) show the difference between this noise signal and the estimation error, $\varepsilon(t)$, obtained from the least squares estimation procedure. This difference generally tends to zero, although on start up and after the change in time delay it requires several sampling instants to settle down to its more usual position. It has been shown, Wellstead et al (1979b) that for self-tuning to occur, in the polynomial case, the error term $\varepsilon(t)$ becomes equal to e(t), i.e. $e(t) - \varepsilon(t) \to 0$. This is discussed with a view to the state space approach in Sec.3.1.

In Figures 2.4-6,(a) gives the polynomial controller response, whereas (b) shows that of the state space controller. Figure 2.5

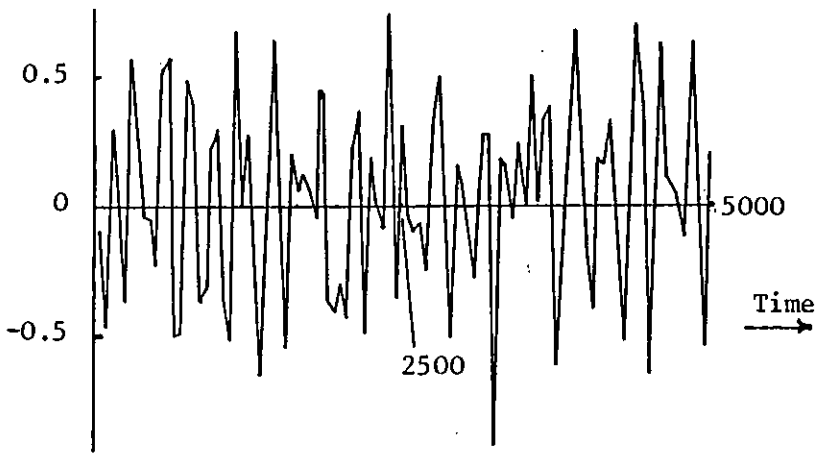


Fig. 2.3. Noise Signal, e(t)

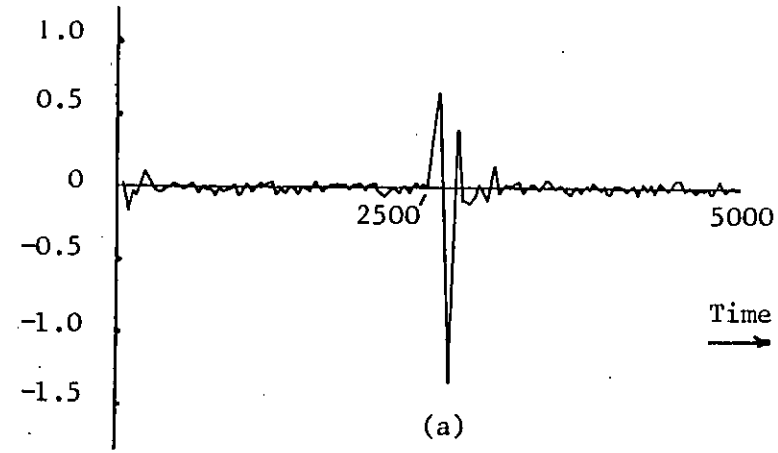
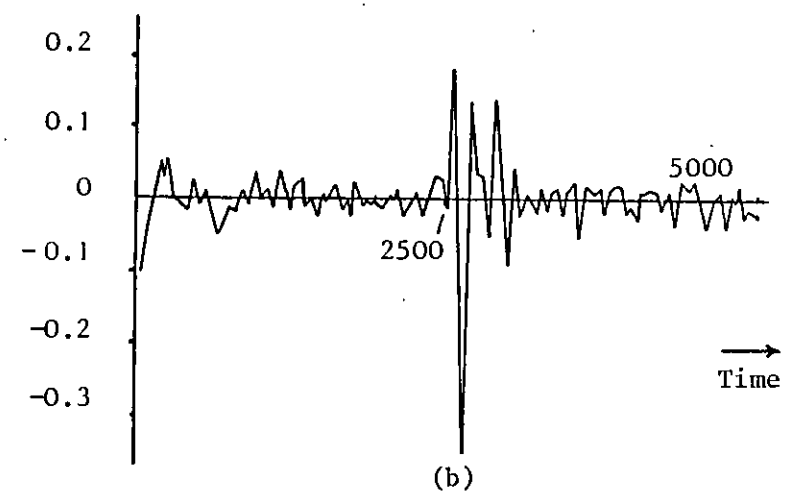


Fig. 2.4 (a) e(t)-ε(t), under polynomial controller.
(b) e(t)-ε(t), under state space controller.



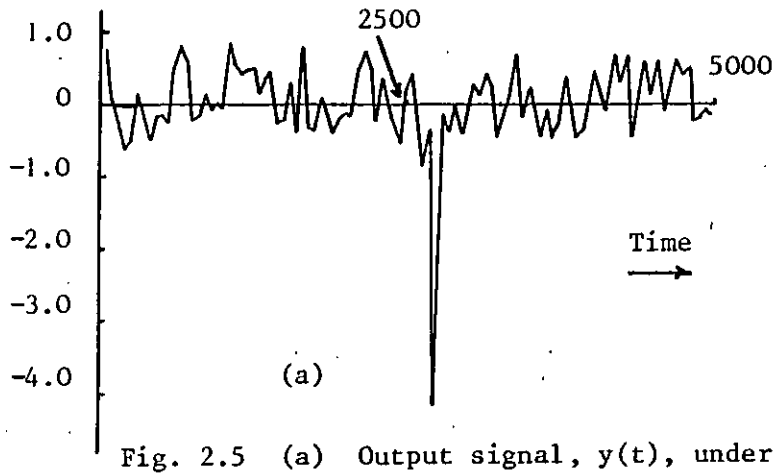
showing the system output and Fig. 2.6 the input.

The asymptotic variances of various signals were obtained by calculations from the signals themselves, and these are given in the following table. The terms in parentheses are the respective theoretical values calculated from the actual system parameters.

Table 2.1

Signal type	Variance before change in the time delay	Variance after change in time delay	Figure
Polynomial Controller			
Output	0.1298(0.1284)	0.1602(0.1575)	2.5(a)
Input	0.1224	0.0116	2.6(a)
$e(t)-\varepsilon(t)$	0.0011	0.0097	2.4(a)
State Space Controller			
Output	0.1513(0.1517)	0.1537(0.1554)	2.5(b)
Input	0.0028	0.0048	2.6(b)
$e(t)-\varepsilon(t)$	0.0004	0.0040	2.4(b)

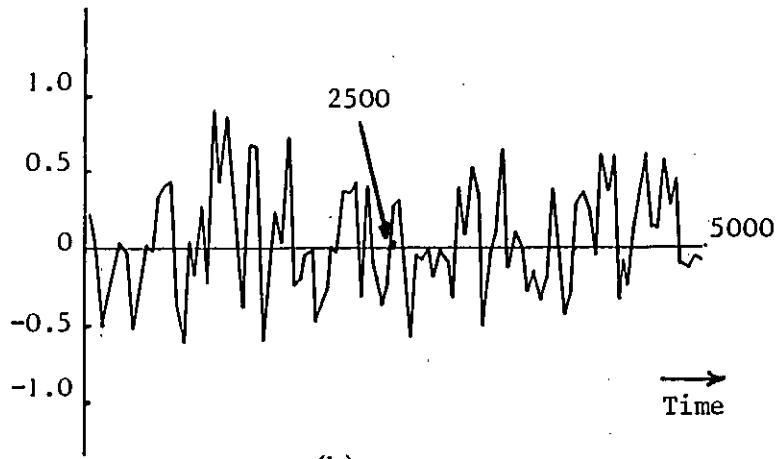
From the values obtained it is shown clearly that, for the example used in the simulation, the polynomial self-tuner has much better output regulation characteristics than the state space method before the change in time delay, yet once the time delay has been increased, a considerable increase in the output variance is apparent.



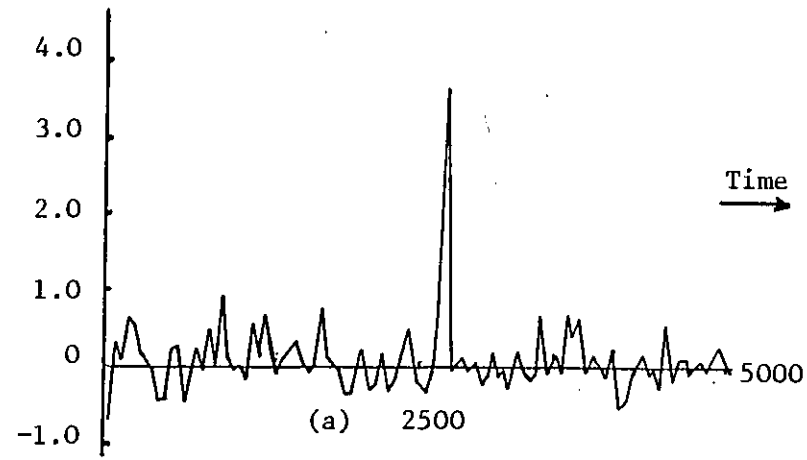
(a)

Fig. 2.5 (a) Output signal, y(t), under polynomial controller.

(b) Output signal, y(t), under state space controller.



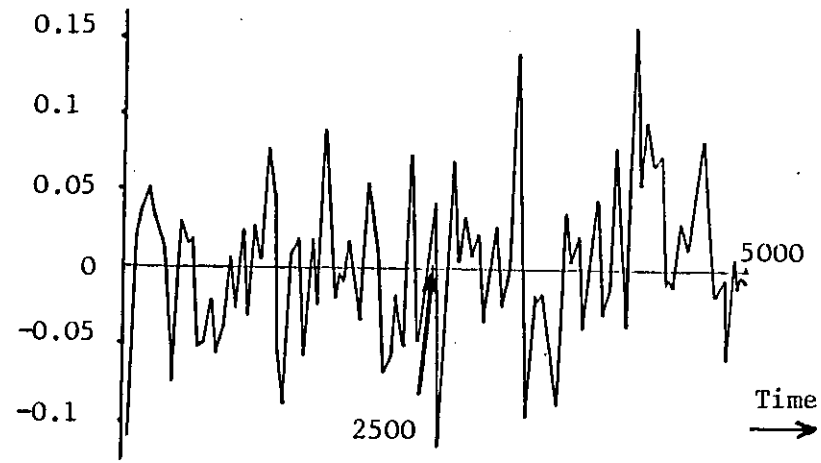
(b)



(a) 2500

Fig. 2.6 (a) Input signal, u(t), under polynomial controller.

(b) Input signal, u(t), under state space controller.



(b)

The state space tuner, however, gives a relatively poor output regulation before the time delay change, but the alteration in the value of variance caused by the change in time delay is much lower. This smoother transition tends to eliminate 'spikes' in the output signal, of the mode found in the polynomial case, Fig.2.5(a).

These 'spikes' also appear in the polynomial control input, Fig.2.6(a), and this raises the question of saturation effects in an actual system, as opposed to a simulated one. Rapidly varying and often excessively large input signals are held to be one of the major disadvantages of the minimum variance type tuners, and therefore the lighter control action employed by the state space method, throughout the simulation, Fig.2.6(b), is a desirable feature in tuning controllers.

Both methods used in the example, though, offer the advantage of dealing readily with nonminimum phase plant characteristics. Due to the possibility of instability the same cannot be said of the minimum variance type tuners, which also have problems in coping with variable system time delays.

2.6 Concluding Remarks

The feedback vector, F(t), obtained from the matrix Riccati equation (2.4.4), is unaffected in value by the choice of $C(z^{-1})$ = unity. In fact if F(t) is found such that u(t) is an optimal control in terms of the stochastic linear regulator problem, then the same control is also optimal in the deterministic case, Davis (1977). The feedback, F(t), obtained from (2.4.7) will be calculated, if the system parameters are known, to be the same whether the system is considered to have a

disturbance or not, and therefore is not dependent on $C(z^{-1})$. The addition of a disturbance is accounted for in the calculation of the state estimate, and hence in the control input. However, the equality (2.4.9) is only of use when a stochastic system is considered, thus the assumption $C(z^{-1}) = 1$, although having no direct affect on F(t), will have an effect on the parameter equalities (2.4.10) when $C(z^{-1}) \neq 1$, i.e. at least one of $c_1, c_2, \ldots, c_{n_c}$ is non-zero, in which case the general expression (1.5.4) must be employed.

In the single stage cost function design, the self-tuning algorithm was considered in the state space with the estimation of the state vector employing all information up to and including the present time, thus giving an overall control action equivalent to that of the polynomial controller. In addition, the system time delay, k, was considered to be known, whereas the state space model and the parameter estimator allow for a variable k. Therefore, if a state space single stage control law with variable k and using the state estimation discussed in Sec. 2.3 is employed, a different, but no longer optimal, control action will be achieved.

The single stage design required a modified estimated state vector, $\underline{\hat{x}}'(t)$, and this was also shown to be the case, Sec.2.5, when a pole placement procedure was considered, where equivalence with the polynomial case was necessary. There are, however, many state estimator configurations which can be used, but not all of these have the polynomial $C(z^{-1})$ as a denominator. With this as the observer polynomial the regulator equations are much simplified, a characteristic shared by controllers using the $\underline{\hat{x}}(t)$ estimate, and this leads to one of the main advantages of self-tuners.

The use of the state space, as opposed to polynomial, design procedure for the pole placement self-tuner results in a distinct reduction in computational effort, despite the fact that a state space framework must be built up from a polynomial system model. This is largely due to the inversion of the W matrix, which by means of Gaussian elimination, needs far fewer calculations than its equivalent in the polynomial case.

CHAPTER 3

PROPERTIES OF STATE-SPACE SELF-TUNERS

Having established the basis for the use of state-space definitions in self-tuning algorithms, it is a consequence that in this chapter some of the properties of this method are discussed. Primarily, the self-tuning property of state-space self-tuners is established in Sec. 3.1, by a generalisation of the proof attached to polynomial type tuners. This leads to a larger number of possible schemes, all of which hold with the self-tuning property, although they provide different controller actions.

In Sec. 3.2, the possibility of tuning, rather than preselecting, the closed loop poles is considered. By this means the variance of the output signal can be lessened by only a slight increase in computational effort when the system model is of low dimension. A simulation study is included in this particular section, whereby the output variance of a nonminimum phase process obtained using tuned poles is compared with that encountered when the poles have been chosen prior to controller start up.

The effect of an external input on the closed loop system, under the operation of a state space controller, is introduced in Sec. 3.3. This is done initially from a pole placement viewpoint, although a comparison is made between this type of set-point following and various methods previously developed with a polynomial framework. Several modes of connection of the external input to the state space system are considered and numerous simulations are carried out to assess the response of the process to a change in reference input.

3.1 Self-Tuning Property

1. Introduction

The fundamental assumption, obtained originally from implicit self-tuning theory, is that for self-tuning to occur, the control input obtained from the estimated system model converges to the value which would be obtained if the actual system parameters were known. Because of this only a limited number of control strategies are found to be applicable. In this section a general rule is formed, whereby the self-tuning property of pole placement self-tuners is obtained. The previous strategies of the original pole placement, Wellstead et al (1979b), and the state space method, Sec. 2.5, are shown to be special cases of this.

The self-tuning property for these controllers was first discussed in Wellstead et al (1979b), with further comments being made in Allidina and Hughes (1980). Now, however the properties of the state space controller developed in Sec. 2.5 must be considered, and this is done by generalising the original proof to account for a larger number of controller parameters. In subsection 2, the order of the numerator and denominator of the control input polynomial transfer function are both increased by unity to allow for the extended version. The general property for the convergence of this modified controller is then discussed in subsection 3, the relationship between the state space and polynomial forms being considered further in subsection 4.

2. Self-tuner formulation

The unknown continuous-time system is modelled in discrete time form as

$$A(z^{-1})y(t) = z^{-k}B(z^{-1})u(t) + C(z^{-1})e(t) \qquad (3.1.1)$$

where

$$\left.\begin{aligned} A(z^{-1}) &= 1 + a_1 z^{-1} + \ldots + a_{n_a} z^{-n_a} \\ B(z^{-1}) &= b_0 + b_1 z^{-1} + \ldots + b_{n_b} z^{-n_b} \\ C(z^{-1}) &= 1 + c_1 z^{-1} + \ldots + c_{n_c} z^{-n_c} \end{aligned}\right\} \qquad (3.1.2)$$

and z^{-1} is the backward shift operator, $k \geq 1$ and $b_0 \neq 0$ by definition. $\{y(t):t\epsilon T\}$and $\{u(t):t\epsilon T\}$ are sequences of output and input variables respectively, measured at the sampling instants, and $\{e(t):t=0,\pm1,\pm2,\ldots\}$ is a zero-mean white noise sequence with finite variance, such that

$$E\{e(i)e^{T}(j)\} = \delta_{ij}\Omega \qquad (3.1.3)$$

where $E\{\cdot\}$ signifies the expected value and δ_{ij} is the Kronecker delta.

The control input is chosen to be,

$$D(z^{-1})u(t) = G(z^{-1})y(t) \qquad (3.1.4)$$

where

$$\left.\begin{aligned} D(z^{-1}) &= 1 + d_1 z^{-1} + \ldots + d_{n_d} z^{-n_d} \\ G(z^{-1}) &= g_0 + g_1 z^{-1} + \ldots + g_{n_g} z^{-n_g} \end{aligned}\right\} \qquad (3.1.5)$$

and $n_d = n_b + k;\ n_g = n_a$.

By substituting (3.1.4) into (3.1.1) the closed loop equation is given by

$$[A(z^{-1})D(z^{-1}) - z^{-k}B(z^{-1})G(z^{-1})]y(t) = C(z^{-1})D(z^{-1})e(t)$$

... (3.1.6)

In the pole placement technique it is now necessary to specify the required closed loop pole polynomial, and this is defined as being

$$T(z^{-1}) = 1 + t_1 z^{-1} + \ldots + t_{n_t} z^{-n_t} \tag{3.1.7}$$

where the constituent parameters $\{t_i : i=1,\ldots n_t\}$ are chosen to meet the selected pole positions.

The control polynomials, $D(z^{-1})$ and $G(z^{-1})$, can now be found from the equation,

$$A(z^{-1})D(z^{-1}) - z^{-k}B(z^{-1})G(z^{-1}) = C(z^{-1})T(z^{-1}) \tag{3.1.8}$$

which is derived from the closed loop equation (3.1.6).

However, the system parameters are unknown, and hence the system model must be estimated by means of the recursive least squares procedure operating on

$$\hat{A}(z^{-1})y(t) = \hat{B}'(z^{-1})u(t) + \varepsilon(t) \tag{3.1.9}$$

where

$$\left.\begin{aligned} \hat{A}(z^{-1}) &= 1 + \hat{a}_1 z^{-1} + \ldots + \hat{a}_{n_a} z^{-n_a} \\ \hat{B}'(z^{-1}) &= \hat{b}_0 z^{-1} + \hat{b}_1 z^{-2} + \ldots + \hat{b}_{n_b+k-1} z^{-(n_b+k)} \end{aligned}\right\} \tag{3.1.10}$$

Once again the control input (3.1.4) may be employed, although the closed loop equation now becomes,

$$[\hat{A}(z^{-1})D(z^{-1}) - \hat{B}'(z^{-1})G(z^{-1})]y(t) = D(z^{-1})\varepsilon(t) \tag{3.1.11}$$

However, to arrive at the specified closed loop pole polynomial form, the parameters of $D(z^{-1})$ and $G(z^{-1})$ must be calculated from the identity,

$$\hat{A}(z^{-1})D(z^{-1}) - \hat{B}'(z^{-1})G(z^{-1}) = T(z^{-1}) \tag{3.1.12}$$

where $T(z^{-1})$ is defined in (3.1.7).

For the self-tuning property to hold it is now required that the parameters of $D(z^{-1})$ and $G(z^{-1})$ obtained recursively from

equation (3.1.12) have a possible convergence point at the values obtained by solving (3.1.8) off line, subject to the convergence of the parameters in the least-squares estimate model (3.1.9). The properties of the least squares process can then be applied with regard to,

$$\left.\begin{array}{ll} \text{i)} & E\{y(t-i),\varepsilon(t)\} \\ \text{ii)} & E\{u(t-j),\varepsilon(t)\} \end{array}\right\} \qquad (3.1.13)$$

for $i = 1,2,\ldots,n_a$ and $j = 1,2,\ldots,n_b+k$.

The closed-loop equation is given, from equations (3.1.11) and (3.1.12) by

$$T(z^{-1})y(t) = D(z^{-1})\varepsilon(t) \qquad (3.1.14)$$

and iff $\varepsilon(t) \to e(t)$ as $t \to \infty$

$$T(z^{-1})y(t) = D(z^{-1})e(t) \qquad (3.1.15)$$

which is the closed loop form obtained from (3.1.6) and (3.1.8).

3. General convergence property

The requirement is that the parameters of the $D(z^{-1})$ and $G(z^{-1})$ polynomials obtained recursively from (3.1.12) can converge to those obtained from (3.1.8), and that on convergence $\varepsilon(t) = e(t)$.

Let $n_\lambda = n_a + n_b + k$.

Then defining a matrix, λ, as

$$\lambda = \begin{array}{c} \uparrow \\ n_a \\ \downarrow \end{array} \left\{ \begin{array}{cccccccccc} 1 & d_1 & \cdots & & d_{n_d} & 0 & \cdots & & 0 \\ 0 & \ddots & \ddots & & & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & & & & \ddots & & 0 \\ 0 & \cdots & 0 & 1 & d_1 & \cdots & & & d_{n_d} \\ \hline g_0 & \cdots & & & g_{n_g} & 0 & \cdots & & 0 \\ 0 & \ddots & & & & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & & & & \ddots & & 0 \\ 0 & \cdots & 0 & g_0 & \cdots & & & & g_{n_g} \end{array} \right\} \begin{array}{c} \uparrow \\ n_b+k \\ \downarrow \end{array} \qquad (3.1.16)$$

where λ is of dimension $n_\lambda \times n_\lambda$ and is almost always of full rank, Warwick (1981b).

We also define a sequence $\{w(t):t\epsilon T\}$ where $w(t)$ is obtained from,

$$\begin{array}{ll} (a) & y(t) = D(z^{-1})w(t) \\ \text{or} \quad (b) & \varepsilon(t) = T(z^{-1})w(t) \end{array} \qquad (3.1.17)$$

where it must be remembered that the order of the $T(z^{-1})$ polynomial, n_t, is such that

$$n_t \leq n_\lambda - n_c \qquad (3.1.18)$$

which is necessary for the identity (3.1.8) to have equality. However, n_t may be limited further by the dimension of the state space, and this possibility is considered in subsection 4.

From (3.1.17) we obtain,

$$\lambda[R_{w\varepsilon}(1),\ldots,R_{w\varepsilon}(n_\lambda)]^T = 0 \qquad (3,1,19)$$

where $$R_{w\varepsilon}(i) = E\{w(t-i),\varepsilon(t)\} \qquad (3.1.20)$$

thus $$R_{w\varepsilon}(i) = 0 \quad \text{for } i = 1,\ldots,n_\lambda \qquad (3.1.21)$$

However, by use of (3.1.6) and (3.1.14),

$$[A(z^{-1})D(z^{-1}) - z^{-k}B(z^{-1})G(z^{-1})]\varepsilon(t) = C(z^{-1})T(z^{-1})e(t) . \qquad \ldots \quad (3.1.22)$$

Both sides of this equation must then be multiplied by $w(t-(n_\lambda+1))$, and expected values taken. From the result, by means of (3.1.21) and (3.1.17a) the following arises,

$$R_{w\varepsilon}(n_\lambda+1) = 0 \quad . \qquad (3.1.23)$$

The same result is also found when both sides of (3.1.22) are multiplied by $w(t-(n_\lambda+2))$, and by continuing this process for all j in $w(t-(n_\lambda+j))$, where $j \geq 1$, it follows that

$$R_{w\varepsilon}(i) = 0 \qquad \text{for all } i \geq 1 \quad . \qquad (3.1.24)$$

It is also apparent that $\varepsilon(t)$ is a white process, as from (3.1.17b),

$$R_{\varepsilon\varepsilon}(i) = R_{w\varepsilon}(i) + t_1 R_{w\varepsilon}(i+1) + \ldots + t_{n_t} R_{w\varepsilon}(i+n_t) \qquad (3.1.25)$$

and by use of (3.1.24) all terms on the right hand side of (3.1.25) are identically zero.

This means that by consideration of (3.1.22),

$$\varepsilon(t) = e(t). \qquad (3.1.26)$$

4. Convergence applied to state-space representation

The state-space description is defined as being,

$$\left.\begin{aligned} &(a) \quad \underline{x}(t+1) = P\underline{x}(t) + Qu(t) + Re(t) \\ &(b) \quad y(t) = H\underline{x}(t) + e(t) \end{aligned}\right\} \qquad (3.1.27)$$

where the constituent matrices are given in Sec. 2.2(2), and the state vector is of dimension $(n_1+k)\times 1$.

The estimation of the state is then calculated from,

$$\hat{\underline{x}}(t) = z^{-1}[I-z^{-1}\bar{P}]^{-1}\{Qu(t) + Ry(t)\} \qquad (3.1.28)$$

where $\bar{P} = P - RH$

and using the control input,

$$u(t) = F\underline{\hat{x}}(t) \tag{3.1.29}$$

where $F = [f_1,\ldots,f_{n_1+k}]$, and is obtained by solving the equation,

$$F^T = W^{-1}(T+R') \tag{3.1.30}$$

in which T and W are defined in Sec. 2.5, and

$$R' = [0,\ldots,0,-a_{n_1},\ldots,-a_1]^T \quad . \tag{3.1.31}$$

By substituting for the control input (3.1.29) into the state equations (3.1.27), the closed loop equation is given by

$$y(t) = [z^{-1}H(I-z^{-1}\bar{F})^{-1}R+1]e(t) \tag{3.1.32}$$

where $\bar{F} = P + QF$.

Here the parameters in the matrix F are found by solving the equation,

$$\det(I-z^{-1}\bar{F}) = T(z^{-1}) \tag{3.1.33}$$

where $T(z^{-1})$ is the operator selected pole polynomial defined in (3.1.7).

In this state-space mode, however, a further restriction is placed on the maximum possible value of n_t, the order of the $T(z^{-1})$ polynomial. In (3.1.18) $n_t \leq n_\lambda - n_c$, where $n_\lambda = n_a + n_b + k$. For (3.1.33) to be compact, though, n_t cannot be of a higher order than the dimension of the state-space. This state-space dimension is given here as n_1+k, where $n_1 = \max\{n_a,n_b,n_c\}$, so that if ℓ is defined to be

$$\ell = \min\{(n_\lambda - n_c),(n_1+k)\} \tag{3.1.34}$$

then $$n_t \leq \ell \quad . \tag{3.1.35}$$

By means of the definitions given for the state-space representation we now wish to show that by using the state-space description,

the closed loop transfer function is identical to that obtained from the generalised polynomial case. With this as a starting point it can then be proved that the generalised self-tuning property obtained is equally applicable to the state space self-tuning controller.

For equivalence of closed loop forms it therefore remains to be shown that (3.1.15) is the same as (3.1.32), and that the control input obtained via the state space description, and given by

$$\{1-z^{-1}F[I-z^{-1}\bar{P}]^{-1}Q\}u(t) = z^{-1}F[I-z^{-1}\bar{P}]^{-1}Ry(t) \qquad (3.1.36)$$

from (3.1.27), (3.1.28) and (3.1.29); is equivalent to the polynomial control input (3.1.4).

Breaking down (3.1.36) for analysis, we have

$$\bar{P} = P-RH = \begin{pmatrix} 0 & \cdots & 0 \\ 1 & \ddots & \vdots \\ 0 & \ddots & 0 \\ \vdots & \ddots & -c_{n_1} \\ 0 & \cdots\ 0\ 1 & -c_1 \end{pmatrix} \qquad (3.1.37)$$

P,R and H being defined in (2.2.4)

Hence, it follows that

$$(I-z^{-1}\bar{P}) = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ -z^{-1} & \ddots & & \vdots \\ 0 & \ddots & 0 & 0 \\ \vdots & \ddots & \vdots & z^{-1}c_{n_1} \\ \vdots & \ddots & 0 & \vdots \\ \cdot & \ddots & 1 & z^{-1}c_2 \\ 0 & \cdots\ 0 & -z^{-1} & 1+z^{-1}c_1 \end{pmatrix} \qquad (3.1.38)$$

which gives

$$\mathrm{Det}(I-z^{-1}\bar{P}) = C(z^{-1}) \qquad (3.1.39)$$

Lemma 3.1 :

$$\{1+Hz^{-1}(I-z^{-1}\bar{F})^{-1}R\}T(z^{-1}) = \{1-Fz^{-1}(I-z^{-1}\bar{P})^{-1}Q\}C(z^{-1}) = D(z^{-1})$$

iff $\mathrm{Det}(I-z^{-1}\bar{F}) = T(z^{-1})$

Proof: can be shown by induction

As an example a simple 2×2 system is considered, where the following hold; $n_a = n_c = 2$, $k = n_b = 1$; then

$$\{1-Fz^{-1}(I-z^{-1}\bar{P})^{-1}Q\}.C(z^{-1}) = 1 + z^{-1}(c_1-b_1f_1-b_0f_2) + z^{-2}(c_2-b_1c_1f_1+b_0c_2f_1-b_1f_2)$$

and $$\{1+Hz^{-1}(I-z^{-1}\bar{F})^{-1}R\} = \frac{1}{\mathrm{Det}(I-z^{-1}\bar{F})}\{1+z^{-1}(c_1-b_1f_1-b_0f_2) + z^{-2}(c_2-b_1c_1f_1+b_0c_2f_1-b_1f_2)\}$$

When $\mathrm{Det}(I-z^{-1}\bar{F}) = T(z^{-1})$ the Lemma is proved.

It follows that; $D(z^{-1}) = 1+z^{-1}(c_1-b_1f_1-b_0f_2) + z^{-2}(c_2-b_1c_1f_1 + b_0c_2f_1-b_1f_2)$

Also from (3.1.36) and (3.1.4) it is necessary that

$$\{z^{-1}F(I-z^{-1}\bar{P})^{-1}R\}.C(z^{-1}) = G(z^{-1}) \qquad (3.1.40)$$

and by re-using the example employed in the proof of Lemma 3.1

$$G(z^{-1}) = z^{-1}(c_2f_1-a_2f_1+c_1f_2-a_1f_2) + z^{-2}(a_1c_2f_1-a_1c_1f_1+c_2f_2-a_2f_2).$$

The representations obtained here for $D(z^{-1})$ and $G(z^{-1})$ lead to a direct comparison with the theoretical generalised polynomial definition (3.1.5).

For the same example the order of the above polynomials is $n_d = n_b+k = 2 = n_a = n_g$.

Therefore, we have from the polynomial description

i) $D(z^{-1}) = 1 + d_1z^{-1} + d_2z^{-2}$

and ii) $G(z^{-1}) = g_0 + g_1z^{-1} + g_2z^{-2}$.

But from the state space example the two polynomials may be rewritten in the form.

$$\text{i)} \quad D(z^{-1}) = 1 + d_1 z^{-1} + d_2 z^{-2}$$

$$\text{ii)} \quad G(z^{-1}) = \quad g_1 z^{-1} + g_2 z^{-2}$$

This means that although the $D(z^{-1})$ polynomial is directly of an identical form, the state space representation causes the g_0 parameter to be zero. As an explanation of this, it must be remembered that the state reconstruction incorporated in the design is only dependent on values of the output sequence $\{y(t)\}$ up to and including time t-1, and not inclusive of time t.

Lemma 3.1 thus shows that the control law employed in the state space is identical to the polynomial control law (3.1.4), in its generalised form, and the closed loop state space transfer function is identical to that obtained in (3.1.15).

It, therefore, must now be shown that solving (3.1.33) is identical to solving (3.1.8), and this is considered in the following Lemma.

Lemma 3.2 :

$$\text{Det}(I-z^{-1}\bar{F}) = A(z^{-1}) \frac{D(z^{-1})}{C(z^{-1})} - z^{-k}B(z^{-1}) \frac{G(z^{-1})}{C(z^{-1})} = T(z^{-1})$$

Proof of Lemma 3.2 is once more carried out simply by means of an inductive process.

Using the same example, as was used in the proof of Lemma 3.1, for continuity,

$$\text{Det}(I-z^{-1}\bar{F}) = 1 + z^{-1}(a_1-b_0f_2-b_1f_1)+z^{-2}(a_2-a_1b_1f_1-b_1f_2+a_2b_0f_1)$$

and for Lemma 3.1 to hold, this is equal to $T(z^{-1})$.

Using the results obtained for the control law polynomials,

$$C^{-1}(z^{-1})\ \{A(z^{-1})D(z^{-1})-z^{-k}B(z^{-1})G(z^{-1})\} = 1 + z^{-1}(a_1-b_0f_2-b_1f_1)$$

$$+z^{-2}(a_2-a_1b_1f_1-b_1f_2+a_2b_0f_1) = T(z^{-1})$$

Hence Lemma 3.2 is proved, and the state feedback terms f_1, f_2 can be calculated without consideration of c_1, c_2. This independence of calculated feedback terms from the parameters of the $C(z^{-1})$ polynomial is not apparent in the polynomial method itself, where d_1, d_2, g_0, g_1 and g_2 are directly related to c_1, c_2 from (3.1.8).

It has been shown, therefore, that the self-tuning property applies to the state space description. Thus the nature of the state space representation containing the estimated model parameters must now be reviewed. The state space formulation being given by,

$$\begin{aligned} &(a) \quad \tilde{\underline{x}}(t+1) = \hat{P}\tilde{\underline{x}}(t) + \hat{Q}u(t) + \hat{R}\varepsilon(t) \\ &(b) \quad y(t) = H\tilde{\underline{x}}(t) + \varepsilon(t) \end{aligned} \tag{3.1.41}$$

where

$$\hat{P} = \begin{pmatrix} 0 & \cdots & \cdots & 0 \\ 1 & \ddots & & \vdots \\ & \ddots & 0 & 0 \\ 0 & \ddots & \ddots & -\hat{a}_n \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & -\hat{a}_1 \end{pmatrix} ; \quad \hat{Q} = \begin{pmatrix} \hat{b}_{n+k_m-1} \\ \vdots \\ \hat{b}_1 \\ \hat{b}_0 \end{pmatrix} \tag{3.1.42}$$

$$R' = [0, \ldots, 0, -\hat{a}_n, \ldots, -\hat{a}_1]$$

and k_m is the maximum possible value of the system integer time delay, k.

The estimated state vector, formed by use of the model estimates is then given as,

$$\hat{\tilde{\underline{x}}}(t) = z^{-1}[I-z^{-1}\hat{P}]^{-1}\{\hat{Q}u(t) + \hat{R}y(t)\} \tag{3.1.43}$$

where $\hat{\bar{P}} = \hat{P} - \hat{R}H$.

The state feedback vector is once more obtained from (3.1.30), although we have

$$R' = [0, \ldots, 0, -\hat{a}_n, \ldots, -\hat{a}_1] = \hat{R} \tag{3.1.44}$$

Lemmas 3.1 and 3.2. both hold with the estimated parameter matrices, and as $C(z^{-1})$ is equal to unity, from Lemma 3.2,

$$\text{Det}(I-z^{-1}\hat{\bar{F}}) = \hat{A}(z^{-1})D(z^{-1}) - \hat{B}'(z^{-1})G(z^{-1}) = T(z^{-1}) \tag{3.1.45}$$

where $$\hat{\bar{F}} = \hat{P} + \hat{Q}\hat{F} \tag{3.1.46}$$

Thus, calculating $\text{Det}(I-z^{-1}\hat{\bar{F}}) = T(z^{-1})$ is equivalent to calculating (3.1.12).

It has been shown that as the parameter estimates converge, the control obtained by calculating the polynomials $D(z^{-1})$ and $G(z^{-1})$ from (3.1.12) has a possible convergence point at the polynomials obtained from an off line calculation of (3.1.8). Hence, the calculation of $\hat{F}$ by means of $\text{Det}(I-z^{-1}\hat{\bar{F}}) = T(z^{-1})$ gives equivalent values for the parameters of $D(z^{-1})$ and $G(z^{-1})$ which can converge to those obtained via $\text{Det}(I-z^{-1}\bar{F}) = T(z^{-1})$.

Therefore, on condition that the model estimates converge, the control obtained recursively by using $u(t) = \hat{F}\hat{\underline{x}}(t)$ can converge to that obtained from the offline solution of $u(t) = F\hat{\underline{x}}(t)$ in the case when the system parameters are known. Thus, although the system parameters are not known, the control calculated via the state space, from the estimated model parameters, can converge to the required control format.

It can be seen from the equations for calculating the closed loop pole polynomial, in the state space, that the state feedback vectors $\hat{F}$ and F are obtained without directly employing any

parameters contained in the $C(z^{-1})$ polynomial, i.e. c_1, c_2, etc. . It follows that as the parameters of $\hat{A}(z^{-1})$ and $\hat{B}'(z^{-1})$ do not converge, respectively, to those of $A(z^{-1})$ and $B(z^{-1})$; unless $c_1 = c_2 = \ldots = c_{n_c} = 0$, so $\hat{F}$ does not converge to F as the estimates converge. The asymptotic values of F can, therefore, only be calculated, if required, by consideration of the asymptotic values of the model estimates, $\hat{a}_1$, $\hat{b}_1$, etc. . This apparent anomaly is eradicated, though, by the estimated state vector obtained using the parameter estimates, $\hat{\underline{x}}(t)$, not converging to the estimated state vector found from the actual system parameters, $\hat{\underline{x}}(t)$. In contrast to this, the overall factor $F\hat{\underline{x}}(t)$, does have a convergence point, as has been shown, at $F\underline{x}(t)$, and hence the control input is able to converge to that calculated offline if the model parameters, a_1, b_1, c_1 etc., were known.

5. Notes on convergence property

It is apparent from (3.1.5) that the constituent parameters of the polynomials $D(z^{-1})$ and $G(z^{-1})$ cannot be calculated explicitly by consideration of only (3.1.8) or (3.1.12), as there exist $n_\lambda+1$ unknowns but only n_λ equations. The various self-tuning methods formulated cope with this problem in a number of ways. In the polynomial type tuner, Wellstead et al (1979b), n_d and n_g are both reduced, in order, by unity, resulting in $n_\lambda-1$ unknowns and $n_\lambda-1$ equations. Hence, with this restriction enforced, only one solution is possible for each parameter of $D(z^{-1})$ and $G(z^{-1})$ at each recursive instant. This solution, though, suffers from an advance side-effect, in that from (3.1.18) the maximum number of closed loop poles, n_t, that may be placed,

are reduced by one. By considering this method from a different viewpoint, the equivalent is arrived at by, effectively, specifying d_{n_d} as being equal to zero. Then, by the nature of the problem, g_{n_g} is also equal to zero, and thus we have reduced the number of unknowns from $n_\lambda+1$ to $n_\lambda-1$ by a reduction of one in the number of equations.

The state space method, Warwick (1981a), has the effect of making g_0 equal to zero, as mentioned in Sec. 3.1(4), in its equivalent polynomial type tuner. The number of unknowns is, by this means, reduced from $n_\lambda+1$ to n_λ, with the total number of equations, n_λ, remaining unaffected. This method, then, also gives rise to a unique solution for all parameters of $D(z^{-1})$ and $G(z^{-1})$ at each iteration, although in this case the maximum number of closed loop poles available for placement, n_t, is not affected.

3.2 Self-Assigning Poles

1. Introduction

Self-tuning controllers based on pole placement principles provide a more robust approach when compared to those based on minimum variance techniques. However, the price paid for this robustness is a small loss in the regulation characteristics. When regulation is of prime importance, as opposed to the system's response to an external input signal, it may be decided that the closed loop pole positions are required to be dependent on the positions necessary to produce the minimum possible variance in the output signal from the unknown process.

Assuming the estimates of the parameters in the system model converge, the state feedback factors $\hat{f}_i$ will also converge to steady values, Sec. 3.1. Thus, as the output signal variance can be calculated, theoretically, by use of these estimates, in addition to the previously chosen closed loop poles, it is therefore possible, by correct choice of these poles, to minimize the variance.

As one assumes no apriori knowledge of the system parameters when carrying out self-tuning, no apriori calculation can be carried out such that this minimization occurs. Thus it is necessary to consider the possible use of a cost function, which is equivalent to the value of 'relative' output variance, this being minimized by the correct choice of poles at the sampling instants.

2. Single pole assignment

The total number of poles that may be assigned is limited by the dimensions of the state, Sec. 3.1, but a further limitation becomes apparent when a self-assigning pole technique is used. It is required that the amount of computation necessary for the overall self-tuning algorithm is kept as small as possible. The placement of poles with respect to a cost functional merely adds to the total computational effort. It will be shown that with low order systems a single pole can be placed optimally with very little extra computational burden, and that there is, in fact, no real necessity for the placement of more than one pole.†

We now consider the form of the closed loop equation obtained by using state-space methods, such that a choice of pole position may be made. The closed loop equation is given from (3.1.32) as,

†Simulated examples have shown that the increased computation necessary for a larger number of poles outweighs the improved performance obtained.

$$y(t) = [z^{-1}H(I-z^{-1}\bar{F})^{-1}R + 1]e(t) \qquad (3.2.1)$$

where $\bar{F} = P + QF$.

This may be rewritten by making use of the identity; $\mathrm{Det}(I-z^{-1}\bar{F}) = T(z^{-1})$, as

$$y(t) = \left\{ \frac{z^{-1}H\{\mathrm{adj}(I-z^{-1}\bar{F})\}R + T(z^{-1})}{T(z^{-1})} \right\} e(t) \qquad (3.2.2)$$

where $T(z^{-1})$ is the pole polynomial defined in (3.1.7).

Here, however, we do not wish to specify t_1, t_2 etc., as exact numerical quantities, but rather as algebraic expressions, relating them to the system model parameters. Although, as only one pole is being adjusted, all other poles will be considered to reside at the origin of the z-plane. Thus the pole polynomial becomes,

$$T(z^{-1}) = 1 + t_1 z^{-1} \qquad (3.2.3)$$

irrespective of the value of n_t, as long as $n_t \geq 1$.

The nature of the cost functional it is required to minimize is now considered by reformulating (3.2.2) as follows,

$$y(t) = (1 + h_1 z^{-1} + h_2 z^{-2} + \ldots)e(t)$$

or

$$y(t) = \left\{ \sum_{i=0}^{\infty} h_i z^{-i} \right\} e(t) \qquad (3.2.4)$$

where $\{h_i : i = 0\} = 1$, and $\{h_i : i \geq 1\}$ are dependent on the t_1 coefficient of the $T(z^{-1})$ polynomial.

Hence,

$$\{\mathrm{Var}[y(t)]\} = (1^2 + h_1^2 + h_2^2 + \ldots)\{\mathrm{Var}[e(t)]\}$$

or Relative Variance = R.V. $= 1^2 + h_1^2 + h_2^2 + \ldots .$ (3.2.5)

and thus R.V. is the cost function which it is desired to minimize, where the minimum value is defined as,

$$\overline{\text{R.V.}} = \min\{\text{R.V.}\}$$

$$\text{and} \qquad \bar{t}_1 = \{t_1 : \text{R.V.} = \overline{\text{R.V.}}\ , \text{ iff } \overline{\text{R.V.}} \text{ exists}\} \qquad (3.2.6)$$

From (3.2.4) and (3.2.2) we obtain

$$\sum_{i=0}^{\infty} h_i z^{-i} = \frac{1+\ell_1 z^{-1} + \ldots + \ell_{n'} z^{-n'} + t_1 z^{-1}}{1 + t_1 z^{-1}}$$

where n' is the dimension of the state space.

$$\text{Then R.V.} = 1^2+\ell_1^2+(\ell_2-\ell_1 t_1)^2+\ldots+ \frac{(\ell_{n'}-\ell_{n'-1} t_1+\ldots(-1)^{n'-1}\ell_1 t_1^{n'-1})^2}{1 - t_1^2} \qquad \ldots(3.2.7)$$

where $\| t_1 \| < 1$ for convergence of the series used in obtaining (3.2.7). This corresponds to the assigned pole remaining within the unit circle in the z-plane; i.e. the pole polynomial must remain stable. As n' is known, this being a dimension of the model, and not necessarily that of the process under control, $\bar{t}_1$ can be found such $\overline{\text{R.V.}} = \text{R.V.}$. Thus it is required that

$$\overline{\text{R.V.}} = 1^2+\ell_1^2+(\ell_2-\ell_1 \bar{t}_1)^2+\ldots+ \frac{(\ell_{n'}-\ell_{n'-1} \bar{t}_1+\ldots(-1)^{n'-1}\ell_1 \bar{t}_1^{n'-1})^2}{1 - \bar{t}_1^2}$$

$$\text{Example: } n'=2 \ ; \ \overline{\text{R.V.}} = \frac{(1+\ell_1^2+\ell_2^2) - 2\ell_1\ell_2\bar{t}_1 - \bar{t}_1^2}{1 - \bar{t}_1^2} \ .$$

But the minimum value of R.V. cannot be obtained by direct partial differentiation of R.V. with respect to t_1, as ℓ_i; $i \geq 2$ depend on F,

which in turn depends on t_1 from (3.1.33). Thus ℓ_1, ℓ_2 etc. must be obtained in terms of a_1, b_1, t_1 etc. before minimization is calculated. From the example, we obtain :

$$\bar{t}_1 = \frac{b_0}{b_1} \quad \text{or} \quad \bar{t}_1 = \frac{b_1}{b_0} .$$

Hence a choice of optimal pole positions is achieved. To retain stability, however, we must employ

$$\bar{t}_1 = \frac{\min\{b_0, b_1\}}{\max\{b_0, b_1\}}$$

which means that as the $T(z^{-1})$ polynomial is always stable, no cancellation of unstable zeros will occur.

The value of the pole, t_1, can now be expressed in terms of the estimated system model parameters obtained from recursive least squares estimation. In this example $\hat{\bar{t}}_1$ is given by,

$$\hat{\bar{t}}_1 = \frac{\min\{\hat{b}_0, \hat{b}_1\}}{\max\{\hat{b}_0, \hat{b}_1\}}$$

Thus, the overall self-tuning algorithm, incorporating the self-assigning pole may be written as

1) Estimate system model parameters via recursive least squares estimation.

2) Calculate the state vector using these estimated values.

3) Obtain $\hat{\bar{t}}_1$

4) Compute state feedback vector from estimated parameters and $\hat{\bar{t}}_1$.

5) Apply new input signal from $u(t) = \hat{\underline{F}}\hat{\underline{x}}(t)$.

6) Repeat.

3. Simulation study

Consider the process described by the transfer function,

$$G(s) = \frac{6.35\ e^{-s\tau}}{s+1}$$

where τ is the transport delay, and the sampling interval is given as T. Then, if $\tau = 0.4$ secs. and $T = 0.5$ secs. the system may be regarded in the z-plane as being

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

and $C(z^{-1})e(t)$ is the system disturbance. The polynomials in the discrete time equation are obtained from the transfer function, G(s), as

$$A(z^{-1}) = 1-0.6065\ z^{-1}$$

$$B(z^{-1}) = 0.6043\ z^{-1} + 1.8942\ z^{-2}$$

and we define $C(z^{-1}) = 1 - 0.4\ z^{-1} + 0.03\ z^{-2}$.

with $\{e(t) : t = 0, \pm1, \pm2, \ldots\}$ a zero-mean, white noise sequence of variance = 0.11. Thus the system is nonminimum phase.

The simulation was operated over a period of 5000 sampling instants, equivalent to 2,500 secs., using the state space pole placement method derived in sec. 2.5(1). The self-assigning pole technique was used during one run, and its regulation characteristics were compared with the same simulation using poles chosen explicitly prior to commencement of the run.

The output variances obtained are given in the following table.

Table 3.1

t_1	Asymptotic Var{y(t)}	Theoretical Var{y(t)}
0.0	0.1177	0.1166
-0.8	0.1246	0.1228
-0.2	0.1183	0.1169
0.6	0.1179	0.1166
S.A.P.	0.1172	0.1163

where S.A.P. $\triangleq$ Self-Assigning Pole.

The optimum pole position for minimizing the output variance is calculated as

$$\bar{t}_1 = b_0/b_1 = 0.3190$$

and $\hat{t}_1$ approached this value asymptotically during the simulation.

4. Convergence and stability

When employing the single pole self-assigning principle the convergence of the estimated system model parameters is of utmost importance. Iff these converge, then the estimated state vector will converge, as it contains only parameters also present in the estimated system model. This is also true of the single pole, calculated as described in Sec. 3.2(2), and hence this can converge to the true value

$\bar{t}_1$ required to obtain the minimum value of relative output variance. Finally, as the value of the assigned pole converges to a steady position, so the state feedback vector will converge, as was proven in Sec. 3.1. Thus the overall self-tuning algorithm will cause the overall control input response obtained to have a possible convergence point at the required optimum value. Hence, although the convergence of the overall control response may be slower, though this is not necessarily true, the property of convergence remains unaffected by the use of the self-assigning pole technique.

The stability of the closed loop transfer function is assured in the steady state when the magnitude of the assigned pole is bounded to be less than unity, and this is also applicable to the case where closed loop poles are specifically chosen prior to tuner operation. Thus the overall system stability remains unchanged by the use of the self-assigning pole technique, provided all poles are bounded to be stable themselves. A limit is placed on the solution to the single self-assigned pole, however, in that it must remain as a real value, although where more than one pole is employed, complex pole pairs are acceptable.

5. Minimization of Relative Variance

It can be seen from (3.2.7) that the dimension of the state-space, n', has a great effect on the nature of the minimization techniques which may be used. By considering this equation to be ,

$$R.V. = 1 + (1-t_1^2)^{-1}\left[\sum_{j=0}^{n'-1}(-1)^j 2\left(\sum_{i=1}^{n'-j}\ell_i\ell_{i+j}\right)t_1^j - \sum_{i=1}^{n'}\ell_i^2\right] \qquad (3.2.8)$$

then R.V. is minimized with respect to t_1. But the degree of (3.2.8), in regard to the highest order of t_1, is equal to $n'-1$, with a minimum in order of two due to the $(1-t_1^2)$ term. Thus problems can arise in obtaining the minimum value of R.V. when n' is large. For $n' = 2$ a typical plot, is shown in Fig. 3.1, of R.V. against the variation of t_1. When $t_1 = \pm 1$, i.e. critical stability points, R.V. becomes infinite, once again because of the $(1-t_1^2)$ term. The minimum value of R.V., $\overline{R.V.}$, is given when $t_1 = \bar{t}_1$, which is retained within the bounds $-1 < t_1 < +1$. However, R.V. can, theoretically be lower in value than this when t_1 is outside the unit disc, as in the case of t_1'. But, two problems arising in this instance, discount the usage of such a closed loop pole. Not only is the closed loop transfer function unstable, but the series used in obtaining R.V. in (3.2.8) does not converge when $\| t_1 \| > 1$. $\overline{R.V.}$, therefore, becomes the only mathematically and physically possible optimum value of R.V.. Where $n' \geq 4$, the solution of (3.2.8) can become far more complicated, a typical plot for $n' = 4$ being given by Fig. 3.2. Although the value of R.V. increases to infinity when $\| t_1 \|$ approaches unity, between these values there may well be more than one minimum value of R.V. . The local minimum obtained with t_a is not, in the figure, a global minima, and thus a more complex algorithm is required to enable the true R.V. value, given when $t_1 = t_b$, to be found.

However, the use of a global minimizing technique may well defeat the property of computational simplicity, which is always a necessary criterion in self-tuning controllers. Moreover, unless a simple method for a particular application can be found, we must consider that the feature of self-assigning poles is only useful for low order

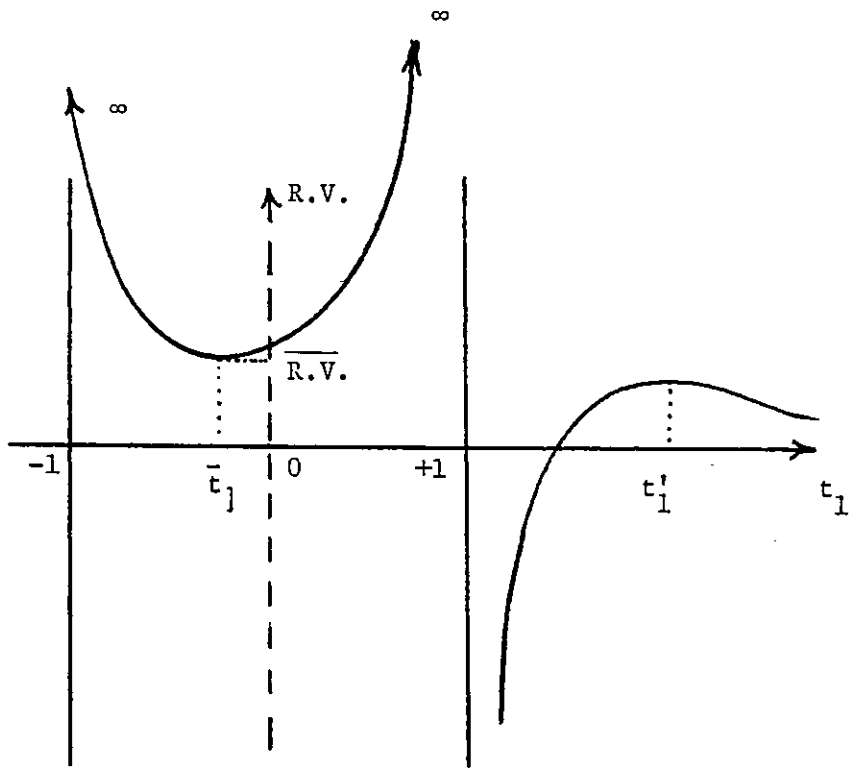


Fig. 3.1. Typical curve of R.V. against one pole, for n' = 2.

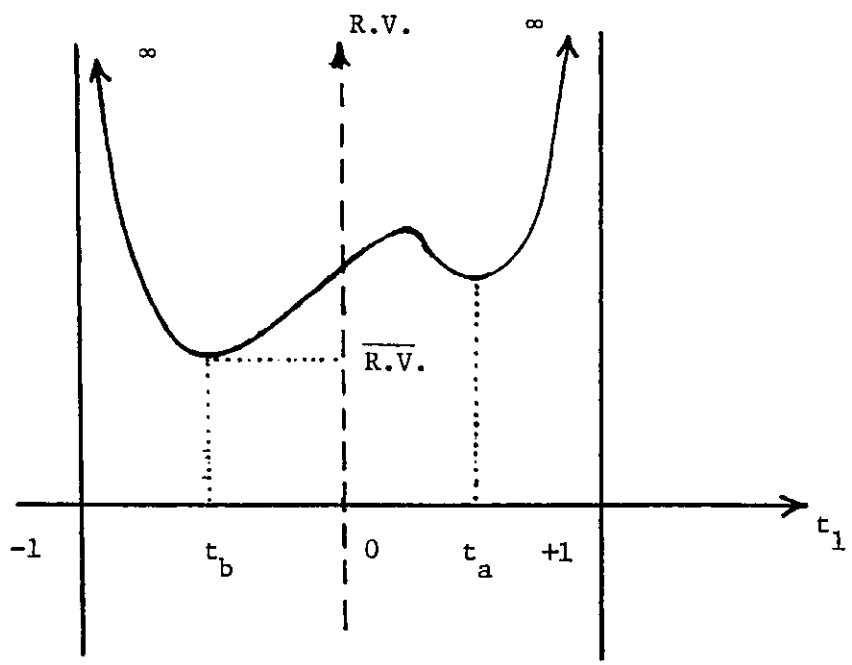


Fig. 3.2. Typical curve of R.V. against a single pole, for n' = 4.

system models. This does not limit its field of usage to a great extent, though, as it is often the case that a high order system may be successfully controlled by means of a low order model because of the characteristics attributed to self-tuners, Åström (1980).

3.3 Response to an External Input Signal

1. Introduction

It has been considered thus far that no external input is present in the system configuration, and hence the problem has remained one of reducing the effect of disturbances on the system output, i.e. output regulation to a zero mean. But now the possibility of there being a reference input, in addition to the control input will be taken into account. This reference input will be regarded as a signal which changes relatively infrequently with respect to time, as is the case with the majority of processes under controller action. It must be remembered, however, that the self-tuning philosophy requires a complete algorithm which retains a computational simplicity and is therefore suitable for recursive controller design and implementation.

2. General controller configuration

The pole assignment format allows for a simple and efficient means of set-point following, such that in the steady state the process output will be forced to comply with the reference input signal. There are, however, several schemes, dependent on how and where the external input is applied, which can lead to steady-state output following. We will therefore provide, initially, a general

scheme, and show how various alternative methods may be derived by using this as a starting point.

The overall control plan is given in Fig. 3.3, where the set-point is considered as a supplementary input, such that the actual system input becomes,

$$u(t) = F\underline{\hat{x}}(t) + Sv(t) \qquad (3.3.1)$$

where v(t) is the external, scalar, reference input.

The definition of S is dependent on the final closed-loop form, as will be seen shortly.

The estimated state vector, $\underline{\hat{x}}(t)$, is characterised by the state equations,

$$\left.\begin{aligned} &(a) \quad \underline{\hat{x}}(t+1) = P\underline{\hat{x}}(t) + Qu(t) + Re(t) \\ &(b) \quad y(t) = H\underline{\hat{x}}(t) + e(t) \end{aligned}\right\} \qquad (3.3.2)$$

Substituting for the input, u(t), in the above equations, from (a) we obtain

$$\underline{\hat{x}}(t) = z^{-1}(I-z^{-1}\bar{F})^{-1}[QSv(t) + Re(t)]$$

where $\bar{F} = P + QF$.

Thus, when (3.3.2(b)) is considered, the closed-loop equation is found to be,

$$y(t) = Hz^{-1}(I-z^{-1}\bar{F})^{-1}QSv(t) + [1 + Hz^{-1}(I-z^{-1}\bar{F})^{-1}R]e(t) \qquad (3.3.3)$$

This can be regarded as,

$$y(t) = y_v(t) + y_e(t) \qquad (3.3.4)$$

where $y_v(t)$ = set-point output.

$y_e(t)$ = Regulator output.

Such that when v(t) = 0, i.e. no external input is present, the set-point output, $y_v(t)$ is also zero, and the system output, y(t), is merely the regulator output as has been considered to be the case

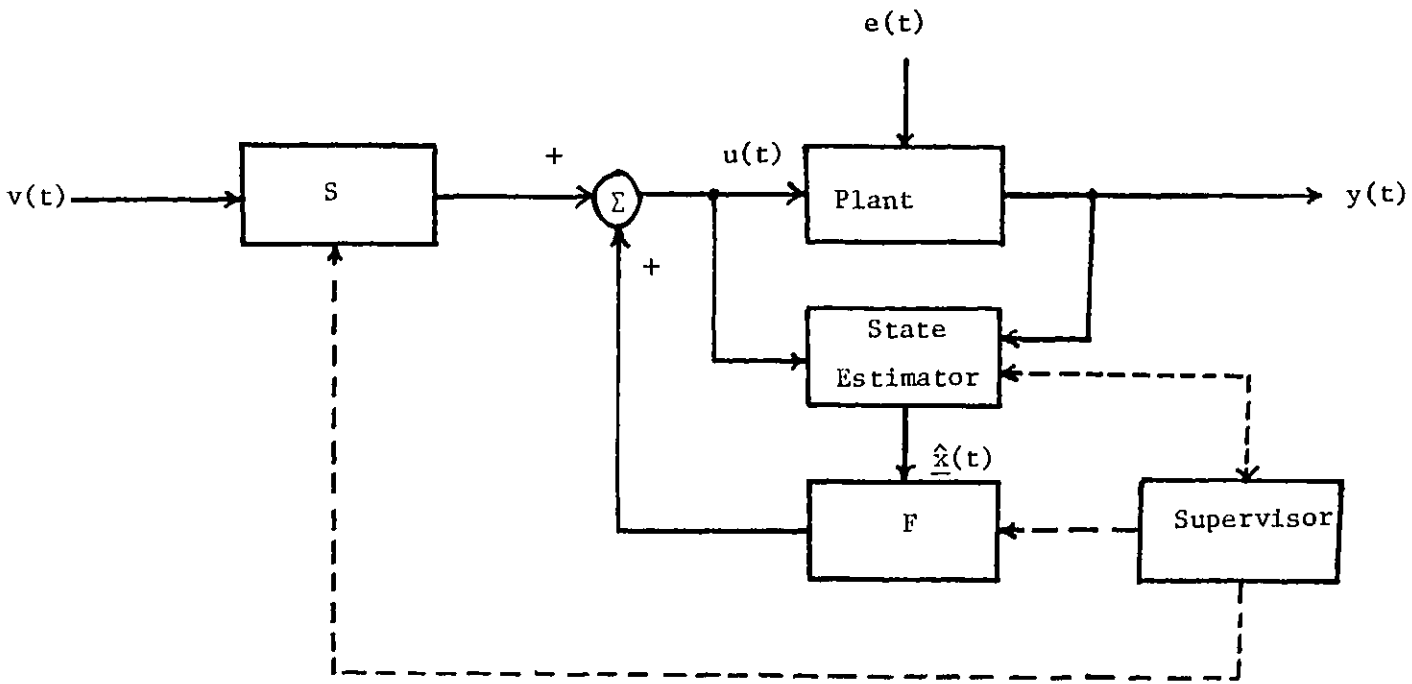


Fig. 3.3. Control of system with External Input

prior to this section. Where $v(t) \neq 0$ and is of a constant value, the variance of the system output is regarded as being with respect to a mean value of $y_v(t)$, and not a zero mean as has been the case.

The external input feedforward gain, S, does not appear in the expression for the regulator output, $y_e(t)$, and therefore the effect of the choice of S on the system output may be viewed solely by inspection of the set-point output, $y_v(t)$. This is re-written as,

$$y_v(t) = \frac{Hz^{-1}\ \mathrm{adj}(I-z^{-1}\bar{F})QS}{\mathrm{Det}(I-z^{-1}\bar{F})}\ v(t) \qquad (3.3.5)$$

where the denominator is given by the solution to the standard state space pole placement identity,

$$\mathrm{Det}(I-z^{-1}\bar{F}) = T(z^{-1}) \qquad (3.3.6)$$

$T(z^{-1})$ being the closed loop pole polynomial chosen prior to controller operation.

The numerator of (3.3.5) is simplified by the following Lemma.

Lemma 3.3 :

$$Hz^{-1}\ \mathrm{adj}(I-z^{-1}\bar{F})Q = z^{-k}B(z^{-1})$$

Proof: given in Appendix 3.1.

Therefore (3.3.5) can be considered as,

$$y_v(t) = \frac{z^{-k}B(z^{-1}).S}{T(z^{-1})}\ v(t) \qquad (3.3.7)$$

Here all the zeros of the $B(z^{-1})$ polynomial are also zeros of the closed loop transfer function between the external input and its respective output. However, to ensure zero steady-state error between

$y_v(t)$ and $v(t)$, we require that in the steady-state $y_v(t) = v(t)$, which is achieved iff $S = \bar{S}_0$, where

$$\bar{S}_0 = \frac{T(1)}{B(1)} \tag{3.3.8}$$

in which $T(1)$ and $B(1)$ are obtained by replacing z by unity in the polynomials $T(z^{-1})$ and $B(z^{-1})$ respectively.

There are, though, several possible definitions of S dependent on the specific requirements of the controller, several examples of which are given here.

Example 1: If the $B(z^{-1})$ polynomial can be defined as $B(z^{-1}) = B_s(z^{-1})B_u(z^{-1})$, where $B_s(z^{-1})$ contains all its zeros within the unit circle and $B_u(z^{-1})$ is the polynomial representing all the unstable zeros of $B(z^{-1})$.

Thus S is defined as

$$S = \bar{S}_0 \cdot S(z^{-1})/B_s(z^{-1})$$

such that S, which remains stable, replaces the stable zeros of $B(z^{-1})$ by those of $S(z^{-1})$, which may be chosen by the operator, where

$$S(z^{-1}) = s_0 + s_1 z^{-1} + \ldots + s_{n_s} z^{-n_s} \tag{3.3.9}$$

The zero steady state error property is retained by the inclusion of $\bar{S}_0$, which must be adjusted accordingly, to account for $S(1)$.

Example 2: Replacement of closed-loop poles.

The closed-loop poles may be replaced by correct choice of S, such that although the output $y_v(t)$ is affected, the original regulator problem, characterised by $y_e(t)$, remains unaltered. In this case S is,

$$S = \bar{S}_0 \cdot T(z^{-1})/S(z^{-1}) \qquad (3.3.10)$$

where S_0 is unity and $T(z^{-1})$ is known, as this has been specifically chosen. Thus $y_v(t)$, from (3.3.7), now becomes

$$y_v(t) = \frac{z^{-k}B(z^{-1}) \cdot \bar{S}_0}{S(z^{-1})} \, v(t)$$

and the closed loop response to the external input, v(t), is dependent on the newly chosen polynomial, $S(z^{-1})$. The use of this definition of S is shown to be a distinct advantage in the following example.

Example 3 : Choice of poles in a SOMI process.

Where the process under control is Single Output Multi-Input (SOMI) in nature, the overall system output may be described by,

$$y(t) = \sum_{i=1}^{p} y_{v_i}(t) + y_e(t) \qquad (3.3.11)$$

where p defines the number of external inputs present, and $y_{v_i}(t)$ is defined as,

$$y_{v_i}(t) = \frac{z^{-k}B(z^{-1})S_i}{T(z^{-1})} \, v_i(t) \qquad (3.3.12)$$

The feedforward factor for each input, S_i, can now be chosen as,

$$S_i = \bar{S}_{0_i} \cdot T(z^{-1})/S_i(z^{-1}) \qquad (3.3.13)$$

where $T(z^{-1})$ is common to all inputs, and $\bar{S}_{0_i} = S_i(1)/B(1)$. Also $S_i(z^{-1})$ is the closed loop pole polynomial required to achieve a particular system response to the i^{th} input. So, finally the regulation characteristics of the output y(t) are specified by the chosen polynomial $T(z^{-1})$, whereas the response of y(t) to each of the inputs is dependent on their respective pole polynomials, $S_i(z^{-1})$,

which are also selected prior to controller operation.

3. Self-tuning with a set-point

The complete controller algorithm, which accounts for variations in an external input signal, is a simple extension to the self-tuning format described previously. The additional computing necessary in this instance, however, is the calculation of the feedforward gain S, denoted by $\hat{S}$ when this is obtained recursively as follows. By the use of recursive least squares estimation, (3.3.7) becomes,

$$y_v(t) = \frac{\hat{B}'(z^{-1}) \cdot \hat{S}}{T(z^{-1})} \quad v(t)$$

where $\hat{S} = T(1)/\hat{B}'(1)$ to achieve zero steady state error, i.e. to normalise $y_v(t)$.

Thus $\hat{S}$ is calculated at each sample period by summing the coefficients contained in the $\hat{B}'(z^{-1})$ polynomial and dividing the total into T(1), which will have been calculated prior to start up. This means that the extra computation necessary for the incorporation of a set-point, where only a normalisation is required, amounts to (n_b+k_m-1) additions and one multiplication (division).

For convergence of $\hat{S}$ it is required that,

1) the parameters in the estimated system model converge to steady values.

2) $T(z^{-1})$ has no roots at or near z = +1.

The first of these is necessary to obtain a direct convergence of $\hat{S}$, although $\hat{S}$ will not converge to S unless $\hat{B}'(z^{-1}) \rightarrow z^{-k}B(z^{-1})$, which may not occur due to the bias in the least squares estimate. On convergence, though, $\hat{S}$ remains the correct value needed to normalise $y_v(t)$.

The second condition ensures $T(1)$ is not equal to zero, as if this were the case $\hat{S}$ would have to be set to zero, i.e. external input removed, to ensure $y_v(t) \not\to \infty$.

4. Simulated examples

The example discussed in Sec. 3.2(3) is reconsidered, where the system is described by

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

and

$$A(z^{-1}) = 1 - 0.6065\ z^{-1}$$

$$B(z^{-1}) = 0.6043 + 1.8942\ z^{-1}$$

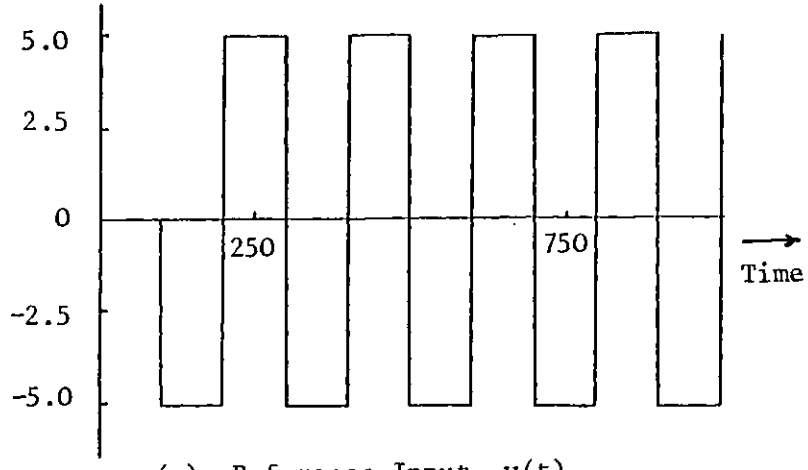
$$C(z^{-1}) = 1 - 0.4\ z^{-1} + 0.03\ z^{-2}$$

the integer time delay k being equal to unity.

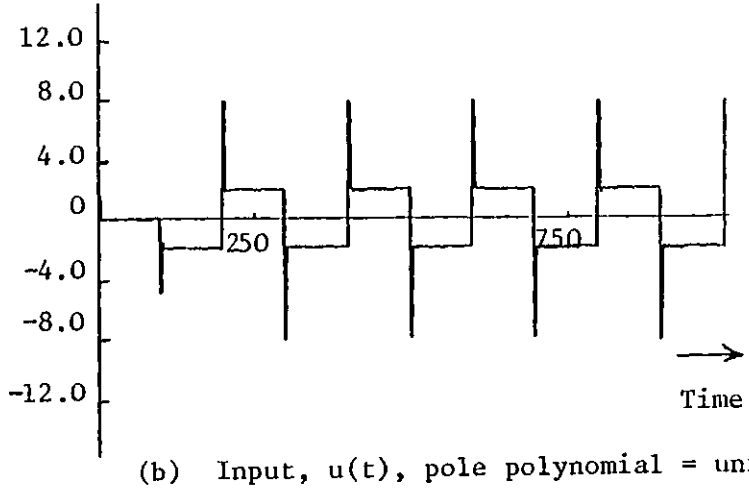
The state estimate is formulated as described in Sec. 2.3, from the estimated model parameters, and the zero-mean white noise sequence has a variance = 0.11.

The system was simulated with the controller operating over 5000 time intervals with the external input, v(t), varying from + 5.0 to − 5.0 in a square wave with a frequency of 200 samples. As each sample period is taken to be 0.5 secs., the frequency of the input is therefore 0.01 Hertz.

Figures 3.4(a) and (b) show v(t) and the system input, u(t), for the first 1000 time steps (500 secs.), Figs. 3.4(c) and (d) meanwhile show the system output, y(t), obtained when in (c) $T(z^{-1}) = 1$ and (d) $T(z^{-1})$ is obtained by the self assigning pole process discussed in Sec. 3.2. . It is noticeable that in the figures concerning the output, values are obtained as steady state limits, that are not equivalent to the reference input, v(t). This is designed to show that

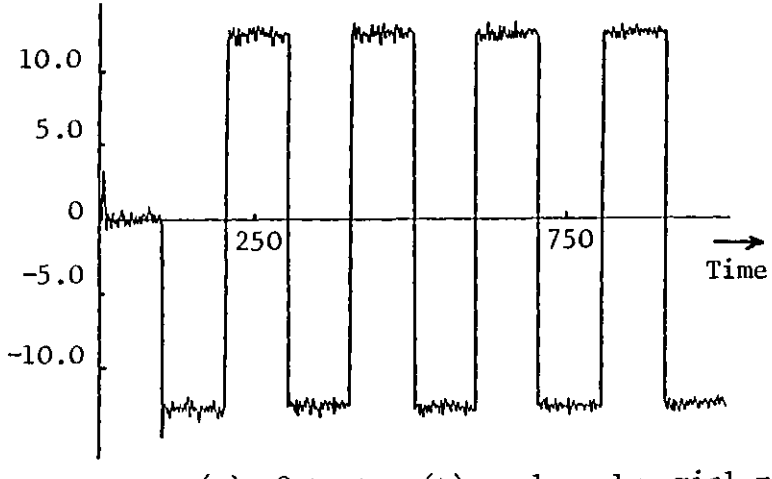


(a) Reference Input, v(t).

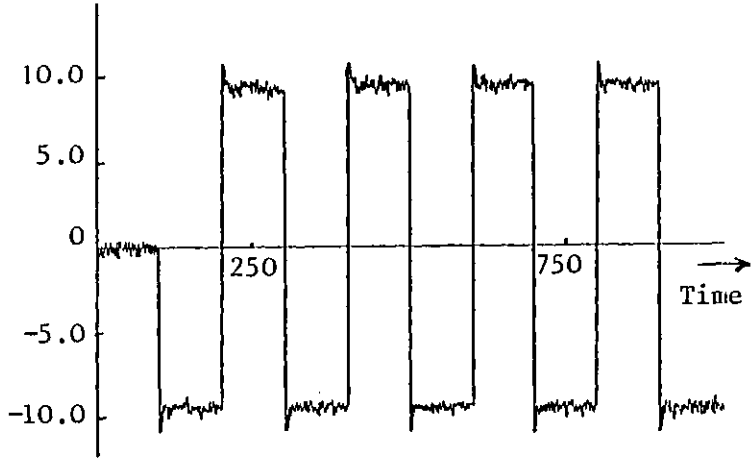


(b) Input, u(t), pole polynomial = unity.

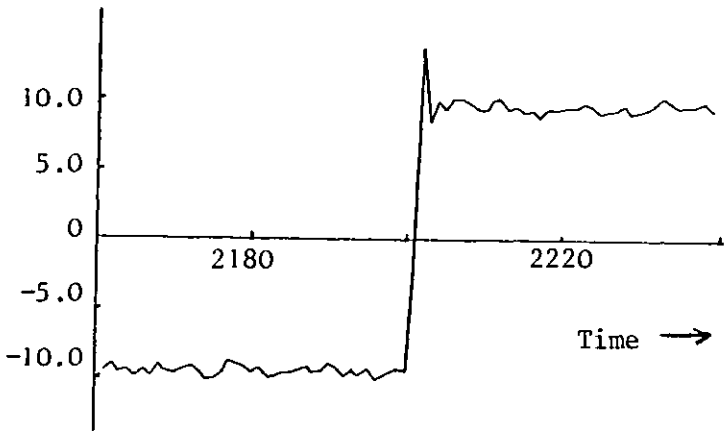
Fig. 3.4 System Response to a reference input signal.



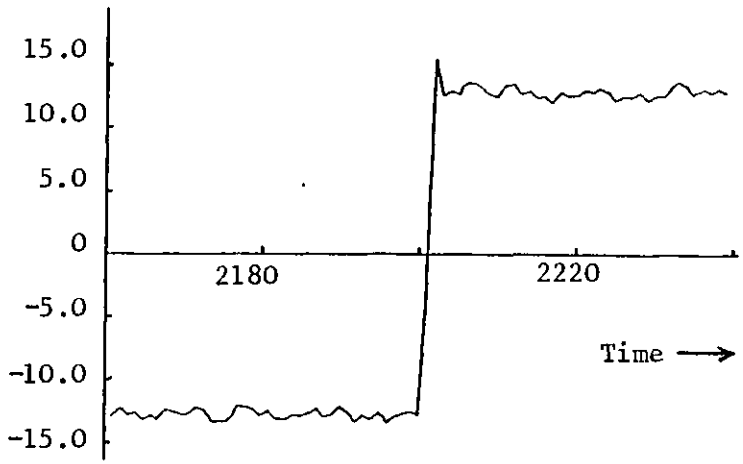
(c) Output, y(t), pole polynomial = unity.



(d) Output, y(t), one tuned pole.

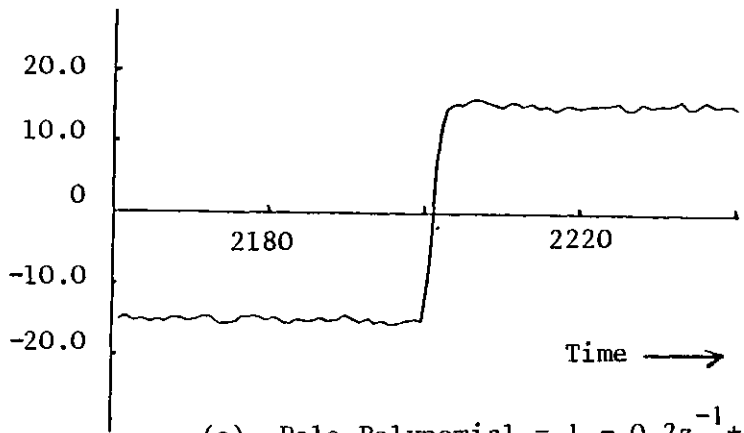


(a) Output, y(t), one tuned pole.

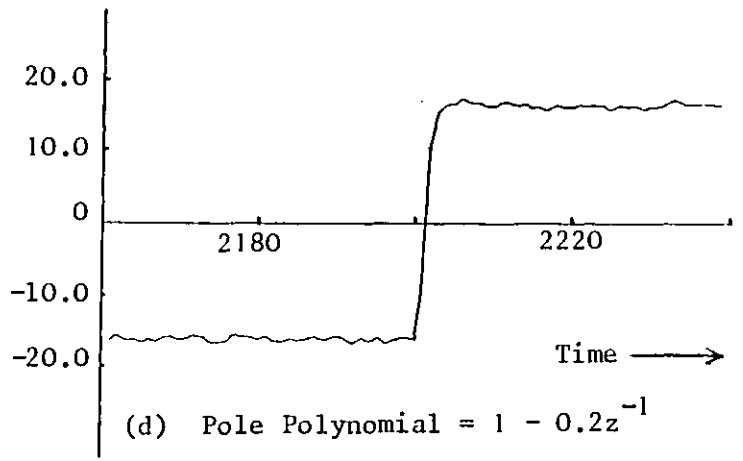


(b) Pole Polynomial = unity

Fig. 3.5 Output, y(t), with various pole positions



(c) Pole Polynomial = $1 - 0.2z^{-1} + 0.01z^{-2}$



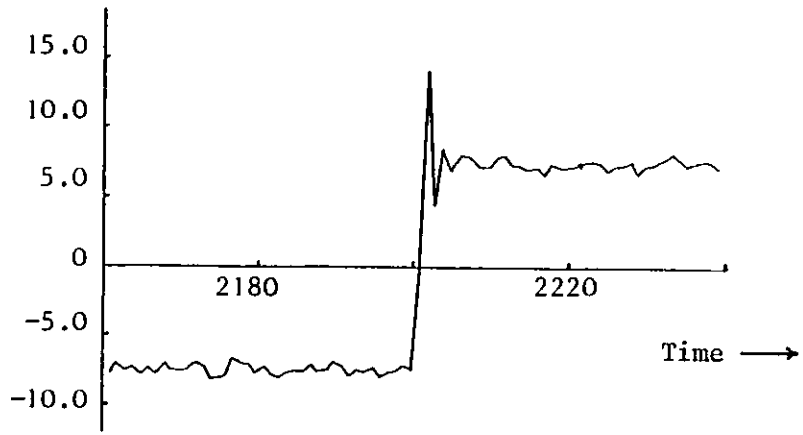
(d) Pole Polynomial = $1 - 0.2z^{-1}$

when $\hat{S} = 1$, i.e. v(t) is connected directly, such that $u(t) = F\underline{\hat{x}}(t) + v(t)$, the effect on the system output magnitude can be hazardous.

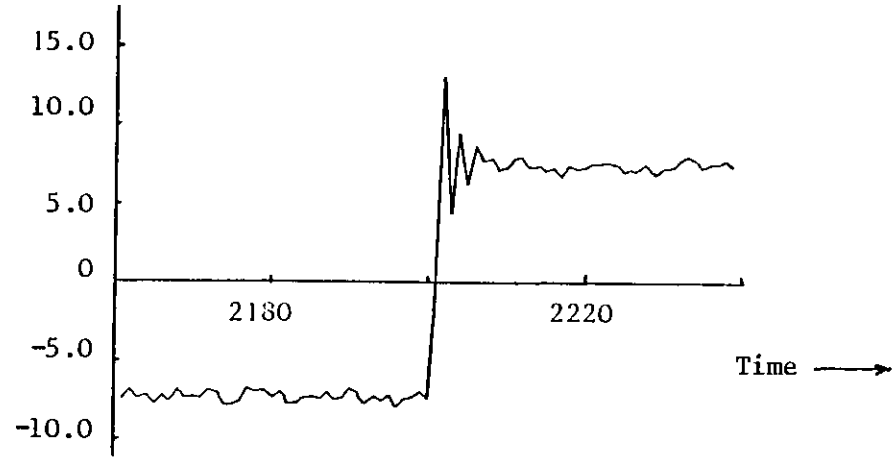
In Figs. 3.5, 3.6 and 3.7 a section of the output response signal is considered in depth over a relatively short time period, such that the transient response can be more readily studied. Figs.3.5 and 3.6 showing the effect of the choice of pole polynomial on the system output, when this has not been normalised, ranging from an overdamped response in Fig. 3.6(c) to an underdamped response in 3.6(b). With this non-normalised output it can be seen in Fig.3.6(d) that where a pole approaches z = +1, the steady-state value of y(t) becomes large in magnitude.

Fig.3.7 shows the normalised output response, and here it is observed that the regulator output, achieved in the steady-state, is of a similar magnitude, whichever pole polynomial is in operation and hence whichever feedforward gain factor is being used.

The theoretical steady state values for each pole polynomial, and their corresponding figures are given in Table 3.2, where $T(z^{-1}) = 1 + t_1 z^{-1} + t_2 z^{-2}$ is the specified closed loop pole polynomial and SAP = Self Assigning Pole.

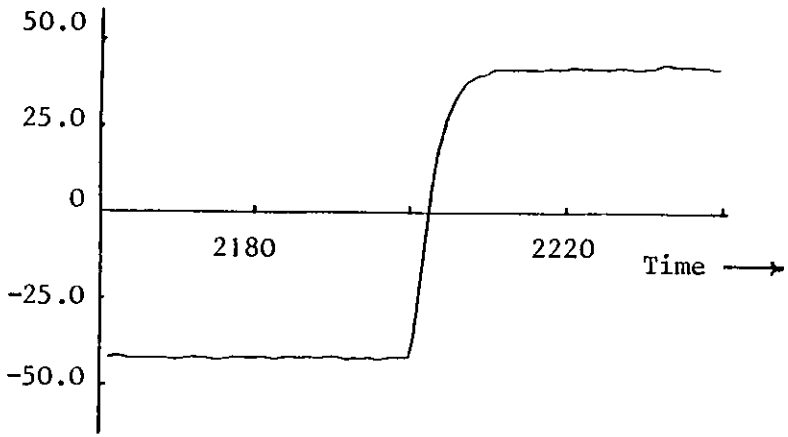


(a) Pole Polynomial = $1 + 0.6z^{-1} + 0.1z^{-2}$

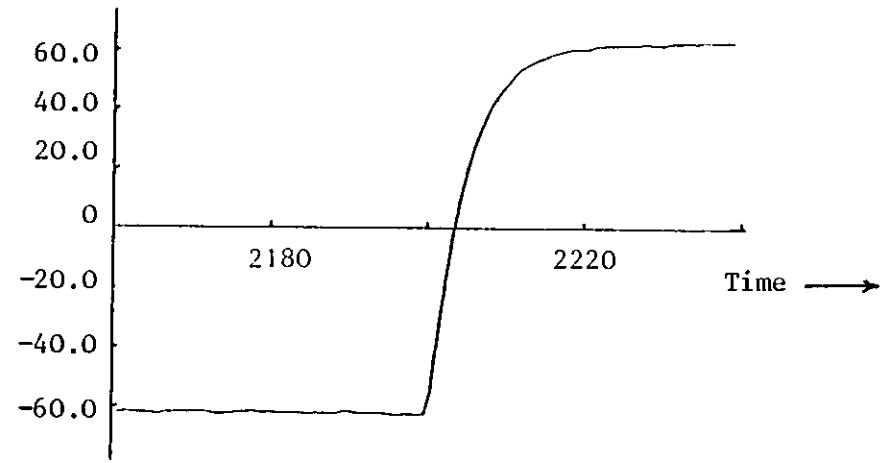


(b) Pole Polynomial = $1 + 0.6z^{-1}$

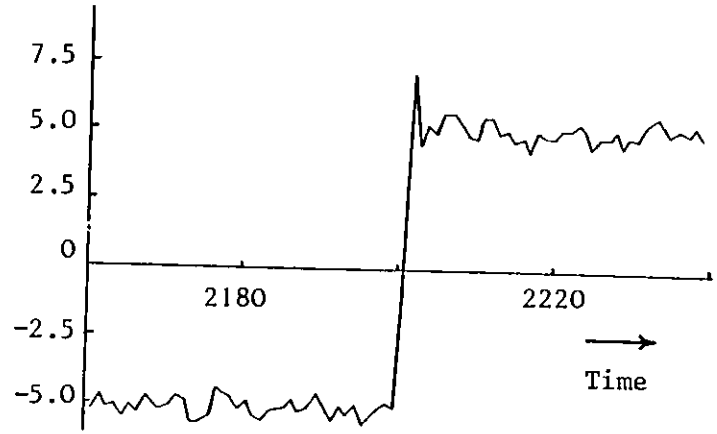
Fig. 3.6 Output, y(t), with various pole positions.



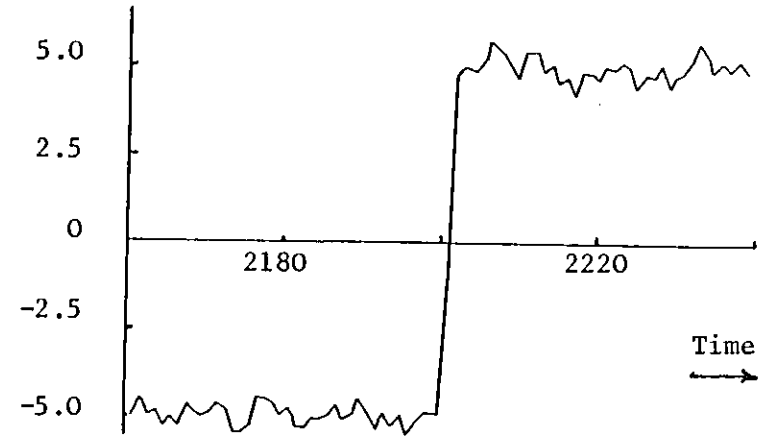
(c) Pole Polynomial = $1 - 0.8z^{-1} + 0.1z^{-2}$



(d) Pole Polynomial = $1 - 0.8z^{-1}$

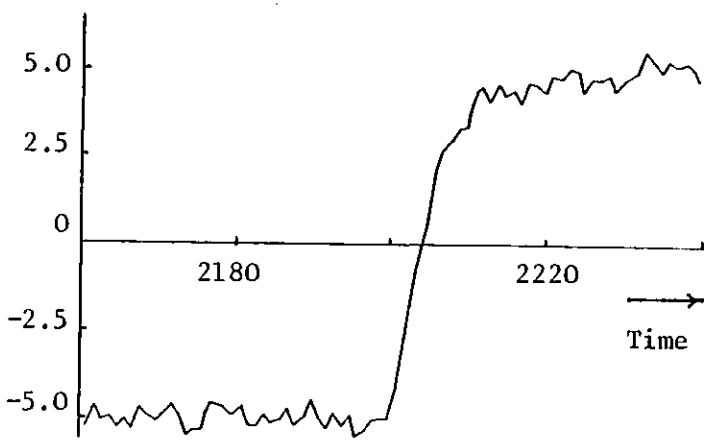


(a) One tuned pole

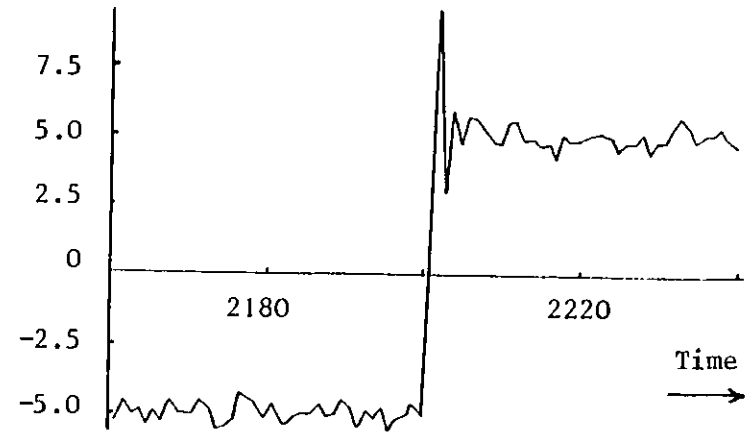


(b) Pole Polynomial = unity

Fig. 3.7. Normalized Output Signals



(c) Pole Polynomial = $1 - 0.8z^{-1}$



(d) Pole Polynomial = $1 + 0.6z^{-1} + 0.1z^{-2}$

Table 3.2

t_1	t_2	Closed Loop Poles	Steady State Output $y_v(t)$	Figs. 3.
0	0	0,0	12.4925	4(c),5(b) 7(b)
SAP	0	+0.3190,0	9.4710	4(d),5(a) 7(a)
-0.2	0.01	+0.1,+0.1	15.4228	5(c)
-0.2	0	+0.2,0	15.6156	5(d)
0.6	0.1	-0.3±j0.3148	7.3485	6(a),7(d)
0.6	0	-0.6,0	7.8078	6(b)
-0.8	0.1	+0.6449,+0.1550	41.6417	6(c)
-0.8	0	+0.8,0	62.4625	6(d),7(c)

5. Variations on the set-point definition

By considering the reference input to be connected elsewhere in the controller circuitry , variations in the basic equations obtained in Sec. 3.3(2) can be achieved. In Fig. 3.8 the set-point is summed with the state estimate, rather than with the control input, as was done in Fig. 3.3. The state estimate is an $(n_1+k)\times 1$ vector, therefore the feedforward gain S' must necessarily also be of this form. The definition is thus made that,

$$S' = [S_0,\ldots,S_{n_1+k-1}]\bar{S}_0 \qquad (3.3.14)$$

where $\bar{S}_0$ is a common factor to all S_i : $i = 0,\ldots,n_1+k-1$.

The system input becomes, in this instance,

$$u(t) = F[\underline{\hat{x}}(t) + S'v(t)] \qquad (3.3.15)$$

and by substitution into the state equations(3.3.2), the closed loop

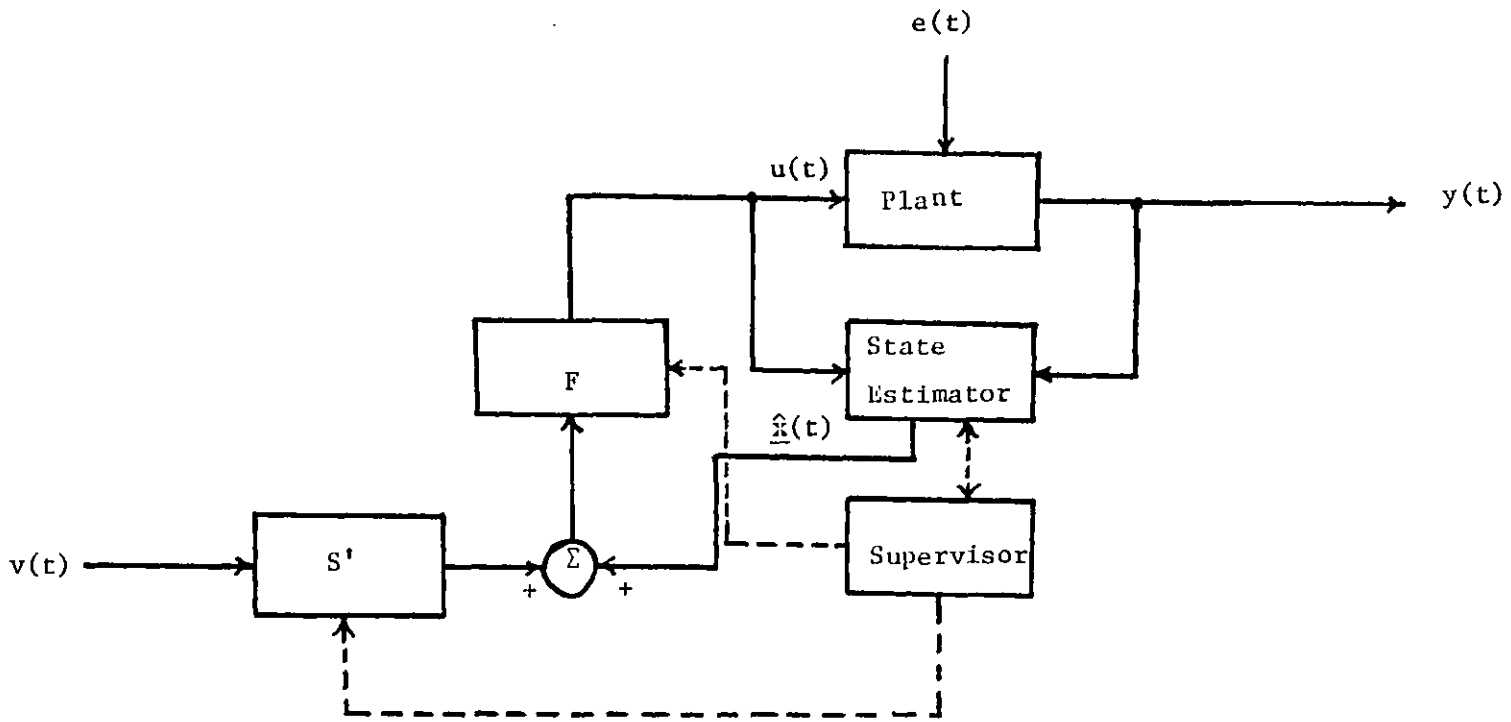


Fig. 3.8. Variation on controller design with External Input

form is given by

$$y(t) = Hz^{-1}(I-z^{-1}\bar{F})^{-1} QFS' v(t) + [I + Hz^{-1}(I-z^{-1}\bar{F})^{-1} R]e(t) \quad (3.3.16)$$

From (3.3.4), $y_e(t)$ is of the same form as in the case previously considered, although now $y_v(t)$ contains a further factor, F, in its transfer function. Hence iff

$$\left.\begin{array}{ll} \text{(a)} & \sum_{i=1}^{n_1+k} f_i S_{i-1} = 1 \\ \text{and} \quad \text{(b)} & \bar{S}_0 = T(1)/B(1) \end{array}\right\} \quad (3.3.17)$$

the two methods arrive at identical, normalised, closed loop transfer functions. The simplest solution to (3.3.17(a)) is obtained by setting $S_{j-1} = f_j^{-1}$ and $\{S_{i-1} = 0;\ i \neq j,\ i = 1,\ldots,n_1+k\}$. This method can, in fact, be regarded as a special case of the general method described in Sec. 3.3(2), where the feedforward gain $S = FS'$.

However, the effect of using the form in Fig. 3.8 gives rise to what can be considered as a modified state estimation scheme which is fed back to provide the next control input, (3.3.15). As $\hat{\underline{x}}_i(t)$ are polynomials in z^{-1}, so S_{i-1} can also be regarded as polynomials of the same type, and this leads to a wide choice of set-point transfer functions

6. Comparison with polynomial case

For polynomial based self-tuners the control input is assumed to be of the form,

$$D(z^{-1})u(t) = G(z^{-1})y(t) \quad (3.3.18)$$

where $D(z^{-1})$ and $G(z^{-1})$ are of degree (n_b+k-1) and (n_a-1) respectively. Various representations of the reference input are shown in Figs. 3.10-3.14, where y(t) is considered to be a direct feedback from the system output. In all cases the transfer function of the feedback from y(t) to

u(t) is of the form (3.3.18), the differences between the diagrams arising merely from the nature of the introduction of v(t), the reference input. The set-point will, therefore, have no effect on the transfer function between the disturbance, e(t), and the output, y(t). Thus $y_e(t)$ in (3.3.4) will, as in Sec. 3.3(2), remain unchanged, and for the purpose of this exercise will be neglected, and that part of the output, $y_v(t)$, determined by v(t) will be considered. In the deterministic case, therefore, the open-loop transfer function is given by

$$A(z^{-1})y(t) = z^{-k}B(z^{-1})u(t) \qquad (3.3.18)$$

Using the control input of Fig.3.10, given by

$$D(z^{-1})u(t) = G(z^{-1})y(t) + Sv(t) \qquad (3.3.19)$$

The closed-loop form is obtained as,

$$[A(z^{-1})D(z^{-1}) - z^{-k}B(z^{-1})G(z^{-1})]y(t) = z^{-k}B(z^{-1})Sv(t)$$

... (3.3.20)

But in the polynomial case, the identity used to obtain $D(z^{-1})$ and $G(z^{-1})$ was constructed as,

$$A(z^{-1})D(z^{-1}) - z^{-k}B(z^{-1})G(z^{-1}) = C(z^{-1})T(z^{-1}) \qquad (3.3.21)$$

Hence, if no pole cancellation is required, S must be chosen as,

$$S = T(1)C(1)/B(1) \qquad (3.3.22)$$

to provide a normalised output. But this means that we need to know the coefficients of the $C(z^{-1})$ polynomial, a requirement additional to those needed in the state-space form. In Wellstead and Sanoff (1981) the $C(z^{-1})$ polynomial is estimated using the extended least squares method, whereas in Clarke and Gawthrop (1975) it is considered that $S = S(z^{-1})$ and the parameters of this new polynomial are included in the estimation scheme.

In the polynomial method, Fig.3.10, of incorporating a set-point, therefore, the computational effort is much increased to allow for a normalisation process to occur.

The control input, Fig.3.11, may be written as

$$D(z^{-1})u(t) = D(z^{-1})Sv(t) + G(z^{-1})y(t) + v(t) \qquad (3.3.23)$$

which leads to the closed loop equation

$$[A(z^{-1})D(z^{-1}) - z^{-k}B(z^{-1})G(z^{-1})]y(t) = z^{-k}B(z^{-1})[1 + D(z^{-1})S]v(t) \qquad (3.3.24)$$

Hence, when the identity (3.3.21) is used, the problem of the $C(z^{-1})$ polynomial is once more apparent, and further, the transient response to a change in reference input may be affected in a retrograde manner.

The value of S required for output normalisation is given now as,

$$S = [C(1)T(1) - B(1)]/B(1)D(1) \qquad (3.3.25)$$

which requires more calculations than has been the case.

The format defined by Fig.3.12 provides a control input.

$$D(z^{-1})u(t) = [D(z^{-1})S+G(z^{-1})]v(t) + G(z^{-1})y(t) \qquad (3.3.26)$$

with a respective closed loop transfer function of,

$$[A(z^{-1})D(z^{-1}) - z^{-k}B(z^{-1})G(z^{-1})]y(t) = z^{-k}B(z^{-1})[D(z^{-1})S + G(z^{-1})]u(t) \qquad (3.3.27)$$

A special case now arises when $A(z^{-1})$ contains a pole at $z = +1$, i.e. the polynomial contains a $(1-z^{-1})$ term,then selecting S as being zero, normalisation will occur, Wellstead et al (1979b). A similar event arises in the controller of Fig. 3.13 where the closed loop equation is

$$[A(z^{-1})D(z^{-1}) - z^{-k}B(z^{-1})G(z^{-1})]y(t) = z^{-k}B(z^{-1})G(z^{-1})Sv(t) \qquad (3.3.28)$$

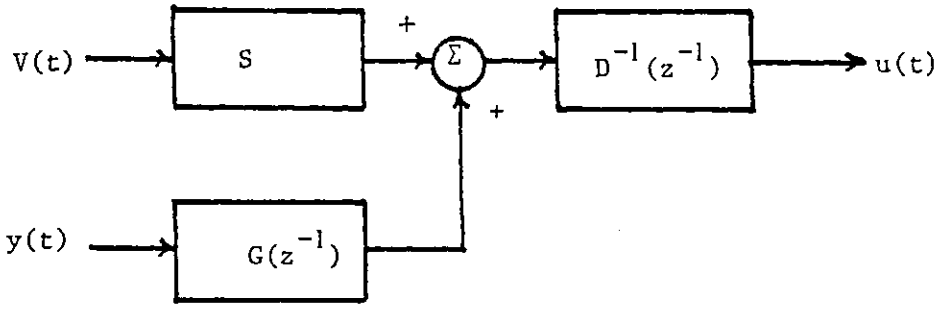


Fig. 3.10.

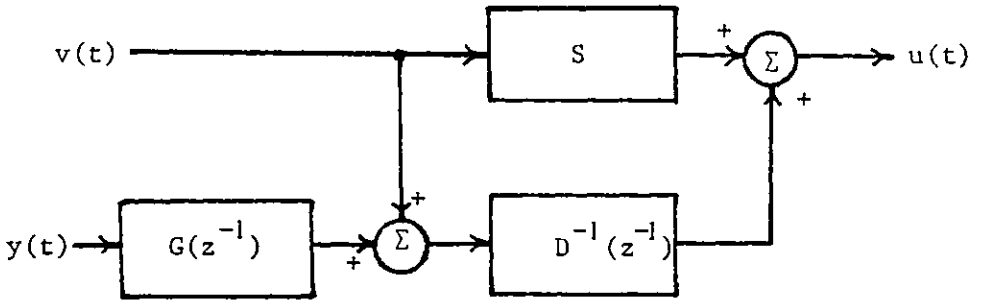


Fig. 3.11.

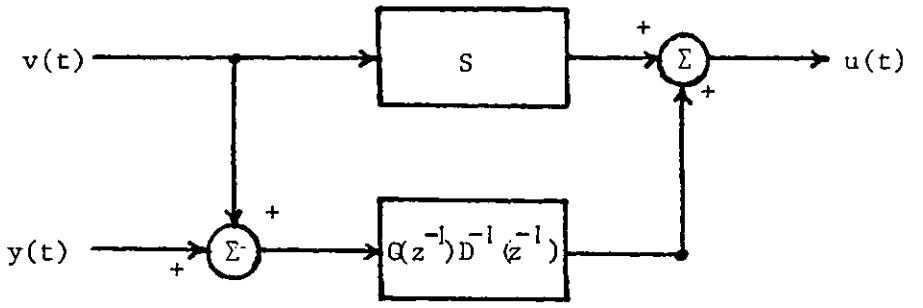


Fig. 3.12

Here, when an integrator term is contained in the $A(z^{-1})$ polynomial, such that $A(1) = 0$, by setting S to unity zero steady state error is obtained between reference input and output, $y_v(t)$. Otherwise, setting S to be

$$S = \bar{S}_0/G(z^{-1}) \tag{3.3.29}$$

where $$\bar{S}_0 = C(1)T(1)/B(1)G(1) \tag{3.3.30}$$

will remove the transient effects caused by $G(z^{-1})$.

The final configuration, Fig 3.14, gives

$$D(z^{-1})u(t) = D(z^{-1})Sv(t) + G(z^{-1})y(t) \tag{3.3.31}$$

which provides a closed loop form,

$$[A(z^{-1})D(z^{-1}) - z^{-k}B(z^{-1})G(z^{-1})]y(t) = z^{-k}B(z^{-1})D(z^{-1})Sv(t) \tag{3.3.32}$$

and from this normalisation is carried out by using

$$\bar{S}_0 = C(1)T(1)/B(1)D(1) \tag{3.3.33}$$

All the polynomial controls mentioned, therefore, provide zero steady-state error by use of $C(1)$. The polynomial form equivalent to that used in the state-space method of Sec. 3.3(2), can, however, be derived from the control input,

$$u(t) = -\frac{D'(z^{-1})}{C(z^{-1})}\,u(t) + \frac{G(z^{-1})}{C(z^{-1})}\,y(t) + Sv(t) \tag{3.3.34}$$

where $$D'(z^{-1}) = D(z^{-1}) - C(z^{-1}) \tag{3.3.35}$$

which is obtained from (3.3.1), see Appendix 3.2, where it is shown that the final closed loop equation is

$$[A(z^{-1})D(z^{-1}) - z^{-k}B(z^{-1})G(z^{-1})]y(t) = z^{-k}B(z^{-1})C(z^{-1})Sv(t) \tag{3.3.36}$$

and if the identity (3.3.21) is used, this becomes

$$T(z^{-1})y(t) = z^{-k}B(z^{-1})Sv(t)$$

which is the final form given by the state space control law (3.3.7).

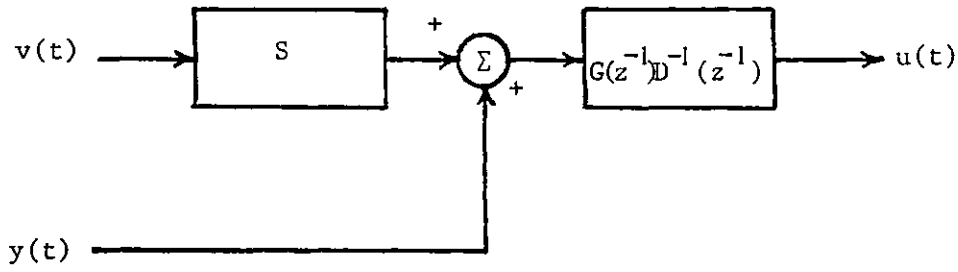


Fig. 3.13

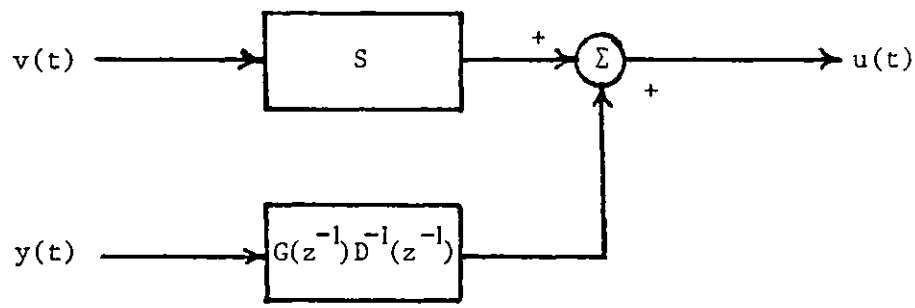


Fig. 3.14

Unfortunately, unless $C(z^{-1})$ is estimated by means of an enlarged estimation scheme, such as extended least squares, the control input (3.3.34) cannot be arrived at by means of a polynomial approach. By setting $C(z^{-1}) = 1$, as in recursive least squares, the control input returns to that of Fig.3.10, which resulted in $C(z^{-1})$ appearing in the closed loop transfer function.

7. Regulator with non-zero set-point

In the deterministic case, the system can be modelled by (3.3.18), which has a state space equivalent as,

$$\left.\begin{aligned} &(a) \quad \underline{x}(t+1) = P\underline{x}(t) + Qu(t) \\ &(b) \quad y(t) = H\underline{x}(t) \end{aligned}\right\} \qquad (3.3.37)$$

when a set-point input is applied it is required that the output of the system will be given by $\bar{y}$, and hence the error between the actual output and this reference value can be described as

$$Wy(t) = \bar{y} - y(t)$$

By similar reasoning, there is also a system input, $\bar{u}$, which will achieve the desired output, and thus we have also,

$$Wu(t) = \bar{u} - u(t)$$

and $$\underline{W}x(t) = \bar{\underline{x}} - \underline{x}(t)$$

where $\bar{\underline{x}}$ is the required state vector, from which

$$\bar{\underline{x}} = P\bar{\underline{x}} + Q\bar{u}$$

and $$\bar{y} = H\bar{\underline{x}}$$

as $\bar{\underline{x}}(t) = \bar{\underline{x}}(t+1)$, etc.

Thus by substituting for $y(t)$, $u(t)$ and $\underline{x}(t)$ in (3.3.37)the state equations may be written as

$$\left.\begin{aligned} &(a) \quad \underline{W}x(t+1) = P\ \underline{W}x(t) + Q\ Wu(t) \\ &(b) \quad Wy(t) = H\ \underline{W}x(t) \end{aligned}\right\} \tag{3.3.38}$$

Similarily, defining the control input from these 'new' state equations as

$$Wu(t) = F\ \underline{W}x(t) \tag{3.3.39}$$

A cost function associated with (3.3.38) can be formulated as

$$\emptyset(Wu) = \sum_{t=t}^{\infty} \{Wy^T(t+1)S_3Wy(t+1) + Wu^T(t)S_2Wu(t)\} \tag{3.3.40}$$

such that, when this is minimized by correct choice of Wu(t), the system is moved optimally from any initial condition to the required reference input (set-point).

By substituting for Wu(t) and $\underline{W}x(t)$ in (3.3.39), the actual system control input becomes,

$$u(t) = F\underline{x}(t) - F\bar{\underline{x}} + \bar{u} \tag{3.3.41}$$

where the constant values $\bar{u}$ and $\bar{\underline{x}}$ are necessary to retain the system at the set-point, and hence we need to find these values to provide an appropriate controller action. If the control input (3.3.41) is now coupled with the original state equations (3.3.37), the overall closed loop equation is obtained as,

$$y(t) = H(I-z^{-1}\bar{F})^{-1}\ Q\ z^{-1}\{\bar{u}-F\bar{\underline{x}}\} \tag{3.3.42}$$

where $$\bar{F} = P + QF \tag{3.3.43}$$

From (3.3.42) we define

$$G_c(z^{-1}) = H(I-z^{-1}\bar{F})^{-1}\ Q\ z^{-1} \tag{3.3.44}$$

Then zero error will occur in the steady state iff

$$\{\bar{u}-F\bar{\underline{x}}\} = G_c^{-1}(1)\bar{y} \tag{3.3.45}$$

where $\bar{y}$ is the reference output value.

The overall control input to the system therefore becomes, from (3.3.41),

$$u(t) = F\underline{x}(t) + G_c^{-1}(1)\bar{y} \qquad (3.3.46)$$

and this is identical, in form, to the control input used in (3.3.1), where $S = G_c^{-1}(1)$ and $v(t) = \bar{y}$.

3.4 Concluding Remarks

In this chapter the self-tuning property has been shown to apply to the state space self-tuners developed in Sec. 2.5, and further modes of operation of this type of self-tuning controller have been considered.

Where a robust controller is required specifically for the purpose of minimizing the variance of the system 's output signal, the self-assigning pole scheme reduces the discrepancy in the value of output variance apparent when the pole placement method is compared with minimum-variance strategies. This is of great benefit, as, if the closed loop poles are chosen, to be constant numerical values, prior to controller operation, certain pole positions can cause higher output variance values. This can be seen in Table 3.1.

In sec. 3.3 an external input was included in the state space format. When this is considered as a set-point which varies relatively infrequently, the computation necessary to ensure that the error between the desired output and the actual output is minimized in the steady state, is negligible. A contributory factor to this is the independence of the transfer functions dependent on disturbance and set-point respectively.

However, in the state space method discussed, the $C(z^{-1})$ polynomial is not apparent in the closed loop form of either transfer function, a characteristic not attributable to most polynomial type tuners, where no cancellation of this particular polynomial occurs in the set-point case.

A common factor between the different tuners, though, is the appearance of the system open loop polynomial, $B(z^{-1})$, in the numerator of the set-point transfer function. This means that if the process is non-minimum phase, this property will be retained in the closed loop form. Any attempt to cancel the unstable zeros of $B(z^{-1})$, by correct choice of reference input feedforward characteristics, will result in unstable poles between the reference input and the final control input, and these are not desirable. However, where a system contains an inherent direct gain factor, this may be considered as a scaling of all parameters in the $B(z^{-1})$ polynomial, and hence will be compensated for by an appropriate division, on the same scale, in the set-point feedforward gain, thus ensuring that the steady state following characteristics of the tuner are not impaired.

CHAPTER 4

EXTENSIONS TO THE FORMULATION

Self-tuning controllers have been introduced and their application by means of a state space framework discussed. This chapter is used to extend the basic assumptions made, by way of building on special properties, covered in the last chapter, attributable to these state space self-tuners.

In Sec. 4.1, linear output feedback (ℓ.o.f.) is combined with the original state feedback, by making use of the extra degree of freedom allowable from the self-tuning property. As the parameter associated with the ℓ.o.f. cannot be calculated, without further alterations being made, from the pole placement criterion, it can either be selected apriori or tuned on line for optimization purposes, the pole placement being carried out by the state feedback parameters alone. Because of problems arising due to matrix, rather than scalar variable, multiplication techniques, self-tuning for multivariable systems has, to the present, only received a limited coverage. Some of these problems are discussed in Sec. 4.2, where the use of multivariable state space self-tuners is considered. Some of the explanation is, however, carried out by means of the equivalent polynomial form for ease of explanation. Section 4.3. briefly describes self-tuning control of non-linear processes, although as this may, to a certain extent, be regarded as an incorrect choice of model for the estimation procedure, the effects of under and over specified models are reviewed in Sec. 4.4.

4.1 State Space Self-Tuning With Linear Output Feedback

1. Introduction

The majority of self-tuning techniques are based on either pole placement or optimization control objectives. In this section a self-tuner is proposed, whereby the control input is dependent not only on state feedback but also linear output feedback (ℓ.o.f.), and by this means a pole placement scheme is augmented with optimal features.

In the state space approach considered thus far, the reconstruction of the state vector is obtained by using information from measurements of the system's input and output signals. The optimal reconstruction employed, however, does not include the present value of output variable, y(t), although this is available. Therefore it will be considered that this output signal is fedback in combination with the usual state feedback, and hence will have a further parameter associated with it. This new parameter can then either be selected off line, as in the case of the required closed loop poles, or be tuned on line, Warwick and Westcott (1982), as is done with the state feedback parameters. Both cases will be looked at, and their effect on the closed loop zeros of the overall transfer function will be investigated.

The application of an external set-point input has been discussed in Sec. 3.3, and here the effect that the additional feedback has on the set-point transfer function is, therefore, included in subsection 3. Finally in subs. 6 numerous simulations are carried out to show the complete self-tuner operating on both a nonminimum phase process, and a process where the integral time delay can vary. The pole

polynomial is also altered between simulations in order to show its effect on the linear output feedback parameter.

2. The application of linear output feedback

Let the unknown system be described in the state space representation as ,

$$\left.\begin{aligned} &(a) \quad \underline{x}(t+1) = P\underline{x}(t) + Qu(t) + Re(t) \\ &(b) \quad y(t) \quad = H\underline{x}(t) + e(t) \end{aligned}\right\} \qquad (4.1.1)$$

where the matrices P, Q, R and H are defined in Sec. 2.2(2). The estimate of the state vector is, therefore, found to be,

$$\underline{\hat{x}}(t) = z^{-1}[I-z^{-1}\bar{P}]^{-1}\{Qu(t) + Ry(t)\} \qquad (4.1.2)$$

where $\bar{P} = P - RH$.

It can be seen that due to the z^{-1} term on the right hand side of (4.1.2), the present value of output variable, y(t), does not appear in the state estimate. The opportunity therefore arises to set the control input, u(t), as a combination of both state and output feedback. It was shown in Warwick (1981a) that by modifying the state along with the addition of a scalar multiple of the present output signal, the control equation of Wellstead et al (1979b) could be achieved. But in Sec. 3.1, both this latter case, and the procedure described in Sec. 2.5(1) using solely state feedback, were shown to be special cases of a more generalised approach. We now wish to make further use of this generalised case to improve on the system response to the controller used. Thus the control input is now defined as,

$$u(t) = F\underline{\hat{x}}(t) + Gy(t) \qquad (4.1.3)$$

where G is a scalar linear output feedback parameter. On substitution of (4.1.3) into the state equations (4.1.1), based on the estimated state form, the following expression for the closed loop form is obtained.

$$[1-Hz^{-1}(I-z^{-1}\bar{F})^{-1}QG]y(t) = [1 + Hz^{-1}(I-z^{-1}\bar{F})^{-1}R]e(t) \qquad (4.1.4)$$

where $\bar{F} = P + QF$.

In this instance, it is still required that the closed loop pole polynomial obtained from (4.1.4) be calculated such that its constituent parameters are equal to those found in the specified pole polynomial, $T(z^{-1})$, which is defined in (3.1.7). Hence these parameters are obtained from,

$$\det(I-z^{-1}\bar{F}) - Hz^{-1}\text{adj}(I-z^{-1}\bar{F})QG = T(z^{-1}) \qquad (4.1.5)$$

However, if (n_1+k) is the dimension of the state space, where k is the integer part of the system time delay and $n_1 = \max\{n_a,n_b,n_c\}$ then (4.1.5) leads to (n_1+k) equations with (n_1+k+1) unknowns, the 'extra' unknown parameter being G. If G is specified prior to tuner operation, though, this problem is eliminated, an example of which is the use of state feedback only, by setting G to zero.

Equation (4.1.5) is easily simplified for calculation purposes by use of the expression

$$Hz^{-1}\text{adj}(I-z^{-1}\bar{F})Q = z^{-k}B(z^{-1}) \qquad (4.1.6)$$

where equality was shown in Lemma 3.3, and the polynomial $B(z^{-1})$ is defined in the system CARMA model (3.1.1).

The state feedback parameters $f_1,\ldots,f_{n_1+k}$ contained in $\bar{F}$ may then be calculated from the equation,

$$\det(I-z^{-1}\bar{F}) - z^{-k}B(z^{-1})G = T(z^{-1}) \qquad (4.1.7)$$

by applying the state feedback of (4.1.3), with G specifically chosen. However, before considering the reasons for a particular choice of parameter G, the self-tuning aspects of the algorithm will be discussed.

The feedback terms obtained from (4.1.7) are the values calculated with regard to the true system parameters, whereas the control input actually entails the use of the parameters found in the estimated system model employed in the recursive least squares estimation procedure. This is given by,

$$\hat{A}(z^{-1})y(t) = \hat{B}'(z^{-1})u(t) + \varepsilon(t) \tag{4.1.8}$$

where the polynomials are defined in (3.1.10), and this is represented in its equivalent state space form as,

$$\left.\begin{aligned} &(a) \quad \underline{\tilde{x}}(t+1) = \hat{P}\underline{\tilde{x}}(t) + \hat{Q}u(t) + \hat{R}\varepsilon(t) \\ &(b) \quad y(t) = H\underline{\tilde{x}}(t) + \varepsilon(t) \end{aligned}\right\} \tag{4.1.9}$$

the matrices $\hat{P}$, $\hat{Q}$ and $\hat{R}$ containing the estimated model parameters being defined in (3.1.42).

By a similar definition to that of (4.1.3), the actual control input provided is given as,

$$u(t) = \hat{\hat{F}}\underline{\hat{\tilde{x}}}(t) + Gy(t) \tag{4.1.10}$$

where, from (4.1.2),

$$\underline{\hat{\tilde{x}}}(t) = z^{-1}[I - z^{-1}\hat{\hat{P}}]^{-1}\{\hat{Q}u(t) + \hat{R}y(t)\} \tag{4.1.11}$$

$$\hat{\hat{P}} = \hat{P} - \hat{R}H$$

and $$\hat{F} = [\hat{f}_1, \ldots, \hat{f}_{n'}]$$

in which n' is the dimension of the state space.

The control input (4.1.10) can now be substituted into the state equations (4.1.9) to obtain the closed loop form,

$$[1 - Hz^{-1}(I - z^{-1}\hat{\hat{F}})^{-1}\hat{Q}G]y(t) = [1 + Hz^{-1}(I - z^{-1}\hat{\hat{F}})^{-1}\hat{R}]\varepsilon(t) \tag{4.1.12}$$

where $$\hat{\hat{F}} = \hat{P} + \hat{Q}\hat{F}.$$

The state feedback factors can, therefore, be evaluated recursively, at each sampling instant, from

$$\det(I-z^{-1}\hat{\underline{\underline{F}}}) - Hz^{-1}\mathrm{adj}(I-z^{-1}\hat{\underline{\underline{F}}})\hat{Q}G = T(z^{-1}) \tag{4.1.13}$$

which, by use of Lemma 3.3, can be rewritten as

$$\det(I-z^{-1}\hat{\underline{\underline{F}}}) - \hat{B}'(z^{-1})G = T(z^{-1}) \tag{4.1.14}$$

The numerator of (4.1.12) being found to be

$$\left.\begin{array}{ll} \text{(a)} & \det(I-z^{-1}\hat{\underline{\underline{F}}}) + Hz^{-1}\mathrm{adj}(I-z^{-1}\hat{\underline{\underline{F}}})\hat{R} \\ \text{or} \quad \text{(b)} & T(z^{-1}) + \hat{B}'(z^{-1})G + Hz^{-1}\mathrm{adj}(I-z^{-1}\hat{\underline{\underline{F}}})\hat{R} \end{array}\right\} \tag{4.1.15}$$

by use of (4.1.14).

Hence the linear output feedback parameter, G, will affect the closed loop zeros, such that by defining

$$Hz^{-1}\mathrm{adj}(I-z^{-1}\hat{\underline{\underline{F}}})\hat{R} = 1 + \hat{\lambda}_1 z^{-1} + \hat{\lambda}_2 z^{-2} + \ldots + \hat{\lambda}_{n'} z^{-n'} \tag{4.1.16}$$

the numerator of (4.1.12) is obtained via (4.1.15b) as

$$1 + z^{-1}(t_1+\hat{b}_0 G+\hat{\lambda}_1) + z^{-2}(t_2+\hat{b}_1 G+\hat{\lambda}_2) + \ldots + z^{-n'}(t_{n'}+\hat{b}_{n'-1}G+\hat{\lambda}_{n'}) \tag{4.1.17}$$

This means that although G was specified such that a unique solution to (4.1.14) can be obtained, the value of G will affect the position of the closed loop zeros, and therefore the variance of the output signal. No zero cancellation will occur unless the n'^{th} term of (4.1.17) is set equal to zero, and as $\hat{b}_{n'-1}$ and $\hat{\lambda}_{n'}$ are quantities which vary, due to parameter uncertainty in the estimation process, an exact cancellation is only a purely theoretical possibility. The actual choice of value for G remains, at this stage, a completely arbitrary selection, dependent on a required function, examples of which are pure state feedback ($G = 0$), noise filtering or to provide a bias on closed loop zeros. For any specific choice of G, it was shown in Sec.3.1 that the self-tuning property is not impaired, and

thus $\varepsilon(t) \to e(t)$ as $t \to \infty$

3. Steady-state set-point following

When the external input applied to the system is no longer equal to zero, the regulation problem becomes one of controlling the variance of the output signal about a reference mean rather than a zero mean. This was considered in detail in Sec. 3.3 with regard to the state space controller using purely state feedback. In Åström and Wittenmark (1980) a deterministic system was employed, whereas when changes in the external input occur relatively infrequently in a stochastic environment, Wellstead et al (1979b), by use of an integrator in the feedback path and by correct choice of feedforward gain, removed problems associated with the $C(z^{-1})$ polynomial mentioned in Sec. 3.4. However, this problem does not occur in the state space controller and thus the same approach is taken here as was done in Sec. 3.3. The control input, with linear output feedback included, may be described as

$$u(t) = F\underline{\hat{x}}(t) + Gy(t) + Sv(t) \qquad (4.1.18)$$

where $\underline{\hat{x}}(t)$ is the estimated state vector and $v(t)$ the external input.

It follows that by substitution of (4.1.18) into the estimated state form of (4.1.1a), the state is given by

$$\underline{\hat{x}}(t) = z^{-1}(I-z^{-1}\underline{F})^{-1}\{QGy(t) + QSv(t) + Re(t)\} \qquad (4.1.19)$$

Therefore, by inclusion of (4.1.1b) the output sequence is obtained as,

$$y(t) = Hz^{-1}(I-z^{-1}\underline{F})^{-1}\{QGy(t) + QSv(t) + Re(t)\} + e(t)$$

and hence the final closed loop form is

$$[1-Hz^{-1}(I-z^{-1}\bar{F})^{-1}QG]y(t) = [1+Hz^{-1}(I-z^{-1}\bar{F})^{-1}R]e(t) + Hz^{-1}(I-z^{-1}\bar{F})^{-1}QSv(t) \quad (4.1.20)$$

The terms e(t) and v(t) therefore retain independent transfer function numerators, and the common denominator is once again equated to the previously selected pole polynomial, $T(z^{-1})$, by calculating the state feedback parameters from (4.1.5).

By use of (4.1.6) the closed loop form may then be written as,

$$T(z^{-1})y(t) = [\det(I-z^{-1}\bar{F}) + Hz^{-1}\mathrm{adj}(I-z^{-1}\bar{F})R]e(t) + z^{-k}B(z^{-1})Sv(t) \quad (4.1.21)$$

The final equation is the overall transfer function, inclusive of external input and linear output feedback, for the state space controller.

Removing the external input, by setting S or v(t) to zero, gives a closed loop equation (4.1.4), whereas removal of the ℓ.o.f. parameter, G = 0, leads to the form (3.3.3). However, where both are present the equivalent closed loop form, obtained when the estimated system model (4.1.8) is employed, must now be considered. By replacing matrices and polynomials in (4.1.21) by their estimated counterparts, we have

$$T(z^{-1})y(t) = \det[(I-z^{-1}\hat{\bar{F}}) + Hz^{-1}\mathrm{adj}(I-z^{-1}\hat{\bar{F}})\hat{R}]e(t) + \hat{B}'(z^{-1})Sv(t) \quad (4.1.22)$$

where the matrices containing the estimated parameters were specified in Sec, 4.1(2), and the state feedback parameters contained within $\hat{\bar{F}}$ are evaluated by means of (4.1.14).

It is apparent that, as in Sec.3.3, the compensator, S, can be chosen to provide any required steady state signal gain between the external input and system output. To achieve direct input steady

state tracking, the scalar term

$$S = T(1)/\hat{B}'(1) \qquad (4.1.23)$$

is calculated at each iteration by summing the terms in the estimated $\hat{B}'(z^{-1})$ polynomial and multiplying the reciprocal of this by the off line summation of the terms in the closed loop pole polynomial, $T(z^{-1})$. The calculation of S is therefore not dependent on the ℓ.o.f. parameter, G, and thus is not affected by this particular modification to the state feedback.

4. Optimization by selection of the ℓ.o.f. parameter

In Sec.4.1(2) it was stated that as the linear output feedback parameter, G , appears in the numerator of the closed loop transfer function, it directly affects the value of output variance obtained. This being the case, it is now required that G be chosen such that the variance of the system output is either reduced from its value when state feedback alone is employed or, better still, that it is minimized with respect to all possible values of G. However, without prior knowledge of the system parameters it is difficult to choose a G before controller start up, such that minimization occurs. It therefore remains to cause the output variance to be minimized by adapting the value of G whilst the tuner is in operation.

The relative output variance is obtained by dividing (4.1.17), the closed loop numerator, by the closed loop denominator, $T(z^{-1})$, and subsequently summing the terms in the resultant polynomial. By defining $H(z^{-1})$ to be the resultant closed loop polynomial, it follows that,

$$y(t) = H(z^{-1})e(t) \tag{4.1.24}$$

where $$H(z^{-1}) = 1 + \sum_{i=1}^{\infty} h_i z^{-i} \tag{4.1.25}$$

from which the relative output variance is obtained as,

$$R.V. = 1 + \sum_{i=1}^{\infty} h_i^2 \tag{4.1.26}$$

The summation of this series is found easily with respect to the parameters of $\hat{B}'(z^{-1})$, $T(z^{-1})$, etc., but computationally this becomes an inefficient procedure to repeat at each iteration. The actual value of relative output variance is, in fact, not required, and this can be shown by consideration of the requirements for G, the necessary condition being,

$$G^* = \arg.\min_{G}\{R.V.; R.V. = 1 + \sum_{i=1}^{\infty} h_i^2, \quad G \in \mathbb{R}\} \tag{4.1.27}$$

where G, a scalar real, is a constituent part of each h_i.

All h_i terms are independently squared, therefore the R.V. is given as a linear quadratic equation in G. Hence G^* can be found by obtaining the standard quadratic solution from,

$$R.V. - 1 = 0 \tag{4.1.27a}$$

But as the number of h_i terms present is much greater than the number of G's, of which there is one only, the solution to the quadratic equation necessarily gives rise to an imaginary G^*, such that

$$\bar{G}^* = g_1 + jg_2 \tag{4.1.28}$$

where $\bar{G}^*$ is the G^* which would be obtained, using j-notation, if an imaginary root was feasible. However, no phase deviation is allowable, due to the scalar real nature of G, thus

$$G^* = g_1 \tag{4.1.29}$$

where G^* is merely the real part of $\bar{G}^*$.

Thus to obtain G^* by a process inclusive of (4.1.26) at each iteration would be, as mentioned previously, extremely time consuming , and is not needed, as G^* may be defined prior to tuner operation as a combination of model estimates and closed loop pole polynomial coefficients.

Note. 4.1: on G^* and $\bar{G}^*$.

$\bar{G}^*$ is the value of G obtained as an exact solution to (4.1.27a). If it were computationally possible to apply this value of G then, as R.V. = 1, a minimum-variance self-tuner, with $\mathrm{Var}\{y(t)\} = \mathrm{Var}\{e(t)\}$ would have been formulated. But G must be a scalar quantity, and therefore it is not possible to apply the imaginary part of $\bar{G}^*$. Only when $g_2 = 0$ does $G^* = \bar{G}^*$ and a minimum variance self-tuner is obtained. This possibility is, however, almost always lost by the use of the pole placement approach to self-tuning.

5. Self-tuning with an adaptive ℓ.o.f. parameter

The overall self-tuning algorithm, inclusive of adaptive ℓ.o.f. parameter, G, may be described as

STEP 1) Estimate model parameters, via recursive least squares procedure, from system input and output readings.

2) Obtain state reconstruction using the estimated model parameters.

3) Calculate $\hat{G}^*$, as an estimate of G^*.

4) Evaluate state feedback vector $\hat{F}$.

5) Formulate the new input signal.

6) Repeat.

Step 4 is the extra insertion into the algorithm, and hence it is necessary that the computational requirements of this step remain small. Here $\hat{G}^*$ is the value of G^* obtained from the model estimates rather than from the actual system parameters. It will be shown shortly that $\hat{G}^*$ can converge to G^*, but firstly an example will be considered such that the number of calculations involved in evaluating $\hat{G}^*$ may be determined.

Example 4.1: For a three dimensional state space system model,

$$\hat{G}^* = \frac{\{G_1 \sum_{i=1}^{3} \hat{\lambda}_i\hat{b}_{i-1} + G_2(\sum_{i=2}^{3} \hat{\lambda}_i\hat{b}_{i-2} + \sum_{i=1}^{2} \hat{\lambda}_i\hat{b}_i) + G_3(\hat{\lambda}_1\hat{b}_2 + \hat{\lambda}_3\hat{b}_0)\}}{\{G_1 \sum_{i=0}^{2} \hat{b}_i^2 + 2G_2 \sum_{i=0}^{1} \hat{b}_i\hat{b}_{i+1} + 2G_3\hat{b}_0\hat{b}_2\}} \quad \ldots \quad (4.1.30)$$

where the following parameter definitions are made,

$$\left.\begin{aligned} &\text{a)} \quad G_1 = (1+t_2) - t_3(t_1+t_3) \\ &\text{b)} \quad G_2 = -t_1(1+t_2) + t_2(t_1+t_3) \\ &\text{c)} \quad G_3 = -t_2(1+t_2) + t_1(t_1+t_3) \end{aligned}\right\} \quad (4.1.31)$$

and

$$\left.\begin{aligned} &\text{a)} \quad \hat{\lambda}_1 = -\hat{a}_1 \\ &\text{b)} \quad \hat{\lambda}_2 = -\hat{a}_2 - (\hat{a}_3\hat{b}_0 - \hat{a}_1\hat{b}_2)\hat{f}_1 - (\hat{a}_2\hat{b}_0 - \hat{a}_1\hat{b}_1)\hat{f}_2 \\ &\text{c)} \quad \hat{\lambda}_3 = -\hat{a}_3 - (\hat{a}_3\hat{b}_1 - \hat{a}_2\hat{b}_2)\hat{f}_1 - (\hat{a}_3\hat{b}_0 - \hat{a}_1\hat{b}_2)\hat{f}_2 \end{aligned}\right\} \quad (4.1.32)$$

The values of $G_i : i = 1,2,3$; however, may be calculated prior to start up of the controller, once the pole polynomial has been specified,

and therefore do not have to be recalculated at each iteration. In the equivalent two dimensional case, $\hat{b}_2, \hat{a}_3$ and t_3 all become equal to zero, and thus it follows that $\hat{\lambda}_3$ is also equal to zero. The equation used to obtain $\hat{G}^*$ is therefore reduced to two G_i terms in both numerator and denominator. A general form for $\hat{G}^*$ in an m-dimensional model, where $m \geq 1$ and is a finite integer, can be formulated by a simple extrapolation of (4.1.30), but this leads to abbreviation difficulties, and hence the use of an example is preferred.

The convergence of $\hat{G}^*$ found from (4.1.30) is discussed via the following lemmas.

Lemma 4.1

The state feedback parameters $f_1 \rightarrow f_{m-1}$ are independent of G^*, and G^* is also independent of f_m, where it is assumed that m is the dimension of the state space.

Proof: If G^* is dependent on f_m, then f_m must appear in at least one λ_i. The λ_i's are formed as the result of pre- and post-multiplication of the adjoint of $(I-z^{-1}\bar{F})$ in (4.1.16), with system parameters substituted, such that

$$Hz^{-1}\text{adj}(I-z^{-1}\bar{F})R = 1 + \lambda_1 z^{-1} + \lambda_2 z^{-2} + \ldots + \lambda_m z^{-m}.$$

Because of the pre-multiplication of this adjoint by H, where $H = [0,\ldots,0,1]$, only the m^{th} row of the adjoint remains. Hence, by the definition of the adjoint of a matrix, no terms occupying positions in the m^{th} column of a matrix will appear in the m^{th} row of its adjoint, unless they also appear in a position other than in the m^{th} column. From the definition of $\bar{F}$, $\bar{F} = P + QF$, all terms involving f_m will

appear in only the m^{th} column of $(I-z^{-1}\bar{F})$, therefore no such terms will appear in H adj$(I-z^{-1}\bar{F})$, leaving G^* independent of f_m.

The independence of the parameters $f_1 \rightarrow f_{m-1}$ follows from this, as G may be replaced in (4.1.5) by its respective function in $f_1 \rightarrow f_{m-1}$.

Lemma 4.2

The convergence of the state feedback parameters is not affected by the adaptive nature of G.

Proof: In Sec.3.1 it was proven that for 'any' set value of G, subject to the model estimates converging, the feedback parameters $\hat{f}_i$ can converge. As, from Lemma 4.1, $\hat{f}_1 \rightarrow \hat{f}_{m-1}$ do not depend on $\hat{G}$, the fact that G is adaptive cannot affect their convergence. But $\hat{G}$ is calculated by means of $\hat{f}_1 \rightarrow \hat{f}_{m-1}$, and no account is taken of $\hat{f}_m$, thus if $\hat{f}_1 \rightarrow \hat{f}_{m-1}$ converge then $\hat{G}$ will do likewise, unless the denominator of $\hat{G} \rightarrow 0$ which is not possible in practice. The one remaining parameter, $\hat{f}_m$, must therefore also converge, as every factor upon which it is dependent has converged.

6. Example and simulations

Consider the system described by,

$$(1-z^{-1}+0.25z^{-2})y(t) = (z^{-1}+1.3z^{-2})u(t)+(1-0.4z^{-1})e(t) \quad (4.1.33)$$

where e(t) is a white noise sequence of variance = 0.1. Using a pole polynomial, $T(z^{-1})$ equal to unity, and the control law given in (4.1.10), the diagram of Fig.4.1 shows how the theoretical output variance is related to a change in the linear output feedback parameter,

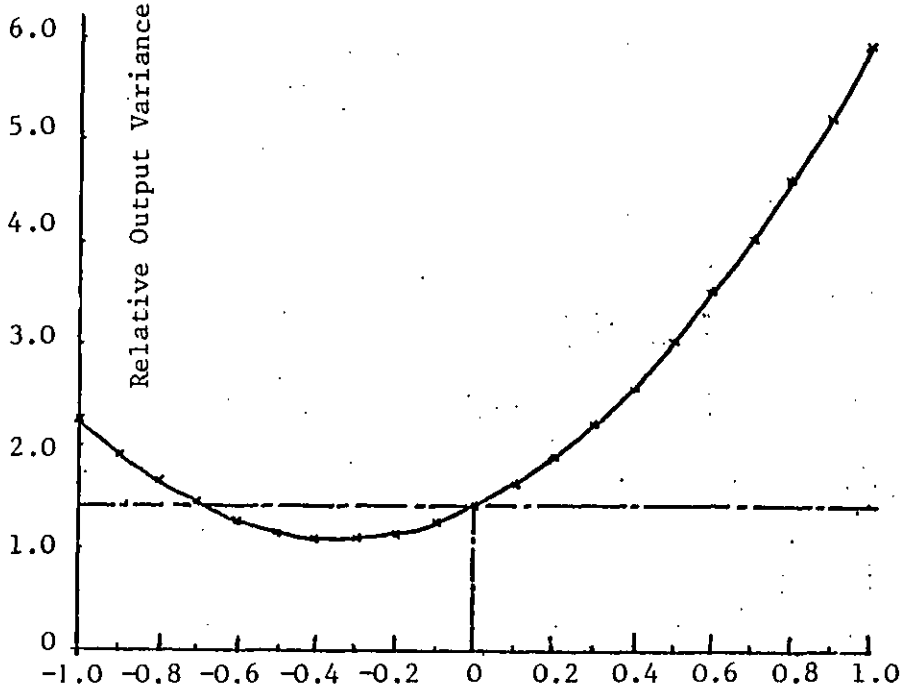


Fig. 4.1. Dependence of Output Variance on L.O.F. Parameter

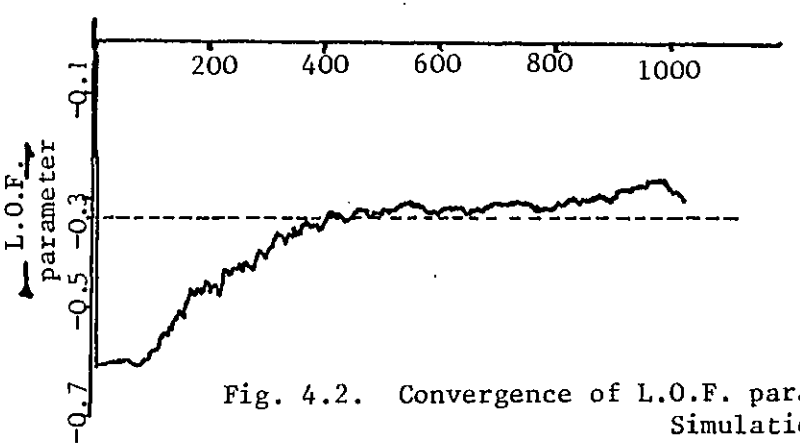


Fig. 4.2. Convergence of L.O.F. parameter, Simulation 1.

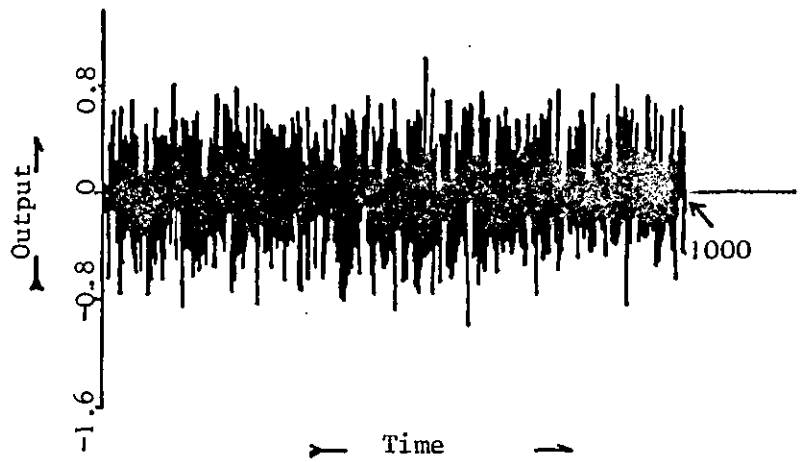


Fig. 4.3. System Output, Simulation 1.

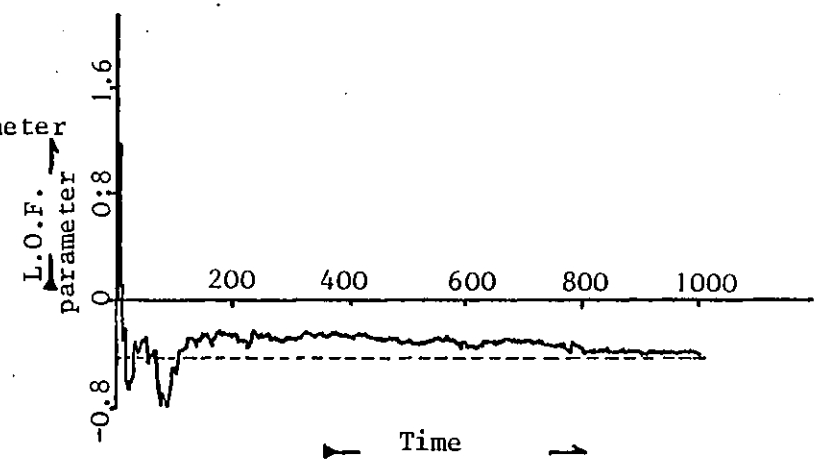


Fig. 4.4. Convergence of L.O.F. Parameter, Simulation 2.

G. Dotted lines show the output variance obtained when solely state feedback is employed, i.e. G is zero. The minimum value of relative output variance is found to be 1.1073 when G = -0.3404, this compares to a relative output variance obtained via the original polynomial method, Wellstead et al (1979b), of 1.1716.

Simulation 1: During these simulations the variable forgetting factor proposed by Wellstead and Sanoff (1981) was employed, with an initial start up value of 0.94 and a window of 100 samples (τ=100). In the first simulation the system given in the above example was considered with $T(z^{-1})$ set out at unity. Fig.4.2 shows the convergence of the linear output feedback parameter and Fig.4.3 the system output, y(t). It can be seen that suitable convergence is obtained, under these conditions, within 1000 time intervals, and hence this remained as the number of periods investigated for the remaining simulations.

Simulation 2: The same process (4.1.33) was considered, although in this case the pole polynomial was chosen to be,

$$T(z^{-1}) = 1 + 0.5z^{-1} \qquad (4.1.34)$$

Fig.4.4, then gives the convergence of the ℓ.o.f. parameter, where the dotted line indicates the value of G obtained when the problem is calculated with known system parameters, i.e. $G = G^*$, and this is the hoped for convergence value of $\hat{G}$.

Simulation 3: The final u(t) parameter, i.e. $1.3z^{-2}$, is omitted from the system given in the example (4.1.33), and the effect of a change in pure system time delay is examined.

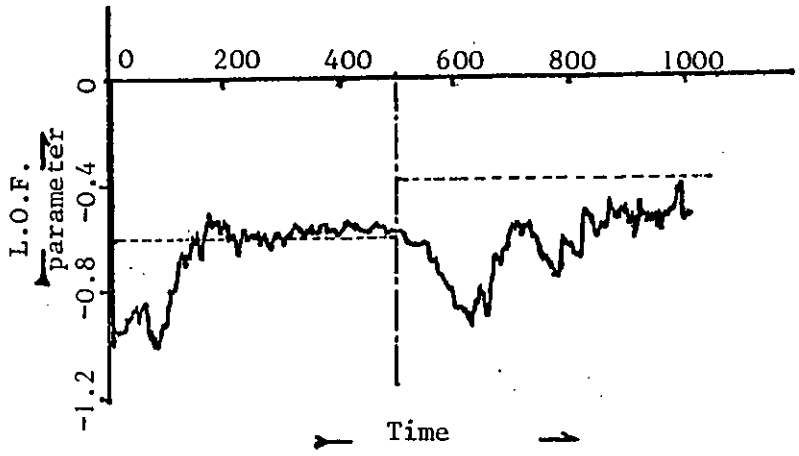


Fig. 4.5. Convergence of L.O.F. Parameter, Simulation 3.

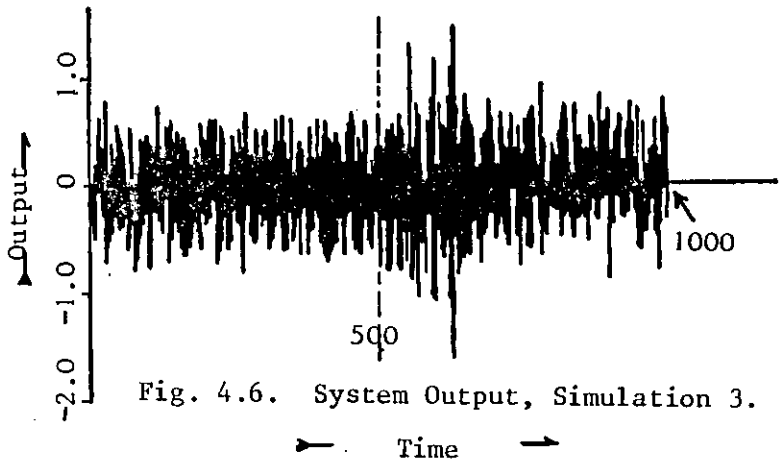


Fig. 4.6. System Output, Simulation 3.

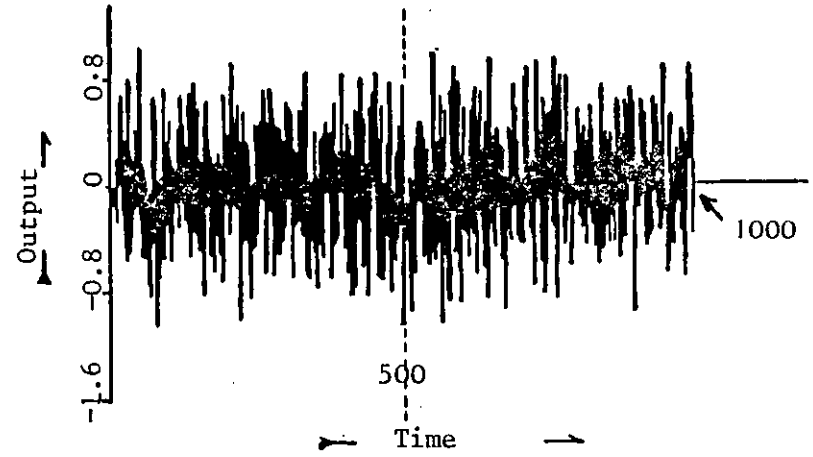


Fig. 4.7. System Output(G=0.0), Simulation 3.

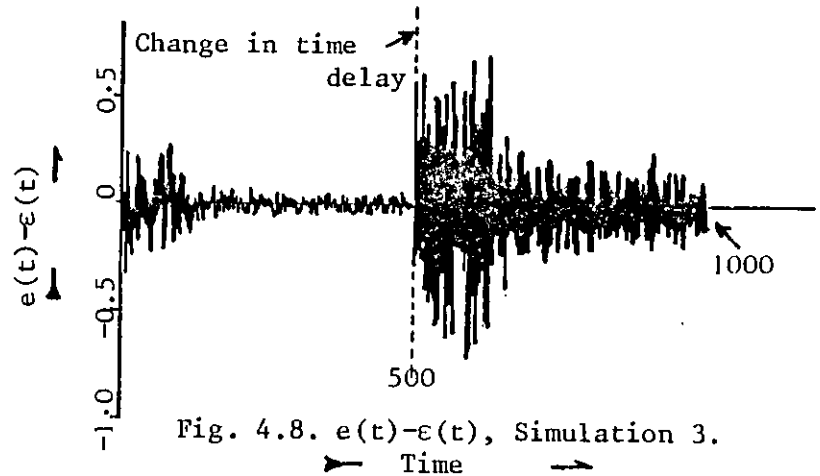


Fig. 4.8. e(t)-ε(t), Simulation 3.

During the first 500 time periods the process is described by,

$$(1-z^{-1}+0.25z^{-2})y(t) = z^{-1}u(t) + (1-0.4z^{-1})e(t) \qquad (4.1.35)$$

After the 500^{th} time instant this changes to,

$$(1-z^{-1}+0.25z^{-2})y(t) = z^{-2}u(t) + (1-0.4z^{-1})e(t) \qquad (4.1.36)$$

The model, from which the estimated parameter values are obtained, remains constant, although in this case the model parameters readjust after the change in time delay has occurred. With the pole polynomial, once again, set to unity, Fig.4.5 shows the value of the ℓ.o.f. parameter throughout the simulation, and it is noticeable that convergence is slower after the time delay alteration. In the system output , Fig. 4.6, large spikes appear after the change, a phenomenon which was seen in the original polynomial simulations carried out in Sec.2.5(5). By setting G equal to zero, Fig.4.7, the spikes are removed, although a larger output variance results.

The difference between the system disturbance, e(t), and the model error, $\varepsilon(t)$, is given in Fig.4.8. This should tend to zero due to the property exhibited by self-tuners that $\varepsilon(t) \to e(t)$ as $t \to \infty$.

Simulation 4: The closed loop pole polynomial is selected to be that of (4.1.34), and simulation 3 is repeated with this singular difference. The convergence of the ℓ.o.f. parameter, Fig.4.9, is again shown to be weaker after the time delay change, and this is combined with a larger value of $e(t) - \varepsilon(t)$, Fig.4.10.

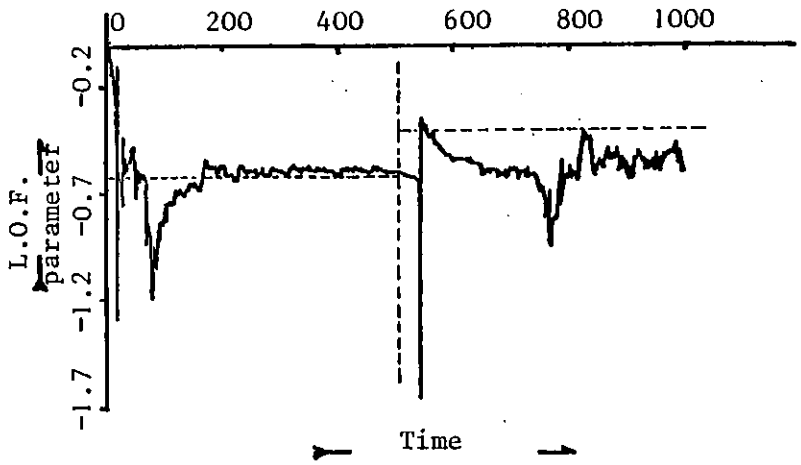


Fig. 4.9 Convergence of L.O.F. Parameter, Simulation 4.

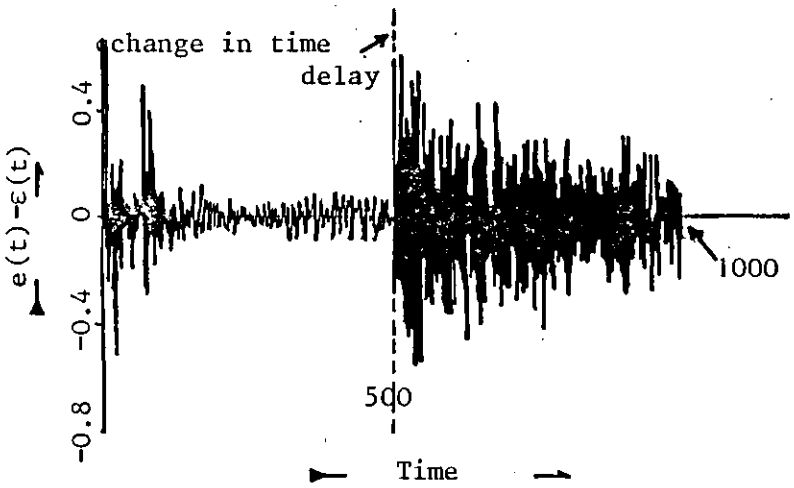


Fig. 4.10. e(t) - ε(t), Simulation 4.

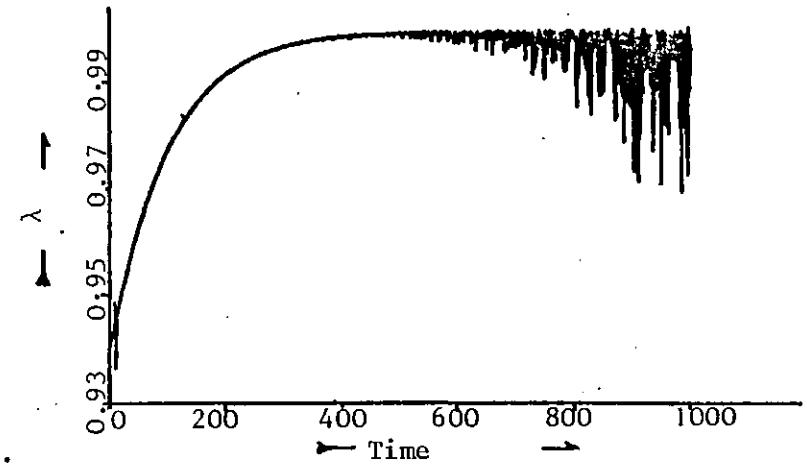


Fig. 4.11, Variable Forgetting Factor (λ), Simulation 1.

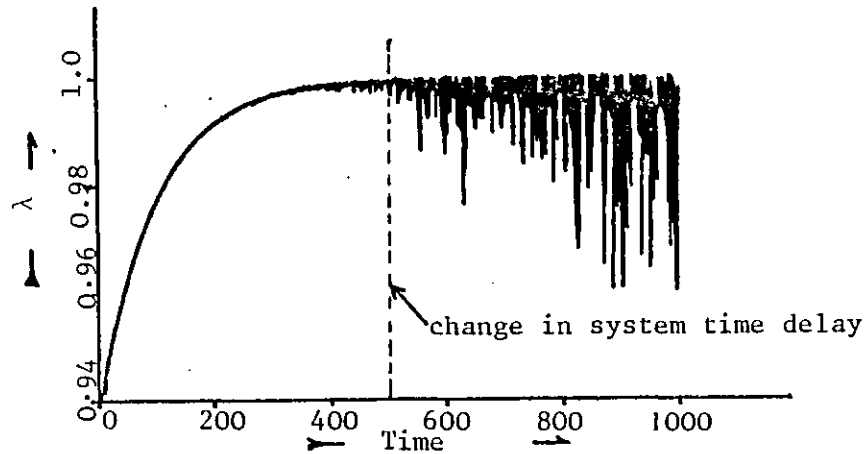


Fig. 4.12. Variable Forgetting Factor (λ), Simulation 3

The asymptotic output variances for the preceding simulations are given in the following table; where (a) denotes the value before the change in time delay and (b) the value after the change.

Table 4.1:

Simulation	G^*	Output Variance (O.V)	O.V. with purely state feedback
1	-0.3404	0.1107	0.1419
2	-0.4097	0.1079	0.1389
3(a)	-0.6	0.1	0.1360
3(b)	-0.35	0.1360	0.1482
4(a)	-0.6	0.1	0.1480
4(b)	-0.35	0.1360	0.1523

In 3(a) and 4(a) of table 4.1, the output variance is equal to the variance of the system disturbance, e(t), i.e. minimum-variance self-tuning has been applied. This is a special case when the system $B(z^{-1})$ polynomial is of order unity, allowing the ℓ.o.f. parameter to converge to $\bar{G}^*$.

4.2 Multivariable Controller Design

1. Introduction

Minimum-variance type controllers were employed with multivariable systems in Borisson (1975), and many of the properties that are apparent in the SISO version also hold in the MIMO case. Unfortunately some of these are disadvantageous, such as the problems associated with nonminimum phase systems and the acute sensitivity shown when alterations in system time delay occur. The system time delay, k, provides a further limitation in the usage of this class of regulator in multivariable control systems, as a basic requirement is that all control loops have identical k values. The number of practical situations where this is the case are relatively few and hence the pole placement approach, which allows for inputs with varying integer time delay values, has been developed, Prager and Wellstead (1981). This type of algorithm provides a more robust control action, which deals with non minimum phase systems, although extra computational effort is required and the regulation characteristics are suboptimal.

In this section the minimum variance type tuners will be reviewed, the progression to the pole placement case discussed, and finally the state space approach to multivariable self-tuning controllers will be introduced.

2. Minimum-variance controllers

The process is modelled by the equation,

$$A(z^{-1})\underline{y}(t) = z^{-k}B(z^{-1})\underline{u}(t) + C(z^{-1})\underline{e}(t) \qquad (4.2.1)$$

where $\{\underline{y}(t) : t\epsilon T\}$ and $\{\underline{u}(t) : t\epsilon T\}$ are vectors of dimension $m\times 1$ denoting the system output and input sequences respectively.

$\{\underline{e}(t) : t = 0, \pm 1, \pm 2, \ldots\}$ is a vector sequence, also of dimension $m\times 1$, which is assumed to be zero-mean with a covariance equal to Ω.

The matrix polynomials in the above equation are defined as,

$$A(z^{-1}) = A_0 + A_1 z^{-1} + \ldots + A_{n_a} z^{-n_a} \qquad (4.2.2)$$

where $B(z^{-1})$ and $C(z^{-1})$ are defined in a similar fashion, being of degree n_b and n_c respectively, although it will be considered that all polynomials are of degree n_l, where n_l is the maximum degree contained by the three polynomials. If a particular polynomial is of degree less than n_l, its higher order, not apparent, matrices are defined to be the null matrix.

In general A_i, B_i, C_i are of dimension $m\times m$, where $i = 0, \ldots, n_l$ and A_0, C_0 are the unit matrix such that $A(z^{-1})$ and $C(z^{-1})$ are monic polynomial matrices. In the minimum variance case it will be considered that all control loops have identical integer time delays, thus k is common to all m inputs.

The matrix polynomial identity,

$$C(z^{-1}) = A(z^{-1})E(z^{-1}) + z^{-k}G(z^{-1}) \qquad (4.2.3)$$

in which $E(z^{-1})$ is monic and of degree $k-1$, and $G(z^{-1})$ is of degree $n_l - 1$, is used to obtain the solution for the control law polynomials $E(z^{-1})$ and $G(z^{-1})$, which are required for the control input found as follows.

The system output vector is predicted k steps ahead of the present time to be,

$$E\{\underline{y}(t+k)\} = \underline{y}(t+k) - E(z^{-1})\underline{e}(t+k) \qquad (4.2.4)$$

and $$E\{\underline{y}(t+k)\} = C^{-1}(z^{-1})[B(z^{-1})E(z^{-1})\underline{u}(t) + G(z^{-1})\underline{y}(t)] \qquad (4.2.5)$$

Thus, if it is required that the expected value of the system output vector is zero at time t+k, the control input must necessarily be,

$$\underline{u}(t) = -B^{-1}(z^{-1})G(z^{-1})E^{-1}(z^{-1})\underline{y}(t) \qquad (4.2.6)$$

where $E(z^{-1})$ and $G(z^{-1})$ are obtained from (4.2.3).

Hence $$\underline{y}(t+k) = E(z^{-1})\underline{e}(t+k) \qquad (4.2.7)$$

or $$\underline{y}(t) = \underline{e}(t) + E_1 e(t-1) + \ldots + E_{k-1} e(t-k+1).$$

Therefore the final control input (4.2.6) minimizes $E\{\underline{y}^T(t)\ \Sigma\underline{y}(t)\}$; Σ being a positive definite symmetric matrix. If the matrix polynomials defined in (4.2.1) are now assumed to be unknown, they may be estimated recursively, on line by means of the recursive least squares procedure. The application of this estimation scheme to the SISO minimum variance case was reviewed in Sec.1.5(1), and the same principles hold here, although rather than an estimation of model parameters taking place, the m×m model parameter matrices must now be estimated. Thus, where once one parameter was estimated, this is replaced by an m×m matrix of parameters. The extension of self-tuning to deal with multivariable systems therefore introduces a large increase in the necessary computing time.

Many of the properties associated with the SISO minimum-variance self-tuning controller carry through to the MIMO case. It has been shown that the control law (4.2.6) and the identity (4.2.3) are of the same form, and this is also true of the stability analysis. Denoting the determinant of the matrix polynomial, $B(z^{-1})$, by $|B(z^{-1})|$, then for stability of the multivariable self-tuning algorithm it is a requirement

that neither $|B(z^{-1})|$ nor $|C(z^{-1})|$ have any unstable roots. Hence the dependence of operation of the minimum variance controller on the minimum phase behaviour of the plant.

As in the SISO case a detuned minimum variance controller may be formulated by placing a costing on the control input as well as the system output.

We will, therefore, consider the function; $E\{\underline{y}^T(t+k)\sum\underline{y}(t+k)+\underline{u}^T(t)\gamma\underline{u}(t)\}$, which must be minimized, by considering γ to be, like $\sum$, a positive definite symmetric matrix.

By partial differentiation of the above function with respect to $\underline{u}(t)$; the required control input is found to be,

$$\underline{u}(t) = -\gamma^{-1}B_0^T\sum E\{\underline{y}(t+k)\} \qquad (4.2.8)$$

where $E\{\cdot\}$ denotes the expected value.

However, from (4.2.4) this may be rewritten as,

$$\underline{u}(t) = -\gamma^{-1}B_0^T\sum[\underline{y}(t+k) - E(z^{-1})\underline{e}(t+k)].$$

By use of (4.2.1) it follows that,

$$\underline{u}(t) = -\gamma^{-1}B_0^T\sum[\underline{y}(t+k)-E(z^{-1})C^{-1}(z^{-1})A(z^{-1})\underline{y}(t+k)+E(z^{-1})C^{-1}(z^{-1})B(z^{-1})\underline{u}(t)]$$

Hence,

$$[E(z^{-1})C^{-1}(z^{-1})B(z^{-1}) + \sum^{-1}B_0^{-T}\gamma]\underline{u}(t) = [E(z^{-1})C^{-1}(z^{-1})A(z^{-1})-I]\underline{y}(t+k)$$

and therefore, if the identity (4.2.3) is employed,

$$[B(z^{-1}) + C(z^{-1})E^{-1}(z^{-1})\sum^{-1}B_0^{-T}\gamma]\underline{u}(t) = -G(z^{-1})E^{-1}(z^{-1})\underline{y}(t) \qquad (4.2.9)$$

. . .

where $\sum$ and γ can be chosen by the system operator.

3 Pole placement controller

Assuming the process to be modelled by (4.2.1), the control law is defined as,

$$\underline{u}(t) = G(z^{-1})D^{-1}(z^{-1})\underline{y}(t) \tag{4.2.10}$$

where $D(z^{-1})$ is a monic matrix polynomial of degree n_b+k-1, $G(z^{-1})$ is of degree n_a-1, and both are of the form (4.2.2).

By substitution of (4.2.10) into (4.2.1) the closed loop equation for the system with this control law applied is found to be,

$$[A(z^{-1}) - z^{-k}B(z^{-1})G(z^{-1})D^{-1}(z^{-1})]\underline{y}(t) = C(z^{-1})\underline{e}(t) \tag{4.2.11}$$

or $$[A(z^{-1})D(z^{-1}) - z^{-k}B(z^{-1})G(z^{-1})]D^{-1}(z^{-1})\underline{y}(t) = C(z^{-1})\underline{e}(t).$$

We now wish to choose the closed-loop poles of the overall system, and by defining $T(z^{-1})$ to be monic and of the same form as (4.2.2), these poles are obtained as the solution to $|T(z^{-1})|$ which may be chosen prior to application of the control law.

Therefore using the identity,

$$[A(z^{-1})D(z^{-1}) - z^{-k}B(z^{-1})G(z^{-1})] = C(z^{-1})T(z^{-1}) \tag{4.2.12}$$

such that the matrices of the polynomials used in the control input equation (4.2.10), $D(z^{-1})$ and $G(z^{-1})$, may be evaluated from this, the closed loop system now becomes,

$$T(z^{-1})D^{-1}(z^{-1})\underline{y}(t) = \underline{e}(t)$$

where a cancellation of the $C(z^{-1})$ matrix polynomial occurs.

The closed loop form may be rewritten as,

$$\underline{y}(t) = D(z^{-1})T^{-1}(z^{-1})\underline{e}(t) \tag{4.2.13}$$

and thus a condition of relative primeness must exist between $D(z^{-1})$ and $T(z^{-1})$, which is almost always true.

Although this method has the advantage, as in the SISO case, over minimum-variance multivariable controllers, in its non-sensitivity to non minimum phase plant, the major advantage is its ability to deal with systems in which a different integer time delay can be associated with each input. The number of multivariable processes that contain uniform delays, k, is extremely small, Borisson (1975), and this greatly limits the field of application for minimum-variance type MIMO controllers,

The use of the recursive least squares estimation scheme, when the system matrices (4.2.1) are unknown, is carried out in a similar fashion to that in the SISO problem, which was dealt with in Sec.1.5(3). Once again, though, where previously single parameters were estimated, these are now replaced by matrices of dimension m×m, all of whose parameters must be estimated. The self-tuning property of the overall algorithm then holds, Prager and Wellstead (1981), provided the maximum value of integer time delay, k, in any loop, does not overshoot the maximum value specified, k_m, which now becomes the maximum possible value connected with any input.

4. State-space controller

The process is considered to be modelled by use of (4.2.1) in the CARMA form. A state space approach, similar to that used for SISO systems may now be defined as,

$$\left.\begin{aligned} \underline{x}(t+1) &= P\underline{x}(t) + Q\underline{u}(t) + R\underline{e}(t) \\ \underline{y}(t) &= H\underline{x}(t) + \underline{e}(t) \end{aligned}\right\} \qquad (4.2.14)$$

where,

$$P = \begin{bmatrix} \mathbf{0} & \cdots & \mathbf{0} \\ I & & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & -A_{n_1} \\ \mathbf{0} & \mathbf{0} & -A_2 \\ \mathbf{0} & I & -A_1 \end{bmatrix}, \quad Q = \begin{bmatrix} B_{n_1} \\ \vdots \\ B_0 \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix} \tag{4.2.15}$$

$$R = \begin{bmatrix} \mathbf{0} \\ \vdots \\ \mathbf{0} \\ C_{n_1} - An_1 \\ \vdots \\ C_1 - A_1 \end{bmatrix} \qquad H^T = \begin{bmatrix} \mathbf{0} \\ \vdots \\ \mathbf{0} \\ I \end{bmatrix}$$

where $n_1 = \max\{n_a, n_b, n_c\}$, and as each matrix A_1, B_1, C_1 etc. is of dimension m×m, the overall dimension of the matrix P is $(n_1+k)m \times (n_1+k)m$. The matrices I and $\mathbf{0}$, the identity and null matrices, are necessarily of dimension m×m and Q, R and H^T are of dimension $(n_1+k)m \times m$. $\{\underline{u}(t) : t\epsilon T\}$ and $\{\underline{y}(t) : t\epsilon T\}$ are m×1 vector input and output sequences respectively and the state, $\underline{x}(t)$, is now of dimension $(n_1+k)m \times 1$ such that the number of states is increased in the MIMO case, from the SISO system, by a factor m.

The simplest method of analysis for the multivariable state space controller is by a simple extrapolation to the SISO model. It was shown (2.3.6) that for a SISO system the estimate of the state vector is given by,

$$\hat{\underline{x}}(t) = (I_1 - z^{-1}\bar{P})^{-1} z^{-1}[Q\underline{u}(t) + R\underline{y}(t)] \tag{4.2.16}$$

where now the original equation has been extended to cover the multivariable case. I_1 is the identity matrix of dimension $(n_1+k)m \times (n_1+k)m$ and here $C(z^{-1})$ is considered to be equal to the $m \times m$ unit matrix I. As $\bar{P} = P - RH$, the estimate may be rewritten,

$$\hat{\underline{x}}(t) = \begin{bmatrix} B_{n_1} z^{-1} \\ B_{n_1-1} z^{-1} + B_{n_1} z^{-2} \\ \vdots \\ z^{-k}(B_0 + z^{-1}B_1 + \ldots + B_{n_1} z^{-n_1}) \end{bmatrix} \underline{u}(t) + \begin{bmatrix} 0 \\ \vdots \\ 0 \\ -A_{n_1} z^{-1} \\ \vdots \\ -A_1 z^{-1} - A_2 z^{-2} - \ldots - A_{n_1} z^{-n_1} \end{bmatrix} \underline{y}(t)$$

. . . (4.2.17)

Although where $C(z^{-1})$ is regarded in its general form,

$$(I_1 - z^{-1}\bar{P}) = \begin{bmatrix} I & 0 & \cdots & 0 \\ -z^{-1}I & \ddots & & \vdots \\ 0 & \ddots & 0 & 0 \\ \vdots & & \ddots & z^{-1}c_{n_1} \\ & & I & z^{-1}c_2 \\ 0 & \cdots & -z^{-1}I & I + z^{-1}c_1 \end{bmatrix} \qquad (4.2.18)$$

This leads to a more complicated expression for $\hat{\underline{x}}(t)$, as in the SISO case, and although this is not of direct importance in this section the estimate is nevertheless rewritten in the form of (4.2.17) in Appendix 4.1, where $(I_1 - z^{-1}\bar{P})^{-1}$ is obtained.

However, where a recursive least squares procedure is employed the state may be reconstructed from the equivalent of (4.2.17), and thus

$C(z^{-1})$ is considered as the unit matrix.

The control input is defined as,

$$\underline{u}(t) = F\underline{\hat{x}}(t) \tag{4.2.19}$$

where F, the state feedback matrix, now contains $m \times (n_1+k)m$ parameters.

By substitution of (4.2.19) into the equations (4.2.14) the closed loop equation is therefore,

$$\underline{y}(t) = [I + Hz^{-1}(I_1 - z^{-1}\bar{F})^{-1}R]\underline{e}(t) \tag{4.2.20}$$

where $\bar{F} = P + QF$.

This final form is similar to (4.1.4) in the SISO case, but because of the properties of the matrix multiplication procedures now involved, especially in finding the inverse of $I_1 - z^{-1}\bar{F}$, any attempt to assign the closed loop poles directly from the above equation is not feasible.

For a simpler approach to the multivariable pole placement problem in a state space framework we will make use of the separation principle, where the estimation of the state vector and the calculation of the controller action may be derived without consideration of their counterpart calculation. As the state estimate has already been dealt with our attention is now focussed on the determination of the controller action by means of the selection of closed loop poles.

The deterministic system is now considered where,

$$\left.\begin{aligned} \underline{x}(t+1) &= P\underline{x}(t) + Q\underline{u}(t) \\ \underline{y}(t) &= H\underline{x}(t) \end{aligned}\right\} \tag{4.2.21}$$

where the matrices are as defined in (4.2.15).

If the control input is chosen to be (4.2.19), the matrix of dynamics of the feedback system is $\bar{F}$. Thus the required closed loop poles may be found by the correct choice of $\bar{F}$ in the matrix

$I_1 - z^{-1}\bar{F}$. However the present state space formulation, with matrices defined in (4.2.15), is not a canonical form. In the SISO case the canonical form of reconstructability was employed, and thus here we will obtain the multivariable canonical form of reconstructability before proceeding with the pole placement. Defining a matrix L, as that which is required in order to obtain a reconstructability canonical form, then

$$P_R = L^{-1}PL, \; Q_R = L^{-1}Q, \; H_R = HL \tag{4.2.22}$$

and

$$\underline{x}_R(t) = L^{-1}\,\underline{x}(t) \tag{4.2.23}$$

To give an equivalent version of (4.2.21) in the required form as,

$$\left.\begin{aligned} \underline{x}_R(t+1) &= P_R\,\underline{x}_R(t) + Q_R\,\underline{u}(t) \\ \underline{y}(t) &= H_R\,\underline{x}_R(t) \end{aligned}\right\} \tag{4.2.24}$$

Now however the state vector is not equivalent to that previously estimated, thus if the state feedback parameter matrix F is found from the form (4.2.24) the control input is then given by,

$$\underline{u}(t) = F\hat{\underline{x}}_R(t) = FL^{-1}\hat{\underline{x}}(t) \tag{4.2.25}$$

The method used to obtain the correct matrix L is set out in many works, e.g. Strejc (1981), the solution to this problem being stated as the $(n_1+k)m \times (n_1+k)m$ matrix,

$$L = L^{-1} = \left(\begin{array}{c|c|c} \begin{matrix} 1 & 0 & \cdots & 0 \\ 0 & 0 & & \vdots \\ \vdots & & & \vdots \\ 0 & \cdots & \cdots & 0 \end{matrix} & \begin{matrix} 0 & \cdots & 0 \\ 1 & & \vdots \\ 0 & & \vdots \\ 0 & \cdots & 0 \end{matrix} & \begin{matrix} 0 & \cdots & 0 \\ \vdots & & \vdots \\ 0 & & 0 \\ 1 & 0 \cdots 0 & 0 \end{matrix} \\ \hline \begin{matrix} 0 & 1 & 0 & \cdots & 0 \\ \vdots & & & & \\ & & & & \\ 0 & \cdots & \cdots & \cdots & 0 \end{matrix} & \begin{matrix} 0 & \cdots & 0 \\ 0 & 1 & 0 \cdots 0 \\ \vdots & & \\ 0 & \cdots & 0 \end{matrix} & \begin{matrix} 0 & \cdots & 0 \\ \vdots & & \vdots \\ 0 & 1 & 0 \cdots 0 \end{matrix} \\ \hline \begin{matrix} 0 & \cdots & 0 & 1 \\ \vdots & & & 0 \\ \vdots & & & \vdots \\ 0 & \cdots & \cdots & 0 \end{matrix} & & \begin{matrix} 0 & \cdots & 0 \\ \vdots & & \vdots \\ \vdots & & 0 \\ 0 & \cdots & 0 & 1 \end{matrix} \end{array}\right) \qquad (4.2.26)$$

Hence L is symmetric.

This new state space form can now be regarded as having matrices with the following structure,

$$H_R = \left(\begin{array}{cccccccc} 0 & \cdots & 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & & 0 & 0 & \cdots & 0 & 1 & \vdots \\ \vdots & & \vdots & \vdots & & & \vdots & \vdots \\ \vdots & & \vdots & \vdots & & 0 & \vdots & 0 \\ 0 & \cdots & 0 & 0 & \cdots & 0 & 0 \cdots 0 & 1 \end{array}\right) \qquad (4.2.27)$$

$$P_R = \left(\begin{array}{c|c|c|c} \begin{matrix} 0 & \cdots & \cdots & 0 \\ \vdots & & & 0 \\ 1 & \ddots & & \vdots \\ \vdots & \ddots & & -a_{n_a}^{11} \\ 0 & \ddots & & \vdots \\ \vdots & \ddots & 0 & \vdots \\ \vdots & \ddots & & \vdots \\ 0 \cdots & 0 & 1 & -a_1^{11} \end{matrix} & \begin{matrix} 0 & \cdots & 0 \\ \vdots & & 0 \\ \vdots & & -a_{n_a}^{12} \\ \vdots & & \vdots \\ \vdots & & \vdots \\ \vdots & & \vdots \\ \vdots & & \vdots \\ 0 \cdots & 0 & -a_1^{12} \end{matrix} & \cdots & \begin{matrix} 0 & \cdots & 0 \\ \vdots & & -a_{n_a}^{1m} \\ \vdots & & \vdots \\ \vdots & & \vdots \\ \vdots & & \vdots \\ \vdots & & \vdots \\ \vdots & & \vdots \\ 0 \cdots & 0 & -a_1^{1m} \end{matrix} \\ \hline \begin{matrix} 0 & \cdots & \cdots & 0 \\ \vdots & & & 0 \\ \vdots & & & \vdots \\ \vdots & & & -a_{n_a}^{21} \\ \vdots & & & \vdots \\ \vdots & & & \vdots \\ 0 & \cdots & \cdots & -a_1^{21} \end{matrix} & \begin{matrix} \cdots & \cdots & \cdots \\ & & \\ & & \\ & & \\ & & \\ & & \\ \cdots & \cdots & \cdots \end{matrix} & \cdots & \begin{matrix} & & \\ & & \\ & & \\ & & \\ & & \\ & & \\ & & \end{matrix} \\ \hline \begin{matrix} & & \\ & & \\ & & \\ & & \\ & & \\ & & \\ & & \end{matrix} & \begin{matrix} & & \\ & & \\ & & \\ & & \\ & & \\ & & \\ & & \end{matrix} & \cdots & \begin{matrix} & & \\ & & \\ & & \\ & & \\ & & \\ & & \\ & & \end{matrix} \\ \hline \begin{matrix} & & \\ & & \\ & & \\ & & \\ & & \\ & & \\ & & \end{matrix} & \begin{matrix} & & \\ & & \\ & & \\ & & \\ & & \\ & & \\ & & \end{matrix} & \cdots & \begin{matrix} 0 & \cdots & 0 \\ \vdots & & 0 \\ 1 & \ddots & \vdots \\ \vdots & \ddots & -a_{n_a}^{mm} \\ 0 & \ddots & \vdots \\ \vdots & \ddots & \vdots \\ 0 \cdots & 0 & 1 \; -a_1^{mm} \end{matrix} \end{array}\right) \qquad (4.2.28)$$

where the $A(z^{-1})$ matrix polynomial is redefined such that each element is of the form,

$$A_i = \begin{bmatrix} a_i^{11} & \cdots & a_i^{1m} \\ \vdots & & \vdots \\ a_i^{m1} & \cdots & a_i^{mm} \end{bmatrix} \qquad (4.2.29)$$

where $i = 1,\ldots,n_a$. For $i = 0$, A_0 is the unit matrix which can be derived as a special case of (4.2.29).

The new state space formulation may therefore be directly obtained from the parameters in the CARMA model. This is also true of the state estimate $\hat{\underline{x}}_R(t)$, where although we now have $\hat{\underline{x}}_R(t) = L\hat{\underline{x}}(t)$, as L contains only zeros and ones, this can be regarded merely as altering the positions of the state estimates in the state vector, no multiplications take place.

Many methods then exist for obtaining the necessary closed loop pole polynomial form. The calculation of the state feedback parameters is much simplified by transforming the state equations into a controllable canonical form, although this leads to a more complicated expression for the associated state vector. The 'best' method, in some sense, is the topic for further research and thus will not be dealt with further here. Defining the specified closed loop matrix polynomial as $T(z^{-1})$, where,

$$T(z^{-1}) = I + T_1 z^{-1} + \ldots + T_{n_t} z^{-n_t} \qquad (4.2.30)$$

and each $T_i = \begin{bmatrix} T_i^1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & T_i^m \end{bmatrix} \qquad (4.2.31)$

Then $T(z^{-1}) =$

$$\begin{bmatrix} T^1(z^{-1}) & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & T^m(z^{-1}) \end{bmatrix}$$

Thus to obtain the necessary closed loop form,

$$(I_1 - z^{-1}\bar{F})^{-1} = X \cdot \begin{bmatrix} \begin{bmatrix} T^1(z^{-1}) & & 0 \\ & \ddots & \\ 0 & & T^1(z^{-1}) \end{bmatrix} & & 0 \\ & \ddots & \\ 0 & & \begin{bmatrix} T^m(z^{-1}) & & 0 \\ & \ddots & \\ 0 & & T^m(z^{-1}) \end{bmatrix} \end{bmatrix}^{-1}$$

. . . (4.2.32)

where X is the residual adjoint matrix, and substituting for this in the closed loop equation (4.2.20), leads to,

$$\underline{y}(t) = [I + z^{-1}YT^{-1}(z^{-1})]\underline{e}(t)$$

where Y is obtained from HXR.

Thus $$\underline{y}(t) = [T(z^{-1}) + z^{-1}Y]\, T^{-1}(z^{-1})\underline{e}(t) \qquad (4.2.33)$$

which produces the necessary pole polynomial matrix.

4.3 Nonlinear Systems Control

1. Introduction

The basic model employed in this thesis, where SISO models are concerned, has regarded the system to be linear, Åström (1970). Because of this, the superposition theorem holds and thus the model, whether state space or polynomial, is much simplified due to the system and measurement disturbances being thought of as one and the same. In practice though nonlinearities, in the process to be controlled, may occur and in this case a self-tuning algorithm cannot be directly implemented, although minor nonlinear effects can, in fact, be dealt with due to the incorporation of an on-line recursive parameter estimator.

Where nonlinearities are much stronger, especially around the point of operation of the self-tuner, this must be accounted for by use of either a more responsive estimation scheme,Wittenmark (1979), or by including nonlinearities in the system model, Anbumani et al (1981). It is generally true, though, that if a nonlinear model is used, many of the properties attributable to self-tuning controllers will no longer apply, and furthermore the formulation of a control law which is suitable for use with the model can become extremely complicated. As in the case of linear systems, the final control scheme must be computationally simple enough to be deployed on a microprocessor in an iterative procedure, and this limits the encroachment into nonlinear theory to a large extent.

The two methods mentioned will be discussed in this section with emphasis placed on their effects on the state space description.

2. Dual estimation method

Kalman filtering estimation techniques, introduced in Sec.1.4, provide a good estimation scheme for a system with parameters which remain constant or vary slowly with respect to time. Where variations are rapid and/or more frequent, the basic estimator may not respond swiftly enough, thus a further 'rough' estimator is appended. The purpose of this second estimator is to choose the best model, from a given set of possible models, such that the primary estimation procedure may be carried out by using this best form.

The standard Kalman filter equations, Åström and Eykhoff (1971), whereby the 'states' in the filter are thought of as being the unknown system parameters, can however be adjusted, with a restriction of one possible model, by the inclusion of a variable forgetting factor, which decreases to counteract a low Kalman gain value. The problem arises, though, that when the model itself is a poor fit, the forgetting factor will remain at a relatively low magnitude and therefore the parameter estimates, obtained from the estimation scheme, may be poor.

Thus, prior to controller operation a set of possible models, which is considered to be known, is specified such that the estimator operates as follows.

The process is described by the equation,

$$A(z^{-1})y(t) = z^{-k}B(z^{-1})u(t) + C(z^{-1})e(t) \qquad (4.3.1)$$

where $A(z^{-1})$ is monic, and $C(z^{-1})$ is defined initially to be unity. The parameters contained in the respective polynomials $A(z^{-1})$ and $B(z^{-1})$ are now time varying and it is assumed that the parameter vector (state) is defined as,

$$\theta^T(t) = [a_1(t), a_2(t), \ldots, a_{n_a}(t),\ b_0(t), b_1(t), \ldots, b_{n_b}(t)] \qquad \ldots \qquad (4.3.2)$$

where $a_1(t)$, $a_2(t), \ldots, b_0(t), b_1(t), \ldots$, are the parameters of the respective polynomials in (4.3.1) which are of order n_a and n_b. This parameter vector is then estimated by means of the primary estimator, to obtain $\hat{\theta}(t)$ at each sampling period.

The set of possible θ at time t is now introduced as $\{\theta_i : i = 1, \ldots, N\}$. Having taken input and output signal samples up to and including those at time t it is therefore possible to test the residuals obtained for each particular θ_i incorporation. Schemes for carrying this out include a Bayesian probabilistic rule or for easier computation the maximization of a log likelihood function. Both methods need to incorporate an updated window, such that only the most recent sampled values are considered, the actual window size being dependent on how fast the system parameters are likely to vary.

At each iteration of this rough estimator, the best θ_i obtained is fed through to form the framework for θ in the next primary estimator calculation. If the θ_i fed forward remains constant over a large time period this means that the primary estimator is able to cope with the parameter variations and hence the system nonlinearities are not as severe as was expected. The effect of this estimation scheme on the state space controller model is to change the dimension of the state space accordingly, although as this is dependent on $\max\{n_a, n_b\}$ rather than $n_a + n_b$, as is the case with $\hat{\theta}$, no change in dimension will take place if $\min\{n_a, n_b\}$ is affected by a new θ_i unless $\min\{n_a, n_b\} = \max\{n_a, n_b\}$.

One distinct advantage of this type of estimation procedure is that where the process is redefined, by internal alterations, the controller need not be reformulated, as the rough estimator will allow for this permanent change.

3. Nonlinear model control

The inclusion of nonlinearities into the system under control have been limited, to date, to minimum variance controllers, the state space equivalent of which having been formulated in Sec.2.4 as a special case of the class of self-tuners in which a weighting is placed on the control input as well as the system output.

By using the linear system description (4.3.1), the input u(t) is now considered to be obtained as,

$$u(t) = \mu_0 + \mu_1\bar{u}(t) + \mu_2\bar{u}^2(t) + \ldots + \mu_N\bar{u}^N(t) \qquad (4.3.3)$$

where $\bar{u}(t)$ is the actual control input to the plant. Rewriting the above equation as,

$$u(t) = \underset{\sim}{\mu}\underset{\sim}{\bar{u}}(t) \qquad (4.3.4)$$

the overall system may be modelled by,

$$A(z^{-1})y(t) = z^{-k}B(z^{-1})\underset{\sim}{\mu}\underset{\sim}{\bar{u}}(t) + C(z^{-1})e(t) \qquad (4.3.5)$$

where the usual definitions (see(1.3.14) for example) apply to the constituent polynomials.

The estimation model is then extended to include the larger number of parameter estimates, such that the final control law for the minimum-variance case is obtained from (1.5.5) as,

$$E(z^{-1})B(z^{-1})\underset{\sim}{\mu}\underset{\sim}{\bar{u}}(t) = -G(z^{-1})y(t) \qquad (4.3.6)$$

where $E(z^{-1})$ and $G(z^{-1})$ are evaluated from the identity,

$$C(z^{-1}) = A(z^{-1})E(z^{-1}) + z^{-k}G(z^{-1}) \qquad (4.3.7)$$

Thus the final control input, $\bar{u}(t)$, is given (4.3.6) as a polynomial of order N, and this necessitates the inclusion of a root extraction algorithm before the control can be applied, and thus N is limited to an odd value such that at least one real root is assured.

This method of dealing with nonlinearities is, at present, limited to use with minimum-variance tuners, although set-point inputs can be allowed for. Despite the fact that minimum-variance control can be obtained via the state space, the advantages of the latter approach lie more in the pole placement field, and therefore the employment of such a nonlinear technique via the state space will lead to no foreseeable improvement.

4.4. Ill-Defined System Models

This section briefly considers the effect of either under or over specification of the system order. When the model, to be used for estimation and control purposes, is decided upon, the order of the model, and thus the dimension of the state space, is usually assumed to be equal to the order of the process to be controlled. But this may well not be the case, especially if little knowledge concerning the process is available. This can be remedied to an extent by the use of a two level estimator, discussed in the preceding section, whereby the higher level decides upon the model order and the lower level estimates the parameters within that bound. But this provides extra computation and as only a finite number of apriori models may be chosen from, the actual process order may still fall outside these available models. Thus consideration must be taken of the effects of using

a model of different order to that of the process itself.

The problem of employing a model which is much simpler than the actual process, an underspecified model, has been found to affect the control provided to only a very limited extent, Åström (1980). Even when the model is several orders lower than the process, because of the self-tuning action, the range over which closed loop stability is highly dependent on a correct model choice is very small. This is, of course, a large bonus for self-tuning controllers, due to the requirement for a simple computational algorithm which necessarily results from a simple model structure. Significantly, in the case of pole placement state space self-tuners, the use of linear output feedback, Sec.4.1, introduces a negligible amount of extra computation when low order models are used, provided that this order is large enough to encompass the specified pole polynomial.

Overspecification of the model order is generally the rule, rather than the exception, in identification procedures, where it is assumed that the model chosen to be used for the system parameter estimation is sufficient to allow for the actual parameters to be estimated. The excess estimates then tend to zero, which means that it is only the control law itself which may be affected under these circumstances. The major problem lies in the inversion, by whatever means, of a coefficient matrix, e.g. the W matrix in Sec. 2.5. Where a column and row of a matrix have as their maximum coefficient a factor tending to zero, a lower bound must be included such that when this is violated by the maximum coefficient the order of the matrix is reduced by unity. This is of special importance when a variable integer time delay k is

allowed for in pole placement self-tuners by the selection of k_{max}. If the actual time delay of the process is less than k_{max}, (k_{max}-k)higher order parameters of the $\hat{B}'(z^{-1})$polynomial will tend to zero, and thus a reduction in the dimension of the state space may be necessary.

4.5 Concluding Remarks

For high order system models the use of linear output feedback, and hence the calculation of G^* can prove to be computationally inefficient, but for relatively low order models the extra computational requirements are minimal in relation to the total necessary in the entire algorithm. This increase in computing time is, however, rewarded by a much improved value of output variance when compared to that obtained if state feedback alone is used. One of the advantages of the state feedback method was that output spikes, occurring when changes in process integer time delay appear, are non existent. This is lost when ℓ.o.f. is used, although the magnitude of the spikes may be reduced by the use of a more active variable forgetting factor. In fact, the variable forgetting factor employed in the simulations of Sec.4.1(6) was of the type proposed by Wellstead and Sanoff (1981), as this was found to be computationally far less time consuming than that of the stochastic version of Cordero and Mayne (1981), although an equivalent factor can result, Omani (1981).

Ultimately, though, the best overall performance would be achieved by the use of ℓ.o.f. under steady conditions, in combination with state feedback, and a cessation of the ℓ.o.f. content when either

a change in time delay occurs or if the model estimates vary rapidly. The problem than arises, however, of detecting changes in the system parameters, although the variable forgetting factor, which decreases rapidly due to a large estimation error, provides one possible measurement.

The problems found in the extention of state space self-tuning to multivariable control systems are centred around the non-commutative property when the multiplication of constituent matrices occurs. However, if a suitable state form, i.e. reconstructibility canonical form, is obtained, this problem is then removed. There still remain several possible ways of obtaining the state feedback parameters via the model described, and this remains as the subject for further research to discover the benefits or detractions of each particular approach.

The state equations used as a starting point for the multivariable case were merely an extension to those employed in the SISO problem, and other representations must therefore not be discounted. One such possibility is the formulation of the state vector by a stacking of past input and output signal values, although this method really represents another way of writing the polynomial equations in matrix form, and hence has not been dealt with in this section.

Due to the use of a simple linear model as a basis for self-tuning controllers, the incorporation of a system which is actually non linear has not, to the present, been considered in great detail. Using a two-level estimator, Sec.4.3(2), has no effect on the final model type, merely on the number of parameters it must incorporate.

Finally, as models, may be employed in self-tuning controllers, which are much simpler than the actual process under control, the

overall computational effort remains small, and hence suitable for real-time microprocessor implementation. As far as model overspecification is concerned, cancelling modes in the open loop system transfer function can lead to an estimation procedure which appears to be overspecified, even though theoretically it is not. Where the cancelling mode lies within the unit circle of the z-plane, stability of the overall closed loop system is not impaired, although unstable, uncontrollable modes can give problems, dependent on the type of control action selected.

CHAPTER 5

FILTERING AND OBSERVER THEORY

Most of the work in the previous chapters has been concerned principally with the formulation of the desired control law, the system state being regarded as known or, at worst, it may be estimated. However, where a state estimation scheme was discussed, Sec.2.3, the underlying filtering techniques used were glossed over simply by looking at the problem as one of back substitution in the state equations.

In this chapter it is intended to reconsider the estimation scheme employed and by doing so to investigate the type of observer necessary to obtain the estimate from system output and control input values. The observer form used is standard, Kwakernaak and Sivan (1972), and defines the basis of other filtering configurations, Vathsal (1980), Warwick (1982). From this starting point the estimator is found, as if the parameters of the system were known, such that in its final format the oberver is considered optimal in the sense that the mean square error in state reconstruction is minimized.

In Sec.5.1, the optimal observer, also known as the Kalman-Bucy filter, is found by using only past values of input and output signals. This provides the state estimate of Sec.2.3, which was later used in self-tuning, specifically pole placement, design. A modified estimate is then looked at in Sec.5.2 by including the present value of output signal in the observer equations. However, the optimal

observer found by this approach is not the one used in other self-tuning systems, and hence in Sec.5.3, a modified non-optimal observer is obtained, which meets the necessary requirement.

The stability and reconstruction error properties of each of the observer forms are discussed, and the possibility of obtaining the modified forms from the original estimate of Sec.5.1 is shown.

5.1 State Reconstruction for the Primary Model

1. Observer formulation

The state estimation procedure, formulated in Sec.2.3, used simple back substitution by means of the state equations in order to arrive at its final form. Here the underlying observer theory connected with such an estimation scheme is discussed.

The state equationsare given in the discrete time by:

$$\left.\begin{aligned} &(a) \quad \underline{x}(t+1) = P\underline{x}(t) + Qu(t) + Re(t) \\ &(b) \quad y(t) \quad = H\underline{x}(t) + e(t) \end{aligned}\right\} \qquad (5.1.1)$$

where the constituent matrices, for this particular case, are defined in sec. 2.2(2), and R e(t) is the state excitation or process noise, whereas e(t) is the the measurement noise. Hence, as the process noise is a linear combination of the measurement noise, the process is singular.

If the system in (5.1.1) is regarded to be of order n, where n is redefined for this chapter, then a full-order observer for the system, is obtained from the n-dimensional equation,

$$\hat{\underline{x}}(t+1) = P\hat{\underline{x}}(t) + Qu(t) + K[y(t) - H\hat{\underline{x}}(t)] \qquad (5.1.2)$$

on condition that $\hat{\underline{x}}(t_0) = \underline{x}(t_0)$, from which it follows that $\hat{\underline{x}}(t) = \underline{x}(t)$; $t \geq t_0$ for all u(t), $t \geq t_0$; Kwakernaak and Sivan (1972).

Previously, Sec. 2.3, it was shown that by setting K = R we obtain,

$$\hat{\underline{x}}(t+1) = [I-z^{-1}\bar{P}]^{-1}\{Ry(t) + Qu(t)\} \tag{5.1.3}$$

where $\bar{P} = P - RH.$ (5.1.4)

This is now examined from a filtering point of view by considering the discrepancy between (5.1.1) and (5.1.2). If the state, $\underline{x}(t)$, and the reconstructed state, $\hat{\underline{x}}(t)$, are identical, these equations are the same. But where they differ it is necessary that some measure of the error between them is taken.

Let this error between the state vectors, known as the reconstruction error, be $\Delta(t)$, when the error is considered at time t, and thus

$$\Delta(t) = \underline{x}(t) - \hat{\underline{x}}(t) \tag{5.1.5}$$

Then, from (5.1.1a) and (5.1.2),

$$\Delta(t+1) = P\Delta(t) + Re(t) - Ky(t) - KH\hat{\underline{x}}(t)$$

and using (5.1.1b),

$$\Delta(t+1) = [P - KH]\Delta(t) + [R - K]e(t). \tag{5.1.6}$$

Thus the error is defined by a linear stochastic difference equation, the mean value of which is given by,

$$E\{\Delta(t+1)\} = [P - KH]E\{\Delta(t)\} \tag{5.1.7}$$

where $E\{\cdot\}$ denotes the expected value, which we wish to be zero in the case of $\Delta(t)$, i.e. $E\{\Delta(t)\} = 0.$ (5.1.7a)

The following definitions are then made.

$$\left.\begin{array}{ll} \text{(a)} & E\{Re(t),\ e^T(t)R^T\} = R\Omega R^T \\ \text{(b)} & E\{Re(t),\ e^T(t)\ \} = R\Omega \\ \text{(c)} & E\{e(t),\ e^T(t)R^T\} = \Omega R^T \\ \text{(d)} & E\{e(t),\ e^T(t)\ \} = \Omega \end{array}\right\} . \qquad (5.1.8)$$

We can now proceed in the investigation of the reconstruction error variance, denoted by $\emptyset(t)$, at time t, and defined by,

$$\emptyset(t) = E\{\Delta(t) - E\{\Delta(t)\}\}\{\Delta(t) - E\{\Delta(t)\}\}^T \qquad (5.1.9)$$

and using (5.1.7a)

$$\emptyset(t) = E\{\Delta(t),\ \Delta^T(t)\} \qquad (5.1.9a)$$

By substitution of (5.1.6) into this equation,

$$\emptyset(t+1) = E\{(P-KH)\Delta(t) + (R-K)e(t)\}\{(P-KH)\Delta(t)+(R-K)e(t)\}^T \qquad (5.1.10)$$

or $$\emptyset(t+1) = E\{(P-KH)\Delta(t) + (R-K)e(t)\}\{\Delta^T(t)(P-KH)^T + e^T(t)(R-K)^T\}$$

which may be expanded such that,

$$\begin{aligned}\emptyset(t+1) = {} & E\{(P-KH)\Delta(t)\ \Delta^T(t)(P-KH)^T\} + E\{Re(t)e^T(t)R^T\} \\ & -E\{Ke(t)e^T(t)R^T\} - E\{Re(t),e^T(t)K^T\} \\ & +E\{Ke(t),\ e^T(t)K^T\} \end{aligned} \qquad (5.1.11)$$

all other terms being equal to zero.

By rewriting (5.1.11) in terms of the identities (5.1.8), we have

$$\emptyset(t+1) = (P-KH)\emptyset(t)(P-KH)^T + R\Omega R^T - K\Omega R^T - R\Omega K^T + K\Omega K^T \qquad (5.1.12)$$

It is now desired to make a choice of the unknown K in order that (5.1.12) is in some sense optimal. As this error has zero mean it is therefore left to select K such that the reconstruction error variance is minimized. This is known as the optimal observer problem.

Rearranging (5.1.12),

$$\begin{aligned}\emptyset(t+1) = {} & P\emptyset(t)P^T + R\Omega R^T - [R\Omega + P\emptyset(t)H^T]K^T \\ & -K[\Omega R^T + H\emptyset(t)P^T] + K[\Omega + H\emptyset(t)H^T]K^T\end{aligned}$$

and completing the squares leads to

$$\begin{aligned}\emptyset(t+1) = {} & P\emptyset(t)P^T + R\Omega R^T - [R\Omega + P\emptyset(t)H^T][\Omega + H\emptyset(t)H^T]^{-1}_x \\ & [\Omega R^T + H\emptyset(t)P^T] + \{K - [R\Omega + P\emptyset(t)H^T][\Omega + H\emptyset(t)H^T]^{-1}\}_x \\ & [\Omega + H\emptyset(t)H^T]\{K - [\Omega R^T + H\emptyset(t)P^T]^T[\Omega + H\emptyset(t)H^T]^{-1}\}^T \end{aligned} \quad \ldots \quad (5.1.13)$$

Thus if $\bar{\emptyset}(t+1) = \min_K\{\emptyset(t+1)\}$ and $K = \bar{K}$ to give $\bar{\emptyset}(t+1)$, from (5.1.13),

$$\bar{K} = [R\Omega + P\emptyset(t)H^T][\Omega + H\emptyset(t)H^T]^{-1} \quad (5.1.14)$$

and it follows that,

$$\left.\begin{aligned} \bar{\emptyset}(t+1) &= P\bar{\emptyset}(t)P^T + R\Omega R^T - \bar{K}[\Omega R^T + H\bar{\emptyset}(t)P^T] \\ \text{or} \quad \bar{\emptyset}(t+1) &= [P - \bar{K}H]\bar{\emptyset}(t)P^T + [R - \bar{K}]\Omega R^T \end{aligned}\right\} \quad (5.1.15)$$

where $[\Omega + H\emptyset(t)H^T]$ is non-singular, and therefore invertible for this to hold.

In its present form (5.1.14) does not present a simple equation from which $\bar{K}$ may be evaluated. But if the steady-state filter format is desired, the sequence of error covariance matrices $\{\bar{\emptyset}(t)\}$ converge to the null(zero) matrix as $t \to \infty$. For this to occur, however, the $C(z^{-1})$ polynomial, whose parameters are present in the R matrix, must contain all stable zeros, Caines (1972), the reason for this will become clear shortly.

With the limitation that $\bar{\emptyset}(t) \to 0$ as $t \to \infty$, (5.1.14) is much simplified to,

$$\bar{K} = R \quad (5.1.16)$$

which means that if K in (5.1.2) is set equal to R, the form becomes that of a steady state optimal observer.

By using the value of $K = \bar{K}$ throughout, (5.1.15) reduces to,

$$\bar{\emptyset}(t+1) = [P - RH]\bar{\emptyset}(t)P^T \quad (5.1.17)$$

The final filter equation being in 'Kalman-Bucy' form, and is the minimum mean square linear estimator, Åström (1970), such that (5.1.2) may now be written as,

$$\underline{\hat{x}}(t+1) = [P - RH]\underline{\hat{x}}(t) + Qu(t) + Ry(t) \qquad (5.1.18)$$

which is equivalent to (5.1.3).

2. Observer stability

The stability of the observer, (5.1.18) is determined by the relationship between $\Delta(t)$ and $\Delta(t+1)$, which gives, among other things, its speed of response.

By considering the deterministic case of (5.1.6),

$$\Delta(t+1) = [P - KH]\Delta(t) \qquad (5.1.19)$$

and it is required that $\Delta(t) \to 0$ as $t \to \infty$ for any $\Delta(t_0)$, on condition that the observer is asymptotically stable. The characteristic values of $[P - KH]$ are called the observer poles, such that by making K very large a rapid reconstruction error convergence to zero is achieved, although the filter then becomes far more sensitive to measurement error. Hence the optimal filter choice for K, $\bar{K}$, found in (5.1.14), which trades off between the two cases.

In order to investigate the observer poles, consider the three dimensional system where ,

$$K = \begin{pmatrix} k_3 \\ k_2 \\ k_1 \end{pmatrix} \; ; \quad H = [0 \quad 0 \quad 1] \; ; \quad P = \begin{pmatrix} 0 & 0 & -a_3 \\ 1 & 0 & -a_2 \\ 0 & 1 & -a_1 \end{pmatrix}$$

It follows that $P-KH = \begin{pmatrix} 0 & 0 & -(a_3+k_3) \\ 1 & 0 & -(a_2+k_2) \\ 0 & 1 & -(a_1+k_1) \end{pmatrix}$

from which the characteristic equation is given by the determinant of

$$\begin{pmatrix} z & 0 & (a_3+k_3) \\ -1 & z & (a_2+k_2) \\ 0 & -1 & z+(a_1+k_1) \end{pmatrix}$$

which is

$$z^3 + (a_1+k_1)z^2 + (a_2+k_2)z + (a_3+k_3)$$

In the optimal case, when $K = \bar{K} = R$, we have that,

$$k_i = c_i - a_i \; ; \quad i = 1,2,3.$$

Thus the observer poles are found from the equation,

$$z^3 + c_1z^2 + c_2z + c_3 = 0.$$

Hence these poles are governed completely by the $C(z^{-1})$ polynomial, which must be stable for the sequence $\{\bar{\emptyset}(t)\}$ to converge to zero as $t \to \infty$.

3. Stability during parameter estimation

As the observer poles are also those of the $C(z^{-1})$ polynomial, this has direct relevance to the type of parameter estimation procedure carried out. Where the recursive least squares method is employed, it is assumed that the $C(z^{-1})$ polynomial is equal to unity, i.e.

$c_1 = c_2 = \ldots = c_{n_c} = 0.$

Thus the characteristic equation becomes, under these conditions

$$z^{n'} = 0.$$

Therefore, all the observer poles may now be regarded as lying at the origin of the z-plane.

Hence, every initial value of reconstruction error is reduced to zero in a maximum of n'steps, where n' is the state space dimension (given as three in the previous example). It follows that the observer becomes what is known as a dead-beat observer under the condition of recursive least squares parameter estimation.

5.2 Modified State Reconstruction : Part I, Optimal Observer

1. Estimator formulation

Although the values of output and input at time t, y(t) and u(t), respectively, were introduced into the observer equation (5.1.2), because of the delay operator in (5.1.3), the state estimate at time t is only dependent on the terms u(t-1), u(t-2),...,y(t-1),y(t-2),... . Hence $\underline{\hat{x}}(t)$ may be written $\underline{\hat{x}}(t/t-1)$, i.e. the estimate of the state at time t, dependent on conditions up to and including those at time t-1.

Thus a controller obtained by employing this state estimate, provides a feedback loop in which y(t), although available, is not apparent. It was shown in Chapter 2 that several polynomial self-tuners use the equivalent of a state estimate form in which y(t) appears, and this alternative method was derived as a modification to the state estimate (5.1.3).

In this section an observer in which the next output value appears will be derived from first principles. The full order observer equation is now, however,

$$\underline{\hat{x}}(t+1) = [I - KH][P\underline{\hat{x}}(t) + Qu(t)] + Ky(t+1) \qquad (5.2.1)$$

where $\underline{\hat{x}}(t) = \underline{\hat{x}}(t/t)$.

Substituting for this state estimate in the state equations (5.1.1), the reconstruction error is

$$\Delta(t+1) = [I - KH]P\Delta(t) + [I - KH]Re(t) - Ke(t+1) \qquad (5.2.2)$$

in which $\Delta(t) = \underline{x}(t) - \underline{\hat{x}}(t)$.

Once more the variance of this reconstruction error may be obtained from,

$$\emptyset(t) = E\{\Delta(t)\ ,\ \Delta^T(t)\}$$

as
$$\begin{aligned}\emptyset(t+1) = {} & [P - KHP]\emptyset(t)[P - KHP]^T \\ & +[I - KH]R\Omega R^T[I - KH]^T + K\Omega K^T \\ & -[I - KH]R\Omega K^T - K\Omega R^T[I - KH]^T \ .\end{aligned}$$

To simplify this somewhat, into a form from which it will be possible to obtain $K = \bar{K}$, the following definition is made.

$$\text{Let } \gamma(t+1) = P\emptyset(t)P^T + R\Omega R^T \qquad (5.2.3)$$

Then it follows that,

$$\begin{aligned}\emptyset(t+1) = {} & \gamma(t+1) - [\gamma(t+1)H^T + R\Omega][H\gamma^T(t+1) + \Omega R^T]_x \\ & [H\gamma(t+1)H^T + HR\Omega + \Omega R^T H^T + \Omega]^{-1} \\ & +\{K - [\gamma(t+1)H^T + R\Omega][H\gamma(t+1)H^T + HR\Omega + \Omega R^T H^T + \Omega]^{-1}\}_x \\ & [H\gamma(t+1)H^T + HR\Omega + \Omega R^T H^T + \Omega]_x \\ & \{K - [H\gamma(t+1) + \Omega R^T]^T [H\gamma(t+1)H^T + HR\Omega + \Omega R^T H^T + \Omega]^{-1}\}^T\end{aligned}$$

Thus the minimum value of $\emptyset(t+1)$ is found when,

$$K = \bar{K} = [\gamma(t+1)H^T + R\Omega][H\gamma(t+1)H^T + HR\Omega + \Omega R^T H^T + \Omega]^{-1} \qquad (5.2.4)$$

and hence ,

$$\bar{\emptyset}(t+1) = \gamma(t+1) - \bar{K}[H\gamma(t+1) + \Omega R^T]$$

or
$$\bar{\emptyset}(t+1) = [I - \bar{K}H]\gamma(t+1) - \bar{K}\Omega R^T \qquad (5.2.5)$$

where
$$\bar{\emptyset}(t) = \min_{K}\{\emptyset(t)\}.$$

The steady-state filter is now required, such that $\bar{\phi}(t) \to 0$ as $t \to \infty$, and by substituting for $\phi(t) = 0$ in (5.2.3) we obtain

$$\gamma(t+1) = R\Omega R^T$$

from which, $\bar{K} = [RR^TH^T + R][HRR^TH^T + HR + R^TH^T + 1]^{-1}$

considering an n dimensional system,

$$HRR^TH^T = (c_1 - a_1)^2 \; ; \quad HR = R^TH^T = (c_1 - a_1)$$

thus $$[HRR^TH^T + HR + R^TH^T + 1]^{-1} = (1 - a_1 + c_1)^{-2} = \bar{E}^{-2}$$

also $$R[R^TH^T + 1] = R(1 - a_1 + c_1) = R\bar{E}$$

Therefore, in general, $\bar{K} = R\bar{E}^{-1}$ (5.2.6)

The steady-state optimal observer for this modified state estimator can now be written as,

$$\underline{\hat{x}}(t+1) = [I - RH\bar{E}^{-1}][P\underline{\hat{x}}(t) + Qu(t)] + R\bar{E}^{-1}y(t+1) \qquad (5.2.7)$$

and defining, $P^* = [I - RH\bar{E}^{-1}]P$, the estimate becomes

$$\underline{\hat{x}}(t) = [I - z^{-1}P^*]^{-1}\{[I - RH\bar{E}^{-1}]Qu(t-1) + R\bar{E}^{-1}y(t)\} \qquad (5.2.8)$$

which is in the same form as (5.1.3), although in this case the estimator at time t contains the present output value, y(t).

2. Estimator equation

In Sec. 2.3, the state estimate $\underline{\hat{x}}(t/t-1)$ was found in terms of parameters of the estimated $A(z^{-1})$ and $B(z^{-1})$ polynomials, $C(z^{-1})$ having been set equal to unity due to the use of a recursive least squares procedure. It is now desired to find a similar form for $\underline{\hat{x}}(t/t)$.

Consider the three dimensional case, where

$$P = \begin{pmatrix} 0 & 0 & -a_3 \\ 1 & 0 & -a_2 \\ 0 & 1 & -a_1 \end{pmatrix} \quad ; \quad R' = \begin{pmatrix} -a_3 \\ -a_2 \\ -a_1 \end{pmatrix} \quad ; \quad H^T = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \quad ; \quad Q = \begin{pmatrix} b_2 \\ b_1 \\ b_0 \end{pmatrix}$$

where R' is the R vector in which all the parameters of the $C(z^{-1})$ polynomial have been set to zero.

Then, from (5.2.8),

$$\hat{\underline{x}}(t) = \frac{1}{\{z+A(z^{-1})(1-z)\}} \left\{ \begin{pmatrix} b_2 \\ b_1 + b_2 z^{-1} \\ b_0 + b_1 z^{-1} + b_2 z^{-2} \end{pmatrix} u(t-1) + \right.$$

$$\left. \begin{pmatrix} -a_3 \\ -a_2 - a_3 z^{-1} \\ -a_1 - a_2 z^{-1} - a_3 z^{-2} \end{pmatrix} y(t) + (1-z^{-1}) \begin{pmatrix} (a_3 b_0 - a_1 b_2) + z^{-1}(a_3 b_1 - a_2 b_2) \\ (a_2 b_0 - a_1 b_1) + z^{-1}(a_3 b_0 - a_1 b_2) \\ 0 \end{pmatrix} u(t-1) \right\}$$

... (5.2.9)

where $A(z^{-1}) = 1 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3}$.

For the full calculation of $\hat{\underline{x}}(t)$ from (5.2.8), see Appendix 5.1.

The expression (5.2.9) is similar to that for $\hat{\underline{x}}(t/t-1)$, see Sec. 2.3, the differences being (a) y(t) is now included instead of y(t-1), (b) the denominator is no longer unity, (c) an extra vector operator on u(t-1) has appeared.

Both estimators, though, may be written in the form,

$$\hat{\underline{x}}(t) = (\det)^{-1}[Mu(t-1) + Ny(t)] \qquad (5.2.10)$$

such that for any n-dimensional system, if M and N are found for $\hat{\underline{x}}(t/t)$:

$$N = \begin{bmatrix} & & & -a_n \\ & & -a_{n-1} & -a_n z^{-1} \\ & \cdot \quad \cdot & & \cdot \\ & \cdot \quad \cdot & & \cdot \\ & \cdot \quad \cdot & & \cdot \\ -a_2 & - \ . \ . \ . \ . \ . \ . \ . & & -a_n z^{-n+2} \\ -a_1 & -a_2 z^{-1} \ . \ . \ . \ . \ . \ . \ . & & -a_n z^{-n+1} \end{bmatrix} \qquad (5.2.11)$$

and $$M = M_1 + M_2 \qquad (5.2.12)$$

where,

$$M_1 = \begin{bmatrix} & & & b_{n-1} \\ & & b_{n-2} & +b_{n-1} z^{-1} \\ & \cdot \quad \cdot & & \cdot \\ & \cdot \quad \cdot & & \cdot \\ & \cdot \quad \cdot & & \cdot \\ b_1 & + \ . \ . \ . \ . \ . \ . & & +b_{n-1} z^{-n+2} \\ b_0 & +b_1 z^{-1} + . \ . \ . \ . \ . \ . & & +b_{n-1} z^{-n+1} \end{bmatrix} \qquad (5.2.13)$$

and,

$$M_2 = (1-z^{-1}) \begin{bmatrix} (a_n b_0 - a_1 b_{n-1}) + z^{-1}(a_n b_1 - a_2 b_{n-1}) + \ldots + z^{-n+2}(a_n b_{n-2} + a_{n-1} b_{n-1}) \\ \vdots \\ (a_2 b_0 - a_1 b_1) + z^{-1}(a_3 b_0 - a_1 b_2) + \ldots \quad + z^{-n+2}(a_n b_0 - a_1 b_{n-1}) \\ 0 \qquad \ldots \end{bmatrix} \qquad (5.2.14)$$

However, as this latter matrix is not a direct stacked vector as in the previous cases, an example for a five dimensional system is obtained as,

$$M_2 = (1-z^{-1})\begin{Bmatrix} (a_5b_0-a_1b_4) + z^{-1}(a_5b_1-a_2b_4) + z^{-2}(a_5b_2-a_3b_4) + z^{-3}(a_5b_3-a_4b_4) \\ (a_4b_0-a_1b_3) + z^{-1}(a_4b_1-a_2b_3) + z^{-2}(a_4b_2-a_3b_3) + z^{-3}(a_5b_2-a_3b_4) \\ (a_3b_0-a_1b_2) + z^{-1}(a_3b_1-a_2b_2) + z^{-2}(a_4b_1-a_2b_3) + z^{-3}(a_5b_1-a_2b_4) \\ (a_2b_0-a_1b_1) + z^{-1}(a_3b_0-a_1b_2) + z^{-2}(a_4b_0-a_1b_3) + z^{-3}(a_5b_0-a_1b_4) \\ 0 \end{Bmatrix}$$

The general rule for building up M_2 can be stated as follows. If every term in each row with a common delay operator forms a subcolumn, such that in the example above there are four subcolumns, then by starting from any term in the lowest non-zero row, to obtain the term in the row above which remains in the same subcolumn, i.e. has the same delay index, unity must be added to the lowest index in either parameter multiplication, until the right hand term has identical indices. If this latter case is apparent, unity is added to the b_i index in the right hand term and to the a_i index in the left hand parameter multiplication.

e.g. if we have $a_4b_i - a_{i+1}b_3$, the term in the row above is, $a_4b_{i+1} - a_{i+2}b_3$, unless $i+1 = 3$, in which case the term in the row above is $a_5b_i - a_{i+1}b_4$, for subcolumn 3.

The final form of (5.2.10) for $\underline{\hat{x}}(t/t)$ will now be compared with that for the state estimate $\underline{\hat{x}}(t/t-1)$. By defining,

$$\underline{\hat{x}}(t/t-1) = (\overline{det})^{-1}[\bar{M}u(t-1) + \bar{N}y(t)] \qquad (5.2.15)$$

the following equalities arise :

1) $\overline{det} = 1$: $det = z + (1-z)A(z^{-1})$.

2) $\bar{M} = M_1$ (or $\bar{M} = M$ where M_2 is the null matrix).

3) $\bar{N} = z^{-1}N$

Because of the difference between the determinants given in

equality (1), a simple calculation of the state estimate at time t from the general form of (5.2.10) is not as obvious as that obtained from (5.2.15). However the estimator $\hat{\underline{x}}(t/t)$ may be rewritten as,

$$\{z+(1-z)A(z^{-1})\}\hat{\underline{x}}(t/t) = Mu(t-1) + Ny(t) \qquad (5.2.16)$$

where $\det(I-z^{-1}P^{*}) = z + (1-z)A(z^{-1})$.

This method of obtaining the state implies that past values of $\hat{\underline{x}}(t/t)$ are required, a feature which is not desirable because of problems due to unknown initial state values. The control input form (5.2.17) is therefore introduced, such that

$$u(t) = F\hat{\underline{x}}(t/t) = \sum_{i=1}^{n} f_i \hat{x}_i(t/t) \qquad (5.2.17)$$

for an n-dimensional state vector.

Let $$\hat{\underline{x}}(t/t) = \{z + (1-z)A(z^{-1})\}^{-1} \hat{\underline{x}}'(t/t) \qquad (5.2.18)$$

then $$\{z + (1-z)A(z^{-1})\}u(t) = F\hat{\underline{x}}'(t/t) \qquad (5.2.19)$$

which follows from the scalar nature of the determinant.

From (5.2.19) by expansion of the determinant,

$$(1-a_1)u(t) = (a_2-a_1)u(t-1)+\ldots+(a_n-a_{n-1})u(t-n+1) - a_n u(t-n) + \sum_{i=1}^{n} f_i \hat{x}_i'(t/t) \qquad (5.2.20)$$

or by setting $\hat{\underline{x}}''(t/t) = (1-a_1)^{-1}\hat{\underline{x}}'(t/t)$

$$u(t) = (1-a_1)^{-1}\{(a_2-a_1)u(t-1)+\ldots+(a_n-a_{n-1})u(t-n+1)-a_n u(t-n)\} + \sum_{i=1}^{n} f_i \hat{x}_i''(t/t).$$

Thus the control law defined by (5.2.17) may be formed from,

$$u(t) = Lu(t-1) + F\hat{\underline{x}}''(t/t) \qquad (5.2.21)$$

or $$u(t) = (L + FM')u(t-1) + FN'y(t) \qquad (5.2.22)$$

where $L = (1-a_1)^{-1}\{a_2+a_3z^{-1}+\ldots+a_nz^{-n+2}-a_1-a_2z^{-1}-\ldots-a_nz^{-n+1}\}$

$M' = (1-a_1)^{-1} M$ and $N' = (1-a_1)^{-1} N$.

The stability of this particular control law is obviously dependent on the term $(1-a_1)$, which it must be remembered is only obtained in the case when $C(z^{-1}) = 1$, e.g. when using recursive least squares estimation. Thus iff the estimate a_1 tends to unity, the state estimate, and hence the final control form, will become unstable.

Calculations to obtain the control input for a general $C(z^{-1})$ polynomial are far more complicated, and in practice not necessary as (a) the actual online evaluations only involve those terms included in (5.2.22) when a recursive least squares procedure is employed, and (b) the stability of the observer may be investigated by means of its characteristic equation.

3. Observer stability

From the definition of the reconstruction error (5.2.2), the characteristic equation is obtained from the matrix[I-KH]P, where the optimal observer poles are found if $K = \bar{K} = RE^{-1}$.

Returning to the three dimensional example,

$$[I-KH]P = \begin{pmatrix} 0 & -k_3 & a_1k_3-a_3 \\ 1 & -k_2 & a_1k_2-a_2 \\ 0 & 1-k_1 & a_1k_1-a_1 \end{pmatrix}$$

and thus the characteristic equation is given from the determinant of,

$$\begin{bmatrix} z & k_3 & a_3-a_1k_3 \\ -1 & z+k_2 & a_2-a_1k_2 \\ 0 & k_1-1 & z+a_1-a_1k_1 \end{bmatrix}$$

Hence the observer poles are obtained as the solution to

$$z^3 + z^2(a_1+k_2-a_1k_1) + z(a_2-a_2k_1+k_3) + (a_3-a_3k_1) = 0.$$

If the Kalman gain K, used to obtain minimum reconstruction error variance, denoted by $\bar{K}$, is employed, we have

$$k_1 = \frac{c_1-a_1}{1-a_1+c_1}, \quad k_2 = \frac{c_2-a_2}{1-a_1+c_1}, \quad k_3 = \frac{c_3-a_3}{1-a_1+c_1}$$

and therefore,

$$z^3(1-a_1+c_1) + z^2(a_1-a_2+c_2) + z(a_2-a_3+c_3) + a_3 = 0$$

or $$A(z) - z\{A(z) - C(z)\} = 0$$

Hence the stability of the optimal observer is highly dependent on the E term, where $E = 1-a_1+c_1$, such that in the case $1 - a_1+c_1 = 0$, an observer pole will appear at infinity, irrespective of the remaining terms in the $A(z^{-1})$ and $C(z^{-1})$ polynomials.

The observer stability now depends on conditions other than the $C(z^{-1})$ polynomial, and thus stability can no longer be ensured by the assumption that $C(z^{-1})$ is stable. As stability of the overall controller is an important factor in self-tuning, we are left with the option that if it is desired to form a state estimate using the present output value, y(t), we must consider a non-optimal observer, that is nevertheless stable, provided $C(z^{-1})$ is stable.

5.3 Modified State Reconstruction : Part II Non-Optimal Observer

1. Introduction

The optimal value of K, given as $\bar{K}$, found in the previous section led to an observer in which the variance of the reconstruction error tended to zero in the steady-state only when the observer polynomial was stable. But by using a non-optimal value of K, so that a stable observer is formed, will have a detrimental effect on the properties of the reconstruction error. In Sec. 5.1 an optimal and stable estimator was achieved, subject to $C(z^{-1})$ being stable, although the present value of output signal was not included. In this section the observer polynomial will be chosen to be $C(z^{-1})$ and the effect this choice has on the Kalman gain, K, and the reconstruction error, $\Delta(t)$, will be discussed.

2. Characteristic equation

To arrive at a characteristic equation equivalent to that achieved in Sec. 5.1, two equalities are necessary. Firstly, by considering the $C(z^{-1})$ polynomial to be unity, as when recursive least squares is employed for parameter estimation, the characteristic equation must have all its poles at the origin of the z-plane, thus leading to a dead-beat observer. Secondly, when a general $C(z^{-1})$ polynomial is included, the observer must be such that its characteristic equation can be regarded as being the $C(z^{-1})$ polynomial itself.

Extrapolating from the characteristic equation arrived at in Section 5.2, in its general form this may be written as,

$$z^n + \sum_{i=1}^{n} \{(1-k_1)a_i + k_{i+1}\}z^{n-i} \qquad (5.3.1)$$

for an n-dimensional system, although this leads to a filter term k_{n+1} which will be dealt with shortly.

Setting (5.3.1) to be equal to C(z), where

$$C(z) = z^n + \sum_{i=1}^{n} c_i z^{n-i} \qquad (5.3.2)$$

we have that

$$(1-k_1)a_i + k_{i+1} = c_i \qquad (5.3.3)$$

or $$k_{i+1} = c_i - a_i(1-k_1) \qquad \text{where } i = 1 , \ldots , n.$$

By redimensioning the state space description to (n+1) from n, (5.3.3) may be described as,

$$PK = R \qquad (5.3.4)$$

and using the case n = 3 as an example, this is

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & -a_3 \\ 0 & 1 & 0 & -a_2 \\ 0 & 0 & 1 & -a_1 \end{pmatrix} \begin{pmatrix} k_4 \\ k_3 \\ k_2 \\ k_1 \end{pmatrix} = \begin{pmatrix} 0 \\ c_3-a_3 \\ c_2-a_2 \\ c_1-a_1 \end{pmatrix}$$

from which P is singular, and therefore not invertible, which means that (5.3.4) leads to n+1 unknowns, k_i , with only n equations.

An extra equation is obtained by remembering that iff $C(z^{-1})$ = unity, we desire the observer to have all its poles at the origin. Thus, from (5.3.3),

$$k_{i+1} = -a_i(1-k_1) \qquad \text{for } i = 1,\ldots,n$$

and this is satisfied when $k_1 = 1$, thus $k_{i+1} = c_i$ such that (5.3.3) holds in general.

Hence the overall state-space dimension must be at least of order unity more than the degree of the $C(z^{-1})$ polynomial to allow for the k_{n+1} term. In Sec. 2.2(2) this was defined as being the case, and further the dimension is such that it will be at least k more than the order of the $C(z^{-1})$ polynomial, where $k \geq 1$ is the system integer time delay. Therefore, when $k > 1$ further Kalman filter terms, k_{i+1}, will be present for $i > n$, but these will all be equal to zero as $c_i = 0$ for $i > n$.

In its final form,

$$K^T = [0,\ldots,0,c_{n_c},\ldots,c_1,1] \qquad (5.3.5)$$

and when $c_i = 0 : i = 1,\ldots,n_c$.

$$K^T = H$$

where H was defined in Sec.2.2(2)

3. Properties of non-optimal observer

Having defined the non-optimal, but stable, observer, this will now be compared with the observer defined in Sec.5.1. Thus if the estimated state vector of this modified observer is denoted by $\underline{\hat{x}}'(t)$, at time t, the following theorem is given

Theorem 5.1

The State Reconstruction,

$$\underline{\hat{x}}'(t+1) = [I - K_2H][P\underline{\hat{x}}'(t) + Qu(t)] + K_2y(t+1)$$

where $PK_2 = R$

can be calculated from the state reconstruction,

$$\underline{\hat{x}}(t+1) = P\underline{\hat{x}}(t) + Qu(t) + K_1[y(t) - H\underline{\hat{x}}(t)]$$

where $K_1 = R$

by means of the equation,

$$\hat{\underline{x}}'(t) = \hat{\underline{x}}(t) + \bar{H}^T[y(t) - H\hat{\underline{x}}(t)]$$

where $\bar{H}^T = K_2$

Proof:

Multiplying both sides of the $\hat{\underline{x}}'(t)$ state equation by the matrix P,

$$P\hat{\underline{x}}'(t+1) = [P - RH][P\hat{\underline{x}}'(t) + Qu(t)] + Ry(t+1)$$

It follows that by adding Qu(t) to both sides,

$$[I - z^{-1}\bar{P}][P\hat{\underline{x}}'(t) + Qu(t)] = Qu(t) + Ry(t)$$

where $\bar{P} = P - RH.$

Thus, as $\hat{\underline{x}}(t) = [I - z^{-1}\bar{P}]^{-1}\,[Qu(t-1) + Ry(t-1)]$

$$P\hat{\underline{x}}'(t) + Qu(t) = \hat{\underline{x}}(t+1)\ .$$

Substituting for this back into the $\hat{\underline{x}}'(t)$ state equation

$$\hat{\underline{x}}'(t) = [\,I - K_2H]\hat{\underline{x}}(t) + K_2y(t)$$

or $$\hat{\underline{x}}'(t) = \hat{\underline{x}}(t) + \bar{H}^T[y(t) - H\hat{\underline{x}}(t)]$$

when $\bar{H}^T = K_2$ Q.E.D.

It was shown in Chapter 2, Lemmas 2.2 and 2.4, that the final form for this modified state reconstruction, $\hat{\underline{x}}'(t)$, was that used in the equivalent of the self-tuning controllers based on polynomial principles for either generalized minimum variance or pole placement objectives. Thus the polynomial self-tuners of Clarke and Gawthrop (1975) and Wellstead et al (1979b) use, when a state space format is considered, a non-optimal observer, which is nevertheless stable subject to the $C(z^{-1})$ polynomial being stable. However, as the observer is non-optimal the variance of the reconstruction error must be non-zero, its exact value being considered in the next subsection.

4. Reconstruction error variance

The general expression for the variance of the reconstruction error was given in Sec. 5.2 as,

$$\emptyset(t+1) = [I - KH]P\emptyset(t)P^T[I - KH]^T + [I - KH]R\Omega R^T[I - KH]^T$$
$$+ K\Omega K^T - [I - KH]R\Omega K^T - K\Omega R^T[I - KH]^T$$

and using K from (5.3.5),

$$[I-KH] = \begin{pmatrix} 1 & 0 & \cdots & 0 & 0 \\ & \ddots & & & \vdots \\ 0 & & \ddots & 0 & 0 \\ & & & \vdots & -c_{n_c} \\ & & & 0 & \\ & & & & -c_1 \\ 0 & \cdots & 0 & 0 & \end{pmatrix}$$

Thus

$$[I-KH]P = \begin{pmatrix} 0 & \cdots & 0 & 0 & 0 \\ & \ddots & & \vdots & \vdots \\ 1 & & \ddots & 0 & 0 \\ & & & 0 & \\ 0 & & & \vdots & -c_{n_c} \quad -a_1 c_{n_c} \\ & & & 0 & \vdots \\ & & & & -a_1 c_{n_a-1} - a_{n_a} \\ & & & 1 & -c_1 \quad a_1 c_1 - a_2 \\ 0 & \cdots & 0 & 0 & 0 \quad 0 \end{pmatrix}$$

The terms incorporating the disturbance variance, Ω, can be rewritten as,

$$\Omega[R - (1 + c_1 - a_1)K][R - (1 + c_1 - a_1)K]^T$$

where the factor $E = (1 + c_1 - a_1)$ will affect the variance of the reconstruction error via this secondary term unless, as in the optimal case of Sec. 5.2, $K = RE^{-1} = \bar{K}$.

Hence, although a stable observer is obtained by means of K in (5.3.5), the reconstruction error variance is now strongly dependent on the factor E, and significantly if $E = 0$.

$$\emptyset(t+1) = [I-KH]P\emptyset(t)P^T[I-KH]^T + R\Omega R^T$$

and therefore the variance will diverge such that $\emptyset(t) \to \infty$ as $t \to \infty$. This can be seen from a simple example, using a two dimensional system where only c_1 and a_1 are non-zero, subject to $E = 0$, thus $a_1 = 1 + c_1$

$$[I-KH]P = \begin{bmatrix} -c_1 & a_1c_1 \\ 0 & 0 \end{bmatrix}$$

and $$RR^T = \begin{bmatrix} 0 & 0 \\ 0 & (c_1-a_1)^2 \end{bmatrix}$$

By defining $$\emptyset = \begin{bmatrix} \emptyset_1 & \emptyset_2 \\ \emptyset_3 & \emptyset_4 \end{bmatrix}$$

It follows that,

$$\emptyset(t+1) = \begin{bmatrix} c_1^2\{\emptyset_1(t) - a_1\emptyset_3(t) + a_1^2\emptyset_4(t) - a_1\emptyset_2(t)\} & 0 \\ 0 & \Omega(c_1-a_1)^2 \end{bmatrix}$$

and as $\emptyset_3(t) = \emptyset_2(t) = 0$, from the previous iteration.

$$\emptyset(t+1) = \begin{bmatrix} c_1^2\{\emptyset_1(t) + a_1^2\emptyset_4(t)\} & 0 \\ 0 & \Omega(c_1-a_1)^2 \end{bmatrix}$$

using $a_1 = 1 + c_1$

$$\emptyset(t+1) = \begin{bmatrix} c_1^2\{\emptyset_1(t) + (1+c_1^2)\Omega\} & 0 \\ 0 & \Omega \end{bmatrix}$$

Therefore $\emptyset_1(t) \to \infty$ as $t \to \infty$ subject to $c_1 \neq 0$; and $\emptyset_4(t)$ is equal to the disturbance variance for all $t \geq 0$.

5.4 Concluding Remarks

It has been shown in this chapter that the state estimation procedure employed for the controller of Sec. 2.5(1), incorporates an optimal observer, which, as its characteristic equation can be regarded as the $C(z^{-1})$ polynomial, can also be considered as being stable. However, this was formulated from past values of system input and output signals, the present output value not being included.

By modifying the estimator to include y(t), the optimal observer produced shows a great dependence, in its stability, on actual system parameters, via the factor $1 + c_1 - a_1$, and thus it is not a suitable choice for a final filter form. Therefore, in Sec. 5.3, a non-optimal observer was discussed, its characteristic equation being, once more, equivalent to the $C(z^{-1})$ polynomial, and hence stable. This was, via theorem 5.1, shown to be the form of observer used when self-tuning is carried out by the state-space equivalent of the original polynomial controllers. But, by the theorem of duality, see for example Strejc (1981), the problem of optimal state estimation and the problem of optimal control result in the same form of Riccati equation. Thus, in the case of minimum variance self-tuners, an optimal control law is combined with a non-optimal state estimator to provide overall controller action, and duality does not therefore hold between the estimation and control.

The variance of the reconstruction error, which was considered to be zero, for all t, in Sec. 5.1., has at best a value greater than zero in the modified estimate of Sec. 5.3 when operating on a stochastic process. Under certain conditions this can, in fact, tend to infinity, although where a recursive least squares estimation procedure is used, and hence $c_i = 0 : i = 1,\ldots, n_c$; a finite value will result.

CHAPTER 6

CONCLUSIONS

Throughout this text conclusions have been drawn at the end of each chapter concerning the work discussed in that respective category.

In this chapter, however, the main points from the above sections will be stressed along with general comments concerning the overall topic of self-tuning via the state-space and areas of this in which further work is either foreseen and/or is regarded as being useful.

Due to the requirement for computational simplicity, parameter estimation has been carried out in all simulations, and has been accounted for in the theory, by the recursive least squares procedure. In the first chapter this scheme was considered along with other techniques, which, although they are applicable, add to the total time necessary, at each sampling interval, for the calculation of the overall control law. In many possible fields of application for self-tuning, e.g. Automotive Engineering, the constraint placed on the time taken for each control law evaluation can be critical, hence the need for a simple, but efficient, recursive estimation technique.

The use of the state-space approach in self-tuning not only provides a further insight into the previous theoretical work, but can also lead to several advantages over the polynomial based tuners. One of these is the reduction in computational requirements needed to calculate the desired feedback parameters. Obviously, if an identical control law is formulated by means of its respective state-space and polynomial forms, then, with regard to the estimated

parameters, the number of essential calculations must also be identical. But when this must be done on a computer the equality disappears. This can best be seen by considering the pole placement objective, where, in the state-space method the matrix W must be inverted, Sec. 2.5(1), and this is equivalent in dimension to the size of the state space, $\max\{n_a, n_b+k\}$, in the polynomial method, however, the matrix to be inverted is of dimension n_a+n_b+k-1, i.e. n_d+n_g+1. Thus, unless special notice is taken of the positions of zeros occurring in the latter matrix, which can only be applicable for that particular model, the state-space scheme will, computationally, be shorter, as long as neither n_a nor n_b+k is of value unity. Further, if the model is altered in any way, the inversion routine for the polynomial method must be respectively altered, the state-space inversion routine only needing to be altered if the dimension of the state-space is directly affected. This reduction in computational effort allows for the calculations necessary to formulate an estimate of the state vector, where the actual value is not available.

A second advantage of the state-space scheme over the polynomial case was shown in the simulations of Sec. 2.5(5), in which a change in the integer part of the process time delay occurred half-way through the simulation run. This had no discernable immediate effect on the state-space tuner, but the polynomial tuner produced large spikes in the control input signal which consequently affected the system output. Large and rapid control input variations, one of the major disadvantages of minimum-variance tuners, can cause saturation effects and hence there is the possibility of loss of control for several sampling periods in practical systems, therefore variations of this nature are

not desired. This state of panic in the control input, found in the polynomial controller, when faced with a change in system integer time delay, manifested itself again in the linear output feedback self-tuner of Chapter 4, and hence is attributable to the use of the present value of output signal, $y(t)$, in forming the control input, $u(t)$.

Two points of note arise from this. Firstly, by employing a more responsive variable forgetting factor, the input and, hence, output spikes can be reduced in magnitude, as the recursive estimator responds more swiftly to the effective change in system parameters. This, however, results in a fall off in the value of output variance achieved, and thus robustness is gained at the expense of overall performance achievement. Secondly, the use of the present output value in the control input calculation, whether in the polynomial controller, Wellstead et al (1979b), or in the state-space controller with linear output feedback, Warwick and Westcott (1982), itself is a trade off towards better performance at the expense of robustness.

In several instances in the text it was found necessary to revert from the state-space to a polynomial form for either explanatory or theoretical purposes. One example of this is the proof of the self-tuning property, Sec.3.1, where this was carried out by means of the polynomial form, the state-space method then being shown to be an equivalent representation, and thus the proof for the state-space model was arrived at. The ease of conversion from one form to the other being largely due to the state-space model chosen in Sec. 2.2(2).

Another instance of the use of the polynomial form was in the explanation of multivariable state-space self-tuning. Although

this was dealt with only briefly, because of the non-commutative property applicable to the majority of the constituent matrices, the state-space final form removes itself to such an extent from the original multivariable CARMA model, that it becomes extremely tedious to detect exactly what controller action is being provided. Further, the whole area of multivariable self-tuning theory is rapidly expanding in its own right, as will be mentioned later, and to have delved any deeper into this topic would have diversified greatly from the scope on which this work is based.

The state-space approach to self-tuning presents no problems in the incorporation of an external input signal, provided this is regarded as a set-point which changes, in a step mode, relatively infrequently. It was shown in Sec. 3.3. that the additional computation necessary to normalize the process output, such that in the steady-state it is equivalent to the set-point value, is negligible in comparison with the total requirements. The inclusion of an external input also has no deterimental effect on the use of self - tuning with linear output feedback, Sec. 4.1(3), which was devised to reduce the variance of the process output signal. It was found from the simulations of chapter 4 that by using ℓ.o.f. the tuner becomes less robust to rapid changes in the system parameters, specifically in the form of time delay excursions, but despite this setback the output variance achieved is better than either the polynomial case of Wellstead et al (1979b) or when state-space feedback alone is used, Warwick (1981a). This performance improvement, however, is gained at the expense of an increase in the number of calculations, although, once again, this is negligible in

comparison with the total when low order system models are employed.

The choice of closed loop polynomial, in the pole placement method, will affect the variance of the output signal, and hence the possibility of using poles which tune themselves to provide an improved variance value was discussed in Sec. 3.2. For low order models this proved to be practically useful, as was shown in later simulations, although as the object of pole placement is really to define the response of the system output to a change in set-point value, this is lost with Self-Assigning Poles (SAP). A preferred scheme would be to use the SAP technique in the steady-state, but return to a pre-assigned pole polynomial if and when a change in set-point occurs, the main difficulty being how and when to judge that the steady-state has once more been arrived at, i.e. for what length of time should the specified pole polynomial be used after each set-point change.

In Chapter 5 the observers used in the various schemes were considered. It was found that where the present output value was not included in the state estimate, and therefore not in the present control input, the optimal observer found was that employed in the state-space method developed in Sec. 2.5(1). However, when this output value, $y(t)$, was included in the state estimate, such as in the equivalent state-space forms of Clarke and Gawthrop (1975) and Wellstead et al (1979b), a non-optimal observer was arrived at. Both observers were, however, found to be stable, on condition that the noise polynomial, $C(z^{-1})$, is stable.

This text has concentrated on the theory of state-space self-tuning controllers including simulation studies, where possible,

to show the various tuners under operating conditions. Nevertheless, this does not take into account the irrational events that occur in a practical situation. It is hoped that further research resulting from this work will be connected with the implementation of the algorithms developed, in an industrial environment. The backbone of the applications being provided by a microprocessor rather than a computer, and to this end computational simplicity has, it is hoped, been stressed throughout the text. The majority of self-tuning applications have, to date, been connected with the Chemical Industry, in which conditions are relatively favourable. But other areas must be considered, e.g. Automotive Engineering, machine tools, etc., where, although a direct usage of self-tuning may be limited because of factors such as excess noise, nevertheless a more robust, rather than performance oriented, algorithm may prove invaluable. This robustness can be regarded as an important point in the control of non-linear plant, which was briefly considered in Sec. 4.3, where a self-tuner providing a control action of some sort, in the presence of continual changes in parameter values, is more favourable than a tuner which provides excellent steady-state control, but goes unstable or provides spurious control signals in the event of parameter alterations. The state-space pole placement controller, Sec. 2.5(1), was shown by simulation to be more robust than its predecessors and therefore more fitted for control of the above.

As with other pole placement methods, the need to only specify the maximum possible system integer time delay, although providing for a more versatile tuner, leads to a gap in mathematical theory. Most stability and/or convergence theory connected with self-tuning

has, to the present time, been concerned with system models in which the time delay, k, is assumed to be known exactly. Although a proof of the self-tuning property for a model with variable k was given in Sec. 3.1, this only shows that, assuming certain conditions hold, one of which is the convergence of the model estimates, the required control law is a possible convergence point for the overall algorithm. Simulations imply that the control law parameters do converge to their desired values, but simulations are not mathematical proofs. Thus further work is necessary to investigate the conditions under which convergence to the correct values does occur, and the effect that this has on the stability of the closed-loop system during the period in which convergence takes place. Unfortunately in adaptive control this is invariably impossible, and therefore it is most likely that mathematical proofs must concentrate on stability and convergence regions, rather than exact values.

Numerous applications involve multivariable processes, and although Single-Input-Multi-Output(SIMO) and Single-Output-Multi-Input (SOMI) can often be regarded as essentially SISO systems, the MIMO case leads to various problems not encountered in SISO self-tuners. With reference to the state-space approach, the number of models, equivalent to the CARMA model, from which to choose is far greater with a MIMO model, the advantages of each particular representation not having been considered extensively in the relevant control literature. Hence much work remains to be done in the area of multivariable theory across the whole field of self-tuning, not least of these being that related to state-space multivariable self-tuning.

The filtering equations obtained in Chapter 5 use , when required

for self-tuning, an observer characteristic equation which is given by the disturbance polynomial, $C(z^{-1})$. However, it was shown that the optimal observer, when y(t) is included in the state estimation identity, is not of this form. Unfortunately it proves to be impractical due to its dubious stability features. Nevertheless, the consideration of the use of other observer polynomials must not be entirely discounted , for example that used if the state-space self-tuner with linear output feedback is considered as being simply a form of modified state feedback. Thus the possibility of other observer forms, although they must be non-optimal, remains to be researched at length with regard to the effect on the self-tuning property the use of an alternative form will have.

Finally, now that self-tuning controllers have been developed by means of the state space, rather than polynomial methods, the numerous fields in control systems in which the vast majority of both theoretical and practical work is carried out by use of the state-space become viable for the investigation of the applicability of self-tuning to that particular area. One example being decentralized control, where, although attempts have been made to employ a polynomial approach, state-space theory predominates. The link between self-tuning and decentralized control appears immediate, as tentative steps, Davison (1978), have already been made from the latter topic towards the field of tuning regulators.

APPENDICES

Appendix 1.1

Independence of $B(z^{-1})$ polynomial from fractional system time delay when the steady-state condition applies.

From (1.2.2) the open loop transfer function is given by

$$G(s) = e^{-s\tau_1} \quad \frac{B_1(s)}{A_1(s)}$$

By factoring $A_1(s)$ into its corresponding distinct roots, G(s) may be rewritten as,

$$G(s) = e^{-s\tau_1} \sum_{i=1}^{m} \frac{\beta_i}{s+\alpha_i} \qquad (A.1.1.1)$$

where m is the order of the $A_1(s)$ polynomial, subject to the order of the $B_1(s)$ polynomial being less than m. The z-transform of $G_0(s)G(s)$ is then found, where $G_0(s)$, the zero-order hold, is defined in (1.2.3), this gives

$$G(z^{-1}) = z^{-k} \frac{B_1(z^{-1})}{A_1(z^{-1})} = z^{-k} \sum_{i=1}^{m} \frac{\beta_{1i}+\beta_{2i}z^{-1}}{(1-e^{-\alpha_1 T_1}z^{-1})} \qquad (A.1.1.2)$$

where T_1 is the sampling interval.

To obtain the actual parameters of the equation (A.1.1.2) the following conditions apply.

$$\left.\begin{aligned} 1)\quad \beta_{1i} &= \frac{\beta_i}{\alpha_i}\{1-e^{-\alpha_i\delta}\} \\ 2)\quad \beta_{2i} &= \frac{\beta_i}{\alpha_i}e^{-\alpha_i T_1}(e^{\alpha_i\tau}-1) \\ 3)\quad \delta &= T_1 - \tau \end{aligned}\right\} \qquad (A.1.1.3)$$

τ being the fractional part of the system time delay.If this fractional part falls to zero, so $\beta_{2i} = 0$ for all i, which means that the order of $B_1(z^{-1})$ falls by unity, from which it follows that $B(z^{-1})$, defined in Sec. 1.2, also has its order reduced by unity.

For the steady-state forward gain characteristic we require B(1), where z is set to unity. The numerator for each i can now be described as $\beta_{1i} + \beta_{2i}$, where

$$\beta_{1i} + \beta_{2i} = \frac{\beta_i}{\alpha_i}(1 - e^{-\alpha_i T_1}) \qquad (A.1.1.4)$$

which is independent of the fractional time delay, τ . As the denominator of (A.1.1.2) is also independent of τ, it follows that $B_1(1)$ must necessarily be independent of τ.

Returning to the definition of $B(z^{-1})$ in Sec. 1.2, the C_3 polynomial has no dependence on the fractional part of the time delay, therefore B(1) must be independent of the same factor.

Q.E.D.

Appendix 2.1

Proof of Lemma 2.2 :

Using the example with n = 2, k = 2 where $n_c = 2$ also, then

$$C(z^{-1}) = 1 + c_1 z^{-1} + c_2 z^{-2}$$

and the polynomial control input is given by,

$$u(t) = \frac{-G(z^{-1})}{S_2' C(z^{-1}) + E(z^{-1})B(z^{-1})}\, y(t)$$

The corresponding input for the state-space controller is,

$$u(t) = F(t)\hat{\underline{x}}(t)$$

It then follows from (2.3.2) that

$$u(t) = F(t)z^{-1}(I - z^{-1}\bar{P})^{-1}\{Qu(t) + Ry(t)\}$$

where $\bar{P} = P - RH = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -c_2 \\ 0 & 0 & 1 & -c_1 \end{pmatrix}$

and $I - z^{-1}\bar{P} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -z^{-1} & 1 & 0 & 0 \\ 0 & -z^{-1} & 1 & c_2 z^{-1} \\ 0 & 0 & -z^{-1} & 1 + c_1 z^{-1} \end{pmatrix}$

Hence $(I - z^{-1}\bar{P})^{-1} = \frac{1}{C(z^{-1})} \begin{pmatrix} C(z^{-1}) & 0 & 0 & 0 \\ z^{-1}C(z^{-1}) & C(z^{-1}) & 0 & 0 \\ z^{-2}(1+c_1 z^{-1}) & z^{-1}(1+c_1 z^{-1}) & 1+c_1 z^{-1} & -c_2 z^{-1} \\ z^{-3} & z^{-2} & z^{-1} & 1 \end{pmatrix}$

F(t) is obtained from (2.4.7) as,

$$f(t) = \frac{-b_0}{S_2+b_0^2}\,[0 \quad 1 \quad -a_1 \quad a_1^2 - a_2]$$

Thus, $F(t)\left(I-z^{-1}\bar{P}\right)^{-1} = \dfrac{-b_0}{C(z^{-1})(S_2+b_0^2)}\,[f_1' \quad f_2' \quad f_3' \quad f_4']$

in which $f_1' = z^{-1}+z^{-2}(c_1-a_1) + z^{-3}(c_2-a_2+a_1^2-a_1c_1) = z^{-1}+z^{-2}e_1+z^{-3}g_0.$

$$f_2' = 1+z^{-1}(c_1-a_1) + z^{-2}(c_2-a_1c_1+a_1^2-a_2) = 1 + z^{-1}e_1+z^{-2}g_0$$

$$f_3' = -a_1+z^{-1}(a_1^2-a_2-a_1c_1) = -a_1+z^{-1}(g_0-c_2)$$

and $f_4' = a_1^2 - a_2 + a_1c_2z^{-1} = g_0 - c_2 + a_1c_1 + a_1c_2z^{-1}.$

where e_1 and g_0 are calculated from the identity (1.5.3).

From the example definition : $Q = \begin{Bmatrix} b_2 \\ b_1 \\ b_0 \\ 0 \end{Bmatrix}$; $R = \begin{Bmatrix} 0 \\ 0 \\ c_2-a_2 \\ c_1-a_1 \end{Bmatrix}$

Therefore, $C(z^{-1})(S_2+b_0^2)u(t) = -b_0\{(f_1'b_2+f_2'b_1+f_3'b_0)u(t-1)$

$$+(f_3'[c_2-a_2]+f_4'[c_1-a_1])y(t-1)\}$$

Thus: $\left[C(z^{-1})\,\dfrac{S_2}{b_0} + b_0 + z^{-1}(b_1+b_0c_1-a_1b_0) + z^{-2}(b_2+b_1e_1+b_0g_0)\right.$

$$\left. + z^{-3}(b_2e_1+b_1g_0) + z^{-4}g_0b_2\right]u(t) = [z^{-1}\{a_1(a_2-c_2+a_1c_1-a_1^2) + (a_2c_1-a_2a_1)\}$$

$$- z^{-2}(g_0a_2)]y(t).$$

Appendix 2.2

Proof of Lemma 2.3 by induction.

Consider the example where $n_a = 2$, $n_b = 1$, $k = 2$ and $C(z^{-1}) = 1$.

In the polynomial pole-placement tuner, the control input is given by the expression,

$$D(z^{-1})u(t) = G(z^{-1})y(t)$$

where

i) $D(z^{-1}) = 1 + d_1 z^{-1} + d_2 z^{-2}$

ii) $G(z^{-1}) = g_0 + g_1 z^{-1}$

as $n_g = n_a - 1$ and $n_d = n_b + k - 1$.

The coefficients of the $D(z^{-1})$ and $G(z^{-1})$ polynomials are then calculated from the expression,

$$A(z^{-1})D(z^{-1}) - z^{-k}B(z^{-1})G(z^{-1}) = T(z^{-1})C(z^{-1})$$

where $T(z^{-1}) = 1 + t_1 z^{-1} + t_2 z^{-2}$.

Thus;

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ a_1 & 1 & -b_0 & 0 \\ a_2 & a_1 & -b_1 & -b_0 \\ 0 & a_2 & 0 & -b_1 \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \\ g_0 \\ g_1 \end{bmatrix} = \begin{bmatrix} t_1 - a_1 \\ t_2 - a_2 \\ 0 \\ 0 \end{bmatrix}$$

It follows that; $d_1 = t_1 - a_1$

$$d_2 = \frac{1}{\Delta}\{(a_2 b_0 b_1 - a_1 b_1^2)(t_1 - a_1) + b_1^2(t_2 - a_2)\}$$

$$g_0 = \frac{1}{\Delta}\{(a_2 b_1 - a_1^2 b_1 + a_1 a_2 b_0)(t_1 - a_1) + (a_1 b_1 - a_2 b_0)(t_2 - a_2)\}$$

$$g_1 = \frac{1}{\Delta}\{(a_2^2 b_0 - a_1 a_2 b_1)(t_1 - a_1) + a_2 b_1 (t_2 - a_2)\}$$

Hence $[C(z^{-1})\frac{S_2}{b_0} + E(z^{-1})B(z^{-1}) + z^{-k}B(z^{-1})g_0]u(t)$

$= [-g_1z^{-1} + g_0(a_1z^{-1} + a_2z^{-2})]y(t)$

and this gives a similar relationship to that obtained in the proof of Lemma 2.1.

Assuming $S_2' = S_2/b_0$, we have

$[C(z^{-1})S_2' + E(z^{-1})B(z^{-1})]u(t) = -G(z^{-1})y(t) + g_0\{A(z^{-1})y(t) - z^{-k}B(z^{-1})u(t)\}$

In this case, however, $A(z^{-1})y(t) - z^{-k}B(z^{-1})u(t) = C(z^{-1})e(t)$, and g_0 acts on all the states; i.e. f_1' , f_2' etc. all contain a g_0 term, whereas when $C(z^{-1}) = 1$, g_0 only acts on the $(n+k)^{th}$ state.

It is now required that $g_0C(z^{-1})e(t)$ is subtracted from the right hand side of the above equation, and this is equivalent to adding $g_0C(z^{-1})e(t)$ to the state vector, as F(t) contains a negation.

We have $f_1'(g_0z^{-3}), f_2'(g_0z^{-2}), f_3'(g_0z^{-1}), f_4'(g_0)$

Where $f_i'(\cdot)$ implies that f_i' contains that specific term .

Therefore we must add e(t) to the $(n+k)^{th}$ state, $c_1e(t)$ to the $(n+k-1)^{th}$ state, etc. Such that when multiplied by the respective g_0 terms, $g_0C(z^{-1})e(t)$ will have been subtracted from the control equation.

Defining the new state as,

$$\underline{x}'(t) = \underline{x}(t) + \bar{H}^T e(t)$$

then $\bar{H} = [0 \quad c_2 \quad c_1 \quad 1]$ in this example.

Although, in the general case:

$$\bar{H} = [\,0,\ldots,0,c_{n_c},\ldots,c_1,1]$$

where $\bar{H}$ contains $n+k-1-n_c$ leading zeros.

where $\Delta = b_1^2 - a_1 b_0 b_1 + a_2 b_0^2$

In the state-space representation, the following matrices are defined

$$P = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & -a_2 \\ 0 & 1 & -a_1 \end{bmatrix}; \quad Q = \begin{bmatrix} b_1 \\ b_0 \\ 0 \end{bmatrix} \quad ; \quad F = [f_1 \quad f_2 \quad f_3]$$

and $$\underline{\hat{x}}(t) = \begin{bmatrix} b_1 z^{-1} \\ z^{-1}(b_0 + b_1 z^{-1}) \\ z^{-2}(b_0 + b_1 z^{-1}) \end{bmatrix} u(t) + \begin{bmatrix} 0 \\ -a_2 z^{-1} \\ -a_1 z^{-1} - a_2 z^{-2} \end{bmatrix} y(t)$$

where the coefficients of the matrix F are obtained from the expression;

$$F^T = W^{-1}(T+R')$$

in which $T^T = [0 \quad t_2 \quad t_1]$, $R'^T = [0 \quad -a_2 \quad -a_1]$

and $$W = \begin{bmatrix} -a_2 b_1 & -a_1 b_1 & -b_1 \\ -a_1 b_1 & -b_1 - a_1 b_0 & -b_0 \\ -b_1 & -b_0 & 0 \end{bmatrix}$$

Therefore, $f_1 = \frac{1}{\Delta}\{-b_1(t_1 - a_1) + b_0(t_2 - a_2)\}$

$$f_2 = \frac{1}{\Delta}\{(a_1 b_1 - a_2 b_0)(t_1 - a_1) - b_1(t_2 - a_2)\}$$

$$f_3 = \frac{1}{\Delta}\{(a_2 b_1 + a_1 a_2 b_0 - a_1^2 b_1)(t_1 - a_1) + (a_1 b_1 - a_2 b_0)t_2 - a_2)\}$$

The control input can now be formed from; $u(t) = F\underline{\hat{x}}(t)$.

Thus: $$\{1 - (f_1 b_1 + f_2 b_0) z^{-1} - (f_2 b_1 + f_3 b_0) z^{-2} - f_3 b_1 z^{-3}\} u(t)$$

$$= -\{(f_2 a_2 + f_3 a_1) z^{-1} + f_3 a_2 z^{-2}\} y(t)$$

and from this the following equalities arise;

i) $f_1 b_1 + f_2 b_0 = -d_1$

ii) $f_2 b_1 = -d_2$

iii) $f_2 a_2 = -g_1$

iv) $f_3 = g_0$

By transferring the state space control law into the polynomial mode, we arrive at

$$(1+d_1 z^{-1}+d_2 z^{-2})u(t) = (g_0+g_1 z^{-1})y(t) + g_0\{z^{-k}B(z^{-1})u(t) - A(z^{-1})y(t)\}$$

Thus the control laws derived via polynomial and state space techniques are identical iff;

$$z^{-k}B(z^{-1})u(t) - A(z^{-1})y(t) = 0$$

This only occurs when $e(t) = 0$, i.e. in a deterministic system model.

Appendix 2.3

Proof of Lemma 2.4

Using the example of Appendix 2.2; where now $C(z^{-1})$ is regarded as,

$$C(z^{-1}) = 1 + C_1 z^{-1} + C_2 z^{-2}.$$

The polynomial control law parameters are obtained from:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ a_1 & 1 & -b_0 & 0 \\ a_2 & a_1 & -b_1 & -b_0 \\ 0 & a_2 & 0 & -b_1 \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \\ g_0 \\ g_1 \end{bmatrix} = \begin{bmatrix} t_1+c_1-a_1 \\ t_2+t_1c_1+c_2-a_2 \\ t_2c_1+t_1c_2 \\ t_2c_2 \end{bmatrix}$$

where $D(z^{-1}) = 1 + d_1 z^{-1} + d_2 z^{-2}$ and $G(z^{-1}) = g_0 + g_1 z^{-1}$ were defined in Appendix 2.2.

In this case, though, $d_1 = t_1 + c_1 - a_1$

$$d_2 = \frac{1}{\Delta}\{b_1^2(t_2 + t_1c_1 + c_2 - a_2 - a_1t_1 - a_1c_1 + a_1^2)$$

$$- b_0b_1(t_2c_1 + t_1c_2 - a_2t_1 - a_2c_1 + a_1a_2) + b_0^2t_2c_2\}$$

$$g_0 = \frac{1}{\Delta}\{(a_1b_1 - a_2b_0)(t_2 + t_1c_1 + c_2 - a_2 - a_1t_1 - a_1c_1 + a_1^2)$$

$$-b_1(t_2c_1 + t_1c_2 - a_2t_1 - a_2c_1 + a_1a_2) + b_0t_2c_2\}$$

$$g_1 = \frac{1}{\Delta}\{a_2b_1(t_2 + t_1c_1 + c_2 - a_2 - a_1t_1 - a_1c_1 + a_1^2)$$

$$- a_2b_0(t_2c_1 + t_1c_2 - a_2t_1 - a_2c_1 + a_1a_2) + (a_1b_0 - b_1)t_2c_2\}$$

where $\Delta = b_1^2 - a_1b_0b_1 + a_2b_0^2$.

When the state space formulation is considered, it is found that the $C(z^{-1})$ polynomial does not affect the values given in Appendix 2.2 for the state feedback parameters f_i, i = 1, 2, 3; as the vector R' is not replaced by R, which contains the parameters of the $C(z^{-1})$ polynomial. The only part of the state space pole placement technique affected, therefore, is the estimation of the state vector, which may now be described as:

$$\underline{\hat{x}}(t) = \frac{1}{C(z^{-1})} \begin{bmatrix} C(z^{-1}) & 0 & 0 \\ z^{-1}(1+c_1z^{-1}) & 1+c_1z^{-1} & -c_2z^{-1} \\ z^{-2} & z^{-1} & 1 \end{bmatrix} \left\{ \begin{bmatrix} b_1 \\ b_0 \\ 0 \end{bmatrix} u(t-1) + \begin{bmatrix} 0 \\ c_2-a_2 \\ c_1-a_1 \end{bmatrix} y(t-1) \right\}$$

This may however, be rewritten to give

$$C(z^{-1})\underline{\hat{x}}(t) = \begin{bmatrix} b_1z^{-1} + b_1c_1z^{-2} + b_1c_2z^{-3} \\ b_1z^{-2} + b_1c_1z^{-3} + b_0z^{-1} + b_0c_1z^{-2} \\ b_1z^{-3} + b_0z^{-2} \end{bmatrix} u(t)$$

$$+ \begin{bmatrix} 0 \\ c_2z^{-1} - a_2z^{-1} - a_2c_1z^{-2} + a_1c_2z^{-2} \\ c_2z^{-2} - a_2z^{-2} + c_1z^{-1} - a_1z^{-1} \end{bmatrix} y(t)$$

and substituting for $\underline{\hat{x}}(t)$ into the expression, $u(t) = F\underline{\hat{x}}(t)$, where F is defined in Appendix 2.2.

$$\{1 + z^{-1}(c_1 - f_1b_1 - f_2b_0) + z^{-2}(c_2 - f_1b_1c_1 - f_2b_0c_1 - f_3b_0 - f_2b_1)$$
$$+ z^{-3}(- f_1b_1c_2 - f_2b_1c_1 - f_3b_1)\}\, u(t) = \{z^{-1}(f_2c_2 - f_2a_2 + f_3c_1 - f_3a_1)$$
$$+ z^{-2}(f_2a_1c_2 - f_2a_2c_1 + f_3c_2 - f_3a_2)\}y(t).$$

The following equalities then arise:

I) $c_1 - f_1 b_1 - f_2 b_0 = d_1$

II) $c_2 - f_1 b_1 c_1 - f_2 b_1 + f_1 b_0 c_2 = d_2$

III) $f_2 c_2 - f_2 a_2 + f_3 c_1 + f_2 c_1 a_1 + f_1 c_2 a_1 = g_1$

IV) $f_3 + f_2 c_1 + f_1 c_2 = g_0$

V) $f_2 a_1 c_2 + f_3 c_2 + f_1 a_2 c_2 = 0$

Hence, the control law obtained from the state space formulation can be considered as

$$D(z^{-1})u(t) = G(z^{-1})y(t) + g_0\{z^{-k}B(z^{-1})u(t) - A(z^{-1})y(t)\}$$

or $D(z^{-1})u(t) = G(z^{-1})y(t) - g_0 C(z^{-1})e(t)$.

For equality of the controllers therefore, the $g_0 C(z^{-1})e(t)$ term must be zero, which is true in the deterministic case. By modifying the state reconstruction technique, to include this extra term, the polynomial pole placement controller can be constructed via the state space. The new estimator required (see Appendix 2.1) is thus

$$\underline{\hat{x}}'(t) = \underline{\hat{x}}(t) + \bar{H}^T \{y(t) - H\underline{\hat{x}}(t)\}$$

where $\bar{H} = [0,\ldots,0,c_{n_c},\ldots,c_1,1]$.

Appendix 3.1

Proof of Lemma 3.3: $Hz^{-1}\text{adj}(I - z^{-1}\bar{F})Q = z^{-k}B(z^{-1})$

A simple inductive example is given, from which a generalized proof may be extrapolated, although this becomes extremely monotonous and illegible.

In this example: $n_a = 2$, $n_b = 2$, $k = 2$.

Then: $P = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -a_2 \\ 0 & 0 & 1 & -a_1 \end{bmatrix}$; $Q = \begin{bmatrix} b_2 \\ b_1 \\ b_0 \\ 0 \end{bmatrix}$; $H^T = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$; $F^T = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix}$

from which $\bar{F} = P + QF = \begin{bmatrix} b_2f_1 & b_2f_2 & b_2f_3 & b_2f_4 \\ 1+b_1f_1 & b_1f_2 & b_1f_3 & b_1f_4 \\ b_0f_1 & 1+b_0f_2 & b_0f_3 & b_0f_4-a_2 \\ 0 & 0 & 1 & -a_1 \end{bmatrix}$

Itfollows that $(I - z^{-1}\bar{F}) = \begin{bmatrix} 1-z^{-1}b_2f_1 & -z^{-1}b_2f_2 & -z^{-1}b_2f_3 & -z^{-1}b_2f_4 \\ -z^{-1}(1+b_1f_1) & 1-z^{-1}b_1f_2 & -z^{-1}b_1f_3 & -z^{-1}b_1f_4 \\ -z^{-1}b_0f_1 & -z^{-1}(1+b_0f_2) & 1-z^{-1}b_0f_3 & z^{-1}(a_2-b_0f_4) \\ 0 & 0 & -z^{-1} & 1+a_1z^{-1} \end{bmatrix}$

As $H = [0\ 0\ 0\ 1]$ we need only consider the fourth row of adj $(I - z^{-1}\bar{F})$ such that:

$$H\ \text{adj}(I - z^{-1}\bar{F}) = [\pi_1, \pi_2, \pi_3, \pi_4]$$

where:
$$\pi_1 = z^{-2}b_0f_1 + z^{-3}(1 + b_1f_1 + b_0f_2)$$
$$\pi_2 = z^{-2}(1 + b_0f_2) - z^{-3}b_0f_1$$
$$\pi_3 = z^{-1} - z^{-2}(b_1f_2 + b_2f_1) - z^{-3}b_2f_2$$
$$\pi_4 = 1 - z^{-1}(b_1f_2 + b_0f_3 + b_2f_1) - z^{-2}(b_1f_3 + b_2f_2) - z^{-3}b_2f_3.$$

Thus $H\ \mathrm{adj}(I - z^{-1}\bar{F})Q = z^{-1}b_0 + z^{-2}b_1 + z^{-3}b_2$

and $z^{-1}H\ \mathrm{adj}(I - z^{-1}\bar{F})Q = z^{-k}B(z^{-1})$

where $B(z^{-1}) = b_0 + b_1 z^{-1} + b_2 z^{-2}$.

Appendix 3.2

Proof of Equation (3.3.34)

In the state-space controller the input is defined by,

$$u(t) = F\underline{\hat{x}}(t) + S\ v(t) \qquad (A.3.2.1)$$

where v(t) is the reference input.

In section 3.1, Lemma 3.1, it was shown that the control input used in the state space form, $u(t) = F\underline{\hat{x}}(t)$, is equivalent, in the polynomial case, to

$$\frac{D(z^{-1})}{C(z^{-1})}\ u(t) = \frac{G(z^{-1})}{C(z^{-1})}\ y(t) \qquad (A.3.2.2)$$

or by defining $D'(z^{-1}) = D(z^{-1}) - C(z^{-1})$

$$u(t) = -\frac{D'(z^{-1})}{C(z^{-1})}\ u(t) + \frac{G(z^{-1})}{C(z^{-1})}\ y(t)$$

where the right hand side of this equation is equivalent to $F\underline{\hat{x}}(t)$.

Hence (A.3.2.1) can be rewritten as:

$$u(t) = -\frac{D'(z^{-1})}{C(z^{-1})}\ u(t) + \frac{G(z^{-1})}{C(z^{-1})}\ y(t) + S\ v(t)$$

$$\text{or } D(z^{-1})\ u(t) = G(z^{-1})\ y(t) + C(z^{-1})S\ v(t) \qquad (A.3.2.3)$$

and substituting this into the standard deterministic system equation,

$$A(z^{-1})\ y(t) = z^{-k}B(z^{-1})\ u(t)$$

the closed-loop equation is obtained as

$$A(z^{-1})\ y(t) = \frac{z^{-k}B(z^{-1})G(z^{-1})}{D(z^{-1})}\ y(t) + \frac{z^{-k}B(z^{-1})C(z^{-1})S}{D(z^{-1})}\ v(t)$$

$$\text{or } [A(z^{-1})D(z^{-1}) - z^{-k}B(z^{-1})G(z^{-1})]\ y(t) = z^{-k}B(z^{-1})C(z^{-1})S\ v(t)$$

By using the identity $AD - z^{-k}BG = CT$, the closed loop equation becomes:

$$y(t) = \frac{z^{-k}B(z^{-1})S\ v(t)}{T(z^{-1})}$$

as in the state space formulation.

Appendix 4.1

The state estimate in a multivariable system for a general $C(z^{-1})$ matrix polynomial.

The estimated state vector is written as,

$$\underline{\hat{x}}(t) = (I_1 - z^{-1}\bar{P})^{-1}\, z^{-1}\,[Qu(t) + Ry(t)] \qquad (A.4.1.1)$$

where $(I_1 - z^{-1}\bar{P}) =$

$$\begin{bmatrix} I & 0 & \cdots & 0 \\ -z^{-1}I & \ddots & & \vdots \\ 0 & \ddots & & z^{-1}C_n \\ \vdots & & I & z^{-1}C_2 \\ 0 & \cdots & -z^{-1}I & I+z^{-1}C_1 \end{bmatrix} \quad \left.\begin{matrix}\\ \\ \\ \\ \end{matrix}\right\} \text{k null matrices}$$

(A.4.1.2)

Thus: (A.4.1.3)

$$(I_1 - z^{-1}\bar{P})^{-1} = |C(z^{-1})|^{-1} \begin{bmatrix} \bar{C}_1 & z^{-1}\bar{C}_1 & \cdots & z^{-(n+k-1)}\bar{C}_1 \\ zC'_1 & & & z^{-(n+k-2)}\bar{C}_2 \\ z^2C'_2 & & & \vdots \\ \vdots & & & \\ z^{n-1}C'_{n-1} & \cdots\; zC'_{n-1} & z^{-1}\bar{C}_{n+k-2} & \\ 0 & \cdots\; 0 & \bar{C}_{n+k-1} & z^{-1}\bar{C}_{n+k-1} \\ \vdots & & & \\ 0 & \cdots\; 0 & & \bar{C}_{n+k} \end{bmatrix}$$

k null matrices

in which $|C(z^{-1})|$ is the determinant of the matrix polynomial $C(z^{-1})$, and

i) $C'_i = |C(z^{-1})|\{\bar{C}_{i+1}|C(z^{-1})|^{-1} - I\}$

ii) $\bar{C}_i = |C(z^{-1})|\{I + C_1z^{-1} + \ldots + C_{i-1}z^{-(i-1)}\}C^{-1}(z^{-1})$

from which, as $C_{n+1} = C_{n+2} = \ldots\ldots = C_{n+k} = 0$ it follows that

$\bar{C}_{n+1} = \bar{C}_{n+2} = \ldots\ldots = \bar{C}_{n+k}$ and $\bar{C}_{n+1} = |C(z^{-1})|\{I + C_1 z^{-1} + \ldots + C_n z^{-n}\}C^{-1}(z^{-1})$

or $\bar{C}_{n+1} = |C(z^{-1})|I$.

Thus $C'_n = |C(z^{-1})|\{\bar{C}_{n+1}|C(z^{-1})|^{-1} - I\}$ and hence, $C'_n = C'_{n+1} = \ldots\ldots$

$= C'_{n+k-1} = 0$.

Therefore these coefficents are not shown on the overall matrix.

The final state estimator form may now be obtained from (A.4.1.1)

Appendix 5.1

Evaluation of the optimal state estimate $\hat{\underline{x}}(t/t)$ for a three dimensional example, the estimate being given as,

$$\hat{\underline{x}}'(t/t) = [I - z^{-1}P^*]^{-1}\{[I - RHE^{-1}]Qu(t-1) + RE^{-1}y(t)\} \quad \ldots\ (A.5.1.1)$$

and $P^* = E \begin{pmatrix} 0 & a_3 & -a_3 \\ 1-a_1 & a_2 & -a_2 \\ 0 & 1 & -a_1 \end{pmatrix}$

where $E = 1 - a_1$ and $C(z^{-1}) =$ unity.

It follows that:

$$[I-z^{-1}P^*]=E^{-1} \begin{pmatrix} (1-a_1) & -z^{-1}a_3 & z^{-1}a_3 \\ z^{-1}(a_1-1) & (1-a_1)-z^{-1}a_2 & z^{-1}a_2 \\ 0 & -z^{-1} & (1-a_1)+z^{-1}a_1 \end{pmatrix}$$

thus $\det (I - z^{-1}P^*) = (1-a_1)\{(1-a_1) + (a_1-a_2)z^{-1} + (a_2-a_3)z^{-2} + a_3z^{-3}\}$

or $\det (I - z^{-1}P^*) = (1 - a_1)\{z + (1 - z)A(z^{-1})\}$

also $\text{adj } (I - z^{-1}P^*) = (1-a_1) \begin{pmatrix} (1-a_1)+(a_1-a_2)z^{-1}+a_2z^{-2} & a_3z^{-1}-a_3z^{-2} & -a_3z^{-1} \\ (1-a_1)z^{-1}+a_1z^{-2} & (1-a_1)+a_1z^{-1} & -a_2z^{-1}-a_3z^{-2} \\ z^{-2} & z^{-1} & (1-a_1)-a_2z^{-1}-a_3z^{-2} \end{pmatrix}$

Therefore, the $(1-a_1)$ term, common to both adjoint and determinant, will cancel to simplify the final expression.

We also have:

$$[I - RH\bar{E}^{-1}]Q = \bar{E}^{-1} \begin{bmatrix} (1-a_1)b_2 + a_3b_0 \\ (1-a_1)b_1 + a_2b_0 \\ b_0 \end{bmatrix}$$

from which:

$$\text{adj}(I-z^{-1}P^*)[I-RH\bar{E}^{-1}]Q = \begin{bmatrix} b_2 + \{(a_3b_0-a_1b_2)+z^{-1}(a_3b_1-a_2b_2)\}(1-z^{-1}) \\ b_1 + b_2z^{-1}+\{(a_2b_0-a_1b_1)+z^{-1}(a_3b_0-a_1b_2)\}(1-z^{-1}) \\ b_0 + b_1z^{-1}+b_2z^{-2} \end{bmatrix}$$

.... (A.5.1.2)

$$\text{and } \text{adj}(I-z^{-1}P^*)R\bar{E}^{-1} = \begin{bmatrix} -a_3 \\ -a_2-a_3z^{-1} \\ -a_1-a_2z^{-1}-a_3z^{-2} \end{bmatrix} \qquad \text{(A.5.1.3)}$$

Substituting for the above in (A.5.1.1) the final estimator form is:

$$\hat{\underline{x}}(t/t) = \frac{1}{\{z+(1-z)A(z^{-1})\}} \left\{ \begin{bmatrix} -a_3 \\ -a_2-a_3z^{-1} \\ -a_1-a_2z^{-1}-a_3z^{-2} \end{bmatrix} y(t) + z^{-1} \begin{bmatrix} b_2 \\ b_1+b_2z^{-1} \\ b_0+b_1z^{-1}+b_2z^{-2} \end{bmatrix} u(t) \right.$$

$$\left. + z^{-1}(1-z^{-1}) \begin{bmatrix} (a_3b_0-a_1b_2)+z^{-1}(a_3b_1-a_2b_2) \\ (a_2b_0-a_1b_1)+z^{-1}(a_3b_0-a_1b_2) \\ 0 \end{bmatrix} u(t) \right\}$$

References

Allidina A.Y. & Hughes F.M. (1980), Generalised self-tuning controller with pole-assignment, Proc. IEE, Vol.127,Pt.D, pp 13-18

Anbumani K., Patnaik L.M. & Sarma I.G. (1981), Self-tuning minimum variance control of nonlinear systems of the Hammerstein model, IEEE Trans. AC-26, No.4, pp. 959-961

Åström K.J. (1970), Introduction to stochastic control theory, Academic Press, New York

Åström K.J. (1974), A self-tuning regulator for nonminimum phase systems, Report No. 7411(c), Lund Inst. of Tech. Lund, Sweden

Åström K.J. (1980), Self-tuning control of a fixed bed chemical reactor system, Int. J. Control, Vol. 32, No.2, pp. 221-256

Åström K.J. & Bohlin T. (1966), Numerical identification of linear dynamic systems from normal operating records, In Theory of Self-Adaptive Control Systems (P.H. Hammond Ed.), Plenum Press

Åström K.J., Borisson U., Ljung L. & Wittenmark B. (1977), Theory and applications of self-tuning regulators, Automatica,Vol. 13, pp. 457-476

Åström K.J. & Eykhoff P. (1971), System identification- a survey, Automatica, Vol. 7, pp. 123-167

Åström K.J. & Wittenmark B(1973), On self-tuning regulators, Automatica, Vol. 9, pp. 185-199

Åström K.J. & Wittenmark B. (1980), Self-tuning controllers based on pole-zero placement, Proc. IEE , Vol. 127, Pt. D, pp120-130.

Bierman G.J. (1977), Factorization methods for discrete sequential estimation, Academic Press, New York.

Bierman G.J. (1981), Efficient time propagation of U-D covariance factors, IEEE Trans. AC-26, No.4, pp. 890-894.

Borisson U, (1975), Self-tuning regulators - industrial application and multivariable theory, Report No. 7513, Lund Inst. of Tech., Lund, Sweden

Bryson A.E. (1977) ed., NASA's advanced control law program for the F-8 DFBW Aircraft, IEEE Trans. AC-22, No. 5, pp. 752-806

Caines P.E. (1972), Relationship between Box-Jenkins-Astrom control and Kalman linear regulator, Proc. IEE, Vol. 119, No.5, pp.615-620

Caines P.E. (1976), Prediction error identification methods for stationary stochastic processes, IEEE Trans. AC-21, pp. 500-506

Clarke D.W. & Gawthrop P.J. (1975), Self-tuning controller, Proc. IEE, Vol. 122, No. 9, pp. 929-934

Clarke D.W. & Gawthrop P.J. (1979a), Self-tuning control, Proc. IEE, Vol. 126, No.6. pp.633-640

Clarke D.W. & Gawthrop P.J. (1979b) Implementation and application of microprocessor based self-tuners, Proc. 5th IFAC Symposium on Identification and System Parameter Estimation, Darmstadt.

Cordero A.O. & Mayne D.Q. (1981), Deterministic convergence of a self-tuning regulator with variable forgetting factor, Proc. IEE, Vol. 128, Pt.D, No.1, pp. 19-23.

Davis M.H.A. (1977), Linear estimation and stochastic control, Chapman & Hall, London

Davison E.J. (1978), Decentralized robust control of unknown systems using tuning regulators, IEEE Trans AC-23, No.2. pp. 276-289

Egardt B. (1978) A unified approach to model reference adaptive systems and self-tuning regulators,Report No. 7134, Lund Inst. of Tech., Lund, Sweden

Fortescue T.R., Kershenbaum L.S. & Ydstie B.E. (1979), Implementation of self-tuning regulators with variable forgetting factors, Dept. of Chem. Eng. Internal Report, Imperial College, London

Gawthrop P.J. (1977), Developments in optimal and self-tuning control theory, D.Phil. Thesis, Oxford University

Goodwin G.C. & Payne R.L. (1977), Dynamic system identification, Academic Press, New York

Holst. J. (1977), Adaptive prediction and recursive estimation, Report No. 1013, Lund Inst. of Tech., Lund, Sweden

Jacobs O.L.R. & Saratchandran P. (1980), Comparison of adaptive controllers, Automatica, Vol. 16, pp. 89-97

Jazwinski A.H. (1970), Stochastic processes and filtering theory, Academic Press, New York

Kalman R.E. (1958), Design of self-optimizing control system, Trans. ASME, Vol. 80, pp. 468-478

Kwakernaak H. & Sivan R. (1972), Linear optimal control systems, Wiley-Interscience, New York

Lam K.P. (1979), Design of stochastic discrete-time linear optimal regulator, OUEL Report 1288/79, Dept. of Engineering Science, Oxford University

Lam K.P. (1980), Implicit and Explicit self-tuning controllers, D. Phil. Thesis, OUEL Report 1334/80, Oxford University

Landau Y.D. (1979), Adaptive control - the model reference approach, Marcel Dekker Inc. New York

Lindorff D.P. & Caroll R.L. (1973), Survey of adaptive control using Lyapunov design, Int. J. Control, Vol. 18, pp. 897-914.

Ljung L. (1978), Convergence analysis of parametric identification methods, IEEE Trans. AC-23, pp. 770-783

Ljung L. (1979), Asymptotic behaviour of the extended Kalman filter as a parameter estimator for linear systems, IEEE Trans. AC-24, pp. 36-50

Melsa J.L. & Sage A.P. (1973), Introduction to probability and stochastic processes, Prentice Hall

Mishkin E. & Braun L. (1961), Eds., Adaptive control systems, McGraw-Hill

Narendra K.S. & Valavani L.S. (1979), Direct and indirect model reference adaptive control, Automatica, Vol. 15, pp.653-664

Omani F. (1981), Design of an adaptive auto-pilot for a sailing yacht, M.Sc. Thesis, Dept. of Elect. Eng., Imperial College, London

Panuska V. (1980), A new form of the extended Kalman filter for parameter estimation in linear systems with correlated noise IEEE Trans. AC-25, pp. 229-235

Peterka V. (1970), A adaptive digital regulation of noisy systems, Proc. 2nd. IFAC Symposium on Identification and System Parameter Estimation, Prague

Prager D.L. & Wellstead P.E. (1981), Multivariable pole-assignment self-tuning regulators, Proc. IEE, Vol. 128, Pt. D, No. 1, pp.9-18.

Rosenbrock H.H. (1970), State space and multivariable theory, Nelson

Söderström T. (1973), An on-line algorithm for approximate maximum likelihood identification of linear dynamic systems, Report No. 7308, Lund. Inst. of Tech., Lund, Sweden

Söderström T., Ljung L & Gustavsson I. (1974), A comparative study of recursive identification methods, Report No. 3085, Lund.Inst. of Tech., Lund, Sweden

Söderström T., Ljung L & Gustavsson I. (1978), A theoretical analysis of recursive identification methods, Automatica, Vol. 14, pp. 231-244.

Strejc V. (1981), State-space theory of discrete linear control, John Wiley

Tsypkin Y.Z., Avedyan E.D. & Gulinskig O.V. (1981), On convergence of the recursive identification algorithms, IEEE Trans. AC-26, No.5, pp. 1009-1018

Vathsal S. (1980), A new configuration for linear filtering problems, IEEE Trans. AC-25, No.6, pp.1206-1207

Warwick K (1981a), Self-tuning regulators - a state space approach, Int. J. Control, Vol. 33, No.5, pp. 839-858

Warwick K. (1981b), Self-tuning property of state space self-tuners, Proc. IEE to be published, also Publication No. EE.CON.81.19, Imperial College, London

Warwick K. (1982), On the Kalman-Bucy filter with specified input, IEEE Trans. AC-27 to be published, also Publication No. EE.CON.81.4, Imperial College, London

Warwick K & Westcott J.H. (1982), Arbitrary linear output feedback self-tuners, Proc. 6th IFAC Symposium on Identification and System Parameter Estimation, Washington, to be published, also Publication No. EE.CON.81.13, Imperial College, London

Water H. van de & Willems J.C. (1981), The certainty equivalence property in stochastic control theory, IEEE Trans. AC-26, No.5, pp.1080-1087.

Wellstead P.E., Edmunds J.M., Prager D & Zanker P. (1979a), Self-tuning pole/zero assignment regulators, Int. J. Control, Vol. 30, pp.1-26.

Wellstead P E., Prager D. & Zanker P. (1979b), Pole-assignment self-tuning regulator, Proc. IEE, Vol. 126, pp.781-787

Wellstead P.E. & Sanoff S.P. (1981), Extended self-tuning algorithm, Control Systems Centre Report No. 506, UMIST

Wittenmark B. (1975), Stochastic adaptive control methods : a survey, Int. J. Control, Vol. 21, pp. 705-730

Wittenmark B, (1979), A two-level estimator for time-varying parameters, Automatica,Vol. 15, pp. 85-89

Gawthrop P.J. (1979), On the stability and convergence of a self-tuning controller, OUEL Report 1293/79, Dept. of Engineering Science, Oxford U.