REDUCTION OF DYNAMICS FOR

OPTIMAL CONTROL OF STOCHASTIC

AND DETERMINISTIC SYSTEMS

by

J.H. Hope

A Doctoral Thesis

Submitted in partial fulfilment of the requirements
for the award of

Doctor of Philosophy of the Loughborough University of Technology

September 1977

Supervisor: Professor C. Storey
Department of Mathematics

ABSTRACT

The optimal estimation theory of the Wiener-Kalman filter is extended to cover the situation in which the number of memory elements in the estimator is restricted. A method, based on the simultaneous diagonalisation of two symmetric positive definite matrices, is given which allows the weighted least square estimation error to be minimised.

A control system design method is developed utilising this estimator, and this allows the dynamic controller in the feedback path to have a low order. A 12-order once-through boiler model is constructed and the performance of controllers of various orders generated by the design method is investigated. Little cost penalty is found even for the one-order controller when compared with the optimal Kalman filter system. Whereas in the Kalman filter all information from past observations is stored, the given method results in an estimate of the state variables which is a weighted sum of the selected information held in the storage elements. For the once-through boiler these weighting coefficients are found to be smooth functions of position, their form illustrating the implicit model reduction properties of the design method.

Minimal-order estimators of the Luenberger type also generate low order controllers and the relation between the two design methods is examined. It is concluded that the design method developed in this thesis gives better plant estimates than the Luenberger system and, more fundamentally, allows a lower order control system to be constructed.

Finally some possible extensions of the theory are indicated. An immediate application is to multivariable control systems, while the existence of a plant state estimate even in control systems of very low order allows a certain adaptive structure to be considered for systems with time-varying parameters.

KEYWORDS

Stochastic systems: Multivariable control: Discrete time: Estimation: Direct digital control: Once-through boiler: Simple controllers: Model reduction: Observers

CONTENTS

CHAPTER 1

INTRODUCTION

1. Outline of Contents of Chapters

The original work of this thesis is largely contained in chapters 3 to 9, and in these a new control system design method is derived and assessed. Chapter 2 generates the optimum control law for the discrete time plant shown in Figure 2.2 and governed by the equations:

$$x_{i+1} = Ax_i + Bu_i + \xi_i \qquad (1.1)$$

$$y_i = Hx_i + \eta_i \qquad (1.2)$$

The plant state at time i is described by a set of variables x(1), x(2),........x(N) which form the elements of the state vector x_i. Similarly the control and observation vectors are u_i and y_i respectively, while ξ_i and η_i are plant disturbances.

The derivation of the optimal control law follows the working of Aoki (reference 1) although equivalent results are given by Aström (reference 2) and Kalman (reference 3), the latter giving his name to the filter which generates the optimal plant state estimate required by the control law. This derivation of the optimal control law is included since it forms the basis upon which the reduced order controller is constructed in subsequent chapters. Computer programs were written to generate the optimal controller, and the method is illustrated by an example of dimension 2, i.e. two variables to describe the plant state. These same programs are later used in chapter 8 to generate the optimal control law for a large dimension plant model.

The remaining chapters deal with the sub-optimal control situation where a simple low order controller is required.

The construction of such a controller is a more complex task than the optimal case, and chapter 3 illustrates the problems that arise in the form of multiple minima of the cost function. However a promising approach is shown to involve the assumption of an "a priori" probability distribution for the plant state as this allows the control law to be derived as for the optimal case.

This approach is taken up in chapter 4 for the general case in which the controller is able to store some, but not all, of its plant information, as this is the restriction which leads to a simple low order controller. The implementation of this controller based on the theory of chapter 4 and the necessary computer program are given in chapter 5.

The performance of the controller depends absolutely upon the choice as to which information to store and an appropriate method of making this selection is required. Chapter 6 describes a suitable method, which is based upon the simultaneous diagonalisation of two symmetric matrices. Again a computer program, now more complex, is written, the same simple example is used and the method is shown to perform very well.

The content of chapters 4,5 and 6 together form a design method which is summarised in Figure 1.1.

To prove the design method more conclusively required a higher order model, and chapter 7 derives a once-through boiler model of order 12. The model is deliberately simplified and in fact is no more than a heat exchanger. When the design method is applied to this model in chapter 8 the structure of the low order controllers generated can be more easily interpreted as a result of this simplification. The low order controllers so generated appear, both in terms of cost function and eigenvalue plots to be very adequate controllers. Unexpectedly some higher order controllers cause the closed loop system to be bordering on instability, and this aspect is discussed.

The reduced order plant state estimator developed in chapters 4,5 and 6 has a number of similarities with observer theory as developed by Luenberger (references 4,5,6) and in chapter 9 these similarities are discussed. Also it is shown that

(i) The reduced order estimator is able to generate a much lower order estimator than is observer theory.

(ii) When the two have the same order the reduced order estimator is not an observer within the Luenberger definition.

(iii) For the same order and for the same design criterion the reduced order estimator provides a better estimate of plant state than does observer theory.

There are a number of interesting and relevant areas which require further study, and these are discussed in chapter 10. Particularly important is the generation of controllers with suitable pre-programmed gain variations, and a method of generating this simple adaptive system is suggested. Also considered are the effects of noise in the controller itself and the structure of controllers for multiloop systems.

The remaining sections of this introduction trace some of the developments in control theory which have led to the present situation in which a design technique for low order controllers is required.

2. Control of Plant

Automatic control of plant is required where either of the following apply:

(a) The task is burdensome on human operators and there is an economic case for reducing staffing by use of automatic control.

(b) The response required is fast and this would be difficult or impossible for an operator to achieve.

Different industries have different problems and may employ automatic control on account of one or both of the above. Fortunately it has been found possible to describe most control situations by a common mathematical notation and it has been in this notation that the traditional methods of control system design have been expressed. Single input, single output, linear control systems can be designed with aid of

(i) Simulation of the system on an analogue or digital computer

(ii) Bode or Nyquist stability criteria in the frequency domain

(iii) The "Root Locus" technique

The latter method (reference 7) comes closest to synthesis of a control system, as the effect of gain changes on the system's roots is traced. This method can be extended from a pure stability assessment (roots in left half plane for stability) to include the desirable attributes of adequate damping and sufficiently fast response.

Rosenbrock (reference 8) has been able to extend the frequency domain description to multivariable systems (i.e. those with more than single inputs and single outputs) and some of these techniques are discussed in section 7 below. In reference 9 the approach is enhanced by the use of interactive computer graphics.

Having carried out a study using one or more of the above techniques a simulation may be decided upon to assess the effects of any known non-linearities, for example controller dead-bands or actuator rate limits. This simulation would also allow the effects of control system failures to be assessed. Further the simulation would allow typical disturbances to be injected, such as, for an aircraft control system, wind gusts.

It is at this stage that shortcomings of a control system often appear. The controller, by means of derivative terms, may give a very stable system but in response to normal disturbances may have an entirely unsatisfactory response. To overcome this sort of difficulty a more direct approach has been developed, for example as set out by Aoki (reference 1) and has come to be known as "Optimal Control Theory".

3. Optimal Control Theory

The approach here is to synthesize a control system directly by first defining

(a) the performance criterion, or cost function

(b) the typical plant disturbances

Taking the example of an aircraft height control system a control law can be derived which will feed back to the control surface the filtered sum of the available measurements in certain proportions. These might include pitch and pitch rate from gyro signals and height information from an altimeter.

The plant state is simply defined by relatively few parameters in the case of the aircraft system, and it was in such a context that optimal control theory was developed. Because it is found that the dynamic system which forms the controller must have the same dimension as the plant itself, the implementation of the theory in cases where the plant model has large dimension is relatively rare. Blomnes et al (reference 10), however, report a field application in the case of a nuclear power station and Herbrik and Jamshidi (reference 11) report a theoretical study for a once-through boiler.

The theory aims to control the plant over a given period or for a certain number of time steps. This will be relevant for a landing or docking manoeuvre but in many instances the control period is infinite, such as for a long-running chemical process or a power station. Curiously the theory in this case is hardly simplified at all, it is merely a question of omitting suffices from certain, otherwise time-varying, quantities. Instead of the optimal feedback gains over a period being generated, an asymptotic feedback gain is found, and if desired direct methods, as discussed in section 9 below, may be utilised for its determination.

4. Model Building and Reduction

As part of a simulation study there may be a requirement to fit a plant model to a given size analogue computer. Or digital computer costs may require a small model which is representative of a large dimension plant. In addition to these rather basic reasons for requiring some reduction in model size, there follows from the optimal control theory approach the consequence that if the model dimension can be reduced then the controller dimension will be similarly reduced, and the control system therefore simplified.

A number of methods are available. Davison (reference 12) has suggested a method which allows system eigenvalues far from the origin to be neglected. Such a modal approach is also used by Porter and Crossley (reference 13) in applications to control system design.

Wilson (reference 14) considers the open loop model reduction problem and gives a method which minimises the weighted mean square difference between outputs of the model and the original system.

The method of Mitra (reference 15) involves projection on to subspaces and in reference 16 Mitra applies this method to a power station boiler system. It is interesting to note that this method, which is in continuous time, involves a simultaneous diagonalisation of two positive definite matrices. This is also a step in the reduction method of chapter 6.

Hickin and Sinha (reference 17) give a method whereby the first few Markov parameters (coefficients of the Taylor expansion in the Laplace operators of transfer function) of the two models are matched.

When reducing the order of the model some criterion is required since clearly almost any approximation is a candidate.

In the context of control theory it should in principle be possible to apply a criterion which will give a model most suitable for the purpose to which it is to be applied, that is as part of the control system design method following the optimal control theory approach. Mitra refers to the desirability of such a criterion.

A novel approach to modelling is adopted in chapters 4 and 5. In effect the modelling is integrated into a control system design method and the requirement to generate a model explicitly is dropped. However since the order of the dynamic system forming the controller has been reduced a low order model must be present implicitly, and some further analysis is carried out in chapter 9 which allows a view to be taken of this implicit model.

5. Random Disturbances

As was pointed out above a perfectly stable control system may be found to respond undesirably, perhaps due to a derivative term, when a disturbance is applied. In general a system may be subject to several such disturbances simultaneously and a convenient method of expressing these disturbances is to use random variables in the analysis and to define the statistical properties of these by means of their probability distributions. The system is now termed "stochastic" and cost functions are now expressed in terms of expected values.

All control systems have a set point and the manner in which this is varied will be relevant to the control system design. The stochastic plant model will be able to embrace this feature and so will allow this disturbance, as it truly is, to be balanced against others in the process of control system design.

The stochastic system description is employed in the theoretical work of all chapters as, for the reasons given above, it appears to be the most general and most relevant to control system design. However deterministic systems are not excluded from consideration since if a particular system input is to be studied for its effect, this input may be given a variance, and, perhaps with other disturbance variances made small, the resulting variances of the state variables may be studied. This is, in effect, equivalent to obtaining the system step response.

The development of the probability distributions involved is based on a "Bayesian" view in which an "a priori" distribution is assumed initially, and this distribution is updated as further observations are made. An exposition of the Bayes approach is given by Raeside in reference 18.

A more general view of a stochastic system would include random changes in the plant characteristics themselves. Such a view was taken by Fel'dbaum in reference 19, where even a simple system is found to require extensive on-line computation in order to determine the optimal control law. Adaptive control must therefore be seen as requiring considerable theoretical effort. However such an approach is particularly relevant to the building of low order models since an adaptive controller could attempt to answer the question: "What characteristics of the plant are to be determined for adequate control?" If one initially has little knowledge of the plant this is a daunting problem. However this learning situation confronts humans continuously, and clearly humans are able to evolve suitable control methods.

There would seem therefore to be no reason why this process could not be automated in a fully adaptive control system design.

Apart from a short discussion of simple adaptive control in chapter 10, this aspect is not considered further in this thesis, but a full understanding of low order controllers would seem to be useful contribution to adaptive control since in this way there are fewer control parameters whose values require optimising.

6. Direct Low Order Control Derivations

From section 4 above it is clear that low order control may require either a model reduction step, or alternatively an estimator reduction step. Several more direct methods have been given whereby a low order controller is assumed and its parameters then adjusted to optimise a criterion. A computer aided design method for various cost functions is suggested by Bereznai and Sinha (reference 20).

Jameson and Rothschild (reference 21) give a method but the designer still needs to specify part of the control structure.

By far the most promising direct method is that of Kurtaran (reference 22) who, for the discrete time case, finds conditions for optimality. However it is stated that no method is currently available to find a solution, and this is understandable in view of the multiple solutions shown to exist in chapter 3.

A gradient method using the Fletcher-Powell routine for the deterministic system is given by Berger (reference 23) and it may be that such a technique could be used to find the Kurtaran solution.

7. Multi-loop Systems

By its structure, optimal control theory will generate plant control inputs which are functions of all the plant outputs. In practice, however, a preferred method is to control one output using perhaps only one input. One finds frequently then a multi-loop situation and this has the following advantages:

(i) Reliability: a failure of any one component will remove from service at most one control loop, which will then be controlled manually.

(ii) Flexibility: during plant start-up control loops may be introduced one by one, allowing any problems to be solved on one loop before passing to the next loop.

Optimal control theory promises to give a performance that is better (in some defined sense) than the multiloop system. However before such methods can gain acceptance some consideration will be required in the above two areas.

If, in a control system, failure of a single transducer, for example, causes the whole plant control system to be removed from service this is clearly undesirable. One is led to the conclusion that a stand-by system is required which would have its own set of transducers. Some form of updating of the stand-by system would ensure that it would be able to take over without unduly disturbing the plant.

A different approach, and one which would allow the loops to be introduced one by one, is based on the work of Rosenbrock (reference 9) where, by forming simple filtered combinations of control or output variables, the frequency domain input - output system matrix is constrained to be "diagonally dominant". By utilising a computer graphical display to generate "Gershgorin bands", which quantify the degree of diagonal dominance, the best design may be found interactively. This technique is particularly powerful since it makes possible the design of each loop separately, there now being little interaction between these loops.

Because some interaction will remain the system will not be optimal, but nevertheless will be close to optimal and will have a relatively simple structure. A system designed in this way will, in common with those designed by optimal control theory, be prone to low reliability since a single transducer may affect several loops. The use of several transducers, one for each input loop, for each output measuring point would seem to overcome this problem, but would introduce additional installation and maintainance costs.

Thus the use of more complex controllers such as envisaged in optimal control theory appears to imply some hardware costs in terms of extra equipment, and in any application this must be balanced against the potential improvement in control system performance.

8. Simple Controllers

As mentioned above the optimal control theory approach will lead to controllers with dimension equal to that of the plant model, whereas it is known that good 3-term control is obtainable (i.e. derivative, proportional and integral terms). On asking why this is so one is led to the conclusion that while the three term controller is sub-optimal it may often be only marginally so. To construct such a simple controller rigorously in the style of optimal control theory appears to require even more complex analysis.

Borg and Giles in reference 24, seek to show that three term controllers are a special case of optimal control theory, but the approach is not constructive in general since a plant model of order 3 is used. Considerable advances have been made in chapters 4 and 5 where the optimal control theory approach is extended to yield a simple controller while still maintaining the optimal control theory advantages, namely

(i) An estimate of plant state is available

(ii) A synthesis method will not require lengthy hill climbing to obtain the best control gains.

9. Computer Methods

The manipulation of matrix equations is a central part of optimal control theory. To assist in computer implementation the MATLAN matrix handling package is used. This package is fully described in reference 25, while a summary of those statements which have been used is given in Appendix 1. The package is designed to be efficient for large matrices where the required storage becomes large. This is particularly encouraging from the point of view of control theory since as confidence is gained larger dimension plant models can be contemplated.

The discrete time matrix Riccati equation occurs in connection with optimal control and estimation and is of the form

$$X_{i+1} = \left[(AX_iA' + Q)^{-1} + B\right]^{-1} \qquad (1.3)$$

If control over an infinite period is being studied then it becomes the asymptotic solution of this equation that is required. In all chapters this asymptotic solution is simply obtained by iterating equation (1.3) until convergence is achieved, as the main interest will be the demonstration of viable methods of control system design. However the inclusion of direct techniques for solving (1.3) would be straight forward.

For continuous time systems the Riccati equation, analogous to the discrete time equation (1.3), is given for example by Barnett (reference 26) as

$$\frac{dX}{dt} = AX + XA' - XBX + Q$$

Barnett also indicates some methods of solution, for example the explicit solution (based on characteristic roots) due to Potter (reference 27) and O'Donnell (reference 28), and iterative methods using Newton's approximation. Repperger (reference 29) has suggested a novel approach to this problem. Analogous methods for the discrete Riccati equation (1.3) would give a computational improvement over the simple asymptotic method used in this thesis.

The foregoing considerations concern off-line computing methods but consideration must be given also to on-line methods. The development of relatively cheap small computers has led to the choice of D.D.C. (direct digital control) in preference to analogue methods in many current control designs and for these the generation of control laws in discrete time is more useful than a continuous time approach. Although a derivation of low order controllers of continuous systems, along the lines of chapters 3, 4 and 5 is probably not difficult this has not been attempted in this thesis as D.D.C. is seen as the more likely application.

For this reason in deriving the discrete-time model of the once through boiler in chapter 7 the control input is specifically assumed to be fixed for the duration of the time interval.

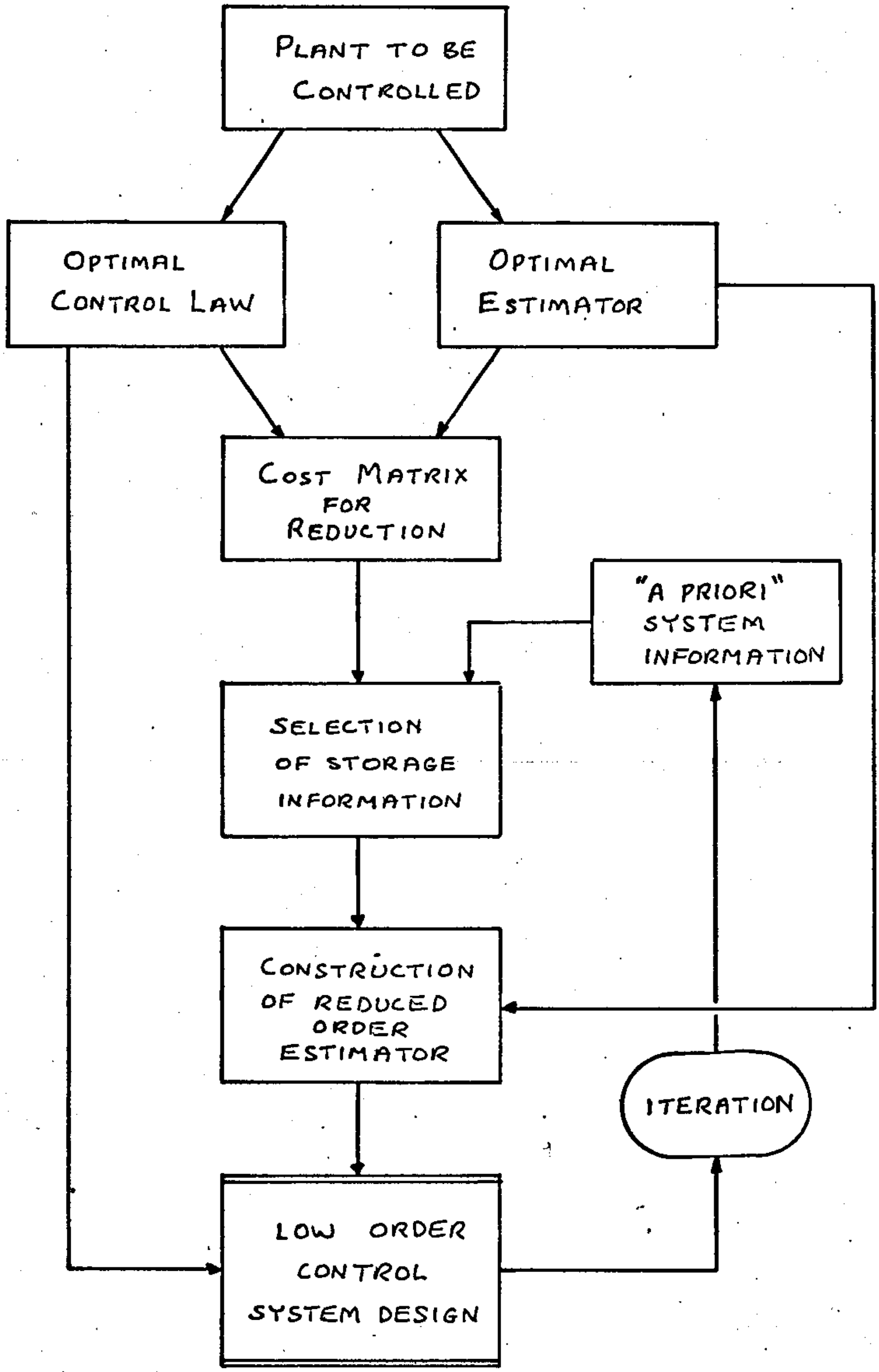


FIGURE 1.1. LOW ORDER CONTROLLER DESIGN METHOD.

CHAPTER 2

OPTIMAL CONTROL AND ESTIMATION

1. Introduction

In this chapter the control and estimation of a linear discrete time stochastic system is considered. The optimal control structure is shown to be separable into an estimator section and a control section. In subsequent chapters in which the estimator is no longer optimal this separation will no longer apply. The derivations set out here are taken from Aoki (Reference 1). Since the results will be frequently referred to and will be illustrated by examples they are included here for completeness. They are also set out here since the optimal case will be used as a basis for comparison with sub-optimal cases.

The system considered is

$$x_{i+1} = Ax_i + Bu_i + \xi_i \tag{2.1}$$

where x_i is an n - dimensional state vector

u_i is a p - dimensional control vector

and where ξ_i is an independent random disturbance vector distributed normally with zero mean and covariance matrix Q.

x_i will be observed by an observation vector y_i according to

$$y_i = Hx_i + \eta_i \tag{2.2}$$

where η_i is an independent random disturbance vector distributed normally with zero mean and covariance matrix R

The performance criterion is

$$J = E\left\{\sum_{1}^{N} x_i{}' V x_i + \sum_{0}^{N-1} u_i{}' Pu_i\right\} \tag{2.3}$$

where E represents expectation

$$J = E \sum_{1}^{N} W_i$$

where $W_i = x_i' V x_i + u_{i-1}' P u_{i-1}$

and where V and P are positive definite $n \times n$ and $p \times p$ matrices respectively.

The various system matrices A, B, Q, R, H, V, P may be made time-dependent with no modification to the derivations. For simplicity of presentation, however, suffices have been omitted, which implies that these matrices do not vary with time.

2. Required Matrix Inversion Formula

A formula which will frequently be used in this and subsequent chapters is derived as follows:-

Let $X = A^{-1} - A^{-1} B (C^{-1} + B' A^{-1} B)^{-1} B' A^{-1}$

Then the product $X (A+BCB')$

$$= I + A^{-1} BCB' - A^{-1} B (C^{-1} + B' A^{-1} B)^{-1} (B' + B' A^{-1} BCB')$$

$$= I + A^{-1} BCB' - A^{-1} B (C^{-1} + B' A^{-1} B)^{-1} (C^{-1} + B' A^{-1} B) CB'$$

$$= I + A^{-1} BCB' - A^{-1} BCB'$$

$$= I$$

It follows that

$$(A + BCB')^{-1} = X$$

$$= A^{-1} - A^{-1} B (C^{-1} + B' A^{-1} B)^{-1} B A^{-1} \qquad (2.4)$$

which is the required formula.

3. Required Probability Relation

A result concerning expected values will be required. This result is obtained as follows:-

$$E (x|a) = \int_x x \, p (x|a) \, dx$$

$$= \int_x x \int_b p (x, b|a) \, db \, dx$$

$$= \int_x x \int_b p (x|a, b) \, p (b|a) \, db \, dx$$

using the chain rule of probabilities.

In the inner integration x is fixed while integration is carried out over the range of b. It is therefore permissible to move x within the inner integration to give

$$E\,(x|a) = \int_x \int_b x\,p\,(x|a,\,b)\,p\,(b|a)\,db\,dx$$

Interchanging the order of integration

$$E\,(x|a) = \int_b \int_x x\,p\,(x|a,\,b)\,p\,(b|a)\,dx\,db$$

Since p (b|a) is independent of x it may be taken outside the inner integration to give

$$E\,(x|a) = \int_b p\,(b|a)\left\{\int_x x\,p\,(x|a,\,b)\,dx\right\}db$$

$$= \int_b p\,(b|a)\left\{E\,(x|a,\,b)\right\}db$$

$$= E\left\{E\,(x|a,\,b)\ |a\right\} \qquad (2.5)$$

This is the required result.

4. Optimal Control Law

Let the information state of the system at time i be written as If_i. This is useful when considering probability distributions based upon this information. For a system with a complete memory If_i will consist of all past observations y_i. For a memory system which is not perfect If_i will contain less information than this.

Let $\lambda_i = E\,(W_i|If_i)$,

Using the principle of dynamic programming, this method of analysis having been set out in a book by Bellman (Reference 30) λ_N is evaluated first.

$$\lambda_N = E\,(W_N|If_{N-1})$$

$$= E\,(x'_N\,Vx_N + u'_{N-1}\,Pu_{N-1}|If_{N-1})$$

$$= E\Big[(Ax_{N-1} + Bu_{N-1} + \xi_{N-1})'\,V\,(Ax_{N-1} + Bu_{N-1} + \xi_{N-1})$$

$$+\ u'_{N-1}\,Pu_{N-1}|If_{N-1}\Big]$$

$$= E\Big[x'_{N-1}\,A'VAx_{N-1} + 2\,x'_{N-1}\,A'VBu_{N-1}$$

$$+ u'_{N-1} (P + B' VB) u_{N-1} | If_{N-1} \Big]$$

$$+ E \xi'_{N-1} V \xi_{N-1} \qquad (2.6)$$

where other terms in ξ_{N-1} vanish since ξ_{N-1} is an independent random vector

This expression is now manipulated to form a square in u'_{N-1}. This is analogous to "completing the square" for a quadratic expression.

$$\lambda_N = E\Big\{[u_{N-1} + (P+B'VB)^{-1}B'VAx_{N-1}]' (P+B'VB) [u_{N-1}+(P+B'VB)^{-1}B'VAx_{N-1}]$$

$$+ x'_{N-1} [A'VA - A'VB(P+B'VB)^{-1}B'VA] x_{N-1} |If_{N-1}\Big\} + \text{trace } QV$$

The trace of a matrix is the sum of its diagonal elements and is used above as a convenient alternative to $\sum_i \sum_j Q_{ji}V_{ij}$

$$\lambda_N = E\Big[(u_{N-1} + \Lambda_{N-1} x_{N-1})' T_1(u_{N-1} + \Lambda_{N-1} x_{N-1}) + x'_{N-1} I_1 x_{N-1} | If_{N-1}\Big]$$

$$+ \text{trace } QV \qquad (2.7)$$

where $\Lambda_{N-1} = (P + B'VB)^{-1} B'VA$ (2.8)

$T_1 = P + B'VB$ (2.9)

$\Pi_1 = A'VB (P + B'VB)^{-1} B'VA$ (2.10)

and $I_1 = A'V - \Pi_1$ (2.11)

The first term in the expression for λ_N is a positive definite quadratic form so it has a minimum value of zero. Consequently λ_N is minimised when

$$u_{N-1} = -\Lambda_{N-1} x_{N-1} \qquad (2.12)$$

This is the required control law when x_{N-1} is known. The feedback control system implied by this equation is shown in Figure 2.1.

In the system being considered x_{N-1} is known only as a result of the observations y_i, that is the information about x_{N-1} may be expressed as the probability distribution

$$p (x_{N-1} | If_{N-1})$$

Let this distribution be Gaussian with mean μ_{N-1} and convariance matrix Γ_{N-1}.

Then

$$\lambda_N = E\left\{[u_{N-1} + \Lambda_{N-1}\mu_{N-1} + \Lambda_{N-1}(x_{N-1}-\mu_{N-1})]' T_1 [u_{N-1} + \Lambda_{N-1}\mu_{N-1} + \Lambda_{N-1}(x_{N-1}-u_{N-}\right.$$

$$\left. + x'_{N-1} I_1 x_{N-1} | If_{N-1}\right\} \quad + \text{trace } QV$$

$$= (u_{N-1} + \Lambda_{N-1}\mu_{N-1})' T_1 (u_{N-1} + \Lambda_{N-1}\mu_{N-1})$$

$$+ 2E\left[(u_{N-1} + \Lambda_{N-1}\mu_{N-1})' T_1 \Lambda_{N-1} (x_{N-1}-\mu_{N-1}) | If_{N-1}\right]$$

$$+ E\left[(x_{N-1}-\mu_{N-1})' \Lambda'_{N-1} T_1 \Lambda_{N-1} (x_{N-1}-\mu_{N-1}) | If_{N-1}\right]$$

$$+ E\, x'_{N-1} I_1 x_{N-1}$$

$$+ \text{trace } QV$$

Now $E(x_{N-1}-\mu_{N-1}) = 0$ and if ν_1 is defined as

$$\nu_1 = \text{trace } QV + E\left[(x_{N-1}-\mu_{N-1})' \Pi_1 (x_{N-1}-\mu_{N-1}) | If_{N-1}\right] \qquad (2.13)$$

then

$$\lambda_N = (u_{N-1} + \Lambda_{N-1}\mu_{N-1})' T_1 (u_{N-1} + \Lambda_{N-1}\mu_{N-1})$$

$$+ E\left[x'_{N-1} I_1 x_{N-1} | If_{N-1}\right] + \nu_1$$

As before λ_N is minimised if the first term on the right hand side, which is a positive definite quadratic form, is made zero.

This is achieved by setting

$$u_{N-1} = -\Lambda_{N-1}\mu_{N-1} \qquad (2.14)$$

Let the corresponding minimum value of λ_N be denoted by γ_N^*, i.e.

$$\gamma_N^* = \min_{u_{N-1}} \gamma_N$$

$$= E\left[x'_{N-1} I_1 x_{N-1} | If_{N-1}\right] + \nu_1$$

It is now necessary to find the optimal control vector u_{N-2} for the preceding point in time. Let

$$\gamma_{N-1} = E\left[W_N + W_{N-1} | If_{N-2}\right]$$

$$= \lambda_{N-1} + E\,(\gamma_N^{*}\,|\,If_{N-2})$$

$$= E\left[x'_{N-1}Vx_{N-1} + u'_{N-2}Pu_{N-2}|If_{N-2} \quad + E\,(\gamma_N^{*}\,|\,If_{N-2}\right)$$

$$= E\left[x'_{N-1}(V+I_1)x_{N-1} + u'_{N-2}Pu_{N-2} + \nu_1|If_{N-2}\right]$$

In the above the relation (2.5) $E\left[E(\theta|If_{N-1})|If_{N-2}\right] = E(\theta|If_{N-2})$ which was proved in section 3 has been used.

Proceeding as before it follows that

$$\gamma^{*}_{N-1} = E\left[x'_{N-2}\, I_2\, x_{N-2} + \nu_2|If_{N-2}\right] \tag{2.15}$$

and $$u^{*}_{N-2} = -\Lambda_{N-2}\mu_{N-2} \tag{2.16}$$

where $P\,(x_{N-2}|If_{N-2})$ is Gaussian with mean μ_{N-2} and covariance matrix Γ_{N-2}

and where $$I_2 = A'\,(V + I_1)\,A - \Pi_2 \tag{2.17}$$

$$\Pi_2 = A'\,(V + I_1)\,B\,(P + B'\,(V + I_1)B\,)^{-1}\,B'\,(V + I_1)\,A \tag{2.18}$$

$$\Lambda_{N-2} = (P + B'\,(V + I_1)\,B)^{-1}\,B'\,(V + I_1)\,A \tag{2.19}$$

$$\begin{aligned}\nu_2 &= \nu_1 + \text{trace}\,(V + I_1)\,Q \\ &\quad + E\left[(x_{N-2}-\mu_{N-2})'\,\Pi_2\,(x_{N-2}-\mu_{N-2})|If_{N-2}\right]\end{aligned} \tag{2.20}$$

Continuing in this way all optimal control vectors can be calculated so that in general

$$\gamma^{*}_{i+1} = E\left[x_i'\, I_{N-i}\, x_i + \nu_{N-i}|If_i\right] \tag{2.21}$$

$$u_i^{*} = -\Lambda_i\,\mu_i \tag{2.22}$$

where

$$I_{N-i} = A'\,(V + I_{N-i-1})\,A - \Pi_{N-i} \tag{2.23}$$

$$\Pi_{N-i} = A'\,(V + I_{N-i-1})B\left[P + B'\,(V + I_{N-i-1})B\right]^{-1}B'\,(V + I_{N-i-1})\,A \tag{2.24}$$

$$\Lambda_i = \left[P + B'\,(V + I_{N-i-1})B\right]^{-1}B'\,(V + I_{N-i-1})\,A \tag{2.25}$$

$$\begin{aligned}\nu_{N-i} &= \nu_{N-i-1} + \text{trace}\left[(V + I_{N-i-1})\,Q_i\right] \\ &\quad + E\left[(x_i - \mu_i)'\,\Pi_{N-i}(x_i - \mu_i)|If_i\right]\end{aligned} \tag{2.26}$$

The probability distribution of x_i, p $(x_i|If_i)$, is Gaussian with mean μ_i and convariance matrix Γ_i.

All the feedback coefficients may be obtained in this way. In the case of perfect observation the coefficients may be calculated independently of the probability distributions p $(x_i|If_i)$.

The structure of the control system is shown in Figure 2.2. It can be seen that the estimator and the controller are separate and in addition that the controller feedback gains are those which are applicable to the determinate case. This property is sometimes called the "certainty equivalence principle", (Reference 1).

5. Optimal Estimator

To provide the above conditional means μ_i for the controller an estimation system is required to accept the new observations y_i given by equation (2.2), i.e.

$$y_i = Hx_i + \eta_i$$

Suppose the state variable distribution at time i is given by p $(x_i|y^i)$ = const. $\exp\left[-\frac{1}{2}(x_i-\mu_i)'\,\Gamma_i^{-1}\,(x_i-\mu_i)\right]$ that is, the distribution is (2.27)
Gaussian with mean μ_i and co-variance matrix Γ_i.

The notation y^i has been used to abbreviate y_i, y_{i-1} ...... y_o, that is all past observations.

Following the observation y_{i+1} it is necessary to calculate p $(x_{i+1}|y^{i+1})$ the state variable probability distribution for the next point in time.

By the chain rule of probabilities

$$\begin{aligned} p\,(x_{i+1}|y^{i+1}) &= p\,(x_{i+1}|y_{i+1}, y^i) \\ &= \frac{p\,(x_{i+1}, y_{i+1}|y^i)}{p\,(y_{i+1}|y^i)} \end{aligned} \qquad (2.28)$$

The numerator may be found from

$$p\ (x_i,\ x_{i+1},\ y_{i+1}|y^i) = p\ (x_{i+1},\ y_{i+1}|x_i,\ y_i)\ p\ (x_i|y^i)$$
$$= p\ (y_{i+1}|x_{i+1},\ x_i,\ y^i)\ p\ (x_{i+1}|x_i,\ y^i)\ p\ (x_i|y^i)$$
$$= p\ (y_{i+1}|x_{i+1})\ p\ (x_{i+1}|x_i)\ p\ (x_i|y^i) \qquad (2.29)$$

Integrating this with respect to x_i gives the numerator of (2.28)

$p\ (x_i|y^i)$ is const. $\exp\left[-\frac{1}{2}\ (x_i-\mu_i)'\Gamma_i^{-1}\ (x_i-\mu_i)\right]$

$p\ (x_{i+1}|x_i)$ is const. $\exp\left[-\frac{1}{2}(x_{i+1} - Ax_i - Bu_i)'\ Q^{-1}\ (x_{i+1} - Ax_i - Bu_i)\right]$

$p\ (y_{i+1}|x_{i+1})$ is const. $\exp\left[-\frac{1}{2}(y_{i+1} - Hx_{i+1})'R^{-1}(y_{i+1} - Hx_{i+1})\right]$

The numerator in (2.28) may therefore be written

$$\int \text{const. } \exp\ (-\tfrac{1}{2}\ E_i)\ dx_i$$

where

$$\begin{aligned} E_i = {} & (x_i - \mu_i)'\Gamma_i^{-1}\ (x_i - \mu_i) \\ & + (x_{i+1} - Ax_i - Bu_i)'Q^{-1}\ (x_{i+1} - Ax_i - Bu_i) \\ & + (y_{i+1} - Hx_{i+1})'\ R^{-1}\ (y_{i+1} - Hx_{i+1}) \end{aligned}$$

To allow integration by x_i this expression must be re-arranged as follows:-

$$\begin{aligned} E_i = {} & (x_i - \mu_i)'\Gamma_i^{-1}\ (x_i - \mu_i) \\ & + (x_{i+1} - A\mu_i - Bu_i - A(x_i - \mu_i))'Q^{-1}(x_{i+1} - A\mu_i - Bu_i - A(x_i - \mu_i)) \\ & + (y_{i+1} - Hx_{i+1})'R^{-1}(y_{i+1} - Hx_{i+1}) \\ = {} & (x_i - \mu_i)'\left[\Gamma_i^{-1} + A'\ Q^{-1}A\right](x_i - \mu_i) \\ & -2\ (x_i - \mu_i)'\ A'\ Q^{-1}(x_{i+1} - A\mu_i - Bu_i) \\ & + (x_{i+1} - A\mu_i - Bu_i)'Q^{-1}(x_{i+1} - A\mu_i - Bu_i) \\ & + (y_{i+1} - Hx_{i+1})'R^{-1}(y_{i+1} - Hx_{i+1}) \end{aligned}$$

Writing $T_i^{-1} = \Gamma_i^{-1} + A'Q^{-1}A$ then by completing the square (2.30)

$$E_i = \left[x_i - \mu_i - \Gamma_i A'Q^{-1}(x_{i+1} - A\mu_i - Bu_i)\right]' T_i^{-1}\left[x_i - \mu_i - \Gamma_i A'Q^{-1}\right.$$
$$\left. x_{i+1} - A\mu_i - Bu_i)\right]$$
$$+ (x_{i+1} - A\mu_i - Bu_i)'(Q^{-1} A\Gamma_i A'Q^{-1})(x_{i+1} - A\mu_i - Bu_i)$$
$$+ (y_{i+1} - Hx_{i+1})'R^{-1}(y_{i+1} - Hx_{i+1}) \tag{2.31}$$

Performing the integration with respect to x_i

$$\int \exp\left[-\tfrac{1}{2} E_i\right] dx_i = \text{const.} \exp\left[-\tfrac{1}{2} E_i'\right]$$

where $E_i' \doteq (x_{i+1} - A\mu_i - Bu_i)'M_{i+1}^{-1}(x_{i+1} - A\mu_i - B\ _i)$

$$+ (y_{i+1} - Hx_{i+1})' R^{-1} (y_{i+1} - Hx_{i+1}) \tag{2.32}$$

and where $M_{i+1}^{-1} = Q^{-1} - Q^{-1} A\ T_i\ A'Q^{-1}$

$$= Q^{-1} - Q^{-1} A\left[\Gamma_i^{-1} + A'Q^{-1}A\right]^{-1} A'Q^{-1}$$
$$= (Q + A\ \Gamma_i\ A')^{-1} \tag{2.33}$$

using the matrix relation of section 2

The expression (2.32) must now be further re-arranged.

$$E_i' = (x_{i+1} - A\mu_i - Bu_i)'\ M_{i+1}^{-1}(x_{i+1} - A\mu_i - Bu_i)$$
$$+ (y_{i+1} - H(A\mu_i + Bu_i) - H(x_{i+1} - A\mu_i + Bu_i))'R^{-1}(y_{i+1} - H(A\mu_i + Bu_i) - H(x_{i+1} - A\mu_i + Bu_i))$$
$$= (x_{i+1} - A\mu_i - Bu_i)'\ (M_{i+1}^{-1} + H'R^{-1}H)(x_{i+1} - A\mu_i - Bu_i)$$
$$-2\ (x_{i+1} - A\mu_i - Bu_i)'\ H'R^{-1}(y_{i+1} - H(A\mu_i + Bu_i))$$
$$+ (y_{i+1} - H(A\mu_i + Bu_i))'R^{-1}(y_{i+1} - H(A\mu_i + Bu_i))$$

Let $\Gamma_{i+1}^{-1} = M_{i+1}^{-1} + H'R^{-1}H$

Then, completing the square

$$E_i' = \left\{x_{i+1} - A\mu_i - Bu_i - \Gamma_{i+1}^{-1} H'R^{-1}\left[y_{i+1} - H(A\mu_i + Bu_i)\right]\right\}$$
$$\times\ \Gamma_{i+1}^{-1}\left\{x_{i+1} - A\mu_i - Bu_i - \Gamma_{i+1}^{-1} H'R^{-1}\left[y_{i+1} - H(A\mu_i + Bu_i)\right]\right\}$$
$$+ (y_{i+1} - H(A\mu_i + Bu_i))'(R^{-1} - H\Gamma_{i+1}H'R^{-1})(y_{i+1} - H(A\mu_i + Bu_i))$$

After integration with respect to x_{i+1}

$$\int \exp\left[-\tfrac{1}{2} E_i'\right] dx_{i+1} = \text{const.} \exp\left[-\tfrac{1}{2} E_i''\right]$$

where $E_i'' = (y_{i+1} - H(A\mu_i + Bu_i))' \Sigma_{i+1}^{-1} (y_{i+1} - H(A\mu_i + Bu_i))$

and where $\Sigma_{i+1}^{-1} = R^{-1} - R^{-1} H \Gamma_{i+1} H' R^{-1}$

Thus $p(y_{i+1}|y^i) = \text{const.} \exp\left[-\tfrac{1}{2} E_i''\right]$ (2.34)

and from (2.28)

$$p(x_{i+1}|y^{i+1}) = \text{const.} \exp\left[-\tfrac{1}{2}(x_{i+1} - \mu_{i+1})' \Gamma_{i+1}^{-1} (x_{i+1}\, \mu_{i+1})\right]$$

where $\mu_{i+1} = A\mu_i + Bu_i + \Gamma_{i+1} H' R^{-1}\left[y_{i+1} - H(A\mu_i + Bu_i)\right]$ (2.35)

and $\Gamma_{i+1}^{-1} = (Q + A\Gamma_i A')^{-1} + H' R^{-1} H$ (2.36)

The required probability distribution for the next time interval has thus been obtained. The equations (2.35, 2.36) provide the relations necessary to calculate all future means and covariance matrices. The discrete time filter system is shown in Figure 2.3, and this filter generates the conditional means required by the control system.

6. Example

A simple example is considered here. The reasons for doing this are firstly to provide an illustration of an optimal controller and in subsequent chapters illustrations of the effect of modifications of this controller and secondly to provide a means of checking computer programs. The example chosen is shown in Figure 2.4.

The example chosen is the combination of two devices each of which performs a purely additive function. Hence there is a similarity with a double integrator continuous system and this similarity becomes rigorous for a sufficiently small time interval. The example has divergent properties when uncontrolled and can be said to be similar to a number of familiar systems.

The equations are

$$(x_1)_{i+1} = (x_1)_i + u_i + (\xi_1)_i$$

$$(x_2)_{i+1} = (x_2)_i + (x_1)_i + (\xi_2)_i$$

This system may be written in the form

$$x_{i+1} = Ax_i + Bu_i + \xi_i \qquad y_i = Hx_i + \eta_i$$

where $A = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$, $B = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ and $H = (0 \quad 1)$

If the aim of the control system is to restrict the value of x_2 then a suitable cost function matrix is

$$V = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$$

The remaining cost, the cost of control,is chosen as

$$P = 3$$

This value is chosen in order to give a simple asymptotic solution for the optimal controller (equation 2.23, 2.24 and 2.25). The controller is given by

$$I = \begin{pmatrix} 6 & 3 \\ 3 & 2 \end{pmatrix}$$

$$\Pi = \begin{pmatrix} 9 & 3 \\ 3 & 1 \end{pmatrix}$$

and $\Lambda = -(1, \frac{1}{3})$

The example is interesting in that the elements of Π show the importance of being able to estimate the current value of x_1, in comparison with the estimation of x_2.

For the optimal estimator the following values of the variances of the disturbances are chosen

$$Q = \begin{pmatrix} 1 & 0 \\ 0 & 4 \end{pmatrix} \text{ and } R = 4$$

With these values the asymptotic solution of the equation for the estimation variance (equation 2.36) is

$$\Gamma = \begin{pmatrix} 3 & 1 \\ 1 & 3 \end{pmatrix}$$

The optimal control and estimation system for this example is shown in Figure 2.5.

7. Optimal Control Subroutine CONTRL

The I.B.M. System/360 Matrix Language System, MATLAN, (reference 25) has been used to implement the recursive equations (2.23, 2.24 and 2.25) for the optimal controller. The subroutine CONTRL accepts the system matrices A, B, the control costs P,V, and a particular value of I and computes the values of I and PI for the previous time point according to equations (2.23 and 2.24). The detail of the subroutine is shown in listing 2.1.

The subroutine was checked using the example. A program was written which read in the various matrices from punched cards, called CONTRL, and printed the resulting values of I, PI and LAMDA. When the asymptotic value of I,

$$I = \begin{pmatrix} 6 & 3 \\ 3 & 2 \end{pmatrix}$$

was read in, the program computed new values of I and LAMDA which agreed with the above example.

A note explaining the meaning of the MATLAN statements used in the computer work is given in Appendix 1.

8. Optimal Estimation Subroutine ESTIM

In a similar way a subroutine ESTIM was written to solve the equation (2.36). The subroutine accepts the system matrices A, B, H, Q, R and a value of G, the covariance matrix of the optimal estimator. The subroutine then calculates the next value of the covariance matrix according to equation (2.36).

A listing of the subroutine, the main program which calls ESTIM, and the program output is given in listing 2.2. The 2 x 2 example of section 6 was used and found to be computed correctly. The subroutines CONTRL and ESTIM were now available for use with larger systems.

9. Plant Models and the Kalman Filter

The "conditional mean generator" developed in section 5 and shown in figure 2.3 is frequently referred to as the "Kalman Filter". This follows from the papers by Kalman where this technique was developed (ref. 3). The filter requires a model of the plant to be set up and for the model to be updated with information from the observed state of the plant. The corrections are in fact proportional to the degree by which the observation does not coincide with the expected value of the observation. This is clearly seen in Figure 2.5, illustrating the example system, where the means μ_1 and μ_2 form a model of the plant analogous to x_1 and x_2, and are updated according to the difference between y and its expected value. This expected value is itself generated by the model. The Kalman Filter thus has a special appeal since the mathematics has generated a system whose functioning is seen to perform in a perfectly understandable fashion. This must set the method apart from methods of filter design based upon optimising techniques where the best parameters are found by searching methods. It should therefore be possible to utilise this structure of the Kalman Filter when other constraints are put upon the system, such as a requirement for the filter to cope with changing plant parameters.

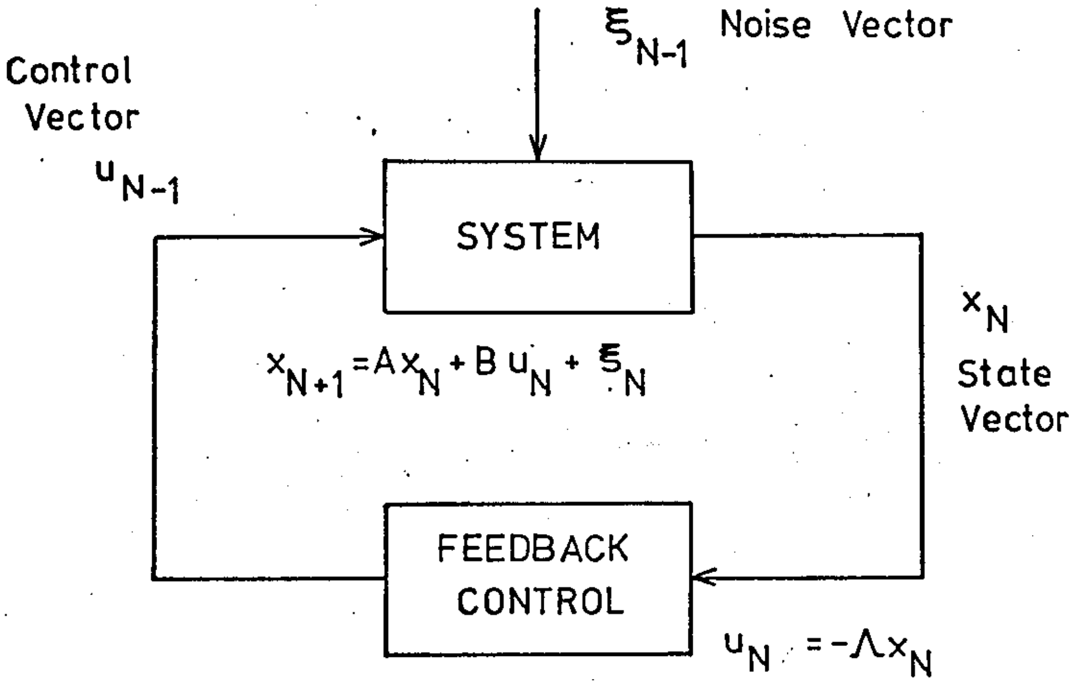


FIGURE 2.1 DETERMINISTIC SYSTEM

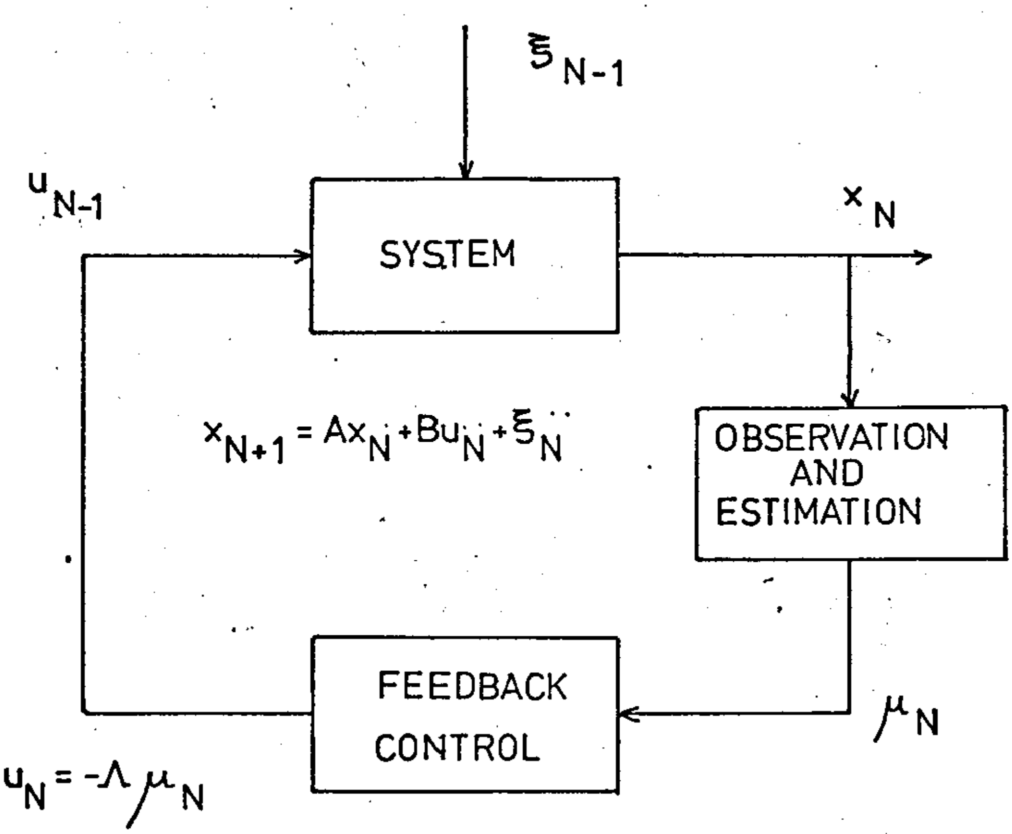


FIGURE 2.2 SYSTEM WITH OBSERVATION

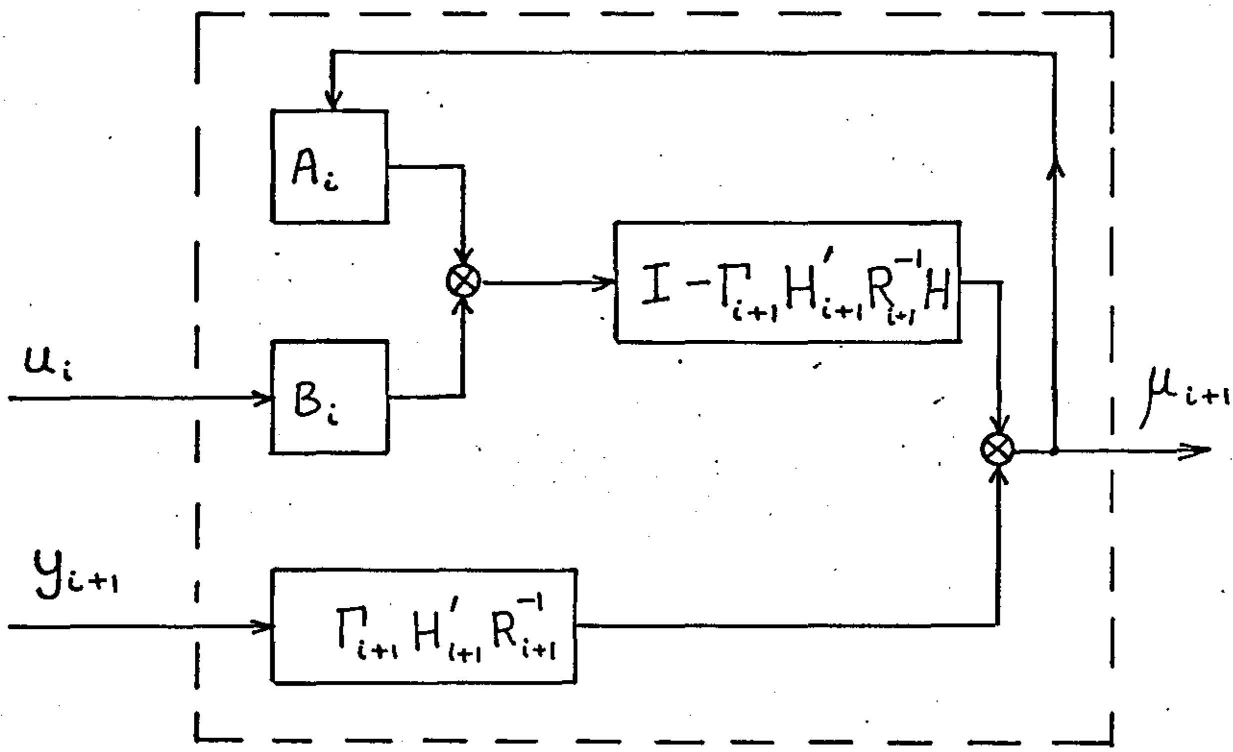
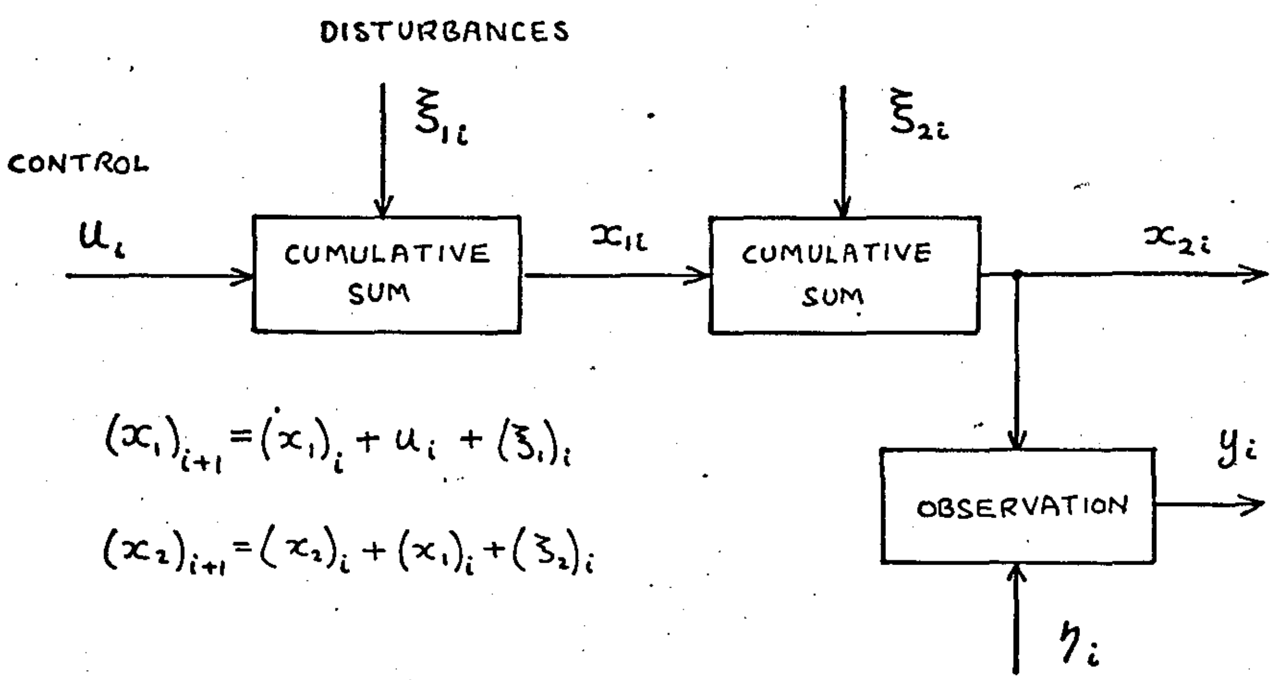


FIGURE 2.3 CONDITIONAL MEAN GENERATOR



$$(x_1)_{i+1} = (x_1)_i + u_i + (\xi_1)_i$$

$$(x_2)_{i+1} = (x_2)_i + (x_1)_i + (\xi_2)_i$$

FIGURE 2.4 EXAMPLE SYSTEM

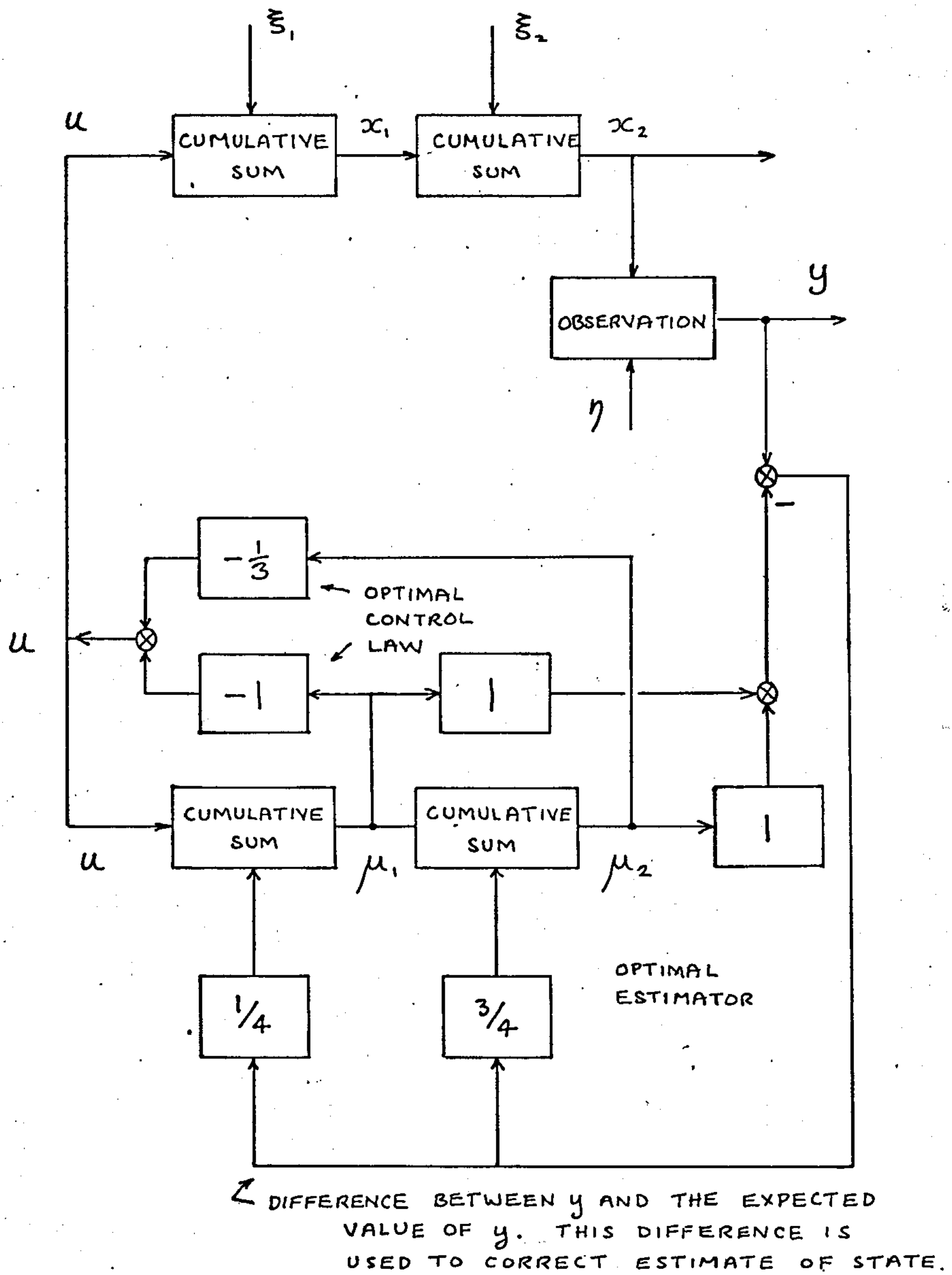


FIGURE 2.5 OPTIMAL CONTROL AND ESTIMATION FOR EXAMPLE SYSTEM

```
STMT     MATLAN STATEMENT

  1              SUBPRO     CONTRL(A,B,P,V,I,LAMDA,PI)
  2              ADD        V,I,VI
  3              MULT       VI,A,VA
  4              TRANS      B,BT
  5              MULT       BT,VA,BA
  6              MULT       VI,B,VB
  7              MULT       BT,VB,BB
  8              ADD        P,BB,PB
  9              DIV        PB,BA,LAMDA
 10              TRANS      BA,AB
 11              MULT       AB,LAMDA,PI
 12              TRANS      A,AT
 13              MULT       AT,VA,AA
 14              SUB        AA,PI,I
 15              RETURN
 16              END
```

<u>Listing 2.1</u> <u>Subroutine CONTRL</u>

Calculates the optimal control law according to the recursive equations (2.23) and (2.24).

```
STMT    MATLAN STATEMENT

  1            MAIN
  2            READ      (A,B,H,Q,R,G)
  3            WRITE     (A,B,H,Q,R,G),FORMAT=A5
  4            CALL      ESTIM(A,B,H,Q,R,G,GNEXT)
  5            WRITE     GNEXT,FORMAT=A5
  6            END
```

```
STMT    MATLAN STATEMENT

  1            SUBPRO    ESTIM(A,B,H,Q,R,G,GNEXT)
  2            TRANS     A,AT
  3            MULT      G,AT,GA
  4            MULT      A,GA,AA
  5            ADD       G,AA,GG
  6            MULT      H,GG,HG
  7            TRANS     H,HT
  8            MULT      HG,HT,HH
  9            ADD       R,HH,RH
 10            DIV       RH,HG,RG
 11            TRANS     HG,HT
 12            MULT      HT,RG,GG
 13            SUB       GG,GG,GNEXT
 14            RETURN
 15            END
```

<u>Listing 2.2</u>   <u>Subroutine ESTIM</u>

Calculates the optimal estimator matrices according to equation (2.36).

Output of the subroutine for the example system.

```
A                          DIMENSIONS = ( 2, 2)

               1              2

     1    1.0000E 00    0.0
     2    1.0000E 00    1.0000E 00

          END OF MATRIX A


B                          DIMENSIONS = ( 2, 1)

               1

     1    1.0000E 00
     2    0.0

          END OF MATRIX B


H                          DIMENSIONS = ( 1, 2)

               1              2

     1    0.0           1.0000E 00

          END OF MATRIX H


Q                          DIMENSIONS = ( 2, 2)

               1              2

     1    1.0000E 00    0.0
     2    0.0           4.0000E 00

          END OF MATRIX Q


R                          DIMENSIONS = ( 1, 1)

               1

     1    4.0000E 00

          END OF MATRIX R
```

<u>Listing 2.2</u> (continued) <u>Output of Subroutine ESTIM</u>

```
G                       DIMENSIONS = ( 2, 2)

                  1                 2

       1     3.0000E 00     1.0000E 00
       2     1.0000E 00     3.0000E 00

             END OF MATRIX G


GNEXT                   DIMENSIONS = ( 2, 2)

                  1                 2

       1     3.0000E 00     1.0000E 00
       2     1.0000E 00     3.0000E 00

             END OF MATRIX GNEXT
```

Listing 2.2 (continued) Output of Subroutine ESTIM

CHAPTER 3

SYSTEMS WITH IMPERFECT MEMORY

1. The Non-Classical Information Pattern

In the previous chapter the optimal design of a linear stochastic control system has been considered and the design has been shown to be separable into the design of an optimal control law and the design of an optimal estimator. The control law is that law which applies in the deterministic case and this property has been called the "certainty equivalence principle".

The optimal estimator will have the same dimension as the system being controlled. In order to represent the system, or plant, accurately in an analysis its dimension could become very large, for example if finite difference methods are being used. However, it is difficult to justify using such a large order estimator, and consequently such a large order control system, in practice as it is known that a control system of order two or three is usually satisfactory. A control system which has a smaller dimension than that of the system it controls corresponds to an estimator which has an imperfect memory.

Witsenhausen (Ref.31) has called the optimal estimatinn system of Chapter 2 the "classical information pattern". This title applies to an estimation system which is able to store all information that it receives. The structure of optimal controllers is considered when the information pattern is non-classical. To illustrate the problems that are encountered in the non-classical situation Witsenhausen uses a simple two stage control problem. This problem will be re-stated below and will later be used to illustrate a method for finding the optimal controller in the non-classical case.

Witsenhausen also uses the example to show how a non-linear controller may be superior to a linear controller in the non-classical case. In order to preserve system structure the work below is restricted to a consideration of linear controllers.

2. Two Stage Control Example*

State equations: $x_1 = x_o + u_1$ (3.1)

$x_2 = x_1 - u_2$ (3.2)

Output equations: $y_o = x_o$ (3.3)

$y_1 = x_1 + v$ (3.4)

Cost function: $k^2 u_1^2 + x_2^2 \qquad k^2 > o$ (3.5)

Stochastic properties: Gaussian where

$E\left\{x_o\right\} = o, \quad E\left\{v\right\} = o$ (3.6)

$E\left\{x_o^2\right\} = \sigma^2, \quad E\left\{v^2\right\} = 1$ (3.7)

Controllers: $u_1 = (\lambda-1)\, y_o$ (3.8)

$u_2 = \mu y_1$ (3.9)

The problem is that of minimising the cost J with respect to λ and μ

$$\begin{aligned} \text{where } J &= E \quad k^2 u_1^2 + x_2^2 \\ &= E \quad k^2 (\lambda-1)^2 \; x_o^2 + (\lambda y_o - \mu\lambda x_o - \mu v)^2 \\ &= k^2 (\lambda-1)^2 \sigma^2 + \mu^2 (\lambda^2\sigma^2 + 1) - 2\mu\lambda^2\sigma^2 + \lambda^2\sigma^2 \qquad (3.10) \end{aligned}$$

---

* The notation used by Witsenhausen is retained

J has a minimum with respect to μ

when

$$\mu = \frac{\lambda^2 \sigma^2}{1 + \lambda^2 \sigma^2} \tag{3.11}$$

and for this μ

$$J = k^2 (1-\lambda)^2 \sigma^2 + \frac{\lambda^2 \sigma^2}{1 + \lambda^2 \sigma^2} \tag{3.12}$$

Witsenhausen establishes that this expression has either one unique minimum or two local minima with respect to λ, depending on the values of k^2 and σ^2.

In the following a particular case is examined with $k = \frac{3}{10}$ and $\sigma = \frac{10}{3}$, which corresponds to the case in which the two local minima are equal. A graph of J against λ for these values is shown in figure 3.1.

The common minimum value of J is 0.91 and the minimising values of λ and μ are

(i) $\lambda = \mu = 0.9$

which corresponds to

$$u_1 = -0.1\, y_o$$
$$u_2 = 0.9\, y_1 \tag{3.13}$$

(ii) $\lambda = \mu = 0.1$

which corresponds to

$$u_1 = -0.9\, y_o$$
$$u_2 = 0.1\, y_1 \tag{3.14}$$

These minimising values have been obtained by an analytical method and no structural significance is apparent. In the following the significance of these solutions is developed.

3. Solution with Classical Information Pattern

This case differs from the above example in that u_2 is able to be a function of the observations y_o and y_1, and also the control u_1. As there is perfect memory the certainty equivalence principle applies and it is first necessary to obtain the optimal deterministic controller.

In this case $E\left\{v^2\right\} = 0$, and the optimal control law is obtained by dynamic programming as

$$\begin{aligned} u_1 &= 0 \quad \text{i.e.} \quad \lambda = 1.0 \\ u_2 &= x_1 \quad \text{i.e.} \quad \mu = 1.0 \end{aligned} \tag{3.15}$$

for which the cost J is zero.

Secondly the estimator must be obtained. Since the estimator has perfect memory y_0 and u_1 are available for the estimation of x_1. Therefore x_1 is known exactly at the second stage and the case becomes identical to the deterministic case.

4. Estimator for Non-Classical Information Pattern

In this case only y_1 is available for the estimation of x_1, so the conditional distribution $p(x_1 \mid y_1)$ is required. Let the first stage control be

$$u_1 = (\lambda-1)\, y_0 \tag{3.16}$$

Then

$$x_1 = x_0 + u_1 = \lambda x_0 \tag{3.17}$$

The distribution of x_0 is Gaussian with mean zero and variance σ^2, so then from equation (3.17) the distribution of x_1 is Gaussian with mean zero and variance $\lambda^2\sigma^2$, that is

$$p(x_1) = \text{const. exp.}\left\{-\frac{1}{2}\,\frac{x_0^{\,2}}{\lambda^2\sigma^2}\right\} \tag{3.18}$$

Since the distribution of v is Gaussian with mean zero and unit variance, it follows from the observation equation (3.4) that

$$p(y_1 \mid x_1) = \text{const. exp.}\left\{-\frac{1}{2}(x_1 - y_1)^2\right\} \tag{3.19}$$

Combining equations (3.18) and (3.19)

$$p\,(x_1, y_1) = p\,(y_1 \mid x_1)\,p\,(x_1)$$

$$= \text{const. exp. } -\tfrac{1}{2}\left\{(x_1 - y_1)^2 + \frac{x_1^{\,2}}{\lambda^2\sigma^2}\right\}$$

$$= \text{const. exp. } -\tfrac{1}{2}\left\{\frac{(x_1 - \sigma_1 y_1)^2}{\sigma_1^{\,2}} + \frac{y_1^{\,2}}{1+\sigma^2}\right\} \qquad (3.20)$$

where $\sigma_1^{\,2} = \dfrac{\lambda^2\sigma^2}{1+\lambda^2\sigma^2}$ (3.21)

Since $p\,(x_1, y_1) = p\,(x_1 \mid y_1)\,p\,(y_1)$ it follows from equation (3.20) that

$$p\,(x_1 \mid y_1) = \text{const. exp. } -\tfrac{1}{2}\left\{\frac{(x_1 - \sigma_1^{\,2} y_1)^2}{\sigma_1^{\,2}}\right\} \qquad (3.22)$$

The conditional distribution of x_1 given observation y_1 has therefore been found to have mean $\sigma_1^{\,2} y_1$ and variance $\sigma_1^{\,2}$.

5. Sub-optimal Control with Reduced Estimator

In order to examine the breakdown of the certainty equivalence principle it is possible to use the deterministic control law, equations (3.15) in conjunction with the reduced estimator obtained in the last section. The resulting controls are

$$u_1 = 0 \qquad \text{i.e.} \qquad \lambda = 1$$

$$u_2 = \sigma_1^{\,2} y_1 \qquad \text{i.e.} \qquad \mu = \sigma_1^{\,2} \qquad (3.23)$$

The cost when using these controls is

$$J = E\left\{k^2 u_1^{\,2} + x_2^{\,2}\right\} = E\left\{x_2^{\,2}\right\}$$

$$= E\left\{(x_1 - u_2)^2\right\}$$

$$= E\left\{(x_1 - \sigma_1^{\,2} y_1)^2\right\}$$

$$= \sigma_1^{\,2} \qquad (3.24)$$

With $\sigma = \frac{10}{3}$, then the controls are $\lambda = 1$, $\mu = 0.917$ and the cost, J is 0.917.

This case does not correspond exactly to either of the optimal solutions obtained in section 2., but can be seen to be close to the optimal solution (i) in which $\lambda = \mu = 0.9$. The cost is only slightly greater than the optimal cost of 0.91.

In this section the certainty equivalence principle has been applied when the estimator had an imperfect memory. Although the optimal solution is not obtained by its use, it can be seen that this sub-optimal control gives a very close approximation to one of the optimal solutions. As this sub-optimal control appears to be an important case, it will be called "pseudo-classical control".

6. Interaction of Control and Estimation

Since the certainty equivalence principle does not hold in the non-classical situation the control law given by equation (3.23) need no longer be related to the deterministic control law. However, instead of falling back on an analytical method of optimising λ and μ as in section 2., it is possible to retain the structure of the classical solution and to modify it to meet the new situation in which the information pattern is non-classical. For this case the cost is

$$J = E\left\{k^2 u_1^2 + x_2^2\right\}$$

$$= k^2 u_1^2 + E\left\{(x_1 - u_2)^2\right\} \qquad (3.25)$$

This cost is minimised with respect to u_2 when

$$u_2 = \hat{x}_1$$

where $\hat{x}_1$ = the mean of the conditional distibution $p(x_1 \mid y_1)$

From equation (3.22)

$$\hat{x}_1 = \sigma_1^2 y_1 \qquad \text{i.e.} \qquad \mu = \sigma_1^2$$

From equation (3.21)

$$\mu = \frac{\lambda^2 \ \sigma^2}{1 + \lambda^2 \sigma^2} \tag{3.26}$$

Although u_2 does in fact correspond to the classical case a new feature becomes apparent when J is minimised with respect to u_1.

Putting $u_2 = \mu y_1$ in equation (3.25)

$$J = E\left\{k^2 u_1^2 + (x_1 - \mu y_1)^2\right\}$$

$$= E\left\{k^2 u_1^2 + (x_1 - \mu x_1 - \mu v)^2\right\}$$

$$= E\left\{k^2 u_1^2 + x_1^2 (1-\mu)^2\right\} + \mu^2 \tag{3.27}$$

The new feature that appears here (as a result of the non-classical information pattern) is the fact that equation (3.27) contains a cost related to x_1. This will result in a non-zero control u_1, and this is at variance with the classical solution. Equation (3.27) can be written

$$J_1 = E\left\{x_1^2 + k_1^2 u_1^2\right\} \tag{3.28}$$

$$\text{where } J_1 = \frac{J}{(1-\mu)^2} - \mu^2 \tag{3.29}$$

$$\text{and } k_1^2 = \frac{k^2}{(1-\mu)^2} \tag{3.30}$$

Completing the square in equation (3.28)

$$J_1 = E\left\{(x_o + u_1)^2 + k_1^2 u_1^2\right\}$$

$$= E\left\{u_1^2 (1+k_1^2) + 2 x_o u_1 + x_o^2\right\}$$

$$= E\left\{(1 + k_1^2)(u_1 + \frac{x_o}{1 + k_1^2})^2 + \frac{k_1^2}{1 + k_1^2} x_o^2\right\}$$

The minimising value of u_1 is

$$u_1 = -\frac{x_0}{1 + k_1^2}$$

$$\text{or } \lambda = 1 - \frac{1}{1 + k_1^2} = \frac{k_1^2}{1 + k_1^2}$$

Using equation (3.30)

$$\lambda = \frac{k^2}{(1-\mu)^2 + k^2} \qquad (3.31)$$

The inter-relation of control and estimation is now evident from equations(3.26) and (3.31) The classical method has been extended using a quadratic minimisation method but the result is that λ depends on μ and that μ depends on λ.

7. Iteration to Optimal Solution

Starting from the pseudo-classical solution $\lambda_0 = 1$, $u_0 = 0.917$ it is possible to iterate using equations(3.25)and(3.31)successively to obtain

$$\lambda_0 = 1.0 \qquad \mu_0 = 0.917$$

$$\lambda_1 = 0.930 \qquad \mu_1 = 0.906$$

$$\lambda_2 = 0.910 \qquad \mu_2 = 0.902$$

This process converges to a solution $\lambda = \mu = 0.9$ as can be seen by substitution. This is precisely the optimal solution (i) obtained in section 2. The significance of this is that it is possible to use methods appropriate to the classical situation to obtain the optimal solution for the non-classical situation. In this process a state estimator is generated and the classical control and estimation structure is maintained.

The solution (ii) of section 2 for which $\lambda = \mu = 0.1$ is known to exist. Starting with values $\lambda_0 = \mu_0 = 0$, iteration using equations (3.26) and (3.31) gives

$$\lambda_0 = 0 \qquad \mu_0 = 0$$
$$\lambda_1 = 0.0826 \qquad \mu_1 = 0.0704$$
$$\lambda_2 = 0.0943 \qquad \mu_2 = 0.0899$$

converging to $\lambda = \mu = 0.1$.

Thus the above method is also capable of yielding the other local minimum provided iteration is begun sufficiently close to it.

8. Application to Larger Systems

The example has shown that despite the problems of the non-classical situation, such as local minima and multiple solutions, it is possible to proceed very much as in the classical situation and to obtain a solution provided the interactions of control and estimation are taken into account.

The most useful area of application for this method is that of continuously running systems and later chapters illustrate this. However, there will be two aspects requiring careful examination.

(i) The example demonstrated the existence of local minima. In the multi-dimensional case these will be far more difficult to detect.

Witsenhausen showed that local minima only occur for $k^2 < \frac{1}{4}$ and with this k^2, local minima occur for σ^2 within a certain range.

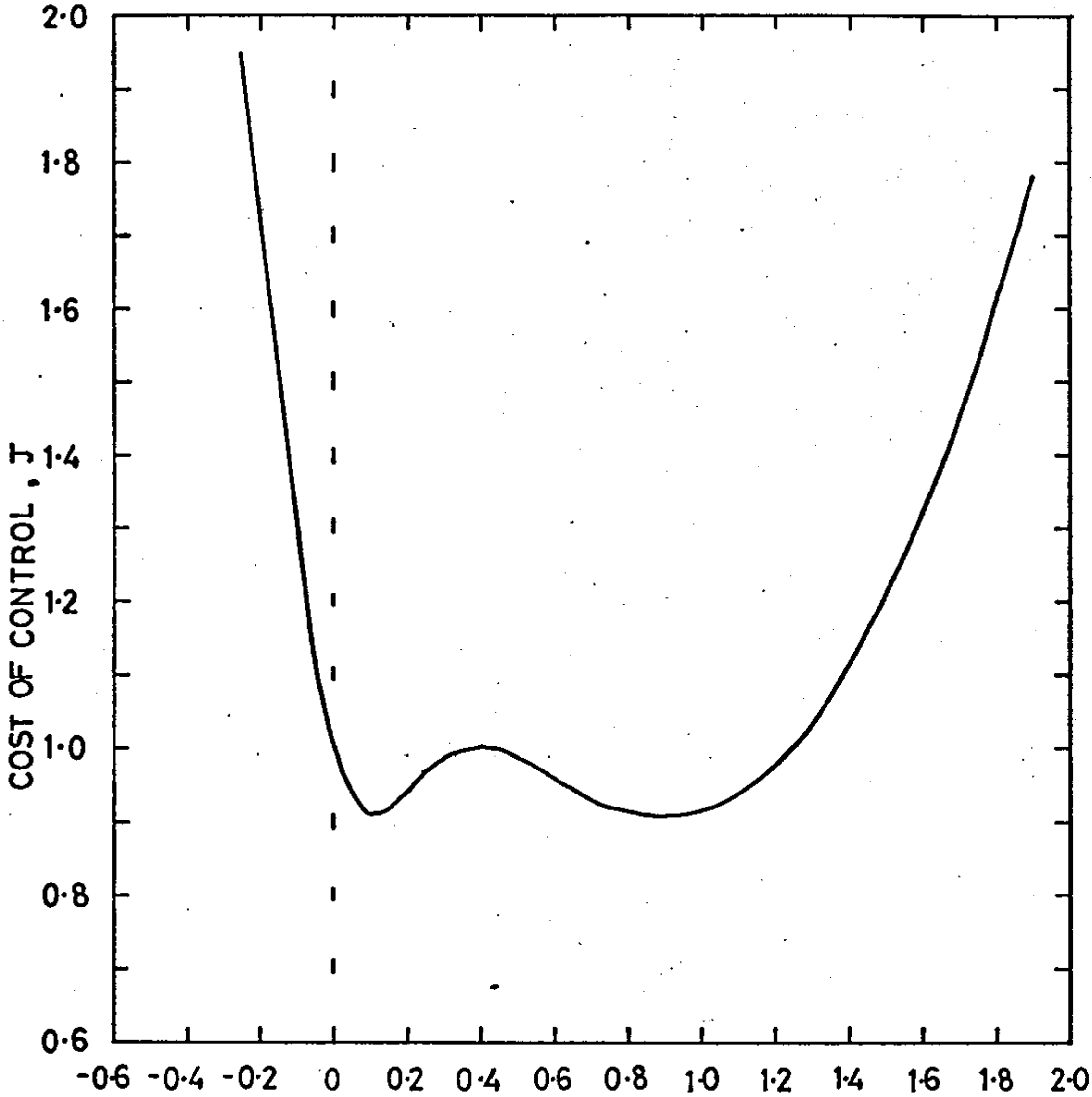
In the light of section 7 it is now possible to interpret these conditions as saying that there will be an optimum that can be reached from the pseudo-classical solution provided k^2 is sufficiently high, that is the cost of transmitting information via the controls is sufficiently high and provided σ^2 is sufficiently high, which corresponds to a poor knowledge of the initial state of the system.

(ii) It is well known that in reducing the order of a control system, such as when a phase advance network is reduced to a proportional network, it is possible that a stable system cannot be designed with the new structure. In the process of reduction from the classical to the non-classical situation, such an effect would manifest itself as the approach of state variable variances to infinity.

In computing terms this phenomenon would be difficult to distinguish from a stable solution with large state variable variances. However, from the point of view of system design this would not be a serious restriction since if the system cost has risen significantly above the optimal classical cost the reduced configuration, even if stable, would not be a viable proposition.

It would be necessary to use alternative methods to investigate the stability of a particular configuration.

MP



GRAPH OF J AGAINST λ WHERE

$$J = k^2(1-\lambda)^2\sigma^2 + \frac{\lambda^2\sigma^2}{1+\lambda^2\sigma^2}$$

AND $k = \frac{3}{10}$, $\sigma = \frac{10}{3}$.

FIGURE 3.1

CHAPTER 4

THEORY OF THE REDUCED ORDER ESTIMATOR

1. Structure of the Reduced System

In Chapter 2 the theory of the optimal estimator was derived and it was shown that this estimator is required if optimal control is to be achieved. In this chapter a structure is considered which leads to an estimator of reduced order. The derivation is similar to the derivation of the optimal estimator of Chapter 2.

The structure of the reduction process is shown in Figure 4.1. The state vector x_i, the control vector u_i, and the observation vector y_i are defined as before. The vector z_i is defined as part of the control system and is a memory element. For the optimal estimator z_i can be identified with the conditional mean μ_i, and will therefore be of order n. For the reduced order estimator z_i is taken to be of order q where

$$1 \leq q < n$$

With this structure an estimate of the state vector, x_i, is made from the information stored in z_i from the previous time point and from the latest observation y_i. These two vectors can be combined together to define a vector v_i of dimension (m + q) which may be called the "information vector", i.e.

$$v_i = \begin{pmatrix} y_i \\ \cdots \\ z_i \end{pmatrix} \tag{4.1}$$

From this vector the estimate of the state vector is constructed, by means of the matrix F_i, according to:-

$$\mu_i = F_i v_i \tag{4.2}$$

The determination of the values of F_i will be described subsequently.

At this point it can be seen that it is necessary to reject some of the information contained in the information vector v_i (dimension $m + q$) in order to construct z_{i+1} (dimension q), the vector containing the information which is to be carried over to the next time step. The most general representation of this rejection process is by means of a non-singular transformation matrix T of dimension $(m + q) \times (m + q)$.

$$\text{Let } v_i = T \begin{pmatrix} z_{i+1} \\ \cdots \\ \alpha_{i+1} \end{pmatrix} \tag{4.3}$$

$$\text{or } v_i = T_\alpha \alpha_{i+1} + T_z z_{i+1} \tag{4.4}$$

where T has been partitioned as

$$T = (T_z \vdots T_\alpha) \tag{4.5}$$

The information which is rejected has been denoted by α_{i+1}, and it will be shown later how the choice of the matrix T can be made in an optimal manner so as to minimise a particular cost criterion.

2. A Priori Distribution of Information Vector, v_i

The probability distribution of α_{i+1} given z_{i+1} will be required and this will be obtained directly from the "a priori" distribution of v_i which can be assumed to be Gaussian with covariance matrix P. So that

$$p(v_i) = \text{const.} \exp (-\tfrac{1}{2} v_i' P^{-1} v_i)$$

A notation will be introduced here to simplify derivations and this writes the above equation as

$$p(v_i) \underset{e}{=} v_i' P^{-1} v_i \tag{4.6}$$

$$\text{or } p(v_i) \underset{e}{=} v_i' P^{-1} (.)$$

where the empty bracket signifies that the transposed expression is simply repeated.

From equations (4.4) and (4.6)

$$p(v_i) = p(\alpha_{i+1}, z_{i+1}) \tag{4.7}$$

$$= p(\alpha_{i+1} \mid z_{i+1})\, p(z_{i+1}) \tag{4.8}$$

$$\underset{e}{=} (T_\alpha \bar{\alpha}_{i+1} + T_z z_{i+1})'\, P^{-1}\, (.)$$

$$= \alpha'_{i+1} T'_\alpha P^{-1} T_\alpha \alpha_{i+1} + 2\alpha'_{i+1} T'_\alpha P^{-1} T_z z_{i+1}$$

$$+ z'_{i+1} T'_z P^{-1} T_z z_{i+1}$$

$$= (\alpha_{i+1} + P_1 T'_\alpha P^{-1} T_z z_{i+1})'\, P_1^{-1} (.)$$

$$+ z'_{i+1} [T'_z P^{-1} T_z - T'_z P^{-1} T_\alpha P_1 T'_\alpha P^{-1} T_z]\, z_{i+1} \tag{4.9}$$

where $P_1^{-1} = T'_\alpha P^{-1} T_\alpha$ (4.10)

Introducing the abbreviations

$$\alpha_1 = \alpha_{i+1} + P_1 T'_\alpha P^{-1} T_z z_{i+1} \tag{4.11}$$

and $P_z^{-1} = T'_z (P^{-1} - P^{-1} T_\alpha P_1 T'_\alpha P^{-1}) T_z$ (4.12)

it follows from equations (4.8) and (4.9) that

$$p(\alpha_{i+1} \mid z_{i+1}) \underset{e}{=} \alpha'_1 P_1^{-1} \alpha_1 \tag{4.13}$$

and $p(z_{i+1}) \underset{e}{=} z'_{i+1} P_z^{-1} z_{i+1}$ (4.14)

3. The Conditional Distribution $p(\alpha_{i+1} \mid z_{i+1}, y_{i+1})$

The information being rejected is contained in the vector α_{i+1}, and it is very relevant to take account of the next observation y_{i+1}. In other words there is no point in storing information if this same information will be available anyway in y_{i+1}. Consequently the distribution $p(\alpha_{i+1} \mid z_{i+1})$ must now be used to obtain $p(\alpha_{i+1} \mid z_{i+1}, y_{i+1})$.

From equation (2.34)

$$p(y_{i+1} \mid \mu_i) \underset{e}{=} (y_{i+1} - H\,(A\mu_i + Bu_i))\, \Sigma^{-1}\, (.) \tag{4.15}$$

where $\Sigma^{-1} = R^{-1} - R^{-1} H\, \Gamma_{i+1} H' R^{-1}$ (4.16)

With the reduced structure of equation (4.2), omitting the suffix from F_i,

$$\mu_i = F v_i$$

$$= F T_\alpha \alpha_{i+1} + F T_z z_{i+1} \tag{4.17}$$

The control vector u_i will, in the present context, be given by

$$u_i = - \Lambda \mu_i \tag{4.18}$$

where Λ is the optimal control feedback matrix obtained in Chapter 2. However the present analysis would apply whatever the origin of Λ.

From equations (4.17) and (4.18) it follows that

$$y_{i+1} - H (A\mu_i + Bu_i)$$

$$= y_{i+1} - H (A - B\Lambda) \mu_i$$

$$= y_{i+1} - H (A - B\Lambda) F (T_\alpha \alpha_{i+1} + T_z z_{i+1})$$

$$= y_{i+1} - H (A - B\Lambda) F (\alpha_{i+1} + P_1 T_\alpha' P^{-1} T_z z_{i+1})$$

$$+ H (A - B\Lambda) F (T_z - T_\alpha P_1 T_\alpha' P^{-1} T_z) z_{i+1}$$

$$= y_{i+1} + H_\alpha \alpha_1 + H_z z_{i+1} \tag{4.19}$$

where $$H_\alpha = H (A - B\Lambda) F T_\alpha \tag{4.20}$$

and $$H_z = H (A - B\Lambda) F (T_z - T_\alpha P_1 T_\alpha' P^{-1} T_z) \tag{4.21}$$

From equations (4.15), (4.19) it follows that

$$p(\alpha_{i+1}, y_{i+1} \mid z_{i+1})$$

$$= p(y_{i+1} \mid \alpha_{i+1}, z_{i+1}) \, p(\alpha_{i+1} \mid z_{i+1}) \tag{4.22}$$

$$= p(y_{i+1} \mid \mu_i) \, p(\alpha_{i+1} \mid z_{i+1})$$

$$\underset{e}{=} (y_{i+1} - H_\alpha \alpha_1 - H_z z_{i+1})' \Sigma^{-1} (.)$$

$$+ \alpha_1' P_1^{-1} \alpha_1$$

$$= (y_{i+1} - H_z z_{i+1})' \Sigma^{-1} (.) - 2 (y_{i+1} - H_z z_{i+1})' \Sigma^{-1} H_\alpha \alpha_1$$

$$+ \alpha_1' (P_1^{-1} + H_\alpha' \Sigma^{-1} H_\alpha) \alpha_1$$

$$= (\alpha_1 - P_2 H_\alpha' \Sigma^{-1} (y_{i+1} - H_z z_{i+1}))' P_2^{-1} (.)$$

$$+ (y_{i+1} - H_z z_{i+1})' (\Sigma^{-1} - \Sigma^{-1} H_\alpha P_2 H_\alpha \Sigma^{-1}) (.)$$

$$= \alpha_2' P_2^{-1} \alpha_2 + (y_{i+1} - H_z z_{i+1})' (\Sigma + H_\alpha P_1 H_\alpha')^{-1} (.) \tag{4.23}$$

where $P_2^{-1} = P_1^{-1} + H_\alpha' \Sigma^{-1} H_\alpha$ (4.24)

$$\alpha_2 = \alpha_1 - P_2 H_\alpha' \Sigma^{-1} (y_{i+1} - H_z z_{i+1}) \tag{4.25}$$

and the matrix inversion relation of equation (2.4) has been used.

Comparing equation (4.22) with the alternative expansion

$$p(\alpha_{i+1}, y_{i+1} \mid z_{i+1}) = p(\alpha_{i+1} \mid y_{i+1}, z_{i+1})\, p(y_{i+1} \mid z_{i+1}) \tag{4.26}$$

it can be deduced from equation (4.23) that

$$p(\alpha_{i+1} \mid y_{i+1}, z_{i+1}) \underset{e}{=} \alpha_2' P_2^{-1} \alpha_2 \tag{4.27}$$

and

$$p(y_{i+1} \mid z_{i+1}) \underset{e}{=} (y_{i+1} - H_z z_{i+1})' (\Sigma + H_\alpha P_1 H_\alpha')^{-1} (.) \tag{4.28}$$

These two probability distributions are required later, the first to express the discarded information in terms of information which is not discarded and the second to construct the "a priori" distribution of the information vector at time (i+1).

4. Probability Distribution of State Variables

It is now possible to proceed to determine the distribution of the state vector x_{i+1}, given the observation y_{i+1}. The analysis follows exactly as for the optimal estimator, with the exception that some information is not available from the last time point. However the result from equations (2.35) and (2.36) may be quoted.

$$p(x_{i+1} \mid y_{i+1}, \mu_i) \underset{e}{=} (x_{i+1} - \mu_{i+1})' \Gamma_{i+1}^{-1} (.) \tag{4.29}$$

where

$$\mu_{i+1} = A\mu_i + Bu_i + \Gamma_{i+1} H' R^{-1} \{y_{i+1} - H (A\mu_i + Bu_i)\} \tag{4.30}$$

and

$$\Gamma_{i+1}^{-1} = (Q + A\Gamma_i A')^{-1} + H' R^{-1} H \tag{4.31}$$

Since, from equation (4.2), μ_i is given by Fv_i and v_i is given by equation (4.4) in terms of α_{i+1} and z_{i+1} it follows that, using also the expression (4.18) for u_i,

$$\mu_{i+1}$$

$$= A_\alpha \alpha_i + A_z z_{i+1} + \Gamma_{i+1} H' R^{-1} [y_{i+1} - H_\alpha \alpha_1 - H_z z_{i+1}] \tag{4.32}$$

where

$$A_\alpha = (A - B\Lambda) FT_\alpha \tag{4.33}$$

and

$$A_z = (A - B\Lambda) F (T_z - T_\alpha P_1 T'_\alpha P^{-1} T_z) \tag{4.34}$$

Using equation (4.25), further rearrangement gives

$$\mu_{i+1}$$

$$= A_{\alpha y} \alpha_2 + A_z z_{i+1} + A_y (y_{i+1} - H_z z_{i+1}) \tag{4.35}$$

where

$$A_{\alpha y} = A_\alpha - \Gamma_{i+1} H' R^{-1} H_\alpha \tag{4.36}$$

and

$$A_y = A_\alpha P_2 H'_\alpha \Sigma^{-1} + \Gamma_{i+1} H' R^{-1} (I - H_\alpha P_2 H'_\alpha \Sigma^{-1}) \tag{4.37}$$

The mean μ_{i+1} is now in a suitable form to use in conjunction with the distribution of α_{i+1} in terms of z_{i+1} and y_{i+1} as given by equation (4.27). By the chain rule of probabilities

$$p(x_{i+1}, \alpha_{i+1} \mid z_{i+1}, y_{i+1})$$

$$= p(\alpha_{i+1} \mid x_{i+1}, z_{i+1}, y_{i+1}) \; p(x_{i+1} \mid z_{i+1}, y_{i+1}) \tag{4.38}$$

and also

$$= p(x_{i+1} \mid \alpha_i, z_{i+1}, y_{i+1}) \; p(\alpha_{i+1} \mid z_{i+1}, y_{i+1}) \tag{4.39}$$

$$= p(x_{i+1} \mid \mu_{i+1}) \; p(\alpha_{i+1} \mid z_{i+1}, y_{i+1})$$

$$\underset{e}{=} (x_{i+1} - A_{\alpha y}\, \alpha_2 - A_z\, z_{i+1} - A_y\, (y_{i+1} - H_z\, z_{i+1}))'\, \Gamma_{i+1}^{-1}\, (.)$$

$$+ \alpha_2'\, P_2^{-1}\, \alpha_2$$

$$\underset{e}{=} (x_{i+1} - \mu_{i+1}^*)\, \Gamma_{i+1}^*\, (.) - 2\alpha_2'\, A_{\alpha y}'\, \Gamma_{i+1}^{-1}\, (x_{i+1} - \mu_{i+1}^*)$$

$$+ \alpha_2'\, (P_2^{-1} + A_{\alpha y}'\, \Gamma_{i+1}^{-1}\, A_{\alpha y})\, \alpha_2$$

where

$$\mu_{i+1}^* = A_z\, z_{i+1} + A_y\, (y_{i+1} - H_z\, z_{i+1}) \tag{4.40}$$

Completing the square for the above probability distribution gives it as

$$\underset{e}{=} [\alpha_2 - P_3\, A_{\alpha y}'\, \Gamma_{i+1}^{-1}\, (x_{i+1} - \mu_{i+1}^*)]'\, P_3^{-1}\, [.]$$

$$+ (x_{i+1} - \mu_{i+1}^*)'\, (\Gamma_{i+1}^{-1} - \Gamma_{i+1}^{-1}\, A_{\alpha y}\, P_3\, A_{\alpha y}'\, \Gamma_{i+1}^{-1})\, (.)$$

$$\underset{e}{=} \alpha_3'\, P_3^{-1}\, \alpha_3 + (x_{i+1} - \mu_{i+1}^*)'\, (\Gamma_i + A_{\alpha y}\, P_2\, A_{\alpha y}')^{-1}\, (.) \tag{4.41}$$

where $P_3^{-1} = P_2^{-1} + A_{\alpha y}'\, \Gamma_{i+1}^{-1}\, A_{\alpha y}$

$$\alpha_3 = \alpha_2 - P_3\, A_{\alpha y}'\, \Gamma_{i+1}^{-1}\, (x_{i+1} - \mu_{i+1})$$

and the inversion relation of equation (2.4) has been used.

Comparing equation (4.41) with the expression (4.38) shows that

$$p(x_{i+1} \mid z_{i+1}, y_{i+1}) \underset{e}{=} (x_{i+1} - \mu_{i+1}^*)'\, \Gamma_{i+1}^{*-1}\, (x_{i+1} - \mu_{i+1}^*) \tag{4.42}$$

where $\Gamma_{i+1}^* = \Gamma_{i+1} + A_{\alpha y}\, P_2\, A_{\alpha y}'$ (4.43)

The estimator equations for determining x_{i+1} given only z_{i+1} and y_{i+1} have now been found and are given by the equation (4.40) for the conditional mean, μ_{i+1} and by the equation (4.43) for the covariance matrix, Γ_{i+1}^*.

5. Modification of Covariance Matrix due to Reduction

Equation (4.43) shows that, by limiting the quantity of information stored by the estimator, the covariance matrix of the state variable distributi is modified by the addition of a positive definite matrix, which may be expressed in a more useful form as follows

$$A_{\alpha y}\ P_2\ A'_{\alpha y}$$

$$= (A_\alpha - \Gamma_{i+1}\ H'R^{-1}\ H_\alpha)(P_I^{-1} + H'_\alpha\ \Sigma^{-1}\ H_\alpha)^{-1}\ (.)'$$

$$= \{(A-B\Lambda)\ FT_\alpha - \Gamma_{i+1}\ H'R^{-1}\ H\ (A-B\Lambda)\ FT_\alpha\}$$

$$\times\ \{T'_\alpha\ P^{-1}\ T_\alpha + T'_\alpha\ F'\ (A-B\Lambda)'\ H'\ \Sigma^{-1}\ H\ (A-B\Lambda)\ FT_\alpha\}^{-1}\ \{.\}'$$

$$= A_\Gamma\ T_\alpha\ \{T'_\alpha\ [P^{-1} + F'\ (A-B\Lambda)'\ H'\ \Sigma^{-1}$$

$$H(A-B\Lambda)\ F]\ T_\alpha\}^{-1}\ T'_\alpha\ A'_\Gamma$$

$$= A_\Gamma\ T_\alpha\ [T'_\alpha\ P_\Sigma^{-1}\ T_\alpha]^{-1}\ T'_\alpha\ A'_\Gamma$$

where the new symbols have been used,

$$A_\Gamma = (I-\Gamma_{i+1}H'R^{-1}\ H)(A-B\Lambda)F \tag{4.44}$$

and

$$P_\Sigma^{-1} = P^{-1} + F'\ (A-B\Lambda)'\ H'\ \Sigma^{-1}\ H\ (A-B\Lambda)F \tag{4.45}$$

The expression for the covariance matrix of the reduced order estimator may therefore be written, from equation (4.43)

$$\Gamma^*_{i+1} = \Gamma_{i+1} + A_\Gamma\ T_\alpha\ [T'_\alpha\ P_\Sigma^{-1}\ T_\alpha]^{-1}\ T'_\alpha\ A'_\Gamma \tag{4.46}$$

In the above derivation of equation (4.46), the various definitions of symbols given in equations (4.36), (4.24), (4.33), (4.10) and (4.20) have been used.

The significance of the expression derived above is that only T_α appears in the positive definite matrix and that it appears in this matrix in a particular manner. This will be amplified in subsequent chapters and used as the basis for choosing T_α according to a certain criterion. However the remainder of this chapter will be confined to establishing the remaining relations which are required in order to construct the complete reduced order estimating system.

6. Storage Relation

Equation (4.3) shows how z_{i+1} is constructed from the information vector v_i, that is

$$v_i = (T_z \vdots T_\alpha) \begin{bmatrix} z_{i+1} \\ \cdots \\ \alpha_{i+1} \end{bmatrix}$$

where $v_i = \begin{bmatrix} y_i \\ \cdots \\ z_i \end{bmatrix}$

It follows that, since T has been assumed non-singular,

$$\begin{bmatrix} z_{i+1} \\ \cdots \\ \alpha_{i+1} \end{bmatrix} = T^{-1} v_i$$

which gives

$$z_{i+1} = (T^{-1})_z v_i \tag{4.47}$$

where the inverse matrix has been partitioned

$$T^{-1} = \begin{bmatrix} (T^{-1})_z \\ \cdots\cdots \\ (T^{-1})_\alpha \end{bmatrix} \tag{4.48}$$

Equation (4.47) defines the storage algorithm to be implemented in a practical estimator since it defines how the stored information z_{i+1} (of dimension q) is made up from the currently available information z_i (dimension q) and y_i (dimension m).

7. New Estimate of State Vector

From equation (4.40) the new estimate of the state vector x_{i+1} is

$$\mu^*_{i+1} = (A_z - A_y H_z) z_{i+i} + A_y y_{i+1}$$

$$= A_{yz} z_{i+1} + A_y y_{i+1} \tag{4.49}$$

where $A_{yz} = (A_z - A_y H_z)$ (4.50)

F_{i+1} may therefore be defined

$$F_{i+1} = (A_y \vdots A_{yz}) \tag{4.51}$$

so that

$$\mu_{i+1} = F_{i+1} v_{i+1} \tag{4.52}$$

where

$$v_{i+1} = \begin{bmatrix} y_{i+1} \\ \cdots\cdots \\ z_{i+1} \end{bmatrix} \tag{4.53}$$

and this new matrix F_{i+1} is available to repeat the reduction process for the next time step.

The expression for A_y is given by equation (4.37) but some simplification is possible as follows

$$\begin{aligned} &A_y \\ &= A_\alpha P_2 H_\alpha' \Sigma^{-1} + \Gamma_{i+1} H' R^{-1} (I - H_\alpha P_2 H_\alpha' \Sigma^{-1}) \\ &= (I - \Gamma_{i+1} H' R^{-1} H) A_\alpha P_2 A_\alpha' H' \Sigma^{-1} + \Gamma_{i+1} H' R^{-1} \\ &= A_{\alpha y} P_2 A_\alpha' H R^{-1} (I - H \Gamma_{i+1} H' R^{-1}) + \Gamma_{i+1} H' R^{-1} \\ &= (A_{\alpha y} P_2 A_{\alpha y}' + \Gamma_{i+1}) H' R^{-1} \\ &= \Gamma_{i+1}^* H' R^{-1} \end{aligned} \tag{4.54}$$

where the various definitions given in equations (4.33), (4.20), (2.34), (4.36) and (4.43) have been used.

Similarly, from equation (4.50)

$$A_{yz} = (I - \Gamma_{i+1}^* H' R^{-1} H) A_z \tag{4.55}$$

8. <u>Prior Distribution for Next Time Step</u>

Associated with the above definition of F_{i+1} is the "a priori" probability distribution of the information vector v_{i+1} for the next time step, defined, as in equation (4.6), by

$$p(v_{i+1}) \underset{e}{=} v_{i+1}' P_{i+1}^{-1} v_{i+1} \tag{4.56}$$

or $\quad E\,\{v_{i+1}\ v'_{i+1}\} = P_{i+1}$ (4.57)

From equation (4.53) it follows that

$$P_{i+1} = \left[\begin{array}{c:c} E(y'_{i+1}\ y_{i+1}) & E(y_{i+1}\ z'_{i+1}) \\ \hdashline E(z_{i+1}\ y'_{i+1}) & E(z_{i+1}\ z'_{i+1}) \end{array}\right] \tag{4.58}$$

The values of these expected values may be obtained using equations (4.14) and (4.28) from which it is possible to write

$$E(z_{i+1}\ z'_{i+1}) = P_z \tag{4.59}$$

and

$$E\{(y_{i+1} - H_z\ z_{i+1})(y_{i+1} - H_z\ z_{i+1})'\} = P_{yz} \tag{4.60}$$

where

$$P_{yz} = \Sigma + H_\alpha\ P_1\ H'_\alpha \tag{4.61}$$

Since $(y_{i+1} - H_z\ z_{i+1})$ and z_{i+1} are independent statistical quantities* it follows that

$$E(y_{i+1} - H_z\ z_{i+1})\ z'_{i+1} = 0$$

or $\quad E(y_{i+1}\ z'_{i+1}) - H_z\ E(z_{i+1}\ z'_{i+1}) = 0$

or, using equation (4.59)

$$E(y_{i+1}\ z'_{i+1}) = H_z\ P_z \tag{4.62}$$

* Provided y_{i+1} and z_{i+1} have a jointly Gaussian distribution and

$$p(y_{i+1}|z_{i+1}) \underset{e}{=} (y_{i+1} - H_z\ z_{i+1})'\ P_{yz}^{-1}\ (.)$$

it can be shown that $(y_{i+1} - H_z\ z_{i+1})$ and z_{i+1} are independent.

From equation (4.60) it follows that

$$E(y_{i+1}\, y'_{i+1}) - \{E(y_{i+1}\, z'_{i+1})\}\, H'_z - H_z\, E(z_{i+1}\, y'_{i+1})$$

$$+ H_z\, \{E(z_{i+1}\, z'_{i+1})\}\, H_z \;=\; P_{yz}$$

and further that, using equation (4.62)

$$E(y_{i+1}\, y'_{i+1}) \;=\; P_{yy} \qquad (4.63)$$

where $P_{yy} \;=\; P_{yz} + H_z\, P_z\, H'_z$ (4.64)

Combining the above results the covariance matrix of the prior probability distribution for the next time step may be constructed, using the partitioning of equation (4.58) as

$$P_{i+1} \;=\; \left[\begin{array}{c:c} P_{yy} & H_z\, P_z \\ \hdashline P_z\, H'_z & P_z \end{array}\right] \qquad (4.65)$$

With P_{i+1} and F_{i+1} now determined for the next time step it is possible to carry out a similar reduction process for the next time step. If required this process may be continued indefinitely until an asymptotic solution is found when the various gains can be incorporated into a practical reduced order estimator algorithm.

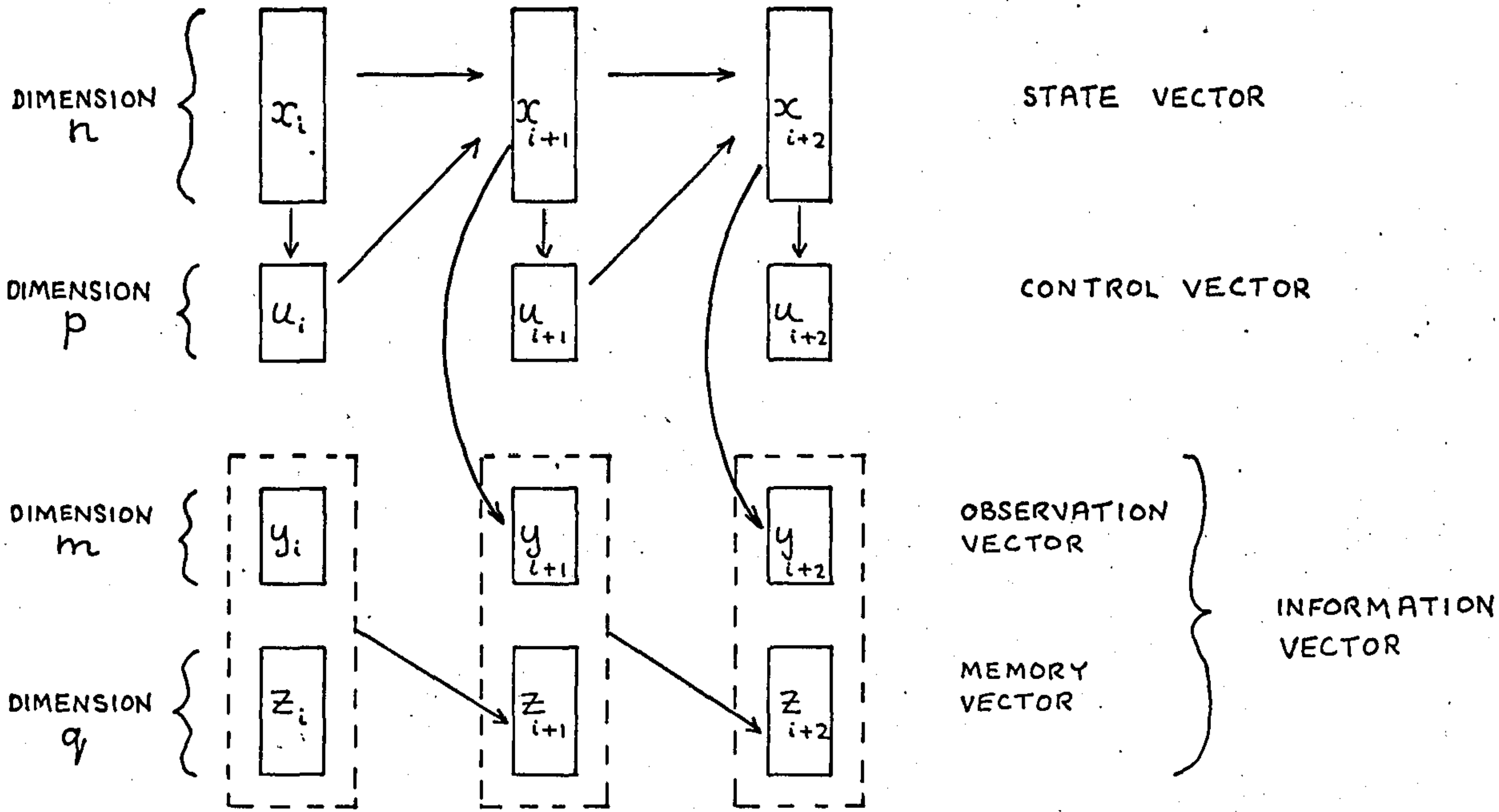


FIGURE 4.1. SCHEMATIC OF INFORMATION VECTOR REDUCTION

CHAPTER 5

THE APPLICATION OF REDUCED ORDER ESTIMATION TO CONTROL

1. The Certainty Equivalence Principle

For the case of control systems with perfect memory the certainty equivalence principle states that the optimal control strategy is that which uses the deterministic control law and derives an estimate of the state vector according to the Kalman filter. In chapter 3 the case of "pseudo-classical" control was examined for a particular example and it was found that very little extra system cost resulted from the adoption of this form of control.

The control system design method was to use the deterministic control law associated with a reduced order observer. Since the certainty equivalence pinciple will no longer apply in this situation, the control law used, say

$$u_i = -\Lambda_i \mu_i \tag{5.1}$$

where Λ_i is derived as in chaper 2, will no longer be optimal. However, in order to gain some experience in the application of this technique the example of chapter 2 will now be re-considered using a reduced order observer. In order to proceed with this a choice has to be made for the reduction matrix T.

2. Choice of Reduction Matrix T

In the expression for the reduced order covariance matrix Γ_{i+1}^{*} (equation 4.46) it is only the sub-matrix T_α of T which appears. This implies that it is only the choice of T_α which can affect the performance of the estimator. The other sub-matrix, T_z, of T will affect the structure of the estimator, that is the values of gains and the amplitudes of the elements of z_i, while not affecting the estimator performance. This can be seen for the case of the example of Chapter 2 where the optimal estimator according to Kalman filter theory had order 2 and for which it is desired to construct a reduced order estimator of order 1. Suppose it is decided that the stored information z_{i+1}(a scalar) is formed by

$$z_{i+1} = e\, z_i + f\, y_i \tag{5.2}$$

From equation (4.47) this implies

$$(T^{-1})_z = (f \quad e) \tag{5.3}$$

If the matrix T is given by

$$T = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \tag{5.4}$$

then

$$T^{-1} = \frac{1}{ad - bc} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix} \tag{5.5}$$

so that comparing (5.3) and (5.5) gives

$$\frac{d}{ad - bc} = f$$

and $\dfrac{-b}{ad - bc} = e$

It follows from (5.4) that

$$T_\alpha = \begin{pmatrix} b \\ d \end{pmatrix} = \frac{1}{ad - bc} \begin{pmatrix} -e \\ f \end{pmatrix} \tag{5.6}$$

In the expression for the covariance matrix of the reduced order estimator, equation (4.46), T_α appears equally inside and outside of the inversion. Consequently the scalar uncertainty in T_α, the factor

$$\frac{1}{ad - bc}$$

will not affect the covariance matrix Γ_{i+1}^{*}. This example has illustrated for a system of order 2 how the choice of estimator structure is equivalent to the choice of a particular T . For higher order systems this is less clear but is implied by the form of equation (4.46).

3. Example of Reduced Order Estimation

The choice of a particular matrix T will now be examined using the same example as was used to illustrate optimal control and estimation in chapter 2. Suppose the stored information is chosen simply to be the previous observation vector y_i, that is

$$z_{i+1} = y_i \tag{5.7}$$

This is a special case of the control method described by Box & Jenkins (ref. 32) where the general form of the controller is

$$u_i + a_1 u_{i-1} + a_2 u_{i-2} + \dots = b_o y_i + b_1 y_{i-1} + b_2 y_{i-2} + \dots$$

$$\text{i.e. } v_{i+1} = \begin{pmatrix} y_{i+1} \\ y_i \end{pmatrix} \tag{5.8}$$

and the conditional mean of the state vector x_{i+1} is thus formed from the two observations y_i and y_{i+1}. A suitable transformation matrix, T, is then the unit matrix so that

$$T^{-1} = T = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{5.9}$$

This choice of T satisfies equations (4.47), (5.7), and (5.8). The control law used can be taken as the steady state solution of the recursive optimal control equations for the example, that is, from Chapter 2

$$u_i = -\Lambda \mu_i$$

$$\text{where } \Lambda = (-1 \ -\tfrac{1}{2}) \tag{5.10}$$

To carry out the computing tasks associated with the generation of the reduced order estimator, a MATLAN subroutine OPRED has been written. This subroutine is suitable for any order system and is described in detail in section 4.

By calling the subroutine successively until convergence is achieved the following results are obtained

(1) The estimator covariance matrix is

$$\Gamma^*_{i\to\infty} = \begin{pmatrix} 3.058 & 0.972 \\ 0.972 & 3.108 \end{pmatrix}$$

and this compares with the optimal estimator result of

$$\Gamma = \begin{pmatrix} 3 & 1 \\ 1 & 3 \end{pmatrix}$$

(2) The prior distribution covariance matrix is

$$\underset{i\to\infty}{P} = \begin{pmatrix} 4.462 & 3.451 \\ 3.451 & 4.462 \end{pmatrix}$$

(3) The estimate of the state vector is given by

$$\mu_i = F_i v_i$$

where $\underset{i\to\infty}{F} = \begin{pmatrix} 0.243 & -0.491 \\ 0.777 & 0.173 \end{pmatrix}$

so that $\mu_1 = 0.243\, y_i - 0.491\, y_{i-1}$

$\mu_2 = 0.777\, y_i + 0.173\, y_{i-1}$

Hence from equation (5.10) it follows that

$$\begin{aligned} u_i &= -\Lambda F v_i \\ &= -(1\ \tfrac{1}{3})\begin{pmatrix} 0.243 & -0.491 \\ 0.777 & 0.173 \end{pmatrix}\begin{pmatrix} y_i \\ y_{i-1} \end{pmatrix} \\ &= -0.502\, y_i + 0.433\, y_{i-1} \end{aligned} \tag{5.11}$$

The resulting control law is seen from this equation to be of a highly derivative nature, which is as would be expected considering that the system has the form of a double integrator, and would therefore require derivative action to achieve stability. Proportional only action of the form

$$u_i = -ky_i$$

can be shown to be unstable for any value of k

When assessing the performance of a control system it is essential to know whether the system cost has been appreciably increased. To this end a further subroutine, SYSTEM, was written to calculate the system cost of any particular control system design and is described in section 5.

The cost of control for the above example is found to be 54.5 which is to be compared with a similar result for the optimal controller of chapter 2 of 54.0. In more detail the results are

	Optimal Controller	Reduced Estimator Controller
Costs on control, u	13.71	13.84
Costs on state vector x_2	40.29	40.62
Co-variance of system $\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$	$\begin{pmatrix} 8.1 & -6.1 \\ -6.1 & 40.3 \end{pmatrix}$	$\begin{pmatrix} 8.2 & -6.1 \\ -6.1 & 40.6 \end{pmatrix}$

Thus the particular choice of reduction matrix T has resulted in a control system whose performance is virtually as good as the controller of chapter 2. If a different choice of reduction matrix had been made a very much poorer system could have resulted. While the arbitrary choice of T in this example was very successful the problem remains of how to choose the most suitable matrix T. A method of selecting T in a near optimal manner is discussed in the next chapter.

4. The subroutine OPRED

The subroutine is shown in Listing 5.1. There are a number of statements whose purpose is not yet described, but are related to the material of chapter 6. Apart from these the subroutine is exclusively concerned with generating the reduced order estimator. In general a set of statements will correspond to a particular equation in the text, and

Table 5.1 sets out the functions of the subroutine statements. The nomenclature is given in Table 5.2.

For input variables the program requires

(a) System matrices A, B, H, R and control matrix Λ.

(b) The optimal covariance matrix Γ_{i+1} as calculated by ESTIM, the associated matrix Σ, the current value of the prior distribution covariance maxtrix P, and the current value of F.

Variables output by the subroutine are the reduced covariance matrix Γ^*_{i+1} the new value of the prior distribution P, and the new value of F.

5. The Computation of System Cost

The system cost for a time step has been defined in Chapter 2. as

$$J_i = x_i' Vx_i + u_i' Pu_i \tag{5.12}$$

It is required to evaluate the expected value of J_i for the general system defined by

$$x_{i+1} = Ax_i + Bu_i + \xi_i \tag{5.13}$$

$$y_i = Hx_i + \eta_i \tag{5.14}$$

$$u_i = Cz_i + Dy_i \tag{5.15}$$

$$z_{i+1} = Ez_i + Fy_i \tag{5.16}$$

Substituting

$$x_{i+1} = (A + BDH)\, x_i + BCz_i + BD\eta_i + \xi_i \tag{5.17}$$

and

$$z_{i+1} = Ez_i + FHx_i + F\eta_i \tag{5.18}$$

Defining the matrices

$$SYS = \begin{pmatrix} A+BDH & BC \\ FH & E \end{pmatrix} \tag{5.19}$$

and

$$SDIST = \begin{pmatrix} I & BD \\ 0 & F \end{pmatrix} \tag{5.20}$$

then equations (5.17) and (5.18) can be written

$$\begin{pmatrix} x \\ z \end{pmatrix}_{i+1} = SYS\begin{pmatrix} x \\ z \end{pmatrix}_i + SDIST\begin{pmatrix} \xi \\ \eta \end{pmatrix}_i \tag{5.21}$$

Again by substitution the expected cost $E(J_i)$ becomes

$$\begin{aligned} E(J_i) &= E(x_i' Vx_i + u_i' Pu_i) \\ &= E\ x_i' Vx_i + z_i' C' PCz_i \\ &+ x_i' H' D' PDHx_i + z_i' C' PDHx_i \\ &+ x_i' H' D' PCx_i + \eta_i' D' PD\eta_i \end{aligned} \tag{5.22}$$

ignoring products involving x and η, z and η since these are independent.

Defining the matrices

$$COST = \begin{pmatrix} (H' D' PDH + V) & H' D' PC \\ C' PDH & C' PC \end{pmatrix} \tag{5.23}$$

$$\text{and } DCOST = \begin{pmatrix} 0 & 0 \\ 0 & D' PD \end{pmatrix} \tag{5.24}$$

then equation (5.22) can be written

$$E(J_i) = E\left\{\begin{pmatrix} x \\ z \end{pmatrix}_i' COST \begin{pmatrix} x \\ z \end{pmatrix}_i + \begin{pmatrix} \xi \\ \eta \end{pmatrix}_i' DCOST \begin{pmatrix} \xi \\ \eta \end{pmatrix}_i\right\} \tag{5.25}$$

The co-variances of the disturbances are known so that

$$E\left\{\begin{pmatrix} \xi \\ \eta \end{pmatrix}_i \begin{pmatrix} \xi \\ \eta \end{pmatrix}_i'\right\} = VDIST = \begin{pmatrix} Q & 0 \\ 0 & R \end{pmatrix} \tag{5.26}$$

and it remains only to evaluate

$$E\left\{\begin{pmatrix} x \\ z \end{pmatrix}_i \begin{pmatrix} x \\ z \end{pmatrix}_i'\right\} = VAR_i \tag{5.27}$$

This co-variance can be evaluated using equation (5.21)

so that

$$\text{VAR}_{i+1} = E\left\{\begin{pmatrix} x \\ z \end{pmatrix}_{i+1} \begin{pmatrix} x \\ z \end{pmatrix}'_{i+1}\right\}$$

$$= E\left\{\text{SYS}\begin{pmatrix} x \\ z \end{pmatrix}_i \begin{pmatrix} x \\ z \end{pmatrix}'_i \text{SYS}'\right\}$$

$$+ E\left\{\text{SDIST}\begin{pmatrix} \xi \\ \eta \end{pmatrix}_i \begin{pmatrix} \xi \\ \eta \end{pmatrix}'_i \text{SDIST}'\right\}$$

$$= \text{SYS}.\text{VAR}_i.\text{SYS}' + \text{SDIST}.\text{VDIST}.\text{SDIST}' \qquad (5.28)$$

where the independence of the disturbance vectors has been used to set some product terms to zero.

For an asymptotically stable system VAR_i will converge, and the resulting value of VAR can be used to determine the system cost. From equation (5.25)

$$E(J_i) = \text{Trace (VAR. COST)} + \text{Trace (VDIST. DCOST)} \qquad (5.29)$$

Various subroutines were written to carry out these calculations as below:

(a) TOTSYS. Listing 5.2

Constructs SYS and SDIST according to equations 5.19 and 5.20.

(b) TOTCST. Listing 5.3

Constructs COST and DCOST according to equations. 5.23 and 5.24.

(c) TOTDST. Listing 5.4

Constructs VDIST according to equation 5.26

(d) PROD. Listing 5.4

Performs the elementwise multiplication required in equation 5.29.

(e) POWER. Listing 5.5

Carries out the iteration to compute VAR according to equation 5.28. The method was made more efficient by the use of a technique described in section 6. of this Chapter.

(f) SYSTEM. Listing 5.6

Calls the above subroutines and computes the cost according to 5.29

The subroutine SYSTEM is called by the main program in order to assess the effectiveness of any control strategy, and serves as an entirely independent check on the validity of any control strategy. For example, if a control strategy is thought to be close to the optimum, this can be checked by running SYSTEM for both cases.

For the particular control systems studied earlier in this chapter the controller is of the form

$$u_i = -\Lambda \mu^*_i$$

$$\mu^*_i = (A_z - A_y H_z)_z + A_y y_i \qquad \text{(from 4.40)}$$

which reduces to the form

$$u_i = Cz_i + Dy_i \qquad (5.15)$$

upon setting

$$C = -\Lambda (A_z - A_z H_z) \qquad (5.30)$$

and $D = -\Lambda Ay$ (5.31)

The form (5.16) for the generation of the storage vector is obtained by defining the matrices E,F by the partition

$$(F \vdots E) = (T^{-1})_z \qquad (5.32)$$

where the right hand side is the partition of the inverse of T given in equation (4.48).

The matrices C, D, E, F are constructed in subroutine OPRED for use in subroutine SYSTEM using the dummy arguments SYSC, SYSD, SYSE, SYSF.

6. A Technique for Rapid Iteration

For the above calculation of system costs it is required to find the asymptotic solution of the following equation

$$C_{i+1} = A C_i A' + B \qquad (5.33)$$

Thus if C_o is any initial positive definite matrix ($C_o = I$ say)

$$C_1 = A\,C_oA' + B$$

and

$$\begin{aligned} C_2 &= A\,C_1A' + B \\ &= A\,(A\,C_oA' + B)\,A' + B \\ &= (A\,A\,)\,C_o(AA)' + (A\,BA' + B) \\ &= A_1C_oA_1' + B_1 \quad \text{where } A_1 = AA \text{ and } B_1 = A\,BA' + B \end{aligned}$$

C_2 can be evaluated directly from C_o in one step by this means. Similarly C_4 can be evaluated from C_o by means of

$$C_4 = A_2C_oA_2' + B_2$$

where A_2, B_2 and further values of A_i and B_i are formed by means of

$$\left.\begin{aligned} A_{i+1} &= A_iA_i \\ \text{and} \quad B_{i+1} &= A_iB_iA_i' + B_i \end{aligned}\right\} \tag{5.34}$$

It is thus possible to advance to the asymptotic solution rapidly such that n evaluations of equations 5.34 allows evaluation of C_i for $i = 2^n$ according to

$$C_i = A_nC_oA_n' + B_n$$

It is this computation that is carried out by subroutine POWER, listing 5.5.

The asymptotic solution of equation (5.33) that is the solution C of the equation

$$C - A\,CA' - B = 0 \tag{5.35}$$

could be obtained by direct solution of the simultaneous equations for the elements of C, but the iterative method was used as it appeared likely to be less complex in terms of computing.

In the theory of the Liapunov stability criterion as set out by Barnett for example (reference 33) there occurs an equation of the same form as (5.35). The numerical solution of this equation is considered by Barnett and Storey (reference 34) involving first a bilinear transformation of the form

$$A' = (I + \underline{A}')(I - \underline{A}')^{-1} \tag{5.36}$$

which brings (5.35) to

$$\underline{A}' C + C\underline{A} = -\tfrac{1}{2}(I - \underline{A}) B (I - \underline{A}') \tag{5.37}$$

This is now the form of the continuous Liapunov equation for which Barnett and Storey give a number of numerical methods, both direct and using series.

7. A Relationship with Liapunov Stability Theory

Reference 33 gives the following theorem for a discrete linear system defined by

$$\mathbf{x}_{i+1} = A\mathbf{x}_i \tag{5.38}$$

"The real matrix A is convergent if and only if for any real symmetric positive definite matrix Q the solution P of the discrete Liapunov matrix equation

$$A'PA - P = -Q \tag{5.39}$$

is also positive definite".

Since A and A' have the same characteristic roots it follows that A' is convergent if and only if A is convergent and thus the Liapunov condition may be written equally as

$$APA' - P = -Q \tag{5.40}$$

The similarity of this equation with the asymptotic co-variance equation (5.35) implies that if Q is taken as the positive definite co-variance matrix of the independent Gaussian disturbance vector ξ_i of the system

$$\mathbf{x}_{i+1} = A\mathbf{x}_i + \xi_i \tag{5.41}$$

then the co-variance matrix of x_i is given by P in (5.40) and, according to the above Liapunov theorem, P will be positive definite if and only if A is convergent.

Thus negative definite matrices will occur as solutions of the asymptotic covariance equation when unstable systems are being studied. While such a result is not constructive since covariance matrices are intrinsically positive definite (or semi-definite), negative definiteness is a preferable outcome for a numerical procedure than is failure to converge, which would result using the method of section 6, above, when treating an unstable system.

This would seem to be a very strong argument in favour of direct methods of solution for the co-variance matrix of stochastic systems when it is not known in advance whether the system is stable.

Statement Numbers	Purpose of Statements	Corresponding Equations in Text
2 - 17	Computation of A_Γ	4.44
19 - 22	Computation of P_Σ	4.45
23	Specification of T. Either explicitly or by call of SIMUL as in Chapter 6.	
24 - 28	Extraction of sub matrix T_α from T	4.5
29 - 33	Construction of reduced estimator covariance Γ^*_{i+1}	4.46
35 - 36	Computation of P_1	4.10
37 - 46	Computation of P_z	4.12
47	Computation of A_z	4.34
48 - 51	Computation of A_y	4.54
53 - 55	Computation of P_{yz}	4.61
56 - 58	Computation of A_{yz}	4.55
59 - 66	Construction of F for next time step from submatrices	4.51
67	Computation of H_z	4.21 and 4.34
68 - 69	Computation of P_{yy}	4.64
70 - 82	Construction of P for next time step from submatrices	4.65
83 - 90	Computation of system matrices for use in subroutine SYSTEM	5.30 to 5.32

Table 5.1 Functions of Statements in Subroutine OPRED

Table 5.2 Nomenclature for Subroutine OPRED

Program Name	Symbol	Program Name	Symbol
B	B	P1	P_1
LAMDA	Λ	PZ	P_z
A	A	AZ	A_z
F	F	AY	A_y
H	H	PYZ	P_{yz}
R	R	AYZ	A_{yz}
HT	H'	PYY	P_{yy}
GNEXT	Γ	TM	T^{-1}
UNIT	I		
AG	A_Γ	SYSTEM MATRICES	
THETA	θ	SYSC	C
P	P	SYSD	D
PM	P^{-1}	SYSE	E
PSIGM	P_Σ	SYSF	F
SIGMA	Σ		
T	T		
TA	T_α		
TZ	T_z		
RGAM	Γ^*		

```
 1        SUBPRC    OPRED,(A,B,H,GNEXT,SIGMA,F,P,THETA,ADIM,
               ,PNEXT,R,RGAM,SYSC,SYSD,SYSE,SYSF,L)
 2        MULT      B,LAMDA,BL
 3        SUB       A,BL,AB                          LAMDA,FNEXT=
 4        CANCEL    BL
 5        MULT      AB,F,ABF
 6        MULT      H,ABF,HABF
      *
 7        DIV       R,H,RH
 8        TRANS     H,HT
 9        MULT      HT,RH,HH
10        CANCEL    HT,RH
11        MULT      GNEXT,HH,GH
      *
12        RDIM      GNEXT,XDIM
13        FORMS     UNIT,(XDIM,XDIM),(1,1),(1,1),XDIM,1.0
14        CANCEL     XDIM
15        SUB       UNIT,GH,GHU
16        CANCEL    GH
17        MULT      GHU,ABF,AG
18        CALL      TLSIDE,(AG,THETA,FTTF)
      *
19        INV       P,PM
20        INV       SIGMA,SIGM
21        CALL      TLSIDE,(HABF,SIGM,PSIGM)
22        ADD       PM,PSIGM,PSIGM
23        CALL SIMUL,(PSIGM,FTTF,T,L)
      *
24        RDIM      T,VDIM
25        SUB       VDIM,ADIM,ZDIM
26        EXSUBM    T,(1,1),(VDIM,ZDIM),TZ
27        ADD       ZDIM,1,Z1
28        EXSUBM    T,(1,Z1),(VDIM,ADIM),TA
29        CALL      TLSIDE,(TA,PSIGM,TPT)
30        INV       TPT,TPTM
31        CALL      TRSIDE,(TA,TPTM,TTPTT)
32        CALL      TRSIDE,(AG,TTPTT,GG)
33        ADD       GNEXT,GG,RGAM
34        CANCEL    TPT,TPTM,TTPTT,GG
      *
35        CALL      TLSIDE,(TA,PM,P1M)
36        INV       P1M,P1
37        CALL      TRSIDE,(TA,P1,TT)
38        MULT      TT,PM,TP
39        MULT      TP,TZ,TP
40        SUB       TZ,TP,TAZ
41        CANCEL    TT,TP
42        MULT      PM,TAZ,PTZ
43        TRANS     TZ,TTZ
44        MULT      TTZ,PTZ,PZM
45        CANCEL    PTZ,TTZ
46        INV       PZM,PZ
47        MULT      ABF,TAZ,AZ
48        INV       R,RM
49        TRANS     H,HT
```

Listing 5.1 Subroutine OPRED

In this chapter T is set to be the unit matrix so that statement 23 would read

'COPY UNIT, T'

Chapter 6 requires a call of SIMUL to generate T.

```
STMT   MATLAN STATEMENT

 50              MULT       RGAM,HT,RH
 51              MULT       RH,RM,AY
 52              CANCEL     HT,RM,RH
 53              CALL       TRSIDE,(TA,P1,TT)
 54              CALL       TRSIDE,(HABF,TT,PYZ)
 55              ADD        PYZ,SIGMA,PYZ
      *
 56              MULT       RGAM,HH,RGH
 57              SUB        UNIT,RGH,RGHU
 58              MULT       RGHU,AZ,AYZ
 59              RDIM       F,RF
 60              CDIM       F,CF
 61              NULLMAT    FNEXT,(RF,CF)
 62              CANCEL     RF,CF
 63              INSUBM     AY,FNEXT,(1,1)
 64              CDIM       AY,DY
 65              ADD        DY,1,D
 66              INSUBM     AYZ,FNEXT,(1,D)                     FNEXT FORME
      *
 67              MULT       H,AZ,HZ
 68              CALL       TRSIDE,(HZ,PZ,PYY)
 69              ADD        PYY,PYZ,PYY
 70              TRANS      HZ,HZT
 71              MULT       PZ,HZT,PHZ
 72              TRANS      PHZ,HPZ
 73              CDIM       PZ,DZ
 74              ADD        DY,1,ZBEG
 75              ADD        DY,DZ,DYZ
 76              NULLMAT    PNEXT,(DYZ,DYZ)
 77              INSUBM     PYY,PNEXT,(1,1)
 78              INSUBM     HPZ,PNEXT,(1,ZBEG)
 79              INSUBM     PHZ,PNEXT,(ZBEG,1)
 80              INSUBM     PZ,PNEXT,(ZBEG,ZBEG)                PNEXT FORME
 81              CANCEL     ZBEG,PHZ,HPZ                        AND THE RES
 82              WRITE      (RGAM,FNEXT,PNEXT),FORMAT=A5
      *
      * CALC. OF SYSTEM MATRICES .. C..D..E..F
      *
 83              MULT       LAMDA,AY,SYSD
 84              MULT       -1,SYSD,SYSD
 85              MULT       LAMDA,AYZ,SYSC
 86              MULT       -1,SYSC,SYSC
      *
 87              INV        T,TM
 88              EXSUBM     TM,(1,1),(ZDIM,DY),SYSF
 89              EXSUBM     TM,(1,D),(ZDIM,ZDIM),SYSE
 90              COPY       TM,L
 91              RETURN
 92              END
```

Listing 5.1 (continued)   Subroutine OPRED

Carries out the functions listed in Table 5.1 in order to construct the optimal estimate of the state vector when the information stored is of reduced order.

```
STMT     MATLAN STATEMENT

  1               SUBPRO     TRSIDE,(A,B,C)
  2               TRANS      A,AT
  3               MULT       A,B,D
  4               MULT       D,AT,C
  5               RETURN
  6               END
```

```
STMT     MATLAN STATEMENT

  1               SUBPRO     TLSIDE,(A,B,C)
  2               TRANS      A,AT
  3               MULT       AT,B,D
  4               MULT       D,A,C
  5               RETURN
  6               END
```

<u>Listing 5.1 (continued) Subroutine OPRED</u>

The subroutines TLSIDE and TRSIDE, above, are called by subroutine OPRED to perform the frequently required computations

TRSIDE: $A = BCB'$

TLSIDE: $A = B'CB$

```
STMT    MATLAN STATEMENT

  1              SUBPRO      TOTSYS(A,B,C,D,E,F,H,SYS,SDIST)
  2               MULT       D,H,DH
  3               MULT       B,DH,BDH
  4               ADD        A,BDH,ABDH
  5               MULT       B,C,BC
  6               MULT       F,H,FH
       *
       *
  7               RDIM       A,DX
  8               RDIM       H,DY
  9               RDIM       E,DZ
 10               CDIM       B,DU
       *
 11               ADD        DX,DZ,RDIM
 12               ADD        DX,1,RBEG
       *
 13               NULLMAT    SYS,(RDIM,RDIM)
 14               INSUBM     ABDH,SYS,(1,1)
 15               INSUBM     BC,SYS,(1,RBEG)
 16               INSUBM     FH,SYS,(RBEG,1)
 17               INSUBM     E,SYS,(RBEG,RBEG)
 18               FORMS      IDX,(DX,DX),(1,1),(1,1),DX,1.0
       *
 19               ADD        DX,DY,DDIST
       *
 20               NULLMAT    SDIST,(RDIM,DDIST)
 21               INSUBM     IDX,SDIST,(1,1)
 22               MULT       B,D,BD
 23               INSUBM     F,SDIST,(RBEG,RBEG)
 24               INSUBM     BD,SDIST,(1,RBEG)
       *
       *
 25               RETURN
 26               END
```

<u>Listing 5.2</u>   <u>Subroutine TOTSYS</u>

Constructs SYS and SDISTaccording to equations (5.19) and (5.20).

```
STMT     MATLAN STATEMENT

  1             SUBPRO     TOTCST(H,D,C,P,V,COST,DCOST)
  2             MULT       D,H,DH
  3             TRANS      DH,TDH
  4             MULT       P,DH,PDH
  5             MULT       TDH,PDH,C10
  6             ADD        C10,V,C11
  7             TRANS      C,TC
  8             MULT       TC,PDH,C21
  9             TRANS      C21,C12
 10             MULT       P,C,PC
 11             MULT       TC,PC,C22
      *
 12             RDIM       V,DX
 13             CDIM       C,DZ
 14             ADD        DX,DZ,RDIM
 15             ADD        DX,1,RBEG
 16             NULLMAT    COST,(RDIM,RDIM)
 17             INSUBM     C11,COST,(1,1)
 18             INSUBM     C12,COST,(1,RBEG)
 19             INSUBM     C21,COST,(RBEG,1)
 20             INSUBM     C22,COST,(RBEG,RBEG)
      *
 21             MULT       P,D,PD
 22             TRANS      D,TD
 23             MULT       TD,PD,D22
      *
 24             CDIM       D,DY
 25             ADD        DX,DY,DDIST
 26             NULLMAT    DCOST,(DDIST,DDIST)
 27             INSUBM     D22,DCOST,(RBEG,RBEG)
 28             RETURN
 29             END
```

<u>Listing 5.3</u>   <u>Subroutine TOTCST</u>

Constructs COST and DCOST according to equations (5.23) and (5.24).

```
STMT     MATLAN STATEMENT

                   SUBPRO      TOTDST(Q,R,VDIST)
                   RDIM        Q,DX
                   CDIM        R,DY
                   ADD         DX,DY,DDIST
                   NULLMAT     VDIST,(DDIST,DDIST)
                   INSUBM      Q,VDIST,(1,1)
                   ADD         DX,1,RBEG
                   INSUBM      R,VDIST,(RBEG,RBEG)
                   RETURN
                   END
```

```
STMT     MATLAN STATEMENT

                   SUBPRO      PROD(V,C,P)
                   EMULT       V,C,VC
                   ROWSUM      VC,ROW
                   CCLSUM      ROW,P
                   RETURN
                   END
```

Listing 5.4

Subroutine TOTSDT — Constructs VDIST according to equation (5.26)

Subroutine PROD — Performs the elementwise multiplication required in equation (5.29).

```
STMT     MATLAN STATEMENT

                  SUBPRO      POWER(INIT,AA,B,CI,NIT)
                  COPY        AA,A
                  COPY        INIT,C
                  LOOP        L2,J,1,NIT
                  TRANS       A,AT
                  MULT        C,AT,CA
                  MULT        A,CA,ACA
                  ADD         ACA,B,CI
                  MULT        B,AT,BA
                  MULT        A,BA,ABA
                  ADD         ABA,B,B
                  MULT        A,A,A
        L2        LOOPEND
                  RETURN
                  END
```

<u>Listing 5.5</u>   <u>Subroutine POWER</u>

Carries out the iteration necessary to find the asymptotic variance, VAR of the system according to equation 5.28.

```
STMT    MATLAN STATEMENT

             SUBPRO    SYSTEM(A,B,H,P,V,Q,R,C,D,E,F,VAR,
             RDIM      A,XDIM
             RDIM      C,ZDIM
             ADD       XDIM,ZDIM,VDIM            SYS,SUM,STEPS)
             NULLMAT   VINIT,(VDIM,VDIM)
             ADD       1.0,VINIT,VINIT
             CALL      TOTSYS(A,B,C,D,E,F,H,SYS,SDIST)
             CALL      TOTDST(Q,R,VDIST)
             CALL      TOTCST(H,D,C,P,V,COST,DCOST)
             TRANS     SYS,TSYS
             TRANS     SDIST,TSDIST
             MULT      VDIST,TSDIST,VTD
             MULT      SDIST,VTD,SS
        *
             CALL      POWER(VINIT,SYS,SS,VAR,STEPS)
             CALL      PROD(VAR,COST,SUMS)
             CALL      PROD(VDIST,DCOST,SUMD)
             ADD       SUMS,SUMD,SUM
             RETURN
             END
```

Listing 5.6   Subroutine SYSTEM

Calls the subroutines required to compute the asymptotic cost of the system according to equation (5.29).

CHAPTER 6

THE CHOICE OF AN OPTIMAL REDUCTION MATRIX

1. Quantifying the Cost of Reduction

In chapter 4 the covariance matrix of the state variables following the estimator reduction process is modified according to

$$\Gamma_{i+1}^{*} = \Gamma_{i+1} + A_r T_\alpha \left[T_\alpha' P_\Sigma^{-1} T_\alpha \right]^{-1} T_\alpha' A_r' \qquad (4.46)$$

The significance in terms of control must be assessed according to the theory of the optimal controller of chapter 2. At a particular time step there will be an expected cost associated with the state variable covariance matrix which is given by equation 2.26.

$$E \left\{ (x_i - \mu_i)' \Pi_{N-i} (x_i - \mu_i) \mid If_i \right\} = \text{Trace } \Gamma_i^{*} \Pi_{N-i} \qquad (6.1)$$

since

$$E \left\{ (x_i - \mu_i)(x_i - \mu_i)' \mid If_i \right\} = \Gamma_i^{*}$$

Thus it is possible to assess quantitatively the effect of the use of a reduced observer and the cost function at each time step is the matrix Π_{N-i}.

It is possible to proceed to minimise the cost associated with the reduced observer by means of the most favourable choice of the matrix T. To make the treatment more general the matrix Π_i, can be a more general cost function, θ, perhaps not derived from control theory, but quantifying the desired estimator properties.

The construction of an appropriate cost function for use in control system design is further considered in section 7. As only the last term on the right hand side of equation (4.46) can be affected by the choice of T the objective is then to minimise the cost.

$$J_i = \text{Trace}\; A_\Gamma T_\alpha \left[T_\alpha' P_\Sigma^{-1} T_\alpha\right]^{-1} T_\alpha' A_\Gamma' \theta \qquad (6.2)$$

Re-arranging equation (6.2) gives

$$\begin{aligned} J_i &= \text{Trace}\; T_\alpha' A_\Gamma' \theta A_\Gamma T_\alpha \left[T_\alpha' P_\Sigma^{-1} T_\alpha\right]^{-1} \\ &= \text{Trace}\; T_\alpha' W T_\alpha \left[T_\alpha' P_\Sigma^{-1} T_\alpha\right]^{-1} \qquad (6.3) \end{aligned}$$

where $W = A_\Gamma' \theta A_\Gamma$ (6.4)

The trace of a matrix is the sum of its eigenvalues so that when seeking a matrix T_α which will minimise the cost J it is natural to look at eigenvalue properties associated with equation (6.3). A method based on eigenvalue properties has been developed and is described below. It employs the simultaneous diagonalisation of the two positive definite symmetric matrices W and P_Σ.

2. Simultaneous Diagonalisation of Matrices

If A and B are positive definite symmetric matrices then there exists a non-singular matrix T such that

$$\left.\begin{aligned} T'AT &= I \\ \text{and} \quad T'BT &= \Lambda \end{aligned}\right\} \qquad (6.5)$$

where Λ is a diagonal matrix of positive elements. T is obtained by applying, in succession, two non-singular transformations R and S (Mirsky, reference 35).

The first, R, is chosen such that

$$R'AR = I \tag{6.6}$$

This is possible since A is positive definite. When this transformation is applied to B a new positive definite matrix C is formed

$$R'BR = C \tag{6.7}$$

The second transformation, S, is orthogonal and chosen to carry C into diagonal form, i.e.

$$S'CS = S'R'BRS = \Lambda \tag{6.8}$$

Since this transformation is orthogonal it leaves the unit matrix unchanged, so that, from (6.6)

$$S'IS = S'R'ARS = I \tag{6.9}$$

Equations (6.8) and (6.9) show that

$$T = RS$$

satisfies equations (6.5) and is the required transformation.

3. Evaluation of the Reduction Cost J_i

Since P_Σ^{-1} and W are positive definite it follows from the above that there exists a non-singular matrix T such that

$$T P_\Sigma^{-1} T = I \tag{6.10}$$

and
$$T WT = \Lambda \tag{6.11}$$

where
$$\Lambda = \begin{pmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_q \end{pmatrix} \tag{6.12}$$

Using the partitions

$$T = \left(T_z \,\vdots\, T_\alpha\right) \tag{6.13}$$

and
$$\Lambda = \left(\begin{array}{c:c} \Lambda_z & 0 \\ \hdashline 0 & \Lambda_\alpha \end{array}\right) \tag{6.14}$$

where T_z is a (q + m) x q matrix

T_α is a (q + m) x m matrix

Λ_z is a q x q diagonal matrix

Λ_α is an m x m diagonal matrix

it follows from (6.10) and (6.11) that

$$\begin{aligned} T_z' P_\Sigma^{-1} T_z &= I_p \\ T_z' P_\Sigma^{-1} T_\alpha &= 0 \\ T_\alpha' P_\Sigma^{-1} T_\alpha &= I_m \end{aligned} \tag{6.15}$$

and

$$\begin{aligned} T_z' W T_z &= \Lambda_z \\ T_z' W T_\alpha &= 0 \\ T_\alpha' W T_\alpha &= \Lambda_\alpha \end{aligned} \tag{6.16}$$

Substituting (6.15) and (6.16) into (6.3)

$$\begin{aligned} J_i &= \text{Trace } (I_m)^{-1} \Lambda_\alpha \\ &= \text{Trace } \Lambda_\alpha \end{aligned}$$

$$J_i = \lambda_{q+1} + \lambda_{q+2} + \cdots\cdots\cdots + \lambda_{q+m} \qquad (6.17)$$

This equation shows that J_i is the sum of the last m diagonal elements of Λ.

4. <u>The Minimisation of J_i</u>

The ordering of the diagonal elements occurs at equation (6.8). It is a simple matter to arrange for the sum of the last m elements to be minimum.

The elements of Λ are the eigenvalues of C and the columns of S are the eigenvectors of C. Λ and S may be constructed so that the eigenvalues of C lie in descending order along the diagonal of Λ. The smallest m eigenvalues will then lie in the last m positions of the diagonal.

T is then formed as the product RS. Since J_i is the sum of the m smallest elements T has been chosen so as to minimise J_i.

The method can be illustrated using a simple example using arbitrarily chosen elements of the matrices A and B.

$$\text{Let } P_{\Sigma}^{-1} = A = \begin{pmatrix} 2 & 1 \\ 1 & 3 \end{pmatrix}$$

$$\text{and} \qquad W = B = \begin{pmatrix} 5 & 4 \\ 4 & 6 \end{pmatrix} \qquad (6.18)$$

The eigenvalues of A are

1.382 and 3.618

and the corresponding eigenvectors are

$$\begin{pmatrix} 0.8507 \\ -0.5257 \end{pmatrix} \text{ and } \begin{pmatrix} 0.5257 \\ 0.8507 \end{pmatrix}$$

Dividing these eigenvectors by the square roots of their corresponding eigenvalues gives the columns of the matrix R.

Thus $R = \begin{pmatrix} 0.7234 & 0.2764 \\ -0.4470 & 0.4473 \end{pmatrix}$

so that

$$R'AR = I$$

and

$$C = R'BR = \begin{pmatrix} 1.228 & 0.6003 \\ 0.6003 & 2.571 \end{pmatrix}$$

The eigenvalues of C are

2.80 and 1.00

and the corresponding eigenvectors are

$$\begin{pmatrix} 0.9342 \\ -0.3567 \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} 0.3567 \\ 0.9342 \end{pmatrix}$$

S is now constructed so that the eigenvalues are in descending order along the diagonal of Λ, that is

$$S'CS = \begin{pmatrix} 2.80 & 0 \\ 0 & 1.00 \end{pmatrix} = \Lambda \qquad (6.19)$$

where $S = \begin{pmatrix} 0.3567 & 0.9342 \\ 0.9342 & -0.3567 \end{pmatrix}$

T is now constructed as the product

$$T = RS = \begin{pmatrix} 0.5162 & 0.5772 \\ 0.2584 & -0.5771 \end{pmatrix} \qquad (6.20)$$

so that $T_\alpha = \begin{pmatrix} 0.5772 \\ -0.5771 \end{pmatrix}$

and the minimum value of J_i is

$$J_i = \text{Trace}\,(T'_\alpha\, P_\Sigma^{-1}\, T_\alpha)^{-1}\, T'_\alpha\, WT_\alpha = 1$$

In general there may be other methods of choosing the T to minimise the reduction cost J_i by the use of hill-climbing subroutines. These methods would involve a number of variables approaching the number of elements of T_α and clearly the eigenvalue method involves working more directly and with a smaller number of variables. However for the above example T_α involves only two variables and a suitable T can be found by an algebraic method which is comparable to the hill-climbing method.

5. <u>The Optimum Choice of $\bar{T}$ from the minima of J_i</u>

$$\text{If } T = (T_\Sigma \vdots T_\alpha) = \begin{pmatrix} T_{11} & T_{12} \\ T_{21} & T_{22} \end{pmatrix}$$

then $$J_i = \text{Trace } (T'_\alpha P_\Sigma^{-1} \; T_\alpha)^{-1} \, T'_\alpha W T_\alpha$$

$$= (2T_{12}^2 + 2T_{12}T_{22} + 3T_{22}^2)^{-1}(5T_{12}^2 + 8T_{12}T_{22} + 6T_{22}^2)$$

If $r = \dfrac{T_{12}}{T_{22}}$ then

$$J_i = \frac{5r^2 + 8r + 6}{2r^2 + 2r + 3}$$

Differentiating to find stationary values

$$\frac{dJ_i}{dr} = \frac{(2r^2 + 2r + 3)(10r + 8) - (5r^2 + 8r + 6)(4r + 2)}{(2r^2 + 2r + 3)^2}$$

$$= \frac{-6(r + 1)(r - 2)}{(2r^2 + 2r + 3)^2} \, .$$

$$= 0 \text{ when } r = -1 \text{ and } r = 2$$

A graph of J_i against r, as in Figure 6.1, shows r = 2 to be a maximum and r = -1 to be the required minimum of J_i.

This minimum value of J_i is 1.

Comparing this result with the eigenvalue method in which, from equation (6.20),

$$T_{12} = 0.5772$$

and $$T_{22} = -0.5771$$

so that $r = \dfrac{T_{12}}{T_{22}} = -1.00$

it can be seen that the two results are in complete agreement.

6. Computation of a Cost Function for Control Application

In Chapter 2 the choice of the control vector u_i was shown to be optimal when chosen as

$$u_i = -\Lambda_i \mu_i \quad . \qquad \text{(equation 2.22)}$$

Since x_i is not known exactly as for the deterministic case, the cost incurred i using u_i as the estimate of x_i instead of x_i itself is quantified by the cost

$$J_i = E\left[(x_i - \mu_i)' \Pi (x_i - \mu_i)\right] \qquad 6.21$$

where the suffix is omitted from Π as it can be assumed to have reached an asymptotic value.

The process of introducing a reduced order estimator necessarily increases the uncertainty in the state x_i and so contributes to an increased system cost. This can be quantified directly from equation (6.21). However to do this would be to overlook the indirect effects of the reduction. These are

1. The uncertainty introduced into the estimate for x_i will cause uncertainties to be introduced into the estimates of x_{i+1}, x_{i+2}, etc. This is apart from the uncertainties introduced by reductions at time i+1, i+2, etc.
2. The choice of u_i according to2.22 may no longer be optimal as the control vector is able to influence the estimation process as a result of the estimator reduction.

The second effect has been discussed in Chapter 3, where an example showed how, even though the certainty equivalence principle no longer strictly applied, the use of a control law according to the principle was likely to give near optimal results. While reduction costs remain small it is unlikely that this choice of the control vector will be downgrading the system significantly.

For this reason further control system study will use the control law given by 2.22.

The first effect, above, implies that the future effects of reduction must be taken into account by means of the cost function. Suppose a suitable cost function is given by

$$J_i = E\left\{(x_i-\mu_i)'\Theta_i(x_i-\mu_i)\right\} \qquad (6.22)$$

Using the dynamic programming approach it follows that

$$E\left\{(x_i-\mu_i)'\Theta_i(x_i-\mu_i)\right\}$$
$$= E\left\{(x_{i+1}-\mu_{i+1})'\Theta_{i+1}(x_{i+1}-\mu_{i+1})\right\} + E\left\{(x_i-\mu_i)'\Pi(x_i-\mu_i)\right\} \qquad (6.23)$$

A recursive equation for Θ_i can now be set up if the relation between $(x_{i+1}-\mu_{i+1})$ and $(x_i-\mu_i)$ is known. From equation (2.35) which defines the optimal estimator structure

$$\begin{aligned}(x_{i+1}-\mu_{i+1}) &= Ax_i + Bu_i + \xi_i \\ &\quad - \left[A\mu_i + Bu_i + \Gamma_{i+1}H'R^{-1}\left(y_{i+1} - H(A\mu_i+Bu_i)\right)\right] \\ &= (I-\Gamma_{i+1}H'R^{-1}H)A(x_i-\mu_i) \\ &\quad + (I-\Gamma_{i+1}H'R^{-1}H)\xi_i - \Gamma_{i+1}H'R^{-1}\eta_i \qquad (6.24)\end{aligned}$$

Substituting into (6.23) and taking expectations, when the terms involving the independent quantities ξ and η vanish, gives

$$\Theta_i = \Pi + A'(I-\Gamma_{i+1}H'R^{-1}H)'\Theta_{i+1}(I-\Gamma_{i+1}H'R^{-1}H)A \qquad (6.25)$$

A subroutine ESTIM2 was written to obtain the asymptotic solution of this recursive equation and is given in listing 6.1. Using the value of Π from the example system of

$$\Pi = \begin{pmatrix} 9 & 3 \\ 3 & 1 \end{pmatrix}$$

and iterating, the subroutine gives an asymptotic solution for θ of

$$\theta = \begin{pmatrix} 20.3 & -.74 \\ -.74 & 2.52 \end{pmatrix} \tag{6.26}$$

As this is identical with the solution which can be obtained analytically as

$$\theta = \frac{4}{27}\begin{pmatrix} 137 & -5 \\ -5 & 17 \end{pmatrix}$$

the example also serves as a check on the programming of ESTIM2.

The calculation of θ has assumed the estimator structure to be optimal, that is with no reduction, and this is not the case as θ will be used as a cost function for the reduction process. However, if the reduction is efficient and the covariance matrices are not greatly altered, the value of θ calculated in this way is likely to be a very relevant cost matrix. If there were sufficient reason in a particular application it could be worthwhile to extend the derivation of θ to include a reduced estimator. This point is discussed further in chapter 10 (section 4).

7. Application of Control Cost Function to the Example System

Having established a suitable value of the cost function it is possible to apply the method of simultaneous diagonalisation of quadratic forms to the choice of the reduction matrix T. The computation is handled by the subroutine SIMUL which is called from subroutine OPRED, which constructs the reduced estimator and has been described in Chapter 5. Subroutine SIMUL performs the matrix transformations and calls a Fortran eigenvector subroutine to perform the necessary diagonalisations. The details of this are given in Section 8.

The construction of the reduced order estimator is now identical to the method of Chapter 5, with θ being supplied as data. After ten iterations all quantities have converged, giving:

$$\Gamma = \begin{pmatrix} 3.01 & 1.003 \\ 1.003 & 3.08 \end{pmatrix}$$

and the control system

$$u_i = 1.83\, z_i - .508\, y_i \tag{6.27}$$

$$z_{i+1} = -\,.0343\, z_i + .246\, y_i \tag{6.28}$$

This control system can be seen to be similar to that of (5.11), but the performance is slightly better, as shown by:

	Optimum Controller	Reduced order estimator using SIMUL
Costs on control, u	13.71	13.75
Costs on state vector x_2	40.29	40.54
Co-variance of system $\begin{pmatrix} x_i \\ x_2 \end{pmatrix}$	$\begin{pmatrix} 8.1 & -6.1 \\ -6.1 & 40.3 \end{pmatrix}$	$\begin{pmatrix} 8.2 & -6.1 \\ -6.1 & 40.5 \end{pmatrix}$

The eigenvalues of the diagonal matrix obtained by SIMUL are 94 and 0.2, demonstating that the cost of reduction is only likely to be 0.2 per time step if a reduced order estimator is used, and this is confirmed by running the subroutine SYSTEM, which gives an independent check, and calculates the above costs. What the example has shown is that the method of simultaneous diagonalisation can construct a reduced order observer with a performance very close to that of the optimal system.

More fully the system is obtained by the simultaneous diagonalisation of (from 6.4)

$$P_{\Sigma}^{-1} = \begin{pmatrix} 0.12 & -0.27 \\ -0.27 & 1.10 \end{pmatrix} \quad \text{and} \quad A'_{\Gamma}\theta A_{\Gamma} = \begin{pmatrix} 5.67 & -0.84 \\ -0.84 & 0.33 \end{pmatrix} \qquad (6.29)$$

giving the transformation matrix

$$T = \begin{pmatrix} 4.21 & 0.14 \\ 1.00 & 0.99 \end{pmatrix} \qquad (6.30)$$

which upon inversion, implies

$$\begin{pmatrix} z_{i+1} \\ \alpha_{i+1} \end{pmatrix} = T^{-1} v_i = \begin{pmatrix} 0.25 & -0.034 \\ -0.25 & 1.05 \end{pmatrix} \begin{pmatrix} y_i \\ z_i \end{pmatrix} \qquad (6.31)$$

giving the relation (6.28).

The system mean is generated from

$$\mu_i = Fv_i = \begin{pmatrix} 0.25 & 2.10 \\ 0.77 & 0.81 \end{pmatrix} \begin{pmatrix} y_i \\ z_i \end{pmatrix} \qquad (6.32)$$

Upon setting the control law

$$u_i = (-\tfrac{1}{3} \;\; -1)\,\mu_i \qquad (6.33)$$

this relation gives the control equation (6.27).

8. The subroutine SIMUL

Listing 6.2 gives the subroutine SIMUL and the main program used to test it using the test matrices A and B of equation 6.18. Subroutine SIMUL calls the Fortran subroutines UNIMAT and EIGMAT, given in listings 6.3 and 6.4 respectively and these each call the Fortran subroutine EIGEN which obtains the eigenvalues and eigenvectors of a symmetric matrix. A listing of EIGEN, which is part of the IBM Scientific Subroutine Package, is given in Appendix 2, together with the job control statements for running MATLAN in conjunction with Fortran and listings of subroutines LOC and MSTR which convert matrices from two dimensional arrays to single dimension arrays and conversely.

UNIMAT constructs the matrix R which transforms A to a unit matrix according to equation (6.6) and EIGMAT constructs the matrix S which diagonalises $R'BR$ according to equation 6.8. The diagonal matrix so obtained is also returned by EIGMAT with eigenvalues in descending order of magnitude, this being a consequence of the operation of EIGEN, so that when SIMUL returns the matrix T as the product RS this will be as required in section 4 for the choice of the reduction matrix.

When the matrices A and B of 6.18 are read by the main program of listing 6.2 the output is as shown in listing 6.5. The matrix T is precisely that of equation 6.20, and the diagonal matrix is precisely that of equation 6.19, giving a check on the programming of SIMUL and associated subroutines. Also given in the output are the products $T'AT$ and $T'BT$ which demonstrate that the matrix T does indeed transform the matrices A and B as required.

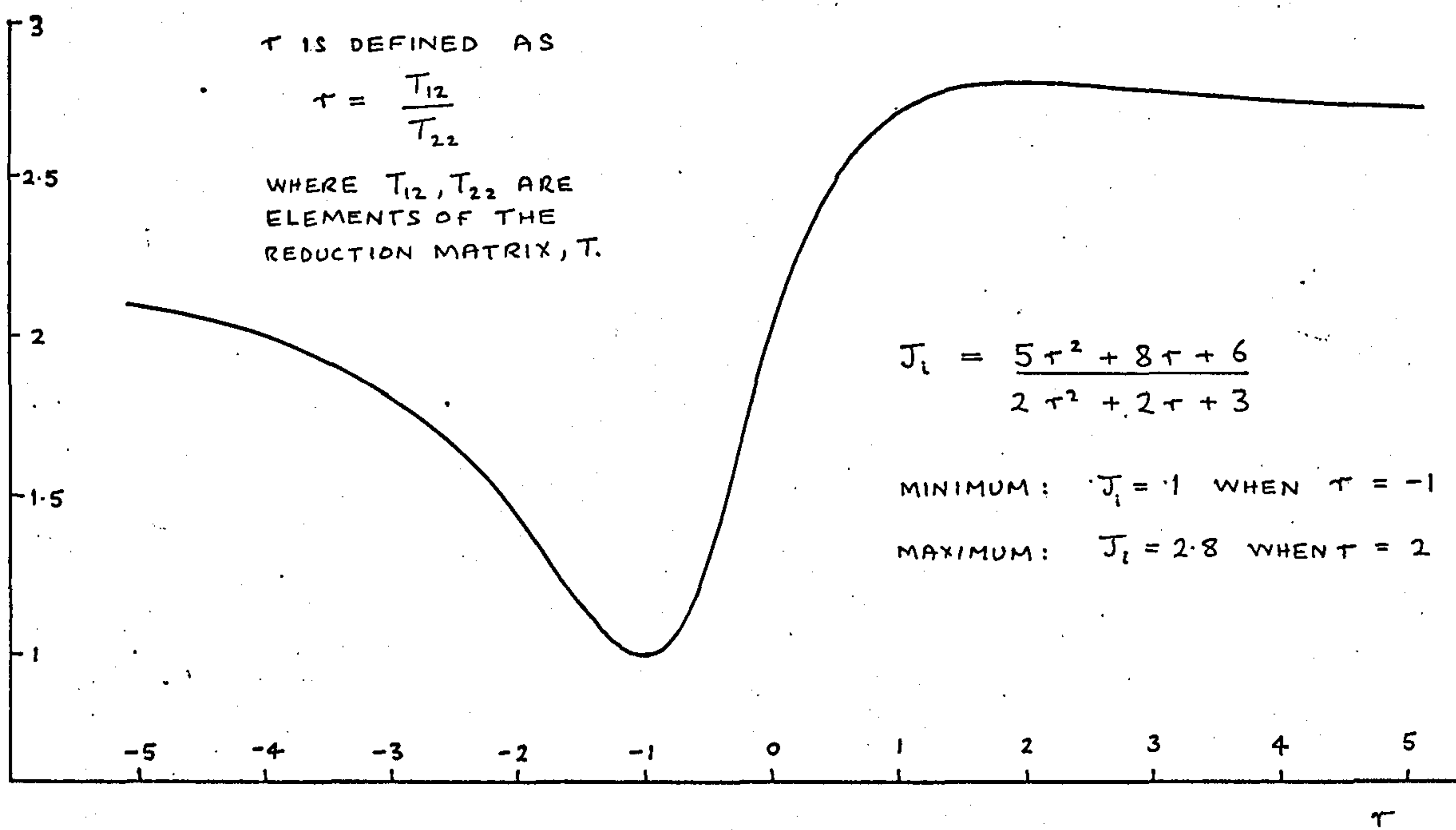


FIGURE 6.1 STATIONARY VALUES OF THE COST FUNCTION J_i

```
MATLAN STATEMENT

          SUBPRO    ESTIM2(A,B,H,Q,R,G,GNEXT,SIGMA,PI,THETA)
          TRANS     A,AT
          MULT      G,AT,GA
          MULT      A,GA,AA
          ADD       Q,AA,GQ
          MULT      H,GQ,HG
          TRANS     H,HT
          MULT      HG,HT,HH
          ADD       R,HH,SIGMA
          DIV       SIGMA,HG,RG
          TRANS     HG,HT
          MULT      HT,RG,GG
          SUB       GQ,GG,GNEXT
          DIV       R,H,RH
          TRANS     H,HT
          MULT      HT,RH,HH
          MULT      GNEXT,HH,GH
          SUB       1.0,GH,IG
          MULT      IG,A,IA
          MULT      THETA,IA,TA
          TRANS     IA,IAT
          MULT      IAT,TA,ATA
          ADD       PI,ATA,THETA
          RETURN
          END
```

<u>Listing 6.1 Subroutine ESTIM 2</u>

Given the control cost matrix Π the subroutine computes the recursive equation for θ given by (6.22).

```
STMT    MATLAN STATEMENT

  1              MAIN
  2              READ       (A,B)
  3              WRITE      (A,B),FORMAT=A5
  4              CALL       SIMUL(A,B,T,X)
  5              WRITE      (T,X),FORMAT=A5
  6              TRANS      T,AT                               **********
  7              MULT       AT,A,TA                            *   CHECK
  8              MULT       TA,T,TAT                           **********
  9              MULT       AT,B,TB
 10              MULT       TB,T,TBT
 11              WRITE      (TAT,TBT),FORMAT=A5
 12              END
```

```
STMT    MATLAN STATEMENT

  1              SUBPRO     SIMUL(A,B,T,X)          X WILL BE VECTOR
  2              RDIM       A,D
  3              ALLOCATE   R,(D,D)
  4              CALL       UNIMAT(A,R,D),F          OF EIGENVALUES
  5              TRANS      R,RT
  6              MULT       RT,B,RTB
  7              MULT       RTB,R,C
  8              WRITE      (R,C),FORMAT=A5
  9              ALLOCATE   S,(D,D)
 10              ALLOCATE   X,(D,1)
 11              CALL       EIGMAT(C,S,X,D),F
 12              MULT       R,S,T
 13              WRITE      (S,T,X),FORMAT=A5
 14              RETURN
 15              END
```

<u>Listing 6.2 Subroutine SIMUL</u>

Finds the transformation matrix T which simultaneously diagonalises the symmetric matrices A and B. The main program reads the test matrices A and B and calculates $T'AT$ and $T'BT$ to check the subroutine. The output is given in listing 6.5.

```
FORTRAN IV G LEVEL 1, MOD 3                UNIMAT                DATE = 70169

                  SUBROUTINE UNIMAT(AAA,RR,XN)
                  DIMENSION AAA( 2, 2),AA( 4),A( 3),R( 4),RR( 2, 2)
      C
      C     THIS NEEDS NEW NUMBERS FOR OTHER CASES
      C
                  N=XN
                  IA=1
                  DO 30 J=1,N
                  DO 40 I=1,N
                  AA(IA)=AAA(I,J)
                  IA=IA+1
               40 CONTINUE
               30 CONTINUE
                  CALL MSTR(AA,A,N,0,1)
                  CALL EIGEN(A,R,N,C)
                  IA=1
                  DO 50 J=1,N
                  DO 60 I=1,N
                  RR(I,J)=R(IA)
                  IA=IA+1
               60 CONTINUE
               50 CONTINUE
                  DO 10 J=1,N
                  CALL LOC(J,J,IA,N,N,1)
                  RT=SQRT(A(IA))
                  DO 20 I=1,N
                  RR(I,J)=RR(I,J)/RT
               20 CONTINUE
               10 CONTINUE
                  RETURN
                  END
```

Listing 6.3 Fortran Subroutine UNIMAT

Subroutine UNIMAT is called by SIMUL in order to construct the transformation matrix R which converts matrix A to the unit matrix.

The operation of the storage conversion subroutines LOC and MSTR is described in Appendix 2.

```
FORTRAN IV G LEVEL 1, MOD 3                 EIGMAT                 DATE = 70169

              SUBROUTINE EIGMAT(CCC,SS,XL,XN)
              DIMENSION XL(2)
              DIMENSION CCC( 2, 2),CC( 4),C( 3),S( 4),SS( 2, 2),
        C
        C  THIS NEEDS NEW NUMBERS FOR OTHER CASES                   L( 2)
        C
              N=XN
              IA=1
              DO 30 J=1,N
              DO 40 I=1,N
              CC(IA)=CCC(I,J)
              IA=IA+1
           40 CONTINUE
           30 CONTINUE
              CALL MSTR(CC,C,N,0,1)
              CALL EIGEN(C,S,N,0)
              IA=1
              DO 50 J=1,N
              DO 60 I=1,N
              SS(I,J)=S(IA)
              IA=IA+1
           60 CONTINUE
           50 CONTINUE
              DO 10 I=1,N
              CALL LOC(I,I,IC,N,N,1)
           10 XL(I)=C(IC)
              RETURN
              END
```

Listing 6.4 Fortran Subroutine EIGMAT

Subroutine EIGMAT is called by SIMUL in order to construct the transformation matrix S which diagonalises the matrix $R'BR$. The diagonal elements are returned in the vector XL.

```
T                         DIMENSIONS = (  2,  2)

                  1               2

     1     5.1640E-01    5.7735E-01
     2     2.5820E-01   -5.7735E-01

           END OF MATRIX T


X                         DIMENSIONS = (  2,  1)

                  1

     1     2.8000E 00
     2     1.0000E 00

           END OF MATRIX X


TAT                       DIMENSIONS = (  2,  2)

                  1               2

     1     1.0000E 00    5.5060E-07
     2     3.0972E-07    1.0000E 00

           END OF MATRIX TAT


TBT                       DIMENSIONS = (  2,  2)

                  1               2

     1     2.8000E 00    0.0
     2     3.0972E-07    1.0000E 00

           END OF MATRIX TBT
```

<u>Listing 6.5 Output for SIMUL Test Matrices</u>

Given the test matrices of (6.18) as input, listing 6.2 gives the above output, checking that the matrix T does indeed diagonalise matrices A and B. The diagonal elements are given in the matrix X.

CHAPTER 7

A ONCE-THROUGH BOILER MODEL

1. Model Description

The theory and associated MATLAN subroutines developed in previous chapters are sufficient to allow the construction of a control system whose order may be chosen to be that of the plant itself, as dictated by optimal control theory and the Kalman filter, or whose order may be chosen to be lower. When considering the control of a plant with a large number of state variables, that is with a high order state vector it is to be expected that the order of the control system can be substantially reduced without any significant penalty in terms of control system performance.

Such a high order plant is typically found in distributed parameter systems since an adequate description of these can only be made by subdivision into a large number of small elements each of which will be represented by one or more state variables.

The example used so far has the smallest possible dimensions and for testing the reduction method developed in previous chapters a model with a larger dimension is required. The model chosen is developed from the finite difference form of the equations for a once-through boiler. While many models of large dimension would be suitable, an advantage of this model is that it is defined by relatively few parameters. Simplifications are made, reducing the model to the equations for a counter-flow heat exchanger. In this form the weighting coefficients determined by optimal control theory, for example, produce recognisable patterns. The form of the heat exchanger is shown in Figure 7.1. The transit times of the two fluids, steam and gas respectively, are assumed to be small, so that at any instant these streams have achieved equilibrium with the local tube metal. The time constants of the system are associated with the time constants of the tube metal. The metal tube is assumed to be thin so that its temperature throughout is defined by one temperature.

2. Finite Difference Model

2.1 Steam Equations

Defining

Steam flow rate	=	WS	kg sec^{-1}
Steam specific heat	=	SHS	J.kg^{-1} $^{o}C^{-1}$
Heat transfer coefficient	=	K	J.$^{o}C^{-1}$ sec^{-1}
Tube metal temperature perturbation	=	T(N)	^{o}C
Steam temperature perturbation	=	TS(N)	^{o}C
Heat Flow into steam	=	Q	J sec^{-1}

where N is the section number, the equations for the steam side are:

(i) Heat Transfer

$$Q = K\left(T(N) - \frac{TS(N) + TS(N+1)}{2}\right) \qquad (7.1)$$

(ii) Temperature Rise

$$Q = WS.\ SHS\left[TS(N) - TS(N+1)\right] \qquad (7.2)$$

Eliminating Q from the above equations gives

$$AS5.TS(N+1) - AS6.TS(N) = T(N) \qquad (7.3)$$

$$\text{where } AS5 = \frac{WS.SHS}{K} + \tfrac{1}{2} \qquad (7.4)$$

$$\text{and } AS6 = \frac{WS.SHS}{K} - \tfrac{1}{2} \qquad (7.5)$$

The complete steam side may be represented in matrix form, taking TS(1), the inlet steam temperature perturbation as zero, as

$$\underline{S1.TS = T} \qquad (7.6)$$

$$\text{where } S1 = \begin{pmatrix} AS5 & 0 & \cdots & 0 \\ -AS6 & AS5 & \ddots & \vdots \\ 0 & \ddots & AS5 & 0 \\ 0 & 0 & -AS6 & AS5 \end{pmatrix} \qquad (7.7)$$

$$TS = \begin{pmatrix} TS(2) \\ TS(3) \\ \vdots \\ TS(N+1) \end{pmatrix} \quad \text{and} \quad T = \begin{pmatrix} T(1) \\ T(2) \\ \vdots \\ T(N) \end{pmatrix} \qquad (7.8)$$

An explicit solution for steam temperature can be obtained from (7.6) as

$$TS = S1^{-1}.T. \qquad (7.9)$$

$$\text{or } TS = S2.T \qquad (7.10)$$

$$\text{where } S2 = S1^{-1} \qquad (7.11)$$

The construction of S2 is carried out by subroutine OTBOIL which is given in listing 7.1

Mean steam temperature in any section is given by:

$$\overline{TS}(N) = \tfrac{1}{2}\,TS(N + 1) + \tfrac{1}{2}TS(N)$$

$$\text{or } \overline{TS} = S4.TS \qquad (7.12)$$

$$\text{where } S4 = \begin{bmatrix} 0.5 & 0 & & & \\ 0.5 & 0.5 & & & \\ 0 & \ddots & \ddots & & \\ & & \ddots & \ddots & \\ 0 & & & 0.5 & 0.5 \end{bmatrix} \qquad (7.13)$$

From equations (7.12) and (7.10) mean steam temperature is given in terms of metal temperature by:

$$\overline{TS} = S5.T \qquad (7.14)$$

$$\text{where } S5 = S4.S2 \qquad (7.15)$$

S4 and S5 are also constructed by OTBOIL.

2.2 Gas equation

For the gas side an entirely similar derivation is possible, with the exception that fluid flow is reversed and the fluid inlet temperature is not assumed constant but is taken as the control input to the system.

The heat transfer and temperature rise equations are:

$$QG = KG\left\{\left[\frac{TG(N + 1) + TG(N)}{2}\right] - T(N)\right\} \qquad (7.16)$$

$$QG = WG.SHG\left[TG(N + 1) - TG(N)\right] \qquad (7.17)$$

and these lead, as before, to

$$AG5.TG(N) - AG6.TG(N + 1) = T(N) \qquad (7.18)$$

where

$$AG5 = \frac{WG.SHG}{KG} + \tfrac{1}{2} \qquad (7.19)$$

$$AG6 = \frac{WG.SHG}{KG} - \tfrac{1}{2} \qquad (7.20)$$

and WG = Gas flow rate

SHG = Gas specific heat

QG = Heat flow from gas to metal

KG = Heat transfer coefficient for gas side

2.3 Tube metal equation

If the element of tube wall has thermal capacity, C J.$^{o}C^{-1}$ then

$$C \frac{d\overline{T}(N)}{dt} = QG - QS \tag{7.31}$$

Substituting for the heat flows from equations (7.1) and (7.16) gives

$$C \frac{d\overline{T}(N)}{dt} = KG \left[\overline{TG}(N) - T(N)\right] - K \left[T(N) - \overline{TS}(N)\right]$$

or

$$\frac{d\overline{T}(N)}{dt} = AM1.\overline{TG}(N) + AM2.\overline{TS}(N) - AM3.T(N) \tag{7.32}$$

$$\left.\begin{array}{l} \text{where } AM1 = \frac{KG}{C}, \; AM2 = \frac{K}{C} \\ \text{and } \quad AM3 = AM1 + AM2 \end{array}\right\} \tag{7.33}$$

Substituting for $\overline{TG}(N)$ and $\overline{TS}(N)$ from (7.12) and (7.29) gives, in matrix form

$$\frac{d}{dt} T = AM1 \left[G5.T + G6.TG(NB + 1)\right] + AM2 \left[S5.T - AM3.T\right] \tag{7.34}$$

or

$$\frac{d}{dt} T = ACON.T + BCON.TG(NB + 1) \tag{7.35}$$

$$\left.\begin{array}{l} \text{where } ACON = AM1.G5 + AM2.S5 - AM3.I \\ \text{and } BCON = AM1.G6 \end{array}\right\} \tag{7.36}$$

Equation (7.35) now describes the plant dynamics in the conventional form

$$\dot{x} = A_c x + B_c u \tag{7.37}$$

and this continuous differential equation requires to be converted to a discrete time form in order to apply the control theory developed in previous chapters.

The generation of ACON, BCON and the other related matrices are carried out by subroutine OTBOIL, Listing 7.1. The matrices so generated using the data of section 5 are shown in Fig. 7.2

3. Discrete Time Model

Using the Crank -Nicholson approximation given in reference 36 gives for equation (7.37)

$$\frac{x_{i+1} - x_i}{\Delta t} = \frac{A_c x_{i+1} + B_c u_{i+1} + A_c x_i + B_c u_i}{2}$$

$$\text{or } (I - \frac{\Delta t}{2} A_c) x_{i+1} = (I + \frac{\Delta t}{2} A_c) x_i + B_c . \Delta t . u_i$$

where u_i is assumed constant throughout the interval. The discrete form is then

$$x_{i+1} = Ax_i + Bu_i \tag{7.38}$$

$$\text{where } A = (I - \frac{\Delta t}{2} A_c)^{-1} (I + \frac{\Delta t}{2} A_c) \tag{7.39}$$

$$\text{and } B = (I - \frac{\Delta t}{2} A_c)^{-1} B_c . \Delta t \tag{7.40}$$

The construction of matrices A and B is carried out in subroutine CRANK which is shown in Listing 7.2.

In a time interval the control variable u is assumed constant. Using this fact it is possible to subdivide the interval to give a more accurate finite difference representation. In such a sub-division

$$x_{i+1} = Ax_i + Bu$$

$$\begin{aligned} \text{and } x_{i+2} &= Ax_i + Bu \\ &= A.Ax_i + A.Bu + Bu \\ &= A.Ax_i + (A+I)Bu \end{aligned}$$

Repeating this process NSUB times allows the system matrices to be built up and the programming necessary is shown in Listing 7.2.

Matrices A and B generated as above with NSUB = 8 and a time step of 2 seconds using the data of section 5 are shown in Figure 7.3.

A check that NSUB = 8 was sufficient subdivision was confirmed by comparison with the transition matrix derived directly from the continuous time matrices as in section 7.7(b).

In matrix form

$$\begin{pmatrix} AG5 & -AG6 & 0 \\ 0 & AG5 & \\ 0 & & \\ & & \\ 0 & & AG5 \end{pmatrix} \cdot \begin{pmatrix} TG(1) \\ \\ \\ \\ TG(N) \end{pmatrix} = \begin{pmatrix} T(1) \\ \\ \\ \\ T(N) \end{pmatrix} + \begin{pmatrix} 0 \\ \\ \\ 0 \\ AG6 \end{pmatrix} . \; TG(NB+1) \qquad (7.21)$$

or

$$G1.TG = T + GU.TG(NB+1) \qquad (7.22)$$

where the matrices are suitably defined.

Further

$$TG = G2.T + G3.TG(NB + 1) \qquad (7.23)$$

$$\text{where } G2 = G1^{-1} \qquad (7.24)$$

$$\text{and } G3 = G1^{-1}.GU \qquad (7.25)$$

Mean gas temperature is given by

$$\overline{TG} = G4.TG + ADDN.TG(NB + 1) \qquad (7.26)$$

where

$$G4 = \begin{pmatrix} 0.5 & 0.5 & & 0 \\ 0 & \ddots & \ddots & \\ & & \ddots & \\ 0 & & & 0.5 \end{pmatrix} \qquad (7.27)$$

and

$$ADDN = \begin{pmatrix} 0 \\ \\ 0 \\ 0.5 \end{pmatrix} \qquad (7.28)$$

Using (7.26) and (7.23)

$$\overline{TG} = G5.T + G6.TG(NB + 1) \qquad (7.29)$$

$$\text{where } G5 = G4.G2 \qquad (7.30)$$

$$\text{and } G6 = G4.G3 + ADDN$$

The matrices G1, G2, G3, G4, G5, and G6 are constructed in OTBOIL from the input data AG5, AG6.

4. Observation Matrix

The observation is taken to be steam temperature only, that is TS(NB + 1). This quantity is one element of the steam temperature vector given by equation (7.9). It is a simple matter then to define the observation vector H in the equation

$$y_i = Hx_i + y_i \tag{2.2}$$

as the bottom row of the matrix S2 and this operation is included in subroutine OTBOIL (Listing 7.1).

5. Data for Model

The following typical data was used, representing a power station once-through boiler unit at full load. Averages were taken so that similar data could be used for each region of the model. Twelve regions (NB = 12) were used, this being typical of the minimum number that could be used for this type of finite difference model.

Mass per unit length of tube:	MW	=	$5934.0\ kgm^{-1}$
Gas flow rate:	WG	=	$315.6\ kg.sec^{-1}$
Steam flow rate:	WS	=	$39.86\ kg.sec^{-1}$
Gas specific heat:	SHG	=	$1.16 \times 10^3\ J.kg^{-1\,o}C^{-1}$
Metal specific heat	SHM	=	$670.0\ J.kg^{-1\,o}C^{-1}$
Length of tube	L	=	7.92 m
Gas heat transfer coefficient	KG	=	$0.1907 \times 10^6\ J^oC^{-1}sec^{-1}$
Steam heat transfer coefficient	K	=	$0.5616 \times 10^6\ J^oC^{-1}sec^{-1}$
Steam specific heat	SHS	=	$13.53 \times 10^3\ Jkg^{-1\,o}C^{-1}$

The thermal capacity of each metal region is then:

$$C = \frac{MW.SHM.L}{NB} \quad A = 2.623 \times 10^6 \quad J^oC^{-1}$$

From the above, using equations (7.4), (7.5), (7.19), (7.20), and (7.33) the following input data for OTBOIL can be obtained

AS5 = 1.46	AG5 = 2.42	AM1 = 0.0727
AS6 = 0.46	AG6 = 1.42	AM2 = 0.2141

6. Construction of System Matrices from Data

The subroutine OTBOIL was run with the above data being read in. The matrices generated were printed and are given in Figures (7.2) and (7.3).

In addition further subroutines were called which caused the matrices generated to be punched on to cards, in formats suitable for use of these cards as data input cards for MATLAN programs or by calling different subroutines, in formats suitable for use as data cards for an eigenvalue package. Details of these subroutines are given in Appendix 3.

It is worthwhile at this point to examine the elements of the system matrices A, B, and H, whose structure largely define the control problem.

(a) Matrix A

The matrix has a clear "banded" structure, all diagonal elements being approximately equal with a value of around 0.68, this corresponding to a time constant of the tube of around 6 seconds. Elements in a line parallel to the main diagonal are also approximately equal, the values becoming smaller away from the main diagonal. This would result from a mechanism whereby tube temperature is influenced more by nearby tube temperature than by remote tube temperatures.

(b) Matrix B

The importance of gas inlet temperature (the control variable) at the gas inlet end of the boiler is shown clearly by the elements of this matrix, their magnitude falling off rapidly from the gas inlet end (element 12).

(c) Matrix H

This matrix demonstrates the dependence of steam outlet temperature on the tube metal temperature at the steam outlet end, elements remote from this end falling in magnitude.

7. Step Response of the System

To give a visual picture of the behaviour of the plant the step response of the system was obtained. The disturbance applied was a 10 degree C step on gas inlet temperature. The response of steam outlet temperature was examined in the following ways:

(a) From System Matrices

The matrices ACON and BCON obtained in section 2 can be used via eigenvalues and the transition matrix, of the form

$$e^{(ACON)t}$$

to give the step response. The result is shown in Figure 7.4. As the transition matrix for 2 seconds was found to be identical with the matrix A as derived in section 3 it follows that the discrete matrix A, if used to generate a step response would give an identical result.

The same program gives the dynamic behaviour of the metal temperatures and these are also shown.

(b) Inverse Laplace Transform

As a check on the matrix modelling an entirely separate method of obtaining the step response was used as described below. Very close agreement with (a) was found.

It can be seen that the response of the steam temperature is in the form of a delayed response with a time constant in the region of 20 seconds. Metal temperatures remote from the gas inlet respond with a longer time constant.

8. Inverse Laplace Transform Method for Step Response

If the frequency domain transfer function can be obtained for a system then numerical methods of Laplace Inversion are available which allow the time response to be obtained. Such a method (reference 37) was used for the present plant model, the transfer function being obtained from the partial differential equations using the method below, which follows a method which has been given in reference 38.

The model equations may be re-cast into continuous space partial differential form, where ΔL is the length of the region used for the discrete space model.

$$C\frac{\partial T}{\partial t} = KG(TG - T) - K(T - TS) \qquad (7.41)$$

$$WG.SHG\frac{\partial TG}{\partial x} = \frac{KG}{\Delta L}(TG - T) \qquad (7.42)$$

$$WS.SHS\frac{\partial TS}{\partial x} = \frac{K}{\Delta L}(T - TS) \qquad (7.43)$$

Putting $s = \frac{\partial}{\partial t}$ in equation (7.41) and using (7.42) and (7.43) gives

$$Cs\ T = KG(TG - T) - K(T - TS)$$

$$\text{and } T = \frac{KG.TG + K.TS}{Cs + KG + K} \qquad (7.44)$$

Substituting into equation (7.41) gives

$$\frac{\partial TG}{\partial x} - AG(s)\ (Cs\ TG + K.TG - K.TS) = 0 \qquad (7.45)$$

$$\text{where } AG(s) = \frac{KG}{WG.SHG.\ \Delta L\ (Cs + KG + K)} \qquad (7.46)$$

Similarly substituting (7.44) into (7.43) gives

$$\frac{\partial TS}{\partial x} - AS(s)\ (-CsTS + KG.TG - KG.TS) = 0 \qquad (7.47)$$

$$\text{where } AS(s) = \frac{K}{WS.SHS.\ \Delta L(Cs + KG + K)} \qquad (7.48)$$

A solution to equations (7.45) and (7.47) can be obtained by seeking a solution

$$TG = BG(s)e^{\emptyset x} \text{ and } TS = BS(s)e^{\emptyset x} \qquad (7.49)$$

Substituting into (7.45) and (7.47) gives

$$\emptyset.BG(s) - AG(s)\left[(Cs + K)BG(s) - K.BS(s)\right] = 0 \qquad (7.50)$$

$$\emptyset.BS(s) - AS(s)\left[-(Cs + KG)BS(s) + KG.BG(s)\right] = 0 \qquad (7.51)$$

From (7.50)

$$BS(s) = \frac{1}{K}\left[-\frac{\emptyset BG(s)}{AG(s)} + (Cs + K)BG(s)\right] \qquad (7.52)$$

Substituting into (7.51) implies

$$\phi^2 + \left[-AG(s)(Cs+K) + AS(s)(Cs+KG)\right]\phi - AG(s)(Cs+K)(Cs+KG)AS(s) + KAS(s)AG(s)K \\ = 0 \qquad (7.53)$$

If the roots of this equation are ϕ_1 and ϕ_2 then the general solution to equations (7.45) and (7.47) is

$$\left.\begin{aligned} TG &= G1\, e^{\phi_1 x} + G2\, e^{\phi_2 x} \\ TS &= S1\, e^{\phi_1 x} + S2\, e^{\phi_2 x} \end{aligned}\right. \qquad (7.54)$$

Boundary conditions are that

$TS = 0$ at $x = 0$

$TG = BG(s)$ at $x = L$

It is required to find the steam temperature TS at $x = L$, as follows:

Substituting (7.54) into equation (7.47) and noting that the coefficients of $e^{\phi_1 x}$ and $e^{\phi_2 x}$ must be zero gives

$$\left.\begin{aligned} S1(\phi_1 + AS(Cs + KG)) - AS.KG.G1 &= 0 \\ S2(\phi_2 + AS(Cs + KG)) - AS.KG.G2 &= 0 \end{aligned}\right\} \qquad (7.55)$$

The boundary condition at $x = 0$ implies

$$S1 + S2 = 0 \qquad (7.56)$$

and that at $x = L$ gives

$$G1e^{\phi_1 L} + G2e^{\phi_2 L} = BG(s) \qquad (7.57)$$

Using (7.56) and (7.57) in (7.55) gives

$$S1 = \frac{AS.KG.BG(s)}{e^{\phi_1 L}(\phi_1 + AS(Cs + KG)) - e^{\phi_2 L}(\phi_2 + AS(Cs + KG))} \qquad (7.58)$$

The solution for the steam outlet temperature is therefore

$$\begin{aligned} TS &= S1\, e^{\phi_1 L} + S2\, e^{\phi_2 L} \\ &= AS.KG \left(\frac{e^{\phi_1 L} - e^{\phi_2 L}}{e^{\phi_1 L}(\phi_1 + AS(Cs+KG)) - e^{\phi_2 L}(\phi_2 + AS(Cs+KG))} \right) BG(s) \end{aligned} \qquad (7.59)$$

If the gas inlet temperature is disturbed as a step function then the response of the steam outlet temperature is given by setting in (7.59)

$$BG(s) = \frac{1}{s}$$

and taking the Inverse Laplace Transform.

A program for obtaining this transform was available from reference 37 and this is given in listing (7.3) together with the programming of equation (7.59), which forms subroutine FS2.

The time response obtained is shown in Figure 7.4 where it can be seen to agree with the time response obtained by matrix methods.

9. Eigenvalue Properties of the Uncontrolled System

To give some insight into the form of the dynamic system represented by the model the eigenvalues of the system matrices were found

(i) Continuous system matrix ACON

The eigenvalues are given in Figure 7.5. It can be seen that the uncontrolled system is stable since the eigenvalues have negative real parts.

(ii) Discrete system matrix A

The eigenvalues of the system are given in Figure 7.6. The discrete system is also stable since all eigenvalues lie within the unit circle.

The eigenvalue/eigenvector method used was from reference 39 .

10. Disturbance Matrices

Some assumed disturbance pattern is required for the model.

(i) Observation noise

Unity disturbance covariance matrix was arbitrarily used.

(ii) Plant noise

The unit matrix was used for the plant disturbance matrix.

11. Cost Matrices

Since the object of control has been taken to be the maintaining of a uniform steam temperature, and a cost of unity per time step may arbitrarily be put upon it. In terms of the plant state vector

$$\text{Cost} = \left[TS(NB + 1)\right]^2 = (H.T)'(H.T)$$
$$= T'H'HT$$

The cost matrix for the plant may therefore be obtained from

$$V = H'H \qquad (7.60)$$

and the matrix so obtained is shown in Figure 7.7.

The cost of control is a scalar quantity and initially was taken as unity, the reasoning being that then both gas inlet temperature (the control vector) and steam outlet temperature would have equal weightings in terms of cost.

Thus $P = 1$ (7.61)

The disturbance and cost matrices are used in the control studies of the following chapters.

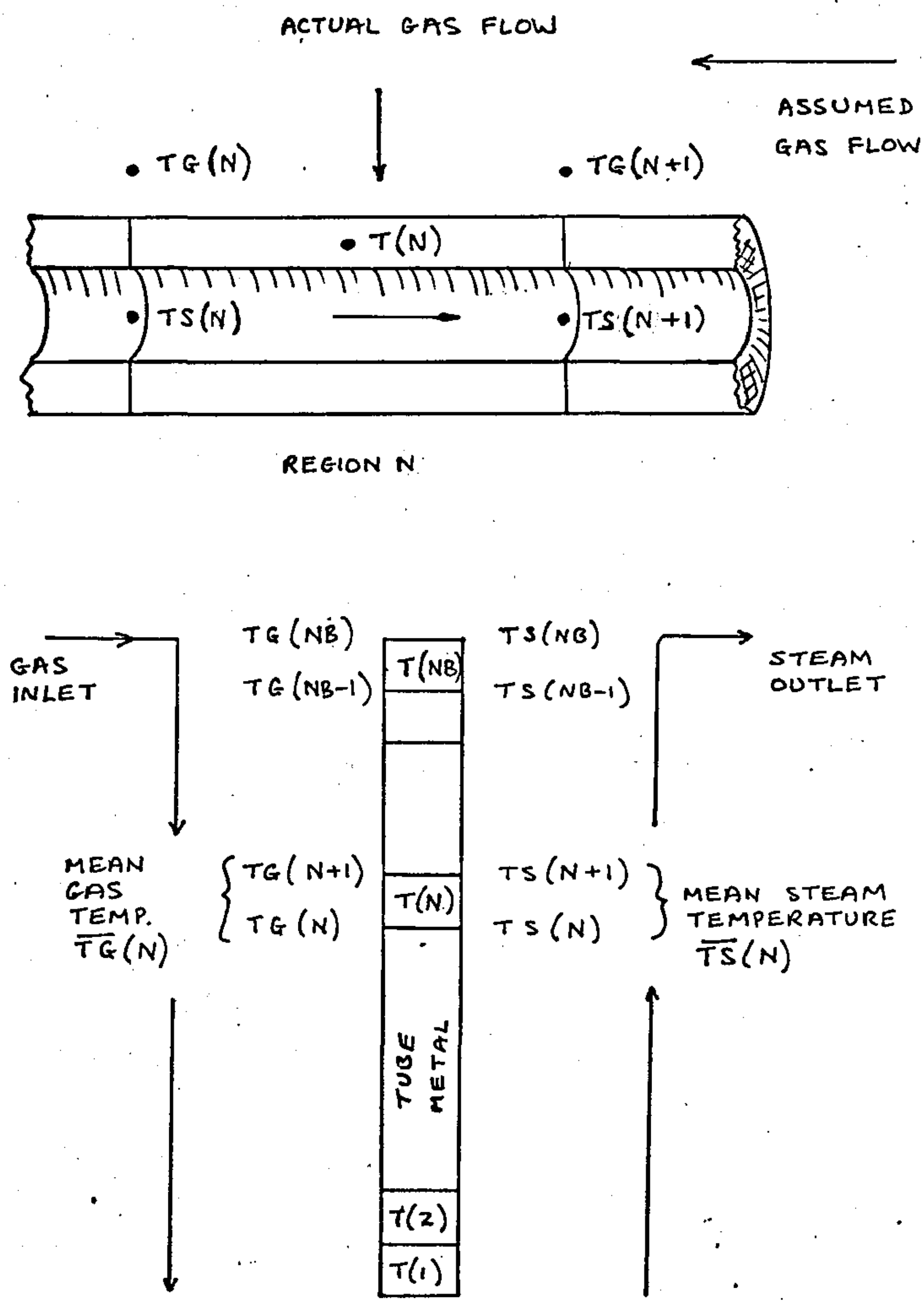


FIGURE 7.1. ONCE-THROUGH BOILER MODEL

ACON DIMENSIONS = (12,12)

	1	2	3	4	5	6
1	-1.9846E-01	2.3837E-02	1.3987E-02	8.2074E-03	4.8159E-03	2.8259E-03
2	9.6417E-02	-1.9846E-01	2.3837E-02	1.3987E-02	8.2074E-03	4.8159E-03
3	3.0378E-02	9.6417E-02	-1.9846E-01	2.3837E-02	1.3987E-02	8.2074E-03
4	9.5712E-03	3.0378E-02	9.6417E-02	-1.9846E-01	2.3837E-02	1.3987E-02
5	3.0156E-03	9.5712E-03	3.0378E-02	9.6417E-02	-1.9846E-01	2.3837E-02
6	9.5011E-04	3.0156E-03	9.5712E-03	3.0378E-02	9.6417E-02	-1.9846E-01
7	2.9935E-04	9.5011E-04	3.0156E-03	9.5712E-03	3.0378E-02	9.6417E-02
8	9.4316E-05	2.9935E-04	9.5011E-04	3.0156E-03	9.5712E-03	3.0378E-02
9	2.9716E-05	9.4316E-05	2.9935E-04	9.5011E-04	3.0156E-03	9.5712E-03
10	9.3626E-06	2.9716E-05	9.4316E-05	2.9935E-04	9.5011E-04	3.0156E-03
11	2.9499E-06	9.3626E-06	2.9716E-05	9.4316E-05	2.9935E-04	9.5011E-04
12	9.2941E-07	2.9499E-06	9.3626E-06	2.9716E-05	9.4316E-05	2.9935E-04

	7	8	9	10	11	12
1	1.6581E-03	9.7296E-04	5.7091E-04	3.3500E-04	1.9657E-04	1.1534E-04
2	2.8259E-03	1.6581E-03	9.7296E-04	5.7091E-04	3.3500E-04	1.9657E-04
3	4.8159E-03	2.8259E-03	1.6581E-03	9.7296E-04	5.7091E-04	3.3500E-04
4	8.2074E-03	4.8159E-03	2.8259E-03	1.6581E-03	9.7296E-04	5.7091E-04
5	1.3987E-02	8.2074E-03	4.8159E-03	2.8259E-03	1.6581E-03	9.7296E-04
6	2.3837E-02	1.3987E-02	8.2074E-03	4.8159E-03	2.8259E-03	1.6581E-03
7	-1.9846E-01	2.3837E-02	1.3987E-02	8.2074E-03	4.8159E-03	2.8259E-03
8	9.6417E-02	-1.9846E-01	2.3837E-02	1.3987E-02	8.2074E-03	4.8159E-03
9	3.0378E-02	9.6417E-02	-1.9846E-01	2.3837E-02	1.3987E-02	8.2074E-03
10	9.5712E-03	3.0378E-02	9.6417E-02	-1.9846E-01	2.3837E-02	1.3987E-02
11	3.0156E-03	9.5712E-03	3.0378E-02	9.6417E-02	-1.9846E-01	2.3837E-02
12	9.5011E-04	3.0156E-03	9.5712E-03	3.0378E-02	9.6417E-02	-1.9846E-01

END OF MATRIX ACON

FIGURE 7.2.

BCON DIMENSIONS = (12, 1)

	1
1	1.6379E-04
2	2.7913E-04
3	4.7570E-04
4	8.1070E-04
5	1.3816E-03
6	2.3546E-03
7	4.0127E-03
8	6.8386E-03
9	1.1654E-02
10	1.9862E-02
11	3.3849E-02
12	5.7686E-02

END OF MATRIX BCON

H DIMENSIONS = (1,12)

	1	2	3	4
1	2.0502E-06	6.6023E-06	2.0955E-05	6.6510E-05

	5	6	7	8
	2.1110E-04	6.7000E-04	2.1265E-03	6.7494E-03

	9	10	11	12
	2.1422E-02	6.7992E-02	2.1580E-01	6.8493E-01

END OF MATRIX H

FIGURE 7.2, CTD. MATRICES BCON AND H AS OUTPUT BY THE MATLAN SUBROUTINE OTBOIL

A DIMENSIONS = (12.12)

	1	2	3	4	5	6
1	6.7642E-01	3.4536E-02	2.1107E-02	1.2892E-02	7.8705E-03	4.8025E-03
2	1.3144E-01	6.7969E-01	3.6508E-02	2.2295E-02	1.3609E-02	8.3022E-03
3	5.4110E-02	1.3268E-01	6.8043E-01	3.6957E-02	2.2566E-02	1.3771E-02
4	2.1866E-02	5.4576E-02	1.3296E-01	6.8060E-01	3.7058E-02	2.2626E-02
5	8.7042E-03	2.2040E-02	5.4681E-02	1.3302E-01	6.8064E-01	3.7081E-02
6	3.4217E-03	8.7689E-03	2.2079E-02	5.4704E-02	1.3303E-01	6.8064E-01
7	1.3308E-03	3.4456E-03	8.7832E-03	2.2088E-02	5.4709E-02	1.3303E-01
8	5.1289E-04	1.3396E-03	3.4508E-03	8.7862E-03	2.2089E-02	5.4709E-02
9	1.9608E-04	5.1609E-04	1.3415E-03	3.4517E-03	8.7862E-03	2.2088E-02
10	7.4417E-05	1.9720E-04	5.1664E-04	1.3415E-03	3.4508E-03	8.7832E-03
11	2.8039E-05	7.4753E-05	1.9720E-04	5.1609E-04	1.3396E-03	3.4456E-03
12	1.0462E-05	2.8039E-05	7.4417E-05	1.9608E-04	5.1289E-04	1.3308E-03

	7	8	9	10	11	12
1	2.9289E-03	1.7853E-03	1.0872E-03	6.6054E-04	3.9771E-04	2.3021E-04
2	5.0625E-03	3.0853E-03	1.8786E-03	1.1412E-03	6.8705E-04	3.9771E-04
3	8.4000E-03	5.1208E-03	3.1190E-03	1.8952E-03	1.1412E-03	6.6054E-04
4	1.3808E-02	8.4210E-03	5.1311E-03	3.1190E-03	1.8786E-03	1.0872E-03
5	2.2639E-02	1.3814E-02	8.4210E-03	5.1208E-03	3.0853E-03	1.7853E-03
6	3.7085E-02	2.2639E-02	1.3808E-02	8.4000E-03	5.0625E-03	2.9289E-03
7	6.8064E-01	3.7081E-02	2.2626E-02	1.3771E-02	8.3022E-03	4.8025E-03
8	1.3303E-01	6.8064E-01	3.7058E-02	2.2566E-02	1.3609E-02	7.8705E-03
9	5.4704E-02	1.3302E-01	6.8060E-01	3.6957E-02	2.2295E-02	1.2892E-02
10	2.2079E-02	5.4681E-02	1.3296E-01	6.8043E-01	3.6508E-02	2.1107E-02
11	8.7689E-03	2.2040E-02	5.4576E-02	1.3268E-01	6.7969E-01	3.4536E-02
12	3.4217E-03	8.7042E-03	2.1866E-02	5.4110E-02	1.3144E-01	6.7642E-01

END OF MATRIX A

B

	1
1	4.0524E-04
2	6.9953E-04
3	1.1636E-03
4	1.9193E-03
5	3.1591E-03
6	5.1958E-03
7	8.5413E-03
8	1.4035E-02
9	2.3054E-02
10	3.7853E-02
11	6.2123E-02
12	1.0191E-01

END OF MATRIX B

FIGURE 7.3. DISCRETE TIME SYSTEM MATRICES A AND B

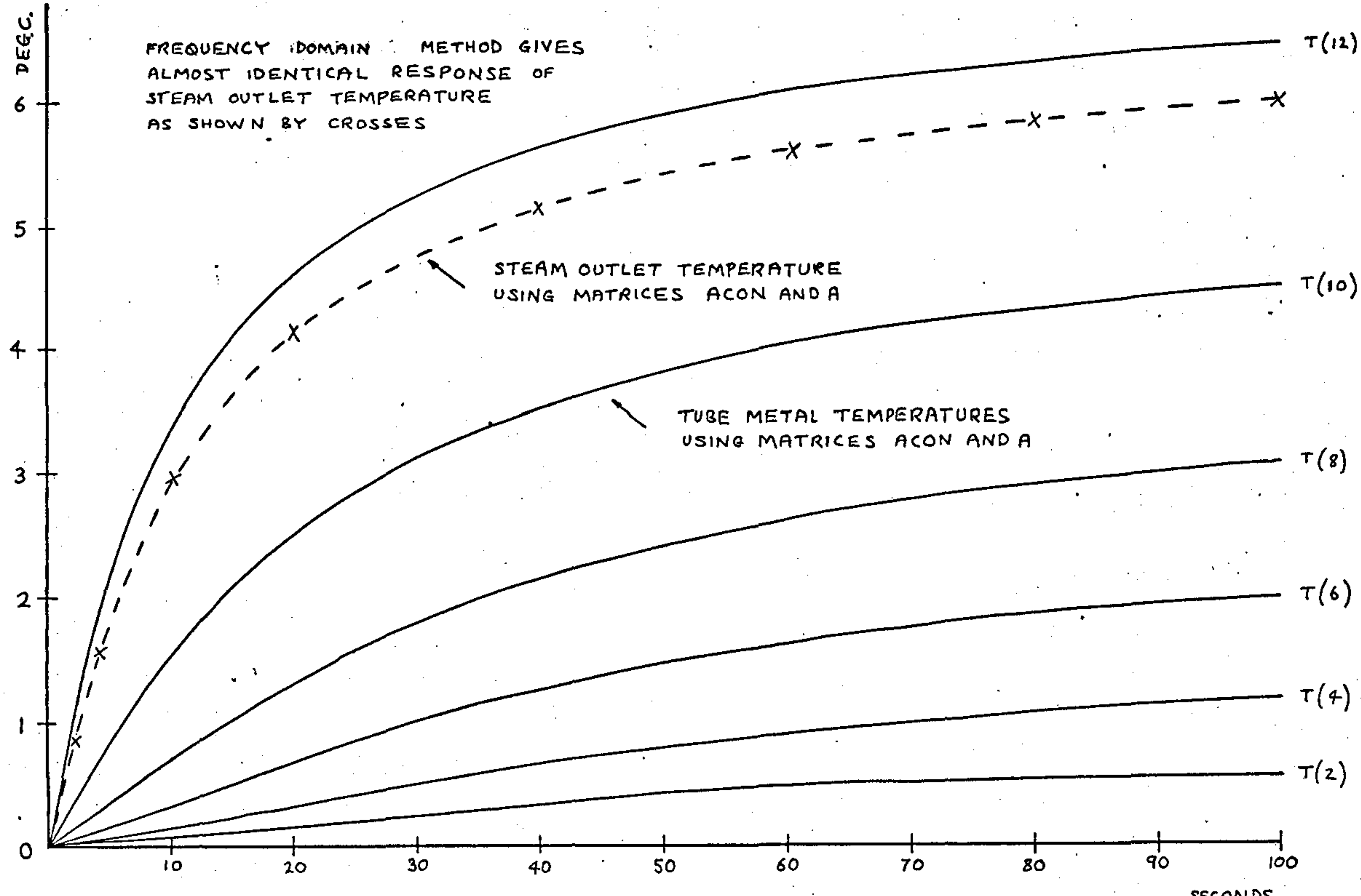
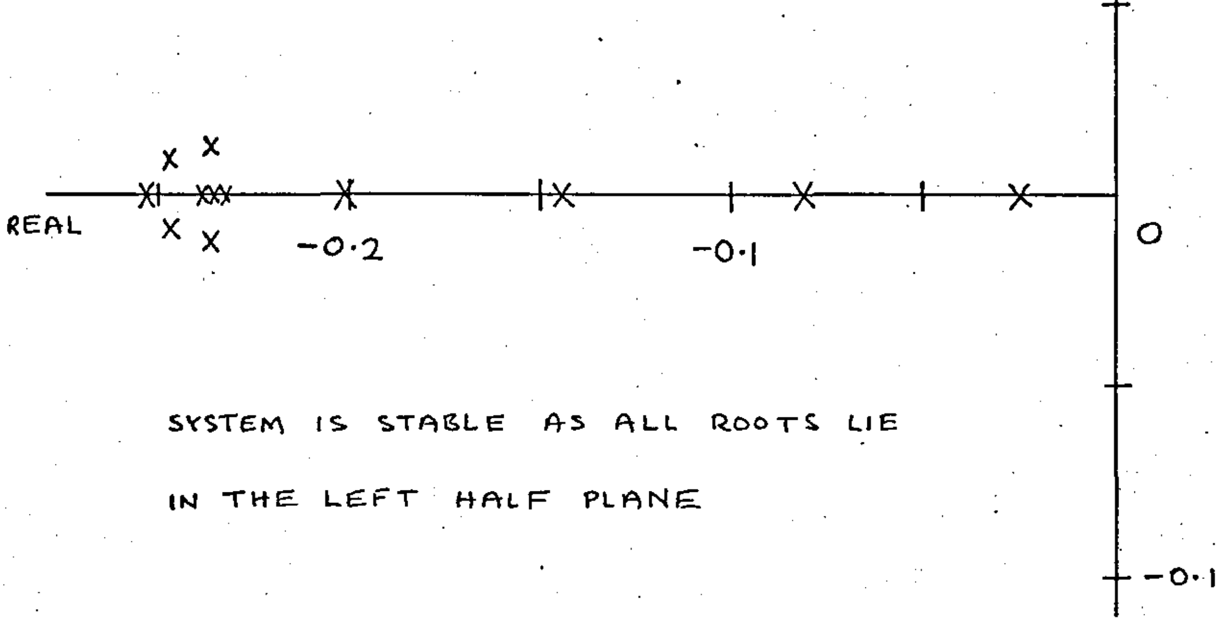


FIGURE 7.4 RESPONSE OF BOILER MODEL VARIABLES TO A 10 DEG.C. STEP INCREASE OF GAS INLET TEMPERATURE

FIGURE 7.5 EIGENVALUES OF CONTINUOUS MATRIX ACON



SYSTEM IS STABLE AS ALL ROOTS LIE IN THE LEFT HALF PLANE

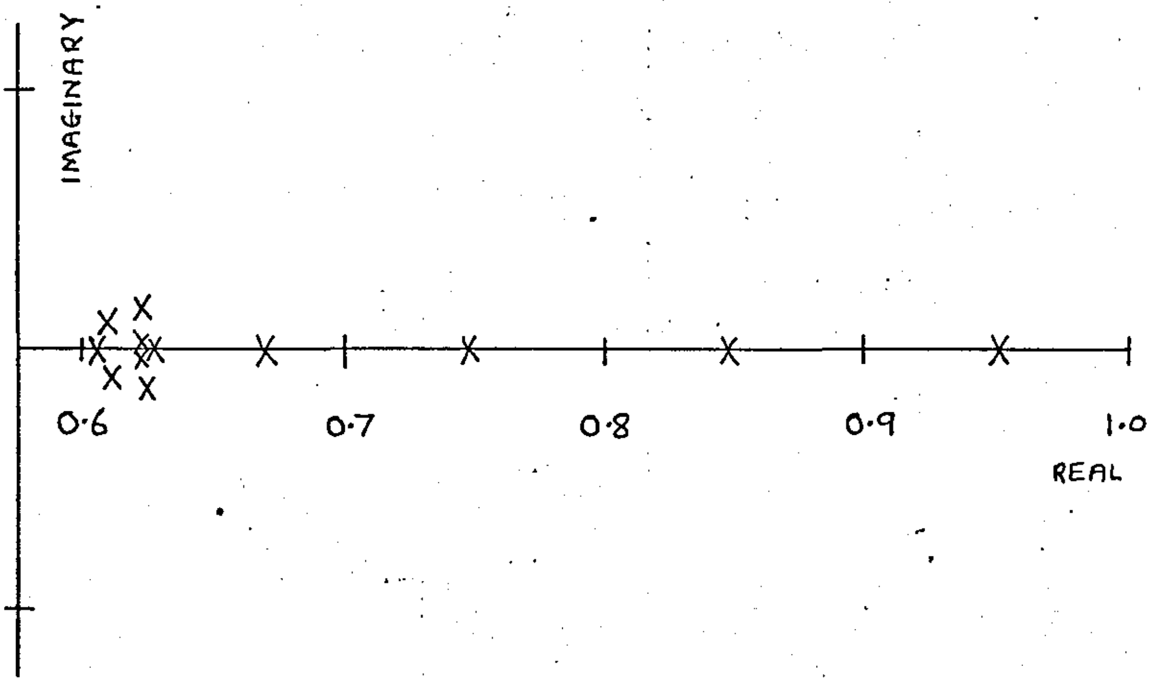


FIGURE 7.6 EIGENVALUES OF THE DISCRETE TIME SYSTEM MATRIX A

SYSTEM IS STABLE AS ALL ROOTS ARE WITHIN THE UNIT CIRCLE

V DIMENSIONS = (12,12)

	1	2	3	4	5	6
1	4.3272E-12	1.3734E-11	4.3591E-11	1.3835E-10	4.3912E-10	1.3937E-09
2	1.3734E-11	4.3591E-11	1.3835E-10	4.3912E-10	1.3937E-09	4.4236E-09
3	4.3591E-11	1.3835E-10	4.3912E-10	1.3937E-09	4.4236E-09	1.4040E-08
4	1.3835E-10	4.3912E-10	1.3937E-09	4.4236E-09	1.4040E-08	4.4562E-08
5	4.3912E-10	1.3937E-09	4.4236E-09	1.4040E-08	4.4562E-08	1.4144E-07
6	1.3937E-09	4.4236E-09	1.4040E-08	4.4562E-08	1.4144E-07	4.4890E-07
7	4.4236E-09	1.4040E-08	4.4562E-08	1.4144E-07	4.4890E-07	1.4248E-06
8	1.4040E-08	4.4562E-08	1.4144E-07	4.4890E-07	1.4248E-06	4.5221E-06
9	4.4562E-08	1.4144E-07	4.4890E-07	1.4248E-06	4.5221E-06	1.4353E-05
10	1.4144E-07	4.4890E-07	1.4248E-06	4.5221E-06	1.4353E-05	4.5555E-05
11	4.4890E-07	1.4248E-06	4.5221E-06	1.4353E-05	4.5555E-05	1.4459E-04
12	1.4248E-06	4.5221E-06	1.4353E-05	4.5555E-05	1.4459E-04	4.5891E-04

	7	8	9	10	11	12
1	4.4236E-09	1.4040E-08	4.4562E-08	1.4144E-07	4.4890E-07	1.4248E-06
2	1.4040E-08	4.4562E-08	1.4144E-07	4.4890E-07	1.4248E-06	4.5221E-06
3	4.4562E-08	1.4144E-07	4.4890E-07	1.4248E-06	4.5221E-06	1.4353E-05
4	1.4144E-07	4.4890E-07	1.4248E-06	4.5221E-06	1.4353E-05	4.5555E-05
5	4.4890E-07	1.4248E-06	4.5221E-06	1.4353E-05	4.5555E-05	1.4459E-04
6	1.4248E-06	4.5221E-06	1.4353E-05	4.5555E-05	1.4459E-04	4.5891E-04
7	4.5221E-06	1.4353E-05	4.5555E-05	1.4459E-04	4.5891E-04	1.4565E-03
8	1.4353E-05	4.5555E-05	1.4459E-04	4.5891E-04	1.4565E-03	4.6229E-03
9	4.5555E-05	1.4459E-04	4.5891E-04	1.4565E-03	4.6229E-03	1.4673E-02
10	1.4459E-04	4.5891E-04	1.4565E-03	4.6229E-03	1.4673E-02	4.6570E-02
11	4.5891E-04	1.4565E-03	4.6229E-03	1.4673E-02	4.6570E-02	1.4781E-01
12	1.4565E-03	4.6229E-03	1.4673E-02	4.6570E-02	1.4781E-01	4.6913E-01

END OF MATRIX V

FIGURE 7.7 STATE VECTOR COST MATRIX V FORMED FROM $H'.H$

```
          SUBPRO    OTBOIL,(AM1,AM2,AS5,AS6,AG5,AG6,NB,ACON,BCON,H)
          ADD       AM1,AM2,AM3
          FORMS     S1,(NB,NB),(1,1),(1,1),NB,AS5
          SUB       NB,1,NB1
          MULT      AS6,-1,MAS6
          INSUBM    MAS6,S1,(2,1),(1,1),NB1
          INV       S1,S2
          FORMS     S4,(NB,NB),(1,1),(1,1),NB,0.5
          INSUBM    0.5,S4,(2,1),(1,1),NB1
          MULT      S4,S2,S5
*  STEAM COMPLETE
          FORMS     G1,(NB,NB),(1,1),(1,1),NB,AG5
          MULT      AG6,-1,MAG6
          INSUBM    MAG6,G1,(1,2),(1,1),NB1
          INV       G1,G2
          EXSUBM    G2,(1,NB),(NB,1),G3
          MULT      AG6,G3,G3
          FORMS     G4,(NB,NB),(1,1),(1,1),NB,0.5
          INSUBM    0.5,G4,(1,2),(1,1),NB1
          MULT      G4,G2,G5
          MULT      G4,G3,G6
          FORMS     ADON,(NB,1),(NB,1),(0,0),1,0.5
          ADD       G6,ADDN,G6
*  GAS COMPLETE
          MULT      AM1,G5,ACON
          MULT      AM2,S5,DUM
          ADD       ACON,DUM,ACON
          SUB       ACON,AM3,ACON
*  CONTINUOUS A FORMED
          MULT      AM1,G6,BCON
*  CONTINUOUS B FORMED
          EXSUBM    S2,(NB,1),(1,NB),H
          RETURN
          END
```

Listing 7.1 Subroutine OTBOIL

This subroutine constructs the matrices describing the relation between gas, metal and water and generates the continuous system matrices ACON and BCON.

```
            SUBPRO    CRANK,(ACON,BCON,DELT,NSUB,A,B)
            ADD       1.0,DIVNS
            FOR       (NSUB,EQ,0,0.001),CONT
            LOOP      J2,J,1,NSUB
            MULT      DIVNS,2,DIVNS
J2         LOOPEND
CONT        DIV       DIVNS,DELT,DEL2
            DIV       2,DEL2,DEL3
            MULT      DEL3,ACON,ACO2
            SUB       1.0,ACO2,IMA
            ADD       1.0,ACO2,IPA
            INV       IMA,IMA
            MULT      IMA,IPA,A
            MULT      IMA,BCON,B
            MULT      B,DEL2,B
*  A,B FOR SUB TIME STEP
            FOR       (NSUB,EQ,0,0.001),EXIT
            LOOP      J1,J,1,NSUB
            MULT      A,B,AB
            ADD       AB,B,B
            MULT      A,A,A
J1          LOOPEND
EXIT        RETURN
            END
```

<u>Listing 7.2</u> <u>Subroutine CRANK</u>

The continuous system matrices ACON and BCON are converted to discrete forms A and B.

```
100 DIMENSION TIME(200),FT(200)
110 COMPLEX C(5),AL(5),S,F,CT,G
130 AL(1)=(1.283767708E1,1.666062584E0)
140 AL(2)=(1.222613148E1,5.012719264E0)
150 AL(3)=(1.093430343E1,8.409672996E0)
160 AL(4)=(8.776434640E0,1.192185390E1)
170 AL(5)=(5.225453367E0,1.572952905E1)
180 C(1)=(-3.690204687E4,1.969904635E5)
190 C(2)=(6.127699970E4,-9.540859887E4)
200 C(3)=(-2.891657227E4,1.816918510E4)
210 C(4)=(4.655360847E3,-1.901772633E0)
220 C(5)=(-1.187414019E2,-1.413036924E2)
230 WRITE(6,10)
240 10 FORMAT(12H+TS,TF,TSTEP)
250 READ(5,*)TS,TF,TSTEP
260 INO=1
270 T=TS
280 40 SUM1=0
290 SUM2=0
300 DO 20 I=1,5
310 S=AL(I)/T
400 CALLFS2(F,S)
470
500 SUM1=SUM1+REAL(C(I))*REAL(F)
510 20 SUM2=SUM2+AIMAG(C(I))*AIMAG(F)
520 FT(INO)=2*(SUM1-SUM2)/T
530 TIME(INO)=T
540 WRITE(6,*)T,FT(INO)
560 INO=INO+1
570 T=T+TSTEP
580 IF(TF-T+1.E-6)30,40,40
590 30 CONTINUE
610 END
620
```

```
620 SUBROUTINE FS2(F,S)
630 COMPLEX S,F,AG,AS,CS,CG,QB,QC,P1,P2,RT,E1,E2
700 AK=.5616E+06
710 AKG=0.1907E+06
720 WS=39.86
730 WG=315.6
740 SHS=13.53E03
750 SHG=1.16E+03
760 DL=7.92/12.
770 C=2.623E+06
780 NB=12
790
800 AG=AKG/(WG*SHG*DL*(C*S+AKG+AK))
810 AS=AK/(WS*SHS*DL*(C*S+AKG+AK))
820 CS=C*S+AK
830 CG=C*S+AKG
840 QB=-AG*CS+AS*CG
850 QC=AG*AS*(AK*AKG-CS*CG)
860 RT=CSQRT(QB*QB-4.*QC)
870 P1=(-QB+RT)/2.
880 P2=(-QB-RT)/2.
890 E1=CEXP(P1*DL*NP)
900 E2=CEXP(P2*DL*NP)
910 F=AS*AKG*(E1-E2)/(E1*(P1+AS*CG)-E2*(P2+AS*CG))
920 F=F*10./S
980 RETURN
990 END
```


Listing 7.3 Inverse Laplace Transform Program

Sub-routine FS2 forms the frequency domain transfer function according to equation (7.59) which is then inverted to give the step response.

CHAPTER 8

REDUCED ORDER CONTROL FOR THE ONCE THROUGH BOILER MODEL

1. Deterministic Control Law

In the previous chapter the plant state cost matrix was found to be

$$V = H'H$$

and is shown in Figure 7.7. The control vector (gas inlet temperature) cost was initially taken as

$$P = 1$$

Using these costs and the derived plant matrices A and B (Figure 7.3) the design method of Chapter 2 may be applied to find the optimal deterministic control law. This will be in the form

$$u_i = - \Lambda x_i \tag{8.1}$$

where the matrix Λ is obtained from subroutine CONTRL (Listing 2.1). The result of a run using the data for the once-through boiler model is shown in Figure 8.1.

As the control law was seen to have such low values (all elements less than 0.1), a further run was carried out to investigate sensitivity to the control cost by setting

$$P = 0.1$$

The resulting control law is also shown in Figure 8.1 and it can be seen that the control is now considerably more active. As this controller would be more likely to require a satisfactory estimator, it was used subsequently in order to more fully test the reduction method.

Associated with the derivation of the control law is the cost matrix PI. Being a 12 x 12 matrix it was punched onto cards for use by subroutine ESTIM2. The instructions to call CONTRL and punch the cards, using the subroutines of Appendix 3, are shown in Listing 8.1. The cost matrix PI associated with the control cost P = 0.1 is shown in Figure 8.2. It is clear that the cost weighting is concentrated near the steam outlet end of the boiler (mesh 12).

In order to investigate the performance of the deterministic system with this controller the system equations were written as

$$x_{i+1} = Ax_i + Bu_i + \xi_i \qquad (2.1)$$

$$= (A - B\Lambda)x_i + \xi_i \qquad (8.2)$$

The matrix $(A - B\Lambda)$ was computed by subroutine CONTRL and this matrix is shown in Figure 8.3. Further the eigenvalues were obtained and these are plotted in Figure 8.4. It can be seen that they are well within the unit circle showing the system to be very stable. There is a considerable similarity to the uncontrolled system eigenvalue plot (Figure 7.6), although the eigenvalues have been shifted towards the origin, implying some improvement in stability.

Before passing to the construction of the Kalman Filter it is possible to note the convergence properties of the matrix Riccati equations solved by subroutine CONTRL. More rapid convergence occurs for mesh points near the steam outlet end of the boiler, but as many as 30 iterations are required near the gas outlet end for convergence to 3 significant figures, indicating relatively slow convergence properties. A more direct method of solving the Riccati equations, similar to those considered in the introduction (Chapter 1) would probably be useful if this calculation were to be required frequently.

2. Control using Kalman Filter

In order to be able to judge the performance of the reduced order estimators to be considered in this chapter it is desirable to first obtain the optimal control performance using the "Kalman Filter" as the plant estimator.

The Kalman Filter is defined when the covariance matrix Γ has been computed. The subroutines ESTIM (Listing 2.2) and the enhanced form ESTIM2 (Listing 6.1) were written to evaluate this matrix. Using the once-through boiler system matrices A, B, H, and the disturbance covariance matrices Q and R defined in the previous chapter as the unit matrix and unity respectively gives the value of Γ as shown in Figure 8.5 as output from ESTIM2. The Kalman Filter estimator (Equation 2.35) when combined with the control law given by the certainty equivalence principle as

$$u_i = -\Lambda\mu_i \tag{2.22}$$

gives

$$\mu_{i+1} = (I - \Gamma H' R^{-1} H)(A - B\Lambda)\mu_i + \Gamma H' R^{-1} y_{i+1} \tag{8.3}$$

These relations can put into the forms

$$u_i = Cz_i + Dy_i \tag{5.15}$$

$$z_{i+1} = Ez_i + Fy_i \tag{5.16}$$

by setting

$$\left.\begin{aligned} C &= -\Lambda(I - \Gamma H'R^{-1}H)(A - B\Lambda) \\ D &= -\Lambda\Gamma H'R^{-1} \\ E &= (I - \Gamma H'R^{-1}H)(A - B\Lambda) \\ F &= \Gamma H'R^{-1} \end{aligned}\right\} \tag{8.4}$$

and identifying the storage vector z_i as the conditional mean μ_{i-1}.

To evaluate these matrices the subroutine SYSOPT was written as shown in Listing 8.2. The simple example system of Chapter 2 was used to confirm the validity of this subroutine.

The properties of the combined plant and controller system, which is of order 24, can now be investigated using the subprogram SYSTEM as described in Chapter 5. The result of this is a cost per time step of 1.475. Also calculated is the 24 x 24 system matrix SYS whose eigenvalues determine the stability of the control configuration. The eigen values were obtained and are plotted in Figure 8.6. There is roughly now a double set of eigenvalues, due to the similarity of the estimator dynamics with the plant dynamics. A new point appears on the real axis at (0.4, 0) and this will be associated with the estimator. Again the eigenvalues are all within the unit circle indicating a stable system.

Subprogram SYSTEM also computes the covariances of the state vector and the storage vector. This 24 x 24 matrix is too large to reproduce but the diagonal elements are plotted in Figure 8.7 where it can be seen that the greatest variance of the state vector occurs near the centre of the boiler, due to

(i) Control action at the steam outlet end holding down the cumulative addition of disturbances at that end.

(ii) The control action will not be suitable for control of state variable variation further down the boiler and will indeed contribute to the variation near the centre, giving the maximum value here.

The variances of the elements of the mean μ (the storage vector) are larger near the steam outlet end, showing the increased activity here, due to updating of these estimates. On the other hand the diagonal elements of Γ, the covariance matrix of the state estimate which are also shown in Figure 8.7 reduce at this end as information from steam outlet temperature is available.

3. Reduced Order Control of the Once-Through Boiler

As the procedure for the design of a reduced order controller has been set out in Chapter 6, it is straightforward to apply the method

4\. for this much larger system. The required matrices will have been generated and punched on to cards. The cost matrix Θ was computed by ESTIM2 and is shown in Figure 8.8. Set out below are the various cases that have been analysed.For the present system the order of the reduced estimator may range from 11 to 1. Not all of these cases were treated, some of the higher order cases being omitted.

Case	Estimator Order	Overall System Order	Reduction ADIM
B	11	23	1
D	9	21	2
F	7	19	3
H	5	17	4
I	4	16	5
J	3	15	6
K	2	14	7
L	1	13	8

Each of the above cases were treated by applying the following processes:

1. <u>OPRED</u>. Successive calls of this subroutine from the main program shown in Listing 8.3 allows a reduced order estimator to be constructed. The order of the estimator is specified via the reduction dimension ADIM (Listing 5.1.).
2. <u>SYSTEM</u>. This set of subroutines provides an assessment of the performance of reduced order controller in terms of cost, and generates state and covariance matrices for the overall system.
3. <u>EIGENVALUES</u>. The state matrix generated by SYSTEM can be assessed for stability directly by the evaluation of its eigenvalues.

4\. <u>Reduced System Costs</u>

The results for each of the reduced cases can now be examined. The most obvious indicator of performance is the overall system cost, as calculated by SYSTEM. The results are shown in Figure 8.9 and it is clear that only when the estimator becomes of very low order is the cost at all increased, the one-dimensional estimator increasing the cost by 0.2% from the optimal (Kalman Filter) system.

For a more difficult control situation it could be expected that the cost curve would rise earlier. This procedure is particularly revealing in terms of control "difficulty" and could be used directly as a control system design technique.

5. Simultaneous Diagonalisation: Eigenvalue Pattern

A central procedure in the choice of a reduced order estimator was the simultaneous diagonalisation of two matrices and the examination of a set of eigenvalues of a positive definite matrix (Chapter 5).

If a large number of these eigenvalues are small then a successful reduction to a lower order estimator could be anticipated. Taking for example case B, the eigenvalues rapidly fall off in value, the second and third eigenvalues being 5% and 0.01% of the first respectively.

Case	Eigenvalues x 10^{-3}			
J	109.6	5.8	0.014	0.2×10^{-6}
K	109.5	5.8	0.005	
L	92.8	2.48		

6. Information Storage

Case L, for example, has a storage vector of dimension one. It is interesting to examine the manner in which the information that is to be stored is selected. As there is the observation vector to complement the stored information, and the observation vector is closely associated with the elements of the state vector at the steam outlet end of the boiler, it would not be surprising to find that the stored information relates to a section of the boiler towards the centre. This can be seen by considering the elements of the matrix F, which was defined in equation (4.2) as

$$\mu_i = F\begin{pmatrix} y_i \\ z_i \end{pmatrix}$$

Thus in the present case where F has dimension 12 x 2 the first column will give the weighting given to the observation y_i and the second column that given to the stored information z_i. A value for F is obtained by the subroutine OPRED and the two columns of F are plotted in Figure 8.10, where it can be seen that the weighting on z was as expected, with a peak towards the boiler centre.

Similar weighting curves can be plotted for the other cases. Figure 8.11 shows the three weighting curves for case K. Two of these are similar to case L, while the third is of different form with some negative values.

This clear visual picture of how the reduced model represents the original model is possible since a plant model has been used with a great deal of similarity between mesh points. Such smooth curves would not in general be encountered. However the technique would be equally valid, even though the reduced variables would in general be combinations of state variables, resulting in weighted averages of such variables as fluid pressure and flow rate.

7. System Stability

The system matrix SYS describes the dynamics of the plant and its associated reduced order controller. The eigenvalues of the matrix SYS for each of the cases B to L are plotted in Figures 8.12 to 8.19. There are two interesting points to emerge from these graphs:

1. A set of eigenvalues around the point (.6, 0) on the real axis remain almost unchanged for each of the cases.
2. For the higher order cases (B to H) there are eigenvalues very close to the unit circle, a situation which is undesirable

since a small change in plant or controller parameters could result in instability.

On the last point the closeness of the eigenvalues to the unit circle is surprising. This is perhaps associated with the redundancy in the estimator system, where such near instability is (mathematically) acceptable provided the behaviour does not contribute to the costs. From a system design viewpoint this behaviour is unacceptable however, and it can be seen that as the order of the estimator is reduced there remain no points very close the the unit circle. The implication here would appear to be that in reducing a system's order the lower order system is to be preferred provided costs are not increased.

That the higher order system's closeness to the unit circle did not result in extra system costs has already been indicated above where the cost did not increase before case L. There is therefore a range of acceptable cases I to K, or L where there is good stability as well as no significant increase in system cost.

8. Proportional Controller

The theory developed for the reduced order controller implicitly assumes that the estimator order is not reduced below unity, that is a storage vector of dimension one. However, simple proportional control has no storage vector, the control being of the form

$$u_i = Dy_i \tag{8.5}$$

To investigate to what extent the control of the once-through boiler would be further degraded by the use of proportional control the set of SYSTEM subroutines were used by setting in equations (5.15, 16)

$E = 0.1$ (arbitrary but within $\pm$ 1.0)

$F = 0.1$ (arbitrary)

and $C = 0.0$

The value of D, the proportional control gain was varied between -0.65 and -1.16. The resulting system costs per time step are shown in Figure 8.20, showing a minimum cost of around 1.53 for a proportional gain of D = -0.94.

This is clearly a further degradation in system performance over case L, and would be the type of cost increase that would be used to justify the use of a controller with some information storage.

Another method of obtaining the optimal proportional control law makes use of the reduced order estimator subroutines but with a minor modification. Suppose, in case L, the stored information z_i, (dimension 1), is in fact discarded, and set to zero. The estimate of plant state must then rely entirely on the observed variable y_i, and this is the proportional control situation.

Rejecting the stored information z_i, with covariance matrix P, increases the covariance matrix Γ, and since covariances are additive the new plant state covariance becomes

$$\Gamma_{prop} = \Gamma + FPF' \qquad (8.6)$$

Setting z to zero was modelled by putting

$$Pprop = 0.00001.P \qquad (8.7)$$

Running the subroutines with these modifications gave a proportional control law of

$$u_i = -1.04 y_i \qquad (8.8)$$

This point is shown on Figure 8.20 and it can be seen to be close to the optimum value.

In addition some insight is gained into the action of the proportional controller as can be seen from Figure 8.21 which shows how the state estimate is calculated based solely on the observation y_i. It is the multiplication of these weighting coefficients by the deterministic control law of Figure 8.1 that gives the simple proportional control law of equation (8.8).

A further case, the uncontrolled system, was examined by setting D = 0.0, and the cost per time step was found to be 1.95.

9. Conclusions

The reduction method applied to the once-through boiler model has provided low order estimator models whose performance is very acceptable. That a simpler model would adequately model the 12 region once through boiler model was clear from the step response shown in Figure 7.4. What has been demonstrated in this chapter is that the reduction technique is capable of making the decisions which effectively determine which part of the model is redundant and can be left out.

The relationship between this simpler model and the original can be seen in terms of weighting coefficient curves which allow a clear visual understanding of how the reduced model is able to represent the original model. It is possible to obtain such a curve even for the extreme case of simple proportional control. Clearly estimates of plant state from such very common simple controllers could have applications in the fields of process control and where plant parameters vary with time.

The positive eigenvalues obtained in the "simultaneous diagonalisation" process appear to provide a qualitative technique for the assessment of control "difficulty".

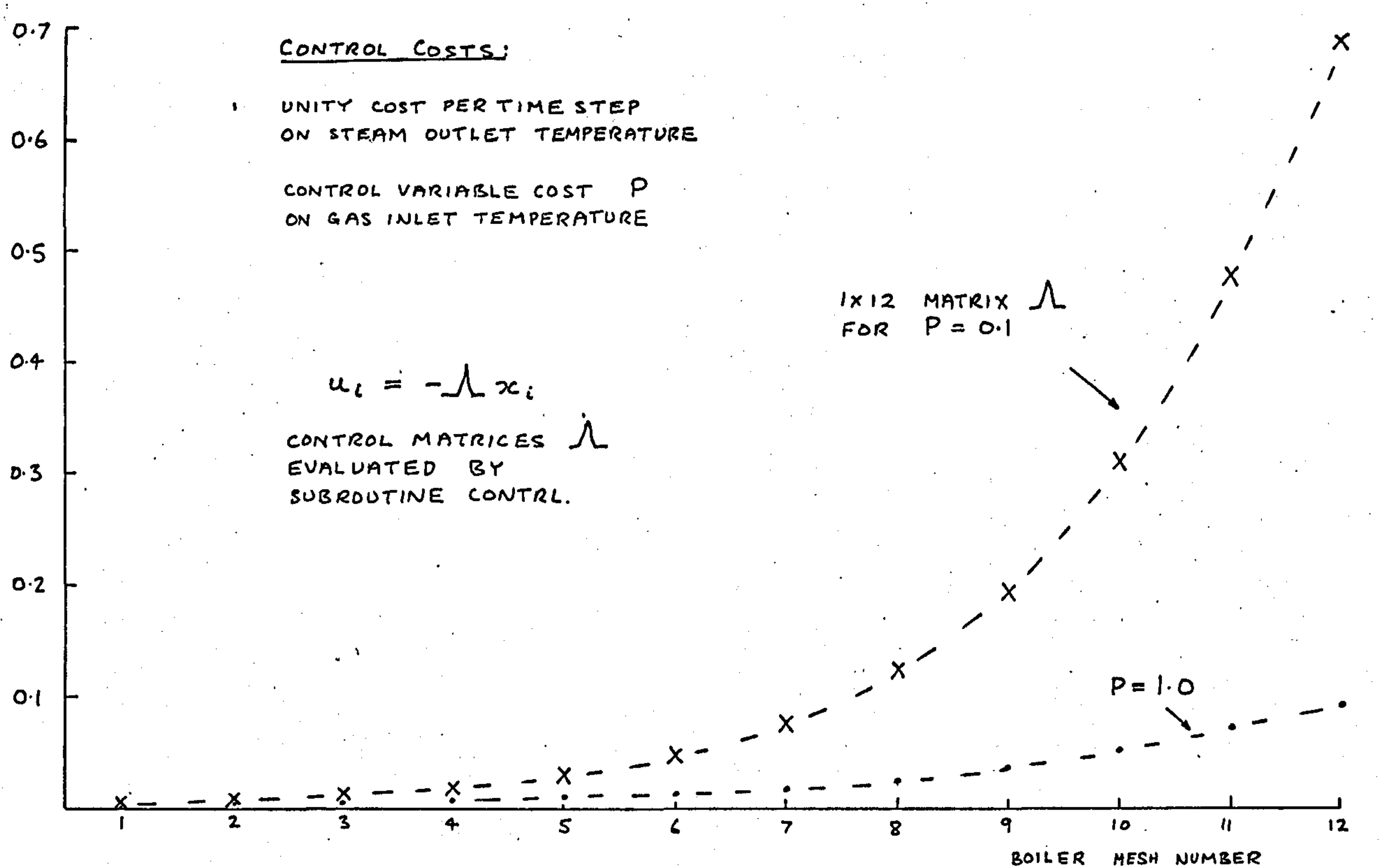


FIGURE 8.1. ELEMENTS OF THE DETERMINISTIC CONTROL LAW MATRIX

PI DIMENSIONS = (12,12)

	1	2	3	4	5	6
1	3.4393E-06	5.4247E-06	8.3063E-06	1.2688E-05	1.9564E-05	3.0577E-05
2	5.4247E-06	8.5560E-06	1.3101E-05	2.0012E-05	3.0857E-05	4.8228E-05
3	8.3063E-06	1.3101E-05	2.0060E-05	3.0642E-05	4.7248E-05	7.3846E-05
4	1.2688E-05	2.0012E-05	3.0642E-05	4.6807E-05	7.2172E-05	1.1280E-04
5	1.9564E-05	3.0857E-05	4.7248E-05	7.2172E-05	1.1128E-04	1.7393E-04
6	3.0577E-05	4.8228E-05	7.3846E-05	1.1280E-04	1.7393E-04	2.7184E-04
7	4.8427E-05	7.6381E-05	1.1696E-04	1.7865E-04	2.7546E-04	4.3054E-04
8	7.7432E-05	1.2213E-04	1.8700E-04	2.8565E-04	4.4045E-04	6.8840E-04
9	1.2412E-04	1.9577E-04	2.9977E-04	4.5790E-04	7.0603E-04	1.1035E-03
10	1.9716E-04	3.1097E-04	4.7616E-04	7.2734E-04	1.1215E-03	1.7529E-03
11	3.0409E-04	4.7963E-04	7.3441E-04	1.1218E-03	1.7297E-03	2.7035E-03
12	4.3716E-04	6.8950E-04	1.0558E-03	1.6127E-03	2.4866E-03	3.8865E-03

	7	8	9	10	11	12
1	4.8427E-05	7.7432E-05	1.2412E-04	1.9716E-04	3.0409E-04	4.3716E-04
2	7.6381E-05	1.2213E-04	1.9577E-04	3.1097E-04	4.7963E-04	6.8950E-04
3	1.1696E-04	1.8700E-04	2.9977E-04	4.7616E-04	7.3441E-04	1.0558E-03
4	1.7865E-04	2.8565E-04	4.5790E-04	7.2734E-04	1.1218E-03	1.6127E-03
5	2.7546E-04	4.4045E-04	7.0603E-04	1.1215E-03	1.7297E-03	2.4866E-03
6	4.3054E-04	6.8840E-04	1.1035E-03	1.7529E-03	2.7035E-03	3.8865E-03
7	6.8187E-04	1.0903E-03	1.7477E-03	2.7761E-03	4.2817E-03	6.1553E-03
8	1.0903E-03	1.7433E-03	2.7944E-03	4.4388E-03	6.8462E-03	9.8420E-03
9	1.7477E-03	2.7944E-03	4.4795E-03	7.1154E-03	1.0974E-02	1.5777E-02
10	2.7761E-03	4.4388E-03	7.1154E-03	1.1302E-02	1.7432E-02	2.5060E-02
11	4.2817E-03	6.8462E-03	1.0974E-02	1.7432E-02	2.6887E-02	3.8652E-02
12	6.1553E-03	9.8420E-03	1.5777E-02	2.5060E-02	3.8652E-02	5.5565E-02

END OF MATRIX PI

Figure 8.2 Cost Matrix PI computed by subroutine CONTRL.

ABL DIMENSIONS = (12,12)

	1	2	3	4	5	6
1	6.7642E-01	3.4533E-02	2.1101E-02	1.2884E-02	7.8580E-03	4.7829E-03
2	1.3143E-01	6.7968E-01	3.6499E-02	2.2281E-02	1.3587E-02	8.2686E-03
3	5.4104E-02	1.3267E-01	6.8042E-01	3.6934E-02	2.2530E-02	1.3715E-02
4	2.1856E-02	5.4559E-02	1.3293E-01	6.8056E-01	3.6999E-02	2.2534E-02
5	8.6871E-03	2.2013E-02	5.4639E-02	1.3295E-01	6.8054E-01	3.6929E-02
6	3.3935E-03	8.7245E-03	2.2011E-02	5.4600E-02	1.3287E-01	6.8039E-01
7	1.2846E-03	3.3726E-03	8.6714E-03	2.1917E-02	5.4445E-02	1.3262E-01
8	4.3684E-04	1.2197E-03	3.2671E-03	8.5057E-03	2.1657E-02	5.4033E-02
9	7.1160E-05	3.1906E-04	1.0398E-03	2.9909E-03	8.0757E-03	2.0977E-02
10	-1.3069E-04	-1.2629E-04	2.1309E-05	5.8484E-04	2.2841E-03	6.9598E-03
11	-3.0857E-04	-4.5616E-04	-6.1573E-04	-7.2568E-04	-5.7506E-04	4.5299E-04
12	-5.4172E-04	-8.4289E-04	-1.2592E-03	-1.8410E-03	-2.6281E-03	-3.5783E-03

	7	8	9	10	11	12
1	2.8980E-03	1.7358E-03	1.0080E-03	5.3466E-04	2.0357E-04	-4.8884E-05
2	5.0091E-03	2.9999E-03	1.7418E-03	9.2391E-04	3.5192E-04	-8.4062E-05
3	8.3113E-03	4.9789E-03	2.8914E-03	1.5338E-03	5.8375E-04	-1.4084E-04
4	1.3661E-02	8.1868E-03	4.7558E-03	2.5228E-03	9.5910E-04	-2.3466E-04
5	2.2398E-02	1.3429E-02	7.8032E-03	4.1396E-03	1.5718E-03	-3.9047E-04
6	3.6688E-02	2.2006E-02	1.2792E-02	6.7862E-03	2.5733E-03	-6.4947E-04
7	6.7999E-01	3.6039E-02	2.0956E-02	1.1118E-02	4.2103E-03	-1.0801E-03
8	1.3196E-01	6.7893E-01	3.4314E-02	1.8206E-02	6.8845E-03	-1.7959E-03
9	5.2945E-02	1.3021E-01	6.7609E-01	2.9796E-02	1.1250E-02	-2.9856E-03
10	1.9191E-02	5.0063E-02	1.2555E-01	6.6868E-01	1.8374E-02	-4.9628E-03
11	4.0293E-03	1.4462E-02	4.2428E-02	1.1336E-01	6.4992E-01	-8.2481E-03
12	-4.3533E-03	-3.7275E-03	1.9384E-03	2.2456E-02	8.2615E-02	6.0623E-01

END OF MATRIX ABL

Figure 8.3 System Matrix for deterministic control of once-through boiler

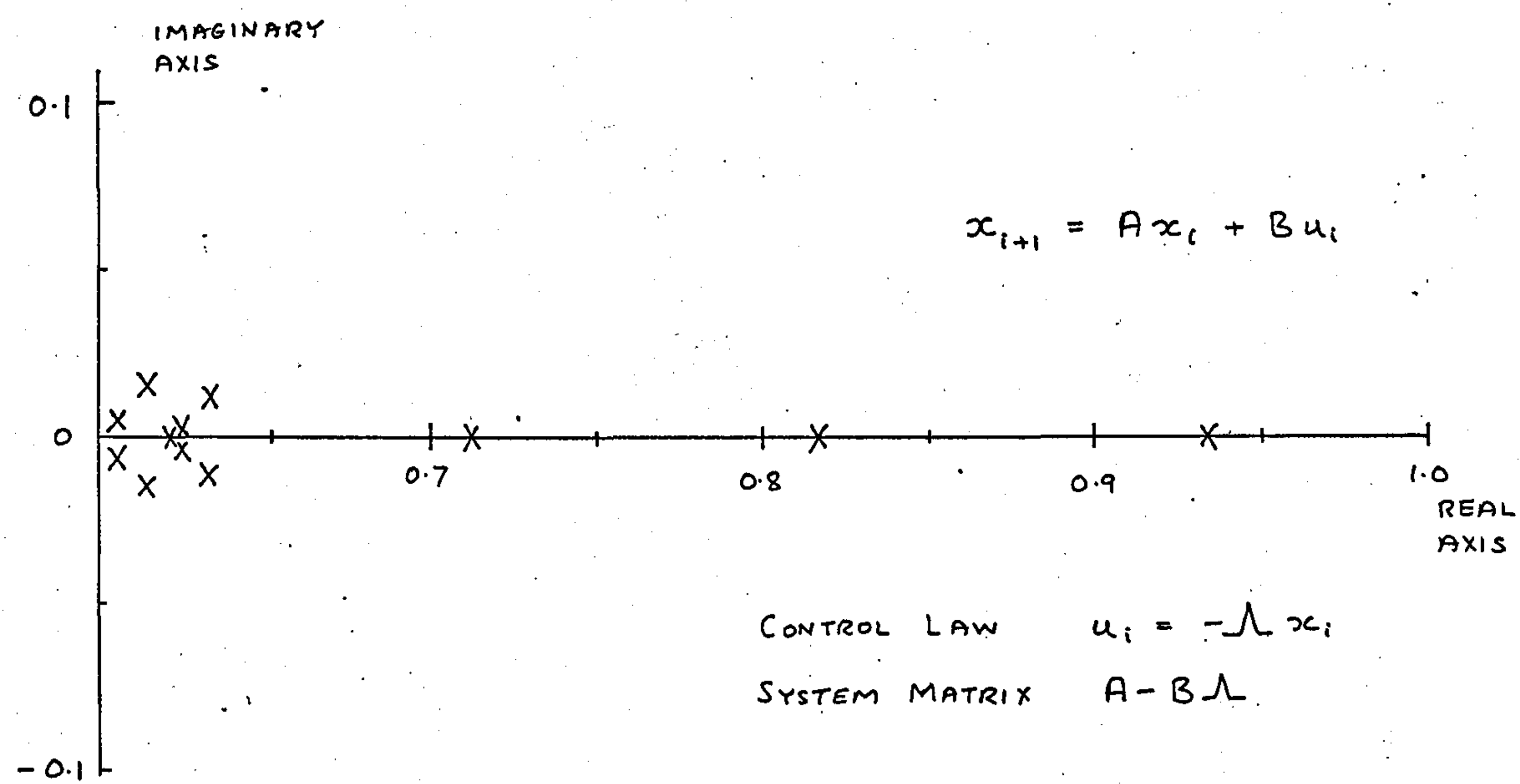


FIGURE 8.4 EIGENVALUES OF THE DETERMINISTIC SYSTEM

GNEXT DIMENSIONS = (12,12)

	1	2	3	4	5	6
1	1.9500E 00	5.2591E-01	4.3159E-01	3.6007E-01	3.0181E-01	2.5132E-01
2	5.2591E-01	2.2790E 00	8.0837E-01	6.7050E-01	5.5877E-01	4.6314E-01
3	4.3159E-01	8.0837E-01	2.5286E 00	1.0224E 00	8.4840E-01	7.0079E-01
4	3.6007E-01	6.7050E-01	1.0224E 00	2.7127E 00	1.1737E 00	9.6540E-01
5	3.0181E-01	5.5877E-01	8.4840E-01	1.1737E 00	2.8334E 00	1.2614E 00
6	2.5132E-01	4.6314E-01	7.0079E-01	9.6540E-01	1.2614E 00	2.8889E 00
7	2.0556E-01	3.7754E-01	5.6994E-01	7.8289E-01	1.0186E 00	1.2824E 00
8	1.6286E-01	2.9340E-01	4.4982E-01	6.1688E-01	8.0044E-01	1.0034E 00
9	1.2215E-01	2.2346E-01	3.3658E-01	4.6124E-01	5.9770E-01	7.4729E-01
10	8.2484E-02	1.5071E-01	2.2692E-01	3.1093E-01	4.0278E-01	5.0304E-01
11	4.2057E-02	7.6776E-02	1.1559E-01	1.5842E-01	2.0530E-01	2.5640E-01
12	-4.0593E-03	-7.4349E-03	-1.1185E-02	-1.5289E-02	-1.9754E-02	-2.4676E-02

	7	8	9	10	11	12
1	2.0556E-01	1.6286E-01	1.2215E-01	8.2484E-02	4.2057E-02	-4.0593E-03
2	3.7754E-01	2.9840E-01	2.2346E-01	1.5071E-01	7.6776E-02	-7.4349E-03
3	5.6994E-01	4.4982E-01	3.3658E-01	2.2692E-01	1.1559E-01	-1.1185E-02
4	7.8289E-01	6.1688E-01	4.6124E-01	3.1093E-01	1.5842E-01	-1.5289E-02
5	1.0186E 00	8.0044E-01	5.9770E-01	4.0278E-01	2.0530E-01	-1.9754E-02
6	1.2324E 00	1.0034E 00	7.4729E-01	5.0304E-01	2.5640E-01	-2.4676E-02
7	2.8750E 00	1.2316E 00	9.1333E-01	6.1328E-01	3.1222E-01	-3.0416E-02
8	1.2316E 00	2.7863E 00	1.1019E 00	7.3655E-01	3.7361E-01	-3.8144E-02
9	9.1333E-01	1.1019E 00	2.6130E 00	8.7742E-01	4.4114E-01	-5.1562E-02
10	6.1328E-01	7.3655E-01	8.7742E-01	2.3309E 00	5.1201E-01	-8.2470E-02
11	3.1222E-01	3.7361E-01	4.4114E-01	5.1201E-01	1.8593E 00	-1.6897E-01
12	-3.0416E-02	-3.8144E-02	-5.1562E-02	-8.2470E-02	-1.6897E-01	8.5678E-01

END OF MATRIX GNEXT

Figure 8.5 Covariance Matrix Γ for Optimal Estimator

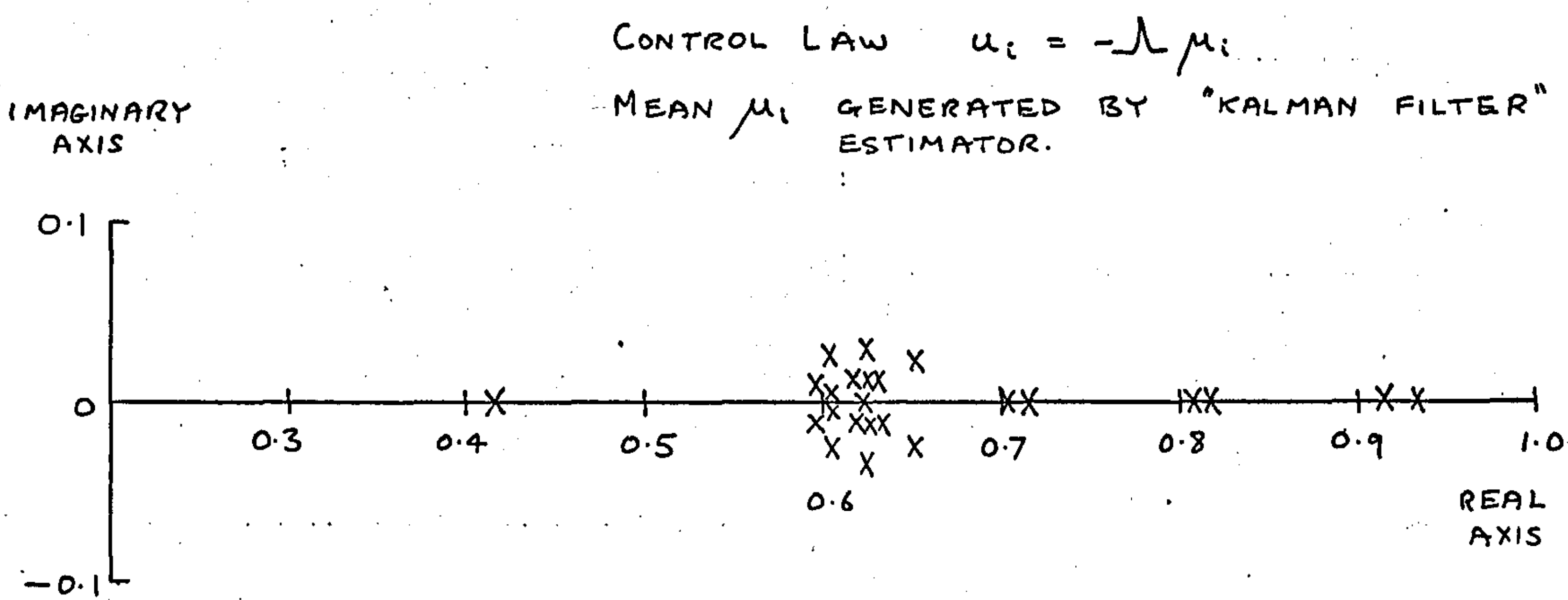


FIGURE 8.6. EIGENVALUES OF THE 24×24 "KALMAN FILTER" CONTROL SYSTEM.

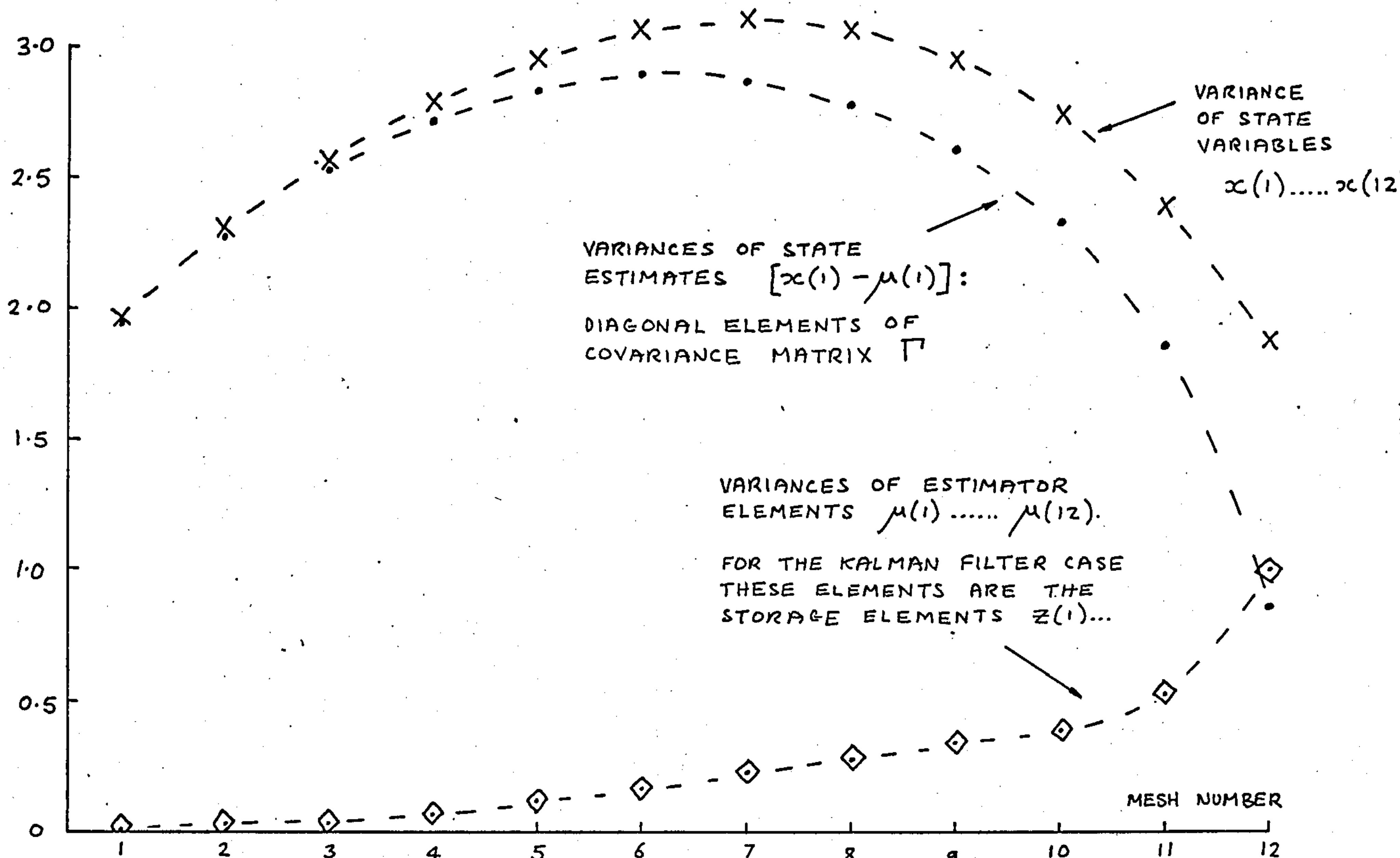


FIGURE 8.7. VARIANCES OF KALMAN FILTER SYTEM.

THETA DIMENSIONS = (12,12)

	1	2	3	4	5	6
1	1.6357E-03	2.1300E-03	2.5508E-03	2.8785E-03	3.0848E-03	3.1343E-03
2	2.1300E-03	2.7970E-03	3.3842E-03	3.8664E-03	4.2044E-03	4.3458E-03
3	2.5508E-03	3.3842E-03	4.1474E-03	4.8111E-03	5.3264E-03	5.6228E-03
4	2.8785E-03	3.8664E-03	4.8111E-03	5.6835E-03	6.4280E-03	6.9572E-03
5	3.0848E-03	4.2044E-03	5.3264E-03	6.4280E-03	7.4536E-03	8.3040E-03
6	3.1343E-03	4.3458E-03	5.6228E-03	6.9572E-03	8.3040E-03	9.5647E-03
7	2.9873E-03	4.2281E-03	5.6096E-03	7.1485E-03	8.8264E-03	1.0566E-02
8	2.6064E-03	3.7854E-03	5.1814E-03	6.8459E-03	8.8058E-03	1.1035E-02
9	1.9662E-03	2.9620E-03	4.2331E-03	5.8721E-03	7.9685E-03	1.0584E-02
10	1.0703E-03	1.7369E-03	2.6926E-03	4.0657E-03	6.0155E-03	8.7210E-03
11	-1.7466E-05	1.7262E-04	5.9011E-04	1.3743E-03	2.7337E-03	4.9647E-03
12	-1.1087E-03	-1.4808E-03	-1.7837E-03	-1.9197E-03	-1.6992E-03	-7.8200E-04

	7	8	9	10	11	12
1	2.9873E-03	2.6064E-03	1.9662E-03	1.0703E-03	-1.7466E-05	-1.1087E-03
2	4.2281E-03	3.7854E-03	2.9620E-03	1.7369E-03	1.7262E-04	-1.4808E-03
3	5.6096E-03	5.1814E-03	4.2331E-03	2.6926E-03	5.9011E-04	-1.7837E-03
4	7.1485E-03	6.8459E-03	5.8721E-03	4.0657E-03	1.3743E-03	-1.9197E-03
5	8.8264E-03	8.8058E-03	7.9685E-03	6.0155E-03	2.7337E-03	-1.6992E-03
6	1.0566E-02	1.1035E-02	1.0584E-02	8.7210E-03	4.9647E-03	-7.8200E-04
7	1.2197E-02	1.3409E-02	1.3700E-02	1.2342E-02	8.4557E-03	1.4028E-03
8	1.3409E-02	1.5629E-02	1.7123E-02	1.6928E-02	1.3646E-02	5.7441E-03
9	1.3700E-02	1.7123E-02	2.0327E-02	2.2216E-02	2.0867E-02	1.3480E-02
10	1.2342E-02	1.6928E-02	2.2216E-02	2.7280E-02	2.9935E-02	2.5978E-02
11	8.4557E-03	1.3646E-02	2.0867E-02	2.9935E-02	3.9213E-02	4.3743E-02
12	1.4028E-03	5.7441E-03	1.3480E-02	2.5978E-02	4.3743E-02	6.3397E-02

END OF MATRIX THETA

<u>Figure 8.8</u> Cost Matrix Θ computed by subroutine ESTIM2.

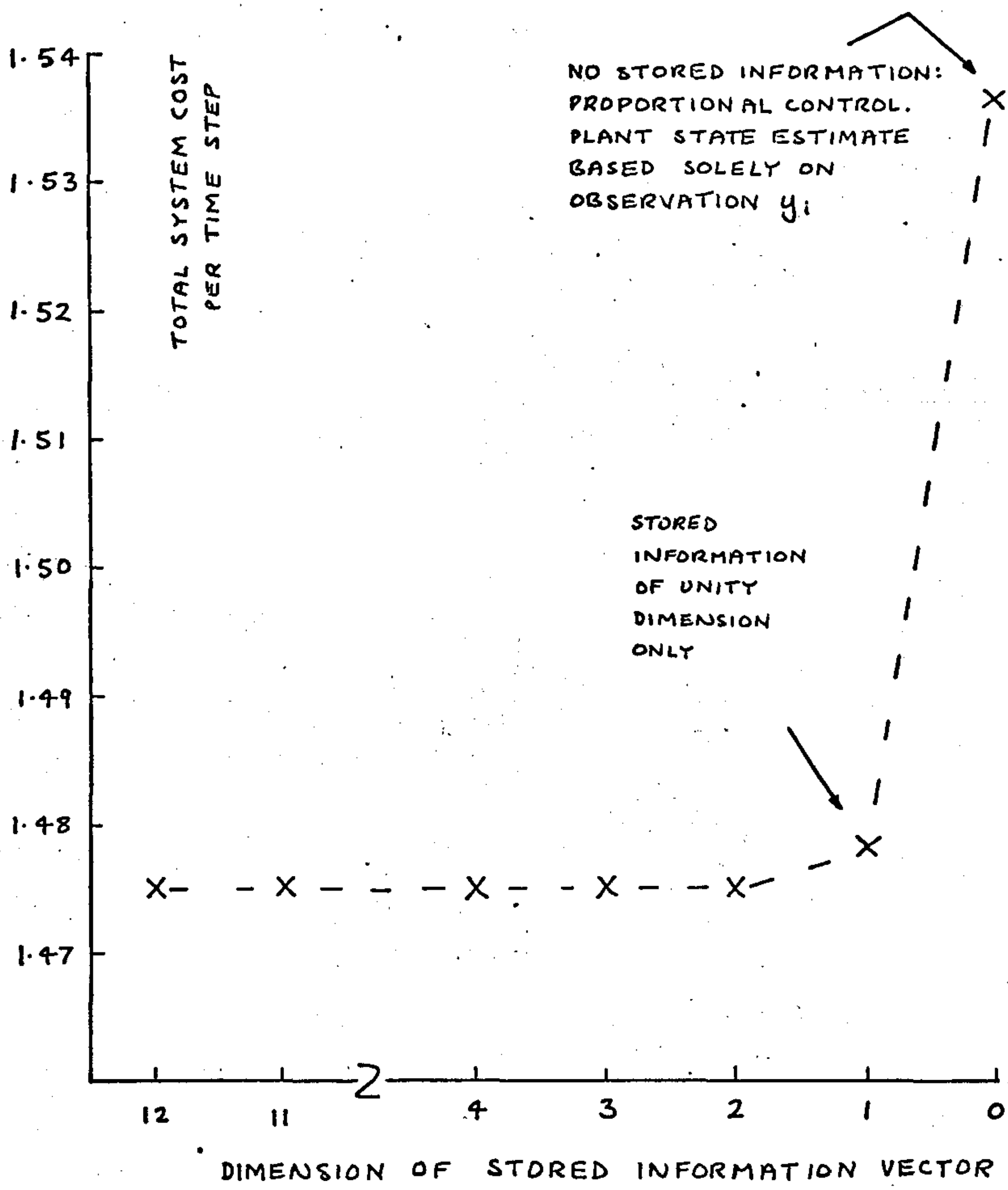


FIGURE 8.9 TOTAL CONTROL SYSTEM COSTS AS ESTIMATOR DIMENSION REDUCES.

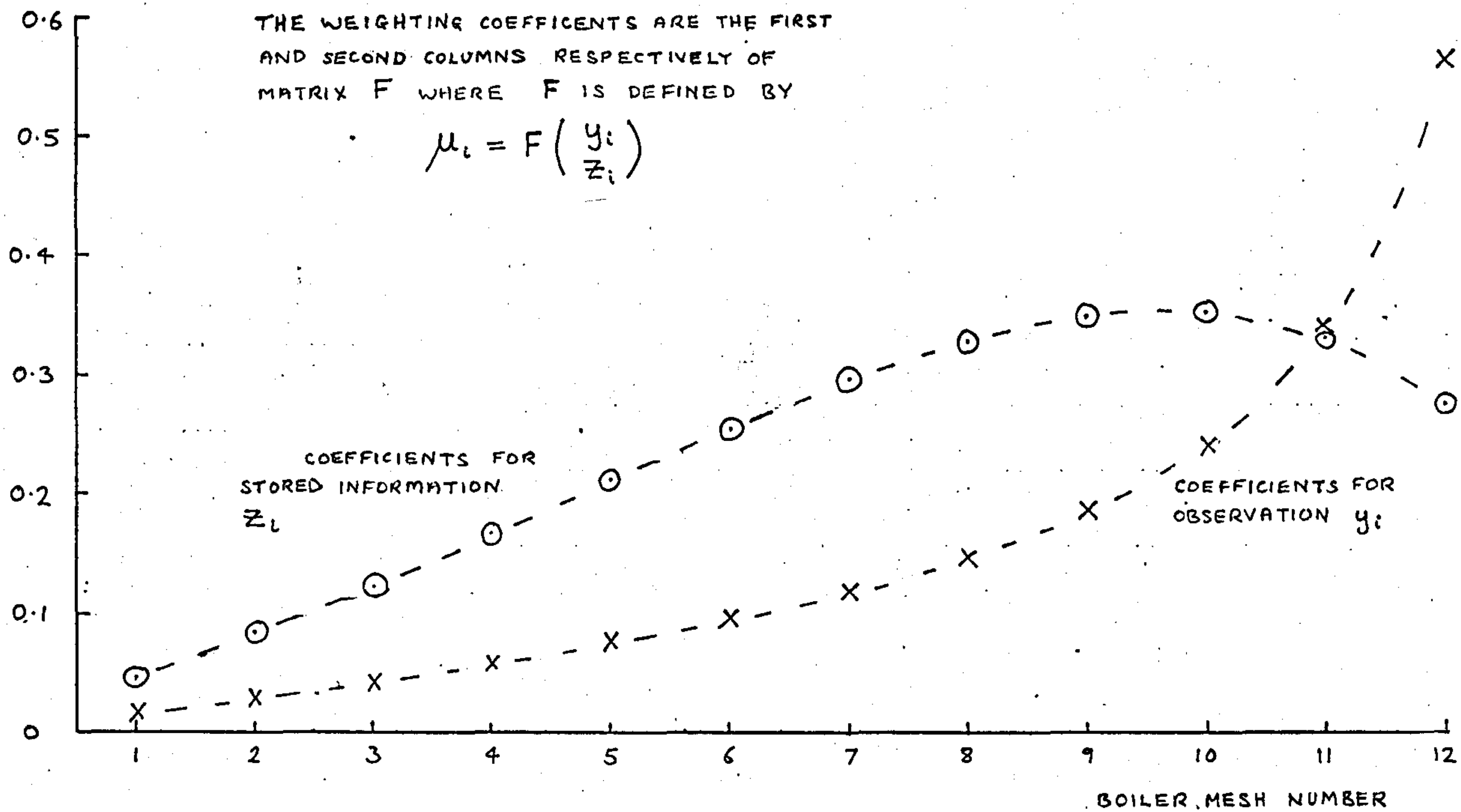


FIGURE 8.10. WEIGHTING COEFFICIENTS FOR PLANT STATE ESTIMATE (CASE L)

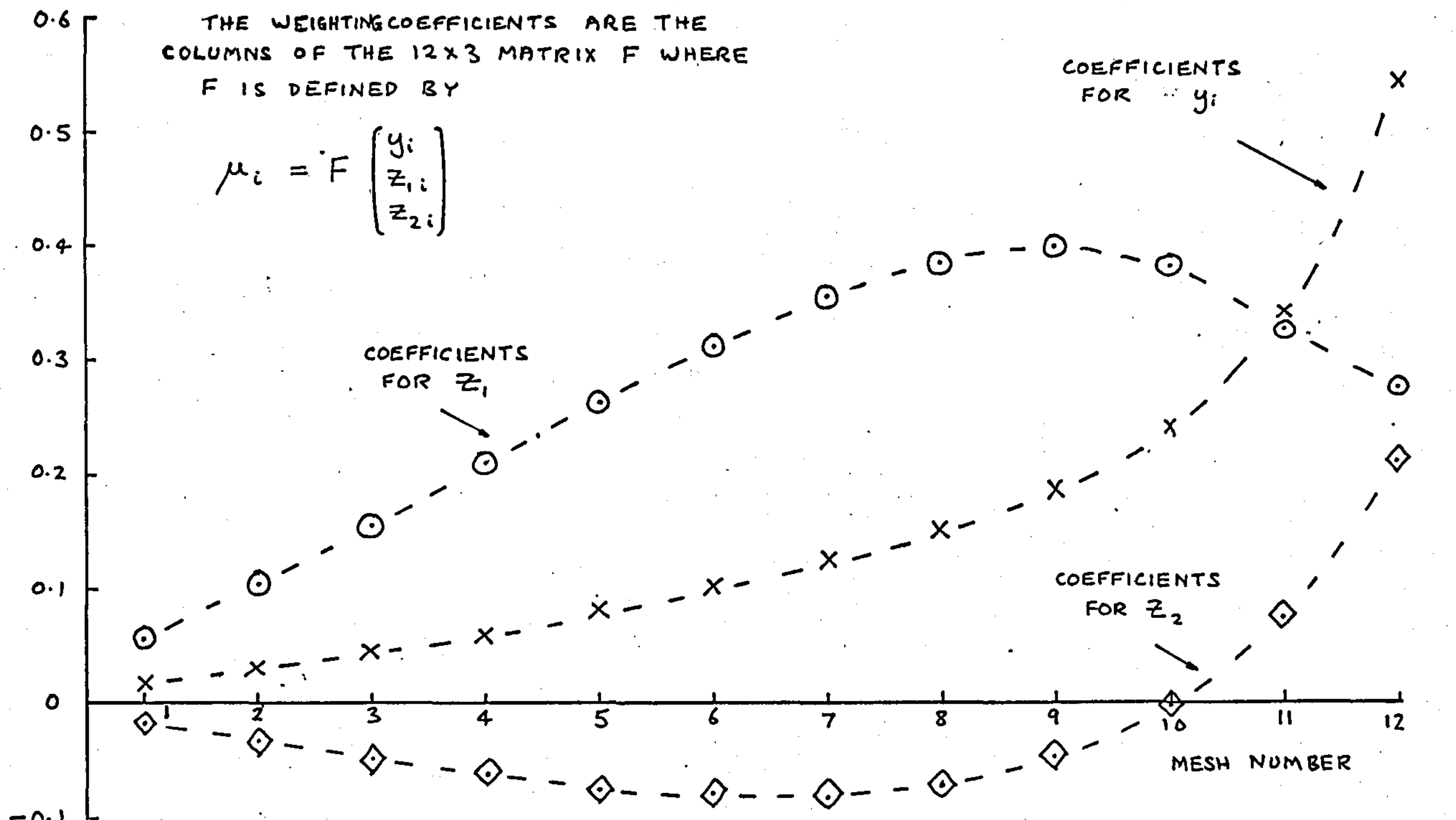


FIGURE 8.11 WEIGHTING COEFFICIENTS FOR PLANT STATE ESTIMATE (CASE K)

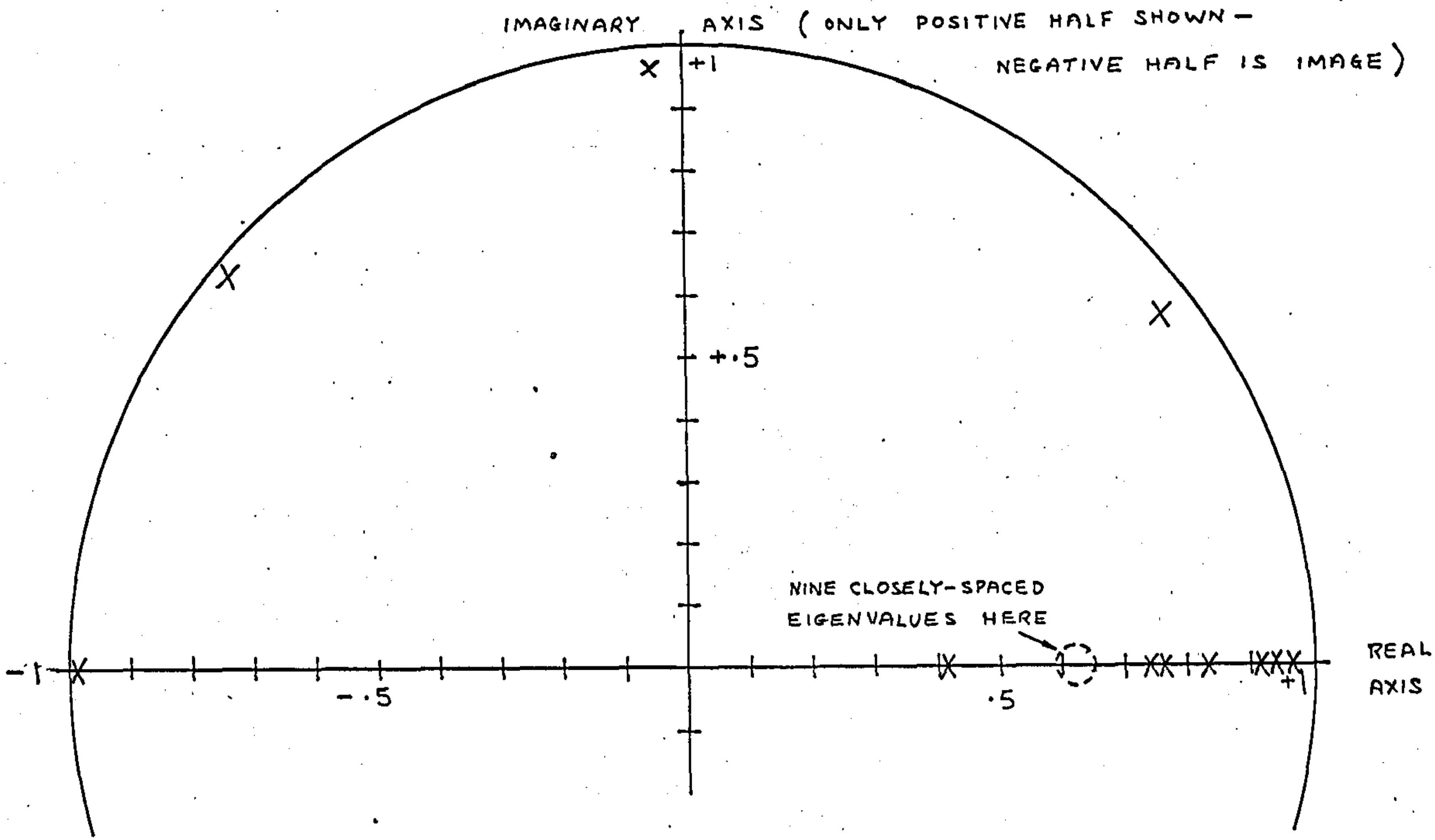


FIGURE 8.12 EIGENVALUE PLOT FOR CASE B (ESTIMATOR ORDER REDUCED FROM 12 TO 11)

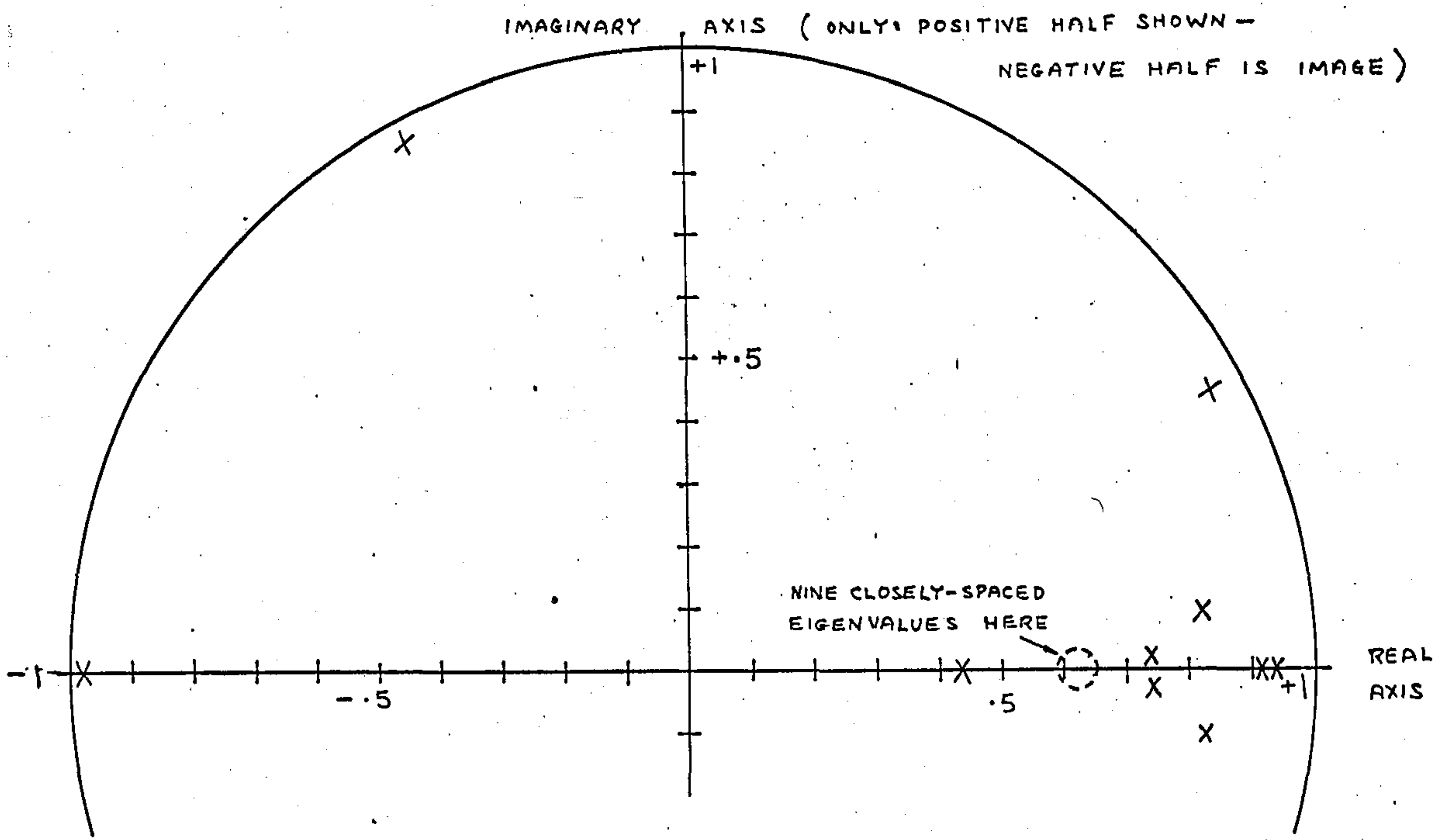


FIGURE 8.13 EIGENVALUE PLOT FOR CASE D (ESTIMATOR ORDER REDUCED FROM 12 TO 9)

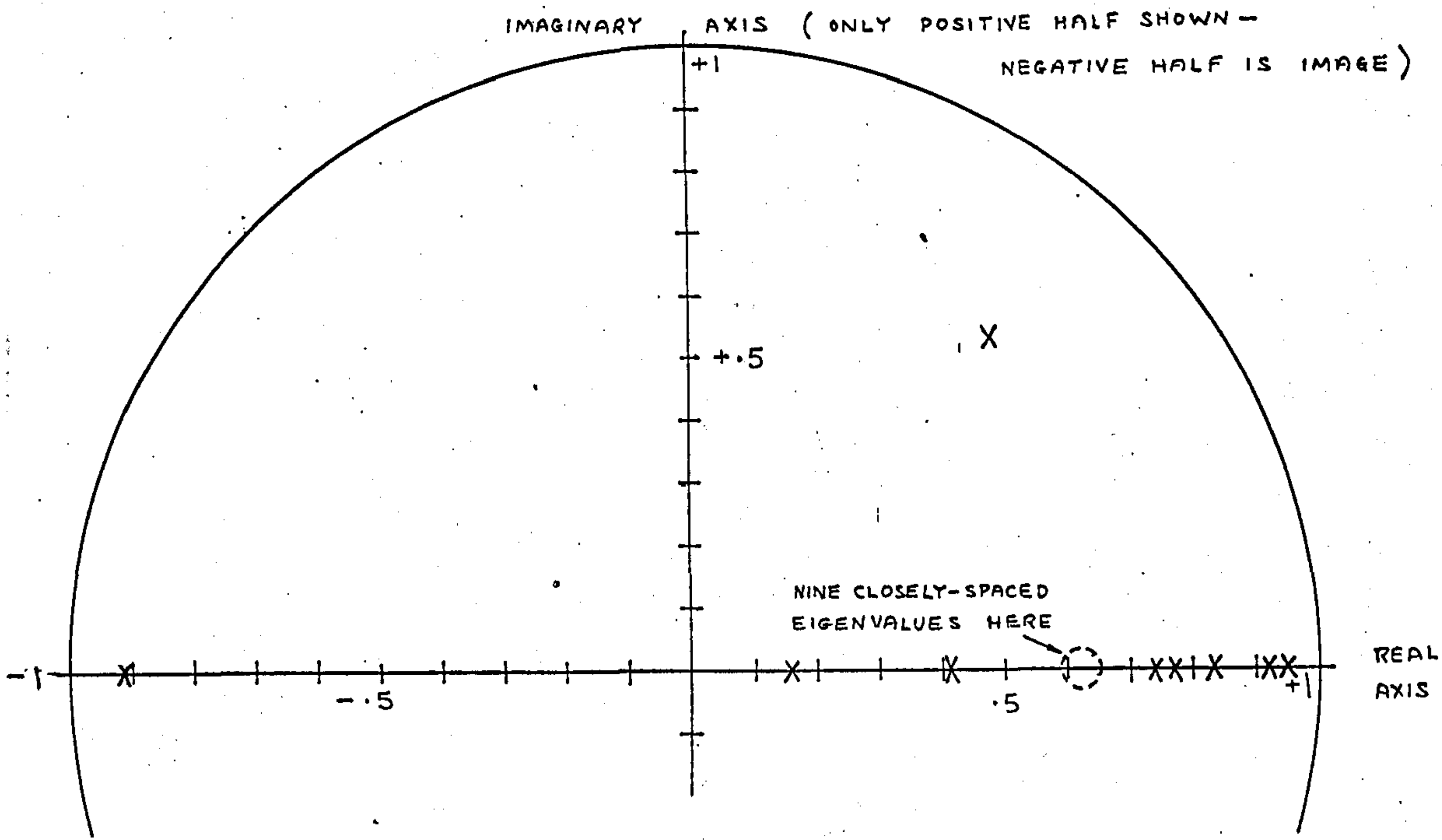


FIGURE 8.14 EIGENVALUE PLOT FOR CASE F (ESTIMATOR ORDER REDUCED FROM 12 TO 7)

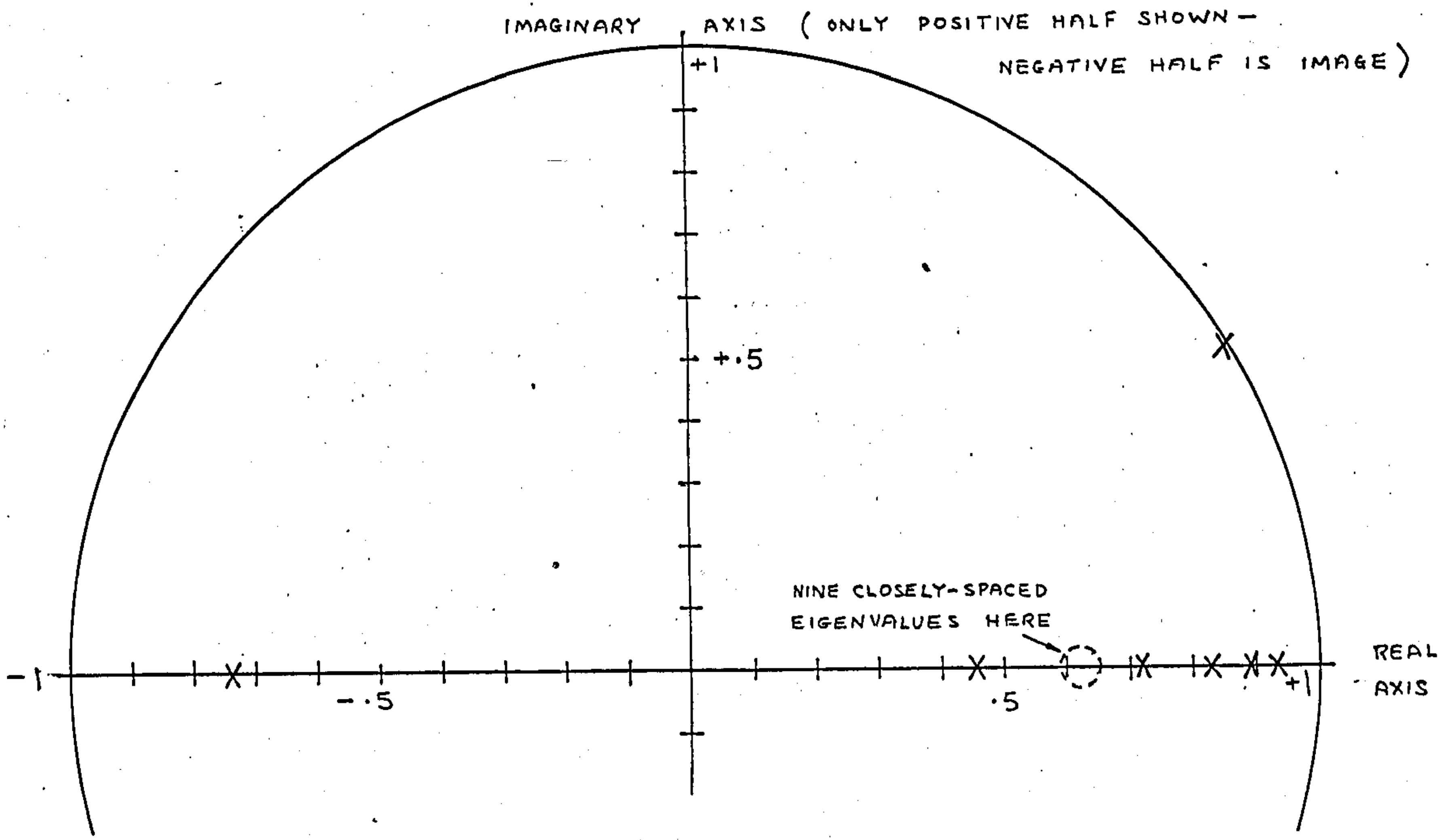


FIGURE 8.15 EIGENVALUE PLOT FOR CASE H (ESTIMATOR ORDER REDUCED FROM 12 TO 5)

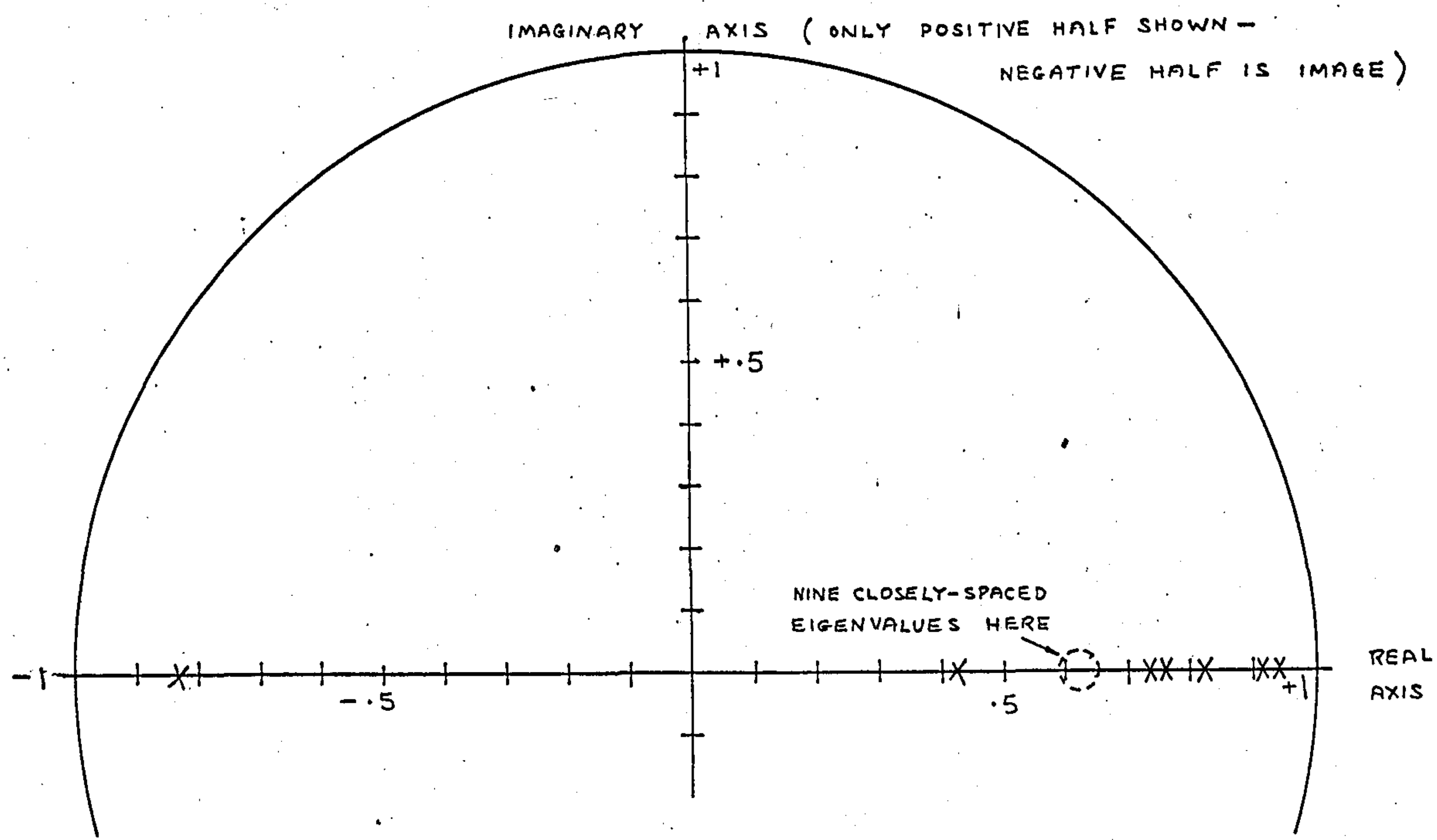


FIGURE 8.16 EIGENVALUE PLOT FOR CASE I (ESTIMATOR ORDER REDUCED FROM 12 TO 4)

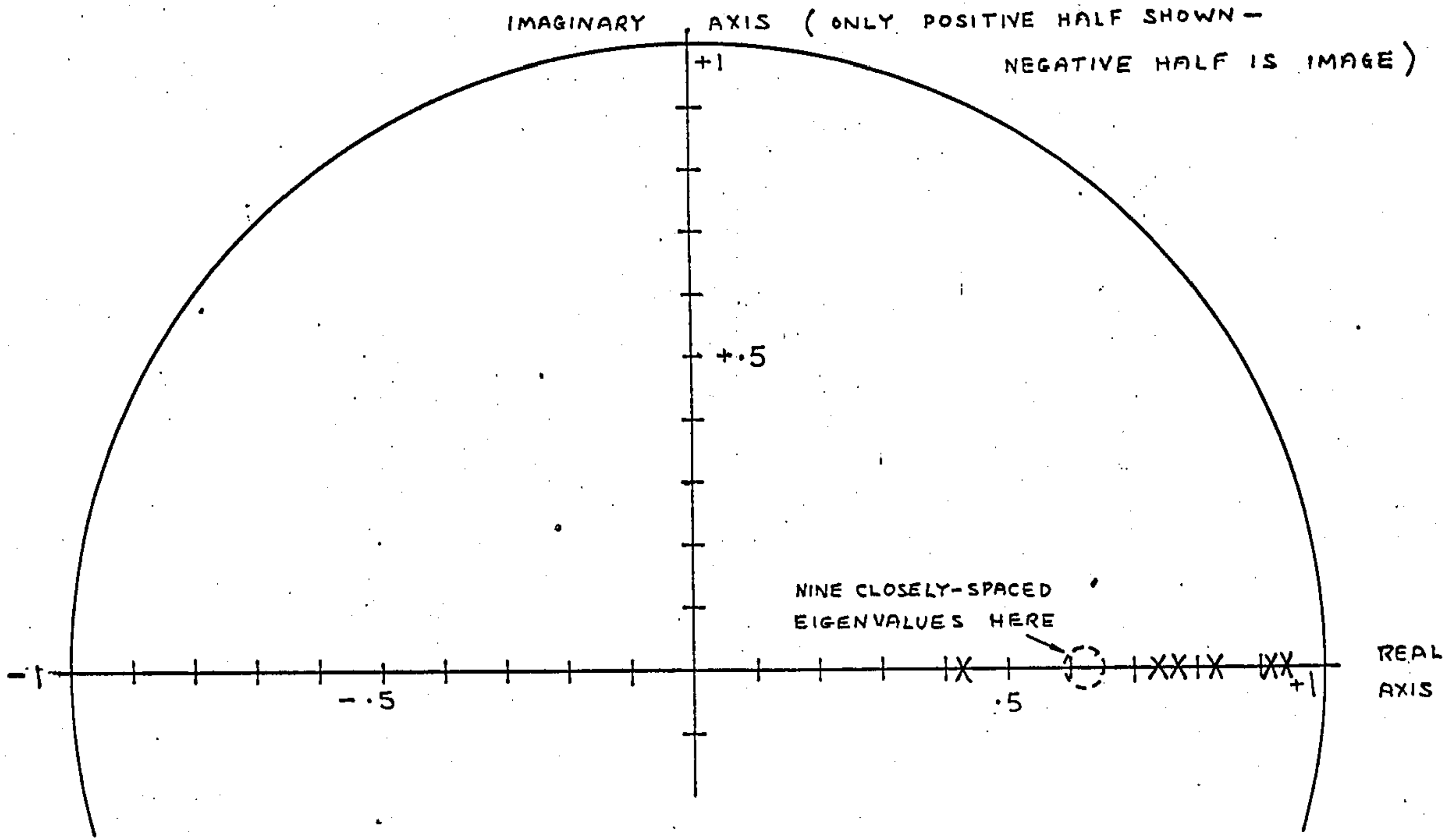


FIGURE 8.17 EIGENVALUE PLOT FOR CASE J (ESTIMATOR ORDER REDUCED FROM 12 TO 3)

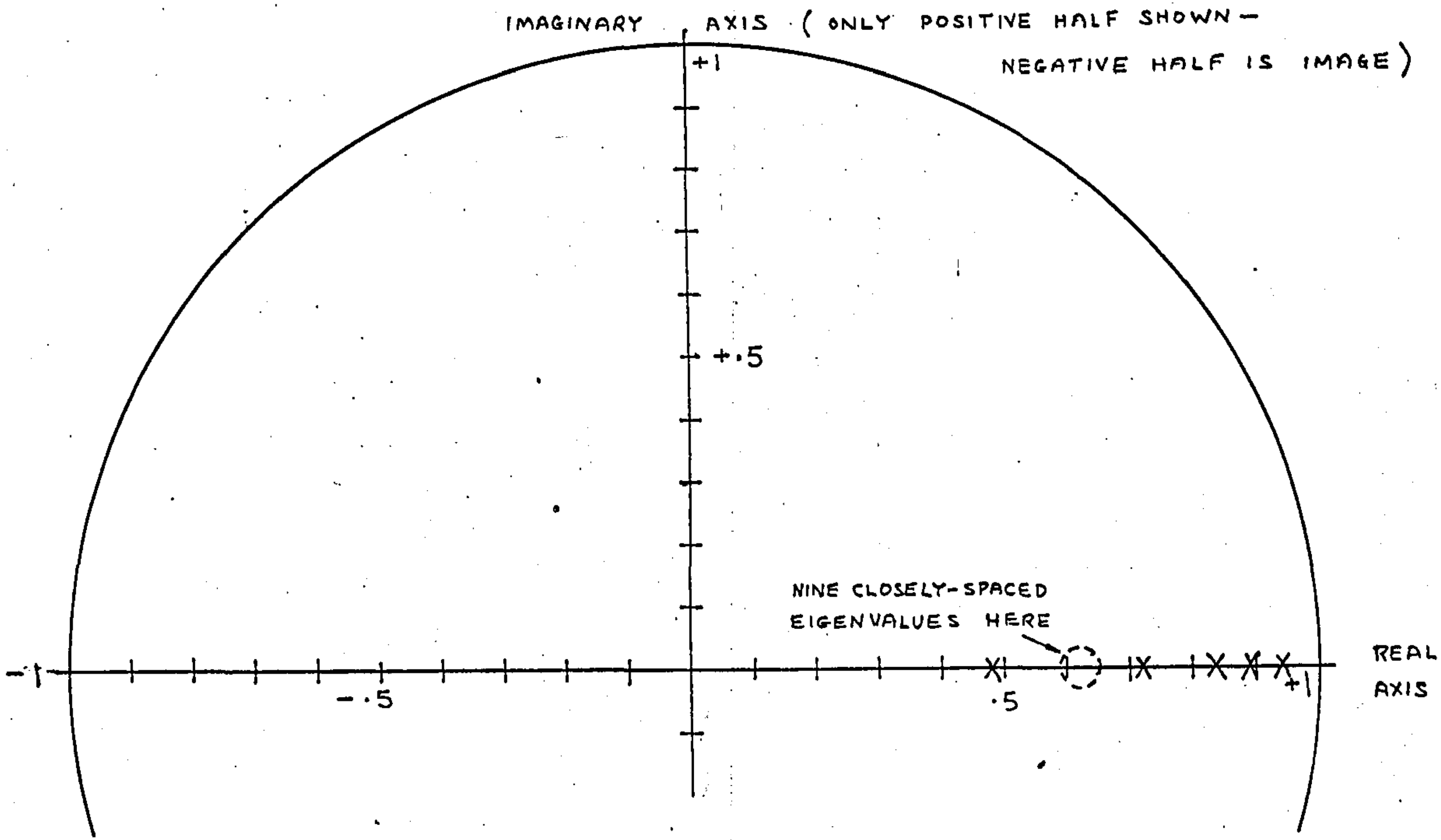


FIGURE 8.18 EIGENVALUE PLOT FOR CASE K (ESTIMATOR ORDER REDUCED FROM 12 TO 2)

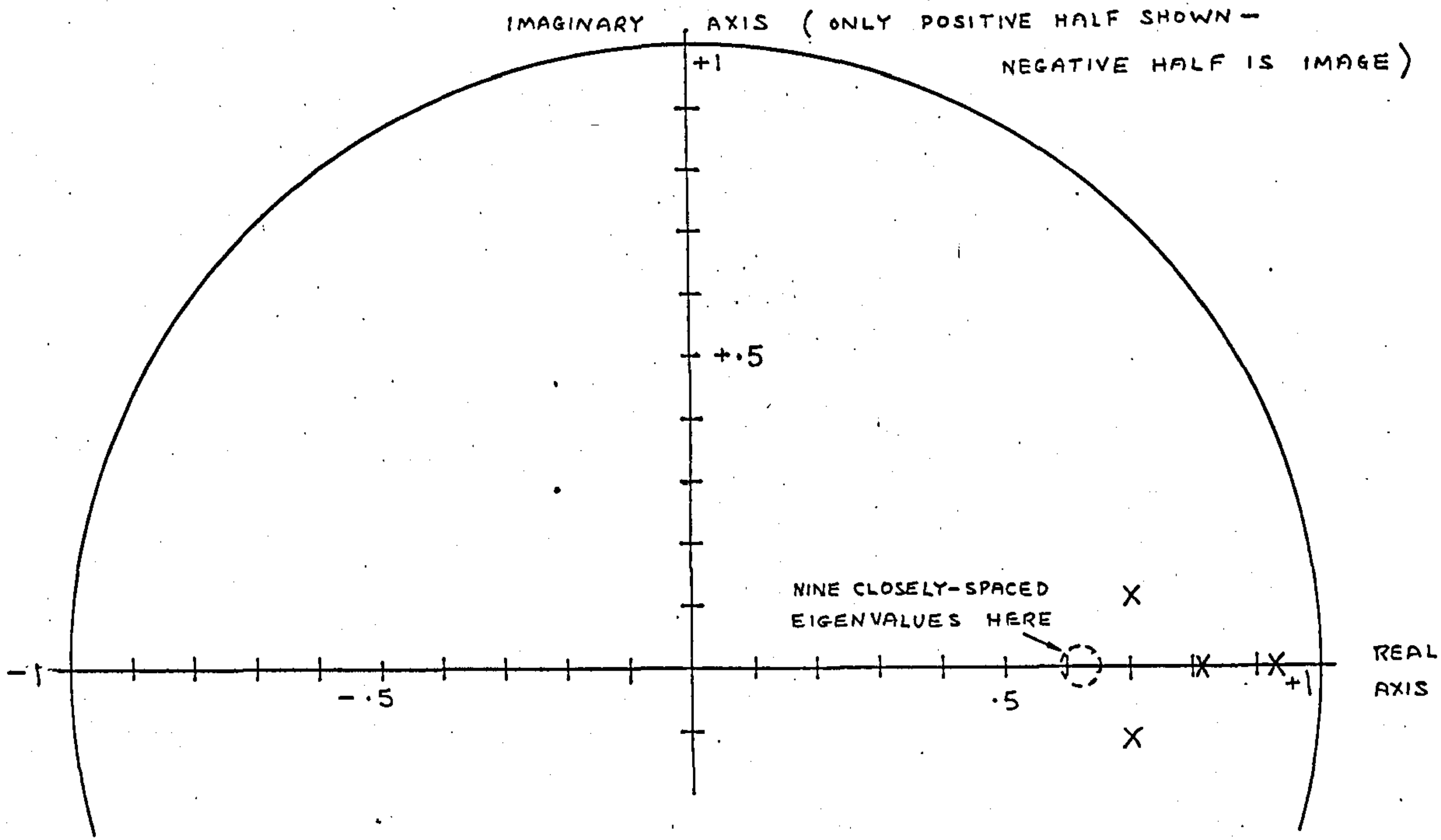


FIGURE 8.19 EIGENVALUE PLOT FOR CASE L (ESTIMATOR ORDER REDUCED FROM 12 TO 1)

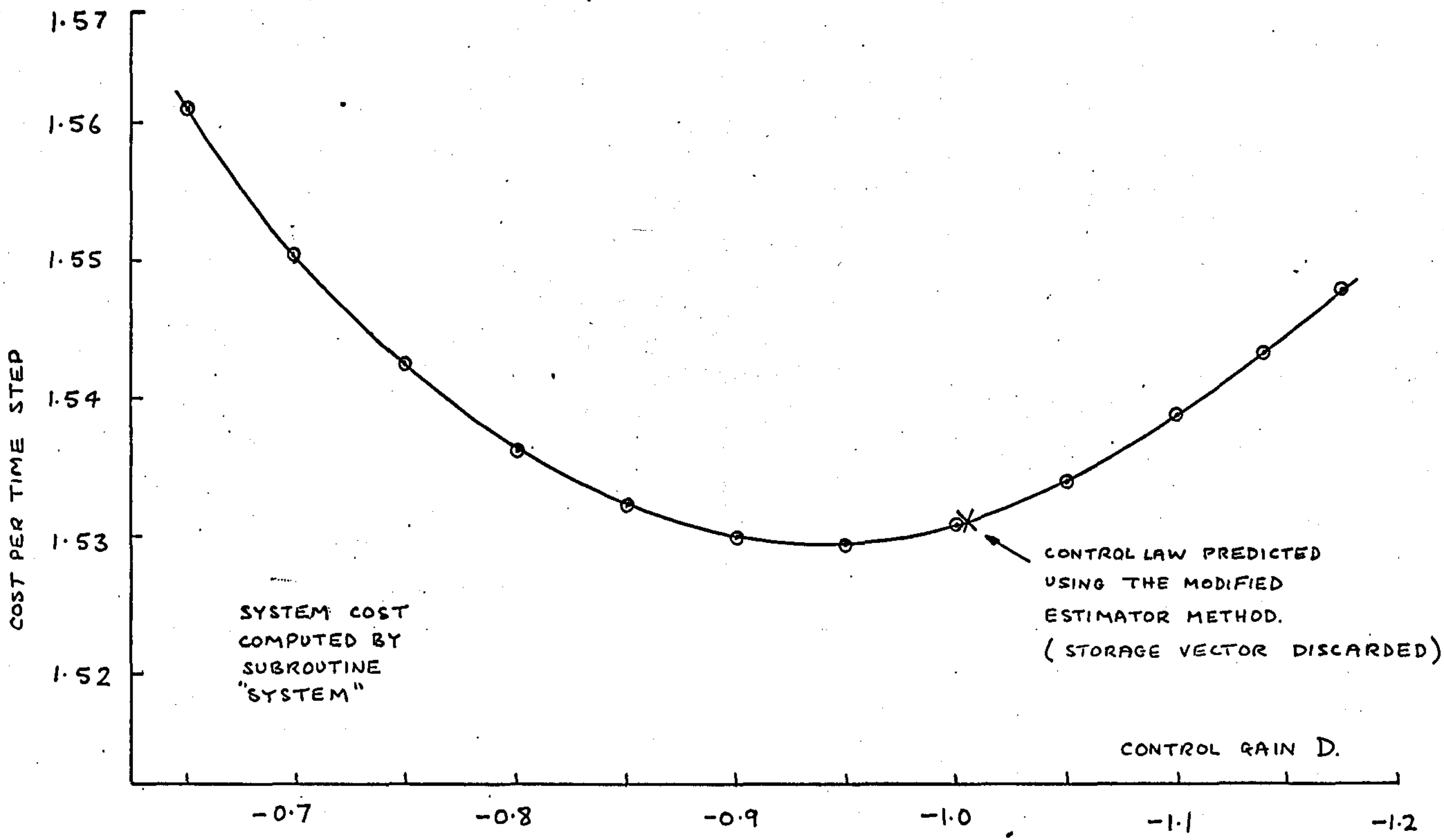


FIGURE 8.20 PROPORTIONAL CONTROL OF BOILER.

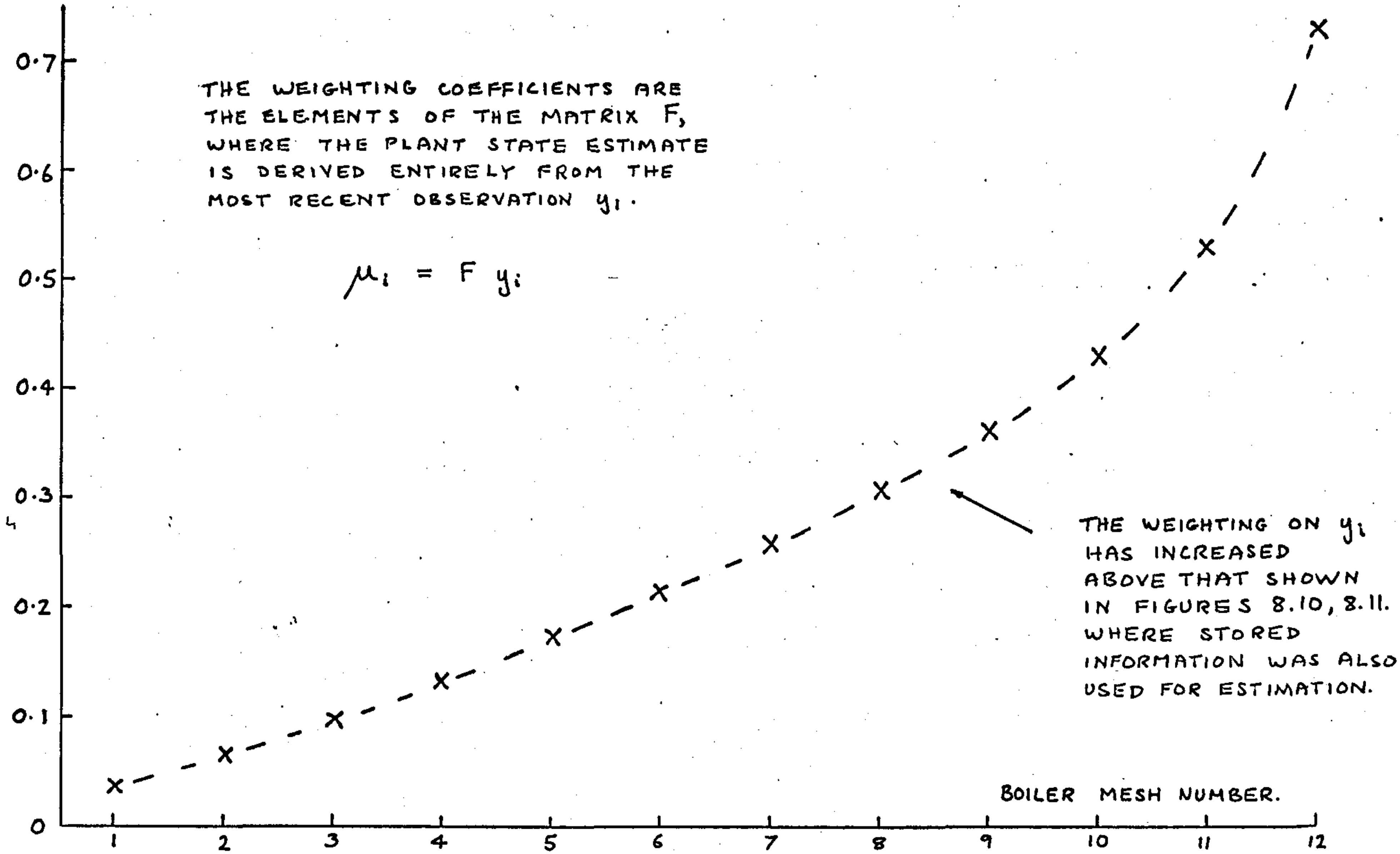


FIGURE 8.21. WEIGHTING COEFFICIENTS FOR PROPORTIONAL CONTROL

```
STMT    MATLAN STATEMENT

  1              MAIN
  2              READ      (A,B,P,H)
  3              WRITE     (A,B,P,H),FORMAT=A5
  4              TRANS     H,HT
  5              MULT      HT,H,V
  6              WRITE     (V),FORMAT=A5
  7              FORMS     I,(12,12),(1,1),(1,1),12,1.0
  8              LOOP      J1,J,1,20
  9              LOOP      I1,K,1,5
 10              CALL      CONTRL(A,B,P,V,I,LAMDA,PI)
 11   I1         LOOPEND
 12              WRITE     (I,PI,LAMDA),FORMAT=A5
 13   J1         LOOPEND
 14              CALL      APUNCH(I),F
 15              CALL      APUNCH(PI),F
 16              CALL      BPUNCH(LAMDA),F
 17              MULT      B,LAMDA,BL
 18              SUB       A,BL,ABL
 19              WRITE     (ABL),FORMAT=A5
 20              CALL      GEDPCH(ABL),F
 21              CALL      GEDPCH(A),F
 22              END
```

Listing 8.1 Matlan program to calculate optimal boiler control law using subroutine CONTRL.

```
STMT    MATLAN STATEMENT

  1            MAIN
  2            READ      (A,B,H,R,G,LAMDA)
  3            WRITE     (A,B,H,R,G,LAMDA),FORMAT=A5
  4           CALL       SYSOPT(A,B,H,R,G,LAMDA,SYSC,SYSD,SYSE,SYSF)
  5            WRITE     (SYSC,SYSD,SYSE,SYSF),FORMAT=A5
  6            CALL      BPUNCH(SYSC),F
  7            CALL      APUNCH(SYSE),F
  8            CALL      BPUNCH(SYSF),F
  9            END
```

```
STMT    MATLAN STATEMENT

  1            SUBPRO    SYSOPT(A,B,H,R,G,LAMDA,SYSC,SYSD,SYSE,SYSF)
  2            MULT       B,LAMDA,BL
  3            SUB       A,BL,ABL
  4            INV       R,RM
  5            TRANS     H,HT
  6            MULT      HT,RM,HR
  7            MULT      G,HR,SYSF
  8            MULT      SYSF,H,GH
  9            SUB       1.0,GH,IG
 10            MULT      IG,ABL,SYSE
 11            MULT      LAMDA,SYSE,SYSC
 12            MULT      -1,SYSC,SYSC
 13            MULT      LAMDA,SYSF,SYSD
 14            MULT      -1,SYSD,SYSD
 15            RETURN
 16            END
```

Listing 8.2 Subroutine SYSOPT

Computes for the optimal "Kalman Filter" system the matrices for use in subroutine SYSTEM.

```
STMT    MATLAN STATEMENT

  1              MAIN
  2              READ      (A,B,H,Q,R,G,LAMDA,PCL,THETA,ADIM)
  3              WRITE     (A,B,H,Q,R,G,LAMDA,PCL,THETA,ADIM),FORMAT=A5
  4              TRANS     H,HT
  5              MULT      HT,H,V
  6              WRITE     (V),FORMAT=A5
  7              RDIM      A,DX
  8              RDIM      H,DY
  9              ADD       0,1,ZDIM
 10              ADD       DY,ZDIM,DYZ
 11              FORMS     P,(DYZ,DYZ),(1,1),(1,1),DYZ,1.0
 12              FORMS     F,(DX,DYZ),(1,1),(1,1),DYZ,1.0
 13              LOOP      J1,J,1,50
 14              CALL      ESTIM(A,B,H,Q,R,G,GNEXT,SIGMA)
 15              CALL      OPRED,(A,B,H,GNEXT,SIGMA,F,P,THETA,ADIM,LAMDA,FNEXT*
                     ,PNEXT,R,RGAM,SYSC,SYSD,SYSE,SYSF,L)
 16              COPY      RGAM,G
 17              COPY      PNEXT,P
 18              COPY      FNEXT,F
 19     J1       LOOPEND
 20              NEWPAGE
 21              WRITE     (ZDIM,G,SIGMA,F,P,L),FORMAT=A5
 22              WRITE     (SYSC,SYSD,SYSE,SYSF),FORMAT=A5
 23              CALL      SYSTEM(A,B,H,PCL,V,Q,R,SYSC,SYSD,SYSE,SYSF,VAR,SYS,*
                     SUM,20)
 24              WRITE     (SYS,VAR,SUM),FORMAT=A5
 25              CALL      GEDFMN(SYS),F
 26              END
```

<u>Listing 8.3</u> Matlan program to call subroutine OPRED and construct the reduced order estimator.

CHAPTER 9

OBSERVER SYSTEMS

1. Luenberger Observer System

The reduced order estimator system that has been developed in earlier chapters has many similarities with the "observer" system as described by Luenberger (references 4, 5 , 6). This observer theory was developed for continuous time systems and the corresponding discrete time observer theory was described by Aoki and Huddle (reference 40). Some other developments in observer theory are discussed in section 8.

The system is described as in Chapter 2 by

$$x_{i+1} = Ax_i + Bu_i + \xi_i \tag{2.1}$$

and

$$y_i = Hz_i + \eta_i \tag{2.2}$$

An estimator is then constructed according to

$$z_{i+1} = Fz_i + Dy_i + Gu_i \tag{9.1}$$

The vector z contains memory elements and is of order l where

$$n - m \leq l \leq n$$

and n and m are the dimensions of x and y respectively.

An estimate of the plant state is taken as the expression

$$\hat{x}_i = Pz_i + Vy_i \tag{9.2}$$

For the deterministic system in which ξ_i and η_i are zero it is stated by Aoki and Huddle that z_i can be related to x_i by a fixed linear transformation T provided z_o is chosen to be equal to Tx_o where T satisfies

$$TA - FT = DH \tag{9.3}$$

and G in (9.1) is given by G = TB.

It is indicated that, with the restriction that l = n - m, a choice of matrix F can be made to ensure that its eigenvalues realise an arbitrarily fast response in the sense that any initial error in z_o is quickly eliminated.

2. Stochastic Observer System

Aoki and Huddle then consider the stochastic case where ξ_i and η_i in (2.1) and (2.2) are non-zero. They state that by choosing in general F and D such that

$$D = TAV \text{ and } F = TAP \tag{9.4}$$

and requiring that T satisfies

$$PT + VH = I_n \tag{9.5}$$

where I_n is the n x n unit matrix, then (9.3) will be satisfied.

They proceed to examine the error covariance matrix of the estimator given by (9.2) and by means of a numerical example show that such an observer system can have a performance nearly as good as the Wiener-Kalman filter system.

This result is very clearly in line with the observations of chapter 4, where a reduced order estimator was constructed. This similarity will be pursued further in section 4.

3. Observers with Minimum Mean-Square Cost

It is possible to choose an observer system according to a design procedure which minimises a weighted-mean-square estimation error. This is the approach of Iglehart and Leondes (ref 41) who carry out the minimisation in a stage by stage manner by direct differentiation of the weighted mean square error cost function. Setting the two differential coefficients so obtained to zero for a minimum value gives a pair of (matrix) simultaneous equations whose solution yields choices for the matrices V and P in (9.2). In turn a matrix T is found which satisfies (9.5).

Although the system is chosen now to be optimal in a given sense, the order of the observer system, l, is again constrained to

$$n - m \leqslant l \leqslant n$$

as was the observer system of Aoki and Huddle. It is further shown that when $l = n$ the observer system is equivalent to the Weiner-Kalman Filter.

There is similarity between the approach to observer design adopted by Iglehart and Leondes and the theory in chapter 5. In both cases a minimisation of a weighted-mean-square estimation error is taken as the criterion of the estimator design. The methods used to accomplish the design of the estimator are, however, entirely different.

4. Observer Dimension Constraints

As mentioned above, the observer system described in sections 2 and 3 has a structural similarity with the reduced order estimator developed in Chapter 4. Both form estimates of the plant state using memory elements z_i together with the most recent observation y_i and both lead to an equation for updating the memory element. However, there are very big differences in approach and in allowed dimensions, as follows:

(a) The "reduced order estimator" of chapter 4 allows the observer order to be reduced by any desired degree down to a minimum of unity. The "observer" system of Aoki and Huddle or Iglehart and Leondes allows the observer to have an order reduced from the plant order n, but only down to a minimum order of n - m, where m is the order of the observation vector.

(b) The extension of the observer approach to low order as in chapter 4 is made possible by use of "a priori" information about the plant parameters. No such information is used in the conventional "observer" system, and this leads to a concept of a "minimal order" observer system of order n - m.

In the once-through boiler model system, of plant model dimension 12, the "minimal order" of a conventional observer would be 11 since the observation vector has dimension 1. As this is typical of a distributed parameter system, the conventional observer approach of Aoki and Huddle can be seen to be severely restricted. On the other hand it was shown in chapter 8 that it was entirely possible to control the plant satisfactorily with an estimator of order 1.

However, despite the difference in approach and the difference in dimensional constraints there remains an area of overlap when the order of the estimator system of chapter 4 is constrained to lie between n and n-m. The relationships between the two systems in this region of overlap is discussed in the next section.

5. Comparison of Estimator Systems

The 2 x 2 system example given by Aoki and Huddle can be adapted to be identical with the 2 x 2 example system used in chapters 2, 4, 5 and 6. Applying the design technique of Aoki and Huddle involves minimising element Γ_{11} of the error covariance matrix Γ in the asymptotic solution where

$$\Gamma_{11} = \frac{8v_1^3 + 4v_1^2 + 1}{v_1(2 - v_1)} \tag{9.6}$$

and v_1 is a partition of matrix V, i.e.

$$V = \begin{pmatrix} v_1 \\ 1 \end{pmatrix} \tag{9.7}$$

Γ_{11} has a minimum when $v_1 = 0.29$ and this gives for the observer system an error covariance matrix:-

$$\Gamma = \begin{pmatrix} 3.09 & 1.16 \\ 1.16 & 4 \end{pmatrix} \tag{9.8}$$

This may be compared with the asymptotic solution for Γ computed for the reduced order estimator of chapter 6 as

$$= \begin{pmatrix} 3.01 & 1.00 \\ 1.00 & 3.08 \end{pmatrix} \tag{9.9}$$

It is clear that the observer design method of Aoki and Huddle has resulted in a poorer estimator than that of chapter 6. The associated gain matrices of (9.1) and (9.2) are

$$\left.\begin{array}{lll} P = \begin{pmatrix} -.29 \\ 0 \end{pmatrix} & V = \begin{pmatrix} .29 \\ 1 \end{pmatrix} & T = (-3.45 \;\; 1) \\ D = .29 & F = .71 & G = -3.45 \end{array}\right\} (9.10)$$

These matrices can be seen to satisfy the constraint (9.5).

The control system that results from this observer system is similar numerically to the reduced order estimator of chapter 6, with the exception that the zero element in P requires that the estimate of x_2 in the state vector is made only on the basis of the observation y and no stored information from z is used. This is clearly a constraint introduced by Aoki and Huddle and explains the poorer estimator performance as shown above in comparing (9.8) and (9.9).

6. Is the reduced order estimator a Luenberger observer?

Having shown that the performance of the reduced order estimator of chapter 6 is superior to the observer system of Aoki and Huddle it is reasonable to investigate whether the reduced order estimator is merely a superior observer system or whether, as it uses "a priori" information which the conventional observer does not, it has no theoretical connection with observer systems of the Luenberger type. A direct test of this is to see whether the reduced order estimator gain matrices satisfy the observer constraint (9.3) Taking the 2 x 2 example system of chaper 6 the gain matrices in the notation of (9.1) and (9.2) can be derived as follows:

Equivalencing (6.32) and (9.2)

$$P = \begin{pmatrix} -2.10 \\ 0.81 \end{pmatrix} \qquad V = \begin{pmatrix} 0.25 \\ 0.77 \end{pmatrix} \tag{9.11}$$

Equivalencing (6.28) and (9.1) is less direct. Setting

$T = (t_1, t_2)$ then

$$G = TB = (t_1, t_2) \begin{pmatrix} 1 \\ 0 \end{pmatrix} = t_1 \tag{9.12}$$

From (6.27)

$$u_i = 1.83 z_i - 0.508 y_i$$

Substituting into (9.1) gives

$$\begin{aligned} z_{i+1} &= Fz_i + Dy_i + Gu_i \\ &= (F + 1.83t_1)\, z_1 + (D - 0.508t_1)\, y_i \end{aligned} \tag{9.13}$$

Equivalencing this with (6.28) gives

$$F = -0.034 - 1.83t_1 \tag{9.14}$$

and $$D = 0.246 + 0.508t_1 \tag{9.15}$$

The relations (9.4) require

$$\left.\begin{aligned} -.25t_1 + 1.02t_2 &= 0.25 \\ 0.27t_1 + 1.29t_2 &= 0.034 \end{aligned}\right\} \tag{9.16}$$

giving a solution

$$T = (t_1, t_2) = (0.485, 0.127) \tag{9.17}$$

Substituting into the left hand side of the constraint equation (9.5) gives

$$PT + VH = \begin{pmatrix} 1.00 & -.02 \\ .39 & .87 \end{pmatrix} \tag{9.18}$$

The observer constraint of (9.5) required that this expression be the unit matrix, and as this is not the case it follows that the reduced order estimator of chapter 6 is not an observer system in the sense of Luenberger.

7. The Observer T Matrix

Central to observer theory is the matrix T which is said to define z in the absence of noise such that

$$z = Tx \tag{9.19}$$

In the derivation of the reduced order observer in chapter 4 no such concept was encountered since the storage element z was generated directly from probability considerations. However such a consideration may be introduced by asking what value would z be likely to have for any given plant state x_i? Again this would have to be calculated in terms of probability distributions using the known distribution, from (4.42), of x_i given z

$$p(x_i|v_i) \underset{e}{=} (x_i - F_i v_i)' \Gamma_i^{*-1}(.) \tag{4.42}$$

where v_i is the information vector

$$v_i = \begin{pmatrix} y_i \\ z_i \end{pmatrix}$$

with "a priori" distribution, from (4.6) of

$$p(v_i) = v_i' P_i^{-1} v_i \tag{4.6}$$

Omitting suffices the above relations give

$$\begin{aligned} p(x, v) &= p(x|v)p(v) \\ &\underset{e}{=} (x - Fv)' \Gamma^{-1}(.) + v'P^{-1}v \\ &= v'(F'\Gamma^{-1}F + P^{-1})v + v'F'\Gamma^{-1}x + x'\Gamma^{-1}Fv + x'\Gamma^{-1}x \\ &= (v - MF'\Gamma^{-1}x)'M^{-1}(.) + x'(\Gamma + FPF')^{-1}x \end{aligned} \tag{9.20}$$

where the matrix M has been defined by

$$M^{-1} = F'\Gamma^{-1}F + P^{-1} \tag{9.21}$$

and the matrix relation of (2.4) has been used.

Since (9.20) may also be written

$$p(x,v) = p(v|x)p(x)$$

it follows that the distribution of v, given x is

$$p(v|x) = (v - MF'\Gamma^{-1}x)M^{-1}(.) \tag{9.22}$$

This distribution defines the expected value of the infomation vector v to be given by

$$\bar{v} = MFGx \qquad (9.23)$$

where the matrix MFG is defined by

$$MFG = MF'\Gamma^{-1} \qquad (9.24)$$

A MATLAN subroutine ADAPT, shown in Listing 9.1, was written to compute M and MFG according to equations (9.21) and (9.24). When applied to the 2 x 2 example system of Chapter 6 and the derived reduced order estimator, subroutine ADAPT gives

$$M = \begin{pmatrix} 4.0 & 0 \\ 0 & 0.314 \end{pmatrix} \qquad MFG = \begin{pmatrix} 0 & 1.0 \\ -.271 & 0.164 \end{pmatrix} \qquad (9.25)$$

so that the best estimate of z is given by

$$\bar{z} = -.27x_1 + .16x_2 \qquad (9.26)$$

Considering that z is known to have a variance of 2.5 (from the prior distribution covariance matrix P) the variance of z about $\bar{z}$, as given by (9.26), of 0.31 is small showing (9.26) to be a very effective estimate.

Equation (9.25) also gives the expected value of y given x_1 and x_2. As this estimate is

$$\bar{y} = Hx$$

it follows that the top row of MFG in (9.25) is the observation matrix H. Also M contains the observation noise covariance matrix R, which has the value 4.0 in this example.

The same technique may be applied to the once-through boiler model of chapters 7 and 8. Taking case L where the storage element is one dimensional, the matrix MFG as given by ADAPT has a bottom row as shown in Figure 9.1. It can be seen that the difference between the weightings of y and z is that those of y decrease more rapidly so that z contains a greater weighting of the state variables x_9, x_8, x_7, etc. This explains why it was that, in chapter 8, the state estimate as expressed by the weightings of figure 8.10 on z peak for x_8 and x_9.

Thus in the context of observer theory the relation (9.26) has been seen to define a relation between the storage element z and the state variable x which is analogous to the relation

$$z = Tx \tag{9.19}$$

which appears for example in reference 5.

This completes the comparison of observer systems with the reduced order estimator as developed in chapter 4. Broadly the reduced order estimator, while not an observer within the Luenberger definition, has been shown to possess all the useful properties of observer systems.

8. Other Developments in Observer Theory

It is clear that observer theory requires the choice of matrices P, V F,D in (9.1) and (9.2) and various methods for choosing these matrices have been developed. Tse and Athans (reference 42), and Tse (reference 43) consider the relationship between deterministic and stochastic minimal order observer systems and also show how, when certain observations are noise-free these may be incorporated into the, otherwise stochastic, observer. Leondes and Novak (reference 44) also deal with this topic and, as in the earlier work by Iglehart and Leondes, the optimal "intermediate order" observer is obtained by differentiation of a cost function. Various limiting cases are examined showing the Kalman Filter and the Tse and Athans observer to be embraced.

The design of a continuous-time observer system is considered by Newmann (reference 45) in the light of uncertainties in the plant initial conditions, although the system is otherwise non-stochastic.

This same viewpoint is adopted by O'Reilly and Newmann (reference 46) for the discrete-time system, and a design method for an observer based control system is developed. Surprisingly perhaps, the certainty equivalence principle is found to be applicable so that control and observer systems may be independently determined. An equivalent "canonical" form for an observer is utilised which by means of an arbitrary gain matrix allows some simplification. Perhaps the reason for the relevance of the certainty equivalence principle is that the only covariance parameter involved, that of the initial state uncertainty, will remain unaffected by the subsequent control actions.

Yoshikawa (reference 47) considers the stochastic discrete-time filter problem. If there are k noise free observations, it is shown that, with certain rank conditions, the optimal filter system is of order (n - k), and the method for constructing this filter is developed. If a small variance was taken for the noise-free observations, then in practice the Kalman Filter would seem to give an equally useful filter system. The method of Yoshikawa does however give a rigorous treatment.

There is published work on the theory of minimal realisations, and Akaike (reference 48) gives a number of references in his paper on the stochastic theory of minimal realisations. This theory would appear to be relevant to observer theory since by using the smallest order model to represent the plant, a reduction of observer or estimator dimension is achievable.

In the various methods of observer construction the restriction to order n - m appears not to make best use of the known nature of the control inputs to the system. In Chapter 8 a successful estimator of much lower order was designed. This relationship between control and estimation is considered for the time-varying case by Asher and Durrett (reference 49) and by Kurtaran (reference 22) for the constant system. In each case the problem is approached directly by means of an augmented state vector and this approach is discussed further in the next chapter.

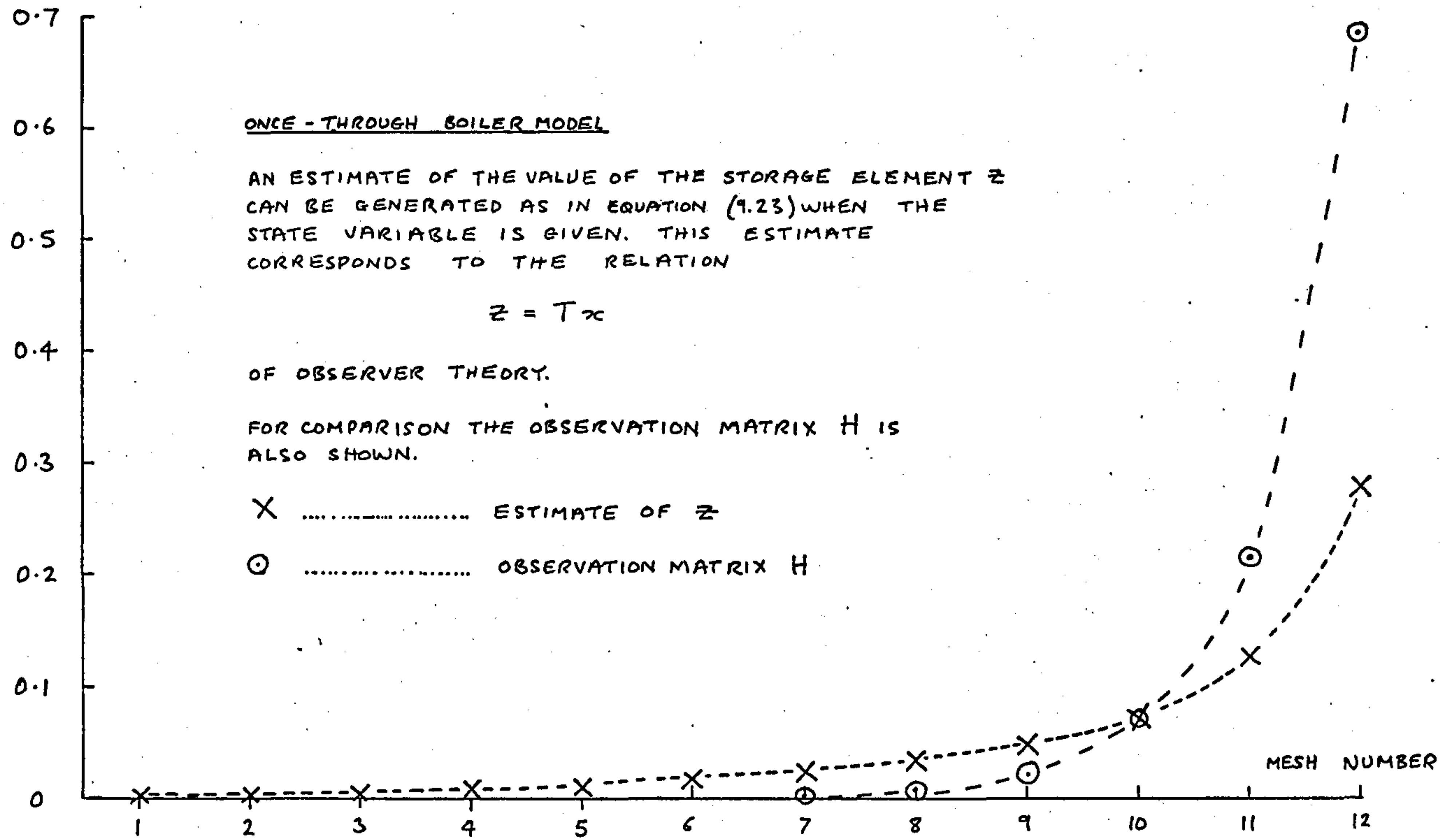


FIGURE 9.1 COMPOSITION OF STORAGE ELEMENT IN TERMS OF STATE VARIABLE

```
STMT    MATLAN STATEMENT

                SUBPRO     ADAPT(F,G,P,M,MFG)
                INV        P,PM
                INV        G,GM
                CALL       TLSIDE(F,GM,FF)
                ADD        FF,PM,FP
                INV        FP,M
                TRANS      F,FT
                MULT       M,FT,MT
                MULT       MT,GM,MFG
                RETURN
                END
```

Listing 9.1 MATLAN Subroutine ADAPT

This subroutine computes the matrices M and MFG according to equations (9.21) and (9.24). MFG estimates the value of the information vector (containing y and z) for any given value of the state variable x.

CHAPTER 10

CONCLUSIONS: A discussion of some possible areas of further investigation and a review of the results of the present study

1. Multi-variable Systems

The example of a once-through boiler model which was examined had 12 variables composing its state vector and the design method was shown to behave satisfactorily for this case. The system nevertheless was single input, single output which is the simplest possible in terms of control. The general case of a multivariable system with many observations and many control inputs, while covered by the theory has not been tested using an example.

Such a case would, for example, arise on a once-through boiler since other parameters such as

(a) steam pressure

(b) gas outlet temperature

will require to be controlled. The available control inputs are

(i) water flow rate

(ii) gas flow rate

Thus the once-through boiler example is capable of extension to a three input, three output system. It is normal practice to construct control loops which typically control (a) above using (i) and control (b) using (ii). Using a design based on a reduced order estimator would result in a multi-variable system in which the stored information z is used to construct each of the control inputs.

The controls would thus be cross-connected in a way that is not usual practice. One of the objects of further study would be to ascertain

(i) To what extent would the "shared z" system improve performance over the present practice.

(ii) To what extent is the stored information in z attributable to separate single loops.

The last point is illustrated by Figure 9.1 which shows that z is largely modelling the regions x_9 to x_{12} of the boiler. In a multi-variable system another element of z might model the regions x_1, x_2 of the boiler and be related strongly to the control of gas outlet temperature. If this were the case then the system would approximate to the present practice of building three separate single input, single output loops. The degree of connectedness would thus become a point of interest in such a study, as would the necessary dimensions required for z.

Further interest would attach to the generation of plant state estimates, this being of particular relevance when the boiler materials have temperature limits which must not be exceeded. At present if such an on-line estimate of plant state was required then a special system would be designed and this would be separate from the control systems. However with a unified control/estimation approach the two functions could be combined using the memory elements z as described in the following section.

2. Combined Control and State Estimation

If a plant state variable (or function of state variables) were being estimated in this way it is possible that, with the estimator design method employed, a poor estimate would result since there might be a low weighting on the variable with respect to control requirements (matrix Θ, chapter 5). It would be a simple step to cover this point by injecting into θ a weighting relating to the particular variable of interest. This, together with an adequate dimension for the storage vector z, would ensure an estimate of the variable as close as required to the optimum as given by the Kalman filter.

A more direct approach would be to recall the selection of the stored information according to equation (4.3), i.e.

$$\begin{pmatrix} y_i \\ z_i \end{pmatrix} = T \begin{pmatrix} z_{i+1} \\ \alpha_{i+1} \end{pmatrix} \qquad (10.1)$$

and to impose conditions on the choice of T in order to ensure optimal estimation of a particular variable. Such an approach would have the drawback that while practical for a single time step, an optimal estimate imposes restrictions on all past choices of T. It is likely that the full Kalman filter would be the necessary result for optimal estimation of just one variable.

Thus the former method in which the weighting matrix is suitably modified would appear to be the more suitable method for joint estimation and control.

3. Integral term in controller

Normal plant controllers similar to the three-term type will contain an integral term whose function in part will be to correct drifting of plant variables. With only proportional control action a drift will be only partly corrected. While it is possible to add an integral term to any designed controller a more basic approach would be to include a drift component in the plant model of the form

$$(x_d)_{i+1} = (x_d)_i + (\xi_d)_i \tag{10.2}$$

where (ξ_d) is a Gaussian independent disturbance.

Other state variables would then be related to this drift variable, according to

$$(x_n^*)_i = (x_n)_i + (x_d)_i \tag{10.3}$$

where x_n is generated in the normal way and x_n^* represents state variables subject to drift. In computing the control law, plant costs would be

$$J = \sum (x_n^*)' V (x_n^*) \tag{10.4}$$

so that in minimising the cost J the deterministic control law (derived as in chapter 2) would require some feedback from the drift variable x_d. This implies the requirement for an estimator for x_d, and such an estimator would perform the integral term function.

In designing such an integral term controller, infinite variances occur since x_d is not controllable. Although a modification to the theory could perhaps allow this situation to be treated an easier solution would be to re-write (10.2) as, say

$$(x_d)_{i+1} = 0{\cdot}999\,(x_d)_i + (\xi_d)_i \tag{10.5}$$

Such a relation would result in a large, but finite, variance for x_d and there should be no computational difficulties while at the same time a suitable integral controller would result. A particularly interesting aspect of further work concerns the structure of the storage elements of such a controller. Would it be obvious that a particular element of z forms the integrator, or would a transformation of z be required to make the presence of the integrating element obvious?

A related topic concerns the number of such drift elements which require to be established for any given multi-variable control system. Clearly a system which already contains an integrating element has little requirement for an integral term in its control system. A proper approach to this aspect, which results in part from use of a linearised system model, is therefore required.

The use of an "optimal control theory" approach to nuclear power station control was reported by B. Blomnes et al in reference 10. Drift elements were found to be essential and the number used was made equal to the number of control inputs.

4. Certainty Equivalence

In deriving the reduced order control schemes it was assumed that the application of the "certainty equivalence principle" would give a performance close to optimal, and this was found to be the case. As the certainty equivalence principle applies only to the case of perfect information storage it follows that some cost penalty must result from using a reduced order estimator. This implies that for the control law given by

$$u_i = \Lambda \mu_i$$

it will be possible to find a control matrix Λ which performs better in the reduced order case than does that given by the certainty equivalence principle.

This altered control law can be seen as a use of the plant itself to store relevant information. That this is the case can be seen by considering a hypothetical plant which contains a storage channel of the form

$$(x_s)_{i+1} = (u_s)_i \qquad (10.6)$$

With complete storage the Kalman Filter system as shown in Figure 10.1 (a) will not utilise the storage channel and u_s will be zero. However it is clear that if the order of the estimator storage elements were to be reduced, as in Figure 10.1 (b) the control performance could be improved by utilising the available storage channel which is part of the plant. Structurally, since a signal now passes through the control variable u_s, the control matrix Λ will have been modified.

While this example makes it clear that the certainty equivalence does not give optimal control, it does not provide any insight into how the best control law may be synthesised. Figure 10.2 shows how estimation costs might be included in the derivation of Λ.

In turn, however, an altered control matrix would result in a modified choice for the reduction matrix T in the estimator, and this is also shown in Figure 10.2. An inter-dependence is thus set up between control and estimation, which is perhaps capable of rigorous analytical treatment. A computer solution by means of iteration would then be practical in order to achieve the overall optimal reduced order system.

It is interesting to note that, in reference 50 the certainty equivalence principle has been extended to the case of non-Gaussian disturbances which are not independent.

A further related problem concerns the generation of the estimation weighting matrix θ. Equation (6.25) gives a recursive relation for θ whose asymptotic solution is used as an input to the subroutine OPRED. However the coefficients in the recursive relation were taken from the optimal Kalman filter case, so that there remains scope for an approach in which these coefficients are altered iteratively as the solution of the reduced order control system develops.

A more general treatment which has the benefit of being completely rigorous is given by Kurtaran (reference 22). In this the storage elements are combined with the state variables to form an augmented state vector. Equations are obtained which are satisfied by the optimal reduced order controller, but Kurtaran states that a method is still needed to solve these equations. However such an approach could perhaps be combined with the iterative method described above and allow a truly optimal controller to be generated.

5. Noisy Storage Elements

It has been assumed that the storage elements z are not subject to noise disturbances. However even for a digital controller rounding error effects will effectively introduce noise into these channels, and Figure 10.3 shows a "noisy estimator" system. It was found in chapter 8 that for some cases the eigenvalues of the system were only just inside the unit circle and hence only just stable. Had noise been modelled in the storage system the resulting system might have been more stable.

To include noise is relatively straightforward, requiring only an observation equation such as

$$w_i = z_i + \delta_i \tag{10.7}$$

where w_i is the corrupted stored signal and δ_i is a Gaussian random variable. The theory of chapter 4 would require the construction of the probability distribution $p\,(x_i | y_i, w_i)$ rather than $p\,(x_i | y_i, z_i)$. The derivation of a reduced order estimator would follow as before, including the choice of a reduction matrix T.

In applying the theory some consideration would need to be given to the choice of convariance matrix, R_z say for the disturbance δ_i. Clearly this needs to be related to the variance of z itself otherwise by simply increasing the scale of z the effect of the noise would be reduced. A simple treatment would be to apply a non-singular transformation to z so that its covariance matrix is unity and then to define the disturbance by

$$R_z = \lambda I \tag{10.8}$$

where I is the unit matrix and λ has a small value in the region of 0.02.

A topic which would require investigation would be the optimal storage of the information in the presence of estimator noise, for example the above "orthogonal" storage system could be examined to see if it is the best.

6. Time-Varying Parameters and Adaptive Control

Most plants under control will possess different control parameters as, for example, throughput or load is increased. Normally this change is intentional, so that "identification" of the new parameters is not required. But for optimal performance gains of control and estimation systems will require considerable adjustment in line with the changed parameters.

The simplest possible "adaptive" system will therefore comprise, as shown in Figure 10.4.

(i) A unit concerned with defining the current plant condition, e.g. "load", usually by means of a single slow-moving parameter.

(ii) A means of using this parameter to adjust gains of the control system in a pre-determined manner.

(iii) A control (and estimation) system.

With an estimation system based on a reduced order estimator, the storage elements, z_i, do not obviously map into corresponding storage elements for a system with changed parameters. This problem is less acute with the full Kalman Filter system since the state vector estimate is of full dimension, and it may be assumed that the state vector in one system may be mapped into the state vector in the one with changed parameters.

Given a plant state estimate

$$\begin{aligned} \mu_i &= F\, v_i \\ &= C\, z_i + D\, y_i \end{aligned} \tag{10.9}$$

and a corresponding estimate in the changed system of

$$\mu_i^* = C^* z_i^* + D^* y_i \tag{10.10}$$

it is possible to assess the estimation error $(\mu_i^* - \mu_i)$ resulting from the transfer from (10.9) to (10.10).

As there is always a degree of freedom to transform z_i^* by means of a non-singular matrix S, this may be utilised to allow z_i to be mapped into z_i^* without modification, so simplifying the adaptive control system structure. The estimation error will then be

$$\mu_i^* - \mu_i = (C^*S - C)\, z_i + (D^* - D)\, y_i \tag{10.11}$$

$$= C^*S\, z_i + \omega_i \tag{10.12}$$

where $$\omega_i = (D^* - D)\, y_i - C\, z_i \tag{10.13}$$

The matrix S may be chosen to minimise the estimation error according to some criterion which may be properly taken, as in chapter 5, to be related to the control costs by a cost function

$$J = E\, (\mu_i^* - \mu_i)'\, \theta\, (\mu_i^* - \mu_i) \tag{10.14}$$

Omitting suffices, and using (10.12)

$$\begin{aligned} J &= E\, (C^*S z + \omega)'\, \Theta\, (C^*S z + \omega) \\ &= E\, [\, z'S'C^{*\prime}\theta C^*S z + \omega'\theta C^*S z \\ &\qquad + z'S'C^{*\prime}\theta\omega + \omega'\theta\omega \,] \end{aligned} \tag{10.15}$$

The joint probability distributions of z and w will be known and if

$$\left.\begin{aligned} &E(z\,z') = P_z, \quad E(wz') = P_{wz} \\ \text{and} \quad &E(w\,w') = P_w \end{aligned}\right\} \tag{10.16}$$

then

$$J = \text{Trace}\left(S'C^{*\prime}\Theta C^* S P_z + \Theta C^* S P'_{wz} + S'C^{*\prime}\Theta P_{wz} + \Theta P_w \right) \tag{10.17}$$

Since, from a relation given by Newmann (reference 51),

$$\frac{d}{dS}\left(\text{Trace } AS\right) = A' \tag{10.18}$$

it follows that J has a minimum when

$$\frac{dJ}{dS} = 2\left(C^{*\prime}\Theta C^* S P_z + C^{*\prime}\Theta P_{wz} \right) = \underline{0} \tag{10.19}$$

Solution of this equation for the elements of S gives the optimal choice of S.

A simple demonstration for the 2 x 2 example system would be to modify the system matrix A to

$$A = \begin{Bmatrix} 1 & 0 \\ 2 & 1 \end{Bmatrix} \tag{10.20}$$

This system, with no other modifications to the example of chapter 6, leads to a reduced order estimator system

$$\mu_i = \begin{pmatrix} -1.88 \\ .70 \end{pmatrix} z_i + \begin{pmatrix} .21 \\ .83 \end{pmatrix} y_i \tag{10.21}$$

$$z_{i+1} = -.56\, z_i + .29\, y_i \tag{10.22}$$

With the cost matrix Θ given by

$$\Theta = \begin{Bmatrix} 20.3 & -.7 \\ -.7 & 2.5 \end{Bmatrix} \tag{10.23}$$

and the joint distribution given by

$E(z^2) = 2.6, \quad E(zy) = 8.5$

and $E(y^2) = 45.4$

which in turn, from (10.13) gives

$$E(wz') = \begin{Bmatrix} 5.12 \\ -1.60 \end{Bmatrix} \text{and } E(ww') = \begin{Bmatrix} 10.11 & -3.09 \\ -3.09 & 1.04 \end{Bmatrix} \qquad (10.24)$$

then from (10.17) the cost J is the quadratic in S

$$J = 194.5\, S^2 - 405.6\, S + 212.2 \qquad (10.25)$$

The minimum value of J is 0.8 and this is achieved when

$$S = -\left(C^{*\prime}\theta C^{*}\right)^{-1} C^{*\prime}\theta P_{wz} P_z^{-1} = 1.04$$

and this transformation may now be applied to the estimator equations (10.21) and (10.22) by replacing z by 1.04 z to give

$$\mu_i = \begin{Bmatrix} -1.96 \\ 0.73 \end{Bmatrix} z_i + \begin{Bmatrix} .21 \\ .83 \end{Bmatrix} y_i \qquad (10.26)$$

$$z_{i+1} = -.56\, z_i + .28\, y_i \qquad (10.27)$$

This completes the demonstration of how the minimising S can be found.

Such a choice of S obtained in this or a similar manner will allow z to be used in a continuous fashion during parameter changes without unduly disturbing the plant. It is interesting to note the large magnitude of the coefficients of S and S^2 in (10.25). Clearly an inappropriate choice of S would result in considerable plant disturbance, and hence costs. Confirmation would be required that such a choice of S would give a reasonable performance when the plant parameter change is reversed. The reverse process could well require a different S, in which case a mean value might be suitable. Extension of this approach over the whole range of parameter changes will allow a simple estimator structure to be maintained. However the constraint is not very severe since the consequence of any poor mapping is only temporary and eventually asymptotic costs will dominate, so that a slightly sub-optimal choice of S would normally be adequate.

Further work in this area requires simulation of an adaptive system and its relationsip with a reduced order estimator. However adaptive situations will have particular constraints and problems so that rather than studying a hypothetical example it would be preferable for the design method to be applied to a suitable real plant system where a consideration of adaptive control is necessary. The requirement for adaptive control of the simplest kind, i.e. control gain variation with load or throughput, is frequently met so that implementation of an "optimal control theory" approach using a reduced order estimator is almost certain to require the consideration of adaptive control.

The approach outlined above provides a practical framework for designing such adaptive systems while still maintaining a simple control structure.

7. A Review of the Results of this Thesis

In the preface to one of his books (reference 9) Rosenbrock observes that space state methods have clarified some questions of structure for automatic control systems, but nevertheless he feels that they have not been able to establish themselves as proven tools for the design of industrial control systems. Part of the reason for this may lie in the complexity both of the off-line design calculation and also of the hardware implementation. The design method derived and evaluated in this thesis removes at least the complexity of the hardware implementation, but at the expense of the off-line design calculation. However to digital computers now in use such off-line calculations are routine and with the advent of graphical display facilities these design procedures become considerably more attractive.

The design method of this thesis (illustrated in Figure 1.1) leads to a fuller appreciation of the structure underlying even the most simple of control systems. For example the proportional control law considered in Chapter 8 (an extreme case with no dynamic elements in the controller) is designed via an estimation of plant state (as in Figure 8.21) and the application of a control law derived from the certainty equivalence principle (perhaps modified as discussed in section 4 above). The uncomplicated scalar gain of the feedback loop thus belies the underlying more complex structure. However this insight into the structure cannot but benefit the designer, and the advantages of the control system design method of this thesis are given below:

(a) The design method provides a viable control system design technique

(b) The technique is applicable to multi-input, multi-output systems

(c) By gauging the effect on the error-squared cost of employing a controller of a given dimension, the lowest dimension controller with adequate performance can be selected.

(d) By setting dimension against cost in this way the technique provides an assessment of "control difficulty" in any situation.

(e) The technique makes available an approach to slowly time-varying systems: the preservation of the state estimate in a varying gain situation is a valuable criterion.

A discussion of the Luenberger type of observer in Chapter 9 leads to the conclusion that the design technique of this thesis gives an estimator which is not of the Luenberger type. The Luenberger observer is shown to be inferior both in performance and, more fundamentally, to be restricted to a higher dimension. A reason for this may be the use made in Chapter 4 of "a priori" information regarding the likely magnitudes and relationships between plant and storage variables. For a closed loop system such information is always available, and it may be the failure of the Luenberger approach to fully make use of it which accounts for its poorer performance.

The design technique of this thesis depends critically on the selection of appropriate information for storage and the method to do this, developed in Chapter 5, has been found to give very satisfactory performance. The criterion, based on the selection of the largest subset of the real positive eigenvalues of a real symmetric positive definite matrix, has the advantage of being straightforward as well as mathematically optimal. The application of the method in Chapter 8 results in storage elements whose respective state estimate weightings (Figures 8.10, 8.11) form a set of smooth curves. These weighting curves illustrate clearly the manner in which the storage elements summarise the available plant information and this gives encouragement that application of the same eigenvalue method to other industrial control problems, where such smooth weighting curves would not necessarily be present, would be successful.

It is a reflection of the central part played by eigenvalue methods in the analysis of linear control systems that eigenvalues (albeit real) should figure so strongly in the design method of this thesis.

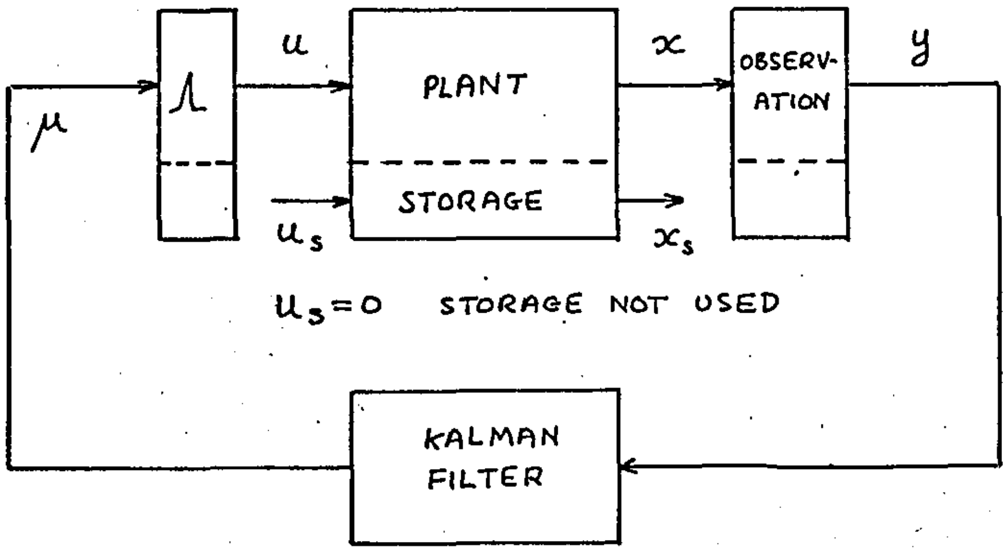


FIGURE 10.1 (a) UNUSED PLANT STORAGE WITH KALMAN FILTER

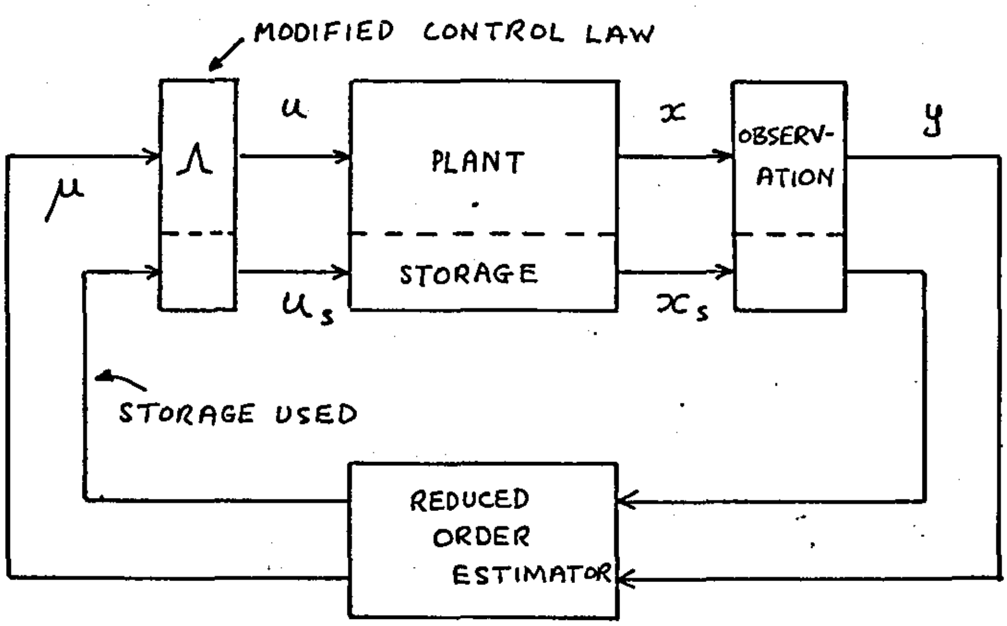


FIGURE 10.1 (b) USE OF PLANT STORAGE WITH REDUCED ORDER ESTIMATOR

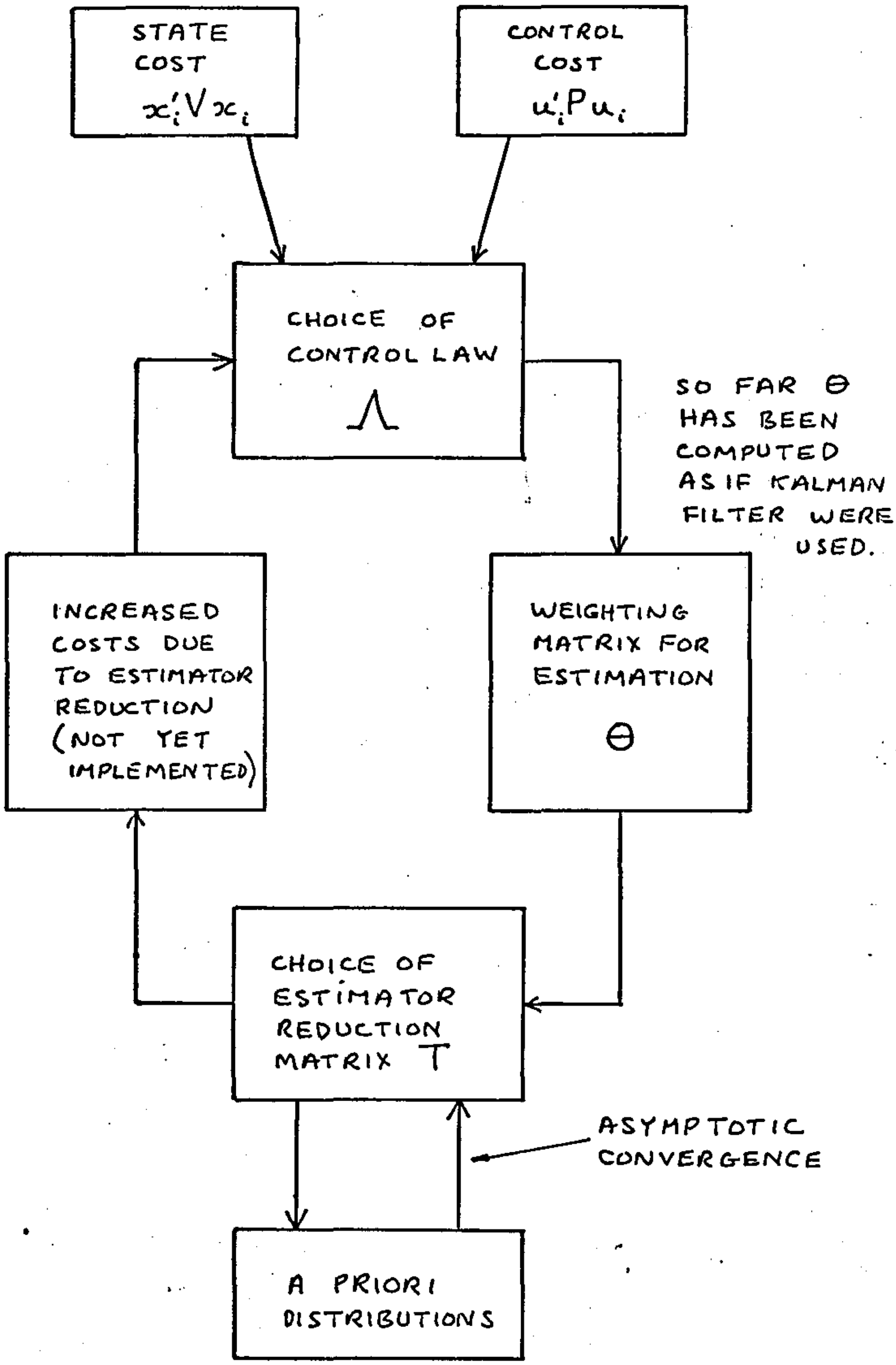


FIGURE 10.2. RELATIONS BETWEEN CONTROL AND ESTIMATION FOR REDUCED ESTIMATOR

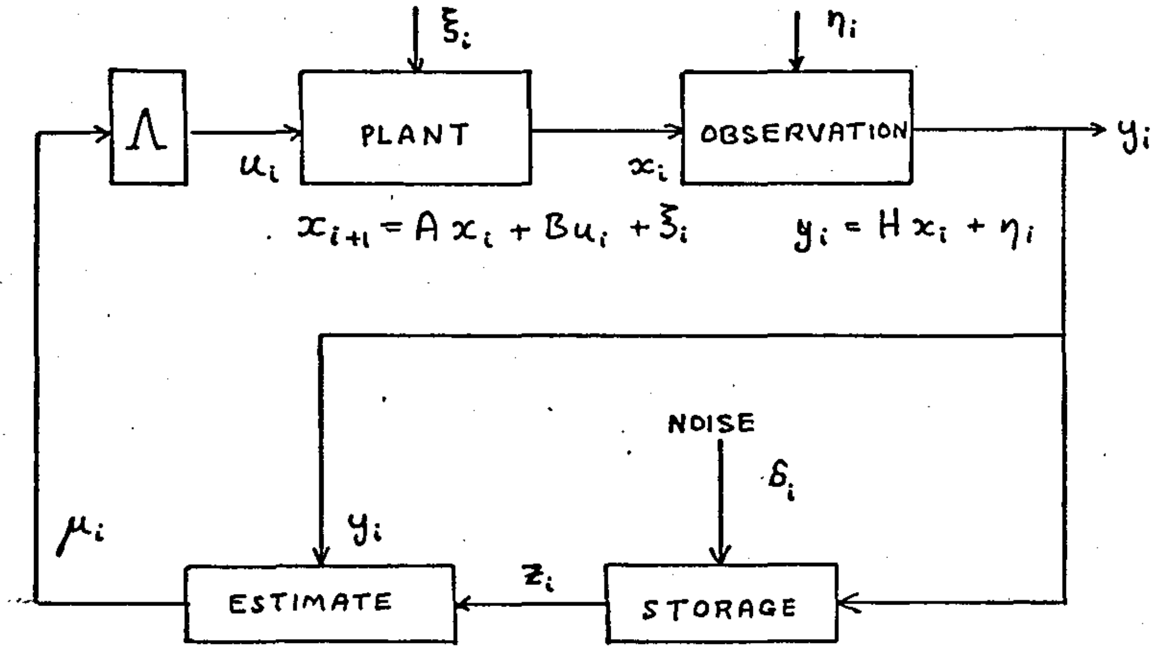


FIGURE 10.3. ESTIMATOR WITH STORAGE NOISE

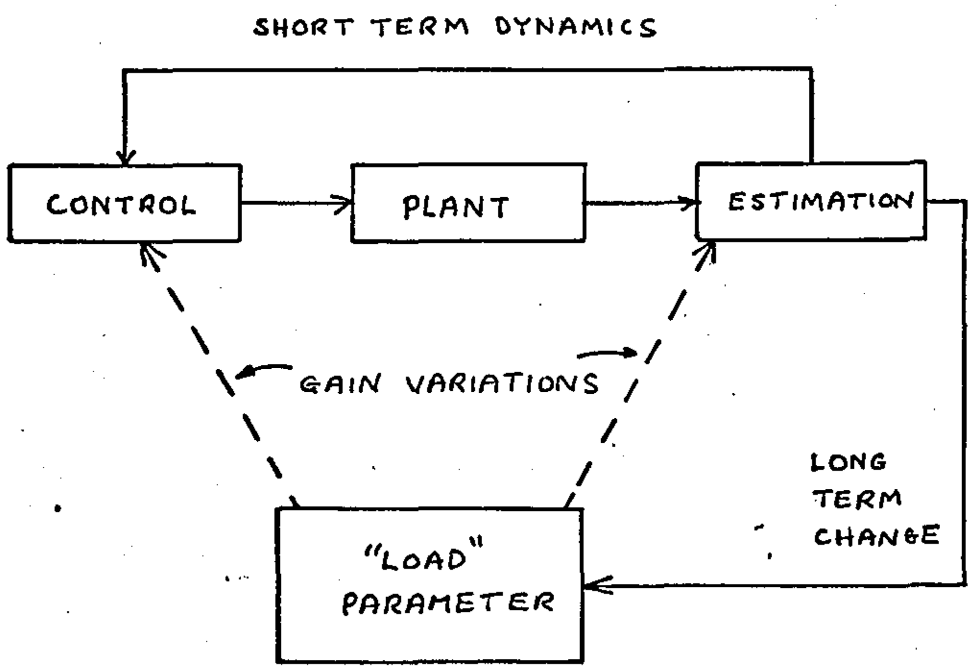


FIGURE 10.4. SIMPLE ADAPTIVE SYSTEM

ACKNOWLEDGMENTS

The author is indebted to Dr W.D. Ray, for encouraging the examination of stochastic systems while the author was with the Group Mathematics Department of Laporte Industries Ltd.

While with the Generation Development and Construction Division of the Central Electricity Generating Board the author has received continuing support. For this, in addition to the opportunity to use the excellent computing facilities, the author is very grateful.

The author would like to thank particularly Professor C. Storey of Loughborough University for giving every possible assistance, and from whom the author has received many useful suggestions.

REFERENCES

1. AOKI, M., "Optimisation of Stochastic Systems", Academic Press, 1967

2. ASTROM, K.J., R.W. KOEPCKE, F. TUNG, "On the Control of Linear Discrete Dynamic Systems with Quadratic Loss", IBM San Jose Research Laboratory Report RJ-222, September 10, 1962

3. KALMAN R.E., "A New Approach to Linear Filtering and Prediction Problems", Trans ASME Ser. D.J. Basic Eng. 82, pp 35-45 (1960)

4. LUENBERGER, D.G., "Observing the State of a Linear System", IEEE Trans Mil Electron., Vol. MIL-8, pp 74-80, April 1964

5. LUENBERGER, D.G., "Observers for Multivariable Systems," IEEE Trans AC, Vol. AC-11, No. 2, pp 190-197, April 1966

6. LUENBERGER, D.G., "An Introduction to Observers" IEEE Trans AC, Vol. AC-16, No.6, pp 596-602, December 1971

7. JACOBS, O.L.R., "Introduction to Control Theory", Clarendon Press, 1974

8. ROSENBROCK, H.H. "State-space and Multivariable Theory", Nelson, 1970

9. ROSENBROCK, H.H. "Computer-Aided Control System Design", Academic Press, 1974.

10. BLOMNES, B., GRUMBACK, R., and LUNDE, J.E. "Experience from the Experimental Implementation of a Modern Control Approach on a Nuclear System". Conference on "Boiler Dynamics and Control in Nuclear Power Stations", London, 1973. British Nuclear Energy Society

11. HEBRIK, R., and JAMSHIDI, M., "Design of an Optimum State Regulator for a Once-Through Boiler", Proc. IFAC, 5th World Congress Paris 12-17 June 1972.

12. DAVISON, E.J., "A Method for Simplifying Linear Dynamic Systems," IEEE Trans AC, Vol. AC-11, No. 1, pp 93-101, January 1966

13. CROSSLEY, T.R., and PORTER, B., "Modal Theory of State Observers", Proc. IEE, Vol 118, No. 12, December 1971

14. WILSON, D.A., "Optimum Solution of the Model-Reduction Problem", Proc IEE, Vol. 117 No. 6, pp 1161-1165, June 1970

15. MITRA, D., "On the Reduction of Complexity of Linear Dynamical Models", UKAEA Report AEEW-R520, 1967

16. MITRA, D., "On the Reduction of Complexity of Linear Dynamical Models - Results of Computer Studies," UKAEA Report AEEW-R535, 1967

17. HICKIN J., and SINHA, N.K., "New Method of Obtainong Reduced Order Models for Linear Multivariable Systems" Elect. Lett. 1976, 12, pp 90-92.

18. RAESIDE, D.E. Bayesian Statistics : Mathematical Tools for Computers", Computer, July 1975, pp 65-73

19. FEL'DBAUM, A.A., "Dual-Control Theory, Parts I-IV", translated from Avtomatika i Telemekhanika Vol.21, No.9, pp 1240-1249, September 1960; Vol.21, No.11, pp 1453-1464, November 1960; Vol.22, No.1, pp 3-16, January 1961; Vol.22, No.2, pp 129-142, February 1961.

20. BEREZNAI G.T. and SINHA, N.K., "Computer-Aided Design of Suboptimal Feedback Controllers using Optimal Low Order Models", Elect. Lett. Vol.7, No. 15, pp 440-442, 29 July 1971.

21. JAMESON, A. and ROTHSCHILD, D., "A Direct Approach to the Design of Asymtotically Optimal Controllers", Int. J. Control Vol. 13, No.6 pp 1041-1050.

22. KURTARAN, B.Z., "Suboptimal Control for Discrete Linear Constant Stochastic Systems," IEEE Trans AC, Vol. AC-21, No.3, pp 423-425, 1971.

23. BERGER, C.S., "An Algorithm for Designing Suboptimal Dynamic Controllers", IEEE Trans AC, Vol. AC-19, No. 5, pp 596-7, October 1974.

24. BORG, D.A. and Giles, R.F., "PID : Is it an Optimal Regulator?" Control Engineering, September 1975, pp 59-60.

25. IBM System/360 Matrix Language (MATLAN) Program Description Manual.

26. BARNETT, S., "Matrices in Control Theory", Van Nostrand, 1971

27. POTTER, J.E., "Matrix Quadratic Solutions", SIAM J. Appl. Math., Vol. 14, pp 496-501, 1966

28. O'DONNELL, J.J. "Asymptotic Solution of the Matrix Riccati Equation of Optimal Control" Proc. 4th Annual Allerton Conference on Circuit and System Theory, University of Illinois, Urband, pp 577-586, 1966.

29. REPPERGER, D.W., "A Square Root of a Matrix Approach to Obtain the Solution to a Steady State Matrix Riccati Equation," IEEE Trans AC, Vol. AC-21, No.5, pp 786-787, October 1976.

30. BELLMAN, R.E., "Dynamic Programming", Princeton : University Press, 1957

31. WITSENHAUSEN, H.S., "A Counterexample in Stochastic Optimal Control" SIAM J. Control, Vol. 6, No.1, 1968.

32. BOX, G.E. and JENKINS, G.M., "Time Series Analysis for Forecasting and Control", Holden and Day, 1970.

33. BARNETT, S., "Introduction to Mathematical Control Theory", Clarendon Press, 1975

34. BARNETT, S., and STOREY, C., "Matrix Methods in Stability Theory", Nelson, 1970

35. MIRKSY, L., "An Introduction to Linear Algebra, "Oxford University Press, 1955

36. ROSENBROCK, H.H., and STOREY, C., "Computational Techniques for Chemical Engineers, "Pergamon Press, 1966.

37. GANNON, D.R., "Constants for the Numerical Inversion of Laplace Transforms, "Ph.D Thesis, UMIST 1972

38. BRERETON, D.R., Central Electricity Research Laboratory, Leatherhead. Private Communication 1975.

39. EMPETT, B., Computing Bureau, CEGB, 85 Park Street, London SE1 9DY. Private Communication, 1971

40. AOKI, M., and HUDDLE, J.R., "Estimation of the State Vector of a Linear Stochastic System with a Constrained Estimator". IEEE Trans AC, Vol. AC-12, pp 432-433, August 1967.

41. IGLEHART, S.C., and LEONDES, C.T., "A Design Procedure for Intermediate - Order Observer - Estimators for Linear Discrete - Time Dynamical Systems", Int. J. Control, Vol.16, No.3 pp 401-415, 1972.

42. TSE, E., and ATHANS, M., "Optimal Minimal - Order Observer - Estimators for Discrete Linear Time-Varying Systems," IEEE Trans AC, Vol. AC-15, No.4, pp 416-426, August 1970.

43. TSE, E., "Observer - Estimators for Discrete Time Systems," IEEE Trans AC, Vol. AC-18, No.1, pp 10-16, February 1973.

44. LEONDES, C.T., and NOVAK, L.M., "Reduced - Order Observers for Linear Discrete - Time Systems", IEEE Trans AC, Vol. AC-19, No. 1, pp 42-46, February 1974.

45. NEWMANN, M.M., "Optimal and Suboptimal Control using an Observer when some of the State Variables are Not Measurable," Int. J. Control, Vol.9, No.3, pp 281-289, 1969.

46. O'REILLY Y.J., and NEWMANN, M.M., "On the Design of Discrete-Time Optimal Dynamical Controllers using a Minimal - Order Observer," Int. J. Control, Vol.23, No.2, pp 257-275, Gebruary 1976.

47. YOSHIKAWA, T., "Minimal - Order Optimal Filter for Discrete - Time Linear Stochastic Systems," Int. J. Control Vol.21 No.1, pp 1-19, January 1975.

48. AKAIKE, H., "Stochastic Theory of Minimal Realisations", IEEE Trans AC, Vol. AC-19, No. 6, pp 667-674, December 1974

49. ASHER, R.B., and DURRETT, J.C., "Linear Discrete Stochastic Control with a Reduced - Order Dynamic Compensator," IEEE Trans AC, Vol. AC-21, No.4, pp 626-627, August 1976.

50. AKASHI, H., and NOSE, K., "On the Certainty Equivalence of Stochastic Optimal Control Problem", Int. J. Control, Vol. 21, No.5, pp 857-863, 1975.

51. NEWMANN, M.N., "A Continuous-Time Reduced-Order Filter for Estimating the State Vector of a Linear Stochastic System", Int. J. Control, 1970, Vol. 11, No. 2, pp 229-239.

APPENDIX 1

A Summary of MATLAN Commands

A full description of the command specification is given in reference 24 but a brief account is given here, only those commands which have appeared in program listings being included.

1. ADD X1, X2, Y

   Performs $Y = X1 + X2$

2. SUB X1, X2, Y

   Performs $Y = X1 - X2$

3. MULT X1, X2, Y

   Performs $Y = X1.X2$

4. DIV X1, X2, Y

   Performs $Y = X1^{-1}X2$

Note: If in any of the above operations square matrices are being used, a scalar may be used for X1 or X2. It will first be multiplied by the unit matrix before the operation is carried out.

5. INV X, Y

   Performs $Y = X^{-1}$

6. ROWSUM X,Y

   Adds the elements forming the rows of X to form the column vector Y.

7. COLSUM X, Y

   Adds the elements forming the columns of X to form the row vector Y.

8. EMULT X1, X2, Y

Each element of X1 is multiplied by a corresponding element of X2 to form the elements of Y. (Used in PROD, Listing 5.4).

9. COPY X, Y

Performs Y = X

10. TRANS X, Y

Performs the transposition of X to form Y, i.e. Y = X'

11. EXSUBM X, (rbeg, cbeg), (rdim, cdim), Y

Allows the partitioning of matrix X and the extraction of submatrix Y, defined by the other parameters.

12. INSUBM X, Y, (rbeg, cbeg)

Allows the construction of the larger matrix Y by the insertion of a sub-matrix X, starting at the given element.

In addition the command INSUBM x, Y, (rbeg, cbeg), (1,1), N allows the scalar element x to be inserted in the matrix Y as a band of N elements starting at a given location. Used in OTBOIL (Listing 7.1).

13. FORMS Y, (rdim, cdim), (rbeg, cbeg), (1,1), repet, val

Constructs a band matrix of dimension rdim x cdim containing zeros and the elements "val". The band starts at (rbeg, cbeg) and extends diagonally with repet elements. Used in OTBOIL (Listing 7.1)

14. NULLMAT Y, (rdim, cdim)

Generates a matrix of given dimensions with zero elements.

15. RDIM X, N

N is set equal to the number of rows of matrix X.

16. CDIM X, N

N is set equal to the number of columns of matrix X.

17. CANCEL X1, X2 ...

Saves storage space if matrices are no longer required.

18. WRITE (X1, X2, ...), FORMAT = 8

Prints matrices or scalars in floating point form with eight significant figures.

19. READ (X1, X2, ...)

Matrices will be read from cards. Generally the card format used has been as below, although other formats are available.

A	12	12	Matrix with dimensions
1.624	0.541	.....	Matrix elements in free format
0.761	....		
⋮			
END			

The subroutines of Appendix 3, were written to allow matrices to be punched on to cards, and the format used for each element was E20.10, with four elements per card.

20. LOOP L1, J, I, N

L1 LOOPEND

Performs a loop for J = I to N

21. SUBPRO Name, (X1, X2, ...)

Specifies a MATLAN subroutine with dummy arguments. Subroutine to finish with RETURN and END.

22. CALL Name, (X1, X2, ...)

Call of above subroutine.

23. CALL Name, (X1, X2, ...), F

The addition of the "F" indicates that the subroutine to be called is a Fortran subroutine. The conversion of the matrix arguments is handled automatically by MATLAN, provided the appropriate job control statements have been supplied as in Appendix 2.

APPENDIX 2

Fortran Subroutines Called by MATLAN

In chapter 6 it was necessary to obtain the eigenvalues and eigenvectors of a real symmetric matrix, and this was achieved by calling the Fortran subroutine EIGEN which is given in listing A2.1. As a description of this subroutine, which forms part of the IBM Scientific Subroutine Package, is given in the comment statements at its head, no further description is given here.

Storage of matrix elements differs between MATLAN and Fortran, and further the Scientific Subroutine package has its own convention for storing elements.

The various methods are as follows:

(a) MATLAN

Matrices are stored by rows in two ways. The first stores all elements (A form) while the second stores only non-zero elements and their co-ordinates (C form). The more economic storage mode is selected internally by MATLAN.

(b) Fortran

A matrix is stored in an array column by column.

(c) IBM Scientific Subroutines

A choice of three storage methods are available, but subroutine

EIGEN assumes that the symmetric matrix whose eigenvalues and eigenvectors are required is stored in storage mode 1. This is a specific mode for symmetric matrices since only the upper triangle is stored, the lower triangle being inferred by symmetry. Storage is then by columns to form a vector of large dimension.

To make the various storage modes compatible requires some detailed programming. Conversion between MATLAN and Fortran is handled automatically with the argument matrices being transposed to allow for the change from row to column storage, and vice versa.

Conversion from Fortran to the IBM Scientific Subroutine Modes is handled by means of the subroutines LOC and MSTR which also form part of the Scientific Subroutine Package. Listing A2.2 shows subroutine LOC, while a call of LOC appears in EIGMAT (Listing 6.4) as a special case of

CALL LOC (I, J, IC, N, M, 1).

This generates the integer IC which indicates the location in the storage vector of the element (I, J) of a N x M matrix when stored in mode 1, i.e. upper triangle only. The matrix element may then be referenced, for example, as A(IC).

The subroutine MSTR is shown in Listing A2.3. This subroutine performs storage conversion for a square matrix. For example, the call of MSTR in UNIMAT (Listing 6.3) is

CALL MSTR (AA, A, N, O, 1).

The vectors AA and A store N x N matrix elements and AA, with storage mode O (general column by column storage into vector) is to be converted into A, with storage mode 1. The subroutine MSTR itself calls LOC to carry out this conversion. Thus within UNIMAT a further conversion is required between the normal Fortran matrix storage and the vector "mode O" storage and this is accomplished by means of a "DO" loop where necessary.

In calling Fortran subroutines it is necessary to place these in a special position in the deck of cards forming a MATLAN job, in order to avoid confusion with MATLAN subroutines and to allow proper compilation.

Listing A2.4 shows the necessary job control cards and the positions of MATLAN, Fortran, and data cards.

This deck was used for all MATLAN work, the Fortran subroutines being omitted when not reouired. The "TIME"parameter causes the job to be cancelled if this length of computing time is exceeded. This was normally set to 30 seconds as a typical run (e.g. chapter 8, cases B to I) occupied around 20 seconds on the CEGB IBM 370 computer.

The following listings have been retained by the author, as they form part of the IBM Scientific Subroutine Package, which is subject to an IBM copyright.

Listing A2.1 Fortran Subroutine Eigen

Finds the eigenvalues and eigenvectors (all real) of a real symmetric matrix.

Listing A2.2 Subroutine LOC

Listing A2.3 Subroutine MSTR

These Fortran subroutines allow the conversion from one matrix storage mode to another.

Further information on the above subroutines can be obtained from the author.

```
//   MSGLEVEL=(1,1),
//      PRTY=01,REGION=210K,TIME=(0,30)
//*MAIN ORG=GEND,CARDS=500,LINES=(3,C),CLASS=ANY,IORATE=LOW
//*PROCESS RICONTL
//*PROCESS MAIN
//*PROCESS PRINT
//*FORMAT PR,DDNAME=SYSMSG,DEST=GEND
//*FORMAT PR,DDNAME=FT03F001,DEST=GEND
//*FORMAT PR,DDNAME=*ACCOUNT,DEST=GEND
//*PROCESS PUNCH
//*FORMAT PU,DDNAME=FT08F001,DEST=LOCAL
//*ENDPROCESS
//CLGL EXEC MATFORT,PARM.G='DCALNG=10'
//M.STEPLIB DD UNIT=3330,VOL=SER=PP,DISP=SHR,DSN=PP.MATLAN
//M.SYSIN DD *
/*
//F.SYSIN DD *
/*
//L.SYSLMOD DD DISP=(NEW,PASS)
/*
//G.FT08F001 DD UNIT=(CTC,,DEFER),DCB=(RECFM=F,BLKSIZE=80)
//G.MTLDAF DD UNIT=UT,SPACE=(TRK,(50)),DCB=BLKSIZE=384
//G.STEPLIB DD UNIT=3330,VOL=SER=PP,DISP=SHR,DSN=PP.MATLAN
//G.DATA DD *
/*
```

MATLAN Main Program and MATLAN Subprograms

Fortran Subroutines

MATLAN data cards to be read in

Listing A2.4 MATLAN Job Control Statements

The above card deck executes a MATLAN job, reads in data on cards, calls the Fortran subroutines and punches cards if required.

APPENDIX 3

Subroutines to Punch Matrices on to Cards

MATLAN will read matrices from cards where the first card gives the matrix name and its dimensions and the remaining cards contain the elements, row by row. For the large matrices being used hand punching was not practical so Fortran subroutines were written to punch suitable cards when the matrix to be punched is passed to the subroutine as an argument. Two basic subroutines were written and are shown in Listings A3.1 and A3.2.

1. APUNCH

   A square matrix is punched on to cards, row by row. E.g. system matrix A.

2. BPUNCH

   A vector is punched on to cards. E.g. control matrix B.

Because of the dimensions used in the examples no other subroutines were required for punching MATLAN matrices.

For punching matrices which could be read by the eigenvalue program GEDES a Fortran subroutine GEDPMN was written which is shown in Listing A3.3. Although normally used for punching square matrices, non-square matrices can be punched if required. The elements are punched row by row together with their co-ordinates, with four elements per card. Before GEDPMN was written a simpler subroutine GEDPCH was written specifically for use with square 12 x 12 matrices. This is shown in Listing A3.4.

```
      SUBROUTINE APUNCH(A)
      DIMENSION A(12,12)
      DO 12 I=1,12
      N=1
   13 NP=N+3
      WRITE(8,2)  (A(I,J),J=N,NP)
      N=N+4
      IF(N.EQ.13) GO TO 12
      GO TO 13
   12 CONTINUE
    2 FORMAT (4(E20.10))
      RETURN
      END
```

```
      SUBROUTINE BPUNCH(B)
      DIMENSION B(12)
      N=1
   13 NP=N+3
      WRITE(8,2)  (B(I),I=N,NP)
      N=N+4
      IF(N.EQ.13) GO TO 12
      GO TO 13
   12 CONTINUE
    2 FORMAT (4(E20.10))
      RETURN
      END
```

<u>Listing A3.1 and A3.2</u> <u>Fortran Subroutines APUNCH and BPUNCH</u>

These subroutines punch matrices on to cards in a format which can be read by the MATLAN read commands.

```
      SUBROUTINE GEDPMN(A)
      DIMENSION A(14,14)
      M=14
      N=M
      I=1
      J=1
    5 CONTINUE
      IF(J.GT.N) GOTO 10
   15 CONTINUE
      I1=I
      J1=J
      A1=A(I,J)
      IF((J.EQ.N).AND.(I.EQ.M)) GOTO 400
      J=J+1
      IF(J.GT.N) GOTO 20
   25 CONTINUE
      I2=I
      J2=J
      A2=A(I,J)
      IF((J.EQ.N).AND.(I.EQ.M)) GOTO 300
      J=J+1
      IF(J.GT.N) GOTO 30
   35 CONTINUE
      I3=I
      J3=J
      A3=A(I,J)
      IF((J.EQ.N).AND.(I.EQ.M)) GOTO 200
      J=J+1
      IF(J.GT.N) GOTO 40
   45 CONTINUE
      I4=I
      J4=J
      A4=A(I,J)
  100 WRITE(8,1) (I1,J1,A1,I2,J2,A2,I3,J3,A3,I4,J4,A4)
      IF((J.EQ.N).AND.(I.EQ.M)) GOTO 500
      J=J+1
      GOTO 5
   10 J=1
      I=I+1
      GOTO 15
   20 J=1
      I=I+1
      GOTO 25
   30 J=1
      I=I+1
      GOTO 35
   40 J=1
      I=I+1
      GOTO 45
  400 WRITE(8,4) (I1,J1,A1)
      GOTO 500
  300 WRITE(8,3) (I1,J1,A1,I2,J2,A2)
      GOTO 500
  200 WRITE(8,2) (I1,J1,A1,I2,J2,A2,I3,J3,A3)
    1 FORMAT (4(I3,I3,1PE14.6))
    2 FORMAT (3(I3,I3,1PE14.6))
    3 FORMAT (2(I3,I3,1PE14.6))
    4 FORMAT   (I3,I3,1PE14.6)
  500 RETURN
      END
```

<u>Listing A3.3</u>   <u>Fortran Subroutine GEDPMN</u>

An arbitrary matrix of dimension M x N is punched on to cards according to a particular format to allow these to be input into an eigenvalue program.

```
      SUBROUTINE GEDPCH(A)
      DIMENSION A(12,12)
      DO 12 I=1,12
      N=1
   13 NP=N+3
      WRITE(8,1) (I,J,A(I,J),J=N,NP)
      N=N+4
      IF(N.EQ.13) GO TO 12
      GO TO 13
   12 CONTINUE
    1 FORMAT (4(I3,I3,1PE14.6))
      RETURN
      END
```

<u>Listing A3.4</u> <u>Fortran Subroutine GEDPCH</u>

A less general subroutine than GEDPMN. Punches square 12 x 12 matrices on to cards, with four elements and element co-ordinates per card.

INDEX TO PROGRAM LISTINGS