AFOSR-TR- 791

(2)

AD-A160 187

ISCT SYSTEMS CONTROL TECHNOLOGY, INC.

1801 PAGE MILL RD. □ P.O. BOX 10180 □ PALO ALTO, CALIFORNIA 94303 □ (415) 494-2233

eport 5400-03

April 1985

ADAPTIVE DECENTRALIZED CONTROL

FINAL REPORT

Prepared by

B. Friedlander
Systems Control Technology, Inc.
1801 Page Mill Road
Palo Alto, CA 94304

Prepared for:

Air Force Office of Scientific Research
Bolling AFB, Washington, D.C. 20332
System Science Program
Directorate of Mathematical
and Information Sciences

Under Contract: F4920-81-C-0051

2

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: AFOSR-TR- 85-0791
2. GOVT ACCESSION NO.: AD-A160 187
3. RECIPIENT'S CATALOG NUMBER:
4. TITLE (and Subtitle): Adaptive Decentralized Control
5. TYPE OF REPORT & PERIOD COVERED: FINAL REPORT, June 81-July 84
6. PERFORMING ORG. REPORT NUMBER:
7. AUTHOR(s): Benjamin Friedlander
8. CONTRACT OR GRANT NUMBER(s): F4920-81-C-0051
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Systems Control Technology, Inc. 1801 Page Mill Road Palo Alto, California 94304
10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 61102F 2304/A6
11. CONTROLLING OFFICE NAME AND ADDRESS: Air Force Office of Scientific Research Bolling Air Force Base Washington, DC 20332
12. REPORT DATE: April 1985
13. NUMBER OF PAGES:
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office):
15. SECURITY CLASS. (of this report): UNCLASSIFIED
15a. DECLASSIFICATION/DOWNGRADING SCHEDULE:
16. DISTRIBUTION STATEMENT (of this Report): Approved for public release; distribution unlimited.
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report):

DTIC ELECTE OCT 11 1985 S D A

18. SUPPLEMENTARY NOTES:
19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Decentralized Control
Adaptive Control
Large Scale Systems
Robustness

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This Final Report summarizes the results of a research effort directed towards the development of adaptive decentralized control systems. The adaptive controller in such a system must operate in the presence of unmodeled dynamics. An input-output approach was developed for analyzing the global stability and robustness properties of adaptive controllers under such circumstances. Conditions for guaranteeing global stability of the error system associated with the adaptive controller, and ensuring boundedness of the adaptive gains, were derived.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

I. INTRODUCTION AND SUMMARY

This final report summarizes work performed on the Adaptive Decentralized Control project (under contract F4920-81-C-0051) during the period June 1981 - July 1984. The objective of this research effort was the development of a new concept for the design of decentralized controllers for large scale systems.

The modeling, analysis and control of large-scale systems is an increasingly important problem in such diverse areas as defense systems, communication and computer networks and transportation systems. The size and complexity of many systems make it difficult or impractical to use centralized control structures. Furthermore, considerations of communication costs, system reliability, computational requirements and response time provide strong incentives for the use of distributed control architectures. The basic focus of our research is on a framework within which decentralized controller structures can be analyzed and developed. The motivation for our proposed approach which we named ADCON (for Adaptive Decentralized CONtrol) comes from the following observations about the current status of control theory.

An important aspect of centralized control has been the study of systems with unknown or uncertain (time varying, random) parameters. The investigation of this problem led to an extensive literature on adaptive control (also called: learning or self-organizing systems). The natural progression in developing centralized controllers was from the non-adaptive case to the more difficult problems addressed by adaptive techniques.

The study of decentralized control seems so far to be almost exclusively devoted to non-adaptive techniques. A possible explanation of this state of affairs is the fact that the area of decentralized control of completely known systems still has many unresolved issues and some basic problems are yet to be answered. Under these conditions, there seemed to be little incentive to tackle the more complex adaptive case which deals with partially known systems. However, this line of thinking is based on the experience gained in centralized control and it may be inapplicable in the context of the decentralized problem, which has radically different characteristics. In fact, adaptive techniques have a central role in decentralized control, which is of a somewhat different nature than the role they play in the centralized problem.

To understand the interrelation between adaptive and decentralized control, we have to re-examine the basic issues underlying the need for decentralized control strategies. The main motivation for considering such strategies arises in the context of complex, large-scale systems where a centralized controller usually requires excessive computational requirements and excessive information gathering networks to make such a controller feasible. In such a system, it is reasonable to assume that the local controller (i.e., the controller of one subsystem in the large system) has only partial information about the rest of the system. Even if the structure of the whole system (i.e., the state equations of all subsystems and their interactions) can be made available to each local controller, the sheer complexity of the problem often limits the usefulness of this information. In fact, attempting to use too much information may be one of the principal stumbling blocks of conventional approaches to decentralized control. Most of these approaches try to solve the (optimal) centralized problem, and then to find clever ways of decentralizing the solution. The shortcomings of this technique and the need for a different point of view are by now widely recognized.

The basic idea underlying our approach is to assume that from the subsystem's point of view, the rest of the system is not exactly known. Thus, the subsystem is aware of its own structure, but it has only an approximate knowledge of the rest of the system, for example, in the form of a reduced order model. (Different subsystems will use different models of the "outside world".) The local controller is then designed on the basis of this partial information. The modeling uncertainty inherent in this procedure makes it necessary to consider robust or adaptive control structures. Note that the uncertainty here is due to the complexity of the system rather than to lack of knowledge or to random effects, which are the traditional sources of uncertainty in centralized control. The idea of replacing a complex deterministic problem by a simple stochastic model is by no means new, and has been used in a variety of physical problems (e.g., statistical thermodynamics).

The use of reduced order models and partial information greatly simplifies the design and implementation of the decentralized controllers. It raises, however, many difficult questions regarding the conditions under which such a scheme will lead to satisfactory system behavior. What is needed is a theory for the control of interconnected subsystems in the presence of model uncertainties. In an earlier report [12] and in some related papers we made a preliminary study of some of these issues.

An even more difficult set of questions arises with regard to the operation of adaptive controllers in the presence of uncertainty. Currently available adaptive control algorithms have been shown to experience severe difficulties in the presence of unmodeled plant dynamics. We were able to derive conditions which guarantee that the adaptive controller will have specified performance despite plant uncertainty and unmodeled dynamics. These conditions provide guidelines for the analysis and design of robust adaptive controllers. A combination of results from robust control and adaptive control theory was used to prove the main theorem. The main theorem was applied to a number of well-known adaptive structures: the direct adaptive controller, an adaptive observer, the indirect adaptive controller, and a general form of the model reference adaptive controller [4]. We believe that this work represents a significant advance in the field of adaptive control.

In [13] we presented an input-output approach for analyzing the global stability and robustness properties of adaptive controllers to unmodeled dynamics. The concept of a tuned system was introduced, i.e., the control system that could be obtained if the plant were known. Comparing the adaptive system with the tuned system results in the development of a generic adaptive error system. Passivity theory was used to derive conditions which guarantee global stability of the error system associated with the adaptive controller, and ensure boundedness of the adaptive gains. Specific bounds are presented for certain significant signals in the control systems. Limitations of these global results are discussed, particularly the requirement that a certain operator be strictly positive real (SPR) -- a condition that is unlikely to hold due to unmodeled dynamics.

The ADCON concept involves many different issues, as can be seen from the earlier discussion and from [4],[9],[12],[13]. So far we have addressed the problem of designing a controller for a single subsystem, when the rest of the system is fixed. This represents only one step in an iterative procedure in which each subsystem performs its own controller design. We have done some investigation extensions of the theory of robust control and adaptive control to the case of interconnected subsystems, in which local controllers are designed sequentially (iteratively) or simultaneously. A number of different information structures were considered. It seems that by providing each subsystem with structural information in addition to an aggregate (reduced order) model of the rest of the systems, it is possible to obtain simpler design schemes. However, no conclusive results are available at this time.

We have also investigated the application of lattice structures to the adaptive control problem. Our work in this area seemed to have generated a considerable amount of interest (cf. [R1]-[R6]). This class of algorithms is especially well suited for large scale problems of the type considered in this project.

In the next section we list the publications prepared under this contract. The key papers are enclosed in the appedices.

2. PROJECT PUBLICATIONS

Journals

1. B. Friedlander, "Lattice Implementation of Some Recursive Parameter Estimation Algorithms;" INT. J. Control, Vol. 37, No. 4, pp. 661-684, 1983.

2. B. Porat and B. Friedlander, "An Efficient Technique for Output Error Model Reduction;" Int. J. Control, Vol. 39, No.1, pp. 95-113, 1984.

3. B. Porat and B. Friedlander, "An Output Error Method for Design of Reduced Order Controllers;" IEEE Trans. Automatic Control, Vol. AC-29, No. 7, pp 629-631, July 1984.

4. R. L. Kosut and B. Friedlander, "Robust Adaptive Control: Conditions for Gloal Stability;" IEEE Trans. Automatic Control, July 1985, to appear.

5. B. Friedlander, "Decentralized Design of Decentralized Controllers," in Advances in Control and Dynamic Systems, Vol. xx, C.T. Leondes, ed.,to appear 1985.

Conferences

6. B. Friedlander, "Lattice Implementation of Some Recursive Parameter Estimation Algorithms," Proc. 6th IFAC Symposium on Identification and System Parameter Estimation, Washington,D.C., June 1982.

7. B. Porat and B. Friedlander, "An Output Error Technique for Design of Reduced Order Controllers," 16th Asilomar Conf. Circuits, Systems and Computers, November 1982.

8. B. Porat and B. Friedlander, "An Efficient Technique for Output Error Model Reduction," 21st IEEE Conference on Decision and Control, December 1982.

9. B. Friedlander and B. Porat, "Adaptive Design of Decentralized Controllers," Proc. 21st IEEE Conference on Decision and Control, December 1982.

10. R.L. Kosut and B. Friedlander, "Performance Properties of Adaptive Control Systems," 21st IEEE Conference on Decision and Control, pp 18-23, Orlando, Florida, December 1982.

11. R.L. Kosut, "Analysis of Performance Robustness for Uncertain Multivariable Systems," Proc. 21-st IEEE Conference on Decision and Control, December 1982.

Reports

12. B. Friedlander, "Adaptive Decentralized Control - Annual Report, September 1982, Systems Control Technology, Inc., Report No. 5400-01.

13. B. Friedlander, Adaptive Decentralized Control - Annual Report, April 1984, Systems Control Technology, Inc., Report No. 5400-02.

REFERENCES

[R1] S.C. Shah, "Adaptive Control and Prediction Using Lattice Structures," Technical Memo 5017-1, October 21, 1981, Integrated Systems, Inc.

[R2] N. Sundarajan and R.C. Montgomery, "Decoupling the Structural Modes Estimated Using Recursive Lattice Filters," IEEE Conference on Decision and Control, December 1982.

[R3] N. Sundarajan and R.C. Montegomery, "Adaptive Identification for the Dynamics of Large Space Structures," AIAA Guidance and Control Conference, San Deigo, California, August 1982.

[R4] S. Olcer and M. Morf, "Adaptive Control by Ladder Forms," submitted for publication.

[R5] D.M. Wiberg, "Frequencies of Vibration Estiated by Lattices," submitted for publication.

[R6] N. Sundarajan and R.C. Montgomery, "Experiments Using Least Squares Lattice Filters for the Identification of Structural Dynamics, submitted for publication.

APPENDIX A

ROBUST ADAPTIVE CONTROL: CONDITIONS FOR GLOBAL STABILITY

ROBUST ADAPTIVE CONTROL: CONDITIONS FOR GLOBAL STABILITY

by

Robert L. Kosut
Integrated Systems, Inc.
151 University Ave
Palo Alto, CA 94301

Benjamin Friedlander
Systems Control Technology, Inc.
1801 Page Mill Road
Palo Alto, CA 94304

ABSTRACT

An input-output approach is presented for analyzing the global stability and robustness properties of adaptive controllers to unmodeled dynamics. The concept of a tuned system is introduced, i.e., the control system that could be obtained if the plant were known. Comparing the adaptive system with the tuned system results in the development of a generic adaptive error system. Passivity theory is used to derive conditions which guarantee global stability of the error system associated with the adaptive controller, and ensure boundedness of the adaptive gains. Specific bounds are presented for certain significant signals in the control systems. Limitations of these global results are discussed, particularly the requirement that a certain operator be strictly positive real (SPR) -- a condition that is unlikely to hold due to unmodeled dynamics.

This work was supported by the Air Force Office of Scientific Research (AFSC), under contract F4920-81-C-0051. The United States government is authorized to reproduce and distribute reprint for governmental purposes notwithstanding any copyright notation thereon.

1. INTRODUCTION

1.1 Background

The analysis and design of adaptive control systems has been the subject of extensive research in the past two decades [1]-[10]. Adaptive techniques provide a way of handling plant uncertainty by adjusting the controller parameters on-line to optimize system performance. An alternative method for handling uncertainty is to use a fixed structure controller designed to provide acceptable performance for a specified range of plant behavior. In principle, adaptive controllers can provide improved performance compared to fixed robust controllers, since they are tuned to the uncertain plant. However, adaptive controllers sometimes exhibit undesirable behavior during the tuning or adaptation process. For example, unmodeled dynamics can cause a rapid deterioration in performance and even instability [11],[12]. This problem is not resolved by increasing the order or complexity of the model. Since the model of any dynamic system, by definition, is not the actual system, it can therefore be argued that unmodeled dynamics are always present, ad infinitum.

The main reason for these difficulties with adaptive controllers seems to be that robustness to unmodeled dynamics was not considered as a design criterion in the development of the adaptive control algorithm. The design objective is global stability of the closed-loop system, e.g., [7], [9] and various assumptions on the structure of the plant are required to achieve that objective. In particular, it is necessary to assume that the plant is linear and time invariant (LTI), that the relative degree of the transfer function is known as well as the sign of the high frequency gain. Such requirements are not practical since real plants are often nonlinear and time-varying and can be accurately represented only by high order (sometimes infinite order [13]) complicated models.

The need for robustness to plant uncertainty is not unique to adaptive control. The problem of robustness is ubiquitous in control theory and has been studied in the context of fixed (nonadaptive) control [14]-[17]. These studies rely on the input/output properties of systems, e.g., [18],[19]. The predominant reason to examine robustness issues in this way is that the characteritics of unmodeled dynamics, such as uncertain model order, are easily represented. Lyapunov theory, on the other hand, is not well suited for this type of uncertainty. Typically, plant uncertainty is characterized by assuming that the plant belongs to a well defined set. For example, a set description of an uncertain LTI plant is to define a "ball" in the frequency domain. The center of the ball is the nominal plant model, and the radius defines the model error. This set model description is one type of a more general set description, referred to as a conic-sector [15]. The uncertainty in the plant induces an uncertainty in the input/output map of the closed-loop system which can, again be characterized by a conic sector. Performance requirements for the control system can be translated into statements on the conic sector which bounds the closed-loop systems, making it possible to check whether a given design meets specifications, and providing guidelines for robust controller design.

In this paper we use the input/output approach to analyze the global stability and robustness properties of continuous-time adaptive controllers with respect to unmodeled dynamics (although we consider only continuous-time algorithms, the input-output formalism can be readily extended to the discrete-time case). By global we mean that no specific magnitude constraint (other than boundedness) is placed on any of the external inputs or initial conditons. We develop an adaptive error system of a general form, by comparing the actual adaptive system with a <u>tuned system</u>, i.e., the control system that could be obtained if the plant were known. This error system is similar to the type used in [7],[8] where the tuned system error output is zero, due to the assumption of perfect modeling. By relaxing this assumption we show that the non-zero outputs of the error system are the inputs to a nonlinear feedback error system consisting of the adaptive algorithm and two feedback (interconnection) operators,denoted by H_{ev} and H_{zv} .

An important consequence of this structure is that the <u>existence</u> of solutions (e.g., tuned system performance) is separated from the <u>stability</u> analysis (e.g., stability of the nonlinear error system). In general, the adaptation law is passive; consequently, if H_{ev} is strictly positive real (SPR), then application of passivity theory [19]-[21], provides global

L_2-stability of the map from the tuned system output to the actual adaptive system output, even though the adaptive parameters may grow beyond all bounds. We provide other conditions (e.g., H_{zv} stable) to insure the L_∞ boundedness of the adaptive gains. Similar results are developed to insure L_∞-stability of the error system by using an exponentially weighted passivity theory [19]. These results are summarized in Theorems 1A and 1B.

As a by product of the input/output view we also obtain specific bounds on the L_2 and L_∞ norms of significant signals in the adaptive system. The results are summarized in Corollary 1.

The results in Theorem 1 and Corollary 1 are not essentially new (see e.g., [7],[8]), although they do provide some extentions to previous results. The main contribution, however, is the fact that all the results can be obtained from a generic error system and from the application of nonlinear stablity theorems based on input-output properties. As a consequence of this approach, it is to be expected that conditions for robustness will arise in a natural way. Such robustness results are obtained, but unfortunately,they have a limited practical use. The main limitation is that the global theory (Theorem 1) requires that $H_{ev} \in$ SPR , which in turn places an upper bound on the size of the unmodeled dynamics in the plant. The details are contained in Lemmas 4.1 and 5.2. This bound is quite restrictive and is easily violated by even the most benign model errors, thus, verifying the results obtained in [11], [12]. To over come this limitation, we construct an SPR compensator, based on the scheme proposed in [22] in the context of robust (non-adaptive) control. Although in the adaptive case the supporing arguments are heuristic, an example simulation shows a positive result.

The input/output analysis presented here provides a generic framework within which it is possible to analyze the robustness of adaptive robust controllers. We believe that this framework can be used to develop practical adaptive control algorithms that can be more readily applied to real systems, than the class of algorithms currently in use.

Since this paper merges ideas from several areas, it is necessary to introduce a number of definitions and concepts.

Since this paper merges ideas from several areas, it is necessary to introduce a number of definitions and concepts.

2. SOME PRELIMINARIES

2.1 Notation

The input/output formulation of multivariable systems is the principal view taken throughout this paper and the notation and terminology used is standard (see e.g. [18],[19]). The input and output signals are assumed to be imbedded in either the normed function space

$$L_p^n = \{x : [0,\infty) \to R^n \mid \|x\|_p < \infty\} \tag{2.1a}$$

or its extention

$$L_{pe}^n = \{x : [0,T] \to R^n \mid \|x\|_{Tp} < \infty, \quad T < \infty\} \tag{2.1b}$$

The respective norms $\|\cdot\|_p$ and $\|\cdot\|_{Tp}$ are defined as follows:

$$\|x\|_p = \lim_{T\to\infty} \|x\|_{Tp} \tag{2.2a}$$

with

$$\|x\|_{Tp} = \begin{cases} \left(\int_0^T |x(t)|^p dt\right)^{1/p}, & p \in [1,\infty) \\ \sup_{t\in[0,T]} |x(t)|, & p = \infty \end{cases} \tag{2.2b}$$

where $|\cdot|$ is the Euclidean norm on R^n. Hence, L_{2e}^n is an inner product space, with inner product $\langle x,y\rangle_T$ of elements $x, y \in L_{2e}^n$ defined by

$$\langle x,y\rangle_T = \int_0^T x(t)'y(t)dt \tag{2.3}$$

and so $\|x\|_{T2} = (\langle x,x\rangle_T)^{1/2}$. If $T \to \infty$ then L_2^n is an inner-product space with inner product $\langle x,y\rangle = \lim_{T\to\infty}\langle x,y\rangle_T$.

2.2 Stability

Systems considered in this paper are described by input/output equations of the form $y = Gu$ where $G: L_{pe}^{m} \to L_{pe}^{n}$ is a causal map from u into y, also denoted $u \to y$. The system G is said to be L_p-stable (or simply stable) if G maps $u \in L_p^m$ into $y \in L_p^n$ and if there exists finite constants k and b such that $||Gu||_{Tp} < k \, ||u||_{Tp} + b$, for all $T > 0$ and all $u \in L_{pe}^{m}$. The smallest k that can be found is referred to as the L_p-gain (or simply gain) of G, denoted $\gamma_p(G)$.

Because we often encounter LTI systems it is convenient to introduce the following notation. Let $R(s)$ and $R_o(s)$ denote the proper and strictly proper rational functions, respectively. Let S and S_o denote functions in R(s) and $R_o(s)$, respectively, whose poles all have negative real parts. Thus, S and S_o are the stable, lumped, LTI systems. Denote multivariable systems with transfer function matrices, by $R(s)^{n \times m}$, $S^{n \times m}$, etc. For example, $G \in S_o^{n \times m}$ means that <u>all</u> elements of G belong to S_o, and so on.

If $G \in S^{n \times m}$ then the following L_p-gains are obtained,

$$\gamma_1(G) < \gamma_\infty(G) = \int_0^\infty \bar{\sigma}[G(t)]dt \tag{2.4}$$

$$\gamma_2(G) = \sup_{\omega \in R} \bar{\sigma}[G(j\omega)] \tag{2.5}$$

where $\bar{\sigma}(A)$ denotes the maximum singular value of the matrix A, defined as the positive square root of the maximum eigenvalue of A*A, where * is the conjugate transpose of A. In (2.4), (2.5) G is the operator, $G(j\omega)$ the transfer function matrix, and G(t) is the impulse response matrix.

2.3 Passivity

The following definitions follow those in [19],[21]. Let $G: L_{1e}^{m} \to L_{1e}^{m}$ and let μ, ρ be constants with $\mu > 0$. Then, $\forall \, u \in L_{2e}^{m}$:

G is passive if,

$$< u, G u>_T > \rho \tag{2.6}$$

G is input strictly passive if,

$$< u, Gu >_T > \rho + \mu \| u \|_{T2} \tag{2.7a}$$

G is output strictly passive if,

$$< u, Gu >_T > \rho + \mu \| Gu \|_{T2} \tag{2.7b}$$

(μ and ρ are not the same throughout). When $G \in S^{mxm}$ satisfies (2.7), G is said to be strictly positive real (SPR), denoted $G \in SPR^m$. Because SPR systems play a crucial role in the proof of stability of adaptive systems, we introduce the following subsets:

$$SPR_+^m = \{G \in S^{mxm} | \underline{\lambda}(\tfrac{1}{2}[G(j\omega) + G(-j\omega)'] - \mu I) > 0, \forall \omega \in R\} \tag{2.8a}$$

$$SPR_0^m = \{G \in S_0^{mxm} | \underline{\lambda}(\tfrac{1}{2}[G(j\omega) + G(-j\omega)'] - \mu\, G(-j\omega)'G(j\omega)) > 0, \forall \omega \in R\} \tag{2.8b}$$

where $\underline{\lambda}(A)$ denotes the smallest eigenvalue of A. Thus, whenever $G \in S^{mxm}$, conditions (2.7) can be tested in the frequency domain. Moreover, SPR_0^m and SPR_+^m , respectively, separate the strictly proper SPR functions from the proper, but not strictly proper, SPR functions. In the scalar case, the frequency domain conditions simplify because $\underline{\lambda}[G(j\omega) + G(-j\omega)'] = 2\, Re[G(j\omega)]$.

Certain unstable systems in $R(s)^{mxm}$ can be passive by virtue of (2.6). In particular, $G \in R(s)^{mxm}$ is passive if G(s) is positive real. The transfer function matrix G(s) is positive real if: (i) it has no poles in Re(s) > 0, (ii) poles on the $j\omega$ axis are simple with a non-negative residue, and (iii) for any $\omega \in R$ not a pole of $G(j\omega) + \underline{G}(-j\omega)' >$.

2.4 Model Error

The cornerstone of robust control design is a quantifiable bound on the error between the model used for control design and the actual plant to be controlled. In the adaptive control case considered here the model is a parametric model, where the parameters are not known exactly. The structure of the parametric model can be obtained analytically from physical laws, but this invariably results in a complicated model. Often a simple structure is selected because it is more convenient for analysis and synthesis.

Let P denote the plant to be controlled. In the broadest sense P is a relation in $L_{1e}^m \times L_{1e}^n$, i.e., the set of all possible ordered pairs $(u,y) \epsilon L_{1e}^m \times L_{1e}^n$ of inputs $u \in L_{1e}^m$ and outputs $y \epsilon L_{1e}^n$ that could be generated by the plant [18]. The uncertainty in the plant is denoted by $(u,y) \in P$.

Let $P_\alpha : L_{pe}^m \to L_{pe}^n$ denote a parametric model of the plant P with parameters $\alpha \in R^k$. The parameters can be selected so as to minimize any discrepancies between the model and the plant, i.e.,

$$\inf_{\alpha \epsilon R^k} \| y - P_\alpha u \|_{Tp} = \| y - P_* u \|_{Tp} \tag{2.9}$$

We will refer to $\alpha_* \epsilon R^k$ as the tuned model parameters and to $P_{\alpha *} = P_*$ as the tuned parametric model of the plant. In general, P_* is dependent on the input/output sequence.

Most of the previous work on adaptive control deals with the case where for every $(u,y) \in P$ there exists a tuned parametric model P_*, such that $P_* = P$. In this paper we consider the presence of unmodeled dynamics, thus, the uncertain plant P cannot be perfectly modeled by any parametric model P_α. Since we will deal exclusively with LTI plants $P \in R(s)^{nxm}$, it is convenient to describe this model error in the frequency-domain. Let $B_S(r)$ denote a "ball" in S of radius r, defined by

$$B_S(r) := \{ G \in S^{nxm} | \ \bar{\sigma}[G(j\omega)] < r(\omega),\ \omega \in R \} \tag{2.10}$$

Let the plant to be controlled be described by

$$P = (I + \Delta)P_* \tag{2.11a}$$

where $P \in R(s)^{nxm}$ is the plant, $P_* \in R(s)^{nxm}$ is the tuned parametric model, and $\Delta \in S^{nxn}$ denotes the unmodeled dynamics. Further, the only knowledge available about Δ is that it is bounded such that

$$\Delta \in B_S(\delta) \tag{2.11b}$$

where $\delta(\omega)$ is known for all frequencies. In other words, while the operator Δ is not precisely known, we do know a bound on its effect. This model description (2.2) is used throughout the paper to precisely define the plant to be controlled in an adaptive system. Following Doyle and Stein [16] we will refer to (2.11b) as an unstructred uncertainty. Note that although Δ is stable, P and P* need not be stable. Hence, the parametric model is implicitly required to capture all unstable poles of the plant. Although this is not severly restrictive - at least on practical grounds - nonetheless, it can be eliminated by definng model error as (stable) deviations in (stable) coprime factors of the plant [23]. As the subsequent analysis is not substantially effected by this choice, we will remain with (2.11) for purposes of illustration.

2.5 Persistent Excitation

From [31], a regulated function $F(\cdot) = R_+ \to R^{nxm}$ is persistently exciting, denoted $F \in PE$, if there exists finite positive constants α_1, α_2, and α_3 such that

$$\alpha_2 I_n > \int_s^{s+\alpha_3} F(t)F(t)'dt > \alpha_1 I_n , \quad \forall s \in R_+ \tag{2.12}$$

The usefulness of a persistently exciting signal is in establishing the exponential stability of the following differential equation which arises in many adaptive and identification schemes, i.e.,

$$\dot{x} = -BFHF'x + w \; , \qquad x(0) \in R^n \tag{2.13}$$

It is shown in [31] that if $B \in R^{n \times m}$, $B = B' > 0$, $H \in SPR_0^m$ or SPR_+^m, and $F \in PE$, then $(w, x(0)) \mapsto x$ is exponentially stable, i.e., $\exists\, m, \lambda > 0$ such that

$$|x(t)| < me^{-\lambda t} |x(0)| + \int_0^t me^{-\lambda(t-r)} |w(\tau)| d\tau \; . \tag{2.14}$$

We will utilize this latter result in section IV in our proof of stablity of the adaptive system.

3. ADAPTIVE ERROR MODEL

In this section we develop a generic adaptive error model which will be used in the subsequent analysis. This requires defining the notions of robust control and tuned control.

Robust and Tuned Control

Consider, for example, the model reference adaptive control (MRAC) depicted in Figure 3.1, consisting of the uncertain plant P, a reference model H_r, and an adaptive controller $C(\hat{\theta})$, where $\hat{\theta}$ is the adaptive gain vector, r is a reference input, d is a disturbance process, and n is sensor noise. Denote by $H(\hat{\theta})$ the closed-loop system relating the external inputs $w = (r', d', n')'$ to the output error e, as depicted in Figure 3.2.. Also, let $w \in W$ denote the admissable class of input signals.

The objective of the adaptive controller is twofold: (1) adjust $\hat{\theta}$ to a constant $\theta_* \in R^k$ such that $H(\theta_*)$ has desireable properties; and (2) during adaptation, as $\hat{\theta}$ is adjusted, the error is well behaved. In the usual formulations [7] only (1) is considered and further it is assumed that there exists a <u>matched gain</u>, denoted by $\bar{\theta} \in R^k$, such that

$$H(\bar{\theta}) = 0 \tag{3.1}$$

The presence of uncertain unmodeled dynamics in the plant eliminate the chance of satisfying the matching condition. Thus, it is more appropriate to define a <u>tuned gain</u>, denoted by $\theta_* \in R^k$, corresponding to each $(u,y,w) \in P \times W$, such that

$$H(\theta_*)w < H(\theta)w \quad , \quad \forall\, \theta \in R^k \tag{3.2}$$

The error signal $e_* := H(\theta_*)w$ is referred to as the <u>tuned error</u>. Note that each $(u,y,w) \in P \times W$ engenders a possibly different θ_*. Also, it is important to distinguish the tuned gain θ_*, from the <u>robust gain</u> $\theta_o \in R^k$, where

$$\sup_{P \times W} H(\theta_o)w < \sup_{P \times W} H(\theta)w \ , \ \forall\, \theta \in R^k \tag{3.3}$$

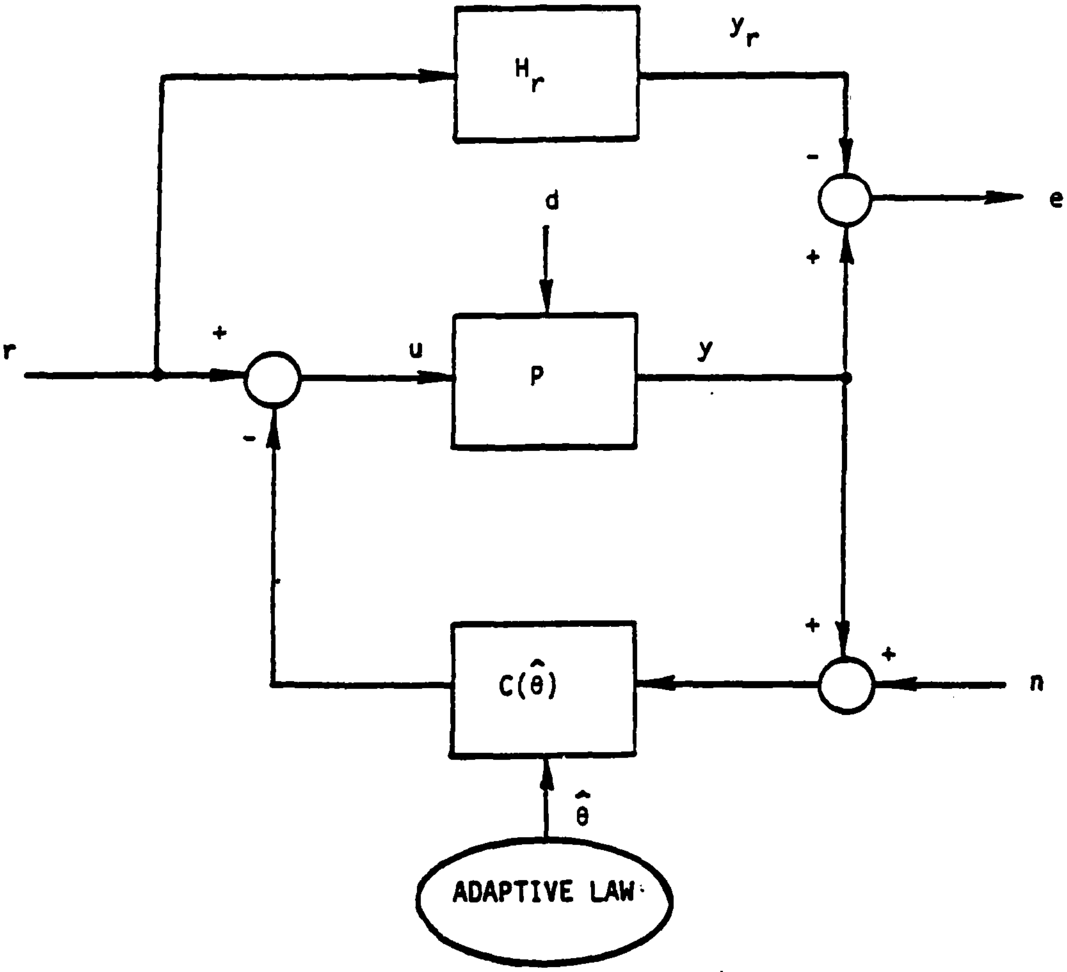


Figure 3.1 A Model Reference Adaptive Controller

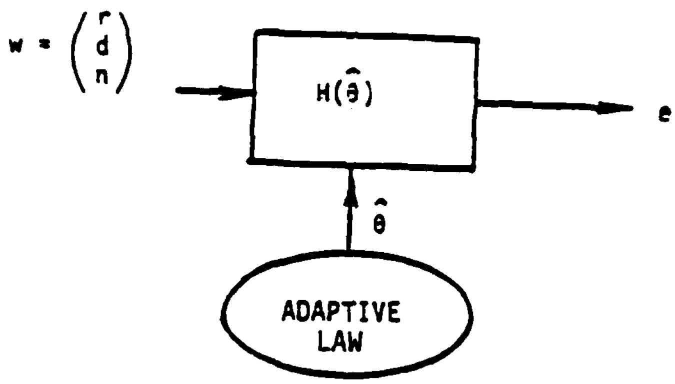


Figure 3.2 Closed-Loop System

The error signal $e_o := H(\theta_o)w$ is referred to as the robust error. It follows from these definitions that the tuned error is always smaller in norm than the robust error, thus $\forall\ w \in W$,

$$e_* = H(\theta_*)w < e_o = H(\theta_o)w \ , \tag{3.4}$$

The tuned controller is, unfortunately, unrealizable since it requires prior knowledge of the actual system $H(\theta)$ (or equivalently, the plant P) and the input w. A practical adaptive controller is likely to have a larger error norm.

Structure of the Adaptive Control

In summary, we consider the multivariable adaptive system, shown in Figure 3.2, and described by

$$e = H(\hat{\theta})w \ . \tag{3.5}$$

where $e(t) \in R^m$ is the error signal to be controlled, $w(t) \in R^q$ is the external input restricted to some set W, and $\hat{\theta}(t) \in R^k$ is the adaptive gain. The class of adaptive controllers considered here are such that the adaptive gains multiply elements of internal signals $z(t) \in R^k$, referred to as the regressor, to produce the adaptive control signals,

$$f_i = \hat{\theta}_i' z_i \quad , \quad i \in [1,m] \tag{3.6}$$

where $\hat{\theta}_i$ and z_i are k_i-dimensional subsets of the elements in $\hat{\theta}$ and z, respectively. Thus,

$$k = \sum_{i=1}^{m} k_i \tag{3.7}$$

Define the adaptive gain error,

$$\theta(t) := \hat{\theta}(t) - \theta_* \tag{3.8}$$

where $\theta_* \in R^k$ is the tuned gain (3.4). Also, define the adaptive control error signals,

$$v_i := \theta_i' z_i \quad , \quad i = 1, \ldots, m \tag{3.9}$$

An equivalent expression is,

$$v = Z'\theta \tag{3.10a}$$

where the time-varying matrix Z is defined by

$$Z = \text{block diag}(z_1, z_2, \ldots, z_m) \tag{3.10b}$$

To describe the relations among the signals e, z, v, and w we introduce the <u>interconnection system</u> $H_I : (w,v) \rightarrow (e,z)$, as shown in Figure 3.3. In particular, let $H_I \in R(s)^{(m+k)x(m+q)}$, and where H_I is defined by,

$$\begin{pmatrix} e \\ z \end{pmatrix} := H_I \begin{pmatrix} w \\ v \end{pmatrix} := \begin{pmatrix} H_{ew} \\ H_{zw} \end{pmatrix} \begin{pmatrix} -H_{ev} & w \\ -H_{zv} & v \end{pmatrix} \tag{3.11}$$

In effect, this structure serves to isolate the adaptive control error v, from the rest of the system. When the adpative control is tuned, $\theta = 0$ and $v = 0$; consequently, the tuned error signal (3.4) is,

$$e_* := H(\theta_*)w = H_{ew}w \tag{3.12}$$

We can also define a tuned regressor signal,

$$z_* := H_{zw}w \tag{3.13}$$

In general, all the subsystems in H_I are dependent on the tuned gains θ_* .

The interconnection system can also be written as,

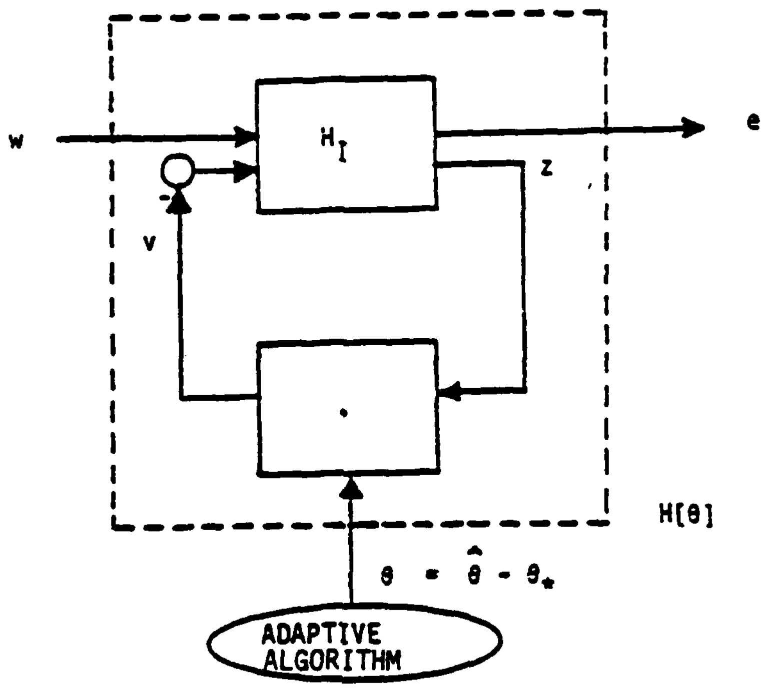


Figure 3.3 Interconnection Structure

$$e = e_* - H_{ev} v \tag{3.14a}$$

$$z = z_* - H_{zv}\, v \tag{3.14b}$$

with v given by (3.10). To complete the error model requires describing the adaptative algorithm, i.e., the means by which $\hat{\theta}(t)$ is generated. We will consider two typical algorithms. A constant gain (gradient) algorithm [7]:

$$\dot{\hat{\theta}} = \Gamma\, Z\, e \tag{3.15}$$

where $\Gamma \in R^{k \times k}$, $\Gamma = \Gamma' > 0$, and a similar but nonlinear gain algorithm:

$$\dot{\hat{\theta}} = \Gamma(Ze - \rho(\hat{\theta})\hat{\theta}) \tag{3.16a}$$

where $\rho : R^k \to R_+$ is a retardation function, whose purpose is to prevent $\hat{\theta}$ from growing too quickly in certain situations. Although many functions will suffice we will select the one proposed in [24], namely:

$$\rho(\hat{\theta}) := \begin{cases} (\|\hat{\theta}\|/c - 1)^2, & \|\hat{\theta}\| > c := \max\|\theta_*\| \\ 0 & , \|\hat{\theta}\| < c \end{cases} \tag{3.16b}$$

The complete adaptive error system, is shown in Figure 3.4. Note that the error system is composed of two subsystems: a linear subsystem Σ_L and a non-linear subsystem Σ_N .

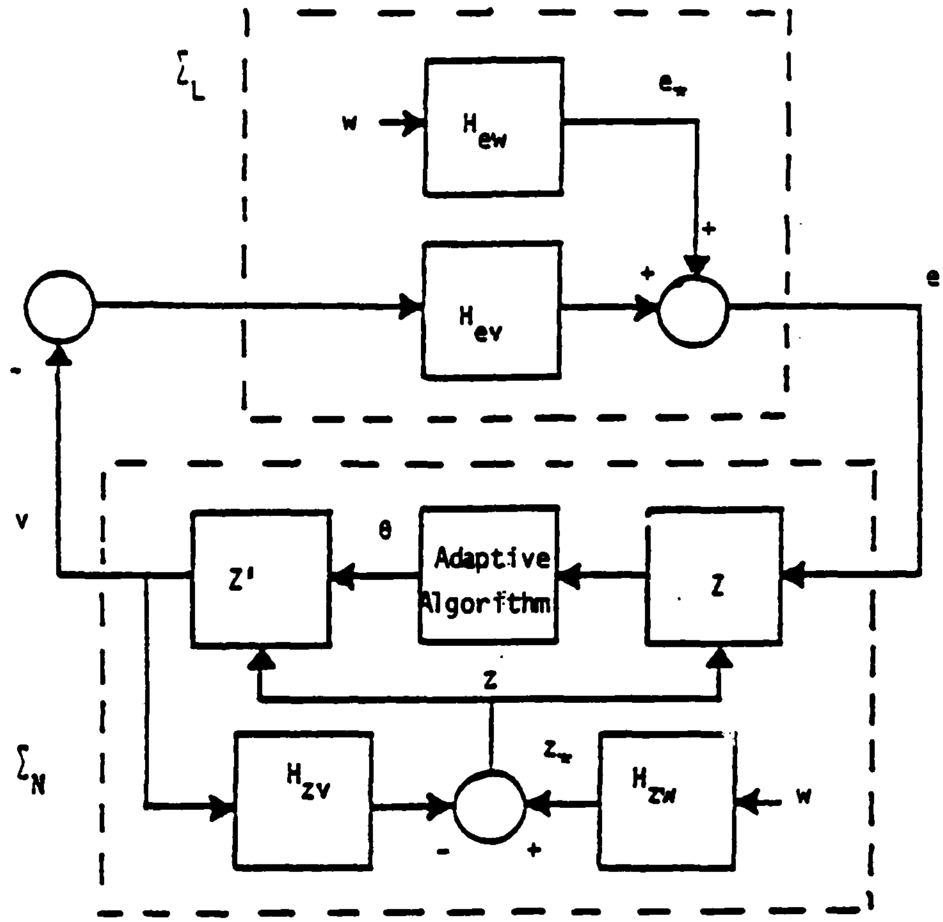


Figure 3.4 Adaptive Error System

4. CONDITIONS FOR GLOBAL STABILITY

The theorems stated below give conditions for which the adaptive error system (Fig. 3.4) is guaranteed to have certain stability and performance properties. Proofs are given in Appendix A. Heuristically, however, the basis for the proofs is application of the Passivity Theorem ([19], pg. 182). It turns out that the map $e \rightarrow v$ is passive. Thus, if H_{ev} is SPR^m, then the map $e_* \rightarrow (e,v)$ is L_2-stable even though z and/or θ can grow without bounds. Further restrictions, provided below, cause θ and z to be bounded. (We use the notation "$x \rightarrow 0$ (exp.)" to mean that $x(t) \rightarrow 0$ (exponentially) as $t \rightarrow \infty$.)

<u>Theorem A: Global Stability</u>

For the adaptive error system shown in Figure 3.4, assume that:

(A1) The system is well-posed in the sense that all inputs $w \in W$ produce signals e, v, z, θ, and $\dot{\theta}$ in $L_{\infty e}$. (4.1a)

(A2) $H_{zv} \in S_0^{k \times m}$ (4.1b)

(A3) $H_{ev} \in SPR_+^m$ (4.1c)

Under these conditions:

(i) If $(e_*, \dot{e}_*) \in L_2^m \cap L_\infty^m$ ($\Longrightarrow e_* \rightarrow 0$) and $(z_*, \dot{z}_*) \in L_\infty^k$ then with algorithm (3.15) or (3.16):

(i-a) $(\theta, \tilde{\theta}) \in L_\infty^k$, $\dot{\theta} \in L_2^k \cap L_\infty^k$, and $\dot{\theta} \rightarrow 0$. (4.a)

(i-b) $e \in L_2^m \cap L_\infty^m$, $\dot{e} \in L_\infty^m$, and $e - e_* \rightarrow 0$. (4.2b)

(i-c) $v \in L_2^m \cap L_\infty^m$, $\dot{v} \in L_\infty^m$, and $v \rightarrow 0$. (4.2c) (i-d) $(z,\dot{z}) \in L_\infty^k$, $(z-z_*, \dot{z}-\dot{z}_*) \in L_2^k \cap L_\infty^k$, and $z-z_* \to 0$ exp. (4.2d)

(i-e) If, in addition, $e_* = 0$ (matched) and $z_* \in PE$ then $(\theta, \dot{\theta}, e-e_*, v, z-z_*) \to 0$ exp. (4.2e)

(ii) If $(e_*, \dot{e}_*) \in L_\infty^m$ and $(z_*, \dot{z}_*) \in L_\infty^k$, then with algorithm (3.15):

(ii-a) $z \in L_\infty^k$ (4.3)

(ii-b) With the addition of either algorithm (3.16) or $z \in PE$ it follows that the elements of $\theta, \dot{\theta}, e, \dot{e}, v, \dot{v}$, and $\dot{z}$ are in L_∞ . (4.4)

Theorem 1B: Global Stability

Replace (A3) in Theorem 1 by

(A3)' $H_{ev} \in SPR_0^m$ (4.5)

(i) If $(e_*, \dot{e}_*) \in L_2^m \cap L_\infty^m$ ($\Rightarrow e_* \to 0$) , and $(z_*, \dot{z}_*) \in L_\infty^k$ then with algorithm (3.15) or (3.16)

(i-a) $(\theta, \tilde{\theta}) \in L_\infty^k$, $\dot{\theta} \in L_2^k \cap L_\infty^k$, $\dot{\theta} \to 0$ (4.6a)

(i-b) $e \in L_2^m \cap L_\infty^m$, $\dot{e} \in L_\infty^m$, $e - e_* \to 0$ (4.6b)

(i-c) $(v,\dot{v}) \in L_\infty^m$ (4.6c)

(i-d) $(z,\dot{z}) \in L_\infty^k$, $(z-z_*, \dot{z}-\dot{z}_*) \in L_2^k \cap L_\infty^k$, and $z-z_* \to 0$. (4.6d)

(i-e) If, in addition, $e_* = 0$ (matched) and $z_* \in PE$, then $(\theta, v) \to 0$ exp. (4.6e)

(ii) If $(e_*, \dot{e}_*) \in L_\infty^m$ and $(z_*, \dot{z}_*) \in L_\infty^k$, then with algorithm (3.15):

(ii-a) $z \in L_\infty^k$ (4.7d)

(ii-b) With the addition of either $z \epsilon PE$ or algorithm (3.16), the elements of θ, $\dot{\theta}$, e, $\dot{e}$, v, $\dot{v}$, and $\dot{z}$ are in L_∞. (4.7b)

Corollary 1: Performance Bounds

Suppose z_* and e_* satisfy the conditions in (i) of Theroems 1A or 1B.

(i) Let $H_{ev} \in SPR_+^m$, i.e., $\exists\, \mu, \gamma > 0$ such that $\forall \omega \epsilon R$,

$$\bar{\sigma}[H_{ev}(j\omega)] < \gamma \text{ and } \tfrac{1}{2}[H_{ev}(j\omega) + H_{ev}(-j\omega)'] > \mu\, I_m \tag{4.8a}$$

Then, bounds on $\|e\|_2$ and $\|\theta\|_\infty$ can be obtained from:

$$\|e-e_*\|_2 < \frac{\gamma}{2\mu}\left[\|e_*\|_2 + \left(\|e_*\|_2^2 + 2\mu\ \theta(0)'\ \Gamma^{-1}\theta(0)\right)^{1/2}\right] \tag{4.8b}$$

$$\|\theta'\Gamma^{-1}\theta\|_\infty < \theta(0)'\ \Gamma^{-1}\theta(0) + 2\|e\|_2\|e-e_*\|_2/\gamma \tag{4.8c}$$

(ii) Let $H_{ev} \in SPR_0^m$, i.e., $\exists\, \mu$, q, k > 0 such that $\forall\ \omega \in R$,

$$\tfrac{1}{2}[H_{ev}(j\omega) + H_{ev}(-j\omega)'] > \mu\ H_{ev}(-j\omega)'\ H_{ev}(j\omega) \tag{4.9a}$$

$$\tfrac{1}{2}[G_{ev}(j\omega) + G_{ev}(-j\omega)'] > k\ I_m \tag{4.9b}$$

$$G_{ev}(s) := (1 + qs)\ H_{ev}(s) \tag{4.9c}$$

Then, bounds on $\|e_2\|$ and $\|\theta\|_\infty$ can be obtained from:

$$\|e\|_2 < \frac{1}{2\mu k}\left[\|e_*+q\dot{e}_*\| + \left(\|e_*+q\dot{e}_*\|_2^2 + 2k^2\mu\theta(0)'\Gamma^{-1}\theta(0)\right)^{1/2}\right] \tag{4.9d}$$

$$\|\theta'\Gamma^{-1}\theta\|_\infty < \theta(0)'\Gamma^{-1}\theta(0) + \frac{1}{k}\|e_*+q\dot{e}_*\|_2\|e\|_2 \tag{4.9c}$$

Discussion

(1) Theorems 1A and 1B give conditions under which the adaptive error system is globally stable. Essentially, conditions are imposed on the interconnection subsystems in H_I. In particular, $H_{ev} \in SPR^m$ and $H_{zv} \in S_0^{k \times m}$ are direct requirements, whereas the restrictions on the tuned signals e_* and z_*, indirectly impose requirements on H_{ew} and H_{zw}. These latter requirements are dependent on knowledge about $w \in W$. For example, if w is a constant, then the assumption that $e_* \to 0$ (Theorem 1A-i) requires that the tuned feedback system is a Type-I robust servomechanism, i.e., the transfer junction $H_{ew}(0) = 0$ for all $(u,y) \in P$.

(2) Corollary 1 gives explicit bounds on signals in the error system. These bounds can be used to evaluate the adaptive system design. Moreover, the bounds allow a coarse determination as to the efficacy of adaptive control vs. robust control. By comparing, for example, the adaptive error $\|e\|_2$ from (4.8) with the robust error $\|e_o\|_2$ from (1.5), it is possible to obtain a quantifiable measure of performance degradation during adaptation.

(3) Although Theorems 1A and 1B are essentially the same, there are slight difference worth noting. These differences arise because in 1A, $H_{ev} \in SPR_+^m \Longrightarrow H_{ev}(s)$ is proper but not strictly proper, whereas in 1B, $H_{ev} \in SPR_0^m \Longrightarrow H_{ev}(s)$ is strictly proper. Thus, comparing part (i) in 1A and 1B, we see that in 1B, $v, \dot{v} \in L_\infty^m$ whereas in 1A, v is additonally in L_2^m and $v \to 0$.

(4) The use of persistent excitation or gain retardation is seen in part (ii) of theorems 1A and 1B to provide the means to guaranty bounded signals. Other schemes based on signal normalizations or dead-zones can provide similar results, e.g. [32],[33]. The effect of these conditions is to provide an L_∞-stability which is not present otherwise. The persistent excitation condition actually supplies exponential stability, which is stronger than L_∞-stability, as provided, for example, by the gain retardation (see proof in Appendix A).

(5) The persistent excitation requirements in parts (i) and parts (ii)

are different. In parts (i), $z_* \epsilon PE$, whereas in parts (ii), $z \epsilon PE$. The different assumptions arise because in parts (i) we enforce the matched condition $e_*=0$. Hence, $z_* \epsilon PE \Rightarrow z \epsilon PE$. This follows from (i-d) where $z - z_* \to 0$ expoentially. Also, with $e_* = 0$, a bounded disturbance added to the reference can cause $z \epsilon$ PE without forcing, $e_* \epsilon L_\infty$. In parts (ii), which is more realistic, we disallow the matched condition, and hence, $e_* \epsilon L_\infty$. Thus, $z \epsilon$ PE is the weakest assumption to make. However, since z is inside the adaptive loop, it is very different to guarantee $z \epsilon$ PE by injecting external signals. Note also (in both parts(ii)) that without retardation or PE it is possible for the regressor to remain bounded even though the adaptive parameters may grow unbounded. Similar results have been reported elsewhere, e.g. [24].

Robustness to Unmodeled Dynamics

Since the theorems impose requriements on the input/output properties of the interconnection system, it follows that the effect of model error on these properties determines the stability robustness of the adaptive system. For example, both theorems require that $H_{ev} \epsilon SPR^m$. Suppose, however, that H_{ev} has the form,

$$H_{ev} = (I + \tilde{H}_{ev})\bar{H}_{ev} \tag{4.10}$$

where $\tilde{H}_{ev}$ is the projection onto H_{ev} of the plant uncertainy operator Δ ; and $\bar{H}_{ev}$ is the nominal transfer function when there is no uncertainty, i.e., when $\Delta = 0$. Thus, $\bar{H}_{ev}$ is a function of the tuned parametric model P_* and the tuned controller gains θ_* . (See Section V for more specific formulae, e.g. (5.5).)

Conditions to insure that $H_{ev} \epsilon SPR_+^m$ despite uncertainty in H_{ev} is provided by the following:

Lemma 4.1: Let H_{ev} be given by (4.3). Then $H_{ev} \epsilon SPR_+^m$ if the following conditions hold:

(i) $\bar{H}_{ev} \epsilon SPR_+^m$ (4.11a)

(ii) $\tilde{H}_{ev} \varepsilon B_S(k)$ where $\forall\ \omega \varepsilon R$, (4.11b)

$$k(\omega) < \frac{1}{2} \underline{\lambda}[\overline{H}_{ev}(j\omega) + \overline{H}_{ev}(-j\omega)']/\overline{\sigma}[\overline{H}_{ev}(j\omega)] \tag{4.11c}$$

<u>Proof:</u> Define $\underline{\mu}(\cdot)$: $C^{mxm} \to R$ by

$$\underline{\mu}\ (A) = \frac{1}{2} \underline{\lambda}(A+A^*)$$

where * denotes conjugate transpose. Then, using definition (2.8) with (4.10) - (4.11) we obtain

$$\underline{\mu}[H_{ev}(j\omega)] = \underline{\mu}[\overline{H}_{ev}(j\omega) + \tilde{H}_{ev}(j\omega)\overline{H}_{ev}(j\omega)]$$
$$> \underline{\mu}[\overline{H}_{ev}(j\omega)] - \overline{\sigma}(\tilde{H}_{ev}(j\omega)]\overline{\sigma}[\overline{H}_{ev}(j\omega)] > 0 .$$

Hnece, $H_{ev} \varepsilon\ SPR_+^m$.

<u>Comments</u>

(1) In order to apply Lemma 4.1 it is necessary to have a detailed description of how the plant uncertainty Δ propagates onto the interconnection uncertainty $\tilde{H}_{ev}$. This type of uncertainty propagation was explored in depth by Safonov [25] and more sophisticated expressions then (4.4b) are available to describe the uncertain operator $\tilde{H}_{ev}$. Section 5 contains more detail on this issue.

(2) In the scalar case (4.11c) becomes

$$\begin{aligned} k(\omega) &< Re[\overline{H}_{ev}(j\omega)]/|\overline{H}_{ev}(j\omega)| \\ &= \cos \measuredangle\ [\overline{H}_{ev}(j\omega)] \end{aligned} \tag{4.12}$$

Since $\overline{H}_{ev}\ \varepsilon\ SPR^m$ by assumption, $k(\omega)$ is always positive for $\omega \varepsilon R$; but because of the cosine function, $k(\omega) < 1$. In Section 6 we show that this limitation on the effect of model error is easily violated by even the most benign type of unmodeled dynamics in the plant. Methods which overcome this limitation are discussed in Section 7. The requirement that $k(\omega) < 1$ also holds for any multivariable $\bar{H}_{ev} \in SPR^m$. To see this let $\bar{H}_{ev}$ have the polar decomposition,

$$\bar{H}_{ev} = G_\ell W_{ev} = W_{ev} G_r \tag{4.13}$$

where G_ℓ , G_r are Hermitian and W_{ev} is unitary. Since $\bar{\sigma}(\bar{H}_{ev}) = \bar{\sigma}(G_\ell) = \bar{\sigma}(G_r)$, it follows that

$$k(\omega) < \bar{\sigma}[W_{ev}(j\omega)] < 1 \tag{4.14}$$

In the case of scalar systems, the condition $k(\omega) < 1$ can be interpreted in terms of a limitation on relative degree of $H_{ev}(s)$. A necessary condition for $H_{ev} \in SPR$ is that the relative degree of $H_{ev}(s)$ does not exceed one i.e., phase limited to ±90°. Rohrs, et al. [12] show that this necessitates precise knowledge of plant order, and hence, is not a feasible requirement in the presence of an unstructured uncertainty (2.12), where the order is unknown. In the multivariable case it is awkward to talk about relative degree or phase, however, (4.14) expresses the same limitation.

(3) In several instances, e.g., [9],[26],[27], it has been reported that the SPR condition has been eliminated. In each case, however, it can be verified that the operator H_{ev} = positive constant , which is SPR. But, these studies do not account for unmodeled dynamics, thus, in the notation of (4.10), only $\bar{H}_{ev}$ = positive constant . Lemma 4.1 then provides the means to evaluate the effect of unmodeled dynamic.

5. APPLICATION TO MODEL REFERENCE ADAPTIVE CONTROL

Consider the model reference adaptive control (MRAC) system, shown in Figure 5.1, consisting of: an uncertain scalar plant $P \in R_0(s)$; a reference model $H_r \in S_0$; and filters with $F \in S_0^{\ell \times 1}$. The plant is affected by a disturbance d and a reference command r. The system equations are:

$$e = y - y_r \tag{5.1a}$$

$$y_r = H_r r \tag{5.1b}$$

$$y = d + Pu \tag{5.1c}$$

$$u = -\hat{\theta}'z = -(\hat{\theta}_1'z_1 + \hat{\theta}_2'z_2) \tag{5.1d}$$

$$z_1 = F u, \; z_2 = F(y-r) \tag{5.1e}$$

Assume that the adaptive law is given by (3.15), thus,

$$\dot{\hat{\theta}} = \Gamma z e \tag{5.1f}$$

Let the plant uncertainty be described by(2.12), i.e.,

$$\Delta := \frac{P-P_*}{P_*} \in B_S(\delta) \tag{5.1g}$$

where $P_* \in R_0(s)$ is a tuned parametric model for P. Let the filter dynamics be given by

$$F(s) = \left(\frac{1}{L(s)}, \frac{s}{L(s)}, \ldots, \frac{s^{\ell-1}}{L(s)}\right)' \tag{5.1h}$$

where L(s) is a stable monic polynomial of degree ℓ . Thus, $\hat{\theta}_1(t), \hat{\theta}_2(t) \in R^{\ell}$ and so $\hat{\theta}(t) \in R^{2\ell}$. Using the definition of tuned gain (3.2) we get,

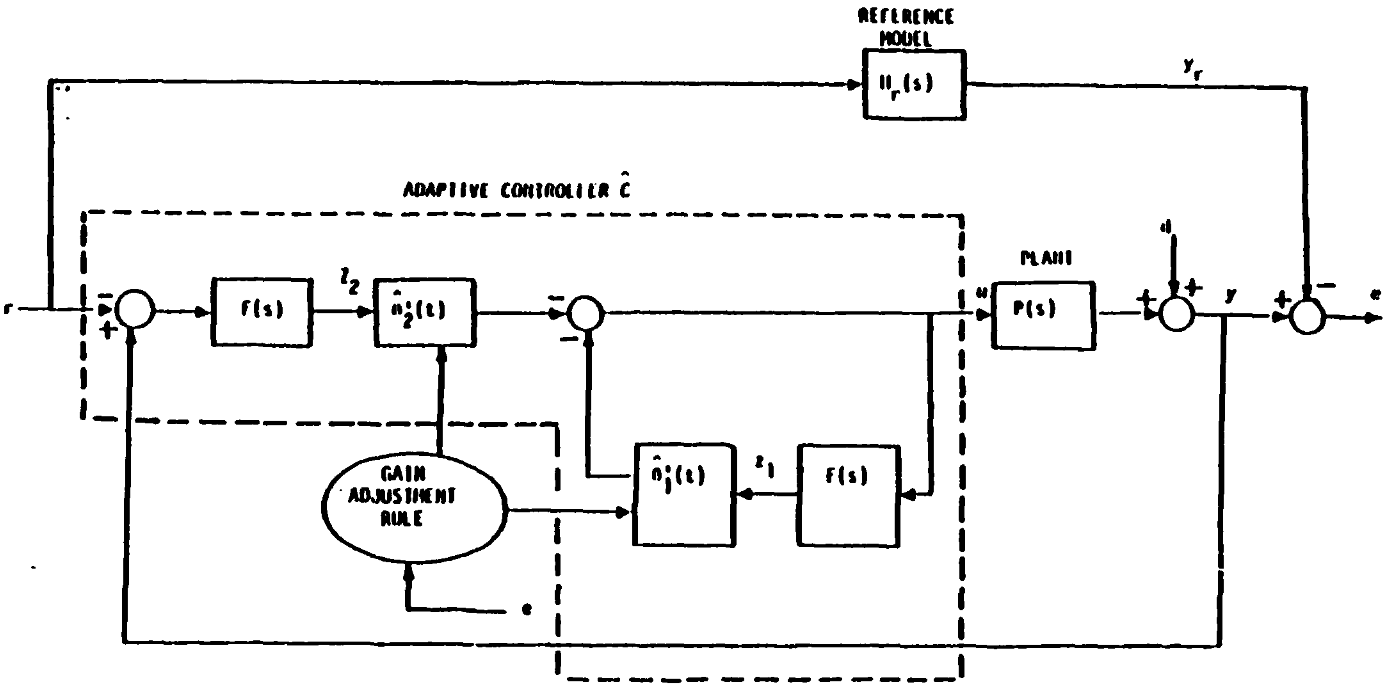


Figure 5.1 MRAC System With Scalar Plant

$$
\begin{aligned}
u &= -\hat{\theta}'z = -(\theta_*+\theta)'z \\
&= -(\theta_{*1}'z_1 + \theta_{*2}'z_2) - v\ , \quad v := \theta'z \ \text{ from (3.6)} \\
&= -\frac{A_{*1}}{L}u + \frac{A_{*2}}{L}(r-y) - v
\end{aligned}
$$

Finally,

$$
u = \frac{A_{*2}/L}{1+A_{*1}/L}(r-y) - \frac{1}{1+A_{*1}/L}v := C_*(r-y) - \frac{1}{1+A_{*1}/L}v \tag{5.2}
$$

where A_* and A_{*2} are polynomials, each of degree $\ell-1$, whose coefficients are the elements of the tuned gains θ_{*1} and θ_{*2} , respectively; and C_* denotes the tuned controller. The tuned system ($\theta=0$) is shown in Figure 5.2.

In terms of the uncertain plant P, the adaptive error system (Fig. 3.4) corresponding to this MRAC system, has tuned signals:

$$
e_* = (1 + PC_*)^{-1}d + [(1+PC_*)^{-1}PC_*-H_r]r \tag{5.3a}
$$

$$
z_* = \begin{bmatrix} F(1+PC_*)^{-1}C_*(r-d) \\ F(1+PC_*)^{-1}(d-r) \end{bmatrix} \tag{5.3b}
$$

and interconnections:

$$
H_{ev} = (1+PC_*)^{-1}P(1+A_{*1}/L)^{-1} \tag{5.3c}
$$

$$
H_{zv} = \begin{bmatrix} F(1+PC_*)^{-1}(1+A_{*1}/L)^{-1} \\ F(1+PC_*)^{-1}P(1+A_{*1}/L)^{-1} \end{bmatrix} \tag{5.3d}
$$

The error system can also be described so as to highlight the model error Δ . The following definitions are convenient:

$$
T_* := (1+P_*C_*)^{-1}P_*C_* := 1 - S_* \tag{5.4a}
$$

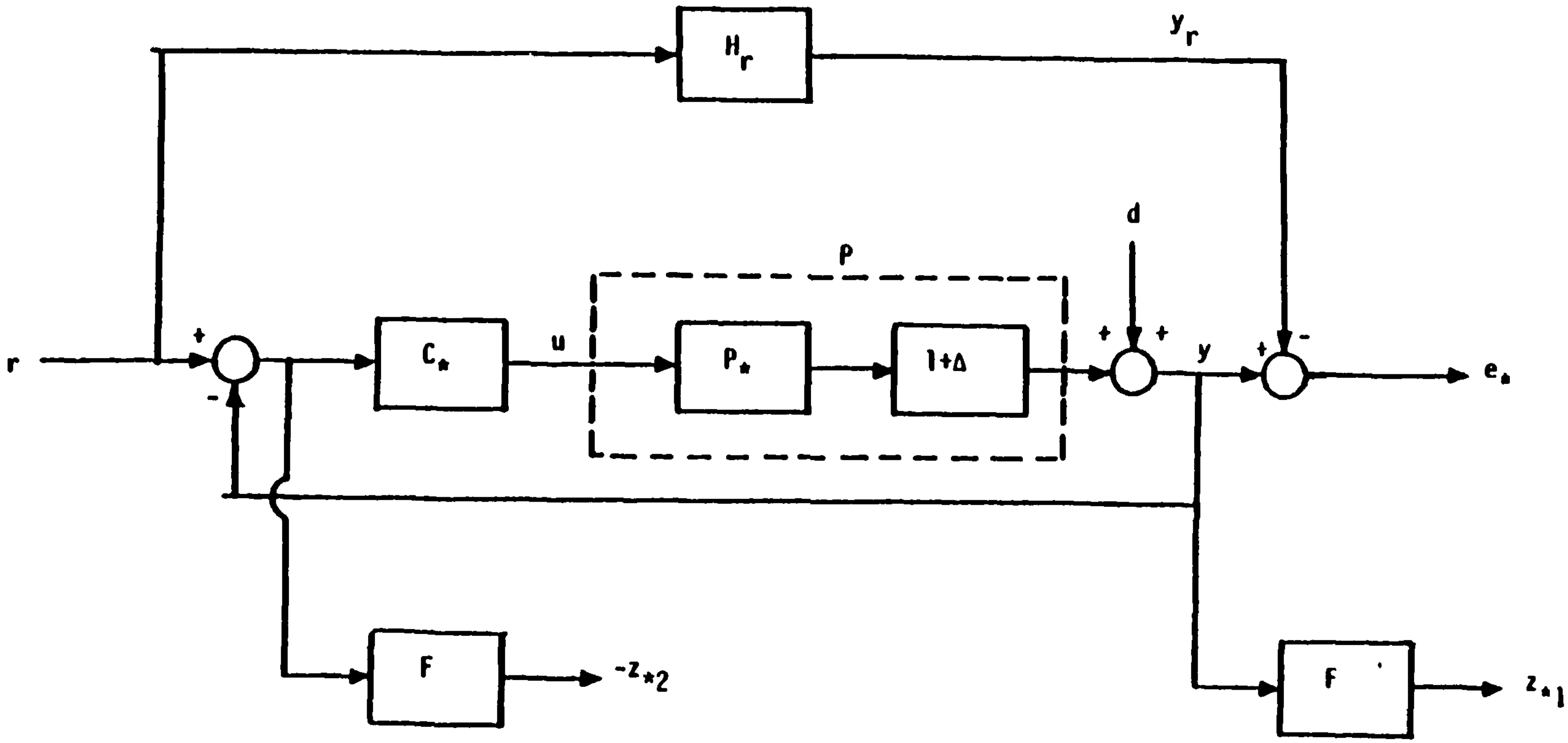


Figure 5.2 Tuned System

$$K_* := H_{ev}\Big|_{\Delta=0} = (1+P_*C_*)^{-1}P_*(1+A_{*1}/L)^{-1} \qquad (5.4b)$$

Thus, the error system (5.3) can be also be expressed as:

$$e_* = S_*(1+\Delta T_*)^{-1}d + (T_*(1+\Delta)(1+\Delta T_*)^{-1}-H_r)r \qquad (5.5a)$$

$$z_* = \begin{bmatrix} F\ S_*C_*(1+\Delta T_*)^{-1}(r-d) \\ F\ S_*(1+\Delta T_*)^{-1}(d-r) \end{bmatrix} \qquad)5.5b)$$

$$H_{ev} = K_*(1+\Delta)(1+\Delta T_*)^{-1} \qquad (5.5c)$$

$$H_{zv} = \begin{bmatrix} F\ K_*P_*^{-1}(1+\Delta T_*)^{-1} \\ F\ K_*(1+\Delta)(1+\Delta T_*)^{-1} \end{bmatrix} \qquad (5.5d)$$

The result that follows in Lemma 5.1 gives conditions under which $H_{ev} \varepsilon\ SPR_o$ and $H_{zv} \varepsilon\ S_o^{2\ell x1}$, despite model error; thus conditions (A1)-(A3) of Theorems 1A and 2B are satisfied. Additional requirements are necessary to establish the class of tuned signals e* and z* as given by (5.5a) and (5.5b), respectively. These requirements are discussed following Lemma 5.1.

Lemma 5.1: For the adaptive system (5.3) or (5.5) $H_{ev} \varepsilon\ SPR_o$ and $H_{zv} \varepsilon\ S_o^{2\ell x1}$ if the following conditions are all satisfied:

(i) $$P_*(s) = \frac{g(s^{n-1}+\beta_1 s^{n-2}+ \dots + \beta_{n-1})}{s^n+\alpha_1 s^{n-1}+ \dots + \alpha_n} = \frac{gN_*(s)}{D_*(s)} \qquad (5.6a)$$

(ii) $N_*(s)$ is a stable monic polynomial (5.6b)

(iii) $g > 0$ (5.6c)

(iv) $K_*(s) = \dfrac{g\ K_1(s)}{K_2(s)}\ \varepsilon\ SPR_o$ where $K_1(s)$ and $K_2(s)$ are monic stable polynomials.

(5.6d)

(v) $\quad \ell = \deg L(s) \geq n + \deg K_1(s) - 1$ (5.6e)

(vi) $\quad \Delta \in B_S(\delta)$ is such that

$$\left.\begin{aligned} \delta(\omega) < \bar{\delta}(\omega) &:= \eta(\omega)[\eta(\omega)|T_*(j\omega)| + |S_*(j\omega)|]^{-1} \\ \eta(\omega) &:= \cos \measuredangle [K_*(j\omega)] \end{aligned}\right\} \forall\, \omega \in R, \tag{5.6f}$$

Proof: See Appendix B.

Discussion

(1) Condition (i)-(v) of Lemma 5.1 are restatements of known results, but normally they apply to the actual plant P, e.g. [7]. In Lemma 5.1, however, these conditions apply to the parametric model P* -- not to the actual plant. As such, they are easier to satisfy, since the parametric model is somewhat arbitrary. This flexibility is penalized by an increase in model error. For example, if the actual plant has a relative degree of 2, then choosing a parametric model of relative degree 1 -- as required by condition (i) -- incrases the high frequency model error.

(2) Condition (vi) imposes an upper bound $\bar{\delta}$ on the model error associated with the chosen parametric model. This condition simultaneously insures that $H_{ev} \in SPR_0$ despite model error, and that the tuned system is stable (see proof in Appendix B).

(3) It is easily verified that $\bar{\delta}(\omega) < 1$, as was discussed following Lemma 4.1. In fact, even the "optimally tight" bound (see [25] for details on this calculation) given by,

$$\bar{\delta} = \frac{1}{2|T|\eta}\left[-|1-T| + \left(|1+T|^2 + 4\eta\ \mathrm{Re}(KT/|K|)\right)^{1/2}\right] \tag{5.7}$$

is also restricted to be less than 1. This limitation severely restricts the type of admissable model error. This issue is pursued in Section 6.

(4) To guarantee global stability using the adaptive law (5.1f), property (i) of Theorem 1 requires that $e_* \to 0$ and $z_*, \dot{z}_* \in L_\infty^{2\ell}$ for all r and d. For example, let r and d be any bounded signals such that $r \to$ constant and $d \to$ constant as $t \to \infty$. Property (i) of Theorem 1 is satisfied if:

$$\delta(0) = 0 \tag{5.8a}$$

$$T_*(0) = H_r(0) = 1 \tag{5.8b}$$

Zero model error at DC (5.8a) is certainly to be expected from even the most crude tuned parametric model.

(5) Let r be bounded such that $r \to$ constant as $t \to \infty$, but let d be just bounded, i.e., $d \in L_\infty$. In this case it is not possible to guarantee $e_* \to 0$, but we can guarantee that $e_* \in L_\infty$. To obtain global stability in this case, requires the introduction of the retardation term (3.16) into the adaptive law (5.1f), see part (ii) of Theorems 1A or 1B.

(6) It is possible to obtain versions of Lemma 5.1 for adaptive systems of different forms, e.g., indirect adaptive [5]. Also, the use of "multipliers", e.g. [4], can be accounted for as well. The multiplier effectively makes use of the availability of $\hat{\theta}$ as a signal; and this allows rel deg $(P_*) = 2$ rather than 1 as required by condition (i) of Lemma 5.1.

6. LIMITATIONS IMPOSED BY THE SPR CONDITION

The fact that the model error bound given in condition (vi) of Lemma 5.1 can not exceed one has unfortunate consequences.

Example 1

Consider a plant with transfer function,

$$P(s) = P_*(s)\,\frac{ab}{(s+a)(s+b)} \tag{6.1}$$

where P_* is the parametirc model, with two unmodeled stable poles at -a and -b. Suppose,also, that b is much greater than a, and that a is much greater than the bandwidth of $P_*(s)$. This situation seems benign -- and most likely a certainty. Comparing (6.1) with (5.1g) gives,

$$\delta(\omega) = \omega\left[\frac{\omega^2 + (a+b)^2}{(\omega^2+a^2)(\omega^2+b^2)}\right]^{1/2} > 1$$

for all frequencies $\omega > (ab/2)^{1/2}$, thus, condition (vi) of Lemma 5.1 is violated, and global stability cannot be guaranteed. The following example illustrates this point.

Example 2

Consider the example MRAC system (Fig. 5.1) studied by Rohrs et al. [12], where:

$$P(s) = \frac{2}{s+1}\,\frac{229}{(s+15)^2 + 4}$$

$$H_R(s) = \frac{3}{s+3}$$

$$u = -\hat{\theta}_1 y + \hat{\theta}_2 r$$

$$\dot{\hat{\theta}}_1 = ye, \quad \hat{\theta}_1(0) = .65$$

$$\dot{\hat{\theta}}_2 = -r\,e, \quad \hat{\theta}_2(0) = 1.14$$

Let r = constant and $d = 0$. Thus, $e_* \to 0$ exponentially when the tuned gains are such that (5.8) is satisfied, i.e.,

$$T_*(0) = \frac{2\theta_{*2}}{1+2\theta_{*1}} = H_r(0) = 1$$

Even though $(\theta_{*1}, \theta_{*2})$ exist to satisfy this, $H_{ev}(s)$ is not SPR, and so global stability is not guaranteed. Simulation runs with $r = .4$ and $r = 4.0$ are shown in Figures 6.1 and 6.2, respectively. With the small input (Fig. 6.1) we see a stable response which tracks the reference very well. With the large input (Fig. 6.2) the response is still stable, but large oscillations are taking place. Larger inputs will eventually drive the system unstable, e.g. [12].

In this example, if the tuned model is taken to be $P_*(s) = 1/(s+1)$ then it is easily verified that model error $\delta(\omega)$ is greater than one at some frequency.

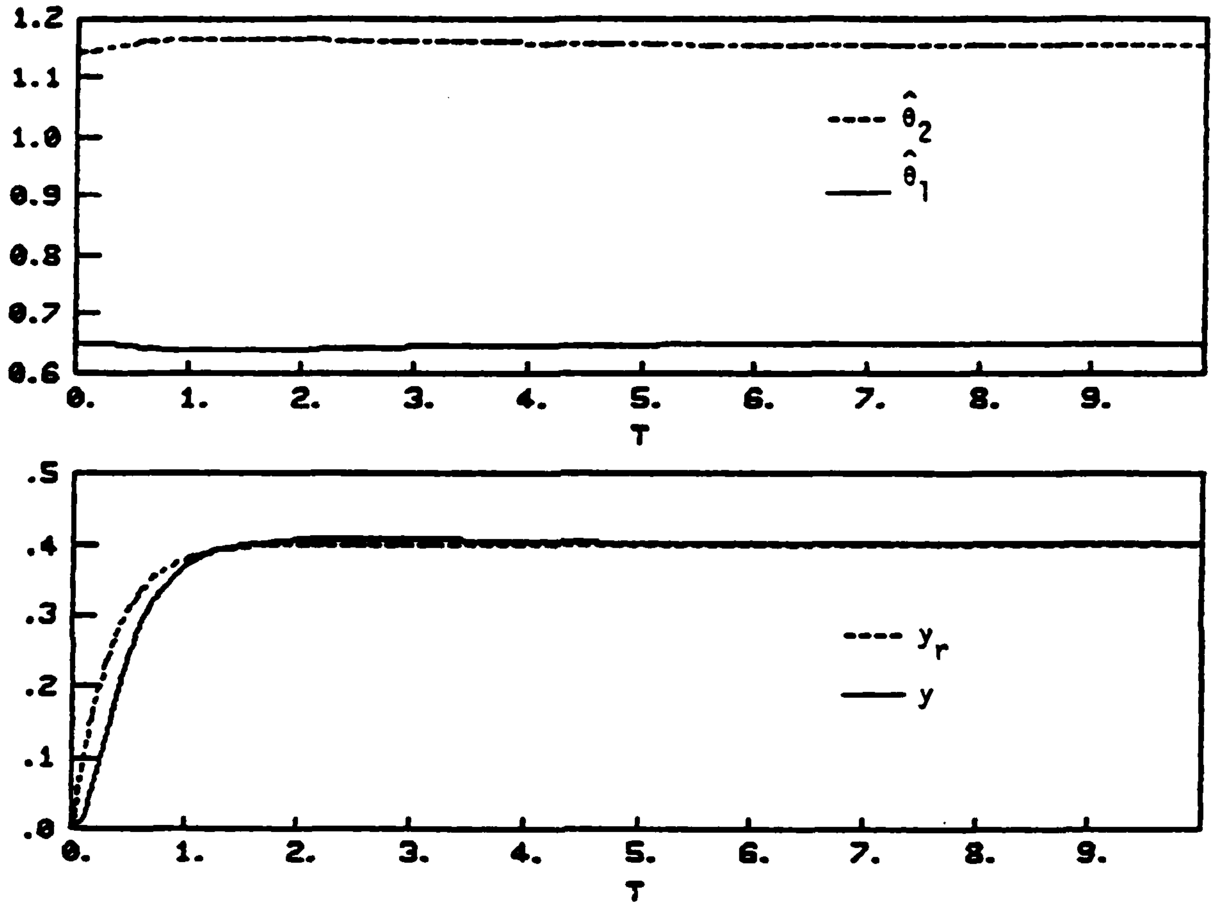


Figure 6.1 Response to r = .4

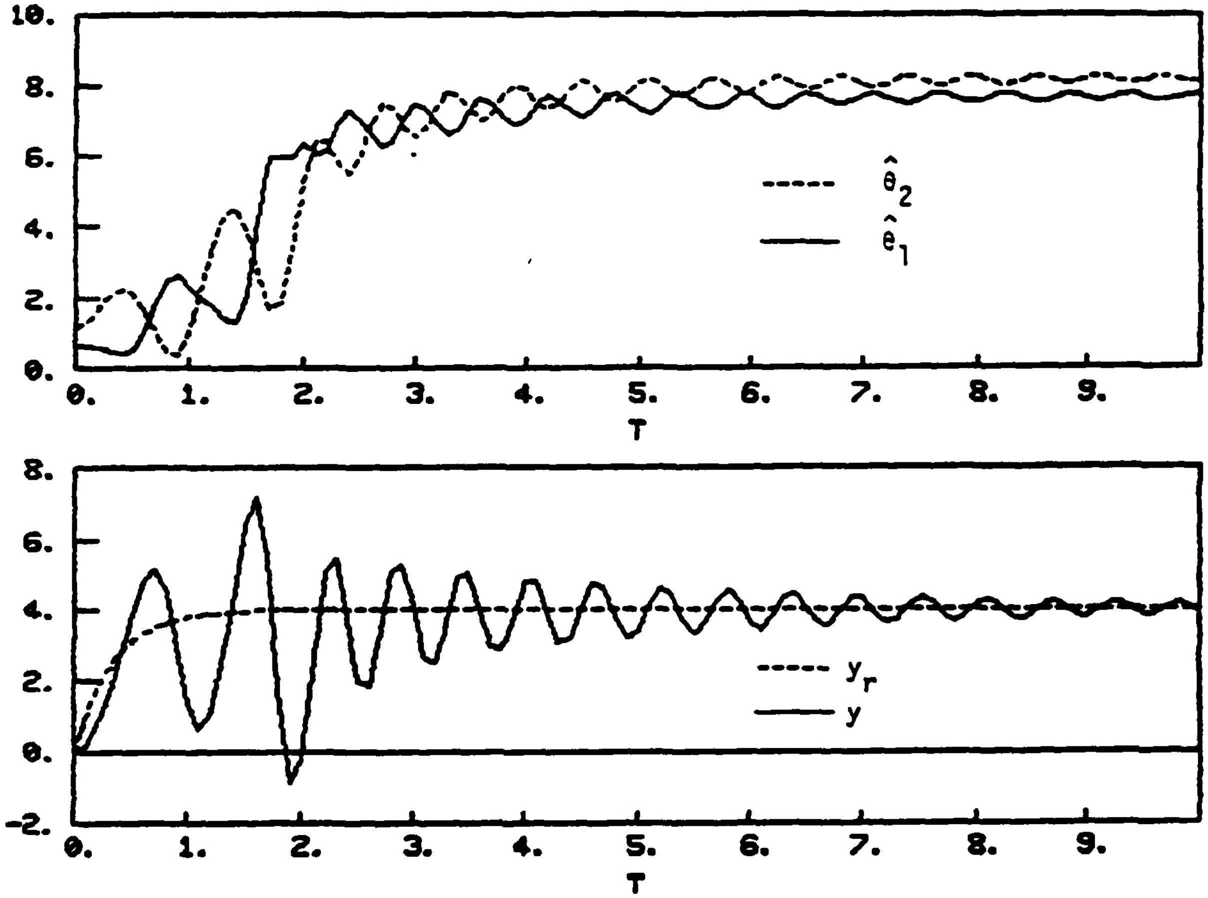


Figure 6.2 Response to r = 4.0

7. SPR COMPENSATION

In this section we heuristically develop a means to obtain global robust adaptive control. Since the SPR condition is violated whenever model error exceeds one, a natural scheme is to construct an SPR compensator which alleivates the problems by "filtering" the plant output; thus, avoiding the trouble. However, direct filtering does not change the size of model error. For example, with the plant $P = (1+\Delta)P_*$, let y_W denote the output of the filtered plant, where

$$y_w := Wy = Wd + (1+\Delta)WP_*u \tag{7.1}$$

Thus, model error is uneffected. Even filtering H_{ev} directly by W offers no help, since the bound (4.4c) is still less than one, i.e.,

$$|\tilde{H}_{ev}| < Re(W\,\bar{H}_{ev})/|W\,\bar{H}_{ev}| < 1 \tag{7.2}$$

for any stable W. What we seek is an SPR compensator which only effects the unmodeled dynamics, but leaves the paramtric model intact.

A compensation scheme, which offers some promise as an SPR compensator, is that proposed in [22], as shown in Figure 7.1. To see the desired result suppose that $P = (1+\Delta)P_m$ with $\Delta \in B_S(\delta)$. Then, the compensator is equivalent to a plant which maps (u,d) into y_c where

$$y_c = Wd + P_cu \tag{7.2a}$$

$$\Delta_c := \frac{P_c - P_m}{P_m} \in B_S(W\delta) \tag{7.2b}$$

Thus, whenver $\delta(\omega) > 1$, select W(s) such that $|W(j\omega)|\delta(\omega) < 1$. The filter W acts like a "frequency switch" whose function is to insure condition (vi) of Lemma 5.1.

There are two ways to implement this compensator in an adaptive system. The first way is to use a fixed model of the plant for P_m, i.e., $P_m = \bar{P}$. The second way is to replace P_m with an adaptive observer, i.e., $P_m = \hat{P}$.

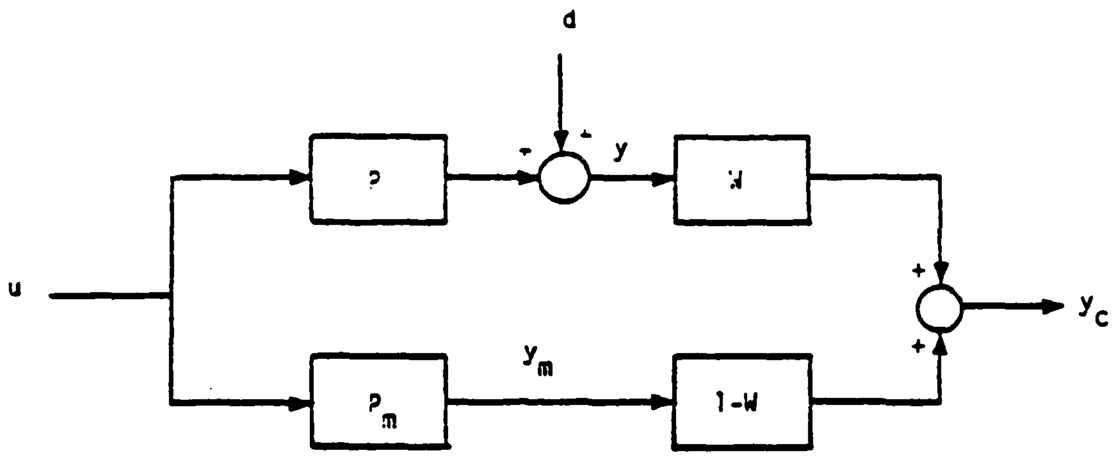


Figure 7.1 SPR Compensation

In either case, to obtain the benefit of the SPR compensator, the signal to be controlled is the compensator output y_c, not the plant output y. Both of these compensators will now be examined.

Fixed SPR Compensator

Let $P_m = \bar{P}$, a fixed model, and let the actual plant be given by (2.17), $P = (1+\Delta)P_*$ with $\Delta \in B_S(\delta)$. Then the fixed compensator plant equivalent model error (7.2b) is:

$$\Delta_c := \frac{P_c - P_*}{P_*} \in B_S(\delta_1) \tag{7.3a}$$

where

$$\delta_1(\omega) := |W(j\omega)|\delta(\omega) + |1 - W(j\omega)| \cdot \left|\frac{\bar{P}(j\omega) - P_*(j\omega)}{P_*(j\omega)}\right| \tag{7.3b}$$

This scheme is motivated by the fact that at low frequencies the tuned parametric model P_* is close to P; thus δ is small and $W \approx 1$. At high frequencies δ is large but $(\bar{P} - P_*)/P_*$ is small, $W \approx 0$ and so δ_1 is small. Of course the compensator is limited if there is large model error at intermediate frequencies.

Example 2

Example 1 is modified to include a fixed SPR compsnator with $W(s) = 1/(s+1)$ and $\bar{P}(s) = 2/(s+1)$. Simulation results with the large step command (r=4) are shown in Figure 7.2. Comparing these to Figure 6.2, without compensation, it is readily verified that the instability tendencies are eliminated. Also, direct calculations reveal that $H_{ev} \in SPR_o$, thus global stability is insured.

Adaptive SPR Compensation

An adaptive SPR compensator, together with the adaptive controller, is shown in Figure 7.3. The adaptive controller is described by,

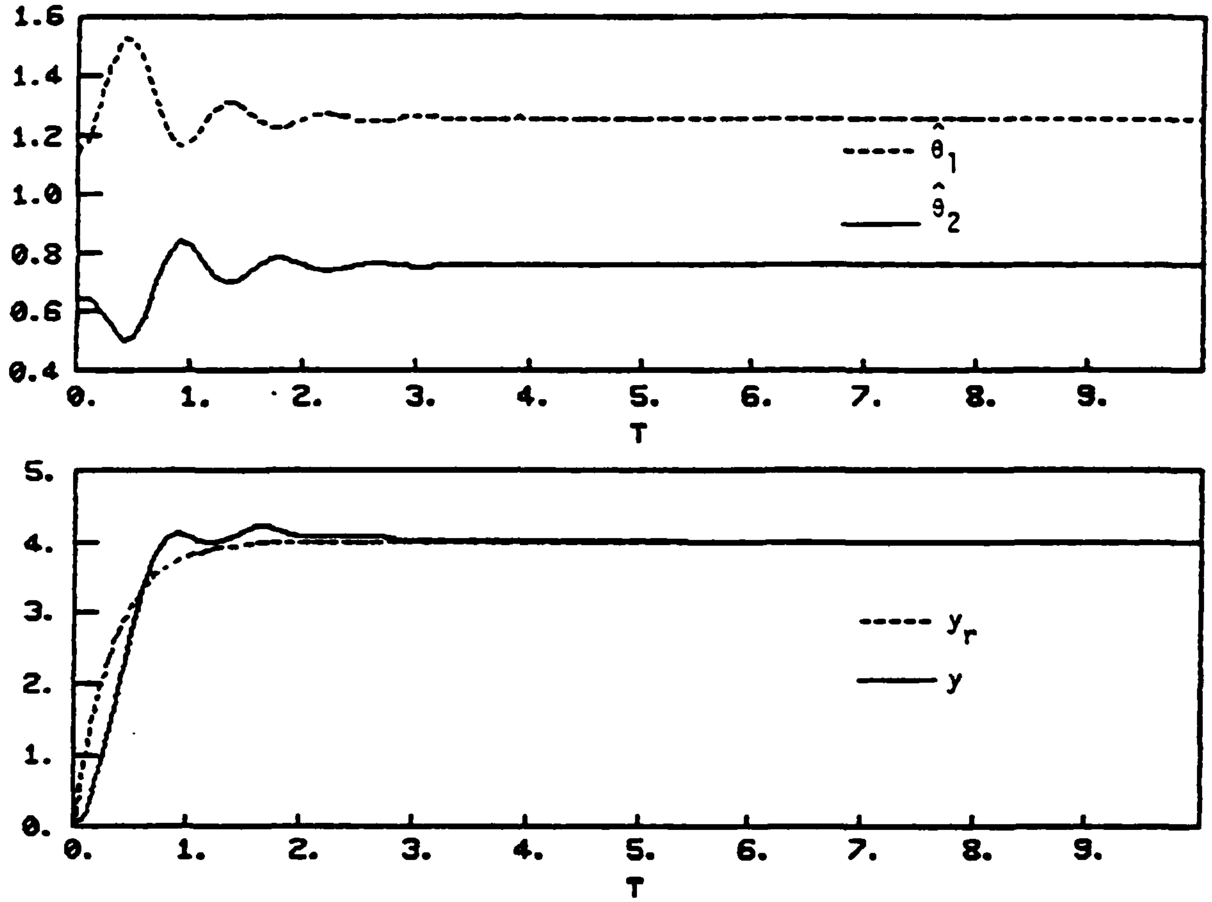


Figure 7.2 MRAC with SPR Compensator, r = 4.0

$$u = -\hat{\theta}_c' z_c \quad , \quad z_c' = \left(F_c' u \ , \ F_c'(y_c - r)\right) \tag{7.4a}$$

$$\dot{\hat{\theta}}_c = \Gamma_c \, z_c \, e_c \ , \quad e_c = y_c - y_r \tag{7.4b}$$

$$F_c'(s) = \left(1/L_c(s), \ldots, s^{n_c - 1}/L_c(s)\right) \ , \ n_c = \deg L_c(s) \tag{7.4c}$$

and the adaptive observer is described by,

$$\hat{y} = \hat{\theta}_o' \, z_o \quad , \quad z_o' = (F_o' \, u. \, -F_o' \, y) \tag{7.4d}$$

$$\dot{\hat{\theta}}_o = \Gamma_o z_o e_o \quad , \quad e_o = y - \hat{y} \tag{7.4d}$$

$$F_o'(s) = \left(1/L_o(s), \ \ldots, \ s^{n_o - 1}/L_o(s)\right) \ , \ n_o = \deg L_o(s) \tag{7.4f}$$

where $L_o(s)$ and $L_c(s)$ are both monic and stable. To generate the error system interconnection operators associated with this system, let θ_{*c} and θ_{*o} denote the tuned parameters with respective gain errors, θ_c and θ_o ; and let $v_c := \theta_c' z_c$ and $v_o := \theta_o' z_o$ be the corresponding adaptive control errors (3.6). By analogy with the procedure used in Section 5 we get,

$$u = C_*(r - y_c) - \frac{1}{I + A_{*1}/L_c} v_c \tag{7.5}$$

$$\hat{y} = -\frac{B_{*1}}{L_o} d + \left(1 - \frac{B_{*1}}{L_o} \Delta\right) P_* u + v_o \tag{7.6}$$

where

$$C_* = \frac{A_{*2}/L_c}{I + A_{*1}/L_c} \tag{7.7}$$

$$P_* = \frac{B_{*2}/L_o}{I + B_{*1}/L_o} = \frac{gN_*}{D_*} \tag{7.8}$$

and where (A_{*1}, A_{*2}) are polynomials whose coefficients are the parameters in θ_{*c} ; (B_{*1}, B_{*2}) are polynomials whose coefficients are the parameters in θ_{*o} ; and N_*, P_* and g are as defined by (5.6a). The adaptive error model is given below in terms of T_*, S_*, and K_* as defined in (5.4). In additon, define:

$$R := 1 + (W-1)\,\frac{D_*}{L_o} \tag{7.9}$$

The tuned signals are:

$$e_{*c} = S_*(1+\Delta RT_*)^{-1}R\,d + \left(T_*(1+\Delta R)(1+\Delta RT_*)^{-1}-H_r\right)r \tag{7.10a}$$

$$e_{*o} = D_*L_o^{-1}(1+\Delta RT_*)^{-1}d + D_*L_o^{-1}T_*\Delta(1+\Delta RT_*)^{-1}r \tag{7.10b}$$

$$z_{*c} = \begin{bmatrix} F_cA_{*2}L_c^{-1}P_*^{-1}K_*(1+\Delta RT_*)^{-1}(r-Rd) \\ F_cS_*(1+\Delta RT_*)^{-1}(Rd-r) \end{bmatrix} \tag{7.10c}$$

$$z_{*o} = \begin{bmatrix} F_oA_{*2}L_c^{-1}P_*^{-1}K_*(1+\Delta RT_*)^{-1}(r-Rd) \\ F_oT_*(1+\Delta RT_*)^{-1}\left(d - (1+\Delta)r\right) \end{bmatrix} \tag{7.10d}$$

The interconnections are:

$$H_{ev} = \begin{bmatrix} K_*(1+\Delta R)(1+\Delta RT_*)^{-1} & -(1-W)S_*(1+\Delta RT_*)^{-1} \\ K_*D_*L_o^{-1}\Delta(1+\Delta RT_*)^{-1} & 1+(1-W)T_*D_*L_o^{-1}(1+\Delta RT_*)^{-1} \end{bmatrix} \tag{7.11a}$$

$$H_{z_cv} = \begin{bmatrix} F_cP_*^{-1}K_*(1+\Delta RT_*)^{-1} & F_cA_{*2}L_c^{-1}P_*^{-1}K_*(1-W)(1+\Delta RT_*)^{-1} \\ F_cK_*(1+\Delta R)(1+\Delta RT_*)^{-1} & -F_cS_*(1-W)(1+\Delta RT_*)^{-1} \end{bmatrix} \tag{7.11b}$$

$$H_{z_0 v} = \begin{bmatrix} F_0 P_*^{-1} K_*(1+\Delta RT_*)^{-1} & F_0 A_{*2} L_c^{-1} P_*^{-1} K_*(1-W)(1+\Delta RT_*)^{-1} \\ -F_0 K_*(1+\Delta)(1+\Delta RT_*)^{-1} & -F_0 T_*(1-W)(1+\Delta)(1+\Delta RT_*)^{-1} \end{bmatrix} \tag{7.11c}$$

The factor $(1+\Delta RT_*)^{-1}$ appears in all the terms above. The transfer function R (7.9) reduces the effect of unmodeled dynamics; however not exactly by the amount anticipated, vis a vis (7.2). This is due to additonal model error introduced by the adaptive observer. Nonetheless, the model error attenuation is greater than with the fixed SPR compensator. In particular, at low frequencies $\Delta \approx 0$ and at high frequencies $R \approx 0$, since $W \approx 0$ and $D_* L_0^{-1} \approx 1$. Without further testing of H_{ev} (7.11a) it is not possible to state that $H_{ev} \in SPR_0$ at intermediate frequencies. Note, however, that the nominal value of H_{ev} is:

$$\bar{H}_{ev} = \begin{bmatrix} K_* & -(1-W)S_* \\ 0 & 1 \end{bmatrix} . \tag{7.12}$$

which is SPR_0 provided that $K_* \in SPR$ and

$$\mathrm{Re}\, K_*(j\omega) > \frac{1}{4}|(1-W(j\omega))S_*(j\omega)|^2 \ , \ \omega \in R \tag{7.13}$$

Applying (4.11) to (7.11a), a tedious procedure, would give an upper bound on model error to insure $H_{ev} \in SPR_0$.

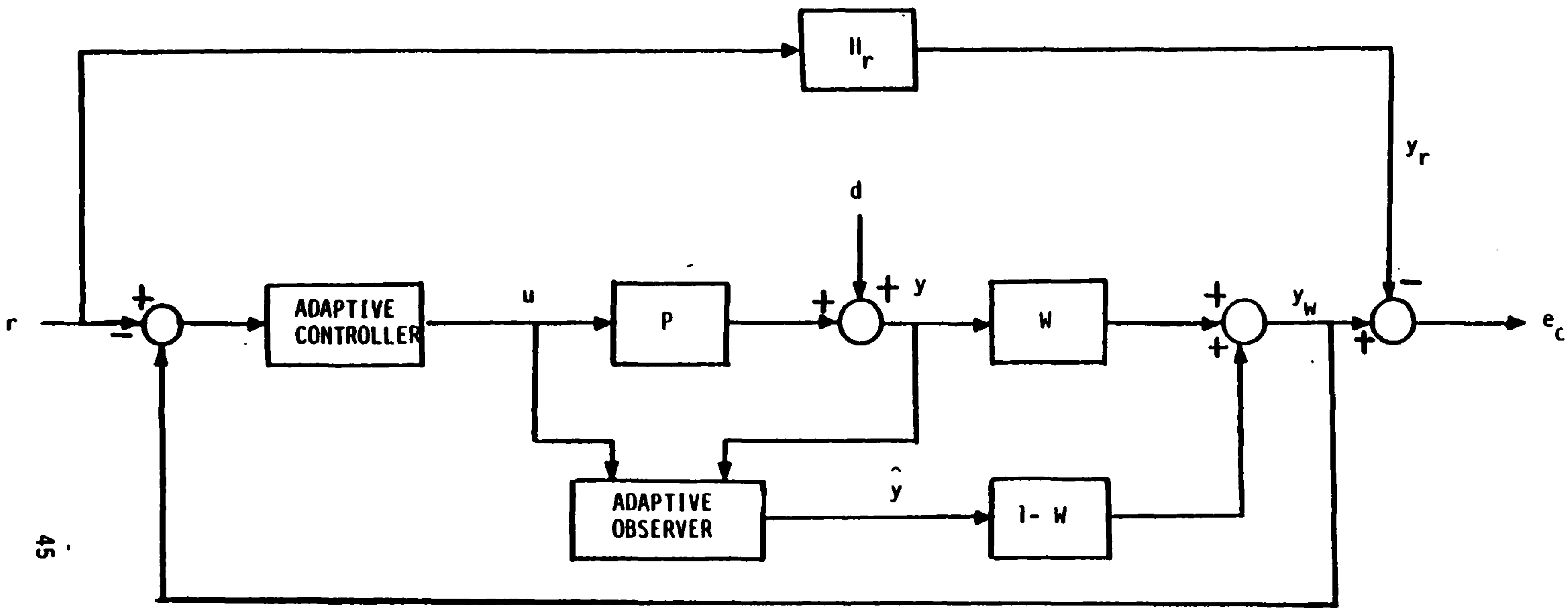


Figure 7.3 Adaptive SPR Compensator

8. CONCLUSIONS

This paper has presented an input/output view of multivariable adaptive control for uncertain linear time invariant plants. The essence of the results are captured in Theorems 1A and 2B which provide conditions that guarantee global stability. Corollary 1 also give specific L_2 and L_∞ bounds on significant signals in the adaptive control system. These bounds, for example, can be used to guarantee that the adaptive system performs as well as a robust (non-adaptive) system using the same structure, but with fixed gains. By distinguishing between a tuned system and a robust system, we establish formulae which can be used to restrict the minimum performance improvement possible with the same control structure.

Although the stability results (Theorem 1A, 1B) are not entirely new (see e.g., [7],[8]), the input/output setting provides the means to directly determine the system robustness properties with respect to model error. The type of model error examined can arise from a variety of causes, such as unmodeled dynamics and reduced order modeling. It is very difficult to treat this type of "unstructured" dynamic model error by using Lyapunov theory, since the system order may not be known -- in fact, it may be infinite. Although infinite dimensional (distributed) systems were not considered here, Theorem 1 can be modified to include them, e.g., [26].

The structure of Theorems 1A and 1B require that a particular subsystem operator, denoted H_{ev}, is strictly positive real (SPR). This requirement is not unique to this presentation - passivity requirements, in one form or another, dominate proofs of global stability for practically all adaptive control systems, including recursive identification algorithms. Unfortunately, although $H_{ev} \in$ SPR is robust to model error (Lemma 4.1), the bound on the model error is too small to be of practical use. Even the most benign neglected dynamics violate the bound.

Although this paper is concerned with continuous-time systems, the theorems carry over virtually intact to discrete-time systems. This is a direct consequence of the portable nature of the input/output view. However, there is an important issue unique to discrete-time systems: plant uncertainty is critical to where performance is actually measured, which is in continuous-time, not at the sampled-data points. As a consequence, it may be necessary to map the discrete portions of the adaptive system (most likely the controller) into continuous-time, i.e., the L_2-gains of the discrete-time operators in the interconnection map, which are associated with the adaptive discrete-time controller, would be needed rather the discrete-time ℓ_2-gains.

Another area worth pursuing is the adaptive control of non-linear plants. The plant uncertainty description (2.11) does not exclude non-linear plants. Note that slowly drifting parameters in an otherwise perfectly known LTI plant could yield the same uncertainty description as a non-linear plant approximated by a parametric LTI model. All that is required is that there exists a (possibly) infinite dimensional LTI system which matches the input/output behavior of the plant for each possible input/output pair. Of course, if the plant is truly non-linear, then the tuned control is likely to be non-linear, which raises some very interesting issues for further research.

One final remark: the stability results presented here, as well as other known results, provide global stability. This is achieved by requiring $H_{ev} \in SPR$, a condition which is difficult to maintain in normal circumstances. On the other hand, this is a sufficient conditon; violation of which does not necessarily lead to instability. The simple example presented here in Figure 6.1-6.2, illustrates the point. Other examples of this phenomena abound, e.g., [12]. It would appear then, that a more valid approach to providing a system-theoretic setting for adaptive control is to develop local stability conditions, which, hopefully, do not require that $H_{ev} \in SPR$. Preliminary results on local stability support this hope, e.g., [33], [34].

ACKNOWLEDGEMENT

We wish to acknowledge many useful discussions on portions of this material with C.R. Johnson, Jr. (Cornell University), B.D.O. Anderson (Australian National University), C.E. Rohrs (Notre Dame University), and C.A. Desoer (University of California, Berkeley).

APPENDIX A

PROOF OF THEOREMS 1 AND 2

Preliminaries

The main ingredient in the proof is to show stability by means of passivity. Although there are many variations on this theme, a general result is given by the following.

Theorem A.1 ([21], [35]

Consider the feedback system of Figure A.1 below with causal operators G_1 and G_2 .

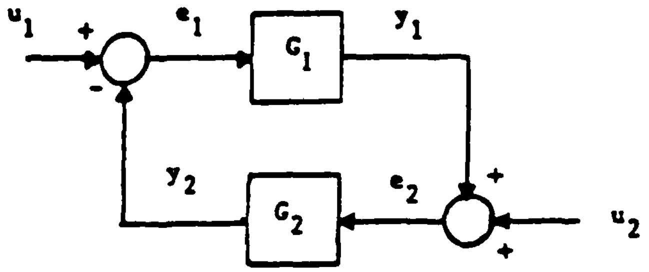


Figure A.1 Feedback System

Suppose there exists real constants ϵ_i, δ_i, α_i, $i=1,2$, such that

$$\langle x, G_i x\rangle_t > \epsilon_i \|x\|_{t2}^2 + \delta_i \|G_i x\|_{t2}^2 + \alpha_i, \ \forall t > 0, \ \forall\ x \in L_2[0,t] \tag{A.1}$$

for $i=1,2$. Then the following holds $\forall\ t > 0$,

$$(\epsilon_2+\delta_1)\|y_1\|_{t2}^2 + (\epsilon_1+\delta_2)\|y_2\|_{t2}^2 < \|y_1\|_{t2}(\|u_1\|_{t2} + 2|\epsilon_2|\cdot\|u_2\|_{t2})$$

$$+ \|y_2\|_{t2}(\|u_2\|_{t2} + 2|\epsilon_1|\cdot\|u_1\|_{t2}) + |\epsilon_1|\cdot\|u_1\|_{t2}^2 + |\epsilon_2| \cdot \|u_2\|_{t2}^2$$

$$+ |\alpha_1| + |\alpha_2| \tag{A.2}$$

Proofs of both theorems also rely on well known results for systems $H \varepsilon S_0^{n \times m}$. The results required here are summarized in the following.

<u>Theorem A-2</u> [see [19], Thm. 9, pg. 59]

Let $H \varepsilon S_0^{n \times m}$; then:

(i) If $u \varepsilon L_2^m$, then $y = Hu \varepsilon L_2^n \;\; L_\infty^n$, $\dot{y} \varepsilon L_2^n$, y is continuous, and $y(t) \to 0$ as $t \to \infty$.

(ii) If $u \varepsilon L_\infty^m$, then $y = Hu \varepsilon L_\infty^n$, $\dot{y} \varepsilon L_\infty^n$, and y is uniformly continuous.

(iii) If $u \varepsilon L_\infty^m$ and $u(t) \to$ constant $c \varepsilon R^m$ as $t \to \infty$, then $y(t) \to H(0)c$ exponentially as $t \to \infty$.

In order to simplify notation we drop the superstrict on L_p^n which indicates vector size.

We will establish Theorem 1A first. Some of the steps will be repeated for 1B. Also, without loss of generality, the matrix Γ in the adaptation law (3.15),(3.16) is set to identity. Corollary 1 is established as a by-product.

<u>Proof of Theorem 1A</u>

<u>Part (i)</u>

Identify G_1, G_2 in Figure A.1 with $e \to v$ and H_{ev} respectively. Also, let $u_1 = e_*$, $u_2 = 0$, $e_1 = e$, $y_1 = e_2 = v$, and $y_2 = H_{ev}v$. . Using adaptive law (3.15) we obtain,

$$\langle e,v\rangle_T = \langle e,Z'\theta\rangle_T = \langle Ze,\theta\rangle_T = \langle \dot{\theta}, \theta\rangle_T \tag{A.4}$$

$$= \frac{1}{2} \|\theta(T)\|^2 - \frac{1}{2} \|\theta(0)\|^2 \tag{A.5}$$

$$\geq - \frac{1}{2} \|\theta(0)\|^2 \tag{A.6}$$

Thus, using (A.1) gives,

$$\epsilon_1 = \delta_1 = 0,\ \alpha_1 = -\frac{1}{2}\|\theta(0)\|^2 \tag{A.7}$$

Since $G_2 = H_{ev} \in SPR_+$ by assumption, $\exists \mu, \gamma > 0$ such that $\forall\ x \in L_{2e}$, $\langle x, H_{ev}x\rangle_T > \mu\|x\|_{T2}^2$, $\|H_{ev}x\|_{T2} < \gamma\|x\|_{T2}$. Hence, from (A.1),

$$\epsilon_2 = \mu,\ \delta_2 = \alpha_2 = 0 \tag{A.8}$$

Using Lemma A.1, together with (A.4)-(A.8) gives,

$$\|v\|_{T2} < \frac{1}{2\mu}\left[\|e_*\|_{T2} + (\|e_*\|_{T2}^2 + 2\mu|\theta(0)|^2)^{1/2}\right] \tag{A.9}$$

$$\|e-e_*\|_{T2} < \gamma\|v\|_{T2} \tag{A.10}$$

$$|\theta(T)|^2 < |\theta(0)|^2 + 2\|e\|_{T2}\|v\|_{T2} \tag{A.11}$$

The bounds shown in (4.8) follow using the assumption $e_* \in L_2$. Hence, $e, v \in L_2$ and $\theta \in L_\infty$.

Having established that $v \in L_2$, Theorem A-2 $\Longrightarrow$ $\tilde{z} := z - z_* \in L_2 \cap L_\infty$, $\dot{\tilde{z}} \in L_2$, $\tilde{z} \to 0$, and $\tilde{z}$ is continuous. Since z_*, $\dot{z}_* \in L_\infty$ by assumption, it follows that $z \in L_\infty$ and $\dot{z} \in L_\infty$ ($\Longrightarrow$ z is uniformly continuous). Using $v = Z'\theta$ with $z, \theta' \in L_\infty \Longrightarrow v \in L_\infty$. Using $e = e_* - H_{ev}v$ with $e_* \in L_\infty$ and $H_{ev} \in S$ (by assumption), and $v \in L_\infty \Longrightarrow e \in L_\infty$. Hence, $\dot{\theta} = Ze \in L_\infty \Longrightarrow \theta$ is uniformly continuous $\Longrightarrow$ $v = Z'\theta$ is uniformly continuous (since z is) $\Longrightarrow$ $v \to 0$ since $v \in L_2$ is established. Using $v \to 0 \Longrightarrow e - e_* \to 0$, and since $e_* \to 0$ by assumption, $e \to 0$. Furthermore, $v \to 0 \Longrightarrow \tilde{z} \to 0$ exp. and $\dot{\theta} = Ze = \tilde{Z}e + Z_*e \to 0$, because $\tilde{z}$ and $e \to 0$. Using $\dot{v} = \dot{Z}'\theta + Z'\dot{\theta}$ with $\dot{z}, \theta, \dot{\theta} \in L_\infty \Longrightarrow \dot{v} \in L_\infty$. Hence, $e' = \dot{e}_* - H_{ev}\dot{v} \in L_\infty$, because $e_* \in L_\infty$ by assumption. Thus, $\ddot{\theta} = \dot{Z}e + Z\dot{e} \in L_\infty$. This establishes properties (i-a)-(i-d).

To show (i-e) consider (3.15) written as:

$$\dot{\theta} = -Z_* H_{ev} Z_*' \theta + w$$

$$w := -(Z_* H_{ev} \tilde{Z}' + \tilde{Z} H_{ev} Z_*' + \tilde{Z} H_{ev} \tilde{Z}')\theta \tag{A.12}$$

Since we have already established that $\tilde{z} \to 0$ exp. and $\theta \in L_\infty$, it follows that $w \to 0$ exp. Since $z_* \in PE$ by assumption (provided $e_* = 0$), $w \mapsto \theta$ is exp. stable by (2.15). Hence, $\theta \to 0$ exp. $\Rightarrow \dot{\theta}$, $v \to 0$ exp. $\Rightarrow e - e_* \to 0$ exp. This completes the proof of part (i) with adaptive law (3.15).

To show that (i-a)-(i-a) hold with adaptive law (3.16) requires showing that $G_1 : e \to v$ is passive. Consider the typical time interval,

$$I = \begin{cases} I_1 = \{t \in [t_0, t_1) \mid \|\hat{\theta}(t)\| < c\} \\ I_2 = \{t \in [t_1, t_2) \mid \|\hat{\theta}(t)\| > c > \max\|\theta_*\|\} \end{cases} \tag{A.13}$$

Hence,

$$\langle e,v\rangle_I = \langle e,v\rangle_{I_1} + \langle e,v\rangle_{I_2} \tag{A.14}$$

Thus,

$$\langle e,v\rangle_{I_1} = \langle \dot{\theta}, \theta\rangle_{I_1} = \frac{1}{2}\|\theta(t_1)\|^2 - \frac{1}{2}\|\theta(t_0)\|^2 \tag{A.15}$$

$$\langle e,v\rangle_{I_2} = \langle \dot{\theta} + (1 - \|\hat{\theta}\|/c)^2 \hat{\theta}, \theta\rangle_{I_2} \tag{A.16}$$

$$= \frac{1}{2}\|\theta(t_2)\|^2 - \frac{1}{2}\|\theta(t_1)\|^2 + (1 - \|\hat{\theta}\|/c)^2 \langle \hat{\theta}, \theta\rangle_{I_2} \tag{A.17}$$

$$> \frac{1}{2}\|\theta(t_2)\|^2 - \frac{1}{2}\|\theta(t_1)\|^2 \tag{A.18}$$

because $\langle \hat{\theta}, \theta\rangle_{I_2} > 0$ from,

$$\hat{\theta}(t)'\,\theta(t) = \hat{\theta}(t)'[\hat{\theta}(t)-\theta_*]$$
$$= \|\hat{\theta}(t)\|^2 - \hat{\theta}(t)'\theta_*$$
$$> \|\hat{\theta}(t)\|^2 - \|\hat{\theta}(t)\|c$$
$$= \|\hat{\theta}(t)\|(\|\hat{\theta}(t)\| - c) > 0, \qquad \forall t \in I_2 \,. \tag{A.19}$$

Thus,

$$\langle e,v\rangle_I > \frac{1}{2}\|\theta(t_2)\|^2 - \frac{1}{2}\|\theta(t_0)\|^2 \tag{A.20}$$

Repeating the above procedure recursively,we eventually conclude that $\langle e,v\rangle_T > -\frac{1}{2}\|\theta(0)\|^2$ as before (A.6), and hence, $G_f e \mapsto v$ is passive. The results in (i) now repeat for adaptive law (3.16). This completes the proof of part(i).

Proof of Theorem 1A,Part (ii)

Theorem 1A, Part (ii) is essentially an L_∞-stability result. The method of proof requires the notion of "exponential weighting" which is a means to obtain L_∞-stability of a system from the L_2-stability of an exponentially weighted version of the system (see e.g., [19], Chapter 9). We require the following:

Definition: Given a real number α define the exponential weighting operator by

$$x^\alpha(t) := \varepsilon^{\alpha t}x(t) \tag{A.21}$$

Consider the system $y = Gu$. An exp. weighted version of this system is denoted by $y^\alpha := G^\alpha u^\alpha$. Note that if G is a convolution operator with transfer function G(s) then G^α is also a convolution operator with transfer function $G(s-\alpha)$. Thus, the corresponding exponentialy weighted error system corresponding is described by

$$
\begin{aligned}
e^{\alpha} &= e^{\alpha}_{*} - H^{\alpha}_{ev}\, v^{\alpha} \\
z^{\alpha} &= z^{\alpha}_{*} - H^{\alpha}_{zv}\, v^{\alpha} \\
v^{\alpha} &= Z'\theta^{\alpha} \\
\dot{\theta}^{\alpha} &= \alpha\theta^{\alpha} + Ze^{\alpha} - \rho(\hat{\theta})\hat{\theta}^{\alpha}
\end{aligned}
\tag{A.22}
$$

where $\alpha > 0$ such that

$$
H^{\alpha}_{ev} \in SPR^{m}_{+} \text{ and } H^{\alpha}_{zv} \in \dot{S}^{k\times m}_{o}
\tag{A.23}
$$

Using Theorem A-1, identify G_1 with $e^{\alpha} \to v^{\alpha}$ and G_2 with H^{α}_{ev}. Note that it is always possible to find some $\alpha > 0$ such that (A.23) holds. We now examine the passivity of G_1: $e^{\alpha} \to v^{\alpha}$. Thus,

$$
\begin{aligned}
\langle e^{\alpha}, v^{\alpha}\rangle_T &= \langle e^{\alpha}, Z'\theta^{\alpha}\rangle_T = \langle Ze^{\alpha}, \theta^{\alpha}\rangle_T \\
&= \langle \theta^{\alpha}, \dot{\theta}^{\alpha} - \alpha\theta^{\alpha} + \rho(\hat{\theta})\hat{\theta}^{\alpha}\rangle_T \\
&= \frac{1}{2}\varepsilon^{2\alpha T}\|\theta(T)\|^2 - \frac{1}{2}\|\theta(0)\|^2 + \langle \rho(\hat{\theta})\hat{\theta}^{\alpha}, \theta^{\alpha}\rangle_T - \alpha\|\theta^{\alpha}\|^2_{T2} \\
&\geq \frac{1}{2}\varepsilon^{2\alpha T}\|\theta(T)\|^2 - \frac{1}{2}\|\theta(0)\|^2 - \alpha\|\theta^{\alpha}\|^2_{T2}
\end{aligned}
\tag{A.24}
$$

The last line follows from (A.19), hence, (A.24) holds with or without the retardation term in the adaptive law. At this point there are two possibilities: either $\theta \in L_{\infty}$ or $|\theta(t)| \to \infty$ as $t \to \infty$. If $\theta \in L_{\infty}$ then $\exists$ constant $c_o < \infty$ such that $\|\theta\|_{\infty} < c_o$. Then,

$$
\begin{aligned}
\langle e^{\alpha}, v^{\alpha}\rangle_T &\geq \frac{1}{2}\varepsilon^{2\alpha T}(\|\theta(T)\|^2\ c_o^2) - \frac{1}{2}\|\theta(o)\|^2 \\
&\geq -\frac{1}{2}\varepsilon^{2\alpha T}c_o^2 - \frac{1}{2}\|\theta(0)\|^2
\end{aligned}
\tag{A.25}
$$

If $|\theta(t)| \to \infty$ as $t \to \infty$ then it is always possible to select an arbitrarily large T such that $\|\theta(T)\| = \|\theta\|_{T\infty}$. Hence, for this T, (A.24) becomes,

$$
\begin{aligned}
\langle e^{\alpha}, v^{\alpha}\rangle_T &\geq \frac{1}{2}\varepsilon^{2\alpha T}(\|\theta(T)\|^2 - \|\theta\|^2_{T\infty}) - \frac{1}{2}\|\theta(0)\|^2 \\
&= -\frac{1}{2}\|\theta(0)\|^2
\end{aligned}
\tag{A.26}
$$

Thus, for some arbitrarily large T, (A.25) and (A.26) have the general form, i.e.,

$$\langle e^{\alpha}, v^{\alpha}\rangle_T \geq -c_1 \varepsilon^{2\alpha T} - c_2 := -c(\alpha T) \tag{A.27}$$

where c_1, c_2 are non-negative constants. Hence,

$$\varepsilon_1 = \delta_1, = 0, \ \alpha_1 = -c(\alpha T) \tag{A.28}$$

Since $G_2 = H^{\alpha}_{ev} \varepsilon SPR_+$, $\exists$ constants μ, $\gamma > 0$ such that

$$\begin{aligned} &\langle x, H^{\alpha}_{ev}\, x\rangle_T \geq \mu \| x\|^2_{T2} \\ &\| H^{\alpha}_{ev}\, x\|_{T2} \leq \gamma\, \| x\|_{T2} \end{aligned} \tag{A.29}$$

Then,

$$\varepsilon_2 = \mu, \ \delta_2 = \alpha_2 = 0 \tag{A.30}$$

Using (A.2), we get

$$\| v^{\alpha}\|_{T2} \leq \frac{1}{2\mu}[\| e^{\alpha}_{*}\|_{T2} + (\| e^{\alpha}_{*}\|^2_{T2} + 2\mu\ c(\alpha T))^{1/2}] \tag{A.31}$$

Since $e_* \varepsilon L_\infty$ by assumption,

$$\| e^{\alpha}_{*}\|_{T2} \leq \varepsilon^{\alpha T}(2\alpha)^{-1/2}\| e_*\|_\infty \tag{A.32}$$

Thus,

$$\| v^{\alpha}\|_{T2} \leq \frac{\varepsilon^{\alpha T}(2\alpha)^{-1/2}}{2\mu}[\| e_*\|_\infty + (\| e_*\|^2_\infty + 4\alpha\ \varepsilon^{-2\alpha T}\mu c(\alpha T))^{1/2}] \tag{A.33}$$

Since $H^{\alpha}_{zv} \varepsilon S^{kxm}_{0}$, we obtain

$$|\tilde{z}(T)| = |\int_0^T H_{zv}(T-\tau)v(\tau)d\tau| \tag{A.34}$$

$$= |e^{-\alpha T}\int_0^T H^{\alpha}_{zv}(T-\tau)v^{\alpha}(\tau)d\tau| \tag{A.35}$$

$$< e^{-\alpha T} \| H^{\alpha}_{zv}(\cdot) \|_1 \cdot \| v^{\alpha} \|_{T2} \tag{A.36}$$

where $H^{\alpha}_{zv}(t)$ is the impulse response matrix associated with H^{α}_{zv}. Substituting (A.33) and (A.27) into (A.36) and noting that $e^{-2\alpha T} c(\alpha T) < c_1 + c_2$, we obtain,

$$|\tilde{z}(T)| < \frac{(2\alpha)^{-1/2}}{\mu} \| H^{\alpha}_{zv}(\cdot) \|_1 \cdot \left[\| e_* \|_{\infty} + \left(\| e_* \|^2_{\infty} + 4\alpha\mu(c_1+c_2) \right)^{1/2} \right] \tag{A.37}$$

Since the right hand side is independent of T, and since T can be selected to be arbitrarily large, it follows that $z \in L_{\infty}$. Assuming there is no retardation or persistent excitation, this completes the proof of (ii-a) to (ii-d).

Assume now that $z \in PE$, which is a noncontradictory assumption since we have already shown that $z \in L_{\infty}$. Hence,

$$\dot{\theta} = -Z\, H_{ev}\, Z'\, \theta + Ze_* \tag{A.38}$$

Since $z \in PE$, $H_{ev} \in SPR_+$ and $z, \varepsilon_* \in L_{\infty}$, it follows from (2.15) that $(Ze_*, \theta(0)) \mapsto \theta$ is exp. stable, thus, $\theta, \dot{\theta} \in L_{\infty}$. The remaining results in (ii-e) follow immediately.

Suppose now that the adaptive law is given by (3.16). Then, we can write,

$$\begin{aligned} \dot{\hat{\theta}} &= Z\, e - \rho(\hat{\theta})\hat{\theta} = Z[e_* - H_{ev} Z'(\hat{\theta}-\theta_*)] - \rho(\hat{\theta})\hat{\theta} \\ &= w - Z\, H_{ev} Z'\, \hat{\theta} - \rho(\hat{\theta})\hat{\theta} \end{aligned} \tag{A.39}$$

where $w := Z\, e_* + Z\, H_{ev} Z'\theta_* \in L_{\infty}$, because $z, e_* \in L_{\infty}$. Consider the candidate Lyapunov function $V : t \mapsto \| \hat{\theta}(t) \|^2$. Hence,

$$\dot{V} = 2\, w'\hat{\theta} - \hat{\theta}'\, Z\, H_{ev} Z'\hat{\theta} - \rho(\hat{\theta})V \tag{A.40}$$

Suppose $\| \hat{\theta}(t) \| \to \infty$ as $t \to \infty$. Then there exists a time $T > 0$ such that

$\|\hat{\theta}(T)\| = \|\hat{\theta}\|_{T\infty} = V_T^{1/2} > c$. Hence,

$$\dot{V}_T < 2\|w\|_\infty V_T^{1/2} + \|z\|_\infty^2 \gamma_\infty(H_{ev})V_T - (1 - V_T^{1/2}/c)^2 V_T \tag{A.41}$$

Clearly, there exists a finite constant c_1 such that when $V_T > c_1$, $\dot{V}_T < 0$. Therefore, $\hat{\theta}$ can not grow beyond all bounds, and hence, $\hat{\theta} \varepsilon L_\infty$. So then is θ and $\dot{\theta}$, and again the result of (ii-e) follow. This completes the proof of Theorem 1A. Note that in this case we do not obtain specific bounds on e, because the proof proceeds by contradition.

Proof of Theorem 1B

Part (i)

Since $H_{ev} \varepsilon SPR_0$, there exists $q > 0$ such that $G_{ev} := (1 + qs)H_{ev} \varepsilon SPR_+$, and furthermore, $G_{ev}^{-1} \varepsilon S$. As a result we can write (3.14a) as,

$$e = -H_{ev}\, y,\; y = v - G_{ev}^{-1}(e_* + q\,\dot{e}_*) \tag{A.42}$$

Referring to Lemma A-1, let $G_1 : v \mapsto e$, $G_2 = H_{ev}$, $u_1 = 0$, and $u_2 = -G_{ev}^{-1}(e_* + q\dot{e}_*)$. Using (A.2) together with (A.42) and the passivity properties of H_{ev} gives,

$$\|e\|_{T2} < \frac{1}{2\mu}[\|u_2\|_{T2} + (\|u_2\|_{T2}^2 + 2\mu|\theta(0)|^2)^{1/2}] \tag{A.43}$$

$$|\theta(T)| < |\theta(0)| + 2\|e\|_{T2} \cdot \|u_2\|_{T2} \tag{A.44}$$

where μ is defined in (4.9a). Using (4.9b) gives, $\|u_2\|_{T2} < (1/k)\|e_* + q\dot{e}_*\|_{T2}$. This together with (A.43), (A.44) and the assumption $e_*, \dot{e}_* \varepsilon L_2$ gives the bounds shown in (4.9). Hence, $e \varepsilon L_2, \theta \varepsilon L_\infty$. However, we can not conclude that $v \varepsilon L_2$ as in Theorem 1A, part (i). From (A.42), we can conclude that $(1 + qs)^{-1} v \varepsilon L_2$. Since $G_{zv} := (1 + qs)H_{zv} \varepsilon S_{o\ell}$, it follows from Lemma A-2 that $z := z - z_* \varepsilon L_2 \cap L_\infty$, $\dot{z} \varepsilon L_2$ and $z \to 0$. Repeated use of Lemma A-2 and the error equations (3.14) gives the results (i-a) - (i-d). (i-e) follows from the arguments in the proof of Theorem 1A, part (i).

Part (ii)

The proof is entirely analgous to that of Theorem 1A, part (ii), where again we use exponential weighting.

APPENDIX B

PROOF OF LEMMA 5.1

The proof utilizes the following known results:

Definition: Let J denote a subset of S, consisting of functions in S whose inverse is also in S.

Fact [29]: If G is any scalar transfer function in R(s), then G has a coprime factorization in S, i.e., there exists N, D, A, and B in S such that $G = N/D$ and $AN + BD = 1$.

Lemma B-1: Consider the tuned adaptive system of Figure 5.2. Let $P_* \in R_0(s)$ and $C_* \in R_0(s)$ have coprime factorizations in S given by $P_* = N_p/D_p$ and $C_* = N_c/D_c$, respectively. Then, the elements of the transfer matrix from (r,d) into (e_*, z_*, y, u) all belong to S, if:

$$\text{(i)} \quad Q := D_pD_c + N_pN_c \in J, \qquad \text{(from [29])} \tag{B.1}$$

and

$$\text{(ii)} \quad \delta(\omega)|T_*(j\omega)| < 1, \ \forall\, \omega \in R, \qquad \text{(from [16])}$$

where

$$T_* := N_pN_c/Q := P_*C_*(1+P_*C_*)^{-1} \tag{B.2}$$

Using the definition of Q we can write H_{ev} and H_{zv} from (5.5) as,

$$H_{ev} = N_pQ^{-1}(1+\Delta)(1+\Delta T_*)^{-1} \tag{B.3}$$

$$H_{zv} = \begin{bmatrix} FD_pQ^{-1}(1+\Delta T_*)^{-1} \\ FN_pQ^{-1}(1+\Delta)(1+\Delta T_*)^{-1} \end{bmatrix} . \tag{B.4}$$

From the definition of K_* (5.4b), we also obtain

$$Q = N_p K_*^{-1} \tag{B.3}$$

<u>Proof of Lemma 5.1</u>

We first show that (i), (ii), and (iv) => $Q \varepsilon J$. Let $P_* = N_p/D_p$ be a coprime factorization of $P*$ such that rel deg $D_p(s) = 0$. Since (i) => rel deg $P_*(s) = 1$, it follows that rel deg $N_p(s) = 1$. Moreover, (iv) => rel deg $K_*(s) = 1$, and that $K_1(s)$ and $K_2(s)$ are stable. This, together with (ii) and (B.3) establishes that $Q \varepsilon J$.

$H_{zv} \varepsilon S_o$ follows immediately by inspection of (B.2), since: $F \varepsilon S_o$ by assumption; D_p, $N_p \varepsilon S$; $Q \varepsilon J$; $\Delta \varepsilon S$ by assumption (vi); and finally (vi) => (ii) of Lemma B-1 => $(1+\Delta T_*)^{-1} \varepsilon S$.

Conditions (iv) and (vi) => $H_{ev} \varepsilon SPR_o$. This follows from Lemma 4.1 by letting $\bar{H}_{ev} = K_*$ and letting $1 + \tilde{H}_{ev} = (1+\Delta)(1+\Delta T_*)^{-1}$. Thus, (4.4a) is satisfied since $K_* \varepsilon SPR_o$ from (iv). Also, from (4.4b),

$$k(\omega) = |\tilde{H}_{ev}(j\omega)| = |\Delta(j\omega)S_*(j\omega)[1-\Delta(j\omega)T_*(j\omega)]^{-1}| \tag{B.4}$$

$$< \frac{\delta(\omega)|S_*(j\omega)|}{1-\delta(\omega)|T_*(j\omega)|} < \bar{k}(\omega) = \eta(\omega) \tag{B.5}$$

The last inequality comes from conditions (vi) and the definition of $\bar{k}(\omega)$ from (4.4b).

The final step in the proof of Lemma 5.1 is to show that there are a sufficient number of parameters in θ_* to insure a solution exists. This is guaranteed by satisfaction of condition (v). To see this combine (B.3) with the definition of Q from (B.1) to get

$$Q := N_c N_p + D_p D_c = N_p K_*^{-1} \tag{B.6}$$

From (5.2), let $N_c = A_{*2}/L$ and $D_c = 1 + A_{*1}/L$ be a coprime factorization of $C*$, and let $N_p = g\, N*/L$ and $D_p = 1 + D_*/L$ be a coprime factorization of P_*, where P_* is as defined in (i). With $K*$ given by (iv), (B.6) becomes the polynomial equation,

$$A_{*1}\, K_1 D_* + A_{*2} K_1 N_* = L(K_2 N_* - K_1 D_*) \tag{B.7}$$

Since $\deg(K_2 N_*) = \deg(K_1 D_*)$ and K_1, K_2, N_*, and D_* are all monic, it follows that $\deg[L(K_2 N_* - K_1 D_*] = \deg(L) + \deg(K_1) + \deg(D_*) - 1$. Then, using known results on polynomial equations, e.g. [30], it can be shown that (v) implies that (B.7) has a solution (A_{*1}, A_{*2}).

REFERENCES

[1] R. Bellman, Adaptive Control Processes: A Guided Tour, Princeton, 1961.

[2] P. C. Parks, "Liapunov Redesign of Model Reference Adaptive Control Systems," IEEE Trans. Aut. Contr., vol. AC-11, July 1966, pp. 362-367.

[3] I. D. Landau, "A Hyperstability Criterion for Model Reference Adaptive Control Systems," IEEE Trans. Aut. Contr., Vol. AC-14, No. 5, Oct. 1969, pp. 552-555.

[4] R. V. Monopoli, "Model Reference Adaptive Control with an Augumented Error Signal," IEEE Trans. Aut. Contr., Vol. Ac-19, Oct. 1974, pp. 474-484.

[5] K. J. Astrom and B. Wittenmark, "On Self Tuning Regulators," Automatica, Vol. 9, pp. 185-199, 1973.

[6] L. Ljung, "On Positive Real Transfer Functions and the Convergence of Some Recursive Schemes," IEEE Trans. Aut. Contr., Vol. AC-22, No. 4, Aug. 1977, pp. 539-551.

[7] K.S. Narendra, L.S. Valavani, "Direct and Indirect Model Reference Adaptive Control," Automatica, Vol. 15, pp. 653-664, 1979.

[8] Y.D. Landau, Adaptive Control: The Model Reference Approach, Dekker, NY, 1979.

[9] G.C. Goodwin, P.J. Ramadge, and P.E. Caines, "Discrete Time Mutivariable Adaptive Control," IEEE Trans. Aut. Contr. , Vol. AC-25, June 1980, pp. 449-456.

[10] G. Kreisselmeier, "On Adaptive State Regulation," Vol. AC-27, No. 1, Feb. 1982, pp. 3-17.

[11] C.E. Rohrs, L. Valavani, M.Athans, and G. Stein, "Analytical Verification of Undesirble Properties of Direct Model Reference Adaptive Contrl Algorithms," Proc. 20th IEEE CDC, pp. 1272-1284, Dec. 1981,San Diego, CA.

[12] ________, "Robustness of Adaptive Control Algorithms in the Presence of Unmodeled Dynamics," Proc. 21st IEEE Conference on Decision and Control, Orlando, Florida, Dec. 1982, pp.

[13] B.D.O. Anderson and C.R. Johnson, Jr., "On Reduced Order Adaptive Output Error Identification and Adaptive IIR Filtering," IEEE Trans. Aut. Contr., Vol. AC-27, No.4, pp. 927-932, Aug. 1982.

[14] I. Horowitz, Synthesis of Feedback Systems, NY, Academic, 1963.

[15] M.G. Safonov, Stability Robustness of Multivariable Feedback Systems, MIT Press, 1980.

[16] J.C. Doyle and G. Stein, "Multivariable Feedback Design: Concepts for a

Modern/Classical Synthesis," IEEE Trans. Aut. Contr., Vol. AC-26, No. 1, pp. 4-17, Feb. 1981.

[17] G. Zames and B. A. Francis, "A New Approach to Classical Frequency Methods: Feedback and Minimax Sensitivity," Proc. 20th CDC, pp. 867-874, Dec. 1981, San Diego, CA.

[18] G. Zames, "On the Input-Output Stability of Time-Varying Nonlinear Feedback Systems," IEEE Trans. on Aut. Contr., Part I: Vol. AC-11, No. 2, pp. 228-238, April 1966; Part II: Vol. AC-11, No. 3, pp. 465-476, July 1966.

[19] C.A. Desoer, M. Vidyasager, Feedback Systems: Input-Output Properties, Academic Press, New York, 1975.

[20] B.D.O. Anderson, "A Simplified Viewpoint of Hyperstability," IEEE Trans. Aut. Contr., Vol. AC-13, pp. 292-294, June, 1968.

[21] D.J. Hill and P.J. Moylan, "Dissipative Dynamical Systems: Basic Input-Output and State Properties," J. Franklin Inst., Vol. 309, No. 5, 1980.

[22] F. Donati and M. Vallauri, "Guaranteed Control of Almost Linear Plants," IEEE Trans. Aut. Contr, Vol. AC-29, No. 1, January 1984.

[23] M. Vidyasager, H. Schneider, and B.A. Francis, "Algebraic and Topological Aspects of Feedback Stabilization," IEEE Trans. Aut. Contr., Vol. AC-27, No.4, pp. 880-894, August 1982.

[24] G. Kreisselmeier and K.S. Narendra, "Stable Model Reference Adaptive Control in the Presence of Bounded Disturbances," IEEE Trans. Aut. Contr, Vol. AC-27, No. 6, pp 1169-1175, Dec 1982.

[25] M.G. Safonov, "Tight Bounds on the Response of Multivariable Systems with Component Uncertainty," Proc. Allerton Conf. on Comm., Contr., and Comput., Monticello, Ill, October, 1978.

[26] I.P. Landau, "Elimination of the Real Positivity Condition in the Design of Parallel MRAS," IEEE Trans. Aut. Contr., Vol. AC-23, No. 6, pp. 1015-1020, Dec. 1978.

[27] B. Egardt, Stability of Adaptive Controllers, Springer-Verlag, 1979.

[28] R.L. Kosut and B. Friedlander, "Performance Robustness Properties of Adaptive Control Systems," 21st IEEE Conference on Dec. and Control, pp 18-23, Orlando, Florida, Dec. 1982.

[29] C.A. Desoer, R.W. Liu, J. Murray, and R. Saeks, "Feedback System Design: The Fractional Representation Approach to Analysis and Synthesis", IEEE Trans. Aut. Contr., Vol. AC-25, No. 3, pp 399-412, June 1980.

[30] V. Kucera, Discrete Linear Control: The Polynomial Equation Approach, Wiley-Interscience, NY.

[31] B.D.O. Anderson, "Exponential Stability of Linear Equations Arising in

[31] B.D.O. Anderson, "Exponential Stability of Linear Equations Arising in Adaptive Identification," IEEE Trans. Aut. Contr., Vol. AC-22, pp. 83-88, February 1977.

[32] B. Peterson and K.S. Narendra, "Bounded Error Adaptive Control," IEEE Trans. Aut. Contr., Vol. AC-27, December 1982.

[33] P. Ioannou and P. Kokotovic, Adaptive Systems with Reduced Models, Springer-Verlag, 1983.

[34] R.L. Kosut and B.D.O. Anderson, "Robust Adaptive Control: Conditions for Local Stability," IEEE Trans. Aut. Contr., submitted, and Proc. 23rd CDC, Las Vegas, Nevada, December 1984.

[35] M. Vidyasager, Input-Output Analysis of Large-Scale Interconnected Systems, Springer-Verlag, 1981.

APPENDIX B

AN EFFICIENT ALGORITHM FOR OUTPUT ERROR MODEL REDUCTION

INT. J. CONTROL, 1984, VOL. 39, NO. 1, 95–113

An efficient algorithm for output-error model reduction†

BOAZ PORAT‡§ and BENJAMIN FRIEDLANDER‡

A new algorithm is presented for reduced-order modelling of linear discrete-time systems, using an output-error criterion. A closed form expression is developed for the gradient of the cost function with respect to the model parameters. A computationally efficient algorithm for computing this gradient is derived. A Fletcher–Powell optimization procedure utilizes the gradient vector to compute the reduced-order model parameters. A special initialization procedure is proposed, and the stability of the reduced-order system is monitored. The performance of the algorithm is illustrated by some numerical examples.

1. Introduction

The problem of mathematical modelling of physical phenomena arises in many scientific disciplines. An important aspect of modelling is the conversion of complex models into simpler ones. It is usually desirable to use models that are as simple as possible yet still capable of capturing the salient features of the underlying phenomena. Model simplification leads to savings in computational requirements and hardware costs and facilitates the analysis and understanding of complex problems. In this paper we consider a technique for the reduced-order modelling of linear discrete-time systems.

The problem of reduced-order modelling (sometimes called rational approximation on the unit circle) can be defined as follows: let $g^0(z)$ be a rational Nth-order transfer function

$$g^0(z)=\frac{b^0(z)}{a^0(z)}=\frac{b_1{}^0 z^{-1}+\ \dots\ +b_N{}^0 z^{-N}}{1+a_1{}^0 z^{-1}+\ \dots\ +a_N{}^0 z^{-N}} \tag{1}$$

where the polynomial $a(z)$ is assumed to be stable, i.e. to have all its roots strictly inside the unit circle. Let

$$g(z)=\frac{b(z)}{a(z)}=\frac{b_1 z^{-1}+\ \dots\ +b_n z^{-n}}{1+a_1 z^{-1}+\ \dots\ +a_n z^{-n}} \tag{2}$$

be an nth-order approximation to $g^0(z)$ (where $n<N$), in the sense that $g(z)$ is ' close ' to $g^0(z)$ under some criterion. The criterion used in this paper is the L_2 norm of the difference on the unit circle, i.e.

$$\begin{aligned} V&=\frac{1}{2\pi}\int_{-\pi}^{\pi}|g^0(\exp(j\omega))-g(\exp(j\omega))|^2\,d\omega \\ &=\frac{1}{2\pi}\int_{-\pi}^{\pi}\left|\frac{b^0(\exp(j\omega))}{a^0(\exp(j\omega))}-\frac{b(\exp(j\omega))}{a(\exp(j\omega))}\right|^2 d\omega \end{aligned} \tag{3}$$

Received 21 April 1983.

† This work was supported by the Air Force Office of Scientific Research under Contract No. F4920-81-C-0051.

‡ Systems Control Technology Inc., 1801 Page Mill Road, Palo Alto, California 94304, U.S.A.

§ Present address: Department of Electrical Engineering, Technion, Israel.

D

Using Parseval's theorem, the criterion can be alternatively specified in the time domain. Let

$$g^0(z) = \sum_{i=1}^{\infty} g_i^0 z^{-i}; \quad g(z) = \sum_{i=1}^{\infty} g_i z^{-i} \tag{4}$$

Then

$$V = \sum_{i=1}^{\infty} (g_i^0 - g_i)^2 \tag{5}$$

Minimization of V as defined by (3) or (5) over all possible parameter values $\{b_i, a_i : 1 \leqslant i \leqslant n\}$, will determine the optimal reduced-order model $g(z)$. We will refer to this procedure as the *output-error* method.

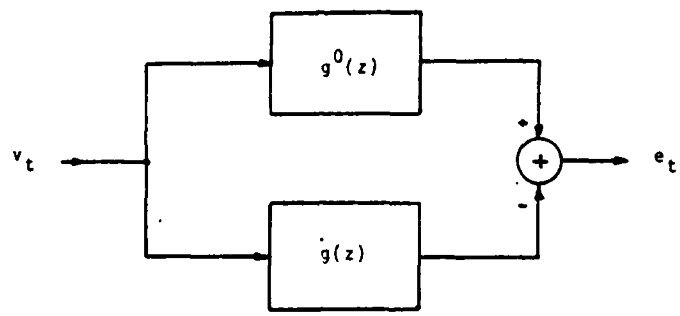


Figure 1. A model for the output error method.

The name 'output-error' comes from the system identification literature (Landau 1979). Consider the problem depicted in Fig. 1: two systems $g^0(z)$ (the real system) and $g(z)$ (the model to be estimated) with a common input process v_t, a unit-variance white noise process. It is desired to estimate the parameters of $g(z)$ so that the mean-square error between the outputs of the two systems $E\{e_t^2\}$ will be minimized. It is a straightforward matter to check that the mean-square error criterion is identical to V as defined earlier.

The output-error criterion seems to be a good candidate in many applications. It uses a physically meaningful error criterion and leads to satisfactory performance in the context of estimation and control problems. The main difficulties with this method are related to the computation of the reduced-order model. First, the error function is a non-quadratic function of the model parameters $(a(z))$. Therefore the minimization of this function involves a non-linear optimization procedure. Such procedures are often complicated and computationally expensive, especially for high-order systems. Second, the error function V will generally have multiple local minima, making it difficult to reach the global minimum.

A number of model-reduction algorithms based on the output error have been proposed, mainly in the context of filter design. Sanathanan and Koerner (1963) have proposed an iterative procedure in which a *conditional output error* is minimized at each stage, where the conditioning is on the denominator polynomial computed at the previous stage. Steiglitz and McBride (1965) used a similar idea in a system identification context. Deczky (1972) proposed a technique for minimizing the p-norm of magnitude error

$$\int (|g(\exp(j\omega))| - |g^0(\exp(j\omega))|)^p \, d\omega$$

and the p-norm of the phase error

$$\int |\text{phase}(g(\exp(j\omega))) - \text{phase}(g^0(\exp(j\omega)))|^p \, d\omega$$

The reduced transfer function $g(z)$ is modelled as a cascade connection of second-order filters. This procedure is extensively used for filter design. Aplevitch (1973) gave a gradient algorithm based on a state-space formulation. Recently, Yahagi (1981) proposed a gradient algorithm for minimizing the output error with respect to a model specified by a finite number of impulse response terms.

A number of alternative procedures have been proposed in the literature, apparently stimulated by the difficulties in computing the output-error reduced-order model parameters. Perhaps the most popular of these is the so-called *equation-error method* which uses an error function of the form

$$\begin{aligned} V &= \frac{1}{2\pi} \int_{-\pi}^{\pi} |a(\exp(j\omega))\, g^0(\exp(j\omega)) - b(\exp(j\omega))|^2 \, d\omega \\ &= \frac{1}{2\pi} \int_{-\pi}^{\pi} |a(\exp(j\omega))|^2 |g^0(\exp(j\omega)) - g(\exp(j\omega))|^2 \, d\omega \end{aligned} \tag{6}$$

Note that this cost function involves a filtered version of the output error. The equation-error method has the advantage of being quadratic in both the $a(z)$ and the $b(z)$ coefficients ; hence the minimization procedure is fairly straightforward. On the other hand, the error function tends to put a small weight on frequencies where the magnitude of the response is large, yielding poor approximations for systems with poles near the unit circle. Even more problematic is the fact that $a(z)$ resulting from minimizing V is not guaranteed to be stable.

Many model-reduction methods that are not based on the output-error technique have been proposed in the literature. We mention in particular the relatively recent development of the balanced realization method (Moore 1978) and the optimal Hankel-norm method (Kung 1980). Other well-established techniques include : dominant mode approximation, aggregation, singular perturbation, Routh approximation and Padé approximation. Here we consider only the output-error method, which appears to work well in various control and signal processing applications.

In this paper we present a new algorithm for the direct minimization of the output error function (3), (5). Through a detailed analysis of the error function V we were able to develop a closed-form expression for the gradient vector (i.e. the derivatives of $V(g)$ with respect to the parameters a_i, b_i). This gradient vector is then used in a Fletcher–Powell minimization algorithm to compute the parameters of the reduced-order model. Using some facts from the theory of discrete Lyapunov equations and Toeplitz matrices we were able to develop an efficient algorithm for computing the gradient vector, requiring of the order of N^2 multiplications and additions. This seems to be by far more efficient than any other existing schemes for computing the gradient. A special initialization procedure based on some properties of orthogonal polynomials is proposed. This procedure seems to provide a good starting point for the subsequent minimization algorithm, leading with high probability to the global minimum. A unique feature of the complete algorithm is that it guarantees stability of the reduced-order model (i.e. of $a(z)$) at each step.

We believe that the technique proposed in this paper provides for the first time a satisfactory solution to the output-error reduced-order modelling problem. The algorithm has a number of properties that distinguish it from previous attempts in this direction : (i) exact closed-form computation of the gradient ; (ii) computational efficiency ; (iii) improved initialization ; and (iv) guaranteed stability of the model. These features make it a viable and practically implementable technique. Our limited computational experience with the algorithm has been very favourable.

The outline of the paper is as follows. In § 2 we derive the closed-form expression for the gradient. In § 3 we discuss the implementation of the method, in particular the efficient computation of the gradient. In § 4 we extend the method to *multivariable* discrete systems. In § 5 we illustrate the performance of the algorithm with some examples.

2. Computation of the gradient vector

In this section we derive explicit expressions for the cost function V and its gradient vector with respect to the coefficients of the polynomials $a(z)$ and $b(z)$. We first express the cost function in terms of three matrices, each of which satisfying a certain matrix Lyapunov equation. These Lyapunov equations are shown to admit closed-form solutions, involving differences of products of triangular Toeplitz matrices. Then we use these expressions to derive a formula for the gradient vector.

2.1. *The cost function*

Let $e(z)$ be the z-transform of the error between the impulse response of the given transfer function and that of the reduced-order approximate model, i.e.

$$e(z)=\frac{b^0(z)}{a^0(z)}-\frac{b(z)}{a(z)} \tag{7}$$

Let $\{h_i^0,\ 0\leqslant i<\infty\}$ and $\{h_i,\ 0\leqslant i<\infty\}$ be the impulse-response sequences of $1/a^0(z)$ and $1/a(z)$ respectively, i.e.

$$\frac{1}{a^0(z)}=\sum_{i=0}^{\infty} h_i^0 z^{-i}\ ; \quad \frac{1}{a(z)}=\sum_{i=0}^{\infty} h_i z^{-i} \tag{8}$$

Using these sequences, we can write (7) in a matrix form. We shall use semi-open brackets to denote semi-infinite vectors and matrices, so that

$$\begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ \vdots \end{bmatrix}=\begin{bmatrix} h_0^0 & & \\ h_1^0 & h_0^0 & \ddots \\ \vdots & \vdots & \\ h_{N-1}^0 & h_{N-2}^0 \cdots & h_0^0 \\ \vdots & & \vdots \end{bmatrix}\begin{bmatrix} b_1^0 \\ \vdots \\ b_N^0 \end{bmatrix}-\begin{bmatrix} h_0 & & \\ h_1 & h_0 & \ddots \\ \vdots & \vdots & \\ h_{n-1} & h_{n-2} \cdots & h_0 \\ \vdots & & \vdots \end{bmatrix}\begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix} \tag{9}$$

or more compactly as

$$\mathbf{e} = H^0 \mathbf{b}^0 - H\mathbf{b} \tag{10}$$

where H^0 has dimensions $\infty \times N$ and H is $\infty \times n$. Assuming that both $a^0(z)$ and $a(z)$ are stable, all semi-infinite entities in (10) have finite norms. Therefore, we can express the squared norm of $\mathbf{e}$ as

$$V(\mathbf{a}, \mathbf{b}) \triangleq \mathbf{e}^{\mathrm{T}}\mathbf{e} = \mathbf{b}^{\mathrm{T}} R \mathbf{b} - 2\mathbf{b}^{0\mathrm{T}} Q \mathbf{b} + \mathbf{b}^{0\mathrm{T}} S \mathbf{b}^0 \tag{11}$$

where

$$R \triangleq H^{\mathrm{T}} H \,; \quad Q \triangleq H^{0\mathrm{T}} H \,; \quad S \triangleq H^{0\mathrm{T}} H^0 \tag{12}$$

The dimensions of R, Q and S are $n \times n$, $N \times n$ and $N \times N$ respectively. Let $p(z)$ be a monic polynomial of degree m, where

$$p(z) = 1 + p_1 z^{-1} + \dots + p_m z^{-m}$$

We define $C(p)$ as the companion matrix of $p(z)$, i.e.

$$C(p) = \begin{bmatrix} -p_1 & -p_2 \cdots -p_m \\ 1 & & 0 \\ & 1 & \\ & & \ddots & \\ 0 & & 1 \end{bmatrix} \tag{13}$$

The matrices R, Q and S can be characterized in terms of the companion matrices of $a(z)$ and $a^0(z)$ as follows.

Lemma 1

Each of the matrices R, Q and S is the unique solution of a matrix Lyapunov equation

$$R - C(a) R C^{\mathrm{T}}(a) = E_{n \times n} \tag{14 a}$$

$$Q - C(a^0) Q C^{\mathrm{T}}(a) = E_{N \times n} \tag{14 b}$$

$$S - C(a^0) S C^{\mathrm{T}}(a^0) = E_{N \times N} \tag{14 c}$$

$E_{k \times l}$ is a matrix of dimension $k \times l$ having 1 in its (1, 1)th entry and zeros elsewhere.

The proof is by a direct substitution using the defining relationships (8). Existence and uniqueness of the solutions are guaranteed by the stability of $a(z)$ and $a^0(z)$ (Lancaster 1969).

The next lemma gives an explicit expression for the solution to a matrix Lyapunov equation of the type appearing in Lemma 1.

Lemma 2

Let $p(z)$ and $q(z)$ be two stable monic polynomials of degree m, and let X satisfy the matrix Lyapunov equation

$$X - C(p) X C^{\mathrm{T}}(q) = E_{m \times m} \tag{15}$$

Then :

(1) X is a Toeplitz matrix (in general, non-symmetric) ;

(2) X^{-1} is equal to $Q_1 P_1^{\mathrm{T}} - P_2 Q_2^{\mathrm{T}}$, defined as

$$
\begin{bmatrix} 1 & & & \bigcirc \\ q_1 & 1 & & \\ \vdots & \ddots & \ddots & \\ q_{m-1} & \cdots & q_1 & 1 \end{bmatrix}
\begin{bmatrix} 1 & p_1 & \cdots & p_{m-1} \\ & \ddots & \ddots & \vdots \\ & & 1 & p_1 \\ \bigcirc & & & 1 \end{bmatrix}
$$

$$
-\begin{bmatrix} p_m & & & \bigcirc \\ p_{m-1} & p_m & & \\ \vdots & \ddots & \ddots & \\ p_1 & \cdots & p_{m-1} & p_m \end{bmatrix}
\begin{bmatrix} q_m & q_{m-1} & \cdots & q_1 \\ & \ddots & \ddots & \vdots \\ & & q_m & q_{m-1} \\ \bigcirc & & & q_m \end{bmatrix} \tag{16}
$$

The lemma can be proved by rather tedious algebraic manipulations or, more easily, by using results from the theory of *bi-orthogonal polynomials on the unit circle*—see, for example, Kailath *et al.* (1978) for a detailed discussion.

To use Lemma 2 for eqn. (15 *b*), a slight modification of this equation is necessary, since $n = \deg a < \deg a^0 = N$. We redefine $a(z)$ as

$$
\tilde{a}(z) = 1 + a_1 z^{-1} + \ldots + a_n z^{-n} + 0z^{-n-1} + \ldots + 0z^{-N} \tag{17}
$$

The polynomial $\tilde{a}(z)$ is only formally different from $a(z)$, i.e. $\tilde{a}(z) = a(z)$ for all numerical values of z. Let $\tilde{Q}$ be the $N \times N$ matrix satisfying the matrix Lyapunov equation

$$
\tilde{Q} - C(a^0)\tilde{Q}C^{\mathrm{T}}(\tilde{a}) = E_{N \times N} \tag{18}
$$

Then it can be verified that $\tilde{Q}$ and Q coincide in their n leftmost columns, or in other words $\tilde{Q}$ is a Toeplitz extension of Q to a square matrix.

2.2. *The gradient*

We now have all the necessary relationships for computing $\partial V/\partial \mathbf{a}$ and $\partial V/\partial \mathbf{b}$. The computation proceeds as follows.

$$
\frac{\partial V}{\partial a_i} = \mathbf{b}^{\mathrm{T}} \frac{\partial R}{\partial a_i} \mathbf{b} - 2\mathbf{b}^{0\mathrm{T}} \frac{\partial Q}{\partial a_i} \mathbf{b} = \mathbf{b}^{\mathrm{T}} \frac{\partial R}{\partial a_i} \mathbf{b} - 2\mathbf{b}^{0\mathrm{T}} \frac{\partial \tilde{Q}}{\partial a_i} \tilde{\mathbf{b}} \tag{19}
$$

where $\tilde{\mathbf{b}}$ is an extension of $\mathbf{b}$ to dimension N by adding $N - n$ zeros. Next recall the following relationships between derivatives of matrices and of their inverses

$$
\frac{\partial R}{\partial a_i} = -R \frac{\partial R^{-1}}{\partial a_i} R \, ; \quad \frac{\partial \tilde{Q}}{\partial a_i} = -\tilde{Q} \frac{\partial \tilde{Q}^{-1}}{\partial a_i} \tilde{Q} \tag{20}
$$

We now use Lemma 2 to express R^{-1} and $\tilde{Q}^{-1}$ as

$$
R^{-1} = A_1 A_1^{\mathrm{T}} - A_2 A_2^{\mathrm{T}} \tag{21 a}
$$

$$
\tilde{Q}^{-1} = \tilde{A}_1 A_1^{0\mathrm{T}} - A_2^0 \tilde{A}_2^{\mathrm{T}} \tag{21 b}
$$

where A_1, A_2, $\tilde{A}_1$, $\tilde{A}_2$, $A_1{}^0$ and $A_2{}^0$ are defined in an obvious manner. Differentiating with respect to a_i we get

$$\frac{\partial R^{-1}}{\partial a_i}=\frac{\partial A_1}{\partial a_i}A_1{}^{\mathrm{T}}+A_1\frac{\partial A_1{}^{\mathrm{T}}}{\partial a_i}-\frac{\partial A_2}{\partial a_i}A_2{}^{\mathrm{T}}-A_2\frac{\partial A_2{}^{\mathrm{T}}}{\partial a_i} \tag{22 a}$$

$$\frac{\partial \tilde{Q}^{-1}}{\partial a_i}=\frac{\partial \tilde{A}_1}{\partial a_i}A_1{}^{0\mathrm{T}}-A_2{}^0\frac{\partial \tilde{A}_2{}^{\mathrm{T}}}{\partial a_i} \tag{22 b}$$

Substituting (20) and (22) into (21), we get

$$\frac{\partial V}{\partial a_i}=-2\mathbf{b}^{\mathrm{T}}R\left(\frac{\partial A_1}{\partial a_i}A_1{}^{\mathrm{T}}-\frac{\partial A_2}{\partial a_i}A_2{}^{\mathrm{T}}\right)R\mathbf{b}$$
$$+2\mathbf{b}^{0\mathrm{T}}\tilde{Q}\frac{\partial \tilde{A}_1}{\partial a_i}A_1{}^0\tilde{Q}\tilde{\mathbf{b}}-2\tilde{\mathbf{b}}^{\mathrm{T}}\tilde{Q}^{\mathrm{T}}\frac{\partial \tilde{A}_2}{\partial a_i}A_2{}^{0\mathrm{T}}\tilde{Q}^{\mathrm{T}}\mathbf{b}^0 \tag{23}$$

It is convenient to introduce the following vectors

$$\left.\begin{array}{c} \mathbf{r}=\tilde{Q}^{\mathrm{T}}\mathbf{b}^0\,; \quad \mathbf{s}=\tilde{Q}\tilde{\mathbf{b}}=Q\mathbf{b}\,; \quad \mathbf{t}=R\mathbf{b} \\ \mathbf{v}_1=A_1{}^{\mathrm{T}}R\mathbf{b}\,; \quad \mathbf{v}_2=A_2{}^{\mathrm{T}}R\mathbf{b} \\ \mathbf{w}_1=A_1{}^0\tilde{Q}\tilde{b}\,; \quad \mathbf{w}_2=A_2{}^0\tilde{Q}^{\mathrm{T}}b^0 \end{array}\right\} \tag{24}$$

and then

$$\frac{\partial V}{\partial a_i}=2\left\{-\mathbf{t}^{\mathrm{T}}\frac{\partial A_1}{\partial a_i}\mathbf{v}_1+\mathbf{t}^{\mathrm{T}}\frac{\partial A_2}{\partial a_i}\mathbf{v}_2+\mathbf{r}^{\mathrm{T}}\frac{\partial \tilde{A}_1}{\partial a_i}\mathbf{w}_1-\mathbf{s}^{\mathrm{T}}\frac{\partial \tilde{A}_2}{\partial a_i}\mathbf{w}_2\right\} \tag{25}$$

Finally we need an expression for the derivatives of the matrices A_1, $\tilde{A}_1$, A_2 and $\tilde{A}_2$. Define an $m\times m$ matrix $Z_m{}^k$ by

$$(Z_m{}^k)_{ij}\triangleq\begin{cases}1\,; & i-j=k\\ 0\,; & \text{otherwise}\end{cases} \tag{26}$$

Then it can be easily verified that

$$\frac{\partial A_1}{\partial a_i}=Z_n{}^i\,; \quad \frac{\partial \tilde{A}_1}{\partial a_i}=Z_N{}^i\,; \quad \frac{\partial A_2}{\partial a_i}=Z_n{}^{n-i}\,; \quad \frac{\partial \tilde{A}_2}{\partial a_i}=Z_N{}^{N-i} \tag{27}$$

Using these expressions in (25) for $1\leqslant i\leqslant n$ and stacking the results in a column vector gives the desired expression for $\partial V/\partial\mathbf{a}$ as

$$\frac{\partial V}{\partial \mathbf{a}}=-2\begin{bmatrix} t_2 & \cdots & t_{n-1} & t_n & 0\\ \vdots & & & & \\ t_{n-1} & & & & \\ t_n & & & & \\ 0 & & & & \bigcirc \end{bmatrix}\mathbf{v}_1+2\begin{bmatrix} t_n & & & \bigcirc\\ t_{n-1} & \ddots & & \\ \vdots & & \ddots & \\ t_1 & \cdots & t_{n-1} & t_n \end{bmatrix}\mathbf{v}_2$$

$$+2\begin{bmatrix} r_2 & \cdots & r_N & 0 \\ \vdots & & & \\ \vdots & & & \bigcirc \\ r_{n+1}\cdots r_N & 0 & & \end{bmatrix}\mathbf{w}_1 - 2\begin{bmatrix} s_N & & & \bigcirc \\ s_{N-1} & \ddots & & \\ \vdots & \ddots & \ddots & \\ s_{N-n+1}\cdots s_{N-1} & s_N & 0\ldots 0 & \end{bmatrix}\mathbf{w}_2 \Bigg\} \qquad (28)$$

Finally, $\partial V/\partial \mathbf{b}$ can easily be computed to be

$$\frac{\partial V}{\partial \mathbf{b}} = 2R\mathbf{b} - 2Q^{\mathrm{T}}\mathbf{b}^0 = 2\mathbf{t} - 2\begin{bmatrix} r_1 \\ \vdots \\ r_n \end{bmatrix} \qquad (29)$$

In the next section we show how the computation can be implemented in an efficient manner and describe other components of the output error algorithm.

3. Implementation of the method

In this section we discuss several issues pertaining to the implementation of the proposed model-reduction procedure. First we consider the initialization problem and show how to obtain a stable reduced-order initial denominator polynomial. Then we discuss a fast computational procedure for the gradient vector. Next we consider the problem of stability monitoring and finally describe the gradient search procedure.

3.1. *Initialization*

The error surface corresponding to the output-error rational approximation method has, in general, several local minima. As any gradient method is only guaranteed to converge to a local minimum, it is imperative to choose an initial condition which is sufficiently close to the global minimum. Furthermore, it is necessary to choose a *stable* initial condition for $a(z)$ and to keep monitoring the stability as the search proceeds.

It has been suggested in the past to use an *equation-error* approximation of the given model as an initial condition (Sanathanan and Koerner 1963). Unfortunately, equation-error approximations are not guaranteed to be stable. A trivial stable initial condition is $a(z) = 1$, but this may be too far from the global minimum to guarantee convergence to this minimum.

We propose choosing the initial $a(z)$ as the nth-order *orthogonal polynomial* of the given $a^0(z)$ on the unit circle (Szegö 1967). In other words, $a(z)$ is defined as the unique solution of the *normal equation*

$$S_{n+1}\begin{bmatrix} 1 \\ a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} R_n^{\,e} \\ \vdots \\ \vdots \\ 0 \end{bmatrix} \qquad (30)$$

where S_{n+1} is the $(n+1)\times(n+1)$ principal minor of the Toeplitz matrix S defined in (12). The polynomial $a(z)$ thus defined has the following properties (Szegö 1967)

(1) it is guaranteed to be stable;

(2) it is an optimal nth-order approximation to $a^0(z)$ in the sense that it minimizes the *prediction* error

$$\frac{1}{2\pi}\int_{-\pi}^{\pi}\left|1-\frac{a(\exp(j\omega))}{a^0(\exp(j\omega))}\right|^2 d\omega$$

(3) it can be efficiently computed using Levinson's algorithm, requiring about n^2 operations (Kailath 1974).

These properties make $a(z)$ given by (30) an especially attractive choice for an initial condition, even though there is no guarantee this will lead to a global minimum.

As we shall now describe, the polynomial $b(z)$ is determined at each iteration by forcing $\partial V/\partial \mathbf{b}$ to zero. Hence no initial condition for $b(z)$ is required here.

3.2. *Efficient computation of the gradient*

As we have shown, both the cost function and the gradient vector require the solution of matrix Lyapunov equations of the form of (15). Specifically, (14 c) needs to be solved only once, while (14 a) and (18) need to be solved at each iteration. Consider (15) and the explicit formula (16) for the inverse of its solution. The matrix X can be obtained by a direct inversion of the right-hand side of (16), but this would require m^3 operations. A more efficient method for inverting this matrix is by the so-called *inverse Levinson algorithm*. This algorithm computes the UDL decomposition of X^{-1} in about $2m^2$ operations (an operation is defined here as one multiplication *and* one addition). As X is Toeplitz, it is fully determined by its first and last columns, and those can be readily computed from the L–D–U factors of X^{-1}. We give below a summary of the inverse Levinson algorithm, skipping the proof (see, for example, Vieira and Kailath (1977) for the symmetric case).

Step 1. Set

$$p_{m,i}=p_i,\quad q_{m,i}=q_i,\quad 0\leqslant i\leqslant m\ ;\quad d_m=1 \tag{31}$$

Step 2. For $i=m$ down to $i=1$, do

$$\rho_i=-p_{i,i},\quad \sigma_i=-q_{i,i},\quad \tau_i=\frac{1}{1-\rho_i\sigma_i} \tag{32 a}$$

$$\begin{bmatrix} 0 & p_{i-1,i-1} & \cdots & p_{i-1,1} & 1 \\ 1 & q_{i-1,1} & \cdots & q_{i-1,i-1} & 0 \end{bmatrix}$$

$$=\tau_i\begin{bmatrix} 1 & \rho_i \\ \sigma_i & 1 \end{bmatrix}\begin{bmatrix} p_{i,i} & p_{i,i-1} & \cdots & p_{i,1} & 1 \\ 1 & q_{i,1} & \cdots & q_{i,i-1} & q_{i,i} \end{bmatrix} \tag{32 b}$$

$$d_{i-1}=\tau_i d_i \tag{32 c}$$

Step 3. Solve the following equations for $\mathbf{x}_1$ and $\mathbf{x}_n$, the first and last columns of X respectively

$$\left.\begin{aligned}
&\begin{bmatrix} 1 & & & \\ p_{1,1} & 1 & & \bigcirc \\ \vdots & & \ddots & \\ p_{m-1,m-1} \cdots p_{m-1,1} & & & 1 \end{bmatrix} \mathbf{x}_1 = \begin{bmatrix} d_0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \\
&\begin{bmatrix} 1 & q_{m-1,1} & \cdots & q_{m-1,m-1} \\ & \ddots & & \\ & & 1 & q_{1,1} \\ \bigcirc & & & 1 \end{bmatrix} \mathbf{x}_n = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ d_0 \end{bmatrix}
\end{aligned}\right\} \tag{33}$$

In counting the number of operations for solving the two Lyapunov equations (14 *a*) and (18), we note the following:

(1) Equation (14 *a*) is symmetric; hence $p_{i,j}=q_{i,j}$, $\rho_i=\sigma_i$ and $\mathbf{x}_1$ is sufficient to determine X. Thus the total count of operation for this equation is $1{\cdot}5n^2$.

(2) Equation (18), while non-symmetric, is 'sparse' in the sense that the corresponding $q(z)$ polynomial is of degree n, rather than N. The total count of operations taking advantage of this sparseness is about $1{\cdot}5N^2+1{\cdot}5n^2$.

The gradient search procedure can be improved by forcing the component $\partial V/\partial \mathbf{b}$ to zero at each iteration. This has the effect of conditionally optimizing V with respect to $\mathbf{b}$ at each iteration (where tne conditioning is on the current value of $\mathbf{a}$), thus reducing the number of free parameters from $2n$ to n. As we see from (29), this achieved by setting $\mathbf{b}$ to

$$\mathbf{b}=R^{-1}Q^{\mathrm{T}}\mathbf{b}^0=R^{-1}\begin{bmatrix} r_1 \\ \vdots \\ r_n \end{bmatrix} \tag{34}$$

The computation of $\boldsymbol{r}$ takes N^2 operations and the solution of (34) takes $2n^2$ operations (e.g. by substituting for R^{-1} its expression given in (16)). The computation of $\mathbf{t}$ is then saved, since now $\mathbf{t}=[r_1 \ldots r_n]^{\mathrm{T}}$.

The total count of operations can now be computed to be about $4{\cdot}5N^2+2Nn+6n^2$. This can certainly be considered as efficient; by comparison, a more conventional solution (say of the form used by Yahagi (1981)) would require a number of operations proportional to nN^6.

3.3. *Stability monitoring*

Stability monitoring can be done, in principle, by solving for the roots of $a(z)$ and checking that they are all inside the unit circle. This, however, is an undesirable approach, since it significantly increases the computational burden if the degree of $a(z)$ is relatively large. Alternatively, the stability of $a(z)$ can be tested by the Schur–Cohn test (Jury 1974), which does not require a factorization of the polynomial. An interesting feature of our algorithm is that it, in fact, includes a stability test. The solution of (14 a) via the algorithm (31)–(33) is equivalent to the Schur–Cohn test in the symmetric case (Vieira and Kailath 1977). The condition

$$|\rho_i| < 1, \quad i = 1, \ldots, n$$

is necessary and sufficient for stability of the given polynomial. Thus our algorithm provides stability monitoring at no extra computational cost.

3.4. *The gradient search procedure*

Once a closed-form expression for the gradient is available, one of many existing gradient methods can be used for minimizing the error function $V(\mathbf{a}, \mathbf{b})$. We have chosen to use the Fletcher–Powell method (Luenberger 1973), known for its excellent convergence rate and relative ease of implementation. An important part of this method (as well as of virtually all gradient methods) is the *line search* procedure, namely, a search for a local minimum of the cost function at the direction used at each iteration, as a function of the step size. In our case, a certain difficulty occurs due to the fact that the $\mathbf{a}$ vector is constrained to be in the open set $\Omega = \{\mathbf{a} : a(z) \text{ is stable}\}$. On the boundary of this set the error $V(\mathbf{a}, \mathbf{b})$ approaches infinity, and it is not defined outside the set. Thus, whenever the poles of $a^0(z)$ are near the unit circle, great care is needed in performing the line search to stay within the permitted region Ω. We have found the *golden section* search procedure (Luenberger 1973) very useful in this case, since it uses the values of the cost function only for magnitude comparisons and makes no use of derivatives. Thus, by assigning very high cost to an unstable $\mathbf{a}$ (say near the value of the computer overflow) the line search can be forced to yield only stable values of $\mathbf{a}$.

4. Model reduction of multivariable systems

In this section we consider the case where both $g^0(z)$ and $g(z)$ are $p \times m$ transfer-function matrices, rather than scalars. A natural rational description of such matrices is in terms of so-called matrix fraction descriptions (MFD) (Kailath 1980). Formulating the output-error model-reduction problem in terms of MFDs leads to Lyapunov equations in *block-companion* form. Unfortunately, such equations do not appear to admit closed-form solutions of the type shown in (16). (It is worth noting that substituting matrices for scalars in (16) *does not* lead to a correct solution of (15) in the matrix case.) Therefore we have chosen not to use MFD representations here but take a different approach.

Let $a^0(z)$ be the characteristic polynomial of the system whose transfer-function matrix is $g^0(z)$. Then $g^0(z)$ can be written as

$$g^0(z) = \frac{B^0(z)}{a^0(z)} = \frac{B_1{}^0 z^{-1} + \ldots + B_N{}^0 z^{-N}}{1 + a_1{}^0 z^{-1} + \ldots + a_N{}^0 z^{-N}} \tag{35}$$

where $\{B_1{}^0, \ldots, B_N{}^0\}$ are $p \times m$ matrices. As before, $g^0(z)$ is assumed to be stable and strictly proper.

We wish to approximate $g^0(z)$ by the nth-order $p\times m$ transfer-function matrix

$$g(z)=\frac{B(z)}{a(z)}=\frac{B_1z^{-1}+\ \dots\ +B_nz^{-n}}{1+a_1z^{-1}+\ \dots\ +a_nz^{-n}} \tag{36}$$

We proceed as in § 2.1, expressing the error as a semi-infinite vector and then expressing V as the l_2 norm of this vector. It will be convenient to introduce the following notation: let $\mathbf{B}_i{}^0$ be a row vector of dimension pm, obtained from $B_i{}^0$ by stacking its rows in their natural order. $\mathbf{B}_i$ is defined in a similar manner. Using these definitions, we can express the error vector as

$$\begin{bmatrix} e_1 \\ e_2 \\ \vdots \end{bmatrix}=\begin{bmatrix} h_0{}^0 & & \bigcirc \\ \vdots & \ddots & \\ h_{N-1}{}^0 & \cdots & h_0{}^0 \\ \vdots & & \vdots \end{bmatrix}\begin{bmatrix} \mathbf{B}_1{}^0 \\ \vdots \\ \mathbf{B}_N{}^0 \end{bmatrix}-\begin{bmatrix} h_0 & & 0 \\ \vdots & \ddots & \\ h_{n-1} & \cdots & h_0 \\ \vdots & & \vdots \end{bmatrix}\begin{bmatrix} \mathbf{B}_1 \\ \vdots \\ \mathbf{B}_n \end{bmatrix} \tag{37}$$

or more compactly as

$$\mathbf{e}=H^0\mathbf{B}^0-H\mathbf{B} \tag{38}$$

The element e_i of $\mathbf{e}$ is now a row vector of dimension pm whose entries are the pm components of the impulse response at time i. The cost function V is now given by

$$V(\mathbf{a},\mathbf{B})\triangleq \operatorname{tr}\{\mathbf{e}^{\mathrm{T}}\mathbf{e}\}=\operatorname{tr}\{\mathbf{B}^{\mathrm{T}}R\mathbf{B}-2\mathbf{B}^{0\mathrm{T}}Q\mathbf{B}+\mathbf{B}^{0\mathrm{T}}S\mathbf{B}^0\} \tag{39}$$

where tr $\{\cdot\}$ denotes the trace operator and R, Q and S are defined as in § 2.1.

The rest of the procedure is similar to the one given in § 2, with some minor modifications. In particular:

(1) the matrices R, $\tilde{Q}$ and S are obtained exactly as before;
(2) the gradient component $\partial V/\partial\mathbf{B}$ is now given by

$$\partial V/\partial\mathbf{B}=2R\mathbf{B}-2Q^{\mathrm{T}}\mathbf{B}^0 \tag{40}$$

By setting

$$\mathbf{B}=R^{-1}Q^{\mathrm{T}}\mathbf{B}^0 \tag{41}$$

at each iteration, the dimensionality of the problem decreases from $(mp+1)n$ to n. This entails a considerable saving in the amount of operations; therefore it is highly recommended here.

(3) The expression (25) for $\partial V/\partial a_i$ basically remains the same, except for the need to take the trace of the right-hand side. This causes some difficulty in obtaining an expression of the form (28) for $\partial V/\partial\mathbf{a}$. However, such an expression is not really needed for practical implementation of the method. It is sufficient to compute the $\partial V/\partial a_i$ individually, and then stack them in a column vector of dimension n.

5. Numerical results

In this section we demonstrate the performance of the algorithm by some numerical examples. We have chosen to test transfer functions which have poles near the unit circle, as these cases are usually quite difficult to handle.

The first case uses the 8th-order transfer function whose denominator and numerator polynomials are

$$a^0(z) = 1 - 4{\cdot}082z^{-1} + 7{\cdot}2269z^{-2} - 6{\cdot}4408z^{-3} + 1{\cdot}8193z^{-4} + 2{\cdot}0443z^{-5} - 2{\cdot}4197z^{-6} + 1{\cdot}0356z^{-7} - 0{\cdot}1516z^{-8} \quad (42\,a)$$

$$b^0(z) = z^{-1} - 0{\cdot}357z^{-2} + 0{\cdot}2036z^{-3} - 0{\cdot}0848z^{-4} - 0{\cdot}0493z^{-5} - 0{\cdot}192z^{-6} \quad (42\,b)$$

The poles of this system are at $-0{\cdot}7$, $0{\cdot}9274 \pm j0{\cdot}3015$, $0{\cdot}7883 \pm j0{\cdot}5730$, $0{\cdot}3$, and $0{\cdot}3827 \pm j0{\cdot}7867$.

The order taken for the reduced model was $n = 4$. Figure 2 shows the impulse response of the approximation corresponding to the initial choice of $a(z)$ as described in § 3.1, against the impulse response of the full model. Figure 3 shows the corresponding frequency responses. The poles of the initial reduced-order model are at $0{\cdot}9205 \pm j0{\cdot}3213$ and $0{\cdot}7754 \pm j0{\cdot}6058$. We see

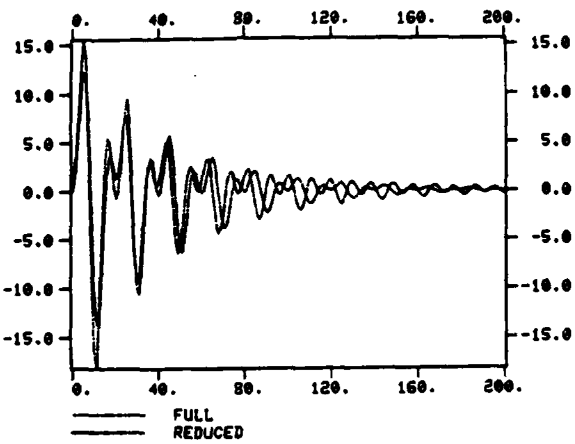


Figure 2. Example 1—Impulse response of initial approximation.

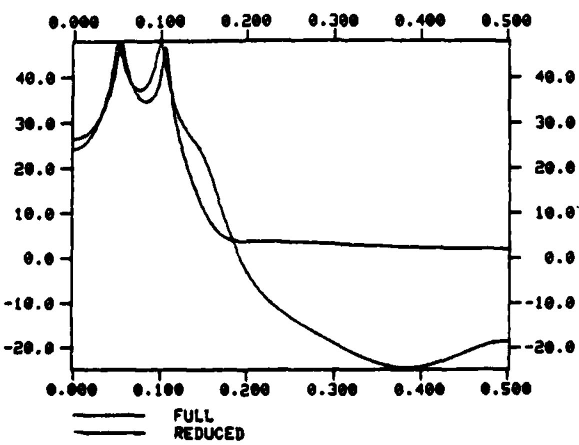


Figure 3. Example 1—Frequency response of initial approximation.

that, even though the poles are rather close to the dominant poles of the full-order model, this approximation is nevertheless poor. The squared error of this approximation is 728·6, or about 25% of the squared impulse response of the full model, which is 2865·2. Figures 4 and 5 show the impulse and frequency responses respectively of the approximation obtained after eight iterations of the algorithm. The transfer function of this approximation is

$$\frac{b(z)}{a(z)}=\frac{-0{\cdot}3945z^{-1}+4{\cdot}8318z^{-2}-4{\cdot}8169z^{-3}+0{\cdot}9268z^{-4}}{1-3{\cdot}4301z^{-1}+4{\cdot}8228z^{-2}-3{\cdot}2599z^{-3}+0{\cdot}9031z^{-4}} \tag{43}$$

The poles are at $0{\cdot}9271 \pm j0{\cdot}3099$ and $0{\cdot}7879 \pm j0{\cdot}5741$. The squared error is 8·48, i.e. about 1% of its initial value! The match of the impulse responses is excellent. The match of the frequency responses is excellent down to 20 dB and then starts deteriorating. This is an obvious result of the fact that the output-error method weighs the error linearly, while the frequency response is

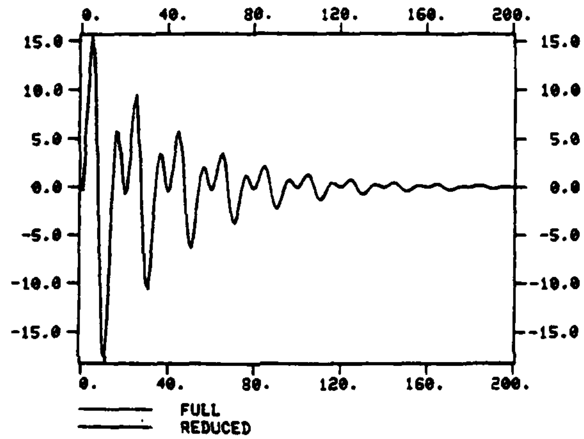


Figure 4. Example 1—Impulse response of final approximation.

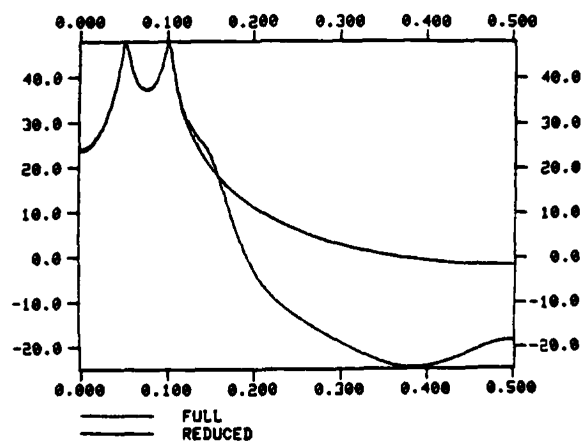


Figure 5. Example 1—Frequency response of final approximation.

shown on a logarithmic scale. Thus one should not expect a good match of the frequency responses at frequencies where the energy density is low.

The second example uses the 10th-order transfer function when denominator and numerator polynomials are

$$a^0(z) = 1 - 4\cdot4158z^{-1} + 8\cdot4582z^{-2} - 8\cdot2398z^{-3} + 2\cdot5658z^{-4} + 3\cdot2817z^{-5} - 4\cdot2475z^{-6} + 1\cdot5203z^{-7} + 0\cdot68197z^{-8} - 0\cdot83162z^{-9} + 0\cdot25151z^{-10} \quad (44\,a)$$

$$b^0(z) = -0\cdot31272z^{-1} - 0\cdot39268z^{-2} + 2\cdot3363z^{-3} - 2\cdot0318z^{-4} + 0\cdot80763z^{-5} - 0\cdot6027z^{-6} - 0\cdot86225z^{-7} + 1\cdot6256z^{-8} + 0\cdot03142z^{-9} - 0\cdot64554z^{-10} \quad (44\,b)$$

This example is taken from Kung and Lin (1980). The poles of this system are at $0\cdot9561 \pm j0\cdot2721$, $0\cdot3827 \pm j0\cdot7867$, $-0\cdot6349 \pm j0\cdot1454$, $0\cdot8711 \pm j0\cdot4517$, and $0\cdot6329 \pm j0\cdot6430$. The reduced order was taken to be $n = 6$.

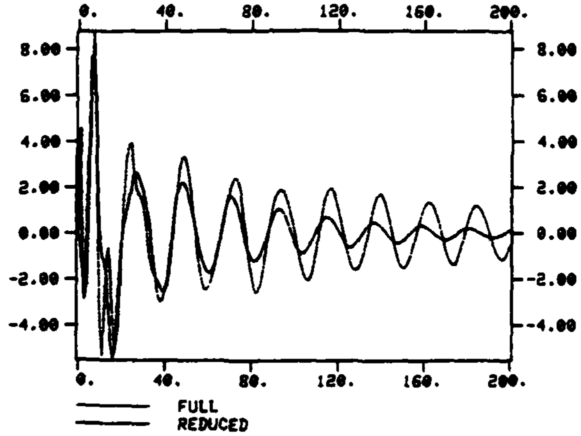


Figure 6. Example 2—Impulse response of initial approximation.

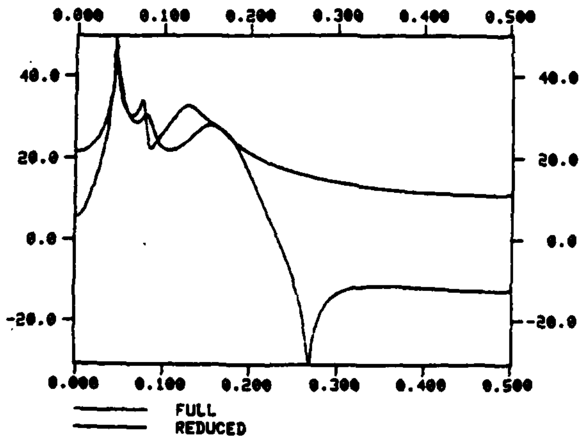


Figure 7. Example 2—Frequency response of initial approximation.

Figures 6 and 7 show the impulse and frequency responses of the initial approximation. The squared error of this approximation is 295·5, while the energy of the full model is 778·9. Again, the initial approximation is definitely poor in this case.

Iterating the algorithm 22 times gave rise to the approximation

$$\frac{b(z)}{a(z)}=\frac{\begin{array}{l}-0{\cdot}111z^{-1}-1{\cdot}4974z^{-2}+3{\cdot}8578z^{-3}\\ \qquad\qquad -4{\cdot}3248z^{-4}+4{\cdot}9636z^{-5}-3{\cdot}0119z^{-6}\end{array}}{\begin{array}{l}1-3{\cdot}7473z^{-1}+6{\cdot}6845z^{-2}-7{\cdot}2682z^{-3}\\ \qquad\qquad +5{\cdot}0934z^{-4}-2{\cdot}2115z^{-5}+0{\cdot}4930z^{-6}\end{array}} \tag{45}$$

The poles are at $0{\cdot}9560 \pm j0{\cdot}2722$, $0{\cdot}5922 \pm j0{\cdot}6391$, $0{\cdot}3254 \pm j0{\cdot}7425$. Somewhat surprisingly, four of the poles are not very close to the dominant poles of the full-order system. The approximation is still very good, as is shown in Figs. 8 and 9. The squared error of the approximation is 23·7, or about 8% of its initial value.

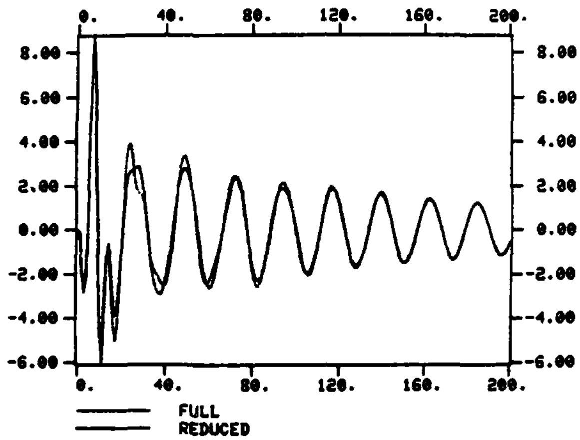


Figure 8. Example 2—Impulse response of final approximation.

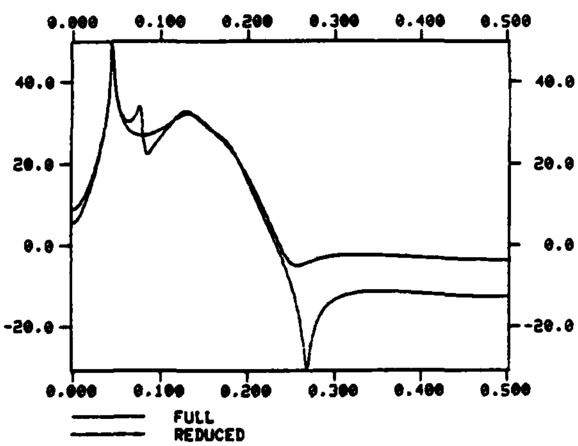


Figure 9. Example 2—Frequency response of final approximation.

The last example again uses the model (44) but takes the reduced order as $n=4$. The initial approximation is shown in Figs. 10 and 11, and the final approximation in Figs. 12 and 13. The initial and final squared errors are 498·9 and 52·5 respectively. The number of iterations needed to reach convergence was eleven. The resulting approximate model is

$$\frac{b(z)}{a(z)}=\frac{2{\cdot}6489z^{-1}-10{\cdot}9680z^{-2}+15{\cdot}5205z^{-3}-7{\cdot}2175z^{-4}}{1-3{\cdot}1856z^{-1}+4{\cdot}2616z^{-2}-2{\cdot}8615z^{-3}+0{\cdot}8285z^{-4}} \tag{46}$$

The poles are at $0{\cdot}9561 \pm j0{\cdot}2724$ and $0{\cdot}6367 \pm j0{\cdot}6579$.

The reader is referred to Kung and Lin (1980) for a comparison with other approximation methods (singular-value decomposition, Hankel-norm approximation, and dominant-mode approximation).

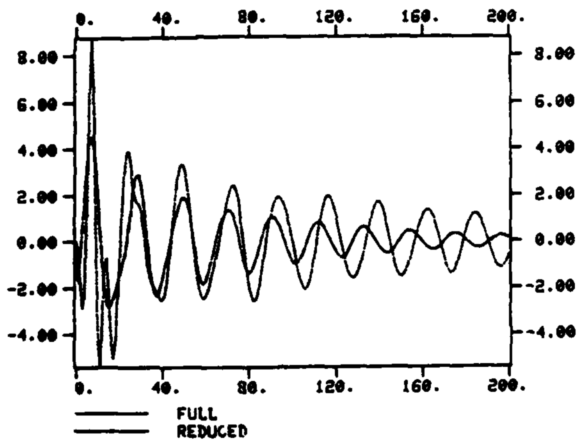


Figure 10. Example 3—Impulse response of initial approximation.

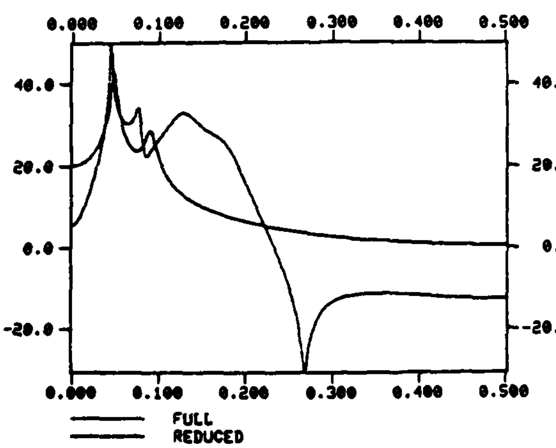


Figure 11. Example 3—Frequency response of initial approximation.

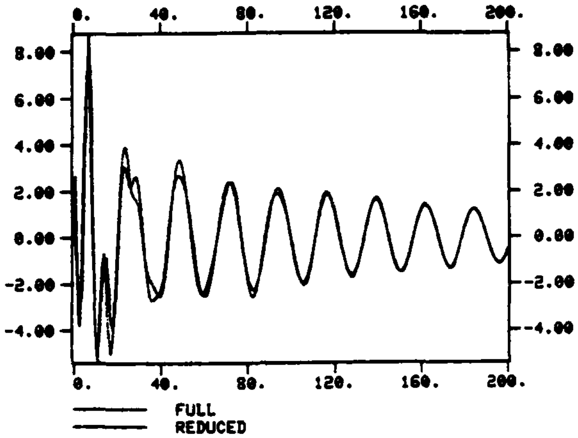


Figure 12. Example 3—Impulse response of final approximation

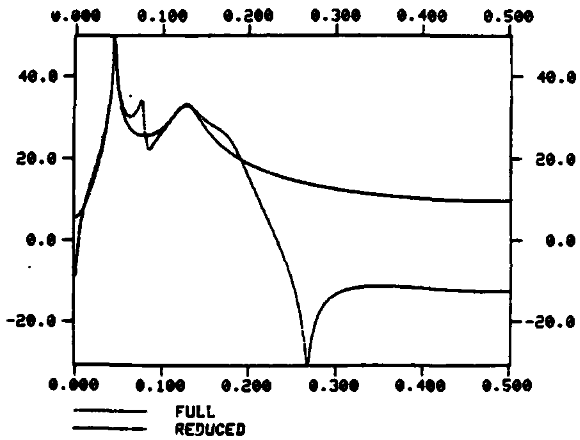


Figure 13. Example 3—Frequency response of final approximation.

6. Conclusions

We have presented an efficient algorithm for output-error model reduction of linear discrete systems. The key step was the development of a closed-form expression for the gradient. Furthermore, this expression was shown to be efficiently computable, requiring a number of operations proportional to N^2, where N is the order of the given model. The proposed optimization technique employs a special initialization procedure and the stability of the algorithm is guaranteed by a built-in stability test.

The method can easily be extended to model reduction of multi-input-multi-output systems. Somewhat surprisingly, this is achieved without the use of matrix fraction descriptions, by dealing with the characteristic polynomials directly.

The new algorithm has potential applications to filter design, system identification and control systems. Our particular motivation in developing the algorithm was for possible application to adaptive control of large-scale systems. Some results obtained in this specific application are reported in Friedlander and Porat (1982).

References

Aplevich, J. D., 1973, *Int. J. Control*, **17,** 565.

Deczky, A. G., 1972, *I.E.E.E. Trans. Audio Electroacoust.*, **20,** 257.

Friedlander, B., and Porat, B., 1982, *Proc. 21st I.E.E.E. Conf. Decision and Control*, Florida, U.S.A.

Jury, E. S., 1974, *Inners and Stability of Dynamic Systems* (Chichester : John Wiley).

Kailath, T., 1980, *Linear Systems* (Englewood Cliffs, N.J. : Prentice Hall) ; 1974, *I.E.E.E. Trans. Inf. Theory*, **20,** 146.

Kailath, T., Vieira, A., and Morf, M., 1978, *SIAM Rev.*, **20,** 106.

Kung, S., 1980, *Proc. Joint Conf. Automatic Control*, Paper No. FA8-D, San Francisco, CA, U.S.A.

Kung, S., and Lin, D. W., 1980, *Reduced Order System Modeling via Singular Value Analysis* (University of Southern California, Dept. Electrical Engineering).

Lancaster, P., 1969, *Theory of Matrices* (New York : Academic Press).

Landau, Y. D., 1979, *Adaptive Control : the Model Reference Approach* (New York : Marcel Dekker).

Luenberger, D., 1973, *Introduction to Linear and Nonlinear Programming* (New York : Addison Wesley).

Moore, B. C., 1978, *Proc. I.E.E.E. Conf. Decision and Control*, New Orleans, LA, 66–73.

Sanathanan, C. K., and Koerner, J., 1963, *I.E.E.E. Trans. autom. Control*, **3,** 56.

Steiglitz, K., and McBride, L. E., 1965, *I.E.E.E. Trans. autom. Control*, **10,** 461.

Szegö, G., 1967, *Orthogonal Polynomials* (Providence, RI : American Mathematical Society), Colloquium Publications No. 23, 3rd edn.

Vieira, A., and Kailath, T., 1977, *I.E.E.E. Trans. Circuits Syst.*, **24,** 218.

Yahagi, T., 1981, *I.E.E.E. Trans. Acoust., Speech, Sig. Processing*, **29,** 245.

APPENDIX C

AN OUTPUT ERROR METHOD FOR REDUCED ORDER CONTROLLER DESIGN

Technical Notes and Correspondence

An Output Error Method for Reduced Order Controller Design

BOAZ PORAT AND BENJAMIN FRIEDLANDER

Abstract—**An efficient computational technique is presented for the design of reduced order controllers for linear discrete-time systems. The technique is based on the minimization of the output error between the closed-loop system and a specified reference model.**

I. Introduction

This note is concerned with the problem of designing reduced order controllers for discrete control systems using a least squares error criterion with respect to a given reference model. Let the plant under consideration be represented by the transfer function

$$G(z)=f(z)/g(z)=\left(\sum_{i=1}^{N} f_i z^{N-i}\right)\Big/\left(z^N+\sum_{i=1}^{N} g_i z^{N-i}\right);\quad (1)$$

also let the reference model be represented by the transfer function

$$G_R(z)=w(z)u(z)=\left(\sum_{i=1}^{M} w_i z^{M-i}\right)\Big/\left(z^M+\sum_{i=1}^{M} u_i z^{M-i}\right).\quad (2)$$

Both the plant and the reference model are assumed to be strictly proper. The reference model is assumed to be asymptotically stable. Denote by $H(z)$ the transfer function of the desired nth order cascade compensator, where

$$H(z)=b(z)/a(z)=\left(\sum_{i=0}^{n} b_i z^{n-i}\right)\Big/\left(z^n+\sum_{i=1}^{n} a_i z^{n-i}\right).\quad (3)$$

The closed-loop transfer function $G_c(z)$ is given by

$$G_c(z)=b(z)f(z)/(a(z)g(z)+b(z)f(z))\triangleq c(z)/d(z).\quad (4)$$

Let $\epsilon(z)$ denote the difference between the transfer functions of the reference model and the actual closed-loop system, i.e., $\epsilon(z)=G_R(z)-G_c(z)$. The cost function to be minimized is given by

$$V(a_1,\cdots,a_n,b_0,\cdots,b_n)=\frac{1}{2\pi}\int_{-\pi}^{\pi}|\epsilon(e^{j\omega})|^2\,d\omega=\sum_{i=1}^{\infty}\epsilon_i^2.\quad (5)$$

The aim is to find a cascade compensator $b(z)/a(z)$ that will minimize V. It will be assumed that $n<N+M$, so that $V=0$ is impossible in general. Previous work on this problem includes [1]–[3]. The contribution of the present work is an efficient computational scheme for the gradient vector of V with respect to the coefficients of $a(z)$ and $b(z)$. Using some facts from theory of discrete Lyapunov equations and Toeplitz matrices, we derive an algorithm for computing the gradient vector in a number of operations proportional to $(N+n)^2$. The gradient is then used in a Fletcher–Powell minimization algorithm to optimize the controller parameters.

Manuscript received March 21, 1983. This paper is based on a prior submission of August 2, 1982. This work was supported by the U.S. Air Force Office of Scientific Research under Contract F4920-81-C-0051.

B. Porat is with the Department of Electrical Engineering, Technion—Israel Institute of Technology, Haifa, Israel.
B. Friedlander is with Systems Control Technology, 1801 Page Mill Rd., Palo Alto, CA 94304.

II. Computation of the Cost Function and the Gradient

Recall that the error between the transfer functions of the reference model and the closed-loop system is given by

$$\epsilon(z)=w(z)/u(z)-c(z)/d(z).\quad (6)$$

The polynomials $c(z)$ and $d(z)$ are of degrees $n+N-1$ and $N+n$, respectively, while $w(z)$ and $u(z)$ are of degrees $M-1$ and M, respectively. It is convenient to multiply both $w(z)$ and $u(z)$ by z^{n+N-M} and redefine

$$u(z)=z^{N+n}+\sum_{i=1}^{N+n} u_i z^{N+n-i};\qquad w(z)=\sum_{i=1}^{N+n} w_i z^{N+n-i}.\quad (7)$$

Let $\{h_i^0;\ 0\le i<\infty\}$ and $\{h_i;\ 0\le i<\infty\}$ be defined by

$$z^{N+n}/u(z)=\sum_{i=0}^{\infty} h_i^0 z^{-i};\qquad z^{N+n}/d(z)=\sum_{i=0}^{\infty} h_i z^{-i}.\quad (8)$$

Then we can write (6) in the form

$$\epsilon=H^0 w-Hc\quad (9)$$

where ϵ is the semi-infinite vector $[\epsilon_1,\epsilon_2\cdots]^T$, H^0 and H are lower trapezoidal semi-infinite Toeplitz matrices of width $N+n$ whose first columns are $[h_0^0,h_1^0\cdots]^T$ and $[h_0,h_1\cdots]^T$, respectively, $w=[w_1\cdots w_{n+N}]^T$ and $c=[c_1\cdots c_{N+n}]^T$. Assuming that both $u(z)$ and $d(z)$ are stable, all semi-infinite entities in (9) have finite norms. Therefore, we can express the square norm of ϵ as

$$\begin{aligned}V&=\epsilon^T\epsilon=c^TH^THc-2w^TH^{0T}Hc+w^TH^{0T}H^0w\\&=c^TRc-2w^TQc+w^TSw\end{aligned}\quad (10)$$

where the definition of R,Q,S is clear from (10). Introduce the notation $C(p)$ for the top-row companion matrix of the polynomial $p(z)$:

$$C(p)=\begin{bmatrix}-p_1 & -p_2 & \cdots & -p_m\\ 1 & & & 0\\ & \ddots & & \vdots\\ & & 1 & 0\end{bmatrix}.$$

The matrices R,Q,S can be expressed in terms of the companion matrices $C(d)$ and $C(u)$ as follows.

Lemma 1: Each of the matrices R, Q, and S is the unique solution of a matrix Lyapunov equation:

$$\begin{aligned}&R-C(d)RC^T(d)=E;\\&Q-C(u)QC^T(d)=E;\\&S-C(u)SC^T(u)=E\end{aligned}\quad (11)$$

where E is a matrix having 1 in its (1,1) position and zeros elsewhere. The proof is by a direct substitution of the definitions of R, Q, and S into (11). Uniqueness of the solution is guaranteed by the stability of $u(z)$ and $d(z)$. An algorithm for solving these equations in a number of operations proportional to $(N+n)^2$ is given, e.g., in [4]. The following lemma gives an explicit expression for the solution to a matrix Lyapunov equation of the type appearing in Lemma 1.

Lemma 2: Let $p(z)$ and $q(z)$ be two stable polynomials of degree m, and let X satisfy the matrix Lyapunov equation

$$X-C(p)XC^T(q)=E.\quad (12)$$

0018-9286/84/0700-0629$01.00 ©1984 IEEE

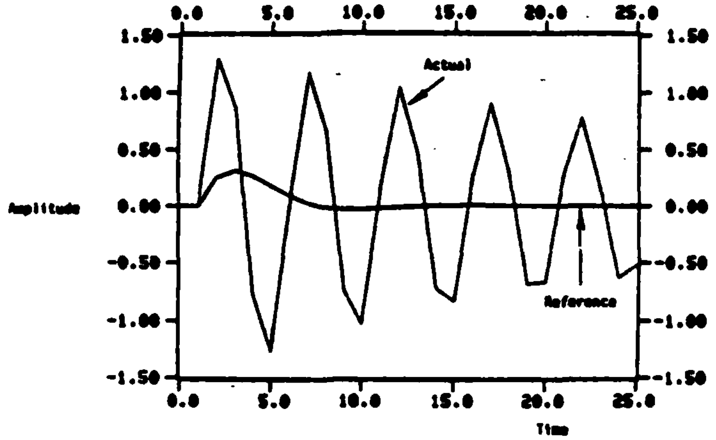


Fig. 1. Impulse response of the initial closed-loop system.

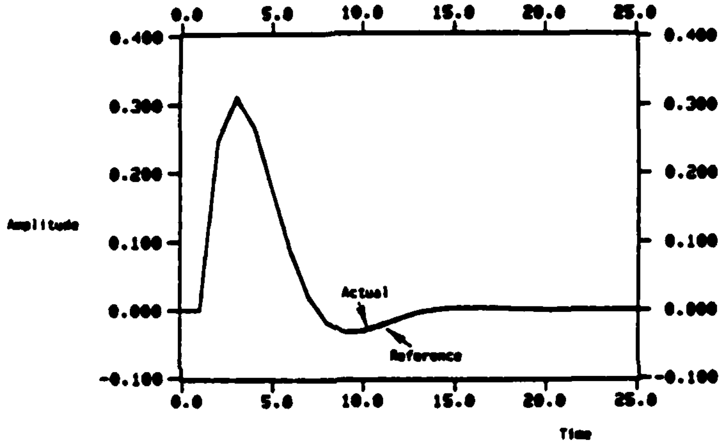


Fig. 2. Impulse response of the final closed-loop system.

Then 1) X is a Toeplitz matrix and 2) X^{-1} is given by

$$X^{-1} = Q_1 P_1^T - P_2 Q_2^T$$
$$\triangleq \begin{bmatrix} 1 & & & 0 \\ q_1 & 1 & & \\ \vdots & \ddots & \ddots & \\ q_{m-1} & \cdots & q_1 & 1 \end{bmatrix} \begin{bmatrix} 1 & p_1 & \cdots & p_{m-1} \\ & 1 & \ddots & \vdots \\ & 0 & \ddots & p_1 \\ & & & 1 \end{bmatrix} - \begin{bmatrix} p_m & & & 0 \\ p_{m-1} & \ddots & & \\ & \ddots & & \\ p_1 & \cdots & p_{m-1} & p_m \end{bmatrix} \begin{bmatrix} q_m & q_{m-1} & \cdots & q_1 \\ & & \ddots & \vdots \\ & 0 & \ddots & q_{m-1} \\ & & & q_m \end{bmatrix}. \tag{13}$$

This lemma can be proven using results from the theory of biorthogonal polynomials on the unit-circle [5]. It will be used here to write down explicit expressions for R^{-1}, Q^{-1}, and S^{-1} as follows. For a polynomial $p(z)$ of degree $m \leq N+n$, let the $(N+n)\times(N+n)$ matrices P_1 and P_2 be defined by

$$(P_1)_{ij} = \begin{cases} p_{i-j}; & 0 \leq i-j \leq m \\ 0: & \text{otherwise} \end{cases};$$
$$(P_2)_{ij} = \begin{cases} p_{m-i+j}; & 0 \leq i-j \leq m \\ 0: & \text{otherwise.} \end{cases}$$

With these definitions we have

$$D_1 = A_1 G_1 + B_1 F_1; \quad D_2 = A_2 G_2 + B_2 F_2 \tag{15}$$

$$R^{-1} = D_1 D_1^T - D_2 D_2^T; \quad Q^{-1} = D_1 U_1^T - U_2 D_2^T; \quad S^{-1} = U_1 U_1^T - U_2 U_2^T. \tag{16}$$

We now have all the necessary relationships for computing $\{\partial V/\partial a_i,\ 1 \leq i \leq n\}$ and $\{\partial v/\partial b_i,\ 0 \leq i \leq n\}$. The computation proceeds as follows:

$$\frac{\partial v}{\partial a_i} = c^T \frac{\partial R}{\partial a_i} c - 2w^T \frac{\partial Q}{\partial a_i} c \tag{17}$$

$$\frac{\partial V}{\partial b_i} = c^T \frac{\partial R}{\partial b_i} c - 2w^T \frac{\partial Q}{\partial b_i} c + 2(c^T R - w^T Q) \frac{\partial c}{\partial b_i}. \tag{18}$$

Recall that

$$\frac{\partial R}{\partial a_i} = -R \frac{\partial R^{-1}}{\partial a_i} R; \quad \frac{\partial Q}{\partial a_i} = -Q \frac{\partial Q^{-1}}{\partial b_i} Q \tag{19}$$

and similarly for the derivatives with respect to b_i. Hence

$$\frac{\partial V}{\partial a_i} = -c^T R \frac{\partial R^{-1}}{\partial a_i} Rc + 2w^T Q \frac{\partial Q^{-1}}{\partial a_i} Qc \tag{20}$$

$$\frac{\partial V}{\partial b_i} = -c^T R \frac{\partial R^{-1}}{\partial b_i} Rc + 2w^T Q \frac{\partial Q^{-1}}{\partial b_i} Qc + 2(c^T R - w^T Q) \frac{\partial c}{\partial b_i}. \tag{21}$$

To get explicit expressions for the derivatives of the inverse, let us define the k-shift matrix of dimensions $(N+n)\times(N+n)$ by

$$Z^k = \begin{cases} 1; & i - j = k \\ 0; & \text{otherwise.} \end{cases} \tag{22}$$

Then it is easy to check that

$$\frac{\partial A_1}{\partial a_i} = Z^i; \quad \frac{\partial A_2}{\partial a_i} = Z^{n-i}; \quad \frac{\partial B_1}{\partial b_i} = Z^i; \quad \frac{\partial B_2}{\partial b_i} = Z^{n-i}. \tag{23}$$

Using (15), (16), and (23) we get

$$\frac{\partial R^{-1}}{\partial a_i} = Z^i G_1 D_1^T + D_1 G_1^T (Z^i)^T - Z^{n-i} G_2 D_2^T - D_2 G_2^T (Z^{n-i})^T \tag{24}$$

$$\frac{\partial R^{-1}}{\partial b_i} = Z^i F_1 D_1^T + D_1 F_1^T (Z^i)^T - Z^{n-i} F_2 D_2^T - D_2 F_2^T (Z^{n-i})^T \tag{25}$$

$$\frac{\partial Q^{-1}}{\partial a_i} = Z^i G_1 U_1^T - U_2 G_2^T (Z^{n-i})^T \tag{26}$$

$$\frac{\partial Q^{-1}}{\partial b_i} = Z^i F_1 U_1^T - U_2^T F_2^T (Z^{n-i})^T. \tag{27}$$

Finally, $\partial \epsilon / \partial b_i$ is given by

$$\frac{\partial \epsilon}{\partial b_i} = \Big[\underbrace{0 \cdots 0}_{i}, \underbrace{f_1 \cdots f_N}_{N}, \underbrace{0 \cdots 0}_{n-i}\Big]^T. \tag{28}$$

Equations (20), (21), (24)–(28) form a complete algorithm for the gradient vector. It can be checked that the number of operations involved in the computation is proportional to $(N + n)^2$.

III. Implementation of the Algorithm

The algorithm described above was implemented on a VAX 780, in Fortran 77. The Lyapunov equations (11) were solved using a nonsymmetric version of the inverse Levinson algorithm [6]. This algorithm was also used for monitoring the stability of the $d(z)$ polynomial at each iteration. The optimization method used was Fletcher–Powell [7], with a golden section line search. The reason for this particular combination was that it requires only one gradient computation per iteration, thus helping to reduce the overall number of computations.

IV. An Example

The following example will serve to illustrate the performance of the algorithm:

$$G(z) = z^4/(z^5 - 3.5z^4 + 4.68z^3 - 2.921z^2 + 0.8263z - 0.085) \tag{29}$$

$$G_r(z) = 0.25z^2/(z^2 - 1.25z + 0.5). \tag{30}$$

The order n of the cascade compensator was chosen as 2. An initial stability compensator, found by trial and error, was taken as

$$H(z) = (1.3z^2 - 2z + 0.825)/z^2. \tag{31}$$

Fig. 1 shows the impulse response of the initial closed-loop system compared to that of the reference model. The optimal controller, obtained after 25 iterations of the algorithm, was found to be

$$H(z) = (0.248z^2 - 0.47z + 0.234)/(z^2 + 0.116z - 0.055). \tag{32}$$

Fig. 2 shows the impulse response of the final closed-loop system computed to that of the reference model. In this example, the total square error of the optimal solution was about 0.05 percent of the reference model impulse response energy.

V. Conclusions

A computationally efficient algorithm for the design of reduced order controllers was presented. The algorithm uses input/output (transfer function) descriptions of the plant, the reference model and the controller, rather than the more commonly used state-space descriptions. The two descriptions are mathematically equivalent, but lead to different computational procedures.

The relative computational efficiency of the proposed algorithm opens the way for using it as part of an adaptive reduced order controller where the design procedure needs to be performed repeatedly. This, however, is outside the scope of this note.

References

[1] W. S. Levine, T. L. Johnson, and M. Athans, "Optimal limited state variable feedback controllers for linear systems," *IEEE Trans. Automat. Contr.*, vol. AC-16, pp. 785–793, Dec. 1971.

[2] R. L. Kosut, "Suboptimal control of linear time invariant systems subject to control structure constraints," *IEEE Trans. Automat. Contr.*, vol. AC-15, pp. 557–563, Oct. 1970.

[3] G. C. Goodwin and P. Z. Ramadge, "Design of restricted complexity adaptive regulators," *IEEE Trans. Automat. Contr.*, vol. AC-24, pp. 584–588, Aug. 1979.

[4] R. R. Bitmead, "Explicit solutions of the discrete-time Lyapunov matrix equation and Kalman–Yakubovich equations," *IEEE Trans. Automat. Contr.*, vol. AC-26, pp. 1291–1294, Dec. 1981.

[5] T. Kailath, A. Vieira, and M. Morf, "Inverses of Toeplitz operators, innovations, and orthogonal polynomials," *SIAM Rev.*, vol. 20, pp. 106–110, Jan. 1978.

[6] A. Vieira and T. Kailath, "On another approach to the Schur–Cohn criterion," *IEEE Trans. Circuits Syst.*, vol. CAS-24, pp. 218–220, Apr. 1977.

[7] D. Luenberger, *Introduction to Linear and Nonlinear Programming*. Reading, MA: Addison-Wesley, 1973.

Nonparametric Algorithm for Input Signals Identification in Static Distributed-Parameter Systems

EWARYST RAFAJLOWICZ

Abstract —In this correspondence, a nonparametric algorithm for identification of input signals in linear, static distributed-parameter systems is proposed and investigated. Integral mean-square convergence of the algorithm is proved for an infinite number of point measurements of the system state. The algorithm is a generalized version of the one recently proposed by Rutkowski [10] for nonparametric function fitting, and in a common area, the presented results are complementary.

I. Introduction

The aim of this correspondence is to propose and investigate an algorithm for identification of an unknown input signal or an excitation of a static, linear distributed-parameter system (DPS) from point measurements of its state. Problems of this type arise in the areas of water and air pollution, electromagnetic heating, vibrations isolation, etc., and have been treated by several authors (mainly from a computational point of view) [3]–[6].

Theoretical analysis of such problems is a difficult task since they are ill-posed in the sense of Hadamard [2], [7]. This difficulty is usually avoided by assuming *a priori* that the unknown excitation belongs to a certain parametric class and only its parameters are estimated [1].

In this correspondence, no assumptions of this type are made, and thus, the proposed algorithm is a nonparametric one. Its main advantage is asymptotic optimality (AO), understood as the integral mean-square convergence (IMSC) as the number of measurements approaches to infinity.

It should be remarked that the proposed algorithm is a modified

Manuscript received July 19, 1983; revised October 17, 1983.

The author is with the Institute of Engineering Cybernetics, Technical University of Wroclaw, Wroclaw, Poland.

0018-9286/84/0700-0631$01.00

APPENDIX D

LATTICE IMPLEMENTATION OF SOME RECURSIVE PARAMETER ESTIMATION ALGORITHMS

INT. J. CONTROL, 1983, VOL. 37, NO. 4, 661–684

Lattice implementation of some recursive parameter-estimation algorithms†

BENJAMIN FRIEDLANDER‡

Linear dynamic models of plants are usually parametrized by the coefficients of difference equations. Lattice structures and their reflection coefficients provide an alternative parametrization that offers several advantages, including numerical robustness, computational efficiency and ease of hardware implementation. The recursive square-root normalized lattice versions of the following well-known parameter-estimation algorithms are presented; recursive least-squares, recursive instrumental variable, extended least-squares, and recursive maximum-likelihood.

1. Introduction

The need for real-time system identification led to the development of numerous recursive parameter-estimation algorithms. The most commonly used algorithms are related to linear input–output models described by difference equations of the type

$$y_t = -\sum_{i=1}^{NA} a_i y_{t-i} + \sum_{i=0}^{NB} b_i u_{t-i} + e_t \tag{1}$$

where u_t, y_t denote the input and output of the plant and e_t denotes a disturbance process. Parametrizing the plant by the coefficients $\{a_i, b_i\}$ of the difference equation seems to be a natural choice, resulting in an estimation problem of the type encountered in regression analysis. The regression variables in this case are simply the input $\{u_t, \ldots, u_{t-NB}\}$ and output $\{y_{t-1}, \ldots, y_{t-NA}\}$ variables. Other parametrizations are, of course, possible. To see this, rewrite (1) in the form

$$y_t = \phi_t^{\mathrm{T}} \theta + e_t \tag{2}$$

where

$$\phi_t = [-y_{t-1}, \ldots, -y_{t-NA}, u_t, \ldots, u_{t-NB}]^{\mathrm{T}}$$
$$\theta = [a_1, \ldots, a_{NA}, b_0, \ldots, b_{NB}]^{\mathrm{T}}$$

The parameter vector θ and the vector of regression variables ϕ_t can be replaced by $\bar{\theta}_t = S\theta$, $\bar{\phi}_t = S^{-1}\phi_t$, where S is an arbitrary (possibly time-varying) non-singular matrix, since clearly

$$y_t = \phi_t^{\mathrm{T}} \theta + e_t = \bar{\phi}^{\mathrm{T}} \bar{\theta} + e_t \tag{3}$$

This leads to an infinite number of possibilities for parametrizing the plant. An interesting choice is to pick S so that the regression variables will be uncorrelated. This can be done by letting S be the lower-triangular square-root

Received 31 August 1982.

† This work was supported in part by the Air Force Office of Scientific Research under Contract No. F49620-81-C-0051. The United States Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright notation thereof.

‡ Systems Control Technology Inc., 1801 Page Mill Road, Palo Alto, California 94304, U.S.A.

of the covarience matrix of ϕ_t, i.e.

$$SS^T = E\{\phi_t \phi_t^T\} \tag{4}$$

in this case

$$E\{\bar{\phi}_t \bar{\phi}_t^T\} = S^{-1} E\{\phi_t \phi_t^T\} S^{-T} = I \tag{5}$$

In other words, the original set of regression variables is replaced by a Gram-Schmidt orthogonalized set. Square-root procedures in linear least-squares estimation are known to have good numerical properties and to be more robust than techniques involving the covariance matrix itself (Bierman 1977, Lawson and Hanson 1974). Note that the least-squares estimate of the parameter vector $\bar{\theta}$ is given by

$$\bar{\theta} = [E\{\bar{\phi}_t \bar{\phi}_t\}]^{-1} E\{\bar{\phi}_t y_t\} = E\{\bar{\phi}_t y_t\} \tag{6}$$

In other words, the parameters can be interpreted as the cross-correlation between the data y_t and the regression variables. Such parameters have been used for quite some time under the name of PARtial CORrelation (PARCOR) coefficients in the analysis of time-series, especially in speech applications (Markel and Gray 1976). This parametrization is related to lattice structures instead of the tapped delay-line structure inherent in the difference equation (1).

Lattice forms are widely used in signal-processing applications involving linear filtering and prediction. They are known to have a number of attractive features including: (i) good numerical behaviour on finite-word-length processors; (ii) an orthogonality (decoupling) property: the signals propagating in a lattice filter are uncorrelated. (One manifestation of this property is the fact that when the filter order is increased, one has to add an additional section to the filter without changing the previous sections. In other words the $(N+1)$th-order lattice predictor is the same as the Nth-order predictor except for the last section. This feature is very useful in handling the problem of model-order determination and reduced-order modelling); (iii) a cascaded structure of the lattice filter (consisting of identical sections) which is very convenient for implementation using special purpose hardware, microprocessors or LSI; (iv) in normalized versions of the lattice filter all the variables are automatically scaled, making it possible to use fixed-point computations. (However, normalization sometimes has an adverse effect on the numerical behaviour of the algorithm (see Samson and Reddy 1982).)

While square-root techniques are sometimes applied to system identification (Strejc 1980), lattice structures are apparently not used. One possible reason is that efficient recursive algorithms for estimating lattice parameters were developed only recently. Another reason is that earlier work on lattice forms was limited to all-pole models, while most realistic plants have both poles and zeros. The work (Morf *et al.* 1977, Lee 1980, Lee *et al.* 1981, Friedlander 1982, Porat *et al.* 1981) on recursive lattice forms provides an elegant solution to the lattice modelling problem for both all-pole and pole-zero plants. This development should encourage the use of lattice forms in system identification and adaptive control.

The purpose of this paper is to present lattice implementations of the following commonly used recursive parameter-estimation algorithms: recursive least-squares (RLS), recursive instrumental variables (RIV), extended least-squares (ELS), and recursive maximum-likelihood (RML) (Söderström *et al.*

1978, Goodwin and Payne 1977). The idea of using recursive lattice forms for system identification was previously proposed in Morf *et al.* (1977). However, the lattice implementations of these four algorithms were not discussed in full detail. In particular, the normalized lattice RIV and the lattice recursions for arbitrary model orders are believed to be presented here for the first time.

The structure of the paper is as follows. In each section we present one of the lattice algorithms and discuss its properties. Owing to space limitations, only brief derivations are included. These derivations assume some familiarity with the projection framework for developing recursive lattice forms which is described in greater detail in Lee (1980), Lee *et al.* (1981), Friedlander (1982) and Porat *et al.* (1981). We have attempted to make this paper self-contained, but some of the background material has been deferred to the references. The results in this paper are presented in the normalized case only. Unnormalized versions of these algorithms can be similarly derived.

2. The lattice recursive least-squares algorithm

In this section we consider models of the type depicted in (1). Given a set of measurements $\{y_0, \ldots, y_T\}$ we can write

$$X_{NA,NB,T}\hat{\theta}_T = y_{0:T} \tag{7}$$

where

$$X_{m,n,T} \triangleq \left[\begin{array}{cccc:cccc} 0 & & & 0 & 0 & & & 0 \\ y'_0 & & & & u'_0 & & & \\ \vdots & \ddots & & & \vdots & \ddots & & \\ \vdots & & & y'_0 & \vdots & & & u'_0 \\ \vdots & & & \vdots & \vdots & & & \vdots \\ y'_{T-1} & \cdots & & y'_{T-m} & u'_{T-1} & \cdots & & u'_{T-n} \end{array}\right]$$

$\hat{\theta}_T$ = the estimate of the parameter vector θ (eqn. (2))

$$y_{0:T} = [y_0, \ldots, y_T]'$$

The least-squares estimate of the parameter vector is given by (indices are omitted for notational convenience)

$$\hat{\theta}_T \triangleq (X'X)^{-1}X'y_{0:T} \tag{8}$$

The associated error vector is given by

$$\epsilon_{0:T} = y_{0:T} - X\hat{\theta}_T = [I - X(X'X)^{-1}X']y_{0:T} \tag{9}$$

where

$$\epsilon_{0:T} = [\epsilon_0, \ldots, \epsilon_T]'$$

The last entry ϵ_T of the error vector is given by

$$\epsilon_T = \pi'[I - X(X'X)^{-1}X]y_{0:T} \tag{10}$$

where

$$\pi = [0, \ldots, 0, I]'$$

Note that $P_X = X(X'X)^{-1}X'$ is a projection operator on the space spanned by the columns of X, i.e. $P_X P_X = P_X$. In Lee (1980), Lee *et al.* (1981), Friedlander (1982), and Porat *et al.* (1981) it was shown that projection operators of this type can be recursively updated as the projection space X is changed by the addition of columns x. More specifically, the following update formula can be derived

$$U'P^c_{X+x}\, V = U'P^c_X\, V - U'P_X{}^c\, x[x'P_X{}^c\, x]^{-1}x'P_X{}^c\, V \tag{11}$$

where U, V, x are vectors (or matrices) of compatible dimensions and $P^c \triangleq I - P$. By proper choices of the projection space X and the vectors U, V, x, the unnormalized lattice recursions are obtained (Lee *et al.* 1981, Friedlander 1982). A normalized lattice form is similarly derived by considering normalized projections

$$\rho_X(U, V) \triangleq [U'P_X{}^c\, U]^{-1/2}U'P_X{}^c\, V[V'P_X{}^c\, V]^{-T/2} \tag{12}$$

which obey the following update formula

$$\begin{aligned}\rho_{X+x}(U, V) = {}&[I - \rho_X(U, x)\rho_X(x, U)]^{-1/2}[\rho_X(U, V)\\ &- \rho_X(U, x)\rho_X(x, V)][I - \rho_X(V, x)\rho_X(x, V)]^{-T/2}\end{aligned} \tag{13}$$

To avoid repeating this expression we will find it convenient to define the functions

$$\left.\begin{aligned}F(u, v, w) &\triangleq [I - ww']^{-1/2}[u - wv][I - v'v]^{-T/2}\\ F^{-1}(u, v, w) &\triangleq [I - ww']^{1/2}u[I - v'v]^{T/2} + wv\end{aligned}\right\} \tag{14}$$

Using the update formula (13) we can derive a large number of lattice algorithms by proper choices of X, x, U, V and proper definitions of variables. Table 1 summarizes the variables involved in the LATTICE RLS algorithm. The quantities $x_{0:T}$ and $x_{0:T}{}^{m,n}$ appearing in the table are defined by

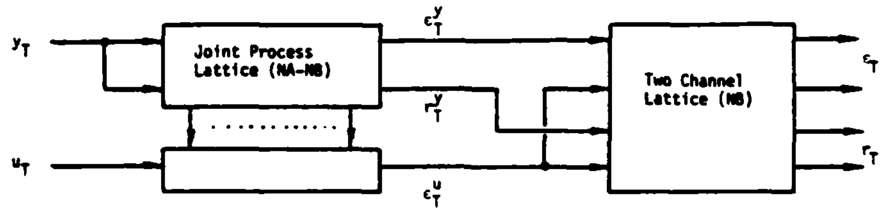
$$x_{0:T} = \begin{bmatrix} y'_0 & u'_0 \\ \vdots & \vdots \\ \vdots & \vdots \\ y'_T & u'_T \end{bmatrix}, \quad x_{0:T}{}^{m,n} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ y'_0 & u'_0 \\ \vdots & \vdots \\ y'_{T-m} & u'_{T-n} \end{bmatrix}, \quad y_{0:T}{}^{m} = \begin{bmatrix} 0 \\ 0 \\ y'_0 \\ \vdots \\ y'_{T-p} \end{bmatrix} \tag{15}$$

$\rho_S(U, V)$	S	U	V	Comments
$\epsilon^y{}_{p,T}$	$X_{p,0,T}$	$y_{0:T}$	π	Joint-process lattice
$\epsilon^u{}_{p,T}$	$X_{p,0,T+1}$	$u_{0:T}$	π	
$r^y{}_{p,T-1}$	$X_{p,0,T}$	$y_{0:T}{}^{p+1}$	π	
$K^y{}_{p+1,T}$	$X_{p,0,T}$	$y_{0:T}$	$y_{0:T}{}^{p+1}$	
$K^u{}_{p+1,T}$	$X_{p,0,T+1}$	$u_{0:T}$	$y_{0:T}{}^{p+1}$	
$\epsilon_{p,T}$	$X_{m,p,T}$	$x_{0:T}$	π	Two-channel lattice $m \triangleq NA - NB + p$
$r_{p,T-1}$	$X_{m,p,T}$	$x_{0:T}{}^{m+1,p+1}$	π	
$K_{p+1,T}$	$X_{m,p,T}$	$x_{0:T}$	π	

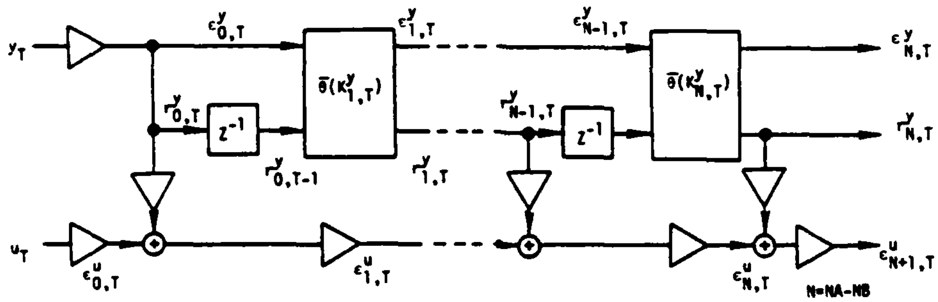
Table 1. Definitions of lattice RLS variables.

As will be shown shortly, these variables define the lattice structure depicted in Fig. 1, where $\bar{\theta}(K)$ is an operator acting on the input vector $[\epsilon', r']$, defined by

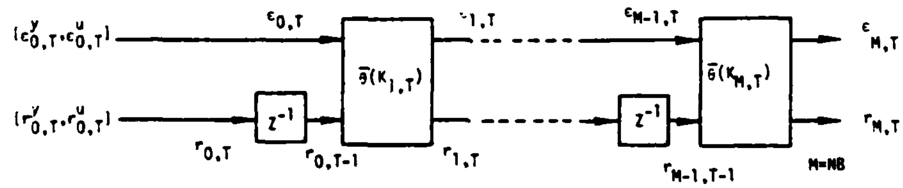
$$\bar{\theta}(K)\begin{bmatrix} \epsilon \\ r \end{bmatrix} = \begin{bmatrix} F(\epsilon, r, K) \\ F(r, \epsilon, K') \end{bmatrix} \tag{16}$$



(*a*)



(*b*)



(*c*)

Figure 1. The RLS lattice form: (*a*) overall lattice structure, (*b*) joint-process lattice, (*c*) two-channel lattice.

These lattice recursions are obtained by making the substitutions in (13) as depicted in Table 2.

The following facts are useful in interpreting the entries of this table:

(i) *order update*

$$X_{m+1,n+1,T} = X_{m,n,T} + x_{0:T}{}^{m+1,n+1} \tag{17}$$

(ii) *order and time update*

$$X_{m+1,n+1,T+1} = X_{m,n,T} + x_{0:T} \tag{18}$$

(iii) *time update*

$$P^c_{X_{m,n,T+s}} = \begin{bmatrix} P^c_{X_{m,n,T-1}} & 0 \\ & \vdots \\ 0 \ldots\ldots\ldots\ldots & 0 \end{bmatrix} \tag{19}$$

X	x	U	V	$\rho_{X+x}(U, V)$	$\rho_X(U, V)$	$\rho_X(x, V)$	$\rho_X(U, x)$
$X_{p,0,T}$	${y_{0:T}}^{p+1}$	$y_{0:T}$	π	${\epsilon^y}_{p+1,T}$	${\epsilon^y}_{p,T}$	${r^y}_{p,T-1}$	${K^y}_{p+1,T}$
$X_{p,0,T}$	$y_{0:T}$	${y_{0:T}}^{p+1}$	π	${r^y}_{p+1,T}$	${r^y}_{p,T-1}$	${\epsilon^y}_{p,T}$	${K^{y'}}_{p+1,T}$
$X_{p,0,T+1}$	${y_{0:T}}^{p+1}$	$u_{0:T}$	π	${\epsilon^u}_{p+1,T}$	${r^y}_{p,T}$	${r^y}_{p,T}$	${K^u}_{p+1,T}$
$X_{p,0,T}$	π	$y_{0:T}$	${y_{0:T}}^{p+1}$	${K^y}_{p+1,T-1}$	${K^y}_{p+1,T}$	${r^{y'}}_{\mu,T-1}$	${\epsilon^y}_{\mu,T}$
$X_{p,0,T+1}$	π	$u_{0:T}$	${y_{0:T}}^{p+1}$	${K^u}_{\mu+1,T-1}$	${K^u}_{p+1,T}$	${r^{y'}}_{p,T}$	${\epsilon^u}_{p,T}$
$X_{m,n,T}$	${x_{0:T}}^{m+1,n+1}$	$x_{0:T}$	π	$\epsilon_{p+1,T}$	$\epsilon_{p,T}$	$r_{p,T-1}$	$K_{p+1,T}$
$X_{m,n,T}$	$x_{0:T}$	${x_{0:T}}^{m+1,n+1}$	π	$r_{p+1,T}$	$r_{p,T-1}$	$\epsilon_{p,T}$	${K'}_{p+1,T}$
$X_{m,n,T}$	π	$x_{0:T}$	${x_{0:T}}^{m+1,n+1}$	$K_{p+1,T-1}$	$K_{p+1,T}$	${r'}_{p,T-1}$	$\epsilon_{p,T}$

$$m = NA - NB + p\,; \quad n = p.$$

Table 2. Derivation of the lattice RLS.

Another point that requires some attention is the interface between the joint-process lattice and the two-channel lattice in Fig. 1. Note that

$$\left.\begin{aligned} \epsilon_{0,T} &= \rho_{X_{NA-NB,0,T}}(x_{0:T}, \pi) \\ r_{0,T-1} &= \rho_{X_{NA-NB,0,T}}({x_{0,T}}^{NA-NB+1,1}, \pi) \end{aligned}\right\} \tag{20}$$

If we assume that the projection operator projects first on ys and then on us (i.e. the square roots in the update formula and the definition of ρ are all lower triangular), then we note the following:

(i) projecting $x_{0:T}$ on $X_{NA-NB,0,T}$ involves the projection of $u_{0:T}$ on $X_{NA-NB+1,0,T+1}$

(ii) projecting ${x_{0:T}}^{NA-NB+1,1}$ on $X_{NA-NB,0,T}$ involves the projection of $u_{0:T}$ on $X_{NA-NB+1,T}$.

Therefore we can conclude that

$$\left.\begin{aligned} \epsilon_{0,T} &= [{\epsilon^{y'}}_{NA-NB,T}, {\epsilon^{u'}}_{NA-NB,T}]' \\ r_{0,T-1} &= [{r^{y'}}_{NA-NB,T-1}, {\epsilon^{u'}}_{NA-NB+1,T-1}] \end{aligned}\right\} \tag{21}$$

We can now summarize the lattice RLS algorithm by reading off the proper entries of Table 2.

Lattice RLS algorithm

The algorithm is presented here for the case $NA \geqslant NB$. If $NA < NB$, simply interchange y and u and NA and NB.

Input parameters

NA, NB = model orders

λ = exponential weighting factor

σ = prior covariance

y_T, u_T = data sequence

Variables

R_T^y, R_T^u = estimated convariance of y, u

$K^y_{p+1,T}, K^u_{p+1,T}$ = reflection coefficients

$K_{p+1,T}$ = reflection coefficients

$\epsilon^y_{p,T}, \epsilon^u_{p,T}, \epsilon_{p,T}$ = forward prediction errors
(dim $\{y\}$, dim $\{u\}$, dim $\{[y', u']'\}$)

$r^y_{p,T}, r_{p,T}$ = backward prediction errors
(dim $\{y\}$, dim $\{[y', u']'\}$)

Initialization

$R_{-1}^u = \sigma I, \; R_{-1}^u = \sigma I$

$K^y_{p,-1} = r^y_{p,-1} = 0, \; p = 1, \ldots, NA - NB$

$K^u_{p,-1} = 0, \; p = 1, \ldots, NA - NB + 1$

$K_{p,-1} = r_{p,-1} = 0, \; p = 1, \ldots, NB$

Main loop

At each time step do the following :

(i) set

$$R_T^y = \lambda R_{T-1}^y + y_T y'_T$$

$$R_T^u = \lambda R_{T-1}^u + u_T u'_T$$

$$\epsilon_{0,T}^y = r_{0,T}^y = (R_T^y)^{-1/2} y_T$$

$$\epsilon_{0,T}^u = (R_T^u)^{-1/2} u_T$$

(ii) update joint-process lattice

For $p = 0, \ldots, \min\{NA - NB, T\}$

$$\left.\begin{aligned} K^y_{p+1,T} &= F^{-1}(K^y_{p+1,T-1}, r^{y'}_{p,T-1}, \epsilon^y_{p,T}) \\ \epsilon^y_{p+1,T} &= F(\epsilon^y_{p,T}, r^y_{p,T-1}, K_{p+1,T}) \\ r^y_{p+1,T} &= F(r^y_{p,T-1}, \epsilon^y_{p,T}, K'_{p+1,T}) \end{aligned}\right\} \text{omit this for } p = \min\{NA - NB, T\}$$

$$K^u_{p+1,T} = F^{-1}(K^u_{p+1,T-1}, r^{y'}_{p,T}, \epsilon^u_{p,T})$$

$$\epsilon^u_{p+1,T} = F(\epsilon^u_{p,T}, r^y_{p,T}, K^u_{p+1,T})$$

(iii) set

$$\epsilon_{0,T} = [\epsilon^{y'}{}_{NA-NB,T}, \epsilon^{u'}{}_{NA-NB+1,T}]'$$

$$r_{0,T} = [r^{y'}{}_{NA-NB,T}, \epsilon^{u'}{}_{NA-NB+1,T}]'$$

(iv) update two-channel lattice

For $p = 0$ to $\min\{NB, T - NA + NB\} - 1$

$$K_{p+1,T} = F^{-1}(K_{p+1,T-1}, r'_{p,T-1}, \epsilon_{p,T})$$

$$\epsilon_{p+1,T} = F(\epsilon_{p,T}, r_{p,T-1}, K_{p+1,T})$$

$$r_{p+1,T} = F(r_{p,T-1}, \epsilon_{p,T}, K'_{p+1,T})$$

Remarks

As the lattice recursions are started they may involve division by zero. It can be shown that the proper procedure is to set to zero the result of such division in the scalar case, or to use pseudo-inverses in the matrix case. (See Porat *et al.* (1981) for details.) For coding purposes it is convenient to make $F(\cdot, \cdot, \cdot)$ and $F^{-1}(\cdot, \cdot, \cdot)$ into subroutine calls. The lattice algorithm then consists of repeated calls of these subroutines.

Running the lattice RLS on data will provide a set of reflection coefficients parametrizing the plant transfer function. In some applications it may be desired to recover the estimates of the $\{a_i, b_i\}$ parameters, rather than to continue with a lattice structure. Assuming that the lattice parameters have converged, this can be done by looking at the impulse response of the filter depicted in Fig. 1. Recall that this filter computes the normalized prediction error sequence ϵ_t where

$$\epsilon_t = \sum_{i=0}^{NA} \bar{a}_i y_{t-i} - \sum_{i=0}^{NA} \bar{b}_i u_{t-i} \tag{22}$$

where $\{\bar{a}_i, \bar{b}_i\}$ are the normalized versions of $\{a_i, b_i\}$. Note that the impulse response from the y input to the ϵ output will be $\{\bar{a}_0, \bar{a}_1, \ldots, \bar{a}_{NA}\}$ while the impulse response from u to ϵ will be $\{\bar{b}_0, \bar{b}_1, \ldots, \bar{b}_{NB}\}$. The unnormalized parameters can be obtained by setting

$$\left.\begin{aligned} a_i &= \bar{a}_0^{-1}\,\bar{a}_i, \quad i = 1, \ldots, NA \\ b_i &= \bar{a}_0^{-1}\,b_i, \quad i = 0, \ldots, NB \end{aligned}\right\} \tag{23}$$

This method of computing $\{a_i, b_i\}$ from the reflection coefficients does not give the exact least-squares estimates of these parameters. However, if the reflection coefficients have converged 'sufficiently' these estimates will be very close to the optimal estimates. A slightly more complicated lattice filter is available for computing the exact least-squares estimates of $\{a_i, b_i\}$, from the information provided by the lattice RLS: see Friedlander (1982) and Porat *et al.* (1981) for a more detailed discussion.

The lattice RLS differs from the standard RLS algorithm in several respects:

(i) *Initialization*. The order recursive nature of the lattice RLS makes it possible to eliminate transient phenomena caused by incorrect initial conditions, leading to faster startup.

(ii) *Normalization*. All quantities the lattice RLS (with the exception of $R^{v}{}_{T}$, $R^{u}{}_{T}$) have magnitudes less than one.

(iii) *Computational requirements*. The lattice RLS is computationally efficient, requiring O(N) operations per time step, where $N = NA + NB$. The usual implementation of the RLS requires O(N^2) operations per time step. Since square-roots are time-consuming operations on general-purpose computers, the efficiency of the lattice forms becomes apparent only for fairly large values of N. However, implementations on special-purpose hardware designed to take advantage of the lattice structure, can be very efficient. Note that the computation of the $\{a_i, b_i\}$ parameters from the reflection coefficients requires O(N^2) operations. This computation can be avoided, however, by reformulating the problem for which the parameters were estimated so that it it will use directly the reflection coefficients.

(iv) *Order-recursive*. The lattice RLS is not only time-recursive, but is also order-recursive. This makes it possible initially to overdetermine the plant order and to choose a lower-order model *after* the parameter estimates are computed.

Finally we note that the lattice algorithm presented in this section was only one of many different lattice forms (the so-called normalized pre-windowed form). The unnormalized lattice recursion and the convariance lattice form are presented in Lee *et al.* (1981) and Porat *et al.* (1981). The pre-windowed form is simpler than the covariance form and is probably better suited for system identification applications.

3. The lattice recursive instrumental variable algorithm

The RLS algorithm provides biased estimates when the disturbance process e_t (see eqn. (1)) is non-white. The intrumental variables method (Young 1970, Söderström and Stoica 1981, Wong and Polak 1967) was derived to eliminate this problem. The parameter estimate $\hat{\theta}_T$ is given by

$$\hat{\theta}_T = (Z'X)^{-1} Z' y_{0:T} \tag{24}$$

where Z is an instrumental variable matrix

$$Z_{m,n,T} = \left[\begin{array}{ccc:ccc} 0 & & 0 & 0 & & 0 \\ \tilde{y}'_0 & & & \tilde{u}'_0 & & \\ \vdots & \ddots & & \vdots & \ddots & \\ \vdots & & \tilde{y}'_0 & \vdots & & \tilde{u}'_0 \\ \vdots & & \vdots & \vdots & & \vdots \\ \tilde{y}'_{T-1} & \cdots & \tilde{y}'_{T-m} & \tilde{u}'_{T-1} & \cdots & \tilde{u}'_{T-n} \end{array}\right] \tag{25}$$

The instrumental variables $\tilde{y}_t, \tilde{u}_t$ can be chosen in different ways. A typical choice is to set $\tilde{u}_t = u_t$ and $\tilde{y}_t$ to be the output of a filter driven by u_t, for example

$$\tilde{y}_t = -\sum_{i=1}^{NA} \hat{a}_i \tilde{y}_{t-i} + \sum_{i=0}^{NB} \hat{b}_i u_{t-i} \tag{26}$$

Note that the projection formula (10) is replaced in this case by

$$\epsilon_T = y'_{0:T}[I - Z(X'Z)^{-1}X']\pi \tag{27}$$

which involves the non-symmetric 'projection operator' ρ_{ZX}. By analogy with the derivation outlined in § 3, we define a normalized projection operator ρ_{ZX} by

$$\rho_{ZX}(U, V) \triangleq [U'P_{ZX}^c U]^{-1/2} U' P_{ZX}^c V [V'P_{ZX}^c V]^{-T/2} \tag{28}$$

In Appendix A we prove the following update formula for this operator

$$\begin{aligned}\rho_{Z+z, X+x}(U, V) = {} & [I - \rho_{ZX}(U, z)\rho_{ZX}^{-1}(x, z)\rho_{ZX}(x, U)]^{-1/2} \\ & \times [\rho_{ZX}(U, V) - \rho_{ZX}(U, z)\rho_{ZX}^{-1}(x, z)\rho_{ZX}(x, V) \\ & \times [I - \rho_{ZX}(V, z)\rho_{ZX}^{-1}(x, z)\rho_{ZX}(x, V)]^{-T/2}\end{aligned}$$

To avoid repeating this complicated expression we will find it convenient to define

$$\bar{F}(u, v, w, q, r, s) = [I - qs^{-1}r]^{-1/2}[u - qs^{-1}v][I - ws^{-1}v]^{-T/2} \tag{30}$$

and its 'inverse'

$$\bar{F}^{-1}(u, v, w, q, r, s) = [I - qs^{-1}r]^{1/2}u[I - ws^{-1}v]^{T/2} + qs^{-1}v$$

Using the update formula (29) we can derive several versions of the lattice RIV by proper choices of Z, X, z, x, U, V. To simplify the presentation we

$\rho_{ZX}(U, V)$	Z	X	U	V	Comments
$\epsilon^x{}_{p,T}$	$Z_{p,T}$	$X_{p,T}$	$x_{0:T}$	π	
$r^x{}_{p,T-1}$	$Z_{p,T}$	$X_{p,T}$	$x_{0:T}{}^{p+1}$	π	Prediction errors
$\epsilon^z{}_{p,T}$	$Z_{p,T}$	$X_{p,T}$	$z_{0:T}$	π	
$r^z{}_{p,T-1}$	$Z_{p,T}$	$X_{p,T}$	$z_{0:T}{}^{p+1}$	π	
$\tilde{\epsilon}^x{}_{p,T}$	$X_{p,T}$	$Z_{p,T}$	$x_{0:T}$	π	
$\tilde{r}^x{}_{p,T-1}$	$X_{p,T}$	$Z_{p,T}$	$x_{0:T}{}^{p+1}$	π	Auxiliary prediction errors
$\tilde{\epsilon}^z{}_{p,T}$	$X_{p,T}$	$Z_{p,T}$	$z_{0:T}$	π	
$\tilde{r}^z{}_{p,T-1}$	$X_{p,T}$	$Z_{p,T}$	$z_{0:T}{}^{p+1}$	π	
$K^{x\tilde{x}}{}_{p+1,T}$	$Z_{p,T}$	$X_{p,T}$	$x_{0:T}$	$x_{0:T}{}^{p+1}$	
$K^{z\tilde{z}}{}_{p+1,T}$	$Z_{p,T}$	$X_{p,T}$	$z_{0:T}$	$z_{0:T}{}^{p+1}$	
$K^{x\tilde{z}}{}_{p+1,T}$	$Z_{p,T}$	$X_{p,T}$	$x_{0:T}$	$z_{0:T}{}^{p+1}$	
$K^{\tilde{x}x}{}_{p+1,T}$	$Z_{p,T}$	$X_{p,T}$	$x_{0:T}{}^{p+1}$	$x_{0:T}$	Reflection coefficients
$K^{\tilde{z}z}{}_{p+1,T}$	$Z_{p,T}$	$X_{p,T}$	$z_{0:T}{}^{p+1}$	$z_{0:T}$	
$K^{\tilde{x}z}{}_{p+1,T}$	$Z_{p,T}$	$X_{p,T}$	$x_{0:T}{}^{p+1}$	$z_{0:T}$	
$K^{xz}{}_{p+1,T}$	$Z_{p,T}$	$X_{p,T}$	$x_{0:T}$	$z_{0:T}$	
$K^{\tilde{x}\tilde{z}}{}_{p+1,T}$	$Z_{p,T}$	$X_{p,T}$	$x_{0:T}{}^{p+1}$	$z_{0:T}{}^{p+1}$	

Table 3. Definitions of the normalized lattice RIV.

consider in this paper only the case where $NA = NB$. The more general case can be derived in a straightforward manner following the steps outlined in § 2. Table 3 summarizes the variables involved in the recursions. We will also use the following definitions

$$X_{p,T} \triangleq X_{p,p,T} \quad Z_{p,T} \triangleq Z_{p,p,T}$$

$$x_{0:T} = \begin{bmatrix} y'_0 & u'_0 \\ \vdots & \vdots \\ y'_T & u'_T \end{bmatrix} \quad z_{0:T} = \begin{bmatrix} \tilde{y}'_0 & \tilde{u}'_0 \\ \vdots & \vdots \\ \tilde{y}'_T & \tilde{u}'_T \end{bmatrix}$$

$$x_{0:T}{}^{p} = \begin{bmatrix} 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ & \\ y'_0 & u'_0 \\ \vdots & \vdots \\ y'_{T-p} & u'_{T-p} \end{bmatrix} \quad z_{0:T}{}^{p} = \begin{bmatrix} 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ & \\ \tilde{y}'_0 & \tilde{u}'_0 \\ \vdots & \vdots \\ \tilde{y}'_{T-p} & \tilde{u}'_{T-p} \end{bmatrix}$$

The recursions are obtained by making the substitutions depicted in Table 4, in the update formula (29).

The following facts are useful in interpreting the entries of this table :

(i) *order updates*

$$X_{p+1,T} = X_{p,T} + x_{0:T}{}^{p+1}, \quad Z_{p+1,T} = Z_{p,T} + z_{0:T}{}^{p+1}$$

(ii) *time and order updates*

$$X_{p+1,T+1} = X_{p,T} + x_{0:T}, \quad Z_{p+1,T+1} = Z_{p,T} + z_{0:T}$$

(iii) *time update*

$$P^{c}{}_{Z_{p,T+\pi},X_{p,T+\pi}} = \begin{bmatrix} P^{c}{}_{Z_{p,T-1},X_{p,T-1}} & 0 \\ & \vdots \\ 0 \ldots\ldots\ldots\ldots\ldots & 0 \end{bmatrix}$$

Note also that $\rho_{ZX}(U, V) = \rho'_{XZ}(V, U)$.

We can now summarize the lattice RIV algorithm by reading off the proper entries of Table 4.

Lattice RIV algorithm

We denote here $M = \dim \{[y'_T, u'_T]'\}$.

Input parameters

N = model order

λ = exponential weighting factor

σ = prior covariance

y_T, u_T = data sequence

$\tilde{y}_T$, $\tilde{u}_T$ = instrumental variable sequence

Z	X	z	x	U	V	$\rho_{Z+s,X+z}(U,V)$	$\rho_{ZX}(U,V)$	$\rho_{ZX}(x,V)$	$\rho_{ZX}(V,z)$	$\rho_{ZX}(U,z)$	$\rho_{ZX}(x,U)$	$\rho_{ZX}(x,z)$
$Z_{p,T}$	$X_{p,T}$	$z_{0:T}^{p+1}$	$x_{0:T}^{p+1}$	$x_{0:T}$	π	$\epsilon^x_{p+1,T}$	$\epsilon^x_{p,T}$	$r^x_{p,T-1}$	$\tilde{r}^{z'}_{p,T-1}$	$K^{x\tilde{z}}_{p+1,T}$	$K^{\tilde{x}x}_{p+1,T}$	$K^{\tilde{x}\tilde{z}}_{p+1,T}$
$Z_{p,T}$	$X_{p,T}$	$z_{0:T}$	$x_{0:T}$	$x_{0:T}^{p+1}$	π	$r^x_{p+1,T}$	$r^x_{p,T-1}$	$\epsilon^x_{p,1}$	$\tilde{\epsilon}^{z'}_{p,T}$	$K^{\tilde{x}z}_{p+1,T}$	$K^{x\tilde{x}}_{p+1,T}$	$K^{xz}_{p+1,T}$
$Z_{p,T}$	$X_{p,T}$	$z_{0:T}^{p+1}$	$x_{0:T}^{p+1}$	$z_{0:T}$	π	$\epsilon^z_{p+1,T}$	$\epsilon^z_{p,T}$	$r^x_{p,T-1}$	$\tilde{r}^{z'}_{p,T-1}$	$K^{z\tilde{z}}_{p+1,T}$	$K^{\tilde{x}z}_{p+1,T}$	$K^{\tilde{x}\tilde{z}}_{p+1,T}$
$Z_{p,T}$	$X_{p,T}$	$z_{0:T}$	$x_{0:T}$	$z_{0:T}^{p+1}$	π	$r^z_{p+1,T}$	$r^z_{p,T-1}$	$\epsilon^x_{p,T}$	$\tilde{\epsilon}^{z'}_{p,T}$	$K^{\tilde{z}z}_{p+1,T}$	$K^{x\tilde{z}}_{p+1,T}$	$K^{xz}_{p+1,T}$
$X_{p,T}$	$Z_{p,T}$	$x_{0:T}^{p+1}$	$z_{0:T}^{p+1}$	$x_{0:T}$	π	$\tilde{\epsilon}^x_{p+1,T}$	$\tilde{\epsilon}^x_{p,T}$	$\tilde{r}^x_{p,T-1}$	$r^{x'}_{p,T-1}$	$K^{\tilde{x}x'}_{p+1,T}$	$K^{x\tilde{z}'}_{p+1,T}$	$K^{\tilde{x}\tilde{z}'}_{p+1,T}$
$X_{p,T}$	$Z_{p,T}$	$x_{0:T}$	$z_{0:T}$	$x_{0:T}^{p+1}$	π	$\tilde{r}^x_{p+1,T}$	$\tilde{r}^x_{p,T-1}$	$\tilde{\epsilon}^z_{p,T}$	$\epsilon^{x'}_{p,T}$	$K^{x\tilde{x}'}_{p+1,T}$	$K^{\tilde{x}z'}_{p+1,T}$	$K^{xz'}_{p+1,T}$
$X_{p,T}$	$Z_{p,T}$	$x_{0:T}^{p+1}$	$z_{0:T}^{p+1}$	$z_{0:T}$	π	$\tilde{\epsilon}^z_{p+1,T}$	$\tilde{\epsilon}^z_{p,T}$	$\tilde{r}^z_{p,T-1}$	$r^{x'}_{p,T-1}$	$K^{\tilde{z}z'}_{p+1,T}$	$K^{z\tilde{z}'}_{p+1,T}$	$K^{x\tilde{z}'}_{p+1,T}$
$X_{p,T}$	$Z_{p,T}$	$x_{0:T}$	$z_{0:T}$	$z_{0:T}^{p+1}$	π	$\tilde{r}^z_{p+1,T}$	$\tilde{r}^z_{p,T-1}$	$\tilde{\epsilon}^z_{p,T}$	$\epsilon^{x'}_{p,T}$	$K^{x\tilde{z}'}_{p+1,T}$	$K^{\tilde{z}z'}_{p+1,T}$	$K^{xz'}_{p+1,T}$
$Z_{p,T}$	$X_{p,T}$	π	π	$x_{0:T}$	$x_{0:T}^{p+1}$	$K^{x\tilde{x}}_{p+1,T-1}$	$K^{x\tilde{x}}_{p+1,T}$	$\tilde{r}^z_{p,T-1}$	$r^z_{p,T-1}$	$\epsilon^z_{p,T}$	$\tilde{\epsilon}^z_{p,T}$	I
$Z_{p,T}$	$X_{p,T}$	π	π	$z_{0:T}$	$z_{0:T}^{p+1}$	$K^{z\tilde{z}}_{p+1,T-1}$	$K^{z\tilde{z}}_{p+1,T}$	$\tilde{r}^z_{p,T-1}$	$r^z_{p,T-1}$	$\epsilon^z_{p,T}$	$\tilde{\epsilon}^z_{p,T}$	I
$Z_{p,T}$	$X_{p,T}$	π	π	$x_{0:T}$	$z_{0:T}^{p+1}$	$K^{x\tilde{z}}_{p+1,T-1}$	$K^{x\tilde{z}}_{p+1,T}$	$\tilde{r}^z_{p,T-1}$	$r^z_{p,T-1}$	$\epsilon^x_{p,T}$	$\tilde{\epsilon}^x_{p,T}$	I
$Z_{p,T}$	$X_{p,T}$	π	π	$x_{0:T}^{p+1}$	$x_{0:T}$	$K^{\tilde{x}x}_{p+1,T-1}$	$K^{\tilde{x}x}_{p+1,T}$	$\tilde{\epsilon}^x_{p,T}$	$\epsilon^x_{p,T}$	$r^x_{p,T-1}$	$\tilde{r}^x_{p,T-1}$	I
$Z_{p,T}$	$X_{p,T}$	π	π	$z_{0:T}^{p+1}$	$z_{0:T}$	$K^{\tilde{z}z}_{p+1,T-1}$	$K^{\tilde{z}z}_{p+1,T}$	$\tilde{\epsilon}^z_{p,T}$	$\epsilon^z_{p,T}$	$r^z_{p,T-1}$	$\tilde{r}^z_{p,T-1}$	I
$Z_{p,T}$	$X_{p,T}$	π	π	$x_{0:T}^{p+1}$	$z_{0:T}$	$K^{\tilde{x}z}_{p+1,T-1}$	$K^{\tilde{x}z}_{p+1,T}$	$\tilde{\epsilon}^z_{p,T}$	$\epsilon^z_{p,T}$	$r^x_{p,T-1}$	$\tilde{r}^x_{p,T-1}$	I
$Z_{p,T}$	$X_{p,T}$	π	π	$x_{0:T}$	$z_{0:T}$	$K^{xz}_{p+1,T-1}$	$K^{xz}_{p+1,T}$	$\tilde{\epsilon}^z_{p,T}$	$\epsilon^z_{p,T}$	$\epsilon^x_{p,T}$	$\tilde{\epsilon}^x_{p,T}$	I
$Z_{p,T}$	$X_{p,T}$	π	π	$x_{0:T}^{p+1}$	$z_{0:T}^{p+1}$	$K^{\tilde{x}\tilde{z}}_{p+1,T-1}$	$K^{\tilde{x}\tilde{z}}_{p+1,T}$	$\tilde{r}^z_{p,T-1}$	$r^z_{p,T-1}$	$r^x_{p,T-1}$	$\tilde{r}^x_{p,T-1}$	I

Table 4. Derivation of the normalized lattice RIV.

Variables

$R_T{}^x$, $R_T{}^z$ = estimated covariance of $[y'_T, u'_T]'$, $[\tilde{y}'_T, \tilde{u}'_T]'$ $(M \times M)$

K^{xz}, K^{zz}, K^{xz}, K^{zx}, K^{zz}, K^{zz}, K^{xz}, K^{zz} = reflection coefficients $(M \times M)$

$\epsilon^x{}_{p,T}$, $\epsilon^z{}_{p,T}$, $r^x{}_{p,T}$, $r^z{}_{p,T}$ = prediction errors $(M \times 1)$

$\bar{\epsilon}^z{}_{p,T}$, $\bar{\epsilon}^z{}_{p,T}$, $\bar{r}^x{}_{p,T}$, $\bar{r}^z{}_{p,T}$ = auxiliary prediction errors $(M \times 1)$

Initialization

$R_{-1}{}^x = R_{-1}{}^z = \sigma I.$

Reflection coefficients and backward prediction errors are all initialized to zero.

Main loop

At each time step do the following :

(i) Set

$$R_T{}^x = \lambda R_{T-1}{}^x + [y'_T, u'_T]'[y'_T, u'_T]$$

$$R_T{}^z = \lambda R_{T-1}{}^z + [\tilde{y}'_T, \tilde{u}'_T]'[\tilde{y}'_T, \tilde{u}'_T]$$

$$\epsilon_{0,T}{}^x = r_{0,T}{}^x = \bar{\epsilon}_{0,T}{}^x = \bar{r}_{0,T}{}^x = (R_T{}^x)^{-1/2}[y'_T, u'_T]'$$

$$\epsilon_{0,T}{}^z = r_{0,T}{}^z = \bar{\epsilon}_{0,T}{}^z = \bar{r}_{0,T}{}^z = (R_T{}^z)^{-1/2}[\tilde{y}'_T, \tilde{u}'_T]'$$

(ii) For $p = 0, \ldots, \min\{N, T\} - 1$, do :

$$K^{xz}{}_{p+1,T} = \bar{F}^{-1}(K^{xz}{}_{p+1,T-1}, \bar{r}^x{}_{p,T-1}, r^x{}_{p,T-1}, \epsilon^z{}_{p,T}, \bar{\epsilon}^z{}_{p,T}, I)$$

$$K^{zz}{}_{p+1,T} = \bar{F}^{-1}(K^{zz}{}_{p+1,T-1}, \bar{r}^z{}_{p,T-1}, r^z{}_{p,T-1}, \epsilon^z{}_{p,T}, \bar{\epsilon}^z{}_{p,T}, I)$$

$$K^{xz}{}_{p+1,T} = \bar{F}^{-1}(K^{xz}{}_{p+1,T-1}, \bar{r}^z{}_{p,T-1}, r^z{}_{p,T-1}, \epsilon^x{}_{p,T}, \bar{\epsilon}_{p,T}{}^x, I)$$

$$K^{zx}{}_{p+1,T} = \bar{F}^{-1}(K^{zx}{}_{p+1,T}, \bar{\epsilon}^x{}_{p,T}, \epsilon^x{}_{p,T}, r^x{}_{p,T-1}, \bar{r}^x{}_{p,T-1}, I)$$

$$K^{zz}{}_{p+1,T} = \bar{F}^{-1}(K^{zz}{}_{p+1,T-1}, \bar{\epsilon}^z{}_{p,T}, \epsilon^z{}_{p,T}, r^z{}_{p,T-1}, \bar{r}^z{}_{p,T-1}, I)$$

$$K^{zz}{}_{p+1,T} = \bar{F}^{-1}(K^{zz}{}_{p+1,T-1}, \bar{\epsilon}^z{}_{p,T}, \epsilon^z{}_{p,T}, r^x{}_{p,T-1}, \bar{r}^x{}_{p,T-1}, I)$$

$$K^{xz}{}_{p+1,T} = \bar{F}^{-1}(K^{xz}{}_{p+1,T-1}, \bar{\epsilon}^z{}_{p,T}, \epsilon^z{}_{p,T}, \epsilon^x{}_{p,T}, \bar{\epsilon}^x{}_{p,T}, I)$$

$$K^{zz}{}_{p+1,T} = \bar{F}^{-1}(K^{zz}{}_{p+1,T-1}, \bar{r}^z{}_{p,T-1}, r^z{}_{p,T-1}, r^x{}_{p,T-1}, \bar{r}^x{}_{p,T-1}, I)$$

$$\epsilon^x{}_{p+1,T} = \bar{F}(\epsilon^x{}_{p,T}, r^x{}_{p,T-1}, \bar{r}^{z'}{}_{p,T-1}, K^{xz}{}_{p+1,T}, K^{zx}{}_{p+1,T}, K^{zz}{}_{p+1,T})$$

$$r^x{}_{p+1,T} = \bar{F}(r^x{}_{p,T-1}, \epsilon^x{}_{p,T}, \bar{\epsilon}^{z'}{}_{p,T}, K^{zz}{}_{p+1,T}, K^{xz}{}_{p+1,T}, K^{xz}{}_{p+1,T})$$

$$\epsilon^z{}_{p+1,T} = \bar{F}(\epsilon^z{}_{p,T}, r^x{}_{p,T-1}, \bar{r}^{z'}{}_{p,T-1}, K^{zz}{}_{p+1,T}, K^{zz}{}_{p+1,T}, K^{zz}{}_{p+1,T})$$

$$r^z{}_{p+1,T} = \bar{F}(r^z{}_{p,T-1}, \epsilon^x{}_{p,T}, \bar{\epsilon}^{z'}{}_{p,T}, K^{zz}{}_{p+1,T}, K^{xz}{}_{p+1,T}, K^{xz}{}_{p+1,T})$$

$$\bar{\epsilon}^x{}_{p+1,T} = \bar{F}(\bar{\epsilon}^x{}_{p,T}, \bar{r}^z{}_{p,T-1}, r^{x'}{}_{p,T-1}, K^{zx'}{}_{p+1,T}, K^{xz'}{}_{p+1,T}, K^{zz'}{}_{p+1,T})$$

$$\bar{r}^x{}_{p+1,T} = \bar{F}(\bar{r}^x{}_{p,T-1}, \bar{\epsilon}^z{}_{p,T}, \epsilon^{x'}{}_{p,T}, K^{zx'}{}_{p+1,T}, K^{zz'}{}_{p+1,T}, K^{xz'}{}_{p+1,T})$$

$$\bar{\epsilon}^z{}_{p+1,T} = \bar{F}(\bar{\epsilon}^z{}_{p,T}, \bar{r}^z{}_{p,T-1}, r^{x'}{}_{p,T-1}, K^{zz'}{}_{p+1,T}, K^{zz'}{}_{p+1,T}, K^{zz'}{}_{p+1,T})$$

$$\bar{r}^z{}_{p+1,T} = \bar{F}(\bar{r}^z{}_{p,T-1}, \bar{\epsilon}^z{}_{p,T}, \epsilon^{x'}{}_{p,T}, K^{xz'}{}_{p+1,T}, K^{zz'}{}_{p+1,T}, K^{xz'}{}_{p+1,T})$$

As can be seen from these equations, the normalized lattice RIV is fairly complex. This is due to the more complicated projection update formula and to the fact that various identities which were true for the symmetric projection operator, no longer hold (e.g. $\rho_{ZX}(U, V) \neq \rho'_{ZX}(V, U)$). The unnormalized version of the lattice RIV turns out to be considerably simpler than the normalized recursions presented above (see Appendix B). This is different from the situation in the lattice RLS where the normalized version is the simpler one. The $\{a_i, b_i\}$ parameters can be recovered by computing the impulse response of the variance normalized lattice RIV.

Finally we note that the unnormalized lattice RIV has been developed independently by several authors (see, for example, Samson 1982, Cadzow and Moses 1981). An approximate lattice RIV was presented by Prevosto *et al.* (1982).

4. The lattice extended least-squares algorithm

In this section we consider the following ARMAX model

$$y_t = -\sum_{i=1}^{NA} a_i y_{t-i} + \sum_{i=0}^{NB} b_i u_{t-i} + \sum_{i=1}^{NC} c_i v_{t-i} + v_t \tag{31}$$

where v_t is an unmeasurable white-noise disturbance process. If it were possible to measure v_t, this would have been a standard linear regression problem, and the RLS algorithm could be applied. The ELS method is based on the idea of replacing v_t by its estimate, the prediction error ϵ_t (Söderström *et al.* 1978, Panuska 1969, Solo 1979). The lattice ELS will, therefore, consist of two steps: (i) use the lattice form as a prediction filter to compute ϵ_t; (ii) use the lattice RLS for the known input case (with y_t, u_t, $v_t = \epsilon_t$) to update the parameter estimates.

To describe the lattice ELS for the model presented above we must first present the basic update formula of the lattice RLS for ARMAX models.

Lattice RLS (ARMAX) algorithm

We assume that $NA \geqslant NB \geqslant NC$. For other cases we simply have to reorder the inputs y_t, u_t, v_t so that the corresponding model orders appear in decreasing order.

$K^y, K^u, K^r, \bar{K}, \bar{K}^r, K$ = reflection coefficients

$\epsilon^y, \epsilon^u, \epsilon^r, \bar{\epsilon}, \bar{\epsilon}^r, \epsilon$ = forward prediction errors (dim $\{y\}$, dim $\{u\}$, dim $\{v\}$, dim $\{[y', u']\}$, dim $\{v\}$, dim $\{[y', u', v']\}$)

$r^y, \bar{r}, r$ = backward prediction errors (dim $\{y\}$, dim $\{[y', u']\}$, dim $\{([y', u', v']\}$)

(i) for $p = 0, \ldots, NA - NB$ (or up to T, during start-up)

$$\left.\begin{aligned} K^y_{p+1,T} &= F^{-1}(K^y_{p+1,T-1}, r^y_{p,T-1}, \epsilon^y_{p,T}) \\ \epsilon^y_{p+1,T} &= F(\epsilon^y_{p,T}, r^y_{p,T-1}, K_{p+1,T}) \\ r^y_{p+1,T} &= F(r^y_{p,T-1}, \epsilon^y_{p,T}, K'_{p+1,T}) \end{aligned}\right\} \quad \text{skip for } p = NA - NB$$

$$K^u_{p+1,T} = F^{-1}(K^u_{p+1,T-1}, r^{y'}_{p,T}, \epsilon^u_{p,T})$$
$$K^v_{p+1,T} = F^{-1}(K^v_{p+1,T-1}, r^{y'}_{p,T}, \epsilon^v_{p,T})$$
$$\epsilon^u_{p+1,T} = F(\epsilon^u_{p,T}, r^y_{p,T}, K^u_{p+1,T})$$
$$\epsilon^v_{p+1,T} = F(\epsilon^v_{p,T}, r^y_{p,T}, K^v_{p+1,T})$$

(ii) set

$$\bar{\epsilon}_{0,T} = [\epsilon^{y'}_{NA-NB}, \epsilon^{u'}_{NA-NB+1,T}]'$$
$$\bar{r}_{0,T} = [r^{y'}_{NA-NB,T}, \epsilon^{u'}_{NA-NB+1,T}]'$$
$$\bar{\epsilon}^v_{0,T} = \epsilon^v_{NA-NB+1,T}$$

(iii) for $p = 0$ to $NB - NC$ (or up to $T - NA + NB$ during start-up)

$$\left.\begin{aligned} \bar{K}_{p+1,T} &= F^{-1}(\bar{K}_{p+1,T-1}, \bar{r}'_{p,T-1}, \bar{\epsilon}_{p,T}) \\ \bar{\epsilon}_{p+1,T} &= F(\bar{\epsilon}_{p,T-1}, \bar{r}_{p,T-1}, \bar{K}_{p+1,T}) \\ \bar{r}_{p+1,T} &= F(\bar{r}_{p,T-1}, \bar{\epsilon}_{p,T}, \bar{K}'_{p+1,T}) \end{aligned}\right\} \text{ skip for } p = NB - NC$$
$$\bar{K}^v_{p+1,T} = F^{-1}(\bar{K}^v_{p+1,T-1}, \bar{r}^{y'}_{p,T}, \bar{\epsilon}^v_{p,T})$$
$$\bar{\epsilon}^v_{p+1,T} = F(\bar{\epsilon}^v_{p,T}, \bar{r}^y_{p,T}, \bar{K}^v_{p+1,T})$$

(iv) set

$$\epsilon_{0,T} = [\bar{\epsilon}'_{0,T}, \bar{\epsilon}^{v'}_{NB-NC+1}]'$$
$$r_{0,T} = [\bar{r}'_{0,T}, \bar{\epsilon}^{v'}_{NB-NC+1}]'$$

(v) for $p = 0$ to $NC - 1$ (or up to $T - NA + NC - 1$ during start-up)

$$K_{p+1,T} = F^{-1}(K_{p+1,T-1}, r'_{p,T-1}, \epsilon_{p,T})$$
$$\epsilon_{p+1,T} = F(\epsilon_{p,T}, r_{p,T-1}, K_{p+1,T})$$
$$r_{p+1,T} = F(r_{p,T-1}, \epsilon_{p,T}, K'_{p+1,T})$$

The corresponding lattice structure is depicted in Fig. 2. The detailed structure of the various sections is similar to that of the lattice RLS of Fig. 1, with some obvious modifications.

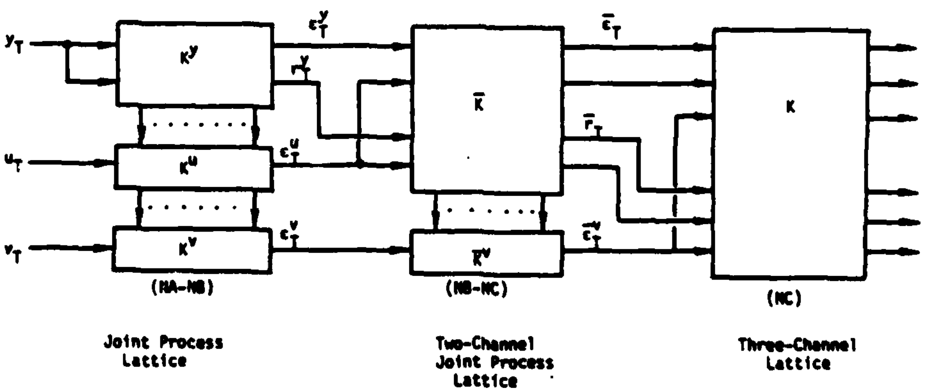


Figure 2. The ARMAX RLS lattice.

The lattice ELS algorithm can now be summarized. The RLS lattice update described above will be used in two modes. One is a prediction mode, in which all the reflection coefficient updates are skipped and the old values of the state variables $\{r^y{}_{p,T-1}, r^u{}_{p,T-1}, r^v{}_{p,T-1}, \bar{r}_{p,T-1}, r_{p,T-1}\}$ are retained. The second is a regular update mode in which both reflection coefficients and state variables are updated.

Lattice ELS algorithm

All matrix square-roots are lower triangular.

Initialization

$$R_{-1}{}^y = R_{-1}{}^u = R_{-1}{}^v = \sigma I$$

All reflection coefficients and backward prediction errors are initialized to zero.

Main loop

At each time step do the following:

(i) compute prediction errors

$$R_T{}^y = \lambda R_{T-1}{}^y + y_T y'_T$$

$$R_T{}^u = \lambda R_{T-1}{}^u + u_T u'_T$$

$$\epsilon_{0,T}{}^y = r_{0,T}{}^y = (R_T{}^y)^{-1/2} y_T$$

$$\epsilon_{0,T}{}^u = (R_T{}^u)^{-1/2} u_T$$

$$\epsilon_{0,T}{}^v = 0$$

Call lattice RLS update (ARMAX), in prediction mode

(ii) set

$$\hat{u}_T = \text{last entry of } \epsilon_{NC,T}$$

(iii) update lattice variables

$$R_T{}^v = \lambda R_{T-1}{}^v + \hat{v}_T \hat{v}'_T$$

$$\epsilon_{0,T}{}^v = (R_T{}^v)^{-1/2} \hat{v}_T$$

Call lattice RLS (ARMAX) in update mode.

As before, the parameter estimates $\{a_i, b_i, c_i\}$ can be recovered by looking at the impulse response of the lattice form depicted in Fig. 2, from the inputs y, u, v to the prediction error output. For a more detailed discussion of the lattice ELS for the case of ARMA (N, N) models (i.e. $NA = NC$, $NB = 0$), see Lee *et al.* (1980). An approximate lattice ELS algorithm for general ARMA processes was presented in Benveniste and Chaure (1981).

5. The lattice recursive maximum-likelihood algorithm

The RML algorithm has improved asymptotic convergence properties compared with the ELS algorithm presented above. To simplify the presentation we consider in this section ARMA models rather than the more general ARMAX models, i.e.

$$y_t = -\sum_{i=1}^{NA} a_i y_{t-i} + \sum_{i=1}^{NC} c_i v_{t-i} + v_t \tag{32}$$

where v_t is a white-noise process. The RML algorithm (Söderström *et al.* 1978, Söderström 1973, Ljung 1977, 1981) can be summarized as follows. Let

$$\theta = [a_1, \ldots, a_{NA}, c_1, \ldots, c_{NC}]^T = \text{parameter vector}$$

$$\phi_t = [-y_{t-1}, \ldots, -y_{t-NA}, e_{t-1}, \ldots, e_{t-NC}]^T = \text{data vector}$$

$$\psi_t = [-\tilde{y}_{t-1}, \ldots, -\tilde{y}_{t-NA}, \tilde{e}_{t-1}, \ldots, \tilde{e}_{t-NC}]^T = \text{filtered data vector}$$

where e_t, $\tilde{e}_t$, $\tilde{y}_t$ will be defined later. The update equations are

$$\left.\begin{aligned} \epsilon_t &= y_t - \phi_t^T \hat{\theta}_{t-1} \\ \hat{\theta}_t &= \hat{\theta}_{t-1} + P_t \psi_t \epsilon_t \\ P_t^{-1} &= P_{t-1}^{-1} + \psi_t \psi_t^T \\ e_t &= y_t - \phi_t^T \hat{\theta}_t \end{aligned}\right\} \tag{33}$$

The filtered quantities are obtained by

$$\left.\begin{aligned} \tilde{e}_t &= [1/\hat{C}_t(z)]e_t \\ \tilde{y}_t &= [1/\hat{C}_t(z)]y_t \end{aligned}\right\} \tag{34}$$

where

$$\hat{C}_t(z) = 1 + \hat{c}_1(t)z^{-1} + \ldots + \hat{c}_{NC}(t)z^{-NC} \tag{35}$$

where z^{-1} is the unit delay operator. Note that (33) can be rewritten as

$$P_t^{-1}\hat{\theta}_t = P_t^{-1}\hat{\theta}_{t-1} + \psi_t\epsilon_t = P_{t-1}^{-1}\hat{\theta}_{t-1} + \psi_t(\epsilon_t + \psi_t^T \hat{\theta}_{t-1}) \tag{36}$$

Let us define

$$x_t = \epsilon_t + \psi_t^T \hat{\theta}_{t-1} \tag{37}$$

and sum up the difference equation (36) to get

$$P_t^{-1}\hat{\theta}_t = \sum_{i=1}^{t} \psi_i x_i \tag{38}$$

or

$$\hat{\theta}_t = \left[\sum_{i=1}^{t} \psi_i \psi_i^T\right]^{-1} \sum_{i=1}^{t} \psi_i x_i \tag{39}$$

Equation (39) can be recognized as the solution to the problem of estimating the process x_t from the components of the vector ψ_t. Thus, given the variables x_t, $\tilde{y}_t$, $\tilde{e}_t$ we can apply the recursive least-squares algorithm to estimate the parameter vector θ. The joint estimation problem described above can be solved by the following joint-process lattice form.

Joint-process two-channel lattice

We assume here $NA = NC = N$.

Initialization

$$R_0 = \sigma I,\ R_0^x = \sigma$$

$$\epsilon_{0,0} = r_{0,0} = 0,\ \epsilon_{0,0}^x = 0$$

$$K_{p,0} = 0,\ K_{p,0}^x = 0 \text{ for } p = 1, \ldots, N$$

$$Z_T = \begin{bmatrix} Y_T \\ U_T \end{bmatrix}$$

Main loop

$$R_T = \lambda R_{T-1} + Z_T Z'_T \qquad R_T^x = \lambda R^x_{T-1} + X_T X'_T \tag{40 a}$$

$$\epsilon_{0,T} = r_{0,T} = R_T^{-1/2} Z_T \quad \epsilon_{0,T}^x = (R_T^x)^{-1/2} X_T \tag{40 b}$$

For $p = 0$ to N, do

$$K_{p+1,T} = F^{-1}(K_{p+1,T-1}, r'_{p,T-1}, \epsilon_{p,T}) \tag{41 a}$$

$$K^x_{p+1,T} = F^{-1}(K^x_{p+1,T-1}, r'_{p,T}, \epsilon^x_{p,T}) \tag{41 b}$$

$$\epsilon_{p+1,T} = F(\epsilon_{p,T}, r_{p,T-1}, K_{p+1,T}) \tag{41 c}$$

$$r_{p+1,T} = F(r_{p,T-1}, \epsilon_{p,T}, K'_{p+1,T}) \tag{41 d}$$

$$\epsilon^x_{p+1,T} = F(\epsilon^x_{p,T}, r_{p,T}, K^x_{p+1,T}) \tag{41 e}$$

The algorithm described above will both update the parameter estimates (reflection coefficients K, K^x) and filter incoming data to compute a set of prediction errors. Sometimes we want to use this lattice structure for filtering only. In this case only (40 *b*), (41 *c*), (41 *d*) and (41 *e*) need to be used. To distinguish between these two cases we will call LATUP the algorithm that performs the full computation (40)–(41) and LATFIL the algorithm that does filtering only.

The lattice structure described above can now be used to implement the RML algorithm (Friedlander *et al.* 1981). This will require several steps of filtering and parameter updating. The following set of equations summarizes the RML algorithm in non-lattice form

$$\epsilon_T = y_T + \sum_{i=1}^{NA} \hat{a}_i(T-1) y_{T-i} - \sum_{i=1}^{NC} \hat{c}_i(T-1) e_{T-i} \tag{42 a}$$

$$-\hat{x}_T = 0 + \sum_{i=1}^{NA} \hat{a}_i(T-1) \tilde{y}_{T-i} - \sum_{i=1}^{NC} \hat{c}_i(T-1) \tilde{e}_{T-i} \tag{42 b}$$

$$x_T = \epsilon_T + \hat{x}_T : \quad \text{perform least-squares parameter update} \tag{42 c}$$

$$e_T = y_T + \sum_{i=1}^{NA} \hat{a}_i(T) y_{T-i} - \sum_{i=1}^{NC} \hat{c}_i(T) e_{T-i} \tag{42 d}$$

$$\tilde{y}_T = y_T + 0 - \sum_{i=1}^{NC} \hat{c}_i(T) \tilde{y}_{T-i} \tag{42 e}$$

$$\tilde{e}_T = e_T + 0 - \sum_{i=1}^{n} \hat{c}_i(T) \tilde{e}_{T-i} \tag{42 f}$$

Call	Y	U	X	ϵ^x	Equation
LATFIL (1)	y_{t-1}	e_{t-1}	y_t	ϵ_t	(42 *a*)
LATFIL (2)	$\tilde{y}_{t-1}$	$\tilde{e}_{t-1}$	0	$-\hat{x}_t$	(42 *b*)
LATUP (2)	$\tilde{y}_{t-1}$	$\tilde{e}_{t-1}$	x_t	ϵ_t^x	(42 *c*)
LATFIL (1)	y_{t-1}	e_{t-1}	y_t	e_t	(42 *d*)
LATFIL (3)	0	$\tilde{y}_{t-1}$	y_t	$\tilde{y}_t$	(42 *e*)
LATFIL (4)	0	$\tilde{e}_{t-1}$	e_t	$\tilde{e}_t$	(42 *f*)

Table 5. The RML lattice.

This set of equations can be implemented by repeated calls of the lattice form described above. The input and output for each lattice call are summarized in Table 5. Note that four different 'state vectors' need to be stored corresponding to θ, ψ and the pre-filters for y and e. These four cases are distinguished in Table 5 by the index of the lattice call (for example, LATFIL (1) represents the filters with y_T, e_T as inputs, while LATFIL (2) has $\tilde{y}_T$, $\tilde{e}_T$ as inputs).

6. Conclusions

The lattice equivalents of several system identification algorithms were presented. These algorithms provide a computationally efficient recursive solution of linear least-squares estimation problems. In the area of digital signal processing lattice filters are often preferred over their tapped-delay-line equivalents because of their relative insensitivity to roundoff errors. In adaptive processing applications, lattice filters have shown improved convergence behaviour compared to the popular Widrow–Hoff LMS algorithm. Lattice structures also lead to processing architectures that are quite different from those related to the RLS and similar algorithms. This modular pipelined architecture has potential advantages in hardware and VLSI implementations of the algorithms.

Relatively little work has been done in the application of lattice forms to system identification and adaptive control. Considerably more analysis and simulation studies are needed to assess the usefulness of the techniques presented in this paper. Of special interest would be tests performed on finite word length machines and plants with high order dynamics. These conditions often lead to numerical problems in standard recursive parameter estimation algorithms. It is hoped that this paper will stimulate research in this area.

Appendix A.

Derivation of the update formula for non-symmetric projection operators

Definitions

$$P_{Z,X} \triangleq Z(X'Z)^{-1}X'$$

$$P^c_{ZX} \triangleq I - P_{ZX}$$

$$Z+z \triangleq [Z \mid z] \quad X+x \triangleq [X \mid x]$$

Update formula for P^c

$$P^c_{Z+z,X+x} = P^c_{ZX} - P^c_{ZX}\, z[x'P^c_{ZX}\, z]^{-1}x'P^c_{ZX} \tag{A 1}$$

Proof

$$P_{Z+z,X+x} = [Z \vdots z]\begin{bmatrix} X'Z & X'z \\ x'Z & x'z \end{bmatrix}^{-1}\begin{bmatrix} X' \\ x' \end{bmatrix} \tag{A 2}$$

Use the following matrix identity

$$\left.\begin{aligned} &\begin{bmatrix} A & B \\ C & D \end{bmatrix}^{-1} = \begin{bmatrix} A^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} -A^{-1}B \\ I \end{bmatrix}\Delta_D^{-1}[-CA^{-1}\; I] \\ &\Delta_D \triangleq D - CA^{-1}B \end{aligned}\right\} \tag{A 3}$$

to invert the matrix in (A 2) to get

$$\left.\begin{aligned} &P_{Z+z,X+x} = Z(X'Z)^{-1}X' + (I - Z(X'Z)^{-1}X')z\Delta^{-1}x'(I - Z(X'Z)^{-1}X') \\ &\Delta = x'[I - Z(X'Z)^{-1}X']z \end{aligned}\right\} \tag{A 4}$$

Equation (A 1) follows directly from (A 4) and the definitions.

Normalization

$$\rho_{ZX}(U, V) \triangleq [U'P^c_{ZX}\, U]^{-1/2}U'P^c_{ZX}\, V[V'P^c_{ZX}\, V]^{-T/2}$$

$$[U'P^c_{ZX}\, U]^{-1/2}U'P^c_{Z+z,X+x}\, V[V'P^c_{ZX}\, V]^{-T/2}$$
$$= \rho_{ZX}(U, V) - \rho_{ZX}(U, z)\rho^{-1}_{ZX}(x, z)\rho_{ZX}(x, V)$$

Set $V = U$

$$[U'P^c_{ZX}\, U]^{-1/2}U'P^c_{Z+z,X+x}\, U[U'P^c_{ZX}\, U]^{-T/2}$$
$$= I - \rho_{ZX}(U, z)\rho_{ZX}^{-1}(z, x)\rho_{ZX}(x, U)$$

Set $U = V$

$$[V'P^c_{ZX}\, V]^{-1/2}V'P^c_{Z+z,X+x}\, V[V'P^c_{ZX}\, V]^{-T/2}$$
$$= I - \rho_{ZX}(V, z)\rho_{ZX}^{-1}(z, x)\rho_{ZX}(x, V)$$

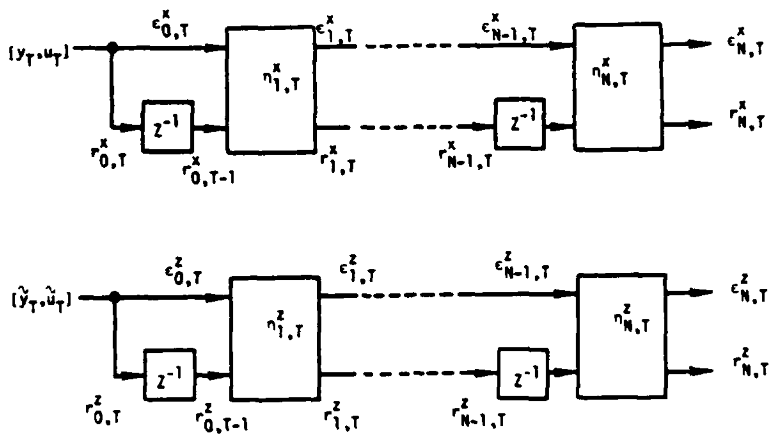
Normalizing $U'P^c_{ZX}\, V'$ by the square-roots of the last two equations gives

$$\begin{aligned} \rho_{Z+z,X+x}(U, V) = {} & [I - \rho_{ZX}(U, z)\rho_{ZX}^{-1}(x, z)\rho_{ZX}(x, U)]^{-1/2} \\ & \times [\rho_{ZX}(U, V) - \rho_{ZX}(U, z)\rho_{ZX}^{-1}(x, z)\rho_{ZX}(x, V)] \\ & \times [I - \rho_{ZX}(V, z)\rho_{ZX}^{-1}(x, z)\rho_{ZX}(x, V)]^{-T/2} \end{aligned}$$

Appendix B

The unnormalized lattice RIV

The unnormalized lattice RIV consists of two RLS-type lattice filters, as depicted in Fig. 3. The input to the upper lattice is the data sequence x_t and to the lower lattice the instrumental variable sequence z_t. The reflection coefficients of the two filters are determined by a common set of coefficients : K^{xz}, K^{zz}, K^{zz}, K^{zx}. Table 6 summarizes the definitions of all the variables in



$$\eta^x_{p,T} = \begin{bmatrix} I & -K^{x\bar{z}}_{p,T}(K^{\bar{x}\bar{z}}_{p,T})^{-1} \\ -K^{\bar{x}z}_{p,T}(K^{xz}_{p,T})^{-1} & I \end{bmatrix}$$

$$\eta^z_{p,T} = \begin{bmatrix} I & -K^{\bar{x}\bar{z}'}_{p,T}(K^{\bar{x}\bar{z}'}_{p,T})^{-1} \\ -K^{x\bar{z}'}_{p,T}(K^{xz'}_{p,T})^{-1} & I \end{bmatrix}$$

Figure 3. The unnormalized lattice RIV.

Z	X	U	V	$U'P^c_{ZX}V$
$Z_{p,T}$	$X_{p,T}$	$x_{0:T}$	π	$\epsilon^x_{p,T}$
$Z_{p,T}$	$X_{p,T}$	$x_{0:T}{}^{p+1}$	π	$r^x_{p,T-1}$
$X_{p,T}$	$Z_{p,T}$	$z_{0:T}$	π	$\epsilon^z_{p,T}$
$X_{p,T}$	$Z_{p,T}$	$z_{0:T}{}^{p+1}$	π	$r^z_{p,T-1}$
$Z_{p,T}$	$X_{p,T}$	$x_{0:T}$	$z_{0:T}{}^{p+1}$	$K^{xz}{}_{p+1,T}$
$Z_{p,T}$	$X_{p,T}$	$x_{0:T}{}^{p+1}$	$z_{0:T}{}^{p+1}$	$K^{xz}{}_{p+1,T}$
$Z_{p,T}$	$X_{p,T}$	$x_{0:T}{}^{p+1}$	$z_{0:T}$	$K^{xz}{}_{p+1,T}$
$Z_{p,T}$	$X_{p,T}$	$x_{0:T}$	$z_{0:T}$	$K^{xz}{}_{p+1,T}$
$Z_{p,T}$	$X_{p,T}$	π	π	$1-\gamma_{p-1,T-1}$

Table 6. Definitions of variables.

Z	X	z	x	U	V	$U'P^c_{Z+z,X+x}V$	$U'P^c_{ZX}z$	$U'P^c_{ZX}z$	$x'P^c_{ZX}z$	$x'P^c_{XZ}V$
$Z_{p,T}$	$X_{p,T}$	$z_{0:T}^{p+1}$	$x_{0:T}^{p+1}$	$x_{0:T}$	π	$\epsilon^x_{p+1,T}$	$\epsilon^x_{p,T}$	$K^{xz}_{p+1,T}$	$K^{xz}_{p+1,T}$	$r^x_{p,T-1}$
$Z_{p,T}$	$X_{p,T}$	$z_{0:T}$	$x_{0:T}$	$x_{0:T}^{p+1}$	π	$r^x_{p+1,T}$	$r^x_{p,T-1}$	$K^{xz}_{p+1,T}$	$K^{xz}_{p+1,T}$	$\epsilon^x_{p,T}$
$X_{p,T}$	$Z_{p,T}$	$x_{0:T}^{p+1}$	$z_{0:T}^{p+1}$	$z_{0:T}$	π	$\epsilon^z_{p+1,T}$	$\epsilon^z_{p,T}$	$K^{xz'}_{p+1,T}$	$K^{xz'}_{p+1,T}$	$r^z_{p,T-1}$
$X_{p,T}$	$Z_{p,T}$	$x_{0:T}$	$z_{0:T}$	$z_{0:T}^{p+1}$	π	$r^z_{p+1,T}$	$r^z_{p,T-1}$	$K^{xz'}_{p+1,T}$	$K^{xz'}_{p+1,T}$	$\epsilon^z_{p,T}$
$Z_{p,T}$	$X_{p,T}$	π	π	$x_{0:T}$	$z_{0:T}^{p+1}$	$K^{xz}_{p+1,T-1}$	$K^{xz}_{p+1,T}$	$\epsilon^x_{p,T}$	$1-\gamma_{p-1,T-1}$	$r^{z'}_{p,T-1}$
$Z_{p,T}$	$X_{p,T}$	π	π	$x_{0:T}^{p+1}$	$z_{0:T}^{p+1}$	$K^{xz}_{p+1,T-1}$	$K^{xz}_{p+1,T}$	$r^x_{p,T-1}$	$1-\gamma_{p-1,T-1}$	$r^{z'}_{p,T-1}$
$Z_{p,T}$	$X_{p,T}$	π	π	$x_{0:T}^{p+1}$	$z_{0:T}$	$K^{xz}_{p+1,T-1}$	$K^{xz}_{p+1,T}$	$r^x_{p,T-1}$	$1-\gamma_{p-1,T-1}$	$\epsilon^{z'}_{p,T}$
$Z_{p,T}$	$X_{p,T}$	π	π	$x_{0:T}$	$z_{0:T}$	$K^{xz}_{p+1,T-1}$	$K^{xz}_{p+1,T}$	$\epsilon^x_{p,T}$	$1-\gamma_{p-1,T-1}$	$\epsilon^{z'}_{p,T}$
$Z_{p,T}$	$X_{p,T}$	$z_{0:T}$	$x_{0:T}$	π	π	$1-\gamma_{p,T}$	$1-\gamma_{p-1,T-1}$	$\epsilon^{z'}_{p,T}$	$K^{xz}_{p+1,T}$	$\epsilon^x_{p,T}$
$Z_{p,T}$	$X_{p,T}$	$z_{0:T}^{p+1}$	$x_{0:T}^{p+1}$	π	π	$1-\gamma_{p,T-1}$	$1-\gamma_{p-1,T-1}$	$r^{z'}_{p,T-1}$	$K^{xz}_{p+1,T}$	$r^x_{p,T-1}$
$Z_{p,T}$	$X_{p,T}$	$z_{0:T}^{p+1}$	$x_{0:T}^{p+1}$	$x_{0:T}$	$z_{0:T}$	$K^{xz}_{p+2,T}$	$K^{xz}_{p+1,T}$	$K^{xz}_{p+1,T}$	$K^{xz}_{p+1,T}$	$K^{xz}_{p+1,T}$
$Z_{p,T}$	$X_{p,T}$	$z_{0:T}$	$x_{0:T}$	$x_{0:T}^{p+1}$	$z_{0:T}^{p+1}$	$K^{xz}_{p+2,T+1}$	$K^{xz}_{p+1,T}$	$K^{xz}_{p+1,T}$	$K^{xz}_{p+1,T}$	$K^{xz}_{p+1,T}$

Table 7. The unnormalized lattice RIV.

the lattice RIV. Making the proper substitutions in the update formula for $V'P^{v}_{ZX}\, V$ gives the set of lattice recursions. The necessary substitutions are summarized in Table 7. Reading off the entries of this table gives the following equations

$$\left.\begin{aligned} \epsilon^{x}_{p+1,T} &= \epsilon^{x}_{p,T} - K^{x\bar{z}}_{p+1,T}(K^{\bar{x}\bar{z}}_{p+1,T})^{-1} r^{x}_{p,T-1} && \text{(B 1)}\\ r^{x}_{p+1,T} &= r^{x}_{p,T-1} - K^{\bar{x}z}_{p+1,T}(K^{xz}_{p+1,T})^{-1} \epsilon^{x}_{p,T} && \text{(B 2)} \end{aligned}\right\} \text{lattice 1}$$

$$\left.\begin{aligned} \epsilon^{z}_{p+1,T} &= \epsilon^{z}_{p,T} - K^{\bar{x}z'}_{p+1,T}(K^{\bar{x}\bar{z}'}_{p+1,T})^{-1} r^{z}_{p,T-1} && \text{(B 3)}\\ r^{z}_{p+1,T} &= r^{z}_{p,T-1} - K^{x\bar{z}'}_{p+1,T}(K^{xz'}_{p+1,T})^{-1} \epsilon^{z}_{p,T} && \text{(B 4)} \end{aligned}\right\} \text{lattice 2}$$

$$\left.\begin{aligned} K^{x\bar{z}}_{p+1,T} &= \lambda K^{x\bar{z}}_{p+1,T-1} + \epsilon^{x}_{p,T} r^{z'}_{p,T-1}/(1-\gamma_{p-1,T-1}) && \text{(B 5)}\\ K^{\bar{x}\bar{z}}_{p+1,T} &= \lambda K^{\bar{x}\bar{z}}_{p+1,T-1} + r^{x}_{p,T-1}\, r^{z'}_{p,T-1}/(1-\gamma_{p-1,T-1}) && \text{(B 6)}\\ K^{\bar{x}z}_{p+1,T} &= \lambda K^{\bar{x}z}_{p+1,T-1} + r^{x}_{p,T-1} \epsilon^{z'}_{p,T}/(1-\gamma_{p-1,T-1}) && \text{(B 7)}\\ K^{xz}_{p+1,T} &= \lambda K^{xz}_{p+1,T-1} + \epsilon^{x}_{p,T}\, \epsilon^{z'}_{p,T}/(1-\gamma_{p-1,T-1}) && \text{(B 8)} \end{aligned}\right\} \text{time updates}$$

$$\left.\begin{aligned} \gamma_{p,T} &= \gamma_{p-1,T-1} + \epsilon^{z'}_{p,T}(K^{xz}_{p+1,T})^{-1}\epsilon^{x}_{p,T} && \text{(B 9)}\\ K^{\bar{x}\bar{z}}_{p+2,T+1} &= K^{\bar{x}\bar{z}}_{p+1,T} + K^{\bar{x}z}_{p+1,T}(K^{xz}_{p+1,T})^{-1}K^{x\bar{z}}_{p+1,T} && \text{(B 10)} \end{aligned}\right\} \text{time and order update}$$

$$\left.\begin{aligned} K^{xz}_{p+2,T} &= K^{xz}_{p+1,T} + K^{x\bar{z}}_{p+1,T}(K^{\bar{x}\bar{z}}_{p+1,T})^{-1}K^{\bar{x}z}_{p+1,T} && \text{(B 11)}\\ \gamma_{p,T-1} &= \gamma_{p-1,T} + r^{z'}_{p,T-1}(K^{\bar{x}\bar{z}}_{p+1,T})^{-1} r^{x}_{p,T-1} && \text{(B 12)} \end{aligned}\right\} \text{order update}$$

Note that we have introduced the exponential weighting factor λ into the time update equations (B 5)–(B 8).

The complete RIV algorithm can be implemented in several ways using these equations. For starting up the algorithm it is necessary to use the order (or time and order) update equations (B 10) and (B 12) for K^{xz}, $K^{\bar{x}\bar{z}}$. Afterwards the time update equations (B 6) and (B 8) will be used instead. The initial conditions for the algorithm at time step T are

$$\epsilon^{x}_{0,T} = r^{x}_{0,T} = x_T$$

$$\epsilon^{z}_{0,T} = r^{z}_{0,T} = z_T$$

$$\gamma_{-1,T} = 0$$

$$K^{\bar{x}\bar{z}}_{0,T} = K^{\bar{x}z}_{0,T+1} = \sum_{i=0}^{T} x_T z'_T = K^{xz}_{0,T-1} + x_T z'_t$$

Before start-up all the reflection coefficients and state variables are set to zero. A complete description of the unnormalized lattice RLS can be found in Lee (1980), Lee *et al.* (1981) and Friedlander (1982). A comparison with eqns. (B 1)–(B 12) leads to one possible implementation of the lattice RIV.

References

Benveniste, A., and Chaure, C., 1981, *I.E.E.E. Trans. autom. Control*, **26**, 1243.

Bierman, G. J., 1977, *Factorization Methods for Discrete Sequential Estimation* (New York : Academic Press).

Cadzow, J. A., and Moses, R. L., 1981, *Proc. ASSP Workshop on Spectral Estimation*, McMaster University, Hamilton, Ontario, Canada.

Friedlander, B., 1982, *Proc. Inst. elect. electron. Engrs*, **70**, 829.

Friedlander, B., Ljung, L., and Morf, M., 1981, *Proc. 20th I.E.E.E. Conf. Decision and Control*, pp. 1083–1084.

Goodwin, G. C., and Payne, R. L., 1977, *Dynamic System Identification : Experiment Design and Data Analysis* (New York : Academic Press).

Lawson, C. L., and Hanson, R. J., 1974, *Solving Least Squares Problems* (Englewood Cliffs, N.J. : Prentice-Hall).

Lee, D., 1980, Ph.D. dissertation, Department of Electrical Engineering, Stanford University, Stanford, California.

Lee, D., Morf, M., and Friedlander, B., 1981, *I.E.E.E. Trans. Acoustics Speech Sig. Process.*, **29**, 627.

Lee, D., Friedlander, B., and Morf, M., 1980, *Proc. 19th I.E.E.E. Conf. Decision and Control*, pp. 1225–1231.

Ljung, L., 1977, *I.E.E.E. Trans. autom. Control*, **22**, 539 ; 1981, *Automatica*, **17**, 89.

Markel, J. D., and Gray, A. H., 1976, *Linear Prediction of Speech* (New York : Springer-Verlag).

Morf, M., Vieira, A., and Lee, D., 1977, *Proc. I.E.E.E. Conf. Decision and Control*, New Orleans, Louisiana, pp. 1074–1078.

Panuska, V., 1969, *Proc. I.E.E.E. Symp. Adaptive Processes, Conf. Decision and Control.*

Porat, B., Friedlander, B., and Morf, M., 1981, *Proc. Conf. Acoustics Speech Signal Processing*, Atlanta, Georgia. (Also in *I.E.E.E. Trans. autom. Control*, 1982, **27**, 813.)

Prevosto, M., Benveniste, A., and Bernoun, B., 1982, IRISA Report No. 130, Rennes, France.

Samson, C., and Reddy, V. U., 1982, *Proc. Int. Conf. Acoustics, Speech, Signal Processing*, Paris, France, pp. 1752–1755.

Samson, C., 1982, *Int. J. Control*, **35**, 909.

Söderström, T., Ljung, L., and Gustavsson, I., 1978, *Automatica*, **14**, 231.

Söderström, T., and Stoica, P., 1981, *Automatica*, **17**, 101.

Söderström, T., 1973, Report 7308, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden.

Solo, V., 1979, *I.E.E.E. Trans. autom. Control*, **24**, 958.

Strejc, V., 1980, *Automatica*, **16**, 535.

Wong, K. Y., and Polak, E., 1967, *I.E.E.E. Trans. autom. Control*, **12**, 707.

Young, P. C., 1970, *Automatica*, **6**, 271.