BOUNDING APPROACHES TO SYSTEM IDENTIFICATION

Edited by
MARIO MILANESE,
JOHN NORTON,
HÉLÈNE PIET-LAHANIER,
and
ÉRIC WALTER

Bounding Approaches to System Identification

Bounding Approaches to System Identification

Edited by

Mario Milanese

Politecnico di Torino
Torino, Italy

John Norton

University of Birmingham
Birmingham, England

Hélène Piet-Lahanier

Office National d'Études et de Recherches Aérospatiales
Châtillon, France

and

Éric Walter

CNRS – École Supérieure d'Électricité
Gif-sur-Yvette, France

Springer Science+Business Media, LLC

Library of Congress Cataloging-in-Publication Data

On file

Chapters 2, 4 – 7, 9, 10, 12 – 15, 17 – 22, 24 – 26, and 30 are published with the permission of Pergamon Press Ltd., Oxford, and

Chapter 3 with the permission of Marcel Dekker, New York. The material has been revised and updated for publication in this volume.

ISBN 978-1-4757-9547-9 ISBN 978-1-4757-9545-5 (eBook)
DOI 10.1007/978-1-4757-9545-5

Originally published by Plenum Press, New York in 1996
Softcover reprint of the hardcover 1st edition 1996

10 9 8 7 6 5 4 3 2 1

Contributors

L. V. R. Arruda • Universidade Estadual de Campinas/FEE/DCA—Cidade Universitaria "Zeferino Vaz," Campinas (SP), Brazil

G. Belforte • Dipartimento di Automatica e Informatica, Politecnico di Torino, 10129 Torino, Italy

T. J. J. van den Boom • Department of Electrical Engineering, Delft University of Technology, 2600 GA Delft, The Netherlands

V. Cerone • Dipartimento di Automatica e Informatica, Politecnico di Torino, 10129 Torino, Italy

A. A. H. Damen • Department of Electrical Engineering, Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands

J. R. Deller, Jr. • Department of Electrical Engineering, Michigan State University, East Lansing, MI 48824

G. Favier • Laboratoire I3S, CNRS URA-1376, Sophia Antipolis, 06560 Valbonne, France

T. F. Filippova • Institute of Mathematics and Mechanics of Russian Academy of Sciences, Ekaterinburg, Russia

G. Fiorio • Dipartimento di Automatica e Informatica, Politecnico di Torino, 10129 Torino, Italy

K. Forsman • ABB Corporate Research, Ideon, S-223 70 Lund, Sweden

P.-O. Gutman • Faculty of Agricultural Engineering, Technion–Israel Institute of Technology, Haifa 32000, Israel

E. Halbwachs • Heudiasyc, CNRS, Université de Technologie de Compiègne, 60206 Compiègne, France

L. Jaulin • Laboratoire des Signaux et Systèmes, CNRS École Supérieure d'Électricité, 91192 Gif-sur-Yvette Cedex, France

B. Z. Kacewicz • Institute of Applied Mathematics, University of Warsaw, 02-097 Warsaw, Poland

K. J. Keesman • Department of Agricultural Engineering and Physics, University of Wageningen, 6703 HD Wageningen, The Netherlands

V. M. Kuntsevich • V. M. Glushkov Institute of Cybernetics, Academy of Sciences of Ukraine, 252207 Kiev, Ukraine

A. B. Kurzhanski • Moscow State University, Moscow, Russia

L. Ljung • Department of Electrical Engineering, Linköping University, S-581 83 Linköping, Sweden

S. Malan • Dipartimento di Automatica e Informatica, Politecnico di Torino, 10129 Torino, Italy

S. M. Markov • Division of Mathematical Modelling in Biology, Institute of Biophysics, Bulgarian Academy of Sciences, BG-1113 Sofia, Bulgaria

D. Meizel • Heudiasyc, CNRS, Université de Technologie de Compiègne, 60206 Compiègne, France

Y. A. Merkuryev • Riga Technical University, LV-1658 Riga, Latvia

M. Milanese • Dipartimento di Automatica e Informatica, Politecnico di Torino, 10129 Torino, Italy.

J. P. Norton • School of Electronic and Electrical Engineering, University of Birmingham, Edgbaston, Birmingham B15 2TT, United Kingdom

H. Piet-Lahanier • Direction des Études de Synthèse/SM Office National d'Études et de Recherches Aérospatiales F-92322, Châtillon Cedex, France

E. D. Popova • Division of Mathematical Modelling in Biology, Institute of Biophysics, Bulgarian Academy of Sciences, BG-1113 Sofia, Bulgaria

A. Preciado-Ruiz • ITESM–Campus Toluca, Toluca, Edo. de Mexico, Mexico

L. Pronzato • Laboratoire I3S, CNRS URA-1376, Sophia Antipolis, 06560 Valbonne, France

A. K. Rao • COMSAT Labs, Clarksburg, MD 20871

K. Sugimoto • Okayama University, Okayama, Japan

T. T. Tay • Department of Electrical Engineering, National University of Singapore, Singapore 0511

I. Vályi • National Bank of Hungary, Budapest, Hungary

S. M. Veres • School of Electronic and Electrical Engineering, University of Birmingham, Edgbaston, Birmingham B15 2TT, United Kingdom

A. Vicino • Facoltà di Ingegneria, Università degli Studi di Siena, 53100 Siena, Italy

É. Walter • Laboratoire des Signaux et Systèmes, CNRS-École Supérieure d'Électricité, 91192 Gif-sur-Yvette Cedex, France

G. Zappa • Dipartimento di Sistemi e Informatica, Università di Firenze, 50139 Firenze, Italy

L. S. Zhiteckij • V. M. Glushkov Institute of Cybernetics, Ukrainian Academy of Sciences, 252207 Kiev, Ukraine

Contents

CHAPTER 6. **Recursive Estimation Algorithms for Linear Models with Set Membership Error**

G. Belforte and T. T. Tay

CHAPTER 7. **Transfer Function Parameter Interval Estimation Using Recursive Least Squares in the Time and Frequency Domains**

P.-O. Gutman

CHAPTER 8. **Volume-Optimal Inner and Outer Ellipsoids**

L. Pronzato and É. Walter

CHAPTER 15. **Set-Valued Estimation of State and Parameter Vectors within Adaptive Control Systems**

V. M. Kuntsevich

CHAPTER 16. **Limited-Complexity Polyhedric Tracking**

H. Piet-Lahanier and É. Walter

CHAPTER 17. **Parameter-Bounding Algorithms for Linear Errors-in-Variables Models**

S. M. Veres and J. P. Norton

CHAPTER 21. **Robust Identification and Prediction for Nonlinear State-Space Models with Bounded Output Error**

K. J. Keesman

CHAPTER 22. **Estimation Theory for Nonlinear Models and Set Membership Uncertainty**

M. Milanese and A. Vicino

CHAPTER 23. **Guaranteed Nonlinear Set Estimation via Interval Analysis**

L. Jaulin and É. Walter

CHAPTER 24. **Adaptive Control of Systems Subjected to Bounded Disturbances**

L. S. Zhiteckij

CHAPTER 25. **Predictive Self-Tuning Control by Parameter Bounding and Worst-Case Design**

S. M. Veres and J. P. Norton

CHAPTER 26. **System Identification for H_∞-Robust Control Design**

T. J. J. van den Boom and A. A. H. Damen

CHAPTER 30. **Robust Performances Control Design for a High Accuracy Calibration Device**

M. Milanese, G. Fiorio, and S. Malan

1

Overview of the Volume

J. P. Norton

The genesis of this volume was the feeling of its editors that bounding had become an important enough topic, and was attracting enough attention, to require a collection of papers as a broad introduction to the field and a review of current progress. The basic idea of describing plant uncertainty by bounds is as old as toleranced engineering design. State bounding was introduced to the control engineering community in the late 1960s and parameter bounding in the early 1980s, but the subject became prominent only in the late 1980s and early 1990s, through workshops in Turin in 1988,(1) Santa Barbara(2) and Sopron(3) in 1992, papers and special sessions at conferences such as the 1988 International Association for Mathematics and Computers in Simulation (IMACS) World Congress in Paris, the International Federation of Automatic Control (IFAC) Budapest and Copenhagen identification symposia in 1991 and 1994, the 1991 Institute of Electrical and Electronics Engineers, Inc. Conference on Decision and Control (IEEE CDC) and 1993 IEEE International Symposium on Circuits and Systems (IEEE ISCAS), and increasing exposure in leading control engineering and signal processing journals.(4,5) The topic is now widespread over a large literature, so this volume is timely.

Before looking at the contents of the volume, let's see what bounding consists of and why it is of interest.

First, what is "bounding"? It is the process of finding bounds on the parameters or state of a given system model that confine within specified ranges the errors between the model inputs and outputs and their observed values. In other words, it

J.P. NORTON • School of Electronic and Electrical Engineering, University of Birmingham, Edgbaston, Birmingham B15 2TT, United Kingdom.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

answers the question "What parameter or state values in this model match the input-output observations to within a given error?" A more mechanistic view of bounding is that it maps the error bounds, through the model and observations, into parameter or state bounds. The error bounds define a set of acceptable error values, and the parameter or state bounds define a *feasible* set (sometimes tautologically called the "membership set"), so the mapping is from one set to another.

Although state and parameter bounding have much in common as in more traditional estimation approaches, this volume concentrates on the bounding of model parameters. Most commonly, symmetrical bounds are specified on instantaneous discrete-time values of individual error variables, *i.e.*, the l_∞ norm is specified for the vector of successive values of each error variable. In some cases, a bound on power or energy, or some other norm of the error, may be more appropriate. Often only a scalar model-output error is considered.

What use is bounding? It provides a way to relate uncertainty in the model to uncertainty in its inputs and outputs, independent of any probabilistic assumptions and referring only to a given data set. It splits parameter or state values into those not excluded by the data and error specification, *i.e.,* the values which must be taken into account when applying the model, and those which *are* excluded and need not be considered. Application of the model's feasible set, for instance in robust control design, gives results which can be relied on only as far as the error specification and data set can. Conversely, the extremes of model behavior over the feasible set give a "worst case" for design that may be conservative if the data set, error specification and model structure allow too wide a range of behavior. The need to impose realistic restrictions is why *parametric* models are considered in most of the work described here, and why algorithms computing approximate bounds seek the tightest possible.

The appeal of bounding lies in its directness, simplicity and need for few assumptions compared with its probabilistic alternatives; its ability to make use of prior knowledge expressed as bounds; and its status as the natural basis for worst case design. The choice of what to bound and what norms to employ gives flexibility not yet fully exploited. An important question in some applications is how to derive point estimates from the feasible set. For example, the values minimizing the maximum model-output error, or the maximum error in each individual parameter, are related to feasible sets in a simple way.

These considerations lead to the first contribution in this volume, a review of optimal estimation theory as a framework for bounding. The theory is able to accommodate a variety of norms and a wide range of problems, including the derivation of point estimates.

Chapters 3 to 16, like a large majority of the publications in the field, consider models linear in their parameters. The feasible set for such a model with instantaneous bounds on its additive output error is a polytope. It is sometimes computable exactly but is often approximated by an ellipsoidal, orthotopic (box) or parallelotopic set, to economize on computing and to simplify its subsequent use. Chapter 4 is a review of ellipsoidal bounding techniques, followed by several chapters on ellipsoidal bounding, which refer also to its connections with least-squares estimation and estimators applying dead zones to the prediction error in the update. Chapter 7 considers the important link between time and frequency domains, reflecting the initial emphasis of robust control analysis and design on frequency-domain bounds through the H_∞ formulation. In Chapters 9 to 13, a number of other bounding techniques for linear models are described. The issue of robustness to outliers, crucial in bounding because of the absence of any averaging, is raised.

The essential difference between state and parameter bounding is the presence in the former of time evolution of the quantities being bounded. The distinction disappears if the parameters are treated as time varying. The evolution requires expansion of the parameter or state bounds to account for the unknown (but bounded) increments from one sampling instant to the next. Computationally, this extra feature is not negligible. It involves vector summing of the prior feasible set and the feasible set of the increments, at every update, rapidly increasing the complexity of the feasible set. Various heuristics have been suggested to lessen this problem; Chapters 14 to 16 discuss how ellipsoids and simplified polyhedra may be used to approximate the evolving set, and how joint bounding of state and parameters may be performed in an adaptive controller.

Nonlinear models are the subject of Chapters 17 to 23. The "errors in variables" regression-type model, linear both in its parameters and in its input and output variables but with uncertainty in all input and output observations, is nonlinear by virtue of containing products of uncertain quantities. If the observation uncertainties are bounded, the parameter bounds due to the observations at any one instant are linear in any orthant. This results from the model's bilinearity: fixing the signs of all parameters determines which bound on each observation error maximizes or minimizes the contribution of that term in the model. The feasible set is therefore composed of polytopes in each orthant, in the absence of any serial dependence between observation errors. However, if the same observation appears in the model at more than one sampling instant (a "dynamic" errors-in-variables model), serial dependence *is* present and renders the parameter bounds nonlinear, even in one orthant. Both dynamic and static errors-in-variables models are considered in Chapters 17 to 19.

Chapters 20 to 23 offer bounding techniques for more general nonlinear models. Not surprisingly, the central issue is the compromise between computing load and resolution. The availability of performance guarantees also plays a large part in selecting an algorithm, as does the ability to handle bounds which are not well behaved locally.

Chapters 24 to 26 present various facets of bounding in robust/adaptive control. An assumption of bounded disturbances is often realistic, and is considered in the context of adaptive control. As previously noted, a model with parameter bounds allows worst-case control design. Worst-case-optimal control synthesis has the potential to guarantee performance. It takes model uncertainty explicitly into account, in contrast to ignoring it by applying certainty equivalence as is usual in adaptive control. Another feature available in bound-based adaptive control is compromise between short-term control performance and longer-term improvement. This is due to reduction of model uncertainty, by optimization of the control input for identification over the set of values giving adequate short-term control. Both these aspects are examined in the setting of predictive control. Identification for H_∞-robust control, a topic stimulating the recent convergence of bound-based identification and control design techniques, is also discussed.

Applications of parameter bounding have had relatively little exposure in the literature. In Chapters 27 to 30, applications in areas as diverse as robotics, image compression, speech processing and high-accuracy calibration are described. The speech processing case is particularly noteworthy as an example of the spread of parametric bounding into signal processing, which has paralleled its growth in control engineering over the past decade.

The technology of parameter bounding is beginning to mature to the point where it should find an established place in the armory of a system modeller or control designer. Important questions remain partly or completely unresolved, though, such as how best to combine distributional information and bounds, or probabilistic and bounding information; how to exploit both time-domain and frequency-domain bounds; how to describe and utilize the complicated bounds typical of nonlinear systems; and how to maximize the information about bounds derived from limited experimentation. Our hope is that readers of this volume will provide some of the answers.

REFERENCES

1. M. Milanese, R. Tempo, and A. Vicino, editors, *Robustness in Identification and Control*, Plenum, London (1989).
2. R. S. Smith and M. Dahleh, editors, *The Modeling of Uncertainty in Control Systems, Lecture Notes in Control & Information Sciences 192*, Springer-Verlag, Berlin (1994).
3. A. B. Kurzhanski and V. M. Veliov, editors, *Modeling Techniques for Uncertain Systems, Vol. 18 Progress in Systems & Control Theory*, Birkhäuser/IIASA, Boston (1994).
4. E. Walter, editor, *Math. Comput. Simul.* **32**, 5&6 (1990).
5. J. P. Norton, editor, *Int. J. Adapt. Control Sig. Proc.* **8**, 1 (1994); **9**, 1 (1995).

2

Optimal Estimation Theory for Dynamic Systems with Set Membership Uncertainty: An Overview

M. Milanese and A. Vicino

ABSTRACT

In many problems, such as linear and nonlinear regressions, parameter and state estimation of dynamic systems, state space and time series prediction, interpolation, smoothing, and functions approximation, one has to evaluate some unknown variable using available data. The data are always associated with some uncertainty and it is necessary to evaluate how this uncertainty affects the estimated variables. Typically, the problem is approached assuming a probabilistic description of uncertainty and applying statistical estimation theory. An interesting alternative, referred to as set membership or unknown but bounded (UBB) error description, has been investigated since the late 60s. In this approach, uncertainty is described by an additive noise which is known only to have given integral (typically l_1 or l_2) or componentwise (l_∞) bounds. In this chapter the main results of this theory are

M. Milanese • Dipartimento di Automatica e Informatica, Politecnico di Torino, 10129 Torino, Italy.
A. Vicino • Facoltà di Ingegneria, Università degli Studi di Siena, 53100 Siena, Italy.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

reviewed, with special attention to the most recent advances obtained in the case of componentwise bounds.

2.1. INTRODUCTION

Estimation theory is concerned with the problem of evaluating some unknown variables depending on given data (often obtained by measurements on a real process). Available data are always known with some uncertainty and it is necessary to evaluate how this uncertainty affects the estimated variables.

Obviously the solution of the problem depends on the type of assumptions made about uncertainty. The cases most investigated so far are unquestionably related to the assumption that uncertainty is given by an additive random noise with a (partially) known probability density function (pdf).

However, in many situations the very random nature of uncertainty may be questionable. For example, the real process generating the actual data may be very complex (large scale, nonlinear, and time varying) so that only simplified models can be practically used in the estimation process. The residuals of the estimated model have a component due to deterministic structural errors. Treating them as purely random variables may lead to unsatisfactory results.

An interesting alternative approach, set membership or unknown but bounded UBB error description has been pioneered by the work of Witsenhausen and Schweppe in the late 60s.(1,2,3) In this approach, uncertainty is described by means of an additive noise which is known only to have given bounds. The motivation for this approach is that in many practical cases the UBB error description is more realistic and less demanding than the statistical description. However, despite the appeal of its features, the UBB approach is not widely used yet. Until the early 80s, reasonable results and algorithms had been obtained only for uncertainty bounds of integral type (mainly l_2), while in practical applications componentwise bounds (l_∞) are mainly of interest.

Real advances have been obtained in the last few years for the componentwise bounds case, leading to theoretical results and algorithms which can be properly applied to practical problems where the use of statistical techniques is questionable.

The purpose of this chapter is to review these results and to present them in a unified framework, in order to contribute the present state of the art in the field and simulate further basic and applied researches.

2.2. PROBLEM FORMULATION

In this section a general framework is formulated such that the main results in the literature can be presented in a unifying view. Such formulation can be sketched as follows.(4,5)

We have a problem element λ (for example a dynamic system or a time function) and a function $S(\lambda)$ of this problem element (for example some parameter of the dynamic system or particular value of the time function) is to be evaluated. Suppose λ is not known exactly, but there is some information on it. In particular assume that it is an element of a set K of possible problem elements and that some function $F(\lambda)$ is measured. Moreover, suppose that exact measurements are not available and actual measurements y are corrupted by some error ρ.

The estimation problem is to find an estimator ϕ providing an approximation $\phi(y) \approx S(\lambda)$ using the available data y and evaluating some measure of such approximation. A geometric sketch is shown in Fig 2.1.

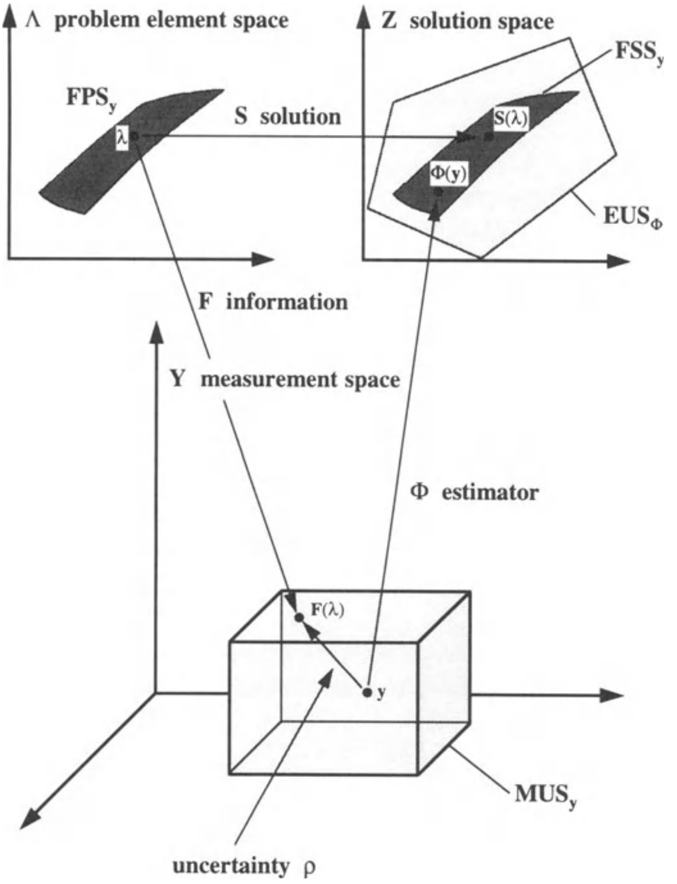


FIGURE 2.1. Generalized estimation problem.

2.2.1. Spaces and Operators

Let Λ be a linear normed n-dimensional space over the real field. Consider a given operator S, called a solution operator, which maps Λ into Z

$$S : \Lambda \rightarrow Z \tag{2.1}$$

where Z is a linear normed l-dimensional space over the real field. The aim is to estimate an element $S(\lambda)$ of the space Z, knowing approximate information about the element λ. Suppose that two kinds of information may be available. The first one (often referred to as *a priori* information) is expressed by assuming that $\lambda \in K$, where K is a subset of Λ. In most cases K is given as

$$K = \{\lambda \in \Lambda; \ ||R(\lambda - \lambda_0)|| \leq 1\} \tag{2.2}$$

where R is a linear operator and λ_0 is a known problem element. The second kind of information is usually provided by the knowledge of some function $F(\lambda)$, where F, called an information operator, maps Λ into a linear normed m-dimensional space Y

$$F : \Lambda \rightarrow Y. \tag{2.3}$$

Spaces Λ, Z, Y are called problem element, solution and measurement spaces respectively. In the following, unless otherwise specified, assume that Λ and Z are equipped with l_∞ norms and Y is equipped with an l_∞^w norm.[1]

In general, due to the presence of noise, exact information $F(\lambda)$ about λ is not available and only perturbed information y is given. In this context, uncertainty is assumed to be additive, i.e.,

$$y = F(\lambda) + \rho \tag{2.4}$$

where the error term ρ is unknown, but bounded by some given positive number ε

$$||\rho|| \leq \varepsilon \tag{2.5}$$

Note that if an l_∞^w norm in measurement space Y is used, componentwise bounds with different values on every measurement can be treated.

An algorithm ϕ is an operator (in general nonlinear) from Y into Z:

$$\phi : Y \rightarrow Z \tag{2.6}$$

i.e., it provides an approximation $\phi(y)$ of $S(\lambda)$ using the available data y. Such an algorithm is also called an estimator.

[1] The l_∞^w norm is defined as $||y||_\infty^w = \max_i \{w_i|y_i|, \quad w_i > 0$

Some examples are now presented in order to show how specific estimation problems fit into this general framework.

2.2.2. Example 1: Parameter Estimation of ARX Models

Consider the ARX model

$$y_k = \sum_{i=1}^{p} v_i y_{k-i} + \sum_{i=1}^{q} \theta_i u_{k-i} + \rho_k \tag{2.7}$$

where y_k is a scalar output, u_k is a known scalar input and ρ_k is an unknown but bounded error such that

$$|\rho_k| \le \varepsilon_k, \quad \forall k \tag{2.8}$$

Suppose that m values $[y_1,\ldots,y_m]$ are known and the aim is to estimate parameters $[\nu_i,\theta_i]$. For the sake of simplicity suppose that $p \ge q$. Λ can be defined as the $(p + q)$-dimensional space with elements

$$\lambda = [\nu_1, \ldots, \nu_p, \theta_1, \ldots, \theta_q]^T. \tag{2.9}$$

If no *a priori* knowledge on parameter λ is available, then $K = \Lambda$.

Z is the $(p + q)$-dimensional space with elements $z = \lambda$, so that $S(\lambda)$ is identity. Y is the $(m - p)$-dimensional space with elements $y = [y_{p+1}, \ldots, y_m]^T$, and consequently $F(\lambda)$ is linear and is given by

$$F(\lambda) = \begin{bmatrix} y_p & \cdots & y_1 & u_p & \cdots & u_{p+1-q} \\ \cdot & \cdots & \cdot & \cdot & \cdots & \cdot \\ y_{m-1} & \cdots & y_{m-p} & u_{m-1} & \cdots & u_{m-q} \end{bmatrix} \lambda. \tag{2.10}$$

2.2.3. Example 2: State Estimation of Linear Dynamic Systems

Consider the problem of estimating the state of the following discrete, linear, time invariant dynamic system

$$\begin{cases} x_{k+1} = Ax_k + Bu_k \\ y_k \quad = Cx_k + \rho_k \end{cases} \tag{2.11}$$

where x_k, y_k, u_k and ρ_k are the state, observation, process noise and observation noise vectors respectively; A, B and C are given matrices. For the sake of simplicity, suppose that x is l-dimensional and y, u, and ρ are scalar variables.

Assume that the samples of process and observation noise are unknown but bounded

$$|u_k| \leq U_k, \quad \forall k \tag{2.12}$$

$$|\rho_k| \leq \varepsilon_k, \quad \forall k. \tag{2.13}$$

Suppose that m values $[y_1, ..., y_m]$ are known and the aim is to estimate x_m. Λ can be defined as the $l + m - 1$-dimensional space with elements

$$\lambda = [x_1^T, u_1, \ldots, u_{m-1}]^T. \tag{2.14}$$

If no *a priori* information on the initial state x_1 is available, K is defined by

$$K = \{\lambda \in \Lambda; |u_j| \leq U_j, j = 1, \ldots, m-1\} \tag{2.15}$$

Z is the l-dimensional space with elements $z = x_m$. Y is the m-dimensional space with elements $y = [y_1, ..., y_m]^T$. Standard computation of solutions of the set of difference Eq. (2.11) shows that the solution and information operator are linear and are given by

$$S(\lambda) = [A^{m-1}, A^{m-2}B, \ldots, AB, B]\lambda \tag{2.16}$$

$$F(\lambda) = \begin{bmatrix} C & 0 & \cdots & 0 & 0 \\ CA & CB & \cdots & 0 & 0 \\ CA^2 & CAB & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ CA^{m-2} & CA^{m-3}B & \cdots & CB & 0 \\ CA^{m-1} & CA^{m-2}B & \cdots & CAB & CB \end{bmatrix} \lambda. \tag{2.17}$$

2.2.4. Example 3: Parameter Estimation of Multiexponential Models

Consider the multiexponential model

$$y(t) = \sum_{i=1}^{l} \mu_i e^{-\nu_i t} + \rho(t) \tag{2.18}$$

where μ_i and ν_i are unknown real parameters and $\rho(t)$ is unknown but bounded by a given $\varepsilon(t)$

$$|\rho(t)| \leq \varepsilon(t). \tag{2.19}$$

Suppose that m values $[y(t_i), \cdots, y(t_m)]$ are known and the aim is to estimate parameters μ_i and ν_i, $i = 1, \ldots, l$.

By setting $\xi_i = e^{-\nu_i}$, $i = 1, \dots, l$, the space Λ is taken as the $2l$-dimensional space with elements

$$\lambda = [\mu_1, \dots, \mu_l, \xi_1, \dots, \xi_l]^T. \tag{2.20}$$

$S(\lambda)$ can be taken as the identity operator. In this way, estimation of variables ξ_i is considered instead of ν_i. Original variables can be obtained by logarithmic transformation.

Y is defined as the m-dimensional space with elements $y = [y(t_1), \dots, y(t_m)]^T$. Then, information operator $F(\lambda)$ becomes the polynomial function

$$\begin{bmatrix} F_1(\lambda) \\ \cdots \\ F_m(\lambda) \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{l} \mu_i \xi_i^{t_1} \\ \cdots \\ \sum_{i=1}^{l} \mu_i \xi_i^{t_m} \end{bmatrix}. \tag{2.21}$$

2.2.5. Example 4: Multistep Prediction with ARX Models

Consider the ARX model Eq. (2.7) and suppose that the aim is to estimate y_{m+h} when past values $[y_1, \dots, y_m]$ are measured (h-step ahead prediction problem). For the sake of simplicity, consider the case $h = 2$.

The space Λ can be defined as the $p + q + 2$-dimensional space with elements

$$\lambda = [\nu_1, \dots, \nu_p, \theta_1, \dots, \theta_q, \rho_{m+1}, \rho_{m+2}]^T. \tag{2.22}$$

If no *a priori* knowledge on parameter λ is available, K is given by

$$K = \{\lambda \in \Lambda;\, |\rho_{m+1}| \le \varepsilon_{m+1}, |\rho_{m+2}| \le \varepsilon_{m+2}\}. \tag{2.23}$$

Z is the 1-dimensional space with elements $z = y_{m+2}$ and consequently $S(\lambda)$ is the polynomial function given by

$$\begin{aligned} S(\lambda) = {} & (\nu_1\nu_1 + \nu_2)y_m + (\nu_1\nu_2 + \nu_3)y_{m-1} + \cdots + \nu_p y_{m-p+1} \\ & + \theta_1 u_{m+1} + (\nu_1\theta_1 + \theta_2)u_m + \cdots + \theta_q u_{m-q+1} \\ & + \nu_1\rho_{m+1} + \rho_{m+2}. \end{aligned} \tag{2.24}$$

Y is an $(m - p)$ dimensional space with elements $y = [y_{p+1}, \dots, y_m]^T$ and $F(\lambda)$ is linear and given by

$$F(\lambda) = \begin{bmatrix} y_p & \cdots & y_1 & u_p & \cdots & u_{p+1-q} & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ y_{m-1} & \cdots & y_{m-p} & u_{m-1} & \cdots & u_{m-q} & 0 & 0 \end{bmatrix} \lambda. \tag{2.25}$$

2.3. MAIN DEFINITIONS AND CONCEPTS

This section provides definitions of the main sets involved in the theory, optimality concepts used to evaluate estimator's performances, and types of estimators investigated.

2.3.1. Relevant Sets

The following sets play key roles in set membership estimation theory: measurement uncertainty set:

$$MUS_y = \{\tilde{y} \in Y : \|\tilde{y} - y\|_\infty^w \le \varepsilon\} \tag{2.26}$$

estimate uncertainty set (for a given estimator ϕ);

$$EUS_\phi = \phi(MUS_y) \tag{2.27}$$

feasible problem elements set;

$$FPS_y = \{\lambda \in K : \|y = F(\lambda)\|_\infty^w \le \varepsilon\} \tag{2.28}$$

and feasible solutions set

$$FSS_y = S(FPS_y). \tag{2.29}$$

Note the difference between EUS_ϕ and FSS_y. The former depends on the particular estimator ϕ used and gives all possible estimated values that could be obtained for all possible measurements consistent with the actual measurement y and the given error bounds. The latter depends only on the problem setting and gives all possible values which are consistent with the available information on the problem.

In the literature on parameter estimation, where problem element λ represents the parameters to be estimated and $S(\lambda)$ is identity (see Section 2.2.2), FPS_y coincides with FSS_y and has been given also different names such as feasible parameters set, membership-set estimate and likelihood set.

An exact description of FSS_y or EUS_ϕ is in general not simple, since they may be very complex sets (e.g. non-convex, not simply connected). For this reason, approximate descriptions are often looked for, using simply shaped sets like boxes or ellipsoids containing (outer bounding) or contained in (inner bounding) the set of interest (see Fig. 2.2). In particular minimum volume outer box (MOB) or ellipsoid (MOE) and maximum volume inner box (MIB) or ellipsoid (MIE) are of interest.

Information of great practical interest is also provided by the values uncertainty intervals (VUI) and estimate uncertainty intervals (EUI), giving the maximum ranges of possible variations of the feasible and the estimated values, respectively. The VUIs are defined as

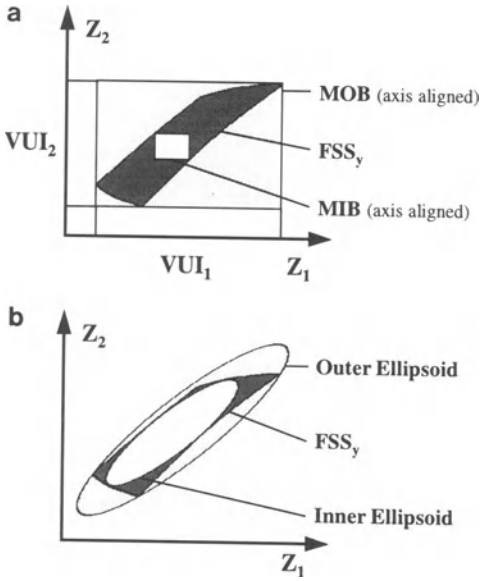


FIGURE 2.2. (a) Box and (b) ellipsoid inner- and outer-bounding.

$$VUI_i = [z_i^m, z_i^M] \quad i = 1, ..., l, \tag{2.30}$$

where

$$z_i^m = \inf_{z \in FSS_y} z_i = \inf_{\lambda \in FPS_y} S_i(\lambda) \quad i = 1, \ldots, l$$

and

$$z_i^M = \sup_{z \in FSS_y} z_i = \sup_{\lambda \in FPS_y} S_i(\lambda) \quad i = 1, \ldots, l. \tag{2.31}$$

Note that the VUIs are the sizes (along coordinate axis) of the axis aligned box of minimal volume containing FSS_y (see Fig. 2.2).

The EUIs are defined in the same way substituting EUS_ϕ for FSS_y.

2.3.2. Errors and Optimality Concepts

Algorithm performance is measured according to the following errors:
Y-local error $E_y^\varepsilon(\phi)$, where

$$E_y^\varepsilon(\phi) = \sup_{\lambda \in FPS_y} \|S(\lambda) - \phi(y)\| \tag{2.32}$$

Λ-local error $E_\lambda^\varepsilon(\phi)$, where

$$E_\lambda^\varepsilon(\phi) = \sup_{y \in MUS_{F(\lambda)}} \|S(\lambda) - \phi(y)\| \tag{2.33}$$

and global error $E^{\varepsilon}(\phi)$

$$E^{\varepsilon}(\phi) = \sup_{y \in Y} E^{\varepsilon}_{y}(\phi) = \sup_{\lambda \in \Lambda} E^{\varepsilon}_{\lambda}(\phi). \tag{2.34}$$

Dependence on ε is dropped out in subsequent notation, except when necessary.

Algorithms minimizing these types of errors are indicated respectively as Y-locally, Λ-locally and globally optimal.

Notice that Y-local optimality is of particular interest in system identification problems, where a set of measurements y is available and one wants to determine the best achievable estimate $S(\lambda)$ for each possible y using an algorithm $\phi(y)$. Also Λ-local optimality is a particularly meaningful property in estimation problems, since it ensures the minimum uncertainty of the estimates for the worst measurement y, for any possible element $\lambda \in K$.

Y- and Λ-local optimality are stronger properties than global optimality, as can be seen from Eq. (2.34). For example, a Y-locally optimal algorithm minimizes the local error $E_y(\phi)$ for all data y, while a globally optimal algorithm minimizes the global error $E(\phi)$ only for the worst data. In other words, a Y-locally optimal algorithm is also globally optimal, while the converse is not necessarily true.

2.3.3. Classes of Estimators

Some classes of estimators whose properties have been investigated in the literature are now introduced.

The first class is related to the idea of taking the Chebicheff center of FSS_y as estimate of $S(\lambda)$. The center of FSS_y, $c(FSS_y)$, and the corresponding radius, $rad(FSS_y)$, are defined by

$$\sup_{z \in FSS_y} \|c(FSS_y) - z\| = \inf_{\tilde{z} \in Z} \sup_{z \in FSS_y} \|\tilde{z} - z\| = rad(FSS_y). \tag{2.35}$$

A central estimator ϕ^c is defined as

$$\phi^c(y) = c(FSS_y) \tag{2.36}$$

The second class includes estimators analogous to unbiased estimators in statistical theory, which give exact values if applied to exact data.

An estimator ϕ is correct if

$$\phi(F(\lambda)) = S(\lambda) \quad \forall \lambda \in \Lambda. \tag{2.37}$$

Such a class is meaningful only for $l \leq m$, that is, when there are at least as many measurements as variables to be estimated (the typical situation in estimation practice). This class contains most of the commonly used estimators, such as projection estimators.

A projection estimator ϕ^P is defined as

$$\phi^P(y) = S(\lambda_y) \tag{2.38}$$

where $\lambda_y \in \Lambda$ is such that

$$\|y - F(\lambda_y)\| = \inf_{\lambda \in \Lambda} \|y - F(\lambda)\|. \tag{2.39}$$

The most widely investigated and used estimators in this class are least square estimators (ϕ^{LS}), which are projection estimators when an l_2 norm is used in space Y. Least-absolute values and least-maximum value estimators have been also considered in the literature, which are projection estimators when l_1 and l_∞ norms are respectively used in space Y.

In the next sections the results available in the literature regarding the following aspects are reviewed: existence and characterization of estimators, optimal with respect to some of the optimality concepts introduced previously; actual computation of the derived optimal estimators; evaluation of the errors of optimal and of projection estimators; and exact or approximate description of feasible sets FPS_y, FSS_y and estimate uncertainty set EUS_ϕ. Whenever possible, a statistical counterpart of the presented results is indicated, based on the analogy:

Y-local optimality ⇔ minimum variance optimality

FSS_y ⇔ minimum variance estimate pdf

EUS_ϕ ⇔ estimate pdf

EUI's ⇔ estimate confidence intervals

VUIs ⇔ Cramer-Rao lower bound confidence intervals.

2.4. NONLINEAR PROBLEMS

A first important result is related to the existence of a Y-locally optimal estimator. No general results are available for Λ-locally optimal estimators. This result also shows that the minimum Y-local error is given by the radius of FSS_y.

Result 1. [4,6] A central estimator ϕ^c is Y-locally optimal

$$E_y(\phi^c) \le E_y(\phi) \quad \forall y \in Y, \forall \phi \tag{2.40}$$

Its Y-local error is

$$E_y(\phi^c) = rad(FSS_y) \tag{2.41}$$

This result holds for any norm in Λ, Z, Y. □

It can be considered as the counterpart of the conditional mean theorem in statistics. As with conditional mean estimators, central estimators are in general difficult to compute. The computation of ϕ^c involves the knowledge of FSS_y, which may be a very complex set (nonconvex, not simply connected).

Several approaches have been proposed to describe FSS_y, mainly in papers related to dynamic system parameter estimation. In Ref. 7 a random sample of parameters is generated by a Monte Carlo technique, and Eqs. (2.4 and 2.5) are used to check if they belong to FSS_y. Global optimization methods based on random search are used in Refs. 8 and 9 to construct the boundary of FSS_y. In Ref. 8 projections of FSS_y onto coordinate one-dimensional or two-dimensional subspaces are looked for. In Ref. 9 intersections of the boundary of FSS_y with bundles of straight lines centered at points inside FSS_y are searched. The optimization methods used in these papers converge in probability to the global maximum or minimum of interest. However, this convergence property is not very useful in practice, because no estimate is given of the distance of the achieved solution from the global solution. Moreover, all these approaches suffer the curse of dimensionality. These reasons motivate the interest in looking for less detailed but more easily computable information on FSS_y.

An important result in this direction is that the computation of ϕ^c and of its Y-local error do not require the exact knowledge of FSS_y, but only of the VUIs.

Result 2.[(5)] The center $c(FSS_y)$ can be computed as

$$c_i(FSS_y) = (z_i^M + z_i^m)/2 \quad i = 1, \dots, l \tag{2.42}$$

The radius $rad(FSS_y)$ can be computed as

$$rad(FSS_y) = \max_i (z_i^M - z_i^m)/2 \tag{2.43}$$

where z_i^m and z_i^M are given by Eq. (2.31). □

Result 2 states that the computation of a central algorithm and of minimum Y-local error is equivalent to the computation of the VUIs, requiring the solution of only $2l$ optimization problems of the type Eq. (2.31).

Equation (2.31) problems are in general not convex, exhibiting local extrema. Any of the general global optimization algorithms available in the literature give approximate solutions converging to the exact ones only in probability and, more seriously, they do *not* provide any assessment on how far the approximate solution is from the correct one.

If $S(\lambda)$ and $F(\lambda)$ are polynomial functions, specific global algorithms exist, for obtaining better results.

Result 3.[(10)] If $S(\lambda)$ and $F(\lambda)$ are polynomial, algorithms exist converging with certainty to global extrema of Eq. (2.31).

Under the assumptions of Result 3, Eq. (2.31) are polynomial optimization problems, in the sense that both cost functions and constraints are polynomials in λ. Polynomial problems are in general nonconvex and may admit local extrema.[11] Nevertheless, if all the variables are strictly positive (in which case the term signomial problems is used), an algorithm is available to find a global maximum.[10,12,13] The underlying idea of this algorithm is to construct a sequence of convex problems approximating the original problem iteratively better. In this way, the algorithm generates a sequence of lower and upper bounds of the global extremum, converging monotonically to it. If the sign of some of the variables is not known, it is possible to reduce a polynomial problem to a signomial problem by setting these variables as the difference of strictly positive new variables.

The hypothesis of Result 2 covers large classes of problems, as shown in examples (2.2.2–2.2.5). The implication is that an optimal estimator and its error can be exactly computed for several nonlinear problems of practical interest. No analogous result is available in the statistical context.

Most of the papers in the literature focus on studying FSS_y, while very few results are available on EUS_ϕ. For any correct estimator, FSS_y is an inner bounding set of EUS_ϕ.[14]

Result 4.[14] If ϕ is correct then

$$FSS_y \subseteq EUS_\phi \quad \forall y \in Y \tag{2.44}$$

□

Hence, for correct estimators the VUIs are lower bounds of the EUIs, that is,

$$VUI_i \subseteq EUI_i \quad i = 1, \ldots, l \tag{2.45}$$

Consider the properties of projection estimators. In general they are not optimal with respect to any of the three considered type of errors.[15] However they are almost Y-locally optimal (within a factor 2) as shown by the following result.

Result 5.[15] A projection algorithm ϕ^p is such that

$$E_y(\phi^p) \le 2\, rad(FSS_y) \le 2E_y(\phi) \quad \forall\, y \in Y, \ \forall\, \phi \tag{2.46}$$

□

Projection estimators enjoy interesting properties of robustness with respect to inexact knowledge of the uncertainty bound ε. Central estimators are not robust in such a sense: a central algorithm computed supposing that $\varepsilon = \varepsilon_0$ may not be optimal if the actual ε is different. A central estimate $\phi^c(y)$ may not even belong to the actual FSS_y and its Y-local error $E_y(\phi^c)$ may be greater than 2 $rad(FSS_y)$.

On the contrary, projection estimators are robustly almost Y-locally optimal, independent of the volume of ε, as shown by the next result.

Result 6.[16] Let ϕ^p be the projection estimator. Then

$$E_y^\varepsilon(\phi^p) \le 2\, rad(FSS_y) \le 2E_y^\varepsilon(\phi) \quad \forall y \in Y, \forall \phi, \forall \varepsilon \tag{2.47}$$

□

Projection estimators also have nice properties in a statistical context. For example, an l_2-projection estimator is the maximum likelihood estimator (MLE) if noise ρ is supposed to be gaussian; an l_1-projection estimator is the MLE if noise is supposed to have a Laplace pdf; an l_∞-projection estimator is the MLE if noise is uniformly distributed. Projection estimators l_1 and l_∞ also have interesting robustness properties with respect to uncertainty in the pdf's knowledge.(17,18,19)

2.5. LINEAR PROBLEMS

Consider the case in which $S(\lambda)$ and $F(\lambda)$ are linear. In this case, Eq. (2.4) is written as

$$y = \mathcal{A}\lambda + \rho \tag{2.48}$$

where $\mathcal{A}$ is a matrix of dimension (m, n).

These assumptions are restrictive, but include cases of practical interest such as parameter estimation of linear regressions, parameter estimation of ARMA models with polynomial trends and harmonic components, state estimation of dynamic systems, and time series forecasting. Moreover, if uncertainty bounds are not too large, linear theory can be used for a first approximate analysis using some linearization techniques.

From Result 1 a central estimator is Y-locally and globally optimal. In the linear case it is also correct and Λ-locally optimal in the class of correct estimators, as shown in the next result.

Result 7.(20) ϕ^c is Y-locally optimal:

$$E_y(\phi^c) \leq E_y(\phi) \quad \forall\ \phi \tag{2.49}$$

ϕ^c is a Λ-locally optimal (among correct estimators)

$$E_\lambda(\phi^c) \leq E_\lambda(\phi) \quad \forall\ \lambda \in K, \forall\ \phi \text{ correct} \tag{2.50}$$

□

In Ref. (15) it is proven that Result 7 holds for any norm in Y.

Under the present assumptions, FSS_y and FPS_y are polytopes. Then from Result 2 it follows that ϕ^c and its Y-local error $E_y(\phi^c)$ can be obtained by solving the $2l$ linear programming problems of Eq. (2.31).

A linear estimator can be computed, which is correct, globally optimal, and Λ-locally optimal within the class of correct estimators. This gives a complete solution to the linear case, representing the counterpart of the Gauss-Markov theory in statistical estimation.

Result 8.(5,14) Let $K = \Lambda$ and $m \geq n$. Then there exists a linear estimator H^* that is correct and globally optimal

$$E(H^*) \leq E(\phi) \ \forall \ \phi \tag{2.51}$$

The linear estimator H^* is Λ-locally optimal (among correct estimators)

$$E_\lambda(H^*) \leq E_\lambda(\phi) \ \forall \ \lambda \in \Lambda, \forall \ \phi \text{ correct} \tag{2.52}$$

Its errors are

$$E(H^*) = E_\lambda(H^*) = E_{\mathcal{A}\lambda}(H^*) = rad(FSS_{\mathcal{A}\lambda}) \ \forall \ \lambda \in \Lambda \tag{2.53}$$

□

Estimator H^* can be computed from the knowledge of the active constraints of the linear programming problems of Eq. (2.31) with $y = 0$.[5,14]

In case that an l_2-norm is used in Y, H^* can be computed by least squares. Under this assumption, the least squares estimator is linear and correct, robustly Y-locally optimal and Λ-locally optimal within the class of correct estimators, as shown by the next result.

Result 9.[15] Let $K = \Lambda$, $m \geq n$ and Y be a Hilbert space. Let ϕ^{LS} be the projection (least square) estimator. Then:
ϕ^{LS} is central, linear, correct and robustly Y-locally optimal

$$E_y^\varepsilon(\phi^{LS}) \leq E_y^\varepsilon(\phi) \ \forall \ y \in Y, \forall \ \phi, \forall \ \varepsilon \tag{2.54}$$

ϕ^{LS} is Λ-locally optimal (among correct estimators)

$$E_\lambda(\phi^{LS}) \leq E_\lambda(\phi) \ \forall \ \lambda \in \Lambda, \forall \ \phi \text{ correct} \tag{2.55}$$

□

The sets FPS_y, FSS_y and EUS_ϕ, (for linear ϕ), are polytopes described by the sets of linear inequalities appearing in Eqs. (2.27–2.29). This is not the simplest way to describe them (for example, many linear inequalities may not concur to defining the boundary of the polytope) and simpler descriptions could be of interest. One way of characterizing a polytope P is through its vertices. Algorithms exist which allow one to compute recursively the vertices of a polytope P_k, defined by the first k measurements, from the knowledge of P_{k-1} and the k-th measurement.[21,22,23,24] The number of vertices may be relatively smaller than the theoretical maximum. For example, Monte Carlo simulations on ARMA models parameter estimation,[23] have shown that the mean number of vertices of FSS_y is approximately constant for $m > 50$. For $l = 4$ and $l = 5$, for example, they are approximately 50 and 150, respectively.

Polytopes can be represented alternatively by describing their faces. This representation is used to derive a recursive algorithm.[25] This approach seems more involved than the previous one, but it also allows the recursive computation of an outer bounding polytope with a fixed number of faces, leading to an approximating description of the polytope of interest by means of a polytope of prescribed complexity.

The most investigated approaches to approximate description of polytopes are through ellipsoids and boxes for the case of parameter estimation, where the polytope of interest is the feasible parameter set.

A recursive algorithm for outer bounding ellipsoid computation has been proposed in.[26] The underlying idea is as follows.

Let OE_{k-1} be the outer ellipsoid bounding P_{k-1}. Let R_k be the feasible parameter set corresponding only to the k-th measurement

$$R_k = \{\lambda \in \Lambda : y_k - \varepsilon_k \leq a_k^T\lambda \leq y_k + \varepsilon_k\} \tag{2.56}$$

where a_k^T is the k-th row of $\mathcal{A}$.

Clearly $P_k \subseteq OE_{k-1} \cap R_k$. OE_k is computed as the minimal volume ellipsoid containing $OE_{k-1} \cap R_k$, and then containing P_k also.

If an ellipsoid OE_k is defined by its centers λ_k^c and positive definite matrix Σ_k according to

$$OE_k = \{\lambda \in \Lambda : (\lambda - \lambda_k^c)^T \Sigma_k^{-1} (\lambda - \lambda_k^c) \leq 1\} \tag{2.57}$$

the following recursive algorithm has been obtained.

Result 10.[26] The ellipsoid OE_k can be computed by the recursion

$$\lambda_k^c = \lambda_{k-1}^c + \frac{\sigma_k V_k a_k \nu_k}{\varepsilon_k^2} \tag{2.58}$$

$$\Sigma_k = (1 + \sigma_k - \frac{\sigma_k \nu_k^2}{\varepsilon_k^2 + \sigma_k \mu_k})V_k \tag{2.59}$$

where

$$V_k = \Sigma_{k-1} - \frac{\sigma_k \Sigma_{k-1} a_k a_k^T \Sigma_{k-1}}{\varepsilon_k^2 + \sigma_k \mu_k} \tag{2.60}$$

$$\nu_k = y_k - a_k^T \lambda_{k-1} \tag{2.61}$$

$$\mu_k = a_k^T \Sigma_{k-1} a_k \tag{2.62}$$

and σ_k is the positive solution of the equation

$$(l - 1)\mu_k^2\sigma_k^2 + [(2l - 1)\varepsilon_k^2 - \mu_k + \nu_k^2]\mu_k\sigma_k + \varepsilon_k^2[l(\varepsilon_k^2 - \nu_k^2) - \mu_k] = 0 \tag{2.63}$$

if a positive solution exists, otherwise $\sigma_k = 0$. □

Computational complexity of this algorithm and slight modifications for implementation on a systolic architecture can be found.[27] A modification of this algorithm with data-dependent updating and forgetting factor has been proposed.(28)

A similar approach can be used for the recursive computation of inner bounding ellipsoids.(29,30) Let IE_{k-1} the inner bounding ellipsoid contained in P_{k-1}. Then IE_k is chosen as the maximal volume ellipsoid such that

$$IE_k \subseteq IE_{k-1} \cap R_k \subseteq P_k \tag{2.64}$$

The resulting recursive algorithm is much as for the outer bounding ellipsoid and is not reported here.

The main drawback of these recursive algorithms is that they do *not* give the minimal and maximal volume ellipsoids bounding the feasible parameter set.(29,31) This is true also for improved versions of the algorithm.(31,32) Since IE_k has an unfortunate tendency to shrink rapidly and vanish,(30) the inclusion $IE_k \subset P_k \subset OE_k$ in practice may not give any reasonable information on the looseness of bound OE_k.

A nonrecursive solution to the problem of finding the minimal volume outer ellipsoid contained in FPS_y (MOE_{FPS}) and the maximal volume inner ellipsoid contained in FPS_y (MIE_{FPS}), has been proposed.(33,34) The solution for MIE_{FPS} is given by the following result.

Result 11.(33) The MIE_{FPS} has center λ^{c*} and matrix Σ^* solution of

$$\max \quad \det(\Sigma) \tag{2.65}$$

subject to

$$\lambda^c, \Sigma: \begin{cases} (u_i^T\lambda^c + c_i)^2 - u_i^T\Sigma u_i \geq 0, \, i = 1, \ldots, 2m \\ u_i^T\lambda^c + c_i \geq 0, \, i = 1, \ldots, 2m \\ \Sigma_i > 0, \, i = 1, \ldots, n \end{cases}$$

where Σ_i, $i = 1, \ldots, n$ are the principal minors of Σ, and matrix $U \in \mathbf{R}^{(2m,n)}$ (with rows denoted by u_i^T) and vector $c \in \mathbf{R}^{2m}$ are given by

$$U = [A^T | - A^T]^T; \, c = [-y_-^T | y_+^T]^T \tag{2.66}$$

$$y_- = [y_1 - \varepsilon w_1, y_2 - \varepsilon w_2, \ldots, y_m - \varepsilon w_m]^T$$

$$y_+ = [y_1 + \varepsilon w_1, y_2 + \varepsilon w_2, \ldots, y_m + \varepsilon w_m]^T \tag{2.67}$$

□

Equation (2.65) is a polynomial optimization problem and can be solved by use of signomial programming.(10) The solution of Eq. (2.65) may be computationally cumbersome, even for a few parameters. Then less general but simpler solutions may be of interest. For example, the maximum ellipsoid of given shape may be sought. Consider that Σ is given except for a scale factor (for example the shape of the outer ellipsoid given by Result 10 can be used). In such a case, Eq. (2.65) reduces to a linear programming problem with $(n + 1)$ variables and $(2m + 1)$ constraints.

The solution for MOE_{FPS} also can be obtained by solving a suitable polynomial problem.(34) Unfortunately, the computational complexity is high for the general case, and does not reduce, as for MIE_{FPS}, if restricted classes of ellipsoids are considered.

For the computation of extremal volume inner and outer boxes definitions are as follows.

A box is defined as:

$$B(\lambda^c, l, R) = \{\lambda \in \Lambda : \|R(\lambda - \lambda^c)\|_\infty^l \leq 1\} \tag{2.68}$$

where R is an orthonormal matrix. The box is described by its center λ^c, axis lengths l_i and rotation R. If $R = I$ the box is aligned with coordinate axis.

A solution to the problem of finding the minimal volume outer box contained in FPS_y (MOB_{FPS}) is provided in Ref. 34 as solution of a suitable polynomial problem. Its computational complexity is high, unless the rotation of the box is given. In such a case the problem can be reduced to a linear programming problem. In particular, if $R = I$ the axis-aligned MOB_{FSS} can be computed directly from Eq. (2.31). This requires the solution of $2l$ linear programming problems with n variables and $2m$ inequalities constraints.

The solution to the problem of finding the maximal volume inner box contained in FPS_y (MIB_{FPS}) is provided by the following result.

Result 12.(33) The MIB_{FPS} has center λ^{c*}, axis length l^* and rotation R^* solution of

$$\max \prod_{i=1}^{n} l_i \tag{2.69}$$

subject to

$$\lambda^c, l, R : \begin{cases} l_i > 0, i = 1, \ldots, n \\ (u_i^T R^T \lambda^c + c_i) - \Sigma_{j=1}^n l_j |u_{ij}| \geq 0, i = 1, \ldots, 2m \\ R^T R = I \end{cases}$$

□

Equation (2.69) is a polynomial optimization problem which can be solved by use of signomial programming. If matrix rotation R is fixed, Eq. (2.69) reduces to a convex problem with $2n$ variables and $(2m + n)$ linear constraints, which can be efficiently solved by means of normally available convex programming algorithms. If axis length l is also fixed except for a scale factor (i.e., the maximum box of a given shape is sought), Eq. (2.69) reduces to a linear programming problem with $(n + 1)$ variables and $(2m + 1)$ constraints.

2.6. OTHER TYPES OF RESULTS

This section briefly recalls papers on topics related to set membership estimation theory, such as experiment design, estimation with reduced order models, and uncertainty in the information operator. Almost all these papers consider linear problems.

2.6.1. Experiment Design

In the previous sections information operator $\mathcal{A}$ is supposed to be given. In some practical application it is possible to choose among different information operators $\mathcal{A}$ (optimal information problem). For example it may be possible to choose the sampling times at which measurements are taken of the input and the output of the dynamic system to be identified. Then a natural choice is the one minimizing the error $E_\lambda(\phi^c)$. In Ref. 35 some results are given for the case in which information is provided by sampling.(35) In Ref. 20 similar results are derived for more general classes of information. In this paper it is also shown that the optimal sampling times can be chosen *a priori*, and no improvements can be obtained by means of more sophisticated sampling schemes.(20) The optimal sampling problem is approached through p-widths theory.(36)

Another criterion is to minimize the volume of FPS_y.(29) In Ref. 29 a recursive selection procedure is given based on heuristics to avoid poor choices without guaranteeing the best. Characterization is given of the minimum number of sampling times assuring minimum volume of the feasible parameter set FPS_y for $y = \mathcal{A}\lambda$ in Ref. 37.

2.6.2. Reduced Order Models

In the previous sections, it is supposed that the structure of the problem is known, for example the number of autoregressive and moving-average terms for an ARMA model. In many cases, however, the structure of the problem and in particular the dimension of space Λ is not known and must be evaluated from the available information (order determination problems). Some methods are analogous to methods widely used for order determination in statistical contexts,(38,39) such as the principal component analysis and singular value decomposition. A method is also proposed, based on the expected behavior of FPS_y for overparameterized and underparameterized structures.

A second important problem is how estimation algorithms can take into account that only approximating structures are used. The usual approach in statistical contexts is to ignore the deterministic nature of modeling errors, and eventually discard badly approximating structures with residuals evidently not satisfying the assumed statistical hypotheses. In the UBB approach, modeling errors can be taken into account in a more natural way, since it is possible to evaluate bounds on such modeling errors.(40,41,42) A deeper analysis considers explicitly that using approximating structures corresponds to restricting the analysis to a subset $K \subset \Lambda$ *not* containing the "true" problem element λ.(43) In this paper, the concept of conditionally central estimator is introduced as an extension of a central estimator, and it is shown to be Y-locally optimal. The same paper shows that there are two possible ways of extending least squares estimators. The first one corresponds to what is usually done (more or less explicitly) when dealing with reduced order models. However, this estimator does not preserve any of the interesting optimality properties of least squares estimators. A second type of extension is introduced, which is shown to have interesting Λ-locally and Y-locally optimality properties.

2.6.3. Uncertain Information Operator

In some papers the case in which information matrix $\mathcal{A}$ is not exactly known is studied. In particular, perturbation of the type $\mathcal{A} = \mathcal{A}_o + \Delta\mathcal{A}$ has been considered, where $\mathcal{A}_o$ is given and $\Delta\mathcal{A}$ is not known but bounded. A modification of the recursive algorithm for outer ellipsoid bounding reported in Result 10 is proposed.(44) Two different extensions of FPS_y are considered in Refs. 45 and 46. FPS_y^1 is defined in Ref. 45 by considering that Eq. (2.28) holds for all $\Delta\mathcal{A}$ and is described by a set of $m2^{n+1}$ linear inequalities. In Ref. 46, FPS_y^2 is defined by considering that Eq. (2.28) holds for some $\Delta\mathcal{A}$, and the problem of finding the corresponding *MOB* by means of suitable linear programming problems is also discussed.

2.7. APPLICATIONS

The UBB approach is now beginning to be used in various application fields. Some papers report applications to real word problems arising in biology,(47,48) pharmacokinetics,(14,49) time series filtering and prediction,(50,51) economics,(52) chemistry,(53) image processing,(54) ecology,(55,56) measurement,(8,57,58) tracking,(59) and speech processing.(27,60,61,62)

Application of set membership estimation theory has also been investigated in the context of identification for robust and adaptive control design,(28,63,64,41,42,65) and in Chapters 27–30 of this volume.

2.8. CONCLUSIONS

In this chapter an outline of the main results in the area of estimation theory for set membership uncertainty has been presented. The main emphasis of the paper is on the following aspects: existence and characterization of worst-case optimal estimators; actual computation of the derived optimal estimators; evaluation of the errors of optimal and of other widely used estimators (least squares, least absolute values, least maximum values); exact or approximate description of feasible parameter and solution sets. A quick reference to less assessed topics such as experiment design, reduced-order modeling, and more general error models are also made in the paper.

Some general considerations may be drawn from this overview.

Concerning linear problems, real advances have been done in the last decade. As a result, properties of estimators and exact or approximate description of feasible parameter and solution sets can be considered subjects with reasonably well understood and usable solutions. In fact, many of the available algorithms have been used for several applications in different real world problems.

Concerning nonlinear problems, in spite of the work done in the last few years, much more remains to be done. Some algorithms for computing exact parameter or solution uncertainty intervals have been proposed. They work reasonably well on problems with a limited number of measurements and parameters. However, their behavior in more complex situations has not been deeply investigated yet.

Several basic problems remain open and need a thorough investigation, both for linear and nonlinear problems, for example the topological properties of the feasible parameter set as a function of the nonlinearity and uncertainty structures, inner and outer bounding for the nonlinear case, the effects of model approximations, the interaction of set membership estimation theory and robust or adaptive control.

ACKNOWLEDGMENTS. This work was partially supported by Ministero della Università e della Ricerca Scientifica and EC project SIMONET.

REFERENCES

1. H. S. Witsenhausen, *IEEE Trans. Autom. Control* **AC-13**, 556 (1968).
2. F. C. Schweppe, *IEEE Trans. Autom. Control* **AC-13**, 22 (1968).
3. F. C. Schweppe, *Uncertain Dynamic Systems*, Prentice Hall, Englewood Cliffs, NJ (1973).
4. J. F. Traub and H. Wozniakowski, *A General Theory of Optimal Algorithms*. Academic Press, New York (1980).
5. M. Milanese and R. Tempo, *IEEE Trans. Autom. Contr.* **AC-30**, 730 (1985).
6. C. A. Micchelli and T. J. Rivlin, in: *A Survey of Optimal Recovery* (C.A. Micchelli and T. J. Rivlin, eds.) *Optimal Estimation in Approximation Theory*, pp. 1–54; Plenum Press, New York (1977).
7. K. Keesman and G. van Straten, *Proceedings of the 12th IMACS World Congress*, Paris (1988).
8. M. K. Smit, *Measurement* **1**, 181 (1983).
9. E. Walter and H. Piet-Lahanier, *Proceedings of the 25th IEEE Conference on Decision and Control*, Athens (1986).
10. M. Milanese and A. Vicino, *Automatica* **27**, 403 (1991).
11. J. G. Ecker, *SIAM Review* **1**, 339 (1980).
12. J. E. Falk, Tech. Rep. T-274, George Washington University, Washington DC (1973).

13. M. Milanese and A. Vicino, in: *Robust Estimation and Exact Uncertainty Intervals Evaluation for Nonlinear Models*, of *Systems Modelling and Simulation* (S. Tzafetas, A. Eisinberg, and L. Carotenuto, eds.) Elsevier Science Publishers (1988).
14. M. Milanese and G. Belforte, *IEEE Trans. Autom. Control* **AC-27**, 408 (1982).
15. B. Z. Kacewicz, M. Milanese, R. Tempo, and A. Vicino, *Systems and Control* **8**, 161 (1986).
16. R. Tempo and G. Wasilkowski, *Systems and Control* **10**, 265 (1988).
17. R. L. Launer and G. N. Wilkinson, Eds. *Robustness in Statistics*, Academic Press, New York (1979).
18. B. T. Poljak and J. Z. Tsypkin, *Automatica* **16**, 53 (1980).
19. A. van den Bos, *Automatica* **24**, 803 (1985).
20. M. Milanese, R. Tempo, and A. Vicino, *J. Complexity* **2**, 78 (1986).
21. V. V. Kapitonenko, *Autom. Remote Control* **22**, 166 (1982).
22. H. Piet-Lahanier and E. Walter, *Proceedings of the 12th IMACS World Congress*, Paris (1988).
23. S. H. Mo and J. P. Norton, *Proceedings of the 12th IMACS World Congress*, Paris (1988).
24. H. Piet-Lahanier and E. Walter, *Proceedings of the 28th IEEE Conference on Decision and Control*, Tampa (1989).
25. V. Broman and M. J. Shensa, *Proceedings of the 12th IMACS World Congress*, Paris (1988).
26. E. Fogel and F. Huang, *Automatica* **18**, 140 (1982).
27. J. R. Deller, *IEEE Acoust., Speech, Signal Process* **6**, 4 (1989).
28. S. Dasgupta and Y. F. Huang, *IEEE Trans. Inf. Theory* **IT-33**, 383 (1987).
29. J. P. Norton, *Automatica* **23**, 497 (1987).
30. J. P. Norton, *Int. J. Control* **50**, 2623 (1989).
31. G. Belforte and B. Bona, *7th IFAC Symposium on Identification and System Parameter Estimation*, York (1985).
32. L. Pronzato, E. Walter, and H. Piet-Lahanier, *Proceedings of the 28th IEEE Conference on Decision and Control*, Tampa (1989).
33. A. Vicino and M. Milanese, *Proceedings of the 28th IEEE Conference on Decision and Control*, Tampa (1989); Also in *IEEE Trans. Autom. Control* **36**, 759 (1991).
34. A. Vicino and M. Milanese, *9th IFAC/IFORS Symposium on Identification and System Parameter Estimation*, Budapest (1991).
35. G. Belforte, B. Bona, and S. Frediani, *IEEE Tans. Autom. Control* **AC-32**, 179 (1987).
36. C. A. Micchelli, in: *Robustness in Identification and Control* (M. Milanese, R. Tempo, and A. Vicino, eds.) Plenum Press, New York (1989).
37. L. Pronzato and E. Walter, *Automatica* **25**, 383 (1989).
38. J. P. Norton, in: *Robustness in Identification and Control* (M. Milanese, R. Tempo, and A. Vicino, eds.) Plenum Press, New York (1989).
39. S. M. Veres and J. P. Norton, *Int. J. Control* **50**, 639 (1989).
40. R. Genesio and M. Milanese, *IEEE Trans. Autom. Control* **AC-21**, 118 (1976).
41. M. K. Lau, R. L. Kosut, and S. Boyd, *29th IEEE CDC*, Honolulu (1990).
42. J. M. Krause, G. Stein, and P. P. Khargonekar, *29th IEEE CDC*, Honolulu (1990).
43. B. Z. Kacewicz, M. Milanese, and A. Vicino, *J. Complex.* **4**, 73 (1988).
44. T. Clement and S. Gentil, *Proceedings of the 12th IMACS World Congress*, Paris (1988).
45. R. Tempo, B. R. Barmish, and J. Trujillo, *Proceedings of the 8th IFAC Symposium on Identification and Parameter Estimation*, Beijing (1988).
46. G. Belforte, B. Bona, and V. Cerone, *Proceedings of the 12th IMACS World Congress*, Paris (1988).
47. G. Belforte, B. Bona, and M. Milanese, *CRC J. Biomed. Eng.* **10**, 275 (1983).
48. J. P. Norton, *Proceedings of the 25th IEEE Conference on Decision and Control*, Athens (1986).
49. R. Gomeni, H. Piet-Lahanier, and E. Walter, *Proceedings of the 3rd IMEKO Congress on Measurements in Clinical Medicine*, Edinburgh (1986).
50. A. Vicino, R. Tempo, R. Genesio, and M. Milanese, *Int. J. Forecasting* **2**, 313 (1987).

51. R. Tempo, *IEEE Trans. Autom. Contr.* **AC-33**, 864 (1988).
52. M. Milanese, R. Tempo, and A. Vicino, *Int. J. Systems Sci.* **19**, 1189 (1988).
53. E. Walter, Y. Lecourtier, J. Happel, and J. Y. Kao, *AIChE J.* **32** , 1360 (1986).
54. A. Venot, L. Pronzato, E. Walter, and J. F. Lebruchec, *Automatica* **22**, 105 (1986).
55. G. Van Straten, *Appl. Math. Comp.* **17**, 459 (1985).
56. K. J. Keesman and G. Van Straten, *Proceedings of the IAWPRC Symposium on System Analysis in Water Quality Management*, Pergamon Press, New York (1987).
57. G. Belforte, B. Bona, E. Canuto, F. Donati, F. Ferraris, I. Gorini, S. Morei, M. Peisino, and S. Sartori, *Annals CIRP* **36**, 359 (1987).
58. J. W. Verhoof and M. K. Smit, *Proceedings of the 12th IMACS World Congress*, Paris (1988).
59. V. Broman and M. J. Shensa, *Proceedings of the 25th IEEE Conference on Decision and Control*, Athens (1986).
60. J. R. Deller and T. C. Luk, *Proceedings of the IEEE International Conference on Acoustic, Speech and Signal Processing*, Dallas (1987).
61. J. R. Deller and D. Hsu, *IEEE Trans. Circ. Syst.* **CAS-34**, 782 (1987).
62. P. L. Combettes and H.J. Trussell, *Proceedings of the IEEE International Conference Acoustics, Speech and Signal Processing,* New York (1988).
63. R. L. Kosut, *Int. J. Adapt. Control Signal Proc.* **2**, 371 (1988).
64. A. Vicino, A. Tesi, and M. Milanese, *IEEE Trans. Autom. Control* **AC-35**, 835 (1990).
65. G. Fiorio, S. Malan, M. Milanese, and A. Vicino, *29th IEEE CDC*, Honolulu (1990).

3

Solving Linear Problems in the Presence of Bounded Data Perturbations

B. Z. Kacewicz

3.1. INTRODUCTION

In most computational problems of engineering or numerical analysis available input data (information) is not exact. Perturbations in data may arise for instance from measurement or round-off errors, to mention only these two possible sources. The problem of how inaccuracy in data influences results (for instance, how does it affect a quality of system identification or signal recovery) attracts attention not only for obvious practical reasons, but also motivates a number of theoretical papers. For example, since a long time the case of stochastic errors in information has been studied by statisticians, to mention only the monograph by Wahba,[1] where extensive references to the subject can be found. On the other hand, an active stream of research is based on deterministic assumptions about the noise. Such assumptions are imposed when no appropriate statistical knowledge about the behavior of data errors is available, or simply when statistical analysis is not of interest. The assumption often made in this framework is that errors in information are unknown but bounded. Among many other papers, the bounding approach is discussed in Refs. 2–5.

B. Z. Kacewicz • Institute of Applied Mathematics, University of Warsaw, 02-097 Warsaw, Poland.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

This chapter presents some results obtained in unknown-but-bounded approach using the tools of Information Based Complexity (IBC), one of the fields of theoretical numerical analysis. Our main objective is to study the minimal cost of solving linear problems in the presence of errors in data. Although the framework of the presentation is rather 'theoretical,' the results and tools of IBC may be useful in the identification field, as shown by a growing interest in such methods.[2,6]

In a general formulation, the problem is to approximate the solution $S(f)$ for elements f belonging to a certain ball K in a linear normed space, where S is a linear continuous operator. To find an approximation, we gather information by successively computing some numbers $z_1, z_2, \ldots$ dependent on f. Each z_n is assumed to be a noisy evaluation of a linear continuous functional at f (e.g., it may be an inaccurate evaluation of a function f) and is available at a certain cost. For a given $\varepsilon > 0$, the aim is to obtain an approximation with error at most ε. That is, data is gathered during n successive steps, where n is the minimal number of evaluations z_1,z_2, ..., z_n which yield an ε-approximation. Obviously, a good termination criterion to stop a data collecting process is needed. A 'good' criterion is meant in the sense that it minimizes (or nearly minimizes) the total cost of obtaining an ε-approximation. A detailed formulation of the problem together with a model example of signal transmission is given in Sections 3.2 and 3.3.

The termination criterion based on the diameter of information, a quantity closely related to the minimal error of an algorithm and often used in IBC, is discussed in Sections 3.4 and 3.5. The choice of this criterion is motivated by the fact that the diameter of information is relatively well studied and 'easy' to compute and manage which makes the criterion useful in further considerations. It turns out that the cost yielded by the diameter termination criterion is (almost) minimal, i.e., the criterion is not only convenient, but also 'optimal.'

Section 3.6 concentrates on results concerning the minimal cost for the diameter termination criterion. Under some assumptions, the minimal cost turns out to be proportional to the minimal number of functionals needed to compute an ε-approximation in the case of *exact* data, multiplied by the cost of obtaining one current information value. The results are illustrated by an example.

Section 3.7 discusses the dependence of the diameter of information on data perturbations for the problem of signal recovery. The problem of how errors in data influence the result is interesting from a general point of view. In this context it is also important when applying the results of the preceding section, where knowledge about such an influence is needed. For the considered problem, the minimal error of an algorithm is bounded from above by a linear function in data errors, with constants dependent on parameters of the problem.

In summary, this chapter is devoted to the general problem of quality of information contaminated by unknown but bounded noise. Results on the minimal cost of computing an ε-approximation are given, as well as on the dependence of the minimal error on data perturbations.

3.2. EXAMPLE: A SIGNAL RECOVERY PROBLEM

To illustrate a general problem formulation coming up in the next section, consider an example motivated by a signal recovery problem which arises, e.g., in speech or image reconstruction.* In this example we assume that a signal is first sampled and its quantization is done to fit a certain number of bits. Next (e.g., after data transmission), the signal is recovered from available (incomplete) information. The reconstruction is to be done with possibly small error which obviously depends on the number of samples and the size of memory used. Or, in an alternate formulation, the size of memory needed to keep the reconstruction error on a prescribed level is to be minimized.

This brief description can be formalized as follows. Let F be the space of real functions in s variables with r ($r \geq 1$) continuous derivatives,

$$F = C^r([0,1]^s),$$

with the norm

$$\|f\| = \max_{0 \leq k_1+\cdots+k_s=i\leq r} \ \sup_{x\in[0,1]^s} \left| \frac{\partial^i f(x)}{(\partial x^1)^{k_1}\cdots(\partial x^2)^{k_s}} \right|, \ f \in F,$$

where $x = [x^1, \ldots, x^s]$. We gather information by sampling the function f,

$$N(f) = [f(t_1), f(t_2), \ldots, f(t_n), \ldots], \tag{3.1}$$

at some points $t_i \in [0,1]^s$, $i \geq 1$. Each value $f(t_i)$ is rounded using binary representation with m_i bits. That is, instead of $f(t_i)$ we have at our disposal a number z_i such that

$$|z_i - f(t_i)| \leq 2^{-m_i}, \ i \geq 1. \tag{3.2}$$

A sequence $[z_1, \ldots, z_n, \ldots]$ is called perturbed information about f.

The aim is to recover a function f with $\|f\| \leq 1$ within a given accuracy $\varepsilon > 0$ using the perturbed information. That is, to find n and a function $g_n = g_n(z_1, ..., z_n)$ in $\mathbf{C}([0,1]^s)$ such that

$$\|g_n - f\|_\infty \leq \varepsilon.$$

The calculation of g_n is connected with a cost which can be measured, e.g., by a number of bits $\Sigma_{i=1}^n m_i$ necessary to store information. This cost should be as small

*This example and the material from Section 3.6 have been extracted from Ref. (3.8) and are included in this chapter courtesy of Marcel Dekker Inc.

as possible. In addition, we want to determine the optimal sampling points t_i, and the optimal number of bits m_i for which the cost is minimized.

We now describe a generalization of the above problem.

3.3. GENERAL PROBLEM FORMULATION

Let S, $S \neq 0$, be a linear continuous operator acting from a Banach space F to a linear normed space G.[†] Let $K = \{f \in F : \|f\| \leq 1\}$. We wish to approximate the solution $S(f)$ for all $f \in K$, based on the knowledge about f restricted only to some perturbed information about f. For $f \in K$, information $N(f)$ is gathered by a successive calculation (or observation) of certain numbers,

$$N(f) = [L_1(f), L_2(f), \ldots], \tag{3.3}$$

where $L_i : F \to \mathbf{R}$ are linear continuous functionals, $\|L_i\| \leq 1$, belonging to a certain class Λ, $i \geq 1$. With no misunderstanding, the operator $N : F \to \mathbf{R}^\infty$ given by Eq. (3.3) will also be called information. Collecting information is continued until some terminating condition is fulfilled.

Assume that instead of the exact values $L_i(f)$ only perturbed values z_i can be evaluated (or observed) such that

$$|z_i - L_i(f)| \leq \Delta_i, \tag{3.4}$$

where $\Delta_i \geq 0$, $i \geq 1$. The sequence $\overline{\Delta} = [\Delta_1, \Delta_2, \ldots] \in \mathbf{R}^\infty$ is called a precision sequence.

The nth approximation g_n to $S(f)$ is obtained based on the values z_i (not on f itself which is unknown) as $g_n = \phi_n(z_1, \ldots, z_n)$, where ϕ_n is a mapping from $\mathbf{R}^n$ to G. The sequence $\phi = \{\phi_n\}_{n=0}^{\infty}$ is called an (idealized) algorithm (ϕ_0 means a fixed element of G).

In the example from the preceding section $F = \mathbf{C}^r([0,1]^s)$, $G = \mathbf{C}([0,1]^s)$, the operator S is given by $S(f) = f$, information functionals are defined by $L_i(f) = f(t_i)$ and $\Delta_i = 2^{-m_i}$.

An algorithm usually produces some error. Results that can be obtained and their interpretation strongly depend on how the error of an algorithm is measured. In this chapter, the nth *error* of ϕ at f is defined as

$$e_n(\phi, N, \overline{\Delta}, f) = \sup\{\|S(f) - \phi_n(z_1, \ldots, z_n)\| : |z_i - L_i(f)| \leq \Delta_i, 1 \leq i \leq n\}. \tag{3.5}$$

[†] The material from Sections 3.3, 3.4 and 3.5 has been extracted from Ref. (3.7) and reprinted here with minor modifications by permission of the American Mathematical Society.

That is, the error is measured for a fixed f as the maximal distance between the solution and the approximation, where the maximum is taken with respect to all possible data perturbations.

Given $\varepsilon > 0$, we compute the values $z_1, z_2, \ldots$ until the error does not exceed ε. Once such an accuracy is achieved, it should be maintained, if for some reason calculations happen to continue. Hence, the number of steps to terminate is equal to

$$n(\phi,N,\overline{\Delta},f)(\varepsilon) = \min\{n \geq 0 : e_j(\phi,N,\overline{\Delta},f) \leq \varepsilon,\ \forall j \geq n\} \tag{3.6}$$

(with the convention $\min \varnothing = +\infty$).

The above termination condition is only 'theoretical.' It reflects demands concerning the termination, but it is not 'computable' as it depends on the unknown element f. The sequel shall define another criterion which is as effective as in Eq. (3.6) but independent of f.

Assume that collecting information is connected with some cost, i.e., we are charged for each evaluation (observation) of a functional. The cost of obtaining a value z such that $|z - L(f)| \leq \Delta$ is assumed to be $c(\Delta)$, where $c : [0, +\infty) \rightarrow [0, +\infty]$ is a given nonincreasing function, positive for sufficiently small $\Delta > 0$ and independent of L, f and z. In the example of Section 3.2 $c(\Delta) = \log_2(1/\Delta)$.

The information cost (or cost) of obtaining an ε-approximation using the algorithm ϕ with information N, the precision sequence $\overline{\Delta}$ and the termination criterion of Eq. (3.6) is defined by

$$C(\phi,N,\overline{\Delta},f)(\varepsilon) = \sum_{i=1}^{m} c(\Delta_i) \tag{3.7}$$

for $m < +\infty$, and $C(\phi,N,\overline{\Delta},f)(\varepsilon) = +\infty$ for $m = +\infty$, where $m = n(\phi,N,\overline{\Delta},f)(\varepsilon)$. (The convention $\Sigma_{i=1}^{0} = 0$ is used.) In Section 3.2 we have $C(\phi,N,\overline{\Delta},f)(\varepsilon) = \Sigma_{i=1}^{m} m_i$.

In addition to the information cost, the actual cost of constructing an approximation also consists of the combinatory cost of calculating $\phi_n(z_1, \ldots, z_n)$, but the latter neglected. It turns out that for many important problems there exists a 'good' algorithm with the combinatory cost relatively small.[10]

The purpose of this chapter is to analyze the behavior of $C(\phi,N,\overline{\Delta},f)(\varepsilon)$ as $\varepsilon \rightarrow 0^+$.

Now turn to defining a termination condition 'equivalent' to that given in Eq. (3.6), but easier to compute. To this end, recall the concept of the nth diameter of information, which is given by

$$d_n(N,\overline{\Delta}) = 2 \cdot \sup\{\|S(h)\| : h \in F,\ \|h\| \leq 1,\ |L_i(h)| \leq \Delta_i,\ 1 \leq i \leq n\}.$$

It is equal (up to a factor of 1/2) to the minimal error of an algorithm for the worst element f.[10,11]

We shall also need the concept of an interpolatory algorithm $\phi^* = \{\phi_n^*\}_{n\geq 0}$ (see [10]). For $n \geq 1$ and $[z_1, z_2, \ldots]$ being perturbed information for some $f \in K$, take an interpolant $\sigma_n = \sigma_n(z_1, \ldots, z_n) \in F$ such that

$$|L_i(\sigma_n) - z_i| \leq \Delta_i, \quad 1 \leq i \leq n,$$

and define an approximation to be the true solution for σ_n,

$$\phi_0^* = 0, \quad \phi_n^*(z_1, \ldots, z_n) = S(\sigma_n), \quad n \geq 1.$$

It is known that

$$e_n(\phi^*, N, \overline{\Delta}, f) \leq d_n(N, \overline{\Delta}), \quad \forall f \in K. \tag{3.8}$$

Hence, if the algorithm ϕ^* is applied, it is enough to compute $n^d(N, \overline{\Delta})(\varepsilon)$ pieces of information to obtain an ε-approximation, where

$$n^d(N, \overline{\Delta})(\varepsilon) = \min\{n \geq 0 : d_n(N, \overline{\Delta}) \leq \varepsilon\}. \tag{3.9}$$

Note that the termination criterion of Eq. (3.9) does not depend on an element f, but only on the class of all elements K. For many problems the behavior of $d_n(N, \overline{\Delta})$ is known.(10,12) The number $n^d(N, \overline{\Delta})(\varepsilon)$ can be computed, in contrast with the quantify $n(\phi, N, \overline{\Delta}, f)(\varepsilon)$. However the criterion of Eq. (3.9) may be useful only if the number $n^d(N, \overline{\Delta})(\varepsilon)$ is not much greater than $n(\phi, N, \overline{\Delta}, f)(\varepsilon)$. As we shall see, this is indeed the case.

The information cost of obtaining an ε-approximation using N, $\overline{\Delta}$ and ϕ^* with the stopping criterion of Eq. (3.9) is independent of f and equal to

$$C^d(N, \overline{\Delta})(\varepsilon) = \sum_{i=1}^{m} c(\Delta_i) \tag{3.10}$$

for $m < +\infty$, and $C^d(N, \overline{\Delta})(\varepsilon) = +\infty$ for $m = +\infty$, where $m = n^d(N, \overline{\Delta})(\varepsilon)$. We call $C^d(N, \overline{\Delta})(\varepsilon)$ the diameter criterion cost. Equation (3.8) yields that, for any $\varepsilon > 0$ and $f \in K$, one has

$$C(\phi^*, N, \overline{\Delta}, f)(\varepsilon) \leq C^d(N, \overline{\Delta})(\varepsilon). \tag{3.11}$$

A deeper relation between the costs of Eqs. (3.7) and (3.10) will be discussed later. It will be shown that the upper bound (3.11) is essentially sharp, i.e., the criterion (3.9) is *not* pessimistic.

Furthermore, it is interesting to determine information N, an algorithm ϕ and a precision sequence $\overline{\Delta}$ for which the cost $C(\phi, N, \overline{\Delta}, f)(\varepsilon)$ grows as slowly as possible as $\varepsilon \to 0^+$. To show what the slowest possible growth is, the next two sections study a relation between $C(\phi, N, \overline{\Delta}, f)(\varepsilon)$ and the minimal diameter criterion cost defined by

$$MC^d(\varepsilon) = \inf_{N,\bar{\Delta}} C^d(N, \bar{\Delta})(\varepsilon), \tag{3.12}$$

the infimum taken with respect to information N consisting of functionals from Λ.

We start with a construction of information N^* and a precision sequence $\bar{\Delta}^*$ which supply an (almost) ε-approximation with cost no greater than $MC^d(\varepsilon)$.

3.4. CONSTRUCTION OF OPTIMAL INFORMATION AND PRECISION SEQUENCE

We first consider problems which are solvable with respect to the criterion of Eq. (3.9), i.e., such that $MC^d(\varepsilon) < +\infty$, $\forall\varepsilon > 0$. The case $MC^d(\varepsilon) = +\infty$ (for small ε) is considered in Theorem 3.4 (ii), which states that the problem is then practically not solvable even if the theoretical criterion of Eq. (3.6) is used.

Assume that the problem is 'hard' in the following sense:

(A) There exist $0 < p < 1$ and $\alpha > 1$ such that

$$MC^d(\alpha \cdot \varepsilon) \le p \cdot MC^d(\varepsilon),$$

for all sufficiently small $\varepsilon > 0$.

Note that the inequality (A) always holds with $p = 1$. For $p < 1$ it states that the minimal diameter criterion cost tends to infinity sufficiently fast as ε decreases. This holds, for example, for the problem described in Section 3.2, see Theorem 3.6.

To define N^* and $\bar{\Delta}^*$, take for $\omega > 1$ and $i \ge 0$ information $N^i = [L_1^i, L_2^i, \ldots]$ consisting of functionals from Λ, a precision sequence $\bar{\Delta}^i = [\Delta_1^i, \Delta_2^i, \ldots]$ and an integer $n_i > 0$ such that

$$d_{n^i}(N^i, \bar{\Delta}^i) \le \frac{1}{\alpha^i}, \tag{3.13}$$

and

$$\sum_{j=1}^{n^i} c(\Delta_j^i) \le \omega \cdot MC^d\left(\frac{1}{\alpha^i}\right). \tag{3.14}$$

This selection is possible for sufficiently large i, $i \ge l$, where $l \ge 0$. Denoting by $N_{n^i}^i$ and $\bar{\Delta}_{n^i}^i$ the first n^i components of N^i and $\bar{\Delta}^i$, respectively, define

$$N^* = [N_{n^l}^l, N_{n^{l+1}}^{l+1}, N_{n^{l+2}}^{l+2}, \ldots], \tag{3.15}$$

and

$$\bar{\Delta}^* = [\bar{\Delta}_{n^l}^l, \bar{\Delta}_{n^{l+1}}^{l+1}, \bar{\Delta}_{n^{l+2}}^{l+2}, \ldots].$$

What are the properties of the information N^* and the precision sequence $\overline{\Delta}^*$ from the point of view of the cost of obtaining an ε-approximation? The following theorem shows that the interpolatory algorithm ϕ^* using N^* and $\overline{\Delta}^*$ produces an (almost) ε-approximation with the cost proportional to $MC^d(\varepsilon)$, if any of the criteria of Eqs. (3.6) or (3.9) is applied.[7]

THEOREM 3.1. Let $MC^d(\varepsilon) < +\infty$ for all $\varepsilon > 0$, and let the condition (A) hold. Then, for all $f \in K$ and all sufficiently small $\varepsilon > 0$ we have that

$$C(\phi^*, N^*, \overline{\Delta}^*, f)(\alpha \cdot \varepsilon) \leq C^d(N^*, \overline{\Delta}^*)(\alpha \cdot \varepsilon) \leq \frac{\omega}{1-p} MC^d(\varepsilon).$$

The above theorem gives only an upper bound on the cost of computing an ε-approximation using N^*, $\overline{\Delta}^*$ and ϕ^*. We now ask: What is the quality of the obtained estimate? Can the upper bound from Theorem 3.1 be improved? In the next section N^* and $\overline{\Delta}^*$ are shown to be (almost) optimal, in the sense that the cost of obtaining an ε-approximation using arbitrary N and $\overline{\Delta}$ cannot be much smaller that $MC^d(\varepsilon)$, even if the theoretical condition of Eq. (3.6) is used.

3.5. LOWER BOUNDS

For arbitrary N, $\overline{\Delta}$ and ϕ, lower bounds on the cost $C(\phi, N, \overline{\Delta}, f)(\varepsilon)$ turn out, roughly speaking, to be given by $MC^d(\varepsilon)$, which shows sharpness of the upper bound from Theorem 3.1. The lower bounds on the cost hold on dense sets of element f. (A set D in a normed space X is called dense if elements of D can be found in any ball in X. That is, each element of X can be approached arbitrarily closely by elements of D.)

Consider first fixed N and $\overline{\Delta}$. Start with the case $C^d(N, \overline{\Delta})(\varepsilon) < +\infty$, for all $\varepsilon > 0$. We have[7]

THEOREM 3.2. Let $C^d(N, \overline{\Delta})(\varepsilon) < +\infty$, $\forall \varepsilon > 0$, and let ϕ be an arbitrary algorithm.

(i) If $d_n(N, \overline{\Delta}) > 0$, $\forall n \geq 0$, then for any function $h : (0, +\infty) \to (0, +\infty)$ with $\lim_{\varepsilon \to 0^+} h(\varepsilon) = 0$ the set

$$A_1 = \{f \in K : \exists C = C(f) \geq 0 \; \exists \varepsilon_0 = \varepsilon_0(f) > 0 \text{ such that for all } 0 < \varepsilon \leq \varepsilon_0$$

$$C(\phi, N, \overline{\Delta}, f)(C \cdot h(\varepsilon) \cdot \varepsilon) \leq C^d(N, \overline{\Delta})(\varepsilon)\}$$

has a dense complement in K, i.e., the set $K - A_1$ is a dense set in K.

(ii) If $d_n(N, \overline{\Delta}) = 0$ for some n, then the set
$A_2 = \{f \in K : \exists C = C(f) \geq 0 \; \exists \varepsilon_0 = \varepsilon_0(f) > 0$ such that for all $0 < \varepsilon \leq \varepsilon_0$

$$C(\phi, N\overline{\Delta}, f)(C \cdot \varepsilon) < C^d(N, \overline{\Delta})(\varepsilon)\}$$

has a dense complement in K, i.e., the set $K - A_2$ is a dense set in K.

Theorem 3.2 provides a lower bound on $C(\phi, N, \overline{\Delta}, f)(\varepsilon)$. To see this, note that by (i) the inequality (3.11) is no more true (on a dense set of fs), if only ε in the left hand side is replaced by a function $h(\varepsilon) \cdot \varepsilon$. This holds no matter what algorithm ϕ is used. The function $h(\varepsilon)$ is arbitrary, i.e., it may tend to 0 arbitrarily slowly with ε, so that replacing ε by $h(\varepsilon) \cdot \varepsilon$ corresponds to a possibly very slight increase in accuracy requirements. Note also that, due to Eq. (3.11), the function $h(\varepsilon)$ cannot be omitted in the formulation of the theorem. In the case (ii), the theorem states that weak inequality (3.11) cannot be replaced by sharp one. Combined upper and lower bounds from Theorems 3.1 and 3.2 imply that, given N and $\overline{\Delta}$, the interpolation algorithm ϕ^* is almost optimal.

In terms of the termination criteria, the above result is somewhat surprising. It states that the theoretical stopping condition of Eq. (3.6) yields larger cost than the criterion of Eq. (3.9), if the accuracy required in Eq. (3.6) is only slightly increased (by a function $h(\varepsilon)$) with respect to the accuracy required in Eq. (3.9).

In the case when $C^d(N, \overline{\Delta})(\varepsilon) = +\infty$ for sufficiently small $\varepsilon > 0$, i.e., for problems which cannot be solved (due to the infinite cost) with respect to the criterion of Eq. (3.9) it holds:[(7)]

THEOREM 3.3. Let $C^d(N, \overline{\Delta})(\varepsilon) = +\infty$ for sufficiently small $\varepsilon > 0$, and let ϕ be an arbitrary algorithm.

(i) If $\lim_{n\to+\infty} d_n(N, \overline{\Delta}) > 0$ and $\Sigma_{i=1}^{\infty} c(\Delta_i) = +\infty$, then for any function $H : (0, +\infty) \to [0, +\infty)$ the set

$$A_3 = \{f \in K : \exists C = C(f) \geq 0 \; \exists \varepsilon_0 = \varepsilon_0(f) > 0 \text{ such that for all } 0 < \varepsilon \leq \varepsilon_0$$

$$C(\phi, N, \overline{\Delta}, f)(C \cdot \varepsilon) \leq H(\varepsilon)\}$$

has a dense complement in K.

(ii) If $\lim_{n\to+\infty} d_n(N,\overline{\Delta}) > 0$ and $\Sigma_{i=1}^{\infty} c(\Delta_i) < +\infty$ or if $\lim_{n\to+\infty} d_n(N,\Delta) = 0$, then the set

$$A_4 = \{f \in K : C(\phi, N, \overline{\Delta}, f)(\varepsilon) < +\infty \quad \forall \varepsilon > 0\}$$

has a dense complement in K.

Theorem 3.3 assures that if the problem cannot be approximated with finite cost using the termination criterion of Eq. (3.9) then it also cannot be approximated even if the 'ideal' criterion of Eq. (3.6) is applied. For any algorithm ϕ, the cost is then arbitrarily large (in the case (i)), or infinite (in the case (ii)), on a dense set of elements f.

Theorem 3.2, 3.3 and the inequality $MC^d(\varepsilon) \leq C^d(N, \overline{\Delta})(\varepsilon)$, (for all $N, \overline{\Delta}$) yield the final lower bound on $C(\phi, N, \Delta, f)(\varepsilon)$.

THEOREM 3.4. Let N, $\overline{\Delta}$ and ϕ be arbitrary information, precision sequence, and algorithm, respectively. We have

(i) If $MC^d(\varepsilon) < +\infty$, $\forall\varepsilon > 0$, then for any function $h : (0, +\infty) \to (0, +\infty)$ with $\lim_{\varepsilon\to 0^+} h(\varepsilon) = 0$ the set

$$B_1 = \{f \in K : \exists C = C(f) \geq 0\ \exists\varepsilon_0 = \varepsilon_0(f) > 0 \text{ such that for all } 0 < \varepsilon \leq \varepsilon_0$$

$$C(\phi,N,\overline{\Delta},f)(C \cdot h(\varepsilon) \cdot\varepsilon) < MC^d(\varepsilon)\}$$

has a dense complement in K.

(ii) If $MC^d(\varepsilon) = +\infty$ for sufficiently small $\varepsilon > 0$, then for any function $H : (0, +\infty) \to [0, +\infty)$ the set

$$B_2 = \{f \in K : \exists C = C(f) \geq 0\ \exists\varepsilon_0 = \varepsilon_0(f) > 0 \text{ such that for all } 0 < \varepsilon \leq \varepsilon_0$$

$$C(\phi, N, \overline{\Delta},f)(C \cdot \varepsilon) \leq H(\varepsilon)\}$$

has a dense complement in K.

In the case $MC^d(\varepsilon) < +\infty$ the cost $C(\phi, N, \overline{\Delta},f)(\varepsilon)$ grows essentially (i.e., up to a function $h(\varepsilon)$) at least as fast as $MC^d(\varepsilon)$, as $\varepsilon \to 0^+$, for f belonging to a dense subset of K. If the problem satisfies the condition (A) then information N^*, the precision sequence $\overline{\Delta}^*$ defined in Section 3.4 and the interpolation algorithm ϕ^* are almost optimal, i.e., $C(\phi^*, N^*, \overline{\Delta},f)(\varepsilon)$ essentially behaves like $MC^d(\varepsilon)$, for all $f \in K$. In the case $MC^d(\varepsilon) = +\infty$, the cost $C(\phi, N, \overline{\Delta},f)(\varepsilon)$ grows arbitrarily fast as $\varepsilon \to 0^+$ for any ϕ, N, and $\overline{\Delta}$, on a dense set of f.

Hence, the problem of finding the optimal N, $\overline{\Delta}$, and ϕ for the 'theoretical' stopping condition of Eq. (3.6) can be essentially reduced to the similar (but easier) problem with the criterion of Eq. (3.9). In both cases, the minimal cost essentially behaves like $MC^d(\varepsilon)$, which means that the diameter termination criterion is as effective as the 'theoretical' one.

3.6. THE MINIMAL DIAMETER CRITERION COST

This section concentrates on results about the minimal diameter criterion cost and shows tight bounds on $MC^d(\varepsilon)$ for some class of problems. Let $d_0 = 2\|S\|$ $(0 < d_0 \leq +\infty)$, and

$$d_n = \inf_N d_n(N, \overline{0}), \quad n \geq 1,$$

where the infimum is taken over all information operators $N = [L_1, L_2, \ldots]$, $L_i \in \Lambda$, and $\overline{0} = [0,0,\ldots]$. The number d_n is thus the nth minimal diameter of *exact* information, well studied in the complexity literature.[10]

Assume that the problem satisfies the following conditions: there exists a constant D, $0 \leq D < +\infty$, such that for any $L \in \Lambda$ and $h \in F$ it holds

$$|L(h)| \leq D \cdot \|S(h)\|. \qquad \text{(A1)}$$

One can easily check that the condition (A1), although restrictive in general, holds with $D = 1$ for the reconstruction problem from Section 3.2.

The second condition deals with the behavior of the minimal diameter in the case when the precision vector has equal components. Let

$$d_n(\Delta) = \inf d_n(N,[\Delta, \Delta, \dots]),$$

where the infimum is taken over all N consisting of functionals from Λ ($d_n = d_n(0)$). Assume that there is a constant M, $0 < M < +\infty$, such that

$$d_n(\Delta) \leq M \cdot (d_n + \Delta) \tag{A2}$$

for $n \geq 1$ and $\Delta \geq 0$.

For $\varepsilon > 0$, let

$$n^*(\varepsilon) = \min\{n \geq 1 : d_n \leq \varepsilon\}.$$

The following result gives bounds on $MC^d(\varepsilon)$ in terms of the minimal number of functionals necessary to solve the problem in the case of exact information $n^*(\varepsilon)$ and the single evaluation cost $c(\Delta)$.

THEOREM 3.5. Let the conditions (Al) and (A2) hold, and $\lim_{n\to\infty} d_n = 0$. Then there are constants M_1 and M_2 such that for all $0 < \varepsilon < \|S\|$

$$n^*(M_2\varepsilon) \cdot c(M_2\varepsilon) \leq MC^d(\varepsilon) \leq n^*(M_1\varepsilon) \cdot c(M_1\varepsilon).$$

If $\varepsilon \geq \|S\|$ then $MC^d(\varepsilon) = 0$.

For many problems the asymptotic behavior of $n^*(\varepsilon)$ and $c(\varepsilon)$ is such that

$$n^*(\alpha\varepsilon) = \Theta(n^*(\varepsilon)) \text{ and } c(\alpha\varepsilon) = \Theta(c(\varepsilon)), \text{ as } \varepsilon \to 0^+,$$

for any $\alpha > 0$. In this case it follows from Theorem 3.5 that

$$MC^d(\varepsilon) = \Theta(n^*(\varepsilon) \cdot c(\varepsilon)), \quad \text{as } \varepsilon \to 0^+.$$

To illustrate the above results recall the example from Section 3.2. Recall that the question under consideration is to minimize the number of bits necessary to store a signal and to recover it with given accuracy ε. It is possible to prove the following[(8)]

THEOREM 3.6. The minimal number of bits necessary to store information which allows to recover all functions $f \in F$, $\|f\| \leq 1$, with the error at most ε is equal to

$$MC^d(\varepsilon) = \Theta\left(\varepsilon^{-s/r} \cdot \log_2\left(\frac{1}{\varepsilon}\right)\right), \quad \text{as } \varepsilon \to 0^+.$$

Furthermore, to achieve $MC^d(\varepsilon)$, it is sufficient to evaluate function values at n uniformly distributed points, $n = \Theta(\varepsilon^{-s/r})$, and store them using the same number of bits, $m_i = m$, where $m = \Theta(\log_2(1/\varepsilon))$, $1 \le i \le n$. The ε-approximation is provided by a piecewise polynomial interpolation.

In the next section, the assumption (A2) of Theorem 3.5, which deals with the dependence of the minimal diameter on data perturbations, is discussed. We show how the diameter is influenced by inaccuracy in data for the problem of recovering band- and energy-limited signals.

3.7. THE DIAMETER OF INACCURATE INFORMATION

This section briefly describes a problem of reconstructing signals from data given by their nonexact samples. Next it presents a formulation of two results concerning the diameter of information, the full proofs of which and other related results can be found in Ref. [9].

Let $L_2 = L_2[-\Omega,\Omega]$ denote the Hilbert space of all square integrable complex valued functions f on the interval $[-\Omega,\Omega]$, and let $B = B(L_2)$ denote the unit ball in L_2. Any function f in B yields a band- and energy-limited signal

$$\check{f}(t) = \int_{-\Omega}^{\Omega} f(\omega)\exp(i\omega t)\,d\omega, \quad t \in \mathbf{R}, \ i = \sqrt{-1}.$$

The bandwidth and the energy of $\check{f}$ are 2Ω and $2\pi\|f\|^2$, respectively. Given real distinct points $t_0, t_1, t_2, \ldots, t_n$ we wish to recover a value $\check{f}(t_0)$ for $f \in B$, with the sole knowledge of $\check{f}$ being a vector $\mathbf{z} \in \mathbf{C}^n$ such that

$$|||\mathbf{z} - N(f)||| \le \Delta, \tag{3.16}$$

where

$$N(f) = [\check{f}(t_1), \check{f}(t_2), \ldots, \check{f}(t_n)]^T.$$

Here Δ is a given nonnegative number and $|||\cdot|||$ is a fixed norm in $\mathbf{C}^n$. That is, the data (information) consists of inaccurate samples of $\check{f}$. The error in data is measured here by an arbitrary norm $|||\cdot|||$, which is a slight generalization with respect to the situation from previous sections, where we have $|||\mathbf{z}||| = \max_{1\le i\le n} |z_i|$.

In an alternate formulation this is the problem of reconstructing the functional S given by

$$S(f) = \langle f, u_0\rangle \tag{3.17}$$

from data $\mathbf{z} \in \mathbf{C}^n$ satisfying the inequality (3.16) with the samples vector reinterpreted as

$$N(f) = [\langle f, u_1\rangle, \langle f, u_2\rangle, \ldots, \langle f, u_n\rangle]^T, \tag{3.18}$$

where $\langle \cdot,\cdot \rangle$ is the inner product in $L_2[-\Omega,\Omega]$ and $u_k(\cdot) = \exp(-it_k\cdot)$ for $k = 0, 1, \ldots, n$.

Results on the diameter of information are formulated in terms of the radius of information $r(\Delta)$, a quantity known to be (for the above problem) the minimal error of an algorithm for the worst f,

$$r(\Delta) = r(u_0; u_1, u_2, \ldots, u_n; \Delta) = \sup\{|S(h)|: h \in B, |||N(h)||| \leq \Delta\}.$$

The diameter of information is equal in this case to $2r(\Delta)$.(10)

How is the radius of information $r(\Delta)$ related to $r(0)$ and the precision Δ? Let

$$\mathbf{d} = [\langle u_0, u_1\rangle, \langle u_0, u_2\rangle, \ldots, \langle u_0, u_n\rangle]^T \tag{3.19}$$

and $G = G(u_{1,} u_{2,} \ldots, u_n)$ be the Gram matrix of the system $\{u_j\}_{j=1}^n$,

$$G = (\overline{\langle u_j, u_k\rangle})_{j,k=1}^n,$$

where $\bar{c}$ is a conjugate to a complex number c. We have(9)

THEOREM 3.7. For any $\Delta \geq 0$

$$r(\Delta) = \sup(|\mathbf{d}^H G^{-1}\mathbf{a}| + (1 - \|G^{-1/2}\mathbf{a}\|_2^2)^{1/2}\, r(0)),$$

where the supremum is taken over all $\mathbf{a} \in \mathbf{C}^n$ such that $|||\mathbf{a}||| \leq \Delta$ and $\|G^{-1/2}\mathbf{a}\|_2 \leq 1$.

Although this theorem gives an exact formula for the radius of information $r(\Delta)$, at first sight the dependence on Δ may be not clear. To see it better, note its consequences: an upper bound and the asymptotic behavior of $r(\Delta)$.

COROLLARY 3.1. For any $\Delta \geq 0$ we have

$$r(\Delta) \leq r(0) + r'(0^+)\Delta$$

and for sufficiently small Δ

$$r(\Delta) = r(0) + r'(0^+)\Delta + \gamma(\Delta),$$

where $r'(0^+) = \sup_{|||\mathbf{a}||| \leq 1} |\mathbf{d}^H G^{-1}\mathbf{a}|$ and $\gamma(\Delta) = O(\Delta^2)$.

The result above holds not only for the specific problem of recovering a signal from its samples, but also in a general situation of approximating a linear functional from information given by (nonexact) inner products.

In the case of signal recovery, the matrix G and the vector $\mathbf{d}$ take the form $G = 2\Omega\mathcal{M}$ and $\mathbf{d} = 2\Omega\mathbf{g}$, respectively, where

$$\mathcal{M} = (\mathrm{sinc}(\Omega(t_j - t_k)))_{j,k=1}^n$$

and $\mathbf{g} = [\mathrm{sinc}(\Omega(t_1 - t_0)), \mathrm{sinc}(\Omega(t_2 - t_0)), \ldots, \mathrm{sinc}(\Omega(t_n - t_0))]^T$. Here sinc stands for the *sinus cardinalis* function, i.e.,

$$\operatorname{sinc}(x) = \begin{cases} \dfrac{\sin(x)}{x} & \text{if } x \neq 0, \\ 1 & \text{if } x = 0. \end{cases}$$

Assuming that $|||\cdot|||$ is the pth norm $||\cdot||_p$ in $\mathbf{C}^n$,

$$||\mathbf{a}||_p = \begin{cases} (\sum_{k=1}^{n} |a_k|^p)^{1/p} & \text{if } 1 \leq p < +\infty, \\ \max_{1 \leq k \leq n} |a_k| & \text{if } p = +\infty, \end{cases}$$

the asymptotic formula in Corollary 3.1 takes the form

$$r(\Delta) = r(0) + ||\mathcal{M}^{-1}\mathbf{g}||_q \, \Delta + O(\Delta^2), \quad \Delta \to 0^+,$$

where $1/p + 1/q = 1$.

In summation, for signal recovery the radius of information (the minimal error of an algorithm) behaves like a linear function of data perturbations with coefficients dependent on $t_0, t_1, ..., t_n$.

Signal recovery is an example of a problem for which the dependence on data errors has been revealed. Results concerning this interesting question for other problems and related topics can be found.(12,13,14)

REFERENCES

1. G. Wahba, SIAM (1990).
2. M. Milanese and A. Vicino, in: *Bounding Approaches to System Identification* (Milanese *et al.*, eds.) Plenum Press, New York, Chap. 2 (1996).
3. S. M. Markov and E. D. Popowa, in: *Bounding Approaches to System Identification* (Milanese *et al.*, eds.) Plenum Press, New York, Chap. 9 (1996).
4. G. Belforte and T. T. Tay, in: *Bounding Approaches to System Identification* (Milanese *et al.*, eds.) Plenum Press, New York, Chap. 6 (1996).
5. G. Favier and L. V. Arruda, in: *Bounding Approaches to System Identification* (Milanese *et al.*, eds.) Plenum Press, New York, Chap. 4 (1996).
6. A. J. Helmicki, C.A. Jacobson, and C.N. Nett, *IEEE Trans. Autom. Control* **36**, 1163 (1991).
7. B. Z. Kacewicz and L. Plaskota, *Math. Comp.* **59**, 503 (1992).
8. B. Z. Kacewicz and L. Plaskota, *Numer. Funct. Anal. Optim.* **11**, 511–529 (1990).
9. B. Z. Kacowicz and M. Kowalski, *Int. J. Adapt. Control Signal Proc.*, in press.
10. J. F. Traub, G. W. Wasilkowski, and H. Woźniakowski, *Information Based Complexity*, Academic Press, New York (1988).
11. C. A. Micchelli and T. J. Rivlin, in: *Optimal Estimation in Approximation Theory* (C. A. Micchelli and T. J. Rivlin, eds.) Plenum Press, New York (1977).
12. A. G. Marchuk and K.Y. Osipenko, *Math. Notes* **17**, 207 (1975).
13. A. Melkman and C. A. Micchelli, *SIAM J. Numer. Anal.* **16**, 87 (1979).
14. D. Lee, T. Pavlidis, and G. W. Wasilkowski, *J. Complexity* **3**, 359 (1987).

4

Review and Comparison of Ellipsoidal Bounding Algorithms

G. Favier and L. V. R. Arruda

ABSTRACT

This chapter is concerned with the problem of robust system identification when no statistical information is available on the noise, but only a bound on its instantaneous values is known. First, various ellipsoidal outer bounding (EOB) algorithms are presented in a unified way. Then, two types of projection algorithms are described, and their link with the EOB algorithms is established. After that, the EOB algorithms are interpreted as robust identification algorithms with a dead zone. The performance of these algorithms is compared through computer simulations where the influence of the choice of the *a priori* error bound is more particularly studied.

4.1. INTRODUCTION

In practice, the identification of a parametric model from measured signals must include both the estimation of the model parameters and an evaluation of the estimated parameter uncertainty. This parametric uncertainty is particularly useful for robust controller design. With the probabilistic approach, the exact distribution

G. Favier • Laboratoire I3S, CNRS URA-1376-Sophia Antipolis, Université de Nice, 06560 Valbonne, France. L.V. R. Arruda • Universidade Estadual de Campinas/FEE/DCA—Cidade Universitaria "Zeferino Vaz," Campinas (SP), Brazil.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

of the estimated parameters can be determined if the statistical description of the input signal and disturbances acting on the system to be identified is known. In real applications, such knowledge is often difficult to formulate. An alternative and certainly more realistic approach to the identification problem is the so-called unknown but bounded error (*UBBE*) approach, which was introduced by Witsenhausen(1) and Schweppe(2) in the context of state estimation, and used by Fogel and Huang(3) for system identification. With this approach, the error that includes the measurement noise and the modeling error is assumed to be unknown but bounded, and the error bounds are assumed to be known. This approach allows us to determine a membership set for the model parameters, the elements of which are compatible with the measurements, the assumed model structure and the *a priori* error bounds.

In the case of regression models which are linear in their parameters, the exact membership set is a polytope, the size of which decreases as the number of measurements increases. Several methods have been recently proposed in the literature for recursively determining the polytope which is characterized by means of its vertices, its edges or its faces.(4–6) The main drawback of these methods is their computational burden when the measurement number increases, implying simultaneous increase of the number of vertices, and therefore of edges and faces, of the polytope. To circumvent this problem, a solution consists in approximating the exact polytope by a region in the parametric space, having a simpler shape such as an ellipsoid or an orthotope (i.e., an hyperrectangle the edges of which are parallel to the co-ordinate axes).

In the case of orthotopic bounding, most of the proposed algorithms(7–12) have the drawback of being non-recursive and time-consuming when the number k of measurements is large, as they must solve $2n$ linear programming problems with n variables and $2k$ constraints, where n is the dimension of the unknown parameter vector. However, new algorithms have recently been provided for recursively determining an orthotopic-outer-bounding approximation of the parameter membership set.(13–15)

In the case of ellipsoidal bounding, various algorithms have been derived by means of a geometrical approach combined with the minimization of a criterion directly linked to the size of the ellipsoid,(3,16,17) or by means of convergence considerations.(18–19) With the *UBBE* approach, robust identification methods can also be obtained from the constrained minimization of different quadratic criteria.(19–21) The resulting algorithms are called "projection algorithms with dead zone," which means that they are stopped when the prediction error becomes smaller than the *a priori* error bound.(22) The main advantage of the ellipsoidal-bounding algorithms is their simplicity due to their recursive formulation. However, they often provide a loose approximation to the exact polytopic region. An improvement in terms of reduction of the ellipsoid size can be achieved by processing all the data several times. The reprocessing takes the final ellipsoid delivered at the (i–1)th iteration as the initial ellipsoid of the ith iteration.(16)

It is suggested that the data be preprocessed by using an ellipsoidal-bounding algorithm to discard some of the constraints, before applying an orthotopic or polytopic-bounding algorithm.(16,23,24) This data preprocessing reduces the computational load of the orthotopic or polytopic-bounding algorithms.

The purpose of the present chapter is first to give a unified presentation of the main *EOB* algorithms, then to show the equivalence between these *EOB* algorithms and the robust identification algorithms with dead zone, and finally to compare the performance of these algorithms through computer simulations. This chapter is organized as follows. Section 4.2 states the parameter estimation problem with the *UBBE* formalism. Section 4.3 shows how various *EOB* algorithms can be derived in a unified way.(25,26) In section 4.4, projection algorithms for robust estimation are presented in the bounded noise case, and their link with the *EOB* algorithms is established. Then, in section 4.5, *EOB* algorithms are interpreted as robust identification algorithms with a dead zone.(22,25,26) In section 4.6, a comparison of the performance of these algorithms is carried out through computer simulations where the influence of the choice of the *a priori* error bound is more particularly studied. Finally, section 4.7 concludes this chapter.

4.2. THE UBBE APPROACH AND MEMBERSHIP SET ESTIMATION

Consider the single-input/single-output linear regression model

$$y_t = \varphi_t^T\theta^* + \omega_{t,} \quad \varphi_t, \theta^* \in \Re^n \tag{4.1}$$

with

$$|\omega_t| \leq \delta_t, \quad \delta_t \geq 0, \ t \geq 0 \tag{4.2}$$

where φ_t and θ^* are the regression and the true parameter vectors respectively, and ω_t is the bounded noise term including the measurement noise, the modeling inaccuracy and the computer round-off errors; the error bound δ_t is assumed to be known *a priori*.

All the parameters θ that are consistent with the model structure (4.1), the *a priori* error bounds (4.2) and the measurements $\{y_t, t \in [1,k]\}$ belong to the so-called parameter membership set,(3) defined as:

$$S(k) = \{\theta \mid y_t - \delta_t \leq \varphi_t^T\theta \leq y_t + \delta_t, \quad t \in [1,k]\} \tag{4.3}$$

$S(k)$ is also called feasible parameter set,(6) parameter uncertainty set,(27) or likelihood set.(28) This set can be viewed as the region of the parametric space that is delimited by k pairs of parallel hyperplanes $H_1(t)$ and $H_2(t)$, $t \in [1,k]$, such that:

$$H_1(t) = \{\theta \mid \varphi_t^T\theta = y_t + \delta_t\} \tag{4.4}$$

$$H_2(t) = \{\theta \mid \varphi_t^T\theta = y_t - \delta_t\} \tag{4.5}$$

Each hyperplane $H_i(t)$, $i = 1, 2$, divides the parametric space into two halfspaces $H_i^+(t)$ and $H_i^-(t)$ defined as:

$$H_1^+(t) = \{\theta \mid \varphi_t^T\theta \le y_t + \delta_t\} \tag{4.6}$$

$$H_1^-(t) = \{\theta \mid \varphi_t^T\theta > y_t + \delta_t\} \tag{4.7}$$

$$H_2^+(t) = \{\theta \mid \varphi_t^T\theta \ge y_t - \delta_t\} \tag{4.8}$$

$$H_2^-(t) = \{\theta \mid \varphi_t^T\theta < y_t - \delta_t\} \tag{4.9}$$

Then, the set $S(k)$ is given by:

$$S(k) = \bigcap_{t=1}^{k} H^+(t) \tag{4.10}$$

$$\text{where: } H^+(t) = H_1^+(t) \cap H_2^+(t) \tag{4.11}$$

The set $S(k)$ is a monotone non-increasing sequence of sets having a polytopic shape, as shown in Fig. 4.1 for $n = 2$ and $k = 4$. Any parameter vector θ belonging to the set $S(k)$ is a valid estimation of θ^*. In practice, the center of $S(k)$ (in some geometrical sense) is chosen as the estimate of θ^*.

Although its size is decreasing, this polytopic region generally becomes very complicated to determine when the number of measurements increases, due to the

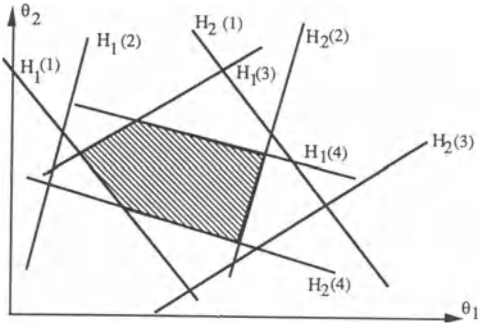


FIGURE 4.1. The parameter uncertainty set $S(k)$.

augmentation of the number of its vertices. An easier solution consists in approximating the convex polytopes $S(k)$ by simpler shaped regions like ellipsoids,(3,16–19) orthotopes,(7–15) or parallelotopes.(29) The corresponding algorithms are respectively called ellipsoidal, orthotopic and parallelotopic outer bounding (*EOB*, *OOB* and *POB*) algorithms. It is also possible to construct ellipsoidal inner bounds(30,31) or orthotopic inner bounds.(11)

Let $M(k)$ be such an outer bounding approximation of $S(k)$:

$$M(k) \supset S(k) \tag{4.12}$$

This region $M(k)$ can be recursively constructed so that

$$M(k) \supset M(k-1) \cap H^{+}(k) \tag{4.13}$$

or, in using (4.11):

$$M(k) \supset M(k-1) \cap H_1^{+}(k) \cap H_2^{+}(k) \tag{4.14}$$

By induction and using Eqs. (4.10, 4.11, and 4.14), it is easy to verify that, if the initial region $M(0)$ is chosen sufficiently large to contain $S(k_0)$, where $k_0 \geq n$ is the first value of k for which n vectors in $\{\phi_t, t \in [1,k_0]\}$ are linearly independent, then the set $M(k)$ satisfies the relation of inclusion (4.12) for all the values of $k \geq k_0$.

In the next section, we show how various *EOB* algorithms can be derived in a unified way.

4.3. A UNIFIED PRESENTATION OF *EOB* ALGORITHMS

In the *EOB* approach, as introduced by Fogel and Huang,(3) the solution consists in recursively determining a sequence of ellipsoids $E(k)$ which enclose $S(k)$. Let us define an initial ellipsoid $E(0)$ by:

$$E(0) = \left\{ \theta \in \Re^n \mid (\theta - \theta_0)^T P_0^{-1} (\theta - \theta_0) \leq \sigma_0^2, P_0 \sigma_0^2 = \frac{1}{\varepsilon} I_n \right\} \tag{4.15}$$

where ε is a sufficiently small number such that $E(0)$ contains $S(k)$ for all $k \geq 0$, σ_0^2 and P_0 represent the *a priori* knowledge about the system to be identified. The ellipsoidal bound $E(k)$ must be chosen in such a way that it contains as tightly as possible the intersection of $E(k–1)$ and $H^{+}(k)$. This ellipsoid $E(k)$ can be defined by means of the following inequality:(25,26)

$$E(k) = \{\theta \mid \alpha_k(\theta - \theta_{k-1})^T P_{k-1}^{-1}(\theta - \theta_{k-1}) + \beta_k(y_k - \varphi_k^T\theta)^2 \leq \alpha_k\sigma_{k-1}^2 + \beta_k\delta_k^2\} \tag{4.16}$$

where $\alpha_k \in \,]0,1]$ is a forgetting factor which weights the old information, while $\beta_k \in [0,1]$ is a selecting factor which weights the new information.

TABLE 4.1. Basic *EOB* Equations

$$\nu_k = y_k - \varphi_k^T\theta_{k-1}$$

$$G_k = \varphi_k^T P_{k-1}\varphi_k$$

$$P_k = \frac{1}{\alpha_k}\left[P_{k-1} - \frac{\beta_k P_{k-1}\,\varphi_k\,\varphi_k^T P_{k-1}}{\alpha_k + \beta_k G_k}\right]$$

$$\theta_k = \theta_{k-1} + \beta_k P_k\varphi_k\nu_k$$

$$\sigma_k^2 = \alpha_k\sigma_{k-1}^2 + \beta_k\delta_k^2 - \frac{\alpha_k\beta_k\nu_k^2}{\alpha_k + \beta_k G_k}$$

The estimated parameters are taken as the coordinates of the center θ_k of the ellipsoid $E(k)$. By simple algebra manipulations, Eq. (4.16) of $E(k)$ can be rewritten as:

$$E(k) = \{\theta \mid (\theta - \theta_k)^T P_k^{-1}(\theta - \theta_k) \le \sigma_k^2\} \tag{4.17}$$

where the ellipsoid parameters θ_k, P_k and σ_k^2 are calculated through the equations of Table 4.1.

In the following, two groups of *EOB* algorithms are derived in a unified way, using the basic *EOB* equations given in Table 4.1 and making different choices for the free parameters α_k and β_k.

Methods minimizing the geometrical size of the ellipsoid E(k): the free parameters α_k and β_k are calculated by minimizing a scalar measure of the size of the matrix P_k, which reflects the geometrical size of the ellipsoid $E(k)$.

Methods based on convergence arguments: the choice of α_k and β_k results from the minimization of a cost function under constraints. This choice is not optimal with respect to the reduction of the geometrical size of the ellipsoid $E(k)$, but this reduction is ensured. We call these methods "degenerate" minimal-volume algorithms.(25)

Before describing these two families of *EOB* algorithms, we give the conditions for the existence of a solution, in terms of the intersection of $E(k-1)$ and $H^+(k)$, and for the redundancy of a measurement. Moreover, we give the formulae for calculating the parameter uncertainty intervals associated with the ellipsoid $E(k)$.

Existence condition: When the intersection $E(k-1) \cap H^+(k)$ is empty, i.e., the ellipsoid $E(k-1)$ is entirely located in one of the two halfspaces $H_i^-(k)$, $i = 1$ or 2, respectively defined by (4.7) and (4.9), the following condition is satisfied:

$$|\nu_k| > \delta_k + \sigma_{k-1} G_k^{1/2} \tag{4.18}$$

In this case, the corresponding measurement must be discarded if the trouble is caused by an outlier (and therefore a bad choice of the *a priori* error bound δ_k), or the algorithm must be reinitialized if the occurrence of (4.18) is due to a bad choice of the initial ellipsoid $E(0)$ or to time variation of the model parameters. This last situation can be detected in incorporating a fault-detection test to the identification algorithm,[32] which leads to an adaptive *EOB* algorithm.

Redundancy condition: Another important particular case occurs when the ellipsoid $E(k-1)$ is entirely located in $H^{+}(k)$, which corresponds to the following condition:[33]

$$|\nu_k| < \delta_k - \sigma_{k-1} G_k^{1/2} \tag{4.19}$$

In this case, the measurement is redundant and can be discarded, so condition (4.19) defines a dead zone for the *EOB* algorithms (see section 4.5 for an interpretation of *EOB* algorithms as robust identification algorithms with dead zone).

Parameter uncertainty intervals: With each coordinate $\theta_k(j), j \in [1,n]$, of the center of the ellipsoid $E(k)$, we can associate the uncertainty interval $[\theta_{k,min}(j), \theta_{k,max}(j)]$, where $\theta_{k,min}(j)$ and $\theta_{k,max}(j)$ are the minimum and maximum values taken by the coordinate $\theta(j)$ of any point of the ellipsoid $E(k)$:

$$\theta_{k,\min}(j) = \underset{\theta \in E(k)}{\text{Min}}\ \theta(j) \quad \text{and} \quad \theta_{k,\max}(j) = \underset{\theta \in E(k)}{\text{Max}}\ \theta(j) \tag{4.20}$$

The bounds of these uncertainty intervals can be calculated by means of the following formulae:[34]

$$\theta_{k,\min}(j) = \theta_k(j) - \sigma_{k-1}\,[P_k(j,j)]^{1/2} \tag{4.21}$$

$$\theta_{k,\max}(j) = \theta_k(j) + \sigma_{k-1}\,[P_k(j,j)]^{1/2} \tag{4.22}$$

where $P_k(j,j)$ is the element (j,j) of the P_k matrix which defines the ellipsoid $E(k)$, as in (4.17).

The two families of *EOB* algorithms are now described.

4.3.1. Methods Minimizing the Geometrical Size of the Ellipsoid E(k)

The methods of this group use the basic *EOB* equations in Table 4.1, with $\alpha_k = 1/\sigma_{k-1}^2$ and $\beta_k = \lambda_k/\delta_k^2$. The variable λ_k is obtained from the minimization of a measure that reflects the geometrical size of the ellipsoid $E(k)$. λ_k is time varying and data dependent. The choice $\lambda_k = 0$ is possible and it means that the information contained in the new observation is redundant. In this case, the ellipsoid stays

TABLE 4.2. Computation of λ_k for Minimal Volume Algorithm

λ_k is the solution of

$$a_1\lambda_k^2 + a_2\lambda_k + a_3 = 0$$

with
$$a_1 = (n-1)\sigma_{k-1}^4 G_k^2$$
$$a_2 = ((2n-1)\delta_k^2 - \sigma_{k-1}^2 G_k + \nu_k^2)\,\sigma_{k-1}^2 G_k$$
$$a_3 = (n(\delta_k^2 - \nu_k^2) - \sigma_{k-1}^2 G_k)\delta_k^2$$

The optimal value of λ_k is then given by:

$$\lambda_k = \begin{cases} 0 & \text{if } a_3 \geq 0 \\ \lambda_k^* & \text{otherwise} \end{cases}$$

with $\lambda_k^* = (-a_2 + (a_2^2 - 4a_1a_3)^{1/2})/2a_1$

unchanged $[E(k-1) \cap H^+(k) = E(k-1)]$ and the parameter estimates are not updated.

Two measures defined on $\Re$ are considered by Fogel and Huang[3] for this minimization: $\mu_v(k)$ = determinant $(\sigma_k^2 P_k)$ and $\mu_T(k)$ = trace $(\sigma_k^2 P_k)$ which are proportional to the volume and to the sum of squares of the semi-axes of $E(k)$ respectively. The corresponding algorithms are called the minimal-volume algorithm and minimal-trace algorithm respectively. The computation of the corresponding optimal values of λ_k are summarized in Tables 4.2 and 4.3.

TABLE 4.3. Computation of λ_k for Minimal Trace Algorithm

λ_k is the solution of

$$\lambda_k^3 + b_1\lambda_k^2 + b_2\lambda_k + b_3 = 0 \qquad \text{(A)}$$

with:

$$b_1 = 3\delta_k^2/(\sigma_{k-1}^2 G_k)$$

$$b_2 = \{\delta_k^2 G_k[\mu_T(k-1)(\delta_k^2 - \nu_k^2) - \sigma_{k-1}^4\gamma_k] + 2\delta_k^2[\delta_k^2 G_k \mu_T(k-1) - \sigma_{k-1}^2\gamma_k(\delta_k^2 - \nu_k^2)]\}/\Psi_k$$

$$b_3 = \delta_k^4[(\delta_k^2 - \nu_k^2)\,\mu_T(k-1) - \sigma_{k-1}^4\gamma_k]/(\sigma_{k-1}^2\Psi_k)$$

$$\gamma_k = \varphi_k^T P_{k-1}^2\varphi_k \text{ and } \Psi_k = \sigma_{k-1}^4 G_k^2[G_k\mu_T(k-1) - \sigma_{k-1}^2\gamma_k]$$

The optimal value of λ_k is then given by:

$$\lambda_k = \begin{cases} 0 & \text{if } b_3 \geq 0 \\ \lambda_k^* & \text{otherwise} \end{cases}$$

where λ_k^* is the positive real root of equation (A).

When only one of the two constraint hyperplanes $H_i(k)$ defined in (4.4 and 4.5) intersects $E(k-1)$, the minimal-volume algorithm does not give the minimal-volume ellipsoid containing $E(k-1) \cap H^{+}(k)$. A smaller volume ellipsoid can be obtained by replacing the non-intersecting hyperplane by a parallel one tangent to $E(k-1)$. The corresponding algorithm is called "Improved Minimal-Volume Algorithm."(16) For this algorithm, the variables δ_k and ν_k in Table 4.1 are replaced by:(33,35)

$$\delta_k' = \begin{cases} \frac{1}{2}(\delta_k + \nu_k + \sigma_{k-1}G_k^{1/2}) & \text{if} -\sigma_{k-1}G_k^{1/2} - \delta_k < \nu_k < -\left|\sigma_{k-1}G_k^{1/2} - \delta_k\right| \\ \frac{1}{2}(\delta_k - \nu_k + \sigma_{k-1}G_k^{1/2}) & \text{if } \left|\sigma_{k-1}G_k^{1/2} - \delta_k\right| < \nu_k < \sigma_{k-1}G_k^{1/2} + \delta_k \\ \delta_k \text{ otherwise} & \end{cases} \tag{4.23}$$

$$\nu'_k = \begin{cases} \frac{1}{2}(\nu_k + \delta_k - \sigma_{k-1}G_k^{1/2}) & \text{if} -\sigma_{k-1}G_k^{1/2} - \delta_k < \nu_k < -\left|\sigma_{k-1}G_k^{1/2} - \delta_k\right| \\ \frac{1}{2}(\nu_k - \delta_k + \sigma_{k-1}G_k^{1/2}) & \text{if } \left|\sigma_{k-1}G_k^{1/2} - \delta_k\right| < \nu_k < \sigma_{k-1}G_k^{1/2} + \delta_k \\ \nu_k \text{ otherwise} & \end{cases} \tag{4.24}$$

and λ_k is calculated as in Table 4.2.

Applying (4.23) is equivalent to reducing the noise upper bound, as we have:

$$0 < \delta_k' \leq \delta_k \tag{4.25}$$

4.3.2. "Degenerate" Minimal-Volume Algorithms

The methods belonging to this group result from a geometrical approach which consists in ensuring that the ellipsoid size is reduced at each time instant k. The ellipsoid $E(k)$ is then determined so that σ_k^2 is minimized, or the sequence $\{\sigma_k^2\}$ is non-increasing, i.e., $\sigma_k^2 \leq \sigma_{k-1}^2$.

The first "degenerate" minimal-volume algorithm, proposed by Dasgupta and Huang(18) is obtained by choosing $\alpha_k = 1 - \lambda_k$ and $\beta_k = \lambda_k$, with λ_k the solution of the following constrained minimization problem:

$$\underset{\lambda_k}{\text{Min}}\ \sigma_k^2 \tag{4.26}$$

$$0 \leq \lambda_k \leq \upsilon < 1$$

where the design variable $\upsilon \in]0,1[$ is introduced to ensure that the matrix P_k will be bounded. The corresponding computation of λ_k is described in Table 4.4.

TABLE 4.4. Computation of λ_k for Dasgupta and Huang's Algorithm

$$\lambda_k = \begin{cases} 0 & \text{if } \gamma_k \geq 1 \\ \lambda_k^* & \text{otherwise} \end{cases}$$

$$\lambda_k^* = \min(\upsilon, \xi_k) \text{ with } 0 < \upsilon < 1$$

$$\gamma_k = \frac{\delta_k^2 - \sigma_{k-1}^2}{\nu_k^2}$$

$$\xi_k = \begin{cases} \upsilon & \text{if } \nu_k = 0 \\ \dfrac{1-\gamma_k}{2} & \text{if } G_k = 1 \\ \upsilon & \text{if } \gamma_k(G_k - 1) + 1 \leq 0 \\ \dfrac{1}{(1-G_k)}\left(1 - \left(\dfrac{G_k}{\gamma_k(G_k-1)+1}\right)^{1/2}\right) & \text{if } \gamma_k(G_k - 1) + 1 > 0 \end{cases}$$

A second "degenerate" minimal-volume algorithm can be derived by considering:(25)

$$\alpha_k = \lambda \quad \text{and} \quad \beta_k = \lambda_k \tag{4.27}$$

where λ is a constant forgetting factor and λ_k is a positive weighting factor to be determined so that it minimizes the criterion (4.26) without the constraint $0 \leq \lambda_k \leq \upsilon$.

Substituting α_k and β_k by their values (4.27) in the equations of θ_k, P_k and σ_k^2 given in Table 4.1, we get:

$$\theta_k = \theta_{k-1} + \frac{\lambda_k P_{k-1}\varphi_k \nu_k}{\lambda + \lambda_k G_k} \tag{4.28}$$

$$P_k = \frac{1}{\lambda}\left[P_{k-1} - \frac{\lambda_k P_{k-1}\varphi_k\varphi_k^T P_{k-1}}{\lambda + \lambda_k G_k}\right] \tag{4.29}$$

$$\sigma_k^2 = \lambda\sigma_{k-1}^2 + \lambda_k\delta_k^2 - \lambda\lambda_k\nu_k^2/(\lambda + \lambda_k G_k) \tag{4.30}$$

The value λ_k^* of λ_k that minimizes σ_k^2 is obtained from $\partial\sigma_k^2/\partial\lambda_k = 0$, which gives:

$$\delta_k^2 - \lambda^2\nu_k^2/(\lambda + \lambda_k G_k)^2 = 0 \tag{4.31}$$

$$\lambda_k = \begin{cases} 0 & \text{if } |\nu_k| \le \delta_k \text{ or } G_k = 0 \\ \lambda_k^* & \text{otherwise} \end{cases} \tag{4.32}$$

$$\lambda_k^* = \frac{\lambda}{G_k}\left(\frac{|\nu_k|}{\delta_k} - 1\right) \tag{4.33}$$

Moreover, for this value of λ_k we have:

$$\left.\frac{\partial^2 \sigma_k^2}{\partial \lambda_k^2}\right|_{\lambda_k=\lambda_k^*} = \frac{2\lambda^2 \nu_k^2 G_k}{(\lambda + \lambda_k^* G_k)^3} > 0 \tag{4.34}$$

which implies that λ_k^* corresponds to minimization of the criterion (4.26).

Finally, a third "degenerate" minimal-volume algorithm can be obtained by using the *EOB* equations in Table 4.1, combined with convergence arguments. This algorithm, proposed in Ref. (19) corresponds to the choice $\alpha_k = 1$ and $\beta_k = \lambda_k$. In contrast to the two previous algorithms in this group, the value of λ_k is not obtained from the minimization of σ_k^2. Indeed, λ_k is determined so that the sequence $\{\sigma_k^2\}$ is non-increasing, i.e., $\sigma_k^2 \le \sigma_{k-1}^2$, while satisfying the constraints $0 \le \lambda_k \le \upsilon \le 1$. Then we get:

$$\lambda_k = \begin{cases} 0 & \text{if } |\nu_k| \le \delta_k^* \\ \lambda_k^* & \text{otherwise} \end{cases} \tag{4.35}$$

with:

$$\lambda_k^* = \frac{\upsilon}{1 + G_k}\left(1 - \frac{\delta_k^*}{|\nu_k|}\right) \text{where } \upsilon \in [0,1] \text{ and } \delta_k^* = (1+\upsilon)^{1/2}\,\delta_k. \tag{4.36}$$

The variable σ_k^2 can be considered as an upper limit for the following quadratic non-negative function:

$$V_k = (\theta_k - \theta^*)^T P_k^{-1} (\theta_k - \theta^*) \tag{4.37}$$

From the analysis of this quadratic function, it is possible to demonstrate the following convergence properties for the degenerate minimal-volume algorithms,[(35,36)] i.e., for the different choices of α_k and β_k:

$$\text{(i)}\ \lim_{k\to\infty}\left\{(\prod_{i=1}^{k} \alpha_i)^{-1} |f(\beta_k)|\right\} = 0 \tag{4.38}$$

$$\text{(ii)}\ \|\tilde{\theta}_k\|^2 \le \rho\, \|\tilde{\theta}_0\|^2 \tag{4.39}$$

where

$$\tilde{\theta}_k = \theta^* - \theta_k, \ \rho = \frac{\mu_{\max}(P_0^{-1})}{\mu_{\min}(P_0^{-1})}, f(\beta_k) = \beta_k \left(\delta_k^2 - \frac{\alpha_k v_k^2}{\alpha_k + \beta_k G_k} \right)$$

and $\mu_{\min}[.]$ ($\mu_{\max}[.]$) = smallest (largest) eigenvalue of [.].

Moreover, assuming that the following persistent excitation condition of the input signal is satisfied:

$$mI_n \geq \sum_{i=k}^{k+N} \beta_i \varphi_i \varphi_i^T \geq MI_n \qquad \forall k \leq k_o \tag{4.40}$$

where m, M and N ($N \geq n$) are positive scalars and k_o is the convergence time of the algorithm, then there exists a positive scalar η such that

$$P_k^{-1} \geq \eta I_n > 0 \tag{4.41}$$

and the following properties hold for the degenerate minimal-volume algorithms:

$$\text{(iii)} \ \lim_{k \to \infty} \|\theta_k - \theta_{k-1}\|^2 = 0 \tag{4.42}$$

$$\text{(iv)} \ \|\tilde{\theta}_k\|^2 \leq \frac{\varepsilon \sigma_o^2}{\eta} \left(\prod_{i=1}^{k} \alpha_i \right) \|\tilde{\theta}_0\|^2 \quad \forall \, k \geq N+1 \tag{4.43}$$

$$\text{(v)} \ \|\tilde{\theta}_k\|^2 \leq \frac{1}{\eta} \sigma_k^2 \tag{4.44}$$

From the properties (iii)–(iv), the degenerate *EOB* algorithms are exponentially convergent. Further, property (v) provides an upper limit ($1/\eta \, \sigma_\infty^2$) for the steady-state estimation error.

4.4. PROJECTION ALGORITHMS WITH DEAD ZONE

The introduction of a dead zone into the estimator equations is a classical procedure to face bounded perturbations. The idea is to stop updating the parameters when the prediction error becomes smaller than some threshold. This threshold defines what is called a dead zone for the estimator. In this section, we present two robust projection algorithms with dead zone. They are obtained from the minimization of two different criteria with a constraint on the a posteriori prediction error.

4.4.1. Robust Projection Algorithm Based on Constrained One-Step-Ahead Criterion Minimization

The estimation problem is considered in terms of minimization of the following cost function[(19)]:

$$J_1\,(\theta_k) = \frac{1}{2}\,\|\theta_k - \theta_{k-1}\|^2 \tag{4.45}$$

under the constraint

$$(e_k^{\mathrm{o}} - i_k^{\mathrm{o}}\delta_k)i_k = 0 \tag{4.46}$$

where e_k^{o} is the residual

$$e_k^{\mathrm{o}} = y_k - \varphi_k^T\theta_k \tag{4.47}$$

and

$$i_k = \begin{cases} 0 & \text{if } |\nu_k| \le \delta_k \\ 1 & \text{otherwise} \end{cases}, \quad i_k^{\mathrm{o}} = \mathrm{sign}(\nu_k) \tag{4.48}$$

The constraint (4.46) means that the residual e_k^{o} is forced to be equal to $\pm\delta_k$ when the absolute value of the prediction error is greater than the noise upper bound δ_k. The constrained minimization problem (4.45) and (4.46) is solved by introducing a Lagrange multiplier λ_k for the constraint, so that the cost function to be minimized becomes:

$$J_1{}'(\theta_k,\lambda_k) = \frac{1}{2}\,\|\theta_k - \theta_{k-1}\|^2 + \lambda_k(e_k^{\mathrm{o}} - i_k^{\mathrm{o}}\delta_k)i_k \tag{4.49}$$

Writing the necessary conditions for a minimum ($\partial J_1' \,/\partial\theta_k = 0$; $\partial J_1' \,/\partial\lambda_k = 0$), we get the following equations:

$$\theta_k = \theta_{k-1} + \lambda_k\varphi_k i_k \tag{4.50}$$

with

$$\lambda_k = \begin{cases} \lambda_k^* & \text{if } i_k = 1 \text{ and } |\varphi_k| \ne 0 \\ 0 & \text{otherwise} \end{cases} \tag{4.51}$$

where

$$\lambda_k^* = \frac{\nu_k - i_k^{\mathrm{o}}\delta_k}{\varphi_k^T\varphi_k} \tag{4.52}$$

The introduction of the dead zone due to the presence of the factor i_k in the correction term (4.50) allows us to turn off the algorithm when the prediction error becomes smaller than the noise bound δ_k. In Table 4.5, the robust projection algorithm (4.50)–(4.52) is compared to the projection algorithm with dead zone introduced by Goodwin and Sin.[(37)] In this last algorithm, the residual e_k^{o} is forced

TABLE 4.5. Projection Algorithms

Algorithms	Constraints	a_k
	Normalized Projection Algorithms	
	$\theta_k = \theta_{k-1} + a_k \frac{\varphi_k}{c + \varphi_k^T \varphi_k} \nu_k$	
Projection Alg. with dead zone[37]	$[y_k - \varphi_k^T \theta_k] i_k = 0$ $i_k = \begin{cases} 0 & \text{if } \lvert\nu_k\rvert \le 2\sup\lvert\omega_k\rvert \\ 1 & \text{otherwise} \end{cases}$	i_k
Robust Projection Algorithm[19]	$[y_k - \varphi_k^T \theta_k - \delta_k \operatorname{sign}(\nu_k)] i_k = 0$ $i_k = \begin{cases} 0 & \text{if } \lvert\nu_k\rvert \le \delta_k \\ 1 & \text{otherwise} \end{cases}$	$\frac{\lvert\nu_k\rvert - \delta_k}{\lvert\nu_k\rvert} i_k$
	Orthogonalized Projection Algorithms	
	$\theta_k = \theta_{k-1} + \frac{a_k P_{k-1} \varphi_k \nu_k}{c + G_k}$	
	$P_k = \frac{1}{\lambda} [P_{k-1} - \frac{P_{k-1} \varphi_k \varphi_k^T P_{k-1}}{c + G_k} a_k]$	
Orthogonalized Projection Algorithm with dead zone ($\lambda = 1$)[37]	$[y_k - \varphi_k^T \theta_k] i_k = 0$ $i_k = \begin{cases} 0 & \text{if } \lvert\nu_k\rvert \le 2\sup\lvert\omega_k\rvert \\ 1 & \text{otherwise} \end{cases}$	i_k
Robust Orthogonalized Projection Algorithm[21]	$[y_k - \varphi_k^T \theta_k - \delta_k \operatorname{sign}(\nu_k)] i_k = 0$ $i_k = \begin{cases} 0 & \text{if } \lvert\nu_k\rvert \le \delta_k \\ 1 & \text{otherwise} \end{cases}$	$\frac{\lvert\nu_k\rvert - \delta_k}{\lvert\nu_k\rvert} i_k$

to zero, while in the first one this error is forced to be equal to $\pm\delta_k$, depending on the sign of the *a priori* prediction error ν_k.

4.4.2. Robust Orthogonalized Projection Algorithm Based on Constrained Least-Squares Criterion Minimization

In this case, the criterion to be minimized is[20,21]

$$J_2(\theta_k) = \sum_{t=1}^{k} \lambda_{k,t} (y_t - \varphi_t^T \theta_k)^2 \tag{4.53}$$

with

$$\lambda_{k,t} = \lambda^{k-t}\lambda_t \tag{4.54}$$

where $\lambda \in \,]0,1]$ is a forgetting factor fixed by the user and $\lambda_t \geq 0$ is a data-dependent weighting factor which is determined in such a way that the constraint (4.46) is satisfied.

Minimization of (4.53) with respect to θ_k, leads to the well known weighted RLS equations:

$$\theta_k = \theta_{k-1} + \frac{\lambda_k P_{k-1}\varphi_k \nu_k}{\lambda + \lambda_k G_k} \tag{4.55}$$

$$P_k = \frac{1}{\lambda}\left[P_{k-1} - \frac{\lambda_k P_{k-1}\varphi_k\varphi_k^T P_{k-1}}{\lambda + \lambda_k G_k}\right] \tag{4.56}$$

where ν_k and G_k are defined in Table 4.1.

Substituting (4.55) for θ_k into the constraint (4.46), when $i_k = 1$ ($|\nu_k| > \delta_k$), gives:

$$e_k^{\mathrm{o}} = \frac{\lambda \nu_k}{\lambda + \lambda_k G_k} = i_k^{\mathrm{o}}\delta_k \tag{4.57}$$

which leads to the following optimal value of the weighting factor λ_k:

$$\lambda_k = \begin{cases} \lambda_k^* & \text{if } |G_k| \neq 0 \text{ and } |\nu_k| > \delta_k \\ 0 & \text{otherwise} \end{cases} \tag{4.58}$$

where:

$$\lambda_k^* = \frac{\lambda}{G_k}\left(\frac{|\nu_k|}{\delta_k} - 1\right) \tag{4.59}$$

Replacing λ_k by its expression (4.58) and (4.59) in (4.55) and (4.56) yields the equations of the robust orthogonalized projection algorithm (also called modified exponentially weighted recursive least squares (*EWRLS*) algorithm), which are given in Table 4.5. This algorithm is compared to the orthogonalized projection algorithm with dead zone.[37] As for the projection algorithms described in section 4.4.1, the orthogonalized and robust orthogonalized projection algorithms are such that the residual e_k^{o} is forced to zero and $\pm\delta_k$ respectively when $|\nu_k|$ is greater than δ_k.

When α_k and β_k are chosen as in (4.27), equations of Table 4.1 for computing P_k and θ_k and expression (4.33) of λ_k^* are identical to (4.55, 4.56 and 4.59). So, we demonstrate the equivalence of the second degenerate minimal-volume algorithm (section 4.3.2) and the robust orthogonalized projection algorithm, which at the same time allows us to give a new geometrical interpretation of the modified *EWRLS* algorithm proposed in Refs. 20 and 21. Moreover, comparing equations of Tables 4.1 and 4.5, one can easily verify that the two families of robust estimation algorithms (*EOB* algorithms and orthogonalized projection algorithms) can be written with the same equations, so that it is possible to interpret the *EOB* algorithms as robust identification algorithms with dead zone. The mathematical equivalence between these two families of algorithms is shown in the next paragraph.

For the projection algorithms described in Table 4.5, division by zero (when $\varphi_k = 0$ or $G_k = 0$) is avoided by adding a small positive constant c to the denominator of the equations which compute θ_k and P_k.

4.5. INTERPRETATION OF *EOB* ALGORITHMS AS ROBUST IDENTIFICATION ALGORITHMS WITH DEAD ZONE

It is now well known that the robustness of classical estimation algorithms, in the sense of a noise sensitivity reduction, can be enhanced by introducing a dead zone in the parameter update equation. Such parameter update law modifications are very useful in the context of adaptive control.[38–40] In this case, the controller adaptation is turned off when the prediction error is smaller than some threshold Δ.

The *EWRLS* algorithm with dead zone, resulting from the minimization of the quadratic criterion (4.53), with:

$$\lambda_{k,t} = \lambda_t \prod_{i=t+1}^{k} \mu_i \tag{4.60}$$

is given in Table 4.6.

TABLE 4.6. *EWRLS* Algorithm with Dead Zone ($f(\nu_k) = \nu_k$)

$$\nu_k = y_k - \varphi_k^T \theta_{k-1}$$

$$G_k = \varphi_k^T P_{k-1} \varphi_k$$

$$P_k = \frac{1}{\mu_k}\left[P_{k-1} - \frac{\lambda_k P_{k-1} \varphi_k \varphi_k^T P_{k-1}}{\mu_k + \lambda_k G_k} i_k\right]$$

$$\theta_k = \theta_{k-1} + \lambda_k P_k \varphi_k f(\nu_k) i_k$$

$$i_k = \begin{cases} 0 & \text{if } \nu_k^2 \le \Delta^2 \\ 1 & \text{otherwise} \end{cases}$$

When the dead zone condition is satisfied ($i_k = 0$), the estimator is then frozen ($\theta_k = \theta_{k-1}, P_k = P_{k-1}$), which requires $\mu_k = 1$ in the computation formula for the matrix P_k.

The most critical point in the application of such identification schemes is the selection of the dead zone Δ. The equations of Table 4.6 look like those of Table 4.1. With the correspondence $\lambda_k = \beta_k$ and $\mu_k = \alpha_k$, it is possible to rewrite the *EOB* algorithms like the *EWRLS* algorithm with dead zone. Indeed, the choice $\beta_k = 0$ in the equation for computing θ_k in Table 4.1, is equivalent to using a dead zone. This dead zone can be explicitly introduced into the basic *EOB* equations of Table 4.1 by rewriting the estimate equation as

$$\theta_k = \theta_{k-1} + \beta_k P_k \varphi_k \nu_k i_k$$

where

$$i_k = \begin{cases} 0 & \text{if "the ellipsoid cannot be reduced"} \\ 1 & \text{otherwise} \end{cases}$$

The condition "the ellipsoid cannot be reduced" defines the dead zone. For the considered *EOB* algorithms, this dead zone is explicitly given by the conditions $a_3 \geq 0$ (Table 4.2), $b_3 \geq 0$ (Table 4.3), $\gamma_k \geq 1$ (Table 4.4), $|\nu_k| \leq \delta_k$ (Eq. (4.32)) and $|\nu_k| \leq \delta_k^*$ (Eq. (4.35)).

TABLE 4.7. Interpretation of *EOB* Algorithms as Robust Identification Algorithms with Dead Zone

Methods	α_k	β_k	$f(\nu_k)$	Δ^2
Minimal-volume algorithm	$1/\sigma_{k-1}^2$	λ_k^*/δ_k^2	ν_k	$\delta_k^2 - \sigma_{k-1}^2\, G_k/n$
Minimal-trace algorithm	$1/\sigma_{k-1}^2$	λ_k^*/δ_k^2	ν_k	$\delta_k^2 - \dfrac{\sigma_{k-1}^2\, \gamma_k}{Tr\, P_{k-1}}$
Improved minimal volume algorithm	$1/\sigma_{k-1}^2$	$\lambda_k^*/\delta_k'^2$	ν_k'	$\delta_k'^2 - \sigma_{k-1}^2\, G_k/n$
First degenerate minimal volume algorithm	$1 - \lambda_k^*$	λ_k^*	ν_k	$\delta_k^2 - \sigma_{k-1}^2$
Second degenerate minimal volume algorithm	λ	λ_k^*	ν_k	δ_k^2
Third degenerate minimal volume algorithm	1	λ_k^*	ν_k	$(1 + \upsilon)\delta_k^2$
RLS algorithm with dead zone	1	1	ν_k	$\delta_k^2(1 + G_k)$

In Table 4.7, we give the dead zone, the forgetting factor α_k, the weighting factor β_k and the function $f(\nu_k)$ for each *EOB* algorithm rewritten with the equations of Table 4.6.

In conclusion, the *EWRLS* and *EOB* algorithms can be written with the same equations; only the dead zone definition and the forgetting and weighting factors are different.

In Table 4.7, the expression of λ_k^* is respectively given in Tables 4.2, 4.3, and 4.4, or Eqs. (4.33) and (4.36), depending on the algorithm which is considered. Moreover, for the improved minimal-volume algorithm, the quantities ν_k' and δ_k' are defined in (4.23 and 4.24).

We have to notice that the dead zone associated with the first family of *EOB* algorithms,(3,16) and with the first degenerate minimal-volume algorithm,(18) results from the computation of the quantities σ_k^2 and P_k, while it depends only on the *a priori* error bound δ_k and the design parameter υ for the second and third degenerate minimal-volume algorithms. As already mentioned, the second degenerate minimal-volume algorithm is identical with the robust orthogonalized projection algorithm described in Table 4.5.

It is easy to prove that the dead zones defined in Table 4.7 have a width larger than the dead zone (4.19), i.e., are more conservative in terms of parameter update. Indeed, the conditions for the existence of a solution to the different optimization problems corresponding to the minimization of the criteria $\mu_v(k)$, $\mu_T(k)$, or σ_k^2 are more restrictive than the condition (4.19) for $E(k-1)$ and $H^+(k)$ to intersect, which means that the existence of such an intersection doesn't always imply the existence of a new ellipsoid containing this intersection and obtained by minimizing one of the above criteria.

In the next section, the performances of the considered *EOB* algorithms are compared on simulated examples.

4.6. SIMULATION RESULTS

In this section, simulation results show the influence of the *a priori* error bound δ_k on the performance of the *EOB* algorithms; they compare the behavior of the *EOB* algorithms in presence of sudden disturbances and time variations of the model parameters.

First consider the following *ARX* model, with constant parameters:

$$y_t = 0.8\, y_{t-1} + 1.2\, u_{t-1} + \omega_t$$

The input signal u_t is a square wave with a period $T = 10$ and an amplitude $A = 1$, and the unmeasurable disturbance ω_t corresponds to an independent random sequence with a uniform distribution in $[-1, 1]$. The initial conditions for the algorithms are $\theta_0 = 0$ and $P_0\, \sigma_0^2 = 100\, I_2$.

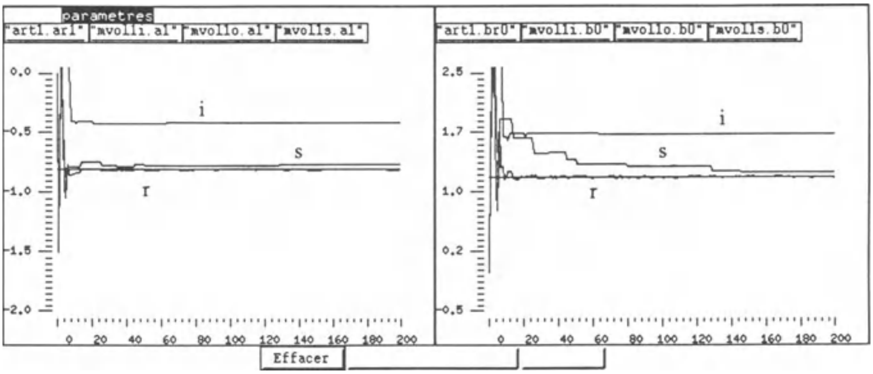


FIGURE 4.2. Estimated parameters $\hat{a}_1$ and $\hat{b}_0$ (minimal volume algorithm).

Two points are studied in these simulations: comparison of the six *EOB* algorithms when the *a priori* error bound is correctly chosen ($\delta_k = \delta_k^r = 1$), and when this bound is underestimated ($\delta_k = \delta_k^i = 0.5$) or overestimated ($\delta_k = \delta_k^s = 3$); and performance comparison for the *EOB* and *EWRLS* algorithms.

Figs. 4.2 through 4.7 show the estimated parameters corresponding to underestimated (plots *i*), good (plots *r*) and overestimated (plots *s*) bounds. Fig. 4.8 shows the results obtained with the *EWRLS* algorithm with dead zone.

Table 4.8 contains the update rate for each algorithm, and for the three values of δ_k which allows comparison of the performance of the *EOB* algorithms, in terms of ability to discard redundant measurements. The update rate is calculated as:

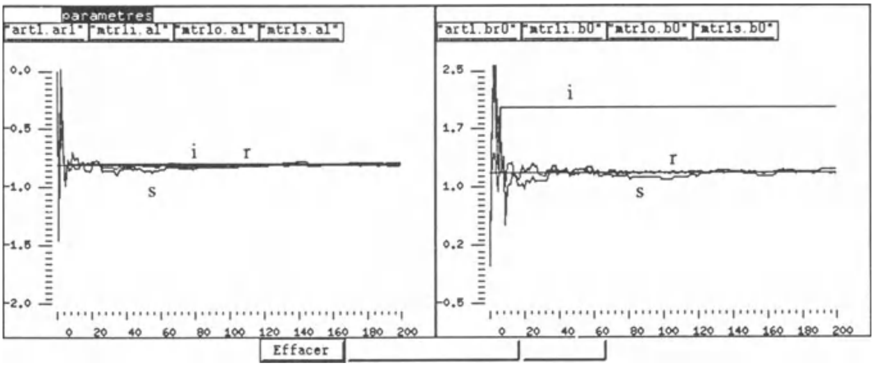


FIGURE 4.3. Estimated parameters $\hat{a}_1$ and $\hat{b}_0$ (minimal trace algorithm).

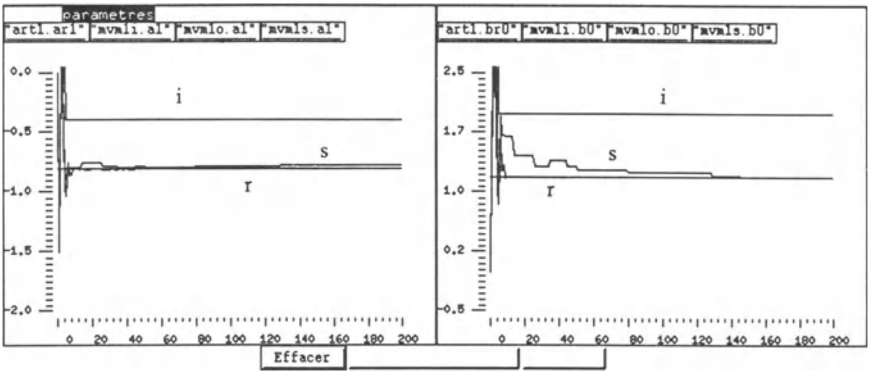


FIGURE 4.4. Estimated parameters $\hat{a}_1$ and $\hat{b}_0$ (improved minimal volume algorithm).

$$R = \frac{\text{number of effective updates}}{\text{total number of iterations}}$$

From these simulation results, we conclude that if the *a priori* error bound is correctly chosen, all the *EOB* algorithms converge to the true parameters, the smallest update rate being obtained with the algorithms belonging to the second group.

By analyzing Figs. 4.2 to 4.4, one can conclude, for the first family of *EOB* algorithms, that convergence is not too much affected by an overestimation of the *a priori* error bound. An underestimation of this *a priori* error bound leads to biased estimated parameters. In this case, Condition (4.18) for the non-existence of a solution is rapidly satisfied. That results in a low update rate (see Table 4.8), which is not indicative of a measurement redundancy but of an inconsistency between the measurements and the assumed error bound δ^i. The convergence is faster, and

TABLE 4.8. Update Rate for *EOB* and *EWRLS* Algorithms

Algorithms	$\delta^i = 0.5$ (%)	$\delta^r = 1.0$ (%)	$\delta^s = 3.0$ (%)
Minimal volume	4	78.5	7
Minimal trace	3	79.5	35
Improved minimal volume	3	61.5	8.5
First degenerate minimal volume	68.5	78.5	1.5
Second degenerate minimal volume	78.5	44.5	1.5
Third degenerate minimal volume	98	18	3.5
EWRLS with dead zone	99.5	56.5	1

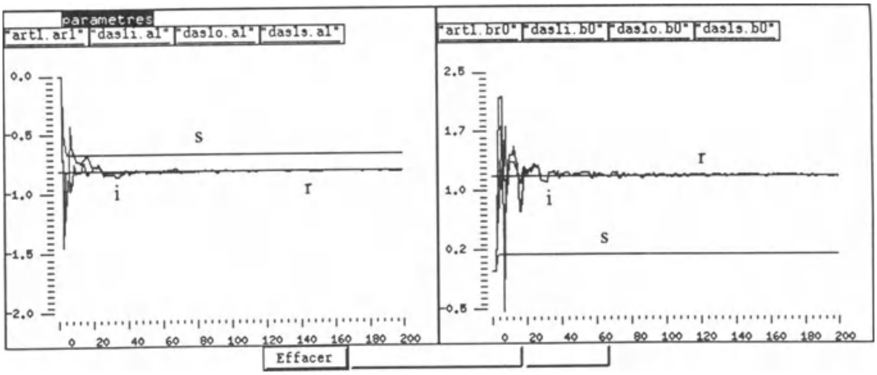


FIGURE 4.5. Estimated parameters $\hat{a}_1$ and $\hat{b}_0$ (first degenerate minimal volume algorithm).

therefore the update rate is smaller, with the improved minimal volume algorithm than with the minimal volume algorithm. When the noise upper bound is well chosen ($\delta_k = \delta_k^r$), the update rates obtained with the minimal volume, minimal trace, and first degenerate minimal-volume algorithms are very similar. As is common, the parameter connected to the input term is not as well estimated as the autoregressive parameter.

By analyzing Figs. 4.5 to 4.7, one can conclude, for the second family of *EOB* algorithms, that the behavior of the degenerate minimal-volume algorithms is the opposite of that of the other *EOB* algorithms. The estimated parameters are biased when the error bound is overestimated, and they are fluctuating around the true values of the parameters when an underestimated bound is used. Moreover, the bias is all the larger as the error bound is more overestimated. This difference in the

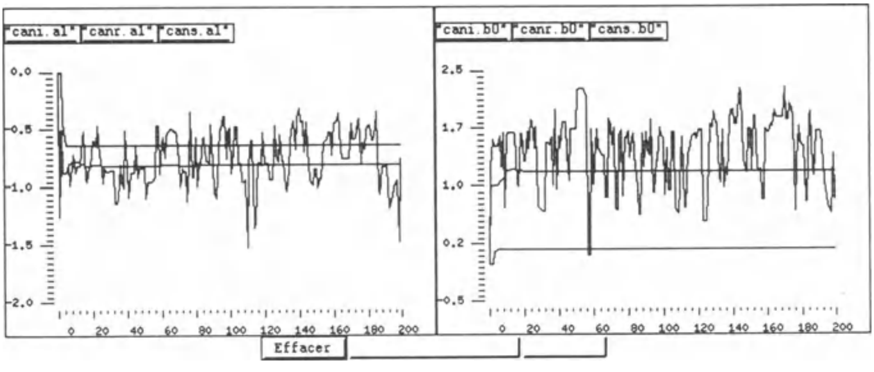


FIGURE 4.6. Estimated parameters $\hat{a}_1$ and $\hat{b}_0$ (second degenerate minimal volume algorithm).

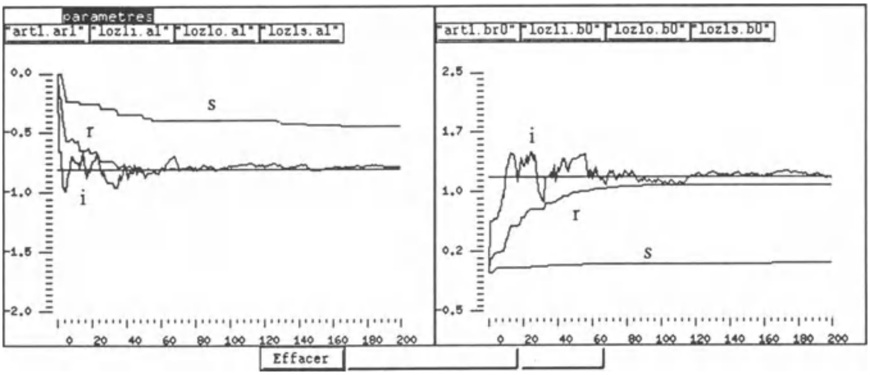


FIGURE 4.7. Estimated parameters $\hat{a}_1$ and $\hat{b}_0$ (third degenerate minimal volume algorithm).

behavior essentially results from the dead zone which is a function of the ellipsoid size in the case of the *EOB* algorithms of the first family. It depends only on the error bound for the second degenerate minimal-volume algorithm, and also the design parameter υ for the third degenerate algorithm. In the case of this last algorithm, the introduction of the factor $(1+\upsilon)$ in the dead zone transforms the good bound δ^r into an overestimated value, which explains the biased estimation of the parameters with $\delta = \delta^r$ (Fig. 4.7). In conclusion, these simulations show that the convergence of the *EOB* algorithms is strongly dependent on the choice of the *a priori* error bound δ_k.

A second simulated example illustrates the behavior of the *EOB* algorithms faced with an abrupt parameter change and an additive sudden disturbance ε_t:

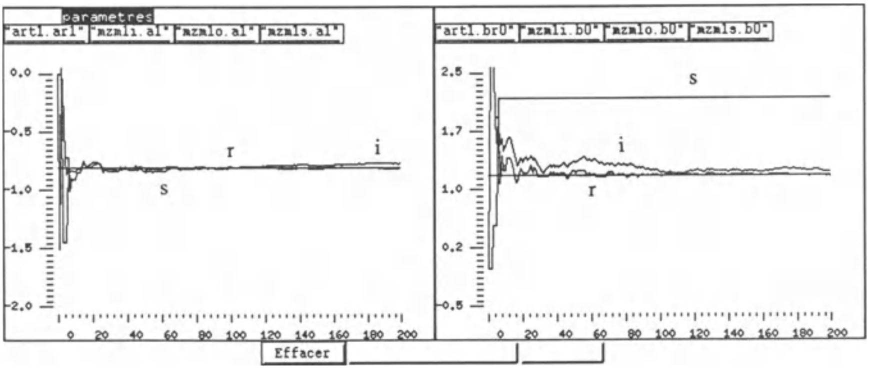


FIGURE 4.8. Estimated parameters $\hat{a}_1$ and $\hat{b}_0$ (*EWRLS* algorithm with dead zone).

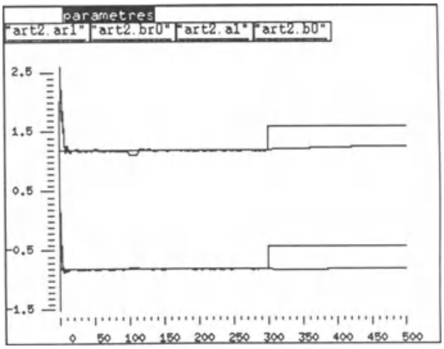


FIGURE 4.9. Estimated parameters $\hat{a}_1$ and $\hat{b}_0$ (improved minimal value algorithm).

$$y_t = a_t^1 \, y_{t-1} + b_t^0 \, u_{t-1} + \omega_t + \varepsilon_t$$

with:

$$a_t^1 = \begin{cases} 0.8 \; t \in [0,300] \\ 0.4 \; t \in [301,500] \end{cases} \qquad b_t^0 = \begin{cases} 1.2 \; t \in [0,300] \\ 1.6 \; t \in [301,500] \end{cases}$$

$$\varepsilon_t \to \mathcal{N}(0,1), \qquad t \in [100,110]$$

The input signal u_t and the non-measurable disturbance ω_t are the same as for the previous simulated example. The additive disturbance ε_t is a zero mean Gaussian noise, with variance equal to one. The initial conditions for the identification algorithms are $\theta_0 = 0$ and $P_0\sigma_0^2 = 100I_2$. The *a priori* error bound is chosen equal to $\delta_k = 1$.

From the previous simulations, the behavior is nearly the same for all the *EOB* algorithms belonging to a same group, so that for this second simulated example, only one algorithm of each family is compared: the improved minimal-volume algorithm (Fig. 4.9) for the first group and the second degenerate minimal-volume

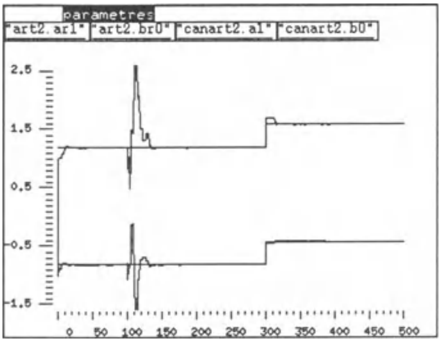


FIGURE 4.10. Estimated parameters $\hat{a}_1$ and $\hat{b}_0$ (second degenerate minimal volume algorithm).

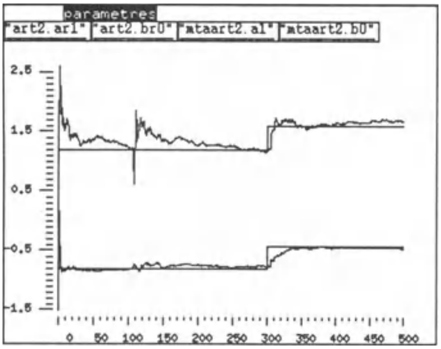


FIGURE 4.11. Estimated parameters $\hat{a}_1$ and $\hat{b}_0$ (adaptive trace algorithm).

algorithm (Fig. 4.10) for the second group. These algorithms are also compared with the adaptive trace algorithm (Fig. 4.11).[32]

The performances of all the *EOB* algorithms are degraded in presence of a sudden additive disturbance. For the degenerate minimal-volume algorithms, this additive disturbance leads to strongly fluctuating estimated parameters.

By examining the plots shown on Fig. 4.9, the *EOB* algorithms of the first family haven't a tracking capability. Indeed, after an abrupt parameter change, the new model parameters generally don't belong to the last ellipsoid which was determined before the parameter change. Then Condition (4.18) is satisfied and the algorithm is stopped, which explains the bias of the estimated parameters. In this case, as suggested in section 4.3, a solution would consist in combining a fault detection test with the *EOB* algorithm and arbitrarily increasing the ellipsoid size when a parameter change is detected.

On the contrary the second family of *EOB* algorithms is naturally adaptive due to the presence of the forgetting factor which permanently ensures a sufficient size of the ellipsoid.

4.7. CONCLUSIONS

In this chapter, various *EOB* algorithms for identifying systems characterized by bounded modeling errors have been presented in a unified way. These algorithms have been reformulated as robust identification algorithms with dead zone, the main differences between them consisting in the computation of the dead zone and the choice of the weighting factors. *EOB* algorithms have thus been proved equivalent to the *EWRLS* algorithm with dead zone.

A comparative analysis of the performances of these *EOB* algorithms have been carried out by means of simulated examples. The influence of the choice of the *a priori* noise upper bound and the tracking capability of these algorithms in presence of an abrupt parameter change have been studied.

REFERENCES

1. H. S. Witsenhausen, *IEEE Trans. Autom. Control* **AC-13**, 556 (1968).
2. F. C. Schweppe, *IEEE Trans. Autom. Control* **AC-13**, 22 (1968).
3. E. Fogel and Y. F. Huang, *Automatica* **18**, 229 (1982).
4. E. Walter and H. Piet-Lahanier, in: *Proceedings of the IEEE Conference on Decision and Control*, Los Angeles, CA, pp. 1921–1922 (1987).
5. V. Broman and M. J. Shensa, *Math. Comput. Simul.* **32**, 469 (1990).
6. S. H. Mo and J. P. Norton, *Math. Comput. Simul.* **32**, 481 (1990).
7. G. Belforte and M. Milanese, *Proceedings of the IFAC/IFORS Symposium on Identification and System Parameter Estimation*, Darmstadt, Germany, pp. 381–385 (1979).
8. M. Milanese and G. Belforte, *IEEE Trans. Autom. Control* **AC-27**, 408 (1982).
9. G. Belforte, B. Bona, and S. Frediani, *Proceedings of the IEEE Conference on Decision and Control*, Las Vegas, NV, pp. 1554–1559 (1984).
10. T. Clement and S. Gentil, *Math. Comput. Simul.* **30**, 257 (1988).
11. A. Vicino and M. Milanese, *Proceedings of the IEEE Conference on Decision and Control*, Tampa, FL, pp. 2576–2580 (1989).
12. M. Milanese and A. Vicino, in: *Proceedings of the IFAC/IFORS Symposium on Identification and System Parameter Estimation*, Budapest, Hungary, pp. 859–867 (1991).
13. G. Belforte and T. T. Tay, in: *Proceedings of the IEEE Conference on Decision and Control*, Honolulu, HI, pp. 3546–3551 (1990).
14. H. Messaoud, G. Favier, and R. Santos Mendes, in: *Proceedings of the IFAC Symposium on Adaptive Systems in Control and Signal Processing*, Grenoble, France, pp. 41–46 (1991).
15. H. Messaoud and G. Favier, in: *Proceedings of the 14th GRETSI Symposium*, Juan-Les-Pins, France, pp. 225–228 (1993).
16. G. Belforte and B. Bona, in: *Proceedings of the IFAC/IFORS Symposium on Identification and System Parameter Estimation*, York, UK, pp. 1507–1512 (1985).
17. L. Pronzato, E. Walter, and H. Piet-Lahanier, in: *Proceedings of the IEEE Conference on Decision and Control*, Tampa, FL, pp. 1952–1955 (1989).
18. S. Dasgupta and Y. F. Huang, in: *Proceedings of the IEEE Conference on Decision and Control*, Ft. Lauderdale, FL, pp. 1067–1071 (1985); Also in *IEEE Trans. Inf. Theory* **IT-33**, 383 (1987).
19. R. Lozano-Leal and R. Ortega, *Automatica* **23**, 247 (1987).
20. C. Canudas de Wit and J. Carrillo, in: *Proceedings of the IFAC Symposium on Identification and System Parameter Estimation*, Beijing, China, pp. 1205–1210 (1988).
21. C. Canudas de Wit and J. Carrillo, *Automatica* **26**, 599 (1990).
22. L. V. R. Arruda, G. Favier, and W. Amaral, *Proceedings of the IEEE Conference on Acoustics, Speech, and Signal Processing*, Albuquerque, NM, USA, pp. 2967–2970 (1990).
23. S. H. Mo and J. P. Norton, *IEEE Proc.* **135**, 127 (1988).
24. G. Belforte, B. Bona, and V. Cerone, *Automatica* **26**, 887 (1990).
25. L. V. R. Arruda and G. Favier, in: *Proceedings of the IFAC/IFORS Symposium on Identification and System Parameter Estimation*, Budapest, Hungary, pp. 1027–1032 (1991).
26. L. V. R. Arruda, G. Favier, and W. Amaral, *Proceedings of the 1st European Control Conference*, Grenoble, France, pp. 1194–1199 (1991).
27. G. Belforte, B. Bona, and V. Cerone, *Measurement* **5**, 167 (1987).
28. E. Walter and H. Piet-Lahanier, *Math. Comput. Simul.* **32**, 449 (1990).
29. A. Vicino and G. Zappa, to appear in *Proceedings of the Workshop on The Modelling of Uncertainty in Control Systems*, Springer-Verlag (1993).
30. J. P. Norton, *Proceedings of the IFAC Symposium on Identification and System Parameter Estimation*, York, UK, pp. 1197–1202 (1985); *Automatica* **23**, 497 (1987).

31. L. Pronzato and E. Walter, *Proceedings of the European Control Conference*, Groningen, The Netherlands, pp. 258–263 (1993).
32. G. Favier, *APII* **22**, 27 (1988).
33. H. Messaoud, *Identification et Commande Robustes: Etude et Comparaison d'Algorithmes*, Doctoral Thesis, University of Tunis, Tunisie (1993).
34. H. Obali, *Etude Comparative d'Algorithmes d'Identification Robuste*, Doctoral Thesis, University of Marrakech, Morocco (1993).
35. L. V. R. Arruda, *Etude d'Algorithmes d'Estimation Robuste et Dèveloppement d'un Système à Système à Base de Connaissance pour l'Identification*, Doctoral Thesis, University of Nice-Sophia Antipolis, Nice, France (1992).
36. L. V. R. Arruda and G. Favier, in: *GRETSI Symposium*, Juan les Pins, France (1991).
37. G. C. Goodwin and K. S. Sin, *Adaptive Filtering, Prediction and Control*, Prentice-Hall Englewood Cliffs, NJ (1984).
38. B. B. Peterson and K. S. Narendra, *IEEE Trans. Autom. Control*, **AC-27**, 1161 (1982).
39. C. Samson, *Automatica* **19**, 81 (1983).
40. L. Praly, *Proceedings of the 3rd Yale Workshop on Applications of Adaptive Systems Theory*, pp. 224–226 (1983).

5

The Dead Zone in System Identification

K. Forsman and L. Ljung

ABSTRACT

A prediction error method for parameter estimation in a dynamical system is studied.

$$\hat{\vartheta} = \arg\min_{\vartheta} \lim_{N\to\infty} \frac{1}{N}\sum_{t=1}^{N} \mathrm{E}l(\varepsilon(t,\vartheta))$$

where ε are the prediction errors of a linear regression. A quadratic norm l is zero within an interval $[-c, c]$. This kind of a dead zone (*DZ*) criterion is very common in robust adaptive control. The following problems are treated in this chapter:

- When is the *DZ* estimate inconsistent, and what is the set of parameters which minimizes the criterion in the case of inconsistency?
- What happens to the variance of the estimate as the *DZ* is introduced?
- Does the *DZ* give a better estimate than least squares (LS) when there are unmodeled deterministic disturbances present?
- What are the relations between identification with a dead zone criterion and so called set membership identification?

K. Forsman • ABB Corporate Research, Ideon, S-223 70 Lund, Sweden. L. Ljung • Department of Electrical Engineering, Linköping University, S-581 83 Linköping, Sweden.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

5.1. INTRODUCTION

Consider a prediction error method for parameter estimation in a dynamical system:

$$\hat{\vartheta} = \arg\min_{\vartheta} V(\vartheta) \tag{5.1}$$

Here V is a *DZ* criterion:

$$V(\vartheta) = \lim_{N\to\infty} \frac{1}{N}\sum_{t=1}^{N} \mathrm{E}l[\varepsilon(t,\vartheta)] \tag{5.2}$$

where ε are the prediction errors of a linear regression:

$$y(t) = \varphi^{\mathrm{T}}(t)\vartheta^{0} + e(t) \tag{5.3}$$

$$\Rightarrow \varepsilon(t,\vartheta) = y(t) - \hat{y}(t,\vartheta) = \varphi^{\mathrm{T}}(t)\tilde{\vartheta} + e(t) \tag{5.4}$$

where

$$\tilde{\vartheta} := \vartheta^{0} - \vartheta \tag{5.5}$$

Furthermore, l is a quadratic norm which is zero within the interval $[-c, c]$:

$$l(x) = \begin{cases} \frac{1}{2}(x-c)^2, & x > c \\ 0, & |x| \le c \\ \frac{1}{2}(x+c)^2, & x < -c \end{cases} \tag{5.6}$$

or, more compactly, $l(x) = \frac{1}{2}[\max(c,|x|) - c]^2$. A typical l is displayed in Fig. 5.1.

The use of such a *DZ* is widespread in adaptive control and system identification. It appears in adaptive regulators used in the industry, but also have theoretically interesting properties, for example in the stability theory of robust adaptive regulators. Still, many properties of the *DZ* estimate seem to have been neglected to some degree. The following questions are addressed in this chapter:

- Under what circumstances will the *DZ* estimate be inconsistent?
- If the estimate is inconsistent, how much can it deviate from the true value of ϑ?
- How does the *DZ* affect the variance of the estimate?
- What is the effect on the *LS* estimate when a *DZ* is introduced, supposing there are unmodeled deterministic disturbances present?

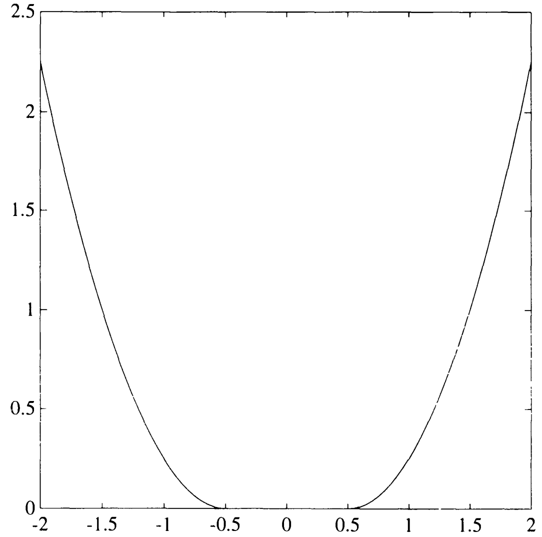


FIGURE 5.1. The norm of the criterion ($c = 0.5$).

- What are the relations between identification with a *DZ* criterion and so called set membership identification?

Notational convention: the notation $\bar{\mathrm{E}}$ is defined by:

$$\bar{\mathrm{E}}x = \lim_{N\to\infty} \frac{1}{N}\sum_{t=1}^{N} \mathrm{E}x(t) \tag{5.7}$$

5.2. CONSISTENCY

In some cases the *DZ* estimate of the parameters will be inconsistent. For instance, if the noise is bounded, the norm of the residuals may be zero at all time instants for any parameter values within a set (non-singleton) in the parameter space. Thus all members of this set are indistinguishable, and only the true parameter is inside the set is known. The following two theorems say what is intuitively clear, namely that the estimate will be inconsistent if and only if the noise is bounded and the *DZ* is too wide.

THEOREM 5.1. Let f be the *PDF* of the noise e in Eq. (5.3) and suppose that f is even. If f is not identically zero outside the interval $[-c, c]$ then V has a global minimum in ϑ_0. Shorter:

$$\mathrm{supp}(f) \not\subset [-c, c] \Rightarrow \hat{\vartheta} = \vartheta_0$$

PROOF. Define $v(b) := E\,l(e+b)$, i.e.,

$$v(b) = \int_{-\infty}^{\infty} l(b+x) f(x)dx$$

It is sufficient to show that $v''(b) > 0$ and that $v'(0) = 0$. Straightforward computations show that $v'(0) = 0$. Furthermore

$$v''(b) = \int_{-\infty}^{-c-b} f(x)dx + \int_{c-b}^{\infty} f(x)dx \tag{5.8}$$

which is nonzero for all b iff $\mathrm{supp}(f) \not\subset [-c, c]$.

In the case of a density that is not even we get a messy implicit expression for the minimizing bias is obtained.

The residuals are defined by Eqs. (5.4 and 5.5). Now, if $|\varepsilon(t,\vartheta)| \leq c$ for all t, $l[\varepsilon(t,\vartheta)] \equiv 0$, which means that a sufficient condition for the estimate to be inconsistent is

$$|\varphi^{\mathrm{T}}(t)\tilde{\vartheta}| + B_1 \leq c \tag{5.9}$$

where B_1 is a bound for the noise e. The following theorem explains exactly the parameter estimates in this case:

THEOREM 5.2. Suppose that the noise and the input are bounded:

$$\forall t:\ |e(t)| \leq B_1 < c,\ |u(t)| \leq B_2$$

and that the system is stable. Then

$$\hat{\vartheta} \supset \{\vartheta | c \geq \alpha(\vartheta)B_3 + \beta(\vartheta)B_2 + B_1\} \neq \{\vartheta_0\}$$

where $B_3 = \|y(t)\|_\infty$, $\alpha(\vartheta) = \|\tilde{\vartheta}_A\|_1$ and $\beta(\vartheta) = \|\tilde{\vartheta}_B\|_1$ are defined via Eq. (5.10).

PROOF. Since it is assumed that the input is bounded we can obtain an estimate of the first term in Eq. (5.9) in the following way: Partition the $\tilde{\vartheta}$ vector in elements corresponding to y and elements corresponding to u:

$$\tilde{\vartheta} = \begin{pmatrix} \tilde{\vartheta}_A \\ \hline \tilde{\vartheta}_B \end{pmatrix},\ \tilde{\vartheta}_A = \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{pmatrix},\ \tilde{\vartheta}_B = \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_k \end{pmatrix} \tag{5.10}$$

and the regression vector φ analogously. If the true system is asymptotically stable the output will be bounded: $|y(t)| \leq B_3$. (Of course, B_3 can be expressed in B_1 and

B_2 if we know the true system parameters ϑ^0.) Hölder's inequality gives us an upper bound for the first term of Eq. (5.9):

$$|\varphi_A^{\mathrm{T}}(t)\tilde{\vartheta}_A| \le \| \begin{pmatrix} -y(t-1) \\ \vdots \\ -y(t-k) \end{pmatrix} \|_\infty \|\tilde{\vartheta}_A\|_1 \le B_3 \sum_{j=1}^{n} |\alpha_j| \tag{5.11}$$

If the B-part of $\varphi^{\mathrm{T}}\tilde{\vartheta}$ is estimated in the same way, then

$$|\varphi^{\mathrm{T}}(t)\tilde{\vartheta}| \le \alpha B_3 + \beta B_2, \quad \alpha := \sum_{j=1}^{n} |\alpha_j|, \quad \beta := \sum_{j=1}^{k} |\beta_j|$$

Since

$$|\varepsilon(t, \vartheta)| \le \alpha B_3 + \beta B_2 + B_1$$

a DZ width which is strictly greater than B_1 will give an estimate that is inconsistent. The set of parameter vectors in which the criterion is zero will be a superset of

$$\{\vartheta | c \ge \alpha(\vartheta)B_3 + \beta(\vartheta)B_2 + B_1\} \tag{5.12}$$

□

It is easy to show that this is the best estimate that Hölder's inequality can produce with the information available.

5.3. VARIANCE

What does the asymptotic covariance matrix of the estimation error look like when we use a DZ, and how does it depend on the width of the DZ? The following important fact can be found in Ref. 5.3:

$$\mathrm{Cov}\,\hat{\vartheta}_N \sim \frac{1}{N}\kappa(l)[\bar{E}\psi_0\psi_0^{\mathrm{T}}]^{-1}, \quad \psi_0 = -\frac{d}{d\vartheta}\varepsilon(t,\vartheta)|_{\vartheta=\vartheta_0} \tag{5.13}$$

where

$$\kappa(l) = \frac{\bar{\mathrm{E}}[l'(e)]^2}{[\bar{\mathrm{E}}l''(e)]^2} \tag{5.14}$$

This formula is valid if l is twice continuously differentiable, which is not true for the squared DZ. The trouble caused by the discontinuity in l'' does not appear to be of that serious a kind, though. Assume that it is of no importance. If the time averaging property of $\bar{\mathrm{E}}$ is disregarded, which is the same as assuming that the PDF of the noise does not vary in time, then

$$\kappa(l) = \frac{\mathrm{E}[l'(e)]^2}{[\mathrm{E}l''(e)]^2} \tag{5.15}$$

Here l is the squared DZ of Eq. (5.6), so

$$l'(\varepsilon) = \begin{cases} \varepsilon - c, & \varepsilon > c \\ 0, & |\varepsilon| \leq c \\ \varepsilon + c, & \varepsilon < -c \end{cases} \tag{5.16}$$

and

$$l''(\varepsilon) = \begin{cases} 1, & |\varepsilon| > c \\ 0, & |\varepsilon| < c \\ \alpha, & |\varepsilon| = c \end{cases} \tag{5.17}$$

where α is a subdifferential, $\alpha = [0, 1]$. If confined to even densities, then

$$\mathrm{E}[l'(e)]^2 = 2\int_c^{\infty} (x-c)^2 f(x)dx \tag{5.18}$$

and

$$\mathrm{E}l''(e) = 2\int_c^{\infty} f(x)dx \tag{5.19}$$

Insert Eqs. (5.19 and 5.18) in Eq. (5.14) to get

$$f\text{even} \Rightarrow \kappa(l) = \frac{\int_c^{\infty} (x-c)^2 f(x)dx}{2\left(\int_c^{\infty} f(x)dx\right)^2} \tag{5.20}$$

supposing the integral in the denominator does not vanish. As expected, κ is strongly dependent on f.

Proving that the variance tends to infinity as the width of the deadzone tends to infinity, even if intuitively very clear, is not trivial.

Here is a theorem which covers many important cases. Different estimates of the quotient Eq. (5.20) are made in the different cases.

DEFINITION 5.1. Let f be a continuously differentiable function that f is *asymptotically decreasing if*

$$\exists N\, \forall x > N\colon\ f'(x) < 0$$

THEOREM 5.3 If f is asymptotically decreasing and either

$$\lim_{x\to\infty} \frac{f(x+1)}{f(x)} > 0 \tag{5.21}$$

or

$$\lim_{x\to\infty} \left|\frac{f'(x)}{f(x)}\right| > 0 \text{ and } \lim_{x\to\infty} \frac{f^2(x)}{f(x+1)} = 0 \tag{5.22}$$

then

$$\kappa(l) \to \infty \text{ as } c \to \infty$$

in Eq. (5.20).

PROOF. Given that $\kappa(l) = T_1 + T_2$ where

$$T_1 = \frac{\int_c^{c+1} (x-c)^2 f(x)dx}{\left(\int_c^{\infty} f\right)^2}, \quad T_2 = \frac{\int_{c+1}^{\infty} (x-c)^2 f(x)dx}{\left(\int_c^{\infty} f\right)^2}$$

it suffices to show that either T_1 or T_2 tends to be ∞ since both are positive. Introducing the notations

$$a(c) = \int_c^{c+1} f, \quad b(c) = \int_{c+1}^{\infty} f$$

it is clear that a and b tend to zero as $c \to \infty$.

First, look at T_1. Assume that Eq. (5.22) holds. Since

$$\int_c^{c+1} (x-c)^2 f(x)dx \geq f(c+1) \int_c^{c+1} (x-c)^2 dx + \frac{1}{3} f(c+1) \tag{5.23}$$

T_1 can be lower bounded:

$$T_1 \geq \frac{1}{3} f(c+1)/(a+b)^2 \tag{5.24}$$

But according to Eq. (5.22)

$$\frac{a^2(c)}{f(c+1)} < \frac{f^2(c)}{f(c+1)} \to 0, \; c \to \infty \tag{5.25}$$

and

$$\lim_{c\to\infty} \frac{a(c)b(c)}{f(c+1)} = 0 \tag{5.26}$$

since

$$\lim_{c\to\infty} \frac{f(c+1)}{b(c)} = \lim_{c\to\infty} \left|\frac{f'(c+1)}{f(c+1)}\right| > 0 \tag{5.27}$$

according to l'Hospital's rule. The same argument can be used to show that

$$\lim_{c\to\infty} \frac{b(c)^2}{f(c+1)} = \lim_{c\to\infty} \frac{f(c+1)}{f'(c+1)} \cdot b(c) = 0 \tag{5.28}$$

In conclusion, $T_1 \to \infty$ as $c \to \infty$.

Now study T_2 under the assumption of Eq. (5.21). This gives

$$\int_{c+1}^{\infty} (x-c)^2 f(x)dx \geq \int_{c+1}^{\infty} f(x)dx \tag{5.29}$$

so that

$$T_2 \geq \frac{b}{(a+b)^2} = \frac{1}{a^2/b + 2a + b} \tag{5.30}$$

If it can be shown that a^2/b tends to zero as $c \to \infty$ the proof is complete. This can be achieved by applying l'Hospital's rule and using Eq. (5.21) to get

$$\lim_{c\to\infty} \frac{a^2(c)}{b(c)} = \lim_{c\to\infty} \frac{f(c)-f(c+1)}{f(c+1)} a(c) = \lim_{c\to\infty} \frac{f(c)}{f(c+1)} a(c) = 0 \tag{5.31}$$

In conclusion, $T_2 \to \infty$ as $c \to \infty$. □

The theorem above covers many of the interesting cases, e.g., that of the normal distribution:

$$f(x) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-x^2/2\sigma^2}$$

For this special case straightforward computations show that

$$\kappa(l) = \frac{(\sigma^2 + c^2)(1 - \Phi(\frac{c}{\sigma})) - \frac{c\sigma}{\sqrt{2\pi}} e^{-c^2/2\sigma^2}}{2(1 - \Phi(\frac{c}{\sigma}))^2}$$

where

$$\Phi(x) := \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} e^{-t^2/2}\, dt$$

5.4. DETERMINISTIC DISTURBANCES

In this section the aim is to investigate what happens as a deterministic disturbance is present in the model. One might believe that in this case an *LS* estimate will always improve if a *DZ* is introduced. However, this is not true. Let us call the deterministic disturbance *d*. Then

$$y(t) = \varphi^{\mathrm{T}}(t)\vartheta^{0} + e(t) + d(t) \tag{5.32}$$

and the residuals are

$$\varepsilon(t,\vartheta) = y(t) - \varphi^{\mathrm{T}}(t)\vartheta = \varphi^{\mathrm{T}}(t)\tilde{\vartheta} + d(t) + e(t) \tag{5.33}$$

These formulas are the analogues of Eqs. (5.3) and (5.4). Let g denote the 'deterministic part' of Eq. (5.33):

$$g(t) := \varphi^{\mathrm{T}}(t)\tilde{\vartheta} + d(t)$$

5.4.1. Minimizing the Criterion

It is possible to derive an explicit expression for the criterion of Eq. (5.2) in the case of Eq. (5.32). It turns out that this expression is extensive and non-suggestive. Luckily, the derivative of the criterion is rather easy to compute, as long as f (the *PDF* of the noise) is even. These computations result in the following equation.

THEOREM 5.4. Given

$$\frac{d}{d\vartheta}V(\vartheta) = 0 \Leftrightarrow \bar{E}\varphi[2g(\vartheta) - \int_{c-g(\vartheta)}^{c+g(\vartheta)} F] = 0 \tag{5.34}$$

PROOF. Recall the following formula from elementary calculus[(4)]:

$$\frac{d}{dx}\int_{\varphi(x)}^{\psi(x)} f(x,y)dy = \int_{\varphi(x)}^{\psi(x)} fx(x,y)dy + f(x,\psi(x))\psi'(x) - f(x,\varphi(x))\varphi'(x) \tag{5.35}$$

Now, for each sample

$$\mathrm{E}l(\varepsilon) = \int_{-\infty}^{-c-g} (x + g + c)^2 f(x)dx + \int_{c-g}^{\infty}(x + g - c)^2 f(x)dx \tag{5.36}$$

where g is a function of ϑ. We want to compute the derivative of Eq. (5.36) w.r.t. ϑ. In order to do this use Eq. (5.35) on each of the terms on the right hand side:

$$A = \frac{d}{d\vartheta} \int_{-\infty}^{-c-g(\vartheta)} (x + g(\vartheta) + c)^2 f(x)dx = 2g'[(g + c)F(-c - g) + \int_{-\infty}^{-c-g} xf(x)dx]$$

$$B = \frac{d}{d\vartheta} \int_{c-g(\vartheta)}^{\infty} (x + g(\vartheta) + c)^2 f(x)dx = 2g'[(g - c)(1 - F(c - g)) + \int_{c-g}^{\infty} xf(x)dx]$$

where $'$ denotes differentiation w.r.t. ϑ. So $A + B = 2g'\Gamma$ where

$$\Gamma = (g + c)F(-c - g) + (g - c)(1 - F(c - g)) - \int_{-c-g}^{c-g} xf(x)dx \tag{5.37}$$

By partial integration,

$$\int_{-c-g}^{c-g} xf(x)dx = [xF(x)]_{-c-g}^{c-g} - \int_{-c-g}^{c-g} F(x)dx$$

$$= (c - g)F(c - g) + (c + g)F(-c - g) - \int_{-c-g}^{c-g} F(x)dx$$

From this

$$\Gamma = g - c + \int_{-c-g}^{c-g} F(x)dx = g - c + \int_{c+g}^{c-g} F(x)dx + \int_{-c-g}^{c+g} F(x)dx$$

$$= g - c - \int_{c-g}^{c+g} F(x)dx + \int_{0}^{c+g} (F(x) + 1 - F(x))dx = 2g - \int_{c-g}^{c+g} F(x)dx \tag{5.38}$$

where $F(-x) \equiv 1 - F(x)$. Finally

$$\frac{d}{d\vartheta} \mathrm{E}l(\varepsilon) = g'(2g - \int_{c-g}^{c+g} F(x)dx) \tag{5.39}$$

To get Eq. (5.34) consider the following a mathematical truth:

$$\frac{d}{d\vartheta}\lim_{N\to\infty}\frac{1}{N}\sum_{t=1}^{N}\mathrm{E}l(\varepsilon)=\lim_{N\to\infty}\frac{1}{N}\sum_{t=1}^{N}\frac{d}{d\vartheta}\mathrm{E}l(\varepsilon) \tag{5.40}$$

Whether there are (many) cases in which this is not so seems to be a difficult question, so we accept that as it is. □

5.4.2. Going from *LS* to *DZ*

The equation to solve in order to obtain the minimizing ϑ for the criterion (2) when there are deterministic disturbances present is now known, as in Eq. (5.32). An interesting question is: Suppose we have an *LS* estimate of the parameters ϑ^0 in Eq. (5.32) and introduce a *DZ* in the criterion, in what way will the estimate change? Mathematically this question can be put like this: Consider the estimate ϑ as a differentiable function of c, what is then $\tilde{\vartheta}'$ at the point $c = 0$? This section uses Eq. (5.34) of the preceding theorem to answer this question. First, redefine $\tilde{\vartheta}$ slightly and note that

$$\frac{d}{dc}\tilde{\vartheta}(c)=\frac{d}{dc}(\vartheta(c)-\vartheta^0)=\frac{d}{dc}\vartheta(c)$$

Differentiating both sides of Eq. (5.34) with respect to the parameter c gives

$$0=\frac{d}{dc}\bar{\mathrm{E}}\varphi[2g-\int_{c-g}^{c+g}F(x)dx]$$

$$=\bar{\mathrm{E}}\varphi[2\varphi^{\mathrm{T}}\tilde{\vartheta}'-F(c+g)(1+\varphi^{\mathrm{T}}\tilde{\vartheta}')+F(c-g)(1-\varphi^{\mathrm{T}}\tilde{\vartheta}')] \tag{5.41}$$

At the point $c = 0$:

$$\bar{\mathrm{E}}\varphi[2\varphi^{\mathrm{T}}\tilde{\vartheta}'(0)-F(g)(1+\varphi^{\mathrm{T}}\tilde{\vartheta}'(0))+(1-F(g))(1-\varphi^{\mathrm{T}}\tilde{\vartheta}'(0))]$$

$$=\bar{\mathrm{E}}\varphi\left[\varphi^{\mathrm{T}}\tilde{\vartheta}'(0)+1-2F(g)\right]=0 \tag{5.42}$$

where, as earlier assumed, f is even. Recall that

$$g[\vartheta(c)]\,|_{c=0}=\tilde{\vartheta}_{LS}^{\mathrm{T}}\varphi+d$$

Hence

$$\bar{\mathrm{E}}[\varphi\varphi^{\mathrm{T}}\tilde{\vartheta}'(0)+\varphi-2F(\tilde{\vartheta}_{LS}^{\mathrm{T}}+d)\varphi]=0 \tag{5.43}$$

and

$$\tilde{\vartheta}'(0)=(\bar{\mathrm{E}}\varphi\varphi^{\mathrm{T}})^{-1}\bar{\mathrm{E}}[(2F(\tilde{\vartheta}_{LS}^{\mathrm{T}}\varphi+d)-1)\varphi] \tag{5.44}$$

$$\tilde{\vartheta}_{LS} = -(\bar{\mathrm{E}}\varphi\varphi^{\mathrm{T}})^{-1}(\bar{\mathrm{E}}\varphi d) \tag{5.45}$$

Substituting this in Eq. (5.44) renders

$$\tilde{\vartheta}'(0) = (\bar{\mathrm{E}}\varphi\varphi^{\mathrm{T}})^{-1}[\bar{\mathrm{E}}(2F(d - \bar{\mathrm{E}}(d\varphi^{\mathrm{T}})(\bar{\mathrm{E}}\varphi\varphi^{\mathrm{T}})^{-1}\varphi) - 1)\varphi]$$

or, with $P = (\bar{\mathrm{E}}\varphi\varphi^{\mathrm{T}})^{-1}$,

$$\tilde{\vartheta}'(0) = P\bar{\mathrm{E}}(2F(d - (\bar{\mathrm{E}}d\varphi)^{\mathrm{T}}P\varphi) - 1)\varphi \tag{5.46}$$

Since Eq. (5.46) is rather difficult to analyze in a general setting, expanding F in a Taylor series may give some information. The following example chooses d so that the first non-zero term of the Taylor expansion had the appropriate sign (to make $\tilde{\vartheta}'(0) > 0$).

Example: Consider the system

$$y(t+1) = -0.7y(t) + \vartheta u(t) + e(t) + d(t), \quad \vartheta^0 = 1$$

where e is Gaussian white noise with unit variance. Choose $u(t) = \sin(0.3t)$ as input and $d(t) = \sin(0.1t)$ as deterministic disturbance. Note that u and d are not correlated. Simulations in MATLAB now indicate that the LS estimate of ϑ is *better* than the estimate obtained with a *DZ* of width $c = 1$:

$$\hat{\vartheta}_{LS} = 0.96, \quad \hat{\vartheta}_{DZ} = 0.92$$

These results were obtained with a data set consisting of 10,000 samples.

For any special case, the way to answer the opening question of this section is to determine the sign of the scalar product $\tilde{\vartheta}_{LS}\tilde{\vartheta}'(0)$. If and only if it is negative, the *DZ* has a positive effect on the *LS* estimate.

5.5. DEAD ZONES AND SET MEMBERSHIP IDENTIFICATION

The traditional description of noise and disturbances influencing a system is to model them as stochastic processes. This leads to the conventional identification methods of maximum likelihood/least squares type. However, there may be reasons to reject this description of the disturbances; see also Ref. (5). If there are measurement errors of quantization type they are bounded. This view has led to the so called "unknown-but-bounded" approach to estimation.[6] The idea is simply to accept all model parameter values that are consistent with a bounded noise assumption:

$$|e(t)| \leq c \tag{5.47}$$

without performing any averaging over the data. This could be described as

$$\hat{\vartheta} = \arg\min \frac{1}{N} \sum_{t=1}^{N} l_c(\varepsilon(t,\vartheta)) \tag{5.48}$$

where

$$l_c(\varepsilon) = \begin{cases} 0, & |\varepsilon| \le c \\ \infty, & |\varepsilon| > c \end{cases} \tag{5.49}$$

or, equivalently,

$$\hat{\vartheta} \in D_N = \{\vartheta \mid \sum_{t=1}^{N} l_c(\varepsilon(t,\vartheta)) = 0\} = \{\vartheta \mid \forall\, t\colon\ |\varepsilon(t,\vartheta)| \le c\} \tag{5.50}$$

The estimate is thus a set, D_N, and the approach is often also called "set membership identification." The set is in practice not found by minimization of Eq. (5.48) but rather by linear programming techniques[(7)] direct calculation[(8)] or outer-bounding by ellipsoids.[(9)]

Now if Eq. (5.47) indeed holds for all disturbances this method works well as does the *DZ* criterion of Eq. (5.2) as found in section 5.1.

However even though there are several reasons to reject the traditional stochastic process description of disturbances, there are also several reasons to reject Eq. (5.47) as the sole description of the noise, i.e., that it possesses no averaging properties whatsoever. It can be argued that a better picture is to describe the noise as

$$e(t) = v(t) + w(t) \tag{5.51}$$

where $v(t)$ is subject to Eq. (5.47) and $w(t)$ has conventional averaging properties, i.e., in the linear regression case

$$\bar{\mathrm{E}}\varphi(t)w(t) = \lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} \mathrm{E}\varphi(t)w(t) = 0 \tag{5.52}$$

The conventional set-membership approach deals with Eq. (5.51) by extending the value c in Eq. (5.47) until D_N in Eq. (5.50) becomes non-empty. This is quite a conservative approach. A seemingly more natural approach would be to use the *DZ* criterion of Eq. (5.2), i.e., to "soften the infinitely steep walls" in Eq. (5.49). However as shown in the preceding section's example there is no guarantee that the *DZ* criterion of Eq. (5.2) performs any better than the conventional quadratic criterion in Eq. (5.51). The value of a *DZ*, although reasonable from an ad hoc point of view, can thus be said to be questionable.

ACKNOWLEDGMENT. This work was financially supported by the Swedish Council for Technical Research (TFR). This work was performed when K. Forsman was with the Department of Electrical Engineering, Linköping University, Sweden.

REFERENCES

1. B. Egardt, *Stability of Adaptive Controllers*, volume 20 of *Lecture Notes in Control and Information Sciences*, Springer (1976).
2. G. C. Goodwin, D. J. Hill, D. Q. Mayne, and R. H. Middleton, *Adaptive Robust Control (Convergence, Stability and Performance)*, Technical Report EE8544, Dept. of Electrical and Computer Engineering, The University of Newcastle, New South Wales, Australia (1985).
3. L. Ljung, *System Identification - Theory for the User*, Prentice-Hall, Englewood Cliffs, NJ, p. 345 (1987).
4. F. B. Hildebrand, *Advanced Calculus for Applications*, Prentice-Hall, Englewood Cliffs, NJ, p. 359 (1962).
5. L. Ljung, *IEEE Control Syst. Mag.* **11**, 25 (1991).
6. F. C. Schweppe, *Uncertain Dynamical Systems*, Prentice-Hall, Englewood Cliffs, NJ (1973).
7. M. Milanese and R. Tempo, *IEEE Trans. Aut. Control* **AC-30**, 730 (1985).
8. E. Walter and H. Piet-Lahanier, in: *IEEE Proceedings of the 26th Conference on Decision and Control*, pp. 1921–1922 (1987).
9. E. Fogel and Y. F. Huang, *Automatica* **18**, 229 (1982).

6

Recursive Estimation Algorithms for Linear Models with Set Membership Error

G. Belforte and T. T. Tay

ABSTRACT

This chapter reviews some of the more recent algorithms for sequential parameter identification in the context of unknown but bounded measurement errors when the model output is linear in the parameters. The properties of the different algorithms are analyzed and compared.

The possibility of evaluating the confidence of the obtained estimates is discussed, particularly information required on the noise structure in order to assess the confidence of the estimates is shown.

Finally, the possibility of using the algorithms for time-varying system identification is considered and the case of uncertain regressors is addressed.

6.1. INTRODUCTION

Data used in parameter estimation are associated with some uncertainty. Traditionally such uncertainty receives a stochastic description, e.g., as an additive

G. Belforte • Dipartimento di Automatica e Informatica, Politecnico di Torino, 10129 Torino, Italy. T. T. Tay • Department of Electrical Engineering, National University of Singapore, Singapore 0511.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

random noise with a given probability density function. The estimation process is then set in a statistical framework. Various techniques such as maximum likelihood estimation are used to exploit the prior information on the noise. The quality of the parameter estimates is then assessed through the use of indices such as the Fisher information matrix. One problem is that the randomness assumption may in many practical cases be rather unrealistic. Moreover, the amount of data available is often not sufficient to check the validity of this assumption.

The unknown but bounded error (*UBBE*) description of the measurement noise was pioneered by Schweppe[1] about 20 years ago. It does not rely on a stochastic framework. The errors are assumed to belong to error sets with some given shapes. There is no easy and straightforward description for sets with arbitrary shapes, but useful results can be obtained in some simple and nevertheless important special cases. One such case is orthotopes, in which each component of the error vector is constrained to belong to some finite interval. Such descriptions have been shown to fit practical applications.[2]

Since the introduction of the *UBBE* description, much work has been done to develop algorithms that exploit this assumption.[3,4,5] The requirements for memory and computing time may however become unrealistic. Hence the interest for recursive algorithms which can update parameter estimates after each new measurement while requiring a limited amount of memory and computing time.

This chapter describes major classes of recursive algorithms available for models linear in their parameters and compare their performances. Particular attention is devoted to algorithms with limited memory and computing time.

Section 6.2 presents the framework and notation of the study. Section 6.3 describes algorithms based on evaluating the exact polytope where the parameter estimate can lie. Recursive fixed memory and computational time algorithms (which are based on overbounding the exact polytopes) are presented in Section 6.4. In Section 6.5, modifications to the various algorithms to cater for time-varying systems are discussed while uncertainties in the system model are addressed in Section 6.6. Simulation results are presented in Section 6.7.

6.2. GENERALITIES

This chapter considers the parameter identification problem for models linear in their parameters described by

$$y_i = a_i^T\theta + e_i \quad i = 1, \ldots, k \tag{6.1}$$

where $y_i \in R$ is the i-th measurement, $a_i^T \in R^p$ is the corresponding regressor, $\theta \in R^p$ is the parameter vector to be estimated and $e_i \in R$ is the measurement error. The measurement error is assumed to be unknown but bounded so that

$$|e_i| \leq \beta w_i = E_i \quad i = 1, \ldots, k \tag{6.2}$$

where E_i is the error bound for the i-th measurement with relative weight w_i, β being a scaling factor.

Two cases, differing by the assumption on the information content about the errors have been addressed in the literature. They are stated as the following conditions.

CONDITION 1. All the bounds E_i $i = 1, \ldots, k$ are known. In this case, without any loss of generality, $\beta = 1$ and the weights w_i $i = 1, \ldots, k$ equal the known error bounds; that is, $w_i = E_i$ $i = 1, \ldots, k$.

CONDITION 2. The exact values of the bounds E_i $i = 1, \ldots, k$ are unknown. However the weights w_i $i = 1, \ldots, k$ are known. In this case the constant scaling factor β can no longer be assumed equal to one. Here $\beta w_i = E_i$ $i = 1, \ldots, k$ are not known.

When a system is described by Eq. (6.1) and the measurement errors are given by Eq. (6.2), the problem of parameter identification, when k measurements are available, usually involves choosing, as parameter estimate, one element of the admissible parameter set $D(k)$,

$$D(k) = \{\theta \in R^p: y_i - E_i \leq a_i^T\theta \leq y_i + E_i \quad i = 1, \ldots, k\} \tag{6.3}$$

$$= \{\theta \in R^p: y_i - \beta w_i \leq a_i^T\theta \leq y_i + \beta w_i \quad i = 1, \ldots, k\}. \tag{6.4}$$

Here $D(k)$ is the set of all the parameters consistent with the given model, the available information on the error and the measurement vector. It is a polytope in the R^p parameter space described by a suitable subset of the planes P_i^- and P_i^+ $i = 1, \ldots, k$ defined by

$$P_i^-: a_i^T\theta = y_i - E_\imath \quad P_i^+: a_i^T\theta = y_i + E_i. \tag{6.5}$$

To each plane are associated half-spaces S_i^+ and S_i^- in R^p defined by

$$S_i^+ = \{\theta \in R^p; a_i^T\theta \leq y_i + E_i\}$$

$$S_i^- = \{\theta \in R^p; a_i^T\theta \geq y_i - E_i\}. \tag{6.6}$$

Let S_i be the set of parameter vectors that are consistent with the i-th measurement. Then

$$S_i = S_i^+ \cap S_i^- \tag{6.7}$$

and

$$D(k) = \bigcap_{i=1}^{k} S_i \tag{6.8}$$

Subsequent sections of the chapter refer to the set of all P_i^- and/or P_i^+ planes defining the boundary of $D(k)$ as $B_D(k)$.

Any point in $D(k)$ can, in principle, be an estimate of the parameter vector while the "size" of $D(k)$ is a measure of the parameters' reliability. Several choices are possible according to different criteria. However, in the unknown but bounded error (UBBE) context, the usual choices are either the Chebicev center $\theta^o(k)$ of $D(k)$* (referred to as the central estimate) or the projection estimate $\theta^P(k)$† that is a minimax estimate.$^{(6,7,8)}$ The central estimate is optimal with respect to the worst case error, while the projection estimate minimizes the l_∞ norm of the prediction error. A recent survey of the properties of these and other possible estimates can be found in.$^{(9)}$

The parameter reliability is usually evaluated by computing the parameter uncertainty intervals $PUI_i(k)$ $i = 1, \ldots, p$ defined as

$$PUI_i(k) = [\theta_i^{\min}(k), \quad \theta_i^{\max}(k)] \quad i = 1, \ldots, p \tag{6.9}$$

where

$$\theta_i^{\min}(k) = \min_{\theta \in D(k)} \theta_i \qquad \theta_i^{\max}(k) = \max_{\theta \in D(k)} \theta_i. \tag{6.10}$$

In general $\theta_i^{\min}(k)$ and $\theta_i^{\max}(k)$ are achieved on a vertex of $D(k)$ where p suitable P_i^+ and/or P_i^- planes intersect. These sets of p planes will be denoted as $B_{D_i^{\min}}(k)$ and $B_{D_i^{\max}}(k)$, $i = 1, \ldots, p$.

Note that for the computation of $D(k)$ the values of the error bounds E_i $i = 1, \ldots k$, must be known exactly. Since the knowledge of $D(k)$ is essential to get the central estimate $\theta^o(k)$ as well as the parameters' uncertainties, it turns out that both the central estimate and the *PUI*s can be exactly computed only when Condition 1 holds.

For the evaluation of the projection estimate $\theta^P(k)$, less information about the measurement error is needed. The problem can be reduced to finding the $\theta^P(k)$ vector and the smallest positive scalar $\alpha(k)$ for which the constraints

*The Chebishev center θ^o of a set D is

$$\theta^o(k){:}\ \sup_{\theta \in D(k)} \|\theta^o - \theta\| = \inf_{\vartheta \in R^p} \ \sup_{\theta \in D(k)} \|\vartheta - \theta\|$$

†Let $A \in R^{k \times p}$ be the matrix whose rows are a_i^T $i = 1, \ldots, k$. Then

$$\theta^P(k){:}\ \|y - A\theta^P(k)\|_\infty^w = \min_\theta \|y - A\theta\|_\infty^w$$

$$y_i - \alpha(k)w_i \leq a_i^T\theta^p(k) \leq y_i + \alpha(k)w_i \quad i = 1, \ldots, k \tag{6.11}$$

are satisfied. Here, only Condition 2 needs to hold. Whenever only the error weights $w_i\, i = 1, \ldots, k$ are known, the $D(k)$ set defined in Eq. (6.4) is undetermined due to the lack of information on β. However, a new admissible parameter set $\tilde{D}(k)$ can be defined in the R^{p+1} extended parameter space of θ and β as

$$\tilde{D}(k) = \{\theta,\beta \in R^{p+1}\colon\ y_i - \beta w_i \leq a_i^T\theta \leq y_i + \beta w_i \quad i = 1, \ldots, k\}. \tag{6.12}$$

Similar to $D(k)$, the set $\tilde{D}(k)$ is a polyhedron. However while $D(k)$ is usually bounded (whenever $k \geq p$ uncorrelated measurements are available), $\tilde{D}(k)$ is unbounded as long as no prior information on β is available.

The polyhedron $\tilde{D}(k)$ can also be described by a suitable subset of the planes $\tilde{P}_i^-$ and $\tilde{P}_i^+$ obtained by rearranging the Eq. (6.12) in the following form

$$\tilde{P}_i^-\colon\ y_i = [a_i^T, w_i]\begin{pmatrix}\theta\\ \beta\end{pmatrix};$$

$$\tilde{P}_i^+\colon\ [a_i^T, -w_i]\begin{pmatrix}\theta\\ \beta\end{pmatrix} = y_i \quad i = 1, \ldots, k. \tag{6.13}$$

To each plane are associated half-spaces $\tilde{S}_i^+$ and $\tilde{S}_i^-$ in R^{p+1} defined by

$$\tilde{S}_i^-\colon\ y_i \leq [a_i^T, w_i]\begin{pmatrix}\theta\\ \beta\end{pmatrix}$$

$$\tilde{S}_i^+\colon\ [a_i^T, -w_i]\begin{pmatrix}\theta\\ \beta\end{pmatrix} \leq y_i \quad i = 1, \ldots, k. \tag{6.14}$$

Again if $\tilde{S}_i$ is the extended parameter set consistent with the i-th measurement,

$$\tilde{S}_i = \tilde{S}_i^+ \cap \tilde{S}_i^- \tag{6.15}$$

and

$$\tilde{D}(k) = \bigcap_{i=1}^{k} \tilde{S}_i. \tag{6.16}$$

Subsequent sections of this chapter refer to the set of all $\tilde{P}_i^-$ and/or $\tilde{P}_i^+$ planes defining the boundary of $\tilde{D}(k)$ as $B_{\tilde{D}}(k)$.

Any point in $\tilde{D}(k)$ can, in principle, be an estimate of the extended parameter vector. However it is reasonable to choose the estimate that minimizes the prediction error. This results in the projection estimate $\theta^p(k)$. In general $\theta^p(k)$ is achieved on a vertex of $\tilde{D}(k)$ where $p + 1$ planes intersect and

$$\alpha(k) = \min_{\theta,\beta \in \tilde{D}(k)} \beta. \tag{6.17}$$

This set of $p + 1$ planes will be referred to as $B_{\tilde{D}}^{\theta_p}(k)$.

Note that for many practical situations, only Conditions 2 can be shown to hold and not Condition 1. Thus the projection estimate is usually more interesting compared to the central estimate which requires Condition 1 to hold. When Condition 1 holds then the $D(k)$ set is the intersection of $\tilde{D}(k)$ with the $\beta = 1$ plane.

To evaluate the central estimate $\theta^{c}(k)$, the projection estimate $\theta^{p}(k)$ and the parameter uncertainty intervals $PUI_i(k)$ $i = 1, \ldots, k$, the exact knowledge of $D(k)$ and/or $\tilde{D}(k)$ is required. The complexity of their exact description can of course become too complicated to be handled.(6,10,11,12) This fact suggested the search for suboptimal algorithms with fixed amount of storage memory and reduced computation requirements. The most popular of these algorithms is the Fogel–Huang algorithm,(13) that computes an ellipsoidal outer bound to $D(k)$. More recently some other algorithms have been proposed. These include computing an orthotopic outer bound(14) to $D(k)$ or selectively storing a small number of suitable past measurements to be used for computing approximated central and projection estimates(15) as well as their uncertainties.(16)

The need for dealing with time varying systems requires that some kind of "aging" of past measurements be introduced so that $D(k)$ and/or $\tilde{D}(k)$ are not constrained to shrink monotonically when the number k of available measurements increases. This allows the parameters to change. The techniques used are similar to the introduction of forgetting factors used in statistical estimation processes.(17,18,19) However other schemes could be proposed and more investigation should be devoted to this topic.

A last point concerns those problems in which the reregresors a_i^T $i = 1, \ldots, k$ are uncertain. This case has been considered in the *UBBE* context.(20,21,22) It must be noted that when all the errors and uncertainties are uncorrelated the $D(k)$ and/or the $\tilde{D}(k)$ sets can still be evaluated. However, if correlation is present only overbounds can be computed.(23)

6.3. EXACT DESCRIPTION OF THE ADMISSIBLE PARAMETER SET

Algorithms where the exact description of $D(k)$ or $\tilde{D}(k)$ is obtained will be considered in this section.

6.3.1. Central Estimate and Parameter Uncertainty Evaluation

The central estimate $\theta^{c}(k)$ and the parameter uncertainty intervals $PUI_i(k)$ $i = 1, \ldots, p$ can be evaluated whenever the exact description of $D(k)$ is available. Three algorithms for the recursive evaluation of $D(k)$ have been proposed(10,11,12) and compared.(24) Their structure can be summarized in the following steps:

Step 1: Initialize the procedure by processing the first p measurements to find the $D(p)$ set, the list of its vertices and the $B_D(p)$ set of all the planes P_i^- and/or P_i^+ describing its boundary.

Step 2a: When the (k+1)-th measurement becomes available, check whether $D(k) = D(k) \cap S_{k+1}^-$. *If yes*: put $D(k') = D(k)$ and go to Step 2b. In this case $B_D(k) = B_{D(}k')$ so that the list of vertices is the same for $D(k')$ and $D(k)$. *If no*: evaluate $D(k') = D(k) \cap S_{k+1}^-$. Construct $B_D(k')$ by adding the plane P_{k+1}^- to $B_D(k)$ and discarding those that no longer define $D(k')$. Then go to Step 2b.

Step 2b: Check whether $D(k') = D(k') \cap S_{k+1}^+$. *If yes*: put $D(k + 1) = D(k')$. Go to Step 2a and wait for a new measurement. In this case $B_D(k + 1) = B_D(k')$ so that the list of vertices is the same for $D(k')$ and $D(k + 1)$. *If no*: evaluate $D(k + 1) = D(k') \cap S_{k+1}^-$. Construct $B_D(k + 1)$ by adding the plane P_{k+1}^+ to $B_D(k')$ and discarding those that no longer define $D(k + 1)$. Then go to Step 2a and wait for a new measurement.

From the vertices of $D(k)$ it is then straightforward to derive the $PUI_i(k)$ $i = 1, \dots, p$ and consequently $\theta^c(k)$.

It should be pointed out that updating the list of the planes that concur to the description of the bound of $D(k + 1)$ is, in general, time consuming, especially when the number of dimension p increases.

6.3.2. Projection Estimate and Parameter Uncertainty Evaluation

The projection estimate $\theta^p(k)$ can be evaluated whenever the exact description of $\tilde{D}(k)$ is available. This can be done when the information about the measurement errors is given by Condition 1 or 2. In principle any algorithm suitable for computing the $D(k)$ set in the R^p parameter space can be used for deriving the $\tilde{D}(k)$ set in the R^{p+1} extended parameter space. Once the vertices of $\tilde{D}(k)$ are available the derivation of $\theta^p(k)$ is straightforward. The structure of such an algorithm is similar to that described before for the central estimate and therefore is not repeated here. An actual implementation of such an algorithm is presented in Ref. 6.

It is important to note that the projection estimate $\theta^p(k)$ is obtained via the determination of $\tilde{D}(k)$. When the measurement error description is given according to Condition 1, the $D(k)$ set can be obtained by intersecting the $\tilde{D}(k)$ set with the plane $\beta = 1$. The $PUI_i(k)$ can then be derived from the knowledge of $D(k)$. This step, although possible, is time consuming.

6.4. APPROXIMATE DESCRIPTION OF THE ADMISSIBLE PARAMETER SET

In the preceding section the algorithms for the recursive computation of central and projection estimates through the exact determination of $D(k)$ and $\tilde{D}(k)$ have been outlined. Those algorithms are, in general, time consuming and require potentially unbounded storage memory although some simulation study shows that this event is unlikely to happen.(11)

The frequent need for fast, online recursive identification with fixed storage memory has motivated the search for simpler recursive algorithms. These would compute some kind of "approximated estimates" provided that their loss in performances remains tolerable. This section presents some of these algorithms that mainly evaluate different kind of outer bounds to $D(k)$ or $\tilde{D}(k)$ and use them to derive suitable parameter estimates as well as to evaluate their reliability. Most of these algorithms require that the error description is given according to Condition 1, and this poses a constraint on their practical use.

The interested reader should refer to the cited literature for an exact description of the algorithms and their properties. Here only some of their common features are listed before presenting them briefly.

- While the central and projection estimates $\theta^{c}(k)$ and $\theta^{p}(k)$ always belong to the $D(k)$ set, the approximate point estimates that can be constructed from these outer bounds are *not* guaranteed to belong to the $D(k)$ set.
- The parameters uncertainty bounds, if computable, are an outer bound to the *PUI*s.
- When the number k of measurements goes to infinite, under fairly general assumptions on the error that cannot however be overbounded, the obtained estimates converge to the true parameter vector that generated the data.(15,25) However the convergence is slower than that of the exact central and projection estimates.

6.4.1. The Fogel–Huang Algorithm

The Fogel–Huang algorithm can be applied only when Condition 1 on the measurement errors is satisfied. The key idea here is to overbound the $D(k)$ set with a suitably chosen ellipsoid $\Phi(k)$. The original algorithm described in Ref. 13 was later improved,(26) and the optimality of this version has been proved.(27)

This algorithm can be summarized in the following steps:

Step 1: Initialize the procedure by selecting an ellipsoid $\Phi(0)$ that contains $D(0)$ (*a priori* information).

Step 2: When the $(k+1)$-th measurement becomes available, find the minimum volume ellipsoid $\Phi(k+1)$ such that $\Phi(k+1) \supseteq \Phi(k) \cap S_{k+1}$.

The center of the ellipsoid $\Phi(k)$ may be used as a point estimate of the parameter at step k while some measure of the extent of the ellipsoid is used to assess the reliability of this point estimate.

Note that this algorithm is sensitive to both the initializing ellipsoid $\Phi(0)$ and the order in which measurements are processed. A reprocessing of past measurements often leads to a drastic reduction of the size of the obtained ellipsoid.(26)

6.4.2. The Pearson Algorithm

The Pearson algorithm[14] can be applied only when Condition 1 on the measurement error is satisfied. The key idea is to overbound the $D(k)$ set with a suitably chosen orthotopic bound $O(k)$.

To run the algorithm, the available measurements must be partitioned into L submatrices of p measurements each. Each submatrix must be nonsingular otherwise it is discarded. For this purpose, *nonredundant partitioning* is defined as any collection $\{X_l\}$ of L $p \times p$ matrices formed from the regressor vectors a_i^T $i = 1, \ldots, k$ such that any regressor does not appear more than once in any X_l matrix.

If all the available data are used, then $k/p \leq L \leq k!/[(k-p)!p!]$. Since the case $L = k!/[(k-p)!p!]$ is not tractable, the following two partitionings are suggested for practical use.

- Disjoint partitioning where $L = k/p$ and the l-th X_l matrix consists of the p regressors a_i^T $i = (l-1)p + 1, \ldots, lp$
- Sliding block partitioning obtained for $L = k-p$ combining each regressor a_i^T with its $(p-1)$ predecessors so that X_l consists of the regressors a_i^T $i = l - j, j = 0, \ldots, p-1$.

Both partitionings are suitable for recursive estimation. The algorithm described as follows is for the sliding block partitioning. Changes to deal with the disjoint partitioning are trivial. The algorithm can be summarized in the following steps:

Step 1: Initialize the procedure by computing the tight outer bounding orthotope O_p of the set X_p of the first p measurements.

Step 2: When the $(k + 1)$-th measurement becomes available, form the new X_{k+1} set and compute its tight outer bounding orthotope $\overline{O}_{k+1}$.

Step 3: Compute the orthotope $O_{k+1} = O_k \cap \overline{O}_{k+1}$, an outer bound to $D(k)$.

The center of $O(k)$ may be used as the parameter estimate at step k while the orthotope $O(k)$ itself accounts for the parameter reliability.

Note that this algorithm is similar to a technique used in Ref. 6.28. It is sensitive to the order in which measurements are processed and a reprocessing of past measurements without a change in their order, will not affect the obtained result.

6.4.3. The ARCE Algorithm

The approximate recursive central estimate (ARCE) algorithm[15] can be applied only when Condition 1 on the measurement error is satisfied. In this algorithm $2p^2$ suitably selected measurements are stored and used to derive an approximated central estimate $\hat{\theta}^c(k)$. Here instead of storing $B_D(k)$, all the planes that define $D(k)$, $2p$ sets (of p planes each) $\hat{B}_{D_i^{\min}}(k)$ and $\hat{B}_{D_i^{\max}}(k)$, $i = 1, \ldots, p$ that define $\hat{\theta}_i^{\min}(k)$ and $\hat{\theta}_i^{\max}(k)$ are stored. $\hat{\theta}_i^{\min}(k)$ and $\hat{\theta}_i^{\max}(k)$ are an outer bound to $\theta_i^{\min}(k)$ and $\theta_i^{\max}(k)$ and they allow to derive $\hat{\theta}^{o}(k)$ whose components are

$$\hat{\theta}_i^{o}(k) = \frac{\hat{\theta}_i^{\min}(k) + \hat{\theta}_i^{\max}(k)}{2} \qquad i = 1, \ldots, p \tag{6.18}$$

When the $(k + 1)$-th measurement becomes available giving rise to the two planes P_{k+1}^{+} and P_{k+1}^{-} define $\bar{D}_i^{\min}(k+1)$, $\bar{D}_i^{\max}(k+1)$ as the admissible parameter sets relative to

$$\hat{B}_{D_i^{\min}(k)} \bigcup P_{k+1}^{+} \bigcup P_{k+1}^{-}, \quad \hat{B}_{D_i^{\max}}(k) \bigcup P_{k+1}^{+} \bigcup P_{k+1}^{-}$$

respectively. The *ARCE* is then updated computing

$$\hat{\theta}_i^{\min}(k+1) = \min_{\theta \in \bar{D}_i^{\min}(k+1)} \theta_i, \qquad \hat{\theta}_i^{\max}(k+1) = \max_{\theta \in \bar{D}_i^{\max}(k+1)} \theta_i, \tag{6.19}$$

and getting $\hat{\theta}^{o}(k+1)$ according to Eq. (6.18).

The implementation of the *ARCE* algorithm can be summarized as follows:

Step 1: Initialize the procedure by processing the first p measurements and find the corresponding central estimate $\theta^{o}(p)$. Let

$$\hat{\theta}^{o}(p) = \theta^{o}(p)$$

and store the $2p$ sets $\hat{B}_{D_i^{\min}}(p) = B_{D_i^{\min}}(p)$ and $\hat{B}_{D_i^{\max}}(p) = B_{D_i^{\max}}(p)$, $i = 1, \ldots, p$.

Step 2: When the $(k + 1)$-th measurement becomes available, test for $i = 1, \ldots, p$ whether $\hat{\theta}_i^{\min}(k) = \hat{\theta}_i^{\min}(k+1)$ (this test can be performed without actually computing $\hat{\theta}_i^{\min}(k+1)$[15]). *If yes*: put $\hat{\theta}_i^{\min}(k+1) = \hat{\theta}_i^{\min}(k)$ and $\hat{B}_{D_i^{\min}}(k+1) = \hat{B}_{D_i^{\min}}(k)$. *If no*: Compute the $\theta_i^{\min}(k+1)$ according to Eq. (6.19) and update $\hat{B}_{D_i^{\min}}(k+1)$. When all the p $\hat{\theta}_i^{\min}$ parameters have been processed go to Step 3.

Step 3: Update $\hat{\theta}_i^{\max}(k+1)$ and $\hat{B}_{D_i^{\max}}(k+1)$ $i = 1, \ldots, p$ with a procedure similar to that of step 2.

Step 4: Compute $\hat{\theta}^{o}(k+1)$ according to Eq. (6.18). Go to Step 2 and wait for a new measurement.

Note that an iterated reprocessing of all the past measurements would lead to the determination of the central estimate.[15]

6.4.4. The ARPE Algorithm

The approximate recursive projection estimate (*ARPE*) algorithm[15] can be applied when the assumption on measurement errors satisfies Conditions 1 or 2. Here only $p + 1$ suitably selected past measurements are stored and the projection estimate relative to this subset of $p + 1$ measurements is computed. In the ARPE algorithm, analogous to the set $B_{\tilde{D}^{\theta_p}}(k)$ that defines the projection estimate $\theta^{p}(k)$, we have a set $\hat{B}_{\tilde{D}^{\theta_p}}(k)$ that defines the ARPE $\hat{\theta}^{p}(k)$.

The *ARPE* algorithm computes $\hat{\theta}^p(k)\hat{\alpha}(k)$ and $\hat{B}_{\tilde{D}}^{\theta_p}(k)$ according to the following steps:

Step 1: Initialize the procedure by processing the first $p + 1$ measurements finding the corresponding projection estimate $\theta^p(p + 1)$ and the associated $\alpha(p + 1)$. Let $\hat{\theta}^p(p + 1) = \theta^p(p + 1)$, $\hat{\alpha}(p + 1) = \alpha(p + 1)$ and $\hat{B}_{\tilde{D}}^{\theta_p}(k + 1) = B_{\tilde{D}}^{\theta_p}(p + 1)$.

Step 2: When the $(k{+}1)$-th measurement becomes available, test whether $\hat{\theta}^p(k)$ and $\hat{\alpha}(k)$ are the projection estimate θ^p and the associated α relative to the set whose bound is described by $\hat{B}_{\tilde{D}}^{\theta_p}(k) \cup P^+_{k+1} \cup P^-_{k+1}$. *If yes*: put $\hat{\theta}^p(k + 1) = \hat{\theta}^p(k)$, $\hat{\alpha}(k + 1) = \hat{\alpha}(k)$ and $\hat{B}_{\tilde{D}}^{\theta_p}(k + 1) = \hat{B}_{\tilde{D}}^{\theta_p}(k)$. Repeat Step 2 when a new measurement becomes available. *If no*: go to Step 3.

Step 3: Compute the projection estimate θ^p and the associated α corresponding to the set whose bound is described by $\hat{B}_{\tilde{D}}^{\theta_p}(k) \cup P^+_{k+1} \cup P^-_{k+1}$. Put $\hat{\theta}^p(k + 1) = \theta^p$, $\hat{\alpha}(k + 1) = \alpha$. Update $\hat{B}_{\tilde{D}}^{\theta_p}(k + 1)$. Go to Step 2 and wait for a new measurement.

Note that an iterated reprocessing of all the past measurements would lead to the determination of the projection estimate.(15)

6.4.5. Approximate PUI Evaluation with ARPE Algorithm

The *ARPE* algorithm does not provide any information about the parameter reliability even when the information about the measurement error, being provided by Condition 1, would allow to derive it. In such case it is convenient to derive some procedure that can provide this information. An exact derivation of the *PUI*s would be optimal, but its evaluation requires the knowledge of $D(k)$ that can only be achieved when exact algorithms are used. It is therefore interesting to investigate the possibility of deriving, with little extra computation, some upper bound to the *PUI*s when using the *ARPE* algorithm.(16)

An approximate evaluation of the parameter uncertainties when the measurement error is described according to Condition 1 and the ARPE algorithm is used, consists of computing, at each step k, the parameter uncertainty intervals relative to the set of $p + 1$ planes in $\hat{B}_{\tilde{D}}^{\theta_p}(k)$. Here, let $\hat{D}(k)$ be the parameter admissible set corresponding to the $p + 1$ measurements of $\hat{B}_{\tilde{D}}^{\theta_p}(k)$ with $\beta = 1$. Then define

$$\hat{\theta}_i^{p\min}(k) = \min_{\theta \in \hat{D}(k)} \theta_i, \qquad \hat{\theta}_i^{p\max}(k) = \max_{\theta \in \hat{D}(k)} \theta_i . \tag{6.20}$$

from which the $P\hat{U}I_i(k)$ $i = 1, \ldots, p$, defined as

$$P\hat{U}I_i(k) = [\hat{\theta}_i^{p\min}(k),\ \ \hat{\theta}_i^{p\max}(k)] \quad i = 1, \ldots, p, \tag{6.21}$$

can be computed. These $P\hat{U}I$s can be regarded as approximations to the true PUIs. Note that their computation must be performed only when the projection estimate $\hat{\theta}^p(k)$ has been updated according to Step 3 of the previous section.

It is easy to construct simple examples showing that the $P\hat{U}I$s do not necessarily shrink for increasing k, in contrast with the *PUI*s. This undesirable feature can however be corrected. In fact it is possible to compute, at each step k, the quantities

$$\bar{\theta}_i^{\min}(k) = \max[\bar{\theta}_i^{\min}(k-1), \hat{\theta}_i^{p\min}(k)]$$

$$\bar{\theta}_i^{\max}(k) = \min[\bar{\theta}_i^{\max}(k-1), \hat{\theta}_i^{p\max}(k)]. \tag{6.22}$$

initializing the procedure with $\bar{\theta}_i^{\min}(p) = \hat{\theta}_i^{p\min}(p)$ and $\bar{\theta}_i^{\max}(p) = \hat{\theta}_i^{\max}(p)$. By construction, $\theta_i^{\min}(k)$ and $\bar{\theta}_i^{\max}(k)$ are monotonic functions of k.

If the $P\bar{U}I_i(k)$ $i = 1, \ldots, p$ are defined as

$$P\bar{U}I_i(k) = [\bar{\theta}_i^{\min}(k), \ \bar{\theta}_i^{\max}(k)] \quad i = 1, \ldots, p \tag{6.23}$$

it is trivial to show that

$$P\hat{U}I_i(k) \supseteq P\bar{U}I_i(k) \supseteq PUI_i(k) \quad i = 1, \ldots, p. \tag{6.24}$$

so that the $P\bar{U}I$s can be used as a better approximation to the PUIs. It is noted that the computation of $P\hat{U}I$ and $P\bar{U}I$ is simple and can be performed according to the results of Lemmas 1 and 2 of Ref. 6.29 mainly requiring the inversion of p $p \times p$ matrices.

6.5. TIME VARYING SYSTEMS

The need to deal with time varying systems has motivated the introduction of forgetting techniques similar to those used in statistical estimation processes.(17,18) The most popular ones are probably windowing over a fixed horizon, where those measurements that are older than a given threshold are discarded, and the use of a forgetting factor where the error bounds E_i (or equivalently the weights w_i) of past measurements are multiplied by a constant γ greater than one, at each new data acquisition.

Both schemes require extensive computation at each step, where the central estimate is concerned. Moreover the central estimate will change at each step k, whenever a forgetting factor is present, since the admissible parameter set $D(k)$ is affected by the forgetting scheme even when it does not depend on the last measure at step k. This feature is not specially convenient and contradicts the intuitive feeling that forgetting scheme should affect the estimates' reliability only and not the estimates themselves. A different approach, that overcomes this defect was recently proposed.(19) It expands the admissible parameter set instead of the error bounds, the expansion being symmetrical with respect to the central estimate.

Things are simpler for projection estimates. The computation required for data updating with the windowing scheme is smaller and a forgetting factor influencing the error weights w_i induces changes on $\alpha(k)$ only. In fact if in Eq. (6.11) the weights

w_i are multiplied by some factor γ, then the minimum value $\alpha(k)$ for which Eq. (6.11) are satisfied must be multiplied by $1/\gamma$ while $\theta^p(k)$ remains unaffected. This kind of consideration holds also for the *ARPE* algorithm.

6.6. UNCERTAINTY IN THE REGRESSORS

There are cases in which the regressor vectors, $a_i^T \in R^p$ $i = 1, \ldots, k$ are uncertain. In the bounded error context this uncertainty can be described assuming that

$$a_i^T = a_i^{*T} + \delta a_i^T \quad i = 1, \ldots, k \tag{6.25}$$

where a_i^{*T} represents the nominal value of the regressor vector while δa_i^T is its uncertainty,, which is assumed to be componentwise bounded so that

$$|\, \delta a_{ij} \,| \leq \Delta a_{ij} \quad i = 1, \ldots, k \quad j = 1, \ldots, p \tag{6.26}$$

where Δa_{ij} $i = 1, \ldots, k$ $j = 1, \ldots, p$ are known quantities.

In such condition it can be shown that in each orthant of the R^p parameter space, the $D(k)$ region is still a polytope, but the P_i^- and P_i^+ $i = 1, \ldots, k$ planes are no longer pairwise parallel.(20,21,22) In fact the $D(k)$ region is described by

$$D(k) = \{\theta \in R^p\colon (a_i^T - \Delta a_i^{oT})\theta \leq y_i + E_i;$$

$$(a_i^T + \Delta a_i^{oT})\theta \geq y_i - E_i \quad i = 1, \ldots, k\}, \tag{6.27}$$

where

$$\Delta a_i^{oT} = [\Delta a_{i1} sgn(\theta_1) \ldots \Delta a_{ip} sgn(\theta_p)]. \tag{6.28}$$

Since all the algorithms presented do not require the planes P_i^- and P_i^+ $i = 1, \ldots, k$ to be pairwise parallel, suitable versions of the algorithms can be implemented to deal with cases with bounded uncertainty on the components of the regressors. It is however important to remark that the computational burden can increase dramatically if there is no prior information on the orthant(s) in which the $D(k)$ region is located. Moreover, in the case in which there is correlation among the regressors' uncertainties, only upper bounds to the $D(k)$ set can in general be obtained.(23) This last case occurs, for example, when *AR*, *MA* or *ARMA* models are considered with bounded noise both on the input and on the output.

6.7. NUMERICAL EXAMPLE

To compare the performances of all the previously described algorithms, they were used for identifying the parameter vector of a third order *MA* system.

Data were obtained from the following simulated model

$$y(k) = 3.0u(k) + 1.5u(k-1) + 0.7u(k-2) + e(k), \tag{6.29}$$

where the error $e(k)$ is white, uniformly distributed so that $e(k) \in [-1,1]$, $\forall k$, and the input vector u belongs to a normally distributed random sequence with mean equal zero and standard deviation equal to one.

Fifty series of inputs and errors were generated. For each of them the six previously presented estimates (Central $\theta^o(k)$, Projection $\theta^p(k)$, Fogel–Huang, Pearson, *ARCE* $\hat{\theta}^o(k)$ and *ARPE* $\hat{\theta}^p(k)$) were computed at each step k. For each parameter and each estimate, the absolute value of the difference between the estimated parameter value and the true one used for generating the data was computed at each step k and averaged over the 50 realizations. The resulting average absolute estimation errors are plotted in Figs. 6.1, 6.2 and 6.3.

The average amplitudes of the *PUI*s, as they can be evaluated when using the various algorithms, were also computed at each step k. In this case note that only five different *PUI*s evaluations are available since the central estimate and the projection estimate have the same parameter bounds. For the *ARPE*, the parameter bounds have been computed using the $\bar{PUI}_i(k)$ and not the $\hat{PUI}_i(k)$ $i = 1, \ldots, p$. Plots of these quantities are reported in Fig. 6.4.

From Figs. 6.1, 6.2 and 6.3, it can be noted that the average absolute error of the *ARCE* and projection estimate are quite close and are just slightly worse than

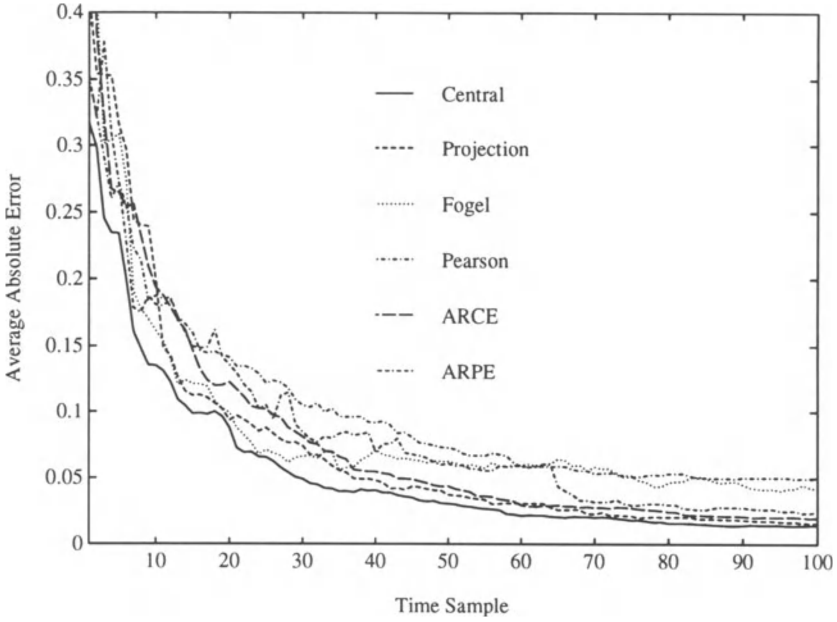


FIGURE 6.1. Average absolute error of the first parameter.

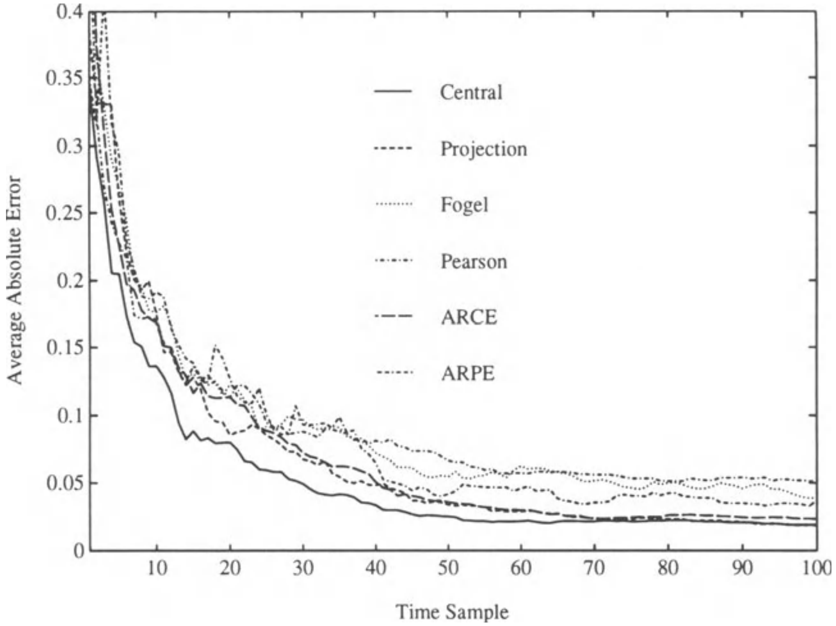


FIGURE 6.2. Average absolute error of the second parameter.

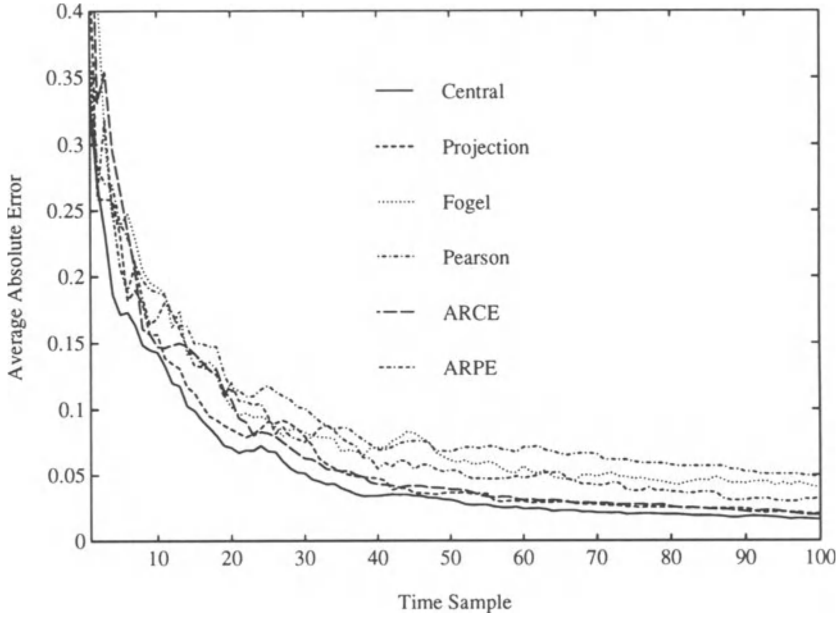


FIGURE 6.3. Average absolute error of the third parameter.

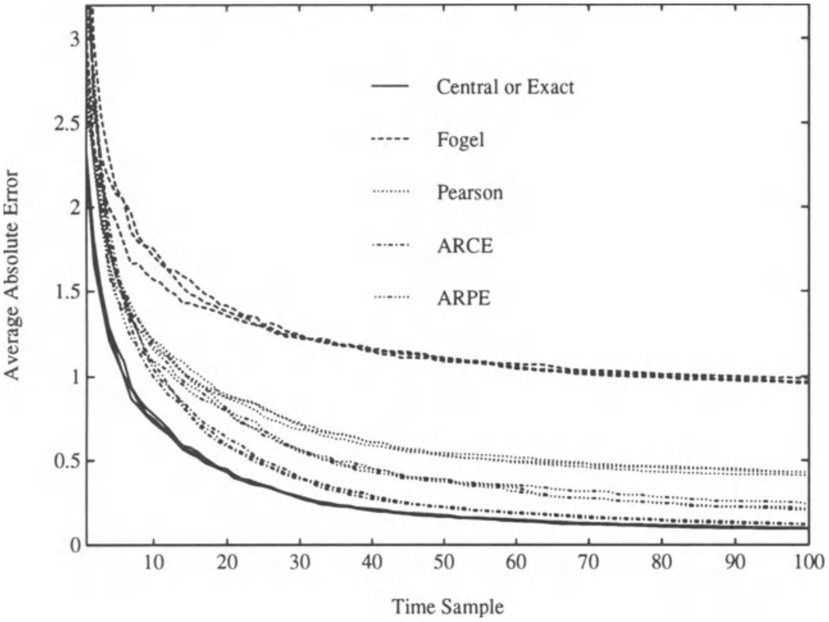


FIGURE 6.4. Average amplitude of the *PUI*s of the three parameters.

the optimal central estimate. Also the *ARPE* performs satisfactorily while the Pearson and Fogel–Huang estimates have larger errors.

From Fig. 6.4 it is even more evident that the parameter uncertainty derived from Pearson's and Fogel–Huang's algorithms is far worse than that obtained with the other algorithms.

Since the *ARPE* algorithm is one of those that requires less computational effort and less information on the error structure, it is probably the most convenient for many practical applications. Furthermore, it can easily deal with time varying systems as outlined in the preceding section.

ACKNOWLEDGMENTS. This work was partially supported by the grants MURST-40% and MURST-60% of the Italian Ministry of University and Scientific Research.

REFERENCES

1. F. C. Schweppe, *Uncertain Dynamic Systems*, Prentice Hall, Englewood Cliffs, N.J. (1973).
2. M. K. Smit, *Measurement* **1**, 181 (1983).
3. J. P. Norton, *Automatica* **23**, 497 (1987).

4. E. Walter and H. Piet-Lahanier, *Math. Comp. Simul.* **32**, 449 (1990).
5. M. Milanese, in: *Robustness in Identification and Control* (M. Milanese, R. Tempo and A. Vicino, eds.), Plenum, New York, pp. 3–24 (1989).
6. E. Walter and H. Piet-Lahanier, in: *Preprints 9th IFAC/IFORS Symp. on Identification and System Parameter Estimation*, Budapest, p. 763 (1991).
7. E. Walter and H. Piet-Lahanier, *Int J. Adapt. Control Signal Process.* **8**, 5 (1994).
8. E. Walter and H. Piet-Lahanier, *Recursive robust minimax estimation* in: *Bounding Approaches to System Identification*, (M. Milanese *et al.*, eds.), Plenum Press, New York (1996).
9. R. Tempo, IBC, in *1992 American Control Conference*, Chicago, IL, p. 237 (1992).
10. V. Broman and M. J. Shensa, *Math. Comput. Simul.* **32**, 469 (1990).
11. S. H. Mo and J. P. Norton, *Math. Comput. Simul.* **32**, 481 (1990).
12. H. Piet-Lahanier and E. Walter, *Math. Comput. Simul.* **32**, 495 (1990).
13. E. Fogel and Y. F. Huang, *Automatica* **18**, 229 (1982).
14. R. K. Pearson, *SIAM J. Matrix An. Appl.* **9**, 513 (1988).
15. G. Belforte and T. T. Tay, *IEEE Trans. Autom. Control* **AC-38**, 1273 (1993).
16. G. Belforte, *Int. J. Adapt. Control Signal Process.* **9**, 97 (1995).
17. G. Belforte, Y. T. Teo and T. T. Tay, in: *Proceedings of the SICICI'92 Singapore International Conference on Intelligent Control and Instrumentation,* Singapore, p. 945 (1991).
18. J. P. Norton and S. H. Mo, *Math. Comput. Simul.* **32**, 527 (1990).
19. H. Piet-Lahanier and E. Walter, in: *IEEE International Symposium on Circuits and Systems*, Chicago, Illinois, p. 782 (1993).
20. G. Belforte, B. Bona, and V. Cerone, *Math. Comp. Simul.* **32**, 561 (1990).
21. V. Cerone, in: *Proceedings of the 9th IFAC/IFORS Symposium on Identification and System Parameter Estimation*, Budapest, Hungary, p. 1518 (1991).
22. V. Cerone, in: *Bounding Approaches to System Identification* (M. Milanese *et al.*, eds.) Plenum Press, New York (1996).
23. G. Belforte, *Syst. Control Lett.* **19**, 425 (1992).
24. H. Piet-Lahanier, S. M. Veres, and E. Walter, *Math. Comput. Simul.* **34**, 515 (1992).
25. S. M. Veres and J. P. Norton, *IEEE Trans. Autom. Control* **AC-36**, 474 (1991).
26. G. Belforte, B. Bona, and V. Cerone, *Automatica* **26**, 887 (1990).
27. L. Pronzato, E. Walter, and H. Piet-Lahanier, in: *Proceedings of the 28th IEEE Conference on Decision and Control*, Tampa, FL, p. 1952 (1989).
28. B. N. Pshenichnyy and V. G. Pokotilo, *Eng. Cybern.* **21**, 94 (1983).
29. M. Milanese and G. Belforte, *IEEE Trans. Autom. Control* **AC-27**, 408 (1982).

7

Transfer Function Parameter Interval Estimation Using Recursive Least Squares in the Time and Frequency Domains

P.-O. Gutman

ABSTRACT

A bank of recursive least squares (RLS) estimators is proposed for the estimation of the uncertainty intervals of the parameters of an equation error model (or RLS model), where the equation error is assumed to lie between a known upper and lower bound. It is shown that the off-line least squares method gives the maximum and minimum parameter values that could have produced the recorded input-output sequence. By modifying the RLS estimator in two ways, it is possible to recursively compute inner and outer bounds of the uncertainty intervals. It is shown that the inner bound is asymptotically tight. It is demonstrated that transfer function parameter intervals can also be estimated, by applying the method to measured frequency function data.

P.-O. GUTMAN • Faculty of Agricultural Engineering, Technion–Israel Institute of Technology, Haifa 32000, Israel.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

7.1. INTRODUCTION

The motivation of this chapter is a desire to make the Horowitz robust control design method[1] adaptive. In the Horowitz method it is suitable to describe the plant uncertainty as transfer function value sets, or alternatively, as plant parameter sets or intervals. The resulting controller consists of a linear time invariant feedback compensator and prefilter. In Yaniv, Gutman, and Neumann[2] a method is suggested how to change, on-line, the parameters of such a robust controller, when it becomes known that the plant parameters each belong to a smaller interval than the original interval on which the design is based. Combined with a parameter interval estimator, an adaptive robust controller is created, based on the principle of robust certainty equivalence,[3] see Fig. 7.1. In Gutman[3] an example from Yaniv, Gutman, and Neumann[2] is simulated with essentially the parameter interval estimator presented here. Conventional adaptive controllers are, on the other hand, in general designed according to the certainty equivalence[4] principle, whereby the adaptation is based on a point estimate of the parameter vector.

The vast literature about the parameter set, or set-membership estimation problem is covered in several informative surveys.[5,6,7] A most attractive method is the one developed by Walter and Piet-Lahanier[8,28] that gives an exact polyhedral description of the feasible parameter set. One might surmise that for most linear problems, it obviates the need for any other method. However, approximants have been proposed, like for instance bounding ellipsoids.[9,10] Therefore, the little idea in this note, originally presented at an IFAC conference,[25] might evoke some interest.

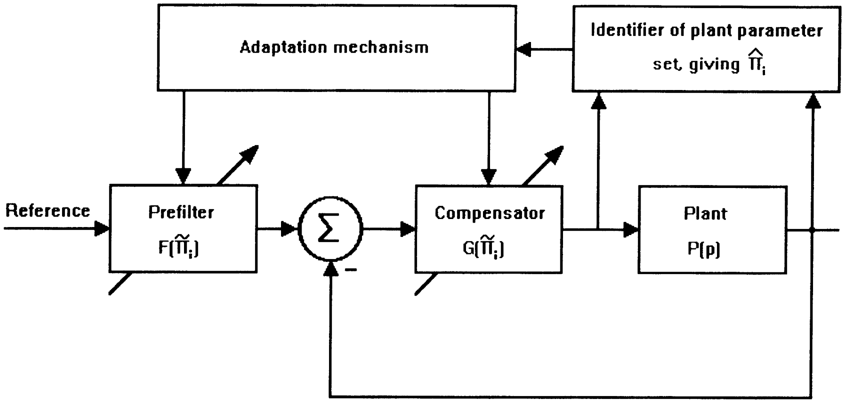


FIGURE 7.1. Block diagram of control system with adaptive feedback and prefilter control, p is the plant parameter vector, $\hat{\Pi}_i$ is the plant parameter uncertainty set estimate, $\tilde{\Pi}_i$ is the plant parameter set on which the design is based, and $\tilde{\Pi}_i \supseteq \hat{\Pi}_i$.

It is shown that a bank of modified RLS estimators, under weak assumptions, gives asymptotically tight inner bounds of the feasible parameter intervals for linear equation error (RLS) models. Another variation yields nontight outer bounds. Hence, the method belongs to the class of inner-bounding and outer-bounding orthotopic estimators.(6,11) It is also shown that the method can be applied directly on frequency function measurement data. The estimator of Kosut,(12) where parameter sets of an output error model with unstructured uncertainty are estimated via the discrete fourier transform (DFT) of the input-output sequences, bears some resemblance to the one analyzed here.

Estimates of value sets in the frequency domain would also serve the initially stated purpose.(13) Goodwin and Salgado(14) estimate value sets directly via a probabilistic RLS approach. LaMaire *et al.*,(15) Wahlberg and Ljung(16) calculate error bounding functions in the frequency domain. The chapter is organized as follows: In section 7.2 the off-line and on-line algorithms are presented and analyzed. Section 7.3 contains four simulated examples. In the last example, the method is applied to frequency function data. In the Conclusions (section 7.4) the proposed algorithm is related to other methods, and advantages and disadvantages are discussed.

7.2. ESTIMATION OF PARAMETER INTERVALS

7.2.1. Off-line Parameter Interval Estimates

Among various algorithms for plant parameter estimation,(17,18) and for recursive estimation,(19,20) we find the popular least squares (LS), and the recursive least squares (RLS) and its relatives. In the above references, conditions on the model, input sequence, and noise sequence are stated for the RLS estimates to converge asymptotically, with and without bias with respect to the true parameter values.

The LS and RLS algorithms will be the point of departure in this chapter. Let the "true" process be

$$y(t) = \theta^T \varphi(t) + v(t) \tag{7.1}$$

where $\theta = (a_1 \dots a_n\, b_1 \dots b_m)^T$ is the parameter vector, and $\varphi(t) = [-y(t-1) \dots -y(t-n)u(t-1) \dots u(t-m)]^T$ is the vector of measured lagged input-output data. The measured input signal is $\{u(t)\}$ and the measured output signal is $\{y(t)\}$. The running sample index is $t = 1, 2, 3, \dots$ The equation error $\{v(t)\}$ includes all effects of measurements noises, mismodelling, disturbances and other uncertainties in the given description of the data. Then the LS estimate, $\hat{\theta}^{LS}(N)$ at time $t = N$ is given by:(17)

$$\hat{\theta}^{\mathrm{LS}}(N) = \left[\frac{1}{N}\sum_{t=1}^{N} \varphi(t)\varphi(t)^T\right]^{-1} \frac{1}{N}\sum_{t=1}^{N} \varphi(t)y(t) \tag{7.2}$$

Introduce the matrix $R(N)$

$$R(N) = \frac{1}{N}\sum_{t=1}^{N} \varphi(t)\varphi(t)^T \tag{7.3}$$

The pure RLS estimate $\hat{\theta}(t)$ is given by:(19)

$$\hat{\theta}(t) = \hat{\theta}(t-1) + P(t)\varphi(t)[y(t) - \hat{\theta}(t-1)^T\varphi(t)] \tag{7.4}$$

$$P(t) = P(t-1) - \frac{P(t-1)\varphi(t)\varphi(t)^T P(t-1)}{1 + \varphi(t)^T P(t-1)\varphi(t)}\,. \tag{7.5}$$

For suitable initial conditions of the RLS algorithm,(19) $\hat{\theta}(t) = \hat{\theta}^{\mathrm{LS}}(t)$ and $P(t) = R(t)^{-1}/t$ for all t; for any positive definite $P(0)$, $\hat{\theta}(t)$ and $P(t)$ converge to $\hat{\theta}^{\mathrm{LS}}(t)$ and $R(t)^{-1}/t$, respectively. For constant θ, the estimate $\hat{\theta}(t)$ converges to θ under ideal conditions.(19) Also under ideal conditions,(17) $P(t)$ is the normalized variance of the estimate:

$$\lambda_0 P(t) = E\{(\hat{\theta}(t) - \theta)(\hat{\theta}(t) - \theta)^T\},$$

where λ_0 is the variance of $v(t)$.

Like the pure RLS estimator, most algorithms give point and variance estimates only. Under ideal conditions, the variance matrix $P(t)$ of the RLS estimator could be used for estimating a likely parameter set. In practice, however, the updating of $P(t)$ in Eq. (7.5) is modified. This would include a forgetting factor, dead zone, or other devices to keep $tr(P(t))$ constant control of $P(t)$,(21,22) e.g., to enhance the tracking ability of the estimator. Then $P(t)$ does not represent the variance of the estimate. It is assumed(3,5–12,23) that the equation error in Eq. (7.1) is bounded:

$$|v(t)| \le V(t) \le V \quad \forall t \tag{7.6}$$

with $V(t)$ or V known. This assumption may be used to compute parameter interval bounds.

It is easy to show(17) that the LS estimation error, $\tilde{\theta}(N) = \hat{\theta}^{\mathrm{LS}}(N) - \theta$ is given by

$$\tilde{\theta}(N) = [R(N)]^{-1}\frac{1}{N}\sum_{t=1}^{N} \varphi(t)v(t) \tag{7.7}$$

Comparing with Eq. (7.2), notice that Eq. (7.7) can be implemented with $\{v(t)\}$ given in the recursive form Eqs. (7.4 and 7.5). With $v(t)$ replacing $y(t)$, and $\tilde{\theta}$ replacing $\hat{\theta}$ in Eq. (7.4), $P(t)$ given by Eq. (7.5):

$$\tilde{\theta}(t) = \tilde{\theta}(t-1) + P(t)\varphi(t)[v(t) - \tilde{\theta}(t-1)^T\varphi(t)] \tag{7.8}$$

The sequence $\{v(t)\}$ is not known however. Hence the estimation error can not be found. It is, however, easy to dream up the worst possible equation error sequence $\{v(t)\}$, satisfying Eq. (7.6), that will yield a maximal upper bound for $|\tilde{\theta}_i(N)|$. For $i = 1, 2, \ldots, (n+m)$, let

$$v_i(t) = V(t) sign\,\{[R(N)]^{-1}\,\varphi(t)\}_{\mathrm{i}} \tag{7.9}$$

and

$$E(i,N) = [R(N)]^{-1}\frac{1}{N}\sum_{t=1}^{N}\varphi(t)v_i(t). \tag{7.10}$$

Then, for each component i,

$$|\tilde{\theta}_i(N)| \le E_i(i,N) \tag{7.11}$$

or, equivalently,

$$\hat{\theta}_i^{\mathrm{LS}}(N) - E_i(i,N) \le \theta_i \le \hat{\theta}_i^{\mathrm{LS}}(N) + E_i(i,N). \tag{7.12}$$

Define

$$M_1 = \{\Theta\colon \hat{\theta}_i^{\mathrm{LS}}(t) - E_i(i,t) \le \theta_i \le \hat{\theta}_i^{\mathrm{LS}}(t) + E_i(i,t) \quad \forall i,\ \forall t \le N\} \tag{7.13}$$

Clearly, M_1 defines the maximal parameter intervals, in which those parameter components are to found that are able to produce the recorded input-output sequence, assuming the model Eqs. (7.1 and 7.6).

Assume[17] that the input $\{u(t)\}$ is quasi-stationary such that $R(N) \to \mathrm{R}^*$, as $N \to \infty$. Assume further that all elements of $\varphi(t)$ are bounded and quasi-stationary, then $H(1/N)\ \Sigma_{t=1}^{N}\ \varphi(t)v_i(t) \to h_i^*$, $\forall i$ as $N \to \infty$. Hence $E(i, N)$ converges as $N \to \infty$.

Equations (7.9–7.11) are suitable for off-line implementation. For on-line use, $P(t)$ can be compute using Eq. (7.5), while the expanding matrix $\Phi(t) = [\varphi(1)\varphi(2)\ldots\varphi\,(t)]$ has to be saved tn order to compute $\mathrm{v}_i(t)$. This may constitute an unacceptable memory burden. The next subsection will treat on-line approximations of Eqs. (7.9 and 7.10).

It may be noticed that V in Eq. (7.6) may be estimated from the residual: Assume that a LS parameter estimate has been computed, $\hat{\theta}(N)$. Compute the residual

$$\varepsilon(t) = y(t) - \varphi^T(t)\hat{\theta}(N) \quad t \le N \tag{7.14}$$

Then $\max_t |\varepsilon(t)|$ may serve as an estimate of V. It is expected that this estimate will be conservative since $\hat{\theta}(N)$ is, in general, a biased estimate of θ.

7.2.2. On-Line Parameter Interval Estimates

In Algorithm 2 of Gutman[3] a bank of RLS estimators was suggested to estimate parameter intervals in essentially the following way: Let, for each i,

$$v_i(t) = V(t)\text{sign}\{[R(t)]^{-1}\varphi(t)\}_i \tag{7.15a}$$

$$D(i,N) = [R(N)]^{-1} \frac{1}{N} \sum_{t=1}^{N} \varphi(t) v_i(t) \tag{7.15b}$$

Comparing with Eqs. (7.9) and (7.10), it is clear that $D_i(i,N) \le E_i(i,N)$ since $\{v_i(t)\}$ is chosen to maximize $E_i(i,N)$. However, Eq. (7.15) is suitable for recursive implementation via (Eq. 7.8), with $P(t)$ given by Eq. (7.5):

$$D(i,t) = D(i,t-1) + P(t)\varphi(t)[v_i(t) - D(i,t-1)^T\varphi(t)] \quad \forall i \tag{7.16}$$

The selection of v in Eq. (7.15a) simply means a "local in t" maximization of $D_i(i,N)$ in Eq. (7.15b and 7.16). Contrast this to the "*a posteriori* at $t = N$" maximization of $E_i(i,N)$ in Eq. (7.10) via the section of v in Eq. (7.9).

Assume that $R(N) \to R^*$ as $N \to \infty$. Then. for every i, $v_i(t) \to v_i(t)$, and hence $D_i(i,t) \to E_i(i,t)$ as $t \to \infty$.

We conclude that $D_i(i,t)$ is a lower, progressively closer bound for the ith parameter interval extension $E_i(i,t)$. An upper bound for $E_i(i,t)$ can also be found.

Let $M = \{m_{ij}\}$ be a matrix whose elements are m_{ij}. Define $\text{abs}(M) = \{|m_{ij}|\}$. Let the definition also hold for vectors. Let

$$F(N) = \text{abs}([R(N)]^{-1}) \frac{1}{N} \sum_{t=1}^{N} \text{abs}(\varphi(t))V(t) \tag{7.17}$$

Comparing with Eq. (7.10) it is immediately clear that $E_i(i,N) \le F_i(N)$ since all elements on the right hand side of Eq. (7.17) are non-negative. Moreover, assuming that $R(N) \to R^*$ as $N \to \infty$, then, of course, abs $([R(N)]^{-1})$ also converges. Assume further that all elements of $\varphi(t)$ are bounded and quasi-stationary, then $(1/N)\ \Sigma_{t=1}^{N}$ $\text{abs}(\varphi(t))V(t)$ converges. Hence $F(N)$ converges as $N \to \infty$.

The convergence limit of $F(N)$ does not seem possible to find without additional specific assumptions, which have to be validated in each particular case. Clearly, from Eqs. (7.10 and 7.17)

$$0 \leq E_i(i,N) - E_i(i,N-1) \leq F_i(N) - F_i(N-1).$$

Assume, for instance, that when $N \to \infty$

$$E\{E_i(i,N) - E_i(i,N-1)\} = xE\{F_i(N) - F_i(N-1)\}$$

where $E\{\cdot\}$ means expected value, and $x \in [0,1]$. Then, with F_0 some constant vector

$$F(N) \to E_i(i,N)/x + F_0, \quad \text{as} N \to \infty.$$

The computation of $F(N)$ is easily made recursive

$$\psi(t) = \psi(t-1) + \text{abs}(\varphi(t))V(t) \tag{7.18a}$$

$$F(t) = \text{abs}(P(t))\psi(t) \tag{7.18b}$$

with $P(t)$ given by Eq. (7.5) and $\psi(0) = 0$.

7.2.3. Summary

From the recorded input-output data and an assumed RLS model of Eq. (7.1) with bounded equation error of Eq. (7.6), the maximally possible parameter intervals of Eq. (7.12) for each parameter θ_i,

$$\theta_i \in [\hat{\theta}_i^{\text{LS}}(N) - E_i(i,N), \hat{\theta}_i^{\text{LS}}(N) + E_i(i,N)] \tag{7.19}$$

have been computed, with $E(i,N)$ given by Eqs. (7.9) and (7.10). Since $E(i,N)$ is not conveniently computed in a recursive way, recursively computable inner, $D(i,t)$ Eqs. (7.15a and 7.16), and outer, $F(t)$ Eq. (7.18), bounds were found:

$$D_i(i,N) \leq E_i(i,N) \leq F_i(N). \tag{7.20}$$

Under weak assumptions, $D(i,N)$, $E(i,N)$, and $F(N)$ converge to their respective limits as $N \to \infty$, with $D(i,N)$ and $E(i,N)$ sharing the same limit.

7.3. EXAMPLES

EXAMPLE 1: Let the "true" process model be given by

$$y(t) = a_1 y(t-1) + b_1 u(t-1) + v(t-1) \tag{7.21}$$

where

$$\theta = (a_1 \ \ b_1)^T = (0.5 \ \ 1)^T$$

is the parameter vector, and

$$\varphi(t) = [y(t-1)\ u(t-1)]^T$$

is the vector of measured lagged input-output data. The input signal $\{u(t)\}$ is chosen to be a uniformly distributed random variable $\in [-1,1]$ independent for each t. The initial condition $y(0)$ was set to 0. The equation error $\{v(t)\}$ is chosen to be a uniformly distributed random variable $\in [-1,1]$ independent for each t, and independent of $\{u(t)\}$. It is assumed known that in Eq. (7.6), $V(t) = V = 1$.

The system is simulated for $t = 1, 2, \ldots, 100$. For one particular simulation, the LS estimate, Eq. (7.2 or 7.4), becomes $\hat{\theta}(100) = (0.5341\ 0.9575)^T$. Such a good estimate is expected because of the nature of $\{v(t)\}$ and $\{u(t)\}$.

From Eq. (7.10), $E(1,t)$ and $E(2,t)$ are computed. The final values are $E_1(1,100) = 0.9317$, and $E_2(2, 100) = 1.6259$, signifying that $a_1 \in [0.5341 \pm 0.9341]$ and $b_1 \in [0.9575 \pm 1.6259]$. From Eq. (7.16), $D(1,t)$ and $D(2,t)$ are computed. The final values are $D_1(1,100) = 0.9102$, and $D_2(2,100) = 1.5974$. From Eq. (7.18), $F(t)$ is computed. The final values $F_1(100) = 0.9369$, and $F_2(100) = 1.6360$.

Fig. 7.2 displays $\hat{a}_1(t)$, $\hat{a}_1(t) \pm E_1(1,t)$, $\hat{a}_1(t) \pm D_1(1,t)$, and $\hat{a}_1(t) \pm F_1(t)$. In Fig. 7.3. $\hat{b}_1(t)$, $\hat{b}_1(t) \pm E_2(2,t)$, $\hat{b}_1(t) \pm D_2(2,t)$, and $\hat{b}_1(t) \pm F_2(t)$ are displayed. From the figures it is seen that Eq. (7.20) holds; $D(i,t)$ seems to be a good approximation of $E(i,t)$ at all times, while $F(t)$ is satisfactory at "steady state" for $t > 30$.

Although the computed parameter intervals may seem exaggerated, worst case parameter combinations, with either a_1 or b_1 at the endpoint of its respective interval, may yield the observed data. In the simulated case, not both a_1 and b_1 may

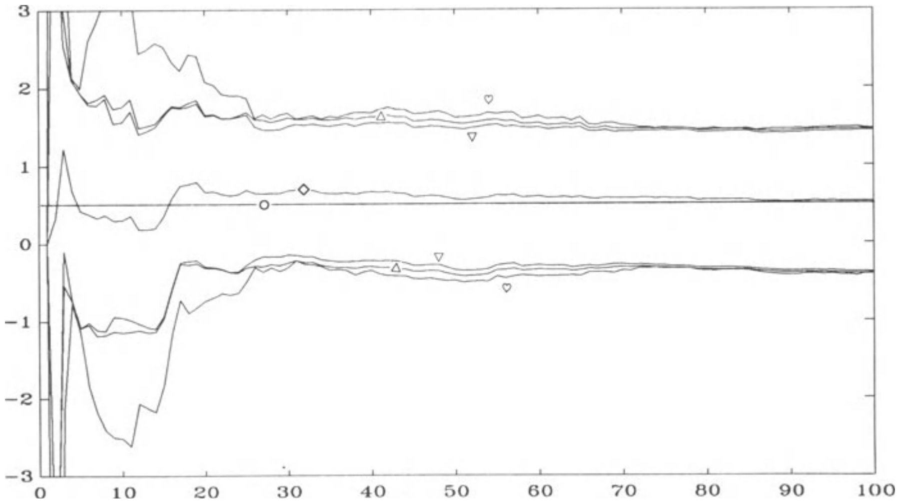


FIGURE 7.2. Example 1 results: $a_1(t)$ (○), $\hat{a}_1(t)$ (◇), $\hat{a}_1(t) \pm E_1(1,t)$ (△), $\hat{a}_1(t) \pm D_1(1,t)$ (▽), and $\hat{a}_1(t) \pm F_1(t)$ (♡).

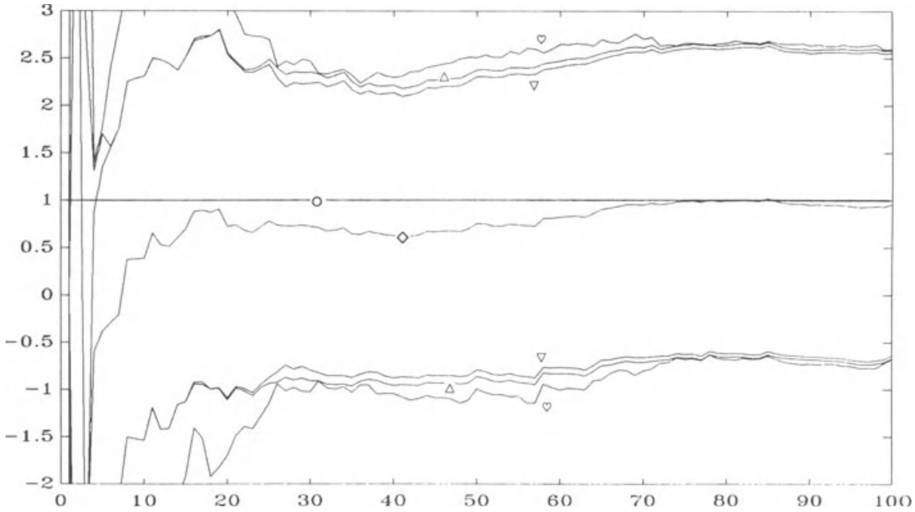


FIGURE 7.3. Example 1 results: $b_1(t)$ (○), $\hat{b}_1(t)$ (◇), $\hat{b}_1(t) \pm E_2(2,t)$ (△), $\hat{b}_1(t) \pm D_2(2,t)$ (▽), and $\hat{b}_1(t) \pm F_2(t)$ (♡).

be at an interval endpoint. This is reflected by the low values of $D_2(1, 100) = 0.03072$, and $D_1(2,100) = 0.0439$. Sharper bounds might be obtained by a linear transformation of the $\{y(t)\ u(t-1)\}$ space, yielding estimates of linear combinations of a_1 and b_1.

Example 2: This example is extreme in the sense that $v(t)$ is highly dependent on $\varphi(t)$. The process is the same as in Example 1, Eq. (7.21), with the exception that $\{v(t)\} = \{u(t)\}$. One simulation is performed. The RLS estimate is very good in the sense that the prediction error is zero: $\hat{\theta}(100) = (0.5\ 2.0)^T$. The knowledge of the size of $v(t)$ gives, however, the opportunity to find other parameter values that could have generated the data.

The bounds $E(1,t)$ and $E(2,t)$ were computed, with the final values $E_1(1,100) = 0.6880$, and $E_2(2,100) = 1.6257$. Consequently, $a_1 \in [0.5 \pm 0.6880]$ and $b_1 \in [2.0 \pm 1.6257]$. The large parameter intervals are justified; the "correct" θ is found in the estimated parameter set. The inner bounds $D(1,t)$ and $D(2,t)$ are computed. The final values are D_1 (1,100) = 0.6815, and D_2 (2, 100) = 1.6089. The outer bound $F(t)$ was computed, with final values $F_1(100) = 0.6933$, and $F_2(100) = 1.6384$.

Fig. 7.4 displays $\hat{a}_1(t)$, $\hat{a}_1(t) \pm E_1(1,t)$, $\hat{a}_1(t) \pm D_1(1,t)$, and $\hat{a}_1(t) \pm F_1(t)$. Fig. 7.5 displays $\hat{b}_1(t)$, $\hat{b}_1(t) \pm E_2(2,t)$, $\hat{b}_1(t) \pm D_2(2,t)$, and $\hat{b}_1(t) \pm F_2(t)$. From Fig. 7.5 it is seen that Eq. (7.20) holds, and that $D(i,t)$ and $F(t)$ seem to be good approximations of $E(i,t)$.

Example 3: In this example, the more "realistic" situation of a first order continuous-time model with step-wise jumping parameters is investigated. The

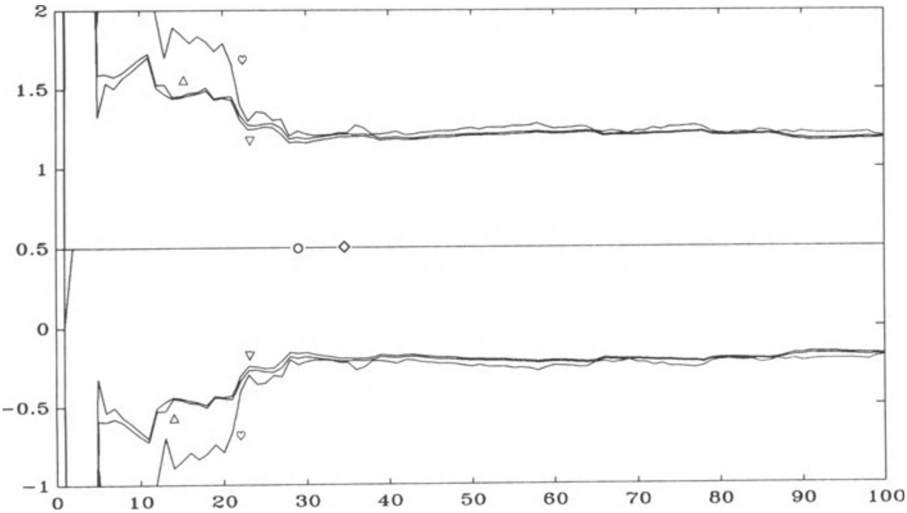


FIGURE 7.4. Example 2 results: $a_1(t)$ (○), $\hat{a}_1(t)$ (◇), $\hat{a}_1(t) \pm E_1(1,t)$ (△), $\hat{a}_1(t) \pm D_1(1,t)$ (▽), and $\hat{a}_1(t) \pm F_1(t)$ (♡).

example is adapted from Gutman[3] To represent the parameter intervals, the inner bounds $D(i,t)$ are computed with a bank of modified RLS estimators, based on Canudas de Wit,[23] of a type that could be used in practice. An aim of this example is to illustrate how the parameter interval estimator fares together with a modified RLS.

The equations of the modified RLS identifier are:

$$\hat{\theta}_t = \hat{\theta}_{t-1} + \lambda_t P_{t-1}\varphi_t(|e_t| - W_t)\text{sign}(e_t) \tag{7.22a}$$

$$\lambda_t = \alpha_t/\gamma_t \tag{7.22b}$$

$$e_t = (y_t - \hat{\theta}_{t-1}\varphi_t) \tag{7.22c}$$

$$\alpha_t = \begin{cases} 0 & \text{if } \gamma_t = 0 \text{ or } |e_t| \le W_t \\ 1 & \text{otherwise} \end{cases} \tag{7.22d}$$

$$\gamma_t = \varphi_t^T P_{t-1}\varphi_t \tag{7.22e}$$

$$P_t = P_{t-1} - \lambda_t P_{t-1}\varphi_t\varphi_t^T P_{t-1}(1 - W_t|e_t|) + (1 - \alpha_t)\cdot f\cdot P_{t-1} \tag{7.22f}$$

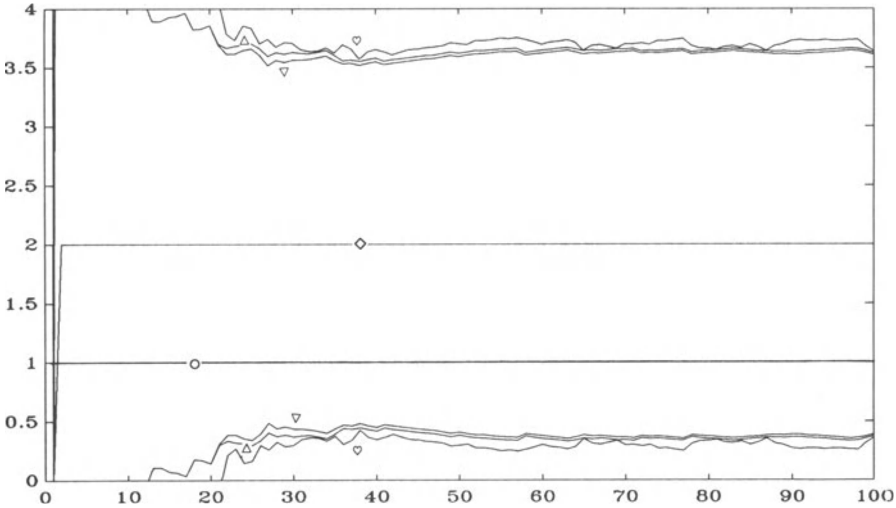


FIGURE 7.5. Example 2 results: $b_1(t)$ (○), $\hat{b}_1(t)$ (◇), $\hat{b}_1(t) \pm E_2(2,t)$ (△), $\hat{b}_1(t) \pm D_2(2,t)$ (▽), and $\hat{b}_1(t) \pm F_2(t)$ (♡).

This algorithm disregards redundant data, and prevents $P_t \to 0$ when such data is received.

Given the first order dynamic system as a function of time τ:

$$\dot{z}(\tau) = -pz(\tau) + ku(\tau) \tag{7.23a}$$

$$y(\tau) = z(\tau) + n(\tau) \tag{7.23b}$$

where $|n(\tau)| \leq 0.01$ is a uniformly distributed measurement noise. The unknown parameters k and p, which occasionally change stepwise, have to be estimated. *A priori* parameter bounds are $k \in [1,10]$, and $p \in [0.5,3]$. By low-pass filtering(24) $y(\tau)$ and $u(\tau)$, one gets an identification model of Eq. (7.1) with $|v_t|$ bounded, whose parameter vector θ is invertibly related to k and p. Let

$$\dot{u}_1(\tau) = -(1/c)u_1(\tau) + (1/c)u(\tau) \tag{7.24a}$$

$$\dot{y}_1(\tau) = -(1/c)y_1(\tau) + (1/c)y(\tau) \tag{7.24b}$$

where the filter time constant is chosen $c = 0.1$. The identification model is then

$$y(\tau) = -a_1 y_1(\tau) + b_1 u_1(\tau) + v(\tau) \tag{7.25}$$

with

$$a_1 = pc - 1$$

$$b_1 = kc$$

$$v(\tau) = n(\tau) + a_1 n_1(\tau)$$

and

$$\dot{n}_1(\tau) = -(1/c)n_1(\tau) + (1/c)n(\tau)\,.$$

Clearly, $|n_1(\tau)| \leq \max(n(\tau))$ since $n_1(\tau)$ equals $n(\tau)$ transmitted through a first order filter. Using the lower limit of p

$$\max(|a_1|) = |0.5 \cdot 0.1 - 1| = 0.95$$

and

$$|v(\tau)| \leq \max|n(\tau)(1 + |a_1|)| = 0.0195.$$

Notice that Eq. (7.25) is valid at all times τ. Hence $y(\tau)$, $y_1(\tau)$, and $u_1(\tau)$ may be sampled at arbitrary times, for instance, with non-uniform sampling periods. Let

$$\theta = (a_1 \; b_1)^T$$

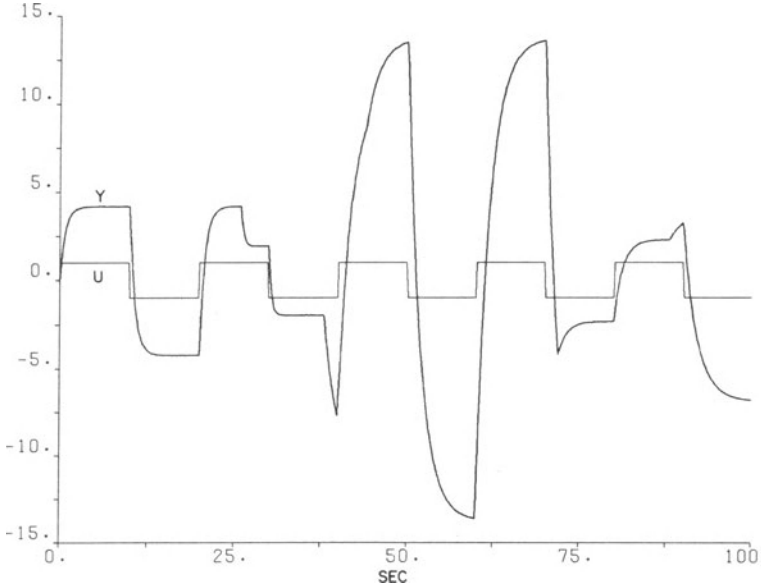


FIGURE 7.6. Input signal $u(\tau)$ and output signal $y(\tau)$ for Example 3. (The noise is hardly noticeable.)

and

$$\varphi_t = (-y_1(\tau)\ u_1(\tau))^T$$

where the index t is the running sample index as in Eq. (7.1).

The system is excited with a square wave input $u(\tau)$ (see Fig. 7.6). According to Figs.7.7 and 7.8, k and p are changing with random steps at random times. Because of the parameter changes, the output $y(\tau)$ varies wildly (see Fig. 7.6).

The modified RLS identifier of Eq. (7.22) is used as a conventional point estimator to estimate $(\hat{k}, \hat{p})$ in the following manner: W_t is used as a tuning parameter and set to 0.013. In Eq. (7.22d) "$\gamma_t = 0$" is replaced by "$|\gamma_t| \leq \varepsilon$" with $\varepsilon = 10^{-6}$. In Eq. (7.22f), "$W_t|e_t|$" is replaced by "$W_t/\max(\varepsilon,|\varepsilon_t|)$", and f is set to 0.05. A uniform sampling period of 0.1 seconds is used.

To compute $D(i,t)$, $i = 1, 2$ for the interval bounds $\hat{k}_{\min}$, $\hat{k}_{\max}$, $\hat{p}_{\min}$, and $\hat{p}_{\max}$, two copies of Eq. (7.22a) are used with $\hat{\theta}_t$ replaced by $D(i,t)$, $i = 1, 2$, and e_t replaced by $(v_i(t) - D(i,t-1)\varphi(t))$, respectively (see Eq. (7.16)).

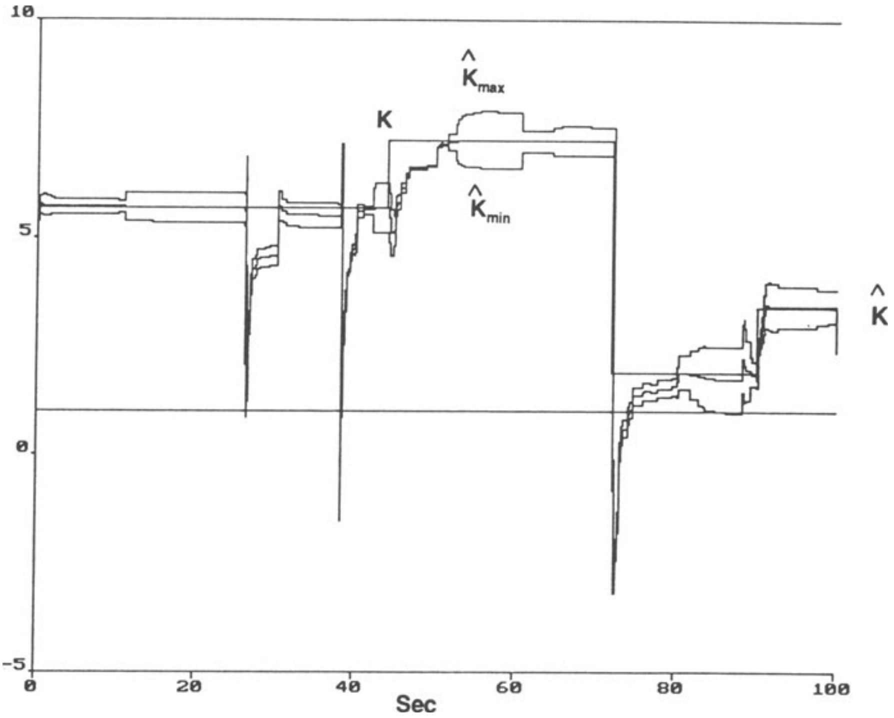


FIGURE 7.7. The true parameter k and its *a priori* bounds, the point estimate $\hat{k}$, and the estimates of the interval bounds $\hat{k}_{\min}$ and $\hat{k}_{\max}$ for Example 3.

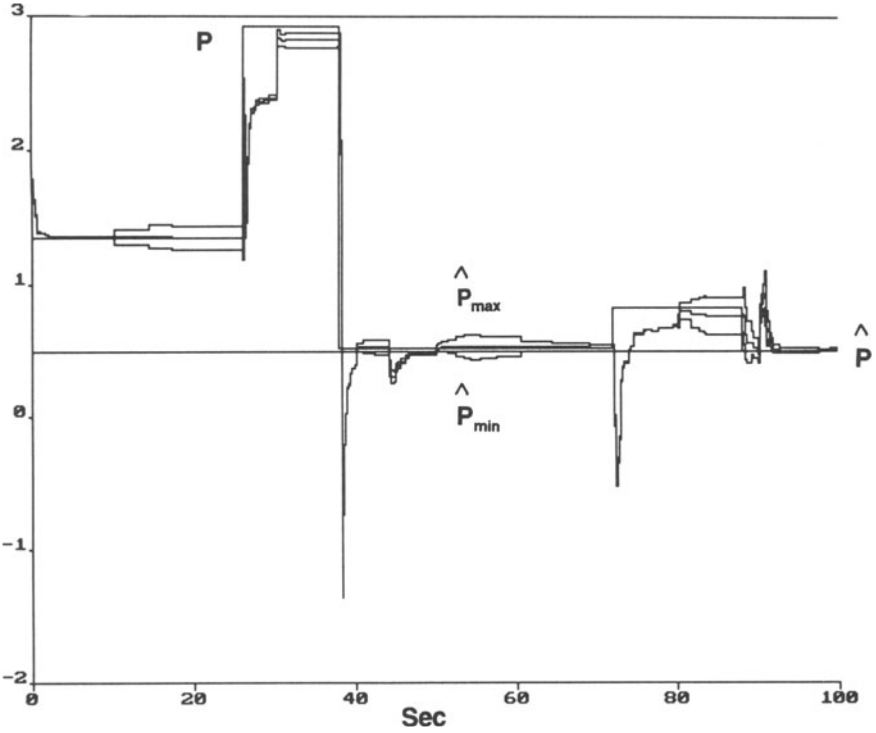


FIGURE 7.8. The true parameter p and its *a priori* bounds, the point estimate $\hat{p}$, and the estimates of the interval bounds $\hat{p}_{\min}$ and $\hat{p}_{\max}$ for Example 3.

Using Eq. (7.15a), $v_i(t)$ is computed with $[R(t)]^{-1}$ replaced by $tP(t)$ and $P(t)$ taken from Eq. (7.22f). $V(t)$ was set to $W_t + \max|v(\tau)| = 0.0325$ to account for both the equation error $v(\tau)$ in Eq. (7.25) and the dead zone W_t in Eq. (7.22).

The identification results are shown in Fig. 7.7, where k, $\hat{k}$, $\hat{k}_{\min}$, $\hat{k}_{\max}$, and the *a priori* bounds are plotted, and in Fig. 7.8 where p, $\hat{p}$, $\hat{p}_{\min}$, $\hat{p}_{\max}$ and the *a priori* bounds are displayed. The estimates are not projected into the *a priori* given parameter set. The estimates converge to their approximate steady state values after steps in $u(\tau)$, i.e., when $u(\tau)$ excites the system. The point estimate $(\hat{k}, \hat{p})$ is of good quality, but exhibits an occasional bias. In most cases, the parameter interval estimates include the true parameters when steady state has been reached. There is cross talk between the parameter estimates: a jump in p influences $\hat{k}$, $\hat{k}_{\min}$, $\hat{k}_{\max}$, and (less so) vice versa. The upper parameter interval estimate F from Eq. (7.18), with $P(t)$ taken from Eq. (7.22) and $V(t)$ as above, diverged since $P(t) \not\to 0$ in Eq. (7.22).

If the parameter interval estimates in this example are to be used for the adaptation of a robust controller,(2) then the robust controller should be based on the full *a priori* parameter uncertainty whenever an abrupt parameter change is sensed. Only when the estimator has reached steady state, could adaptation take place using the interval estimates with appropriate safety margins.

EXAMPLE 4: Application of the method on measured frequency response data. Consider the same process as in Example 1, Eq. (7.21), with the same $y(0)$, $\{u(t)\}$ and $\{v(t)\}, t = 1, 2, \ldots, N$, and $N = 100$. Let $G(q) = B(q)/A(q)$ be the true input-output transfer function of Eq. (7.21), with q denoting the forward shift operator; $qy(t) = y(t + 1)$.

An empirical transfer function estimate (ETFE)(17) $\hat{G}(e^{j\omega_k})$, with $\omega_k = 2\pi k/N$ [rad/s], and $k = 0, 1, \ldots, N - 1$, is obtained by dividing $Y_N(\omega)$, defined as the discrete Fourier transform (DFT) of $\{y(t)\}$, with $U_N(\omega)$, the DFT of $\{u(t)\}$, $t = 1, 2, \ldots, N$. Moreover, let $V_N(\omega)$ be the DFT of $\{v(t)\}$, $t = 1, 2, \ldots, N$. The ETFE, $\hat{G}(e^{j\omega_k})$, is considered as the measured data in this example.

Let $\Theta = (a_1\ b_1)^T$ as in Eq. (7.21). Define $\phi(q) = [\hat{G}(q)\ 1]^T$ and $\psi(q) = q\hat{G}(q)$. Lilja(26) (and references given therein, e.g., Levy(27)) shows that

$$\psi(e^{j\omega_k}) = \Theta^T \phi(e^{j\omega_k}) + w(j\omega_k) \tag{7.26}$$

constitutes a LS model in the frequency domain with the equation error $w(j\omega_k)$ (see Eq. (7.1)).

In order to apply the parameter interval estimator on Eq. (7.26), it is necessary to compute $W(j\omega_k)$ such that $|w(j\omega_k)| \leq W(j\omega_k)$. Using Ljung(17) [Eq. (6.28)], and noting that $w(j\omega_k)$ is an equation error and not an output error,

$$w(j\omega_k) = A(e^{j\omega_k})R_N(\omega_k)/U_N(\omega_k) + V_N(\omega_k)/U_N(\omega_k) \tag{7.27}$$

where, according to Ljung,(17) [Eq. (2.54)]

$$|R_N(\omega_k)| \leq 2C_u G_G/\sqrt{N} \tag{7.28}$$

with C_u such that $|u(t)| \leq C_u$. Denoting the impulse response of $G(q)$ by $g(t)$, $C_G = \Sigma_{t=1}^{\infty} t|g(t)|$.

Clearly, from Eq. (7.27),

$$|w(j\omega_k)| \leq W(j\omega_k) = |A(e^{j\omega_k})R_N(\omega_k)/U_N(\omega_k)| + |V_N(\omega_k)/U_N(\omega_k)| \tag{7.29}$$

The first member of the right hand side of Eq. (7.29) is deterministic since $u(t)$ is assumed to be known. It can be estimated via Eq. (7.28) and an estimate of $A(e^{j\omega_k})$. Assuming that Eq. (7.21) is the sampled version of a stable continuous system, $|A(e^{j\omega_k})| \leq \max_{a_1 \in [0,1)}(|e^{j\omega_k} - a_1|)$. In Eq. (7.28), $C_u = 1$ and $C_G < \infty$ for $a_1 \in [0,1)$. Consequently $W(j\omega_k)$ is a very large number that makes the method useless in this case.

However, change the example such that $u(t)$ is periodic with a period of N so that $R_N(\omega_k) = 0$ according to Ljung[17] and the first member of the right hand side of Eq. (7.29) equals zero. Choosing a sum of sinus signals at five frequencies such that their range covers the expected bandwidth of Eq. (7.21):

$$u(t) = 0.2 \sum_{m=0}^{4} \sin(\omega_{(3 \cdot 2^m)} t), \quad t = 1, \ldots, N \tag{7.30}$$

Then, the standard deviation of the second member of the right hand side of Eq. (7.29) which is random with respect to $v(t)$ (Ljung,[17] page 149) and whose mean is zero, can be computed as follows: the power of $u(t)$ equals $0.2^2/2$, at each of the five frequencies $\omega_{(3 \cdot 2^m)}$, $m = 0, 1, \ldots, 4$, otherwise the power is zero. The power spectral density of $v(t)$ ideally equals the variance of $v(t)$ (= 1/3) divided by $N/2$, i.e., 1/150. Hence, it can be estimated (Ljung,[17] Eq. (6.34a)) that

$$\sigma(V_N(\omega_k)/U_N(\omega_k)) \approx \sqrt{1/3}, \quad \text{for } k = 3 \cdot 2^m, \quad m = 0, 1, \ldots, 4 \tag{7.31}$$

Although not strictly correct but common in practice, let $W(j\omega_k)$ equal three standard deviations, i.e., $W(j\omega_k) = \sqrt{3}$. Applying the offline algorithm in Section 7.2.1 to Eq. (7.26) at the five frequencies defined in Eq. (7.31), with $|w(j\omega_k)| \leq \sqrt{3}$, the LS estimate $\hat{\Theta}(5) = (0.5020 \ 1.2702)^T$ with, $E_1(1, 5) = 1.0161$ and $E_2(2, 5) = 1.8010$, signifying that modulo the 3σ assumption, $a_1 \in [0.5020 \pm 1.0161]$ and $b_1 \in [1.2702 \pm 1.8010]$. This estimate is of roughly the same quality as the one in Example 1.

We have demonstrated that if a frequency function estimate is given, generated by a known periodic input sequence, then the proposed algorithm can be used to bound the estimates of the coefficients of transfer function numerator and denominator polynomials.

7.4. CONCLUSIONS

From the recorded input-output data and an assumed RLS model with bounded equation error, the feasible parameter intervals for each parameter θ_i,

$$\theta_i \in [\hat{\theta}_i^{\text{LS}}(N) - E_i(i,N), \hat{\theta}_i^{\text{LS}}(N) + E_i(i,N)]$$

have been computed. Since $E(i,N)$ is not conveniently computed in a recursive way, a recursively computable inner, $D(i,N)$, and outer, $F(N)$, bounding is found: $D_i(i,N) \leq E_i(i,N) \leq F_i(N)$. Under weak assumptions, $D(i,N)$, $E(i,N)$, and $F(N)$ converge to their respective limits as $N \to \infty$, with $D(i,N)$ and $E(i,N)$ sharing the same limit.

Note that $E_i(i,N)$ *exactly* describes the feasible parameter interval in the worst case: there exists a feasible equation error sequence that could have produced the input-output data with the parameter value belonging to the interval. Moreover, $D_i(i,N)$ is an asymptotically exact inner bound of $E_i(i,N)$. Hence it is believed that a contribution of this chapter is the development of orthotopic[6,11] inner bounding. It should however be remarked that the bounds in this chapter are inner and outer bounds of the estimate uncertainty intervals due to the LS algorithm.[5] Since the LS algorithm is asymptotically correct, $D_i(i,N)$ tends to the inner bound, and $\pm E_i(i,N)$ tend to the inner and outer bounds, respectively, of the feasible parameter intervals.[5,11]

The proposed algorithm is obviously not a special case of projecting ellipsoidal inner and outer boundings[5–7,9,10] along the parameter axes, since the covariance matrix $P(t)$ is not used for the interval estimate. Instead, a specially constructed, worst-case equation error sequence is employed. Moreover, It has been demonstrated that bounding ellipsoids are very crude;[6,7] in particular the inner ellipsoid tends to vanish. Our inner bound, $D_i(i,N)$, is asymptotically exact. The weak point of the algorithm is the convergence limit of $F_i(N)$, which in general does not equal the limit of $E_i(i,N)$. Under additional assumptions on the data, the distance between the limits may be established. The simulations of Example 1 and 2 indicate, however, that there are cases when the limits are close. The algorithm possesses the same tracking ability as the RLS on which it is based, since the estimated intervals are centered around their respective point estimates. However, the interval estimates may be unreliable during transients (Example 3).

In most cases, methods giving an exact description[8] of the feasible parameter set should be preferable. But for a practitioner who already has a well oiled RLS or one of its cousins running in his system, only a marginal additional effort is necessary to include the parameter set estimator proposed. The algorithm is a small contribution to combining statistical and set-membership estimators.[6]

Finally, it was demonstrated in an example that the algorithm can be used in the frequency domain, given a frequency function measurement with a periodic input signal.

ACKNOWLEDGMENTS. This research was supported by the Fund for the promotion of research at the Technion, and by the Swedish Board for Technical and Industrial Development, NUTEK.

REFERENCES

1. I. M. Horowitz, and M. Sidi, *Int. J. Control* **16**, 287 (1972).
2. O. Yaniv, P. O. Gutman, and L. Neumann, *Automatica* **26**, 709 (1990).
3. P. O. Gutman, in: *Proceedings of the 1988 IFAC Workshop on Robust Adaptive Control*, Newcastle, N.S.W., Australia (1988).

4. K. J. Åström, and B. Wittenmark, *Adaptive Control*, Addison-Wesley, Reading, MA (1989).
5. M. Milanese, and A. Vicino, in: *Bounding Approaches to System Identification* (M. Milanese *et al.*, eds.) Plenum Press, New York, Chap. 2 (1996).
6. E. Walter, and H. Piet-Lahanier, *Math. Comp. Simul.* **32**, 449 (1990).
7. J. P. Norton, *Automatica* **23**, 497 (1987).
8. E. Walter, and H. Piet-Lahanier, *IEEE Trans. Autom. Control* **34**, 911 (1989).
9. E. Fogel and Y. F. Huang, *Automatica* **18**, 229 (1982).
10. S. Dasgupta and Y. F. Huang, *IEEE Trans. Inf. Theory* **33**, 3 (1987).
11. M. Milanese and G. Belforte, *IEEE Trans. Autom. Control* **27**, 408 (1982).
12. R. L. Kosut, *Int. J. Adapt. Control Signal Proc.* **2**, 371 (1988).
13. M. Gevers, in: *IFAC Ident. Syst. Param. Est.*, Budapest, Hungary, pp. 1–10 (1991).
14. G. C. Goodwin and M. E. Salgado, *Int. J. Adapt. Control Signal Proc.* **3**, 333 (1990).
15. R. O. LaMaire, L. Valavani, M. Athans, and G. Stein, *Automatica* **27**, 23 (1991).
16. B. Wahlberg and L. Ljung, *IEEE Trans. Autom. Control* **37**, 900 (1992).
17. L. Ljung, *System Identification: Theory for the User*, Prentice-Hall, Englewood Cliffs, New Jersey (1987).
18. T. Söderström and P. Stoica, *System Identification*, Prentice-Hall International (UK) Ltd, Hertfordshire, UK (1989).
19. L. Ljung and T. Söderström, *Theory and Practice of Recursive Identification*, MIT Press, Cambridge, MA (1983).
20. G. C. Goodwin and K. S. Sin, *Adaptive Filtering, Prediction, and Control*, Prentice-Hall, Englewood Cliffs, N.J. (1984).
21. R. H. Middleton, G. C. Goodwin, D. J. Hill, and D. Q. Mayne, *IEEE Trans. Aut. Contr.* **33**, 50 (1988).
22. T. Hägglund, *New Estimation Techniques for Adaptive Control*, Ph.D. thesis, Lund Institute of Technology, Lund, Sweden (1983).
23. C. A. Canudas de Wit, *Commande Adaptatives pour des Systèmes Partiallement Connus*, Ph.D. thesis, Laboratoire d'Automatique de Grénoble, E.N.S.I.E.G., Saint Martin d'Hères, France (1987).
24. R. Johansson, *Identification of Continuous Time Dynamic Systems*, Internal Report, CODEN: LUTFD2/(TFRT-7318)/1-29, Lund Institute of Technology, Lund, Sweden (1986).
25. P. O. Gutman, On-line Identification of Transfer Function Value Sets, *IFAC Ident. Syst. Param. Est.,* Budapest, Hungary, pp. 758–762 (1991).
26. M. Lilja, *Controller Design by Frequency Domain Approximation*, Ph.D. thesis, Internal Report, CODEN: LUTFD2/ (TFRT-1031) /1-107, Lund Institute of Technology, Lund, Sweden (1989).
27. E. C. Levy, *I.R.E. Trans. Aut. Contr.* **4**, 37 (1959).
28. E. Walter and H. Piet-Lahanier, in: *Bounding Approaches to System Identification* (M. Milanese *et al.*, eds.) Plenum Press, New York, Chap. 12 (1996).

8

Volume-Optimal Inner and Outer Ellipsoids

L. Pronzato and É. Walter

8.1. INTRODUCTION AND PROBLEM STATEMENT

Approximating a complex set $\mathcal{K}$ by a simple geometrical form (such as a polytope, an orthotope, a sphere or an ellipsoid) is often of practical interest. Consider for instance the situation where a vector **u** has to be chosen so as to satisfy the property

$$\mathbf{p}(\mathbf{u},\mathbf{x}) \in \mathcal{T}, \forall\, \mathbf{x} \in \mathcal{S}, \tag{8.1}$$

where **x** and **p**(.,.) are vector-valued and where $\mathcal{T}$ and $\mathcal{S}$ are given sets. This can be of interest for instance in robust control, where the controller characterized by **u** must be designed in order to guarantee some given performances—at least stability—corresponding to a target set $\mathcal{T}$ for the process under study, given the information that the model parameters **x** lie in some specified feasible domain $\mathcal{S}$. The information about $\mathcal{S}$ can be derived using the parameter bounding methodology, where one assumes that observations with bounded errors are performed on the process.[1]

Two methods can be used to replace the initial problem by a simpler one. The first one is to replace Eq. (8.1) by the sufficient condition $\mathbf{p}(\mathbf{u}, \mathbf{x}) \in \mathcal{T}, \forall\, \mathbf{x} \in \mathcal{O} \supset \mathcal{S}$, with $\mathcal{O}$ an outer approximation of $\mathcal{S}$, e.g. an ellipsoid $\mathcal{E}_o$. The second one is to

L. PRONZATO • Laboratoire 13S, CNRS URA-1376, Sophia Antipolis, 06560 Valbonne, France.
É. WALTER • Laboratoire des Signaux et Systèmes, CNRS-École Supérieure d'Electricité, 91192 Gif-sur-Yvette Cedex, France.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

replace it by the sufficient condition $\mathbf{p}(\mathbf{u}, \mathbf{x}) \in I \subset \mathcal{T}$, $\forall$ $\mathbf{x} \in \mathcal{S}$, where I might be for instance an ellipsoid $\mathcal{E}_i$.

In Ref. 8.2, an algorithm from the field of experimental design is used to construct the minimum-volume ellipsoid containing $\mathcal{S}$. One may hope that, with this optimal ellipsoidal approximation, robust control laws will be obtained that are less conservative than those derived from coarser approximations.(3) Various statistical applications of volume-optimal outer ellipsoids are suggested in Ref. 8.4. See especially the robust estimation of correlation coefficients.(5,6,7)

Ellipsoidal inner approximation is of interest in the context of tolerance design (design centering).(8) It is a basic tool for efficient methods in convex programming.(9,10) In the context of parameter bounding, characterizing $\mathcal{S}$ by outer and inner ellipsoidal approximations permits evaluation of the accuracy of these characterizations.(11)

Let $\mathcal{K}$ be a bounded convex body of the Euclidian space $\mathbb{R}^p$. From the Loewner-Behrend theorem,(12) there exists a unique ellipsoid $\mathcal{E}_o^*$ of minimal volume containing $\mathcal{K}$, and, from Ref. 8.3, there also exists a unique ellipsoid $\mathcal{E}_i^*$ of maximal volume contained in $\mathcal{K}$.

We shall denote $P_o(\mathcal{K})$ (resp. $P_i(\mathcal{K})$) the problem corresponding to the determination of $\mathcal{E}_o^*(\mathcal{K})$ (resp. $\mathcal{E}_i^*(\mathcal{K})$). Both problems are simpler when the center of the ellipsoid to be determined is fixed *a priori*, and they will then be denoted by $P_{o_c}(\mathcal{K})$ and $P_{i_c}(\mathcal{K})$. When $\mathcal{K}$ is a polytope, all these problems can be solved by classical nonlinear programming approaches (constrained Newton, path-following Newton methods...). Moreover, the solutions of $P_o(\mathcal{K})$, $P_{o_c}(\mathcal{K})$ and $P_{i_c}(\mathcal{K})$ can be obtained through the solution of a problem $P_i(.)$.(10) We consider a more general situation where $\mathcal{K}$ is not necessarily a polytope (and even not necessarily convex for problems $P_o(\mathcal{K})$, $P_{o_c}(\mathcal{K})$).

An algorithm for solving $P_o(\mathcal{K})$, with $\mathcal{K}$ not necessarily convex, is given in Section 8.2. Some basic results about inner and outer ellipsoids are recalled in Section 8.3, to be used for the solution of $P_i(\mathcal{K})$, with $\mathcal{K}$ a convex set. This problem is considered in Section 8.4. When $\mathcal{K}$ is a polytope, $\mathcal{E}_i^*(\mathcal{K})$ is then obtained through the solution of a series of problems $P_o(.)$ for polytopes. When $\mathcal{K}$ is a general convex set, $P_i(\mathcal{K})$ is solved via a relaxation procedure, i.e., it is decomposed into a series of subproblems $P_i(.)$ for polytopes. Finally, $P_i(\mathcal{K})$ when $\mathcal{K}$ is a polytope determined recursively is considered. Various illustrative examples are presented.

8.2. MINIMUM-VOLUME OUTER ELLIPSOID

Problems P_{o_c} and P_o are known to be respectively duals of a D-optimal experiment-design problem(14) and a D_s-optimal design problem.(15) When $\mathcal{K}$ is a polytope characterized by its vertices, we simply have to determine the minimal ellipsoid containing a finite set of points. The equivalence between P_{o_c} and P_o is then shown in Ref. 8.10, and the case $p = 2$ can be solved with a finite exact algorithm,[4,16] which is not considered here. Consider $P_o(\mathcal{K})$, with $\mathcal{K}$ a given compact set of $\mathbb{R}^p$ (not necessarily convex). Let Ξ be the set of all normalized distributions of weights on $\mathcal{K}$,

$$\int_{\mathcal{K}} \xi(d\mathbf{x}) = 1,$$

and define $\mathbf{M}(\xi)$ and $\mathbf{c}(\xi)$ as

$$\mathbf{M}(\xi) := \int_{\mathcal{K}} \mathbf{x}\mathbf{x}^T\xi(d\mathbf{x}),\ \ \mathbf{c}(\xi) := \int_{\mathcal{K}} \mathbf{x}\xi(d\mathbf{x})\,. \tag{8.2}$$

The following theorem states the equivalence between $P_o(\mathcal{K})$ and the determination of an optimal distribution ξ^* on $\mathcal{K}$.

Theorem 8.1. $\mathcal{E}_o^*(\mathcal{K})$ corresponds to

$$\mathcal{E}(\mathbf{c}^*,\mathbf{A}^*) := \{\mathbf{x} \in \mathbb{R}^p \mid (\mathbf{x} - \mathbf{c}^*)^T\mathbf{A}^*(\mathbf{x} - \mathbf{c}) \le p\},$$

with

$$\mathbf{c}^* := \mathbf{c}(\xi^*),\ \mathbf{A}^* := (\mathbf{M}(\xi^*) - \mathbf{c}(\xi^*)\mathbf{c}^T(\xi^*))^{-1},$$

and ξ^* obtained by

$$\xi^* := \arg\max_{\xi\in\Xi} \ln\det[\mathbf{M}(\xi) - \mathbf{c}(\xi)\mathbf{c}^T(\xi)]. \tag{8.3}$$

Proof: See Refs. 8.2 and 8.15. □

The solution of $P_o(\mathcal{K})$ thus amounts to the determination of ξ^* in Eq. (8.3). This optimal distribution satisfies the following properties.

Theorem 8.2.

(i) A distribution ξ^* supported by at most $\frac{p(p+3)}{2}$ (and at least $p + 1$) points of $\mathcal{K}$ always exists. These support points are located on the boundary of the convex closure of $\mathcal{K}$. When there are only $p + 1$ support points, they are weighted uniformly and the center $\mathbf{c}^*$ of the ellipsoid corresponds to their center of gravity.

(ii) ξ^* is not necessarily unique (although $\mathcal{E}(\mathbf{c}^*,\mathbf{A}^*)$ is), but the set of all optimal distributions satisfying Eq. (8.3) is convex.

(iii) $\forall\ \mathbf{x} \in \mathcal{K}$, $d(\mathbf{x},\xi^*) \le 0$, with

$$d(\mathbf{x},\xi) := \mathbf{x}^T\mathbf{M}^{-1}(\xi)\mathbf{x} + \frac{[\mathbf{x}^T\mathbf{M}^{-1}(\xi)\mathbf{c}(\xi) - 1]^2}{1 - \mathbf{c}^T(\xi)\mathbf{M}^{-1}(\xi)\mathbf{c}(\xi)} - (p + 1). \tag{8.4}$$

(iv) $\max_{\mathbf{x}\in\mathcal{K}} d(\mathbf{x},\xi^*) = \min_{\xi\in\Xi}\max_{\mathbf{x}\in\mathcal{K}} d(\mathbf{x},\xi)$.

Proof: (i) follows from Caratheodory's theorem;[12] (ii–iv) result from the concavity of the criterion

$$\Phi(\xi) := \ln \det[\mathbf{M}(\xi) - \mathbf{c}(\xi)\mathbf{c}^T(\xi)]. \tag{8.5}$$

A detailed proof can be found in the experimental design literature.[17,18] □

From (i), in practice it is always possible to restrict attention to discrete distributions of weights λ_i on support points $\mathbf{x}_i$, $i = 1, \ldots, n$, with $n \leq \frac{p(p+3)}{2}$. The integrals in Eq. (8.2) then reduce to discrete sums.

The main interest of Theorem 8.2 is perhaps the availability of efficient algorithms developed in the context of experimental design. It can also be used to prove the global convergence (whatever the choice of the initial distribution ξ^0) of the following vertex-direction algorithm.[19,20,21]

AP_0: (Algorithm for problem P_o)

Step (i) Choose ε such that $0 < \varepsilon << 1$, and a discrete distribution ξ^0 such that $\mathbf{M}(\xi^0)$ is invertible. Set $k = 0$.

Step (ii) Compute

$$\mathbf{x}^+ := \arg \max_{\mathbf{x} \in \mathcal{X}} d(\mathbf{x}, \xi^k). \tag{8.6}$$

If $d(\mathbf{x}^+, \xi^k) < \varepsilon$, stop.

Step (iii) Compute ξ^{k+1} as a distribution whose support points $\mathbf{x}_i$ are $\mathbf{x}^+$ and those of ξ^k and whose weights λ_i are the best in the sense $\Phi(\xi)$ in Eq. (8.5). Remove any support point with zero weight from ξ^{k+1} . $k \leftarrow k + 1$, go to Step (ii).

Step (ii) involves the maximization of a convex quadratic function over $\mathcal{X}$. Local methods may thus not converge to the global optimum. When $\mathcal{X}$ is a polytope, from Theorem 8.2 (i), the only candidates for $\mathbf{x}^+$ in Eq. (8.6) are the vertices of $\mathcal{X}$ so that the global solution is obtained at a very low computational cost when these vertices are known. Step (iii) corresponds to the maximization of $\Phi(\xi)$ in Eq. (8.5), which is a concave function of the weights λ_i , $i = 1, \ldots, n$, with the constraints $\lambda_i \geq 0$, $\Sigma_{i=1}^n \lambda_i = 1$. A constrained Newton method (which amounts to solving a sequence of convex quadratic programming problems) can thus be used. The following expressions for the gradient and Hessian of $\Phi(\xi)$ allow an easy implementation of the algorithm.

$$\frac{\partial \Phi(\xi)}{\partial \lambda_i} = \mathbf{x}_i^T \mathbf{M}^{-1}(\xi)\mathbf{x}_i + \frac{[\mathbf{x}_i^T \mathbf{M}^{-1}(\xi)\mathbf{c}(\xi) - 1]^2 - 1}{1 - \mathbf{c}^T(\xi)\mathbf{M}^{-1}(\xi)\mathbf{c}(\xi)},$$

$$\frac{\partial^2 \Phi(\xi)}{\partial \lambda_i \partial \lambda_j} = (\mathbf{x}_j^T \mathbf{M}^{-1}(\xi)\mathbf{x}_i)^2 + \frac{2\mathbf{x}_j^T \mathbf{M}^{-1}(\xi)\mathbf{x}_i}{1 - \mathbf{c}^T(\xi)\mathbf{M}^{-1}(\xi)\mathbf{c}(\xi)} \times$$

$$\{\mathbf{c}^T(\xi)\mathbf{M}^{-1}(\xi)(\mathbf{x}_i + \mathbf{x}_j) - [\mathbf{x}_i^T \mathbf{M}^{-1}(\xi)\mathbf{c}(\xi)]\,[\mathbf{x}_j^T \mathbf{M}^{-1}(\xi)\mathbf{c}(\xi)] - 1\} +$$

$$\frac{[\mathbf{x}_i^T \mathbf{M}^{-1}(\xi)\mathbf{c}(\xi)][\mathbf{x}_j^T \mathbf{M}^{-1}(\xi)\mathbf{c}(\xi)]}{[1 - \mathbf{c}^T(\xi)\mathbf{M}^{-1}(\xi)\mathbf{c}(\xi)]^2} \times$$

$$\{2\mathbf{c}^T(\xi)\mathbf{M}^{-1}(\xi)(\mathbf{x}_i + \mathbf{x}_j) - [\mathbf{x}_i^T\mathbf{M}^{-1}(\xi)\mathbf{c}(\xi)][\mathbf{x}_j^T\mathbf{M}^{-1}(\xi)\mathbf{c}(\xi)] - 4\}.$$

The Newton method can be initialized with the distribution ξ^{k+1} that would be chosen by a Fedorov-like algorithm,(19)

$$\xi = (1 - \alpha)\xi^k + \alpha\xi_{x^+},$$

where ξ_{x^+} gives a unit mass to the single support point $\mathbf{x}^+$ of Eq. (8.6) and α is given by

$$\alpha = \frac{d(\mathbf{x}^+,\xi^k)}{(p + 1)[d(\mathbf{x}^+,\xi^k) + p]}.$$

Note that, in this case, optimizing the weights is not necessary to insure global monotone convergence. It is, however, highly recommended to obtain a satisfactory transient. Applications of this algorithm to parameter bounding can be found in Ref. 8.22.

REMARK 8.1. The construction of the minimum-volume sphere containing $\mathcal{K}$ (not necessarily convex) coincides with the determination of the Chebychev center of $\mathcal{K}$ for the Euclidian norm. One can easily show(23) that the center $\mathbf{c}^*$ and radius r^* of the minimum covering sphere satisfy $\mathbf{c}^* = \mathbf{c}(\xi^*)$ and $r^* = r(\xi^*)$, where

$$r^2(\xi) := trace[\mathbf{M}(\xi) - \mathbf{c}(\xi)\mathbf{c}^T(\xi)],$$

and

$$\xi^* = \arg \max_{\xi \in \Xi} r^2 (\xi),$$

with $\mathbf{M}(\xi)$ and $\mathbf{c}(\xi)$ given by Eq. (8.2). An algorithm similar to AP_o can be derived, that converges globally to the optimal distribution ξ^* for the criterion $r(.)$. When $\mathcal{K}$ is a polytope, a finite algorithm for the determination of its Chebychev center (of the same type as the simplex algorithm for linear programming) is described in Ref. 24.

Duality properties between inner and outer ellipsoids will now be presented, to be used in Section 8.4 for the solution of P_i.

8.3. DUALITY PROPERTIES

Consider the set $C_o(\mathbb{R}^p)$ of convex compact subsets $\mathcal{K}$ of $\mathbb{R}^p$ that contain the origin O in their interior ($O \in \text{int}(\mathcal{K})$), and the transformation $T(.)$ given by

$$C_o(\mathbb{R}^p) \to C_o(\mathbb{R}^p)$$

$$\mathcal{K} \mapsto T(\mathcal{K}) := \{\phi \in \mathbb{R}^p \mid \phi^T \mathbf{x} \le p, \forall\, \mathbf{x} \in \mathcal{K}\}. \tag{8.7}$$

The following properties, illustrated by Fig. 8.1, then hold true.

LEMMA 8.1.

(i) $T(.)$ defines a relation of duality in the following sense:

—$\forall \mathcal{K} \in C_o(\mathbb{R}^p)$, $T[T(\mathcal{K})] = \mathcal{K}$,

—for any given polytope $\mathcal{K}$ in $C_o(\mathbb{R}^p)$, $T(.)$ defines a one-to-one relation between the q-faces of $\mathcal{K}$ and the $(p-q-1)$-faces of $T(\mathcal{K})$ (for a polytope in $\mathbb{R}^p$, a vertex is a 0-face, an edge is a 1-face, a hyperplane is a $(p-1)$-face, a q-face is a face of a $(q+1)$-face).

(ii) When $\mathcal{K}$ is a polytope of $C_o(\mathbb{R}^p)$ defined by

$$\mathcal{K} = \{\mathbf{x} \in \mathbb{R}^p \mid \mathbf{a}_i^T\mathbf{x} \leq b_i, i = 1, \ldots, m\},$$

the vertices of $T(\mathcal{K})$ belong to the set $\{p\mathbf{a}_i/b_i, i = 1, \ldots, m\}$.

(iii) $T(.)$ satisfies

$$\forall\ (\mathcal{K}_1, \mathcal{K}_2) \in C_o(\mathbb{R}^p)^2,\ \mathcal{K}_1 \subset \mathcal{K}_2 \Leftrightarrow T(\mathcal{K}_2) \subset T(\mathcal{K}_1).$$

(iv) $T[\mathcal{E}(\mathbf{c},\mathbf{A})] = \mathcal{E}(\mathbf{c}',\mathbf{A}')$, with

$$\begin{cases} \mathbf{c}' = -\dfrac{p\mathbf{A}\mathbf{c}}{p - \mathbf{c}^T\mathbf{A}\mathbf{c}}, \\ \mathbf{A}'^{-1} = \dfrac{p}{p - \mathbf{c}^T\mathbf{A}\mathbf{c}}(\mathbf{A} + \dfrac{\mathbf{A}\mathbf{c}\mathbf{c}^T\mathbf{A}}{p - \mathbf{c}^T\mathbf{A}\mathbf{c}}), \end{cases}$$

where $\mathcal{E}(\mathbf{c},\mathbf{A})$ denotes the ellipsoid defined by

$$\mathcal{E}(\mathbf{c},\mathbf{A}) := \{\mathbf{x} \in \mathbb{R}^p \mid (\mathbf{x} - \mathbf{c})^T\mathbf{A}(\mathbf{x} - \mathbf{c}) \leq p\}, \tag{8.8}$$

with $O \in \text{int}[\mathcal{E}(\mathbf{c},\mathbf{A})]$ (i.e., $\mathbf{c}^T\mathbf{A}\mathbf{c} < p$).

PROOF:

(i) The duality property is proved in Ref. 8.12.

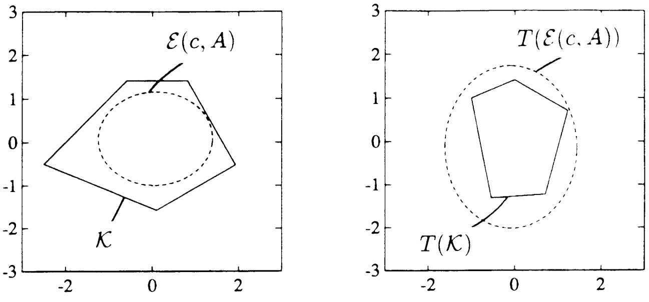


FIGURE 1. Illustration of Lemma 8.1.

(ii) $O \in \text{int}(\mathcal{K})$ implies $b_i > 0$, $i = 1, \dots, m$, and thus

$$\mathcal{K} = \{\mathbf{x} \in \mathbb{R}^p \mid \frac{p}{b_i}\mathbf{a}_i^T\mathbf{x} \leq p, i = 1, \dots, m\}.$$

If the hyperplane defined by $(p/b_i)\mathbf{a}_i^T\mathbf{x} = p$ is a $(p-1)$-face of $\mathcal{K}$, it is transformed by $T(.)$ into the 0-face (vertex) $(p/b_i)\mathbf{a}_i$ of $T(\mathcal{K})$.

(iii) Assume first that $\mathcal{K}_1 \subset \mathcal{K}_2$. Then, $\forall\ \phi \in T(\mathcal{K}_2)$, $\max_{\mathbf{x}\in\mathcal{K}_2}\phi^T\mathbf{x} \leq p$, so that

$$\max_{\mathbf{x}\in\mathcal{K}_1} \phi^T\mathbf{x} \leq p,$$

and $\phi \in T(\mathcal{K}_1)$, which implies $T(\mathcal{K}_2) \subset T(\mathcal{K}_1)$. Assume now that $T(\mathcal{K}_2) \subset T(\mathcal{K}_1)$. This implies $T[T(\mathcal{K}_1)] \subset T[T\mathcal{K}_2)]$ and thus from (i), $\mathcal{K}_1 \subset \mathcal{K}_2$.

(iv) $T[\mathcal{E}(\mathbf{c},\mathbf{A})] = \{\phi \in \mathbb{R}^p \mid \max_{\mathbf{x}\in\mathcal{E}(\mathbf{c},\mathbf{A})} \phi^T\mathbf{x} \leq p\}$, so we first have to compute $\max_{\mathbf{x}\in\mathcal{E}\mathbf{c},(\mathbf{A})}\phi^T\mathbf{x}$. Elementary calculations (using the Lagrangian method) give

$$\max_{\mathbf{x}\in\mathcal{E}(\mathbf{c},(\mathbf{A})} \phi^T\mathbf{x} = \phi^T\mathbf{c} + (p\phi^T\mathbf{A}^{-1}\phi)^{1/2},$$

which implies

$$T[\mathcal{E}(\mathbf{c},\mathbf{A})] = \{\phi \in \mathbb{R}^p \mid \phi^T\mathbf{c} + (p\phi^T\mathbf{A}^{-1}\phi)^{1/2} \leq p\}$$
$$\subset \{\phi \in \mathbb{R}^p \mid p\phi^T\mathbf{A}^{-1}\phi \leq (p - \phi^T\mathbf{c})^2\}.$$

This last set is easily shown to correspond to $\mathcal{E}(\mathbf{c}',\mathbf{A}')$. Now,

$$\max_{\phi\in\mathcal{E}(\mathbf{c}',\mathbf{A}')} \phi^T\mathbf{c} = \mathbf{c}^T\mathbf{c}' + (p\mathbf{c}^T\mathbf{A}'^{-1}\mathbf{c})^{1/2}$$
$$= pf(u),$$

with $f(u) := (u + u^2)^{1/2} - u$, and $u = \mathbf{c}^T\mathbf{A}\mathbf{c}/(p - \mathbf{c}^T\mathbf{A}\mathbf{c})$, which is strictly positive since $O \in \text{int}(\mathcal{E}(\mathbf{c},\mathbf{A}))$. One can then check that $f(u) < 1$, so that $\phi^T\mathbf{c} < p$, $\forall\ \phi \in \mathcal{E}(\mathbf{c}',\mathbf{A}')$. One therefore has $p\phi^T\mathbf{A}^{-1}\phi \leq (p - \phi^T\mathbf{c})^2 \Rightarrow \phi^T\mathbf{c} + (p\phi^T\mathbf{A}^{-1}\phi)^{1/2} \leq p$, which finally gives $T[\mathcal{E}(\mathbf{c},\mathbf{A})] = \mathcal{E}(\mathbf{c}',\mathbf{A}')$. □

Corollary 8.1. The volumes of $\mathcal{E}(\mathbf{c},\mathbf{A})$ and $T[\mathcal{E}(\mathbf{c},\mathbf{A})]$ satisfy

$$\text{vol}[\mathcal{E}(\mathbf{c},\mathbf{A})]\ \text{vol}\{T[\mathcal{E}(\mathbf{c},\mathbf{A})]\} = \beta^2(p)\left(1 - \frac{\mathbf{c}^T\mathbf{A}\mathbf{c}}{p}\right)^{-\frac{p+1}{2}} \geq \beta^2(p), \tag{8.9}$$

with $\beta(p)$ the volume of the p-dimensional ball $\mathcal{B}(\mathbf{0}, p^{1/2})$. The inequality is strict when $\mathbf{c} \neq \mathbf{0}$.

Proof. One has

$$\text{vol}\{T[\mathcal{E}(\mathbf{c},\mathbf{A})]\} = \beta(p)(\det \mathbf{A}')^{-1/2},$$

and

$$\mathrm{vol}[\mathcal{E}(\mathbf{c},\mathbf{A})] = \beta(p)(\det \mathbf{A})^{-1/2},$$

with

$$(\det \mathbf{A}')^{-1} = \left(1 - \frac{\mathbf{c}^T\mathbf{A}\mathbf{c}}{p}\right)^{-(p+1)} \det \mathbf{A}. \qquad \square$$

In what follows we shall also need to consider translations in $\mathbb{R}^p$. For any **c** in $\mathbb{R}^p$ define $\tau_c(.)$ by

$$\mathbb{R}^p \to \mathbb{R}^p$$

$$\mathbf{z} \mapsto \mathbf{z} - \mathbf{c}.$$

Obviously, $\tau_c(.)$ preserves volumes of sets, and $\tau_{-c}(.) = \tau_c^{-1}(.)$.

Consider first ellipsoids with a fixed center $\mathbf{c} \in \mathcal{K}$. The following theorem, where $\circ$ denotes the composition of operators, relates P_{i_c} and P_{o_c}.

THEOREM 8.3. $P_{i_c}(\mathcal{K})$ is equivalent to $P_{o_c}(T \circ \tau_c(\mathcal{K}))$.

PROOF. We have from Eq. (8.9),

$$\mathrm{vol}\{T \circ \tau_c[\mathcal{E}(\mathbf{c},\mathbf{A})]\} = \mathrm{vol}\{T\,[\mathcal{E}(\mathbf{0},\mathbf{A})]\} = \frac{\beta^2(p)}{\mathrm{vol}[\mathcal{E}(\mathbf{0},\mathbf{A})]} = \frac{\beta^2(p)}{\mathrm{vol}[\mathcal{E}(\mathbf{c},\mathbf{A})]}\,.$$

From Lemma 8.1 (iii), $\mathcal{E}(\mathbf{c},\mathbf{A})$ is contained in $\mathcal{K}$ if and only if $T \circ \tau_c[\mathcal{E}(\mathbf{c},\mathbf{A})]$ contains $T \circ \tau_c(\mathcal{K})$. Maximizing $\mathrm{vol}[\mathcal{E}(\mathbf{c},\mathbf{A})]$ is equivalent to minimizing $\mathrm{vol}\{T \circ \tau_c[\mathcal{E}(\mathbf{c},\mathbf{A})]\}$, which states the proof. $\square$

Consider now ellipsoids whose centers are chosen optimally. The following result permits increasing the volume of an inner ellipsoid for $\mathcal{K}$ through the solution of a problem P_o.

THEOREM 8.4. Let $\mathcal{E}(\mathbf{c},\mathbf{A})$ be an inner ellipsoid for $\mathcal{K}$, we have

$$\tau_{-c} \circ T\{\mathcal{E}_o^*[T \circ \tau_c(\mathcal{K})]\} \subset \mathcal{K},$$

and

$$\mathrm{vol}(\tau_{-c} \circ T\{\mathcal{E}_o^*[T \circ \tau_c(\mathcal{K})]\}) \geq \mathrm{vol}[\mathcal{E}(\mathbf{c},\mathbf{A})].$$

PROOF: From Lemma 8.1 (i, iii), $\tau_c(\mathcal{K}) \supset T\{\mathcal{E}_o^*[T \circ \tau_c(\mathcal{K})]\}$, and the first part of the theorem is proved. Let $\mathcal{E}_{o_o}^*[T \circ \tau_c(\mathcal{K})]$ be the minimum-volume ellipsoid with center O containing $T \circ \tau_c(\mathcal{K})$. A smaller ellipsoid can be obtained if its center is chosen optimally,

$$\mathrm{vol}\{\mathcal{E}_o^*[T \circ \tau_c(\mathcal{K})]\} \leq \mathrm{vol}\{\mathcal{E}_{o_o}^*[T \circ \tau_c(\mathcal{K})]\}\,. \tag{8.10}$$

From Eq. (8.9),

$$\mathrm{vol}(\tau_{-c} \circ T\{\mathcal{E}_o^*[T \circ \tau_c(\mathcal{K})]\}) \geq \frac{\beta^2(p)}{\mathrm{vol}\{\mathcal{E}_o^*[T \circ \tau_c(\mathcal{K})]\}}, \tag{8.11}$$

which together with Eq. (8.10) gives

$$\mathrm{vol}(\tau_{-c} \circ T\{\mathcal{E}_o^*[T \circ \tau_c(\mathcal{K})]\}) \geq \frac{\beta^2(p)}{\mathrm{vol}\{\mathcal{E}_{o_o}^*[T \circ \tau_c(\mathcal{K})]\}}, \tag{8.12}$$

Using Theorem 8.3 and Eq. (8.9),

$$\mathrm{vol}\{\mathcal{E}_{o_o}^*[T \circ \tau_c(\mathcal{K})]\} = \frac{\beta^2(p)}{\mathrm{vol}\{\mathcal{E}_{i_o}^*[\tau_c(\mathcal{K})]\}} = \frac{\beta^2(p)}{\mathrm{vol}[\mathcal{E}_{i_c}^*(\mathcal{K})]}, \tag{8.13}$$

where $\mathcal{E}_{i_c}^*(\mathcal{K})$ is the maximum volume ellipsoid with center **c** contained in $\mathcal{K}$. We thus have $\mathrm{vol}[\mathcal{E}_{i_c}^*(\mathcal{K})] \geq \mathrm{vol}[\mathcal{E}(\mathbf{c},\mathbf{A})]$, which together with Eqs. (8.12) and (8.13) states the second part of the proof. □

We can thus increase the volume of an inner ellipsoid $\mathcal{E}(\mathbf{c},\mathbf{A})$ for $\mathcal{K}$ through the construction of a minimum-volume outer ellipsoid for $T \circ \tau_c(\mathcal{K})$. From Lemma 8.1 (ii), this set is known analytically when $\mathcal{K}$ is a polytope characterized by linear inequalities. Theorem 8.4 will be the cornerstone of the algorithms described in the next section.

8.4. MAXIMUM-VOLUME INNER ELLIPSOID

Throughout this section, $\mathcal{K}$ is assumed to be convex and bounded, so that there exists a unique ellipsoid $\mathcal{E}_i^*(\mathcal{K})$ of maximal volume contained in $\mathcal{K}$. First assume that $\mathcal{K}$ is a polytope.

8.4.1. Polytopic Case

Consider a polytope defined by

$$\mathcal{K} = \{\mathbf{x} \in \mathbb{R}^p \mid \mathbf{a}_i^T\mathbf{x} \leq b_i, i = 1, \ldots, m\}. \tag{8.14}$$

We suggest the following algorithm for solving $P_i(\mathcal{K})$.

Algorithm for problem P_i for polytopes (AP_iP)

Step (i): Choose ε such that $0 < \varepsilon << 1$, and $\mathbf{c}^0 \in \mathrm{int}(\mathcal{K})$. Set $k = 0$.

Step (ii): Compute the m vectors

$$\mathbf{v}_i^k := \frac{p\mathbf{a}_i}{b_i - \mathbf{a}_i^T\mathbf{c}^k}, \; i = 1, \ldots, m, \tag{8.15}$$

and the minimum-volume ellipsoid $\mathcal{E}(\mathbf{c}^{*k},\mathbf{A}^{*k})$ containing them (using AP_o of Section 8.2).

Step (iii): Compute

$$\mathbf{e}^k := \frac{p\mathbf{A}^{*k}\mathbf{c}^{*k}}{p - \mathbf{c}^{*k^T}\mathbf{A}^{*k}\mathbf{c}^{*k}}, \tag{8.16}$$

$$\mathbf{c}^{k+1} := \mathbf{c}^k + \mathbf{e}^k . \tag{8.17}$$

If $||\mathbf{e}^k|| < \varepsilon$, compute

$$\mathbf{B}^{k+1} := (\mathbf{A}^{*k^{-1}} - \frac{\mathbf{c}^{*k}\mathbf{c}^{*k^T}}{p}) \frac{(p - \mathbf{c}^{*k^T}\mathbf{A}^{*k}\mathbf{c}^{*k})}{p}, \tag{8.18}$$

take $\mathcal{E}(\mathbf{c}^{k+1},\mathbf{B}^{k+1})$ as an approximation of $^*\mathcal{E}_i^*(\mathcal{K})$; else $k \leftarrow k + 1$, go to Step (ii).

When the vertices of $\mathcal{K}$ are not known, the choice of $\mathbf{c}^0$ at Step (i) may be nontrivial (because $\mathbf{c}^0$ must not belong to the boundary of $\mathcal{K}$). However, $\mathbf{c}^0$ can be obtained through the construction of a series of outer ellipsoids. For instance, the following procedure, based on the shallow-cut ellipsoid method,(25) yields an ellipsoid contained in $\mathcal{K}$.

Step (0-i): Choose $\mathbf{c}^0$, $\mathbf{B}^0$ such that $\mathcal{K} \subset \mathcal{E}[\mathbf{c}^0,(\mathbf{B}^0)^{-1}]$, set $k = 0$. Compute

$$\rho := \frac{1}{(p+1)^2}, \sigma := \frac{p^3(p+2)}{(p+1)^3(p-1)}, \tau := \frac{2}{p(p+1)}, \zeta := 1 + \frac{1}{2p^2(p+1)^2} .$$

Step (0-ii): Compute

$$r = \min_i \left\{ b_i - \mathbf{a}_i^T\mathbf{c}^k - \frac{[p\mathbf{a}_i^T(\mathbf{B}^k)^{-1}\mathbf{a}_i]^{1/2}}{p+1} \right\} .$$

Let j be the argument of the minimum. If $r \geq 0$, stop; we have:

$$\mathcal{E}(\mathbf{c}^k,(p+1)^2\mathbf{B}^{k^{-1}}) \subset \mathcal{K}$$

(and $\mathcal{K} \subset \mathcal{E}[\mathbf{c}^k(\mathbf{B}^k)^{-1}]$).

Step (0–iii): Compute

$$\mathbf{c}^{k+1} = \mathbf{c}^k - \rho \frac{p^{1/2}\mathbf{B}^k\mathbf{a}_j}{(\mathbf{a}_j^T\mathbf{B}^k\mathbf{a}_j)^{1/2}},$$

$k \leftarrow k + 1$, go to Step (0-ii).

This procedure terminates in a finite number of steps.(25) Note that the condition $\mathcal{K} \subset \mathcal{E}(\mathbf{c}^0,(\mathbf{B})^{0^{-1}})$ of Step (0-i) is easily fulfilled by choosing $\mathbf{B}^0 = \gamma I_p$, with I_p the p-dimensional identity matrix and γ large enough. Much faster ellipsoidal procedures can also be considered, such as the deep-cut or central-cut algorithms.(26,27) However, one must then check that $\mathbf{c}^0$ thus obtained does not lie on the boundary of $\mathcal{K}$.

The algorithm AP_iP has been independently suggested in Ref. 10, with no proof of convergence. The following result provides such a proof.

THEOREM 8.5. For any choice of $\mathbf{c}^0 \in \operatorname{int}(\mathcal{K})$ AP_iP generates a sequence of inner ellipsoids $\mathcal{E}(\mathbf{c}^k,\mathbf{B}^k)$ (where $\mathbf{c}^k,\mathbf{B}^k$ are respectively defined by Eq. (8.17) and (8.18)) converging monotonically in the sense of the volume to $\mathcal{E}_i^*(\mathcal{K})$.

PROOF. We first prove that AP_iP generates a sequence of inner ellipsoids with monotonically increasing volume. Assume that $\mathbf{c}^k \in \operatorname{int}(\mathcal{K})$; we thus have $O \in \operatorname{int}[\tau_{c^k}(\mathcal{K})]$, with

$$\tau_{c^k}(\mathcal{K}) = \{\mathbf{x} \in \mathbb{R}^p \mid \mathbf{a}_i^T\mathbf{x} \leq b_i - \mathbf{a}_i^T\mathbf{c}^k,\ i = 1, \ldots, m\}.$$

From Lemma 8.1 (ii), the vectors $\mathbf{v}_i^k$, $i = 1, \ldots, m$ of Eq. (8.15) then correspond to all possible vertices of $T \circ \tau_{c^k}(\mathcal{K})$, and $\mathcal{E}(\mathbf{c}^{*k},\mathbf{A}^{*k}) = \mathcal{E}_o^*(T \circ \tau_{c^k}(\mathcal{K}))$. From Lemma 8.1 (iv) and the matrix inversion lemma, $\mathcal{E}(\mathbf{e}^k,\mathbf{B}^{k+1}) = T\{\mathcal{E}_o^*[T \circ \tau_{c^k}(\mathcal{K})]\}$, and $\mathcal{E}(\mathbf{c}^{k+1},\mathbf{B}^{k+1}) = \tau_{-c^k} \circ T\{\mathcal{E}_o^*[T \circ \tau_{c^k}(\mathcal{K})]\}$. Theorem 8.4 then yields the result.

The sequence of volumes $\operatorname{vol}[\mathcal{E}(\mathbf{c}^k,\mathbf{B}^k)]$ is monotonously increasing and bounded by $\operatorname{vol}(\mathcal{K})$, thus it converges. Let $\mathcal{E}(\mathbf{c}^k,\mathbf{B}^k)$ be such that

$$\operatorname{vol}[\mathcal{E}(\mathbf{c}^k,\mathbf{B}^k)] = \operatorname{vol}[\mathcal{E}(\mathbf{c}^{k+1},\mathbf{B}^{k+1})]. \tag{8.19}$$

The inequality (8.11) then becomes an equality, and

$$\begin{aligned}&\operatorname{vol}\{\mathcal{E}_o^*[T \circ \tau_{c^k}(\mathcal{K})]\}\operatorname{vol}(T\{\mathcal{E}_o^*[T \circ \tau_{c^k}(\mathcal{K})]\})\\ &\qquad = \operatorname{vol}[\mathcal{E}(\mathbf{c}^{*k},\mathbf{A}^{*k})]\operatorname{vol}\{T[\mathcal{E}\mathbf{c}^{*k},\mathbf{A}^{*k})]\} = \beta^2(p),\end{aligned}$$

which from Eq. (8.9) implies $\mathbf{c}^{*k} = \mathbf{0}$ and thus $\mathbf{c}^{k+1} = \mathbf{c}^k$, $\mathbf{B}^{k+1} = (\mathbf{A}^{*k})^{-1}$. By construction, $\mathcal{E}(\mathbf{c}^{*k},\mathbf{A}^{*k})$ is the minimum-volume ellipsoid containing the vectors $\mathbf{v}_i^k$, $i = 1$, $\ldots, m$. From the results given in Section 8.2 we thus have[2]

$$\mathbf{c}^{*k} = \sum_{i=1}^{m} \lambda_i^* \mathbf{v}_i^k = \mathbf{0},$$

$$\mathbf{A}^{*k} = \left(\sum_{i=1}^{m} \lambda_i^* \mathbf{v}_i^k \mathbf{v}_i^{k^T} - \mathbf{c}^{*k}\mathbf{c}^{*k^T}\right)^{-1} = \left(\sum_{i=1}^{m} \lambda_i^* \mathbf{v}_i^k \mathbf{v}_i^{k^T}\right)^{-1},$$

$$\lambda_i^*(\mathbf{v}_i^{k^T}\mathbf{A}^{*k}\mathbf{v}_i^k - p) = 0,\ i = 1, \ldots, m, \tag{8.20}$$

$$\lambda_i^* \geq 0,\ i = 1, \ldots, m, \tag{8.21}$$

and, moreover, $\Sigma_{i=1}^m \lambda_i^* = 1$. Using the definition of $\mathbf{v}_i^k$, see Eq. (8.15),

$$\sum_{i=1}^{m} \lambda_i^* \frac{\mathbf{a}_i}{b_i - \mathbf{a}_i^T \mathbf{c}^k} = \mathbf{0} \tag{8.22}$$

$$\mathbf{B}^{k+1} = \sum_{i=1}^{m} \lambda_i^* \frac{p \mathbf{a}_i \mathbf{a}_i^T}{\mathbf{a}_i^T \mathbf{A}^{*k} \mathbf{a}_i}. \tag{8.23}$$

Lagrangian theory will now be used to show that $\mathcal{E}(\mathbf{c}^{k+1},\mathbf{B}^{k+1})$ coincides with $\mathcal{E}_i^*(\mathcal{K})$, which completes the proof of convergence.

Determining $\mathcal{E}_i^*(\mathcal{K}) = \mathcal{E}(\mathbf{c}^*,\mathbf{B}^*)$ corresponds to minimizing ln det **B** with respect to **B** and **c**, with the constraints

$$\mathbf{a}_i^T \mathbf{x} \le b_i, i = 1, \dots, m \quad \forall\, \mathbf{x} \in \mathcal{E}(\mathbf{c},\mathbf{B}),$$

or equivalently, to minimizing-ln det **A**, with $\mathbf{A} = \mathbf{B}^{-1}$ and the constraints

$$\mathbf{a}_i^T \mathbf{c} + (p \mathbf{a}_i^T \mathbf{A} \mathbf{a}_i)^{1/2} \le b_i, i = 1, \dots, m. \tag{8.24}$$

The function –ln det **A** is convex in **A**, but unfortunately the feasible set for **A** is not convex. However, **A** must belong to the set $\mathbf{M}_{p^+}$ of all $p \times p$ symmetric positive definite matrices. Elementary calculations then show that setting $\mathbf{A} = \mathbf{H}^T\mathbf{H}$ yields a convex feasible set for **H**. The constraints of Eq. (8.24) can then be written as

$$p \ln(p \mathbf{a}_i^T \mathbf{H}^T \mathbf{H} \mathbf{a}_i) \le 2p \ln(b_i - \mathbf{a}_i^T \mathbf{c}), \quad i = 1, \dots, m,$$

which yields the Lagrangian

$$\mathcal{L}(\mathbf{H},\mathbf{c},\lambda) = -\ln \det(\mathbf{H}^T\mathbf{H}) + \sum_{i=1}^{m} \lambda_i \left[p \ln (p \mathbf{a}_i^T \mathbf{H}^T \mathbf{H} \mathbf{a}_i) - 2p\ln(b_i - \mathbf{a}_i^T \mathbf{c})\right].$$

From the Kuhn–Tucker theorem, we know that the solution of the problem is obtained for **H***, **c***, λ^* such that

$$\frac{\partial \mathcal{L}(\mathbf{H},\mathbf{c},\lambda)}{\partial \mathbf{H}}\Big|_{\mathbf{H}^*,\mathbf{c}^*,\lambda^*} = \mathbf{0}, \tag{8.25}$$

$$\frac{\partial \mathcal{L}(\mathbf{H},\mathbf{c},\lambda)}{\partial \mathbf{c}}\Big|_{\mathbf{H}^*,\mathbf{c}^*,\lambda^*} = \mathbf{0}, \quad \text{and}$$

$$\lambda_i^* \left[\ln(p \mathbf{a}_i^T \mathbf{H}^{*T} \mathbf{H}^* \mathbf{a}_i) - 2 \ln(b_i - \mathbf{a}_i^T \mathbf{c}^*)\right] \le 0, \quad i = 1, \dots, m,$$

$$\lambda_i^* \ge 0, \quad i = 1, \dots, m.$$

The first condition Eq. (8.25) can also be written

$$\frac{\partial \mathcal{L}(\mathbf{H},\mathbf{c},\lambda)}{\partial \mathbf{A}}\Big|_{\mathbf{H}^*,\mathbf{c}^*,\lambda^*}\frac{\partial \mathbf{A}}{\partial \mathbf{H}}\Big|_{\mathbf{H}^*}=\mathbf{0}.$$

One can then easily check that λ^*, $\mathbf{A}^{k^*}=(\mathbf{B}^{k+1})^{-1}$ and $\mathbf{c}^{k+1}=\mathbf{c}^k$ satisfying Eqs. (8.20–8.23) are solutions. □

Remark 8.2.

(i) The stopping rule in Step (iii) of AP_iP could also rely on the difference between two consecutive volumes, on $\|\mathbf{c}^{*k}\|$, or on $\mathbf{c}^{*k^T}\mathbf{A}^{*k}\mathbf{c}^{*k}$.[9] Further studies are required to investigate whether the corresponding sequences decrease monotonically.

(ii) The proof assumed that $\mathcal{E}_o^*[T_c^k(\mathcal{K})]$ can be obtained without any approximation using AP_o (Step (ii)). In practice, this algorithm contains an ε-stopping rule so that $\mathcal{E}_o^*(T_c^k(\mathcal{K}))$ is not obtained exactly. Practical rules for choosing an ε′ at Step (iii) of AP_iP could be derived from the general ideas presented in Ref. 28.

(iii) The distribution ξ^0 used to initialize AP_o can be taken equal to the optimal distribution corresponding to previous ellipsoid $\mathcal{E}(\mathbf{c}^{*k-1},\mathbf{A}^{*k-1})$.

(iv) The determination of $\mathcal{E}_i^*(\mathcal{K})$ does not require the calculation of the vertices of $\mathcal{K}$. The complexity is related to the dimension of the vector of weights λ in AP_o. This dimension increases at most linearly with the number of constraints that define $\mathcal{K}$

(v) The complexity of the solution of $P_i(\mathcal{K})$ is considered in Ref. 8.10, where another algorithm is suggested, also based on the solution of a sequence of subproblems (see also Ref. 29).

(vi) The determination of the maximum-volume inner sphere for the polytope $\mathcal{K}$corresponds to a linear-programming problem:[8] let $\mathbf{c}$ and r respectively denote the center and the radius of the maximum inscribed sphere, one has to maximize r with the constraints $\mathbf{a}_i^T\mathbf{c}+r\|\mathbf{a}_i\|\le b_i$, $i=1,\ldots,m$. Maximum-volume inner ellipsoids for polytopes are considered in Ref. 8.30. A signomial algorithm is suggested for the general case, and the situation where the shape of the ellipsoid is fixed is shown to correspond to a linear programming problem.

Example 1:

Consider the AR-2 model

$$y(k)=-0.4y(k-1)-0.85y(k-2)+\varepsilon(k),\ k=3,\ldots,25,$$

$$y(1)=\varepsilon(1),y(2)=\varepsilon(2),$$

with $\varepsilon(k)$ uniformly distributed in $[-1,1]$, $k=1,\ldots,25$. It corresponds to a linear model structure, $\eta(\theta,k)=\mathbf{a}_k^T\theta$, $\mathbf{a}_k=[y(k-1),y(k-2)]^T$, with bounded disturbances $\varepsilon(k)$. The true value of the parameters, given by $\theta^*=(-0.4,-0.85)^T$, is indicated by

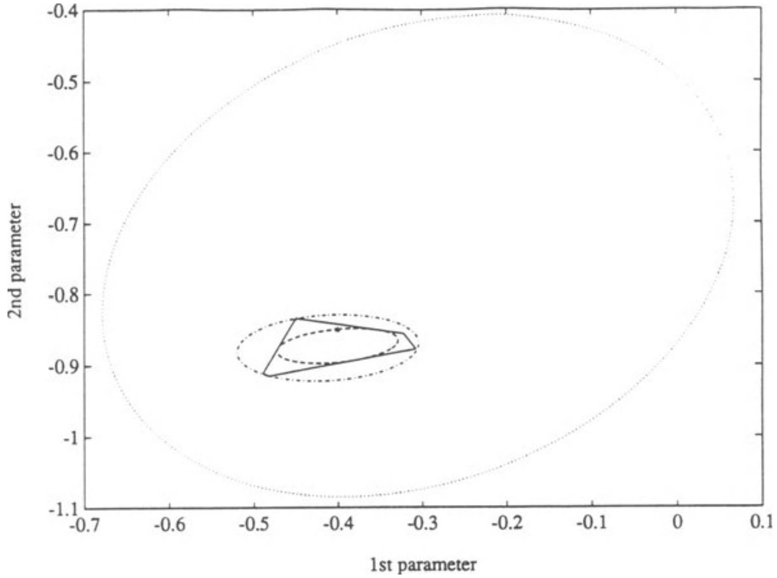


FIGURE 8.2. $\mathcal{S}$, recursive outer and volume-optimal inner and outer ellipsoids.

a star on Fig. 8.2. We are interested in characterizing the set $\mathcal{S}$ of all parameter vectors consistent with the data, model structure and error bounds, given by

$$\mathcal{S} = \{\theta \in \mathbb{R}^2 \mid -1 \leq y(k) - \mathbf{a}_k^T\theta \leq 1, k = 1, \ldots, 25\}.$$

The same example is considered in Ref. 22, where various approaches are used to give an ellipsoidal outer bound for $\mathcal{S}$. The ellipsoid $\mathcal{E}(N)$ obtained via the classical recursive procedure of Fogel and Huang,(31) even improved according to Ref. 32, appears very pessimistic (see Figure 8.2). Recirculations of data(33) yield smaller outer ellipsoids. However, even when the number of recirculations gets very large, the ellipsoid obtained is still significantly larger than the minimum-volume outer ellipsoid $\mathcal{E}_o^*(\mathcal{S})$ (see also Examples 2 and 3). The set $\mathcal{S}$ and the ellipsoid $\mathcal{E}_o^*(\mathcal{S})$, obtained via AP_o of Section 8.2, are given in Fig. 8.2. Figure 8.2 also presents the maximum-volume inner ellipsoid $\mathcal{E}_i^*(\mathcal{S})$ obtained after three iterations of AP_iP.

Example 2: Consider the AR-3 model described by

$$y(k) = -0.5y(k-1) - 0.75y(k-2) + 0.1y(k-3) + \varepsilon(k), k = 4, \ldots, 100,$$

$$y(k) = \varepsilon(k), k = 1, 2, 3,$$

with $\varepsilon(k)$ uniformly distributed in $[-1, 1]$, $k = 1, \ldots, 100$. For a typical simulation, the following results were obtained. The exact polytope $\mathcal{S}$ has 20 vertices. Table

TABLE 8.1. Volumes and Centers of Outer and Inner Ellipsoids, Example 2

Ellipsoid	Center	Volume
$\mathcal{E}(N)$	(−0.328, −0.622, 0.219)	2.47×10^{-1}
$\mathcal{E}(N,100)$	(−0.508, −0.776, 0.044)	4.45×10^{-3}
$\mathcal{E}_o^*(\mathcal{S})$	(−0.500, −0.772, 0.058)	1.09×10^{-3}
$\mathcal{E}_i^*(\mathcal{S})$	(−0.509, −0.778, 0.043)	1.60×10^{-4}

8.1 gives the volumes and centers of the recursive ellipsoid $\mathcal{E}(N)$ determined with the algorithm described in Ref. 8.32, the recursive ellipsoid $\mathcal{E}(N, 100)$ obtained after 100 circulations of the data, and the ellipsoids $\mathcal{E}_o^*(\mathcal{S})$ and $\mathcal{E}_i^*(\mathcal{S})$. For comparison, the exact volume of $\mathcal{S}$ (calculated with the algorithm given in Ref. 8.34) is 3.63×10^{-4}.

Example 3: Consider the AR-5 model described by

$$\begin{aligned} y(k) &= -0.4y(k-1) - 0.85y(k-2) - 0.1y(k-3) - 0.02y(k-4) - 0.05y(k-5) \\ &\quad + \varepsilon(k),\ k = 6, \ldots, 100, \\ y(k) &= \varepsilon(k),\ k = 1, \ldots, 5, \end{aligned}$$

with $\varepsilon(k)$ uniformly distributed in $[-1,1]$, $k = 1, \ldots, 100$. For a typical simulation, the following results were obtained. The exact polytope $\mathcal{S}$ has 132 vertices. Table 8.2 gives the volumes and centers of the recursive ellipsoid $\mathcal{E}(N)$ determined with the algorithm described in Ref. 8.32, the recursive ellipsoid $\mathcal{E}(N, 100)$ obtained after 100 circulations of the data, and the ellipsoids $\mathcal{E}_o^*(\mathcal{S})$ and $\mathcal{E}_i^*(\mathcal{S})$.

We consider now the more general case where $\mathcal{K}$ is any bounded convex set of $\mathbb{R}^p$.

TABLE 8.2. Volumes and Centers of Outer and Inner Ellipsoids, Example 3

Ellipsoid	Center	Volume
$\mathcal{E}(N)$	(−0.343, −0.874, 0.053, −0.029, 0.163)	1.39
$\mathcal{E}(N,100)$	(−0.387, −0.886, −0.064, 0.006, −0.009)	1.33×10^{-3}
$\mathcal{E}_o^*(\mathcal{S})$	(−0.383, −0.848, −0.059, 0.014, −0.016)	4.93×10^{-5}
$\mathcal{E}_i^*(\mathcal{S})$	(−0.389, −0.866, −0.066, 0.006, −0.010)	3.93×10^{-6}

8.4.2. Convex Sets: General Case

When some of the constraints defining $\mathcal{K}$ are nonlinear, it remains possible, using a relaxation procedure, to derive a globally convergent algorithm. However, the sequence of ellipsoids will no longer be contained in $\mathcal{K}$.

$\mathcal{K}$ is the convex closure of its extreme points, the number of which may now be infinite. Let $\mathcal{X}(\mathcal{K})$ denote the set of all extreme points of $\mathcal{K}$. With any $\mathbf{x} \in \mathcal{X}(\mathcal{K})$ one can associate (at least) one supporting hyperplane $\mathcal{H}$ tangent to $\mathcal{K}$. Let $\mathbf{a}^T\mathbf{x} = b$ be the equation of any one of these hyperplanes, with $\mathbf{a}$ and b such that $\mathbf{a}^T\mathbf{x} \leq b$ $\forall\, \mathbf{x} \in \mathcal{K}$. $\mathcal{K}$ is thus included in any polytope $\mathcal{P}(\mathcal{H}_1, \ldots, \mathcal{H}_m)$ defined by m such hyperplanes, i.e., given by Eq. (8.14). The relaxation procedure consists in taking only a finite number of hyperplanes into account at each iteration, thereby constructing a sequence of polytopes containing $\mathcal{K}$ and a sequence of inner ellipsoids for these polytopes.

Algorithm for problem P_i for bounded Convex Sets (AP_iCS)

Step (i) Choose m extreme points of $\mathcal{K}$, $m \geq p$, such that the corresponding supporting hyperplanes $\mathcal{H}_i$ yield a closed and bounded polyhedron $\mathcal{P}^m = \mathcal{P}(\mathcal{H}_1, \ldots, \mathcal{H}_m)$ (i.e., such that the vectors $\mathbf{a}_i = 1, \ldots, m$, span $\mathbb{R}^p$). Choose ε such that $0 < \varepsilon << 1$, set $k = m$.

Step (ii) Determine $\mathcal{E}_i^*(\mathcal{P}^k) = \mathcal{E}(\mathbf{c}^k, \mathbf{B}^k)$ using AP_iP.

Step (iii) Compute

$$\mathbf{x}^- := \arg \min_{\mathbf{x} \in \mathcal{X}(\mathcal{K})} (\mathbf{x} - \mathbf{c}^k)^T\mathbf{B}^k(\mathbf{x} - \mathbf{c}^k). \tag{8.26}$$

If $(\mathbf{x}^- - \mathbf{c}^k)^T\mathbf{B}^k(\mathbf{x}^- - \mathbf{c}^k) > p - \varepsilon$, stop: take $\mathcal{E}(\mathbf{c}^k,\mathbf{B}^k)$ as an approximation of $\mathcal{E}_i^*(\mathcal{K})$. Otherwise determine the supporting hyperplane $\mathcal{H}_{k+1}$ passing through $\mathbf{x}^-$, take $\mathcal{P}^{k+1} = \mathcal{P}(\mathcal{H}_1, \ldots, \mathcal{H}_k, \mathcal{H}_{k+1})$, $k \leftarrow k + 1$, go to Step (ii).

The global convergence of AP_iCS follows from the convexity of $\mathcal{K}$ and the convergence of AP_iP. The main difficulty lies in the computation of $\mathbf{x}^-$ in Eq. (8.26), which corresponds to a nonconvex minimization problem. Global optimization methods(35) (e.g., based upon interval analysis(36)) are advisable. It is sometimes enough to consider a combination of local minimizations only (one minimization per constraint defining $\mathcal{K}$). Note that a precise determination of the global minimum is not crucial (as it is enough to determine $\mathbf{x}^-$ such that $(\mathbf{x}^- - \mathbf{c}^k)^T\mathbf{B}^k(\mathbf{x}^- - \mathbf{c}^k)$ $\leq p - \varepsilon$).

REMARK 8.3.

(i) During the successive calls to AP_iP, the center $\mathbf{c}^k$ of the ellipsoid previously determined can be taken as the initial vector $\mathbf{c}^0$. The number of vectors $\mathbf{v}_i^k$ in Eq. (8.15) increases by one at each iteration. When AP_iP calls AP_o for the first time (see Remark 8.2 (iii)), the initial distribution ξ^0 can be chosen to give the same weights to the new vector $\mathbf{v}_i^k$ and the vectors that were support points for the previous ellipsoid.

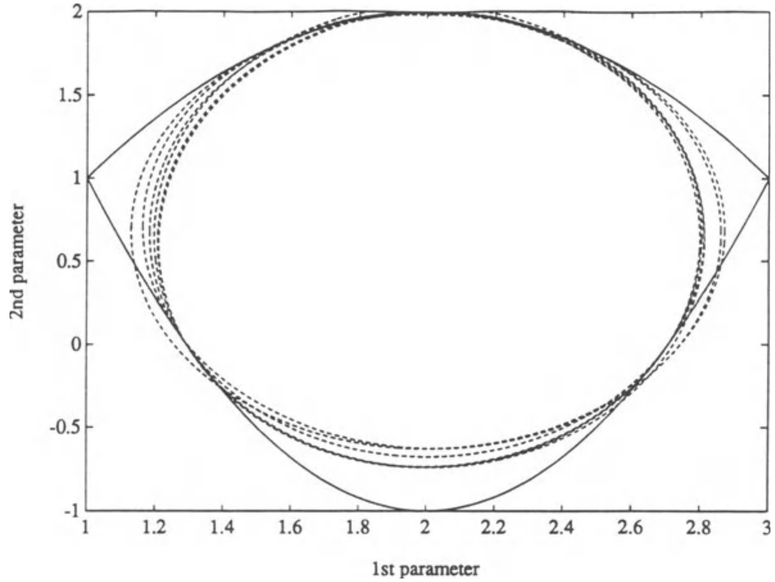


FIGURE 8.3. Target set (solid line) and sequence of ellipsoids converging to the minimum-volume inner ellipsoid.

(ii) Remark 2 (ii) applies for this algorithm too.

(iii) The sequence of volumes $\{\text{vol}[\mathcal{E}(\mathbf{c}^k,\mathbf{B}^k)]\}_k$ is monotonically decreasing.

Example 4: Consider the target set

$$\mathcal{T}= \{\mathbf{x} \in \mathbb{R}^2 \,|x_2 - 2(x_1 - 1)(x_1 - 3) \geq 1,\ x_2 + (x_1 - 2)^2 \leq 2\}.$$

This set is presented on Figure 8.3 (solid line), together with the sequence of ellipsoids $[\mathcal{E}_i^*(\mathcal{P}^k)]_k$ generated by AP_i. The initial extreme points of $\mathcal{K}$are $(\frac{3}{2}, -\frac{1}{2})$, $(\frac{3}{2}, \frac{7}{4})$, $(\frac{5}{2}, -\frac{1}{2})$, $(\frac{5}{2}, \frac{7}{4})$.

So far, we have considered the non-recursive determination of $\mathcal{E}_i^*(\mathcal{K})$, i.e., the case where all constraints defining the target set $\mathcal{K}$are taken into account at the same time. However, in practice one may wish to take any new constraints into account upon arrival. A recursive determination of $\mathcal{E}_i^*(\mathcal{K})$ is then of interest since it will reduce the computational cost and may allow real-time processing.

8.4.3. Pseudo-Recursive Algorithm

We restrict our attention to the case of linear constraints, i.e., to convex polyhedral sets. A recursive algorithm for the determination of an inner ellipsoid has already been suggested[11] in the context of parameter bounding. However, it does not yield the maximum-volume inner ellipsoid. Moreover, the ellipsoid obtained tends to vanish after a modest number of iterations (i.e., a small number of new constraints defining $\mathcal{K}$). The algorithm suggested here is not fully recursive, in the sense that the information to be stored grows with the number of constraints on $\mathcal{K}$. However, a simple test indicates whether new constraints can be rejected or not.

The constraints $\mathbf{a}_i^T\mathbf{x} \leq b_i$ defining $\mathcal{K}$ will be transformed into $\mathbf{a}_i^T\mathbf{x} \leq p$ (which means that $\mathbf{0} \in \mathcal{K}$). This can always be obtained by a suitable translation in $\mathbb{R}^p$ (which may have to be performed at any iteration when the problem occurs). Let $\mathcal{K}^k$ be the polyhedron defined by the first k constraints,

$$\mathcal{K}^k = \{\mathbf{x} \in \mathbb{R}^p \mid \mathbf{a}_i^T\mathbf{x} \leq p, i = 1, \dots, k\},$$

and let $\mathcal{L}^k$ be the set of extreme points of the convex closure of the vectors $\mathbf{a}_i$, $i = 1, \dots, k$ (i.e., corresponding to active constraints). The algorithm can be summarized as follows.

Recursive algorithm for problem P_i for Polytopes (RAP_iP)
Initialization: Consider the first k vectors $\mathbf{a}_i$ that span $\mathbb{R}^p$ $(k \geq p + 1)$. Compute

$$\mathcal{E}_i^*(\mathcal{K}^k) = \mathcal{E}(\mathbf{c}^k, \mathbf{B}^k) \text{ (using } AP_iP\text{) and } \mathcal{L}^k.$$

Iteration: Let $\mathbf{a}_{k+1}^T\ \mathbf{x} \leq p$ be the new constraint.

If $\mathbf{a}_{k+1} \in \overline{\mathcal{L}}^k$ (the convex closure of $\mathcal{L}^k$), set

$$\mathcal{E}_i^*(\mathcal{K}^{k+1}) = \mathcal{E}_i^*(\mathcal{K}^k),\ \mathcal{L}^{k+1} = \mathcal{L}^k.$$

Otherwise take $\mathcal{L}^{k+1}$ as the set of extreme points of $\mathcal{L}^k \cup \{\mathbf{a}_{k+1}\}$, if

$$\mathbf{a}_{k+1}^T\mathbf{c}^k + [p\mathbf{a}_{k+1}^T(\mathbf{B}^k)^{-1}\mathbf{a}_{k+1}]^{1/2} \leq p, \tag{8.27}$$

$$\text{set } \mathcal{E}_i^*(\mathcal{K}^{k+1}) = \mathcal{E}_i^*(\mathcal{K}^k),$$

otherwise, compute $\mathcal{E}_i^*(\mathcal{K}^{k+1})$ by applying AP_iP to the polytope defined by the vectors $\mathbf{a}_i$ in $\mathcal{L}^{k+1}$.

AP_iP is based on the construction of $\mathcal{E}_o^*[T \circ \tau_c(\mathcal{K}^k)]$, with $\mathbf{c} \in \mathcal{K}^k$,

$$\tau_c(\mathcal{K}^k) = \{\mathbf{x} \in \mathbb{R}^p \mid \mathbf{a}_i^T\mathbf{x} \leq p - \mathbf{a}_i^T\mathbf{c}, i = 1, \dots, k\},$$

and from Lemma 8.1 (ii), the possible vertices of $T \circ \tau_c(\mathcal{K}^k)$ are thus given by

$$\mathbf{v}_i = \frac{p\mathbf{a}_i}{p - \mathbf{a}_i^T\mathbf{c}},\ i = 1, \dots, k.$$

The active constraints of $\mathcal{K}^k$ correspond to the vectors $\mathbf{a}_i$ associated with the vertices of $T \circ \tau_c(\mathcal{K}^k)$. $\mathcal{E}_o^*[T \circ \tau_c(\mathcal{K}^k)]$ only depends on these vertices, i.e., the points lying on the boundary of the convex closure of the vectors $\mathbf{v}_i$. One can easily check that for any $\mathbf{c}$ in int$(\mathcal{K}^k)$, these extreme $\mathbf{v}_i$ in the dual space are associated with the extreme $\mathbf{a}_i$ in the primal space. Only the vectors in $\mathcal{L}^k$ have thus to be stored. Now, if

$\mathbf{a}_{k+1} \in \overline{\mathcal{L}}^{k^c}$, then $\mathcal{K}^{k+1} = \mathcal{K}^k$, otherwise $\mathcal{L}^k$ has to be updated. However, it remains to be tested whether $\mathcal{E}_i^*(\mathcal{K}^k)$ is cut by the new constraint. This corresponds to Eq. (8.27).

REMARK 8.4. If the set $\mathcal{L}^k$ of active constraints is determined without computing the vertices of $\mathcal{K}$, see, e.g., Ref. 37, this algorithm gives a recursive construction of $\mathcal{E}_i^*(\mathcal{K})$ without requiring a recursive characterization of $\mathcal{K}$ through its vertices.

8.5. CONCLUSIONS

The determination of the minimum-volume ellipsoid containing a compact set $\mathcal{K} \subset \mathbb{R}^p$ (problem P_o) is strongly connected to experimental design, and an efficient algorithm has already been suggested.[2,15,22] It can be used when $\mathcal{K}$ is not convex (one then has to solve a series of p-dimensional global-optimization problems), while more traditional approaches based on convex programming seem to be restricted to the case where $\mathcal{K}$ is a polytope. This optimal ellipsoidal outer approximation might prove particularly useful in parameter bounding, where large uncertainty sets lead to conservative robust control laws.[3] When the model is linear in the parameters $\mathcal{K}$ is a polytope, the description of which might reveal very complex. The algorithm presented permits reducing this complexity drastically, but still requires the exact description to be obtained.

The determination of the maximum-volume ellipsoid contained in a polytope $\mathcal{K}$ can be performed through the solution of a series of problems P_o. It does not require the knowledge of the vertices of $\mathcal{K}$. Other algorithms can also be used for that purpose.[10,29,38] Further studies are required concerning the complexity of the algorithm presented here, and its potential interest in nonlinear programming.[9,38]

When the polytope is constructed recursively, the recursive determination of the volume-maximal inner ellipsoid requires storage of a possibly growing amount of information (corresponding to the active constraints). The only approach suggested so far to the best of our knowledge does not yield an ellipsoid with maximum volume.[11] Moreover, the inner ellipsoid obtained tends to quickly vanish when the number of linear constraints increases. This is not the case with the maximum-volume inner ellipsoid as obtained from the algorithm suggested here. Finally, when $\mathcal{K}$ is only convex, the problem P_i can be decomposed into a series of problems involving polytopes, and global convergence to the maximum inscribed ellipsoid can still be guaranteed.

REFERENCES

1. E. Walter, editor. *Mathematics and Computers in Simulation* **32**(5,6) (1990).
2. L. Pronzato and E. Walter, *Int. J. Adapt. Control Sig. Proc.* **8**, 15 (1994).
3. B. Wahlberg and L. Ljung, *IEEE Trans. Autom. Control* **AC-37**, 900 (1992).

4. B. W. Silverman and D. M. Titterington, *SIAM J. Sci. Stat. Comput.* **1**, 401 (1980).
5. D. M. Titterington, *J. Royal Stat. Soc.* **C27**(3), 227 (1978).
6. L. Davies, An. Stat. **20**, 1828 (1992).
7. R. D. Cook, D. M. Hawkins, and S. Weisberg, *Stat. Prob. Lett.* **16**, 213 (1993).
8. E. M. T. Hendrix, C. J. Mecking, and T. H. B. Hendriks, *A Mathematical Formulation of Finding Robust Solutions for a Product Design Problem*, Technical Report 93-02, Wageningen Agricultural University, Department of Mathematics, 6703HA Wageningen, The Netherlands (1993).
9. S. P. Tarasov, L.G. Khachiyan, and I. I. Erlich, *Sov. Math. Dokl.* **37**, 226 (1988).
10. L. G. Khachiyan and M. J. Todd, *On the Complexity of Approximating the Maximal Inscribed Ellipsoid for a Polytope*, Technical Report 893, School of Operations Research and Industrial Engineering, College of Engineering, Cornell University, Ithaca, New York (1990).
11. J. P. Norton, *Int. J. Control* **50**, 2423 (1989).
12. M. Berger, *Géométrie*, CEDIC/Nathan, Paris (1979).
13. V. L. Klee, editor. L. Danzer, B. Grünbaum, and V. Klee, in: *Proceedings of Symposia in Pure Mathematics, Convexity,* Ann. Math. Soc., Providence, Rhode Island (1963). vol. VII, pp. 101–180.
14. R. Sibson, *J. Roy. Stat. Soc.* **B34**, 181 (1972).
15. D. M. Titterington, *Biometrika* **62**, 313 (1975).
16. E. Welzl, in: *Lecture Notes in Computer Science Vol. 555*, Springer-Verlag, Berlin, pp. 359–370 (1991).
17. K. Kiefer and J. Wolfowitz, *Can. J. Math.* **12**, 363 (1960).
18. S. D. Silvey, *Optimal Design*, Chapman & Hall, London (1980).
19. V. V. Fedorov, *Theory of Optimal Experiments*, Academic Press, New York (1972).
20. C. L. Atwood, *An. Stat.* **4**, 1124 (1976).
21. C. F. Wu, *An. Stat.* **6**, 1286 (1978).
22. L. Pronzato and E. Walter, *Automatica* **30**, 1731 (1994).
23. D. J. Elzinga and D. W. Hearn, *Manag. Sci.* **19**, 96 (1972).
24. N. D. Botkin and V. L. Turova-Botkina, *An Algorithm for Finding the Chebyshev Center of a Convex Polyhedron*, Technical Report 395, Institüt für Angewandte Mathematik und Statistik, Universität Würzburg, W-8700 Würzburg (1992).
25. M. Grötschel, L. Lovász, and A. Schrijver, *Geometric Algorithms and Combinatorial Optimization*, Springer, Berlin (1980).
26. R. G. Bland, D. Goldfarb, and M. J. Todd, *Op. Res.* **29**, 1039 (1981).
27. L. Pronzato, E. Walter, and H. Piet-Lahanier, in: *Proceedings of the 28th IEEE Conference on Decision and Control*, Tampa, FL, pp. 1952–1955 (1989).
28. E. Polak, *Computational Methods in Optimization: a Unified Approach,* Academic Press, New York (1971).
29. N. Z. Shor and O. A. Berezovski, *Optim. Method. Software* **1**, 283 (1992).
30. A. Vicino and M. Milanese, *IEEE Trans. Autom. Control* **36**, 759 (1991).
31. E. Fogel and Y. F. Huang, *Automatica* **18**, 229 (1982).
32. G. Belforte and B. Bona, in: *Prep. 7th IFAC/IFORS Symposium on Identification and System Parameter Estimation*, York, pp. 1507–1512 (1985).
33. G. Belforte, B. Bona, and V. Cerone, *Automatica* **26**, 887 (1990).
34. J. B. Lasserre, *J. Optim. Theory App.* **39**, 363 (1983).
35. R. Horst and H. Tuy, *Global Optimization*, Springer, Berlin, Germany (1990).
36. H. Ratschek and J. Rokne, *New Computer Methods for Global Optimization*, Ellis Horwood Limited, Chichester (1988).
37. L. Devroye, *Inf. Proc. Lett.* **11**, 53 (1980).
38. Y. Nesterov and A. Nemirovskii, *Interior-Point Polynomial Algorithms in Convex Programming*, SIAM, Philadelphia (1994).

9

Linear Interpolation and Estimation Using Interval Analysis

S. M. Markov and E. D. Popova

ABSTRACT

This chapter considers interpolation and curve fitting using generalized polynomials under bounded measurement uncertainties from the point of view of the solution set (not the parameter set). It characterizes and presents the bounding functions for the solution set using interval arithmetic. Numerical algorithms with result verification and corresponding programs for the computation of the bounding functions in given domain are reported. Some examples are presented.

9.1. INTRODUCTION: FORMULATION OF THE PROBLEM

We consider the problems of interpolation and curve fitting in the presence of unknown but bounded errors in the output measurements. Let $\eta(\lambda;\cdot) : D \rightarrow R, D \subseteq R^k$, be a model function depending on a real argument $\xi \in D$, and on a parameter vector $\lambda \in \Lambda \subset R^m$. The following hypotheses are assumed:[1,2,3,4]

S. M. Markov and E. D. Popova • Division of Mathematical Modelling in Biology, Institute of Biophysics, Bulgarian Academy of Sciences, BG-1113 Sofia, Bulgaria.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

Assumptions on the modeling function: The modeling function $\eta(\lambda;\cdot)$ defined on some domain $D \subseteq R^k$ is a generalized polynomial depending linearly on m parameters:

$$\eta(\lambda;\xi) = \sum_{i=1}^{m} \lambda_i \varphi_i(\xi) = \varphi(\xi)^{\mathrm{T}} \lambda, \quad \xi \in D, \tag{9.1}$$

where $\varphi(\cdot) = [\varphi_1(\cdot), \ldots, \varphi_m(\cdot)]^{\mathrm{T}}$ is a vector of m continuous on D functions and $\lambda = (\lambda_1, \ldots, \lambda_m)^{\mathrm{T}} \in R^m$ is a vector of m (unknown) parameters. For any $(x_1', \ldots, x_m'), x_i' \in D, i = 1, \ldots, n$, the vector $\varphi(\cdot)$ generates a matrix defined by

$$\begin{pmatrix} \varphi_1(x_1') & \cdots & \varphi_m(x_1') \\ \vdots & \ddots & \vdots \\ \varphi_1(x_m') & \cdots & \varphi_m(x_m') \end{pmatrix}. \tag{9.2}$$

We shall assume that (9.2) is not singular whenever $(x_1', \ldots, x_m')$ is such that $x_i' \neq x_j', i \neq j$. A set φ of functions satisfying the above assumption will be further called a (Chebyshev) system of basic functions. The class of all modeling functions of the form (9.1) where φ is a system of basic functions is denoted by $\mathcal{L}_m(D, \varphi)$ or $\mathcal{L}$.

Assumptions on the type of errors in the data: The input data are error-free and the output data errors are unknown but bounded (UBB).[5,6] This means that there are n distinct (input) data $x_j \in D \subseteq R^k, j \in J = \{1, \ldots, n\}$, and there are n (output) interval measurements $Y_j = [y_j^-, y_j^+], j \in J$, which contain the correct values of the corresponding measured quantities.

Denote the system of input data by $\mathbf{x} = (x_1, x_2, \ldots, x_n)^{\mathrm{T}} \in R^{n \times k}$ and the system of output measurements by $Y = (Y_1, \ldots, Y_n)^{\mathrm{T}} \in IR^n$, where IR^n is the set of all n-dimensional interval vectors.[7,8,9] Geometrically, the pairs $(x_j, Y_j), j \in J$, can be considered as n vertical segments in the $(k + 1)$-dimensional space $Ox_1x_2 \ldots x_ky$.

Throughout the chapter it is assumed that $m \leq n$. Section 9.3 considers the problem of finding bounds for the set of modeling functions $\eta \in \mathcal{L}_m(D)$ interpolating the vertical segments $(x_j,Y_j), j \in J$. More precisely, for a fixed $\xi \in D$, we look for the set of values at ξ of all modeling functions η interpolating the segments $(x_j,Y_j), j \in J$, that is the set:

$$\{\eta(\lambda;\xi) \mid \eta \text{ is such that } \eta(\lambda;x_j) \in Y_j, j \in J\}, \quad \xi \in D. \tag{9.3}$$

The requirement that the values of η at x_j range in the corresponding intervals Y_j leads to a system of inequalities for λ

$$\eta(\lambda;x_j) = \varphi(x_j)^{\mathrm{T}} \lambda \in Y_j, \quad j \in J, \tag{9.4}$$

which can be written in matrix form as

$$\Phi(\mathbf{x})\lambda \in Y, \tag{9.5}$$

where $\Phi(\mathbf{x})$ is the following $(n \times m)$-matrix of full rank $\Phi(\mathbf{x}) = m$:

$$\Phi(\mathbf{x}) = \begin{pmatrix} \varphi_1(x_1) & \cdots & \varphi_m(x_1) \\ \vdots & \ddots & \vdots \\ \varphi_1(x_n) & \cdots & \varphi_m(x_n) \end{pmatrix} = \begin{pmatrix} \varphi(x_1)^{\mathrm{T}} \\ \vdots \\ \varphi(x_n)^{\mathrm{T}} \end{pmatrix}.$$

In (9.4) the data **x** and Y are known; the parameter λ is unknown. We thus have to solve a system of n algebraic inclusions for the m-dimensional parameter λ. Any λ satisfying (9.4) is called a feasible parameter. Every feasible parameter λ generates a solution function $\eta(\lambda;\cdot) \in \mathcal{L}_m(D)$. Denote by Λ the set of all feasible parameters, and by $\eta(\Lambda;\xi)$ the set of values of all solution functions at $\xi \in D$, respectively

$$\Lambda = \{\lambda \in R^m \mid \Phi(\mathbf{x})\lambda \in Y\}, \tag{9.6}$$

$$\eta(\Lambda;\xi) = \{\varphi(\xi)^{\mathrm{T}}\lambda \mid \lambda \in \Lambda\}, \xi \in D. \tag{9.7}$$

The set $\eta(\Lambda;\xi)$ defined by Eq. (9.7) is an interval for any fixed $\xi \in D$. Thus Eq. (9.7) defines an interval-valued function (briefly, interval function) on D, which will be further denoted by $\eta(\mathbf{x},Y;\cdot)$. Note the difference between $\eta(\Lambda;\cdot) = \{\varphi(\cdot)^{\mathrm{T}}\lambda \mid \lambda \in \Lambda\}$ and $\eta(\mathbf{x}, Y;\cdot)$: the former is a set of solution functions defined on D (sometimes called feasible solution set), whereas the latter is an interval function defined on D. Of course for a fixed $\xi \in D$ we have $\eta(\Lambda;\xi) = \eta(\mathbf{x},Y;\xi)$. We shall be particularly concerned with characterizing and computing the bounding lower and upper functions $\eta^-(\mathbf{x},Y;\cdot)$, $\eta^+(\mathbf{x},Y;\cdot)$ of the interval function $\eta(\mathbf{x},Y;\cdot)$, which are called enveloping functions for the feasible solution set $\eta(\Lambda;\cdot)$.(10)

We can compute $\eta(\mathbf{x},Y;\xi)$ for $\xi \in D$ by solving two constrained linear optimization problems(4,6)

$$\eta(\mathbf{x},Y;\xi) = \left[\min_{\lambda\in\Lambda}\{\varphi\,(\xi)^{\mathrm{T}}\lambda\},\ \max_{\lambda\in\Lambda}\{\varphi\,(\xi)^{\mathrm{T}}\lambda\}\right]. \tag{9.8}$$

Another approach(3) is to enclose Λ by an interval vector (box) Λ^I, and then find an enclosure for $\eta(\mathbf{x},Y;\xi)$ by $\eta(\mathbf{x},Y;\xi) \subseteq \varphi(\xi)^{\mathrm{T}}\Lambda^I$.

The problem of finding/enclosing the interval function $\eta(\mathbf{x},Y;\cdot)$ is different from the problem of finding/enclosing the parameter set Λ defined by Eq. (9.6).(2,3,5,6,11) The set Λ is an m-dimensional polytope, whereas $\eta(\mathbf{x},Y;\xi)$ is a closed one-dimensional interval for a fixed ξ. The presentation or computation of $\eta(\mathbf{x},Y;\xi)$ in a given domain for ξ can be of practical importance. In the case of one-dimensional argument ξ, we characterize the interval function $\eta(\mathbf{x},Y;\cdot)$ and propose methods for its presentation and computation. A computer program written in PASCAL-SC[12] is reported, which efficiently computes the interval function $\eta(\mathbf{x},Y;\xi)$ in a given interval.

If the interpolation problem has no solution, then one often wants to solve it by choosing another family of modeling functions (e.g., by changing either the number of parameters or the system of basic functions). One may choose to reformulate the interpolation problem as a curve fitting (estimation) problem.[5,6] Assume that the inclusions of Eq. (9.4) can be violated, which practically means that the errors in the measurements are assumed to be of a stochastic nature.

Section 9.4 considers the problem of finding the set of parameters λ, respectively the set of modeling functions $\eta(\lambda;\cdot)$, such that

$$\eta(\lambda;x_j) \approx Y_j, \quad j \in J, \tag{9.9}$$

in matrix form $\Phi(\mathbf{x})\lambda \approx Y$, where the symbol $\approx$ means that the values $\eta(\lambda;x_i) = \varphi(x_j)^{\mathrm{T}}\lambda$ are "close" to the measurement intervals Y_j. For the numerical (single-valued) case $Y = y \in R^n$ the curve fitting problem (9.9) is mathematically formulated by choosing an operator (called estimator) $\phi(y)$ producing from a data set $(\mathbf{x},y)$ a solution function $\eta(\lambda_y;\cdot)$ from $\mathcal{L}_m(X)$. The operator ϕ is chosen in accordance with the hypothesis on the statistical nature of the errors in the measurements (for instance, a least-square estimator is chosen if the errors in y are assumed to have normal distribution). Let us restrict ourselves to so-called projection estimators[1,5] of the form $\phi(y) = \eta(\lambda_y;\cdot)$, with λ_y minimizing some functional of the form

$$\|y - \Phi(\mathbf{x})\lambda_y\| = \inf_{\lambda\in K}\|y - \Phi(\mathbf{x})\lambda\|, \quad K \subseteq R^m, \tag{9.10}$$

where $\|\cdot\|$ is a norm in R^n. Assume as before that the measurement interval Y contains the true values of the measured quantities. As proposed[1,5,6] consider the set of solution functions corresponding to the data $(\mathbf{x},Y)$, defined by

$$\{\eta(\lambda_y;\cdot) \mid y \in Y\} = \{\varphi(\cdot)^{\mathrm{T}}\lambda_y \mid y \in Y\}, \tag{9.11}$$

where λ_y is given in Eq. (9.10).

Let Λ_ϕ be the set of all λ_y, produced by the estimator $\phi(y)$, whenever the numerical vector y ranges in the interval measurement vector $Y = (Y_1, \ldots, Y_n)$,

$$\Lambda_\phi = \{\lambda_y \in K \subseteq R^m, \lambda_y \text{ satisfies Eq. (9.10)} \mid y \in Y\}. \tag{9.12}$$

The set Λ_ϕ is called the estimate uncertainty set.[5,6] The set Λ_ϕ generates a corresponding (estimate) solution set

$$\eta(\Lambda_\phi;\cdot) = \{\varphi(\cdot)^{\mathrm{T}}\lambda \mid \lambda \in \Lambda_\phi\}. \tag{9.13}$$

For a fixed $\xi \in D$

$$\eta(\Lambda_\phi;\xi) = \{\varphi(\xi)^{\mathrm{T}}\lambda \mid \lambda \in \Lambda_\phi\}, \quad \xi \in D. \tag{9.14}$$

Equality (9.14) defines an interval-valued function; Section 9.4 is devoted to its presentation and computation.

This chapter considers the interval-valued functions generated by the solution sets both for the interpolation and for the curve fitting problems. In some special cases the interval solution functions have simple presentation in subregions of D and can be easily computed. The next subsection gives a brief introduction to the necessary concepts of interval arithmetic.

9.2. INTERVAL ARITHMETIC: BASIC CONCEPTS

By IR denote the set of all intervals Y of the form $Y = [y^-,y^+] = \{y \mid y^- \leq y \leq y^+\}$, $y^-,y^+ \in R$. This chapter uses two simple interval arithmetic operations:(7,8) one for addition of two intervals X, $Y \in IR$ and one for multiplication by a real number $\alpha \in R$ defined as follows:

$$X + Y = [x^- + y^-, x^+ + y^+],$$

$$\alpha X = [\alpha x^{-\sigma(\alpha)}, \alpha x^{\sigma(\alpha)}] = \begin{cases} [\alpha x^-, \alpha x^+], & \alpha \geq 0, \\ [\alpha x^+, \alpha x^-], & \alpha < 0. \end{cases}$$

wherein $\sigma(\alpha) = \{-, \alpha < 0; +, \alpha \geq 0\}$, $x^{--} = x^+$, $x^{-+} = x^-$.

The following is a simple application of interval arithmetic. Given a real valued vector $\alpha = (\alpha_1, \ldots, \alpha_n)$ and an interval valued vector $Y = (Y_1, \ldots, Y_n)^{\mathrm{T}}$ write

$$\{\alpha y \mid y \in Y\} = \{\alpha_1 y_1 + \alpha_2 y_2 + \ldots + \alpha_n y_n \mid y_1 \in Y_1, \ldots, y_n \in Y_n\}$$

$$= \alpha_1 Y_1 + \alpha_2 Y_2 + \ldots + \alpha_n Y_n = \alpha Y. \tag{9.15}$$

A standard way to present the set $\{\alpha y \mid y \in Y\}$ via the end-points of Y is

$$\{\alpha y \mid y \in Y\} = \left[\sum_{i=1}^{n} \alpha_i y_i^{-\sigma(\alpha_i)}, \sum_{i=1}^{n} \alpha_i y_i^{\sigma(\alpha_i)}\right]. \tag{9.16}$$

The interval expression (9.15) is much shorter than expression (9.16), which does not make use of interval arithmetic.

Remark: A similar expression (9.16) can be obtained by using a presentation of the intervals via centers and radii (see e.g., Ref. 6, Proposition 1). Denoting the center of the interval Y_i by y_i^c and its radius by y_i^r we obtain the expression

$$\{\alpha y \mid y \in Y\} = \left[\sum_{i=1}^{n} \alpha_i (y_i^c - \sigma(\alpha_i) y_i^r), \sum_{i=1}^{n} \alpha_i (y_i^c + \sigma(\alpha_i) y_i^r)\right],$$

which is also clumsy, whereas the interval expression αY is brief and offers convenience.

More generally, if A is a real valued $(k \times n)$-matrix

$$A = \begin{pmatrix} a_{11}, & a_{12}, & \dots, & a_{1n} \\ & & \dots & \\ a_{k1}, & a_{k2}, & \dots, & a_{kn} \end{pmatrix} = \begin{pmatrix} a_1 \\ \vdots \\ a_k \end{pmatrix}$$

then Eq. (9.15) yields for the k-dimensional set $\{Ay \mid y \in Y\}$ the following inclusion

$$\begin{aligned} \{Ay \mid y \in Y\} &= \{(a_1 y, a_2 y, \dots, a_k y) \mid y \in Y\} \\ &\subseteq (a_1 Y, \dots, a_k Y) = AY. \end{aligned} \tag{9.17}$$

Inclusion (9.17) is often known as "wrapping effect."[8] The set AY is the smallest k-dimensional box (orthotope, interval vector) enclosing the set $\{Ay \mid y \in Y\}$.

9.3. LINEAR INTERPOLATION UNDER INTERVAL MEASUREMENTS

9.3.1. The Multidimensional Case

First consider the general situation $k \geq 1$, $D \subseteq R^k$ and the problem of finding the interpolation interval function of (9.8).

DEFINITION 9.1. For a fixed class $\mathcal{L} = \mathcal{L}_m(D, \varphi)$ of modeling functions a system of vertical segments $(\mathbf{x}, \tilde{Y})$, $\mathbf{x} = (x_1, \dots, x_n)^{\top}$, $\tilde{Y} = (\tilde{Y}_1, \dots, \tilde{Y}_n)^{\top}$, is called $\mathcal{L}$-compatible (or just compatible), if for any $i \in J = \{1, \dots, n\}$ and $y_i \in \tilde{Y}_i$ there is an element η of $\mathcal{L}$, with $\eta(x_i) = y_i$, such that $\eta(\lambda; x_j) \in \tilde{Y}_j$ for $j = 1, \dots, n$, $j \neq i$.

In the situation when the data matrix $\mathbf{x}$ is fixed (as is the case in this chapter) one shall sometimes say "$\tilde{Y}$ is $\mathcal{L}$-compatible", instead of "$(\mathbf{x}, \tilde{Y})$ is $\mathcal{L}$-compatible."

Denote $\tilde{Y} = (\tilde{Y}_1, \tilde{Y}_2, \dots, \tilde{Y}_n)^{\top}$, $\tilde{Y}_i = \eta(\mathbf{x}, Y; x_i)$, $i = 1, \dots, n$. Then $\tilde{Y}_i = \eta(\mathbf{x}, \tilde{Y}; x_i)$, that is, the interval vectors $\tilde{Y}$ and Y generate same feasible solution sets. The compatible segments $(x_i, \tilde{Y}_i)$, $i \in J$, have the property of possessing no "excess points," that is, such points through which no individual solution function η passes.[13]

Two systems $(\mathbf{x}, Y)$, $(\mathbf{x}, \tilde{Y})$, generating same feasible solution sets are called equivalent. The problem of finding a solution set corresponding to the data $(\mathbf{x}, Y)$, can be divided into two steps: 1) to find an $\mathcal{L}$-compatible system $(\mathbf{x}, \tilde{Y})$ which is equivalent to $(\mathbf{x}, Y)$, and 2) to find the solution set generated by $(\mathbf{x}, \tilde{Y})$.

Every feasible parameter $\lambda \in \Lambda$ generates a vector $y_{\lambda} = (y_1, \dots, y_n)^{\top} \in R^n$ by

$$y_j = \eta(\lambda; x_j), \quad j \in J, \tag{9.18}$$

in matrix form $y_\lambda = \Phi(\mathbf{x})\lambda$. The set of all vectors y defined by Eq. (9.18) for some $\lambda \in \Lambda$ will be denoted

$$Y' = \{y_\lambda = (\eta(\lambda;x_1), \ldots, \eta\,(\lambda;x_n))^{\mathrm{T}} \mid \lambda \in \Lambda\}$$

$$= \{\Phi(\mathbf{x})\lambda \mid \lambda \in \Lambda\}. \tag{9.19}$$

In other words Y' is the subset of all y, $y \in Y$, for which the system $y = \Phi(\mathbf{x})\lambda$ is consistent. For a compatible set of data ($\mathbf{x}$, Y) the interval Y_j is the projection of the set Y' defined by Eq. (9.19) on the j-th coordinate axis.

First consider the case $n = m$ when the number of data equals the number of parameters. In this case $Y' = Y$ (since $y = \Phi(\mathbf{x})\lambda$ is consistent for all $y \in Y$) and we can express the solution set by means of the following proposition.

Proposition 9.1. For $m = n$ we have

$$\eta(\mathbf{x},Y;\xi) = (\varphi(\xi)^{\mathrm{T}}\Phi^{-1}(\mathbf{x}))Y. \tag{9.20}$$

Proof: A modeling function $\eta(\lambda;\cdot) = \varphi(\cdot)^{\mathrm{T}}\lambda$ from $\mathcal{L}_m(X)$, which interpolates a set of m data $(\mathbf{x},Y)$, satisfies a system $\Phi(\mathbf{x})\lambda \in Y$ of m algebraic inclusions for the m unknown parameters, or $\Phi(\mathbf{x})\lambda = y$, $y \in Y$. For $n = m$ we have $Y' = Y$. Since det $\Phi(\mathbf{x}) \neq 0$, every $y \in Y$ generates a $\lambda = \Phi^{-1}(\mathbf{x})y$. For the set of values of the modeling function interpolating ($\mathbf{x}$, Y) at a fixed $\xi \in D$

$$\eta(\mathbf{x},Y;\xi) = (\varphi(\xi)^{\mathrm{T}}\lambda \mid \lambda \in \Lambda\} = \{\varphi(\xi)^{\mathrm{T}}(\Phi^{-1}(\mathbf{x})y) \mid y \in Y\}$$

$$= \{(\varphi(\xi)^{\mathrm{T}}\Phi^{-1}(\mathbf{x}))y \mid y \in Y\}$$

$$= (\varphi(\xi)^{\mathrm{T}}\Phi^{-1}(\mathbf{x}))Y.$$

The interval function (9.20) will be further called simple interval interpolation function (SII-function).

Remark: Proposition 9.1 shows that the SII-function can be computed for every ξ in interval arithmetic using the simple interval-arithmetic expression (9.20). In (9.20) the vector $\varphi(\xi)^{\mathrm{T}}\Phi^{-1}(\mathbf{x})$ is multiplied by the interval vector Y in the sense of Eq. (9.15). Such an exact presentation cannot be given for the parameter set Λ because of the wrapping effect.[8] Indeed for the set Λ of feasible parameters

$$\Lambda = \{\Phi^{-1}(\mathbf{x})y \mid y \in Y'\} = \{\Phi^{-1}(\mathbf{x})y \mid y \in Y\} \subseteq \Phi^{-1}(\mathbf{x})Y.$$

Using interval arithmetic gives the inclusion $\Lambda \subseteq \Phi^{-1}(\mathbf{x})Y = \Lambda^I$, which may be rough; Λ is a convex polytope, whereas Y, and hence Λ^I, is an m-dimensional box.[8] The above consideration also demonstrates the importance of the brackets in Eq. (9.20) . A change of the place of the brackets leads to an inclusion

$$(\varphi(\xi)^{\mathrm{T}}\Phi^{-1}(\mathbf{x}))Y \subseteq \varphi(\xi)^{\mathrm{T}}(\Phi^{-1}(\mathbf{x})Y).$$

Indeed $\Lambda \subseteq \Phi^{-1}(\mathbf{x})Y = \Lambda^l$ implies

$$\eta(\mathbf{x},Y,\xi) = \{\varphi(\xi)^{\mathrm{T}}\lambda \mid \lambda \in \Lambda\} \subseteq \{\varphi(\xi)^{\mathrm{T}}\lambda \mid \lambda \in \Lambda^l\}$$

$$= \varphi(\xi)^{\mathrm{T}}(\Phi^{-1}(\mathbf{x})Y).$$

Now consider the case $m < n$. In this case $Y' \subseteq Y$ and the inclusion $\Lambda \subseteq \Lambda^l$ due to

$$\Lambda = \{\Phi^{-1}(\mathbf{x})y \mid y \in Y'\} \subseteq \{\Phi^{-1}(\mathbf{x})y \mid y \in Y\} \subseteq \Phi^{-1}(\mathbf{x})Y = \Lambda^l$$

can be rough. The following proposition gives a characterization of the solution set $\eta(\mathbf{x},Y;\cdot)$. See Lemma 9.2 from Ref. 6.

Proposition 9.2. There exists a subset Q of the index set $J = \{1, \ldots, n\}$ consisting of m elements ($Q \subseteq J$, card(Q) = m), such that for every $l \in Q$ at least one of the two equalities $\eta^-(\mathbf{x},Y;x_l) = Y_l^-$, $\eta^+(\mathbf{x},Y;x_l) = Y_l^+$ hold.

The proof of this Proposition is given in Ref. 6. Proposition 9.2 shows that the solution set reaches the end-points of at least m input intervals Y_l, $l \in Q \subseteq J$.

Let the index set Q be a subset of the index set J with m elements: $Q \subseteq J$, *card*(Q) = m. Assume that Q is ordered in increasing order and let $q(i)$ be the i-th element of Q. Denote by $\mathbf{x}^Q = (x_{q(1)}, \ldots, x_{q(m)})^{\mathrm{T}}$ the matrix $\mathbf{x}$ reduced to the index set Q. Analogously $Y^Q = (Y_{q(1)}, \ldots, Y_{q(m)})^{\mathrm{T}}$ is the vector Y reduced to Q.

To find the set of functions from $\mathcal{L}_m(X)$ interpolating a reduced set of m data $(\mathbf{x}^Q,Y^Q)$ consider the corresponding system $\Phi(\mathbf{x}^Q)\lambda \in Y^Q$, which is a system of m algebraic inclusions for m unknown parameters and applying Eq. (9.20), obtain $\eta(\mathbf{x}^Q,Y^Q;\xi) = (\varphi(\xi)^{\mathrm{T}}\Phi^{-1}(\mathbf{x}^Q))Y^Q$.

Proposition 9.3. The value of $\eta(\mathbf{x},Y;\cdot)$ at a point ξ is given by

$$\eta(\mathbf{x},Y;\xi) = \bigcap_{Q \subseteq J} \eta(\mathbf{x}^Q,Y^Q;\xi) = \bigcap_{Q \subseteq J} (\varphi(\xi)^{\mathrm{T}}\Phi^{-1}(\mathbf{x}^Q))Y^Q. \tag{9.21}$$

The proof is obvious. Proposition 3 shows that the value of $\eta(\mathbf{x},Y;\cdot)$ at ξ can be determined by an intersection of $\binom{n}{m}$ *SII*-functions.

The intervals Y_j can be reduced to $\mathcal{L}$-compatible intervals $\tilde{Y}_j$ using the following

Proposition 9.4. For the $\mathcal{L}$-compatible intervals we have

$$\tilde{Y}_j = Y_j \cap \bigcap_{Q \subseteq J \;\; j \notin Q} \eta(x^Q,Y^Q;x_j).$$

The following methods are suggested for the computation of $\eta(\mathbf{x},Y;\cdot)$ at a point $\xi \in D$:

A. Compute $\eta(\mathbf{x},Y;\cdot)$ at ξ by means of Proposition 9.3, that is, by intersecting the values of all simple interval interpolating functions at ξ. The latter are computed by means of Proposition 9.1. If at some point ξ the interval values of two simple interval interpolating functions are disjointed, their intersection is an empty set and the set of solution functions is void.

B. For $\xi \in D$ compute $\eta(\mathbf{x},Y;\xi)$ by solving two constrained linear optimization problems of Eq. (9.8).

C. Compute first the $\mathcal{L}$-compatible intervals $\tilde{Y}_i$ by means of Proposition 9.4. Then compute $\eta(\mathbf{x},Y;\cdot) = \eta(\mathbf{x},\tilde{Y};\cdot)$ at arbitrary ξ by using method **A** or **B** for the compatible intervals.

Below we look for effective methods for the presentation and computation of η in the one-dimensional case $k = 1$.

9.3.2. The One-Dimensional Case

In the remaining part of this section assume $k = 1$, that is, the input data $\mathbf{x}$ is a vector of real components and will be denoted by $x = (x_1, \ldots, x_n)$. Assume that the components of x belong to an interval $X = [x^-,x^+]$ and that $x_0 = x^- \leq x_1 < x_2 < \ldots < x_n \leq x^+ = x_{n+1}$. Use the letter k to denote a fixed subinterval $[x_k, x_{k+1}]$ to be considered. The following theorem gives an additional characterization of the boundary functions of the solution set.

We first give a definition which will be used in the proof of the next proposition.

Definition 9.2. For $l \leq m$ a l-face of Λ is a subset of Λ defined by

$$y_j^- \leq \varphi(\xi_j)^{\mathrm{T}}\lambda \leq y_j^+, \quad j \in J,$$

where $m - l$ of the above linear independent inequalities transform into equalities.[11,14]

Proposition 9.5. Let the set $\eta(\lambda;\cdot)$ of all functions from $\mathcal{L}_m(X)$ which interpolate (x,Y) be not empty and let the interval function $\eta(x,Y;\cdot)$ be the envelope of this set.[15,16,17] Then in every (x_k, x_{k+1}), $k = 0,1, \ldots, n,$ the upper and lower boundary functions of $\eta(x,Y;\cdot)$ are functions from $\mathcal{L}_m(X)$ generated by some parameters $\lambda_k^-,\lambda_k^+ \in \Lambda$.

Proof: Proposition 9.5 states that for every subinterval $[x_k, x_{k+1}]$ there exist two parameters λ_k^-, $\lambda_k^+ \in \Lambda \subseteq R^m$ generating the envelope in the whole subinterval, that is,

$$\eta^-(x,Y;\xi) = \eta(\lambda_k^-;\xi) = \varphi(\xi)^{\mathrm{T}}\lambda_k^-, \quad \xi \in (x_k, x_{k+1}),$$

$$\eta^+(x,Y;\xi) = \eta(\lambda_k^+;\xi) = \varphi(\xi)^{\mathrm{T}}\lambda_k^+, \quad \xi \in (x_k, x_{k+1}).$$

Assuming the opposite, there exist a point $\xi_s \in (x_k, x_{k+1})$ and two parameters $\lambda^1,\lambda^2 \in R^m$, $\lambda^1 \neq \lambda^2$, such that $\eta^+(x,Y;\xi_s) = \varphi(\xi_s)^{\mathrm{T}}\lambda^1 = \varphi(\xi_s)^{\mathrm{T}}\lambda^2$. On the other hand

$$\eta^+(x,Y;\xi_s) = \max_{\lambda\in\Lambda} \varphi(\xi_s)^\top\lambda. \tag{9.22}$$

Because the set of optimal points of the linear programming problem (9.22) is convex all points of the segment $[\lambda^1,\lambda^2]$ are optimal. The set of all optimal points of (9.22) is a l-face of Λ, where $l \geq 1$and the vector $\varphi(\xi_s)$ is perpendicular to this l-face of Λ. The l-face is an intersection of $(m - l)$ hyperplanes with linear independent normal vectors $a^1, \dots, a^{m-l} \in \{\varphi(\xi_j), j \in J\}$ (the linear independence follows from the assumption that the modeling function is from $\mathcal{L}_m(X)$). Thus the vector $\varphi(\xi_s)$ is a linear combination of $a^1, \dots, a^{m-l}$. This is a contradiction to the assertion that φ is a system of basic functions. For the lower function η^- the arguments are analogous. □

Proposition 9.5 shows that under the given assumptions the upper and lower boundary functions $\eta(x,Y;\xi)$ for all $\xi \in (x_k, x_{k+1})$ are themselves elements of $\mathcal{L}_m(X)$. Therefore, to find $\eta(x,Y;\xi)$ for $\xi \in (x_k, x_{k+1})$ we have to determine expressions for these two functions. Such expressions can be found ether in terms of some subset (x^Q,Y^Q) of the given data or in terms of λ depending on the method used: intersection of *SII*-functions (method **A**) or constrained optimization (method **B**).

In some cases it can be preferable to use method **C** which prescribes first the computation of the compatible intervals. The next proposition shows that, if the set of data (x,Y) is $\mathcal{L}$-compatible, then $\eta(x,Y;\xi)$ may be determined by an intersection of a reduced number of simple interval interpolating functions.

PROPOSITION 9.6. If the set of data (x,Y) is $\mathcal{L}$-compatible, then for every $k = 0, \dots, n$ the following formula holds

$$\eta(\mathbf{x},Y;\xi) = \bigcap_{Q\in Q(k)} \eta(x^Q,Y^Q;\xi) \text{ for } \xi \in [x_k, x_{k+1}],$$

where $Q(k)$ is the set of all subsets Q of J consisting of m elements (notationally, $Q \subseteq J$, $card(Q) = m$), such that

$$k,k+1 \in Q, \quad \text{if } 0 < k < n,$$

$$1,n \in Q, \quad \text{if } k = 0 \text{ or } k = n.$$

If $m = 2$ the set $Q(k)$, for every k, $0 \leq k \leq n$, consist of one single pair, namely

$$Q(k) = \begin{cases} \{k,k+1\}, & \text{if } 0 < k < n, \\ \{1,n\}, & \text{if } k = 0 \text{ or } k = n. \end{cases}$$

For the interval solution $\eta(\lambda;\cdot)$ in this case ($m = 2$, compatible data) the following simple formula holds in $[x_k, x_{k+1}]$:

$$\eta(x,Y;\xi) = \eta(x^{Q(k)},Y^{Q(k)};\xi) = (\varphi(\xi)^\top\Phi^{-1}(\mathbf{x}^{Q(k)}))Y^{Q(k)}$$

$$= \begin{cases} \dfrac{\Delta(\xi,x_{k+1})}{\Delta(x_k,x_{k+1})} Y_k + \dfrac{\Delta(x_k,\xi)}{\Delta(x_k,x_{k+1})} Y_{k+1}, & \text{if } 0 < k < n, \\ \dfrac{\Delta(\xi,x_n)}{\Delta(x_1,x_n)} Y_1 + \dfrac{\Delta(x_1,\xi)}{\Delta(x_1,x_n)} Y_n, & \text{if } k = 0 \text{ or } k = n, \end{cases}$$

wherein

$$\Delta(x',x'') = \begin{vmatrix} \varphi_1(x') & \varphi_2(x') \\ \varphi_1(x'') & \varphi_2(x'') \end{vmatrix} .$$

Clearly, finding the compatible intervals in the case $m = 2$ solves the problem of finding the *SII*-function. It is then obtained by connecting the upper, resp., lower end-points of each two neighboring segments (x_i,Y_i), (x_{i+1},Y_{i+1}) via generalized linear functions. Proposition 9.6 is proved (for the polynomial case) in Ref. 16.

Numerical algorithm (for $k = 1$): Compute $\eta(x,Y;\cdot)$ at some point ξ_i from the open interval (x_i,x_{i+1}), e.g. $\xi_i = (x_{i+1} + x_i)/2$, using method **A** or **B**. Proposition 5 states that there are two unique generalized polynomials $\eta_i^- = \eta(\lambda_i^-;\xi)$, $\eta_i^+ = \eta(\lambda_i^+;\xi)$ which are the boundary functions of $\eta(x,Y;\cdot)$ in the interval $[x_i, x_{i+1}]$. We can find expressions for the boundary functions η_i^-,η_i^+ by any one of the methods **A** or **B**. Using method **A** we obtain two m-dimensional subsets Q_i^-,Q_i^+ of J and two m-dimensional sets of binary variables $\mathcal{A}^- = (\alpha_{q(1)}^-, \ldots, \alpha_{q(m)}^-)$, $\mathcal{A}^+ = (\alpha_{q(1)}^+, \ldots, \alpha_{q(m)}^+)$, $\alpha_{q(i)}^-$, $\alpha_{q(i)}^+ \in \{+,-\}$, $i = 1, \ldots, n$, such that for $\xi \in [x_i, x_{i+1}]$:

$$\eta^-(x,Y;\xi) = [\varphi(\xi)^{\mathsf{T}}\Phi^{-1}(\mathbf{x}^{Q_i^-}]\,(Y^{Q_i^-})^{\alpha_{q(i)}^-},$$

$$\eta^+(x,Y;\xi) = [\varphi(\xi)^{\mathsf{T}}\Phi^{-1}(\mathbf{x}^{Q_i^+})]\,(Y^{Q_i^+})^{\alpha_{q(i)}^+}$$

(note that the pairs $(Q_i^-,\mathcal{A}_i^-)$, $(Q_i^+,\mathcal{A}_i^+)$ may not be unique, and any pair can be used).

Alternatively, if method **B** is used then we can determine λ_i^-,λ_i^+ as defined by Proposition 9.5.

According to Proposition 9.5 the expressions for the functions η_i^-, η_i^+ can be used for presentation or computation of $\eta(x,Y;\cdot)$ at any point in the subinterval $[x_i, x_{i+1}]$.

9.3.3. The Polynomial Case

If the basic functions are of the form $\varphi_i(x) = x^{i-1}$, $i = 1, \ldots, m$, then (9.2) is the Vandermond's determinant: det $\Phi(x') = \Pi_{i>j}\,(x_i - x_j)$, which does not vanish. $\mathcal{L}_m(X)$ is the class of polynomial functions defined on $X = R$ of $(m - 1)$-st degree of the form $\eta_{m-1}(\lambda;\xi) = \lambda_1 + \lambda_2\xi + \ldots + \lambda_m\xi^{m-1}$.

In the case $n = m$ Eq. (9.20) for the *SII*-function obtains the form[18,19]

$$\eta_{m-1}(x,Y;\xi) = l(x;\xi)^{\mathrm{T}}Y, \quad l_i(x;\xi) = \prod_{k=1,\dots,m,\ k\neq i} \frac{\xi - x_k}{x_i - x_k}.$$

This interval function has been studied (without using interval arithmetic).(20)

Equation (9.21) for $n > m$ in the polynomial case reads:(13)

$$\eta_{m-1}(x,Y;\xi) = \bigcap_{Q\subseteq J} l(x^Q;\xi)^{\mathrm{T}}Y^Q, \quad l_i(x^Q;\xi) = \prod_{k=1,\dots,m,\ k\neq i} \frac{\xi - x_{q(k)}}{x_{q(i)} - x_{q(k)}},$$

wherein $Q = \{q(k)\}_{k=1}^{m}$.

The intervals Y_j can be reduced to compatible intervals $\tilde{Y}_j$ by(13)

$$\tilde{Y}_j = Y_j \cap \bigcap_{Q\subseteq J\ j\notin Q} l(x^Q;x_j)^{\mathrm{T}}Y^Q, \quad j\in J.$$

For $m = 2$ and applying Proposition 9.6 gives for $\xi \in [x_k, x_{k+1}]$, $0 \le k \le n$, the following simple expression(13)

$$\eta_1(x,Y;\xi) = \begin{cases} \dfrac{\xi - x_{k+1}}{x_k - x_{k+1}} Y_k + \dfrac{\xi - x_k}{x_{k+1} - x_k} Y_{k+1}, & \text{if } 0 < k < n, \\ \dfrac{\xi - x_n}{x_1 - x_n} Y_1 + \dfrac{\xi - x_1}{x_n - x_1} Y_n, & \text{if } k = 0 \text{ or } k = n, \end{cases}$$

where the data (x, Y) are assumed compatible.

Next are two examples for polynomial functions. The computations are performed by a program written in PASCAL-SC,(12) based on the method **A**.

Example 1: Let the following set of data be given

$$(x,Y) = \begin{pmatrix} 1, & 2, & 4, & 6 \\ [1,3], & [1,2], & [1.5,2.5], & [2,3] \end{pmatrix}^{\mathrm{T}},$$

and let the modeling functions be second order polynomials of the form

$$\eta_2(\lambda;\xi) = \lambda_1 + \lambda_2\xi + \lambda_3\xi^2.$$

The graph of the interval function $\eta_2(x,Y;\cdot)$ is presented on Fig. 9.1. For comparison the simple interval polynomial $\eta_3(x,Y;\cdot)$ is also presented. In order to recognize both interval interpolating functions on Fig. 9.1 keep in mind that $\eta_2 \subseteq \eta_3$.

According to Proposition 9.3 the bounding functions of $\eta_2(x,Y;\cdot)$ pass through at least three end-points of the interval segments, which fully determine them. The program gives results for $\eta_2(x,Y;\cdot)$ presented in Table 9.1. Note that the computed

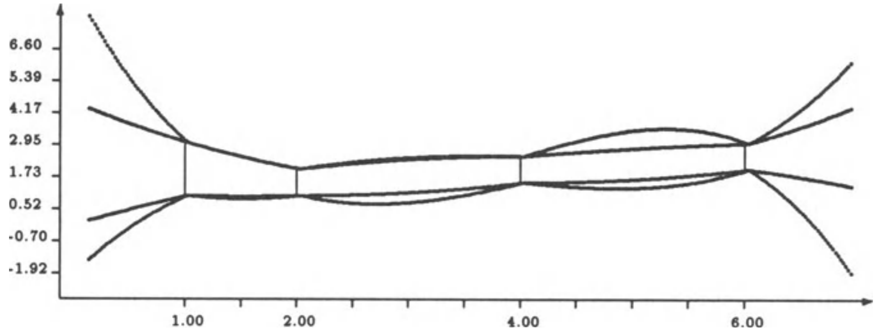


FIGURE 9.1. Graphs of the interval polynomials from Example 1.

compatible intervals coincide with the input intervals, that is, the input data are compatible.

Remark: To demonstrate the advantages of direct computation of the interval function $\eta_2(x,Y;\cdot)$ compute the solution set for this example through the parameter set Λ. Assume that Λ is computed exactly. Then optimally enclose Λ to obtain an interval vector Λ^I. The best result for the upper function is

$$\eta_2^+(\Lambda^I;\xi) = 4.5 + 1.25\xi + 0.25\xi^2,$$

and for the lower function

$$\eta_2^-(\Lambda^I;\xi) = -0.1 - 1.75\xi - 0.15\xi^2.$$

The width of $\eta_2(\Lambda^I;\xi)$ at $\xi = 6$ is

$$\omega[\eta_2(\Lambda^I;6)] = \eta_2^+(\Lambda^I;6) - \eta_2^-(\Lambda^I;6) = 37.$$

TABLE 9.1. Bounding Functions and Compatible Intervals for the Problem of Example 1

	Bounding Functions		
Subinterval	Lower	Upper	Compatible Intervals
$[x_{-\infty}, x_1]$	$Y_1^- Y_3^+ Y_4^-$	$Y_2^+ Y_3^- Y_4^+$	$Y_1 = [1,3]$
$[x_1, x_2]$	$Y_1^- Y_2^- Y_4^+$	$Y_2^+ Y_3^- Y_4^+$	$Y_2 = [1,2]$
$[x_2, x_3]$	$Y_2^- Y_3^- Y_4^+$	$Y_2^+ Y_3^+ Y_4^-$	$Y_3 = [1.5,2.5]$
$[x_3, x_4]$	$Y_2^+ Y_3^- Y_4^-$	$Y_1^- Y_3^+ Y_4^+$	$Y_4 = [2,3]$
$[x_4, x_\infty]$	$Y_1^- Y_3^+ Y_4^-$	$Y_2^+ Y_3^- Y_4^+$	

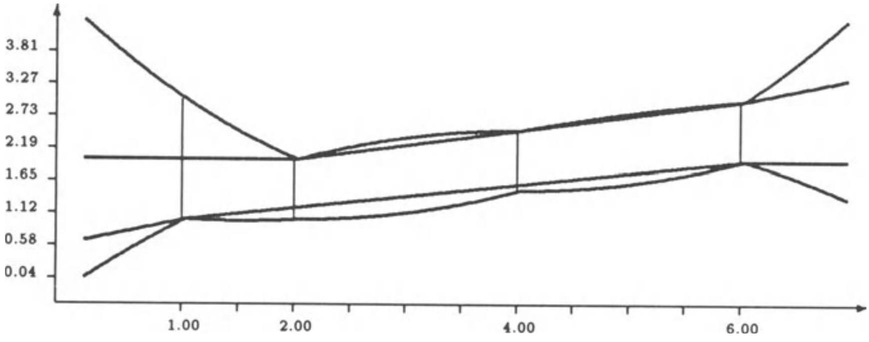


FIGURE 9.2. Graphs of the interval polynomials related to Example 2.

The width of $\eta_2(\Lambda;6) = \eta_2(x,Y;6)$ as computed by our method is

$$\eta_2^+(x,Y;6) - \eta_2^-(x,Y;6) = 1.$$

Example 2: For the same set of data and for the set of linear modeling functions $\eta_1(\lambda;\xi) = \lambda_1 + \lambda_2\xi$ we obtain the results presented in Table 9.2.

The interval function $\eta_1(x,Y;\cdot)$, comprising the set of linear modeling functions is presented on Fig. 9.2. For comparison the function $\eta_2(x,Y;\cdot)$ is given (the latter also appears in Fig. 9.1). To recognize both functions on Fig. 9.2 recall that $\eta_1 \subseteq \eta_2$.

Example 3: Next consider an example using 6 knots

$$(x,Y) = \begin{pmatrix} 0, & [1, 1.02] \\ 1, & [0.99, 1.25] \\ 2, & [1.04, 1.06] \\ 3, & [1.07, 1.09] \\ 4, & [1.16, 1.18] \\ 5, & [1.23, 1.25] \end{pmatrix}$$

TABLE 9.2. Bounding Functions and Compatible Intervals for the Problem of Example 2

	Bounding Functions		
Subinterval	Lower	Upper	Compatible Intervals
$[x_{-\infty}, x_1]$	$Y_1^- Y_4^+$	$Y_2^+ Y_4^-$	$Y_1 = [1,2]$
$[x_1, x_2]$	$Y_1^- Y_4^-$	$Y_2^+ Y_4^-$	$Y_2 = [1.2,2]$
$[x_2, x_3]$	$Y_1^- Y_4^-$	$Y_2^+ Y_4^+$	$Y_3 = [1.6,2.5]$
$[x_3, x_4]$	$Y_1^- Y_4^-$	$Y_3^+ Y_4^+$	$Y_4 = [2,3]$
$[x_4, x_\infty]$	$Y_2^+ Y_4^-$	$Y_1^- Y_4^+$	

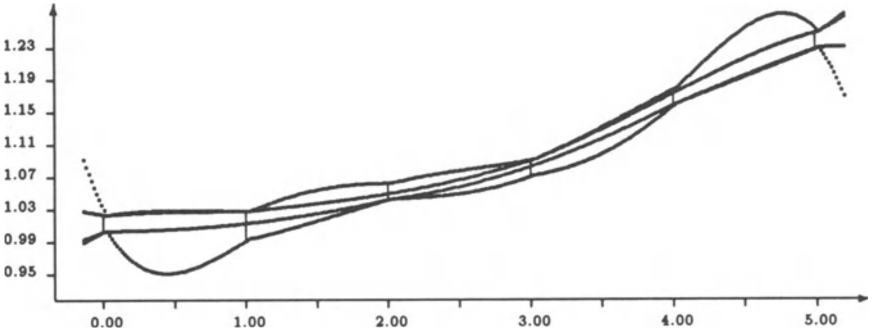


FIGURE 9.3. Graphs of the interval polynomials related to Example 3.

Fig. 9.3 presents the corresponding polynomials $\eta_5(x,Y;\cdot)$ and $\eta_4(x,Y;\cdot)$. Of course, $\eta_4 \subseteq \eta_5$.

9.4. LINEAR ESTIMATION UNDER INTERVAL MEASUREMENTS

This section shall assume that the parameter λ_y defined by Eq. (9.10) depends linearly on y, i.e., $\lambda_y = Hy$, where $H \in R^{m\times n}$, and $H = H(\mathbf{x})$ may depend on $\mathbf{x}$ but not on y. Equation (9.12) can be written as

$$\Lambda_\phi = \{\lambda_y \in K \subseteq R^m, \lambda_y = Hy \mid y \in Y\}$$

$$= \{Hy \mid y \in Y\} \subseteq HY, \tag{9.23}$$

whereby the last inclusion relation Eq. (9.17) has been used.

Assume as before that $\mathcal{L}$ is a class of linear on λ functions of the form $\eta(\lambda;\cdot) = \varphi(\cdot)^{\mathrm{T}}\lambda$ defined on D. For a fixed $\xi \in D$ the estimate solution set can be written in the form

$$\eta(\Lambda_\phi;\xi) = \{\eta(\lambda;\xi) \mid \lambda \in \Lambda_\phi\}$$

$$= \{\varphi(\xi)^{\mathrm{T}}\lambda, \lambda = Hy \mid y \in Y\}$$

$$= \{\varphi(\xi)^{\mathrm{T}}(Hy) \mid y \in Y\} = \{(\varphi(\xi)^{\mathrm{T}}H)y \mid y \in Y\}$$

$$= (\varphi(\xi)^{\mathrm{T}}H)Y = \Gamma(\xi)Y. \tag{9.24}$$

Note that the interval-valued function (9.24) gives the exact bounds for the solution set. Next consider a special case of least-square estimator illustrating the above approach.

Multiple linear regression: In the case of multiple linear regression, denote $\xi = (1, \xi_1, \ldots, \xi_{m-1})$ and assume $\varphi_i(\xi) = \xi_i$, $i = 0, \ldots, m-1$, so that

$$\eta(\lambda;\xi) = \varphi(\xi)^{\mathrm{T}}\lambda = \lambda_0 + \lambda_1\xi_1 + \ldots + \lambda_{m-1}\xi_{m-1} = \xi\lambda.$$

Denoting

$$X = \begin{pmatrix} 1 & x_{11} & \ldots & x_{1m-1} \\ & & \ldots & \\ 1 & x_{n1} & \ldots & x_{nm-1} \end{pmatrix},$$

we obtain from (9.10) with an l_2 norm the matrix H in the form $H = (X^{\mathrm{T}}X)^{-1}X^{\mathrm{T}}$. Substituting in (9.23) and (9.24) gives

$$\Lambda_\varphi \subseteq HY = ((X^{\mathrm{T}}X)^{-1}X^{\mathrm{T}})Y, \tag{9.25}$$

$$\eta(\Lambda_\varphi;\xi) = \Gamma(\xi)Y = (\xi H)Y = (\xi(X^{\mathrm{T}}X)^{-1}X^{\mathrm{T}})Y, \tag{9.26}$$

where $\Gamma(\xi) = \xi(X^{\mathrm{T}}X)^{-1}X^{\mathrm{T}} = (\gamma_1(\xi), \ldots, \gamma_n(\xi))$.

In the case of $m = 2$, the approximating function is linear of the form $f(\lambda;\xi) = \lambda_0 + \lambda_1\xi$. For the components $\gamma_i(\xi)$ of the n-dimensional vector $\Gamma(\xi)$ we obtain

$$\begin{aligned}\gamma_i(\xi) &= (\xi(X^{\mathrm{T}}X)^{-1}X^{\mathrm{T}})_i \\ &= (x_i - \bar{x})(\xi - \bar{x})/S_{xx} + 1/n, \quad i = 1, \ldots, n,\end{aligned}$$

where

$$\bar{x} = \sum^n x_i/n, \quad S_{xx} = \sum^n x_i^2 - n\bar{x} = \sum^n (x_i - \bar{x})^2.$$

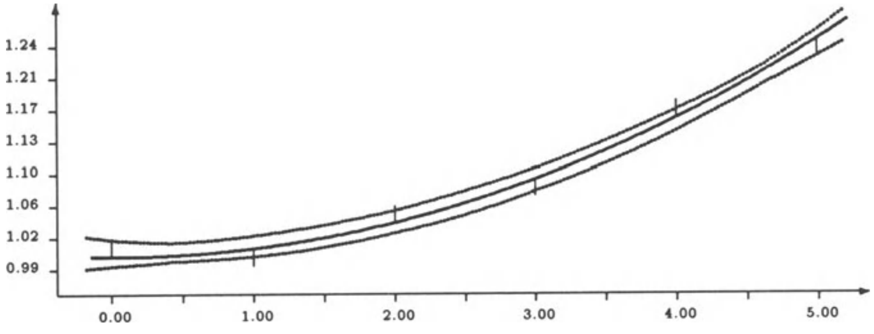


FIGURE 9.4. Graphs of the interval polynomials related to Example 4.

The boundary functions of the interval function $L(\xi) = \Gamma(\xi)Y$ are lines in each interval with end-points two neighboring ξ_i, i = 1, . . . , n, where ξ_i are defined by $\gamma_i(\xi_i) = 0$, that is,

$$\xi_i = \bar{x} + S_{xx}/[n(\bar{x} - x_i)].$$

The polynomial and multinomial cases produce similar results under the corresponding choice of the matrix X.

Example 4: Consider the data

$$(x,Y) = \begin{pmatrix} 0, & [1, 1.02] \\ 1, & [0.99, 1.01] \\ 2, & [1.04, 1.06] \\ 3, & [1.07, 1.09] \\ 4, & [1.16, 1.18] \\ 5, & [1.23, 1.25] \end{pmatrix}.$$

For the given data, the set of interpolating polynomials of degree $m - 1 = 2$ consists of only one single-valued interpolation polynomial. The latter serves also for an unique solution of the same problem with $m - 1 = 3$ and $m - 1 = 4$ (Fig. 9.4). The solution set is empty for the same problem with $m - 1 < 2$. The envelope of the set of least-square approximation polynomials of second degree for the given interval data is also presented in Figure 9.4.

Example 5: For the set of data of Example 1 and for modeling functions which are second order polynomials, the corresponding sets of solutions both for the interpolation and the least-square approximation problems are presented on Fig. 9.5.

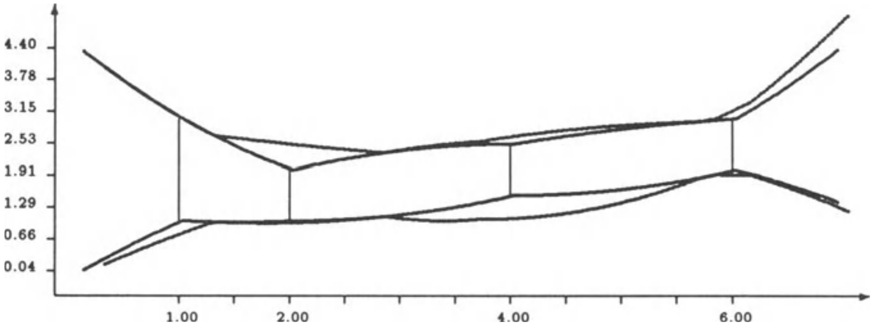


FIGURE 9.5. Graphs of the interval polynomials related to Example 5.

9.5. CONCLUSION

Both interpolation and curve fitting problems involving generalized polynomials and interval data have been studied. In certain special cases exact interval-arithmetic expressions for the envelopes of the sets of solution functions are obtained (see Eqs. (9.20, 9.21, 9.24, and 9.26)). In the one-dimensional case when the solutions are functions of one variable, the enveloping functions are characterized to show that they are piece-wise generalized polynomials. These interval-arithmetic expressions can be effectively computed in a software environment which supports interval arithmetic like recently developed SC-language(12) (or computer algebra systems *Maple* and *Mathematica*). Such an environment provides computer operations with directed roundings, so that the computed interval bounds are automatically rounded toward outside and contain with guarantee the true results. Thus, the computed bounds comprise all possible kinds of input and computational errors. This fact opens a new way to the practical implementation and interpretation of the computed results especially with respect to the interpolation problem. For example, assume that one knows that the experimentally obtained measurement intervals Y contain with guarantee the true values of the measured quantities. Assume that $\eta(x,Y;\cdot)$ is the interval solution function computed from these measurements and that the model function η belongs to $\mathcal{L}$. Assume that an experiment provides us with a *new* measurement (x_N,Y_N) such that $\eta(x,Y;x_N) \cap Y_N = \emptyset$. The correct conclusion, then, is that the class $\mathcal{L}$ of model functions is inadequate for the description of the experimental data.

Therefore the approach and programming tools can be used by experimental scientists for checking hypotheses with respect to the type of the modeling functions. New data can be easily checked whether they intersect the available interval solution sets. If some of these intersections are empty, then it follows that the type of the modeling functions is wrong. Another type of modeling function (possibly involving more parameters or other type of basic functions) should be taken in consideration.

In the above arguments it is assumed that Y_i are measurement intervals, containing *with guarantee* the true values of the measured quantities. It seems that experimental scientists can provide such intervals in most situations. Moreover, the provision of guaranteed bounds seems to be a substantial part of the experiment. At present, experimental scientists often do not care about obtaining such bounds, which diminishes the value of the experiment. A possible explanation for such an attitude is that few mathematical tools and methods dealing with interval problems have been developed. Measurement tools and instruments also sometimes fail to provide the necessary guaranteed bounds for the data to be read. The guaranteed numerical "interval approaches" should be employed for guaranteed interval data, possibly obtained using high quality "interval measurement" tools.

ACKNOWLEDGMENTS. This work has been partially supported by the National Foundation for Scientific Research under contract No. MM521/95.

REFERENCES

1. M. Milanese and A. Vicino, in: *Bounding Approaches to System Identification* (M. Milanese *et al.*, eds.), Plenum Press, New York, Chap. 2 (1996).
2. J. P. Norton, *Automatica* **23**, 497 (1987).
3. E. Walter, editor, *Math. Comput. Simul.* **32**, 447 (1990).
4. A. P. Voschinin and G. R. Sotirov, *Optimization with Uncertainties*, Moscow Energy Institute, Moscow, in Russian (1989).
5. M. Milanese, in: *Robustness in Identification and Control* (M. Milanese, R. Tempo, and A. Vicino, eds.), pp. 3–24, Plenum Press, New York (1989).
6. M. Milanese and G. Belforte, *IEEE Trans. Autom. Control* **AC-27**, 408 (1982).
7. G. Alefeld and J. Herzberger, *Introduction to Interval Computations*, Academic Press, New York (1983).
8. R. E. Moore, *Interval Analysis*, Prentice-Hall, Englewood Cliffs, N.J. (1966).
9. L. Jaulin and E. Walter, in: *Bounding Approaches to System Identification* (M. Milanese *et al.*, eds.), Plenum Press, New York, Chap. 23 (1996).
10. G. Sotirov, in: *Scientific Computation and Mathematical Modeling* (S. M. Markov, ed.), DATECS Publishing, Sofia, pp. 31–36 (1993).
11. S. H. Mo and J. P. Norton, in: *Proceedings of the 12th IMACS World Congress*, Paris (1988).
12. U. Kulisch, editor, *PASCAL-SC: A PASCAL-Extension for Scientific Computation: Information Manual and Disks for IBM PC*, Wiley-Teubner, Chichester (1987).
13. S. M. Markov, in: *Computer Arithmetic, Scientific Computation and Mathematical Modeling* (E. Kaucher, S. M. Markov, and G. Mayer, eds.), IMACS, pp. 251–262 (1991).
14. S. N. Tschernikow, *Lineare Ungleichungen*, Deutcher Verlag der Wissenschaften Berlin, Germany (1971).
15. S. Markov, E. Popova, U. Schneider, and J. Schulze, submitted to *Math. Comput. Simul.* (1993).
16. U. Schneider, *Modellierung unscharfer Daten: Intervallinterpolationsproblem*, Diplomawork, Fachbereich Mathematik und Informatik, Merseburg, Germany (1992).
17. U. Schneider and J. Schulze, in: *Scientific Computation and Mathematical Modeling* (S. M. Markov, ed.), DATECS Publishing, Sofia, pp. 173–176 (1993).
18. J. Herzberger, *SIAM J. Appl. Math.* **34**, 4 (1978).
19. J. Garloff, *Z. Angew. Math. Mech.* **59**,T59 (1979).
20. M. A. Crane, *SIAM J. Appl. Math.* **29**, 751 (1975).
21. S. M. Markov, in: *Contributions to Computer Arithmetic and Self-Validating Numerical Methods* (Ch. Ullrich, ed.) J. C. Baltzer AG, Basel, Switzerland, pp. 133–147 (1990).

10

Adaptive Approximation of Uncertainty Sets for Linear Regression Models

A. Vicino and G. Zappa

ABSTRACT

This chapter deals with the problem of uncertainty evaluation in linear regression models, representing either purely parametric models or mixed parametric/nonparametric (restricted complexity) models. The hypothesis is that disturbance information and prior knowledge on the unmodeled dynamics are available as deterministic bounds. A procedure is proposed for constructing recursively an outer bounding parallelotopic estimate of the parameter uncertainty set, which can be considered as an alternative description to commonly used ellipsoidal approximations. This new type of approximation is motivated by recent developments in the robust control field, where descriptions like hyperrectangular or polytopic domains have led to appealing stability and performance robustness properties of uncertain feedback systems.

A. Vicino • Facoltà di Ingegneria, Università degli Studi di Siena, 53100 Siena, Italy. G. Zappa • Dipartimento di Sistemi e Informatica, Università di Firenze, 50139 Firenze, Italy.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

10.1. INTRODUCTION

Recent years have seen a renewed and stronger interest in system identification.(1,2) Research activity has been mainly stimulated by the growing need for techniques providing the basic information required by advanced robust and adaptive control schemes developed in the past decade.(2) Both soft (stochastic) and hard (deterministic) bound settings have been widely investigated(2,3) for soft-bound, mixed parametric/nonparametric approaches, for H_∞ or l_1 nonparametric techniques,(4,5) for hard-bound purely parametric approaches,(6,7,8) and for hard-bound mixed parametric/nonparametric approaches.(9,10,11) Mixed parametric/nonparametric approaches appear promising for providing the necessary information for applicability of the techniques recently devised in the robust control field for structured and unstructured uncertainties.(12–15)

This chapter is embedded in a hard-bound setting, where knowledge about disturbances and *a priori* information is given in terms of deterministic bounds. A fixed-order model and a possible block accounting for unmodeled dynamics are allowed. The contribution of this chapter is in the spirit of Wahlberg and Ljung.(11) The distinguishing feature is that instead of constructing adaptive ellipsoidal approximations for the parameter uncertainty set, i.e., the set of parameters compatible with the disturbance bounds and the *a priori* knowledge on the unmodeled dynamics, it proposes recursive approximations of orthotopic or parallelotopic shape.

Beyond the intrinsic interest from a theoretical standpoint, the main practical motivation for this different characterization of the parameter uncertainty set estimates lies in the recent results found in the robust control field when the nominal plant model is affected by parametric or mixed parametric/nonparametric perturbations. Most of these contributions refer to uncertainty regions in plant parameter space of hyperrectangular or polytopic shape.(14–16) The main purpose of these references is to characterize extremal subsets of the uncertainty region providing worst-case properties of the uncertain system from the stability or performance viewpoint. The interesting feature of polytopic regions is that it is possible to find very ‘small’ subsets (made of vertices or edges) providing the ‘worst-case’ information contained in the whole uncertainty set.

This chapter provides an adaptive algorithm for constructing recursively an outer bounding parallelotopic approximation of the parameter uncertainty set. The procedure represents a counterpart of the algorithm originally proposed by Fogel and Huang(17) and successively modified by Belforte *et al.*(18) It can be employed both in a purely parametric or in a mixed parametric/nonparametric setting of the identification problem. Though the computational burden of the algorithm is comparable to that in,(17,18) it is a good candidate to provide better approximations of the parameter uncertainty set, on the grounds that the family of approximating parallelotopes is parameterized according to a larger number of degrees of freedom than ellipsoids.

The chapter is structured as follows. Section 10.2 introduces notation and problem formulation. Section 10.3 presents basic results for optimal approximation of the uncertainty set, while the adaptive algorithm is discussed in Section 10.4. Concluding remarks are reported in Section 10.5.

10.2. NOTATION AND PROBLEM FORMULATION

Consider the linear regression equation

$$y(k) = \phi'(k)\theta + e(k), \;\; k = 1,2, \ldots \tag{10.1}$$

where $y(k)$ is the k-th scalar measurement on the system under investigation, $\theta = [\theta_1, \ldots, \theta_n]'$ is the model parameter vector, $\phi(k) = [\phi_1(k), \ldots, \phi_n(k)]'$ is the regressor and $e(k)$ represents an error term such that

$$|e(k)| \leq r(k), \;\; k = 1, 2, \ldots \tag{10.2}$$

where $r(k) > 0$ is a known sequence of error bounds. Notice that, as is better specified at the end of this section, θ may include parameters of a fixed-order nominal model and parameters describing the unmodelled dynamics possibly associated with the nominal model.(11) Denote by $\Theta(k)$ the uncertainty parameter set at time k, i.e., the set of θ consistent with the model Eq. (10.1) and the error bound Eq. (10.2) up to the k-th measurement, i.e.,

$$\Theta(k) = \{\bigcap_{l=1}^{k} \Sigma(l)\}, \tag{10.3}$$

where $\Sigma(l)$ is the set of parameters consistent with the l-th measurement

$$\Sigma(l) = \{\theta \in R^n \colon |y(l) - \phi'(l)\theta| \leq r(l)\}.$$

A set in R^n defined as $\Sigma(l)$ will be called a ‘strip’. It is easy to check that $\Theta(k)$ is a convex polytope. Assume that $\Theta(k)$ is nonempty for any k.

The next sections approximate $\Theta(k)$ through simple-shaped regions like parallelotopes; a description of such regions is introduced. Denote by $\mathcal{R}(\theta^c)$ the unit ball in the l_∞ norm centered at θ_c

$$\mathcal{R}(\theta^c) = \{\theta : \max_{i=1,\ldots,n} |\theta_i - \theta_i^c| \leq 1\}.$$

A parallelotope can be defined through $\mathcal{R}(\theta^c)$ and a nonsingular transformation $T \in R^{n,n}$

$$\mathcal{P}(T,\theta^c) = \{\theta : \theta = T\tilde{\theta}, \tilde{\theta} \in \mathcal{B}(\theta^c)\} = \{\theta : \|P(\theta - \theta^c)\|_\infty \leq 1\} \tag{10.4}$$

where $P = T^{-1}$. Denote by $t_j, j = 1, \ldots, n$ and $p_i', i = 1, \ldots, n$ the columns and rows of matrices T and P, respectively. It is easy to verify that

$$\theta \in \mathcal{P} \Leftrightarrow \theta = \theta^c + \sum_{i=1}^{n} \alpha_i t_i, \quad \alpha_i \in [-1,1]. \tag{10.5}$$

Alternatively, the parallelotope $\mathcal{P}(T,\theta^c)$ can be expressed as the intersection of n strips in parameter space,

$$\mathcal{P}(T,\theta^c) = \left\{ \bigcap_{i=1}^{n} \mathcal{S}_i \right\}, \tag{10.6}$$

where

$$\mathcal{S}_i = \{\theta : |p_i'\theta - c_i| \leq 1\}, \quad c_i \doteq p_i'\theta^c. \tag{10.7}$$

Moreover, denote by σ_i^+ and σ_i^- the bounding hyperplanes of $\mathcal{S}_i$, i.e.,

$$\sigma_i^+ \doteq \{\theta : p_i'\theta - c_i = 1\}, \quad \sigma_i^- \doteq \{\theta : p_i'\theta - c_i = -1\}. \tag{10.8}$$

Since one looks for 'optimal', in the sense of minimal volume, outer approximations of $\Theta(k)$, choose as 'measure' μ of a parallelotope in R^n its volume

$$\mu[\mathcal{P}(T,\theta^c)] = \mathrm{vol}[\mathcal{P}(T,\theta^c)].$$

Recall the relationship between the volumes of a unit ball $\mathcal{B}(\theta^c)$ and $\mathcal{P}(T,\theta^c)$

$$\mu[\mathcal{P}(T,\theta^c)) = 2^n|\det(T)| = 2^n/|\det(P)|. \tag{10.9}$$

Hence, the requirement of minimal volume for a parallelotopic domain is equivalent to one of minimum (maximum) determinant magnitude for the matrix $T(P)$ defining the parallelotope.

Now formulate the problem solved in the forthcoming section. Consider the linear regression model Eq. (10.1) with error bounds given by Eq. (10.2). Let an outer estimate of $\Theta(k)$ be given at time k in the form of a parallelotope $\mathcal{P}(T,\theta^c)$. Suppose that an additional measurement at time $k + 1$ becomes available. The problem is to use the new information to update in an optimal way the parallelotopic estimate. More precisely, denoting by $\mathcal{P}_k$ the parallelotopic estimate of $\Theta(k)$ at time k, i.e., $\mathcal{P}_k = \mathcal{P}(T(k),\theta^c(k))$, find the minimal-volume parallelotope $\mathcal{P}_{k+1}$ consistent with the preceding estimate $\mathcal{P}_k$, the new measurement $y(k + 1)$ and the corresponding error bound $r(k + 1)$. Of course, *a priori* information on the system and on the data is assumed to be available in order to determine a suitable initial estimate $\mathcal{P}_0$ and evaluate the error bounds $r(k)$.

The above problem formulation includes both purely parametric model estimation, classical in the set membership uncertainty community,(6–8) and mixed parametric and nonparametric identification in a hard bound context.(9–11) The major requirement is a linear parameterization of the model. Hence, ARMA models can be dealt with in an equation-error approach. Output-error models where the parametric part is an FIR model or a linear combination of orthogonal filters (like Laguerre of Kautz filters),(11) can be tackled equally well. When dealing with purely parametric models, the *a priori* information generally consists in an initial uncertainty parallelotope for the parameters and a measurement error bound. When mixed parametric/nonparametric models are of concern, *a priori* information on the nonparametric part of the model becomes of crucial importance and it requires suitable techniques to translate it into an initial parallelotopic estimate $\mathcal{P}_0$. A good example models the nonparametric part via a FIR model cascaded with a suitable shaping filter.(11) The corresponding *a priori* information is mapped into an ellipsoid in the FIR parameter space. The *a priori* information which can be assumed in the context of parallelotopic approximations may be given in terms of

- a hard bound on the tail contribution of the nonparametric part of the model (equivalent to assuming a certain rate of decay of the impulse response of the nonparametric part);
- hard bounds on the errors between the first n samples of the 'true' impulse response samples and the FIR model parameters;
- hard bounds on discrepancies between the frequency-response magnitude of the nonparametric part and the truncated approximation.

10.3. OPTIMAL ADAPTATION OF THE PARALLELOTOPIC APPROXIMATION

In this section, a solution is provided to the following problem: given the parallelotope $\mathcal{P}_k$ and the new strip $\Sigma(k+1)$ provided by the $(k+1)$-th measurement, find the minimal-volume parallelotope $\mathcal{P}_{k+1}$ containing the polytope $\mathcal{V} \doteq \mathcal{P}_k \cap \Sigma(k+1)$.

Notice that $\mathcal{V}$ is the intersection of $n+1$ strips in the parameter space, each bounded by a pair of parallel hyperplanes. Clearly, some of these hyperplanes may not be tangent to $\mathcal{V}$; elementary geometrical considerations show that $\mathcal{V}$ is bounded by m supporting hyperplanes, with m varying from $n+1$ up to $2n+2$.

In order to compute the optimal outer-bounding parallelotope, it is necessary to check whether the corresponding hyperplanes are tangent to $\mathcal{V}$ for each strip. In fact, if both the bounding hyperplanes are not tangent to $\mathcal{V}$, then the strip does not provide any new information and therefore can be discarded. In this case, the problem is trivially solved since $\mathcal{V}=\mathcal{P}_{k+1}$ is the intersection of the remaining n strips. Conversely, if for a given strip only one hyperplane is not tangent to $\mathcal{V}$, then the strip must be 'tightened' by replacing the non-tangent hyperplane by a new parallel tangent hyperplane. Iterating this tightening procedure for all the $n+1$ strips leads to the following description of $\mathcal{V}$

$$\mathcal{V}=\left\{\bigcap_{i=1}^{n+1} \mathcal{S}_i\right\} \tag{10.10}$$

where $\mathcal{S}_i$ are defined as in Eq. (10.7) and all the strips are tight, i.e., all the hyperplanes $\sigma_i^+, \sigma_i^-, i=1, \ldots, n+1$ defined as in Eq. (10.8) are tangent to $\mathcal{V}$. Notice that the result of the tightening procedure is independent of the order according to which the strips are tightened. Implementation aspects will be discussed in the next section.

The next lemma provides a parameterization of a generic strip outer-bounding a polytope described by the intersection of tight strips.

LEMMA. Let $\mathcal{V}=\{\cap_{j=1}^{n+1} \mathcal{S}_j\}$ and let $\mathcal{S}_j, j=1, \ldots, n+1$, be tight with respect to $\mathcal{V}$. Then any strip $\bar{\mathcal{S}}_i$ outer-bounding $\mathcal{V}$ can be expressed as

$$\bar{\mathcal{S}}_i \doteq \{\theta : |\bar{p}_i'\theta - \bar{c}_i| \leq 1\} \tag{10.11}$$

where $\bar{p}_i$ and $\bar{c}_i$ are given by

$$\bar{p}_i=\sum_{j=1}^{n+1} a_{ij} p_j, \quad \bar{c}_i=\sum_{j=1}^{n+1} a_{ij} c_j,$$

with

$$\sum_{j=1}^{n+1} |a_{ij}| \leq 1. \tag{10.12}$$

In order to find the minimal-volume parallelotope, define, for $j=1, \ldots, n+1$, the $n+1$ matrices and vectors

$$P^j \doteq [p_1, \ldots, p_{j-1}, p_{j+1}, \ldots, p_{n+1}]' \in R^{n,n}$$

$$c^j \doteq [c_1, \ldots, c_{j-1}, c_{j+1}, \ldots, c_{n+1}] \in R^n \tag{10.13}$$

The result of this section can be stated in the following theorem.

THEOREM. The minimal-volume parallelotope $\mathcal{P}(T, \theta^c)$ outer-bounding $\mathcal{V}=\cap_{j=1}^{n+1} \mathcal{S}_j$ is given by

$$T=(P^{j^*})^{-1}, \quad \theta^c=(P^{j^*})^{-1} c^{j^*},$$

where

$$j^* = \arg \max_{j=1,\dots,n+1} \{|\det P^j|\}.$$

Proof. According to the Lemma, any parallelotope containing $\mathcal{V}$ can be expressed as the intersection of n strips given by Eq. (10.11). Therefore, it is clear from Eq. (10.9) that the problem of finding the minimum volume parallelotope outer-bounding $\mathcal{V}$amounts to the following mathematical programming problem

$$\max_{a_{ij}} \left\{ \left| \det \left[\sum_{j=1}^{n+1} a_{1j} p_j, \dots, \sum_{j=1}^{n+1} a_{nj} p_j \right] \right| \right\}, \tag{10.14}$$

subject to the constraints of Eq. (10.12). In fact, the coefficients $\{a_{ij}\}$ for which the maximum is attained in Eq. (10.14) provide the parametrization of the optimal parallelotope. Exploiting the linear dependence of the determinant on the coefficients a_{1j}, Eq. (10.14) can be rewritten as

$$\max_{a_{1j}} \left\{ \left| \sum_{j=1}^{n+1} a_{1j} \max_{a_{ij,i>1}} \left\{ \det \left[p_j, \sum_{j=1}^{n+1} a_{2j} p_j, \dots, \sum_{j=1}^{n+1} a_{nj} p_j \right] \right\} \right| \right\}, \tag{10.15}$$

which, taking into account the constraints (12) on the coefficients a_{1j}, reduces to

$$\max_j \left\{ \max_{a_{ik,i>1,k\neq j}} \left\{ \left| \det \left[p_j, \sum_{k=1}^{n+1} a_{2k} p_k, \dots, \sum_{k=1}^{n+1} a_{nk} p_k \right] \right| \right\} \right\}. \tag{10.16}$$

Notice that the constraint $k \neq j$ in Eq. (10.16) allows one to rule out the possibility that det[·] becomes null. Repeating the same argument for the other rows of the matrix in Eq. (10.14), one finds out that the optimal parallelotope is determined by the matrix P^j with maximal determinant. Hence the theorem is proved. □

Remark 10.1. The preceding theorem implies that the minimal-volume parallelotope is given by the intersection of n out of the $n + 1$ strips defining $\mathcal{V}$. Moreover, as can be easily checked by the proof, the result can be generalized to the case when $\mathcal{V}$is given by the intersection of an arbitrary number $N > n$ of tight strips.

10.4. RECURSIVE UNCERTAINTY SET ESTIMATION

This section presents a recursive algorithm for outer-bounding the parameter set via parallelotopes. The input of the algorithm at time $k + 1$ is the estimate $\mathcal{P}_k$ and the strip $\Sigma(k + 1)$ representing the $k + 1$-th measurement.

In order to apply the Theorem proved in the preceding section, the tightening procedure must be carried out for the $n + 1$ strips defining $\mathcal{V}$. Therefore, for each strip, one must check if the supporting hyperplanes are tangent to $\mathcal{V}$, i.e., if they intersect the parallelotope defined by the remaining n strips. Let us now illustrate this tightening procedure by considering the strip $\Sigma(k+1)$. In this case, one has to check if the hyperplanes

$$\sigma^+ = \{\theta : \phi'(k)\theta - y(k)\} = r(k)$$

$$\sigma^- = \{\theta : \phi'(k)\theta - y(k)\} = -r(k) \tag{10.17}$$

intersect the parallelotope $\mathcal{P} = \mathcal{P}(T(k),\theta^c(k))$. (Explicit dependence of ϕ, y, r, $\mathcal{P}$, and so forth, on k will be dropped hereafter to simplify notation).

Since from Eq. (10.5)

$$\begin{aligned} +\max_{\theta\in\mathcal{P}} \{\phi'\theta - y\} &= \phi'\theta^c - y + \sum_{i=1}^{n} |\phi' t_i| \doteq r^+ \\ \min_{\theta\in\mathcal{P}} \{\phi'\theta - y\} &= \phi'\theta^c - y - \sum_{i=1}^{n} |\phi' t_i| \doteq r^- \end{aligned} \tag{10.18}$$

the hyperplane σ^+ (σ^-) intersects $\mathcal{P}$ if

$$r^+ \geq r \; (r^- \leq -r). \tag{10.19}$$

Thus, if one of the two conditions of Eq. (10.19) does not hold, the strip

$$\Sigma = \{\theta : |\phi'\theta - y| \leq r\}$$

must be modified. The tightened strip, denoted by $\mathcal{S}_{n+1}$, will be given by

$$\mathcal{S}_{n+1} = \{\theta : |p'_{n+1}\theta - c_{n+1}| \leq 1\} \tag{10.20}$$

where

$$p_{n+1} = 2\phi/(\bar{r} - \underline{r}), \; c_{n+1} = y + (\bar{r} + \underline{r})/2 \tag{10.21}$$

and

$$\bar{r} \doteq \min(r, r^+), \; \underline{r} \doteq \max(-r, r^-).$$

Clearly, if both conditions of Eq. (10.19) hold, then $\mathcal{S}_{n+1} \equiv \Sigma$. Notice that the procedure outlined above must be applied also to each of the n strips defining $\mathcal{P}_k$. This is due to the fact that $\mathcal{V}_k$ is a polytope not necessarily preserving the parallelotopic structure of $\mathcal{P}_k$.

An algorithm for recursive parameter uncertainty estimation is based on the results presented before.

Step 1 Compute a description of $\mathcal{V}_k \doteq \mathcal{P}_k \cap \Sigma(k+1)$ in terms of $n+1$ tightened strips $\mathcal{S}_i$ like in Eq. (10.10), following the procedure outlined above.

Step 2 Form the $n+1$ matrices P^j and vectors c^j defined in Eq. (10.13).

Step 3 Solve

$$j^* = \arg\{\max_{j=1,\ldots,n+1} \{|\det P^j|\}\}.$$

Step 4 Set

$$\mathcal{P}_{k+1} = \bigcap_{i=1, i\neq j^*}^{n+1} \mathcal{S}_i$$

and compute

$$T(k+1) = (P^{j^*})^{-1}, \ \theta^c(k+1) = (P^{j^*})^{-1} c^{j^*}.$$

Remark 2. The present version of the algorithm, requiring several matrix inversions, is computational heavy. However, it can be shown that exploiting the close relationships among the matrices P^j defined in Eq. (10.13), Steps 1–3 can be carried out without any matrix inversion or determinant computation.(19) Therefore, only the matrix inversion of Step 4 is required.

Remark 3. As already noticed, if both the supporting hyperplanes of the i-th strip are nontangent to $\mathcal{V}$, then the other strips are tight and $j^* = i$. It can be also shown that if only one hyperplane of the i-th strip is nontangent, then $j^* = i$, independently of the fact that the other strips are tight or not.(19) This has an important implication for the parallelotope orientation. In fact, if the diameter of $\mathcal{P}_k$ is smaller than the width of the strip $\Sigma(k+1)$ associated to the $(k+1)$ measurement, then, necessarily, at least one hyperplane of $\Sigma(k+1)$ does not intersect $\mathcal{P}_k$. In this case $\mathcal{P}_k$ and $\mathcal{P}_{k+1}$ will share the same orientation.

Remark 4. A simplified version of the recursive algorithm can be employed for deriving orthotopic approximations of the parameter uncertainty set. For this problem, only Step 1 of the algorithm needs be performed. In fact, orientations of the hyperplanes bounding the approximating parallelotope are not free in this case; they are fixed by the orthotopic shape assumption.

10.5. CONCLUDING REMARKS

In this chapter an algorithm has been proposed for recursive estimation of the parameter uncertainty set in a linear regression model. A hard-bound setting of the underlying identification problem has been considered. The procedure provides an outer approximation of the uncertainty set alternative to commonly used ellipsoidal bounds. Several ramifications of the problem solved in this chapter may be the object of further investigation. Numerical efficiency and robustness of different techniques for implementing the algorithm; convergence of the algorithm to the minimum-volume parallelotope bounding the true parameter uncertainty set; performance evaluation of the parallelotopic approximation as compared to the ellipsoid-based techniques; mapping different kinds of prior knowledge on measurement noise and unmodelled dynamics into initial uncertainty estimates represent but some of the interesting and widely open problems deserving attention in future investigation.

ACKNOWLEDGMENTS. This work was partially supported by funds of the Ministero della Università e della Ricerca Scientifica e Tecnologica and CNR.

REFERENCES

1. L. Ljung, *System Identification: Theory for the User*, Prentice-Hall, Englewood Cliffs, N.J. (1987).
2. M. Gevers, in: *Proceedings of the 9th IFAC Symposium on Identification and System Parameter Estimation*, Budapest, Hungary, pp. 1–10 (1991).
3. G. C. Goodwin and M. E. Salgado, *Int. J. Adapt. Control Signal Proc.* **3**, 333 (1990).
4. A. J. Helmicki, C. A. Jacobson and C. N. Nett, *IEEE Trans. Autom. Control* **36**, 1163 (1991).
5. D. N. C. Tse, M. A. Dahleh and J. N. Tsitsiklis, in: *Proceedings of the International Workshop on Robust Control*, CRC Press, San Antonio, TX, pp. 311–328 (1991).
6. J. P. Norton, *Automatica* **23**, 497 (1987).
7. E. Walter and H. Piet-Lahanier, *Math. Comput. Simul.* **32**, 449 (1990).
8. M. Milanese and A. Vicino, *Automatica* **27**, 997 (1991).
9. R. C. Younce and C. E. Rohrs, in: *Proceedings of the 29th CDC*, Honolulu, HI, pp. 3154–3161 (1990).
10. R. L. Kosut, M. K. Lau and S. P. Boyd, *IEEE Trans. Autom. Control* **37**, 929 (1992).
11. B. Wahlberg and L. Ljung, *IEEE Trans. Autom. Control* **37**, 900 (1992).
12. J. C. Doyle, Analysis of feedback systems with structured uncertainties, *IEE Proc., Part D 129*, 242–250 (1982).
13. J. C. Doyle, J. E. Wall and G. Stein, in: *Proceedings of the 21st IEEE CDC*, Orlando, FL, pp. 629–636 (1982).
14. H. Chapellat, M. Dahleh, and S. P. Bhattacharyya, *IEEE Trans. Autom. Control* **35**, 1100 (1990).
15. M. Dahleh, A. Tesi and A. Vicino, *Automatica* **29**, 707 (1993).
16. A. C. Bartlett, C. V. Hollot and L. Huang, *Math. Control. Signal, and Syst. 1*, 61 (1988).
17. E. Fogel and Y. F. Huang, *Automatica* **18**, 229 (1982).
18. G. Belforte, B. Bona and V. Cerone, *Automatica* **26**, 887 (1990).
19. A. Vicino and G. Zappa, *Sequential Approximation of Parameter Sets for Identification with Parametric and Nonparametric Uncertainty*, Tech. Rep. DSI/RT-12/93, Università di Firenze, Firenze, Italy (1993).

11

Worst-Case l_1 Identification

M. Milanese

ABSTRACT

In this chapter recent results on nonparametric and mixed parametric-nonparametric l_1 identification are reviewed. These results mainly concern the evaluation of the identification errors, the design of experiment, the selection of the model structure, the construction of optimal and almost optimal algorithms, and the convergence properties of the identification algorithms.

11.1. INTRODUCTION

Most of the literature on set membership identification developed in the 70s and 80s focused on parametric approaches of the problem.[1–4] A review of the literature can also be found in Chapter 2 of this volume. In the parametric approaches, the structure of the model to be estimated is supposed to be given, typically a difference or differential equation of fixed order. The aim is to estimate the vector of unknown parameters to represent the equation coefficients.

In the 90s, much attention has been devoted to nonparametric approaches. The problem is to estimate the impulse response or the transfer function of time invariant, linear, possibly infinite dimensional systems.[5] In this way, weaker assumptions on the system to be identified are used, rather than with a parametric approach. However, as expected, very large estimation errors are obtained generally. To overcome these problems, mixed parametric-nonparametric approaches

M. MILANESE • Dipartimento di Automatica e Informatica, Politecnico di Torino, 10129 Torino, Italy.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

have been investigated recently, where it is considered that the system to be identified can be described by a parametric model perturbed by a nonparametric error system, which represents the unmodeled dynamics.[6–9]

Most of literature on these topics can be classified, according to the norm measuring the estimation errors, in two main categories, namely H_∞ and l_1 identification.

The aim of this chapter is to review recent results on nonparametric and mixed parametric-nonparametric l_1 identification of discrete time invariant linear systems. The motivation for studying worst-case l_1 identification is twofold. First, the model minimizing the l_1 norm of the impulse response error gives the minimal absolute prediction error. Second, l_1 identification provides the information needed to apply l_1 to modern robust control design techniques.[10]

11.2. PROBLEM FORMULATION

The class of plants considered in this chapter consists of causal, single-input single-output, linear, time-invariant, and discrete-time systems. This class is identified with the space H of one-sided, real sequences $h = \{h_0, h_1, \ldots\}$, representing the impulse response of the plants. The aim is to estimate the first $n + 1$ samples of h, that is to estimate $h^n = T^n h$, where T^n is the truncation operator:

$$h^n = T^n h = [h_0, \ldots, h_n]^t \qquad (11.1)$$

Suppose that two kinds of information may be available. The first one, often referred to as *a priori information* is expressed by assuming that $h \in K$, where K is a subset of H. From a modeling point of view, K is used to restrict the class of models, which the system to be identified is supposed belonging to. An important distinction among parametric and nonparametric identification methods can be made according to the dimensionality of the set K. This classification has particular relevance in connection with the achievable levels of the identification errors and the "informational complexity" of the identification procedure, as discussed in Section 11.3.

Nonparametric identification methods are characterized by large dimensionality of the set K. Typical sets considered in the nonparametric approaches are:

$$K_S = \{h \in H \colon \Sigma_{j=0}^{\infty} |h_j| < \infty\}:$$

set of BIBO stable systems.

$$K_E = \{h \in H \colon |h_j| \le L\rho^{-j}, \rho > 1, j = r, \ldots, \infty\}:$$

set of exponentially stable systems with a given degree of stability, if $r = 0$. Unless specified otherwise, $r = 0$ is assumed.

$$K_F = \{h \in H: |h_j| = 0, j = r, \ldots, \infty\}:$$

set of FIR systems of order r.

Parametric identification methods assume as K the set of the impulse responses $h(p)$ of a set of parametric models $\mathcal{M}(p)$ depending on a (possibly) low dimensional parameter vector p. In order to take explicitly into account that real systems cannot be exactly represented by low order models, a mixed parametric-nonparametric approach can be taken. This chapter considers *a priori* information of the type:

$$K_M = \{h \in H: h_j = h_j^{\mathcal{M}}(p) + h_j^{\varepsilon},$$

$$p \in \Pi \subseteq R^l, |h_j^{\varepsilon}| \leq L\rho^{-j}, \rho > 1, j = 0, 1, \ldots, \infty\}:$$

set of mixed parametric-nonparametric models, with the parametric part $\mathcal{M}(p)$ depending on an l dimensional parameter vector p and with exponentially stable unmodeled dynamics.

Only classes of models $\mathcal{M}(p)$ linear in the parameters are considered, having the impulse response samples linear functions of p:

$$h_j^{\mathcal{M}}(p) = \sum_{i=1}^{l} m_{ji} p_i = (Mp)_j \quad j = 0, 1, \ldots, \infty \tag{11.2}$$

There are several ways of representing models linear in the parameters, e.g., the FIR, Laguerre, and Kautz models. Alternatively, models nonlinear in the parameters, such as ARX models, may be linearized.(6,7,9)

The second kind of information is usually provided by a finite number of measurements performed during some experiments on the system to be identified. Consider experimental conditions consisting in the knowledge of the first $N + 1$ components of m output sequences $y^{(i)}$ related to m one-sided input sequences $u^{(i)}$ by

$$y_j^{(i)} = \sum_{k=0}^{j} h_k u_{j-k}^{(i)} + e_j^{(i)}, \; j = 0, 1, \ldots, N, \quad i = 1, 2, \ldots, m \tag{11.3}$$

Assume zero initial condition and $\|u^{(i)}\|_\infty \leq 1, \forall\, i$. The disturbance sequences $e^{(i)}$ are unknown but l_∞^w bounded, i.e.,

$$|e_j^{(i)}| \leq w_j \varepsilon \quad j = 0, 1, \ldots, N, \quad i = 1, 2, \ldots, m \tag{11.4}$$

where w_j are given positive weights. For the sake of simplicity, consider $w_j = 1, \forall j$, though several of reported results extend to the general case.

Equation (11.3) can be rewritten in a more compact form as

$$\mathbf{y} = Fh + \mathbf{e} \tag{11.5}$$

where

$$\mathbf{y} = [y_0^{(1)}, \ldots, y_N^{(1)}, \ldots, y_0^{(m)}, \ldots, y_N^{(m)}]^t,$$

$$\mathbf{e} = [e_0^{(1)}, \ldots, e_N^{(1)}, \ldots, e_0^{(m)}, \ldots, e_N^{(m)}]^t$$

and F is the linear operator

$$F = \begin{bmatrix} U_1 T^N \\ U_2 T^N \\ \ldots \\ U_m T^N \end{bmatrix} \tag{11.6}$$

where U_i is the lower triangular Toeplitz matrix formed by input $u^{(i)}$:

$$U_i = \begin{bmatrix} u_0^{(i)} & 0 & \cdots & 0 \\ u_1^{(i)} & u_0^{(i)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ u_N^{(i)} & u_{N-1}^{(i)} & \cdots & u_0^{(i)} \end{bmatrix}. \tag{11.7}$$

An identification algorithm is a (possibly nonlinear) operator ϕ: $Y \to R^{n+1}$ providing an estimate $\hat{h}^n = \phi(\mathbf{y})$ to $T^n h$, using the corrupted information $\mathbf{y}$.

Look for estimates minimizing the l_1 error. The interest in using this measure of the identification error is twofold. First, interesting techniques exist for robust control design techniques in the l_1 setting.[10] Second, a model minimizing the l_1 impulse response error gives minimal absolute prediction error. If $y_j^o = \Sigma_{k=0}^j h_k^o u_{j-k}$ is the output of the "true" plant h^o and $\hat{y}_j = \Sigma_{k=0}^j \hat{h}_k^\infty u_{j-k}$ is the predicted output using an estimate $\hat{h}^\infty$ of h^o, the following tight bound holds:[11]

$$|y_j^o - \hat{y}_j| \le \|h^o - \hat{h}^\infty\|_1 \cdot \|u\|_\infty, \quad \forall j, \ \forall u \tag{11.8}$$

Since h^o is not known, a worst case approach is taken as usual in the set membership identification, by defining the identification error as

$$E(\phi,\varepsilon) = \sup_{\mathbf{y}} \sup_{h \in FSS_{\mathbf{y}}} \|T^n h - \phi(\mathbf{y})\|_1 \tag{11.9}$$

where $FSS_{\mathbf{y}}$ is the feasible system set, i.e., the set of plants consistent with corrupted information $\mathbf{y}$,

$$FSS_y = \{h \in K : |y_j^{(i)} - \sum_{k=0}^{j} h_k u_{j-k}^{(i)}| \le \varepsilon \quad j = 0, 1, \ldots, N, \quad i = 1, 2, \ldots, m\} \tag{11.10}$$

It is clear from Eq. (11.10) that if no *a priori* information is given ($K = H$), measurements give no information on h_k for $k > N$. In most cases *a priori* information is used to give information on the behavior of the impulse response just for $k > N$, while information for $k \leq N$ can be derived directly from the measurements. In order to make use of *a priori* information only when necessary, only sets K giving limitations on h_k for $k > N$ only may be considered. *A priori* information of this type, will be denoted as residual. For example, *a priori* information provided by K_F or K_E, with $k = N$, is residual.

Assuming that the reader is familiar with the main concepts and results in set membership estimation theory, as briefly reported in Chapter 2 of this book, the following few other concepts are needed.

The minimal worst case error is called radius of information $R(\varepsilon)$

$$R(\varepsilon) = \inf_{\phi} E(\phi,\varepsilon) \tag{11.11}$$

Useful bounds on $R(\varepsilon)$ are often found in terms of the diameter of information $D(\varepsilon)$, expressed as

$$D(\varepsilon) = 2 \sup_{h \in FSS_0} \|T^n h\|_1. \tag{11.12}$$

In fact, if K is balanced (i.e., symmetric with respect the origin) and convex, then[12]

$$0.5D(\varepsilon) \leq R(\varepsilon) \leq D(\varepsilon) \tag{11.13}$$

Note the sets K considered above are balanced and convex.

The overall estimation process is indicated as identification procedure and is defined by specifying K, u, ϕ, ε, y, N, m, n. For example, two identification procedures may only differ by *a priori* assumptions on K or because different inputs are used. To put in evidence the dependence of E, R and D on N, m and n, the notation $E_n^{N,m}$, $R_n^{N,m}$ and $D_n^{N,m}$ are used when necessary.

An identification procedure is called convergent if

$$\lim_{\varepsilon \to 0} \lim_{N \to \infty} E(\phi,\varepsilon) = 0. \tag{11.14}$$

Note that in the literature there is not yet a unified terminology for convergence concepts. Some authors use the terms robust convergence or uniform convergence for the above definition. Other authors use the term robust convergence for weaker or stronger convergence concepts. For the sake of simplicity, only results related to the above definition are reported.

Results related to the evaluation of the identification errors, the design of experiment, the selection of the model structure, the construction of optimal and almost optimal algorithms, and the convergence properties of identification algorithms are reviewed.

11.3. IDENTIFICATION ERRORS AND MODEL STRUCTURE SELECTION

Most of the results are based on the analysis of the diameter of information, which is easier to evaluate than the radius of information, and is related to the radius through by Eq. (11.13). Moreover, the diameter of information provides a tight bound of the identification errors of almost optimal algorithms, such as interpolatory or projection algorithms (see next section).

In view of Eq. (11.8) the case $n = \infty$ is of particular interest. Most papers study $D_\infty^{N,m}(\varepsilon)$. This diameter is bounded below by $D_\infty^{N,m}(0)$, representing the inherent uncertainty, due to the limited number of measurements used for the identification. References 13 and 14 show that $D_\infty^{N,1}(0)/2$ is related to the Kolmogorov and Gelfand N-widths of set K, well known concepts in approximation theory.[15] Reference 14 shows that if $K = K_E$

$$\frac{2L}{\rho^{N-1}(\rho-1)} \leq D_\infty^{N,1}(0), \quad \forall u. \tag{11.15}$$

This bound is tight in the sense that equality holds for some u, e.g., the unit impulse sequence.

From Eq. (11.12) it follows that

$$2 \sup_{h \in K} \sum_{k=N}^{\infty} |h_k| \leq D_\infty^{N,m}(\varepsilon) \leq D_N^{N,m}(\varepsilon) + 2 \sup_{h \in K} \sum_{k=N}^{\infty} |h_k| \tag{11.16}$$

where the right inequality is an equality if $K = K_E$ or $K = K_F$. In particular, if $K = K_E$

$$D_\infty^{N,m}(\varepsilon) = D_N^{N,m}(\varepsilon) + \frac{2L}{\rho^{N-1}(\rho-1)} \tag{11.17}$$

From Eq. (11.16), it is clear that if no *a priori* information is assumed or if $K = K_S$, the identification error $E_\infty^{N,m}$ (ϕ,ε) is not finite whatever algorithm ϕ is used.

Eqs. (11.16) and (11.17) allow evaluation or bounding of $D_\infty^{N,m}(\varepsilon)$ in terms of $D_N^{N,m}(\varepsilon)$ and the assumed *a priori* information. Now, some results related to $D_N^{N,m}(\varepsilon)$ are reported.

In the case of residual *a priori* information, a simple lower bound has been derived for $m = 1$:[16,17]

$$2N\varepsilon \leq D_N^{N,1}(\varepsilon), \quad \forall\, u. \tag{11.18}$$

This bound is tight since, if u is the unit impulse sequence, then[16]

$$2N\varepsilon = D_N^{N,1}(\varepsilon) \tag{11.19}$$

It follows that in order to go below $2N\varepsilon$, it is necessary to have stronger information than the residual one (which does not assume any *a priori* information on h_k, $k = 0, 1, \ldots, N$) or to use more than one input sequence ($m > 1$).

If $K = K_E$, the following lower and upper bounds on $D_N^{N,1}(\varepsilon)$ have been derived in:[18]

$$\max_{0 \le k \le N-1} \min\{\varepsilon \sum_{j=0}^{k} |v_{k-j}|, L\rho^{-k}\} \le D_N^{N,1}(\varepsilon)$$

$$\le \sum_{k=0}^{N-1} \min\{2\varepsilon \sum_{j=0}^{k} |v_{k-j}|, 2L\rho^{-k}\} \tag{11.20}$$

where $\nu_0, \nu_1, \ldots, \nu_{N-1}$ are the elements of the first column of the matrix $V = U^{-1}$.

Even if only residual *a priori* information is assumed, $D_N^{N,m}(\varepsilon)$ can be reduced down to 2ε, using suitable inputs and a sufficiently large number of experiments m. It has been shown[16] that if $K = H$ and u is the sequence of all binary N-tuples of ± 1 (Galois sequence), then

$$D_N^{N,2^{N+1}}(\varepsilon) = 2\varepsilon \tag{11.21}$$

In the same references it is also shown that if $m < 2^{N-1}$, the diameter is strictly greater than 2ε. Note that a diameter lower than 2ε can be achieved only under very strong (and implausible) *a priori* assumptions. Provided the system has $|h_k| > \varepsilon$ for some $k \le N$, then $D_N^{N,m}(\varepsilon) \ge 2\varepsilon$.[16]

In the case of residual *a priori* information, a lower bound, function of m, has been obtained:[19]

$$\frac{2(N+1)}{1 + \sqrt{2(N+1)\ln[2m(N+1)(N+2)]}}\,\varepsilon \le D_N^{N,m}(\varepsilon), \quad \forall u \tag{11.22}$$

This bound implies that the number of measurements needed to obtain a diameter not exceeding a given threshold grows exponentially with the number impulse response samples to be estimated. Denoting by m_c the minimum number of experiments such that $D_N^{N,m}(\varepsilon) \le 2c\varepsilon$, $c > 1$, gives[16]:

$$m_c \ge \frac{2^N}{(N+1)(N+2)\sum_{k=0}^{\nu_c} \binom{N+1}{k}} \tag{11.23}$$

where $\nu_c = \lceil (N+1)(c-1)/2c \rceil$ and $\lceil\ \rceil$ indicates the roundup function.

Similar lower bounds have been derived(20,21) for the case $m = 1$. In particular, denoting by N_c the minimum number of measurements such that $D_n^{N,1}(\varepsilon) \le 2c\varepsilon$, $c > 1$, the following asymptotically tight lower bound has been derived for the case $K = K_F$ and $n = r$:(20)

$$N_c \ge 2^{ng(1/c)-1} - n + 2(n/c - 1) \qquad (11.24)$$

where

$$g(\alpha) = 1 + \frac{(1-\alpha)}{2}\log_2\frac{(1-\alpha)}{2} + \frac{(1+\alpha)}{2}\log_2\frac{(1+\alpha)}{2}.$$

Equations (11.23) and (11.24) indicate that nonparametric l_1 identification suffers from large "informational complexity," i.e., the number of measurements needed to assure a given level of identification error grows exponentially with the number of impulse response samples to be estimated. This fact has been sometimes interpreted as a confirmation of the common belief that worst case estimation is too pessimistic. However, one reason for this exponential growth is that the representation of systems through impulse response samples is not "parsimonious," while it is well known that the use of parsimonious models is a key point for obtaining reliable identification results. Mixed parametric and nonparametric models have been investigated(9) in order to overcome such complexity problems. If $K = K_M$, it is shown that:

$$D_\infty^{N,1} = 2 \sup_{p \in FPS_0} \|h^{\mathcal{M}}(p)\|_1 + \varepsilon^{\mathcal{E}} \qquad (11.25)$$

where

$$\varepsilon^{\mathcal{E}} = \frac{L\rho}{(\rho-1)} \text{ and } FPS_0 = \{p \in R^l : \|FM_N p\|_\infty \le \varepsilon + \varepsilon^{\mathcal{E}}\}.$$

and M_N is the matrix formed by the first N rows of the seminfinite matrix M in Eq. (11.2).

The quantity

$$D^{\mathcal{M}} = 2 \sup_{p \in FPS_0} \|h^{\mathcal{M}}(p)\|_1 \qquad (11.26)$$

is the diameter of information for the parametric class of models $\mathcal{M}(p)$, and a method for its computation can be found in the above reference. Suppose the parametric part $\mathcal{M}(p)$ is refined by increasing the dimension l. The value of $\varepsilon^{\mathcal{E}}$, due to unmodeled dynamics, decreases while $D^{\mathcal{M}}$ increases, becoming unbounded for $l > N$. Thus the total diameter is minimal for some value $l^* \le N$. Typically, if parsimonious classes of models are used, such as linearized ARX models, l^* may be very low, considerably reducing the dimensionality of the problem. The results confirm that the informational complexity may be largely reduced with respect to the nonparametric approach.(9)

The above considerations suggest also that Eq. (11.25) can be used to compare the "goodness" of different classes of models. In particular, it can be used for the selection of the order the parametric part. The diameter of information represents a measure of the "predictive ability" of the considered class of models with respect to absolute error, analogous to statistical criteria such as FPE, AIC, and so forth, which give a measure of the predictive ability with respect to mean value error.(8,9,11,13)

The identification error of the least squares algorithm ϕ^{LS}, perhaps the most popular and widely used algorithm in system identification, has been also investigated. We report here some results related to the case $m = 1$, $K = K_M$. The results for $K = K_F$, with $l \leq N$, can be obtained as particular cases, while ϕ^{LS} is not a meaningful algorithm for $K = K_S$ or $K = K_E$.

If $m = 1$ and $K = K_M$, the least squares algorithm is the linear algorithm given by:

$$\phi^{LS}(\mathbf{y}) = A\mathbf{y} = M_n(M_N^T U^T U M_N)^{-1} M_N^T U^T \mathbf{y} \tag{11.27}$$

provided that the indicated inverse exist. M_n is the matrix formed with the first n rows of the semiinfinite matrix M in Eq. (11.2).

The following expression of its identification error is obtained:(9,22)

$$E_n^{N,1}\,(\phi^{LS},\varepsilon) = \varepsilon\|A\| + \frac{L\rho}{\rho - 1} \tag{11.28}$$

where $\|A\| = \sup_{\|\mathbf{y}\|_\infty \leq 1}\|Ay\|_1$.

The computation of $\|A\|$ can be performed by means of convex optimization programs, but it may become cumbersome for large N. In such a case, standard lower and upper bound of norms can be used,(22,23) leading to

$$\min\{\max_j \sum_i^n |a_{ij}|, \max_i \sum_j^N |a_{ij}|\} \leq \|A\| \leq n \min\{\max_j \sum_i^n |a_{ij}|, \max_i \sum_j^N |a_{ij}|\} \tag{11.29}$$

Note that error (11.28) can be arbitrarily larger than the radius of information.(22)

11.4. OPTIMAL AND ALMOST-OPTIMAL ALGORITHMS

Optimal algorithms, i.e., algorithms whose error equals the radius of information, can be found as central algorithms.(12,24) A central algorithm is obtained by finding the Chebyshev center of the set $FSS_y^n = T^n FSS_y$. This set is a polytope and its center may be derived by finding its vertices $[h^{(1)}, \ldots, h^{(v)}]$ and computing the point of minimum distance from all the vertices, i.e., solving the problem:

$$\min_{h^n \in R^n} \max_{k=1,\ldots,v} \|h^n - h^{(k)}\|_1 = rad(FSS_y^n) \tag{11.30}$$

Optimization Eq. (11.30) can be solved by linear programming.[25] However, in nonparametric approaches, the complexity of the computation of the vertices of FSS_y^n increases combinatorically with n.[26] Just for $n = 20 \div 30$, the computation complexity becomes quite large.

Complexity can be overcome by using mixed parametric-nonparametric classes of models. If $K = K_M$, Eq. (11.30) can be reduced to the computation of the vertices of an l-dimensional polytope.[9] Recall that if "parsimonious" models are used, the value of l may be quite small. Moreover, the number of vertices also is typically low. In fact, if linearized ARX models are used for the parametric part $\mathcal{M}(p)$, Monte Carlo simulations have shown that the mean number of vertices tends to be constant as N increases. In the example reported in Ref. 27, the mean number of vertices is 50 and 150, for $l = 4$ and $l = 5$, respectively.

Almost optimal algorithms (i.e., optimal within a factor of 2), can be constructed more easily.

An interpolatory algorithm $\phi^I(\mathrm{y})$ is defined as

$$\phi^I(\mathrm{y}) = T^n h_\mathrm{y}, \quad \text{where } h_\mathrm{y} \in FSS_y \tag{11.31}$$

Interpolatory algorithms are almost optimal, since:[12]

$$E(\phi^I, \varepsilon) \leq D(\varepsilon) \leq 2R(\varepsilon) \tag{11.32}$$

Interpolatory algorithms require finding a feasible point of the polytope FSS_y^n. This can be obtained, for example, through the solution of the linear program:

$$h_\mathrm{y}^n = \arg \min_{h^n \in FSS_y^n} \mathcal{L}h^n \tag{11.33}$$

where $\mathcal{L}$ is any given linear functional.

In this way, however, the feasible point is obtained on the edges of FSS_y^n, while more "centered" points should be desirable, as suggested by optimality of central algorithms. For example, the center of the maximal volume ball contained in FSS_y^n can be looked for. Reference 28 shows how to solve this problem by means of one linear program.

An alternative solution is to use the projection algorithm ϕ^P, given by:

$$\phi^P(y) = \hat{h}_y^n, \quad \hat{h}_y^n = \arg \min_{h^n \in T^n K} \|y - Fh^n\|_\infty. \tag{11.34}$$

If $n > N$, Eq. (11.34) has to be solved by substituting N for n. Note that the solution of Eq. (11.34) can be obtained by means of linear programming.(29) The projection algorithm is an almost optimal algorithm(30) which does not require the knowledge of ε. Clearly, this is an appreciable property for all problems where it is not possible or easy to have reliable information on the value of ε.

Generally, the least squares algorithm ϕ^{LS} is neither optimal nor almost optimal in l_1 identification.(30) However, ϕ^{LS} is optimal if FIR models are considered ($K = K_F$), and impulse or step sequences are used as inputs.(22) This result easily extends to the case $K = K_E$.

11.5. CONVERGENCE PROPERTIES

Most of the convergence results are based on the analysis of the diameter of information. As follows from Eq. (11.13), if $\lim_{\varepsilon \to 0} \lim_{N \to \infty} D(\varepsilon) = 0$, an identification procedure using optimal or almost-optimal algorithms is convergent. On the contrary, if the diameter of information is not convergent, no identification procedure can be convergent, whatever algorithm is used. In such a case, the only way to obtain a convergent procedure is to modify the *a priori* information or the experimental conditions.

The case of $n = \infty$ has been investigated mostly. In such a case, no convergent identification procedure can be found, unless some suitable *a priori* information is assumed. In fact, from Eq. (11.16) it follows that convergent identification procedures exist only if

$$\lim_{N \to \infty} \sup_{h \in K} \sum_{k=N}^{\infty} |h_k| = 0. \tag{11.35}$$

Equation (11.35) does not hold for K_S, while it holds true for K_E and K_F, and for K_M if $\mathcal{M}(p)$ is stable $\forall p \in FPS_0$.

Reference 31 shows that if $K = K_E$, the input has length $N \geq n + 2^{n+1}$, containing all possible Galois sequences, and the projection algorithm is used, then the obtained identification procedure is convergent. Similar results have been derived by using general results on the asymptotic behaviour of the diameter of information.(32) If $K = K_E$, any interpolatory algorithm using a nonzero input sequence is convergent.(18)

Despite the previous results, it may still be possible in the absence of *a priori* information to estimate an arbitrarily large number of impulse response samples as accurately as desired. From the results of Refs. 16 and 17, it follows that if $n = N$

and $K = H$, Eq. (11.21) holds. This implies that using $m = 2^{N-1}$ experiments, Galois sequences as inputs and an interpolatory algorithm, the resulting identification procedure gives an error in estimating N impulse response samples, which tends to zero as ε tends to zero without using any *a priori* information.

Convergence properties of the least squares algorithm, when $K = K_F$ or $K = K_M$, have been investigated.[22] It is shown that $E^N(\phi^{LS},\varepsilon)$ may be unbounded as $N \to \infty$. However, ϕ^{LS} is convergent if the system to be identified is stable, u and e are quasi-stationary, uncorrelated, and u is persistently exciting. These assumptions are the deterministic analogy of the typical conditions assuring consistency of the least squares estimates in a stochastic setting.

ACKNOWLEDGMENTS. Funds of Ministero dell'Università e della Ricerca Scientifica e Tecnologica and EC project SIMONET.

REFERENCES

1. M. Milanese and A. Vicino, *Automatica* **27**, 997 (1991).
2. E. Walter and H. Piet-Lahanier, *Math. Comp. Sim.* **32**, 499 (1990).
3. A. B. Kurzhanski, *Control and Observation under Conditions of Uncertainty*, Nauka, Moscow, Russia (1977).
4. V. M. Kuntzevich and M. Lychak, *Guaranteed Estimates, Adaptation and Robustness in Control Systems*, Lectures Notes in Control and Information Sciences, Springer-Verlag, Berlin, Germany (1992).
5. M. Milanese and A. Vicino, *J. Complex.* **9**, 427 (1993).
6. G. C. Goodwin, M. Gevers, and B. Ninness, *IEEE Trans. Autom. Control* **37**, 913 (1992).
7. B. Wahlberg and L. Ljung, *IEEE Trans. Autom. Control* **37**, 900 (1992).
8. R. L. Kosut, M. K. Lau, and S. P. Boyd, *IEEE Trans. Autom. Control* **37**, 929 (1992).
9. N. Elia and M. Milanese, in: *Proceedings of the 32nd IEEE CDC,* San Antonio, Texas, pp. 545–550 (1993).
10. M. A. Dahleh and J. B. Pearson, *IEEE Trans. Autom. Control* **32**, 314 (1987).
11. C. A. Desoer and M. Vidyasagar, *Feedback Systems: Input-Output Properties,* Academic Press, New York (1975).
12. J. F. Traub, G. W. Wasilkowski, and H. Woźniakowski, *Information-Based Complexity*, Academic Press, New York (1988).
13. P. M. Makila and J. R. Partington, in: *Proceedings of the 1991 ACC*, pp. 70–76 (1991).
14. L. Lin, L. Y. Wang, and G. Zames, in: *Proceedings of the ACC 1992*, pp. 296–300 (1992).
15. A. Pinkus, *n-Widths in Approximation Theory*, Springer-Verlag, Berlın, Germany (1985).
16. B. Kacewicz and M. Milanese, in: *Proceedings of the IEEE 31st Control and Decision Conference*, Tucson, AZ, pp. 56–61 (1992).
17. B. Kacewicz and M. Milanese, *Jour. Adapt. Contr. Signal Process* **9**, 87–96 (1994).
18. J. Chen, C. N. Nett, and M. K. Fan, in: *Proceedings of the 1992 American Control Conference*, pp. 279–286, Chicago, IL (1992).
19. M. Milanese and B. Kacewicz, in: *IIASA 1992 Workshop on Modeling Techniques for Uncertain Systems* (Kurzhansky and Veliov, eds.), Birkhauser, Boston, MA (1994).
20. M. A. Dahleh, T. Theodosopoulos, and J. N. Tsitsiklis, in: *Proceedings of the 32nd IEEE CDC*, San Antonio, Texas (1993).

21. K. Poolla and A. Tikku, *Proceedings of the 1993 ACC*, San Francisco, pp. 141–145 (1993).
22. M. Milanese, *Proceedings of IFAC-SYSID'94*, Copenhagen (1994), also *Automatica* **31**, 327 (1995).
23. N. Dunford and J. T. Schwarz, *Linear Operators*, vol. 1, Interscience, New York (1958).
24. M. Milanese and R. Tempo, *IEEE Trans. Autom. Control* **30**, 730 (1985).
25. K. Trustrum, *Linear Programming*, Chap. 2, Routledge and Kegan Paul, London (1971).
26. T. H. Matheiss and D. S. Rubin, *Math. Oper. Res.* **5**, 167 (1980).
27. S. H. Mo and J. P. Norton, *Math. Conput. Simul.* **32**, 481 (1990).
28. A. Vicino and M. Milanese, *IEEE Trans. Autom. Control* **36**, 759 (1991).
29. M. L. Overton, in: *Nonlinear Optimization 1981* (M. J. D. Powell, ed.), Academic Press, New York (1982).
30. B. Z. Kacewicz, M. Milanese, R. Tempo, and A. Vicino, *Syst. Control Lett.* **8**, 161 (1986).
31. P. M. Mäkilä, *International J. Control* **54**, 1189 (1991).
32. D. C. N. Tse, M. A. Dahleh, and J. N. Tsitsiklis, in: *Control of Uncertain Dynamic Systems* (S. P. Bhattacharyya and L. H. Keel, eds.), CFC Press, pp. 311–328 (1991).
33. C. A. Jacobson and C. N. Nett, in: *Proceedings of the 1991 American Control Conference*, Boston (1991).
34. P. M. Makila and J. R. Partington, in: *Proceedings of the 1992 American Control Conference*, pp. 301–306, Chicago (1992).
35. L. Ljung, *System Identification: Theory for the User*, Prentice-Hall, Englewood Cliffs, NJ (1987).

12

Recursive Robust Minimax Estimation

É. Walter and H. Piet-Lahanier

ABSTRACT

An important problem arising when one wants to estimate the parameters of a model in a bounded-error context is the specification of reliable bounds for this error. In early phases of development, when no prior information is available, one may wish to know the minimum upper bound for the amplitude of the error such that the feasible parameter set is not empty. This corresponds to using a minimax estimator. For models linear in their parameters, we describe a method that takes advantage of a reparametrization in order to recursively obtain the minimax estimates and associated bounds for the error. It also provides the set of parameters compatible with any upper bound of the error. This procedure is extended to output-error models, which are nonlinear in their parameters. Its robustness to outliers is discussed and a technique is described to detect and discard them.

12.1. INTRODUCTION

The problem of estimating the parameters of a model together with their uncertainty in the presence of noise has been widely discussed. The approach

É. WALTER • Laboratoire des Signaux et Systèmes, CNRS École Supérieure d'Électricité, 91192 Gif-sur-Yvette Cedex, France. H. PIET-LAHANIER • Direction des Études de Synthèse/SM Office National d'Études et de Recherches Aérospatiales F-92322, Châtillon Cedex, France.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

known as set membership estimation assumes that a nonstatistical description of the noise, under the form of bounds on its realizations, is available for each measurement.(1) Set membership estimation aims at characterizing the region in the parameter space that contains all parameter values consistent with the data, model structure, and bounds on the acceptable error between the data and model output. Here this set will be referred to as the (posterior) feasible parameter set $\mathbb{S}$. Several techniques have been developed to either determine $\mathbb{S}$ exactly or characterize a simple-shaped set containing it. For models linear in their parameters, $\mathbb{S}$, if it exists, is a convex polyhedron which, when bounded, i.e. when a polytope, can be approximated by ellipsoids,(2–4) or orthotopes(5) containing it. This polyhedron can also be described exactly by enumerating its vertices, unbounded edges and supporting hyperplanes. Broman and Shensa,(6) and Mo and Norton(7) present methods that are limited to the study of bounded polyhedra, whereas the technique Walter and Piet-Lahanier(8) developed is not. For models nonlinear in their parameters, various methods exist for determining an approximation of $\mathbb{S}$. Linear techniques have been extended to the nonlinear case using multiple linearization of the model.(9) For specific model structures, such as output-error models with a deterministic recurrence equation, it is possible to obtain sets of linear inequalities that must be satisfied for the parameters to belong to $\mathbb{S}$.(10,11) Signomial programming has also been suggested to compute an orthotope containing $\mathbb{S}$.(12) Random search methods have been designed either to compute points belonging to $\mathbb{S}$(13) or determine points on its boundary.(14)

In the bounded-error context, each new measurement is associated with two inequalities that the parameter vector must satisfy to belong to $\mathbb{S}$. These inequalities are functions of the bounds assumed for the acceptable error. Optimistic bounds may lead to the conclusion that no parameter vector is consistent with all data and that $\mathbb{S}$ is empty, whereas pessimistic bounds inflate the set and, therefore, overestimate the uncertainty on the parameters. In practice, the bounds on the error are usually defined by taking into account the user's knowledge of the system or the technical specifications the manufacturers provide for the measuring devices. However, during the early phases of the study of a new system, one may be at a loss to define reliable bounds on the errors. A possible way to partly overcome this difficulty consists in describing $\mathbb{S}$ as a function of the bound on the error. This involves computing the minimum value of the bound that results in a non-empty $\mathbb{S}$.

This chapter presents a method to determine such a bound for models linear in their parameters. An algorithm is described that recursively updates the bound whenever a new measurement is taken into account. As a by-product, it provides a description of $\mathbb{S}$ for any bound larger than the minimum value. An extension of this technique to the study of output-error models is considered. The robustness of the method to outliers in the data is discussed and a technique is described to detect and discard them.

12.2. PROBLEM STATEMENT

The error $\varepsilon(k, \boldsymbol{\theta})$ is defined as

$$\varepsilon(k, \boldsymbol{\theta}) = y(k) - \boldsymbol{\phi}^{\mathrm{T}}(k)\boldsymbol{\theta}, \qquad (12.1)$$

where $y(k)$, $k = 1, \ldots, \mathrm{N}$, are the data, $\boldsymbol{\theta}$ is the p-dimensional vector of the parameters to be estimated and $\boldsymbol{\phi}(k)$ is the kth regressor. In the bounded-error context, the error should satisfy

$$|\varepsilon(k, \boldsymbol{\theta})| \leq \varepsilon_{\mathrm{max}}(k). \qquad (12.2)$$

To be consistent with the hypotheses, $\boldsymbol{\theta}$ must then belong to the solution set $\mathbb{S}$ of the following set of inequalities

$$|\boldsymbol{\phi}^{\mathrm{T}}(k)\boldsymbol{\theta} - y(k)| \leq \varepsilon_{\mathrm{max}}(k), \ k = 1, \ldots, N. \qquad (12.3)$$

In most papers, $\varepsilon_{\mathrm{max}}(k)$ is assumed to be known *a priori*. However, it may happen that the available information is too scarce to allow one to define reliable bounds $\varepsilon_{\mathrm{max}}(k)$. This chapter considers such a situation and assumes that all $\varepsilon_{\mathrm{max}}(k)$ are equal to $\varepsilon_{\mathrm{max}}$, which is unknown. Provided that $\varepsilon_{\mathrm{max}}$ is large enough, any $\boldsymbol{\theta}$ can be considered as acceptable. One would then like

(i) to estimate the minimum value $\hat{\varepsilon}_{\mathrm{max}}$ of $\varepsilon_{\mathrm{max}}$ associated with a non-empty $\mathbb{S}$; and
(ii) to obtain a description of all sets $\mathbb{S}$ associated with $\varepsilon_{\mathrm{max}} \geq \hat{\varepsilon}_{\mathrm{max}}$.

The set $\mathbb{S}$ obtained as the solution of problem (i) is a minimax estimate of $\boldsymbol{\theta}$, given by

$$\mathbb{S}_{\mathrm{mm}} = \{\hat{\boldsymbol{\theta}} \mid \hat{\boldsymbol{\theta}} = \operatorname{Arg} \min_{\boldsymbol{\theta}} \max_{k} |\boldsymbol{\phi}^{\mathrm{T}}(k)\boldsymbol{\theta} - y(k)|\}. \qquad (12.4)$$

Laplace[15] seems to have introduced this minimax (L_∞) estimator and later Fourier[16] and Cauchy[17] developed it. A very interesting account of the historical development of L_∞ estimation can be found in Farebrother.[18] It is well known that the criterion associated with minimax estimation is not differentiable everywhere, especially in a neighborhood of the optimum, so that specific algorithms are needed. An important special case is polynomial approximation, where the regressor takes the form

$$\boldsymbol{\phi}(k) = [1, t_k, t_k^2, \ldots, t_k^{p-1}]^{\mathrm{T}}, \ k = 1, \ldots, N. \qquad (12.5)$$

For this problem, the exchange algorithm of Remes[19] is especially appealing, because of its extreme simplicity. It would be tempting to transpose it to the more general problem of linear minimax estimation. This may lead to erroneous results, as illustrated in,[20] because the regressor does not have the same properties as the ones defined by Eq. (12.5).

The minimax estimation problem of Eq. (12.4) can be transformed into a differentiable problem under constraints[21,22] by introducing an additional variable x and determining $\boldsymbol{\theta}$ such that

$$\hat{\boldsymbol{\theta}} = \operatorname*{Arg\,min}_{\boldsymbol{\theta}} x, \tag{12.6}$$

subject to the constraints

$$x - \boldsymbol{\phi}^{\mathrm{T}}(k)\boldsymbol{\theta} + y(k) \geq 0, k = 1, \ldots, N, \tag{12.7a}$$

$$x + \boldsymbol{\phi}^{\mathrm{T}}(k)\boldsymbol{\theta} - y(k) \geq 0, k = 1, \ldots, N. \tag{12.7b}$$

Example 1: Suppose that four measurements have been performed on a system and that the results are those given in Table 12.1. These data are to be described by

$$y(k) = \theta\ \phi(k) + \varepsilon(k). \tag{12.8}$$

In Fig. 12.1, the constraints of types Eqs. (12.7a) and (12.7b) associated with the data of Table 12.1 are drawn in the (x, θ) plane. The shaded area corresponds to the set of all pairs (x, θ) consistent with all constraints. The minimax estimate of θ can be read directly as $\hat{\theta} = 0.75$ and $\hat{\varepsilon}_{\max} = 0.75$.

Remarks: The set of all (x, θ) consistent with all constraints is a polyhedral cone, i.e., an *unbounded* polyhedron.

To obtain the feasible parameter set associated with any given value of $\varepsilon_{\max}$, one only has to add the inequality $x \geq \varepsilon_{\max}$ to Eqs. (12.7(a) and (b)). For instance, if $\varepsilon_{\max} = 1$, one immediately obtains $\theta \in [0.5, 1]$. The exact description of the feasible polytope for (x, θ) thus contains the exact description of $\mathbb{S}$ for any $\varepsilon_{\max} \geq \hat{\varepsilon}_{\max}$ as a by-product.

The problem defined by eqs. (12.6) and (12.7(a) and (b)) can be viewed as a linear programming problem and thus could be solved by classical techniques such as the simplex[23] or projection[24] algorithms. In their basic form, these algorithms

TABLE 12.1. Data Set for Example 1

k	ϕ	y
1	1	1.5
2	2	1.5
3	3	2
4	−1	0

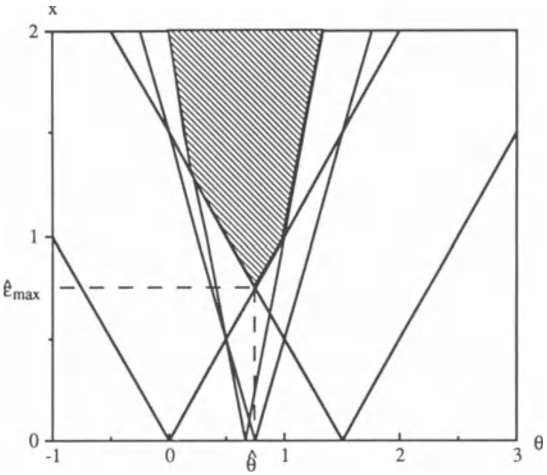


FIGURE 12.1. Geometrical interpretation of Overton's reformulation.

are not recursive and cannot be used for real-time minimax estimation, so that recursive variants would have to be used.[25,26] They would not, however, provide a description of the set of all solutions when this set is not a singleton, a situation that may be encountered even for the minimax estimate. This is why we suggest using a non-pivoting method[8] derived from the double-description method of Motzkin *et al.*[27] The next paragraph presents this algorithm in the context of recursive minimax estimation.

12.3. EXACT CONE UPDATING METHOD

The parameter vector $\boldsymbol{\theta}$ and x must satisfy Eqs. (12.6) and (12.7(a) and (b)). A classical approach[27] to solve sets of inhomogeneous inequalities such as Eqs. (12.7a) and (12.7b) is to convert them into sets of homogeneous inequalities by introducing a new variable v^h. This modification amounts to transforming a convex polyhedron in a $(p + 1)$-dimensional space into a polyhedral cone in a $(p + 2)$-dimensional space

$$x^h - \boldsymbol{\phi}^{\mathrm{T}}(k)\, \boldsymbol{\theta}^h + y(k)v^h \geq 0, k = 1, \ldots, N, \tag{12.9a}$$

$$x^h + \boldsymbol{\phi}^{\mathrm{T}}(k)\, \boldsymbol{\theta}^h - y(k)v^h \geq 0, k = 1, \ldots, N. \tag{12.9b}$$

where $v^h > 0$. This set of inequalities can be written in matrix form as

$$\mathbf{Aw} \geq \mathbf{0}, \tag{12.10}$$

where $\mathbf{A}$ is a $(2N, p + 2)$ matrix and $\mathbf{w} = (x^h, \boldsymbol{\theta}^{h\mathrm{T}}, v^h)^{\mathrm{T}}$. Here and in what follows, vector inequalities are to be understood componentwise. The solution set is then the intersection of $2N$ half-spaces of the form $\mathbf{a}_i^{\mathrm{T}}\mathbf{w} \geq 0$, where $\mathbf{a}_i^{\mathrm{T}}$ is the ith row of the matrix $\mathbf{A}$. The hyperplane $\mathbf{a}_i^{\mathrm{T}}\mathbf{w} = 0$ associated with each half-space is a supporting hyperplane of the set if it is associated with a non-redundant constraint. Each d-face of the set is defined as the intersection of $(p + 2 - d)$ supporting hyperplanes. Edges are 1-faces and vertices are 0-faces. The solution set of Eq. (12.10) is a polyhedral cone which can be fully described by enumerating its edges and supporting hyperplanes. When the set of all $\mathbf{w}$ solutions of Eq. (12.10) has been determined, the solution set of Eqs. (12.7(a) and (b)) is obtained as a set of vertices and edges. The vertices are obtained by

$$x = w_1/w_{p+2} = x^h/v^h, \tag{12.11a}$$

$$\theta_k = w_{k+1}/w_{p+2} = \theta_k^h/v^h, \quad k = 1, \ldots, p, \tag{12.11b}$$

for any $w_{p+2} > 0$.

Let $\mathbf{a}_j^{\mathrm{T}}$ be the jth row of $\mathbf{A}$, C_j be the polyhedral cone associated with the first j inequalities of Eq. (12.10), and $\mathbf{S}_j$ be a matrix the columns of which are the direction vectors of the edges of C_j. The algorithm is as follows:

Initialization: The method requires that

$$\mathbf{w} \geq \mathbf{0}. \tag{12.12}$$

From Eq. (12.11) and the meaning of x and v^h, this corresponds to $\boldsymbol{\theta} \geq \mathbf{0}$. For the presentation of the algorithm, we shall therefore assume that $\boldsymbol{\theta} \geq \mathbf{0}$, but we shall see later how this assumption can be avoided.

Equation (12.12) defines C_0 as the non-negative orthant. A description of this cone consists of the matrix $\mathbf{S}_0$ of the direction vectors of its edges, and for each edge of the list of all hyperplanes to which it belongs and the list of all other edges of the cone which are adjacent to it. In C_0, the ith edge admits as supporting hyperplanes the $(p + 1)$ hyperplanes of the form $w_j = 0, j \neq i$, and is adjacent with any of the remaining $(p + 1)$ edges of C_0.

Iteration: Suppose that the first $(j - 1)$ inequalities have been processed so that $\mathbf{S}_{j-1}$ and the associated lists of supporting hyperplanes and adjacent edges are available.

The new inequality $\mathbf{a}_j^{\mathrm{T}}\mathbf{w} \geq 0$ defines a hyperplane $H_j = \{\mathbf{w}|\mathbf{a}_j^{\mathrm{T}}\mathbf{w} = 0\}$ and two half-spaces $H_j^+ = \{\mathbf{w}|\mathbf{a}_j^{\mathrm{T}}\mathbf{w} \geq 0\}$ and $H_j^- = \{\mathbf{w}|\mathbf{a}_j^{\mathrm{T}}\mathbf{w} \leq 0\}$. The solution set of the j inequalities is the intersection of C_{j-1} and H_j^+ (in a $(p + 2)$-dimensional space). If $\mathbf{z}^{\mathrm{T}} = \mathbf{a}_j^{\mathrm{T}}S_{j-1}$, the sign of the ith component of $\mathbf{z}$ indicates whether or not the ith edge of C_{j-1} belongs to H_j^+. Three patterns have to be considered:

1. $\mathbf{z} < \mathbf{0}$. No edge of C_{j-1} belongs to the intersection. The solution set for the homogeneous system of Eq. (12.10) reduces to the singleton $\{\mathbf{0}\}$. The original system, Eq. (12.8), has no solution. This case should not take place in the context of minimax estimation.
2. $\mathbf{z} \geq \mathbf{0}$. C_{j-1} is included within H_j^+. No updating is necessary. The inequality is redundant. Set $C_j = C_{j-1}$ and introduce the next inequality.
3. The components of $\mathbf{z}$ are of different signs. H_j separates the edges of C_{j-1}. The inequality is not redundant and the description of the cone must be updated.

When Pattern 3 occurs, $\mathbf{S}_j$ differs from $\mathbf{S}_{j-1}$. All columns of $\mathbf{S}_{j-1}$ that correspond to a non-negative component of $\mathbf{z}$ are kept. If any component of $\mathbf{z}$ is equal to zero, the list of the supporting hyperplanes of the associated edge must be completed by introducing H_j. New edges must be determined, which are located on the faces of C_{j-1} intersected by H_j. These faces are associated with two adjacent edges of C_{j-1}, one giving a positive component of $\mathbf{z}$, denoted by $\mathbf{s}_i^+$, and the other one giving a negative component of $\mathbf{z}$, denoted by $\mathbf{s}_k^-$. The new edge $\mathbf{s}_{ik}^j$ is a linear combination of these two edges and must belong to H_j. The suitable linear combination is then

$$\mathbf{s}_{ik}^j = -z_k\mathbf{s}_i^+ + z_i\mathbf{s}_k^-. \tag{12.13}$$

The list of supporting hyperplanes associated with this new edge is determined by keeping only the supporting hyperplanes common to $\mathbf{s}_i^+$ and $\mathbf{s}_k^-$, and by including H_j. When all new edges with their list of supporting hyperplanes have been computed, the list of adjacent edges must be updated for all edges of C_j. Two edges are adjacent if and only if their lists of supporting hyperplanes have at least p hyperplanes in common and no other edge admits these p hyperplanes as supporting hyperplanes.

The polyhedral cone C_j is then obtained in the form of its matrix of edges $\mathbf{S}_j$ and lists of adjacency and supporting hyperplanes.

Using Eqs. (12.11(a) and (b)), vertices of the solution set of Eqs. (12.7(a) and (b)) can be obtained. The solution of the problem defined by Eqs. (12.6) and (12.7(a) and (b)) is determined by looking for the minimal value of x over the set of vertices and finding the associated value(s) of $\boldsymbol{\theta}$.

REMARKS: The initialization of the method requires that $\boldsymbol{\theta} \geq \mathbf{0}$. If a lower bound $\boldsymbol{\theta}_{\min}$ for $\boldsymbol{\theta}$ is known, it is possible to perform a transformation of the parameter vector $\boldsymbol{\theta}' = \boldsymbol{\theta} - \boldsymbol{\theta}_{\min}$ so that $\boldsymbol{\theta}'$ satisfies $\boldsymbol{\theta}' \geq \mathbf{0}$. If no lower bound is available for $\boldsymbol{\theta}$, the change of variable

$$\theta_i = v_i - v_i', i = 1, \ldots, p, \tag{12.14}$$

where v_i and v_i' are both positive, allows the method to be used at the cost of doubling the number of unknowns.

The description obtained for the solution set of Eq. (12.8) contains as a by-product the set $\mathbb{S}$ associated with any given value of $\varepsilon_{\max} \geq \hat{\varepsilon}_{\max}$. Obtaining $\mathbb{S}$ only requires taking the additional constraint $x \geq \varepsilon_{\max}$ into account.

Example 2: Consider an *ARX* system described by

$$y(k+1) = -a_1^* y(k) - a_2^* y(k-1) + b_1^* u(k) + b_2^* u(k-1) + \varepsilon(k+1). \qquad (12.15)$$

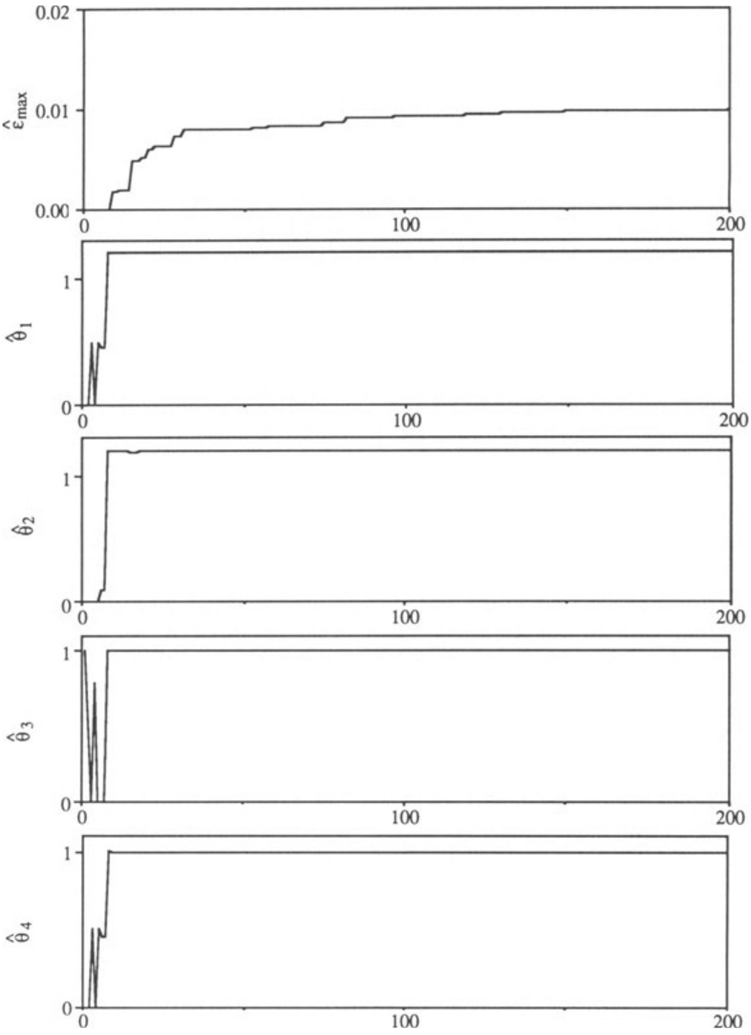


FIGURE 12.2. Evolution of the minimax estimates for Example 2 as a function of the number of constraints taken into account.

One hundred data points have been simulated according to Eq. (12.15), with $a_1^* = 1.2, a_2^* = 1.2$ (so that the system is unstable), $b_1^* = b_2^* = 1$, $y(0) = y(1) = 0$, $u(k) = (1 + (-1)^{(k-1)})/2$. Each $\varepsilon(k)$ was generated according to a uniform distribution in [–0.01, 0.01], so that the true value of $\varepsilon_{\max}$ is $\varepsilon_{\max}^* = 0.01$. The following model was used to fit the data

$$\boldsymbol{\phi}^{\mathrm{T}}(k+1)\boldsymbol{\theta} = -a_1 y(k) - a_2 y(k-1) + b_1 u(k) + b_2 u(k-1), \tag{12.16}$$

where $\boldsymbol{\theta} = (a_1, a_2, b_1, b_2)^{\mathrm{T}}$. Fig. 12.2 illustrates the evolution of $\hat{\boldsymbol{\theta}}$ and $\hat{\varepsilon}_{\max}$ with the number of constraints taken into account.

12.4. EXTENSION TO OUTPUT-ERROR MODELS

The method described so far can only handle models linear in their parameters, which is restrictive and is not in particular the case for output-error models. To extend the approach to such models, we shall use the method proposed in Ref. 11, which makes it possible to deduce affine inequalities from those defining the feasible parameter set associated with an output-error model of the form

$$y_m(k, \boldsymbol{\theta}) = -\sum_{j=1}^{n_a} a_j y_m(k-j, \boldsymbol{\theta}) + \sum_{j=1}^{n_b} b_j u(k-j), \tag{12.17}$$

with

$$y(k) = y_m(k, \boldsymbol{\theta}) + \varepsilon(k, \boldsymbol{\theta}), \quad k = 1, \ldots, \mathrm{N}, \tag{12.18}$$

where $\boldsymbol{\theta} = (a_1, \ldots, a_{n_a}, b_1, \ldots, b_{n_b})^{\mathrm{T}}$ and where the initial conditions $y_m(k, \boldsymbol{\theta})$ $(k = 0, \ldots, 1 - n_a)$ are assumed to be known. One can write

$$y(k) = \boldsymbol{\phi}^{\mathrm{T}}(k)\boldsymbol{\theta} + \varepsilon(k, \boldsymbol{\theta}), \tag{12.19}$$

where $\varepsilon(k, \boldsymbol{\theta})$ is the output error, assumed here to satisfy $|\varepsilon(k, \boldsymbol{\theta})| \leq \varepsilon_{\max}$. The first n_a terms of $\boldsymbol{\phi}(k)$ are unknown but bounded. This corresponds to an "errors-in-variables problem."(28) Each new observation $y(k)$ yields a pair of piecewise-linear bounds on $\boldsymbol{\theta}$, because each change of sign of an autoregressive parameter θ_j $(j = 1, \ldots, n_a)$ changes the bound used to replace $y_m(k-j, \boldsymbol{\theta})$ in the regressor.(10) Any $\boldsymbol{\theta}$ belonging to $\mathbb{S}$ satisfies the following (necessary but not sufficient) inequalities(11)

$$\sum_{j=1}^{n_a} \theta_j[-y(k-j) - \mathrm{sgn}(\theta_j)\varepsilon_{\max}] + \sum_{j=1}^{n_b} \theta_{n_a+j} u(k-j) \leq y(k) + \varepsilon_{\max}, \tag{12.20}$$

and

$$\sum_{j=1}^{n_a} \theta_j[-y(k-j) + \mathrm{sgn}(\theta_j)\varepsilon_{\max}] + \sum_{j=1}^{n_b} \theta_{n_a+j}u(k-j) \geq y(k) - \varepsilon_{\max}, \tag{12.21}$$

for $k = 1, \ldots, N$. Since $\varepsilon_{\max}$ is unknown, these inequalities are nonlinear in $(\boldsymbol{\theta}, \varepsilon_{\max})$. Therefore we suggest replacing $\varepsilon_{\max}$ in the left-hand side of Eqs. (12.20) and (12.21) by the most recently available $\hat{\varepsilon}_{\max}$. The corresponding inequalities can then be written as

$$\varepsilon_{\max} + \sum_{j=1}^{n_a} \theta_j[y(k-j) + \mathrm{sgn}(\theta_j)\hat{\varepsilon}_{\max}(k-1)]$$
$$- \sum_{j=1}^{n_b} \theta_{n_a+j}u(k-j) + y(k) \geq 0, \tag{12.22}$$

$$\varepsilon_{\max} + \sum_{j=1}^{n_a} \theta_j[-y(k-j) + \mathrm{sgn}(\theta_j)\hat{\varepsilon}_{\max}(k-1)]$$
$$+ \sum_{j=1}^{n_b} \theta_{n_a+j}u(k-j) - y(k) \geq 0. \tag{12.23}$$

If the signs of all autoregressive parameters $\theta_j (j = 1, \ldots, n_a)$ are known *a priori*, then Eqs. (12.22 and 12.23) are linear in $\boldsymbol{\theta}$ and $\varepsilon_{\max}$. It is therefore possible with the exact cone updating technique to recursively obtain $\hat{\boldsymbol{\theta}}$ and $\hat{\varepsilon}_{\max}$. If the signs of some of the autoregressive parameters are not known, all possible combinations of signs have to be investigated.

REMARK: $\hat{\boldsymbol{\theta}}$ and $\hat{\varepsilon}_{\max}$ are no longer an exact minimax solution, and $\hat{\boldsymbol{\theta}}$ may not belong to $\mathbb{S}$ because of the approximation involved in the transformation of the nonlinear inequalities into linear ones. $\hat{\boldsymbol{\theta}}$ may nevertheless correspond to a good point estimate of $\boldsymbol{\theta}$, as evidenced by the following example.

EXAMPLE 3: Consider an output-error system described by

$$y_m(k, \boldsymbol{\theta}^*) = -a_1^* y_m(k-1, \boldsymbol{\theta}^*) - a_2^* y_m(k-2, \boldsymbol{\theta}^*) + b_1^* u(k-1) + b_2^* u(k-2),$$
$$y(k, \boldsymbol{\theta}^*) = y_m(k) + \varepsilon(k). \tag{12.24}$$

One hundred data points have been simulated according to Eq. (12.24) with $a_1^* = 1, a_2^* = 1, b_1^* = 1, b_2^* = 1, y_m(0) = y_m(1) = 0, u(k) = (1 + (-1)^{(k-1)})/2$. Each $\varepsilon(k)$ was generated according to a uniform distribution in $[-0.02, 0.02]$. Fig. 12.3

illustrates the evolution of $\hat{\theta}$ and $\hat{\varepsilon}_{\max}$ with the number of constraints taken into account. Although $\hat{\varepsilon}_{\max}$ remains very optimistic, the estimated values of the parameters are very close to the true values.

Remark: In Example 3, for $(\hat{\theta}, \varepsilon_{\max}) = (\hat{\theta}, \hat{\varepsilon}_{\max})$, 97% of the pairs of inequalities associated with Eq. (12.20) and (12.21) but only 11% of the pairs of inequalities associated with Eqs. (12.17) and (12.18) are satisfied. This suggests

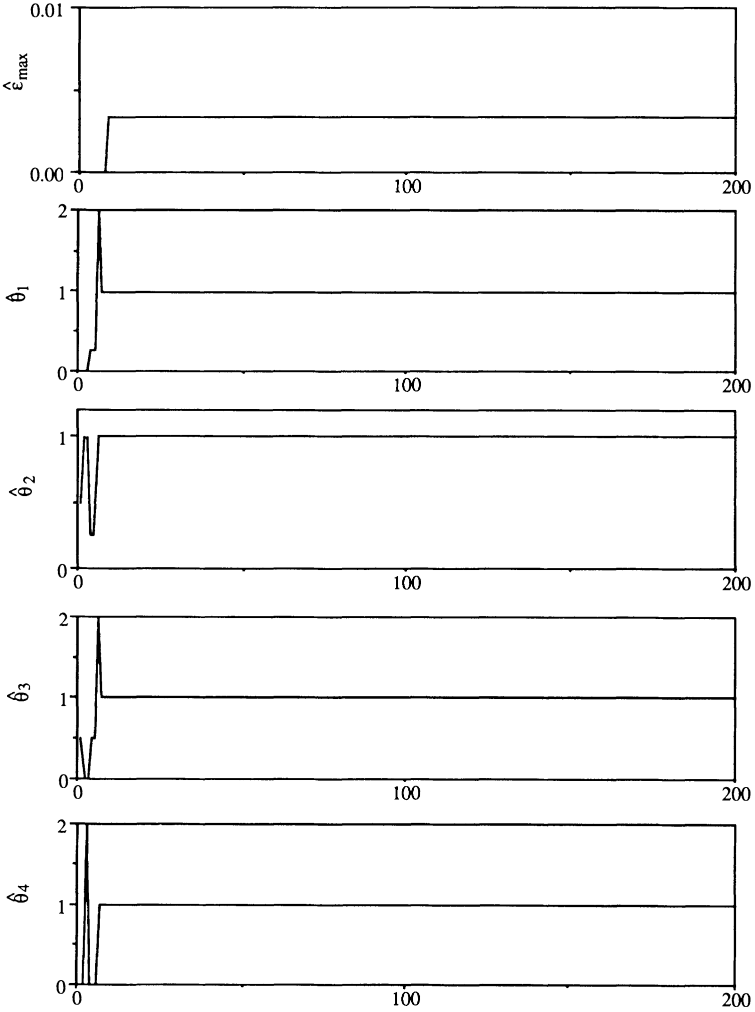


FIGURE 12.3. Evolution of the minimax estimates for Example 3 as a function of the number of constraints taken into account.

a nonrecursive method for correcting $\hat{\varepsilon}_{\max}$. Assume that $|\varepsilon(k)|$ is uniformly distributed between 0 and $\varepsilon^*_{\max}$. The percentage α of the inequalities associated with Eqs. (12.17 and 12.18) such that $|y(k) - y_m(k, \boldsymbol{\theta}^*)| \leq \hat{\varepsilon}_{\max}$ then satisfies

$$\alpha \rightarrow \frac{\hat{\varepsilon}_{\max}}{\varepsilon^*_{\max}} \text{ when } n \rightarrow \infty.$$

An estimate $\hat{\alpha}$ of α is thus given by the percentage of the inequalities associated with Eqs. (12.17 and 12.18) such that

$$|y(k) - y_m(k, \hat{\boldsymbol{\theta}})| \leq \hat{\varepsilon}_{\max}.$$

For Example 3, $\hat{\alpha} = 11\%$, and a corrected value for $\hat{\varepsilon}_{\max}$ is $\hat{\hat{\varepsilon}}_{\max} = \hat{\varepsilon}_{\max}/\hat{\alpha} \approx 0.03$, much closer to the true value. Another possibility worth investigating would be to adjust $\hat{\varepsilon}_{\max}$ upward until the inequalities deriving from Eqs. (12.17) and (12.18) are all satisfied.

12.5. DETECTION OF OUTLIERS

Consider a situation where a large number of data points are associated with an error satisfying $|\varepsilon| \leq \varepsilon^*_{\max}$ but where a few data points are associated with a very much larger error, because of some failure in the procedure for data collection. The value of $\hat{\varepsilon}_{\max}$ can only increase or remain unchanged when a new data point is taken into account. Therefore, if one uses the algorithm presented in Section 12.3, $\hat{\varepsilon}_{\max}$ is then much larger than what would have been obtained had the data been correctly collected. One may then wish to identify the data points associated with exceptionally large errors as outliers, to discard them. If the regressor does not contain past output values, discarding an outlier merely requires ignoring the two inequalities associated with it. On the other hand, if the regressor depends on past values of the output, discarding an outlier requires ignoring not only the two inequalities where it appears as a measurement value y, but also all other inequalities in which it appears as a coefficient of the regressor.

The problem can be viewed as one of fault detection, for which a number of methods have been proposed.(29) The method currently implemented in the algorithm is based on mean-value testing.(30) For each new constraint, the new value of the minimax bound on the error is determined and compared to the mean value of the minimax bounds obtained from the previous constraints. If the difference between the mean and the new value is higher than a given threshold, the corresponding data is considered as an outlier and rejected, so that the cone is not modified.

EXAMPLE 4: One hundred data points have been simulated according to Eq. (12.15) with $a^*_1 = 1$, $a^*_2 = 1$, $b^*_1 = 1$, $b^*_2 = 1$, $y(0) = y(1) = 0$, $u(k) = (1 + (-1)^{(k-1)})/2$.

Each $\varepsilon(k)$ was generated according to a uniform distribution in [–0.01, 0.01]. For three data points, corresponding to $k = 25$, 50 and 70, (constraints # 50, 100 and 140) the value of $y(k)$ was replaced by an outlier, obtained by adding 25 to the value. The resulting value of $y(k)$ was recorded for the data points but not used in computing $y(k + 1)$ and $y(k + 2)$ (this corresponds to an error in records, not a jump in the system). The model Eq. (12.16) was used to fit the data. Fig. 12.4 illustrates

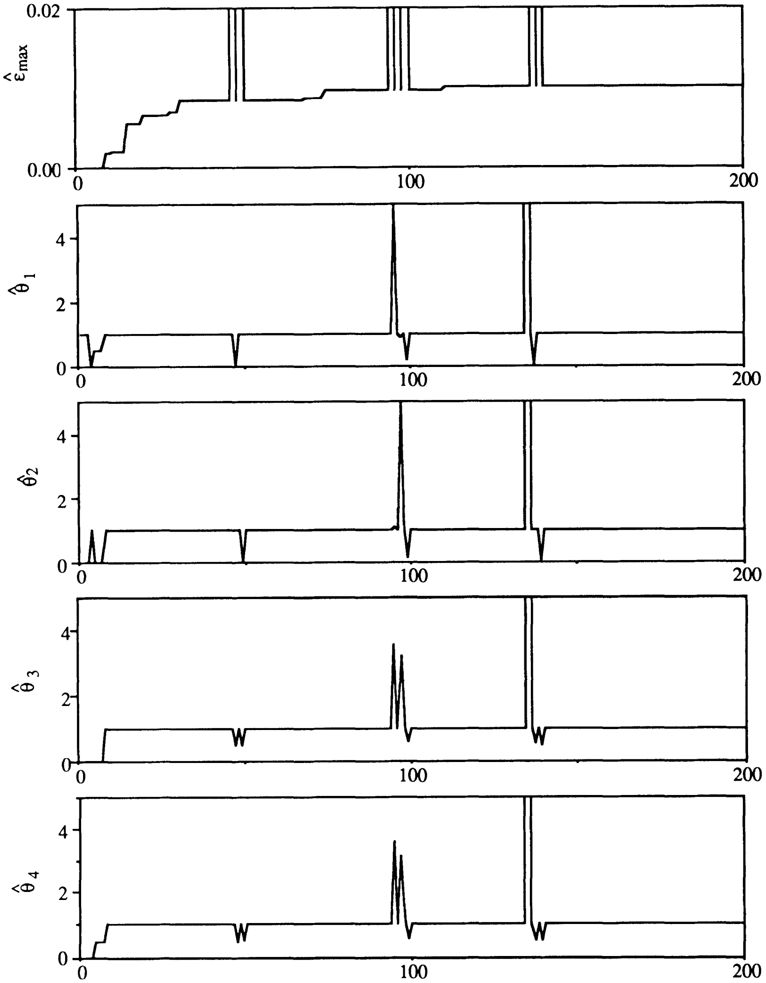


FIGURE 12.4. Evolution of the minimax estimates for Example 4 as a function of the number of constraints taken into account.

the evolution of $\hat{\boldsymbol{\theta}}$ and $\hat{\varepsilon}_{\max}$ with the number of constraints taken into account. The values indicated in Fig. 12.4 are those given by the algorithm, even when an outlier has been detected. Hence the jumps in the estimates. In practice, of course, one would then keep the previous estimates, so that these jumps would not occur.

The method is easy to implement and requires very little additional computation at each iteration. However, the determination of a suitable threshold may be critical, as too small a value would lead to an underestimation of the minimax bound by rejection of regular data, whereas too large a value would make the test totally useless. When the number of data points taken into account increases, the probability of a large increase of $\varepsilon_{\max}$ decreases, so that adapting the threshold seems of interest.

12.6. CONCLUSIONS

A method has been presented that provides the minimal value of the bound on the error that ensures the nonemptiness of the feasible parameter set. By a suitable reparametrization of the problem, this value can be obtained from the description of the (unbounded) solution set of a system of linear inequalities. A method has been described which recursively updates an exact description of this set whenever a new datum is taken into account and provides a minimax estimate of the parameters and bound for the error. This description contains the exact description of any feasible set associated with a larger bound on the error as a by-product. The method was initially designed for the study of models linear in their parameters but can be extended to output-error models. In this case, the estimated value of the minimax bound for the error is a lower bound for the true bound. However, the resulting parameter estimates prove to remain very close to the true values. The value of the minimax bound for the error can only increase or remain unchanged when a new data point is taken into account. If the data set contains outliers, i.e., data associated with a very much larger error than the rest of the set because of some failure in the procedure for data collection, the estimated minimax bound for the error would be drastically increased because of these data. This is why a simple procedure has been suggested to protect one against such occurrences.

REFERENCES

1. E. Walter and H. Piet-Lahanier, *Math. Comput. Simul.* **32**, 449 (1990).
2. E. Fogel and Y. F. Huang, *Automatica* **18**, 229 (1982).
3. L. Pronzato, E. Walter, and H. Piet-Lahanier, in: *Proceedings of the 28th IEEE Conference on Decision and Control*, Tampa, FL, pp. 1952–1955 (1989).
4. G. Belforte, B. Bona, and V. Cerone, *Automatica* **26**, 887 (1990).
5. M. Milanese and G. Belforte, *IEEE Trans. Autom. Control* **AC-27**, 408 (1982).

6. V. Broman and M. J. Shensa, in: *Proceedings of the 25th IEEE Conference on Decision and Control*, Athens, Greece, pp. 1749–1752 (1986).
7. S. H. Mo and J. P. Norton, *Math. Comput. Simul.* **32**, 481 (1990).
8. E. Walter and H. Piet-Lahanier, *IEEE Trans. Autom. Control* **AC-34**, 911 (1989).
9. G. Belforte and M. Milanese, in: *Proceedings of the 1st IASTED International Symposium on Modeling Identification and Control*, Davos, Switzerland, pp. 75–79 (1981).
10. J. P. Norton, *Int. J. Control* **45**, 375 (1987).
11. T. Clement and S. Gentil, *Math Comput. Simul.* **30**, 257 (1988).
12. M. Milanese and A. Vicino, in: *System Modelling and Simulation* (S. Tzafestas *et al.* eds.) Elsevier, Amsterdam, The Netherlands, pp. 91–96 (1989).
13. K. Keesman, *Math. Comput. Simul.* **32**, 535 (1990).
14. H. Piet-Lahanier and E. Walter, *Math. Comput. Simul.* **32**, 553 (1990).
15. P. S. Laplace, in: *Mémoires de l'Académie des Sciences de Paris*, pp. 1–87 (1793); *Oeuvres*, Vol. 11, (Reprint) Gauthier-Villars, Paris, pp. 447–558 (1895).
16. J. B. Fourier, in: *Histoire de l'Académie pour 1824*, pp. 325–328 (1824); *Oeuvres* (Reprint) Gauthier-Villars, Paris (1890).
17. A. L. Cauchy, *Journal de l'École Polytechnique* **13**, 175 (1831); *Oeuvres* (Reprint) Series 2, Vol. 1, Gauthier-Villars, Paris (1905).
18. R. W. Farebrother, in: *Statistical Data Analysis Based on the L_1-Norm* (Y. Dodge, ed.) Elsevier, Amsterdam, The Netherlands, pp. 37–63 (1987).
19. A. Ralston, *A First Course in Numerical Analysis*, McGraw-Hill, New York (1965).
20. E. Walter and H. Piet-Lahanier, in: *Prep. 9th IFAC/IFORS Symposium on Identification and System Parameter Estimation*, Budapest, Hungary, pp. 763–768 (1991).
21. M. L. Overton, in: *Nonlinear Optimization 1981* (M. J. D. Powell, ed.) Academic Press, New York (1982).
22. A. van den Bos, in: *Prep. 7th IFAC/IFORS Symposium on Identification and System Parameter Estimation*, pp. 173–177 (1985).
23. G. Dantzig, *Linear Programming and Extensions*, Princeton University Press, Princeton, NJ (1963).
24. N. Karmarkar, *Combinatorica* **4**, 373 (1984).
25. G. Dantzig and P. Wolfe, *Econometrica* **29**, 767 (1961).
26. G. Goffin and J. P. Vial, *J. of Optim. Theory and Applic.* **65**, 409 (1990).
27. T. S. Motzkin, H. Raiffa, G. L. Thompson, and R. M. Thrall, in: *Contributions to the Theory of Games*, Vol. 2, *Annals of Mathematics Study* **28**, (H. W. Kuhn and A. W. Tucker, eds.) Princeton University Press, Princeton, NJ, pp. 51–73 (1953).
28. B. D. O. Anderson, *Automatica* **21**, 709 (1985).
29. M. Basseville, *Automatica* **24**, 309 (1988).
30. A. Willsky, *IEEE Trans. Autom. Control* **AC-21**, 108 (1976).

13

Robustness to Outliers of Bounded-Error Estimators and Consequences on Experiment Design

L. Pronzato and É. Walter

ABSTRACT

If proper precautions are not taken, bounded-error estimators are not robust to outliers, i.e., to data points where the actual error is larger than assumed when specifying the error bounds. The outlier minimal number estimator (OMNE) has been designed to overcome this difficulty and has proved on various examples to be particularly insensitive to outliers. This chapter is devoted to a theoretical study of its robustness. The notion of breakdown point, introduced to quantify the robustness of point estimators, is extended to set-estimators. When the model output is linear in the parameters, OMNE is shown to possess the highest achievable breakdown point. A bound on the bias due to outliers is established and used to define a new policy for optimal experimental design aimed at providing a higher protection against outliers than conventional D-optimal design.

L. PRONZATO • Laboratoire I3S, CNRS URA-1376, Sophia Antipolis, 06560 Valbonne, France.
É. WALTER • Laboratoire des Signaux et Systèmes, CNRS-École Supérieure d'Electricité, 91192 Gif-sur-Yvette Cedex, France.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

13.1. INTRODUCTION

The purpose of robust estimation[1] is to provide estimates that are not dramatically affected if the hypotheses made on the measurement errors are not entirely satisfied, either because of a misspecification of the distribution or because of the presence of outliers. Least squares estimators are not robust to outliers, to the point where a single erroneous datum can ruin the estimate obtained from a large set of otherwise regular data. The notion of breakdown point, introduced in the context of point estimation,[2] is useful to quantify robustness and to compare the performances of estimators. Loosely speaking, the breakdown point of an estimator is the minimum percentage of outliers that must be introduced in a data set for the estimator to produce a meaningless result. In this chapter, this notion is extended to set estimators such as those encountered in the context of bounded-error estimation,[3,4,5,6] which is recalled in Section 13.2. The aim of bounded-error estimation is to characterize the set of all parameter vectors such that the residuals lie between some prior bounds. In this context, outliers are any data points for which these bounds are too optimistic. Many bounded-error estimators are not robust, in the sense that a single outlier may make the set of possible values for the parameters empty. OMNE, however, has proved to be particularly insensitive to outliers.[7,8,9] When the model output is linear in the parameters, OMNE is shown in Section 13.3 to reach the highest possible breakdown point. A bound is given to the bias due to outliers, which suggests a new policy for optimal experiment design aimed at providing a high protection against outliers. This policy is described in Section 13.4, and compared on an illustrative example to conventional D-optimal design.

13.2. BOUNDED-ERROR ESTIMATION

Given a n-sample $\mathcal{Z}$ of data points $(\mathbf{x}_i, y_i)$, $i = 1, \ldots, n$, where y_i denotes the measurement obtained under the ith experimental conditions $\mathbf{x}_i$, and a model structure $\eta(\theta, \mathbf{x})$ with a p-dimensional parameter vector θ, bounded-error estimation aims at characterizing the set of all vectors θ such that all differences $y_i - \eta(\theta, \mathbf{x}_i)$ lie between some known bounds $-\varepsilon_i^m$ and ε_i^M. This posterior feasible parameter set[10] (or membership set[11]), denoted in what follows by $\mathcal{S}$, is then given by

$$\mathcal{S}(\mathcal{Z}) = \{\theta \in \mathbb{R}^p \mid -\varepsilon_i^m \le y_i - \eta(\theta, \mathbf{x}_i) \le \varepsilon_i^M, i = 1, \ldots, n\}. \tag{13.1}$$

As in classical point estimation, the observations y_i can be assumed to correspond to the model response $\eta(\theta^*, \mathbf{x}_i)$ obtained at some unknown true value θ^* of the parameters, corrupted by some unknown errors b_i,

$$y_i = \eta(\theta^*, \mathbf{x}_i) + b_i, \; i = 1, \ldots, n.$$

If the errors b_i are only known to satisfy

$$-\varepsilon_i^m \leq b_i \leq \varepsilon_i^M, \quad i = 1, \ldots, n, \tag{13.2}$$

any θ in $\mathcal{S}(\mathcal{Z})$ is a possible candidate to being the true value θ^*. Note that if b_i is assumed to be a random variable with a probability density function equal to zero when (and only when) Eq. (13.2) is not satisfied, then $\mathcal{S}(\mathcal{Z})$ corresponds to the set of all parameter vectors with a non-zero likelihood. For that reason, $\mathcal{S}(\mathcal{Z})$ has also been called posterior likelihood set. It must be emphasized that the definition of $\mathcal{S}$ in Eq. (13.1) does not suppose the existence of a true parameter vector θ^*. The structure of the model used in the definition of $\mathcal{S}$ can be quite different from that of the process generating the data, which allows simple model structures to be used to describe the behaviour of complex processes. In such a situation, the errors b_i may be essentially deterministic, so that the underlying assumptions of classical approaches for point-estimation such as maximum likelihood may no longer be valid. Note that $\mathcal{S}(\mathcal{Z})$ can also be written as

$$\mathcal{S}(\mathcal{Z}) = \{\theta \in \mathbb{R}^p \,|\, -1 \leq z_i - \frac{\eta(\theta, \mathbf{x}_i)}{\varepsilon_i} \leq 1, \quad i = 1, \ldots, n\},$$

where $\varepsilon_i = (\varepsilon_i^M + \varepsilon_i^m)/2$, and $z_i = y_i/\varepsilon_i + (\varepsilon_i^m - \varepsilon_i^M)/(2\varepsilon_i)$, so that we shall assume with no loss of generality that the bounds ε_i^m and ε_i^M are symmetrical and identical for all data points, i.e.,

$$\varepsilon_i^m = \varepsilon_i^M = \varepsilon, \quad i = 1, \ldots, n.$$

In what follows, the model output is assumed to be a linear function of θ, so that it can be written as

$$\eta(\theta, \mathbf{x}_i) = \mathbf{x}_i^T\theta, \; i = 1, \ldots, n,$$

or equivalently with a vector notation

$$\eta_X(\theta) = \mathbf{X}\theta,$$

where the ith row of $\mathbf{X}$ is equal to $\mathbf{x}_i^T$. The posterior feasible parameter set $S(Z)$ associated with the n measurements is then given by

$$\mathcal{S}(\mathcal{Z}) = \{\theta \in \mathbb{R}^p \,|\, -\varepsilon \leq y_i - \mathbf{x}_i^T\theta \leq \varepsilon, i = 1, \ldots, n\}. \tag{13.3}$$

When rank($\mathbf{X}$) $= p$, $\mathcal{S}(\mathcal{Z})$ is a convex polyhedron that can be given an exact recursive parametric description,(12) and an experimental design policy aimed at minimizing the volume of $\mathcal{S}(\mathcal{Z})$ has already been described.(13,14) When the inequalities $|\, y_i - \mathbf{x}_i^T\theta \,| \leq \varepsilon, i = 1, \ldots, n$, cannot be satisfied simultaneously, $\mathcal{S}(\mathcal{Z})$ is empty. This can be due to two different reasons: (i) the model structure is incorrect; and (ii) the data are corrupted by outliers, which should be rejected. We shall assume in what follows that we are in the second situation. The rejection policy, motivated by robustness regarding outliers, is described in the next section.

13.3. ROBUST PARAMETER BOUNDING

13.3.1. Outlier Minimal Number Estimator

Let $\mathcal{J}$ be a finite set of distinct indices, defined as follows

$$\mathcal{J} = \{i_j \in \mathbb{N} \mid i_j \le n,\ i_j \ne i_k \text{ if } j \ne k, j = 1, \dots, h,\ h \le n\}.$$

Let $\mathcal{S}_{\mathcal{J}}(\mathcal{Z})$ be the posterior feasible set associated with those data points $(\mathbf{x}_i, y_i)$ from a n-sample $\mathcal{Z}$ that are such that $i \in \mathcal{J}$. Define the set $\mathcal{S}^{\#h}(\mathcal{Z})$ as

$$\mathcal{S}^{\#h}(\mathcal{Z}) = \bigcup_{\#(\mathcal{J})=h} \mathcal{S}_{\mathcal{J}}(\mathcal{Z}), \tag{13.4}$$

where $\#(\mathcal{J})$ denotes the cardinal of $\mathcal{J}$. OMNE then corresponds to the set $\mathcal{S}^{\#h^*}(\mathcal{Z})$, with

$$h^*(\mathcal{Z}) = \arg\max\{h \mid \mathcal{S}^{\#h}(\mathcal{Z}) \ne \varnothing\}. \tag{13.5}$$

$\mathcal{S}^{\#h^*}(\mathcal{Z})$ is denoted by $\mathcal{S}^*(\mathcal{Z})$ in what follows. The set $\mathcal{S}^{*(}\mathcal{Z})$ thus corresponds to all values of the parameter vector θ that are consistent with the largest possible number of observations. Note that no attempt is made at pinpointing which bad items in $(y_i)_i$, and/or $(\mathbf{x}_i)_i$ have given rise to the outlier data points $(\mathbf{x}_j, y_j)$ (a non-trivial problem if the same item appears at more than one j, as in AR models[3]).

When $\mathcal{Z}$ consists of regular data points, which means that $\mathcal{S}$ as defined in Eq. (13.3) is not empty, $h^*(\mathcal{Z}) = n$, and $\mathcal{S}^*(\mathcal{Z}) = \mathcal{S}_{i}^{\#n}(\mathcal{Z}) = \mathcal{S}(\mathcal{Z})$. OMNE has proved on various examples to be particularly insensitive to numerous and severe outliers.[7,9] Its theoretical robustness properties will now be studied in more detail in terms of its breakdown point.

13.3.2. Breakdown Point

Consider a n-sample $\mathcal{Z}$ of regular data points ($\mathcal{S}(\mathcal{Z}) \ne \varnothing$), and a corrupted sample $\mathcal{Z}'$ obtained from $\mathcal{Z}$ by replacing m original points by arbitrary outliers. One wishes the optimal set $\mathcal{S}^*(\mathcal{Z}')$ to satisfy

$$\mathcal{S}^*(\mathcal{Z}') = \mathcal{S}^{\#n-m}(\mathcal{Z}') = \mathcal{S}_{\mathcal{J}_\cap}(\mathcal{Z}),$$

where the set $\mathcal{J}_\cap$ corresponds to the regular data kept in $\mathcal{Z}'$, i.e.

$$\mathcal{J}_\cap = \{i_j \in \mathbb{N} \mid (\mathbf{x}_{i_j}, y_{i_j}) \in \mathcal{Z} \cap \mathcal{Z}', j = 1, \dots, n-m\}.$$

This would correspond to the rejection of the m outliers and of no regular data. Note that in this case, if a true value θ^* can be defined for the model parameters, then θ^* belongs to $\mathcal{S}^*(\mathcal{Z}')$. However, less favorable situations can be encountered where $\mathcal{S}^*(\mathcal{Z}') \neq \mathcal{S}_{\mathcal{J}\cap}(\mathcal{Z})$, which corresponds to nonrejected outliers. There is then no reason for $\mathcal{S}^*(\mathcal{Z}')$ to contain θ^*. Practical experience indicates, however, that $\mathcal{S}^*(\mathcal{Z}')$ generally remains close to θ^*.[15] Intuitively, a maximal value m^* should exist for m, such that the distance $d[\mathcal{S}^*(\mathcal{Z}),\mathcal{S}^*(\mathcal{Z}')]$ between $\mathcal{S}^*(\mathcal{Z})$ and $\mathcal{S}^*(\mathcal{Z}')$ remains bounded when $m < m^*$. The ratio m^*/n corresponds to the notion of breakdown point of an estimator,[2] here extended to set estimators such as those encountered in parameter bounding. We allow here contaminated experimental conditions, i.e., the outliers may be due to errors in the $\mathbf{x}_i$s. The breakdown point of a point estimator without contaminated experimental conditions is considered in Ref. 16, with special attention to breakdown point maximizing experimental designs.

DEFINITION 13.1. The breakdown point of a set estimator $\hat{\mathcal{S}}$ associated with a regular n-sample $\mathcal{Z}$ is given by

$$m^*[\hat{\mathcal{S}}(\mathcal{Z})] = \min_{\mathcal{Z}'} \{\frac{m}{n} \mid d[\hat{\mathcal{S}}(\mathcal{Z}),\hat{\mathcal{S}}(\mathcal{Z}')] = \infty\},$$

with the convention

$$d(\hat{\mathcal{S}}(\mathcal{Z}),\varnothing) = \infty,$$

where $\mathcal{Z}'$ is a corrupted n-sample obtained from Z by replacing m original points by arbitrary outliers.

REMARK 13.1. The breakdown point of the posterior feasible set $\mathcal{S}(\mathcal{Z})$ as defined in Eq. (13.3) is $1/n$ since a single outlier can make $\mathcal{S}$ empty. $\mathcal{S}(\mathcal{Z})$ is therefore not robust to outliers.

To investigate the robustness of OMNE to outliers, we shall need the following definition.

DEFINITION 13.2. A set estimator $\hat{\mathcal{S}}$ is regression equivariant if it satisfies

$$\hat{\mathcal{S}}(\mathcal{Z}_2) = T_v[\hat{\mathcal{S}}(\mathcal{Z}_1)],$$

for any p-dimensional vector $\mathbf{v}$, where $\mathcal{Z}_1$ and $\mathcal{Z}_2$ are two data sets respectively defined by

$$\mathcal{Z}_1 = \{(\mathbf{x}_1,y_1), \ldots, (\mathbf{x}_n,y_n)\},\ \mathcal{Z}_2 = \{(\mathbf{x}_1,y_1 + \mathbf{x}_1^T\mathbf{v}), \ldots, (\mathbf{x}_n,y_n + \mathbf{x}_n^T\mathbf{v})\},$$

and where $T_v(.)$ is the translation associated with $\mathbf{v}$.

LEMMA 13.1. The posterior feasible set $\mathcal{S}$ is regression equivariant.

PROOF. We can write

$$\mathcal{S}(\mathcal{Z}_2) = \{\theta \in \mathbb{R}^p \mid -\varepsilon \leq y_i - \mathbf{x}_i^T(\theta - \mathbf{v}) \leq \varepsilon, i = 1, \ldots, n\}$$

$$= \{\theta \in \mathbb{R}^p \mid \theta - \mathbf{v} \in \mathcal{S}(\mathcal{Z}_1)\} = T_v(\mathcal{S}(\mathcal{Z}_1)). \qquad \square$$

COROLLARY 13.1. OMNE is regression equivariant.

PROOF: We use the same notation as in Lemma 1. OMNE for Z_2 can be written as

$$\begin{aligned} S^*(Z_2) &= \bigcup_{\#(\mathcal{J})=h^*(Z_2)} S_{\mathcal{J}}(Z_2) \\ &= \bigcup_{\#(\mathcal{J})=h^*(Z_2)} T_v[S_{\mathcal{J}}(Z_1)] \\ &= T_v[S^*(Z_1)], \end{aligned}$$

where h^* is defined as in Eq. (13.5). □

REMARK 13.2. The notions of scale equivariance and affine equivariance$^{(2)}$ can also be extended to set estimators, and OMNE can be shown to be scale equivariant (provided that the bounds are modified according to the same scale as the data) and affine equivariant.

The following theorem then extends to parameter bounding the results obtained by Rousseeuw and Leroy$^{(2)}$ in the context of robust point estimation.

THEOREM 1.

(i) The breakdown point of any regression-equivariant set estimator $\hat{S}$ associated with a n-sample Z satisfies

$$m^*[\hat{S}(Z)] \leq \frac{\lfloor \frac{n-p}{2} \rfloor + 1}{n}, \tag{13.6}$$

where $\lfloor x \rfloor$ stands for the largest integer less than or equal to x.

(ii) If the experimental conditions are chosen in such a way that any $p \times p$ submatrix of $\mathbf{X}$ has full rank, the breakdown point of OMNE satisfies

$$m^*[S^*(Z)] = \frac{\lfloor \frac{n-p}{2} \rfloor + 1}{n}, \tag{13.7}$$

PROOF. (i): Suppose that $m^*[\hat{S}(Z)] > (\lfloor (n-p)/2 \rfloor + 1)/n$. Any corrupted sample Z' deduced from Z by replacing $\lfloor (n-p)/2 \rfloor + 1$ points is then such that $d[\hat{S}(Z),\hat{S}(Z')] < \beta$, with β bounded. Such a sample Z' contains $q = n - \lfloor (n-p)/2 \rfloor - 1$ data points of Z. If $n-p$ is odd, then $2q-(p-1) = n$, otherwise $2q-(p-1) = n-1$. Anyway, $2q-(p-1) \leq n$. We construct two corrupted n-samples $Z'(\mathbf{v})$ and $Z'(-\mathbf{v})$ whose $2q-(\mathrm{p}-1)$ first points are respectively defined by

$$(\mathbf{x}_1,y_1), \ldots, (\mathbf{x}_{p-1},y_{p-1}), (\mathbf{x}_p,y_p), \ldots, (\mathbf{x}_q,y_q),(\mathbf{x}_p,y_p + \mathbf{x}_p^T\alpha\mathbf{v}), \ldots, (\mathbf{x}_q,y_q + \mathbf{x}_q^T\alpha\mathbf{v}),$$

and

$$(\mathbf{x}_1,y_1),\ldots,(\mathbf{x}_{p-1},y_{p-1}),(\mathbf{x}_p,y_p),\ldots,(\mathbf{x}_q,y_q),(\mathbf{x}_p,y_p-\mathbf{x}_p^T\alpha\mathbf{v}),\ldots,(\mathbf{x}_q,y_q-\mathbf{x}_q^T\alpha\mathbf{v}),$$

with $\alpha \in \mathbb{R}$, $\mathbf{v} \neq \mathbf{0}$ and

$$\mathbf{x}_i^T\mathbf{v}=0,\ i=1,\ldots,p-1. \tag{13.8}$$

If $n-p$ is even, the nth data points of $\mathcal{Z}'(\mathbf{v})$ and $\mathcal{Z}'(-\mathbf{v})$ are still free. Let $(\mathbf{x}_n,y_n)$ be the nth datum of $\mathcal{Z}'(\mathbf{v})$, the nth datum of $\mathcal{Z}'(-\mathbf{v})$ is chosen as $(\mathbf{x}_n,y_n-\mathbf{x}_n^T\alpha\mathbf{v})$. $\mathcal{Z}'(\mathbf{v})$ and $\mathcal{Z}'(-\mathbf{v})$ both contain q points of $\mathcal{Z}$, so that

$$d\{\hat{\mathcal{S}}(\mathcal{Z}),\hat{\mathcal{S}}[\mathcal{Z}'(\mathbf{v})]\}<\beta', \tag{13.9}$$

$$d\{\hat{\mathcal{S}}(\mathcal{Z}),\hat{\mathcal{S}}[\mathcal{Z}'(-\mathbf{v})]\}<\beta''. \tag{13.10}$$

Taking Eq. (13.8) into account, one can easily check that $\mathcal{Z}'(-\mathbf{v})$ can also be deduced from $\mathcal{Z}'(\mathbf{v})$ (up to a reindexation of the elements, see Fig. 13.1) by replacing each datum $(\mathbf{x}_i,y_i)$ by $(\mathbf{x}_i,y_i-\mathbf{x}_i^T\alpha\mathbf{v})$. The regression equivariance of $\hat{\mathcal{S}}$ then implies that $\hat{\mathcal{S}}(\mathcal{Z}'(-\mathbf{v}))=T_{-\alpha\mathbf{v}}\{\hat{\mathcal{S}}[\mathcal{Z}'(\mathbf{v})]\}$, which contradicts Eqs. (13.9) and (13.10) for values of α large enough.

(ii): From Corollary 13.1, the breakdown point of $\mathcal{S}^*$ satisfies Eq. (13.6). Let us prove that the bound is reached. Suppose that $m=\lfloor(n-p)/2\rfloor$ points of $\mathcal{Z}$ are replaced to give a modified sample $\mathcal{Z}'$. $\mathcal{S}^*(\mathcal{Z}')=\mathcal{S}^{\#h^*}(\mathcal{Z}')$, with h^* defined as in Eq. (13.5). One obviously has $h^*(\mathcal{Z}')\geq n-m$. Consider then one of the sets $\mathcal{S}_{\mathcal{J}}(\mathcal{Z}')$, with $\#(\mathcal{J})=h^*(\mathcal{Z}')$, and denote it by $\tilde{\mathcal{S}}(\mathcal{Z}')$. Let $\mathcal{G}$ be the set of regular data points that contribute to defining both $\mathcal{S}^*(\mathcal{Z})$ and $\tilde{\mathcal{S}}(\mathcal{Z}')$. One has

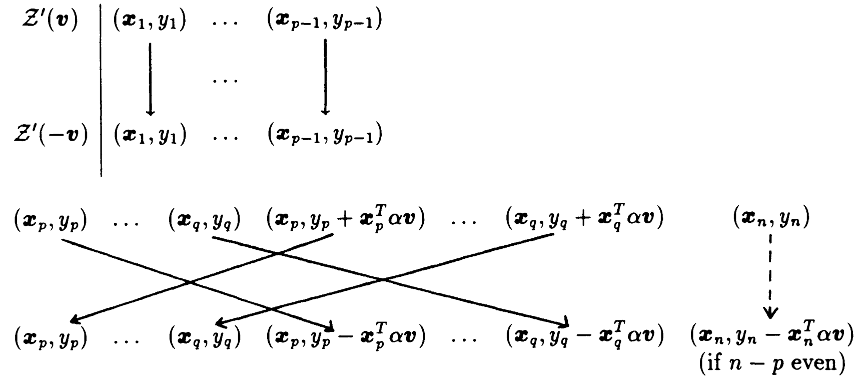

$\mathcal{Z}'(\boldsymbol{v})$
$(\boldsymbol{x}_1,y_1)$ ... $(\boldsymbol{x}_{p-1},y_{p-1})$
...
$\mathcal{Z}'(-\boldsymbol{v})$
$(\boldsymbol{x}_1,y_1)$ ... $(\boldsymbol{x}_{p-1},y_{p-1})$
$(\boldsymbol{x}_p,y_p)$ ... $(\boldsymbol{x}_q,y_q)$ $(\boldsymbol{x}_p,y_p+\boldsymbol{x}_p^T\alpha\boldsymbol{v})$ ... $(\boldsymbol{x}_q,y_q+\boldsymbol{x}_q^T\alpha\boldsymbol{v})$ $(\boldsymbol{x}_n,y_n)$
$(\boldsymbol{x}_p,y_p)$ ... $(\boldsymbol{x}_q,y_q)$ $(\boldsymbol{x}_p,y_p-\boldsymbol{x}_p^T\alpha\boldsymbol{v})$ ... $(\boldsymbol{x}_q,y_q-\boldsymbol{x}_q^T\alpha\boldsymbol{v})$ $(\boldsymbol{x}_n,y_n-\boldsymbol{x}_n^T\alpha\boldsymbol{v})$
(if $n-p$ even)

FIGURE 13.1. The two corrupted samples in the proof of Theorem 13.1 (i).

$$\#(\mathcal{G}) \geq h^*(\mathcal{Z}') - m \geq n - 2m = n - 2\lfloor \frac{n-p}{2} \rfloor \geq p. \tag{13.11}$$

Let $\mathcal{J}_{\mathcal{G}}$ be the set of indices associated with data points in $\mathcal{G}$, $\mathcal{S}_{\mathcal{J}_{\mathcal{G}}}(\mathcal{Z}') = \mathcal{S}_{\mathcal{J}_{\mathcal{G}}}(\mathcal{Z}) = \mathcal{S}_{\mathcal{J}_{\mathcal{G}}}$. The set $\tilde{\mathcal{S}}(\mathcal{Z}')$ is included in $\mathcal{S}_{\mathcal{J}_{\mathcal{G}}}$, and any $p \times p$ submatrix of $\mathbf{X}$ has full rank, so $\mathcal{S}_{\mathcal{J}_{\mathcal{G}}}$ is bounded. From Eq. (13.4), $\mathcal{S}^*(\mathcal{Z}')$ is included in the union of such sets, so the distance $d[\mathcal{S}^*(\mathcal{Z}),\mathcal{S}^*(\mathcal{Z}')]$ is bounded. □

REMARK 13.3. Note that the bound in Eq. (13.6) only depends on $\mathcal{Z}$ through the number of data points.

REMARK 13.4. When the number of measurements tends to infinity, $\mathcal{S}^*$ can accommodate up to 50% outliers. This is obviously the largest possible percentage if the outliers are allowed to be organized in such a way that they can be described by the model. Note that in practice the outliers are seldom organized in this way, so that OMNE can perform satisfactorily even on cases where there is a large majority of outliers.

REMARK 13.5. Other regression equivariant parameter bounding policies could be defined, with a high breakdown point. A possible choice corresponds to sets $\mathcal{S}^{\#h}(\mathcal{Z})$ with fixed h. Suppose that m points of $\mathcal{Z}$ are replaced to give a corrupted sample $\mathcal{Z}'$, and that the experimental conditions are such that any $p \times p$ submatrix of $\mathbf{X}$ has full rank. One wants $d[\mathcal{S}^{\#h}(\mathcal{Z}),\mathcal{S}^{\#h}(\mathcal{Z}')]$ to be bounded whatever the outliers may be, so that h should satisfy $n - m \geq h$ (one must have $\mathcal{S}^{\#h}(\mathcal{Z}') \neq \emptyset$), and $h - m \geq p$ (any set $\mathcal{S}_{\mathcal{J}}(\mathcal{Z}')$, with $\#(\mathcal{J}) = h$, must contain at least p regular data points to be bounded). The maximal value for m which allows these inequalities to be satisfied is $m = \lfloor (n-p)/2 \rfloor$. The value of h given by $\hat{h} = \lfloor n/2 \rfloor + \lfloor (p+1)/2 \rfloor$ then allows the bound of Eq. (13.6) given in Theorem 13.1 (i) to be reached.

REMARK 13.6. The least median of squares (LMS) estimator[2] and the set estimator $\mathcal{S}^{\#\hat{h}}$ defined in Remark 13.5 both neglect up to 50% of the data when n tends to infinity. This systematic rejection of a large part of the data leads to a loss of information when there are fewer than 50% outliers. $\mathcal{S}^*$ does not reject any data *a priori* and therefore does not have such a drawback.

13.3.3. Bias Due to Outliers

The distance $d[\mathcal{S}^*(\mathcal{Z}),\mathcal{S}^*(\mathcal{Z}')]$, where $\mathcal{Z}'$ is a corrupted n-sample obtained from a regular n-sample $\mathcal{Z}$ by replacing m original points by arbitrary outliers, can be seen as a bias due to these outliers. Provided that $m \leq \lfloor (n-p)/2 \rfloor$ and that any $p \times p$ submatrix of $\mathbf{X}$ has full rank, this bias is known from Theorem 13.1 to be bounded. We now derive an expression for such a bound, which is used in Section 13.4 to define an optimality criterion for experimental design.

Define $\tilde{\mathcal{S}}(\mathcal{Z}')$ as in the proof of Theorem 13.1 (ii). Equation (13.11) implies that at least p regular data points of $\mathcal{Z}$ contribute to the definition of $\tilde{\mathcal{S}}(\mathcal{Z}')$. Let $\tilde{\mathcal{J}}$ be a set of p indices associated with any subset (with cardinal p) of these regular data points. One obviously has

$$\mathcal{S}^*(\mathcal{Z}) \subseteq \mathcal{S}_{\tilde{\mathcal{J}}}(\mathcal{Z}),$$

$$\tilde{\mathcal{S}}(\mathcal{Z}') \subseteq \mathcal{S}_{\tilde{\mathcal{J}}}(\mathcal{Z}),$$

so that

$$d[\mathcal{S}^*(\mathcal{Z}),\tilde{\mathcal{S}}(\mathcal{Z}')] \leq \max_{\theta,\theta' \in \mathcal{S}_{\tilde{\mathcal{J}}}(\mathcal{Z})} d(\theta,\theta').$$

Finally, from Eq. (13.4) and the definition of $\tilde{\mathcal{S}}(\mathcal{Z}')$, we get

$$d[\mathcal{S}^*(\mathcal{Z}),\mathcal{S}^*(\mathcal{Z}')] \leq \max_{\mathcal{J}|\#(\mathcal{J})=p} \; \max_{\theta,\theta' \in \mathcal{S}_{\mathcal{J}}(\mathcal{Z})} d(\theta,\theta').$$

As $d(\theta,\theta')$, we shall use the Euclidean distance $||\theta - \theta'||$. We first evaluate

$$\Delta[\mathcal{S}_{\mathcal{J}}(\mathcal{Z})] = \max_{\theta,\theta' \in \mathcal{S}_{\mathcal{J}}(\mathcal{Z})} ||\theta - \theta'||.$$

We assume, with no loss of generality, that $\mathcal{S}_{\mathcal{J}}(\mathcal{Z})$ is defined by the first p data. $\mathcal{S}_{\mathcal{J}}(\mathcal{Z})$ is a convex polyhedron with p pairs of parallel faces (parallelotope). The ith pair of faces is defined by

$$\theta^T\mathbf{x}_i = y_i + \varepsilon, \;\; \theta^T\mathbf{x}_i = y_i - \varepsilon, \;\; i \in \mathcal{J}.$$

Take one of the vertices of $\mathcal{S}_{\mathcal{J}}(\mathcal{Z})$ as the origin, and let $\mathbf{s}_i$, $i = 1, \ldots, p$ be the vectors of coordinates of the adjacent vertices. The maximum value of $||\theta - \theta'||$ is obtained when θ and θ' are vertices of $\mathcal{S}_{\mathcal{J}}(\mathcal{Z})$, so that

$$\Delta[\mathcal{S}_{\mathcal{J}}(\mathcal{Z})] = \max_{\mathbf{u} \in \mathcal{C}_p} ||\mathbf{Su}||, \tag{13.12}$$

where

$$\mathcal{C}_p = \{\mathbf{u} \in \mathbb{R}^p \mid u_i = \pm 1, i = 1, \ldots, p\},$$

and where the ith row of $\mathbf{S}$ is given by $\mathbf{s}_i^T$. The ordering of the vertices can be chosen such that any $\mathbf{s}_k$, $k \neq i$, belongs to the ith pair of faces of $\mathcal{S}_{\mathcal{J}}(\mathcal{Z})$, which can be written as

$$\mathbf{x}_i^T\mathbf{s}_k = 0, \;\; k \neq i, \;\; \text{with} \;\; i, k \in \mathcal{J}. \tag{13.13}$$

The vertex $\mathbf{s}_i$ does not belong to the ith face of $\mathcal{S}_{\mathcal{J}}(\mathcal{Z})$. It satisfies $\mathbf{x}_i^T\mathbf{s}_i = \pm 2\varepsilon$, and the origin can be chosen such that

$$\mathbf{x}_i^T\mathbf{s}_i = 2\varepsilon. \tag{13.14}$$

Let $\mathbf{X}_{\mathcal{J}}$ be the $p \times p$ matrix the ith row of which is equal to $\mathbf{x}_i^T$. This matrix has full rank, and from Eqs. (13.13) and (13.14) $\mathbf{S}$ satisfies

$$\mathbf{S} = 2\varepsilon \mathbf{X}_{\mathcal{J}}^{-1}.$$

From Eq. (13.12), $\Delta[\mathcal{S}_{\mathcal{J}}(\mathcal{Z})]$ can therefore be written as

$$\Delta[\mathcal{S}_{\mathcal{J}}(\mathcal{Z})] = \max_{\mathbf{u} \in \mathcal{C}_p} 2\varepsilon \, ||\mathbf{X}_{\mathcal{J}}^{-1}\mathbf{u}||,$$

or equivalently

$$\Delta[\mathcal{S}_{\mathcal{J}}(\mathcal{Z})] = \max_{\mathbf{X}_{\mathcal{J}}\mathbf{w} \in \mathcal{C}_p} 2\varepsilon ||\mathbf{w}||.$$

Replacing $\mathbf{w}$ by $\mathbf{v}/\rho$, with $||\mathbf{v}|| = 1$, one can write $\Delta[\mathcal{S}_{\mathcal{J}}(\mathcal{Z})]$ as

$$\Delta[\mathcal{S}_{\mathcal{J}}(\mathcal{Z})] = \max_{\mathbf{v} \in \mathcal{S}_{\mathcal{J},\rho}} \frac{2\varepsilon}{\rho},$$

where

$$\mathcal{S}_{\mathcal{J},\rho} = \{\mathbf{v} \in \mathbb{R}^p \mid ||\mathbf{v}|| = 1, \mathbf{v}^T\mathbf{x}_\mathbf{i} = \pm\rho, i \in \mathcal{J}\}.$$

The bound on $d[\mathcal{S}^*(\mathcal{Z}), \mathcal{S}^*(\mathcal{Z}')]$ is finally obtained by

$$d[\mathcal{S}^*(\mathcal{Z}), \mathcal{S}^*(\mathcal{Z}')] \leq \max_{\mathcal{J} | \#(\mathcal{J}) = p} \max_{\mathbf{v} \in \mathcal{S}_{\mathcal{J},\rho}} \frac{2\varepsilon}{\rho},$$

or equivalently

$$d[\mathcal{S}^*(\mathcal{Z}), \mathcal{S}^*(\mathcal{Z}')] \leq \frac{2\varepsilon}{\rho^*(\mathbf{X})}, \tag{13.15}$$

with

$$\rho^*(\mathbf{X}) = \min\{\rho \mid \exists \mathbf{v} \in \mathbb{R}^p, ||\mathbf{v}|| = 1, \exists p \text{ rows } \mathbf{x}_i^T \text{ of } \mathbf{X}, \mid \mathbf{x}_i^T\mathbf{v} \mid = \rho\}. \tag{13.16}$$

REMARK 13.7. If a true value θ^* can be defined for the model parameters, it belongs to $\mathcal{S}^*(\mathcal{Z})$, and from Eq. (13.15) any θ' in $\mathcal{S}^*(\mathcal{Z}')$ then satisfies

$$||\theta' - \theta^*|| \leq \frac{2\varepsilon}{\rho^*(\mathbf{X})}.$$

The bound $2\varepsilon/\rho^*(\mathbf{X})$ can be used as a quantitative measure of the robustness of the estimator. It depends on the experimental conditions through the value of ρ^*, hence the idea of designing the experiment so as to make ρ^* as large as possible.

13.4. EXPERIMENTAL DESIGN TOWARD ROBUSTNESS

A first qualitative condition to ensure robustness of OMNE with respect to outliers is given in Theorem 13.1 (ii): any $p \times p$ submatrix of **X** must have full rank. A quantitative criterion for designing experiments intended to yield a high protection against outliers can further be obtained from the expression of the bound on the bias due to outliers given by Eq. (13.15) .

DEFINITION 13.1. A $n \times p$ design matrix **X** is ρ-optimal if it maximizes the criterion $\rho^*(.)$ given by (16).

Experimental design for robust estimation seems to have received little attention in the literature. The only study we are aware of[17] concerns the minimization of the discrepancy of the predicted outputs $\mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$ obtained by standard least squares (SLS) when outliers are present, where **y** is the vector of measurement outputs. However, the SLS estimator has a breakdown point equal to $1/n$, and this policy should therefore be rejected when severe outliers are to be feared. Note that the bound on the bias due to outliers obtained in Ref. 2, Chapter 8 for the least median of squares (and other related) estimator(s), especially designed against severe outliers, is also related to $1/\rho^*$. Maximizing ρ^* can thus be of interest for these estimators as well. Further studies are required to investigate the theoretical properties of this new design policy, and to develop algorithmic procedures. (See also Chapter 8, Ref. 16.) The importance of a proper choice of the design matrix **X** for the robustness of OMNE is here simply stressed by an example.

EXAMPLE: Assume that $p = 2$ and consider the following feasible region for the regressors

$$\mathcal{X} = \{\mathbf{x} = (x_1, x_2)^T \in \mathbb{R}^p \mid 0 \leq x_1, 0 \leq x_2, x_1^2 + x_2^2 \leq 1\}.$$

When four measurements are to be performed, the maximal value of ρ^* in Eq. (13.16) is equal to $\sin(\pi/12)$ and is obtained for the design matrix

$$\mathbf{X} = \begin{pmatrix} 1 & 0 \\ \cos(\pi/6) & \sin(\pi/6) \\ \cos(\pi/3) & \sin(\pi/3) \\ 0 & 1 \end{pmatrix}. \tag{13.17}$$

Note that no replications are involved, contrary to classical D-optimal design. The ρ-optimal experiment defined by Eq. (13.17) is also $\hat{V}$-optimal.[13,14] It minimizes the volume of the estimated feasible set defined by

$$\mathcal{S}(\mathbf{X}, \theta_p) = \{\theta \in \mathbb{R}^p \mid -\varepsilon \leq x_i^T(\theta - \theta_p) \leq \varepsilon, \ i = 1, \ldots, n\},$$

where θ_p is any prior value for θ. The estimated feasible set corresponding to the design matrix **X** given by Eq. (13.17) is presented in Fig. 13.2 (solid lines). The volume of $\mathcal{S}(\mathbf{X},\theta_p)$ is always greater than or equal to the volume of $\mathcal{S}(\mathcal{Z})$. Assume that there are no outliers and that the four measurements are given by

$$\mathbf{y} = (5.1, 9.5, 11, 10.3)^T,$$

with bounds $\varepsilon = 0.5$. Fig. 13.2 presents $\mathcal{S}^*(\mathcal{Z})$ (dashed lines), which coincides here with $\mathcal{S}(\mathcal{Z})$.

The breakdown point of $\mathcal{S}^*$ given by Eq. (13.7) is here equal to 50%, which means (since $n = 4$) that up to one arbitrary outlier can be handled.

Suppose that a problem occurred in the registration of the last data point, so that it is replaced by the outlier $\mathbf{x}_4 = (0,1)^T$, $y_4 = 20.3$. The corresponding set $\mathcal{S}^*(\mathcal{Z}')$ is presented in Fig. 13.3, together with the outlier minimal number estimate associated with the measurements $(5.1, 5.4, 10.3, 20.3)^T$ and the D-optimal design matrix

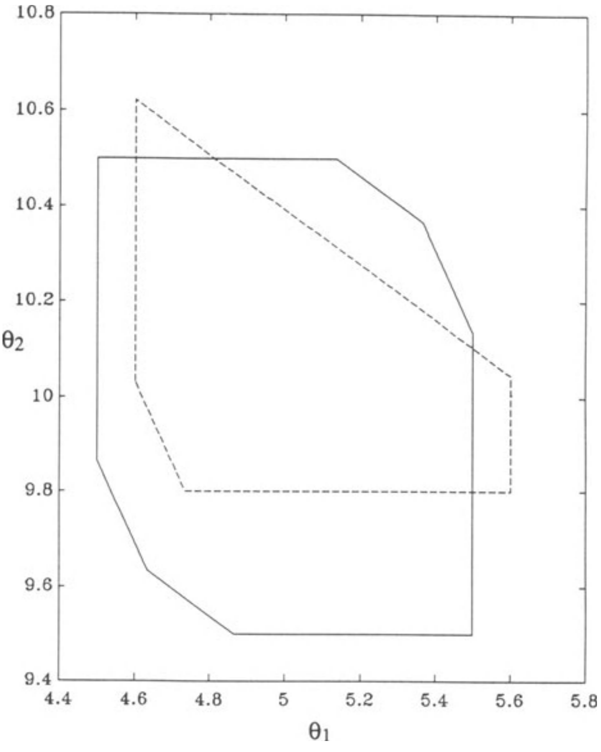


FIGURE 13.2. Estimated feasible set $\mathcal{S}(\mathbf{X},\theta_p)$ when $\theta_p = (5, 10)^T$ (solid lines), and posterior feasible set $\mathcal{S}^*(\mathcal{Z})$ (dashed lines) for regular data points, with the design matrix given by Eq. (13.17).

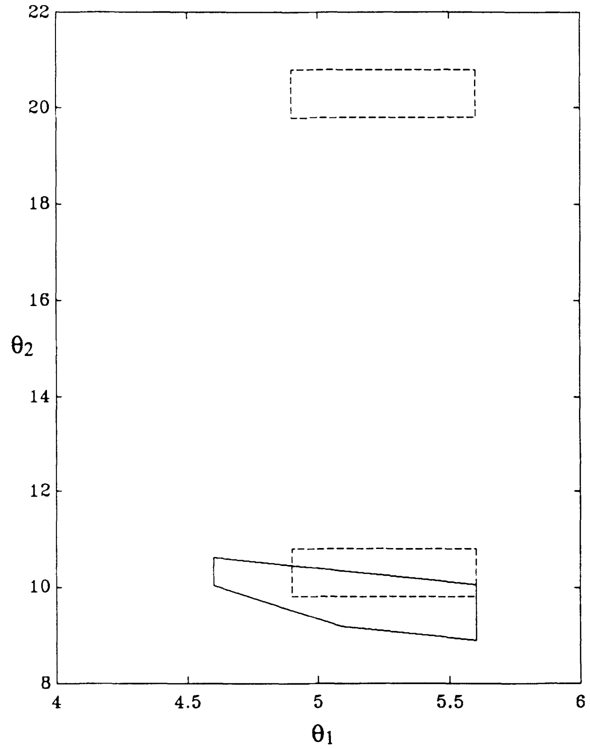


FIGURE 13.3. OMNE for the ρ-optimal design matrix of Eq. (13.17) (solid lines), and for the *D*-optimal design matrix of Eq. (13.18) (dashed lines) in the presence of one outlier.

$$\mathbf{X}_D = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{pmatrix}. \tag{13.18}$$

The presence of replications implies that $\rho(\mathbf{X}_D) = 0$. The conditions for OMNE to have a high breakdown point are therefore no longer fulfilled. Should y_4 tend to infinity, the maximum distance between $\mathcal{S}^*(\mathcal{Z})$ and $\mathcal{S}^*(\mathcal{Z}')$ would tend to infinity. As a consequence, classical *D*-optimal design should be avoided if robustness to outliers is an issue.

13.5. CONCLUSIONS

When outliers are to be expected and bounds are available on regular errors, OMNE is a powerful alternative to classical robust point estimators. Its breakdown point has been evaluated, and it reaches the highest achievable value. The bias due to the presence of outliers depends on the choice of the experimental conditions, which permits the definition of a new criterion for experimental design. This criterion may also be of interest for robust point estimators such as the least median of squares. Further studies are required to investigate the properties of the corresponding optimal design policy and to develop specific optimization procedures. Contrary to this new design policy, classical *D*-optimal design usually leads to replication of measurements. It may have disastrous consequences on robustness to outliers, as has been illustrated by a simple example.

REFERENCES

1. R. L. Launer and G. N. Wilkinson, editors, *Robustness in Statistics.* Academic Press, New York (1979).
2. P. J. Rousseeuw and A. M. Leroy, *Robust Regression and Outlier Detection.* Wiley, New York (1987).
3. J. P. Norton, *Automatica* **23**, 497 (1987).
4. J. R. Deller, *IEEE ASSP Mag.* **6**, 4 (1989).
5. M. Milanese, in: *Robustness in Identification and Control* (M. Milanese, R. Tempo, and A. Vicino, eds.) Plenum Press, New York pp. 3–24 (1989).
6. E. Walter and H. Piet-Lahanier, *Math. Comput. Simul.* **32**, 449 (1990).
7. E. Walter and H. Piet-Lahanier, in: *Proceedings of the 25th IEEE Conference on Decision and Control,* Athens, Greece pp. 1037–1042 (1986).
8. H. Lahanier, E. Walter, and R. Gomeni, *J. Pharm. Biopharm.* **15**, 203 (1987).
9. E. Walter and H. Piet-Lahanier, in: *Robustness in Identification and Control* (M. Milanese, R. Tempo, and A. Vicino, eds.) Plenum Press, New York pp. 67–76 (1989).
10. J. P. Norton, in: *Proceedings of the 25th IEEE Conference on Decision and Control,* Athens, Greece pp. 286–290 (1986).
11. D. P. Bertsekas and I. B. Rhodes, *IEEE Trans. Autom. Control* **16**, 117 (1971).
12. E. Walter and H. Piet-Lahanier, *IEEE Trans. Autom. Control* **34**, 911 (1989).
13. L. Pronzato and E. Walter, in: *Optimal Design and Analysis of Experiments* (Y. Dodge, V. V. Federov, and H. P. Wynn, eds.) North-Holland, Amsterdam, The Netherlands, pp. 195–205 (1988).
14. L. Pronzato and E. Walter, *Automatica* **25**, 383 (1989).
15. E. Walter and H. Piet-Lahanier, *Math. Biosci.* **92**, 55 (1988).
16. Ch. H. Müller, *J. Stat. Plan. Inf.* (1995) in press.
17. G. E. P. Box and N. R. Draper, *Biometrika* **62**, 347 (1975).

14

Ellipsoidal State Estimation for Uncertain Dynamical Systems

T. F. Filippova, A. B. Kurzhanski, K. Sugimoto, and I. Vályi

ABSTRACT

This chapter gives a concise description of effective solutions to the guaranteed state estimation problems for dynamic systems with uncertain items being unknown but bounded. It indicates a rigorous theory for these problems based on the notion of evolution equations of the "funnel" type which could be further transformed, through *exact* ellipsoidal representations, into algorithmic procedures that allow effective simulation, particularly with computer graphics. The estimation problem is also interpreted as a problem of tracking a partially known system under incomplete measurements.

Mathematically, the technique described in this chapter is based on a theory of set-valued evolution equations with the ellipsoidal-valued functions formulating approximation of solutions in terms of set-valued calculus.

T. F. Filippova • Institute of Mathematics and Mechanics of Russian Academy of Sciences, Ekaterinburg, Russia. A. B. Kurzhanski • Moscow State University, Moscow, Russia. K. Sugimoto • Okayama University, Okayama, Japan. I. Vályi • National Bank of Hungary, Budapest, Hungary.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

14.1. INTRODUCTION

The topic of this paper is motivated by problems of state estimation of dynamic processes described by ordinary differential equations with uncertain parameters or differential inclusions.(1,2,3,7,17,18,21,23) This topic already has a fairly large literature so that the published overviews are hardly able to give a full picture of the available achievements and research history. The aim of the present paper is to complement the available literature on the subject.

An uncertain system is said to be one of type

$$\dot{x}(t) \in A(t)x(t) + u(t), \quad t_0 \le t \le t_1, \quad x(t_0) = x_0, \tag{14.1}$$

where $A(t) \in \mathcal{R}^{n\times n}$ and $u(t) \in \mathcal{R}^n$ is the unknown but bounded input (disturbance). It is presumed that the *initial state* $x_0 \in \mathcal{R}^n$ is also unknown but bounded, so that

$$u(t) \in \mathcal{P}(t), \quad t_0 \le t \le t_1, \quad x_0 \in \mathcal{X}_0, \tag{14.2}$$

where the set $\mathcal{X}_0 \in \mathrm{conv}\mathcal{R}^n$ and the continuous set-valued function $\mathcal{P}(t) \in \mathrm{conv}\mathcal{R}^n$, $t_0 \le t_1$ are given (conv $\mathcal{R}^n$ denotes the family of all convex compact subsets of $\mathcal{R}^n$).

Equation (14.1) of the plant may be complemented by a state constraint

$$G(t)x(t) \in \mathcal{K}(t), \quad t_0 \le t \le t_1 \tag{14.3}$$

where $G(t) \in \mathcal{R}^{m\times n}$ and $\mathcal{K}(t) \in \mathrm{conv}\ \mathcal{R}^m$, $m \le n$. The constraint (1.3) may be particularly generated by a measurement equation

$$y(t) = G(t)x(t) + v(t), \quad t_0 \le t \le t_1, \tag{14.4}$$

with an unknown but bounded error

$$v(t) \in Q(t), \quad t_0 \le t \le t_1, \tag{14.5}$$

where $Q(t) \in \mathrm{conv}\ \mathcal{R}^m$, $t_0 \le t_1$. With the realization $y(\cdot)$ being known, restriction Eqs. (14.4 and 14.5) become

$$G(t)x(t) \in y(t) - Q(t), \quad t_0 \le t \le t_1, \tag{14.6}$$

so that $y(t) - Q(t)$ now substitutes for $\mathcal{K}(t)$ (the whole function $y(\cdot)$ may however not be known in advance, arriving *on-line*).

The objective will be to estimate the system output

$$w(t) = Hx(t), \quad w \in \mathcal{R}^r, \quad r \le n, \quad t_0 \le t \le t_1 \tag{14.7}$$

with $H \in \mathcal{R}^{r\times n}$, at a prescribed instant of time t, either for Eqs. (14.1 to 14.3) for the attainability problem under state constraints, or for Eqs. (14.1, 14.2 and 14.6) for the Guaranteed State Estimation Problem.

The solution approaches to both problems are well known.(7,9,11,21) The aim, however, is not to repeat this information but to rewrite the theoretical results focusing on the main objective: a constructive algorithmic procedure based on ellipsoidal techniques that allows a simulation with graphical representations.

14.2. THE ESTIMATION PROBLEMS

We start with the attainability problem. Let $x[\cdot] = x(\cdot,t_0,x_0)$ stand for an isolated solution of system Eq. (14.1) that starts at point $x_0 = x(t_0)$. As is well known, the attainability domain for Eqs. (14.1 to 14.3) at time $t \in [t_0,t_1]$ from point $x_0 \in \mathcal{R}^n$ is the cross-section at $t \in [t_0,t_1]$ of the tube $\mathcal{X}(\cdot,t_0,x_0)$ of all trajectories $x[\cdot] = x(\cdot,t_0,x_0)$ that satisfy Eqs. (14.1 to 14.3). Further, let $\mathcal{X}[t] = \mathcal{X}(t,t_0,\mathcal{X}_0)$ be defined by the relation

$$\mathcal{X}[t] = \cup\ \{\mathcal{X}(t,t_0,x_0)\ |\ x_0 \in \mathcal{X}_0\} \tag{14.8}$$

then $\mathcal{X}[t]$ is the attainability domain at time t from set $\mathcal{X}_0$.

The multivalued map $\mathcal{X}[\cdot]$ generates a generalized dynamic system. Namely the mapping

$$\mathcal{X}: [t_*,t_1] \times [t_*,t_1] \times \mathrm{conv}\mathcal{R}^n \rightarrow \mathrm{conv}\mathcal{R}^n$$

possesses a semigroup property, that is whatever are the values $t_* \le t_0 \le t \le \tau \le \theta \le t_1$ we have

$$\mathcal{X}[\theta,t,\mathcal{X}[t]] = \mathcal{X}\{\theta,\tau,\mathcal{X}[\tau,t,\mathcal{X}[t]]\}.$$

Also, if $m = n$ and $G(t) \equiv I(t_0 \le t \le t_1)$ in Eq. (14.3), the set-valued map, or in other words, the tube $\mathcal{X}[t]$, $(t_0 \le t \le t_1)$ satisfies an evolution equation—a 'funnel' equation(9,11,20)—which is

$$\lim_{\sigma \rightarrow +0} \sigma^{-1} h(\mathcal{X}[t+\sigma], ((I + A(t)\sigma)\mathcal{X}[t]$$

$$+ \sigma\mathcal{P}(t)) \cap \mathcal{K}(t+\sigma)) = 0, \quad t_0 \le t \le t_1,$$

$$\mathcal{X}[t_0] = \mathcal{X}_0, \tag{14.9}$$

Here $h(\mathcal{X}',\mathcal{X}'')$ stands for the Hausdorff distance between $\mathcal{X}',\mathcal{X}'' \in \mathrm{conv}\ \mathcal{R}^n$, namely

$$h(\mathcal{X}',\mathcal{X}'') = \max\ \{h_+(\mathcal{X}',\mathcal{X}''), h_-(\mathcal{X}',\mathcal{X}'')\},$$

$$h_+(\mathcal{X}',\mathcal{X}'') = \min\{\alpha \ge 0\ |\ \mathcal{X}' \subset \mathcal{X}'' + \alpha\mathcal{S}\},$$

$$h_-(\mathcal{X}',\mathcal{X}'') = h_+(\mathcal{X}'',\mathcal{X}').$$

with $\mathcal{S}$ being the unit ball in $\mathcal{R}^n$ and h_+, h_- called Hausdorff semi-distances.

Equation (14.9) is correctly posed and under some assumptions[11] has a unique solution that defines the tube $\mathcal{X}[\cdot] = \mathcal{X}(\cdot,t_0,\mathcal{X}_0)$ for system Eqs. (14.1 to 14.3). One of the assumptions mentioned above is the Lipschitz continuity of the set-valued map $\mathcal{K}(\cdot)$:

$$h(\mathcal{K}(t'), \mathcal{K}(t'')) \leq k\,|\,t' - t''\,|$$

for some $k > 0$ and for any $t', t'' \in [t_0,t_1]$.

Using only one of the Hausdorff semi-distances in Eq. (14.9) leads to the loss of *uniqueness* of the solutions, but complemented with an extremality condition, alternative descriptions for the multivalued map $\mathcal{X}[\cdot]$ are obtained. On one hand, consider

$$\lim_{\sigma\to+0} \sigma^{-1}h_-(\mathcal{W}[t+\sigma], \{[I + A(t)\sigma]\mathcal{W}[t] + \sigma\mathcal{P}(t)\} \cap \mathcal{K}(t+\sigma)) = 0, \quad t_0 \leq t \leq t_1,$$

$$\mathcal{W}[t_0] = \mathcal{X}_0,$$

A set-valued map $\mathcal{X}_-[\cdot]$ will be defined as a *minimal solution* of Eq. (14.10) if it satisfies Eq. (14.10) for almost all $t \in [t_0,t_1]$ and if there exists no other solution $\mathcal{W}[\cdot]$ to Eq. (14.10) such that $\mathcal{X}_-[t] \supset \mathcal{W}[t]$ for all $t \in [t_0,t_1]$ and $\mathcal{X}_-[\cdot] \neq \mathcal{W}[\cdot]$. Equation (14.10) has a unique minimal solution under the conditions required for the existence and uniqueness of the solutions to Eq. (14.9). In this case (using the notation of Eq. (14.8), $\mathcal{X}[\cdot] = \mathcal{X}_-[\cdot]$. On the other hand,[12]

$$\lim_{\sigma\to+0} \sigma^{-1}h_+(\mathcal{V}[t+\sigma], ((I + A(t)\sigma)\mathcal{V}[t] \cap \mathcal{K}(t))$$

$$+ \sigma\mathcal{P}(t)) = 0, \quad t_0 \leq t \leq t_1,$$

$$\mathcal{V}[t_0] = \mathcal{X}_0, \tag{14.11}$$

has a unique maximal solution $\mathcal{X}_+[\cdot]$ (defined analogously to the minimal solution of Eq. (14.10)) if, for example, $\mathcal{K}(\cdot)$ is upper semicontinuous.[1] If so then, as previously, $\mathcal{X}[\cdot] = \mathcal{X}_+[\cdot]$.

The *Guaranteed State Estimation Problem* may now be formulated as follows. Suppose that the measurement $y^*(\cdot)$ due to system Eqs. (14.1 to 14.4) is given. It is generated by an unknown triplet

$$\zeta^*(t) = \{x_0^*,u^*(t),v^*(t)\}, \quad t_0 \leq t \leq t_1, \tag{14.12}$$

which complies with the constraints of Eqs. (14.2 and 14.5). Then the tube of attainability domains $\mathcal{X}^*[\cdot]$ generated by Eqs. (14.1, 14.2 and 14.6); $y[\cdot] = y^*[\cdot]$ always contain the unknown actual trajectory of the system $x^*[\cdot]$, that is generated by $\zeta^*(\cdot)$. The tube $\mathcal{X}^*[\cdot]$, therefore, gives a guaranteed estimate of the state of system

Eq. (14.1) on the basis of a measurement $y^*(\cdot)$ of Eq. (14.4) under the constraints Eqs. (14.2 and 14.5). The solution of the problem is to specify the tube $\mathcal{X}^*[t]$, $t_0 \le t \le t_1$.

The set $\mathcal{X}[t] = \mathcal{X}(t,t_0,\mathcal{X}_0)$ is the domain of states $x(t)$ of system Eq. (14.1) at time t that, given $y(\tau)$, $t_0 \le \tau \le t$, are consistent with the constraints Eqs. (14.2 and 14.6) The attainability domain for system Eqs. (14.1, 14.2, and 14.6) is also known as the 'informational domain',(5) the 'domain of consistency', or the 'feasibility domain',(18,21,23) for the state estimation Eqs. (14.1, 14.2, 14.4, and 14.5).

Presume that $y(\cdot)$ is Lipschitz-continuous, to conform with the assertions above. The situation allows a generalization to the case when $y(\cdot)$ is a function measurable on $[t_0,t_1]$. The respective mathematical details, however, are beyond the scope of this chapter.

The solutions to the above estimation problems are given through the evolution Eqs. (14.9 and 14.10). An alternative approach to handle state constraints, based on the singular perturbation technique is also be presented. Now continue by devising an algorithmic scheme for solving the evolution equations.

14.3. THE DISCRETE-TIME SCHEME

Equations (14.9 and 14.10) yield a natural discrete-time scheme that can be given in two versions reflecting Eqs. (14.9 to 14.11). These two are first-order schemes:

$$\mathcal{X}[t+\sigma] = ((I+\sigma A(t))\mathcal{X}[t] + \sigma\mathcal{P}(t)) \cap \mathcal{K}(t+\sigma) \tag{14.13}$$

$$\mathcal{X}[t+\sigma] = ((I+\sigma A(t))\mathcal{X}[t] \cap \mathcal{K}(t)) + \sigma\mathcal{P}(t) \tag{14.14}$$

that yield convergence to the continuous-time solutions. The main problem is that the $\mathcal{X}[t]$s are arbitrary, convex, and compact sets mathematically described through infinite-dimensional elements, e.g., their support functions $\rho(l \mid \mathcal{X}[t])$. The objective is to give a constructive scheme for their description by approximating them through finite-dimensional elements which, in this chapter, are taken as ellipsoids and approximating the corresponding convex set-valued maps through ellipsoidal-valued functions.

14.4. THE ELLIPSOIDAL TECHNIQUES

Denote a nondegenerate ellipsoid as

$$\mathcal{E}(a,S) = \{x \mid (S^{-1}(x-a), x-a) \le 1\}$$

where $a \in \mathcal{R}^n$ is its center and the symmetric matrix $S > 0$ determines its configuration. From here

$$\rho(l \mid \mathcal{E}(a,S)) = (l,a) + (Sl,l)^{1/2}$$

where the latter description also allows $\det S = 0$.

Suppose the sets $\mathcal{X}_0$, $\mathcal{P}(t)$, $\mathcal{Q}(t)$, $\mathcal{K}(t)$ $(t_0 \le t \le t_1)$ are ellipsoids, so that

$$\mathcal{X}_0 = \mathcal{E}(x_0, X_0), \quad \mathcal{P}(t) = \mathcal{E}(p(t), P(t)), \tag{14.15}$$

$$\mathcal{Q}(t) = \mathcal{E}(q(t), Q(t)), \quad \mathcal{K}(t) = \mathcal{E}(k(t), K(t)), \tag{14.16}$$

and the conditions

$$X_0 \ge 0, \quad P(t) \ge 0, \quad Q(t) > 0, \quad K(t) > 0$$

hold.

The discrete-time schemes of Eqs. (14.13 and 14.14) then make it necessary to handle the following operations:

$$[\mathcal{E}(a_1,Q_1) + \mathcal{E}(a_2,Q_2)] \cap \mathcal{E}(a_3,Q_3)$$

$$[\mathcal{E}(a_1,Q_1) \cap \mathcal{E}(a_2,Q_2)] + \mathcal{E}(a_3,Q_3)$$

with $\mathcal{E}(a_i,Q_i)$, $Q_i \ge 0$, $i = 1, 2, 3$ given. This can be done through a combination of the following relations:

The sum of ellipsoids: Given ellipsoids $\mathcal{E}(a_i,Q_i)$, $i = 1, 2$, their sum $\mathcal{E}_s = \mathcal{E}(a_1,Q_1) + \mathcal{E}(a_2,Q_2)$ which need not be an ellipsoid, could be approximated from above as

$$\mathcal{E}_s \subset \mathcal{E}(a_1 + a_2, Q(\pi)), \tag{14.17}$$

where

$$Q(\pi) = (1 + \pi^{-1})Q_1 + (1 + \pi)Q_2, \quad \pi > 0.$$

LEMMA 14.1. The inclusion Eq. (14.17) is true whatever is the coefficient $\pi > 0$. The following relation holds:

$$\mathcal{E}_s = \cap \, \{\mathcal{E}(a_1 + a_2, Q(\pi)) \mid \pi > 0\}. \tag{14.18}$$

The intersection of ellipsoids: The intersection $\mathcal{E}_i = \mathcal{E}(a_1,Q_1) \cap \mathcal{E}(a_2,Q_2)$ can be approximated from above as

$$\mathcal{E}_i \subset \mathcal{E}(B_1 a_1, B_1 Q B_1') + \mathcal{E}(B_2 a_2, B_2 Q B_2'), \tag{14.19}$$

$$B_1 + B_2 = I, \tag{14.20}$$

where $B_i \in \mathcal{R}^{n\times n}$, $I \in \mathcal{R}^{n\times n}$ is the identity and the prime stands for the transpose.

LEMMA 14.2. The inclusion Eq. (14.19) is true for any matrices $B_i \in \mathcal{R}^{n\times n}$, $i =$ 1,2 that satisfy Eq. (14.20). The following equality is true

$$\mathcal{E}_i = \cap \{\mathcal{E}(B_1a_1,B_1QB_1') + \mathcal{E}(B_2a_2,B_2QB_2') \mid B_1 + B_2 = I. \qquad (14.21)$$

The following result is used in Section 14.6 and may be considered as a special case of Lemma 14.1.

The direct product of ellipsoids: Given ellipsoids $\mathcal{E}(p,P) \subset \mathcal{R}^k$, $\mathcal{E}(q,Q) \subset \mathcal{R}^m$, their direct product $\mathcal{E}(p,P) \times \mathcal{E}(q,Q) \subset \mathcal{R}^{k+m}$ can be approximated from above as

$$\mathcal{E}(p,P) \times \mathcal{E}(q,Q) \subset \mathcal{E}(z,Z(\pi)) \subset \mathcal{R}^{k+m} \qquad (14.22)$$

where $z = \{p,q\} \in \mathcal{R}^{k+m}$ and

$$Z(\pi) = \begin{pmatrix} (1+\pi^{-1})P & 0 \\ 0 & (1+\pi)Q \end{pmatrix}, \quad \pi > 0.$$

LEMMA 14.3. The Eq. (14.22) is true for any coefficient $\pi > 0$. The following relation holds:

$$\mathcal{E}(p,P) \times \mathcal{E}(q,Q) = \cap \{\mathcal{E}(z,Z(\pi)) \mid \pi > 0\}. \qquad (14.23)$$

The combination of Eqs. (14.18, 14.21 and 14.23) gives an exact external approximation of the sets in Eqs. (14.13 and 14.14) by a family of ellipsoids that can be simulated through parallelization. Among these one may also select an optimal ellipsoid. A somewhat different scheme can be given along the lines of Refs. (2,21,22).

Under the constraints of Eqs. (14.15) the attainability problem for the system is

$$\dot{x}(t) \in A(t)x(t) + \mathcal{E}(p(t),P(t)), \quad t_0 \le t \le t_1 \qquad (14.24)$$

$$x(t_0) \in \mathcal{E}(x_0,X_0) \qquad (14.25)$$

$$x(t) \in \mathcal{E}(k(t),K(t)), \quad t_0 \le t \le t_1. \qquad (14.26)$$

Ellipsoidal-valued functions may approximate the attainability tube $\mathcal{X}[\cdot]$ both internally and externally for Eqs. (14.24 and 14.26). Further sections deal only with the former case. (The schemes of internal ellipsoidal approximation for various attainability problems can be found in Refs. 29 and 22).

Consider the evolution equation

$$\lim_{\sigma\to+0} \sigma^{-1}\cdot h_+(\{[I+A(t)\sigma]\mathcal{E}[t] \cap \mathcal{E}(k(t),K(t))$$

$$+\ \sigma\mathcal{E}(p(t),\mathcal{P}(t)),\ \mathcal{E}[t+\sigma]\} = 0, \quad t_0 \le t \le t_1,$$

$$\mathcal{E}[t_0] = \mathcal{E}(x_0,X_0). \tag{14.27}$$

A function $\mathcal{E}_+[\cdot]$ is defined as a solution to Eq. (14.27) if it satisfies Eq. (14.27) for almost all $t \in [t_0,t_1]$ and is ellipsoidal-valued. Obviously the solution $\mathcal{E}_+[\cdot]$ is nonunique and satisfies the inclusion

$$\mathcal{E}_+[t] \supset \mathcal{X}[t], \quad t_0 \le t \le t_1, \quad \mathcal{E}_+[t_0] = \mathcal{X}[t_0].$$

Moreover, as a consequence of Lemmas 14.1, 14.2,

THEOREM 14.1. For any $t_0 \le t \le t_1$ the equality

$$\mathcal{X}[t] = \cap\ \{\mathcal{E}_+[t] \mid \mathcal{E}_+[\cdot] \text{ is a solution to (14.27) }\}.$$

The ellipsoidal solutions $\mathcal{E}_+[\cdot] = \mathcal{E}(x_+(\cdot),X_+(\cdot))$ to Eq. (14.27) allow explicit representations through appropriate systems of ODEs for the centers $x_+(\cdot)$ and the matrices $X_+(\cdot) > 0$ of these ellipsoids.(2,14,15,16,22)

14.5. ESTIMATION THROUGH PARAMETRIZATION

The point of interest of this Section is to study the set of all solutions $x[t] = x(t,t_0,x_0)$ to a nonlinear differential inclusion

$$\dot{x} \in \mathcal{F}(t,x), \quad t \in T = [t_0,t_1], \tag{14.28}$$

that are emitted by the initial compact subset $\mathcal{X}_0 \subset \mathcal{R}^n$ so that

$$x(t_0) = x_0, \quad x_0 \in \mathcal{X}_0 \tag{14.29}$$

where $\mathcal{F}(t,x)$ is a multivalued map ($\mathcal{F}$: $T \times \mathcal{R}^n \to \text{conv}\mathcal{R}^n$).

A further problem that concerns the set of these solutions is to single out a subset of those trajectories $x[t] = x(t,t_0,x_0)$ that satisfy both Eq. (14.28) and a restriction on the state vector (the "viability" constraint). (See also Sections (14.1 and 14.2.)

$$G(t)x[t] \in \mathcal{K}(t) = y(t) - \mathcal{Q}(t). \tag{14.30}$$

In a more general form this equation may be written as

$$y(t) \in G(t,x[t]), \tag{14.31}$$

where $G(t,x)$ is a multivalued map (G: $T \times \mathcal{R}^n \to \text{conv}\mathcal{R}^m$) or taking

$$G^{*}(t,x) = G(t,x) - y(t)$$

and omitting the asterisk, as

$$0 \in G(t,x[t]). \tag{14.32}$$

The requirements on $\mathcal{F}(t,x)$, $G(t,x)$ are given in Ref. 11.

DEFINITION. A trajectory $x[t] = x(t,t_0,x_0)$ ($x_0 \in \mathcal{X}_0$, $t \in T$) of the differential inclusion (14.28) is defined to be viable on $[t_0,\tau]$ if

$$0 \in G(t,x[t]) \quad \text{for all } t \in [t_0,\tau]. \tag{14.33}$$

The "guaranteed" estimation problem thus consists in describing the set

$$\mathcal{X}[\cdot] = \cup \,\{x(\cdot,t_0,x_0) \,|\, x_0 \in \mathcal{X}_0\}$$

of solutions $x(\cdot,t_0,x_0)$ to the system Eqs. (14.28, 14.29 and 14.30) (viable trajectories). The crossection $\mathcal{X}[t]$ of this set will be the set-valued estimate itself (see Section 14.2).

In this Section the description of trajectory tubes $\mathcal{X}[t]$ is reduced to the treatment of trajectory tubes for a variety of specially designed new differential inclusions without state constraints. These new inclusions are designed depending upon certain parameters and have a relatively simple structure. The overall solution is then presented as an intersection over the parameters of the parallel solution tubes to the new inclusions.

The restriction $\mathcal{F}_G(t,x)$ of the map $\mathcal{F}(t,x)$ to a multifunction $G(t,x)$ (at time t) is given by

$$\mathcal{F}_G(t,x) = \begin{cases} \mathcal{F}(t,x), & 0 \in G(t,x) \\ \varnothing, & 0 \notin G(t,x) \end{cases}$$

The next property follows directly from the definition of viable trajectories.

LEMMA 14.4. An absolutely continuous function $x(t)$ defined on the interval $[t_0,\tau]$ with $x_0 \in \mathcal{X}_0$ is a viable trajectory to Eq. (14.28) for $t \in [t_0,\tau]$ if and only if the inclusion

$$\dot{x}(t) \in \mathcal{F}_G(t,x)$$

is true for almost all $t \in [t_0,\tau]$.

Now represent $\mathcal{F}_G(t,x)$ as an intersection of certain multifunctions. The first step to achieve that objective is to indicate the following auxiliary assertion.

LEMMA 14.5. Suppose A is a bounded set, B a convex closed set, $A \subset \mathcal{R}^n$, $B \subset \mathcal{R}^m$. Then

$$\bigcap \,\{A + LB \mid L \in \mathfrak{R}^{n\times m}\} = \begin{cases} A, & 0 \in B \\ \varnothing, & 0 \notin B \end{cases}$$

where $\Re^{n\times m}$ is the space of all $n \times m$ matrices.

From Lemmas 14.4 and 14.5 one obtains the following characterization of viable trajectories.

THEOREM 14.2. An absolutely continuous function $x(\cdot)$ defined on an interval $[t_0,t_1]$ with $x(t_0)$ is a viable trajectory to Eq. (14.28) for $[t_0,\tau]$ iff the inclusion

$$\dot{x}(t) \in \cap \{(\mathcal{F}(t,x) + LG(t,x)) \mid L \in \Re^{n\times m}\}$$

is true for almost all $t \in [t_0,\tau]$.

A variety of differential inclusions that depend on a matrix parameter $L \in \Re^{n\times m}$ are given by

$$\dot{z} \in \mathcal{F}(t,z) + LG(t,z),$$

$$z(t_0) \in \mathcal{X}_0, \quad t_0 \le t \le t_1 \tag{14.34}$$

By $z[\cdot] = z(\cdot,\tau,t_0,z_0,L)$ denote the trajectory to Eq. (14.34) defined on the interval $[t_0,\tau]$ with $z[t_0] = z_0 \in \mathcal{X}_0$. Also denote

$$Z(\cdot,\tau,t_0,\mathcal{X}_0,L) = \cup \{Z(\cdot,\tau,t_0,z_0,L) \mid z_0 \in \mathcal{X}_0\}$$

where $Z(\cdot,\tau,t_0,z_0,L)$ is the bundle of all the trajectories $z[\cdot] = z(\cdot,\tau,t_0,z_0,L)$ issued at time t_0 from point z_0 and defined on $[t_0,\tau]$. The crossections of the set $Z(\cdot,\tau,t_0,\mathcal{X}_0,L)$ at time t are then denoted as $Z(\tau,t_0,\mathcal{X}_0,L)$.

THEOREM 14.3. For each $\tau \in [t_0,t_1]$ one has

$$\mathcal{X}(\cdot,\tau,t_0,\mathcal{X}_0) = \cap \{Z(\cdot,\tau,t_0,\mathcal{X}_0,L) \mid L \in \Re^{n\times m}\}.$$

Moreover, the following inclusion is true

$$\mathcal{X}[\tau] = \mathcal{X}(\tau,t_0,\mathcal{X}_0) \subseteq \cap \{Z(\tau,t_0,\mathcal{X}_0,L) \mid L \in \Re^{n\times m}\}.$$

PROOF. Theorem 14.3 is a direct consequence of Theorem 14.2. □

Now replace the constant matrix L in Eq. (14.34) by a continuous function $L(\cdot) \in \Re^{n\times m}[t_0,t_1]$, coming thus to the differential inclusion

$$\dot{z} \in \mathcal{F}(t,z) + L(t)G(t,z),$$

$$z(t_0) \in \mathcal{X}_0, \quad t_0 \le t \le t_1 \tag{14.35}$$

and keeping the earlier notation for its trajectory bundle $Z(\cdot,\tau,t_0,\mathcal{X}_0,L(\cdot))$ and $Z(\tau,t_0,\mathcal{X}_0,L(\cdot))$ for the τ-cross-section of the bundle.

What follows is a more precise version of Theorem 14.3.

THEOREM 14.4. For each $\tau \in [t_0,t_1]$ one has

$$\mathcal{X}(\cdot,\tau,t_0,\mathcal{X}_0) = \cap \{Z(\cdot,\tau,t_0,\mathcal{X}_0,L) \mid L \in \Re^{n\times m}[t_0,\tau]\}. \tag{14.36}$$

Moreover,

$$\mathcal{X}[\tau] = \mathcal{X}(\tau,t_0,\mathcal{X}_0) \subseteq \cap\ \{\mathcal{Z}(\tau,t_0,\mathcal{X}_0,L) \mid L \in \Re^{n\times m}[t_0,\tau]\} \tag{14.37}$$

where $\Re^{n\times m}[t_0,\tau]$ is the space of all continuous $n \times m$-matrix functions $L(t)$, $t \in [t_0,t_1]$.

The main point is that the Eq. (14.37) actually turns to be an equality if set-valued functions $\mathcal{F}(t,x)$, $G(t,x)$ are linear in x,

$$\mathcal{F}(t,x) = A(t)x + \mathcal{P}(t), \quad G(t,x) = G(t)x - y(t) + \mathcal{Q}(t) \tag{14.38}$$

see also Sections 14.1 and 14.2. The respective result is given by the following theorem.

THEOREM 14.5. Assume that both mappings $F(t,x)$, $G(t,x)$ are linear, (14.38). Then for each $\tau \in [t_0,t_1]$ one has

$$\mathcal{X}[\tau] = \mathcal{X}(\tau,t_0,\mathcal{X}_0) = \cap\ \{\mathcal{Z}(\tau,t_0,\mathcal{X}_0,L) \mid L \in \Re^{n\times m}[t_0,\tau]\}.$$

14.6. THE SINGULAR PERTURBATION TECHNIQUES

We now briefly describe another technique for solving the state estimation problem under state constraints, which may be useful particularly when $\mathcal{K}(\cdot)$ is discontinuous or only measurable in time t.

Taking the system Eqs. (14.1 to 14.3), we substitute the last relation by a singularly perturbed differential inclusion:

$$L(t)\dot{y}(t) \in -\,G(t)x(t) + \mathcal{K}(t), \quad t_0 \le t \le t_1 \tag{14.39}$$

$$y(t_0) \in \mathcal{Y}_0 \tag{14.40}$$

where $\mathcal{Y}_0 \in \mathrm{conv}\mathcal{R}^m$ is a given set, and $L(\cdot) \in \mathcal{M}^{m\times m}[t_0,t]$ and $\mathcal{M}^{m\times m}[t_0,t]$ denotes the space of continuous $m \times m$ matrix-valued functions with invertible values defined on the interval $[t_0,t]$. The above system has to be treated together with the differential inclusion

$$\dot{x}(t) \in A(t)x(t) + \mathcal{P}(t), \quad t_0 \le t \le t_1 \tag{14.41}$$

$$x(t_0) \in \mathcal{X}_0 \tag{14.42}$$

that follows from Eqs. (14.1 and 14.2). Equations (14.39 to 14.42) form a system

$$\dot{z}(t) \in B(t)z(t) + \mathcal{H}(t), \quad t_0 \le t \le t_1 \tag{14.43}$$

$$z(t_0) \in \mathcal{Z}_0 \tag{14.44}$$

with state space vector

$$z(t) = \begin{pmatrix} x(t) \\ y(t) \end{pmatrix}, \quad z(t) \in \mathcal{R}^{n+m},$$

parameters

$$\mathcal{B}(t) = \begin{pmatrix} A(t) & 0 \\ L^{-1}(t)G(t) & 0 \end{pmatrix}, \quad \mathcal{H}(t) = \begin{pmatrix} \mathcal{P}(t) \\ L^{-1}(t)\mathcal{K}(t) \end{pmatrix},$$

and the initial set of Eq. (14.44) taking the form

$$\mathcal{Z}_0 = \begin{pmatrix} \mathcal{X}_0 \\ \mathcal{Y}_0 \end{pmatrix} = \mathcal{X}_0 \times \mathcal{Y}_0.$$

Denote $\Pi_x z \in \mathcal{R}^n$ to be the projection of vector $z \in \mathcal{R}^{n+m}$ on the subspace corresponding to the state vectors $x(t)$ of Eq. (14.41). Given set $\mathcal{Z} \subset \mathcal{R}^{n+m}$, define

$$\Pi_x \mathcal{Z} = \{x \in \mathcal{R}^n \,|\, x = \Pi_x z, z \in \mathcal{Z}\}.$$

If we take $\mathcal{Z}_L[t] = \mathcal{Z}_L(t,t_0,\mathcal{Z}_0)$ to be the solution tube for system Eq. (14.43), then the following theorem turns out to be true.[10]

THEOREM 14.6. For any $t \in [t_0,t_1]$ and $\mathcal{Y}_0 \in \operatorname{conv}\mathcal{R}^m$

$$\mathcal{X}(t,t_0,\mathcal{X}_0) = \Pi_x(\cap \{\mathcal{Z}_L(t,t_0,\mathcal{Z}_0) \,|\, L(\cdot) \in \mathcal{M}^{m\times m}[t_0,t]\}). \tag{14.45}$$

where $\mathcal{X}(t,t_0,\mathcal{X}_0)$ is the attainability set under state constraint for the system of Eqs. (14.1, 14.2, 14.4 and 14.5).

A slight modification of this theorem is needed for the case when the initial set $\mathcal{Z}_0 \neq \mathcal{X}_0 \times \mathcal{Y}_0$ but the projection $\Pi_x\mathcal{Z}_0 = \mathcal{X}_0$.[3]

THEOREM 14.7. The following formula is true for any $\mathcal{Z}_0 \in \operatorname{conv}\mathcal{R}^{n+m}$, $t \in [t_0,t_1]$

$$\mathcal{X}(t,t_0,\Pi_x\mathcal{Z}_0) = \Pi_x(\cap \{\mathcal{Z}_L(t,t_0,\mathcal{Z}_0) \,|\, L(\cdot) \in \mathcal{M}^{m\times m}[t_0,t]\}).$$

An ellipsoidal version of Theorems 14.6 and 14.7 is based on analogous schemes to those in Section 14.4.

Assume, in addition to Eqs. (14.15 and 14.16)

$$\mathcal{Y}_0 = \mathcal{E}(y_0,Y_0), \quad Y_0 \geq 0. \tag{14.46}$$

Introduce the system

$$\dot{x}(t) \in A(t)x(t) + \mathcal{E}(p(t),P(t)), \tag{14.47}$$

$$L(t)\dot{y}(t) \in -\,G(t)x(t) + \mathcal{E}[k(t),K(t)], \tag{14.48}$$

$$\{x(t_0),y(t_0)\} \in \mathcal{E}(\{x_0,y_0\},\tilde{Z}_0), \tag{14.49}$$

$$\tilde{Z}_0 = \begin{pmatrix} X_0 & 0 \\ 0 & Y_0 \end{pmatrix} \tag{14.50}$$

with $L(\cdot) \in \mathcal{M}^{m\times m}[t_0,t_1]$, which actually is the system of Eqs. (14.43 and 14.44) for the data

$$\mathcal{H}(t) = \mathcal{E}(p(t),P(t)) \times \mathcal{E}(k(t),K(t)),$$

$$\mathcal{Z}_0 = \mathcal{E}(\{x_0,y_0\},\tilde{Z}_0).$$

The attainability set $\mathcal{Z}_L(t,t_0,\mathcal{Z}_0)$ of Eqs. (14.47 to 14.50) in general is not an ellipsoid, but one can introduce external ellipsoidal approximations for $\mathcal{Z}_L(t,t_0,\mathcal{Z}_0)$ following, for example, the techniques given in Refs. 2, 3 and 15. This yields the inclusion

$$\mathcal{Z}_L(t,t_0,\mathcal{Z}_0) \subset \mathcal{E}\{z[t,t_0,L(\cdot)],Z[t,t_0,L(\cdot),\pi(\cdot),\sigma(\cdot)]\},$$

where

$$z[t,t_0,L(\cdot)] = \begin{pmatrix} x(t) \\ y(t) \end{pmatrix} \tag{14.51}$$

$$Z[t,t_0,L(\cdot),\pi(\cdot),\sigma(\cdot)] = \begin{pmatrix} Z_1(t) & Z_2(t) \\ Z_2'(t) & Z_3(t) \end{pmatrix} \tag{14.52}$$

are the solutions to systems

$$\dot{x}(t) = A(t)x(t) + p(t) \tag{14.53}$$

$$x(t_0) = x_0 \tag{14.54}$$

$$\dot{y}(t) = -L^{-1}(t)G(t)x(t) + L^{-1}(t)k(t) \tag{14.55}$$

$$y(t_0) = y_0 \tag{14.56}$$

and

$$\begin{aligned} \dot{Z}_1(t) &= A(t)Z_1(t) + Z_1(t)A'(t) \\ &\quad + \sigma^{-1}(t)Z_1(t) + \sigma(t)[1 + \pi^{-1}(t)]P(t) \end{aligned} \tag{14.57}$$

$$Z_1(t_0) = X_0 \tag{14.58}$$

$$\dot{Z}_2(t) = -Z_1(t)G'(t)L'^{-1}(t) + [A(t) + \sigma^{-1}(t)I]Z_2(t) \tag{14.59}$$

$$Z_2(t_0) = 0 \tag{14.60}$$

$$Z_3(t) = -L^{-1}(t)G(t)Z_2(t) - Z_2'(t)G'(t)L'^{-1}(t) + \sigma^{-1}(t)Z_3(t)$$

$$+ \sigma(t)[1 + \pi(t)]L^{-1}(t)K(t)L'^{-1}(t) \tag{14.61}$$

$$Z_3(t_0) = Y_0 \tag{14.62}$$

THEOREM 14.8. For any $t \in [t_0,t_1]$ and $L(\cdot) \in \mathcal{M}^{m\times m}[t_0,t]$, the following equality is true

$$\mathcal{Z}_L(t,t_0,\mathcal{Z}_0) = \cap \, \{\mathcal{E}(z[t,t_0,L(\cdot)],Z(t,t_0,L(\cdot),\pi(\cdot),\sigma(\cdot))) \mid \pi(\cdot),\sigma(\cdot) \in C_+[t_0,t]\}$$

where $C_+[t_0,t]$ is the class of all positive scalar valued functions continuous on $[t_0,t]$.

Theorems 14.6–14.8 together yield:

THEOREM 14.9. Given instant $t \in [t_0,t_1]$ and $\mathcal{Y}_0 = \mathcal{E}(y_0,Y_0)$, the following equality is true

$$\mathcal{X}(t,t_0,\mathcal{X}_0) = \Pi_x(\cap \, \{\mathcal{E}(z[t,t_0,L(\cdot)],Z(t,t_0,L(\cdot),\pi(\cdot),\sigma(\cdot))) \mid$$

$$L(\cdot) \in \mathcal{M}^{m\times m}[t_0,t],\ \pi(\cdot),\sigma(\cdot) \in C_+[t_0,t]\}). \tag{14.63}$$

This allows one to present the attainability set under state constraint $\mathcal{X}(t,t_0,\mathcal{X}_0)$ as the projection of an intersection of ellipsoids. The important question of specifying the minimal class of functions $L(\cdot),\pi(\cdot)$ over which it would suffice to take Eq. (14.63) is not discussed in this chapter. There are examples, however, when the variety of such functions is *finite* (see also Section 14.7 and Figs. 14.7 and 14.8). Further examples of the application of the singular perturbation techniques to the problem of state estimation are given in Ref. 3.

For the technique of this section to be applicable it is enough that the set-valued map $\mathcal{K}(\cdot)$ (and, therefore, also the functions $k(\cdot),K(\cdot)$) are integrable. This allows a robust simulation of the solution to the problem with irregular noise in Eq. (14.4).

What follows in Section 14.9 are the results of numerical simulations for the estimation problems, including the tracking type representation of the solutions, as well as for the singular perturbation techniques.

14.7. GUARANTEED STATE ESTIMATION AS A TRACKING PROBLEM

One of the conventional guaranteed estimates for the unknown states $x(t)$ is the "Chebyshev center" $x^0(t)$ for $\mathcal{X}[t]$ which is given by the relation[7]

$$\max\{\|x^0(t) - x\| \mid x \in \mathcal{X}[t]\}$$

$$= \min\{\max\{\|z - x\| \mid x \in \mathcal{X}[t]\} \mid z \in \mathcal{X}[t]\}$$

(this also allows one to mention the guaranteed estimation techniques as those of "minimax estimation"). The calculation of these estimates is discussed in Ref. 7.

The difficult point is that the vector $x^0(t)$ usually does not satisfy any "nice" differential equation (except when the restrictions on the unknowns are symmetrical in some sense). The respective applications may not require a precise calculation of $x^0(t)$, however. On the other hand, the Chebyshev center for an ellipsoid $\mathcal{E}(c,P)$ is precisely the point c. We shall therefore indicate a scheme where $x^0(t)$ is substituted for $x_+(t)$: the center of one of its external ellipsoidal estimates $\mathcal{E}_+[t]$.

According to the previous Sections 14.5 and 14.6, the center $x_+(t)$ of each the external tubes $\mathcal{E}_+[t]$, $t_0 \leq t \leq t_1$, allows a representation

$$\dot{x}_+(t) = A(t)x_+(t) + L(t)G(t)x_+(t) + p(t) - L(t)q(t) - L(t)y(t), \quad x_+(t_0) = x_0,$$

where $L(t)$ is a matrix parameter or $L(t)$ is substituted by a functional $L(t,\cdot) = L[t,y_t(\cdot)]$ with memory $y_t(\cdot) = y(t + \sigma)$, $t_0 - t \leq \sigma \leq 0$.

The actual trajectory to be estimated is defined according to Eq. (14.12), by $x^*(\cdot)$. By the construction, the inclusion

$$\mathcal{E}_+[t] \supseteq \mathcal{X}^*[t], \quad t_0 \leq t \leq t_1$$

holds. Therefore, the result of the approximate estimation procedure is that the center $x_+(t)$ tracks $x^*(t)$ on the basis of the measurement $y^*(\tau)$ with $t_0 \leq \tau \leq t$. The ellipsoid $\mathcal{E}_+[t]$ around it plays the role of a guaranteed confidence region. According to the terminology used in identification theory, the set $\mathcal{X}^*[t]$ is the error set of the estimation process.

The matrix parameter $L(t)$ may here act as a control to minimize or guarantee a fixed value of the maximal error

$$\max\{\|x_+(t) - x^*(t)\| \mid u(\cdot), v(\cdot), x_0, \quad \text{Eqs. (14.1–14.5)}\}$$

either for a specified instant $t = \theta$ or for any $t \geq t' > t_0$, or to ensure that the integral cost

$$\max\{\int_{t_0}^{\theta} \|x_+(t) - x^*(t)\|^2 dt$$

$$+ \|x_+(\theta) - x^*(\theta)\|^2 \mid u(\cdot), v(\cdot), x_0, \text{ Eqs. (14.1–14.5)} \}$$

would be minimal.

This procedure is similar in nature to a differential game of observation.(5) A feedback duality theory for differential games of observation and control is indicated in Ref. 6.

14.8. THE DETERMINISTIC AND THE STOCHASTIC FILTERING APPROACHES

Suppose that the system of Eqs. (14.1 to 14.4) is specified as follows

$$\dot{x} \in A(t)x + \mathcal{P}(t), \tag{14.64}$$

$$y(t) \in G(t)x + Q(t), \tag{14.65}$$

$$x(t_0) \in X_0, \tag{14.66}$$

where Eq. (14.65) is the measurement (observation) equation. The continuous multifunction $Q(t)$ (Q: $T = [t_0, t_1] \to \text{conv}R^n$) reflects the restriction on the unknown but bounded noise w in the observations as indicated in Eq. (14.5).

Given the measurement $y = y^*(t)$, $t \in [t_0, \tau]$ the guaranteed state estimation problem as indicated in the Introduction is to specify at a given time-instant τ the set $X[\tau]$ of all states $x[\tau]$ of Eq. (14.64) that are consistent with the Eqs. (14.64 to 14.66) when $y(t) \equiv y^*(t)$.

In other words, one is to find the crossections at time $t = \tau$ of the "viability tube" $X[t]$ for Eqs. (14.64 to 14.66), $y(t) \equiv y^*(t)$. In the state estimation context the set $X[\tau]$ is known as the informational domain, or consistency domain (see Section 14.2). This set depends on the measurement $y_\tau(\sigma) = y(\tau + \sigma)$, $t_0 - \tau \le \sigma \le 0$, namely,

$$X[\tau] = X[\tau, y_\tau(\cdot)].$$

For the linear system under consideration $X[\tau, y_\tau(\cdot)] \in \text{conv}R^n$.

The problem of finding $X[\tau]$ is further propagated into one of describing the evolution of $X[\tau] = X[\tau, y_\tau(\cdot)]$ in time. The evolution Eq. (14.9) for $X[\tau]$ would, therefore, be the "guaranteed filtering" Eqs. (14.64 to 14.66) for the system with unknown but bounded uncertainties.

Needless to say that the evolution Eq. of type (14.9) serve to be the solution to this problem (provided, of course, that $\mathcal{K}(t) = y(t) - Q(t)$ as indicated above, and that $\mathcal{K}(t)$ satisfies the respective assumptions).

It is well known, however, that a conventional stochastic filtering technique is given by the equations of the "Kalman filter" which turn to solve the stochastic filtering problem for linear systems with Gaussian noise. Can the equations of the Kalman filter also be used to describe the informational domain $X[\tau,y_\tau(\cdot)]$ for the guaranteed estimation problem of the above?

On one hand, the tube $X[t]=X[t,y_\tau(\cdot)]$ may be described through linear-quadratic approximations.[11] On the other hand, it may be described by the well established connections between the Kalman filtering equations and the solutions to the linear-quadratic problem of control.

Using the solutions of the previous Sections, fix a triplet

$$k(\cdot)=k^*(\cdot)=\{v^*(\cdot),w^*(\cdot),x_0^*\}$$

with

$$k^*(\cdot)\in\{\mathcal{P}(\cdot)\times[y(\cdot)-Q(\cdot)]\times X_0\}$$

and consider the stochastic differential equations

$$dz=[A(t)z+v^*(t)]dt+\sigma(t)d\xi \tag{14.67}$$

$$dq=[G(t)z+w^*(t)]dt+\sigma_1(t)d\eta \tag{14.68}$$

$$z(0)=x_0^*+\zeta,\quad q(0)=0 \tag{14.69}$$

where ξ,η are standard, normalized Brownian motions. They have continuous diffusion matrices $\sigma(t)$, $\sigma_1(t)$ and

$$\det(\sigma(t)\sigma'(t))\neq 0 \text{ for all } t\in T,$$

ζ is a Gaussian vector with zero mean and variance $M^*=\sigma_0\sigma_0'$.

Denoting

$$\sigma(t)\sigma'(t)=R^*(t),\ \sigma_1(t)\sigma_1'(t)=H^*(t)$$

and treating $q=q(t)$ as the available measurement one may find the equations for the minimum variance estimate

$$z^*(t)=E(z(t)\,|\,q(s),\ t_0\leq s\leq t)$$

(the respective "Kalman filter").

These are

$$dz^*(t)=[A(t)-\Sigma(t)G'(t)H^{*-1}(t)G(t)]z^*(t)dt+$$

$$+\,\Sigma(t)G'(t)w^*(t)dt + \Sigma(t)G'(t)dq(t) + v^*(t)dt,\ z^*(t_0) = x_0^*, \tag{14.70}$$

$$\Sigma(t) = A(t)\Sigma(t) + \Sigma(t)A'(t) -$$

$$-\,\Sigma(t)G'(t)H^{*-1}(t)G(t)\Sigma(t) + R^*(t),\ \Sigma(t_0) = M^* \tag{14.71}$$

The estimate $z^*(t)$ depends on the triplets

$$k^*(\cdot) \text{ and } \Lambda^* = \{M^*, R^*(\cdot), H^*(\cdot)\} \in \Im.$$

Consider the set

$$Z^*(t) = Z^*(t,\Lambda^*) = \cup\ \{z^*(t)\,|\,k^*(\cdot) \in \{\mathcal{P}(\cdot) \times \mathcal{K}(\cdot) \times X_0\}\}$$

which, with a given realization $q(t)$, is the attainability domain for Eq. (14.70).

THEOREM 14.10. Assume the equalities

$$M^* = M^{-1},\quad R^*(t) \equiv R^{-1}(t),\quad H^*(t) \equiv H^{-1}(t) \tag{14.72}$$

to be true and $\mathcal{K}(t) = y(t) - Q(t)$, $t \in T$. Also assume

$$q(t) \equiv \int_{t_0}^{t} y(\tau)d\tau \tag{14.73}$$

Then the following equality is true

$$X[\tau] = \cap\ \{Z^*(\tau,\Lambda^*)\,|\,\Lambda^* \in \Im\} \tag{14.74}$$

The last results describe a clear connection between the solutions to the linear-quadratic Gaussian filtering problem (the Kalman filter), and the solutions to the deterministic guaranteed state estimation problems for uncertain systems with unknown but bounded "noise" in the nonquadratic case of the instantaneous constraints on the unknowns.

14.9. NUMERICAL EXAMPLES

Study a four-dimensional system of Eqs. (14.1) and (14.2) over the time interval [0,5] to consider the *Attainability Problem under State Constraints.*

The initial state is bounded by the ellipsoid $X_0 = \mathcal{E}(x_0, X_0)$ at the initial moment $t_0 = 0$ with

$$x_0 = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \end{pmatrix} \text{ and } X_0 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{14.75}$$

Consider a case when the right hand side is constant:

$$A(t) \equiv \begin{pmatrix} 0 & 1 & 0 & 0 \\ -8 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -4 & 0 \end{pmatrix}, \tag{14.76}$$

describing the position and velocity of two independent oscillators. Inputs $u(t)$ are also bounded by time independent constraints $\mathcal{P}(t) = \mathcal{E}(p(t),P(t))$ with

$$p(t) \equiv \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \text{ and } P(t) \equiv \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.01 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0.01 \end{pmatrix}. \tag{14.77}$$

This form of the bounding sets makes the system coupled. State Eq. (14.3) is defined by the data

$$G(t) \equiv \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad k(t) \equiv \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad K(t) \equiv \begin{pmatrix} 16 & 0 \\ 0 & 25 \end{pmatrix}. \tag{14.78}$$

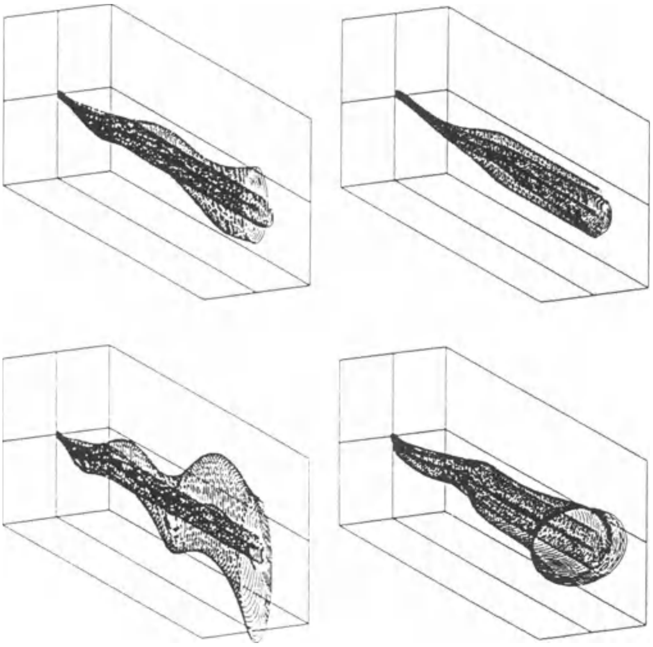


FIGURE 14.1. Tube of external ellipsoidal estimates of attainability sets.

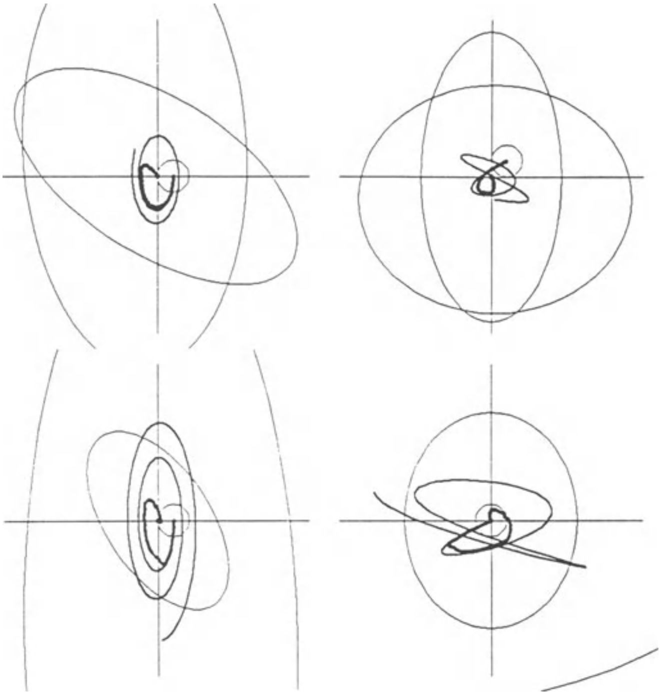


FIGURE 14.2. Trajectories of the centers and final estimates in phase space.

Fig. 14.1 shows the graph of external ellipsoidal estimates of the system outputs (with and without constraints), presenting them in four windows. Here, as well as in Figs. 14.2 to 14.6, the matrix H of Eq. (14.7) is equal to four projections of the phase space vector to the planes $\{x_1,x_2\}$, $\{x_3,x_4\}$, $\{x_1,x_3\}$, $\{x_2,x_4\}$ in a clockwise order starting from bottom left. The drawn segments of coordinate axes corresponding to the output variables range from –30 to 30. The skew axis in Fig. 1 is time, ranging from 0 to 5.

Calculations are based on the discretized version of Eqs. (14.24 to 14.26, and 14.13). Trajectories of the centers are also drawn. The thick line corresponds to estimates of the nonconstrained outputs. Fig. 14.2 shows the trajectory of the centers, initial sets, and the ellipsoidal estimates of the outputs in phase space with the coordinate axes ranging from –10 to 10.

Turn now to the guaranteed state estimation problem interpreted as a tracking problem, of Eqs. (14.1, 14.6, and 14.7). [Keep the above parameter values of the time interval, $A(t),\mathcal{E}(x_0,X_0),\mathcal{E}[p(t),P(t)],G(t)$ and H. In Eq. (14.6) that now replaces Eq. (14.3), take

$$q(t) \equiv k(t) \quad \text{and} \quad Q(t) \equiv K(t). \tag{14.79}$$

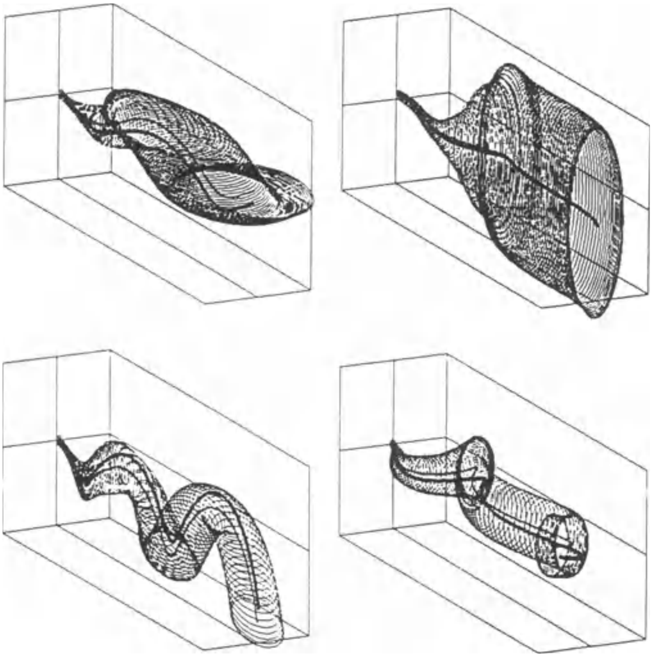


FIGURE 14.3. Time representation of ellipsoidal tracking (worst noise).

We model the trajectory $x^*(\cdot)$ and the outputs $z^*(\cdot)$, those to be tracked, by using the following construction for the triplet

$$\zeta^*(\cdot) = \{x_0^*, u^*(\cdot), v^*(\cdot)\}.$$

The initial value x_0^* is a (randomly selected) element at the boundary of the initial set $X_0 = \mathcal{E}(x_0, X_0)$. The input $u^*(\cdot)$ is of the so called extremal bang–bang type: the time interval is divided into subintervals of constant lengths. A value u is chosen randomly at the boundary of the respective bounding set, that is, in case of the input

$$u^*(t), \text{ of } \mathcal{P}(t) = \mathcal{E}(p(t), P(t))$$

and its value is then defined as $u^*(t) = u$ over all the first interval, and as $u^*(t) = -u$ over the second. Then a new random value for u is selected and the above procedure is repeated for the next pair of intervals, etc.

For modeling the measurement noise $v^*(\cdot)$ (generating together with x_0^* and $u^*(\cdot)$ the actual measurement $y^*(\cdot)$), use a similar procedure. As is well known, the size of the error set of the estimation depends on the nature of $v^*(\cdot)$. According to Ref. 6, if one chooses it in such a way that it takes a constant value at the boundary of $\mathcal{E}[q(t), Q(t)]$ over all the time interval under study, then it corresponds to the worst

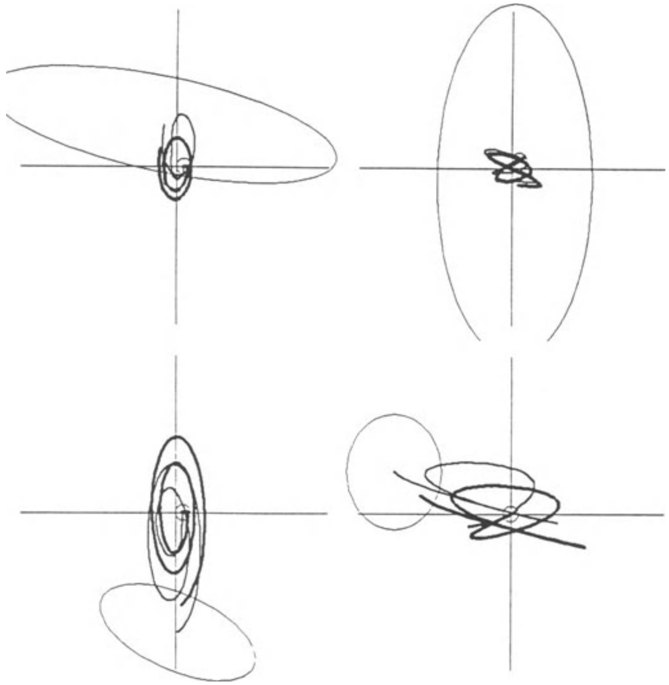


FIGURE 14.4. Phase space representation of ellipsoidal tracking (worst noise).

case. In large confidence regions using, e.g., the extremal bang–bang construction, 'good' noises are created, which reduces the confidence region size.

Fig. 14.3 shows the process developing over time. The drawn segments of coordinate axes correspond to the output variables range from –20 to 20. In Fig. 14.4, the initial sets of uncertainty (appearing as circles) are displayed in phase space, as well as the confidence region at the final moment. Coordinate axes range from –10 to 10. The trajectory drawn with the thick line is the actual output $z^*(\cdot) = Hx^*(\cdot)$. The thin line represents the trajectory of the centers $Hx_{-}(\cdot)$ of the projections of the tracking ellipsoids. Figs. 14.5 and 14.6 show how much the estimation can improve if the noise changes from worst to better. Although, one obtains only external ellipsoidal estimates of the true error sets. Opposed to the above, where the noise was constant, one chooses its length to be 0.05. Again, the range of coordinate axes is –20 to 20.

To illustrate the singular perturbation technique, we chose a system of two dimensions, and a scalar measurement equation, by taking the first two state variables, and the first coordinate of the measurement Equation (14.6) of the above example over the same time period. This means taking the first two entries of the

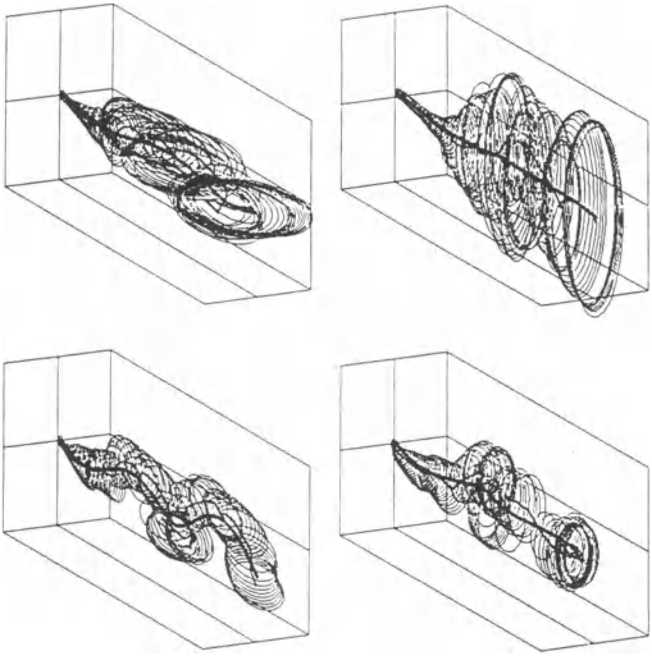


FIGURE 14.5. Time representation of ellipsoidal tracking (better noise).

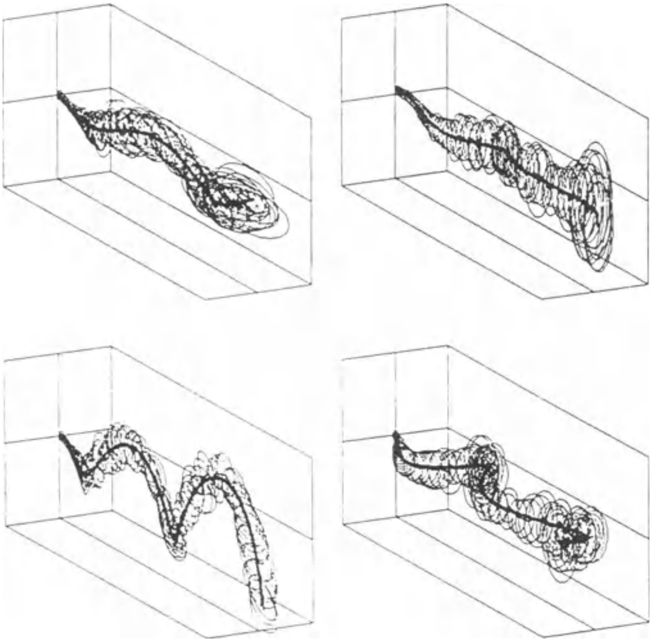


FIGURE 14.6. Phase space representation of ellipsoidal tracking (better noise).

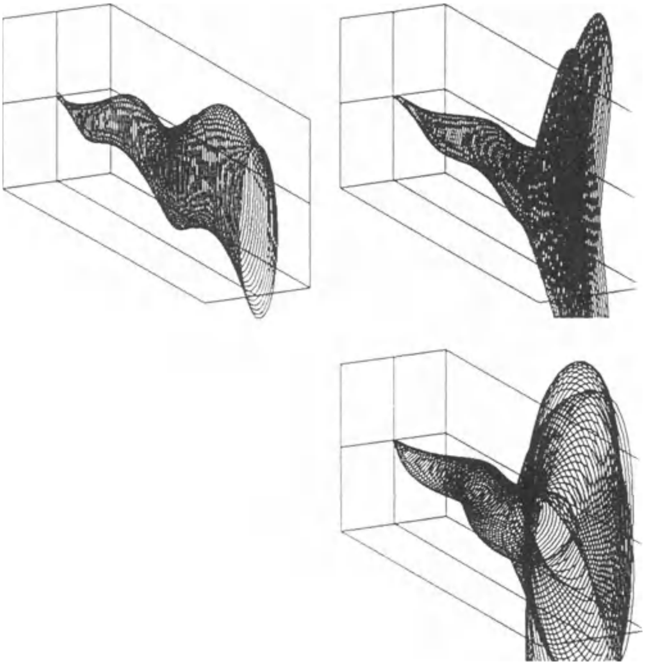


FIGURE 14.7. Ellipsoidal estimates developing over time: singular perturbation technique.

vectors and the upper left 2 by 2 block of the matrices in Eqs. (14.75) to (14.77). Further take

$$G(t) \equiv (0 \;\; 1), \quad q(t) \equiv 1, \quad Q(t) \equiv (1).$$

The two estimates shown correspond to the following choices for the function L:

$$L_+(t) = \begin{cases} 1 & \text{if } t \in [0,3.5] \\ 0.3 & \text{if } t \in [3.5,5], \end{cases} \quad L_-(t) = \begin{cases} 1 & \text{if } t \in [0,3.5] \\ -0.3 & \text{if } t \in [3.5,5]. \end{cases} \tag{14.80}$$

Additionally, suppose the initial condition:

$$y(0) \in [-10^{-5}, 10^{-5}].$$

Fig. 14.7 shows the two estimates developing over time with the range of coordinate axes being –30 to 30. The left upper window shows the projections onto the plane spanned by the two state variables. Here they coincide as expected. In the right upper window note the projection of the two estimating tubes onto the plane of the measurement variable and the first state variable, while in the lower window onto the plane of the measurement variable and the second state variable. In Figure

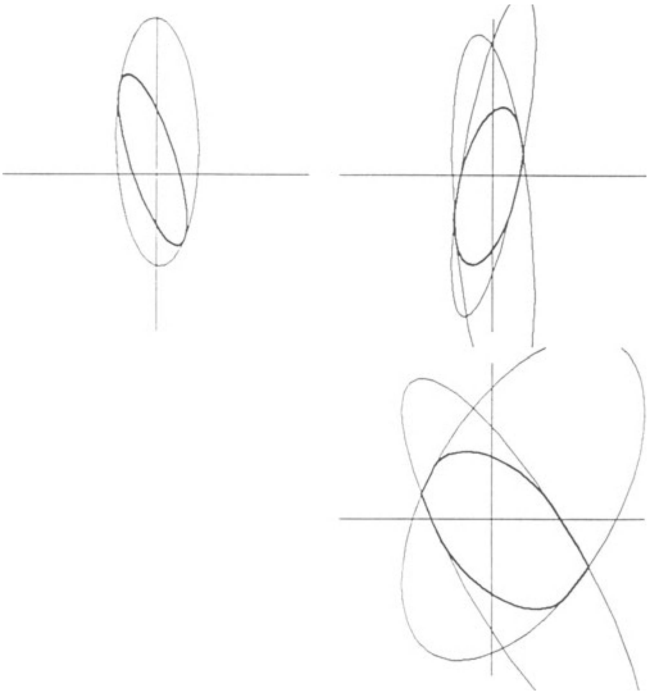


FIGURE 14.8. Ellipsoidal estimates and the projection of their intersection: singular perturbation technique.

14.8 note the estimates (in the same arrangement of the windows and in the same scale) at the moment $t = 4.25$, indicated by thin lines, and the projection of their intersection, indicated by a thicker line. In the space of the first two variables, the projections of the two estimates coincide again, but the projection of their intersection is a proper subset.

14.10. CONCLUSIONS

This chapter indicates constructive approaches with algorithmic ellipsoidal procedures for the state estimation problem for dynamic systems under unknown errors bounded by given instantaneous constraints.

Specifically, the guaranteed estimator may be presented as a system that tracks the unknown actual trajectory of the system. The procedures allow effective graphic simulation that is demonstrated on second and fourth order systems.

The connections between "Kalman" stochastic end deterministic "guaranteed" filtering problems with magnitude bounds on the unknowns are also specified.

ACKNOWLEDGMENTS. The research of T.F. Filippova and A.B. Kurzhanski was supported by grants RFFR 94-01-00803 and RFFR-ISF NBV 300.

REFERENCES

1. J.-P. Aubin and I. Ekeland, *Applied Nonlinear Analysis*, Wiley, New York (1984).
2. F. L. Chernousko, *Estimation of the Phase State of Dynamical Systems*, Nauka, Moscow (1988).
3. T. F. Filippova, A. B. Kurzhanski, K. Sugimoto, and I. Valyi, *Ellipsoidal Calculus, Singular Perturbations and State Estimation Problems for Uncertain Systems*. IIASA, WP-92-51 (1992).
4. N. N. Krasovskii, *The Control of a Dynamic System*, Nauka, Moscow, Russia (1968).
5. A. B. Kurzhanski, *Sov. Math. Dok.* **3**, 207 (1972).
6. A. B. Kurzhanski, *Izvestia A. N. SSR, Techn. Kibernetika* No. 5 (1973).
7. A. B. Kurzhanski, *Control and Observation under Conditions of Uncertainty*, Nauka, Moscow (1977).
8. A. B. Kurzhanski, in: *From Data to Model* (J. Willems, ed.), Springer-Verlag, Berlin, Germany (1988).
9. A. B. Kurzhanski and T. F. Filippova, in: *Les Annales de l'Institut Henri Poincare, Analyse Non-lineaire*, Paris, pp. 339–363 (1989).
10. A. B. Kurzhanski and T. F. Filippova, *Sov. Math. Dok.* **3**, 454 (1991).
11. A. B. Kurzhanski and T. F. Filippova, *On the Theory of Trajectory Tubes: A Mathematical Formalism for Uncertain Dynamics, Viability and Control, The Fields Institute for Research in Mathematical Sciences*, FI93-DS08, pp. 1–67 (1993); *Advances in Nonlinear Dynamics and Control: A Report from Russia*, Birkhäuser, Boston, MA (1993).
12. A. B. Kurzhanski and O. I. Nikonov, in: *Perspectives in Control Theory*, Vol. 2 of *Progress in Systems and Control Theory* (B. Jakubczyk, K. Malanowski, and W. Respondek, eds.) Birkhäuser, Boston, pp. 143–153 (1990).
13. A. B. Kurzhanski and O. I. Nikonov, *Dok. Akad. Nauk SSSR* **311**, 788 (1990).
14. A. B. Kurzhanski and I. Vályi, in: *Analysis and Optimization of Systems*, Vol. 111 of *Lecture Notes in Control and Information Sciences* (A. Bensoussan and J. L. Lions, eds.) Springer-Verlag, Berlin, Germany, pp. 775–785 (1988).
15. A. B. Kurzhanski and I. Vályi, *Dynamics and Control* **1**, 357 (1991).
16. A. B. Kurzhanski and I. Vályi, *Dynamics and Control* **2**, 87 (1992).
17. M. Milanese and A. Vicino, *Automatica* **27**, 997 (1991).
18. J. P. Norton, *Automatica* **23**, 4 (1987).
19. A. I. Ovseevich and F. L. Chernousko, *Prikl. Mat. Mech.* **46**, 5 (1982).
20. A. I. Panasyuk and V. I. Panasyuk, *Asymptotic Magistral Optimization of Controlled Systems*, Nauka i Technika, Minsk (1986).
21. F. C. Schweppe, *Uncertain Dynamic Systems*, Prentice-Hall, Englewood Cliffs, NJ (1973).
22. I. Vályi, in: *Modelling and Adaptive Control*, Vol. 105 of *Lecture Notes in Control and Information Sciences* (A. B. Kurzhanski and C. I. Byrnes, eds.) Springer-Verlag, Berlin, Germany, pp. 361–384 (1986).
23. E. Walter and H. Piet-Lahanier, in: *Proceedings of the 12th IMACS World Congress* (R. Vichnevetsky, P. Borne, and J. Vignes, eds.) IMACS (1988).

15

Set-Valued Estimation of State and Parameter Vectors within Adaptive Control Systems

V. M. Kuntsevich

ABSTRACT

The problem under consideration is that of obtaining simultaneously set-valued estimates for state and parameter vectors of linear (in parameters and in phase coordinates) discrete-time systems under uncontrollable bounded disturbances and given bounded noise in measurements.

There is no other *a priori* information on disturbances and noise except for they are bounded. It is shown that in the absence of noise in measurements and in the presence only of uncontrollable additive disturbances having an effect on stationary plants being investigated, the problem of obtaining set-valued parameter estimates is equivalent to the problem of determining a set-valued solution of a set of linear algebraic equations under uncertainty in their right-hand sides. With additive measurement noise, set-valued estimation procedure should be changed considerably since in this case one has to determine the whole set of solutions of a set of algebraic equations under uncertainty in coefficients as well as in right-hand sides. The problem of simultaneous estimation of state and parameter vectors can be reduced in the long run to the last-mentioned algebraic one.

V. M. KUNTSEVICH • V. M. Glushkov Institute of Cybernetics, Academy of Sciences of Ukraine, 252207 Kiev, Ukraine.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

The problem of set-valued estimation for nonstationary systems with restricted parameter drift rate is also considered.

15.1. INTRODUCTION

Set-valued estimation has been widely used in solving identification problems in the last decade. A number of publications[1–23] have been devoted to the range of problems under consideration. The problem of simultaneous estimation of state and parameter vectors holds a special place in this series. It is met in particular in unstable plants control when the parameter identification process cannot be separated from the control process itself as well as when both of the problems should be carried out simultaneously within adaptive control system. The above-mentioned problems are considered consecutively in the chapter.

15.2. SET-VALUED PARAMETER ESTIMATION FOR LINEAR NONSTATIONARY SYSTEMS

Recall the main idea of set-valued estimation with the simplest example namely with the parameter identification problem for plants without memory, which is widely known. One of the earliest general schemes suggested for obtaining a set-valued estimates of the parameters is considered below.[4]

Let a class of plants under consideration be stated by the equation:

$$y_n = L^T u_n + f_n, \quad n = 1,2,\ldots, \tag{15.1}$$

where L is k-dimensional vector of unknown but constant parameters; f_n is uncontrollable bounded disturbance (noise) with given *a priori* interval estimate

$$f_n \in \mathfrak{f} \quad \forall\, n > 0, \tag{15.2}$$

where

$$\mathfrak{f} = \{f\colon\ |f| \le \Delta = const\}. \tag{15.3}$$

No other information about disturbance f_n exists besides that it is bounded in terms of Eq. (15.3).

Let the estimate for vector L be known at the nth instant:

$$\mathrm{L} \in \mathfrak{L}_n, \tag{15.4}$$

where $\mathfrak{L}_n$ is a given convex set. At $n = 0$, $\mathfrak{L}_0$ is given *a priori*. It is required to obtain *a posteriori* estimate of L using measured values of u_n, y_n and estimate Eq. (15.4).

Using Equation (15.1) gives the estimate for L with known u_{n+1} and y_{n+1}

$$L \in \tilde{\mathfrak{L}}_{n+1} = \{L: \bigcup_{f_n \in \mathfrak{f}} u_n^{\mathrm{T}} L = y_n - f_n\}. \tag{15.5}$$

Then *a posteriori* estimate can be obtained by intersecting two noncontradictory set-valued estimates in Eqs. (15.4 and 15.5):

$$L \in \mathfrak{L}_{n+1} = \tilde{\mathfrak{L}}_{n+1} \cap \mathfrak{L}_n. \tag{15.6}$$

Obviously one can claim that estimate $\mathfrak{L}_{n+1}$ fulfills the relation $\mathfrak{L}_{n+1} \subseteq \mathfrak{L}_n$ and no more in general. The case $\mathfrak{L}_{n+1} = \mathfrak{L}_n$ corresponds to noninformative measurement of Eq. (15.1). In some particular cases,(12,21,23) procedure Eqs. (15.5 and 15.6) enable an identification process to be completed by obtaining a pointwise set containing the only true value of vector L in a finite number of steps. The pointwise set-valued estimate might be obtained for a particular class of uncontrollable disturbances f_n which meet an additional constraint besides Eq. (15.3),

$$\lim_{N\to\infty} 1/N \,|\Sigma_{n+1}^{N} f_n| = 0,$$

if this feature of sequence $\{f_n\}$ is taken into account in a proper way modifying a procedure.(8,23)

Some necessary and (or) sufficient conditions of noninformativity of Eqs. (15.5) can be pointed out. Nevertheless, the complexity of checking them is commensurable with that of carrying out an Eq. (15.6), which makes them useless.

Since $\tilde{\mathfrak{L}}_{n+1}$ is a hyperband in the parameter space, the result of intersection Eq. (15.6) is a convex polyhedron if $\mathfrak{L}_0$ is given in the form of convex polyhedron. In particular, when dealing with polyhedra described by their vertices, the algorithm (and its program) of two convex polyhedra intersecting is suggested.(8,23) Since the number of vertices of polyhedron $\mathfrak{L}_n$ varies and cannot be determined in advance, one has to operate with data array of a varied volume. This disadvantage of the algorithms of precise set-valued estimation is why a line of investigations are developed essentially for a class of polyhedra for estimation with approximating ellipsoids.(1,2,5,6,7,10,12,21) However, considerably less computational complexity of estimation with ellipsoids is achieved at the cost of the set-valued estimate's quality. The error of such approximation increases with the increase of parameter vector dimension. That is why methods of set-valued estimation in both classes have been developed in parallel. Areas of an application of polyhedral and ellipsoidal estimates presumably can be determined as follows. For problems of comparatively small dimension ($k \leq 10$), preference should be given to precise set-valued estimates. For problems of a middle dimension ($10 \leq k \leq 10^2$), ellipsoidal estimates are preferable. For problems of a large dimension ($k > 10^2$), an application of both methods is connected with essential difficulties, since approximation error increases sharply for ellipsoidal methods while memory and computational complexity increase rapidly for polyhedron methods.

A similar approach can be used for obtaining set-valued estimates of dynamic system parameters. Indeed, let a class of discrete-time control systems be given by the equation

$$X_{n+1} = AX_n + Bu_n + Cf_n, \quad n = 0,1,2,\ldots, \tag{15.7}$$

where X_n is m-dimensional state vector, A is $(m \times m)$-matrix, B and C are m-dimensional vectors, u_n is scalar control (input), and f_n is uncontrollable disturbance as shown above.

For simplicity assume that vector X_n is available for measurement without noise. Assume that matrix A and vectors B and C in Eq. (15.7) are of the canonic structure, i.e.,

$$A = \left\| \begin{matrix} 0; I_{m-1} \\ \cdots \\ A_m^{\mathrm{T}} \end{matrix} \right\|; \quad B = \left\| \begin{matrix} 0 \\ \cdots \\ b_m \end{matrix} \right\|; \quad C = \left\| \begin{matrix} 0 \\ \cdots \\ 1 \end{matrix} \right\| \tag{15.8}$$

where I_{m-1} is a unit matrix $(m-1) \times (m-1)$.

Let an *a priori* estimate be given for the vector of unknown but constant parameters $L^{\mathrm{T}} = (A_m^{\mathrm{T}}, b_m)$:

$$L \in \mathfrak{L}_0, \tag{15.9}$$

where $\mathfrak{L}_0$ is a given convex set (polyhedron).

It is needed to obtain an estimate of vector L using measured values of X_n and u_n and an *a priori* estimate in Eq. (15.9). Making use of Eqs. (15.7 and 15.8) gives

$$x_{m,n+1} = A_m^{\mathrm{T}} X_n + b_m u_n + f_n \tag{15.10}$$

or

$$x_{m,n+1} = L^{\mathrm{T}} Z_n + f_n \tag{15.11}$$

where

$$Z_n^{\mathrm{T}} = (X_n^{\mathrm{T}}, u_n), \quad L^{\mathrm{T}} = (A_m^{\mathrm{T}}, b_m). \tag{15.12}$$

Here $x_{m,n+1}$ is the mth component of vector X_{n+1}. With measured values of X_{n+1}, X_n and u_n, and taking into account Eqs. (15.3) and (15.11) one can obtain an estimate

$$\mathrm{L} \in \tilde{\mathfrak{L}}_{n+1} = \{L\colon \bigcup_{f_n \in \mathfrak{f}} Z_n^{\mathrm{T}} L = X_{m,n+1} - f_n\}. \tag{15.13}$$

If the set-valued estimate of L is available at the nth step, that is

$$L \in \mathfrak{L}_n, \tag{15.14}$$

then the result is an *a posteriori* estimate from Eqs. (15.13) and (15.14):

$$L \in \mathfrak{L}_{n+1} = \tilde{\mathfrak{L}}_{n+1} \cap \mathfrak{L}_n. \tag{15.15}$$

Thus, the problem of set-valued estimation of the parameter vector of dynamic system Eq. (15.7) has been reduced to a procedure identical to that for solving the problem of parameter identification of a plant without memory (see Eq. (15.1)).

Obtaining estimates of vector L using procedure Eqs. (15.13 and 15.15) is, in effect, equivalent to determining a set-valued solutions of a set of linear algebraic equations with uncertain right-hand sides.(16,24) Equation (15.11) can actually be written in the form

$$Z_N L = S_N - F_N, \tag{15.16}$$

where

$$Z_N = \left\| \begin{matrix} Z_0^{\mathrm{T}} \\ Z_1^{\mathrm{T}} \\ \vdots \\ Z_{N-1}^{\mathrm{T}} \end{matrix} \right\|, \quad S_N = \left\| \begin{matrix} X_{m,1} \\ X_{m,2} \\ \vdots \\ X_{m,N} \end{matrix} \right\|, \quad F_N = \left\| \begin{matrix} f_0 \\ f_1 \\ \vdots \\ f_{N-1} \end{matrix} \right\| . \tag{15.17}$$

Matrix Z_N and vector S_N are known exactly in Eq. (15.16). For vector F_N, one has only its *a priori* set-valued estimate

$$F_N \in \mathfrak{F}_N = \mathfrak{f} \times \mathfrak{f} \times \ldots \times \mathfrak{f}. \tag{15.18}$$

As shown,(16,24) the recurrent procedure of Eq. (15.15) is nothing else but for obtaining the whole set of solutions of a set of linear equations with uncertain right-hand sides projected onto *a priori* estimate $\mathfrak{L}_0$.

With additional noise in the measurement of vector X_n, considerable changes to the process of finding the whole set of solutions of a set of linear equations should be made. Indeed, let vector

$$X_n = \|x_{i,n}\|_{i=1}^{m} = \|x_{n-i}\|_{i=0}^{m-1}$$

and let X_n be measured with noise v_n, i.e., let

$$y_n = x_n + v_n, \tag{15.19}$$

where

$$v_n \in \mathfrak{v} \quad \forall n \geq 0, \quad \mathfrak{v} = \{v\colon \left| v \right| \leq \delta = const\}. \tag{15.20}$$

Then instead of Eq. (15.13), Eqs. (15.7 and 15.8) give

$$y_{m,n+1} = A_m^{\mathrm{T}} Y_n + b_m u_n + f_n - A_m^{\mathrm{T}} V_n - v_{m,n+1}, \tag{15.21}$$

where $y_{m,n+1}$ and $v_{m,n+1}$ are respective mth components of vectors

$$Y_n = \| y_{i,n} \|_{i=1}^{m} = \| y_{n-i} \|_{i=0}^{m-1}, \quad \text{and} \quad V_n = \| v_{i,n} \|_{i=1}^{m} = \| v_{n-i} \|_{i=0}^{m-1},$$

or

$$(Y_n - V_n)^{\mathrm{T}} A_m + u_n b_m = y_{m,n+1} + s_n, \tag{15.22}$$

where

$$s_n = f_n - v_{m,n+1}. \tag{15.23}$$

Only an *a priori* estimate

$$V \in \mathfrak{B} = \mathfrak{v} \times \mathfrak{v} \times \ldots \times \mathfrak{v}, \tag{15.24}$$

is given for vector V_n by virtue of Eq. (15.20). An estimate

$$s_n \in \mathfrak{S} = \mathfrak{f} + \mathfrak{v} \tag{15.25}$$

is given for variable s_n on the strength of Eqs. (15.3, 15.20, and 15.23), where a sum of sets is taken as a Minkowski sum. Thus the presence of a multiplicative member $Z_n^{\mathrm{T}} A_m$ by means of a procedure like Eq. (15.15), with observation Eq. (15.22), introduces valuable changes. In this case the whole procedure of Eq. (15.15) proves to be equivalent to determining a set-valued solution of a set of linear algebraic equations under uncertainty in their both sides.(16,24) This procedure is considered below in detail.

15.3. SIMULTANEOUS SET-VALUED ESTIMATION OF GUARANTEED ESTIMATES OF PARAMETER AND STATE VECTORS FOR LINEAR STATIONARY SYSTEMS

A designer of control systems often finds himself in a situation when rough *a priori* estimates of parameters of a controlled plant are given. Hence they need more exact definition. A state vector of a system is measured under noise which cannot be neglected. Consider this situation as applied to the class of dynamic system Eqs. (15.7) analyzed above. Entering necessary changes and alterations into its description. Thus, assume that Eq. (15.7) describes as before the motion of the dynamic system under consideration. Matrix A and vectors B and C of canonic structure and an *a priori* estimates given for matrix A and vector B are

$$A \in \mathfrak{A}_0, \quad B \in \mathfrak{B}_0, \tag{15.26}$$

where $\mathfrak{A}_0$ and $\mathfrak{B}_0$ are given convex sets.

Construct a sequence of estimates of state vector X_n and that of elements of matrix A and vector B using the result of Eq. (15.19) and *a priori* Eqs. (15.3, 15.20, and 15.26). A number of publications are devoted to obtaining a solution of a more simple problem, namely, to estimate states of a dynamic system with known parameters.(2,3,10,12)

If vector X_n is of the form

$$X_n = \left\| x_{i,n} \right\|_{i=1}^{m} = \left\| x_{n-i} \right\|_{i=0}^{m-1}, \tag{15.27}$$

then obtaining an estimate of vector X_n is reduced to that of estimation of its first component x_n. Thus, construct a sequence of set-valued estimates of vectors X_n and L. Make use of Eqs. (15.21) and (15.19), i.e.,

$$x_{n+1} = A_m^{\mathrm{T}} X_n + b_m u_n + f_n, \tag{15.28}$$

$$y_{n+1} = x_{n+1} + v_{n+1}, \tag{15.29}$$

with *a priori* estimates Eqs. (15.3) and (15.20) and

$$L = \left\| \begin{matrix} A_m \\ b_m \end{matrix} \right\| \in \mathfrak{L}_n, \tag{15.30}$$

$$\mathrm{X}_\mathrm{n} \in \mathfrak{X}_n \tag{15.31}$$

and known values u_n and y_{n+1}. Equation (15.28) determines prognostic estimate of value x_{n+1} in the form

$$x_{n+1} \in \mathfrak{x}_{n+1} = \bigcup_{\substack{x_n \in \mathfrak{x}_n, \\ L \in \mathfrak{L}_n, \\ f_n \in \mathfrak{f}}} (A_m^{\mathrm{T}} X_n + b_m u_n + f_n), \tag{15.32}$$

and Eq. (15.29) gives the estimate

$$x_{n+1} \in \tilde{\mathfrak{x}}_{n+1} = y_{n+1} - \mathfrak{v}. \tag{15.33}$$

Here, in the same manner as everywhere below, a sum of sets is taken as a Minkowski sum.

Using two noncontradictory Eqs. (15.32) and (15.33) gives

$$X_{n+1} \in \mathfrak{X}_{n+1} = \tilde{\mathfrak{X}}_{n+1} \cap \bar{\mathfrak{X}}_{n+1}. \tag{15.34}$$

Estimate $\mathfrak{X}_{n+1}$ is used further to obtain the estimates of vector L from Eq. (15.28), that is,

$$L \in \tilde{\mathfrak{L}}_{n+1} = \{L\colon \bigcup_{\substack{x_{n+1} \in \mathfrak{x}_{n+1} \\ f_n \in \mathfrak{f} \\ z_n \in \mathfrak{z}}} (Z_n^{\mathrm{T}} L + f_n - x_{n+1} = 0)\}. \tag{15.35}$$

Vector $Z_{n+1}^{\mathrm{T}} = (X_n^{\mathrm{T}}, u_n)$ introduced here is estimated with set $\mathfrak{z}$ obviously of the form

$$Z_n \in \mathfrak{z} = \mathfrak{X} \times u_n. \tag{15.36}$$

A posteriori estimate of vector L is determined with estimates Eqs. (15.30) and (15.36), that is

$$L \in \mathfrak{L}_{n+1} = \tilde{\mathfrak{L}}_{n+1} \cap \mathfrak{L}_n. \tag{15.37}$$

With calculations made by Eq. (15.37) the cycle of estimation of vectors L and X_n is completed.

Without dwelling on all the necessary details of application of general scheme presented here are some general comments. Clearly, the following relations between sets $\tilde{\mathfrak{X}}_{n+1}$ and $\bar{\mathfrak{X}}_{n+1}$ may take place in general:

1) $\bar{\mathfrak{X}} \subset \tilde{\mathfrak{X}}_{n+1}$;

2) $\tilde{\mathfrak{X}}_{n+1} \subset \bar{\mathfrak{X}}_n$; and

3) $\bar{\mathfrak{X}}_{n+1} \cap \tilde{\mathfrak{X}}_{n+1} \neq \varnothing$.

Thus, only the first and the third ones involve prognostic estimate $\bar{\mathfrak{X}}_{n+1}$. This allows one to obtain an estimate of vector X_n better than $\tilde{\mathfrak{X}}_{n+1}$ obtained as a result of measuring with noise. In the latter case, an estimate of parameters of a dynamic system is rough. Using prognostic estimate $\bar{\mathfrak{X}}_{n+1}$ one cannot refine estimate $\tilde{\mathfrak{X}}_{n+1}$. Thus, the quality of estimate $\mathfrak{L}_n$ has an immediate effect on the quality of estimate $\mathfrak{X}_{n+1}$ to be obtained.

To dwell more elaborately on some details of obtaining sets $\mathfrak{L}_{n+1}$ and $\mathfrak{X}_{n+1}$, determine interval set $\mathfrak{X}_{n+1}$ defined by Eq. (15.32), and use the following designations:

$$\bar{\mathfrak{X}}'_{n+1} = \bigcup_{\substack{A_m \in \mathfrak{A}_n^{(m)} \\ X_n \in \mathfrak{X}_n}} (A_m^{\mathrm{T}} X_n), \tag{15.38}$$

$$\bar{\mathfrak{X}}''_{n+1} = \bigcup_{b_m \in \mathfrak{b}_n^{(m)}} (b_m u_n), \tag{15.39}$$

where $\mathfrak{A}_n^{(m)} = P_A(\mathfrak{L}_n)$ is a projection of set $\mathfrak{L}_n$ onto a subspace of elements A_m; and $\mathfrak{b}_n^{(m)} = \mathrm{P_b}(\mathfrak{L}_n)$ is a projection of the same set onto Ob_m axis.

Designate also

$$\sup_{\substack{A_m \in \mathfrak{A}_n^{(m)};\\ X \in \mathfrak{X}_n}} \{A_m^{\mathrm{T}} X_n\} = \bar{\sigma}'_{n+1}, \tag{15.40}$$

$$\inf_{\substack{A_m \in \mathfrak{A}_n^{(m)};\\ X_n \in \mathfrak{X}_n}} \{A_m^{\mathrm{T}} X_n\} = \sigma'_{-n+1}, \tag{15.41}$$

To find values $\bar{\sigma}'_{n+1}$ and σ'_{-n+1}, state the following.[16,25]

Statement 15.1: If X_i and Y_j are vertices of arbitrary polyhedra $\mathfrak{X}$ and $\mathfrak{Y}$ respectively then

$$\sup_{\substack{X \in \mathfrak{X}\\ Y \in \mathfrak{Y}}} \{v = X^{\mathrm{T}} Y\} = \sup_{\substack{i \in \overline{1,N}\\ j \in \overline{1,M}}} \{\tilde{v}_{ij} = \tilde{X}_i^{\mathrm{T}} \tilde{Y}_j\} \tag{15.42}$$

and

$$\inf_{\substack{X \in \mathfrak{X}\\ Y \in \mathfrak{Y}}} \{v = X^{\mathrm{T}} Y\} = \inf_{\substack{i \in \overline{1,N}\\ j \in \overline{1,M}}} \{\tilde{v}_{ij} = \tilde{X}_i^{\mathrm{T}} \tilde{Y}_j\} \tag{15.43}$$

where $\tilde{X}_i$ and $\tilde{Y}_j$ are vertices of convex hulls $\tilde{\mathfrak{X}}$ and $\tilde{\mathfrak{Y}}$ of sets $\mathfrak{X}$ and and $\mathfrak{Y}$ respectively; N and M are the numbers of these vertices.

The proof of this statement is obvious enough and it is based on the properties of linear functional.

Let sets $\mathfrak{L}_n$ and $\mathfrak{X}_n$ be convex polyhedra defined in spaces of respective dimensions $\mathbb{R}^{m+1}$ and $\mathbb{R}^m$ defined by their vertices L_n^i, $i \in \overline{1,N_n}$ and X_n^j, $j \in \overline{1,M_n}$, where N_n and M_n are the numbers of vertices of $\mathfrak{L}_n$ and $\mathfrak{X}_n$, respectively, with given matrixes, their vertices

$$G_L^n = \|L_n^1, L_n^2, \ldots, L_n^{N_n}\|, \text{ and} \tag{15.44}$$

$$G_x^n = \|X_n^1, X_n^2, \ldots, X_n^{M_n}\|. \tag{15.45}$$

The same set of vertices for $\mathfrak{A}_n = P_A(\mathfrak{L}_n)$ has the form

$$G_A^n = \|A_n^1, A_n^2, \ldots, A_n^{S_n}\|. \tag{15.46}$$

Making use of Statement 15.1 gives

$$\overline{\sigma}'_{n+1} = \sup_{\substack{j=\overline{1,S_n}\\ i=\overline{1,M_n}}} \{(A^j_n)^{\mathrm{T}} X^i_n\}, \text{ and} \tag{15.47}$$

$$\underline{\sigma}'_{n+1} = \inf_{\substack{j=\overline{1,S_n}\\ i=\overline{1,M_n}}} \{(A^j_n)^{\mathrm{T}} X^i_n\}. \tag{15.48}$$

Hence set $\mathfrak{X}'_{n+1}$ can be determined now in the form

$$\mathfrak{X}'_{n+1} = \{x: \underline{\sigma}'_{n+1} \le x \le \overline{\sigma}'_{n+1}\}. \tag{15.49}$$

Set $\mathfrak{b}^{(m)}_n = P_b(\mathfrak{L}_n)$ is interval one, i.e.,

$$\mathfrak{b}^{(m)}_n = \{b: \underline{\mathfrak{b}}^{(m)}_n \le b \le \overline{\mathfrak{b}}^{(m)}_n\}$$

where $\underline{\mathfrak{b}}$ and $\overline{\mathfrak{b}}$ are numbers defined as a result of projecting set $\mathfrak{L}_n$ onto the $\mathrm{O}b_m$ axis. In accordance with Eq. (15.39) set $\mathfrak{X}''_{n+1}$ is obtained in the form

$$\overline{\mathfrak{X}}''_{n+1} = \{X: u_n \underline{b} \le x \le u_n \overline{b}\} \text{ at } u_n > 0, \text{ and} \tag{15.50}$$

$$\overline{\mathfrak{X}}''_{n+1} = \{X: u_n \overline{b} \le x \le u_n \underline{b}\} \text{ at } u_n \le 0. \tag{15.51}$$

On the strength of Eqs. (15.32), (15.38), and (15.39), $\overline{\mathfrak{X}}_{n+1}$ can be finally represented as

$$\overline{\mathfrak{X}}_{n+1} = \overline{\mathfrak{X}}'_{n+1} + \mathfrak{X}''_{n+1} + \mathfrak{f}, \tag{15.52}$$

where sets $\overline{\mathfrak{X}}'_{n+1}$ and $\overline{\mathfrak{X}}''_{n+1}$ are determined by Eqs. (15.49), (15.50), and (15.51), respectively.

Now dwell on obtaining set $\tilde{\mathfrak{L}}_{n+1}$ determined by Eq. (15.35). Parameter identification has known values of state vector X_n when set $\mathfrak{L}_{n+1}$ is represented by a hyperband in the parameter space. The case of uncertain state coordinates, unlike the case considered above, only when the inclusion $X_n \in \mathfrak{X}_n$ is to be used, is essentially more complicated. In this connection, consider in greater detail the problem of finding a set-valued solution of a set of linear algebraic equations under uncertainty mentioned above. With this aim in view, consider a set of linear algebraic equations in a standard notation

$$AX = B, \tag{15.53}$$

where X is l-dimensional vector to be determined, A is a rectangular matrix of $l \times N$ dimension in general, and B is n-dimensional vector. Assume that estimates for matrix A and vector B are given by

$$A \in \mathfrak{A}, \quad B \in \mathfrak{B} \tag{15.54}$$

where $\mathfrak{A}$ and $\mathfrak{B}$ are convex sets (polyhedra). An *a priori* estimate is also given for vector X

$$X \in \mathfrak{X}_0, \tag{15.55}$$

where $\mathfrak{X}_0$ is a convex set (polyhedron). Set-valued solution of Eq. (15.53) under Eqs. (15.54) and (15.55) is to be found.

Solving a set of linear algebraic equations is an ancient problem in mathematics. Existence of unavoidable errors in coefficients and in left-hand sides which are caused either by inaccuracy in initial data or by a finite accuracy of a computer, or by both the first and the second, leads to an uncertainty of a solution. Varying coefficients of a set of equations within the accuracy of their assignments, one can obtain different solutions and pretend that each one is equally true. But existing methods for solving a set of linear equations, such as least squares techniques and others, are oriented for obtaining the only (pointwise) solution. They might be considered as a particular illustration of "subjective aversion for problems not having a univalent answer."(26)

A wide range of problems have been specified in the last few years, particularly control, identification, and filtration under uncertainty of non-stochastic nature. For solving them, one has to obtain the whole set of solutions of a set of linear equations with uncertain values in both sides of each equation. Such problems are of interest to adherents of a new scientific branch called "interval mathematics."(27,28)

Definition 15.1. Set $\mathfrak{X}$ containing all the points $X \in \mathbb{R}^l$ satisfying Eq. (15.53) at each particular A and B from Eq. (15.55) is a set-valued solution of Eqs. (15.53) and 15.54).

Naturally, in the general case of having no additional conditions, all the points of set $\mathfrak{X}$ (if it is not empty) are equivalent in a sense that none of them can pretend to the role of the only "right" solution.

Obtaining set $\mathfrak{X}$ in the form

$$\mathfrak{X} = \mathfrak{f}(\mathfrak{A}, \mathfrak{B}) = \bigcup_{A \in \mathfrak{A}} \ \bigcup_{B \in \mathfrak{B}} f(A,B). \tag{15.56}$$

Let set $\mathfrak{B}$ be of the form

$$\mathfrak{B} = \bigcup_{r} \mathfrak{B}_r, \quad r \in \overline{1, N_B}, \tag{15.57}$$

where $\mathfrak{B}_r$ is a subset of $\mathfrak{B}$, N_B is the number of subsets $\mathfrak{B}_r$, and

$$\mathfrak{A} = \bigcup_{k} \mathfrak{A}_k \text{ and } k \in \overline{1, N_A}, \tag{15.58}$$

respectively, where $\mathfrak{A}_k$ is a subset of $\mathfrak{A}$, and N_A is a number of subsets $\mathfrak{A}_r$. Substituting Eqs. (15.57 and 15.58) into Eq. (15.56) gives

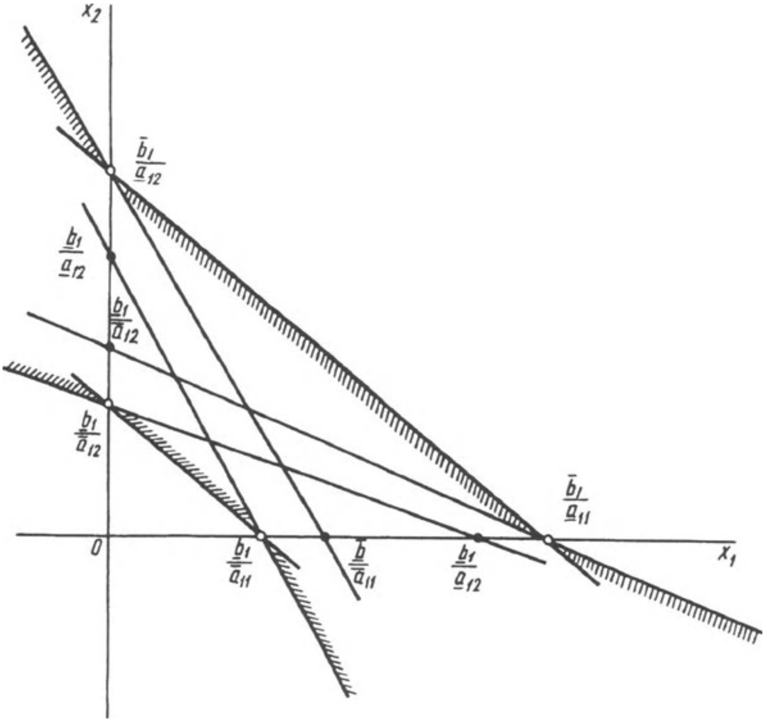
$$\mathfrak{X} = \bigcup_k \bigcup_r \mathfrak{X}_{kr}, \quad k \in \overline{1,N_A}, \quad r \in \overline{1,N_B}. \tag{15.59}$$

where

$$\mathfrak{X}_{kr} = \bigcup_{A \in \mathfrak{A}} \bigcup_{B \in \mathfrak{B}} f(A,B). \tag{15.60}$$

From Eqs. (15.59) and (15.60), the set $\mathfrak{X}$ to be found is a solution of a set of linear Equations (15.53) under Eq. (15.54), and in terms of Definition 15.1. It has a property called compositivity. Namely set $\mathfrak{X}$ can be obtained from Eq. (15.60) as a union of subsets. It means that determining set $\mathfrak{X}$ makes use of Eq. (15.57) and (15.58).

Consider solving Eq. (15.53) (in terms of Definition 15.1) for interval A and B:



x_2
$\frac{\bar{b}_1}{\underline{a}_{12}}$
$\frac{\underline{b}_1}{\underline{a}_{12}}$
$\frac{\underline{b}_1}{\bar{a}_{12}}$
$\frac{b_1}{\bar{a}_{12}}$
0
$\frac{b_1}{\bar{a}_{11}}$
$\frac{\bar{b}}{\bar{a}_{11}}$
$\frac{\underline{b}_1}{\underline{a}_{12}}$
$\frac{\bar{b}_1}{\underline{a}_{11}}$
x_1

FIGURE 15.1. Set of solutions of linear equation with $\underline{a}_{11} > 0$, $\underline{a}_{12} > 0$.

$$\underline{a}_{nj} \le a_{nj} \le \bar{a}_{nj}, \quad n \in \overline{1,N}, \quad j \in \overline{1,l}, \tag{15.61}$$

$$\underline{b}_n \le b_n \le \bar{b}_n, \tag{15.62}$$

where $\underline{a}_{nj}$, $\bar{a}_{nj}$, $\underline{b}_n$, and $\bar{b}_n$ are given values.

It is easy to verify that in a general case set $\mathfrak{X}$ is nonconvex. Indeed, consider one row of Eq. (15.53) with two-dimensional vector X, i.e., at $l = 2, N = 1$ (the two-dimensional case is suitable for geometric interpretation on a plane). Set $\mathfrak{X}$ of various forms is presented at Figs. 15.1, 15.2, and 15.3 for $\underline{b}_1 > 0$, and different values of a_{11} and a_{12} as a crosshatched region. Figure 15.1 corresponds to the case of $\underline{a}_{11} > 0, \underline{a}_{12} > 0$. Figure 15.2 corresponds to the case of $\underline{a}_{11} < 0, \bar{a}_{11} > 0$, and $\underline{a}_{12} > 0$. Figure 15.3 corresponds to the case of a $\underline{a}_{11} < 0, \bar{a}_{11} > 0, \underline{a}_{12} < 0$, and $\bar{a}_{12} > 0$.

Consider the sth orthant ($s = 1, \ldots, 2^l$) of space $\mathbb{R}^l$. Determine the set of indices $\{e_j^s\}$ for this orthant at $j = 1, \ldots, l$ as follows: $e_j^s = 0$ if the value of a component x_j of vector X is positive in this orthant, $e_j^s = 1$ otherwise. Then, diagonal matrix $G_s = \text{diag}\{e_1^s, \ldots, ke_l^s\}$ is characterized by the equation

$$(1 - 2G_s)X \ge 0. \tag{15.63}$$

For each s, introduce matrices $\underline{C}_s(.)$ and $\bar{C}_s(.)$ with coefficients determined as follows

$$\underline{C}_{nj}(s) = a_{nj}^{e_j^s}, \quad \bar{C}_{nj}(s) = a_{nj}^{1-e_j^s}, \quad n \in \overline{1,N}, \quad j \in \overline{1,l}. \tag{15.64}$$

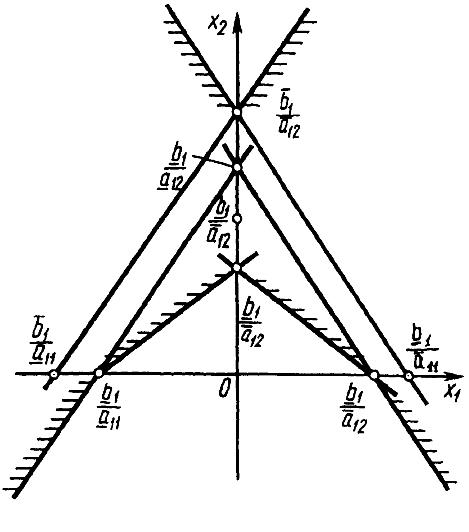

x_2
$\frac{\bar{b}_1}{\underline{a}_{12}}$
$\frac{\underline{b}_1}{\underline{a}_{12}}$
$\frac{\underline{b}_1}{\bar{a}_{12}}$
$\frac{\underline{b}_1}{\bar{a}_{12}}$
$\frac{\bar{b}_1}{\underline{a}_{11}}$
$\frac{\underline{b}_1}{\bar{a}_{11}}$
$\frac{\underline{b}_1}{\underline{a}_{11}}$
0
$\frac{\underline{b}_1}{\bar{a}_{12}}$
x_1

FIGURE 15.2. Set of solutions of linear equation with $\underline{a}_{11} < 0, \bar{a}_{11} > 0, \underline{a}_{12} > 0$.

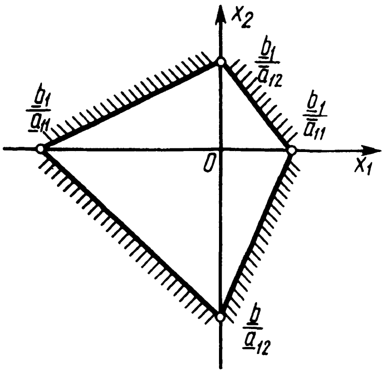

x_2
$\frac{\underline{b}_1}{\overline{a}_{12}}$
$\frac{\underline{b}_1}{\underline{a}_{11}}$
$\frac{\underline{b}_1}{\overline{a}_{11}}$
0
x_1
$\frac{\underline{b}}{\underline{a}_{12}}$

FIGURE 15.3. Set of solutions of linear equation with $\underline{a}_{11} < 0, \overline{a}_{11} > 0, \underline{a}_{12} < 0, \overline{a}_{12} > 0$.

Introduce vectors

$$\underline{B}^{\mathrm{T}} = (\underline{b}_1, \ldots, \underline{b}_N), \text{ and } \overline{B}^{\mathrm{T}} = (\overline{b}_1, \ldots, \overline{b}_N).$$

A set of linear equations

$$\overline{C}_s X \leq \overline{B},$$

$$\underline{C}_s X \leq \underline{B}, \tag{15.65}$$

combined with Eq. (15.33) separates out the set $\mathfrak{X}^s$, which makes it a convex polyhedron.

The following theorem is correct.

THEOREM 15.1. A set of solutions of Eq. (15.53) under Eq. (15.54) represented by Eqs. (15.61 and 15.62) should be the set

$$\mathfrak{X} = \bigcup_{s=1}^{2^l} \mathfrak{X}^s. \tag{15.66}$$

A proof of Theorem 15.1 can be found.[23,24]

Naturally, any particular subset $\mathfrak{X}^s$ may turn to be empty, i.e., the set of Eqs. (15.63) and (15.65) is contradictory for the sth orthant. Comparing Eqs. (15.53) and (15.65) (taking Eq. (15.64) in view), it is clear that each row from Eq. (15.53) produces two linear inequalities in the form of Eq. (15.65) in the respective orthant. All these pairs of inequalities are independent with each other. Subset $\mathfrak{X}^s$ can be reduced, subsequently adding respective pairs of inequalities and eliminating non-informative ones. With this aim in view rewrite Eq. (15.53) in the form

$$A_n^{\mathrm{T}} X = b_n, \quad n \in \overline{1,N}, \tag{15.67}$$

where A_n^{T} is the nth row of the matrix A. To each scalar equation from Eq. (15.69), set $\mathfrak{X}_n$ corresponds as follows: $X \in \mathfrak{X}_n$ such $A \in \mathfrak{A}_n$ and $b_n \in \mathfrak{b}$ exist that Eq. (15.69) is true for the taken n. According to Theorem 15.1, the set $\mathfrak{X}_n$ for any n can be presented in the form

$$\tilde{\mathfrak{X}}_n = \bigcup_{s=1}^{M} \tilde{\mathfrak{X}}_n^s, \quad M = 2, \tag{15.68}$$

where $\tilde{\mathfrak{X}}_n$ is a convex subset completely belonging to the sth orthant of space $\mathbb{R}^l$. It is separated out in this space by two scalar inequalities

$$\begin{aligned} &\overline{C}_{ns}^{\mathrm{T}} X - \underline{b}_n \geq 0, \\ &\underline{C}_{ns}^{\mathrm{T}} X - \overline{b}_n \leq 0, \end{aligned} \tag{15.69}$$

where $\underline{C}_{ns}^{\mathrm{T}}$ and $\overline{C}_{ns}^{\mathrm{T}}$ are the nth rows of matrices $\underline{C}_s$ and $\overline{C}_s$ respectively.

On the other hand, taking into account that *a priori* estimates for matrix A and vector B coefficients are independent of each other one can claim that

$$\mathfrak{X} = \tilde{\mathfrak{X}}_1 \cap \tilde{\mathfrak{X}}_2 \cap \ldots \cap \tilde{\mathfrak{X}}_N. \tag{15.70}$$

Therefore, set evolution equation is represented with

$$\mathfrak{X}_{n+1} = \tilde{\mathfrak{X}}_{n+1} \cap \mathfrak{X}_n, \quad \mathfrak{X}_1 = \tilde{\mathfrak{X}}_1, \quad n \in \overline{1,N}, \tag{15.71}$$

and

$$\mathfrak{X} = \mathfrak{X}_N. \tag{15.72}$$

From Eqs. (15.68) and (15.71) a set of independent difference equations are obtained:

$$\begin{aligned} &\mathfrak{X}_{n+1}^i = \tilde{\mathfrak{X}}_{n+1}^i \cap \mathfrak{X}_n^i, \quad i \in \overline{1,M}, \quad n \in \overline{1,N} \\ &\mathfrak{X}_1^i = \tilde{\mathfrak{X}}_1, \end{aligned} \tag{15.73}$$

which describe the evolution of convex polyhedra in each orthant separately (except for those orthants where these sets are empty). As this takes place, performing intersection of convex sets $\mathfrak{X}_n^i$ and $\mathfrak{X}_{n+1}^i$ remains similar to the methods described above, since one must intersect a polyhedron with Eq. (15.69) and reject the cut-off part only.

Given an *a priori* estimate $\mathfrak{X}_0$ of set $\mathfrak{X}$ is represented as a union of convex polyhedra $\mathfrak{X}_0^i$:

$$\mathfrak{X}_0 = \cup\, \mathfrak{X}_0^i, \quad i \in \overline{1,M}, \tag{15.74}$$

so it is appropriate and can be used in the recurrent procedure described above. With this aim in view, Eq. (15.73) should also be extended to the value $n = 0$. Substitute Eq. (15.74) for the initial condition $\mathfrak{X}_1^i = \tilde{\mathfrak{X}}_1^i$.

Examples illustrate the above method for constructing the set $\mathfrak{X}$ of solutions of the set of Eq. (15.53) under the given conditions of Eqs. (15.54) and (15.55).[11,23]

Now, extend the method suggested above to the case of state vector estimation when vector X_n is an arbitrary one. Equation (15.19) represents the measurements that should be substituted by the equation

$$Y_n = X_n + V_n, \tag{15.75}$$

where V_n is m-dimensional vector of noise with *a priori* set-valued estimate

$$V_n \in \mathfrak{B} \quad \forall\, n \geq 0. \tag{15.76}$$

Here $\mathfrak{B}$ is a given convex set (polyhedron).

The case under consideration differs from the one considered above from the only viewpoint. Use the estimate

$$X_{n+1} \in \tilde{\mathfrak{X}}_{n+1} = Y_{n+1} - \mathfrak{B} \tag{15.77}$$

instead of Eq. (15.32) and the following one instead of Eq. (15.33), therefore, on the strength of Eq. (15.28) it takes the form

$$X_{n+1} = \left\| \begin{matrix} \underline{X}_{n+1} \\ \cdots \\ \mathfrak{X}_{m,n+1} \end{matrix} \right\| \in \overline{\mathfrak{X}}_{n+1} = \left\| \begin{matrix} \underline{\mathfrak{X}}_{n+1} \\ \cdots \\ \mathfrak{r}_{m,n+1} \end{matrix} \right\|, \tag{15.78}$$

$$\underline{\mathfrak{X}}_{n+1} = \bigcup_{X_n \in \mathfrak{X}_n} (\underline{A} X_n), \text{ and} \tag{15.79}$$

$$\overline{\mathfrak{r}}_{m,n+1} = \bigcup_{\substack{A_m \in \mathfrak{A}_n^{(m)},\, X_n \in \mathfrak{X}_n \\ b_m \in \mathfrak{b}^{(m)},\, f_n \in \mathfrak{f}}} (A_m^{\mathrm{T}} X_n + b_m u_n + f_n). \tag{15.80}$$

A matrix $\underline{A}$ is obtained from the matrix A by deleting its last row, and $\underline{X}$ is a vector obtained from the vector X by deleting its last element.

All the other steps of the identification procedure applied for vectors L and X_n to obtain set-valued estimates $\mathfrak{L}_n$, $\mathfrak{X}_n$ remain with no change.

It is also easy to prove that if matrix A is not of a canonic form, nothing, in principal, should be changed (considering the general scheme described above).

15.4. OBTAINING SET-VALUED ESTIMATES OF STATE AND PARAMETER VECTORS FOR LINEAR NONSTATIONARY SYSTEMS

Let a class of linear nonstationary systems be stated by the difference equation

$$X_{n+1} = A_n X_n + B_n u_n + Cf_n, \tag{15.81}$$

where all the designations have the same sense as above; matrix A_n and vector B_n are unknown and vary arbitrarily in time elements, for which only certain *a priori* estimates are given. Clearly, with general suggestions on A_n and B_n one cannot obtain valuable results in solving the identification problem, and must restrict a subclass of plants for which a rate of change of the coefficients of A_n and B_n are bounded with known bounds.

Assume as above that A_n and B_n have a canonic form

$$A_n = \left\| \begin{matrix} 0 \vdots I_{m-1} \\ \cdots \\ A_{m,n} \end{matrix} \right\|, \quad B_n = \left\| \begin{matrix} 0 \\ \cdots \\ b_{m,n} \end{matrix} \right\|, \tag{15.82}$$

and *a priori* set-valued estimate $\mathfrak{L}_0^0$ for vector $L_0^{\mathrm{T}} = (A_{m,0}^{\mathrm{T}}, b_{m,0})$ is given by

$$L_0 \in \mathfrak{L}_0^0 \tag{15.83}$$

in the form of a convex set (polyhedron). In addition, assume the rate of change of the parameter vector $L_n^{\mathrm{T}} = (A_{m,n}^{\mathrm{T}}, b_{m,n})$ to be bounded, i.e.,

$$\|\Delta L_n = L_{n+1} - L_n\| \le \delta = \text{const.} \tag{15.84}$$

The vector norm is determined as

$$\|X\| = \max_{i=\overline{1,m}} |x_i|.$$

Equation (15.84) gives an *a priori* estimate for vector $\Delta L_n = R_n$

$$\Delta L_n = R_n \in \mathfrak{R} = \{R\colon \|R\| \le \delta\}. \tag{15.85}$$

Clearly, since Eq. (15.81) is nonstationary in L, the only alterations in the suggested general scheme deal with estimation of vector L_n. In this connection, a radically unremovable delay for stepwise operating discrete-time system exists in identification process or a nonstationary plant. Indeed, as follows from Eq. (15.81) with the value of vector X_n measured at the (n+1)th instant, this equation determines set $\tilde{\mathfrak{L}}_n^{n+1}$ as an *a posteriori* estimate of the parameter vector value at the previous instant. Vector L_n, jointly with the previous *a priori* estimate for the nth instant of time, is

$$L_n \in \mathfrak{L}_n^n, \tag{15.86}$$

and determines *a posteriori* estimate of vector L

$$L_n \in \mathfrak{L}_n^{n+1} = \tilde{\mathfrak{L}}_n^{n+1} \cap \mathfrak{L}_n^n. \tag{15.87}$$

Since solving particular problem at the (*n*+l)th instant estimate of vector L_{n+1} is required at the same time, solve an extrapolation problem using Eq. (15.87). It is necessary in one way or another to obtain an estimate of vector L_{n+1} at the (*n*+1)th instant of time

$$L_{n+1} \in \mathfrak{L}_{n+1}^{n+1}. \tag{15.88}$$

Clearly, Eq. (15.84) or (15.85) is the only source to obtain it. Since vectors L_n and ΔL_n are given only in terms of their set-valued estimates, it is clear that vector L_{n+1} can be estimated in the form

$$L_{n+1} \in \mathfrak{L}_{n+1}^{n+1} = \mathfrak{L}_n^{n+1} + \mathfrak{R}. \tag{15.89}$$

Looking at Eq. (15.89) one can find increasing volume of a set-valued estimate caused by the operation of summation of sets and decreasing volume caused by their intersection. Due to this fact, Eqs. (15.87) and (15.89) at $n \to \infty$ cannot be divergent. Nevertheless, one can give a rigorous statement and prove it.

Theorem 15.2. For parameter vector of discrete-time Eq. (15.81) satisfying the difference equation

$$L_n = L_{n-1} + \Delta L_{n-1},$$

where vector ΔL_{n-1} is bounded by Eq. (15.85). For vector L_{n-1} with its estimate $\mathfrak{L}_{n-1}^{n-1}$ given at the (n−1)th step with no disturbances and noise, i.e., at $\mathfrak{f} = \varnothing$ and $\mathfrak{B} = \varnothing$, the recursion

$$L_n \in \mathfrak{L}_n^n = \mathfrak{L}_n^{n-1} + \mathfrak{R},$$

where

$$L_{n-1} \in \mathfrak{L}_{n-1}^n = \tilde{\mathfrak{L}}_{n-1}^n \cap \mathfrak{L}_{n-1}^{n-1}, \text{ and}$$

$$\mathfrak{L}_{n-1}^n = \{L_{n-1}: \bigcup_{L_{n-1} \in \mathfrak{L}_{n-1}^{n-1}} (X_{n-1}^{\mathrm{T}} A_{m,n-1} + u_{n-1} b_{m,n-1} - x_{m,n} = 0)\}$$

with linearly-independent vectors $Z_{n-1}^{\mathrm{T}} = (X_{n-1}^{\mathrm{T}}, u_{n-1})$ determines a sequence of bounded sets $\mathfrak{L}_n^n$, i.e., the diameter $\delta(\mathfrak{L}_n^n) < \infty$ at $n \to \infty$.

In addition, taking expressions Eqs. (15.87) and (15.89) into account one can claim that sets $\mathfrak{L}_n^n$ are convex since the class of convex polyhedra is closed with respect to taking a sum of sets and intersecting.

15.5. CONCLUSIONS

The algorithms of simultaneous set-valued estimation of state and parameter vectors for linear dynamic systems have been described above. Making an assumption of a very general nature of the form of Eqs. (15.3) and (15.20) put on uncontrollable disturbances (noise), the procedures of parameter identification described above generally should be terminated with obtaining unimprovable estimates. The availability of such "residual" (unimprovable) uncertainty in parameter identification also provides the existence of generally unremovable uncertainty in state estimation.

For the class of nonstationary dynamic systems, which are unremovable in principle, uncertainty exists naturally. Nevertheless, this uncertainty remains bounded even at arbitrary large time interval.

The algorithms of set-valued estimation of state and parameter vectors can be applied to designing adaptive control systems of a wide class, particularly to unstable plants.

REFERENCES

1. F. C. Schweppe, *Uncertain Dynamic Systems*, Prentice-Hall, Englewood Cliffs, NJ (1973).
2. F. L. Chernousko and A. A. Melikjan, *Game Problems of Control and Search*, Nauka, Moscow, Russia (1973).
3. A. B. Kurzhanski, *Control and Observation Under Conditions of Uncertainty*, Nauka, Moscow, Russia (1977).
4. V. M. Kuntsevich and M. M. Lychak, *Autom. Remote Control* **1**, 77 (1979).
5. G. M. Bakan, *Sov. Autom. Control* **2**, 38 (1980).
6. M. Milanese and G. Belforte, *IEEE Trans. Autom. Control* **AC-27**, 408 (1982).
7. G. M. Bakan, *Autom. Remote Control* **9**, 81 (1980).
8. V. M. Kuntsevich and M. M. Lychak, *Synthesis of Optimal and Adaptive Control Systems: Game Approach*, Naukova dumka, Kiev, Russia (1985).
9. J. P. Norton, *Inter. J. Control* **45** 375 (1987).
10. F. L. Chernousko, *Estimation of the Phase State of Dynamic Systems*, Nauka, Moscow, Russia (1988).
11. V. M. Kuntsevich, M. M. Lychak and A. S. Nikitenko, in: *8th IFAC/IFORS Symposium*, Vol. 2, pp. 1237–1241, Beijing, P.R. China (1988).
12. A. B. Kurzhanski, *Identification Theory of Guaranteed Estimates*, IIASA Working Paper, Laxenburg, Austria (1989).
13. H. Piet-Lahanier and E. Walter, in: *Proceedings of the 28th IEEE Conference on Decision and Control* Tampa, FL (1989).
14. M. Milanese and A. Vicino, *Automatica* **27**, 403 (1991).
15. G. M. Bakan and N. N. Kussul, *Avtomatika* **5**, 11 (1989).
16. E. Walter and H. Piet-Lahanier, *Math. Comp. Sim.* **32**, 468 (1990).
17. G. M. Bakan and N. N. Kussul, *Avtomatika* **3**, 29 (1990).
18. D. C. N. Tse, M. A. Dahleh and I. N. Tsitsikeis, in: *Proceedings of the 1991 IEEE Conference on Decision and Control*, pp. 623–628, Brighton, United Kingdom (1991).
19. S. M. Veres and J. P. Norton, *Inter. J. Control* **50**, 639 (1989).

20. A. B. Kurzhanski and I. Valyi, in: *Nonlinear Synthesis, Progress in Systems and Control Theory* (Ch. I. Byrnes and A. B. Kurzhanski, eds.) Birkhauser, Boston, MA, pp. 184–196 (1991).
21. A. B. Kurzhanski, *Avtom. Telemekh.* **4**, 3 (1991).
22. M. Milanese and A. Vicino, *Automatica* **27**, 977 (1991).
23. V. M. Kuntsevich and M. M. Lychak, in: *Lecture Notes in Control and Information Sciences*, 196, Springer-Verlag, Berlin, Germany (1992).
24. V. M. Kuntsevich, M. M. Lychak, and A. S. Niktienko, *Kibernetika* **4**, 47 (1988).
25. V. M. Kuntsevich, *Dokl. AN SSR* **288**, 321 (1986).
26. R. E. Kalman, *Us. Mat. Nauk* **10**, 117 (1984).
27. R. E. Moore, *Interval Analysis*, Prentice-Hall, Englewood Cliffs, NJ (1966).
28. R. E. Moore, *Interval Analysis*, Prentice-Hall, Englewood Cliffs, NJ (1966).
29. V. M. Kuntsevich, *Avtom. Telemekh.* **2**, 79 (1980).
30. V. M. Kuntsevich and A. S. Nikitenko, *Kibernetika* **5**, 38 (1990).

APPENDIX

The proof of Theorem 15.2: It was already shown above that Eqs. (15.87 and 15.89) are equivalent to solving the respective set of linear algebraic equations under uncertainty in right-hand sides. This statement is correct also for the case of solving a set of linear equations under conditions of Theorem 15.2, based on the successive elimination of unknown variables. Indeed, a set of equations at $n \geq (m+1)$ is of the form

$$x_{m,n+1} = A^{\mathrm{T}}_{m,n}X_n + b_{m,n}u_n$$

and, respectively,

$$x_{m,n+1-k} = X^{\mathrm{T}}_n\left(A_{m,n} - \sum_{j=1}^{k} \Delta A_{m,n-1-j}\right) + \left(b_{m,n} - \sum_{j=1}^{k} \Delta b_{m,n-1-j}\right)u_n \quad \text{at } k < n.$$

Assuming $k = 1, 2, \ldots, m+1$, one obtains the following set of equations

$$x_{m,n} = X^{\mathrm{T}}_{n-1}\, A_{m,n-1} + b_{m,n-1}u_{n-1},$$

$$x_{m,n-1} = X^{\mathrm{T}}_{n-1}\left(A_{m,n-1} - \Delta A_{m,n-2}\right) + \left(b_{m,n-1} - \Delta b_{m,n-2}\right)u_{n-2},$$

$$\vdots$$

$$x_{m,n-m} = X^{\mathrm{T}}_{n-m-1}\left(A_{m,n-1} - \sum_{j=1}^{m} \Delta A_{m,n-1-j}\right) + \left(b_{m,n-1} - \sum_{j=1}^{m} \Delta b_{m,n-1-j}\right)u_{n-m-1}.$$

With the designations

$$\hat{X}^{\mathrm{T}}_n = (x_{m,n}, x_{m,n-1}, \ldots, x_{m,n-m-1}), \quad Z^{\mathrm{T}}_{n-1} = (X^{\mathrm{T}}_{n-1}, u_{n-1}),$$

$$\tilde{Z}_{n-1} = \begin{Vmatrix} Z^{\mathrm{T}}_{n-1} \\ Z^{\mathrm{T}}_{n-2} \\ \vdots \\ Z^{\mathrm{T}}_{n-m-1} \end{Vmatrix} \qquad Y_{n-1} = \begin{Vmatrix} 0 & 0 \\ \Delta A^{\mathrm{T}}_{m,n-2} & \Delta b_{m,n-2} \\ \vdots & \vdots \\ \sum\limits_{j=1}^{m} \Delta A^{\mathrm{T}}_{m,n-1-j} & \sum\limits_{j=1}^{m} \Delta b_{m,n-1-j} \end{Vmatrix}$$

rewrite them in the form

$$\hat{X}_n = \tilde{Z}_{n-1}(L_{n-1} - Y_{n-1}).$$

If det $\tilde{Z} \neq 0$ we obtain from above

$$L_{n-1} = \tilde{Z}^{-1}_{n-1}\hat{X}_n + \tilde{Z}^{-1}_{n-1}Y_{n-1}.$$

The exact values of vector $\Delta L_n = (\Delta A^{\mathrm{T}}_{m,n}, \Delta b_{m,n})$ are unknown, i.e., it is known only that $Y_{n-1} \in \mathfrak{Y}_{n-1}$. Hence only estimate $L_{n-1} \in \mathfrak{L}_{n-1}$ can be obtained. However, since set $\mathfrak{Y}_{n-1}$ is bounded, set $\mathfrak{L}_{n-1}$ is bounded at det $\tilde{Z}_{n-1} \neq 0$. Thus, estimates $\mathfrak{L}^{n}_{n}$ are bounded at $\forall n \geq 0$.

16

Limited-Complexity Polyhedric Tracking

H. Piet-Lahanier and É. Walter

ABSTRACT

When the errors between the data and model outputs are affine in the parameter vector $\boldsymbol{\theta}$, the set of all values of $\boldsymbol{\theta}$ such that these errors fall within known prior bounds is a polytope (under some identifiability conditions, which can be described exactly and recursively. However, this polytope may turn out to be too complicated for its intended use. In this chapter, an algorithm is presented for recursively computing a limited-complexity approximation guaranteed to contain the exact polytope. Complexity is measured by the number of supporting hyperplanes. The simplest polyhedric description that can thus be obtained is in the form of a simplex, but polyhedra with more faces can be considered as well. A polyhedric algorithm is also described for tracking time-varying parameters, which can accommodate both smooth and infrequent abrupt variations of the parameters. Both algorithms are combined to yield a limited-complexity polyhedric tracker.

H. PIET-LAHANIER • Direction des Études de Synthèse/SM Office National d'Etudes et de Recherches Aérospatiales F-92322, Châtillon Cedex, France. É. WALTER • Laboratoire des Signaux et Systèmes, CNRS-École Supérieure d'Électricité, 91192 Gif-sur-Yvette Cedex, France.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

16.1. INTRODUCTION

Bounded-error estimation(1–3) initially dealt with time-invariant models. When the error to be bounded is affine in the parameters $\boldsymbol{\theta} \in R^n$ to be estimated, the (posterior) feasible set $\mathbb{S}$ for $\boldsymbol{\theta}$ is a convex polyhedron. Assume for the sake of simplicity that this polyhedron is bounded (i.e., a polytope), which can be interpreted as an identifiability condition. This polytope can then be characterized exactly as the convex hull of its vertices. Proper book-keeping of the relations of adjacency between vertices (e.g., via lists of supporting hyperplanes) makes it possible to characterize $\mathbb{S}$ recursively.(4–6) Section 16.2 recalls an algorithm for that purpose, which forms the basis of the procedure to be used for limited-complexity polyhedric tracking. This exact description is often much simpler than might be feared (because a large number of inequalities do not contribute to the definition of the boundary of $\mathbb{S}$). It may nevertheless turn out to be too complicated for its intended use. This is the main motivation for attempting to approximate $\mathbb{S}$ by simpler sets guaranteed to contain it. Ellipsoids,(7–9) axis-aligned orthotopes(10) or generic parallelotopes(11) have been considered for that purpose. In Section 16.3, a modification of the exact polyhedric description is presented that allows a recursive determination of a limited-complexity polyhedron guaranteed to contain $\mathbb{S}$. The complexity of the polyhedron will be characterized by the number of its supporting hyperplanes. It is, for instance, possible to require the polyhedron to be a simplex.

In many practical applications, it is necessary to allow parameter variations to account for the unmodeled behavior of the system. Various extensions of the original techniques to time-varying parameters have been presented in the literature.(12,13) Most are derived from ellipsoidal-bounding algorithms. In Section 16.4, an algorithm for polyhedric tracking of time-varying parameters is described. Although mostly designed to follow smooth parameter variations, it can also accommodate infrequent abrupt changes of the parameter vector. It combines two algorithms recently proposed (14) and makes it possible to limit the complexity of the polyhedra obtained. Illustrative examples are presented in Section 16.5.

16.2. POLYHEDRIC DESCRIPTION FOR TIME-INVARIANT SYSTEMS

If the error $e(t, \boldsymbol{\theta})$ is affine in $\boldsymbol{\theta}$, it can be written as

$$e(t, \boldsymbol{\theta}) = \boldsymbol{\alpha}^{\mathrm{T}}(t)\boldsymbol{\theta} - \beta(t), \tag{16.1}$$

where $\boldsymbol{\alpha}(t)$ and $\beta(t)$ are known. In bounded-error estimation, one is interested in characterizing the posterior feasible set for the parameters, i.e., the set of all values of $\boldsymbol{\theta}$ such that the error satisfies

$$e_{\min}(t) \le e(t, \boldsymbol{\theta}) \le e_{\max}(t) = 1, 2, \ldots, N, \tag{16.2}$$

where the bounds $e_{\min}(t)$ and $e_{\max}(t)$ are known *a priori*. (For a situation where the bounds are not known, see Ref. 15.) From Eqs. (16.1) and (16.2), $\boldsymbol{\theta}$ must satisfy

$$\boldsymbol{\alpha}^{\mathrm{T}}(t)\boldsymbol{\theta} \ge \beta(t) + e_{min}(t), \tag{16.3}$$

$$\boldsymbol{\alpha}^{\mathrm{T}}(t)\boldsymbol{\theta} \le \beta(t) + e_{\max}(t), \tag{16.4}$$

which define a feasible strip $\mathbb{H}_t$ bounded by two parallel hyperplanes. It seems worth noting that the algorithms to be presented do not require these two hyperplanes to be parallel, so that they also apply to the type of pairs of linear inequalities obtained by the errors-in-variables approach. When $\boldsymbol{\theta}$ is assumed to be time-invariant, $\mathbb{S}$ is the intersection of N such feasible strips. If it is not empty, it is a convex polyhedron, which can be described exactly by enumerating its vertices and/or supporting hyperplanes. The determination of the solution set associated with Eqs. (16.3) and (16.4) can be performed recursively(4–6) by processing one inequality of type (16.3) or (16.4) at each iteration. Taking a new datum into account thus requires two iterations. The polyhedron $\mathbb{P}^k$ of all values of $\boldsymbol{\theta}$ consistent with the first k inequalities is described by the set of its vertices $\{\mathbf{v}_i^k\}$ completed for each of them by a list of its adjacent vertices $adj^k(i)$ and a list of its supporting hyperplanes $hyp^k(i)$.

The kth inequality to be taken into account can be written as

$$\boldsymbol{\alpha}_k^{\mathrm{T}}\boldsymbol{\theta} \ge b_k. \tag{16.5}$$

At iteration k, the intersection $\mathbb{P}^k$ of $\mathbb{P}^{k-1}$ with the new feasible half space

$$\mathbb{H}_k^+ = \{\boldsymbol{\theta} \mid \mathbf{a}_k^{\mathrm{T}}\boldsymbol{\theta} \ge b_k\} \tag{16.6}$$

is computed. Initialization is performed by defining a prior feasible polyhedron $\mathbb{P}^0$ described by its set of vertices $\{\mathbf{v}_i^0\}$ and associated $adj^0(i)$ and $hyp^0(i)$ lists. If no vertex of $\mathbb{P}^{k-1}$ satisfies

$$\mathbf{a}_k^{\mathrm{T}}\mathbf{v}_i^{k-1} - b_k \ge 0, \tag{16.7}$$

then the intersection is empty and the algorithm stops. If all vertices of $\mathbb{P}^{k-1}$ satisfy Eq. (16.7), then $\mathbb{P}^k = \mathbb{P}^{k-1}$ and the constraint is redundant. Otherwise, it is necessary to update $\mathbb{P}^{k-1}$ to $\mathbb{P}^k$ as follows. Let $\mathbf{v}_{k/k-1}$ be any vertex of $\mathbb{P}^{k-1}$ satisfying Eq. (16.6) and thus kept in $\mathbb{P}^k$. If $\mathbf{a}^{\mathrm{T}}\mathbf{v}_{k/k-1} = b_k$ then $\mathbb{H}_k = \{\boldsymbol{\theta} \mid \mathbf{a}_k^{\mathrm{T}}\boldsymbol{\theta} = b_k\}$ must be added to the list of supporting hyperplanes of $\mathbf{v}_{k/k-1}$ which otherwise remains unchanged.

Consider now the set of all vertices of $\mathbb{P}^{k-1}$ that are adjacent to $\mathbf{v}_{k/k-1}$. Any of these vertices that satisfies Eq. (16.5) also belongs to $\mathbb{P}^k$ and remains adjacent to $\mathbf{v}_{k/k-1}$. Any of these vertices that does not satisfy Eq. (16.5) is discarded and replaced by a new vertex located at the intersection of $\mathbb{H}_k$ with the edge linking $\mathbf{v}_{k/k-1}$ to the vertex discarded. This new vertex is obviously adjacent to $\mathbf{v}_{k/k-1}$ and to all other new vertices created from $\mathbf{v}_{k/k-1}$, which makes the updating of the list of adjacent vertices simple. The list of supporting hyperplanes for each vertex created from $\mathbf{v}_{k/k-1}$ is obtained by appending $\mathbb{H}_k$ to those supporting both $\mathbf{v}_{k/k-1}$ and the vertex discarded. All vertices of $\mathbb{P}^k$ are thus determined, together with their lists of supporting hyperplanes. All vertices of $\mathbb{P}^k$ that are in $\mathbb{P}^{k-1}$ also have a complete list of adjacent vertices, but the lists of adjacent vertices associated with the newly created vertices remain to be completed. This is performed by considering all pairs of newly created vertices originating from different vertices of $\mathbb{P}^{k-1}$ and comparing their list of supporting hyperplanes. Any pair of vertices that have at least $(n-1)$ hyperplanes in common and are such that no other vertex has a list of supporting hyperplanes containing these $(n-1)$ hyperplanes are adjacent. The polyhedron $\mathbb{P}^k$ is then obtained as a set of its vertices, with lists of adjacent vertices and supporting hyperplanes.

16.3. APPROXIMATE DESCRIPTION FOR TIME-INVARIANT SYSTEMS

Provided that the number n of the parameters to be estimated is not too large, $\mathbb{P}^{2N}$ is often surprisingly simple, even if the number of inequalities to be taken into account is quite large. A large number of inequalities turn out to be redundant. It is possible, however, that the complexity of the exact description obtained is too high for its intended use. It is then necessary to look for a coarser characterization. So far as the approximating set is guaranteed to contain the actual feasible set, it can be considered as an expansion of the actual feasible set, and therefore already confers on the algorithm some tracking ability. The method presented in this section aims at determining a series of limited-complexity polyhedra guaranteed to contain $\mathbb{S}$. The parameters to be estimated are still assumed to be time-invariant. The procedure is initialized by choosing as a prior feasible set some polyhedron $\mathbb{P}^0$ defined by at most n_h supporting hyperplanes. The simplest possible case is when $n_h = n + 1$, which corresponds to a polyhedron with $n + 1$ vertices, or simplex. Since any vertex of a simplex is adjacent to all others, the associated *adj* lists are trivial. Since any vertex belongs to all supporting hyperplanes of the simplex but one, the *hyp* list associated with any vertex can be replaced by the index of this supporting hyperplane. This allows one to further simplify the algorithm.

Let $\mathbb{L}^{k-1}$ be the limited-complexity polyhedron obtained when $k-1$ inequalities have been taken into account. The algorithm of Section 16.2 can be used to compute the intersection $\mathbb{P}$ of $\mathbb{L}^{k-1}$ and $\mathbb{H}_k^+$. If the number of faces of $\mathbb{P}$ is lower than or equal to n, then set $\mathbb{L}^k = \mathbb{P}$. Else compute the n_h polyhedra defined by n_h faces among the $n_h + 1$ faces of $\mathbb{P}$ and select the one with minimum volume as

$\mathbb{L}^k$. Note that this policy does not give the minimum-volume limited-complexity polyhedron guaranteed to contain $\mathbb{S}$. As for ellipsoidal approximation, a better approximation can be obtained by recirculating the inequalities in the algorithm. The computation of the optimal limited-complexity polyhedron guaranteed to contain $\mathbb{S}$ remains an open problem, except for some very restricted classes of polyhedra, such as axis-aligned boxes.

In the general case, the volume of polyhedra can be computed by the method described in Ref. 16. When the polyhedron is a simplex in an n-dimensional space, with vertices $\mathbf{v}_0,\mathbf{v}_1,\ldots,\mathbf{v}_n$, its volume V_n is easily obtained by the recursive formula[17]

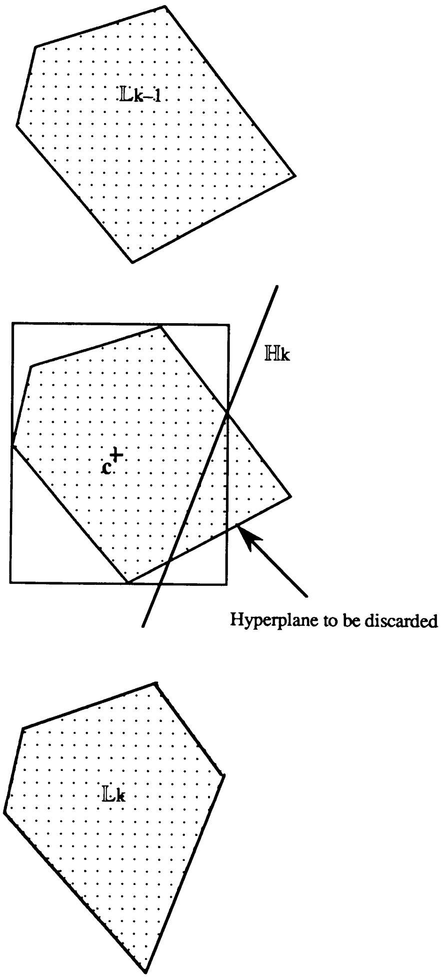


FIGURE 16.1. Suboptimal method for selecting the face to be discarded.

$$V_{\mathrm{n}} = \mathrm{h}V_{\mathrm{n-1}}/\mathrm{n}, \tag{16.8}$$

where h is the Euclidian distance between the vertex $\mathbf{v}_n$ and the $(n - 1)$-dimensional space containing the simplex with vertices $\mathbf{v}_0, \mathbf{v}_1, \ldots, \mathbf{v}_{n-1}$, and volume V_{n-1}.

When $n_h > n + 1$, the computation of the volume of each of the polyhedra candidates to become $\mathbb{L}^k$ may turn out to be too complex. An easier (but less effective) way of selecting the face to be discarded is as indicated on Fig. 16.1. Let $\mathbf{c}$ be the Chebyshev center of $\mathbb{P}$ for the L_∞-norm, with components given by

$$c_j = \frac{1}{2}[\min_i v_j(i) + \max_i v_j(i)], \tag{16.9}$$

where $\mathbf{v}(i)$ is the ith vertex of $\mathbb{P}$. Compute the distance to $\mathbf{c}$ of each supporting hyperplane $\mathbb{H}_j$ of $\mathbb{P}$, as

$$d(\mathbb{H}_j) = |\mathbf{a}_j^{\mathrm{T}}\mathbf{c} - b_j|/\|\mathbf{a}_j\|. \tag{16.10}$$

If the n_h hyperplanes with the smallest values of the distance are linearly independent, then set $\mathbb{L}^k$ is equal to the nondegenerate polytope defined by these faces by using the algorithm of Section 16.2. Otherwise, use the complete algorithm. Note that $\mathbb{L}^k$ is already partially obtained after $\mathbb{P}$ has been computed. From the lists of supporting hyperplanes associated with all vertices, it is easy to find the vertices of $\mathbb{P}$ belonging to $\mathbb{L}^k$.

16.4. POLYHEDRIC TRACKING FOR TIME-VARYING PARAMETERS

When the parameters are allowed to vary, the parameter set obtained from past observations should be modified to reflect the possible variations of $\boldsymbol{\theta}$ between past and present observations, before being intersected with the feasible strip defined by the two inequalities associated with the present observation. The method presented here(18) combines two algorithms proposed in Ref. 14. The polyhedron obtained from the previous measurements is first expanded, and the result is then intersected with $\mathbb{II}_t$. This involves using the basic intersection algorithm of Section 16.2 twice. If the intersection turns out to be empty, the expanded polyhedron is translated so as to move its Chebyshev center to the median hyperplane of the feasible strip associated with the present observation, which ensures a nonempty intersection even when some hypotheses of the method are not satisfied (e.g., when the bounds on the error are too optimistic or when the parameter variation is too abrupt for the expansion policy).

The algorithm is again initialized by choosing some prior feasible parameter domain $\mathbb{P}^0$ described by its set of vertices $\{\mathbf{v}_i^0\}$ and associated $adj^0(i)$ and $hyp^0(i)$

lists. Contrary to Section 16.2, a pair of inequalities are considered at each iteration, so that the current polyhedron at time t will be denoted by $\mathbb{P}^t$.

Let $\mathbf{c}^{t-1}$ be the Chebyshev center of $\mathbb{P}^{t-1}$ for the L_∞-norm. When no specific information is available on the possible speed of variation of each parameter, expansion of $\mathbb{P}^{t-1}$ may be performed by replacing each of its vertices $\mathbf{v}_i^{t-1}$ by ${}^{e}\mathbf{v}_i^{t-1}$, defined by

$${}^{e}\mathbf{v}_t^{t-1} = \mathbf{v}_i^{t-1} + \lambda\left(\mathbf{v}_i^{t-1} - \mathbf{c}^{t-1}\right), \tag{16.11}$$

where $\lambda > 0$ is some scalar expansion factor (vaguely similar to the forgetting factor of recursive least squares, although the analogy should not be pushed too far). The resulting expanded polyhedron ${}^{e}\mathbb{P}^{t-1}$ is then intersected with the feasible strip $ℿ^t$ associated with the measurement at time t. If the resulting intersection is empty, then ${}^{e}\mathbb{P}^{t-1}$ is translated orthogonally to the median hyperplane $\mathbb{H}$ of $ℿ^t$ so that its Chebyshev center lies on $\mathbb{H}$. Let $\mathbb{H}^-$ and $\mathbb{H}^+$ be the two (possibly non-parallel) hyperplanes limiting $ℿ^t$.

$$\mathbb{H}^- = \{\boldsymbol{\theta}: \boldsymbol{\alpha}_-^{\mathrm{T}}\boldsymbol{\theta} = \beta_-\}, \tag{16.12}$$

$$\mathbb{H}^+ = \{\boldsymbol{\theta}: \boldsymbol{\alpha}_+^{\mathrm{T}}\boldsymbol{\theta} = \beta_+\}. \tag{16.13}$$

The median hyperplane is then given by

$$\mathbb{H} = [\boldsymbol{\theta}: (\boldsymbol{\alpha}_-^{\mathrm{T}} + \boldsymbol{\alpha}_+^{\mathrm{T}})\boldsymbol{\theta} = \beta_- + \beta_+]. \tag{16.14}$$

Any vertex ${}^{e}\mathbf{v}_i^{t-1}$ of ${}^{e}\mathbb{P}^{t-1}$ is translated according to

$${}^{et}\mathbf{v}_i^{t-1} = {}^{e}\mathbf{v}_i^{t-1} + \mu(\boldsymbol{\alpha}_- + \boldsymbol{\alpha}_+), \tag{16.15}$$

with

$$\mu = \frac{\beta_- + \beta_+ - (\boldsymbol{\alpha}_- + \boldsymbol{\alpha}_+)^{\mathrm{T}}\mathbf{c}^{t-1}}{(\boldsymbol{\alpha}_- + \boldsymbol{\alpha}_+)^{\mathrm{T}}(\boldsymbol{\alpha}_- + \boldsymbol{\alpha}_+)}, \tag{16.16}$$

which ensures that the Chebyshev center of the translated polyhedron ${}^{et}\mathbb{P}^{t-1}$ belongs to $\mathbb{H}$. In the special case where $\mathbb{H}^-$ and $\mathbb{H}^+$ are defined by Eqs. (16.3) and (16.4) and where $e_{\min} = -e_{\max}$, this policy simplifies into that proposed in Ref. 18.

This algorithm has several advantages over those described in Ref. 13. First, it is recursive. Second, its expansion policy modifies neither the relationships of adjacency between vertices nor the directions of supporting hyperplanes. The *adj* and *hyp* lists, therefore, need not be modified after the expansion phase. Moreover, as the relative distance between two adjacent vertices of the polyhedron increases, adjacent vertices can never merge, so that degeneration of faces can not occur.

When more information is available on the dynamics of the components of $\boldsymbol{\theta}$, it can be taken into account by replacing Eq. (16.11) by a similar expression, where each component of the expanded vertices has its own expansion coefficient, i.e.,

$$^{e}v_i^{t-1}(j) = v_i^{t-1}(j) + \lambda_j[v_i^{t-1}(j) - c^{t-1}(j)], j = 1, \dots, n. \tag{16.17}$$

The expansion factor along the jth axis $\lambda_j \geq 0$ must be chosen *a priori*. The larger λ_j, the faster the jth component of $\boldsymbol{\theta}$ can vary. If any component of $\boldsymbol{\theta}$ is assumed to be time-invariant, one may choose the associated expansion factor equal to zero. If the variation of the jth parameter between two measurements is assumed to be less than $vmax_j$, then a time-varying λ_j may be chosen as

$$\lambda_j(t) = vmax_j \left[\frac{2}{\max\limits_i(v_i^{t-1}(j)) - \min\limits_i(v_i^{t-1}(j))} \right]. \tag{16.18}$$

The relationships of adjacency are not altered after an expansion according to Eq. (16.18).[14] The overall description of the polyhedron in terms of lists of supporting hyperplanes, therefore, does not need to be modified, but the faces of the expanded polyhedron are no longer parallel to the initial ones.

The expansion-translation algorithm used for polyhedric tracking is easily combined with the approximate description of Section 16.3 to yield a limited-complexity polyhedric tracker.

16.5. EXAMPLE

One thousand data points have been generated by simulating the ARX system

$$\begin{aligned} y_t &= 0.1\, y_{t-1} + 0.72\, y_{t-2} \\ &+ \left[-0.5 + \frac{t-1}{500} \right] u_t - 1.8\, u_{t-1} + \varepsilon_t, t = 1, \dots, 300, \end{aligned} \tag{16.19}$$

and

$$\begin{aligned} y_t &= -1.7\, y_{t-1} - 0.72\, y_{t-2} \\ &+ \left[-0.5 + \frac{t-1}{500} \right] u_t - 1.8\, u_{t-1} + \varepsilon_t, t = 301, \dots, 1000, \end{aligned} \tag{16.20}$$

with ε_t a sequence of independent random variables uniformly distributed between −0.1 and 0.1. The input u alternates sequences of fifty identical values ±1. The initial conditions are

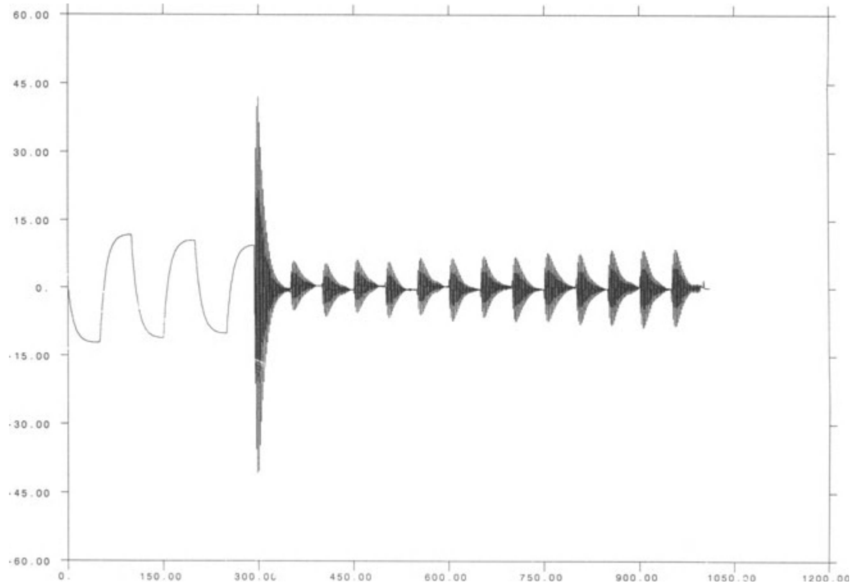


FIGURE 16.2. Data for the example.

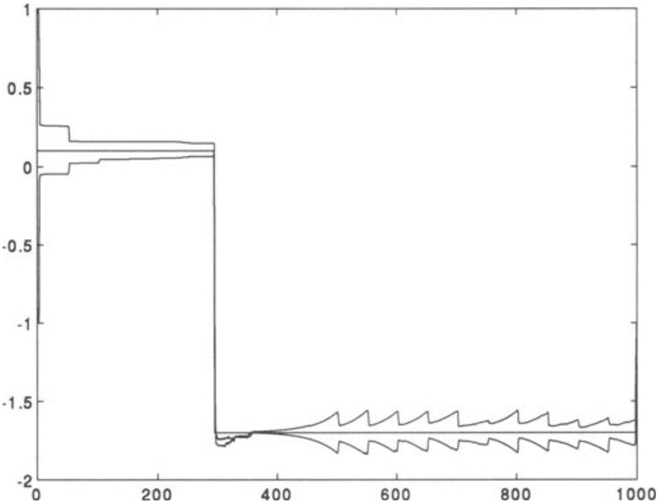


FIGURE 16.3. Evolution of the true value and estimated parameter uncertainty interval for θ_1.

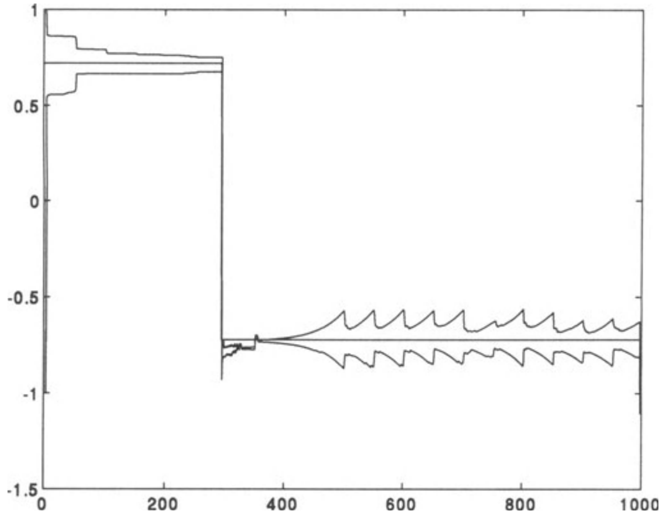


FIGURE 16.4. Evolution of the true value and estimated parameter uncertainty interval for θ_2.

$$y_0 = y_{-1} = 0. \tag{16.21}$$

Fig. 16.2 presents the data obtained, which are used to estimate the parameters of the model

$$y_m(t, \boldsymbol{\theta}) = \theta_1 y_{t-1} + \theta_2 y_{t-2} + \theta_3 u_t + \theta_4 u_{t-1}. \tag{16.22}$$

From the equations used to generate the data, it can be seen that the true values of θ_1 and θ_2 jump at $t = 300$, while the true value of θ_3 is slowly varying and that of θ_4 remains constant. The prior feasible set for the parameters is a simplex, large enough to be guaranteed to contain the true value for the parameter vector. At each iteration, the number of supporting hyperplanes is limited to $n_h = 5$. The limited-complexity polyhedric tracker is used for that purpose, with the suboptimal procedure of Figure 16.1 for selecting the face to be discarded in any polyhedron with more than five supporting hyperplanes. The expansion coefficients are given by $\lambda = 0.1$ for θ_3 and $\lambda = 0.05$ for all other parameters. To be considered as feasible, the parameters must satisfy

$$|\mathrm{y}_t - y_m(t, \boldsymbol{\theta})| \leq 0.1. \tag{16.23}$$

Figs. 16.3 to 16.6 give the evolution, with the number of data points taken into account, of the four parameter uncertainty intervals obtained by projecting the simplex on the axes of the parameter space. Before the jump at $t = 300$ occurs, the

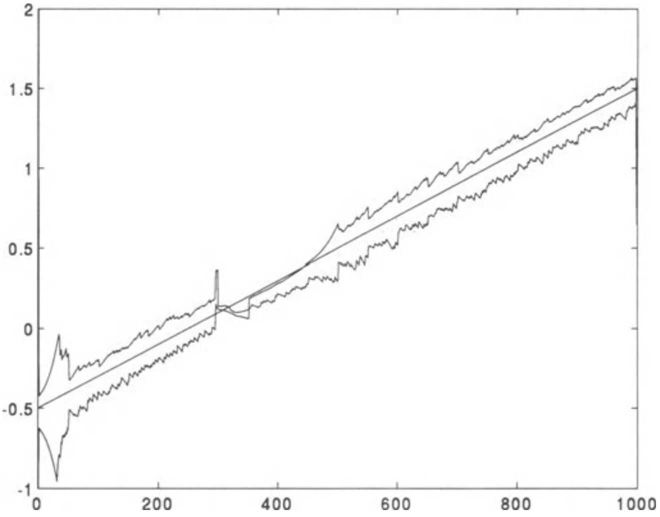


FIGURE 16.5. Evolution of the true value and estimated parameter uncertainty interval for θ_3.

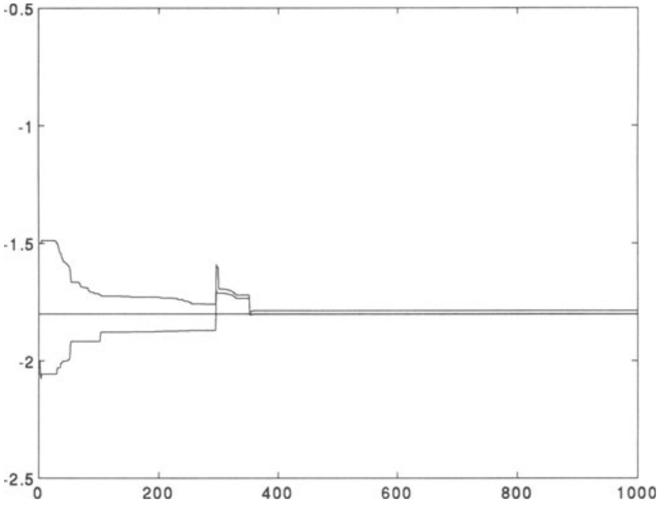


FIGURE 16.6. Evolution of the true value and estimated parameter uncertainty interval for θ_4.

expanded polyhedron is never translated. The translation is performed as soon as the feasible simplex becomes empty. This policy makes it possible to recover correct parameter uncertainty intervals in about 100 measurements and no further translation is needed. The same example treated with an exact description instead of a simplex gives figures very similar to those presented here. Examples involving 10 parameters and 1000 data points have also been treated using a simplex approximation. The corresponding exact polyhedron soon becomes intractable.

16.6. CONCLUSIONS

Ellipsoidal outer bounding of feasible parameter sets has long been thought of as the only viable option when the number of parameters to be estimated was large. The polyhedric approach can now also be considered for large-scale problems, because of the availability of methods for limiting the complexity of the resulting descriptions. The limited-complexity approach advocated here is only one among many that can be considered, but corresponds to a large class of algorithms. It can be combined with various expansion policies to allow the tracking of time-varying parameters. The recursive expansion policy described in this chapter makes the updating of the lists describing adjacency and supporting hyperplane relationships trivial. It also ensures that no degeneration of faces can occur. By tuning individual expansion factors along each axis, it is possible to take bounds on speeds of variation of the parameters into account. Rare abrupt changes of parameters that cannot be accounted for by the expansion policy chosen are taken care of by a translation of the expanded polyhedron so that its Chebyshev center lies on the median plane of the feasible strip associated with the new datum.

REFERENCES

1. F. C. Schweppe, *IEEE Trans. Automat. Control* **13**, 22 (1968).
2. E. Walter and H. Piet-Lahanier, *Math. and Comput. in Simul.* **32**, 449 (1990).
3. M. Milanese and A. Vicino, *Automatica* **27**, 997 (1991).
4. V. Broman and M. J. Shensa, *Math. and Comput. in Simul.* **32**, 469 (1990).
5. E. Walter and H. Piet-Lahanier, *IEEE Trans. Autom. Contr.* **34**, 911 (1989).
6. S. H. Mo and J. P. Norton, *Math. and Comput. in Simul.* **32**, 481 (1990).
7. E. Fogel and Y.-F. Huang, *Automatica* **18**, 229 (1982).
8. G. Belforte, B. Bona, and V. Cerone, *Automatica* **26**, 887 (1990).
9. J. R. Deller, *IEEE Trans. Acoustic Speech and Signal Processing* **37**, 1432 (1989).
10. M. Milanese and G. Belforte, *IEEE Trans. Automat. Control* **27**, 408 (1986).
11. A. Vicino and G. Zappa, in: *Proc. Workshop on the Modeling of Uncertainty in Control Systems* (M. Dahleh, J. Doyle, and R. Smith, eds.) Springer-Verlag, Berlin, Germany (1992).
12. S. Dasgupta and Y. F. Huang, *IEEE Trans. Informat. Theory*, **33**, 383 (1987).
13. J. P. Norton and S. H. Mo, *Math. Comput. and Simul.* **32**, 527 (1990).

14. H. Piet-Lahanier and E. Walter, in: *31st IEEE Conf. on Decision and Contr.*, pp. 66–67 (1992); *IEEE Trans. on Autom. Control* **39**, 1661 (1994).
15. E. Walter and H. Piet-Lahanier, in: *Bounding Approaches to System Identification* (M. Milanese *et al.*, eds.), Plenum Press, New York, Chap. 12 (1996).
16. J. B. Lasserre, *J. of Optimization Theory and Applications* **39**, 363 (1983).
17. M. Berger, *Géométrie*, Tome 3, Cedic/Nathan, Paris, p. 95 (1979).
18. H. Piet-Lahanier and E. Walter, in: *Proc. IEEE Int. Symp. on Circuits and Systems*, pp. 782–785 (1993).

17

Parameter-Bounding Algorithms for Linear Errors-in-Variables Models

S. M. Veres and J. P. Norton

ABSTRACT

Computational techniques are considered for the errors-in-variables (EIV) problem with bounds specified on the errors in all variables. The significant difference in difficulty in bounding the parameters of a dynamic EIV model, compared with the static case, is explained. Conditions for the feasible set of the parameters to be the union of polytopes are discussed, and a search technique to find the nonlinear bounds for the dynamic EIV problem is described. A simulation example compares EIV and equation-error bounding. Techniques for shortening the computation of EIV parameter bounds, and for finding polytope and ellipsoid approximations, are given.

17.1. INTRODUCTION

The EIV problem is that of estimating parameters in a linear-in-parameters model when some or all explanatory variables, as well as the output, are uncertain

S. M. Veres and J. P. Norton • School of Electronic and Electrical Engineering, University of Birmingham, Edgbaston, Birmingham B15 2TT, United Kingdom.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

(noisy). It has been extensively discussed in the statistical literature.[(1–4)] Recently dynamic EIV models have had attention.[(5–9)] Most of this work assumes that the errors are statistically specified. Following the appearance of deterministic parameter-bounding algorithms,[(10–18)] it is of interest to consider the EIV problem in a deterministic parameter-bounding context. Statistical assumptions on the errors (e.g., uncorrelatedness or restrictions on distribution) are replaced by the single requirement that the errors should lie within specified bounds. The bounded-error case can be considered as a special case of statistical modeling, with independent and uniformly distributed errors. However, such a view is not necessary and may complicate motivation, interpretation or analysis.

The EIV parameter-bounding problem can be formulated as computing bounds on the p-vector θ of parameters in the model

$$x_t = f(\phi_t, \theta) + e_t \quad t = 1,2,\ldots,N \tag{17.1}$$

where f is a known function, scalar x_t is known to be within ε_x of its observed value x_t^o. The error in x_t^o is bounded by $|\tilde{x}_t| \equiv |x_t - x_t^o| \le \varepsilon_x$, the errors $\tilde{\phi}_t \equiv \phi_t - \phi_t^o$ in observations ϕ_t^o of the q-vector $\phi_t \equiv [\phi_t^1 \ldots \phi_t^q]^{\mathrm{T}}$ of explanatory variables are bounded by $|\tilde{\phi}_t^i| \le \varepsilon_\phi^i$, i = 1,2,...,q and equation error e_t (structural error, due to linearization, reduction of the model order or omission of a significant term) is bounded by $|e_t| \le \varepsilon_e$. The EIV problem is called linear if f is linear in both θ and ϕ_t, so that

$$x_t = f^{\mathrm{T}}(\phi_t)\theta + e_t \quad t = 1,2,\ldots,N \tag{17.2}$$

with f a known vector function. More symmetrically,

$$f'^{\mathrm{T}}(\phi'_t)\theta' = -e_t \quad t = 1,2,\ldots,N \tag{17.3}$$

where

$$\phi'_t \equiv [x_t \ \phi_t^{\mathrm{T}}]^{\mathrm{T}}, \ \theta' \equiv [-1 \ \theta^{\mathrm{T}}]^{\mathrm{T}}$$

and f is correspondingly augmented to f'. One knows that $f'(\phi'_t)$ is in the set

$$\mathcal{F}_t = \{f'(\phi'_t) \mid |\tilde{x}_t| \le \varepsilon_x, |\tilde{\phi}_t^i| \le \varepsilon_\phi^i, i = 1,\ldots,q\} \tag{17.4}$$

and the feasible parameter set (FPS) after processing all data up to time N is

$$\mathcal{D}_N \equiv \{\theta \mid |f'^{\mathrm{T}}(\phi'_t)\theta'| \le \varepsilon_e, f'^{\mathrm{T}}(\phi'_t) \in \mathcal{F}_t, t = 1,\ldots,N\} \tag{17.5}$$

The problem is dynamic if consecutive vectors ϕ are related (deterministically), for instance, by a sample of a variable appearing in successive ϕs; otherwise it is static. If all errors are treated as part of e_t, the problem simplifies to the standard equation-error problem but with complicated bounds on e_t. In a statistical framework, lumping all the errors into e_t makes it correlated with the observed explanatory variables, which causes bias in standard estimators such as least squares.(3)

Section 17.2 shows how identification of dynamic EIV models relates to identification of equation-error and static models, and illustrates by an example that the dynamic bounding problem is much less straightforward than the static problem. In Section 17.3, conditions for the parameter bounds to be the union of a set of polytopes are discussed. A search technique to find the nonlinear parameter bounds for the dynamic EIV problem is described, and a simulation example compares EIV and equation-error bounding. Section 17.4 presents techniques for shortening the computation of EIV parameter bounds, and Section 17.5 describes algorithms for computing polytope and ellipsoidal parameter bounds for the EIV problem.

17.2. BOUNDING IN DYNAMIC EIV MODELS

In dynamic models, the set $\mathcal{F} = \mathcal{F}_1 \times \mathcal{F}_2 \times \ldots \mathcal{F}_N$ defining the uncertainty in the variables is reduced in dimension by the relations between successive ϕ's. An example is the ARX MISO model

$$
\begin{aligned}
y_t &= a_1 y_{t-1} + \ldots + a_k y_{t-k} \\
&\quad + \sum_{i=1}^{m} (b_1^i u_{t-d-1}^i + \ldots + b_l^i u_{t-d-l}^i) + e_t \quad t = 1, \ldots, N
\end{aligned}
\tag{17.6}
$$

Each successive sample of y is known within bounds $|y_{t-j} - y_{t-j}^o| \le \varepsilon_y$. Each sample of u^i is known within $|u_{t-j}^i - u_{t-j}^{oi}| \le \varepsilon_u^i$, and equation error e_t is bounded by $|e_t| \le \varepsilon_e$. Although

$$
\phi'_t \equiv [y_t \;\; y_{t-1} \cdots y_{t-k} \;\; u_{t-d-1}^1 \ldots u_{t-d-1}^m \ldots u_{t-d-l}^1 \cdots u_{t-d-l}^m]^{\mathrm{T}}
$$

and

$$
\theta' \equiv [-1 \;\; a_1 \ldots a_k \;\; b_1^1 \ldots b_1^m \ldots b_l^1 \ldots b_l^m]^{\mathrm{T}}
$$

have $k + 1 + ml$ elements, the model embodies altogether only y_{1-k} to y_N and u_{1-d-l}^i to u_{N-d-1}^i, $i = 1, \ldots, m$, so $\mathcal{F}$ is of dimension $N(m + 1) + k - 2m + ml$, not $N(k + 1 + ml)$.

The exact parameter bounds in a linear, equation-error model are all hyperplanes and form a polyhedron (a polytope so long as the normals span the space). It is often readily computed in realistic cases, but may be inconveniently complicated. If so, an outer bound such as an ellipsoid(12) or a box(13) may be computed cheaply. The bounds are more complicated in the EIV case, but in the linear *static* case, which is how the problem has been treated so far,(19,20) they remain piecewise linear. The relations between successive ϕ's in the *dynamic* case make the exact parameter bounds deviate from piecewise linearity, greatly increasing the difficulty of computing them, exactly or approximately. To illustrate, Example 1 considers just two successive sampling instants for a 1st-order ARX model:

$$y_t = ay_{t-1} + bu_{t-1} + e_t$$

$$y_{t+1} = ay_t + bu_t + e_{t+1} \tag{17.7}$$

Here y_t is common to both equations, appearing in both ϕ'_t and ϕ'_{t+1}. Figure 17.1 shows bounds on $\theta \equiv [a\ b]^{\mathrm{T}}$ for $\varepsilon_y = 0.4$, $\varepsilon_u = 0.4$, $\varepsilon_e = 0.1$ and $a = 0.6$, $b = -0.9$, due to observations $u^o_{t-1} = 1$, $u^o_t = 2$, $y^o_{t-1} = 2$, $y^o_t = 0.3$, $y^o_{t+1} = -1.62$. The FPS $\mathcal{D}$ is the intersection of a family of quadrilaterals generated as $\tilde{y}_t$ varies over its range[–0.1,0.7]; each is given by the four inequalities

$$-y^o_t - \varepsilon_e \le \tilde{y}_t - (\phi^o_t + \tilde{\phi}_t)^{\mathrm{T}}\theta \le -y^o_t + \varepsilon_e \tag{17.8}$$

$$-y^o_{t+1} - \varepsilon_e \le \tilde{y}_{t+1} - (\phi^o_{t+1} + \tilde{\phi}_{t+1})^{\mathrm{T}}\theta \le -y^o_{t+1} + \varepsilon_e \tag{17.9}$$

Fig. 17.1 shows 20 such quadrilaterals, and reveals that $\mathcal{D}$ has nonlinear boundaries. The reason is the presence of $\tilde{y}_t$ both in $a\tilde{y}_t$ within $\tilde{\phi}^{\mathrm{T}}_{t+1}\theta$ in Eq. (17.9) and on its own in Eq. (17.8). In the space of $(\tilde{y}_t, \theta)$, Eq. (17.8) gives hyperplane bounds which intersect hyperbolic-section bounds due to Eq. (17.9). The intersections are curves *projecting* to curved bounds in (θ_1,θ_2)-space, even though any (θ_1,θ_2) *cross section* of the bounds in $(\tilde{y}_t, \theta)$-space is a quadrilateral. Other terms such as $b\tilde{u}_{t-1}$ have no such effect, since the corresponding curved bounds intersect only hyperplane bounds at the extreme values of the uncertain variable, and those intersections are linear. Clearly, nonlinear bounds on θ will occur whenever a sample of an uncertain variable appears more than once in the model and at least once in a nonlinear combination with a parameter. The dashed lines in Fig. 17.1 show the much larger

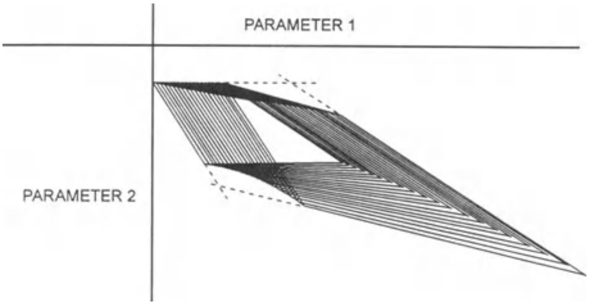


FIGURE 17.1. Example 1: nonlinear parameter bounds for the linear model (17.7).

$\mathcal{D}$, a polytope, obtained if Eqs. (17.7) are treated as independent, i.e., if the dynamic problem is treated as static.

17.3. POLYTOPE BOUNDS ON THE PARAMETERS OF EIV MODELS

The feasible parameter set $\mathcal{D}_N$ defined by Eq. (17.5) may be written as

$$\mathcal{D}_N \equiv \mathcal{E}_N/\mathcal{F} \equiv \{\theta \,|\, f'^{\mathrm{T}}(\phi'_t)\theta' \in \mathcal{E}_t \text{ and} f'(\phi'_t) \in \mathcal{F}_t, \quad t = 1,2,\ldots,N\} \quad (17.10)$$

where

$$\mathcal{E}_N \equiv \{e_t, \quad t = 1,2,\ldots,N| \; |e_t| \leq \varepsilon_e, \quad t = 1,2,\ldots,N\} \quad (17.11)$$

and $\mathcal{F}_t$ is defined by Eq. (17.4). Under some conditions on $\mathcal{F}_t$, both $\mathcal{D}_N$ and

$$\mathcal{F}_t{*}\mathcal{D}_N \equiv \{f'^{\mathrm{T}}(\phi'_t)\theta' \,|\, f'(\phi'_t) \in \mathcal{F}_t, \theta \in \mathcal{D}_N\} \quad (17.12)$$

are polytopes or sets of polytopes. (A polytope is defined as a connected region of a Euclidean space with the union of linear faces as its boundary. Linear faces are defined by induction from the zero-dimensional face, which is a point: a k-dimensional face ($k > 0$) is a connected subset on a k-dimensional manifold with boundary the union of less-than-k-dimensional face.)

LEMMA 17.1. If each $\mathcal{F}_t$, $t = 1, 2, \ldots, N$ has an orthotope range and $\mathcal{F}$ is the Cartesian product of $\mathcal{F}_1$ to $\mathcal{F}_N$, then for any orthotope $\mathcal{E}$, $\mathcal{E}/\mathcal{F}$ is the union of a set of polytopes. Lemma 17.1 follows from Lemma 17.2.

In static EIV parameter bounding, $\mathcal{F}$ is indeed the Cartesian product of its components: if $f'^{\mathrm{T}}_1, \ldots, f'^{\mathrm{T}}_N$ are, respectively, in $\mathcal{F}_1 \ldots, \mathcal{F}_N$, then $[f'_1 \; f'_2 \; \ldots f'_N]^{\mathrm{T}} \in \mathcal{F}$. By Lemma 17.1, $\mathcal{E}/\mathcal{F}$ is linear, i.e., the union of a set of polytopes. However, Example 1 shows that in the dynamic case the boundaries of $\mathcal{F}$ may be linear and convex, with $\mathcal{E}$ an orthotope, without $\mathcal{E}/\mathcal{F}$ having linear boundaries. In the dynamic case, the uncertainties in $f'_1, \ldots, f'^{\mathrm{T}}_N$ are not independent.

The next lemma refers to the shape of the parameter bounds imposed by y^o_t and ϕ^o_t obtained at a single time instant. Denote by $\mathcal{K}$ a full set of binary p-vectors $\kappa_j, j = 0,1,2,\ldots, 2^p - 1$ with elements κ_{ij}, $i = 1,2,\ldots, p$ each 1 or -1. To each κ_j corresponds a parameter-space orthant

$$O_j \equiv \{\theta \,|\, \theta_i \kappa_{ij} \geq 0, i = 1, \ldots, p\}.$$

LEMMA 17.2. The feasible parameter set due to a single $\phi^{o\prime}_t \equiv [y^o_t \; \phi^{o\mathrm{T}}_t]^{\mathrm{T}}$ can be obtained as the union of its parts in the orthants $O_j, j = 0,1, \ldots, 2^p{-}1$:

$$\mathcal{D}(\phi_t^{o\prime}) = \bigcup_{\kappa_j \in \mathcal{K}} \{\theta \in O(\kappa_j) \mid \sum_{i=1}^{p} \theta_i(\phi_t^{oi} - \kappa_{ij}\varepsilon_\phi^i) \leq y_t^o + \varepsilon_y, \tag{17.13}$$

$$\sum_{i=1}^{p} \theta_i(\phi_t^{oi} + \kappa_{ij}\varepsilon_\phi^i) \geq y_t^o - \varepsilon_y\}$$

In each orthant, $\mathcal{D}(\phi_t^{o\prime})$ is seen to be bounded by two hyperplanes with $\phi_t^o \pm vec(\kappa_{ij}\varepsilon_\phi^i)$ as normals.

PROOF: If $\theta \in \mathcal{D}(\phi_t^{o\prime})$ then there is a ϕ_t with $|\phi_t^i - \phi_t^{oi}| < \varepsilon_\phi^i$, $i = 1, \ldots, p$ such that

$$y_t^o - \varepsilon_y \leq \sum_{i=1}^{p} \theta_i \phi_t^i \leq y_t^o + \varepsilon_y$$

from which, with $\theta_i\kappa_{ij} \geq 0$, $i = 1, \ldots, p$ for $\theta \in O(\kappa_j)$,

$$y_t^o - \varepsilon_y \leq \sum_{i=1}^{p} \theta_i\phi_t^i = \sum_{i=1}^{p} \theta_i(\phi_t^{oi} + \varepsilon_\phi^i) \leq \sum_{i=1}^{p} \theta_i(\phi_t^{oi} + \kappa_{ij}\varepsilon_\phi^i)$$

and

$$\sum_{i=1}^{p} \theta_i(\phi_t^{oi} - \kappa_{ij}\varepsilon_\phi^i) \leq \sum_{i=1}^{p} \theta_i(\phi_t^{oi} + \varepsilon_\phi^i) = \sum_{i=1}^{p} \theta_i\phi_t^i \leq y_t^o + \varepsilon_y$$

proving that $\mathcal{D}(\phi_t^{o\prime})$ is a subset of the union on the right hand side of Eq. (17.13). To prove the reverse, that any $\theta \in O(\kappa_j)$ satisfying

$$\sum_{i=1}^{p} \theta_i(\phi_t^{oi} - \kappa_{ij}\varepsilon_\phi^i) \leq y_t^o + \varepsilon_y$$

and

$$\sum_{i=1}^{p} \theta_i(\phi_t^{oi} + \kappa_{ij}\varepsilon_\phi^i) \geq y_t^o - \varepsilon_y \tag{17.14}$$

is in $\mathcal{D}(\phi_t^{o\prime})$,consider the strip

$$\mathcal{L}_t = \{\phi_t \mid y_t^o - \varepsilon_y \leq \theta^T\phi_t \leq y_t^o + \varepsilon_y\}$$

and box

$$\mathcal{B}_t = \{\phi_t \,|\, \phi_t^{oi} - \varepsilon_\phi^i \le \phi_t^i \le \phi_t^{oi} + \varepsilon_\phi^i, i = 1, \ldots, p\}$$

in ϕ_t-space. It need only be shown that for any $\theta \in O(\kappa_j)$ satisfying Eq. (17.14), $\mathcal{L}_t$ intersects $\mathcal{B}_t$. Now $\phi_t = vec(\phi_t^{oi} - \kappa_{ij}\varepsilon_\phi^i)$ and $\phi_t = vec(\phi_t^{oi} + \kappa_{ij}\varepsilon_\phi^i)$ are opposing vertices of $\mathcal{B}_t$. By Eq. (17.14), each of the half spaces making up $\mathcal{L}_t$ contains a section of the line joining those vertices. The sections cannot be disjoint, as no point of ϕ_t-space is in either half space. Hence the sections overlap; there exists at least one point on both, which is in $\mathcal{L}_t \cup \mathcal{B}_t$. □

Example 2 considers the feasible parameter set for an ARX model with $q = 1$, $k = 1$ and $l = 1$. The first observations u_{1-d}^o, y_o^o and y_1^o give

$$\mathcal{D}_1 = \{\theta = [a\ b]^{\mathrm{T}} \,|\, \exists\, y_o, u_{1-d}, e_1 \ :\ y_1 = ay_o + bu_{-d} + e_1,$$

$$|\tilde{y}_o| \le \varepsilon_y,\ |\tilde{u}_{-d}| \le \varepsilon_u,\ |e_1| \le \varepsilon_e\}.$$

A little reflection shows that $\mathcal{D}_1$ is as in Fig. 17.2, where

$$a_1 = \frac{y_1^o + \varepsilon_e + \varepsilon_y}{y_o^o - \varepsilon_y},\ a_2 = \frac{y_0^o - \varepsilon_e - \varepsilon_y}{y_o^o + \varepsilon_y},\ b_1 = \frac{y_1^o + \varepsilon_e + \varepsilon_y}{u_{-d}^o - \varepsilon_u},\ b_2 = \frac{y_1^o - \varepsilon_e - \varepsilon_y}{u_{-d}^o + \varepsilon_u},$$

$$a_3 = \frac{y_1^o + \varepsilon_e + \varepsilon_y}{y_o^o + \varepsilon_y},\ a_4 = \frac{y_1^o - \varepsilon_e - \varepsilon_y}{y_o^o - \varepsilon_y},\ b_3 = \frac{y_1^o + \varepsilon_e + \varepsilon_y}{u_{-d}^o + \varepsilon_u},\ b_4 = \frac{y_1^o - \varepsilon_e - \varepsilon_y}{u_{-d}^o - \varepsilon_u}.$$

and

$$b_{\min} = \min\{b_1,b_2,b_3,b_4\},\ b_{\max} = \max\{b_1,b_2,b_3,b_4\},\ a_{\min} = \min\{a_1,a_2,a_3,a_4\} \text{ and}$$

$$a_{\max} = \max\{a_1,a_2,a_3,a_4\}.$$

In Fig. 17.2 the slopes of the bounds depend on the selected orthant: in the orthant with sign vector κ_j, the slopes are

$$\frac{(y_1^o + \varepsilon_e + \varepsilon_y)/(y_o^o - \kappa_{1j}\varepsilon_y)}{(y_1^o + \varepsilon_e + \varepsilon_y)/(u_{-d}^o - \kappa_{2j}\varepsilon_u)}\ and\ \frac{(y_1^o - \varepsilon_e - e_y)/(y_o^o + \kappa_{1j}\varepsilon_y)}{(y_1^o - \varepsilon_e - \varepsilon_y)/(u_{-d}^o + \kappa_{2j}\varepsilon_u)}.$$

The exact $\mathcal{D}_N$ due to a succession of observations is not simply the intersection of sets as in Fig. 17.2 for $t = 1,2, \ldots, N$, although this intersection does include $\mathcal{D}_N$. The intersection can be computed readily by established polytope exact updating procedures.[14–16] The exact $\mathcal{D}_N$ may be computed as follows, taking Example 2 for illustration, i.e.,

$$y_t = ay_{t-1} + bu_{t-d-1} + e_t,\quad t = 1,2, \ldots, N;$$

$$|\tilde{y}_t| \le \varepsilon_y,\quad t = 0,1, \ldots, N;\ |\tilde{u}_t| \le \varepsilon_u,\quad t = -d, 1-d, \ldots,$$

$$N-d-1;\ |e_t| \le \varepsilon_e,\quad t = 1,2, \ldots, N$$

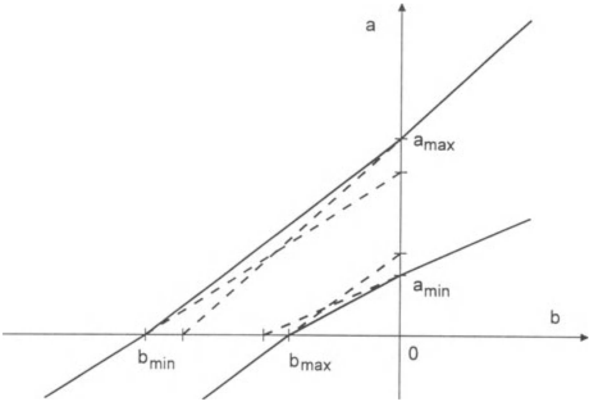


FIGURE 17.2. Example 2: parameter bounds imposed by $\phi_1^{o\prime}$.

The parameter bounds will be found only in two dimensions, but the method may be extended to more dimensions.[21] Parameter b is set to equally spaced values $b \in B \equiv \{b_o + i\delta b, i = 0,1, \ldots, K\}$. At each b, the extreme feasible values of a are found by a halving-doubling search. A trial $[a\ b]^{\mathrm{T}} \in \mathcal{D}$ iff the N-dimensional parallelepiped

$$\mathcal{P}_N = \{Ay^N \mid y^N \in \prod_{t=1}^{N} [y_t^o - \varepsilon_y, y_t^o + \varepsilon_y]\}$$

and the orthotope

$$Q_N = \prod_{t=1}^{N} [b(u_{t-d-1} - sign(b)\varepsilon_u) - \varepsilon_e, b(u_{t-d-1} + sign(b)\varepsilon_u) + \varepsilon_e]$$

intersect, where

$$A = \begin{bmatrix} 1 & a & 0\ldots & 0 \\ 0 & 1 & a\ldots & 0 \\ \vdots & & & \vdots \\ 0 & 0 & 1\ldots & a \\ 0 & \cdots & 0 & 1 \end{bmatrix} \quad \text{and } y^N = [y_N y_{N-1} \ldots y_1]^{\mathrm{T}}$$

Checking all the faces of Q_N and edges of $\mathcal{P}_N$ for intersections would be a heavy computation. Instead, the structure of matrix A can be exploited in a recursive

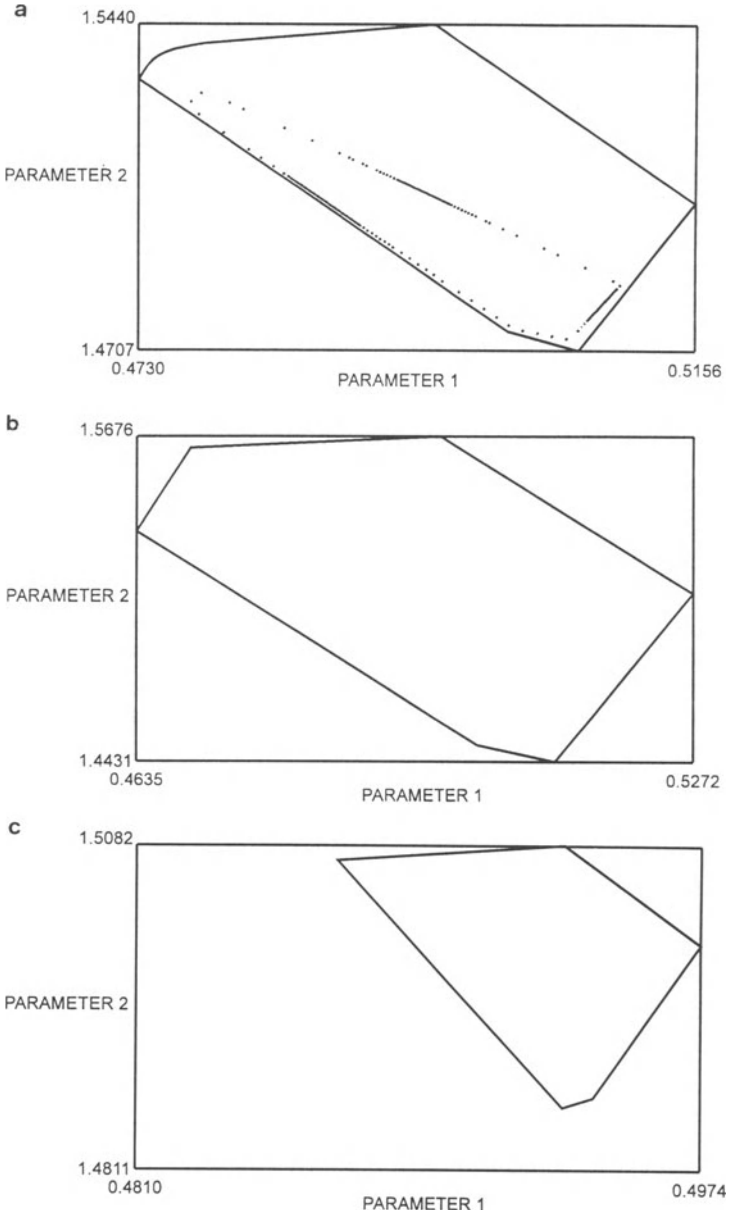


FIGURE 17.3. FPS given by various methods for Example 2 with $N = 30$; (a) dynamic-case EIV feasible parameter set (b) static-case EIV approximation to feasible parameter set (c) feasible parameter set given by equation-error approach.

procedure to decide if $\mathcal{P}_N \cap Q_N \neq \varnothing$. Denote by $[y_t^l, y_t^u]$ the feasible range of y_t on the basis of observations $y_o^o, \dots, y_t^o, u_{-d}^0, \dots, u_{t-d-1}^o$ and bounds $\varepsilon_y, \varepsilon_u, \varepsilon_e$. Then

$$[y_1^l, y_1^u] = [-a(y_o^o + sign(a)\varepsilon_y) + b(u_{-d}^o - sign(b)\,\varepsilon_u) - \varepsilon_e,$$

$$-a(y_o^o - sign(a)\,\varepsilon_y) + b(u_{-d}^o + sign(b)\,\varepsilon_u) + \varepsilon_e] \cap [y_1^o - \varepsilon_y, y_1^o + \varepsilon_y]$$

and thenceforth the feasible range of y is updated recursively by

$$[y_t^l, y_t^u] = [-ay_{t-1}^u + b(u_{t-d-1}^o - sign(b)\,\varepsilon_u) - \varepsilon_e, -ay_{t-1}^l$$

$$+ b(u_{t-d-1}^o - sign(b)\,\varepsilon_u) - \varepsilon_e]$$

$$\cap [y_t^o - \varepsilon_y, y_t^o + \varepsilon_y] \text{ if } a > 0,$$

$$[y_t^l, y_t^u] = [-ay_{t-1}^l + b(u_{t-d-1}^o - sign(b)\,\varepsilon_u) - \varepsilon_e, \; - ay_t^u + b(u_{t-d-1}^o - sign(b)\,\varepsilon_u) - \varepsilon_e]$$

$$\cap [y_t^o - \varepsilon_y, y_t^o + \varepsilon_y] \text{ if } a < 0.$$

If $[y_t^l, y_t^u] = \varnothing$ for some $t \leq N$ then $[a\ b]^{\mathrm{T}} \notin \mathcal{D}_N$.

Fig. 17.3 illustrates the differences between $\mathcal{D}_N$s found from an equation-error model, the static-EIV-case approximation and the exact dynamic-case FPS. Here N is 30, and successive inputs are independent and uniformly distributed (IUD) in $[-1,1]$. The structural error e_t is IUD in $[-0.01, 0.01]$, and the errors in y and u are IUD in $[-0.02, 0.02]$. The records are given in the Appendix. Figure 17.3(a) shows the dynamic-case $\mathcal{D}_N$ found by accurately assuming the error bounds to be $\varepsilon_e = 0.01$, $\varepsilon_y = \varepsilon_u = 0.02$. Figure 17.3(b) shows $\mathcal{D}_N$ given by the static-case treatment, and Fig. 17.3(c) the results of exact polytope bounding with equation-error bounds $\varepsilon = 0.01 + 1 \times 0.02 + 2 \times 0.02 + 0.02 = 0.09$ obtained by assuming reasonable *a priori* parameter bounds $|a| \leq 1$, $|b| \leq 2$. Use of narrower (erroneous) equation-error bounds can easily give mistaken results. For instance, $\varepsilon = 0.04$ gives an empty FPS from this data set. $\mathcal{D}_N$ obtained by the equation-error approach is larger than by either of the other methods.

17.4. FAST REJECTION OF PARTS OF PARAMETER SPACE FROM FPS

Calculating the polytope comprising the FPS in each orthant and finding out whether each is empty may take a great deal of time, so in this section we present algorithms to exclude some orthants from the FPS rapidly.

LEMMA 17.3. If for $i = 1, \dots, p$ the numbers

$$\frac{y_t^o + \varepsilon_y}{\phi_t^{oi} + \varepsilon_\phi^i}, \frac{y_t^o - \varepsilon_y}{\phi_t^{oi} - \varepsilon_\phi^i}, \frac{y_t^o + \varepsilon_y}{\phi_t^{oi} - \varepsilon_\phi^i}, \frac{y_t^o - \varepsilon_y}{\phi_t^{oi} + \varepsilon_\phi^i},$$

have the same signs as κ_{it}, $i = 1, \ldots, p$, then the orthant specified by parameter signs $\{-\kappa_{it}, i = 1, \ldots, p\}$ can be excluded from the FPS.

Remark: If $\varepsilon_y \to 0$ and $\varepsilon_\phi^i \to 0$ then the condition of Lemma 17.3 is satisfied. That is, for sufficiently small ε_y and ε_ϕ^i at least one orthant can be excluded from the parameter set if $y_t^o \neq 0$ and $\phi_t^{oi} \neq 0$.

PROOF: For the first and fourth expressions to have the same sign, $y_t^o + \varepsilon_y$ and $y_t^o - \varepsilon_y$ must have the same sign, denoted by κ_{ot}. Similarly $\phi_t^{oi} - \varepsilon_\phi^i$ and $\phi_t^{oi} + \varepsilon_\phi^i$ have the same sign κ_{ot}/κ_{it}. Consequently y_t^o has sign κ_{ot} and ϕ_t^{oi} has sign κ_{ot}/κ_{it}. Hence if each element θ_i of θ has sign $-\kappa_{it}$, i.e., if θ is in the orthant specified by parameter signs $\{-\kappa_{it}, i = 1, \ldots, p\}$,

$$sign(\theta_i \phi_t^{oi}) = sign(-\kappa_{it}\kappa_{ot}/\kappa_{it}) = -\kappa_{ot} \neq sign(y_t^o), \quad i = 1,2, \ldots, p$$

which implies that $\theta_1\phi_t^{o1} + \ldots \theta_p\phi_t^{op} \neq y_t^o$. Thus there is no feasible parameter point in the orthant. □

This lemma provides an easily applied sign test. For narrow error bounds it is often satisfied, allowing a high proportion of the orthants to be excluded quickly from the feasible parameter set.

A second test excludes some orthants not excluded by the sign test. Recall that by Lemma 17.2 the set of observations pertaining to time t bounds the FPS in orthant $O(\kappa)$ by a pair of hyperplanes: $H_t^1[\kappa]$ and $H_t^2[\kappa]$. The halfspaces bounded by $H_t^1[\kappa]$ and $H_t^2[\kappa]$ are denoted by $S_t^1[\kappa]$ and $S_t^2[\kappa]$, respectively. Note that these halfspaces depend on the orthant selected, as well as on the sampling instant. The calculations are made parsimonious by a series of tests, where each complicated test is carried out only if no decision has been made in the previous tests.

The tests for

$$S_{t_1}^1[\kappa] \cap S_{t_1}^2[\kappa] \cap S_{t_2}^1[\kappa] \cap S_{t_2}^2[\kappa] \cap O(\kappa) \neq \varnothing$$

are as follows:

1. $\mathbf{0} \in S_{t_1}^1[\kappa]$ *and* $\mathbf{0} \in S_{t_1}^2[\kappa]$ *and* $\mathbf{0} \in S_{t_2}^1[\kappa]$ *and* $\mathbf{0} \in S_{t_2}^2[\kappa]$ can be verified by checking four inequalities $y_{t_1} + \varepsilon_y \geq 0$ *and* $y_{t_1} - \varepsilon_y \leq 0$ *and* $y_{t_2} + \varepsilon_y \geq 0$ *and* $y_{t_2} - \varepsilon_y \leq 0$. If no decision has been made by (1) then apply Test (2).
2. $S_{t_j}^i[\kappa] \cap S_{t_l}^k[\kappa] \cap O(\kappa) = \varnothing$ can be checked by solving a simple set of inequalities. If it is found to be true for any $1 \leq i, j, k, l \leq 2$ with $(i,j) \neq (k,l)$ then $S_{t_1}^1[\kappa] \cap S_{t_1}^2[\kappa] \cap S_{t_2}^1[\kappa] \cap S_{t_2}^2[\kappa] \cap O(\kappa) = \varnothing$. If the orthant has still not been rejected, apply Test (3).
3. For $p \geq 3$ if $H_{t_1}^1[\kappa] \cap H_{t_1}^2[\kappa] \cap H_{t_2}^1[\kappa] \cap S_{t_2}^2[\kappa] \cap O(\kappa) \neq \varnothing$ or $H_{t_1}^1[\kappa] \cap H_{t_1}^2[\kappa] \cap S_{t_2}^1[\kappa] \cap H_{t_2}^2[\kappa] \cap O(\kappa) \neq \varnothing$ *or* $H_{t_1}^1[\kappa] \cap S_{t_1}^2[\kappa] \cap H_{t_2}^1[\kappa] \cap S_{t_2}^2[\kappa] \cap O(\kappa) \neq \varnothing$ *or* $S_{t_1}^1[\kappa] \cap H_{t_1}^2[\kappa] \cap H_{t_2}^1[\kappa] \cap S_{t_2}^2[\kappa] \cap O(\kappa) \neq \varnothing$ (each of which

leads to a simple set of inequalities in (p–3) variables) then $S^1_{t_1}[\kappa] \cap S^2_{t_1}[\kappa] \cap S^1_{t_2}[\kappa] \cap S^2_{t_2}[\kappa] \cap O(\kappa) \neq \emptyset$ is concluded. If Tests (1) to (3) have not resolved the issue, Test (4) is applied.

4. Define a large simplex $\mathcal{L}(M)$ fitted into the orthant $O(\kappa)$ with vertices

$$V_o = [0 \ldots 0]^{\mathrm{T}}$$

$$V_1 = [M\kappa_{o1}\ 0 \ldots 0]^{\mathrm{T}}$$

$$V_2 = [0\ M\kappa_{o2}\ 0 \ldots 0]^{\mathrm{T}}$$

$$\vdots$$

$$V_p = [0 \ldots 0\ M\kappa_{op}]^{\mathrm{T}}$$

where

$$M = \max_{i=1\ldots p} \mathit{inf}\{x \,|\, y_{t_1} + \varepsilon_y \geq x(\phi_{t_1} - \varepsilon^i_\phi)\}.$$

Alternatively, M can be chosen as the largest allowed absolute value of the parameter components. A p-dimensional polytope-updating procedure is then applied to decide whether $\mathcal{L}(M) \cap S^1_{t_1}[\kappa] \cap S^2_{t_1}[\kappa] \cap S^1_{t_2}[\kappa] \cap S^2_{t_2}[\kappa] \neq \emptyset$, which determines whether $S^1_{t_1}[\kappa] \cap S^2_{t_1}[\kappa] \cap S^1_{t_2}[\kappa] \cap S^2_{t_2}[\kappa] \cap O(\kappa) \neq \emptyset$.

The relatively large amount of calculation in Test (4) is incurred only if Tests (1) to (3) permit no decision.

17.5. EXACT-POLYTOPE AND ELLIPSOIDAL ALGORITHMS FOR THE EIV PROBLEM

One can now describe algorithms to compute polytope and ellipsoidal parameter bounds for the errors-in-variables problem. Both algorithms first discard orthants as above (which restricts the sign combination of the parameters) then update the polytope or ellipsoid in each remaining orthant $O(\kappa)$, $\kappa \in \mathcal{K}$.

For polytope updating, the procedure starts from a cube in $O(\kappa)$ having one vertex at the origin, or from a simplex obtained by cutting the orthant $O(\kappa)$ with a hyperplane. The simplex needs fewer vertices and faces. Polytope updating involves intersecting the polytope with the halfspaces defined by hyperplanes $H^i_{t_1}[\kappa]$ and $H^j_{t_2}[\kappa]$. The updated $\mathcal{D}$ is the union of the updated polytopes over all orthants $O(\kappa)$, $\kappa \in \mathcal{K}$.

For ellipsoid updating, the procedure starts from a sphere $\mathcal{L}_0(\kappa)$ which must contain a sufficiently large cube contained in $O(\kappa)$ and having a vertex at the origin. The ellipsoid is intersected with the halfspaces defined by $H^i_{t_1}[\kappa]$ and $H^j_{t_2}[\kappa]$, using a simple modification of the basic ellipsoidal-bounding algorithm.[12,22] The updated $\mathcal{D}$ is the union of the updated ellipsoids over all orthants $O(\kappa)$, $\kappa \in \mathcal{K}$.

Polytope updating may become complicated for high parameter dimensions, but ellipsoid bounding remains practicable for large numbers of parameters and observations. The ellipsoid bounds may, however, become much looser than the polytope bounds. Walter and Piet-Lahanier[14] and Mo and Norton[15] describe economical procedures for intersecting a polytope with a halfspace. The complexity of the exact bounding polytope varies from case to case, so the computing load cannot be predicted. The computing load for ellipsoidal bounding is calculable from the number of parameters, an important point for on-line applications.

17.6. CONCLUSIONS

Parameter bounding of linear-in-the-parameters and errors-in-variables models has been considered. Such models meet the frequent practical need to distinguish between modeling errors, input errors and output errors. Parameter bounding suits the situation where statistical assumptions cannot be made about the modeling error but error bounds can be specified. The significant difference in difficulty between the static and dynamic parameter-bounding EIV problems has been shown. Two algorithms are available for the static problem, yielding polytope or ellipsoid bounds. Fast procedures for discarding empty orthants in parameter space have been described. A bounding procedure based on two-dimensional boundary searches has been presented for the dynamic problem.

ACKNOWLEDGMENTS. The Estimation and Control Group, School of Electronic and Electrical Engineering, University of Birmingham produced the examples by a software package for system identification. The package includes a wide range of modeling techniques, including several parameter-bounding algorithms. The parameter-bounding features were written as part of an SERC project (under grant GR/E 28536) to investigate the potential of bounding for applications. The work continues and inquiries about possible applications of parameter or state bounding are welcome.

REFERENCES

1. A. Madansky, *JASA* **54**, 173 (1959).
2. P. A. P. Moran, *J. Multivar. Anal.* **1**, 232 (1971).
3. M. G. Kendall and A. Stuart, *The Advanced Theory of Statistics, vol. 2*, Griffin, London (1974).
4. D. J. Aigner and A. S. Goldberger, *Latent Variables in Socio-economic Models*, North Holland, Amsterdam, The Netherlands (1977).
5. T. Söderström, *Automatica* **17**, 713 (1981).
6. B. D. O. Anderson and M. Deistler, *J. Time Series Anal.* **5**, 1 (1984).

7. B. D. O. Anderson, *Automatica* **21**, 709 (1985).
8. M. Deistler, in: *Essays in Time Series Analysis, Lecture Notes in Statistics* (J. Gani and M. Priestley, eds.), Springer-Verlag, Berlin, Germany (1986).
9. M. Deistler, *Linear Errors in Variables Models: Some Structure Theory*, private communication (1988).
10. G. Belforte and M. Milanese, in: *Fifth IFAC Symposium on Identification & System Parameter Estimation*, Darmstadt, Germany, pp. 381–385 (1979) .
11. E. Fogel, *IEEE Trans. Autom. Control* **AC 24**, 752 (1979).
12. E. Fogel and Y. F. Huang, *Automatica* **18**, 229 (1982).
13. M. Milanese and G. Belforte, *IEEE Trans. Autom. Control* **27**, 408 (1982).
14. E. Walter and H. Piet-Lahanier, *IEEE Trans. Autom. Control* **34**, 911 (1989).
15. S. H. Mo and J. P. Norton, *Math. Comput. Simul.* **32**, 481 (1990).
16. V. Broman and M. J. Shensa, *Math. Comput. Simul.* **32**, 469 (1990).
17. E. Walter and H. Piet-Lahanier, in: *Proceedings of the IEEE International Symposium on Circuits & Systems*, Chicago, IL, pp. 774–777 (1993).
18. E. Walter and H. Piet-Lahanier, *Math. Comput. Simul.* **32**, 449 (1990).
19. J. P. Norton, *Automatica* **23**, 497 (1987).
20. T. Clement and S. Gentil, in: *Proceedings of the 12th IMACS World Congress*, Paris, France, pp. 484–486 (1988).
21. J. P. Norton and S. M. Veres, in: *Bounding Approaches to System Identification* (Milanese *et al.*, eds.) Plenum Press, New York, Chap. 20 (1996).
22. G. Belforte and B. Bona, in: *Seventh IFAC/IFORS Symposium on Identification & System Parameter Estimation*, York, United Kingdom, 1507–1512 (1985).

18

Errors-in-Variables Models in Parameter Bounding

V. Cerone

ABSTRACT

When all observed variables of a model are affected by noise, parameter estimation is known as the *errors-in-variables* problem. While parameter bounding methods and algorithms have been extensively developed in the case of exactly known regressor variables, little attention has been paid to the bounded errors-in-variables problem. This chapter gives a formal proof of a previous result on the description of the feasible parameter region for models linear in the parameters in the presence of bounded errors in all variables. Topological features of the feasible parameter region, such as convexity and connectedness, are also discussed. Finally, approximate parameter uncertainty intervals are derived for ARMAX models when all the observed variables are affected by bounded noise. For an example involving extensive simulations, central estimates obtained by means of the bounded errors-in-variables approach and least squares estimates are computed and compared.

V. CERONE • Dipartmento di Automatica e Informatica, Politecnico di Torino, 10129 Torino, Italy.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

18.1. INTRODUCTION

Identification is carried out on the basis of input-output observations taken from the system to be identified. Most identification methods rely on the assumption that the input is exactly known and the so called *equation-error* approach is used, where all noise is considered as additive equation error.[1] However, due to measurement errors, the assumption of noise-free input might often be unrealistic. The equation error is then correlated with the measured input, leading to bias both in statistical parameter estimates and in parameter bounds.[2] Problems where all observed variables are noisy are referred to as *errors-in-variables* problems. See Söderström[3] for the case of stochastic errors.

Despite their popularity, statistical methods for parameter estimation suffer from some drawbacks. It is well known that, due to cost constraints, there are situations where the number of collected observations cannot be large, as in biological systems, for example. In those cases, available measurement records are not long enough to check *a posteriori* from the residuals whether the probability density function assumed for the noise is appropriate or not; in other words, it might be impossible to realize that a statistical hypothesis is not met. Moreover, there are situations where the errors are better characterized in a deterministic way: systematic and class errors in measurement equipment, rounding and truncation errors in A/D converters are some.

An appealing alternative to the stochastic characterization of uncertainty in measurements is the bounded-error description. In this approach, the uncertain variables are no longer considered as random, but are assumed to belong to a given set. In this context, the outcome of identification is a set of parameter values, each a feasible solution of the estimation problem. In other words, all parameter vectors belonging to the *feasible parameter region* (*FPR*) are consistent with the measurements vector, the measurement error bounds and the assumed model structure.

When instantaneous constraints on the measurement error in a model linear in its parameter are available, and deterministic regressors are considered, the *FPR* turns out to be a polytope. Due to the possible complex shape of the *FPR*, some classes of methods have been proposed which compute a simpler set containing it. Milanese and Belforte[4] suggest to bound the feasible parameter region by an orthotope aligned with the parameter coordinate axes; the orthotope is computed by linear programming. A recursive algorithm given by Fogel and Huang[5] provides an ellipsoidal set outer-bounding the *FPR*. A combined use of the two algorithms, outer-bounding by linear programming and ellipsoidal outer-bounding, has been tested by Mo and Norton[6] and Belforte, Bona and Cerone.[7] To approximate the *FPR* simplexes have also been used, as well as parallelotopes[8] and other limited complexity polyhedra.[9] Should the approximation turn out to be too crude, then an exact description of the *FPR* is sometimes possible by means of recursive algorithms.[10–12] Inner approximation of the *FPR* is considered by Vicino and

Milanese;(13) they showed how to compute maximal balls in l_∞ norms (boxes), l_2 norms (ellipsoids) and l_1 norms (diamonds) when their shape is either known or partially free. The problem of computing maximum-volume ellipsoid inner bounds is also considered in Refs. 14–19.

Surprisingly, parameter bounding in the case of bounded errors-in-variables has received much less attention so far. Norton(2) gave an insight on the identification of ARMAX models in the bounding context, paving the way for the bounded errors in variables approach. Belforte, Bona and Cerone(20) give a result on how to describe the *FPR* for such problems; that result has been formally proved in Cerone(21,22) and is reported in Section 18.3 of this chapter. Clement and Gentil(23,24) address the problem of parameter bounding for output-error models when the output is a noisy vector, while the input is exactly known. Merkuryev(25) gives a solution based on interval analysis methods,(26) which deals with linear static models when both input and output signals have interval nature. Veres and Norton(27) discussed and pointed out the differences between static and dynamic errors-in-variables models as far as parameter bounds are concerned.

The purpose of this chapter is to address the problem of parameter bounding for linear models, when *all the observed variables* are affected by *bounded noise*. The chapter is organized in the following way. Section 18.2 states the estimation problem and introduces further notation and definitions. In Section 18.3, a formal proof of a previous result on the description of the feasible parameter region for linear models in the presence of bounded errors in all variables is given, together with further results which give an insight into the shape and structure of the *FPR*. Topological features of the *FPR*, such as convexity and connectedness, are discussed in Section 18.4. In Section 18.5 parameter bounds for ARMAX models by means of the errors-in-variables approach are derived.

18.2. PROBLEM FORMULATION AND NOTATION

Suppose that a given system is described by the linear model

$$w_t = \mathbf{x}_t^T \boldsymbol{\theta}, \tag{18.1}$$

where $w_t \in R$ is the hypothetical noise-free output, $\mathbf{x}_t \in R^p$ is the hypothetical noise-free regressor vector and $\boldsymbol{\theta} \in R^p$ is the unknown parameter vector. Due to measurement and model errors, the variables actually observed are

$$\boldsymbol{\varphi}_t = \mathbf{x}_t + \boldsymbol{\delta\varphi}_t, \tag{18.2}$$

$$y_t = w_t + \delta y_t; \quad t = 1, \ldots, N; \tag{18.3}$$

where N is the number of output samples, y_t is the output measured at time t, δy_t is the output measurement uncertainty, $\boldsymbol{\varphi}_t$ is the measured regressor and $\boldsymbol{\delta\varphi}_t$ is its uncertainty. Reduction of Eqs. (18.1), (18.2) and (18.3) leads to

$$y_t = (\boldsymbol{\varphi}_t - \boldsymbol{\delta\varphi}_t)^T\boldsymbol{\theta} + \delta y_t. \tag{18.4}$$

Given symmetric bounds Δy_t on the output measurement uncertainty and $\Delta\varphi_{tj}$ on regressors uncertainty, i.e.,

$$|\delta y_t| \leq \Delta y_t, \tag{18.5}$$

$$|\delta\varphi_{tj}| \leq \Delta\varphi_{tj},\quad j = 1, \ldots, p, \tag{18.6}$$

the problem addressed in this chapter is that of finding the feasible parameter region, defined as follows. Let $\mathcal{D}_t$ be the feasible parameter set associated with the observation at time t

$$\mathcal{D}_t = \{\boldsymbol{\theta} \in R^p \colon y_t = (\boldsymbol{\varphi}_t - \boldsymbol{\delta\varphi}_t)^T\boldsymbol{\theta} +$$

$$\delta y_t;\ |\delta y_t| \leq \Delta y_t;\ |\delta\varphi_{tj}| \leq \Delta\varphi_{tj};\ j = 1, \ldots, p\}. \tag{18.7}$$

The *FPR* corresponding to the whole set of observations is then $\mathcal{D} = \cap_{t=1}^{N} \mathcal{D}_t$. It is assumed that the components φ_{tj}'s ($j = 1,2, \ldots, p$) of the regressor vector $\boldsymbol{\varphi}_t$ are permitted to vary independently. The difficulty of describing $\mathcal{D}$ arises from the nonlinear relationship between the unknown uncertainties $\delta\varphi_{tj}$ and the unknown parameters θ_j in Eq. (18.4).

To simplify algebra, without loss of generality, the output measurement y_t is considered as a regressor variable and δy_t as regressor uncertainty, by setting

$$\boldsymbol{\varphi}_t^{*T} = \left[\boldsymbol{\varphi}_t^T \vdots -y_t\right],\ \boldsymbol{\delta\varphi}_t^{*T} = [\boldsymbol{\delta\varphi}_t^T \vdots -\delta y],\ \boldsymbol{\Delta\varphi}_t^{*T} = [\boldsymbol{\Delta\varphi}_t^T \vdots \Delta y],\ \boldsymbol{\theta}^{*T} = [\boldsymbol{\theta}^T \vdots 1].$$

By this means Eq. (18.4) becomes

$$(\boldsymbol{\delta\varphi}_t^* - \boldsymbol{\varphi}_t^*)^T\boldsymbol{\theta}^* = 0; \tag{18.8}$$

while Eqs. (18.5 and 18.6) can be written as

$$|\delta\varphi_{tj}^*| \leq \Delta\varphi_{tj}^*,\quad j = 1, \ldots, p. \tag{18.9}$$

Any orthant in the parameter space Θ can be defined as

$$O(\boldsymbol{\alpha}) = \{\boldsymbol{\theta} \in R^p\colon\ \alpha_j\theta_j \geq 0,\ \ j = 1, \ldots, p\},$$

where $\boldsymbol{\alpha}$ is the vector of the signs of the components of $\boldsymbol{\theta}$ in this orthant.

18.3. DESCRIPTION OF THE FEASIBLE PARAMETER REGION

To describe the feasible parameter region defined in the preceding section, recall the following theorem, which is stated in Belforte, Bona and Cerone[(20)] and receives a formal proof in Cerone.[(21,22)]

THEOREM 18.1. *A necessary and sufficient condition for* $\boldsymbol{\theta}$ *to belong to the set* $\mathcal{D}_t$ *is*

$$|y_t - \boldsymbol{\varphi}_t^T\boldsymbol{\theta}| \leq \sum_{j=1}^{p} \Delta\varphi_{tj}|\theta_j| + \Delta y_t. \tag{18.10}$$

or, equivalently

$$|\boldsymbol{\varphi}_t^{*T}\boldsymbol{\theta}^*| \leq \sum_{j=1}^{p+1} \Delta\varphi_{tj}^*|\theta_j^*|. \tag{18.11}$$

Before the proof is given, it is useful to explain Eqs. (18.8 and 18.9) geometrically in the $(p + 1)$-dimensional space of uncertainties $\delta\varphi_{tj}^*$. Equation (18.8) represents a hyperplane passing through $\boldsymbol{\varphi}_t^*$ and normal to $\boldsymbol{\theta}^*$. Equation (18.9), which describes the feasible uncertainty region (*FUR*), is an axes-aligned orthotope centered at the origin and with vertices in $\pm\Delta\varphi_{tj}^*$. The supporting hyperplanes of the *FUR* normal to $\boldsymbol{\theta}^*$ satisfy

$$\boldsymbol{\delta\varphi}_t^{*T}\boldsymbol{\theta}^* = \pm \sum_{j=1}^{p+1} \Delta\varphi_{tj}^*|\theta_j^*|. \tag{18.12}$$

Proof of necessity: One has to prove that for all $\delta\varphi_{tj}^*$ satisfying Eqs. (18.8 and 18.9), Eq. (18.11) is true. From Eq. (18.8) one gets

$$\boldsymbol{\varphi}_t^{*T}\boldsymbol{\theta}^* = \sum_{j=1}^{p+1} \delta\varphi_{tj}^*\theta_j^*. \tag{18.13}$$

Taking absolute values in Eq. (18.13), using the triangle inequality and taking account of Eq. (18.9), the proof of necessity is obtained

$$|\boldsymbol{\varphi}_t^{*T}\boldsymbol{\theta}^*| = |\sum_{j=1}^{p+1} \delta\varphi_{tj}^*\theta_j^*| \leq \sum_{j=1}^{p+1} |\delta\varphi_{tj}^*|\ |\theta_j^*| \leq \sum_{j=1}^{p+1} \Delta\varphi_{tj}^*|\theta_j^*|. \tag{18.14}$$

Proof of sufficiency: To prove sufficiency, one has to show that for all θ^* satisfying Eq. (18.11), there exists some $\delta\varphi_{tj}^*$ satisfying both Eqs. (18.8 and 18.9). Using Eq. (18.8), Eq. (18.11) becomes

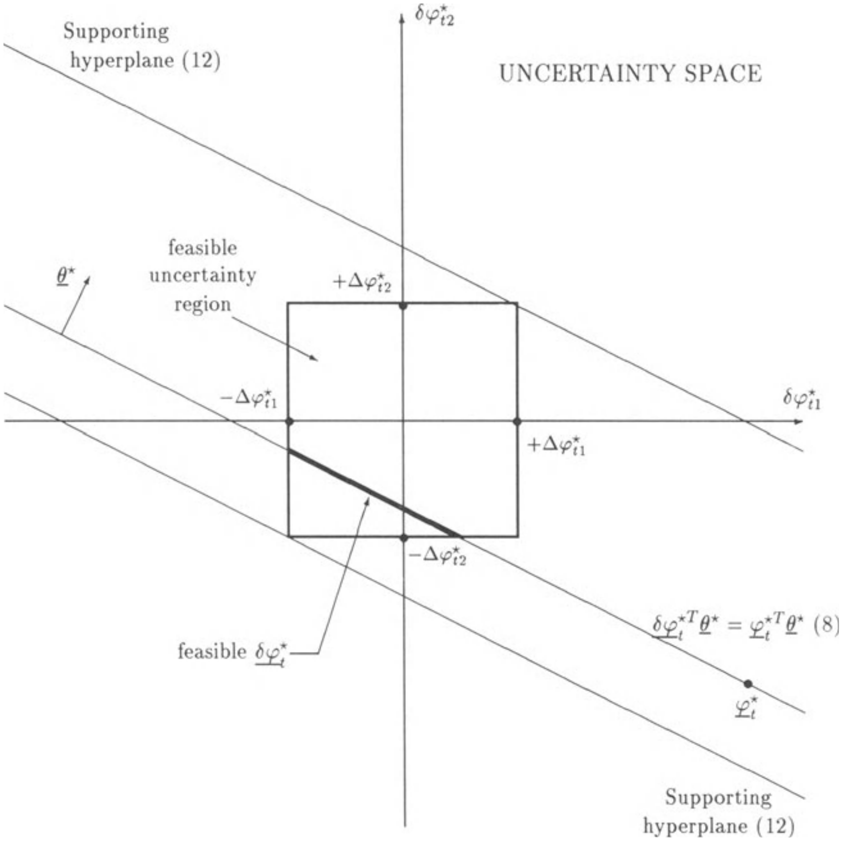

$\delta\varphi^\star_{t2}$
UNCERTAINTY SPACE
Supporting hyperplane (12)
feasible uncertainty region
$\underline{\theta}^\star$
$+\Delta\varphi^\star_{t2}$
$-\Delta\varphi^\star_{t1}$
$\delta\varphi^\star_{t1}$
$+\Delta\varphi^\star_{t1}$
$-\Delta\varphi^\star_{t2}$
$\delta\underline{\varphi}^{\star T}_t \underline{\theta}^\star = \underline{\varphi}^{\star T}_t \underline{\theta}^\star$ (8)
feasible $\delta\underline{\varphi}^\star_t$
$\underline{\varphi}^\star_t$
Supporting hyperplane (12)

FIGURE 18.1. A two-dimensional illustration of Theorem 18.1

$$|\delta\boldsymbol{\varphi}_t^{*T}\boldsymbol{\theta}^*| \le \sum_{j=1}^{p+1} \Delta\varphi_{tj}^*|\theta_j^*|. \tag{18.15}$$

Hyperplane of Eq. (18.8) (normal to $\boldsymbol{\theta}^*$) lies between the two supporting hyperplanes of Eq. (18.12) (normal to $\boldsymbol{\theta}^*$) of the feasible uncertainty region (Eq. (18.9)). Hyperplane of Eq. (18.8) always either cuts the *FUR* or lies on one edge of it, proving that there are $\delta\varphi_{tj}^*$ satisfying both Eq. (18.8 and 18.9). A two-dimensional geometric illustration of Theorem 18.1 is given in Fig. 18.1.

Proposition 18.1 gives an insight into the shape and structure of the *FPR* $\mathcal{D}$.

Proposition 18.1. The feasible parameter region $\mathcal{D}$ is the union of at most 2^p convex sets, each the intersection of $\mathcal{D}$ with a single orthant of parameter space, i.e.,

$$\mathcal{D} \bigcup_{\boldsymbol{\alpha}\in\Gamma} \mathcal{D}(\boldsymbol{\alpha}), \tag{18.16}$$

where

$$\mathcal{D}(\boldsymbol{\alpha}) = \bigcap_{t=1}^{N} \mathcal{D}_t(\boldsymbol{\alpha}), \tag{18.17}$$

and

$$\mathcal{D}_t(\boldsymbol{\alpha}) = \{\boldsymbol{\theta} \in \mathcal{O}(\boldsymbol{\alpha}) : y_t + \Delta y_t \geq \sum_{j=1}^{p} (\varphi_{tj} - \alpha_j \Delta\varphi_{tj})\theta_j,$$

$$y_t - \Delta y_t \leq \sum_{j=1}^{p} (\varphi_{tj} + \alpha_j \Delta\varphi_{tj})\theta_j\}. \tag{18.18}$$

Proof. The feasible parameter region $\mathcal{D}$ can obviously be decomposed into at most 2^p subsets consisting of its intersections with each of the 2^p orthants. It remains to be proven that each such subset is a convex set. First note that Eq. (18.18), which describes $\mathcal{D}_t$, in a given orthant $\mathcal{O}(\boldsymbol{\alpha})$, is a result of Theorem 18.1; it can be obtained directly from Eq. (18.10), setting $|\theta_j| = \theta_j \mathrm{sgn}(\theta_j)$ and $\alpha_j = \mathrm{sgn}(\theta_j)$. In a given orthant $\boldsymbol{\alpha}$ is fixed, which means that Eq. (18.18) gives a region bounded by two hyperplanes. It is easy to see that the above hyperplanes are not, in general, parallel; they lie symmetrically with respect to the hyperplane $y_t - \boldsymbol{\varphi}_t^T\boldsymbol{\theta} = 0$ (as a consequence of assuming symmetrical bounds in Eqs. (18.5 and 18.6)) and cannot intersect in the considered orthant. Hence, $\mathcal{D}_t(\boldsymbol{\alpha})$ is a convex region and $\mathcal{D}(\boldsymbol{\alpha})$, which is given by Eq. (18.17), can only be a convex set (if not empty).

Proposition 18.2. *If*

$$|y_t| > \Delta y_t, \ |\varphi_{tj}| > \Delta\varphi_{tj}, \ j = 1, \ldots, p, \tag{18.19}$$

there is no intersection between $\mathcal{D}_t$ and the orthant characterized by $\mathrm{syn}(\theta_j) = -sgn(y_t)sgn(\varphi_{tj})$.

Proof. (By contradiction) From Eq. (18.8)

$$|\delta\boldsymbol{\varphi}_t^{*T}\boldsymbol{\theta}^*| = |\boldsymbol{\varphi}_t^{*T}\boldsymbol{\theta}^*| = |y_t - \boldsymbol{\varphi}_t^T\boldsymbol{\theta}| = \Big|\, |y_t|\,\mathrm{sgn}(y_t)$$

$$- \sum_{j=1}^{p} |\varphi_{tj}|\,\mathrm{sgn}(\varphi_{tj})\,|\theta_j|\,\mathrm{sgn}(\theta_j)\Big|. \tag{18.20}$$

Now, if there are some $\boldsymbol{\theta}$, belonging to $\mathcal{D}_t$, such that

$$\operatorname{sgn}(\theta_j)\operatorname{sgn}(\varphi_{tj}) = -\operatorname{sgn}(y_t),\ (j = 1,2,\ldots,p),$$

then one gets

$$|\boldsymbol{\delta\varphi}_t^{*T}\boldsymbol{\theta}^*| = |y_t| + \sum_{j=1}^{p} |\varphi_{tj}|\,|\theta_j|, \tag{18.21}$$

and, finally, taking Eq. (18.19) into account one obtains

$$|\boldsymbol{\delta\varphi}_t^{*T}\boldsymbol{\theta}^*| > \sum_{j=1}^{p} \Delta\varphi_{tj}|\theta_j| + \Delta y_t = \sum_{j=1}^{p+1} \Delta\varphi_{tj}^*|\theta_j^*|, \tag{18.22}$$

which contradicts Eq. (18.11) for $\boldsymbol{\theta}$ to be feasible.

From the geometrical point of view, it can be seen that, when Eq. (18.19) and thus Eq. (18.21) are satisfied, the hyperplane of Eq. (18.8) (normal to $\boldsymbol{\theta}^*$) always lies outside the region included between the two supporting hyperplanes (normal to $\boldsymbol{\theta}^*$) of the feasible uncertainty region (Eq. (18.9)). This means that the hyperplane of Eq. (18.8) never cuts the *FUR*, proving that there are no $\delta\varphi_{tj}^*$ satisfying both Eq. (18.8 and 18.9). More precisely, in any $\boldsymbol{\delta\varphi}_t^*$ satisfying Eq. (18.8), there are some components $\delta\varphi_{tj}^*$ such that $|\delta\varphi_{tj}^*| > \Delta\varphi_{tj}^*$, which contradicts Eq. (18.19).

REMARK 1. Proposition 18.2 gives a sufficient condition for a whole orthant to be unfeasible. Hypothesis Eq. (18.19) is equivalent to assuming that the relative error on the output and each regressor is lower than 100%.

18.4. TOPOLOGICAL FEATURES OF THE FEASIBLE PARAMETER REGION

Convexity and connectedness of the feasible parameter region $\mathcal{D}$ are now discussed. Consider the following static model

$$y_t = (\varphi_{t1} - \delta\varphi_{t1})\theta_1 + (\varphi_{t2} - \delta\varphi_{t2})\theta_2 + \delta y_t. \tag{18.23}$$

$\{\delta y_t\}$ and $\{\delta\varphi_{tj}\}$ are random sequences uniformly distributed in [–1,1]. Regressors φ_{tj} are generated randomly and uniformly distributed in [–10,10]. Numerical simulations with a true value for the parameters given by $\theta_1^\circ = 0.8$ and $\theta_2^\circ = 0.5$ have been carried out.

Figures 18.2 to 18.5 show some features of the feasible parameter region. When regressors are exactly known, the *FPR* $\mathcal{D}_t$ is bounded by two parallel hyperplanes. When regressors are noisy as well, the *FPR* $\mathcal{D}_t$ is not convex either (see Fig. 18.2) and the final $\mathcal{D}$ will not generally be convex (see Fig. 18.3 and Fig. 18.4). This has already been noted[2,17] as resulting from uncertainty in the AR variables of an ARMAX model. Moreover, the above result is in agreement with

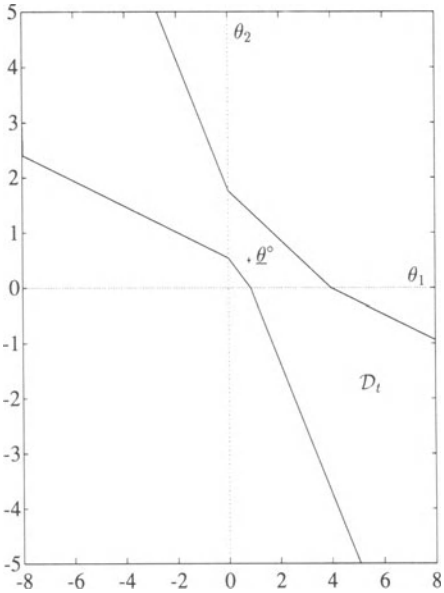


FIGURE 18.2. Feasible parameter region associated with a single measurement.

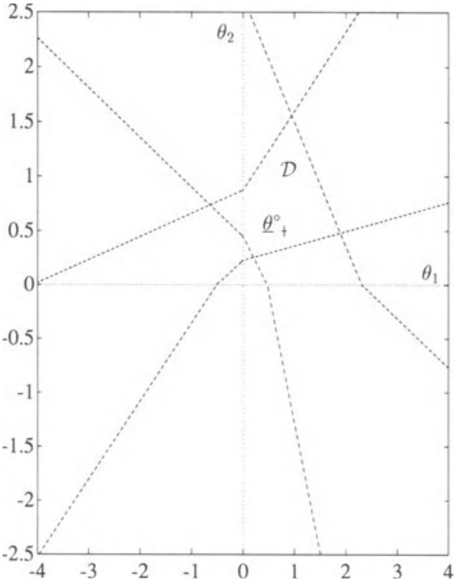


FIGURE 18.3. Parameter bounds from two observations.

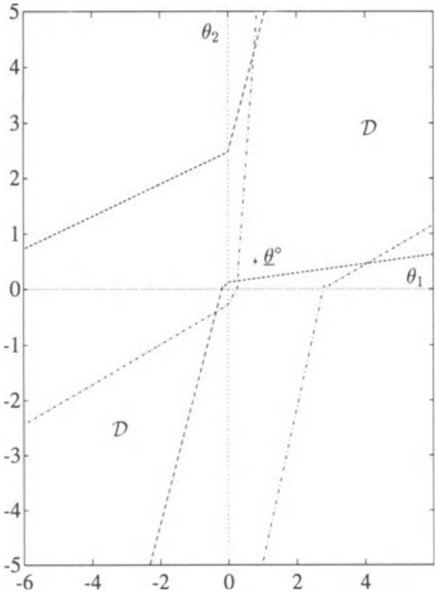


FIGURE 18.4. Nonconnected and unbounded *FPR* from two measurements.

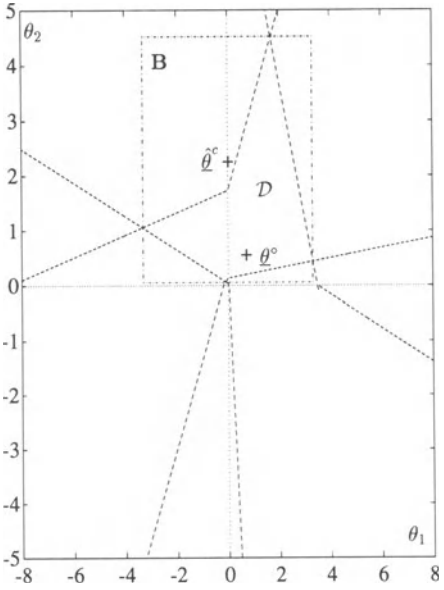


FIGURE 18.5. Feasible parameter region whose Chebyshev center ($\hat{\boldsymbol{\theta}}^c$) is not a feasible point.

Barmish and Sankaran.(28) They considered the dynamical system $\omega(t+1) = \Psi(t)\omega(t)$ and showed that convexity of the feasible state set can be lost because of independently varying uncertainties in the entries of $\Psi(\cdot)$.

Figure 18.4 shows that $\mathcal{D}$ may be a non-connected and unbounded set.

The Chebyshev center of $\mathcal{D}$, $\hat{\theta}^c$ (central estimate), is an optimal estimate in the sense that it minimizes the maximal distance from the unknown parameter vector that generated the data.(29) As far as linear models with exactly known regressors are concerned, $\hat{\theta}^c$ always belongs to $\mathcal{D}$ and, for l_∞ normed parameter space, it coincides with the geometrical center of the minimum-volume box B containing $\mathcal{D}$. However, as shown in Fig. 18.5, when regressors are noisy $\mathcal{D}$ may be a non-convex set, and $\hat{\theta}^c$ may not belong to the feasible parameter region.

18.5. PARAMETER BOUNDING IN ARMAX MODELS

In this section, Theorem 18.1 is applied to the problem of parameter bounding for ARMAX models when all the observed variables are affected by bounded noise.(30,31) Tempo, Barmish and Trujillo(32) present an approach to robust estimation and prediction of ARMA models when both the output errors and the noise are bounded. Clement and Gentil(23,24) address the problem of parameter bounding for output error models. Veres and Norton(27) discuss and point out the differences between static and dynamic errors-in-variables models as far as parameter bounds are concerned.

Consider a single-input, single-output, linear and discrete-time system where the true input signal, x_t, and the noise-free output, w_t, are related through the linear difference equation

$$A(q^{-1})w_t = B(q^{-1})x_t. \tag{18.24}$$

Polynomials $A(\cdot)$ and $B(\cdot)$ are polynomials in the backward shift operator q^{-1}, $(q^{-1}w_t = w_{t-1})$:

$$A(q^{-1}) = 1 + a_1 q^{-1} + \ldots + a_{n_a} q^{-n_a},$$

and

$$B(q^{-1}) = b_0 + b_1 q^{-1} + \ldots + b_{n_b} q^{-n_b}. \tag{18.25}$$

Let y_t and u_t be the noise-corrupted measurements of w_t and x_t, respectively,

$$y_t = w_t + \eta_t,$$

and

$$u_t = x_t + \xi_t, \quad t = 1, \ldots, N; \tag{18.26}$$

N is the number of output samples. Uncertainties are known to vary within given bounds, i.e.,

$$|\eta_t| \leq \Delta\eta_t, \tag{18.27}$$

and

$$|\xi_t| \leq \Delta\xi_t. \tag{18.28}$$

The unknown parameter vector $\boldsymbol{\theta} \in R^p$ is defined as

$$\boldsymbol{\theta}^T = [a_1 \ldots a_{n_a} \; b_0 \, b_1 \ldots b_{n_b}], \tag{18.29}$$

where $n_a + n_b + 1 = p$. The feasible parameter region is defined as

$$\mathcal{D} = \{ \boldsymbol{\theta} \in R^p : A(q^{-1})[y_t - \eta_t] \\ = B(q^{-1})[u_t - \xi_t];\ |\eta_t| \leq \Delta\eta_t;\ |\xi_t| \leq \Delta\xi_t;\ t = 1, \ldots, N\}. \tag{18.30}$$

It is well known that the feasible parameter region for linear static models is a polytope. Due to serial dependence between output samples at different time, exact parameter bounds for dynamic models are no longer linear.(27) In this section, polytopic outer approximations $\mathcal{D}'$ of the exact *FPR* $\mathcal{D}$ is presented. Still, since $\mathcal{D}'$ may become fairly complex for large N, orthotope-outer bounding algorithms are considered, which compute orthotopic sets $\mathcal{B}$ containing $\mathcal{D}'$. They provide guaranteed parameter uncertainty intervals (*PUI*s), where

$$PUI_j = \left[\theta_j^{\min}, \theta_j^{\max}\right],\ j = 1, \ldots, p; \tag{18.31}$$

$$\mathcal{B} = \{\boldsymbol{\theta} \in R^p\!: \theta_j = \hat{\theta}_j^c + \delta\theta_j, |\delta\theta_j| \leq \Delta\theta_j/2,\ j = 1, \ldots, p\}, \tag{18.32}$$

with

$$\hat{\theta}_j^c = \frac{\theta_j^{\min} + \theta_j^{\max}}{2}, \tag{18.33}$$

$$\Delta\theta_j = |\theta_j^{\max} - \theta_j^{\min}|, \tag{18.34}$$

and

$$\theta_j^{\min} = \min_{\theta_j \in \mathcal{D}'} \theta_j,\quad \theta_j^{\max} = \max_{\theta_j \in \mathcal{D}'} \theta_j. \tag{18.35}$$

18.5.1. Bounded Equation Error Model

Reducing Eqs. (18.24 and 18.26) yields the following noisy linear regression model

$$y_t = \varphi_t^T\boldsymbol{\theta} + r_t, \tag{18.36}$$

where

$$r_t = A(q^{-1})\eta_t - B(q^{-1})\xi_t = \eta_t + \sum_{j=1}^{n_a} a_j\eta_{t-j} - \sum_{j=0}^{n_b} b_j\xi_{t-j}, \tag{18.37}$$

and

$$\varphi_t^T = [-y_{t-1} \cdots -y_{t-n_a}\ u_t\ u_{t-1} \cdots u_{t-n_b}]. \tag{18.38}$$

denote the equation error and the regressor vector respectively. If the equation error bounds are available, i.e., Δr_t such that $|r_t| \le \Delta r_t$, a set over bounding the exact feasible parameter region is known to be given by

$$\mathcal{D}'_{ee} = \{\boldsymbol{\theta} \in R^p\text{: } |y_t - \varphi_t^T\boldsymbol{\theta}| \le \Delta r_t,\ \ t = 1, \ldots, N\}. \tag{18.39}$$

Unfortunately, the main difficulty arises in specifying noise bounds Δr_t on the equation error r_t from those available on the input and output measurement errors. The equation error r_t depends on the measurement errors η_t and ξ_t, and unknown vector $\boldsymbol{\theta}$, according to Eq. (18.37). Hence, at least in principle, bounds on r_t cannot be computed.

18.5.2. Bounded Errors-in-Variables Model

Based on the result given in Section 18.3, one can give a different solution which allows a direct use of bounds on measurement errors. Reduce Eqs. (18.24 and 18.26) to the following form

$$y_t = -\sum_{j=1}^{n_a} (y_{t-j} - \eta_{t-j})a_j + \sum_{j=0}^{n_b} (u_{t-j} - \xi_{t-j})b_j + \eta_t. \tag{18.40}$$

Equations (18.27, 18.28 and 18.40) fit in the framework of the bounded-errors-in-variables model outlined in Section 18.3. Thus, Eq. (18.18) implies that $\mathcal{D}_t$ is described by

$$(\varphi_t - \Delta\varphi_j^{\mathrm{o}})^T\boldsymbol{\theta} \le y_t + \Delta\eta_t, \tag{18.41}$$

and

$$(\varphi_t + \Delta\varphi_t^{\mathrm{o}})^T\boldsymbol{\theta} \ge y_t - \Delta\eta_t, \tag{18.42}$$

where

$$\Delta\varphi_t^{\mathrm{o}T} = [\Delta\eta_{t-1}\mathrm{sgn}(a_1), \ldots, \Delta\eta_{t-n_a}\mathrm{sgn}(a_{n_a}),$$

$$\Delta\xi_t \operatorname{sgn}(b_0), \Delta\xi_{t-1} \operatorname{sgn}(b_1), \ldots, \Delta\xi_{t-n_b} \operatorname{sgn}(b_{n_b})]. \tag{18.43}$$

REMARK 2. In Theorem 18.1 it is assumed that the components φ_{tj}s ($j = 1,2,\ldots,p$) of the regressor vector $\boldsymbol{\varphi}_t$ are permitted to vary independently. In the case of ARMAX models, however, there is serial dependence among them; consequently, the set $\mathcal{D}'_{ev}$, obtained by intersecting the sets described by Eqs. (18.41) and (18.42), only includes the exact feasible parameter region, i.e., $\mathcal{D}'_{ev} \supset \mathcal{D}$.

REMARK 3. Suppose the input $\mathbf{u} = [u_1\ u_2\ \ldots\ u_N]^T$ is assumed to be precisely known and the measurement vector $\mathbf{y} = [y_1\ y_2\ \ldots\ y_N]^T$ is corrupted by noise. Equation (18.43) reduces to

$$\boldsymbol{\Delta\varphi}_t^{oT} = [\Delta\eta_{t-1} \operatorname{sgn}(a_1), \ldots, \Delta\eta_{t-n_a} \operatorname{sgn}(a_{n_a}), 0, 0, \ldots, 0], \tag{18.44}$$

which, together with conditions Eqs. (18.41 and 18.42), forms the result given by Clement and Gentil.(23,24)

REMARK 4. In the case of linear static models, conditions (41) and (42) reduce to the result given by Merkuryev [25]; see inequalities (6) in that paper.

18.5.3. Numerical Results and Discussion

The system considered here is an ARMAX model, characterized by (24) and (26) with

$$A(q^{-1}) = (1 - 1.1q^{-1} + 0.28q^{-2}),$$

$$B(q^{-1}) = (q^{-1} + 0.5q^{-2})$$

and $w_0 = 0$, $w_1 = 0$. Thus, the true parameter vector is

$$\boldsymbol{\theta}^o = [a_1\ a_2\ b_1\ b_2]^T = [-1.1\ 0.28\ 1\ 0.5]^T.$$

The system has two real poles for $z_1 = 0.4$ and $z_2 = 0.7$ and a zero at $z_3 = -0.5$. Bounded relative errors have been used, i.e.,

$$\eta_t = \varepsilon_t^y y_t,\ |\varepsilon_t^y| \le \Delta\varepsilon_t^y,\ \xi_t = \varepsilon_t^u u_t,\ |\varepsilon_t^u| \le \Delta\varepsilon_t^u.$$

In simulation, bounds on the errors at the input and the output are set as equal, i.e., $\Delta\varepsilon^y = \Delta\varepsilon^u = \Delta\varepsilon$; this is a realistic assumption since one may use the same measurement equipment to collect samples of u_t and y_t. The input sequence $\{x_t\}$ is uniformly distributed in $[-10,10]$.

Two noise distributions have been chosen for comparison purposes. One is the uniform distribution $U[-\Delta\varepsilon,\Delta\varepsilon]$ and the other is the normal distribution with zero mean and variance $\sigma_\varepsilon^2 = (\Delta\varepsilon/3)^2$ truncated at $\pm 3\sigma_\varepsilon$. With this choice, the two errors distributions are bounded by the same quantity $\pm\Delta\varepsilon$ and, of course, have different variances. Four different values of uncertainties bounds are chosen, namely $\Delta\varepsilon =$

2%, 5%, 10%, 20%. For a given $\Delta\varepsilon$ ten different values of N have been considered (N = 100,200, . . . , 1000) and for a fixed N, 100 independent sets of data are generated.

Parameter bounding from these records has been carried out by the bounded errors-in-variables approach, computing central estimates (*CE*) and orthotopic approximations of the feasible parameter set according to the scheme outlined in Section 18.5.2. Least squares estimates (*LSE*) have also been computed in order to give a comparison solution by a classical method. To compare the accuracy of the estimates $\hat{\boldsymbol{\theta}}$, the following parameter error norm has been introduced

$$\frac{\|\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}^o\|_2}{\|\boldsymbol{\theta}^o\|_2}$$

where $\|\cdot\|_2$ is the Euclidean norm. Figure 18.6 shows the mean values for 100 runs of the parameter error norms for central estimates and least squares estimates in the case of uniformly distributed errors. The results against the number of observations and for different values of noise level are grouped together in order to facilitate the comparison. Figure 18.7 depicts the results obtained when the errors belong to the truncated normal distribution. In order to clarify the results shown in Fig. 18.6 and Fig. 18.7, piece-wise straight-line interpolations are used for the discrete values.

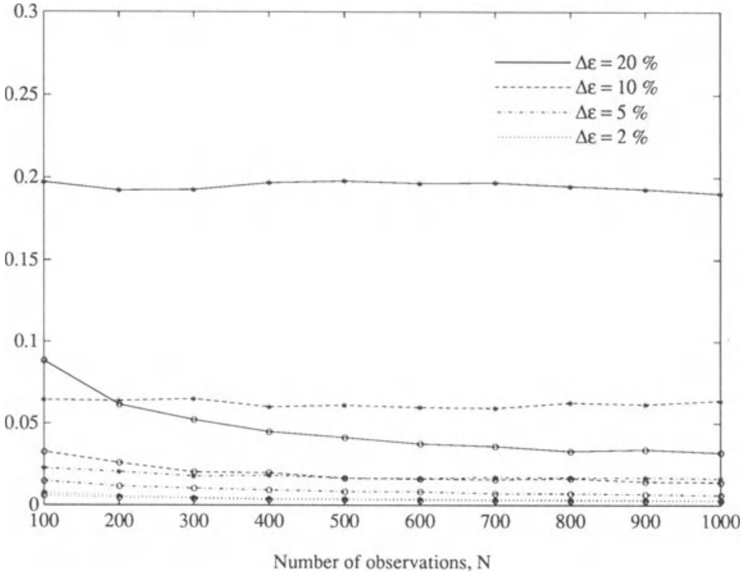


FIGURE 18.6. Mean values for 100 runs of parameter error norms for central estimates (CE), o, and least squares estimates (LSE), *, from records corrupted with uniformly distributed noise.

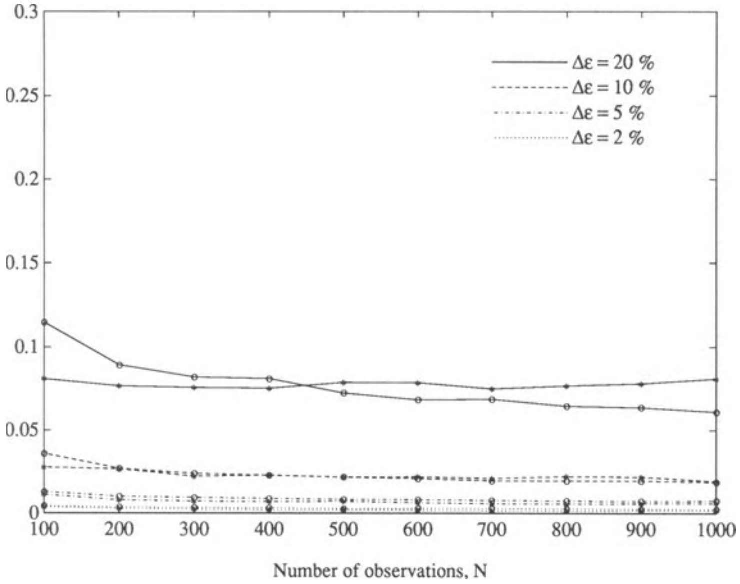


FIGURE 18.7. Mean values for 100 runs of parameter error norms for central estimates (CE), ○, and least squares estimates (LSE), *, from noisy records when the corrupting error sequence belongs to a truncated normal distribution.

Both the *CE* and the *LSE* give satisfactory estimation of the true parameters when the noise level is low (Δε ≤ 5%), for both uniform and normal distributions cases. Increasing the noise level (Δε ≥ 10%), decreases the accuracy of both *CE* and *LSE*. Central estimates are always more accurate than the least squares estimates in the case of uniformly distributed corrupting errors and for large *N* when the sequence of noise belongs to the truncated normal distribution. In the latter case, *LSE* are slightly more accurate than *CE* for high noise level and small *N*. When switching from uniform to normal distributions, *CE* exhibit about the same accuracy for Δε = 2%, 5%, 10% and become slightly less accurate for Δε = 20%. *LSE* obtained in the case of truncated Gaussian noise are always more accurate than those obtained by processing the data corrupted with uniformly distributed noise.

An informal explanation of the above performances is the following. It is well known that, if the noise is equipped with an l_∞ norm, central estimates are optimal, the optimality criterion being the maximum possible distance between the estimate and the true value. Least squares estimates are optimal in this sense when the errors are bounded by the l_2 norm.(29) The disturbances of the simulation presented in Section 18.5.3 are bounded by the l_∞ norm. This explains the good behavior of *CE*. The fact that *LSE* give better results when used with the truncated normal distribution suggests that they take advantage of the information contained in that kind of distribution.

18.6. CONCLUSIONS

Parameter bounding in models from records with bounded errors in both input and output data has been addressed. A formal proof of a previous result on the description of the feasible parameter region (*FPR*) for linear models in the presence of bounded errors in all variables is given, together with further results which give an insight into the shape and structure of the *FPR*. Topological features of the *FPR* have also been discussed. It may be *not* convex and *not* connected; moreover, its Chebyshev center may not be a feasible point.

Parameter outer-bounding for ARMAX models with both input and output bounded errors have also been presented. Central Estimates (*CE*) obtained with the bounded errors-in-variables approach, and least square estimates (*LSE*) have been computed for a hundred independent realizations of a simulated example, which shows the superiority of *CE* to *LSE* in the case of uniformly distributed corrupting errors. When the sequence of output noise belongs to a truncated normal distribution, both *CE* and *LSE* exhibit, approximatively, the same accuracy.

ACKNOWLEDGMENTS. This research was partly developed while Dr. Cerone was honorary research fellow at the School of Electronic and Electrical Engineering, University of Birmingham, UK, and the Politecnico di Torino. The italian Ministry of Universities and Research in Science and Technology (MURST), under the plan "Fondi per la Ricerca Scientifica – 40% e 60%" and a CNR scholarship supported this research.

REFERENCES

1. T. Söderström and P. Stoica, *System Identification*, Prentice-Hall, Englewood Cliffs, N.J. (1989).
2. J. P. Norton, *Int. J. Control* **45**, 375 (1987).
3. T. Söderström, *Automatica* **17**, 713 (1981).
4. M. Milanese and G. Belforte, *IEEE Trans. Autom. Control* **27**, 408 (1982).
5. E. Fogel and Y. F. Huang, *Automatica* **18**, 229 (1982).
6. S. H. Mo and J. P. Norton, in: *IEE Proceedings, part D*, No. 135, pp. 127–132 (1988).
7. G. Belforte, B. Bona, and V. Cerone, *Automatica* **26**, 887 (1990).
8. A. Vicino and G. Zappa, in: *Proceedings of the 32nd IEEE Conference on Decision and Control*, San Antonio, TX, pp. 2044–2049 (1993).
9. H. Piet-Lahanier and E. Walter, in: *Bounding Approaches to System Identification* (Milanese *et al.*, eds.) Plenum Press, New York, Chap. 16 (1996).
10. E. Walter and H. Piet-Lahanier, *IEEE Trans. Automat. Control* **34**, 911 (1989).
11. V. Broman and M. J. Shensa, *Math. Comput. Simul.* **32**, 469 (1990).
12. S. H. Mo and J. P. Norton, *Math. Comput. Simul.* **32**, 481 (1990).
13. A. Vicino and M. Milanese, *IEEE Trans. Automat. Control* **36**, 759 (1991).
14. J. P. Norton, *Int. J. Control* **50**, 2423 (1989).
15. L. Pronzato and E. Walter, *Int. J. Adapt. Control Signal Process.* **18**, 15 (1994).
16. L. Pronzato and E. Walter, in: *Bonding Approaches to System Identification* (Milanese *et al.*, eds.) Plenum Press, New York, Chap. 8 (1996).

17. J. P. Norton, *Automatica* **23**, 497 (1987).
18. E. Walter and H. Piet-Lahanier, *Math. Comput. Simul.* **32**, 449 (1990).
19. M. Milanese and A. Vicino, *Automatica* **27**, 997 (1991).
20. G. Belforte, B. Bona, and V. Cerone, *Math. Comput. Simul.* **32**, 561 (1990).
21. V. Cerone, in: *Prep. 9th IFAC/IFORS Symposium on Identification and System Parameter Estimation* (C. Banyasz and L. Keviczky, eds.), Budapest, Hungary, pp. 1518–1523 (1991).
22. V. Cerone, *Automatica* **29**, 1551 (1993).
23. T. Clement and S. Gentil, *Math. Comput. Simul.* **30**, 257 (1988).
24. T. Clement and S. Gentil, *Math. Comput. Simul.* **32**, 505 (1990).
25. Y. A. Merkuryev, *Int. J. Control* **50**, 2333 (1989).
26. R. E. Moore, *Methods and Applications of Interval Analysis*, SIAM Studies in Applied Mathematics, Philadelphia, PA (1979).
27. S. M. Veres and J. P. Norton, in: *9th IFAC Symposium on Identification and System Parameter Estimation* (C. Banyasz and L. Keviczky, eds.) Budapest, Hungary, pp. 1038–1043 (1991).
28. B. R. Barmish and J. Sankaran, *IEEE Trans. on Autom. Control* **24**, 346 (1979).
29. B. Z. Kacewicz, M. Milanese, R. Tempo, and A. Vicino, *Syst. Control Lett.* **8**, 161 (1986).
30. V. Cerone, in: *Prep. 9th IFAC Symposium on Identification and System Parameter Estimation* (C. Banyasz and L. Keviczky, eds.), Budapest, Hungary, pp. 1419–1424 (1991).
31. V. Cerone, *Int. J. Control* **57**, 225 (1993).
32. R. Tempo, B. R. Barmish, and J. Trujillo, in: *Prep. 8th IFAC/IFORS Symposium on Identification and System Parameter Estimation*, Beijing, P.R. China, pp. 1136–1140 (1988).

19

Identification of Linear Objects with Bounded Disturbances in Both Input and Output Channels

Y. A. Merkuryev

19.1. PROBLEM FORMULATION

The problem under consideration is to identify an object that is described by a linear equation

$$y = a_1 x_1 + \ldots + a_n x_n, \tag{19.1}$$

where $x_1, \ldots, x_n$ are input scalar signals, y is an output scalar signal, and $a_1, \ldots, a_n$ are the model coefficients, which must be estimated.

The object has been investigated experimentally. The input and output signals have been measured in m experiments: estimates $\tilde{x}_{ij}$ for x_{ij} (i.e., the input x_i in the jth experiment) and $\tilde{y}_j$ for y_j (i.e., the output y in the jth experiment) are known for all n inputs and m experiments. The measurements have been made with bounded additive disturbances:

$$\begin{cases} c_{ij}^- \le c_{ij} \le c_{ij}^+ & i = 1, \ldots, n, \\ d_j^- \le d_j \le d_j^+ & j = 1, \ldots, m, \end{cases} \tag{19.2}$$

Y. A. MERKURYEV • Riga Technical University, LV-1658 Riga, Latvia.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

where c_{ij} is the disturbance associated with x_{ij}, d_j is the disturbance associated with y_j, and c_{ij}^-, c_{ij}^+, d_j^- and d_j^+ are known. Note that in such a situation the bounds of the experimental input and output signals are known:

$$\begin{cases} x_{ij}^- \le x_{ij} \le x_{ij}^+, \\ y_j^- \le y_j \le y_j^+, \end{cases} \tag{19.3}$$

where

$$x_{ij}^- = \tilde{x}_{ij} + c_{ij}^-, \quad x_{ij}^+ = \tilde{x}_{ij} + c_{ij}^+,$$

$$y_j^- = \tilde{y}_j + d_j^-, \quad y_j^+ = \tilde{y}_j + d_j^+,$$

and

$$i = 1, \ldots, n, \quad j = 1, \ldots, m.$$

The identification problem may now be formulated as follows: to build the region W for possible values of the parameters $a_1, \ldots, a_n$ that is consistent with Eq. (19.3). For each vector $A^{\text{o}} \in$ W there are x_{ij}^{o} and y_j^{o} in Eq. (19.3) such that the relation

$$y_j^{\text{o}} = A^{\text{o}} X_j^{\text{o}}$$

holds for each $j = 1, \ldots, m$. Here

$$A^{\text{o}} = [a_1^{\text{o}} \ldots a_n^{\text{o}}], \quad X_j^{\text{oT}} = [x_{1j}^{\text{o}} \ldots x_{nj}^{\text{o}}].$$

An exact description of W is often problematic. It can be defined for up to six or seven model coefficients, but then becomes numerically intractable. The identification problem is generally reformulated to seeking the enclosing set W^*. Two different approaches are mainly used to describe W^*. The first uses hypercubes as enclosed sets,[1] and the second uses ellipsoidal sets.[2] This chapter shall study the first approach.

An exact description of W is considered in Refs. 3 and 4. A number of papers on parameter-bounding identification methods (for instance, Refs. 5 and 6) are surveyed in Ref. 7.

To start to solve the identification problem, first we follow Merkuryev.[8] Then, a method for situations when the signs of the model coefficients are not known *a priori*, is discussed.

19.2. IDENTIFICATION WHEN THE SIGNS OF THE COEFFICIENTS ARE KNOWN

It is possible to describe the region W by means of a system of linear inequalities. The method that may be used to build this system depends on *a priori* information about the signs of the coefficients $a_1, \ldots, a_n$. First assume that these signs are known *a priori* (for instance from physical interpretation of the coefficients). Later, drop this assumption.

If the signs of the coefficients are known, then, using interval arithmetic, it is possible to write expressions for the minimum and maximum values of the right-hand side of Eq. (19.1) if the input signals belong to the intervals in Eq. (19.3). For example, if all the coefficients are positive, the expressions for the minimum and the maximum are

$$\begin{cases} (AX)_j^- = a_1x_{1j}^- + \ldots + a_nx_{nj}^- \\ (AX)_j^+ = a_1x_{1j}^+ + \ldots + a_nx_{nj}^+, \quad j = 1, \ldots, m. \end{cases} \tag{19.4}$$

These expressions allow construction of the inequalities describing the region W:

$$\begin{cases} y_j^+ \geq (AX)_j^- \\ y_j^- \leq (AX)_j^+, \quad j = 1, \ldots, m. \end{cases} \tag{19.5}$$

These inequalities arise because each experiment admits values of the coefficients $a_1, \ldots, a_n$ such that the intervals for the left- and right-hand sides of Eq. (19.1), which appear because of Eq (19.3), intersect. Each pair of inequalities describes those values of the coefficients that give such an intersection in the corresponding experiment; thus these values are consistent with both Eqs. (19.1 and 19.3) in this experiment. The region W consists of those points that are common to all the experiments; these points are described by means of the full system of inequalities in Eq. (19.5).

The enclosed hypercube W^* may be obtained as a set of parameter uncertainty intervals:

$$W^* = \{\mathrm{A} \in R^n | a_i \in [a_i^-, a_i^+], i = 1, \ldots, n\},$$

where

$$a_i^- = \min_{A \in W} a_i, \quad a_i^+ = \max_{A \in W} a_i.$$

The bounds a_i^- and a_i^+ can be obtained by solving a corresponding linear programming problem. This problem is based on the system of linear inequalities (19.5) and uses a_i as a cost function to be correspondingly minimized or maximized.[7,9] It can be solved, for instance, by a simplex method.[10]

19.3. DETERMINATION OF THE SIGNS OF THE COEFFICIENTS

The way to build the region W when the signs of the coefficients $a_1, \ldots, a_n$ are not known *a priori* has been discussed. When these signs are not known, it is necessary:

(i) to find those sets of signs that are consistent with the experimental information;
(ii) to build a subregion Wj for each such set of signs; and
(iii) to consider the region W as the union of subregions

$$W = \underset{j}{U}\, Wj. \tag{19.6}$$

The main problem is to realize the first step. After it has been made, one can use each set of signs, as has been done before.

The present aim is to survey one method of determination of the signs of the coefficients to be found. (Two other methods are considered in Merkuryev.[(8)]) The method comprises the following stages.

1. Evaluate the vector $A = [a_1 \ldots a_n]$ by means of some traditional method of evaluation (e.g., the least-squares method), or by solving a system of n equations which make use of Eq. (19.1) and first n experiments: further on signs of the coefficients that have been found are used and refined;

2. Make an attempt to build up the region W_1 for the set of coefficients signs from the preceding step. Due to the convex nature of the region W_1, Eq. (19.5) should be solved by linear programming. First, if necessary, Eq. (19.5) should be rewritten so that it may be investigated by linear programming. This demands the left-hand sides y_j^-, y_j^+ and unknown coefficients a_i in Eq. (19.5) not to be negative. Therefore in case of the negative signs of the left-hand sides y_j^-, y_j^+, their inequalities should be multiplied by -1; similarly 'minus' signs of the negative coefficients a_i to be found are to be referred to the multipliers of these coefficients, thus reversing their signs. For instance, if the coefficient a_i in a product a_iz_i (where z_i equals to x_{ij}^- or x_{ij}^+) is negative, this element should be rearranged to the form of $-(-a_i)z_i$, where $-a_i$ is already positive. This actually carries out a transition from coefficients a_i to non-negative auxiliary coefficients b_i:

$$a_i = \text{sign}(a_i)b_i,\ b_i = |a_i| \geq 0,\ i = 1, \ldots, n.$$

For example, an inequality

$$5 \geq a_1 3 + a_2 4,$$

where $a_1 \geq 0$ and $a_2 \leq 0$, should be rewritten in the following manner:

$$5 \geq b_1 3 - b_2 4 \quad \text{or} \quad 5 \geq 3b_1 - 4b_2,$$

where $b_1 = a_1 \geq 0$ and $b_2 = -a_2 \geq 0$.

Minimization and maximization of the functions $f_i = b_i$, $i = 1, \ldots, n$, if $b_i \geq 0$ and limitations Eq. (19.5) are accordingly recorded, give corresponding minimum and maximum values of the model coefficients a_i which correspond to the region W_1. In case of Eq. (19.5) incompatibility, this fact is ascertained during the optimization process;

3. If the linear programming procedure results in the lack of solution of Eq. (19.5) (i.e., if there is no W_1 for the combination of the signs that have been found at the first stage) it is necessary to extend the boundaries of output disturbances. Instead of the values d_j^- and d_j^+ in Eqs. (19.2 and 19.3) the values $d_j^- - g$ and $d_j^+ + g$ are to be used here accordingly, where $g \geq 0$. The value g needs to be increased from 0 until system Eq. (19.5) remains compatible. Having achieved this try to decrease the value g and to change the signs of the coefficients in such a way that, when the value of g decreases, the corresponding Eq. (19.5) remains compatible. This may be done by means of iterative change of the signs of those coefficients that reach zero in the current situation (i.e., when the current value of g and the current signs of the coefficients are used). As a result, this iterative procedure gives the situation with $g = 0$ and the Eq. (19.5) being compatible. The corresponding subregions W_1 may be used in the following way.

If some coefficients in W_1 reach zero, their opposite signs should be checked: subregions W_2, $W_3, \ldots$ are then built. If some coefficients in this subregions reach zero then they also should be checked for opposite signs, and so on. When all subregions W_j have been found, the region W may be taken as their union.

The method that has been described may be illustrated by the following examples.

Suppose the results of five experiments for an object with two inputs are represented in Table 19.1; the corresponding maximum errors of measurements being:

$$c_{ij}^- = -0.1, \quad c_{ij}^+ = 0.1,$$

$$d_j^- = -0.5, \quad d_j^+ = 0.5, \quad i = 1,2, \quad j = 1, \ldots, 5.$$

Primarily describing the given object by the equation

$$y = a_1 x_1 + a_2 x_2, \tag{19.7}$$

TABLE 19.1. Experimental Data for Example 1

j	$\tilde{x}_1$	$\tilde{x}_2$	$\tilde{y}$
1	2.00	1.00	6.73
2	3.00	0.00	6.31
3	0.00	3.00	8.69
4	1.00	3.00	11.37
5	3.00	1.00	8.91

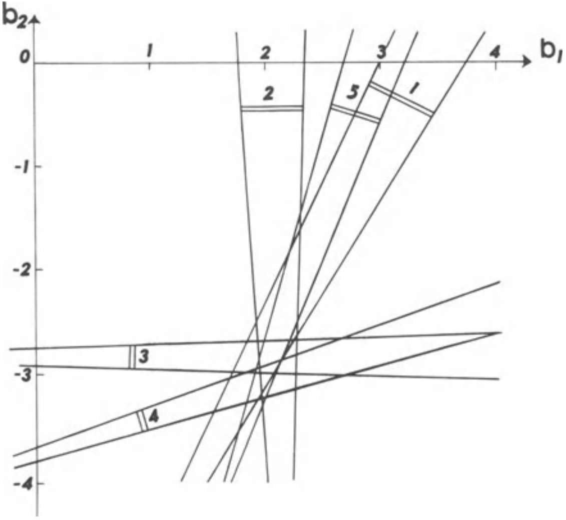


FIGURE 19.1. Lack of the region W_1 in Example 1 when $a_1 > 0$, $a_2 < 0$, $g = 0$.

choose the following combination of the coefficients signs: $a_1 \geq 0$, $a_2 \leq 0$ (such a combination of signs is chosen in deliberately to demonstrate the algorithm). Therefore the Eq. (19.5) here looks as follows:

$$\begin{cases} y_j^+ \geq b_1 x_{1j}^- - b_2 x_{2j}^+ \\ y_j^- \leq b_1 x_{1j}^+ - b_2 x_{2j}^-, \quad j = 1, \ldots, 5. \end{cases} \tag{19.8}$$

As stated above, investigating the system makes use of the linear programming method. With this goal Eq. (19.8) is rewritten as follows:

$$\begin{cases} y_j^+ \geq b_1 x_{1j}^- + b_2(-x_{2j}^+) \\ y_j^- \leq b_1 x_{1j}^+ + b_2(-x_{2j}^-), \quad j = 1, \ldots, 5, \end{cases} \tag{19.9}$$

which allows the specified method to be used. The solution of Eq. (19.9) is to be searched in the first quadrant. Fig. 19.1 shows that there is no such solution: Eq. (19.9) is incompatible in the first quadrant (here the regions, which correspond to individual measurements, are marked with corresponding figures). Therefore the value g has to be increased from 0 until the system becomes compatible in quadrant 1.

When $g = 9.5$, there is a region of solution W_1' touching the axis a_1 as shown on Fig. 19.2. Search for the solution changing the sign of the coefficient a_2 and decreasing the value of g. The final region $W_1' = W_1$ has been found if $a_1 > 0$, $a_2 > 0$ and $g = 0$ is represented in Fig. 19.3. By means of solving the corresponding linear

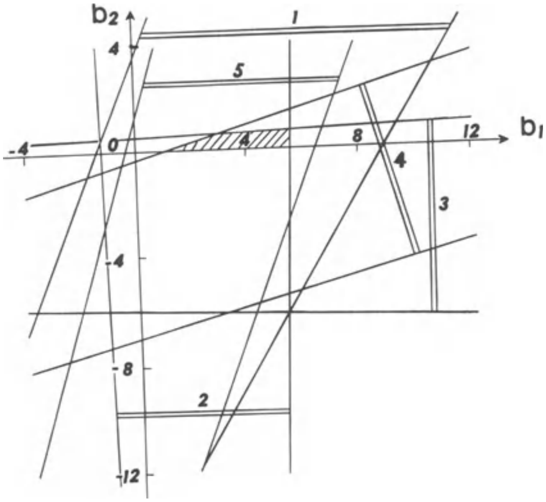


FIGURE 19.2. The region W_1' in Example 1 when $a_1 > 0$, $a_2 < 0$, $g = 9.5$.

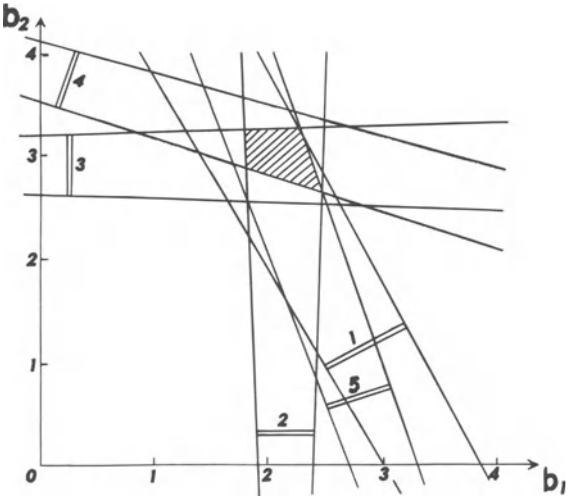


FIGURE 19.3. The region W_1 in Example 1 when $a_1 > 0$, $a_2 > 0$, $g = 0$.

TABLE 19.2. Experimental Data for Example 2

j	$\tilde{x}_1$	$\tilde{x}_2$	$\tilde{x}_3$	$\tilde{x}_4$	$\tilde{y}$
1	0.00	2.00	4.00	6.00	−19.64
2	2.00	4.00	6.00	0.00	16.17
3	4.00	6.00	0.00	2.00	−19.82
4	6.00	0.00	2.00	4.00	0.89
5	6.00	4.00	2.00	0.00	8.24
6	4.00	2.00	0.00	6.00	−27.71
7	2.00	0.00	6.00	4.00	7.98
8	0.00	6.00	4.00	2.00	−11.77
9	0.00	4.00	2.00	6.00	−33.95
10	4.00	2.00	6.00	0.00	26.75

programming problem the boundaries of the rectangle which circumscribes the region are

$$a_1^- = 1.77, \quad a_1^+ = 2.43,$$

$$a_2^- = 2.64, \quad a_2^+ = 3.25.$$

Consider a more complex example with four input signals and ten experiments as it is represented in Table 19.2; maximum errors of measurements are the same as in the first example.

The results of the first four experiments are used for preliminary estimation of the signs of the coefficients $a_1 \dots a_4$ to be found. This approach gives the following system:

$$\begin{cases} 0a_1 + 2a_2 + 4a_3 + 6a_4 = -19.64 \\ 2a_1 + 4a_2 + 6a_3 + 0a_4 = \ \ 16.17 \\ 4a_1 + 6a_2 + 0a_3 + 2a_4 = -19.82 \\ 6a_1 + 0a_2 + 2a_3 + 4a_4 = \ \ \ 0.89. \end{cases}$$

TABLE 19.3. Stage 1 for Example 2 ($g = 33.4$)

j	1	2	3	4
sign a_j	1	1	1	1
a_j^-	0.50	0.00	0.00	0.00
a_j^+	1.59	0.03	0.06	0.02

TABLE 19.4. Stage 2 for Example 2 ($g = 18.1$)

j	1	2	3	4
sign a_j	1	−1	1	1
a_j^-	1.70	−7.95	3.72	0.00
a_j^+	2.05	−7.32	4.01	0.03

TABLE 19.5. Stage 3 for Example 2 ($g = 0.0$)

j	1	2	3	4
sign a_j	1	−1	1	−1
a_j^-	1.62	−3.47	3.53	−5.45
a_j^+	2.55	−2.56	4.48	−4.49

The solution gives $a_1 = 2.10$, $a_2 = -3.06$, $a_3 = 4.03$, $a_4 = -4.94$. Thus the primary combination of the signs is as follows: $a_1 > 0$, $a_2 < 0$, $a_3 > 0$, $a_4 < 0$. But further on it is demonstrated that these are the real signs of the coefficients to be found. Thus if the specified combination of the signs is used as an initial one it results in the failure to demonstrate the functioning of the identification algorithm that has been described: the Eq. (19.5) becomes compatible at once with $g = 0$. Therefore, choose a deliberately false combination of the signs as an initial one, e.g., $a_i > 0$, $i = 1, \dots, 4$. Successive tables of intermediate results obtained by the specified algorithm illustrate the major stages of solving the second example by means of changing the signs of the coefficients a_i and the value g.

Table 19.3 represents the first step: change the value of g from 0 to 33.4, when the inequalities in Eq. (19.5) are compatible for the initial signs of the coefficients. Here the lower meanings of a_2, a_3 and a_4 are equal to zero, so one can change the sign of anyone from them. Invert the sign of a_2 and reduce g up to $g = 18.1$; the compatible Eq. (19.5) is represented in Table 19.4. Then, invert the sign of a_4 and reduce g until $g = 0$, to obtain the final decision, which is reflected in Table 19.5.

19.4. CONCLUSIONS

A method to identify linear objects with unknown bounded disturbances in both input and output channels has been presented. It includes construction and investigation of possible values of model parameters that are consistent with both the structure of the model and the experimental information.

The region W of possible values of the model parameters is described by a system of linear inequalities. It is possible to investigate this region using linear programming. A method to construct W in situations where signs of the model parameters are not known *a priori* has been described and illustrated using two concrete situations: with (1) two input channels and five experiments, and (2) four input channels and ten experiments.

REFERENCES

1. M. Milanese and G. Belforte, *IEEE Trans. Autom. Control* **27**, 408 (1982).
2. E. Fogel and Y.-F. Huang, *Automatica* **18**, 229 (1982).
3. J. P. Norton and S. H. Mo, *Comput. Simul.* **32**, 527 (1990).
4. E. Walter and H. Piet-Lahanier, in: *Proceedings of the 26th IEEE Conference on Decision and Control*, pp. 1921–1922 (1987).
5. J. P. Norton, *Int. J. Control* **45**, 375 (1987).
6. V. Cerone, in: *Prep. 9th IFAC/IFORS Symposium on Identification and System Parameter Estimation*, pp. 1518–1522 (1991).
7. E. Walter and H. Piet-Lahanier, *Math. Comput. Simul.* **32**, 449 (1990).
8. Y. A. Merkuryev, *Int. J. Control.* **50**, 2333 (1989).
9. Y. A. Merkuryev, *Minimax Estimation of the Model Parameters for Control Objects when the Initial Information is of Interval Character*, Ph.D. Thesis, Riga Polytechnical Institute, Riga, Latvia (1982).
10. W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes, The Art of Scientific Computing,* Cambridge University Press, Cambridge (1986).

20

Identification of Nonlinear State-Space Models by Deterministic Search

J. P. Norton and S. M. Veres

ABSTRACT

An economical technique for tracing the boundary of a two-dimensional cross section of the feasible parameter set for a model with bounded output error is described. It allows exploration of a boundary which is not piecewise linear and may not be convex. First a point on the boundary is found, then a line search is executed, adapting to local behavior of the boundary. Resolution may be traded against computational speed by choice of the search parameters.

20.1. INTRODUCTION

A search method for computing an approximation to the active parameter bounds of bounded-error nonlinear state-space models is presented. The method is fast enough to allow exploration, using a PC, of complicated parameter bounds of non-linear models. The bounds are those implied by bounds on the model-output error, given a set of observations and a model structure. They represent a simple

J. P. Norton and S. M. Veres • School of Electronic and Electrical Engineering, University of Birmingham, Edgbaston, Birmingham B15 2TT, United Kingdom.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

description of model uncertainty, based on minimal information about the discrepancy between the model output and the observations. If the model-output error bounds and model structure are well chosen, the parameter bounds forming the boundary of the feasible parameter set (FPS) may be used to classify the output record (as in the example given later), or for experiment design, worst-case control design or robust prediction.

The simplest possible approach[(1)] is to compute boundary points of one parameter for a range of values of a second, with all other parameters fixed, thereby generating a two-dimensional cross section of the FPS. The obvious drawback of this method is the need to compute values over a fine grid of parameter values in order to obtain a clear indication of the shape of the FPS; this limits its use to short records and models with few parameters. Choice of grid spacing is a non-trivial task.

A Monte Carlo technique has been used for non-linear parameter bounding[(2)] by Smit,[(3)] and Smit and van Vliet[(4)] devise a technique for computing estimates of lower-dimension projections of the FPS.[(4)] Lahanier, Walter, and Gomeni[(5)] give a random-search algorithm which generates a cloud of points on the boundary of the FPS. It is able to deal with complicated boundaries and with feasible sets which consist of a number of separate subsets,[(6)] but is computationally expensive. It has been applied to the identification of pharmacokinetic models.

Recently developed methods are based on signomial programming and subdivision of parameter space into ever smaller orthotopes.[(7–9)] These methods are guaranteed to determine the boundary of the feasible set to a specified resolution. The line-searching method presented here, an early version of which is described in Ref. 10 is deterministic, like those of Refs. 7–9, in that the computation of the boundary is entirely determined by the behavior of the boundary, once a starting point has been specified. The line searching adjusts directions and step sizes to match the local surface shape. This improves efficiency of the computation; random searching to identify independent points on the boundary is slow by comparison. The adaptive features of the algorithm make precomputation of the resolution complicated, although bounds on the errors in the boundary can be computed on line from local step sizes and angles of direction changes.

20.2. PROBLEM FORMULATION

The discrete-time state-space model

$$\mathbf{x}_t = \mathbf{f}(\mathbf{x}_{t-1},\boldsymbol{\theta}),\ \ y_t = g(\mathbf{x}_t) + e_t,\ \ t = 1,2,\ldots,N \tag{20.1}$$

(which may result from discretization of a continuous-time model) is considered. Here

$$y_t \in \mathbb{R}^1, \mathbf{x}_t \in \mathbb{R}^n,$$

and **f** is a possibly non-linear analytic function, fully determined for a given feasible parameter vector $\boldsymbol{\theta} \in \mathrm{D} \subset \mathbb{R}^k$ and three times differentiable in $\boldsymbol{\theta}$ on the open connected parameter domain $\mathrm{D} \subset \mathbb{R}^k$. The model-output error e_t has specified bounds $|e_t| \leq \delta$ (the symmetry of which is achieved by trivial adjustment of y_t if necessary). The error term combines observation error and any structural error in the state and observation equations. Separate state-equation error, although sometimes of interest, is not considered here.

The problem is to compute the initial state $\mathbf{x}_o$ and all values of parameters $\boldsymbol{\theta}$ consistent with Eq. (19.1) and the model-output error bounds, i.e., the set

$$\mathcal{D} = \{(\mathbf{x}_o,\boldsymbol{\theta}) \in I \times D \mid \mathbf{x}_t = \mathbf{f}(\mathbf{x}_{t-1},\boldsymbol{\theta}), |y_t - g(\mathbf{x}_t)| \leq \delta, t = 1,2,\ldots,N\}$$

where $S \subset \mathbb{R}^k$ is the set of initial states known *a priori* to be possible. If $\mathbf{x}_o$ is known,

$$\mathcal{D} = \{\boldsymbol{\theta} \in D \mid \mathbf{x}_t = \mathbf{f}(\mathbf{x}_{t-1},\boldsymbol{\theta}), |y_t - g(\mathbf{x}_t)| \leq \delta, t = 1,2,\ldots,N\}.$$

20.3. COMPUTATION OF TWO-DIMENSIONAL CROSS-SECTIONS

Two-dimensional cross-sections of the FPS $\mathcal{D}$ are computed by fixing $k-2$ of the parameters and exploring in the plane of the remaining two. Extension to higher dimensions is described in Section 20.4. The procedure presented has a number of advantages. There is no need to know an initial feasible point; all that is required is specification of a bounded, finite parameter/initial state set to be explored. The ability of the procedure to follow the boundary of the feasible set is limited only by the resolution implied by the minimum length and direction change specified for the search steps. Stage I of the algorithm, described below, can be repeated to detect disjoint parts of the FPS (which occur in quite simple practical situations(6,11)).

To simplify the notation, $\mathbf{x}_o$ is included in the parameter vector $\boldsymbol{\theta}$, increasing its dimension k. Assume that the cross-section to be explored is in the plane of the first two parameters, and denote $\boldsymbol{\theta}$ by $[\theta_1,\theta_2,\theta_3^{\mathrm{T}}]^{\mathrm{T}}$, with $\boldsymbol{\theta}_3$ fixed. The algorithm searches the (θ_1,θ_2)-plane for all points such that $[\theta_1,\theta_2,\theta_3^{\mathrm{T}}]^{\mathrm{T}} \in \mathcal{D}$. Let $\mathcal{A}$ denote the prior feasible (θ_1,θ_2) set, usually much larger than the cross section of $\mathcal{D}$ to be computed. The prior feasible set $\mathcal{A}$ may depend on $\boldsymbol{\theta}_3$. At each step, the procedure determines whether a trial $\boldsymbol{\theta}$ belongs to $\mathcal{D}$ and calculates the gradient of the largest model-output error with respect to $\boldsymbol{\theta}$. The error sequence and its gradients $\boldsymbol{\psi}$ are computed recursively by

$$\mathbf{x}_t = \mathbf{f}(\mathbf{x}_{t-1},\boldsymbol{\theta}), \;\; e_t = y_t - g(\mathbf{x}_t,\boldsymbol{\theta}), \;\; t = 1,2,\ldots,N \qquad (20.2)$$

$$\frac{\partial}{\partial \boldsymbol{\theta}^*}\mathbf{x}_t = \frac{\partial}{\partial \boldsymbol{\theta}^*}\mathbf{f}(\mathbf{x}_{t-1},\boldsymbol{\theta}) + \frac{\partial}{\partial \mathbf{x}_{t-1}}\mathbf{f}(\mathbf{x}_{t-1},\boldsymbol{\theta})\frac{\partial}{\partial \boldsymbol{\theta}^*}\mathbf{x}_{t-1}, \text{ and}$$

$$\psi_t \equiv \frac{\partial}{\partial \boldsymbol{\theta}^*} e_t = -\frac{\partial}{\partial \boldsymbol{\theta}^*} g(\mathbf{x}_t,\boldsymbol{\theta}) - \frac{\partial}{\partial \mathbf{x}_t} g(\mathbf{x}_t,\boldsymbol{\theta})\frac{\partial}{\partial \boldsymbol{\theta}^*}x_t, \quad t = 1,2,\ldots,N \tag{20.3}$$

where $[\theta_1\theta_2]^{\mathrm{T}}$ has been written as $\boldsymbol{\theta}^*$. Let the largest output error be

$$C(\boldsymbol{\theta}) = \max\{|e_t(\boldsymbol{\theta})|, t = 1,\ldots,N\}$$

and let $t_1(\boldsymbol{\theta}),\ldots,t_{r(\boldsymbol{\theta})}(\boldsymbol{\theta})$ be the time instants where the errors take this maximal value within a tolerance κ, i.e.,

$$e_{t_i}(\boldsymbol{\theta}) \simeq C(\boldsymbol{\theta}), i = 1,2,\ldots,r(\boldsymbol{\theta}) \tag{20.4}$$

or

$$e_{t_i}(\boldsymbol{\theta}) \simeq -C(\boldsymbol{\theta}), i = 1,2,\ldots,r(\boldsymbol{\theta}) \tag{20.5}$$

where approximate equality $a \simeq b$ holds within tolerance κ if $|a - b| < \kappa$. In the algorithm below, all approximate equalities are understood to hold in this way. The number $r(\boldsymbol{\theta})$ of instants at which the largest model-output error occurs is in practice usually one or two, but consider the general case for completeness. To simplify notation, denote by $\psi_i \in \mathbb{R}^2, i = 1,\ldots,r(\boldsymbol{\theta})$ the gradients

$$\partial e_{t_i}(\boldsymbol{\theta})/\partial\boldsymbol{\theta}^* \text{ or } -\partial e_{t_i}(\boldsymbol{\theta})/\partial\boldsymbol{\theta}^*,$$

according as Eq. (20.4) or (20.5) holds, with $\psi_{ij} \equiv \pm\, \partial e_{t_i}(\boldsymbol{\theta})/\partial\theta_j^*$.

The algorithm starts from a finite set $\mathcal{J}$ of initial points $\boldsymbol{\theta}_o^* = [\theta_{1o}\ \theta_{2o}]^{\mathrm{T}} \in \mathcal{A}$ not necessarily in the projection of $\mathcal{D}$ onto the $\boldsymbol{\theta}^*$ plane. Finite set $\mathcal{J}$ may be chosen at will, but is best spread uniformly around $\mathcal{A}$.

20.3.1. Stage I: Reach the Boundary of the FPS

Step (1) Calculate normalized gradient

$$\psi = \psi_o/\|\psi_o\| \text{ at } \boldsymbol{\theta}_o \text{ and set } \boldsymbol{\theta} \text{ to } \boldsymbol{\theta}_o.$$

Step (2) Search for a point on the half-line

$$\boldsymbol{\theta}' = \boldsymbol{\theta} + \lambda\psi, \lambda > 0,$$

at which $r(\boldsymbol{\theta}') \geq 2$. The search successively halves and doubles λ as necessary to search for a point with

$$C(\boldsymbol{\theta}') < \delta \text{ and } t_1(\boldsymbol{\theta}') = t_1(\boldsymbol{\theta})$$

close to another point

$$\boldsymbol{\theta}'' = \boldsymbol{\theta} + \lambda'\boldsymbol{\psi}$$

(with $\lambda \simeq \lambda'$) such that

$$C(\boldsymbol{\theta}'') \geq \delta \text{ or } t_1(\boldsymbol{\theta}'') \neq t_1(\boldsymbol{\theta}).$$

If $C(\boldsymbol{\theta}'') \simeq \delta$, pass to Stage II.

If $r(\boldsymbol{\theta}') \geq 2$ during the search there are three possible cases:

(a) If

$$C(\boldsymbol{\theta}') > \delta \text{ and at } \boldsymbol{\theta}', \mathbf{0} \in Conv\{\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_r\} \text{ (the convex hull of } \boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_r),$$

then for any given small change $\delta\boldsymbol{\theta}$ in $\boldsymbol{\theta}'$, some positive-weighted sum of the resulting changes $\boldsymbol{\psi}_i^{\mathrm{T}}\delta\boldsymbol{\theta}$, $i = 1,2, \ldots, r$ in maximal errors is zero. Hence not all the changes are of the same sign, so one cannot simultaneously reduce all $|e_{t_i}|$, which is necessary to reduce $C(\boldsymbol{\theta}')$. Restart, therefore, from another initial $\boldsymbol{\theta} \in \mathcal{J}$;

(b) If

$$C(\boldsymbol{\theta}') > \delta \text{ and at } \boldsymbol{\theta}', \mathbf{0} \notin Conv\{\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_r\},$$

set $\boldsymbol{\theta}$ to $\boldsymbol{\theta}'$ then

$$\boldsymbol{\theta}' = \boldsymbol{\theta} + \lambda \frac{1}{r(\boldsymbol{\theta})} \sum_i \boldsymbol{\psi}_i, \quad \lambda < 0,$$

in an attempt to get both $r(\boldsymbol{\theta}') \geq 2$ and $C(\boldsymbol{\theta}') \simeq \delta$, and return to the start of Step (2);

(c) If

$$\mathbf{0} \in Conv\{\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_r\} \text{ and } C(\boldsymbol{\theta}') \simeq \delta$$

(at the boundary), check whether all $\boldsymbol{\psi}_i$, $i = 1, \ldots, r$ can be written as $\boldsymbol{\psi}_i = \lambda_i\boldsymbol{\psi}$ for some $\boldsymbol{\psi}$. If so, every $|e_{t_i}|$ is unchanged by $\delta\boldsymbol{\theta}$ orthogonal to $\boldsymbol{\psi}$, so in Stage II search the boundary of $\mathcal{D}$ along the line orthogonal to $\boldsymbol{\psi}$.

If one reaches the specified maximum number of iterations of this search procedure, restart from another initial $\boldsymbol{\theta} \in \mathcal{J}$. If all $\boldsymbol{\theta} \in \mathcal{J}$ have been tried, exhaustive checking of points on a uniform grid filling the whole of $\mathcal{A}$ can be used to find an internal point of the feasible parameter set $\mathcal{D}$. Alternatively, the (non-linear) least-squares estimate may be an internal point. A search then starts for a boundary point of $\mathcal{D}$, using the Procedure B described later in place of the procedure of Stage I.

20.3.2. Stage II: Follow the Boundary of the FPR $\mathcal{D}$

Stage I provides a point $\boldsymbol{\theta}_1$ on the boundary of $\mathcal{D}$, and the maximal-error gradients $\boldsymbol{\psi}_i$, $i = 1, \ldots, r(\boldsymbol{\theta}_1)$. If $r(\boldsymbol{\theta}_1) = 1$ then the initial search direction

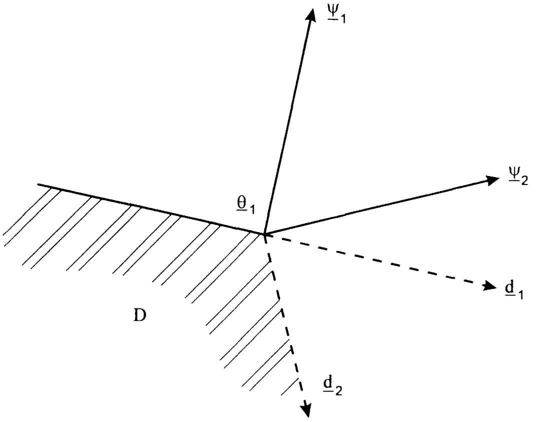


FIGURE 20.1. Direction of search from $\boldsymbol{\theta}_1$ for $r(\boldsymbol{\theta}_1) = 2$.

$\mathbf{d} \equiv [d_1\ d_2]^{\mathrm{T}}$ should turn at right angles to $\boldsymbol{\psi}_1$ along the boundary, so $\mathbf{d} = [\psi_{12} - \psi_{11}]^{\mathrm{T}}$ will do. If $r(\boldsymbol{\theta}_1) = 2$, $\boldsymbol{\theta}_1$ is at the intersection of two bounds and the appropriate direction is $\mathbf{d} = [\psi_{12}\ {-\psi_{11}}]^{\mathrm{T}}$ if $\psi_{12}\psi_{21} < \psi_{22}\psi_{11}$ and $\mathbf{d} = [\psi_{22}\ {-\psi_{21}}]^{\mathrm{T}}$ if $\psi_{12}\psi_{21} \geq \psi_{22}\psi_{11}$. Fig. 20.1 illustrates this situation. The dashed lines show the two tangents to the FPS boundary at $\boldsymbol{\theta}_1$; gradient $\boldsymbol{\psi}_2$ is obtained by turning $\boldsymbol{\psi}_1$ in a clockwise direction by less than π rad, which gives $\psi_{12}\psi_{21} < \psi_{22}\psi_{11}$. Similarly, if $r(\boldsymbol{\theta}_1) > 2$ one selects from $\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_{r(\boldsymbol{\theta}_1)}$ that $\boldsymbol{\psi}_k$ from which all the other $\boldsymbol{\psi}_i$s can be obtained by turning $\boldsymbol{\psi}_k$ in a positive direction by an angle less than or equal to π. The initial direction of search from $\boldsymbol{\theta}_1$ is then $\mathbf{d} = [\psi_{k2} - \psi_{k1}]^{\mathrm{T}}$. In the unlikely event that no such $\boldsymbol{\psi}_k$ exists, the two-dimensional cross section in the neighborhood of $\boldsymbol{\theta}_1$ consists of the singleton $\{\boldsymbol{\theta}_1\}$.

Starting from $\boldsymbol{\theta}_1$ the feasibility of three trial points A_1, A_2, A_3, displaced from $\boldsymbol{\theta}_1$ by vectors of lengths $\lambda_{1,2,3}$ and angles $\alpha_{1,2,3}$, is checked. The lowest acceptable resolution fixes the initial values of $\lambda_{1,2,3}$ and the spacing of $\alpha_{1,2,3}$. The latter are successive integer multiples of a specified angle α_0, not critical but typically between 10° and 20°. In Fig. 20.2, "×" indicates a point in $\mathcal{D}$ and "o" a point not in $\mathcal{D}$. In Fig. 20.2(a), the boundary crosses the intervals $[\boldsymbol{\theta}_1, A_i]$ and $[A_i, A_j]$ for some i and $i \neq j$, $i,j = 1,2,3$. In Fig. 20.2(b), the step size μ should be reduced by a specified factor, e.g., 2. In Fig. 20.2(c), the direction of search should be turned in a positive or negative direction by α_0. The left picture in Fig. 20.2(a) shows that it is not enough to look for a boundary point between $\boldsymbol{\theta}_1$ and a point outside $\mathcal{D}$, as the boundary crossing may coincide with $\boldsymbol{\theta}_1$, in which case a boundary point must be sought between the A_is. The right picture in Fig. 20.2(a) shows, however, that a boundary point between $\boldsymbol{\theta}_1$ and A_i should be sought first to turn at the right place.

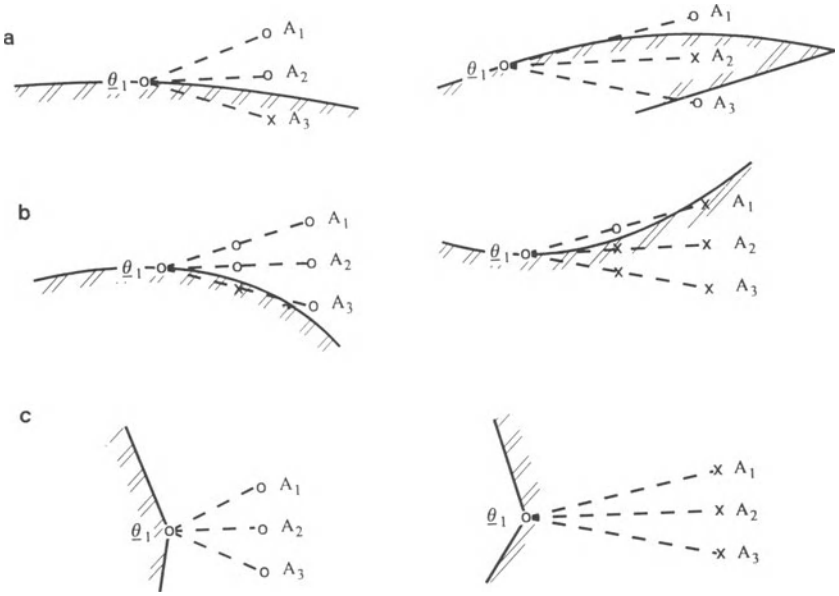


FIGURE 20.2. Examples of behavior during boundary search. (a) Trial points of differing feasibility; (b) Trial points of same feasibility; and (c) At a nondifferentiable boundary point ($r(\boldsymbol{\theta}_1) > 1$)

At this stage a procedure is needed to find a boundary point between an inside and an outside point. Let $\boldsymbol{\theta}_x$ be the point inside $\mathcal{D}$, $\boldsymbol{\theta}_y = \boldsymbol{\theta}_x + \lambda\mathbf{d}$ the next trial point and a $\mathcal{F}$ Boolean variable with initial value $\mathcal{F} = \{\boldsymbol{\theta}_y \in \mathcal{D}\}$. The following "halving-doubling" procedure finds a boundary point. It uses $\mathcal{F}$ to indicate whether the previously considered point is inside or outside the FPS.

```
$\boldsymbol{\theta}_z := \boldsymbol{\theta}_x$;
start: $\boldsymbol{\theta}_x := \boldsymbol{\theta}_z + \lambda\mathbf{d}$;
    if ($\mathcal{F}$ is true and $\boldsymbol{\theta}_x \in \mathcal{D}$) then $\lambda := 2\lambda$;
    if ($\mathcal{F}$ is true and $\boldsymbol{\theta}_x \notin \mathcal{D}$) then ($\boldsymbol{\theta}_z := \boldsymbol{\theta}_z + \lambda\mathbf{d}/2$; $\lambda := \lambda/4$;
        $\mathcal{F}$ := false; go to start);
    if ($\mathcal{F}$ is false and $\boldsymbol{\theta}_x \in \mathcal{D}$) then ($\boldsymbol{\theta}_z := \boldsymbol{\theta}_x$; $\lambda := \lambda/2$);
    if ($\mathcal{F}$ is false and $\boldsymbol{\theta}_x \notin \mathcal{D}$) then $\lambda := \lambda/2$;
    if $\lambda < \kappa$ then stop else go to start.
```

The algorithm has worked well in many examples with piecewise differentiable boundaries. Fig. 20.3 shows a low-resolution solution with large initial and maximum search-step sizes, and a higher-resolution solution with smaller step sizes. Noteworthy features are the complexity of the boundary formed by a modest

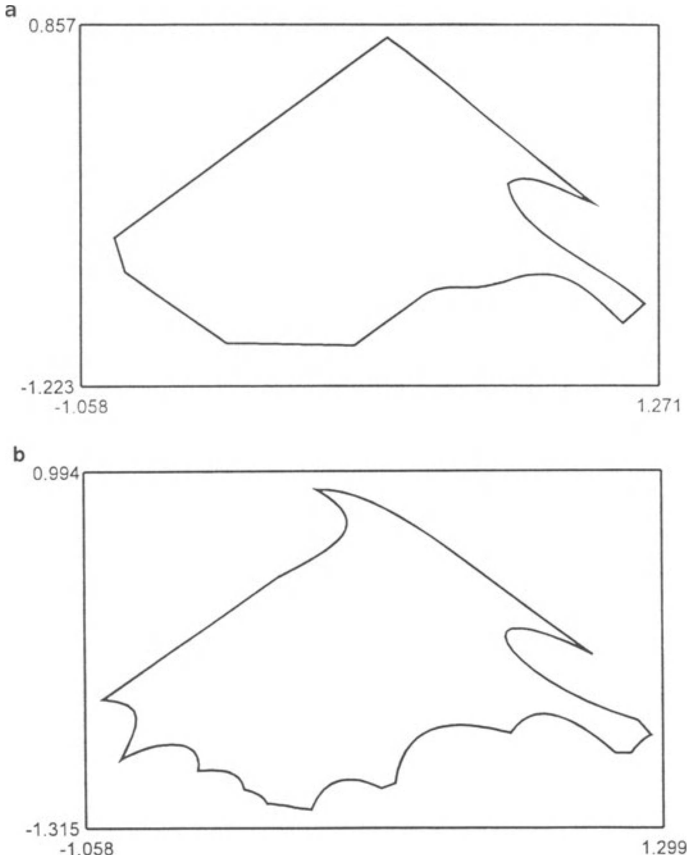


FIGURE 20.3. Cross-sections obtained by two-dimensional boundary search (a) low resolution (b) high resolution

number of simple bounds, and the ability of the algorithm to follow it approximately even when the search parameters are not well chosen.

20.4. EXTENSION TO MORE DIMENSIONS

The extension of the algorithm described above is an indexed set of two-parameter searches. Starting from $\boldsymbol{\theta}_o \in \mathcal{D}$, boundary points are found in a succession of uniformly distributed directions. The direction vectors $\mathbf{d}_i$ are all of the form $[\beta_1\ \beta_2 \ldots \beta_k]^{\mathrm{T}}$, where

$$\beta_1 = \cos\alpha_1,\ \beta_2 = \sin\alpha_1 \cos\alpha_2,\ \beta_3 = \sin\alpha_1 \sin\alpha_2 \cos\alpha_3, \ldots$$

$$\ldots, \beta_{k-1} = \prod_{j=1}^{k-2} \sin \alpha_j \cos \alpha_{k-1}, \beta_k = \prod_{j=1}^{k-1} \sin \alpha_j.$$

Uniform distribution is arranged by setting

$$\alpha_j = 2\pi m_j/m, j = 1,2, \ldots, k-1$$

with each integer m_j indexed over $1,2, \ldots, m$, generating altogether m^{k-1} direction vectors. The search is confined to two dimensions by fixing the values of all but one m_j. Each two-dimensional boundary exploration, as in Section 20.3, yields a closed polygon of successful linear search steps. The end result is a set of two-dimensional cross-sections of $\mathcal{D}$ at m^{k-1} points of a $(k - 1)$-dimensional grid covering the surface of the k-dimensional FPS.

The initial search for a boundary point along the line $\boldsymbol{\theta}_o + \lambda\mathbf{d}_i$, $\lambda \geq 0$ is carried out by halving and doubling the search step in the procedure described in the preceding section.

As the list of vertices or edges of the polygons approximating the two-dimensional cross sections makes up a complicated description of $\mathcal{D}$, there remains the non-trivial problem of finding an economical and readily comprehensible way to characterize $\mathcal{D}$ adequately for the intended application of the model. However, considerable insight into the character of the FPS. can be obtained by inspecting two- or three-dimensional cross sections of its boundary, as in the example below.

20.5. EXAMPLE

Assessment of the uncertainty in the model parameters by bounds rather than by a covariance matrix is attractive when the central-limit theorem is inapplicable, e.g., with small samples and/or heavily structured errors. The example chosen therefore involves a nonlinear model with only a small number of samples in the input-output record. Fig. 20.4 shows the response of blood-plasma concentration in a human subject to a rapid oral dose of methionine to test liver function. The response suggests a two-exponential model

$$y(t) = a[\exp b_1(t - \tau) - \exp b_2(t - \tau)] + e(t) \tag{20.6}$$

where t denotes time and τ an unknown pure delay. (Because the sampling is nonuniform in time, one cannot rewrite the model as a difference equation linear in its parameters as usual.) The model-output error bounds are provisionally specified from knowledge of instrument and experimental accuracy to be $|e(t)| \leq 10$. Bounds on the parameter vector $\boldsymbol{\theta} = [a\ b_1\ b_2\ \tau]^{\mathrm{T}}$ are to be computed.

The initial value $\boldsymbol{\theta}_o = [229\ {-0.94}\ {-3.4}\ 0.15]^{\mathrm{T}}$ was found to be inside the feasible parameter set. Figure 20.5 shows two-dimensional cross-sections through

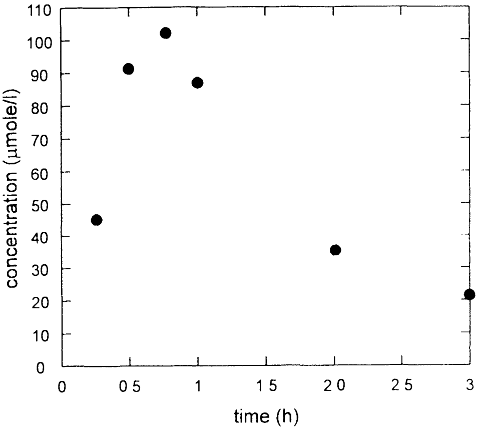


FIGURE 20.4. Methionine tolerance test response: evolution of plasma concentration of methionine.

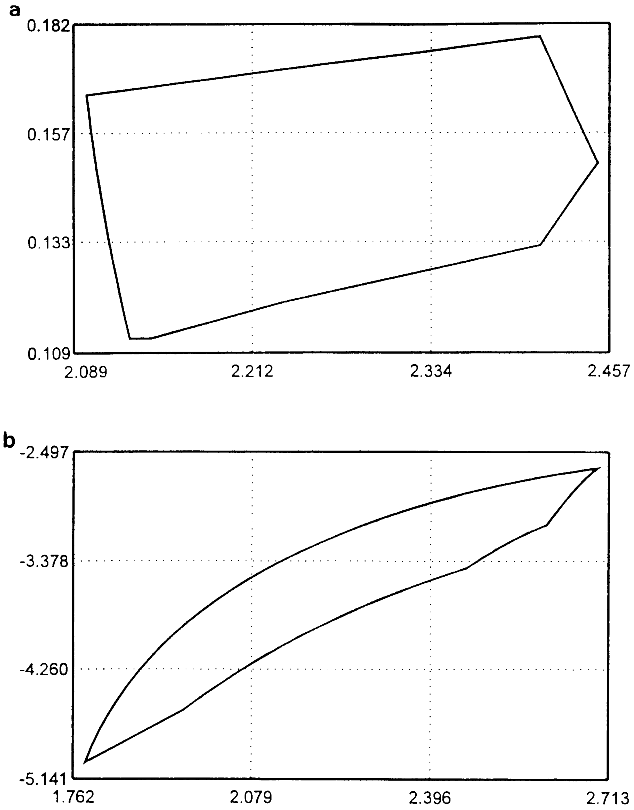


FIGURE 20.5. Two-dimensional cross sections of the feasible parameter set, all through $\boldsymbol{\theta}_o$.

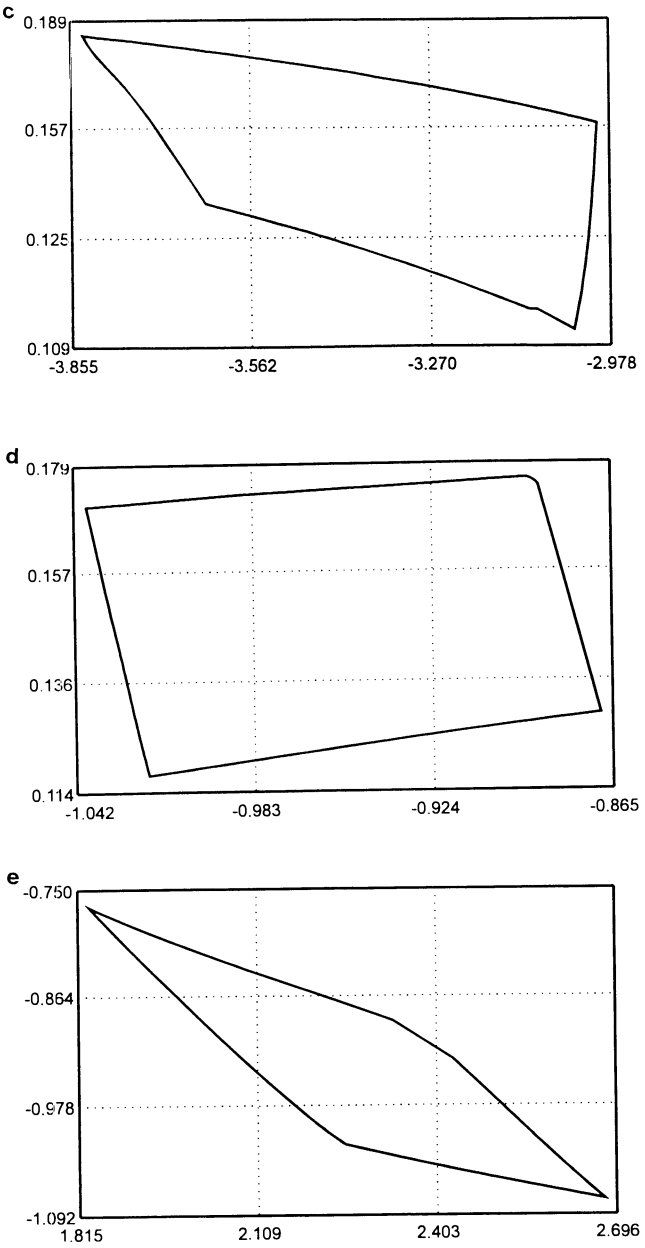


FIGURE 20.5. (Continued)

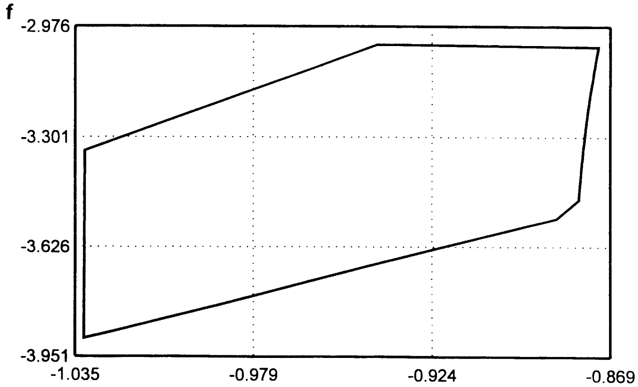


FIGURE 20.5. (Continued)

θ_o. The first parameter a has been normalized by 100 to make its magnitude comparable with those of the other parameters. There are as many two-dimensional cross sections as the number of ways to chose two out of four parameters, i.e., six. The 1-2 and 1-3 sections in Fig. 20.5 suggest that the FPS almost falls within a linear subspace. At least one linear function of the parameters is near-redundant, which implies that one or more parameters can be eliminated with little detriment to the model. A clearer indication that this is so comes in Fig. 20.6, which shows isometric views of the three-dimensional cross section of the FPS at $a = 229$. Fig. 20.6 shows that the FPS is thin in directions orthogonal to $[0\ 1\ -0.7\ -2]^T$.

The nonlinear bounding facilities offered by the University of Birmingham identification package also include projection of the FPS onto any specified two-dimensional subspace. Fig. 20.7 shows the FPS of Fig. 20.6 projected onto the b_2:τ plane. The package provides advice to the user in two forms: a HELP facility and a file of information about the progress of the search, accumulated automatically as the search proceeds and accessible at any stage. This file contains, for instance, the number of search line segments so far and the worst-case parameter precision achieved. The user may adjust the search parameters on the basis of this information.

The results screens show the line search in progress, with the next few trial points and search directions. The user may decide to intervene according to this information. The package also provides for linear transformation of the parameters, which is useful in cases of near-redundancy like that in Fig. 20.6. It helps in optimizing the search-direction density. The package contains a parser to interpret state and observation equations typed in algebraic form in a standard notation.

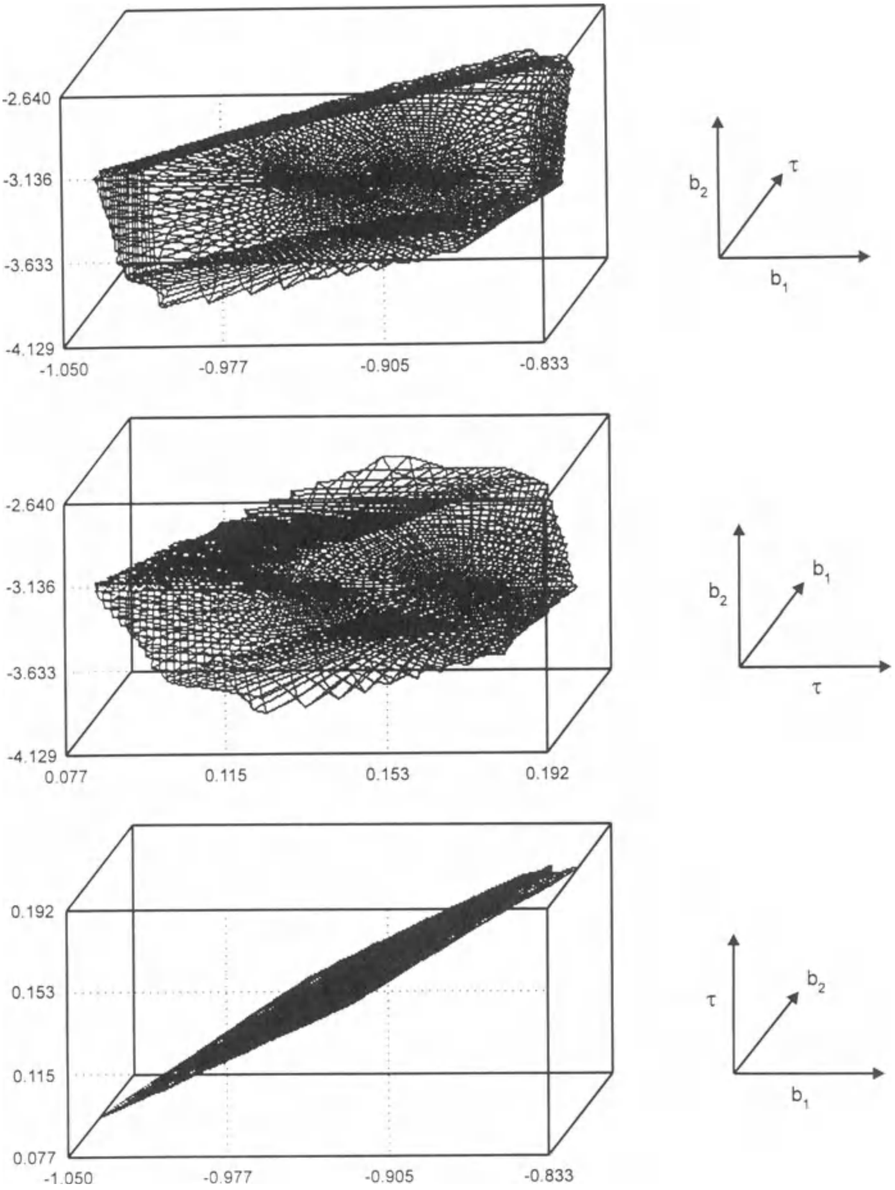


FIGURE 20.6. Isometric views of three-dimensional cross section of FPS at $a = 229$.

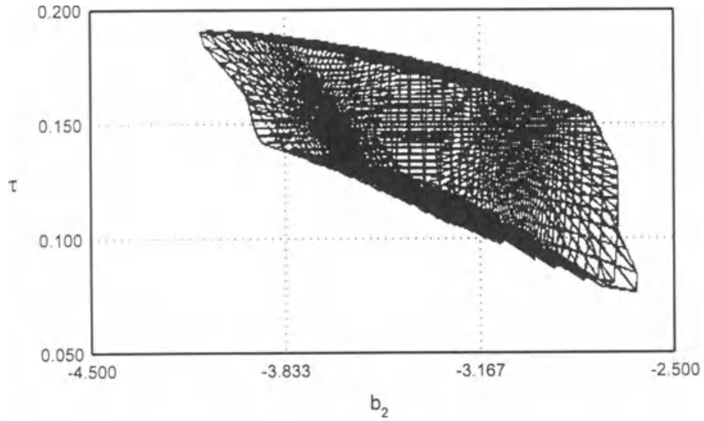


FIGURE 20.7. Projection of FPS onto plane.

20.6. CONCLUSIONS

A method has been presented for computing the boundary, possibly nonconvex and complicated, of the feasible parameter set of a nonlinear state-space model. It performs line searches along the boundaries of two-dimensional cross sections of the FPS. Higher-dimensional regions are explored by a succession of two-dimensional searches. The computing load makes nonlinear bounding mainly an off-line identification technique. The fundamental difficulty of finding comprehensible and economical characterizations of possibly complicated multidimensional surfaces also has to be faced. Nevertheless, the algorithm enables the user to examine two-dimensional cross sections and projections of the FPS boundary, which can give valuable insight into the adequacy of the model parameterization and experimental conditions. The nonlinear FPS boundary computation is embedded in the University of Birmingham identification package, which has a number of features to facilitate interactive exploration.

REFERENCES

1. J. P. Norton, *Biomed. Meas. Inf. Control.* **2**, 101 (1987).
2. K. Fedra, G. Van Straten, and M. B. Beck, *Eco. Model.* **13**, 87 (1981).
3. M. K. Smit, *Measurement* **1**, 181 (1983).
4. M. K. Smit and Ch. Van Vliet, *Measurement* **1**, 209 (1983).
5. H. Lahanier, E. Walter, and R. Gomeni, *J. Pharmacokin.* **15**, 203 (1987).
6. H. Piet-Lahanier and E. Walter, *Math. Comput. Simul.* **32**, 553 (1990).
7. M. Milanese and A. Vicino, *Automatica* **27**, 404 (1991).
8. L. Jaulin and E. Walter, *Automatica* **29**, 1053 (1993).

9. L. Jaulin and E. Walter, *Math. Comput. Simul.* **35**, 123 (1993).
10. S. M. Veres and J. P. Norton, in: *Proceedings of the 9th IASTED International Conference on Modelling, Identification & Control*, Innsbruck, Austria, pp. 367–370 (1990).
11. H. Piet-Lahanier, *Estimation de Paramètres pour des Modèles à Erreur Bornée*, Thèse de Docteur en Sciences Univ. de Paris-Sud, Centre d'Orsay, Paris (1987).

21

Robust Identification and Prediction for Nonlinear State-Space Models with Bounded Output Error

K. J. Keesman

21.1. INTRODUCTION

An important application of mathematical models is prediction of the future system behavior. Due to incomplete system knowledge as well as errors in the observations obtained from the "real" system, these models will always contain some uncertainty. Hence, for the credibility of model predictions, it is desirable to quantify the prediction uncertainty. From this point of view, a single future trajectory suggest an unrealistic reliability.

In a large number of applications, prediction uncertainty is dominated by uncertainty in uncontrolled future system inputs, which is always speculative. In order to illustrate the contribution of other uncertainties, an appropriate model structure is first presented. As a result of it, consider the following finite-dimensional, continuous-discrete time, nonlinear, time-invariant state-space model structure without system noise,

K. J. KEESMAN • Department of Agricultural Engineering and Physics, University of Wageningen, 6703 HD Wageningen, The Netherlands.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

$$\begin{aligned} & d\mathbf{x}(t,\boldsymbol{\theta})/dt = \mathbf{f}[\mathbf{x}(t,\boldsymbol{\theta}),\mathbf{u}(t),t;\boldsymbol{\theta}] \\ & \mathbf{x}(t_0,\boldsymbol{\theta}) = \mathbf{x}_0 \\ & \mathbf{y}(t_k) = \mathbf{g}[\mathbf{x}(t_k,\boldsymbol{\theta}),\mathbf{u}(t_k),t_k;\boldsymbol{\theta}] + \mathbf{e}(t_k) \quad t_k = t_1, \ldots, t_N \end{aligned} \tag{21.1}$$

where $\mathbf{x}(.)$, $\mathbf{u}(.)$ and $\boldsymbol{\theta}$ are the state, input and parameter vector. In the discrete-time observation equation, $\mathbf{y}, \mathbf{e} \in \mathbb{R}^s$ are the observation and output-error vector. Notice that the output-error represents uncertainty due to both the measurement and the modeling process. From Eq. (21.1) also note that, apart from the uncertainty in future inputs, the prediction uncertainty is also been determined by uncertainty in initial conditions ($\mathbf{x}_0$), model structure ($\mathbf{f}$, $\mathbf{g}$) due to unmodelled phenomena, and model parameter vector $\boldsymbol{\theta}$. Unlike the future input uncertainty, which is not considered in what follows, these uncertainties are quantified on the basis of available measurements which have been corrupted with noise.

The ultimate aim of this contribution is to provide a framework for identification and prediction of grey box models, in the form of a nonlinear state space representation, from data with bounded noise. The evaluation of the prediction uncertainty from different uncertainty sources is herein emphasized.

Conventionally, the evaluation of the prediction uncertainty is performed within a stochastic framework, that is, by employing random differential equations, first-order variance propagation analysis, or Monte Carlo simulation analysis. In the 1980s, however, a set-membership approach to prediction(1–7) has been developed as well. Within this approach, the only assumption with respect to the uncertainty is that it is pointwise bounded with known bounds, which implies that for each t_k the output-error in (21.1) belongs to a set. In mathematical notation: $\mathbf{e}(t_k) \in \Omega_e(t_k)$, where

$$\Omega_e(t_k) = \{\mathbf{e}(t_k) \in \mathbb{R}^s\colon \mathbf{e}(t_k)^- \le \mathbf{e}(t_k) \le \mathbf{e}(t_k)^+\} \quad \text{for } t_k = t_1, \ldots, t_N \tag{21.2}$$

and $\mathbf{e}(t_k)^-$, $\mathbf{e}(t_k)^+$ are the lower and upper bound, respectively. Hence, the parameter estimates and the instantaneous predictions also belong to a set. This approach is very much appealing when no detailed statistical model of the uncertainty can be found as, for instance, in situations with sparse data.

For a reliable assessment of the prediction uncertainty one needs to have a valid description of the uncertainties at the beginning of the prediction stage. Therefore, in Section 21.2 the identification of parametric and (nonparametric) modeling uncertainty is evaluated in detail for the class of state-space models represented by Eq. (21.1), and uncertainty model Eq. (21.2). Within this set-theoretic framework, robust estimates of parameters and modeling uncertainty result. In Section 21.3 two examples are presented which will illustrate the set-membership approach to prediction. First, a simple hypothetical example is presented, which shows the effect of both uncertainty components on the prediction uncertainty. Secondly, a simple "real world" example of modeling dissolved oxygen (DO) concentrations in a lake is used for the validation of the approach to one-step and multiple-steps ahead predictions. In addition, long-term predictions are evaluated and compared with available measurements. Finally, in Section 21.4, some concluding remarks are presented.

21.2. ROBUST IDENTIFICATION AND PREDICTION

21.2.1. Parameter and Modeling Error Estimation

Within the set-membership context, the problem is to identify a set of feasible parameter vectors (denoted by Ω_θ) consistent with the model Eq. (21.1), the error characterization Eq. (21.2), and a predefined parameter set. From this formulation, it is clear that the feasible parameter vectors as well as the predicted outputs are robust with respect to all disturbances which satisfy Eq. (21.2). Apart from the literature cited above with respect to prediction, there is a growing amount of literature on the set-membership approach to identification; see Refs. 8–10 for an overview. However, most algorithms are merely applicable to models that are linear in the parameters. For models nonlinear in the parameter, in addition to successive linearization,(11) OMNE developed by Lahanier, Walter, and Gomeni,(12) and the Monte Carlo Set-Membership (MCSM) algorithm,(13,14) identification methods based on boundary search,(15) signomial programming(16) and interval analysis(17) have also been developed recently.

One of these, the widely applicable MCSM algorithm, is characterized by global random scanning in a predefined parameter space, which is updated occasionally by a parameter space rotation procedure based on principal component analysis of the feasible realizations. Notice that this algorithm is clearly based on a discrete (numerical) approximation of the nonlinear identification problem. Hence, for the system represented by Eqs. (21.1 and 21.2), the algorithm only identifies Ω_θ exactly for an infinite number of realizations from a predefined parameter set that contains the exact solution set.

From Eq. (21.1) notice that the uncertainty in model parameters, initial conditions, and model structure is strongly related to the behavior of the output-error vector sequences and the prior characterization Eq. (21.2). As for the initial conditions, it is common practice to augment the parameter vector with the unknown initial conditions. If the model is exact, the parametric uncertainty due to measurement noise only is explicitly represented by the set of feasible parameter vectors. Otherwise, Ω_θ will also represent some or all uncertainty due to model misspecifications. In what follows the uncompensated part of this is indicated as modeling error. For the characterization of the *j*th element of the modeling error vector, $w_j \in \Omega_w(j)$, the following expression in terms of the output-error has been stated,(6)

$$\Omega_w(j) = \{w_j \in \mathbb{R}: |w_j| \leq w_j^M\} \tag{21.3}$$

$$w_j^M = \max_{t_k} \{ \min_{i=1,\ldots,M} \Omega_\varepsilon(j,t_k)\} \tag{21.4}$$

where $\Omega_\varepsilon(j,t_k)$ is a (finite) set of absolute residual output errors at time instant t_k and related to the *j*th observation element. Excluding residuals related to outliers, $\Omega_\varepsilon(j,t_k)$ is defined as

$$\Omega_\varepsilon(j,t_k) = \{\varepsilon_{ijk} : \varepsilon_{ijk}$$
$$= |y_j(t_k) - \mathbf{g}_j[\mathbf{x}(t_k,\boldsymbol{\theta}_i), \mathbf{u}(t_k), t_k;\boldsymbol{\theta}]|;\ \forall\ \boldsymbol{\theta}_i \in \Omega_\theta,\ i = 1, \ldots, M\}$$

where *M* is an appropriate number, which depends on the description of the feasible parameter set. In the case of an exact polytopic solution, *M* is equal to the number of vertices, while for a discrete approximation *M* is identical to card(Ω_θ). Notice that for a discrete solution obtained from MCSM, the modeling error also represents the uncertainty introduced by an inner-bounding solution of the identification problem. As an alternative to the upper bound description of the modeling error, suitable for robust long range predictive controller design or guaranteed scenario analysis, for short-term prediction the conservatism in the estimate may be reduced by taking into account the time structure of $\{w_j^M(t_k)\}$.[18]

21.2.2. Exact and Approximate Modeling

After having presented the key ideas behind the identification of both the parametric and the (nonparametric) modeling uncertainty for a general class of state-space models with output error, consider the following cases:

A. Consider an *exact* model and *exact* measurements. This situation occurs when one formulates, for example, a (*N*-1)th order polynomial model on the basis of a finite number (*N*) of accurate measurements. The set Ω_θ reduces then to a singleton which implies a single predicted output trajectory.

B. Consider an *exact* model and *noisy* measurements. Assume, furthermore, that the noise is governed by a random mechanism which has been characterized exactly in terms of upper bounds. For the linear case and *N*→∞ the set Ω_θ converges with probability one to a singleton,[19] denoted as the minimax or Chebyshev estimate, or in terms of Tempo *et al.*,[5] the maximally robust estimate. For the nonlinear case, a nonlinear optimization problem remains which for *N*→∞ does not necessarily result in a single maximally robust estimate. For output prediction one of these "optimal" estimates is selected, which results in a single trajectory. On the contrary, when *N* is finite, more than one feasible parameter vector is most likely to be found. Hence, Ω_θ is associated with a (finite) set of feasible model response trajectories,

$$\Omega_{\hat{y}}(t_k) = \{\hat{\mathbf{y}} \in \mathbb{R}^s: \hat{\mathbf{y}}(t_k;\boldsymbol{\theta}) = \mathbf{g}[\mathbf{x}(t_k,\boldsymbol{\theta}), \mathbf{u}(t_k), t_k;\ \boldsymbol{\theta}];\ \forall\ \boldsymbol{\theta} \in \Omega_\theta\} \tag{21.5}$$

for $t_k = 1, \ldots, t_{N+P}$, where P is the prediction horizon.

C. Consider an *approximate* model and *exact* measurements. A first step towards the solution of this problem is to hypothesize that the model is exact, which implies that is assumed that the measurements are corrupted with (colored) noise (Case B). From these conditions note that for $N \rightarrow \infty$ and an exact unknown-but-bounded error characterization it suffices to solve a minimax estimation problem in order to obtain the feasible parameter set. However, under the conditions originally considered, the resulting minimax estimate alone does not represent the uncertainty caused by the approximate model. In the second step, estimate an upper bound on this modeling uncertainty from Eq. (21.4) which is exact for $N \rightarrow \infty$. Hence, for output prediction based on the model structure given in Eq. (21.1), the interval vector $[-w^M, +w^M]$, containing the estimated upper bounds on the modeling uncertainty, must be added to the minimax output vectors at the time instants $t_{N+1}, \ldots, t_{N+P}$. When N is finite or when the output-error bound is chosen too large, the estimated bounds must be added to a set of output vectors at each time instant instead of a single minimax output vector.

D. Consider an *approximate* model and *noisy* measurements. Notice that for $N \rightarrow \infty$ the set Ω_θ is empty for an exact characterization of the noise originating from the measurement process alone, because of the presence of modeling uncertainty in addition to the measurement error. From the viewpoint of model selection, this implies then that an empty parameter set indicates the presence of modeling uncertainty. Thus, the model structure is an incorrect or incomplete representation of the system under study. In practice, however, measurement error bounds can seldom be specified exactly. In this situation, the specified error set Eq. (21.2), if chosen sufficiently large, represents both measurement and modeling uncertainty, which will also be reflected in Ω_θ. In a previous paper[6] various situations with respect to Ω_θ and the choice of bounds on $\{\mathbf{e}(t_k)\}$ have been evaluated. In those cases where Ω_θ does not represent the modeling uncertainty completely (see Case C), an instantaneous estimate of the upper bound on this uncertainty is provided by Eq. (21.4). Hence, for realistic output predictions, based on a state-space model formulation with output-error, the vector sum of the modeling error set $\Omega_w(t_k)$ and the set $\Omega_{\hat{y}}(t_k)$ (Eq. (21.5)) must be determined for $t_k = t_{N+1}, \ldots, t_{N+P}$.

Thus, from a system-theoretic point of view, robust estimates of both the model parameters and the modeling uncertainty are provided, which contribute to robust output predictions.

21.3. EXAMPLES

In this section two examples illustrate the application of the procedures previously presented. A more complex "real world" example of predicting algal growth in a water quality system under environmental change has been reported.[7,20]

21.3.1. Hypothetical Example

Consider the following measurements,

$x(t_k)$	1	2	3	4	5
$y(t_k)$	5.2	5.3	5.1	4.5	5.0

which originate from a random process with $y(t_k) \in U[4.5,5.5]$.

If both model and measurements are assumed to be exact, it can easily be verified that the model,

$$\hat{y}(t_k;\theta) = \theta_0 + \theta_1 x(t_k) + \theta_2 x^2(t_k) + \theta_3 x^3(t_k) + \theta_4 x^4(t_k) \tag{21.6}$$

satisfies the assumed conditions for $\theta_0 = 6.4997$, $\theta_1 = -2.9661$, $\theta_2 = 2.2830$, $\theta_3 = -0.68325$, and $\theta_4 = 0.06666$. The output prediction, consisting of one single trajectory, is presented in Fig. 21.1, Case A.

Alternatively, consider the case in which it is assumed that the noisy measurements are obtained from a process that is exactly represented by the model,

$$\hat{y}(t_k;\theta) = \theta_0 \tag{21.7}$$

and measurement uncertainty that is bounded on the interval $[-0.5, 0.5]$ for all t_k. Then, the bounds on the model output can easily be calculated from the measured minimum and maximum value plus or minus the noise bound, that is

$$\Omega_{\hat{y}}(t_k) = \{\hat{y}(t_k) \in \mathbb{R}: 4.8 \le \hat{y}(t_k;\theta) \le 5.0\} \quad \text{for } t_k = t_1, \ldots, t_{N+P}$$

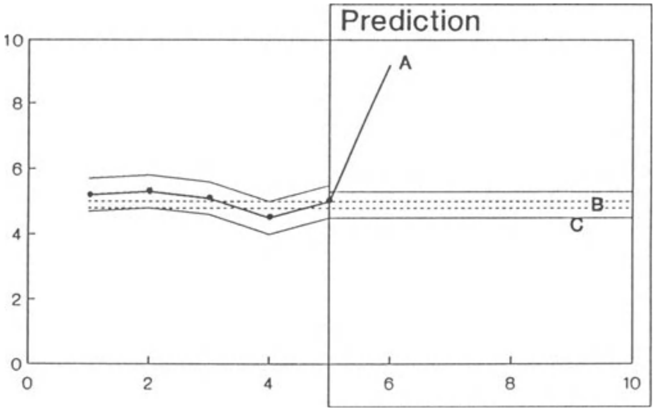


FIGURE 21.1. Prediction uncertainty evaluation for the hypothetical example.

(see Fig. 21.1, Case B, where $P = 5$).

If, on the other hand, the measurements are assumed to be exact and the model is an approximate representation of the process, a modeling error must be added to the feasible model output set resulting from analysis with an exact model and noisy measurements. The upper bound on the modeling uncertainty is equal to 0.3, that is $\max\{y(t_k)\}$ minus the upper bound on $\Omega_{\hat{y}}$, or $|\min\{y(t_k)\} - 4.8|$. The set of corresponding output predictions is presented in Fig. 21.1, Case C.

At last, consider the most realistic case in which both the model and measurements are uncertain. Let $|e(t_k)| \leq \eta$; then for $0.4 \leq \eta \leq 0.8$, the ultimate set of output predictions is equal to the previous one. Notice that the contribution of the additive modeling error diminishes from 0.4 to zero. Hence, for $\eta > 0.8$, $w^M = 0$ and $\Omega_{\hat{y}}(t_k)$ for $t_k = t_1, \ldots, t_{N+P}$ contains the set of Case C. Clearly, by selecting the upper error bound η one can weigh the trade-off between a parametric and a nonparametric uncertainty description.(21)

21.3.2. "Real World" Example

For this example, measurements of dissolved oxygen concentrations in a lake are used and have been presented in previous work.(6) The dynamic behavior of the dissolved oxygen concentration (C) can be described by,

$$dC(t)/dt = K_r[C_s(t) - C(t)] + \alpha I(t) - R \tag{21.8a}$$

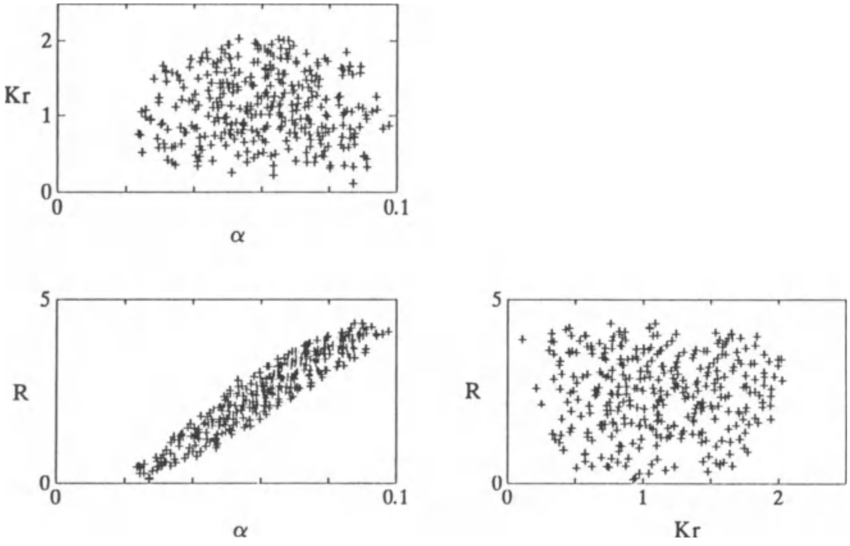


FIGURE 21.2. Parameter estimation results.

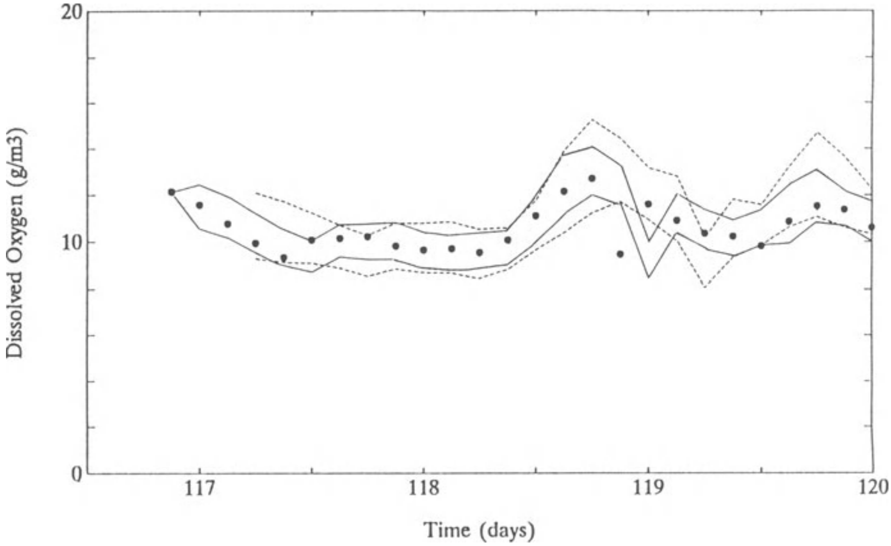


FIGURE 21.3. Prediction uncertainty evaluation for the "real world" dissolved oxygen example.

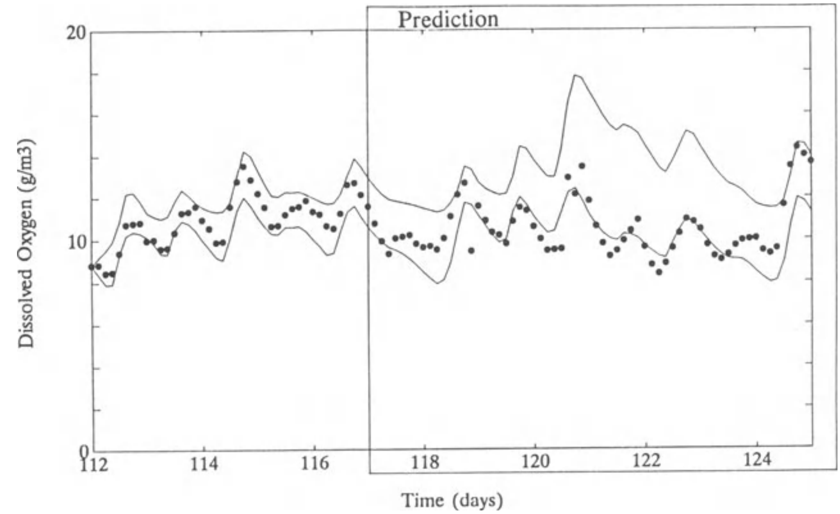


FIGURE 21.4. Long-term predictions for the "real world" dissolved oxygen example.

$$y(t_k) = C(t_k) + e(t_k) \qquad (21.8b)$$

where $C_s(t)$ is the saturated DO concentration and $I(t)$ the radiation. The parameters K_r, α, and R represent the reaeration coefficient, the photosynthetic production rate, and the oxygen consumption rate, respectively. The posterior parameter set Ω_θ has been identified from 40 measurements (five days with sampling interval of three hours) using the MCSM algorithm with $\eta = 1.5$ g/m^3 (see Fig. 21.2). The maximum

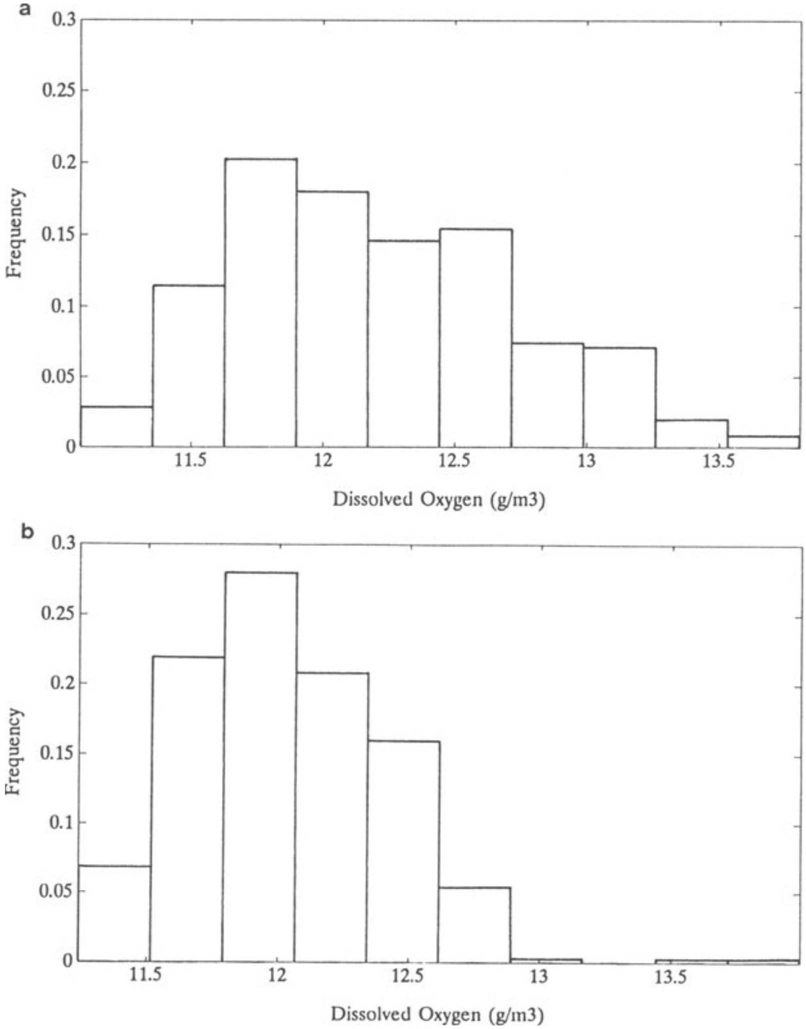


FIGURE 21.5. Prediction frequency distributions at $k = 120$ and $k = 125$.

distance between individual measurements and associated model output set generated by Ω_θ over all sampling instants t_k offers, then, a bound on the modeling uncertainty, that is, $w^M = 0.17$ g/m^3.

For $t_k \in \{117, 117.125, 117.25, \ldots, 120\}$, the one-step and three-steps-ahead predictions in terms of lower and upper bounds are presented in Fig. 21.3. Notice the effect of the outlier at time instant $t_k = 118.875$ on the predictions.

The long-term bounded predictions, as a result of this one-step procedure of identification and prediction, can be seen in Fig. 21.4. Clearly, not all observations are contained in the predicted model output set. Most likely this has been caused mainly by unmodeled dynamics due to incomplete knowledge of processes related to oxygen production by radiation and oxygen consumption.

In addition to the prediction uncertainty bounds, frequency distributions are also available as a result of the sampled parameter space. From the positively skewed frequency distributions for $k = 120$ and 125 it can be concluded that the high prediction values are especially determined by only a few parameter combinations. However, recall that this additional information is not essential for the procedure; the primary interest is the prediction uncertainty bounds to be employed in robust predictive controller design or guaranteed scenario analysis.

21.4. CONCLUSIONS

Robust identification of both the model parameters and the modeling uncertainty within the context of a state-space model formulation on behalf of a realistic evaluation of the prediction uncertainty have been the main themes of this chapter. Within the set-membership approach, the MCSM algorithm, which is applicable to a broad class of (nonlinear) estimation problems, provides a finite set of robust parameter estimates Ω_θ. The modeling error set Ω_w, including errors due to the inner-bounding characteristics of MCSM, is obtained from analysis of the residual output error set. Robust output predictions for the class of state-space models with output-error result, then, from the vector sum of Ω_w and the set of model responses $\Omega_{\hat{y}}(t_k)$ for $t_k = t_{N+1}, \ldots, t_{N+P}$, which is determined by Ω_θ, the posterior parameter set.

REFERENCES

1. B. R. Barmish and J. Sankaran, *IEEE Trans. Autom. Control* **24**, 346 (1979).
2. K. Fedra, G. van Straten, and M.B. Beck, *Ecol. Model.* **13**, 87 (1981).
3. M. Milanese and R. Tempo, *IEEE Trans. Autom. Control* **30**, 730 (1985).
4. A. Vicino, R. Tempo, R. Genesio, and M. Milanese, *J. Forecast.* **3**, 313 (1987).
5. R. Tempo, B. R. Barmish and J. Trujillo, in: *Proceedings of the 8th IFAC Symposium on Identification and System Parameter Estimation*, Beijing, P.R. China, pp. 821–825 (1988).
6. K. J. Keesman and G. van Straten, *Int. J. Control* **49**, 2259 (1989).

7. K. J. Keesman and G. van Straten, *Water Resour. Res.* **26**, 2643 (1990).
8. J. P. Norton, *Automatica* **23**, 497 (1987).
9. E. Walter and H. Piet-Lahanier, *Math. Comput. Simul.* **32**, 449 (1990).
10. M. Milanese and A. Vicino, *Automatica* **27**, 997 (1991).
11. G. Belforte and M. Milanese, in: *Proceedings of the 1st IASTED Symposium on Modeling, Identification and Control*, Davis, Switzerland, pp. 75–79 (1981).
12. H. Lahanier, E. Walter, and R. Gomeni, *J. Pharmacokin. Biopharm.* **15**, 203 (1987).
13. K. J. Keesman and G. van Straten, in: *Proceedings of the IAWPRC Symposium on Systems Analysis in Water Quality Management*, pp. 297–308, Pergamon Press, Oxford (1987).
14. K. J. Keesman, *Math. Comput. Simul.* **32**, 535 (1990).
15. J. P. Norton and S. M. Veres, in: *Proceedings of the 9th IFAC Symposium on Identification and System Parameter Estimation*, Budapest, Hungary, pp. 363–368 (1991).
16. M. Milanese and A. Vicino, *Automatica* **27**, 403 (1991).
17. L. Jaulin and E. Waller, *Automatica* **29**, 1053 (1993).
18. K. J. Keesman, *Envirometrics* **6**, 445 (1995).
19. A. B. Kurzhanski, Identification—A Theory of Guaranteed Estimates, Working Paper, WP-88-55, IIASA, Laxenburg, Austria, p. 78 (1988).
20. G. van Straten and K. J. Keesman, *J. Forecasting* **10**, 163 (1991).
21. K. J. Keesman, in: Upper Error Bound Selection in Bounded Parameter Estimation, *Proceedings of the 10th IFAC Symposium on System Identification*, Copenhagen, Denmark (1994).

22

Estimation Theory for Nonlinear Models and Set Membership Uncertainty

M. Milanese and A. Vicino

ABSTRACT

This chapter studies the problem of estimating a given function of a vector of unknowns, called the problem element, by using measurements depending nonlinearly on the problem element and affected by unknown but bounded noise. Assuming that both the solution sought and the measurements depend polynomially on the unknown problem element, a method is given to compute the axis-aligned box of minimal volume containing the feasible solution set, i.e., the set of all unknowns consistent with the actual measurements and the given bound on the noise. The center of this box is a point estimate of the solution, which enjoys useful optimality properties. The sides of the box represent the intervals of possible variation of the estimates. Important problems, like parameter estimation of exponential models, time series prediction with ARMA models and parameter estimates of discrete time state space models, can be formalized and solved by using the developed theory.

M. Milanese • Dipartimento di Automatica e Informatica, Politecnico di Torino, 10129 Torino, Italy.
A. Vicino • Facoltà di Ingegneria, Università degli Studi di Siena, 53100 Siena, Italy.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

22.1. INTRODUCTION

In this chapter the following problem, referred to as the (generalized) estimation problem,$^{(11)}$ is addressed. Given a problem element λ (for example the vector of parameters of a dynamic system or a time function), evaluate a vector valued function $S(\lambda)$ of this problem element (for example, some functions of parameters of the dynamic system or particular values of the time function). The element λ is not exactly known and there is only partial information on it. In particular, assume that it belongs to a set K of possible problem elements and that further information on λ is given by the knowledge of a function $F(\lambda)$, representing measurements performed on variables depending on λ. Suppose that exact measurements are not available and actual measurements y are corrupted by some error ρ according to the equation

$$y = F(\lambda) + \rho. \tag{22.1}$$

The estimation problem consists in finding an algorithm (estimator) ϕ that provides an approximation $\phi\,(y) \approx S(\lambda)$, as a function of the available data y and in evaluating a measure of the approximation error.

Many different problems such as linear and nonlinear regressions, parameter or state estimation of dynamic systems, state-space and ARMA models prediction, filtering, smoothing, time series forecasting, interpolation, and function approximation can be formulated in a general unifying framework based on the above concepts.

The solution of the estimation problem depends on the type of assumptions made on ρ. Most of the cases investigated in the literature on estimation theory are undoubtedly related to the assumption that the error vector ρ is statistically modeled as an at least partially known probability distribution. Within this context the most important and widely used results are related to the theory of maximum likelihood estimators (MLE). Despite the large amount of theoretical results developed on MLE, the application to real world problems may be not appropriate due to a number of possible drawbacks. These include

1. Actual computation of MLE usually requires a search of the global extremum of functions which are, in general, multimodal. Since general optimization algorithms (including the so called global ones, based on random search) are not guaranteed to achieve the global extremum, the estimate obtained may be far from MLE;
2. Even though MLE are asymptotically efficient, it is difficult to evaluate whether the available data are sufficient to ensure that the covariance matrix estimate is "close" to the Cramer–Rao lower bound or not;
3. For small data sets, it is useful to have lower and upper bounds of the estimate covariance matrix; indeed, tight upper bounds are difficult to

evaluate. Moreover, in this condition even the evaluation of the Cramer–Rao lower bound may be not significant;

4. It is difficult to evaluate the effect of non-exact matching of the assumed statistical hypotheses on ρ. In particular, there is no theory for taking into account the presence in ρ of modeling errors.

In more recent years a new approach, referred to as "set membership error description" or "unknown but bounded error (UBBE)", has been investigated.(1) In this case, the error vector ρ is assumed to be an element of an admissible error set described by a norm operator as

$$\|\rho\| \leq \varepsilon \tag{22.2}$$

where ε is a known quantity. A case of great concern is when l_∞ norms are adopted; in this case, each component of the error vector is known to be bounded by given values. Motivation for this kind of error representation is the fact that in many practical cases the UBBE information is more realistic than statistical assumptions with respect to the measurement error.(2,3) In this context, a possible approach to the estimation problem consists in finding the feasible solution set, i.e., the set of volus $S(\lambda)$ such that λ is consistent with the measurements y and the error model of Eq. (22.2). Any element of this set represents a possible estimate, although the center or the minimum norm element of the set enjoy interesting optimality properties.(4–7) The size of the set represents a measure of the estimate reliability.

Unfortunately, an exact representation of the feasible solution set is in general not simple, since it may be not convex and not connected. It is therefore convenient to look for simpler, although approximate, descriptions of this set. To this extent, the use of simply shaped sets, like axis-aligned boxes (referred to as boxes for short) or ellipsoids, has been proposed to approximate the feasible solution set.(3,8) Ellipsoids may approximate the shape of the feasible solution set better than boxes. Unfortunately, algorithms for computing ellipsoidic approximations are known for linear $S(\cdot)$ and $F(\cdot)$ only.(8,9) Moreover, the obtained approximations may not be tight.(9,10) On the other hand, important information can be obtained by box approximation. In particular, the minimal volume box containing the feasible solution set, minimal outer box (MOB), has the following properties

- the length of each of its sides along the corresponding i-th coordinate axis gives the maximum range of possible variation of $(S(\lambda))_i$ (called Uncertainty Interval UI_i);
- the center of MOB is the (Chebyshev) center of the feasible solution set and hence it is an estimate of $S(\lambda)$ enjoying several optimality properties.(6,7)

For linear problems, the MOB can be computed easily by solving suitable linear programming problems.(3) Unfortunately, many practical estimation problems, even if related to linear dynamic models, lead to nonlinear $S(\cdot)$ and $F(\cdot)$ (see

Section 22.3). Several approaches have been proposed to evaluate MOB when $F(\cdot)$ is nonlinear and $S(\cdot)$ is identity. In Ref. 11 a solution is found in the case in which Eq. (22.1) represents model output-error equations. In Ref. 12 a method of successive linearization is proposed to construct a sequence of boxes contained in the MOB, but no guarantee of convergence to the MOB is given. In Refs. 13 and 14 optimization methods are used to construct the boundary of the feasible solution set. In particular, the random search algorithm used[14] generates a sequence of boxes contained in the MOB and converging monotonically to it with probability one. However, this convergence property is not particularly useful in practice, because no estimate is given of the distance of the achieved solution from the global solution.

This chapter shows that if $S(\cdot)$ and $F(\cdot)$ are polynomial functions, a sequence of boxes contained in the MOB can be constructed, converging to it. Moreover, an estimate of the distance of the estimated box from the MOB is provided at each iteration. It is also shown that the hypothesis of $S(\lambda)$ and $F(\lambda)$ polynomial covers large classes of problems of practical interest such as, for example, the identification of multiexponential, ARMA and state-space discrete time models.

The chapter is organized as follows. Section 22.2 introduces the spaces and operators needed to build a general framework for estimation problems. Section 22.3 shows how some significant estimation problems lead to polynomial $S(\lambda)$ and $F(\lambda)$. Section 22.4 presents an optimization algorithm which allows one to derive a guaranteed global solution for the class of polynomial problems mentioned above. The effectiveness of the proposed approach is demonstrated by some examples reported in Section 22.5.

22.2. A GENERAL FRAMEWORK FOR ESTIMATION PROBLEMS

Let Λ be a linear normed n-dimensional space on the real field (called the *problem element space*). Consider a given operator S, called the solution operator mapping Λ into Z

$$S\colon \Lambda \to Z \tag{22.3}$$

where Z is a linear normed l-dimensional space on the real field. In estimation theory, the aim is to estimate an element $S(\lambda)$ belonging to the *solution space* Z, knowing approximate information about the element λ.

The available information on the problem is contained in the space Λ and in an additional linear space Y which is introduced below. The first kind of information, which is referred to as *a priori* information, is generally provided by letting λ belong to a subset K of Λ. In the chapter problems are considered for which either $K = \Lambda$ (i.e., no *a priori* information is available), or K is given as

$$K = \{\lambda \in \Lambda\colon \|P(\lambda - \lambda_0)\| \leq 1\} \tag{22.4}$$

where P is a linear operator and λ_0 is a known problem element. Despite the above assumption, many of the results presented in the chapter hold also for more general structures of the set K. As for the second kind of information, assume that some function $F(\lambda)$ is given; called *information operator*, mapping Λ into a linear normed m-dimensional space Y (called *measurement space*)

$$F: \rightarrow Y. \tag{22.5}$$

Assume that Z and Y are equipped with (weighted) l_∞ norms.*

In general, due to the presence of noise, exact information $F(\lambda)$ about λ is not available and only perturbed information y is given. In this context, information uncertainty ρ is assumed to be additive, i.e.,

$$y = \mathrm{F}(\lambda) + \rho \tag{22.6}$$

where the error term ρ is unknown but bounded by a given positive value ε according to an l_∞^w norm

$$\|\rho\|_\infty^w \leq \varepsilon. \tag{22.7}$$

Notice that the use of an l_∞^w norm in the measurement space Y allows one to consider different error bounds on every measurement. An *algorithm* ϕ is an operator (in general nonlinear) from Y into Z

$$\phi\colon Y \rightarrow Z \tag{22.8}$$

which provides an approximation $\phi(y) \simeq S(\lambda)$ using the available data y. Such an algorithm is also referred to as an *estimator*.

As a simple example of how a specific estimation problem fits into the general framework outlined above, consider the problem of parameter estimation of a time function belonging to a finite dimensional space, using data obtained by sampling and measuring it at a number of instants. Roughly speaking, the problem element space is the space of the considered class of functions, identified as the space of the unknown function parameters; the space Y is the space of available samples (possibly corrupted by noise); the solution operator is the identity operator and the information operator is the sampling operator.

Now, introduce the following set, which plays a key role in the development of the theory

*A weighted l_∞ norm, denoted by l_∞^w, is defined as

$$\|y\|_\infty^w = \max_i w_i|y_i|, \quad w_i > 0$$

$$T(y) = \{\lambda \in K : ||y - F(\lambda)||_{\infty}^{w} \leq \varepsilon\}. \tag{22.9}$$

The set $T(y)$ contains all λ compatible with the information F, the data y and the bound ε on the noise; $S[T(y)]$ represents the already mentioned *feasible solution set*. Make some technical assumptions about this set. First, there exists a set $Y_0 \subseteq Y$ such that for each $y \in Y_0$, $T(y)$ is nonempty, i.e., the model structure is able to represent all the data y belonging to the set Y_0. Secondly, $T(y)$ does not contain isolated (discrete) points. Third, $T(y)$ is bounded; if this was not true, $F(\lambda)$ would be too poor to solve the problem with finite error, indicating the presence of unidentifiability conditions in the problem formulation. Notice that the above hypotheses are almost always implicitly assumed in the great majority of identification problems.

Algorithm approximation will be measured according to the following *local* and *global* errors:

1. *Y-local* error $E(\phi,y)$

$$E(\phi,y) = \sup_{\lambda \in T(y)} ||S(\lambda) - \phi(y)||. \tag{22.10}$$

2. Λ-*local* error $E(\phi,\lambda)$

$$E(\phi,\lambda) = \sup_{y : ||y - F(\lambda)||_{\gamma}^{w} \leq \varepsilon} ||S(\lambda) - \phi(y)||. \tag{22.11}$$

3. *global* error $E(\phi)$

$$E(\phi) = \sup_{y \in Y_0} E(\phi,y) = \sup_{\lambda \in K} E(\phi,\lambda). \tag{22.12}$$

Algorithms minimizing these types of errors are called Y-*locally*, Λ-*locally* and *globally* optimal, respectively. Notice that the above errors, and related optimality concepts, are relevant to estimation problems. In fact, the Λ-local error measures the maximum uncertainty of the estimates induced by the perturbation affecting the *exact* information $F(\lambda)$, for a given problem element λ . On the other hand, the Y-local error measures the uncertainty affecting an estimate of $S(\lambda)$, for a given set of data y, λ being unknown. The global error represents a *worst case* cost function, in the sense that it measures the largest estimation uncertainty arising for the worst data realization and the worst problem element in the set $T(y)$ of admissible problem elements.

As already mentioned, the set $T(y)$ plays a key role in the present theory. In particular, if $z^c \in Z$ is the Chebyshev center of $S[T(y)]$,* the algorithm ϕ^c, called the *central algorithm*, defined by

*z^c is defined as $\sup_{z \in S(T(y))} ||z^c - z|| = \inf_{\tilde{z} \in Z} \sup_{z \in S(T(y))} ||\tilde{z} - z||$

$$\phi^c(y) = z^c \tag{22.13}$$

is known to be Y-locally and globally optimal.[4,5] In addition, Λ-locally optimality of ϕ_c has been proven under mild assumptions[6,7] for the case where $S(\cdot)$ and $F(\cdot)$ are linear.

Important information can be also derived from the knowledge of the quantities z_i^m and z_i^M, solutions of the following optimization problems

$$z_i^m = \inf_{\lambda \in T(y)} [S(\lambda)]_{i;}\ i = 1, \ldots, l$$

$$z_i^M = \sup_{\lambda \in T(y)} [S(\lambda)]_{i;}\ i = 1, \ldots, l. \tag{22.14}$$

More precisely, observe that

- the intervals

$$UI_i = [z_i^m, z_i^M],\ \ i = 1, \ldots, l. \tag{22.15}$$

represent the range of possible variations of the unknown solution components;
- the MOB containing $S[T(y)]$ is obtained as the cartesian product of the UI_i

$$MOB = [UI_1 \times UI_2 \times \cdots \times UI_l]. \tag{22.16}$$

- the central algorithm ϕ^c can be computed componentwise as[11]

$$[\phi^c(y)]_i = z_i^c = (z_i^m + z_i^M)/2;\ \ i = 1, \ldots, l \tag{22.17}$$

Unfortunately, finding global solutions of problems Eq. (22.14) is in general a difficult task. If no further assumptions on S and F are made, the use of general global optimization algorithms based on random search[16,17] assure at most convergence in probability to global extreme. More importantly, these methods do not provide any measure of how far is the computed solution from the global minimum. However, in many estimation problems $S(\lambda)$ and $F(\lambda)$ are polynomial functions of λ (as shown in next section). In these cases, it is possible to design algorithms (as the one presented in Section 22.4) which ensure certain convergence to global extrema, and give at each step a measure of how far is the actual solution from the global one.

22.3. NONLINEAR ESTIMATION OF DYNAMIC MODELS

As already mentioned, the general framework presented in Section 22.2 can be used to deal with several estimation problems such as dynamic model parameter estimation, prediction, filtering, and so forth. This section shows how to formulate some of them, leading to polynomials S and F.

22.3.1. Parameter Estimation of Exponential Models

Consider the multiexponential model

$$y(t) = \sum_{i=1}^{l} \mu_i e^{-\nu_i t} + e(t) \tag{22.18}$$

where μ_i and ν_i are unknown real parameters and $e(t)$ is unknown but bounded by a given $\varepsilon(t)$

$$|e(t)| \leq \varepsilon(t). \tag{22.19}$$

Suppose that m values $[y(t_i), \ldots, y(t_m)]$ are known and the aim is to estimate parameters μ_i and ν_i, $i = 1, \ldots, l$. Problems of this type arise in many applications, e.g., in pharmacokinetics and biomedical problems. By setting $\xi_i = e^{-\nu_i}$, $i = 1, \ldots, l$, the space Λ is the $2l$-dimensional space of $\lambda = [\mu_1, \ldots, \mu_l, \xi_1, \ldots, \xi_l]^t$ and $Z = \Lambda$, so that S is the identity operator. Y is an m-dimensional space whose elements are given as $[y_1, \ldots, y_m]^t = [y(t_1) \ldots, y(t_m)]^t$.

The information operator $F(\cdot)$ is given by:

$$\begin{bmatrix} F_1(\lambda) \\ \cdots \\ F_m(\lambda) \end{bmatrix} = \begin{bmatrix} \Sigma_{i=1}^{l} \mu_i \xi_i^{t_1} \\ \cdots \\ \Sigma_{i=1}^{l} \mu_i \xi_i^{t_m} \end{bmatrix} \tag{22.20}$$

where it is apparent that each component of $F(\lambda)$ is a polynomial function of μ_i and ξ_i.

22.3.2. Parameter Estimation of ARMA Models

Consider the ARMA model

$$y_k = \sum_{i=1}^{p} \delta_i y_{k-i} + \sum_{i=1}^{q} \theta_i e_{k-i} + e_k \tag{22.21}$$

where e_k is an unknown but bounded sequence

$$|e_k| \leq e_k, \quad \forall\, k. \tag{22.22}$$

To keep notation as simple as possible, consider the case $p = q$. Suppose that m values $[y_1, \ldots, y_m]$ are known and the aim is to estimate parameters δ_i, θ_i. The problem element space Λ can be defined as the $2p + m - 1$-dimensional space with elements

$$\lambda = [\delta_1, \ldots, \delta_p, \theta_1, \ldots, \theta_p, e_1, \ldots, e_{m-1}]^t \tag{22.23}$$

and the subset K is defined by Eq. (22.22), where e_k is replaced with its expression obtained by Eq. (22.21). Space Z is $2p$-dimensional with elements

$$z = [\delta_1, \ldots, \delta_p, \theta_1, \ldots, \theta_p]^t. \tag{22.24}$$

The operator S(λ) is linear and is given by

$$S(\lambda) = [I_{2p}\ \varnothing]\ \lambda \tag{22.25}$$

where I_{2p} is the identity matrix of dimension $(2p,2p)$ and $\varnothing$ is the null matrix of dimension $(2p,\ m{-}1)$. Space Y is an $m{-}p$ dimension with elements $y = [y_{p+1}, \ldots, y_m]^t$. The information operator $F(\cdot)$ is given by:

$$\begin{bmatrix} F_1(\lambda) \\ \cdots \\ F_{m-p}(\lambda) \end{bmatrix} = \begin{bmatrix} \delta_1 y_p + & \cdots & +\delta_p y_1 + \theta_1 e_p + & \cdots & +\theta_p e_1 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \delta_1 y_{m-1} + & \cdots & +\delta_p y_{m-p} + \theta_1 e_{m-1} + & \cdots & +\theta_p e_{m-p} \end{bmatrix}. \tag{22.26}$$

As it can be easily checked, $S(\lambda)$ is linear and $F(\lambda)$ is polynomial (actually linear in δ_i and bilinear in θ_i and e_i).

The same technique can be used to deal with more general models such as ARMAX, bilinear, quadratic, and so forth.

23.3.3. Multistep Prediction with ARMA Models

Consider the ARMA Eq. (22.21) and suppose that the aim is to estimate y_{m+h} when past values $[y_1, \ldots y_m]$ are known (*h-step ahead prediction problem*). This problem can be embedded in the framework of Section 22.2 by defining all spaces and functions as for the case of ARMA parameter estimation, except for Λ and $S(\lambda)$. For the sake of notation simplicity, consider the case $h = 2$. The space Λ is a $(2p{+}m{+}2)$ dimensional space with elements

$$\lambda = [\delta_i, \ldots, \delta_p, \theta_1, \ldots, \theta_p, e_1, \ldots, e_{m+2}]^t \tag{22.27}$$

Z is the one dimensional space with elements, $z = y_{m+2}$. The operator $S(\cdot)$ is no longer linear and is given by

$$S(\lambda) = \sum_{i=1}^{p} (\delta_1\delta_i - \delta_{i+1})y_{m-i+1} + \sum_{i=0}^{p} (\delta_1\theta_i + \theta_{i+1})e_{m-i+1} + \delta_1 e_{m+1} + e_{m+2} \tag{22.28}$$

where

$$\delta_i = 0,\ \ \theta_i = 0 \text{ for } i > p \text{ and } \theta_0 = 1\ .$$

Note that the evaluation of the expression of $S(\lambda)$ requires symbolic computations which for large h may become cumbersome. If necessary, such symbolic computations may be performed by symbolic manipulation codes like MACSYMA, REDUCE, MAPLE, and so forth.

22.3.4. Parameter Estimation of Discrete Time State Space Models

Consider the h-th order linear discrete time dynamic model

$$\begin{cases} x_{k+1} = A(p)x_k + B(p)u_k \\ y_k = C(p)x_k + e_k \end{cases} \tag{22.29}$$

where the system matrices entries are polynomial functions of physical unknown parameters $p \in R^l$, u_k is a known sequence and, e_k is an unknown but bounded sequence. Suppose, for ease of presentation, that the system is single output and that m values of the output $[y_1, \ldots, y_m]$ are known. The aim is to estimate unknown parameters p and system initial condition $x_0 \in R^h$. The problem can be embedded in the framework of Section 22.2 as follows. The space Λ is identified as the $(l + h)$ dimensional space of vectors $\lambda = [px_0]^t$; $K = \Lambda$ (if no *a priori* information is available on physical parameters); $Z = \Lambda$ and S is identity. Y is an m-dimensional space and $F(\lambda)$ is the m-dimensional vector valued function given by

$$\begin{bmatrix} F_1(\lambda) \\ \cdots \\ F_m(\lambda) \end{bmatrix} = \begin{bmatrix} C(p)A(p)x_0 + C(p)B(p)u_0 \\ \cdots \\ C(p)A^m(p)x_0 + \Sigma_{i=0}^{m-1}\ C(p)A^{m-i-1}(p)B(p)u_i \end{bmatrix} . \tag{22.30}$$

Again, the operator $F(\cdot)$ is polynomial in the parameter vector p and linear in the initial condition x_0. Note that symbolic computation of the polynomial expressions of $F_i(\lambda)$ in Eq. (22.30) is required. For large values of m, symbolic evaluation of Eq. (22.30) may become cumbersome due to the fast increase of the number of terms in each component of $F(\lambda)$.

22.4. AN ALGORITHM FOR THE EXACT COMPUTATION OF SOLUTION UNCERTAINTY INTERVALS

If $S(\cdot)$ and $F(\cdot)$ are polynomial functions, Eq. (22.14) is of the form

$$\min\ (\max)\ f_0(\lambda) \tag{22.31}$$

subject to

$$|f_i(\lambda)| \leq \varepsilon_i, \; i = 1, \ldots, m$$

where functions $f_i(\lambda)$ have the structure

$$f_i(\lambda) = \sum_{k=1}^{q} \alpha_{ik} \lambda_1^{a_{ki1}} \lambda_2^{a_{ki2}} \cdots \lambda_n^{a_{kin}}. \tag{22.32}$$

For example, in parameter estimation of exponential models of section 22.3.1 one of the problems to be solved is

$$z_1^m = \mu_1^m = \min \mu_1 \tag{22.33}$$

subject to

$$\left| y(t_j) - \sum_{i=1}^{l} \mu_i \xi_i^{t_j} \right| \leq \varepsilon(t_j), \; j = 1, \ldots, m.$$

The above optimization problem can be transformed into a *signomial* programming problem. Such problems are in general not convex and may exhibit local extreme. An algorithm is presented due to Ref. 20, its original version, which guarantees convergence to the global extremum. The iterative algorithm allows one to evaluate upper and lower bounds on the absolute extremum at each iteration. The sequences of upper and lower bounds converge monotonically to the global solution.

A signomial optimization problem is defined as follows

$$\min \; \{h_0(\lambda) - g_0(\lambda)\} \tag{22.34}$$

subject to

$$\begin{cases} k_k(\lambda) - g_k(\lambda) \leq 1, \; k = 1, \ldots, 2m \\ \lambda_i > 0, \; i = 1, \ldots, n \end{cases}$$

where $h_k(\lambda)$ and $g_k(\lambda)$ ($k = 0, \ldots, 2n$) are *posynomials*, i.e., polynomials with nonnegative coefficients such that

$$\begin{aligned} h_k(\lambda) &= \Sigma_{i \in I_1(k)} \alpha_i \Pi_{j=1}^{n} \lambda_j^{a_{ij}} \\ &\qquad\qquad\qquad , \quad k = 0, \ldots, 2m \\ g_k(\lambda) &= \Sigma_{i \in I_2(k)} \alpha_i \Pi_{j=1}^{n} \lambda_j^{a_{ij}} \end{aligned} \tag{22.35}$$

where exponents a_{ij} are real numbers, α_i are positive reals and $I_{1(2)}(k)$, $k = 0, \ldots, 2m$ are sets of integers that are disjointed for each k. Note that functions $h_k(\lambda)$ and

$g_k(\lambda)$ are, in general, not convex. Nevertheless, one can introduce new variables x_i with the aim of transforming h_k and g_k into convex functions of the variables x_i

$$\lambda_i = e^{x_i}, \quad i = 1, \ldots, n. \tag{22.36}$$

Eqs. (22.35) become

$$H_k(x) = [h_k(\lambda)]_{\lambda_i = e^{x_i}} = \sum_{i \in I_1(k)} \alpha_i \, e^{(a_i, x)}$$

$$G_k(x) = [g_k(\lambda)]_{\lambda_i = e^{x_i}} = \sum_{i \in I_2(k)} \alpha_i e^{(a_i, x)} \tag{22.37}$$

where $(\cdot,\cdot)$ denotes inner product and $a_i = [a_{i1}, \ldots, a_{in}]^t$. Equation (22.34) is transformed into the equivalent problem

$$\min \{H_0(x) - G_0(x)\} \tag{22.38}$$

subject to

$$H_k(x) - G_k(x) \le 1, \quad k = 1, \ldots, 2m.$$

The algorithm given below generates a tree whose node τ are associated with convex problems Q^τ which approximate the signomial Eq. (22.38) (called P). Problems Q^τ are obtained by suitable linear overestimates of $G_k(x)$ as follows.

Suppose that an *a priori* upper and lower bounds x^m and x^M of a global solution x^* of Eq. (22.38) are given

$$x_j^m \le x_j^* \le x_j^M; \quad j = 1, \ldots, n \tag{22.39}$$

and that

$$f^* = H_0(x^*) - G_0(x^*)$$

is the global minimum of Eq. (22.38). Let $\mathcal{S}^\tau$ be the set defined as

$$\mathcal{S}^\tau = \{x: \; r_i^\tau \le (a_i, x) \le R_i^\tau, \; i \in I_2(k)\} \tag{22.40}$$

Variables r_i^τ and R_i^τ are recursively computed using the rules of Steps 5 and 6 below, starting from the initial values

$$r_i^1 = \sum_{j=1}^{n} \min\{a_{ij} x_j^m, a_{ij} x_j^M\}, \quad i \in I_2(k) \tag{22.41}$$

$$R_i^1 = \sum_{j=1}^{n} \max\{a_{ij}x_j^m, a_{ij}x_j^M\}, \; i \in I_2(k) \tag{22.42}$$

Approximating problems Q^τ are of the form

$$\min \; \{H_0(x) - L_0^\tau(x)\} \tag{22.43}$$

subject to

$$\begin{cases} H_k(x) - L_k^\tau(x) \leq 1, \; k = 1, \ldots, 2m \\ x_j^m \leq x_j \leq x_j^M, \qquad j = 1, \ldots, n \end{cases}$$

where

$$L_k^\tau(x) = \sum_{i \in I_2(k)} \left(\frac{\alpha_i}{R_\imath^\tau - r_\imath^\tau}\right) [(R_i^\tau e^{r_\imath^\tau} - r_i^\tau e^{R_\imath^\tau}) + (e^{R_\imath^\tau} - e^{r_\imath^\tau})(a_i, x)] \overset{\Delta}{=} \sum_{\imath \in I_2(k)} \mathcal{L}_i^\tau(x) \tag{22.44}$$

Note that since the terms $L_k^\tau(x)$ are linear and functions $H_k(x)$ are convex, problems Q^τ are convex. Global solutions x^τ for these problems, with minimum $\mathrm{v}^\tau = H_0(x^\tau) - L_0(x^\tau)$, can be found by any convex optimization algorithm. Also notice that $L_k^\tau(x) \geq G_k(x) \;\; \forall x \in S^\tau$, and consequently if $x^\tau \in S^\tau$, then $v^\tau \leq f^*$.

The algorithm generates new approximating problems, by selecting an existing node τ, according to Step 3 of the algorithm below, and refining the linear approximation of the corresponding problem Q^τ according to rules of Steps 5 and 6. Only two problems are generated at each stage, so that after stage s has been completed, problems $Q^1, Q^2, \ldots, Q^{2s+1}$ are generated.

Let $J(s)$ be the set of all nodes τ which have not been selected as branching nodes at stages preceding stage s (see Step 3 of the algorithm below). Define V^s and U^s as

$$V^s = \min_{\tau \in J(s)} \mathrm{v}^\tau \tag{22.45}$$

$$U^s = \min_{\tau=1,\ldots,2s-1} \{H_0(x^\tau) - G_0(x^\tau)\}. \tag{22.46}$$

Note that approximations of functions $G_k(x)$ are performed by constructing linear envelopes, so that the minima of the two approximating problems generated at each stage s are larger than the minimum of the problem which generated them at stage $s-1$. This guarantees that the sequence of lower bounds V^s to the global minimum never decreases. Moreover, the way in which the upper bounds U^s are generated ensures that they form a non-increasing sequence. More importantly, using the results in Ref. 21, Ref. 20 shows that the sequence x^τ contains a subsequence converging monotonically to the global solution x^* and

$$\lim_{s\to\infty} V^s = \lim_{s\to\infty} U^s = f^*. \tag{22.47}$$

The algorithm consists of the following steps:

- Step 1: Initialization.
  Generate and solve Q^1, obtaining x^1, v^1, V^1, U^1. Set $s = 1$, $\tau = 1$, $J(s) = \{1\}$
- Step 2: Check for solution.
  If $V^s = U^s$ then a global solution of problem P is

$$x^* = x^\tau;\ f^* = V^s \tag{22.48}$$

  Otherwise go to Step 3.
- Step 3: Choose a branching node τ.
  Select $\tau \in J(s)$ such that $v^\tau = V^s$
- Step 4: Choose a term of $G_k(x)$ to be approximated.
  Select $k^* \in \{0,1,\ldots,2m\}$ maximizing $L_k^\tau(x^\tau) - H_k(x^\tau)$. Select $i^* \in I_2(k^*)$ maximizing $\mathcal{L}_i^\tau(x^\tau) - \alpha_i e^{(a_i, x^\tau)}$.
- Step 5: Generate problem Q^{2s}.
  Set

$$r_i^{2s} = r_i^\tau;\ R_i^{2s} = R_i^\tau,\ \forall i \in I_2(k^*), i \neq i^*$$

$$R_{i^*}^{2s} = (a_{i^*}, x^\tau),\ r_{i^*}^{2s} = r_{i^*}^c \tag{22.49}$$

- Step 6: Generate problem Q^{2s+1}.
  Set

$$r_i^{2s+1} = r_i^\tau;\ R_i^{2s+1} = R_i^\tau,\ \forall i \in I_2(k^*), i \neq i^*$$

$$r_{i^*}^{2s+1} = (a_{i^*}, x^\tau),\ R_{i^*}^{2s+1} = R_{i^*}^c \tag{22.50}$$

- Step 7: Solve problems Q^{2s} and Q^{2s+1}.
  Solve problems Q^{2s} and Q^{2s+1}, obtaining x^{2s}, x^{2s+1}, v^{2s}, v^{2s+1}. Compute V^{s+1}, U^{s+1} according to Eqs. (22.45 and 22.46). Update the set $J(s)$: add the two nodes $\tau = 2s$, $\tau = 2s + 1$ and delete the node τ selected at Step 3. Set $s = s + 1$ and go to Step 2.

Some considerations on the estimation algorithm proposed above.

REMARK 1. Computation of U^s may be improved by using a local solution in Eq. (22.46) to the true problem P, computed by an iterative algorithm starting from x^τ, instead of using $\{H_0(x^\tau) - G_0(x^\tau)\}$.

REMARK 2. The condition $\lambda_i > 0$ in Eq. (22.34) is not a serious restriction. In fact, it is possible to bring the set $T(y)$ in the first orthant of Λ by means of a suitable translation of the origin of the problem element space. Another way of dealing with this problem is to express unknown sign variables as differences of auxiliary strictly positive variables.

REMARK 3. The convergence speed of the algorithm is, in general, quite sensitive to the sizes of intervals $x_i^M - x_i^m$. In solving the $2l$ optimization Eq. (22.14), information gained by the solved ones can be used to shrink such intervals as much as possible. This is particularly simple for parameter estimation problems where $z_i = [S(\lambda)]_i = \lambda_i$. The following heuristic strategy can be used for handling this problem. A certain number of runs of the $2l$ optimization Eq. (22.14) are performed, stopping the algorithm after few stages $\bar{s}$ (say $\bar{s} = 5$), without waiting for convergence of upper and lower bounds. When solving the first problem of Eq. (22.14), i.e., finding $z_1^m = \mu_1^m$, x_i^m and x_i^M can be derived by *a priori* information provided by the set K. When solving the second problem, i.e., computation in Eq. (22.14), set (recall Eq. (22.36)) $x_1^m = \ln V_1$, where V_1 is the lower bound of z_1^m obtained by the preceding run of the algorithm stopped at stage $\bar{s}$. In solving the third problem (computation of z_2^m), set $x_1^M = \ln U_1$, where U_1 is the upper bound of z_1^M provided by the preceding run of the algorithm stopped at stage $\bar{s}$, and so on. This procedure is iterated until it is able to tighten the bounds x_i^m or x_i^M. Successively, the limitation on the number of stages is removed and each extremum problem is solved by letting the algorithm reach convergence.

Such a shrinking procedure has been used in working out the numerical examples reported in next section. It has proven to be very effective in leading to considerable computing time reductions.

22.5. NUMERICAL EXAMPLES

22.5.1: Example 1: Parameter Estimation of a Multiexponential Model

The following model is considered

$$y(t) = \mu_1 e^{-\nu_1 t} + \mu_2 e^{-\nu_2 t} + e(t). \tag{22.51}$$

The data used are reported in TABLE 22.1. They have been generated from (22.51) with the following nominal parameter values

$$\mu_1 = 20.0, \ \nu_1 = 0.4, \ \mu_2 = -8.0, \ \nu_2 = 0.1. \tag{22.52}$$

The bound on measurement errors is supposed to be:

$$|e(t_k)| \le 0.05|y(t_k)| + 0.1. \tag{22.53}$$

A priori information set K is defined by the following inequalities

TABLE 22.1. Data for Example 1

k	1	2	3	4	5	6	7	8	9	10
t_k	0.75	1.5	2.25	3.0	6.0	9.0	13.0	17.0	21.0	25.0
$y(t_k)$	7.39	4.09	1.74	0.097	–2.57	–2.71	–2.07	–1.44	–0.98	–0.66

$$K: \begin{cases} 2.0 \le \mu_1 \le 60.0 \\ 0.0 < \nu_1 \le 1.0 \\ -30.0 \le \mu_2 \le -1.0 \\ 0.0 < \nu_2 \le 0.5 \end{cases} \tag{22.54}$$

The estimation results obtained are reported in TABLE 22.2. They refer to convergence within 2% of upper and lower bounds of the signomial algorithm for each extremization problem of Eq. (22.14).

The total computing time of the algorithm, using the shrinking procedure outlined in Remark 3 of Section 22.4, is about 10 minutes on a VAX 8800 computer. Convergence within the mentioned tolerance, without using the shrinking procedure, has not been reached after a computing time of about one order of magnitude larger.

22.5.2. Example 2: Multistep Prediction with an AR Model

The following AR model is considered

$$y_k = \delta_1 y_{k-1} + \delta_2 y_{k-2} + e_k \tag{22.55}$$

The data used, which are reported in TABLE 22.3, have been generated from Eq. (22.55) with the nominal parameter values

$$\delta_1 = 0.3, \ \delta_2 = -0.69 \tag{22.56}$$

assuming e_k uniformly distributed and such that

$$|e_k| \le 0.5. \tag{22.57}$$

TABLE 22.2. Uncertainty Intervals and Central Estimates for Example 1

	μ_1	ν_1	μ_2	ν_2
UI	[17.2,26.9]	[0.30,0.49]	[–16.1,–5.4]	[0.077,0.136]
Central estimates	22.05	0.395	–10.75	0.1065

TABLE 22.3. Data for Example 2

k	1	2	3	4	5	6	7	8	9	10	11	12
y_k	0.19	–0.72	–0.82	–0.22	0.88	0.80	–0.20	–0.88	0.31	0.32	–0.33	–0.63

Multistep predictions from 1 to 4 steps ahead have been computed by considering information the set *K a priori* defined by the following inequalities

$$K: \begin{cases} 0.19 \le \delta_1 \le \ \ 0.4 \\ -0.8 \le \delta_2 \le -0.51 \end{cases} \tag{22.58}$$

The uncertainty intervals in Eq. (22.58) have been obtained by a preliminary analysis of maximal and minimal feasible parameters of the linear Eq. (22.55) by means of linear programming.

The results obtained are reported in TABLE 22.4. They refer to convergence within 2% of upper and lower bounds of the signomial algorithm for each extremization problem of Eq. (22.14). The last line of TABLE 22.4 reports the predictions (called nominal predictions) obtained by the minimum mean square predictor of Eq. (22.55) with the nominal parameter values of Eq. (22.56).

The total computing time for obtaining these results is about 3 minutes on a VAX 8800 computer.

22.6. CONCLUSIONS

A method has been proposed for parameter estimation and prediction in a set membership uncertainty context, when measurements are nonlinear functions of the variables to be estimated. A procedure has been presented which allows one to compute exact uncertainty intervals of the estimated variables for the case when

TABLE 22.4. Uncertainty Intervals, Central Predictions and Nominal Predictions for Example 2

	y_{13}	y_{14}	y_{15}	y_{16}
UI	[–0.53,0.56]	[0.33,1.21]	[–0.875,1.19]	[–1.75,0.87]
Central Prediction	0.01	0.44	0.16	–0.44
Nominal Prediction	0.04	0.45	0.11	–0.28

measurements depend polynomially on model parameters. Some examples have been worked out to show the performance of the proposed algorithm.

ACKNOWLEDGMENTS. This work was partially supported by funds of Ministero della Università e della Ricerca Scientifica e Tecnologica and EC Project SIMONET.

REFERENCES

1. M. Milanese, in: *Robustness in Identification and Control* (M. Milanese, R. Tempo, and A. Vicino, eds.) Plenum Press, New York (1989).
2. F. C. Schweppe, *Uncertain Dynamic Systems*, Prentice-Hall, Englewood Cliffs, NJ (1973).
3. M. Milanese and G. Belforte, *IEEE Trans. Autom. Control* **AC-27**, 408 (1982).
4. J. F. Traub and H. Wozniakowski, *A General Theory of Optimal Algorithms*, Academic Press, New York (1980).
5. C. A. Micchelli and T. J. Rivlin, *Optimal Estimation in Approximation Theory* (C. A. Micchelli and T. J. Rivlin, eds.) Plenum, New York, pp. 1–54 (1977).
6. B. Z. Kacewicz, M. Milanese, R. Tempo, and A. Vicino, *Systems and Control Letters* **8**, 161 (1986).
7. M. Milanese, R. Tempo, and A. Vicino, *J. Complexity* **2**, 78 (1986).
8. E. Fogel and F. Huang, *Automatica* **18**, 140 (1982).
9. J. P. Norton, *Automatica* **23**, 497 (1987).
10. G. Belforte and B. Bona, in: *Proc. 7th IFAC Symp. on Identification and System Parameter Estimation*, York, pp. 1507–1511 (1985).
11. M. Milanese and R. Tempo, *IEEE Trans. Automat. Contr.* **AC-30**, 730 (1985).
12. T. Clement and S. Gentil, *Math. and Comput. in Simulation* **30**, 257 (1988).
13. G. Belforte and M. Milanese, in: *Proc. 1st IASTED Symp. Modelling, Identification and Control*, Davos, Switzerland, pp. 75–79 (1981).
14. M. K. Smit, *Measurement* **1**, 181 (1983).
15. E. Walter and H. Piet-Lahanier, in: *Proc. 25th Conf. on Decision and Control*, Athens, Greece (1986).
16. P. M. Pardalos and J. B. Rosen, *Constrained Global Optimization*, Springer-Verlag, Berlin, Germany (1987).
17. P. J. M. van Laarhoven and E. H. Aarts, *Simulated Annealing: Theory and Applications*, Reidel Publishing Company (1987).
18. K. Godfrey, *Compartmental Models and Their Applications*, Academic Press, New York (1983).
19. J. G. Ecker, *SIAM Review* **1**, 339 (1980).
20. J. E. Falk, *Global Solutions of Signomial Programs*, Tech. Rep. T-274, George Washington Univ., Washington, DC (1973).
21. R. M. Soland, *Management Science* **17**, 759 (1971).

23

Guaranteed Nonlinear Set Estimation via Interval Analysis

L. Jaulin and É. Walter

23.1. INTRODUCTION

Many methods have been developed for solving problems arising in mathematics and physics which are formulated in such a way as to require a point solution (e.g., a real number or vector). However, because of the uncertainty attached to the data and numerical errors induced by the finite-word-length representation in the computer, these methods are generally not appropriate to accurately characterize the uncertainty with which the solution is obtained. It is then difficult to assess the validity of the result.

Set formulation of problems replaces the search for a point solution by that of a feasible solution set that may contain a non-denumerable set of vectors. It is then possible to take uncertainty on the data as well as numerical errors into account and to get a global and guaranteed result. Uncertainty on this result can be computed rigorously, contrary to the classical point approaches. Interval analysis is one of the main tools that can be used to characterize sets obtained as the results of computations on sets. It generalizes real and vector calculi to intervals and vector intervals (or boxes). The manipulated subsets are approximated by sets consisting of unions of boxes (or subpavings). In set-inversion problems, which constitute a large part of set problems, the solution set is defined as the reciprocal image of a given set by

L. Jaulin and É. Walter • Laboratoire des Signaux et Systèmes, CNRS École Supérieure d'Electricité, 91192 Gif-sur-Yvette Cedex, France.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

a known function. The simple common structure of these problems makes it possible to derive a single algorithm that can be used to approximate the solution set for any set-inversion problem. This chapter applies this general approach to the problem of bounded-error estimation in the nonlinear case.

Let $M(.)$ be a set of models parameterized by a vector $\mathbf{p} \in \mathbb{R}^{n_p}$. To each value of $\mathbf{p}$ corresponds a model $M(\mathbf{p})$. Let $\mathbf{y} \in \mathbb{R}^{n_y}$ be the vector of all available experimental data, which may consist of measurements performed at various times. The corresponding model output will be denoted by $\mathbf{y}_m(\mathbf{p}) \in \mathbb{R}^{n_y}$. The dependency of $\mathbf{y}$ and $\mathbf{y}_m$ in the experimental conditions (inputs, measurement times, and so forth) is omitted to simplify notation. The output error is defined as

$$\mathbf{e}(\mathbf{p}) = \mathbf{y} - \mathbf{y}_m(\mathbf{p}). \tag{23.1}$$

Bounded-error estimation aims at characterizing the set $\mathbb{S}$ of all values of $\mathbf{p}$ such that $\mathbf{y}_m(\mathbf{p})$ is feasible in the sense that $\mathbf{e}(\mathbf{p})$ belongs to some prior feasible set for the errors $\mathbb{E}$. It is easy to deduce from $\mathbb{S}$ a point estimate $\hat{\mathbf{p}}$ for the parameters, as well as the uncertainty attached to it.

This chapter is organized as follows. Section 23.2 shows how bounded-error estimation can be formulated as a set-inversion problem and gives some illustrative test cases. The notions of interval analysis needed for the algorithm to be proposed are then presented in Section 23.3. Section 23.4 explains how pavings and subpavings can be used to approximate and bracket solution sets. Section 23.5 presents the set-inversion algorithm applied in Section 23.6 to the test cases presented in Section 23.2.

23.2. BOUNDED-ERROR ESTIMATION AS A SET-INVERSION PROBLEM

A MATLAB-like notation is used for vector equations and inequalities. Vectors and vector-valued functions are denoted by bold lower-case letters. Equalities and inequalities are to be understood componentwise. Note that some precautions are required in the manipulation of such operators. For instance, the contraposite of $\mathbf{u} \le \mathbf{v}$ is not $\mathbf{u} > \mathbf{v}$ since the two proposals may be false simultaneously. Usual real functions such as sin, exp, and so forth, when their arguments are vectors, become vector functions and are also written in bold. They are evaluated component by component. For instance

$$\mathbf{sin}(\mathbf{u}) = \mathbf{sin}\begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} = \begin{pmatrix} \sin(u_1) \\ \sin(u_2) \\ \sin(u_3) \end{pmatrix}. \tag{23.2}$$

Let $\mathbf{f}$: $\mathbb{R}^n \to \mathbb{R}^p$ be a continuous function and $\mathbb{Y}$ be a closed subset of $\mathbb{R}^p$. Solving the associated set-inversion problem means characterizing

$$\mathbb{X} = \mathbf{f}^{-1}(\mathbb{Y}) = \{\mathbf{x} \mid \mathbf{f}(\mathbf{x}) \in \mathbb{Y}\}. \tag{23.3}$$

$\mathbb{X}$ is the solution set of the problem. The set function $\mathbf{f}^{-1}$ is the reciprocal function of **f**. The direct image of $\mathbb{X}$ by **f** is defined by

$$\mathbf{f}(\mathbb{X}) = \{\mathbf{f}(\mathbf{x}) \mid \mathbf{x} \in \mathbb{X}\}. \tag{23.4}$$

The set $\mathbf{f}(\mathbb{R}^n)$ is a differential manifold, called the image manifold. When $p > n$, it is almost surely n-dimensional. Otherwise, it is almost surely p-dimensional. From elementary set theory, $\mathbf{f}(\mathbb{X}) \subset \mathbb{Y}$. In many practical problems, $\mathbb{Y}$ can be defined by a finite set of inequalities

$$\mathbb{Y} = \{\mathbf{y} \mid \mathbf{g}(\mathbf{y}) \leq \mathbf{0}\}. \tag{23.5}$$

The following equivalences then hold true

$$\mathbf{x} \in \mathbb{X} \Leftrightarrow \mathbf{f}(\mathbf{x}) \in \mathbb{Y} \Leftrightarrow \mathbf{g} \circ \mathbf{f}(\mathbf{x}) \leq \mathbf{0}. \tag{23.6}$$

If $\mathbf{h} = \mathbf{g} \circ \mathbf{f}$, $\mathbb{X}$ can be described by the finite set of inequalities

$$\mathbb{X} = \{\mathbf{x} \mid \mathbf{h}(\mathbf{x}) \leq \mathbf{0}\}. \tag{23.7}$$

Solving a set-inversion problem thus often amounts to characterizing a set defined by inequalities. When **h** is linear, $\mathbb{X}$ is a polyhedron, and its characteristics, such as its volume, the smallest box or ellipsoid containing it, can be computed accurately. When **h** is nonlinear, the techniques based on interval analysis presented in what follows make it possible to bracket $\mathbb{X}$ between simpler sets consisting of unions of boxes.

In the context of bounded-error estimation, the posterior feasible set for the parameters can be written as

$$\mathbb{S} = \{\mathbf{p} \mid \mathbf{e}(\mathbf{p}) \in \mathbb{E}\} = \mathbf{e}^{-1}(\mathbb{E}). \tag{23.8}$$

Characterizing $\mathbb{S}$ is, therefore, a problem of set inversion. The parameter vector **p**, the error function **e** and the prior feasible set for the errors $\mathbb{E}$, respectively, stand for **x**, **f** and $\mathbb{Y}$. In what follows, assume that $\mathbb{E}$ can be defined by a finite set of inequalities. Two test-cases are now introduced to illustrate the notions presented.

TEST-CASE 1: (Parameter estimation) Consider a two-parameter problem(1) where

$$\mathbf{y} = (0.1, 0.1)^{\mathrm{T}}. \tag{23.9}$$

These data correspond to two scalar measurements performed at times

$$\mathbf{t} = (0.5, 1)^{\mathrm{T}}. \tag{23.10}$$

The corresponding output for a model $M(\mathbf{p})$ is given by

$$\begin{aligned} \mathbf{y}_{\mathrm{m}}(\mathbf{p}) &= (0.5\cos(p_1) + 1.25)\,\mathbf{cos}(p_2\mathbf{t}) \\ &= \begin{pmatrix} (0.5\cos(p_1) + 1.25)\cos(p_2/2) \\ (0.5\cos(p_1) + 1.25)\cos(p_2) \end{pmatrix}, \end{aligned} \tag{23.11}$$

where the ith component of $\mathbf{y}_m(\mathbf{p})$ is computed for the ith component of $\mathbf{t}$. For $\mathbf{p}$ to be feasible, the error must satisfy

$$\mathbf{e}(\mathbf{p}) = \mathbf{y} - \mathbf{y}_m(\mathbf{p}) \in \mathbb{E} = \{\mathbf{e} \mid -\mathbf{0.75} \leq \mathbf{e} \leq \mathbf{0.75}\}, \tag{23.12}$$

where **0.75** is a vector with all entries equal to 0.75. The set to be characterized is given by $\mathbb{S} = \mathbf{e}^{-1}(\mathbb{E})$.

TEST-CASE 2: (State estimation) Consider the discrete-time state space model

$$\begin{cases} x_1(k+1) = \cos(x_1(k)\, x_2(k)) \\ x_2(k+1) = 3x_1(k) - \sin(x_2(k)), \\ y_m(k) = x_1^2(k) - x_2(k) \end{cases} \quad \mathbf{x}(0) = \begin{pmatrix} x_1(0) \\ x_2(0) \end{pmatrix} = \begin{pmatrix} p_1 \\ p_2 \end{pmatrix} = \mathbf{p}. \tag{23.13}$$

In order to estimate the unknown initial conditions, ten measurements $y(k)$ ($k = 0, \ldots, 9$) have been generated by simulating the model with $\mathbf{x}(0) = (2, 1)^{\mathrm{T}}$, which therefore correspond to the true value for the parameters. Adding a random error ε to each of these noise-free outputs, such that $-0.5 \leq \varepsilon \leq 0.5$, the resulting data set is then

$$\begin{aligned} \mathbf{y} &= (y(0), \ldots, y(9))^{\mathrm{T}} \\ &= (3, -5, 0.6, 2.2, -3.8, -1.4, 0.4, -1.2, -1.8, 2.6)^{\mathrm{T}}. \end{aligned} \tag{23.14}$$

The set $\mathbb{S}$ to be characterized is that of all $\mathbf{x}(0) = \mathbf{p}$ such that $\mathbf{e}(\mathbf{p}) \in \mathbb{E}$ with

$$\mathbb{E} = 0.5\,[-\mathbf{1}, \mathbf{1}], \tag{23.15}$$

where $[-\mathbf{1}, \mathbf{1}]$ stands for an axis-aligned hypercube centered on the origin and with width two. The error function $\mathbf{e}$ could be given a formal expression, but the result would be very complex. On the other hand, $\mathbf{e}$ is easily obtained by an algorithm. Using pseudo PASCAL, $\mathbf{e}(\mathbf{p})$ can be computed by

```
$x_1(0) := p_1$; $x_2(0) := p_2$;
For $k$ := 0 to 9 do
  begin
    $y_m(k) := x_1^2(k) - x_2(k)$;
    $e(k) := y(k) - y_m(k)$;
    $x_1(k+1) := \cos(x_1(k) * x_2(k))$;
    $x_2(k+1) := 3\, x_1(k) - \sin(x_2(k))$;
  end;                                                    (23.16)
```

and

$$\mathbf{e}(\mathbf{p}) := \begin{pmatrix} e(0) \\ \cdots \\ e(9) \end{pmatrix}. \tag{23.17}$$

Again, the set to be characterized is $\mathbb{S} = \mathbf{e}^{-1}(\mathbb{E})$.

23.3. INTERVAL ANALYSIS

Interval calculus can be seen as a simple formalism for manipulating inequalities. In the interval approach to numerical computations,(2–4) any uncertain number is replaced by an interval guaranteed to contain it. Intervals are manipulated as a new type of numbers represented by an ordered pair of real numbers associated with the extremities of the interval. Intervals thus have a dual nature of numbers and infinite sets. Many algorithms take advantage of this duality and combine operations on sets, such as union and intersection, with arithmetical operations. High level languages implementing interval calculus are readily available.(5,6)

An interval $[x] \in \mathbb{R}$ or real interval is a closed, connected, and bounded subset of $\mathbb{R}$, such that

$$[x] = [x^-, x^+] = \{x \mid x^- \leq x \leq x^+\}. \tag{23.18}$$

The set of all real intervals will be denoted by $\mathbb{IR}$. Interval arithmetic generalizes addition, subtraction, multiplication, and division to intervals. If, for instance, $x^- \leq x \leq x^+, y^- \leq y \leq y^+$ and $z = x + y$, then $x^- + y^- \leq z \leq x^+ + y^+$ so that the addition of two intervals is defined as

$$[x] + [y] = \{x + y \mid x \in [x] \text{ and } y \in [y]\} = [x^- + y^-, x^+ + y^+], \tag{23.19}$$

Similarly,

$$-[x] = \{-x \mid x \in [x]\} = [-x^+, -x^-], \tag{23.20}$$

$$[x] - [y] = \{x - y \mid x \in [x] \text{ and } y \in [y]\} = [x^- - y^+, x^+ - y^-], \tag{23.21}$$

$$\text{If } 0 \notin [x], \text{ then } 1 / [x] = \{ 1/x \mid x \in [x] \} = [1/x^+, 1/x^-], \tag{23.22}$$

$$[x] * [y] = [\min(x^- y^-, x^- y^+, x^+ y^-, x^+ y^+), \max(x^- y^-, x^- y^+, x^+ y^-, x^+ y^+)], \tag{23.23}$$

$$[x]^2 = \{x^2 \mid x \in [x]\}. \tag{23.24}$$

Note that $[x]^2 \neq [x] * [x]$. For instance, if $[x] = [-1, 1]$, then $[x]^2 = [0, 1]$ whereas $[x] * [x] = [-1, 1]$. It is easy to show that multiplication and addition are both associative and commutative. In general, however, addition is not distributive with respect to multiplication. The subdistributivity property guarantees that

$$[x] * ([y] + [z]) \subset [x] * [y] + [x] * [z]. \tag{23.25}$$

REMARK: When implementing interval arithmetic on a computer, one must take into account that not all intervals can be represented exactly and that approximations are committed at most arithmetical operations. It is necessary, therefore, to perform outwards rounding so as to insure that the exact results are contained in the intervals computed. What follows does not consider these problems of implementation, which have little influence on the results for the problems treated.

The function $\mathbb{f}$: $\mathbb{IR} \rightarrow \mathbb{IR}$ is an inclusion function of the continuous function f: $\mathbb{R} \rightarrow \mathbb{R}$ if it satisfies

$$f([x]) \subset \mathbb{f}([x]) \tag{23.26}$$

for any $[x]$. Computing the interval $f([x])$ would require solving two global optimization problems, which is often exceedingly time consuming. On the other hand, for most real functions f, it is easy to obtain an inclusion function, as will be seen later. If the real x is known to belong to $[x]$, then $f(x)$ is guaranteed to belong to $\mathbb{f}([x])$. For any given f, there are, of course, infinitely many inclusion functions. One of them, denoted $[f]$, is minimal in the inclusion sense and satisfies $[f]([x]) = \mathrm{f}([x])$ for any $[x]$. Call it the minimal inclusion function. For all elementary functions such as sin, cos, exp, log, arcsin, arccos, and so forth, this minimal inclusion function is easy to compute, as illustrated by the two following examples.

EXAMPLE 1: Since the exponential function is increasing, [exp] is given by

$$[\exp]([x]) = [\exp]([x^-, x^+]) = [\exp(x^-), \exp(x^+)] = \exp([x]). \tag{23.27}$$

To get the image interval, it therefore suffices to compute the image by exp of the extremities of $[x]$. This holds true for any real monotonic function.

EXAMPLE 2: Since the sine function is not monotonic, the technique of Example 1 cannot be applied to compute [sin]. It is easy, however, to show that $[\sin]([x])$ can be computed as

$$\begin{array}{ll}
\text{If } \exists\, k \in \mathbb{Z} \mid 2k\pi - \pi/2 \in [x] & \text{then } \sin^-([x]) := -1 \\
 & \text{else } \sin^-([x]) := \min(\sin x^-, \sin x^+); \\
\text{If } \exists\, k \in \mathbb{Z} \mid 2k\pi + \pi/2 \in [x] & \text{then } \sin^+([x]) := 1 \\
 & \text{else } \sin^+([x]) := \max(\sin x^-, \sin x^+); \\
[\sin]([x]) := [\sin^-([x]), \sin^+([x])]. &
\end{array} \tag{23.28}$$

If f results from the composition of real operators or elementary functions, it is not possible to compute $[f]$. An inclusion function called *natural interval extension* can instead be obtained by replacing, in the formal expression for f, its argument x by the interval $[x]$ and the elementary functions and operators by the associated minimal inclusion functions. The natural interval extension is usually far from minimal, and may be much improved by suitably rewriting the formal expression of f or by taking advantage of the fact that the intersection of inclusion functions is an inclusion function.

EXAMPLE 3: Let $f_1(x) = x - x^2$ and $f_2(x) = x(1 - x)$. Although $f_1 \equiv f_2$, the two corresponding natural interval extensions $\mathbb{f}_1([x]) = [x] - [x]^2$ and $\mathbb{f}_2([x]) = [x] * (1 - [x])$ are different. The subdistributivity property of Eq. (23.25) implies $\mathbb{f}_2([x]) \subset$

$\mathbb{f}_1([x])$. For instance, if $[x] = [0, 1]$, $\mathbb{f}_1([x]) = [-1, 1]$ and $\mathbb{f}_2([x]) = [0, 1]$, when $f([x]) = [0, 1/4]$.

Vector calculus can similarly be extended to intervals by replacing vectors of $\mathbb{R}^n$ by boxes. A *box*, or *vector interval*, $[\mathbf{x}]$ of $\mathbb{R}^n$ consists of the Cartesian product of n scalar intervals. Boxes are indifferently denoted by

$$[\mathbf{x}] = [x_1^-, x_1^+] \times \ldots \times [x_n^-, x_n^+] = [x_1] \times \ldots \times [x_n] = [\mathbf{x}^-, \mathbf{x}^+] = \begin{pmatrix} [x_1^-, x_1^+] \\ \ldots \\ [x_n^-, x_n^+] \end{pmatrix}, \tag{23.29}$$

where $\mathbf{x}^- = (x_1^-, x_2^-, \ldots, x_n^-)^{\mathrm{T}}$ and $\mathbf{x}^+ = (x_1^+, x_2^+, \ldots, x_n^+)^{\mathrm{T}}$. The scalar intervals $[x_i] = [x_i^-, x_i^+]$ are the components of the box $[\mathbf{x}]$. The set of all boxes of $\mathbb{R}^n$ is denoted by $\mathbb{IR}^n$. The *width* of $[\mathbf{x}] \in \mathbb{IR}^n$ is given by

$$w([\mathbf{x}]) = \max_i \{x_i^+ - x_i^-\}. \tag{23.30}$$

When $w([\mathbf{x}]) = 0$, $[\mathbf{x}]$ degenerates into the vector $\mathbf{x}$, so that vectors can also be considered as belonging to $\mathbb{IR}^n$, with $\mathbf{x}^- = \mathbf{x}^+ = \mathbf{x}$. A *principal plane* of $[\mathbf{x}]$ is a symmetry plane of this box that is orthogonal to an axis i associated with a side of maximal length, i.e., $i \in \{j \mid w([\mathbf{x}]) = w([x_j])\}$.

Fig. 23.1 presents a two-dimensional box with its principal plane, a straight line here. The *enveloping box* $[\mathbb{A}]$ of a bounded set $\mathbb{A} \subset \mathbb{R}^n$ is the smallest box (in the sense of inclusion) of $\mathbb{IR}^n$ that contains $\mathbb{A}$.

$$[\mathbb{A}] = \cap \{[\mathbf{x}] \in \mathbb{IR}^n \mid \mathbb{A} \subset [\mathbf{x}]\}. \tag{23.31}$$

Vector addition and external multiplication can be extended to boxes:

$$[\mathbf{x}] + [\mathbf{y}] = \{\mathbf{x} + \mathbf{y} \mid \mathbf{x} \in [\mathbf{x}], \mathbf{y} \in [\mathbf{y}]\} = [\mathbf{x}^- + \mathbf{y}^-, \mathbf{x}^+ + \mathbf{y}^+], \tag{23.32}$$

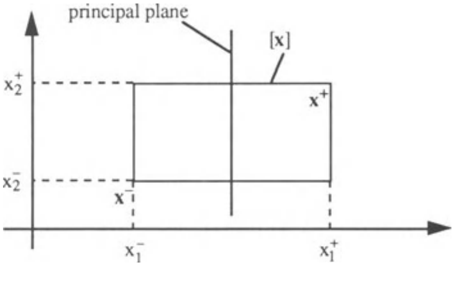


FIGURE 23.1. Box $[\mathbf{x}]$ with its principal plane.

$$\lambda[\mathbf{x}] = \{\lambda\mathbf{x} \mid \mathbf{x} \in [\mathbf{x}]\}. \tag{23.33}$$

The set function $\mathbb{f} : \mathbb{IR}^n \to \mathbb{IR}^p$ is an inclusion function of $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^p$ if and only if for any $[\mathbf{x}]$

$$\mathbf{f}([\mathbf{x}]) \subset \mathbb{f}([\mathbf{x}]). \tag{23.34}$$

It will be said to be *convergent* if for any sequence of boxes $[\mathbf{x}]$

$$w([\mathbf{x}]) \to 0 \Rightarrow w(\mathbb{f}([\mathbf{x}])) \to 0. \tag{23.35}$$

Convergent inclusion functions exist if and only if $\mathbf{f}$ is continuous. Among all possible inclusion functions $\mathbb{f}$ of $\mathbf{f}$: $\mathbb{R}^n \to \mathbb{R}^p$, one,

$$[\mathbf{f}]\text{: } \mathbb{IR}^n \to \mathbb{IR}^p;\ [\mathbf{x}] \to [\{\mathbf{f}(\mathbf{x} \mid \mathbf{x} \in [\mathbf{x}]\}], \tag{23.36}$$

is minimal in the sense of inclusion. Therefore, $[\mathbf{f}]([\mathbf{x}])$ is the enveloping box of the set $\mathbf{f}([\mathbf{x}])$. Fig. 23.2 illustrates the notion of inclusion function.

EXAMPLE 4: Consider the function

$$f\text{: } \mathbb{R}^4 \to \mathbb{R};\ \mathbf{x} \to x_1 \exp(x_2) + x_3 \exp(x_4).$$

From the monotonicity of the exponential function,

$$\exp([x_i]) = [\exp]\ ([x_i]) = [\exp(x_i^-), \exp(x_i^+)].$$

Therefore

$$[\mathrm{f}]\text{: } \mathbb{IR}^4 \to \mathbb{IR};\ [\mathbf{x}] \to [x_1] * \exp([x_2]) + [x_3] * \exp([x_4]).$$

Assume now that $x_1 > 0$ and $x_3 > 0$. Then

$$[f]([\mathbf{x}^-, \mathbf{x}^+]) = [x_1^- \exp(x_2^-) + x_3^- \exp(x_4^-), x_1^+ \exp(x_2^+) + x_3^+ \exp(x_4^+)].$$

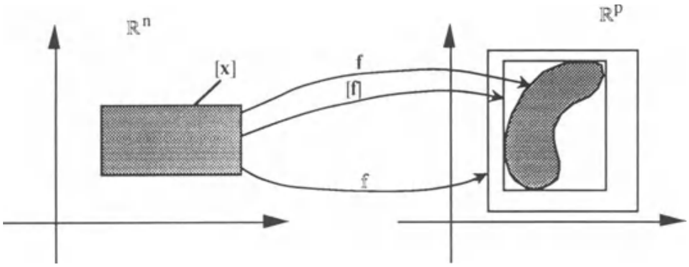


FIGURE 23.2. Minimal inclusion function [f] and inclusion function $\mathbb{f}$.

EXAMPLE 5: Consider the function f: $\mathbb{R}^2 \to \mathbb{R}$; $\mathbf{x} \to x_1 \sin x_2$. Since x_1 and x_2 appear independently in the formal expression of f, it is trivial to show that

$$[f]:\mathbb{I}\,\mathbb{R}^2 \to \mathbb{I}\,\mathbb{R};\ \mathbf{x} \to\ [x_1]*[\sin]([x_2]).$$

When **f** takes its values in $\mathbb{R}^p$, the coordinate functions of **[f]** are the minimal inclusion functions $[f_i]$ associated with the coordinate functions f_i of **f** ($i = 1, \dots,$ p). As in the scalar case, a natural interval extension for **f** can be obtained by replacing in its formal expression (or in the algorithm describing **f**):

— the coordinates x_i of the argument **x** by the components $[x_i]$ of [**x**];
— all arithmetic operators by the corresponding operators for intervals; and
— all elementary functions by the corresponding minimal inclusion functions.

If each component of **x** appears at most once in the formal expression of a given coordinate function, then the natural interval extension is a minimal inclusion function.

TEST-CASE 1 (continued): The natural interval extension $\mathbb{e}([\mathbf{p}])$ for $\mathbf{e}(\mathbf{p})$ is given by

$$\begin{aligned}\mathbb{e}([\mathbf{p}]) &= \mathbf{y} - (0.5\,[\cos]([p_1]) + 1.25)\,[\mathbf{cos}]([p_2]\,\mathbf{t}) \\ &= \begin{pmatrix} 0.1 - (0.5\,[\cos]([p_1]) + 1.25)\,[\cos]([p_2]/2) \\ 0.1 - (0.5\,[\cos]([p_1]) + 1.25)\,[\cos]([p_2]) \end{pmatrix},\end{aligned} \tag{23.37}$$

where $[\cos]([x]) = [\sin](\pi/2 - [x])$ and [sin] is as in Example 2. Note that this inclusion function is minimal, so that $\mathbb{e}([\mathbf{p}]) = [\mathbf{e}]([\mathbf{p}]) = [\mathbf{e}([\mathbf{p}])]$.

TEST-CASE 2 (continued): The natural interval extension $\mathbb{e}([\mathbf{p}])$ for $\mathbf{e}(\mathbf{p})$ can be computed by the following pseudo-PASCAL code, where the inputs are $[p_1]$ and $[p_2]$

```
$[x_1] := [p_1]$; $[x_2] := [p_2]$;
For $k := 0$ to 9 do
  begin
    $[y_m] := [x_1]^2 - [x_2]$;
    $[e](k) := (y(k) - [y_m])^2$;
    $[x_1'] := \cos([x_1]*[x_2])$;                    (23.38)
    $[x_2'] := 3\,[x_1] - \sin([x_2])$;
    $[x_1] := [x_1']$;
    $[x_2] := [x_2']$;
  end;
```

The 10-dimensional box $\mathbb{e}([\mathbf{p}])$ whose kth component is given by $[e](k-1)$ is a convergent inclusion function for the error function $\mathbf{e}(\mathbf{p})$.

23.4. SET BRACKETING AND SUBPAVINGS

The solution set $\mathbb{S}$ to be characterized can usually be defined exactly, e.g., by the nonlinear inequalities of Eq. (23.7). However, the resulting description is often too complex to be of any use. It may, for instance, be difficult to know whether $\mathbb{S}$ is empty, whether it is connected, and what its volume or shape is. Another approach is to approximate $\mathbb{S}$ by more tractable sets, such as unions of boxes called subpavings. It will then become possible to approximate some characteristics of $\mathbb{S}$ by computing the corresponding characteristic of the approximating set.

A *subpaving* $\mathbb{K}$ of $\mathbb{R}^n$ is a set of non-overlapping boxes of $\mathbb{I}\mathbb{R}^n$ with non-zero volume. If $\mathbb{A}$ is the subset of $\mathbb{R}^n$ consisting of the union of all boxes of $\mathbb{K}$, then $\mathbb{K}$ is a *paving* of $\mathbb{A}$. When there is no ambiguity, the set $\{\mathbb{K}\}$ consisting of the union of all boxes of $\mathbb{K}$ will also be denoted by $\mathbb{K}$. Subpavings are easily represented in a computer and readily amenable to set manipulation with the help of interval calculus. They are used to approximate, and more precisely bracket, the sets to be characterized. For almost any $\mathbb{X}$, it is possible to find two finite subpavings $\mathbb{X}^-$ and $\mathbb{X}^+$ such that $\mathbb{X}^- \subset \mathbb{X} \subset \mathbb{X}^+$. The subpavings to be considered here always satisfy $\mathbb{X}^- \subset \mathbb{X}^+$ in the sense that each box of $\mathbb{X}^-$ is also a box of $\mathbb{X}^+$. The quantity $\Delta\mathbb{X} = \mathbb{X}^+ - \mathbb{X}^-$, therefore, is a subpaving, the *uncertainty layer*, which comprises all vectors for which it is not known whether they belong to the interior or exterior of $\mathbb{X}$. Fig. 23.3 illustrates the bracketing of a compact set between subpavings and the associated uncertainty layer. Let $V(\mathbb{X})$ be the set of all compacts $\mathbb{X}'$ such that $\mathbb{X}^- \subset \mathbb{X}' \subset \mathbb{X}^+$. In the Hausdorff distance sense, the diameter of $V(\mathbb{X})$ can be made as small as desired for almost any $\mathbb{X}$.[7] $V(\mathbb{X})$, therefore, is a neighborhood of $\mathbb{X}$.

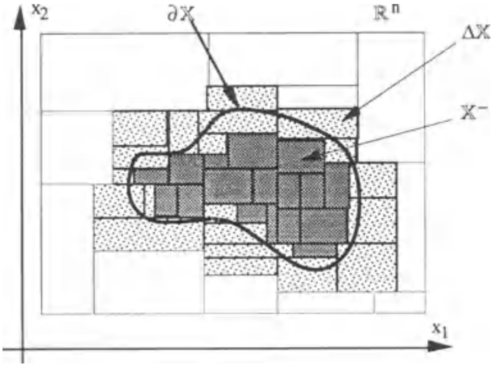


FIGURE 23.3. Bracketing a compact set between two subpavings.

23.5 SET INVERSION

To characterize $\mathbb{X} = f^{-1}(\mathbb{Y})$, assume that a convergent inclusion function $\mathbb{f}$ is known for f and that $\mathbb{Y}$ is compact. The notions of set inversion and bracketing of the solution set by subpavings are illustrated by Fig. 23.4. Note that

$$\mathbf{f}(\mathbb{X}) = \mathbf{f} \circ \mathbf{f}^{-1}(\mathbb{Y}) = \mathbb{Y} \cap \mathbf{f}(\mathbb{R}^n) \subset \mathbb{Y},$$

with $\mathbf{f}(\mathbb{X}) = \mathbb{Y}$ only if $\mathbf{f}$ is surjective, which is never true in the type of applications considered here. The algorithm *Set Inverter Via Interval Analysis* (SIVIA) will now be used to obtain the subpavings $\mathbb{X}^-$ and $\mathbb{X}^+$. It can also be used to bracket any quantity $Z(\mathbb{X})$ monotonic over $\mathbb{X}$ with as much precision as desired.[8] For simplicity, $\mathbb{X}$ is assumed to be bounded and included in a known prior box $[\mathbf{x}](0)$, which is used as the initial search domain. Extension to unbounded sets would involve the use of unbounded boxes (or generalized vector intervals).

A box $[\mathbf{x}]$ of $\mathbb{IR}^n$ is feasible if $[\mathbf{x}] \subset \mathbb{X}$, unfeasible if $[\mathbf{x}] \cap \mathbb{X} = \emptyset$, and ambiguous otherwise. Interval analysis provides two conditions, illustrated by Fig. 23.5, to test a box $[\mathbf{x}]$ for feasibility:

$$\mathbb{f}([\mathbf{x}]) \subset \mathbb{Y} \Rightarrow [\mathbf{x}] \subset \mathbb{X} \text{ ([}\mathbf{x}\text{] is feasible)}, \tag{23.39}$$

$$\mathbb{f}([\mathbf{x}]) \cap \mathbb{Y} = \emptyset \Rightarrow [\mathbf{x}] \cap \mathbb{X} = \emptyset \text{ ([}\mathbf{x}\text{] is unfeasible)}. \tag{23.40}$$

In all other cases, $[\mathbf{x}]$ is indeterminate. Note that indeterminate boxes may be feasible, unfeasible or ambiguous, but that any ambiguous box is indeterminate. Fig. 23.5 shows how an unfeasible box may be indeterminate, which explains why the two previous conditions are only sufficient.

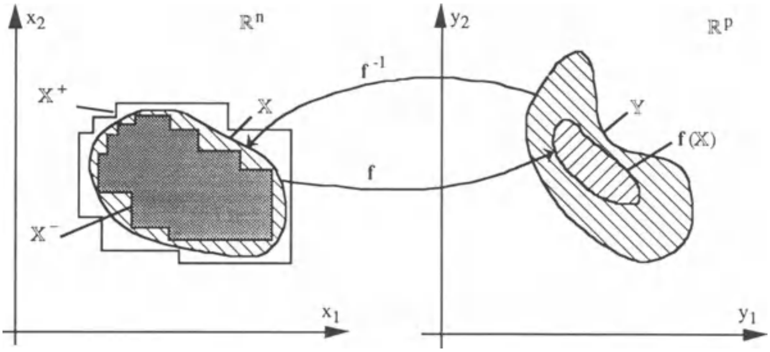


FIGURE 23.4. Bracketing the solution set of the set-inversion problem between two subpavings.

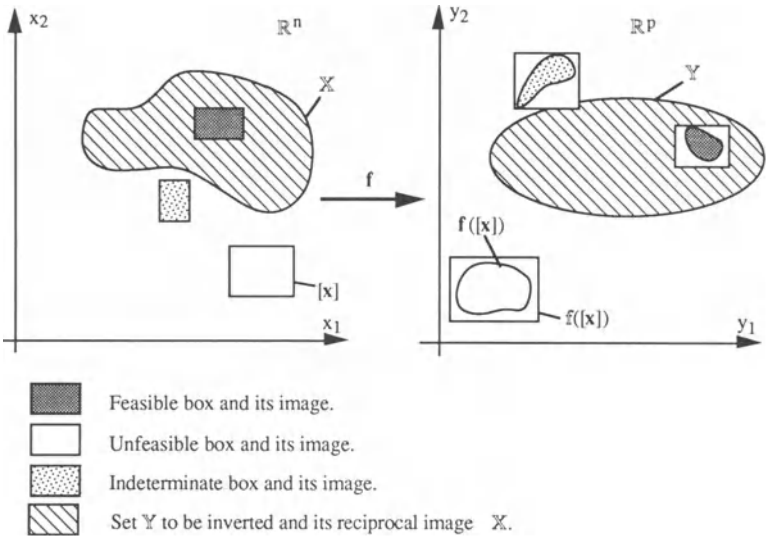

x_2
$\mathbb{R}^n$
$\mathbb{X}$
f
[x]
x_1
y_2
$\mathbb{R}^p$
$\mathbb{Y}$
f([x])
$\mathbb{f}$([x])
y_1
Feasible box and its image.
Unfeasible box and its image.
Indeterminate box and its image.
Set $\mathbb{Y}$ to be inverted and its reciprocal image $\mathbb{X}$.

FIGURE 23.5. Sufficient conditions for a box to be feasible or unfeasible.

SIVIA involves three basic steps:

- the definition of a box of interest [**x**](0), on which the search will be performed;
- the choice of a paving $\mathbb{K}$ for [**x**](0); and
- the computation of $\mathbb{f}$([**x**]) for each box of $\mathbb{K}$.

Three cases are then possible for any given box [**x**]:

- if $\mathbb{f}([\mathbf{x}]) \subset \mathbb{Y}$ then $[\mathbf{x}] \subset \mathbb{X}$, ([**x**] is feasible);
- if $\mathbb{f}([\mathbf{x}]) \cap \mathbb{Y} = \varnothing$ then $[\mathbf{x}] \cap \mathbb{X} = \varnothing$, ([**x**] is unfeasible); and
- else, [**x**] is indeterminate.

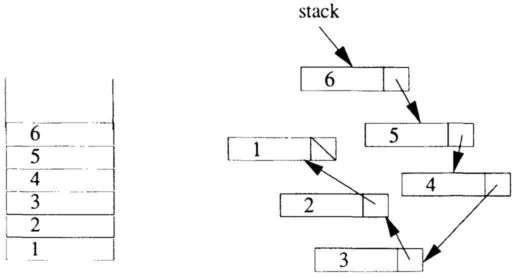
The paving $\mathbb{K}$ is thus partitioned into three subpavings $\mathbb{X}^-$, $\Delta\mathbb{X}$ et $\overline{\mathbb{X}^+}$, which correspond respectively to the sets of all feasible, indeterminate and unfeasible boxes. Since $\mathbb{X}^+ = \mathbb{X}^- \cup \Delta\mathbb{X}$,

$$\mathbb{X}^- \subset \mathbb{X} \subset \mathbb{X}^+, \tag{23.41}$$

$$\partial\mathbb{X} \subset \Delta\mathbb{X}, \tag{23.42}$$

$$\text{vol}\ (\mathbb{X}^-) \leq \text{vol}\ (\mathbb{X}) \leq \text{vol}\ (\mathbb{X}^+), \tag{23.43}$$

$$[\mathbb{X}^-] \subset [\mathbb{X}] \subset [\mathbb{X}^+]. \tag{23.44}$$



(a) Conventional representation. (b) Computer representation.

FIGURE 23.6. Representation of a stack.

SIVIA recursively implements the idea of bracketing by subpavings that has just been presented. A *stack* of boxes (think of a stack of plates) is used, in which each element knows the location of the one located beneath it. Fig. 23.6 illustrates the representation of a stack on a computer. On this figure, a stack of six elements, numbered from 1 to 6, is presented under its traditional form (a) and as stored on the computer (b). Element 1 is at the bottom of the stack and Element 6 on top. The arrows represent the pointers, and the right-hand-side box of each element of the stack stands for the memory cell containing the address of the element beneath. Since it is at the bottom, Element 1 points to no other element. The topology of the stack is independent of its representation on the computer. Such a representation makes it possible to modify relationships without having to move the elements involved, which drastically speeds up the management of the memory.

Only three operations are possible on a stack, namely *stacking* (i.e., putting an element on top of the stack), *unstacking* (i.e., removing the element located on top of the stack) and testing the stack for emptiness. The box considered at iteration k is denoted by $[\mathbf{x}](k)$. The required accuracy for the subpavings $\mathbb{X}^-$ and $\Delta\mathbb{X}$ will be denoted by ε_r. After completion of the algorithm, all indeterminate boxes have a width smaller than or equal to ε_r. The inputs of SIVIA are the inclusion function $\mathbb{f}$, the set to be inverted $\mathbb{Y}$, the domain of interest $[\mathbf{x}](0)$ and the required accuracy ε_r. The initialization is performed by setting

$$[\mathbf{x}] = [\mathbf{x}](0), \text{stack} := \varnothing, \mathbb{X}^- := \varnothing, \Delta\mathbb{X} := \varnothing, \tag{23.45}$$

and iteration is given by

Step 1 If $\mathbb{f}([\mathbf{x}]) \subset \mathbb{Y}$, then $\mathbb{X}^- := \mathbb{X}^- \cup [\mathbf{x}]$. Go to Step 4.

Step 2 If $\mathbb{f}([\mathbf{x}]) \cap \mathbb{Y} = \varnothing$, then go to Step 4.

```
Step 3   If $w([\mathbf{x}]) \leq \varepsilon_r$,   then $\Delta\mathbb{X} := \Delta\mathbb{X} \cup [\mathbf{x}]$,
         else bisect $[\mathbf{x}]$ along a principal plane and stack the two
         resulting boxes.
Step 4   If the stack is not empty, then unstack into $[\mathbf{x}]$ and go to Step 1.
End.                                                                    (23.46)
```

SIVIA thus generates two subpavings $\mathbb{X}^-$ and $\Delta\mathbb{X}$. The dependency of these subpavings on ε_r is omitted to simplify notation. For almost any $\mathbb{X}$, the resulting bracketing ,

$$\mathbb{X}^- \subset \mathbb{X} \subset \mathbb{X}^+ = \mathbb{X}^- \cup \Delta\mathbb{X} \tag{23.47}$$

defines a neighborhood of $\mathbb{X}$ with a diameter that tends to zero with ε_r.. The convergence conditions are studied in Ref. 7. The main limitation of SIVIA lies in its computing time, which is proportional to the number of elements in $\mathbb{K}$ and increases exponentially with the number of parameters.[8]

When one is only interested in computing a characteristic of the solution set $\mathbb{X}$ such as its enveloping box $[\mathbb{X}]$ or its volume vol($\mathbb{X}$), only the stack takes a significant place in memory. It is possible to avoid storing the subpavings $\mathbb{X}^-$ and $\Delta\mathbb{X}$ with the help of a recursive technique. Note, however, that the paving must be explored, even if it needs not be stored, so that the computing time is not shortened. To understand how one can avoid storing subpavings, let us modify SIVIA to recursively bracket the volume of $\mathbb{X}$. The program is initialized by setting stack := $\varnothing$, vol$^-$:= 0, vol$^+$:= 0, $[\mathbf{x}] := [\mathbf{x}](0)$. The iteration is as follows

```
Step 1 If $\mathbb{f}([\mathbf{x}]) \subset \mathbb{Y}$, then  $\text{vol}^- := \text{vol}^- + \text{vol}([\mathbf{x}])$,
                               $\text{vol}^+ := \text{vol}^+ + \text{vol}([\mathbf{x}])$, go to Step 4.
Step 2 If $\mathbb{f}([\mathbf{x}]) \cap \mathbb{Y} = \varnothing$ , then go to Step 4.
Step 3 If $w([\mathbf{x}]) \leq \varepsilon_r$, then $\text{vol}^+ := \text{vol}^+ + \text{vol}([\mathbf{x}])$,
       else bisect $[\mathbf{x}]$ along a principal plane and stack the two resulting boxes.
Step 4 If the stack is not empty, then unstack into $[\mathbf{x}]$ and go to Step 1.
End.                                                                    (23.48)
```

As the volume is a monotonously increasing characteristic, completion gives

$$\text{vol}^- \leq \text{volume}(\mathbb{X}) \leq \text{vol}^+, \tag{23.49}$$

without having stored any subpaving. The transposition to the computation of the enveloping box $[\mathbb{X}]$ is trivial. The number of elements in the stack satisfies:[8]

$$\#\text{stack} < n \text{ int } (\log_2(\text{w}([\mathbf{x}](0))) - \log_2(\varepsilon_r) + 1), \tag{23.50}$$

where int(r) stands for the integer part of a real r. Even for large n, the size of the stack remains reasonable. For instance if $n = 100$, $w([\mathbf{x}](0)) = 10^4$, and $\varepsilon_r = 10^{-10}$, then Eq. (23.50) implies that #stack < 4600.

SIVIA can easily be parallelized, and the following version can be implemented on r processors, which would similarly shorten the computing time. The inputs and initialization are as in the previous version of SIVIA. Iteration is as follows

```
Step 1 Split $[\mathbf{x}]$ into $r$ boxes forming a subpaving $\mathbb{Z}$.
Step 2 Store in $\mathbb{X}^-$ all boxes $[\mathbf{z}]$ of $\mathbb{Z}$ such that $\mathbb{f}([\mathbf{z}]) \subset \mathbb{Y}$.
Step 3 Eliminate from $\mathbb{Z}$ all boxes $[\mathbf{z}]$ such that $\mathbb{f}([\mathbf{z}]) \cap \mathbb{Y} = \varnothing$.
Step 4 Store all boxes $[\mathbf{z}]$ such that $w([\mathbf{z}]) \leq \varepsilon_r$ in $\Delta\mathbb{X}$.
Step 5 Stack all remaining boxes of $\mathbb{Z}$.
Step 6 If the stack is not empty, unstack into $[\mathbf{x}]$ and go to Step 1.
End.                                                            (23.51)
```

Steps 2 to 5 are shared by all processors.

Frequently the parameter space is not isotropic because the sensitivities of **f** relative to the various components of **x** do not have the same order of magnitude. Bisecting along a principal plane, as suggested in the above description of SIVIA, may then turn out to be rather inefficient. The problem is to find a strategy for bisection to speed up the convergence. One way is to weight each component of **x** in such a way as to compensate for the anisotropy. It seems difficult, however, to suggest a rational strategy for the choice of the weights since the anisotropy may strongly depend on the position in the parameter space.

Another algorithm for set inversion was developed independently by Moore.[9] The main difference is that Moore's algorithm uses a queue when SIVIA uses a stack. The required memory for the queue is larger than for the stack by several orders of magnitude.

It may often be helpful to reformulate the problem of set inversion as that of finding any set $\hat{\mathbb{X}}$ such that

$$\mathbf{f}^{-1}(\mathbb{Y}^-) \subset \hat{\mathbb{X}} \subset \mathbf{f}^{-1}(\mathbb{Y}^+),$$

given two sets $\mathbb{Y}^-$ and $\mathbb{Y}^+$ such that $\mathbb{Y}^- \subset \mathbb{Y} \subset \mathbb{Y}^+$. The program is initialized by setting stack $:= \varnothing$, $\hat{\mathbb{X}} := \varnothing$, $[\mathbf{x}] := [\mathbf{x}](0)$. Iteration is as follows

```
Step 1 If $\mathbb{f}([\mathbf{x}]) \subset \mathbb{Y}^+$, then $\hat{\mathbb{X}} := \hat{\mathbb{X}} \cup [\mathbf{x}]$, go to Step 4.
Step 2 If $\mathbb{f}([\mathbf{x}]) \cap \mathbb{Y}^- = \varnothing$, then go to Step 4.
Step 3 Bisect $[\mathbf{x}]$ along a principal plane and stack the two resulting boxes.
Step 4 If the stack is not empty, then unstack into $[\mathbf{x}]$ and go to Step 1.
End.                                                            (23.52)
```

Even if $\mathbb{Y}$ is not known accurately, one then gets a characterization of the uncertainty attached to **x**. Moreover, this algorithm does not require the specification of ε_r. Provided that $\partial\mathbb{Y}^- \cap \partial\mathbb{Y}^+ = \emptyset$, $\hat{\mathbb{X}}$ can be obtained in finite time. Even if $\mathbb{Y}$ is known exactly, such an approach remains of interest in the context of bounded-error estimation. By setting $\mathbb{Y}^- = \mathbb{E}$ and $\mathbb{X} = \mathbb{S}$, one gets $\mathbb{S} \subset \hat{\mathbb{S}}$, so that a set guaranteed to contain the posterior feasible set for the parameters is obtained. The set $\mathbb{Y}^+$ then plays the role of the stopping criterion. If the layer $\mathbb{Y}^+ - \mathbb{Y}^-$ is thick enough, this stopping criterion should make it possible to obtain a result comparable to that of SIVIA much more quickly. The set $\hat{\mathbb{S}}$ contains all the parameter vectors that are consistent with the available information, and none of those that are indisputably inconsistent (in the sense that their image is outside $\mathbb{Y}^+$).

23.6. EXAMPLES

Test-case 1: For a required accuracy $\varepsilon_r = 0.04$ and a prior domain of interest $[\mathbf{p}](0) = [-10, 10] \times [-10, 10]$, SIVIA generates the paving presented on Fig. 23.7 in less than 160 seconds on a Compaq 386/33. It keeps less than 16 boxes in the stack at any given time (Eq. (23.50) predicts a number lower than or equal to 18).

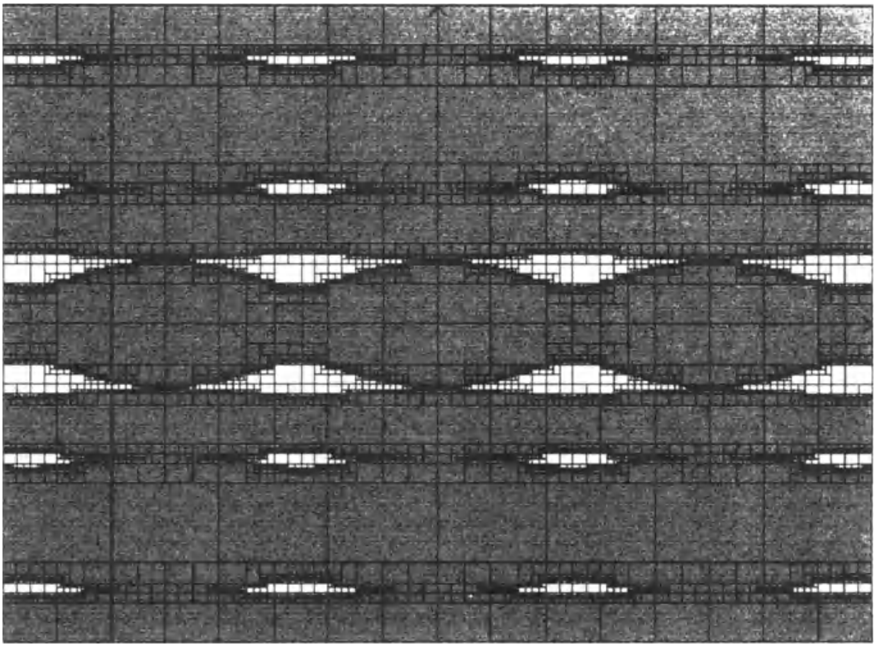


FIGURE 23.7. Paving generated by SIVIA for Test-case 1 in the (p_1, p_2) space. The frame corresponds to the search domain $[\mathbf{p}](0) = [-10, 10] \times [-10, 10]$.

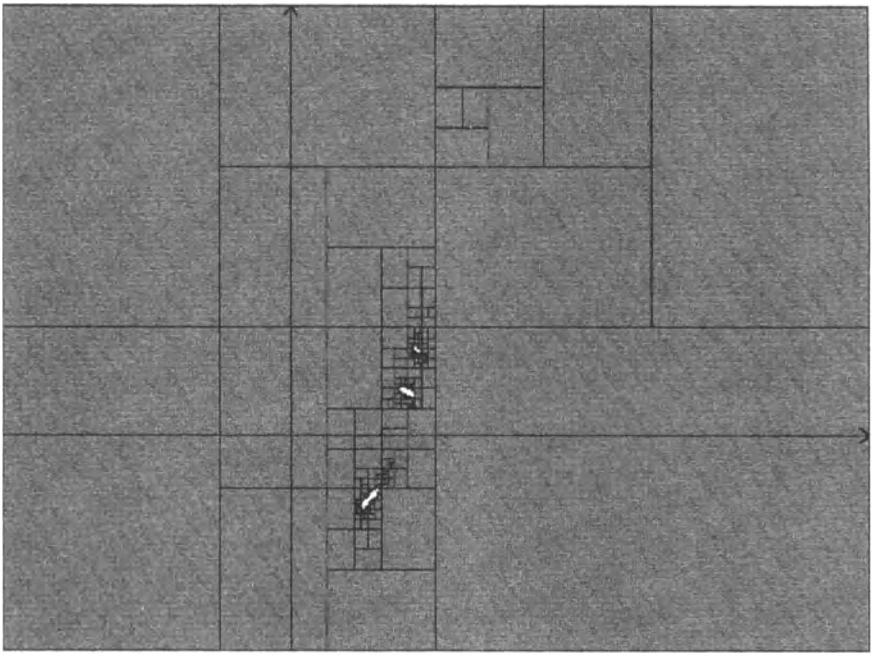


FIGURE 23.8. Paving generated by SIVIA for Test-case 2 in the (p_1, p_2) space. The frame corresponds to the search domain $[\mathbf{p}](0) = [-5, 10] \times [-5, 10]$.

The subpavings $\mathbb{S}^-$ and $\overline{\mathbb{S}^+}$ are filled in with white and grey, respectively. The volume of $\mathbb{S} \cap [\mathbf{p}](0)$ is guaranteed to satisfy

$$35 \leq \mathrm{vol}(\mathbb{S} \cap [\mathbf{p}](0)) \leq 43. \tag{23.53}$$

The posterior feasible set $\mathbb{S}$ for the parameters turns out to be unconnected. One may wonder about the meaning of point estimation in this context.

Test-case 2: For $\varepsilon_r = 0.05$ and $[\mathbf{p}](0) = [-5, 10] \times [-5, 10]$, the paving presented in Fig. 23.8 is generated in less than 10 seconds. The stack never contains more than 14 boxes ((Eq. 23.50) predicts a number lower than or equal to 18). The subpavings $\overline{\mathbb{S}^+}$ and $\Delta\mathbb{S}$ are filled in with grey and white, respectively. No box has been found in $\mathbb{S}^-$.

A random exploration of $[\mathbf{p}](0)$ with a uniform distribution for more than half an hour does not produce any feasible value for $\mathbf{p}$. To understand the difficulty of the problem, zoom around the true value for $\mathbf{p}$. For a required accuracy of $\varepsilon_r = 0.0001$, and $[\mathbf{p}](0) = [1.98, 2.02] \times [0.98, 1.02]$, SIVIA generates the paving presented in Fig. 23.9 in less than ten minutes. The subpavings $\mathbb{S}^-$ and $\overline{\mathbb{S}^+}$ are filled

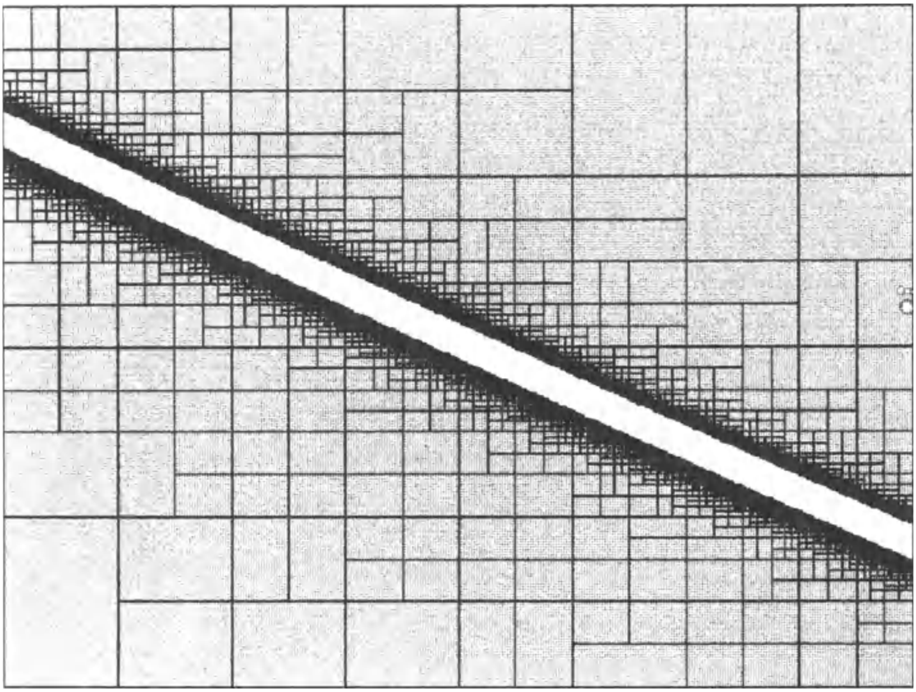


FIGURE 23.9. Paving generated by SIVIA for Test-case 2 in the (p_1, p_2) space. The frame corresponds to the search domain $[\mathbf{p}](0) = [1.98, 2.02] \times [0.98, 1.02]$.

in with white and grey, respectively. The posterior feasible set $\mathbb{S}$ is so narrow that it is almost impossible to reach it by random exploration.

23.7. CONCLUSIONS

Set inversion is particularly suitable to characterize the set of all values of parameters that are feasible in the sense that they satisfy a finite number of (possibly nonlinear) inequalities. The problem of estimating the parameters of a nonlinear model from bounded-error data is easily cast into this framework, which makes it possible to obtain approximate but guaranteed and global results in a finite number of operations.

The tools of interval analysis and the concept of subpaving have been used to derive efficient methods for the solution of the set-inversion problem. To the best of our knowledge, the only approach capable of providing guaranteed global results in nonlinear bounded-error estimation that is not based on interval analysis is the signomial approach advocated by Milanese and Vicino(10) and also presented in this volume. Signomial analysis makes it possible to bracket the enveloping box $[\mathbb{S}]$ of $\mathbb{S}$ between two boxes. It gives $[\mathbb{S}]$ faster than SIVIA, which characterizes $\mathbb{S}$ in a much more detailed way and applies to a larger class of problems (e.g., problems involving trigonometric functions).

SIVIA eliminates a large portion of the domain of interest quickly before concentrating on the boundary of $\mathbb{S}$. It cannot provide great precision when there are more than a few parameters. However, it can still be useful provided that the required accuracy is suitably decreased. It may be, therefore, a powerful tool for a preliminary analysis before turning to local or random searches.

Among the many other problems that can be cast in the framework of set inversion, one may mention the problem of analyzing the robust stability of an uncertain time-invariant linear system.(11)

REFERENCES

1. L. Pronzato and E. Walter, *Math. Comput. Simul.* **32**, 571 (1990).
2. R. E. Moore, *Methods and Applications of Interval Analysis*, SIAM, Philadelphia, PA (1979).
3. A. Neumaier, *Interval Methods for Systems of Equations*, Cambridge University Press, Cambridge, United Kingdom (1990).
4. A. Ratschek and J. Rokne, *New Computer Methods for Global Optimization*, Ellis Horwood Limited, John Wiley & Sons, New York (1988).
5. IBM, *High-Accuracy Arithmetic Subroutine Library, (ACRITH): Program Description and User's Guide*, SC 33-6164-02, 3rd Ed. (1986).
6. R. Klate, U. W. Kulisch, M. Neaga, D. Ratz, and C. Ullrich, *PASCAL-XSC: Language Reference with Examples*, Springer-Verlag, Heidelberg, Germany (1992).
7. L. Jaulin and E. Walter, *Automatica* **29**, 1053 (1993).
8. L. Jaulin and E. Walter, *Math. Comput. Simul.* **35**, 123 (1993).
9. R. E. Moore, *Math. Comput. Simul.* **34**, 113 (1992).
10. M. Milanese and A. Vicino, *Automatica* **27**, 403 (1991).
11. E. Walter and L. Jaulin, *IEEE Trans. Autom. Control* **39**, 886 (1994).

24

Adaptive Control of Systems Subjected to Bounded Disturbances

L. S. Zhiteckij

24.1. INTRODUCTION

In practical adaptive control systems which use identification procedures, the effect of disturbances on the system behavior is the important factor. The above effect is investigated from statistical considerations.(1) This approach requires some knowledge of disturbance statistics. However, in various control applications, the assumptions regarding the disturbance statistics may be invalid. In these cases, the statistical approach is unsuitable. Meanwhile, in most cases the available *a priori* information about the disturbance is given not in statistical terms but as bounds on its absolute value. In the cases mentioned, the bounding approaches are appropriate. These approaches are developed in the identification and control theory.(2–6)

Recently, the important results have been obtained in many works in which are considered adaptive control systems in the presence of bounded disturbances.(6–18) One of the approaches to the solution of the adaptive control problem for bounded disturbance case which has been proposed in several papers,(9,11,12,14) allows reduction of the problem of the adaptive estimation to finding a single unknown plant parameter vector. An alternative approach is to find an *a posteriori* member-

L. S. Zhiteckij • V. M. Glushkov Institute of Cybernetics, Ukrainian Academy of Sciences, 252207 Kiev, Ukraine.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

ship set of the unknown parameter vector in some parameter space.(13,15) Based on the second approach, the methods of an ellipsoidal estimation are explored in.(7,10,16)

Within the general area of bounding approaches, an original direction was formed in adaptive control theory.(8) This direction reduces the derivation of the adaptive control algorithms to solving some inequalities by using recursive projection procedures which must converge at finite time. Such algorithms ensure the suboptimal adaptive control of system subjected to chaotic but bounded disturbances. These algorithms(17–20) can be modified to cope with various types of bounding uncertainty, including the case when the disturbances have independent bounded time increments.(17)

The main assumption which is usually made in bounded disturbance case is that bounds on the disturbances are known. It turns out that it is also possible to design the adaptive control system in the presence of bounded disturbances with unknown symmetric bounds if a membership set of plant parameters is known.(18) The key idea proposed in Ref. 18 is to exploit one remarkable property of the recursive projection type algorithms which converge at finite time.(8) This property allows adjustment of one additional parameter which is an estimate of an upper bound on the size of disturbance. A different situation is when bounds on the disturbance are unknown and, possibly, asymmetric. A worse situation arises when not only bounds but also a class of the disturbances are unknown *a priori*. For instance, it is unknown whether the disturbance itself is the so-called non-regular bounded signal, as in Ref. 8, or this disturbance is a signal with the non-regular bounded time increments as in Ref. 17. But it turns out(19) that the technique of Ref. 18 can also be extended to these difficult cases.

This chapter is a broadened version of the paper Ref. 19 and is organized in the following way. In Section 24.2, the assumptions regarding the parameters of a plant and disturbances are made and the problem is formulated. Section 24.3 outlines the main features of the adaptive control algorithms which must be convergent during finite time. In Section 24.4, the optimal control law are presented for the known parameter cases. The main result is given in Section 24.5 which synthesizes the adaptive identification algorithms for control of systems subjected to bounded disturbances with unknown parameters. Section 24.6 presents a report on the simulation. Section 24.7 concludes the chapter.

24.2. PROBLEM STATEMENT

24.2.1. Description of the Plant

Consider a plant as a discrete-time, linear, time-invariant, lth order system which, for simplicity of exposition, has the unit delay and satisfies the difference equation

$$y(t) + a_1 y(t-1) + \cdots + a_l y(t-l)$$

$$= b_1 u(t-1) + \cdots + b_l u(t-l) + \zeta(t) \tag{24.1}$$

with

$$b_1 \neq 0 \tag{24.2}$$

where $y(t) \in \mathbb{R}^1$, $u(t) \in \mathbb{R}^1$ are measurable output and control input, respectively, and $\zeta(t) \in \mathbb{R}^1$ is an unmeasurable disturbance at discrete time t ($t = 0,1,2, \cdots$). This equation may be rewritten in an equivalent compact form as

$$A(\mathbf{a},z^{-1})y(t) = B(\mathbf{b},z^{-1})u(t) + \zeta(t) \tag{24.3}$$

with the polynomials

$$A(\mathbf{a},z^{-1}) = 1 + a_1 z^{-1} + \cdots + a_l z^{-l} \tag{24.4}$$

$$B(\mathbf{b},z^{-1}) = b_1 z^{-1} + \cdots + b_l z^{-l} \tag{24.5}$$

in which z^{-1} denotes the unit delay operator and $\mathbf{a}^{\mathrm{T}} = [a_1, \ldots, a_l]$ and $\mathbf{b}^{\mathrm{T}} = [b_1, \ldots, b_l]$ are the parameter vectors.

As in Ref. 1, Eq. (24.1) can also be represented in the form

$$y(t) = \theta_{\mathrm{o}}^{\mathrm{T}} \mathbf{w}(t-1) + \zeta(t) \tag{24.6}$$

where

$$\theta_{\mathrm{o}} = [\mathbf{a}^{\mathrm{T}}, \mathbf{b}^{\mathrm{T}}]$$

and

$$\mathbf{w}^{\mathrm{T}}(t) = [-y(t), \cdots, -y(t-l+1), u(t), \cdots, u(t-l+1)].$$

The following assumptions regarding the plant parameters are made.

- *Assumption* 1a: Sign b_1 is known.
- *Assumption* 1b: The parameter vectors **a**, **b** are unknown but it is known that

$$\mathbf{a} \in \Omega_{\mathrm{a}} \subset \mathbb{R}^l \qquad \mathbf{b} \in \Omega_{\mathrm{b}} \subset \mathbb{R}^l \tag{24.7}$$

  in which Ω_{a} and Ω_{b} are known bounded, convex regions in the l-dimensional Euclidean space $\mathbb{R}^l$.
- *Assumption* 2: The coefficients of $B(\mathbf{b},z^{-1})$ satisfy

$$zB(\mathbf{b}, z^{-1}) \neq 0 \ \text{ for all } |z| \geq 1. \tag{24.8}$$

That is, Eq. (24.3) describes a minimum phase plant. Moreover, it is assumed that $zB(\hat{\mathbf{b}}, z^{-1}) \neq 0$ for all $|z| \geq 1$ and any $\hat{\mathbf{b}} \in \Omega_{\mathrm{b}}$.

Before making the assumptions concerning the disturbance $\zeta(t)$ first give the following definition.

- *Definition 1*:[(8)] The signal $\xi(t) = \xi(t,\omega)$ which depends on t and on an abstract parameter ω making the sense of a distinctive "event" parameter is said to be non-regular (non-stochastic) in a bounded set Ξ if for any natural N and for any $\xi_1, \cdots, \xi_{\mathrm{N}}$ from Ξ there is always an $\omega \in \{\omega\}$ such that

$$\xi(1,\omega) = \xi_1, \cdots, \xi(N,\omega) = \xi_{\mathrm{N}}.$$

Using this definition, distinguish between the following two cases.

- *Case* 1: $\zeta(t)$ is a non-regular bounded signal, i.e.,

$$\varepsilon_i \leq \zeta(t) \leq \varepsilon_{\mathrm{s}} \tag{24.9}$$

where bounds ε_i and ε_{s} are finite. An interval $\Xi = [\varepsilon_i, \varepsilon_{\mathrm{s}}]$ may here be both asymmetric when $\varepsilon_i \neq -\varepsilon_{\mathrm{s}}$ and symmetric when $\varepsilon_{\mathrm{s}} = -\varepsilon_i = \varepsilon$. In this well-known latter case

$$|\zeta(t)| \leq \varepsilon. \tag{24.10}$$

- *Case* 2: $\zeta(t)$ has non-regular bounded time increments $\xi(t)$. In this case $\zeta(t)$ obeys the equation

$$\zeta(t) = \zeta(t-1) + \xi(t) \tag{24.11}$$

where $\xi(t)$ satisfies

$$|\xi(t)| \leq \varepsilon_{\mathrm{V}}. \tag{24.12}$$

- *Remark* 1: Obviously, Case 1 and Case 2 are obtained as particular cases from the equation

$$\zeta(t) + g\cdot\zeta(t-1) = \xi(t) \tag{24.13}$$

in which $\xi(t)$ is a non-regular signal. Indeed, Eq. (24.13) leads to Case l for $g = 0$ and $\xi(t) \in [\varepsilon_i, \varepsilon_{\mathrm{s}}]$ and to Case 2 for $g = -1$ and $\xi(t) \in [-\varepsilon_{\mathrm{V}}, \varepsilon_{\mathrm{V}}]$.
- *Remark* 2: If $0 < |g| < 1$ and Eq. (24.12) holds, then

$$\zeta(t) = \zeta(t-1) + \xi_1(t) \tag{24.14}$$

which is similar to Eq. (24.11), is satisfied formally with

$$|\xi_1(t)| \leq 2(1+g)^{-1}(\varepsilon_{\mathrm{s}} - \varepsilon_i) < \infty.$$

Although $\xi_1(t)$ in Eq. (24.14) is also a bounded signal, it is not non-regular (in contrast to $\xi(t)$ in Eq. (24.11)), and it is essential.

- *Remark* 3: It is not hard to see that if Eq. (24.9) holds then

$$|\nabla\zeta(t)| \leq 2\varepsilon \tag{24.15}$$

where

$$\nabla\zeta(t) \overset{\Delta}{=} \zeta(t) - \zeta(t-1) \tag{24.16}$$

and

$$\varepsilon = (\varepsilon_s - \varepsilon_i)/2. \tag{24.17}$$

In this case, Eq. (24.11) is satisfied with $|\xi(t)| \leq 2\varepsilon$. Meanwhile, $\xi(t)$ is not a non-regular signal.

Now, make the following assumptions about the disturbance $\zeta(t)$.

- *Assumption* 3a: It is known that $\zeta(t)$ satisfies either Eq. (24.9) or Eq. (24.11) together with Eq. (24.12).
- *Assumption* 3b: The bounds ε_i, ε_s in Eq. (24.9) and ε_∇ in Eq. (24.12) are unknown.
- *Assumption* 3c: It is unknown which of the cases, namely Case 1 or Case 2, takes place (in addition to Assumption 3b).

24.2.2. Control Objective

Let $y^0 \equiv$ const be a desired plant output (a fixed set-point). Our problem is to minimize

$$J \overset{\Delta}{=} \limsup_{t\to\infty} |e(t)| \tag{24.18}$$

by choosing the control signal sequence $\{u(t)\} = \mathrm{u}(1), \mathrm{u}(2), \cdots$ where $e(t)$ is an output error defined by

$$e(t) \overset{\Delta}{=} y^0(t) - y(t). \tag{24.19}$$

- *Definition* 2: The control is said to be optimal if

$$J = J^0 \overset{\Delta}{=} \min_{\{u(\mathrm{t})\}} J. \tag{24.20}$$

- *Definition* 3: The control is said to be suboptimal if

$$J \leq J^{\mathrm{o}} + \delta \tag{24.21}$$

for arbitrary pre-specified positive constant δ.

The control objective is to design an adaptive controller which ensures the suboptimal control under Assumptions 1–3.

Note that if bounds on the disturbances are symmetric and known *a priori* then the optimal adaptive control can be designed.(12) However, there is no solution of the optimal adaptive control problem for the case of no *a priori* information about the above bounds.

24.3. PRELIMINARIES

Adaptive control algorithms are derived on the basis of the following common scheme.(8) First design the optimal control law assuming for a while that all parameters of Eq. (24.3) and of Eq. (24.13) are known. To make the control law adaptive replace the true parameters by estimated parameters. The recursive projection procedure for solving an infinite system of inequalities is chosen as an estimation algorithm. It is essential that the above algorithm must converge at finite time. The finite convergence is known to be achieved only if boundedness of system variables is guaranteed.(8) To establish such a boundedness, one needs the following key technical lemma which is the reformulation of Theorem 4.П.3 given in Ref. 8.

LEMMA 1. Consider a plant described by equation

$$\tilde{A}(\mathbf{a},z^{-1})y(t) = B(\mathbf{b},z^{-1})\tilde{u}(t) + \tilde{\zeta}(t) \tag{24.22}$$

in which $\tilde{A}(\mathbf{a},z^{-1})$ is an arbitrary monic polynomial of degree $\tilde{l} \geq l$. Suppose that Eq. (24.8) is satisfied, and $\zeta(t)$ is bounded in modulus. Let

$$|e(t)| \leq \text{const} + \delta_* \|\mathbf{v}(t-1)\| \tag{24.23}$$

be satisfied for some $\delta_* > 0$ where

$$\mathbf{v}^{\mathrm{T}}(t) = [-y(t), \cdots, -y(t-\tilde{l}+1), \tilde{u}(t), \cdots, \tilde{u}(t-l+1)]$$

and $\tilde{u}(t)$ and $\tilde{\zeta}(t)$ denote an equivalent control signal and an equivalent disturbance, respectively. Then the closed-loop system, consisting of a plant described by Eq. (24.22) and a controller which causes Eq. (24.23), is stable in the sense that for any initial $\mathbf{v}(0)$ (irrespective of the chosen control law) there exists a bounded region $D_{\mathrm{v}} \subset \mathbb{R}^{2l}$ and a finite t_{D} such that

$$\|\mathbf{v}(t)\| \leq D_{\mathrm{v}} < \infty \quad \text{for all } t \geq t_{\mathrm{D}} \tag{24.24}$$

if δ_* satisfies

$$\delta_*[\tilde{l} + l\cdot\max_{\lambda:\,|\lambda|=1} |\tilde{A}(\mathbf{a},\lambda)B^{-1}(\mathbf{b},\lambda)|^2] < 1 \tag{24.25}$$

where λ is the complex variable and $\|\cdot\|$ denotes the Euclidean vector norm.

PROOF. (See Section 4.П.5° of Ref. 8.) □

24.4. OPTIMAL CONTROL OF SYSTEMS IN THE PRESENCE OF BOUNDED DISTURBANCES WITH KNOWN PARAMETERS

24.4.1. Case 1

For the time being suppose that $\zeta(t)$ satisfies Eq. (24.9) with known ε_i, ε_s. Let $\boldsymbol{\theta}_0$ be known. To derive the feedback control law, rewrite Eq. (24.1) in the equivalent form as

$$\begin{aligned} y(t) + a_1 y(t-1) + \cdots + a_l y(t-l) + \bar{\varepsilon} \\ = b_1 u(t-1) + \cdots + b_l u(t-l) + \tilde{\zeta}(t) \end{aligned} \tag{24.26}$$

where

$$\bar{\varepsilon} = (\varepsilon_i + \varepsilon_s)/2 \tag{24.27}$$

and $\tilde{\zeta}(t)$ is an equivalent disturbance satisfying

$$|\tilde{\zeta}(t)| \leq \varepsilon \tag{24.28}$$

with ε given by Eq. (24.17).

Equation (24.26) can be presented by

$$y(t) = \tilde{\boldsymbol{\theta}}_0^{\mathrm{T}} \cdot \tilde{\mathbf{w}}^{\mathrm{T}}(t-1) + \tilde{\zeta}(t) \tag{24.29}$$

where

$$\tilde{\boldsymbol{\theta}}_0^{\mathrm{T}} = [\boldsymbol{\theta}_0^{\mathrm{T}}, \bar{\varepsilon}] \tag{24.30}$$

$$\tilde{\mathbf{w}}^{\mathrm{T}}(t) = [\mathbf{w}^{\mathrm{T}}(t), 1]. \tag{24.31}$$

The following results from the optimal control.

LEMMA 2. Suppose that (a) Eqs. (24.2 and 24.8) are satisfied, (b) $\zeta(t)$ is a non-regular signal described by Eq. (24.9), and (c) ε_s and ε_i are known.

Then Eq. (24.20) in which

$$J^{\mathrm{o}} = \varepsilon \tag{24.32}$$

is achieved if the control law is chosen as

$$y^{\mathrm{o}} = \tilde{\boldsymbol{\theta}}_{\mathrm{o}}^{\mathrm{T}} \cdot \tilde{\mathbf{w}}(t). \tag{24.33}$$

PROOF: This is a straightforward application of Theorem 3.2.1 of Ref. 8, Eq. (24.29), Eq. (24.28), and the fact that $\tilde{\zeta}(t)$ is a non-regular signal in the symmetric interval $[-\varepsilon, \varepsilon]$. □

24.4.2. Case 2

As before, assume that $\boldsymbol{\theta}_{\mathrm{o}}$ is known but $\zeta(t)$ now satisfies Eq. (24.11) in which $\xi(t)$ is bounded in modulus by a constant ε_{∇}. In view of Eqs. (24.6 and 24.11), write

$$y(t) = y(t-1) + \boldsymbol{\theta}_{\mathrm{o}}^{\mathrm{T}} \cdot \nabla\mathbf{w}(t-1) + \xi(t) \tag{24.34}$$

where

$$\nabla\mathbf{w}(t) \overset{\Delta}{=} \mathbf{w}(t) - \mathbf{w}(t-1). \tag{24.35}$$

The following result regarding the optimal control can be shown to be valid.

LEMMA 3.[17] Suppose that $\zeta(t)$ is a disturbance of the form of Eq. (24.11) in which $\xi(t)$ is a non-regular time increment satisfying Eq. (24.12). Under assumption (a) given in Lemma 2 the control law

$$y^{\mathrm{o}} = y(t) + \boldsymbol{\theta}_{\mathrm{o}}^{\mathrm{T}} \cdot \nabla\mathbf{w}(t) \tag{24.36}$$

achieves Eq. (24.20) with

$$J^{\mathrm{o}} = \varepsilon_{\nabla}. \tag{24.37}$$

PROOF: Eq. (24.37) follows immediately from Theorem 3.2.1 of Ref. 8 applied to Eq. (24.34) with $\xi(t) \in [-\varepsilon_{\nabla}, \varepsilon_{\nabla}]$. □

REMARK 4: Recalling Remark 3, one can show that under conditions of Lemma 2, Eq. (24.36) ensures that $J = 2\varepsilon$. This means that for Case 1, the performance of the Eq. (24.33) is twice as good as that of the Eq. (24.36).

24.5. SUBOPTIMAL ADAPTIVE CONTROL OF SYSTEMS IN THE PRESENCE OF BOUNDED DISTURBANCES WITH UNKNOWN PARAMETERS

24.5.1. Basic Ideas

Now, assume that both the parameters of Eq. (24.3) and bounds on the signal $\xi(t)$ that causes the disturbance $\zeta(t)$ (Eq. (24.13)) are unknown. Clearly, if $g = -1$ and $-\varepsilon_{\nabla} \leq \xi(t) \leq \varepsilon_{\nabla}$ (Case 2) then the problem can be solved[18] to Eq. (24.34) because $\xi(t)$ is a non-regular bounded signal with symmetric bounds and Eq. (24.7) implies that $\boldsymbol{\theta}_{\mathrm{o}} \in \Omega_{\mathrm{o}}$, where

$$\Omega_{\text{o}} \triangleq \Omega_{\text{a}} \times \Omega_{\text{b}} \subset \mathbb{R}^{2l} \tag{24.38}$$

is the known and bounded set. It may seem at first sight that if $g = 0$ and $\xi(t) \in [\varepsilon_i, \varepsilon_s]$ (Case 1), then the above problem can also be reduced to the problem of the suboptimal adaptive control of the equivalent Eq. (24.29) subjected to equivalent disturbance $\zeta(t)$ with unknown but symmetric bound ε. In this case, the equation

$$y^{\text{o}} = \tilde{\boldsymbol{\theta}}^{\text{T}}(t)\cdot\tilde{\mathbf{w}}(t) \tag{24.39}$$

can be chosen as a control law. One is obtained from Eq. (24.33). Replace unknown $\tilde{\boldsymbol{\theta}}_{\text{o}}$ by some $\tilde{\boldsymbol{\theta}}(t)$ defined as

$$\tilde{\boldsymbol{\theta}}^{\text{T}}(t) = [\boldsymbol{\theta}_{\text{l}}^{\text{T}}(t), \bar{\varepsilon}(t)] \tag{24.40}$$

where

$$\boldsymbol{\theta}_{\text{l}}^{\text{T}}(t) = [\mathbf{a}_{\text{l}}^{\text{T}}(t), \mathbf{b}_{\text{l}}^{\text{T}}(t)] \tag{24.41}$$

is an estimate of unknown $\tilde{\boldsymbol{\theta}}_{\text{o}}$ and $\bar{\varepsilon}(t)$ is an estimate of unknown ε at time t. Next, find $\tilde{\boldsymbol{\theta}}(t)$ by solving the system of the inequalities

$$|\tilde{\boldsymbol{\theta}}^{\text{T}}\cdot\tilde{\mathbf{w}}(t-1) - y(t)\,| \le \varepsilon_{\text{l}}^{\text{o}}(t) \tag{24.42}$$

$$(t = 1, 2, \cdots)$$

which can be obtained using Eqs. (24.28 and 24.29), and replacing $\tilde{\boldsymbol{\theta}}_{\text{o}}$ by $\tilde{\boldsymbol{\theta}}$ and ε by some $\varepsilon_{\text{l}}^{\text{o}}(t)$.

However, this is far from being the case. When the upper bound on $\tilde{\zeta}(t)$ in Eq. (24.29) is unknown, then some bounded membership set of $\tilde{\boldsymbol{\theta}}_{\text{o}}$ must be known *a priori.*[18] Nevertheless, from the Eq. (24.30), $\boldsymbol{\theta}_{\text{o}} \in \Omega_{\text{o}}$, and Eq. (24.27) implying that $\bar{\varepsilon} \in \Xi$ where

$$\Xi \triangleq [\varepsilon_i, \varepsilon_{\text{s}}] \subset \mathbb{R}^1 \tag{24.43}$$

one knows only that $\tilde{\boldsymbol{\theta}}_{\text{o}} \in \Theta$ where

$$\Theta \triangleq \Omega_{\text{o}} \times \Xi \subset \mathbb{R}^{2l+1}. \tag{24.44}$$

Although the membership set Θ of $\tilde{\boldsymbol{\theta}}_{\text{o}}$ given by Eq. (24.44) is bounded, this set is unknown, since bounds ε_i and ε_{s} of Ξ which is the membership set of unmeasurable disturbance $\zeta(t)$ are unknown. However, the solution to the problem is possible.

Although this problem seems hopeless at first, the author proposes an approach based on two basic ideas. The first idea is to design an *a posteriori* membership set $\Xi(t) \subset \mathbb{R}^1$ of unmeasurable $\zeta(t)$. Bounds on $\Xi(t)$ can be evaluated from Eq. (24.6) which yields

$$\zeta_i(t) \leq \zeta(t) \leq \zeta_s(t) \qquad (24.45)$$

$$(t = 1, 2, \cdots)$$

where

$$\zeta_i(t) = y(t) - \max_{\theta \in \Omega_0} \theta^T \mathbf{w}(t-1) \qquad (24.46)$$

$$\zeta_s(t) = y(t) - \min_{\theta \in \Omega_0} \theta^T \mathbf{w}(t-1). \qquad (24.47)$$

It follows from Eqs. (24.46 and 24.47) that the set

$$\Xi(t) = [\min_{1 \leq \tau \leq t} \zeta_i(\tau), \max_{1 \leq \tau \leq t} \zeta_s(\tau)] \supseteq \bigcup_{\tau=1}^{t} [\zeta_i(\tau), \zeta_s(\tau)] \qquad (24.48)$$

contains unmeasurable $\zeta(t)$ for every $t \geq 1$. Since the *a priori* set Ω_0 is known, one can always find the bounds on $\Xi(t)$ using the measurement data and Eqs. (24.46 to 24.48).

Fig. 24.1 illustrates the Eq. (24.48) together with Eq. (24.45) for $t = 2$. This figure shows an example when the intervals $[\zeta_i(t), \zeta_s(\tau)]$ and $[\zeta_i(t-1), \zeta_s(t-1)]$ have a non-empty intersection (this condition is not necessary at all).

In order to make use of the first idea, a boundedness of $\Xi(t)$ must be guaranteed.$^{(18)}$ Meanwhile, such a boundedness is still not provided since there is no guarantee that $\|\mathbf{w}(t)\|$ is not unbounded. Therefore $\Xi(t)$ defined by Eq. (24.48) together with Eqs. (24.46 and 24.47) is unbounded as $t \to \infty$ (see Fig. 24.1). Nevertheless, it is almost obvious that the vector $\mathbf{w}(t)$ can successfully be kept within a bounded region $D^0 \subset \mathbb{R}^{2l}$, if there is another controller, which comes into operation whenever $\mathbf{w}$(t) comes out from some subregion $\overline{D}^0 \subset D^0$. This controller must return $\mathbf{w}(t)$ into $\overline{D}^0$ at a finite time. But, again, to design such a controller it is

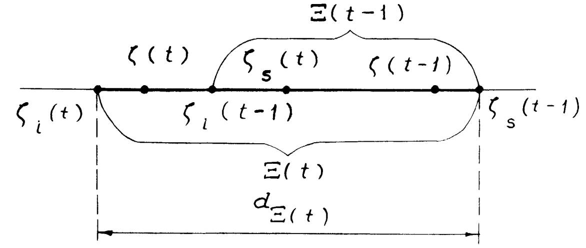


FIGURE 24.1. Construction of the *a posteriori* membership set $\Xi(t)$ given by Eq. (24.48).

necessary to obtain suitable estimates of the unknown plant parameters. All this leads to the second idea: use the second adaptive controller as a stabilizing controller. It turns out that this controller can simultaneously be suboptimal if Case 2 takes place.

24.5.2. Stabilizing Controller

24.5.2.1. Adaptive Control Algorithm

To derive an adaptive law for the second controller, replace the true parameter vector $\boldsymbol{\theta}_o$ in Eq. (24.36) by an estimate parameter vector $\boldsymbol{\theta}(t)$ defined as

$$\boldsymbol{\theta}^{\mathrm{T}}(t) = [\mathbf{a}_2^{\mathrm{T}}(t), \mathbf{b}_2^{\mathrm{T}}(t)] \tag{24.49}$$

where $\mathbf{a}_1^{\mathrm{T}}(t) \not\equiv \mathbf{a}_2^{\mathrm{T}}(t)$ and $\mathbf{b}_1^{\mathrm{T}}(t) \not\equiv \mathbf{b}_2^{\mathrm{T}}(t)$ in general. Then

$$y^{\mathrm{o}} = y(t) + \boldsymbol{\theta}^{\mathrm{T}}(t)\cdot\nabla\mathbf{w}(t), \tag{24.50}$$

which is the second adaptive control law.

Determine $\boldsymbol{\theta}(t)$ using a recursive algorithm for solving the system of inequalities[17]

$$|\boldsymbol{\theta}^{\mathrm{T}}\nabla\mathbf{w}(t-1) - \nabla y(t)| \le \varepsilon_2^{\mathrm{o}}(t) \qquad (t = 1, 2, \cdots) \tag{24.51}$$

where

$$\nabla y(t) \overset{\Delta}{=} y(t) - y(t-1)$$

and $\varepsilon_2^{\mathrm{o}}(t)$, which is specified later, is an estimate of an upper bound on $|\nabla\zeta(t)|$. Equation (24.51) is obtained from Eq. (24.6). Use Eqs. (24.16), (24.35), and the fact that Eqs. (24.12 and 24.15) imply that $|\nabla\zeta(t)| \le E$ where

$$E = \begin{cases} 2\varepsilon & \text{if } \zeta(t) \text{ satisfies Eq. (24.9)} \\ \varepsilon_\nabla & \text{if } \zeta(t) \text{ satisfies Eq. (24.11)} \end{cases} \tag{24.52}$$

is the upper bound on $|\nabla\zeta(t)|$.

The value of $\boldsymbol{\theta}(t)$ is found by the following several subsequent steps.

Step 1: Calculate

$$q_2(t) = \boldsymbol{\theta}^{\mathrm{T}}(t-1)\nabla\mathbf{w}(t-1) - \nabla y(t) \tag{24.53}$$

by substituting $\boldsymbol{\theta} = \boldsymbol{\theta}(t-1)$ in the left side of Eq. (24.51), and

$$\varepsilon_2(t) = \varepsilon_2^{\mathrm{o}}(t) - \delta/2. \tag{24.54}$$

Step 2: Determine

$$\boldsymbol{\theta}'(t) = \boldsymbol{\theta}(t-1) - f(q_2(t),\varepsilon_2^{o}(t),\varepsilon_2(t))\|\nabla\mathbf{w}(t-1)\|^{-2}\cdot\nabla\mathbf{w}(t-1) \tag{24.55}$$

where δ is some constant and

$$f(q(t),\varepsilon^{o},\varepsilon) = \begin{cases} q(t) - \varepsilon & \text{if } q(t) > \varepsilon^{o} \\ 0 & \text{if } |q(t)| \le \varepsilon^{o} \\ q(t) + \varepsilon & \text{if } q(t) < -\varepsilon^{o} \end{cases} \tag{24.56}$$

is the dead zone function such that $0 < \varepsilon < \varepsilon^{o}$.

Comments: Equation (24.55) is a known recursive projection procedure which is investigated in Ref. 8 for the case of

$$\varepsilon_2^{o}(t) \ge E + \delta/2, \tag{24.57}$$

where the dead zone Eq. (24.56) differs from the one used in Refs. 1,2,6 and 12 to 14.

Note Eqs. (24.53, 24.55, and 24.56). Suppose *t*th Eq. (24.51) is satisfied by substituting $\boldsymbol{\theta} = \boldsymbol{\theta}(t-1)$. That is, $\boldsymbol{\theta}(t-1)$ lies inside a band

$$S(t) \triangleq \{\boldsymbol{\theta} \in \mathbb{R}^{2l} : |\, \boldsymbol{\theta}^{\mathrm{T}}\nabla\mathbf{w}(t-1) - \nabla y(t)\,| \le \varepsilon_2^{o}(t)\},$$

Then set $\boldsymbol{\theta}'(t) = \boldsymbol{\theta}(t-1)$. Otherwise, project $\boldsymbol{\theta}(t-1)$, which lies outside $S(t)$, onto the closest of the hyperplanes

$$T_{\varepsilon}^{\pm} \triangleq \{\boldsymbol{\theta} \in \mathbb{R}^{2l}\colon \boldsymbol{\theta}^{\mathrm{T}}\nabla\mathbf{w}(t-1) - \nabla y(t) = \pm\varepsilon_2(t)\}.$$

The last case is illustrated in Figs. 24.2 and 24.3, where

$$T_{\varepsilon^{o}}^{\pm} \triangleq \{\boldsymbol{\theta} \in \mathbb{R}^{2l}\colon \boldsymbol{\theta}^{\mathrm{T}}\nabla\mathbf{w}(t-1) - \nabla y(t) = \pm\varepsilon_2^{o}(t)\}$$

are the hyperplanes.

The boundaries of $S(t)$ are also depicted (for simplicity, drop the argument *t* in the notations of all hyperplanes here).

Step 3: Find $\boldsymbol{\theta}(t)$ as

$$\boldsymbol{\theta}(t) = \arg\min_{\boldsymbol{\theta}\in\Omega_o} \|\boldsymbol{\theta} - \boldsymbol{\theta}'(t)\|. \tag{24.58}$$

Comments: Eq. (24.58) is defined as the orthogonal projection of $\boldsymbol{\theta}'(t)$ onto the known convex, closed and bounded set Ω_0. This procedure is used to ensure $\boldsymbol{\theta}(t) \in \Omega_0$ for all *t*. When the result of Eq. (24.55) lies in Ω_o as shown in Figure 24.2, then $\boldsymbol{\theta}(t) = \boldsymbol{\theta}'(t)$. Otherwise, updated $\boldsymbol{\theta}(t)$ is obtained via mapping $\boldsymbol{\theta}'(t)$ into the closest point $\boldsymbol{\theta}(t)$, which lies on the surface of Ω_o (see Fig. 24.3).

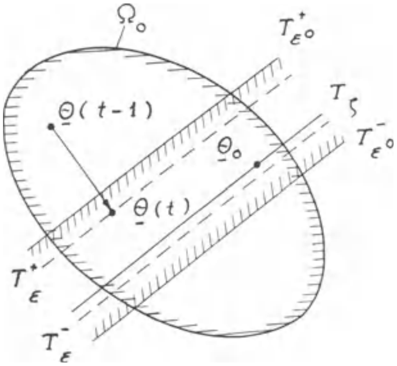


FIGURE 24.2. Geometric interpretation of the estimation Eqs. (24.55 and 24.58) for the case $\mathbf{\theta}_o \in S(t)$. The result of Eq. (24.55) lies inside Ω_o.

Remark 5: Assumption 1 implies that the hyperplane $\{\hat{\mathbf{b}} \in \mathbb{R}^l : \hat{b}_1 = 0\}$ does not intersect Ω_o. It can be proven that the first component of $\mathbf{b}_2(t)$ in $\mathbf{\theta}(t)$, is the coefficient of $u(t)$ in Eq. (24.50), is always nonzero.

Note Eqs. (24.6), (24.16), and (24.35). Suppose $\varepsilon_2^o(t) \geq |\nabla\zeta(t)|$ so some t. Then $\mathbf{\theta}_o$, which lies always on the hyperplane

$$T_\zeta(t) \overset{\Delta}{=} \{\mathbf{\theta} \in \mathbb{R}^{2l} : \mathbf{\theta}^T \nabla \mathbf{w}(t-1) - \nabla y(t) = \nabla\zeta(t)\},$$

also lies inside the band $S(t)$ (see Fig. 24.2). Otherwise $\mathbf{\theta}_o$ lies outside $S(t)$ as depicted in Fig. 24.3.

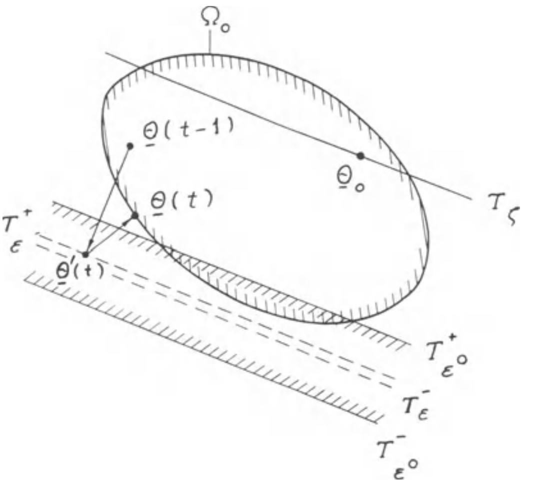


FIGURE 24.3. Geometric interpretation of the estimation Eqs. (24.55 and 24.58) for the case $\mathbf{\theta}_o \notin S(t)$. The result of Eq. (24.55) lies outside Ω_o.

It can be proven that if there exists a finite t_2^o such that the condition

$$\varepsilon_2^o(t) > \limsup_{t\to\infty} |\nabla\zeta(t)| + \delta/2 \tag{24.59}$$

is satisfied for all $t \ge t_2^o$, then there exists a finite $\tilde{t}_2^o \ge t_2^o$ such that

$$|\boldsymbol{\theta}_o^T \nabla \mathbf{w}(t-1) - \nabla y(t)\,| \le \varepsilon_2(t) < \varepsilon_2^o(t) \tag{24.60}$$

for all $t \ge \tilde{t}_2^o$. In this case, the intersection

$$H \triangleq \bigcap_{\tau=\tilde{t}_2^o}^{\infty} S(\tau)$$

includes the true parameter vector $\boldsymbol{\theta}_o$ and its neighborhood. This guarantees that the system of Eq. (24.51) is compatible for $t = \tilde{t}_2^o$, $\tilde{t}_2^o + 1, \cdots$. However, if there is no finite t_2^o such that Eq. (24.59) is satisfied for all $t \ge t_2^o$, then it is not guaranteed that there exists a $\hat{t}_2^o$ such that the intersection

$$S(\hat{t}_2^o) \cap S(\hat{t}_2^o + 1) \cap \cdots$$

is non-empty. In last case, the system of Eq. (24.51) may be incompatible for $t = \hat{t}_2^o$, $\hat{t}_2^o + 1, \cdots$. For this reason, $\varepsilon_2^o(t)$ should be large enough for all sufficiently large t in order to avoid an incompatibility of the above system. In the other hand, $\varepsilon_2^o(t)$ should be small enough to ensure the control suboptimality for Case 2.

To obtain a suitable value of $\varepsilon_2^o(t)$, the author proposes a recursive procedure in the form of two subsequent steps.

Step 1: Determine

$$\varepsilon_2^o(t) = \begin{cases} \varepsilon_2^o(t-1) & \text{if } \eta_2(t-1) \le d^2 \\ \varepsilon_2^o(t-1) + \delta/2 & \text{otherwise} \end{cases} \tag{24.61}$$

where

$$d \triangleq \max_{\boldsymbol{\theta}', \boldsymbol{\theta}'' \in \Omega_o} \|\boldsymbol{\theta}' - \boldsymbol{\theta}''\| \tag{24.62}$$

is the diameter of known Ω_o.

Step 2: Determine

$$\eta_2(t) = \begin{cases} \eta_2(t-1) & \text{for } |q_2(t)| \le \varepsilon_2^o(t) \\ \eta_2(t) + \|\nabla \mathbf{w}(t-1)\|^{-2} \cdot [|q_2(t)| - \varepsilon_2(t)]^2 & \\ & \text{for } |q_2(t)| > \varepsilon_2^o(t) \end{cases} \tag{24.63}$$

$$\text{if } \eta_2(t-1) \le d^2$$

$\eta_2(t) = 0$ otherwise.

Comments: Observe from Eq. (24.61) that the estimate $\varepsilon_2^o(t)$ changes as a piecewise constant, monotonic nondecreasing function in t. Such a change arises whenever the auxiliary inequality $\eta_2(t-1) \le d^2$ is violated at some $t = t_2(k)$ (k = 1, 2, ···). Then associated with Eq. (24.61), $\varepsilon_2^o(t)$ has a jump at $t = t_2(k)+1$. Suppose there exists a finite $\bar{k}$ such that Eq. (24.60) is satisfied for all $t \ge t_2(\bar{k}) + 1$. In this case, the inequality $\eta_2(t-1) \le d^2$ cannot be violated at $t \in [t_2(\bar{k}) + 1, \infty)$ if the initial estimates $\boldsymbol{\theta}(0)$ and $\eta_2(0)$ are chosen so that $\boldsymbol{\theta}(0) \in \Omega_o$ and $\eta_2(0) = 0$. Indeed, assume that Eq. (24.51) is violated at $t = t_2^j$ for the jth time by substituting $\boldsymbol{\theta} = \boldsymbol{\theta}(t_2^j - 1)$ ($j = 1,2, \cdots$). Use the proof of Theorem 2.1.la of Ref. 8 and Eq. (24.62). Since $\boldsymbol{\theta}(t) \in \Omega_o$ for all t, we establish from Eq. (24.63) that the estimation procedure represented by Eq. (24.55), together with Eq. (24.58), has the following remarkable property:

$$\eta_2(t) = \sum_{t_2(\bar{k}) + 2 \le t_2^j \le t} \alpha(t_2^j) \le \|\boldsymbol{\theta}_o - \boldsymbol{\theta}(t)\|^2 \le d^2 \tag{24.64}$$

for all $t_2(\bar{k}) + 2 \le t < \infty$ where

$$\alpha(t_2^j) = \|\nabla \mathbf{w}(t_2^j - 1)\|^{-2} \cdot [|q_2(t_2^j)| - \varepsilon_2(t_2^j)]^2. \tag{24.65}$$

Since Eq. (24.63) yields $\eta_2(t_2(\bar{k}) + 1) = 0$, then from Eq. (24.64) $\eta_2(t) \le d^2$ follows all for $t \ge t_2(\bar{k}) + 1$.

Furthermore, exploit the key property (Eq. 24.64)) together with Lemma 1 to establish the finite convergence of $\{\boldsymbol{\theta}(t)\}$ in some stabilizing system $\bar{S}$ which comprises the plant (Eq. (24.3)), the adaptive controller (Eq. 24.50)) and the identifier (Eq. (24.53–24.56, 24.58, and 24.61–24.63)).

24.5.2.2. Properties of Stabilizing System

The following important preliminary results regarding the properties of the system $\bar{S}$ with the second adaptive controller serve as a foundation for establishing the conditions under which the first adaptive controller (Eq. (24.39)) may come into operation.

Lemma 4. Let Assumption 1 and 3, and Eq. (24.8) be valid. Consider the system $\bar{S}$. Suppose that the Eq. (24.50) starts with some $t = t_i \ge 0$. Choose an arbitrary $\delta > 0$. Then, for $\varepsilon_2^o(0) = \delta$, $\eta_2(0) = 0$, and any $\boldsymbol{\theta}(0) \in \Omega_o$

(i) there exists a finite t_2^+ and ε_2^* such that

$$\varepsilon_2^o(t) = \varepsilon_2^* \equiv \text{const for all t} \ge t_2^+$$

where

$$\varepsilon_2^* < E + \delta \tag{24.66}$$

(ii) there exists a finite $t_2^* \geq t_2^+$ and $\boldsymbol{\theta}^* \in \Omega_0$, $\boldsymbol{\theta}^* \equiv$ const such that $\boldsymbol{\theta}(t) = \boldsymbol{\theta}^*$ and $|y^0 - y(t)| \leq \varepsilon_2^*$ for all $t \geq t_2^*$.

Proof: The proof proceeds by an argument analogous to that used in the proof of Theorem 1 of Ref. 18 via applying Lemma 1 to Eq. (24.22) with

$$\tilde{A}(\mathbf{a},z^{-1}) = (1 - z^{-1})A(\mathbf{a},z^{-1})$$

where $\tilde{u}(t) \overset{\Delta}{=} u(t) - u(t-1)$ and $\tilde{\zeta}(t) \overset{\Delta}{=} \zeta(t) - \zeta(t-1)$, and using Eq. (24.64) together with Eqs. (24.65 and 24.60). (Details are omitted here.) □

Lemma 5. If Assumption 2 holds in addition to the conditions of Lemma 4, then for any $\delta' > 0$ there exists finite $\tau^0 = \tau^0(\delta') \geq t_2^*$ such that

$$|\nabla u(t)| \leq \kappa M \varepsilon_2^* + \delta' \text{ for all } t \geq \tau^0 \tag{24.67}$$

where

$$M = 3 + 2 \max_{\hat{\mathbf{a}} \in \Omega_a} \sum_{\mu=1}^{l} |\hat{a}_\mu| \tag{24.68}$$

$$\kappa = \max_{\hat{\mathbf{b}} \in \Omega_b} \sum_{\nu=0}^{\infty} |\beta_\nu(\hat{\mathbf{b}})| < \infty \tag{24.69}$$

in which $\beta_\nu(\hat{\mathbf{b}})$ are the coefficients of the series

$$z^{-1}B^{-1}(\hat{\mathbf{b}},z^{-1}) = \sum_{\nu=o}^{\infty} \beta_\nu(\hat{\mathbf{b}})z^{-\nu}.$$

Proof: Eq. (24.67) together with Eq. (24.68) follows directly from Lemma 4 and Property D_1 of Ref. 20. The value of κ defined by Eq. (24.69) is obtained in the same way as the similar value given by Theorem 10 in Section 6 of Ref. 21 for continuous-time case. □

24.5.3. Adaptive Control Design

Now return to the derivation of the first estimation algorithm which is a recursive procedure for solving the system of Eq. (24.42). Introducing a variable

$$\rho(t) = \begin{cases} \rho(t-1) + q_2(t) & \text{if } |q_2(t)| \leq \varepsilon_2^0(t) \text{ and } \rho(t-1) \leq 2\varepsilon_1^0(t-1) \\ 0 & \text{otherwise} \end{cases} \tag{24.70}$$

design this algorithm as the following several steps.

Step 1: Calculate

$$\varepsilon_1(t) = \varepsilon_1^{\mathrm{o}}(t) - \delta/2 \tag{24.71}$$

$$q_1(t) = \tilde{\boldsymbol{\theta}}^{\mathrm{T}}(t-1)\tilde{\mathbf{w}}(t-1) - y(t) \tag{24.72}$$

$$d_{\Theta(t)} = (d^2 + d^2_{\Xi(t)})^{1/2} \tag{24.73}$$

where $d_{\Theta(t)}$ is the diameter of the set $\Theta(t) \overset{\Delta}{=} \Omega_{\mathrm{o}} \times \Xi(t)$ and

$$d_{\Xi(t)} = \max_{1 \le \tau \le t} \zeta_s(\tau) - \min_{1 \le \tau \le t} \zeta_i(\tau) \tag{24.74}$$

is the diameter of the set $\Xi(t)$ (see Fig. 24.1).

Step 2: Determine the first estimate $\varepsilon_1^{\mathrm{o}}(t)$ by

$$\varepsilon_1^{\mathrm{o}}(t) = \begin{cases} \min\{\varepsilon_2^{\mathrm{o}}(t),\ \varepsilon_1^{\mathrm{o}}(t-1) + \delta/2\} & \text{if } \eta_1(t-1) > d^2_{\Theta(t)} \\ & \text{or if } \rho(t-1) > 2\varepsilon_1^{\mathrm{o}}(t-1) \\ \varepsilon_2^{\mathrm{o}}(t)/2 & \text{if } \varepsilon_1^{\mathrm{o}}(t-1) < \varepsilon_2^{\mathrm{o}}(t)/2,\ \eta_1(t-1) \le d^2_{\Theta(t)} \\ & \text{and } \rho(t-1) \le 2\varepsilon_1^{\mathrm{o}}(t-1) \\ \varepsilon_1^{\mathrm{o}}(t) & \text{otherwise.} \end{cases} \tag{24.75}$$

Step 3a: Choose $\tilde{\boldsymbol{\theta}}(t) = [\boldsymbol{\theta}_t^{\mathrm{T}}(t), \bar{\varepsilon}(t)]^{\mathrm{T}}$ with $\boldsymbol{\theta}_1(t)$ and $\bar{\varepsilon}(t)$ satisfying

$$\boldsymbol{\theta}_1(t) \in \Omega_{\mathrm{o}} \text{ and } \bar{\varepsilon}(t) \text{ are arbitrary} \tag{24.76}$$

if $\varepsilon_1^{\mathrm{o}}(t) = \varepsilon_2^{\mathrm{o}}(t) = \varepsilon_2^{\mathrm{o}}(0)$.

Step 3b: Set

$$\tilde{\boldsymbol{\theta}}(t) = \tilde{\boldsymbol{\theta}}(t-1) \text{ if } \varepsilon_1^{\mathrm{o}}(t) = \varepsilon_2^{\mathrm{o}}(t) \ne \varepsilon_2^{\mathrm{o}}(0). \tag{24.77}$$

Step 4: Determine

$$\tilde{\boldsymbol{\theta}}'(t) = \tilde{\boldsymbol{\theta}}(t-1) - f(q_1(t), \varepsilon_1^{\mathrm{o}}(t), \varepsilon_1(t))\|\tilde{\mathbf{w}}(t-1)\|^{-2} \cdot \tilde{\mathbf{w}}(t-1) \tag{24.78}$$

if $\varepsilon_1^{\mathrm{o}}(t) < \varepsilon_2^{\mathrm{o}}(t)$.

Step 5: Find $\tilde{\boldsymbol{\theta}}(t)$ from the condition

$$\tilde{\boldsymbol{\theta}}(t) = \arg\min_{\tilde{\boldsymbol{\theta}} \in \Theta(t)} \|\tilde{\boldsymbol{\theta}} - \tilde{\boldsymbol{\theta}}'(t)\|. \tag{24.79}$$

Step 6: Determine $\eta_1(t)$ as follows:

$$\eta_1(t) = 0$$

if $\varepsilon_1^{\mathrm{o}}(t) = \varepsilon_2^{\mathrm{o}}(t) = \varepsilon_2^{\mathrm{o}}(0)$, and

$$\eta_1(t) = \eta_1(t-1)$$

if $\varepsilon_1^{\rm o}(t) = \varepsilon_2^{\rm o}(t) \neq \varepsilon_2^{\rm o}(0)$, and

$$\eta_1(t) = \begin{cases} \eta_1(t-1) & \text{for } |q_1(t)| \leq \varepsilon_1^{\rm o}(t) \\ \eta_1(t) + \|\tilde{\mathbf{w}}(t-1)\|^{-2}\cdot[|q_1(t)| - \varepsilon_1(t)]^2 & \\ & \text{for } |q_1(t)| > \varepsilon_1^{\rm o}(t) \end{cases} \tag{24.80}$$

$$\text{if } \eta_1(t-1) \leq d^2_{\Theta(t)},$$

$$\eta_1(t) = 0 \text{ otherwise}$$

if $\varepsilon_1^{\rm o}(t) < \varepsilon_2^{\rm o}(t)$.

Now, it only remains to formulate a rule of the controllers switching as follows. Set $\varepsilon_1^{\rm o}(0) = \varepsilon_2^{\rm o}(0)$. The second controller (Eq. (24.50)) comes always into operation at $t = t_{\rm o} = 0$. This controller continues to be active as long as $\varepsilon_1^{\rm o}(t) = \varepsilon_2^{\rm o}(t)$. If $\varepsilon_1^{\rm o}(t) < \varepsilon_2^{\rm o}(t)$ (the case $\varepsilon_1^{\rm o}(t) > \varepsilon_2^{\rm o}(t)$ is impossible due to Eq. (24.75)) at some $t = t_1$, then the first controller (Eq. (24.39)) comes into operation and the controller (Eq. (24.50)) becomes inactive. Starting with $t = t_1$, the controller (Eq. (24.39)) continues to be active until either $|e(t)| > 2\varepsilon_1^{\rm o}(t)$ or $\varepsilon_1^{\rm o}(t) = \varepsilon_2^{\rm o}(t)$ occur at some $t = t_2$. Then the controller (Eq. (24.50)) starts at $t = t_2$ and the controller (Eq. (24.39)) becomes inactive, again. The controller (Eq. (24.50)) continues to be active until all the conditions $|e(t-n)| \leq \varepsilon_2^{\rm o}(t)$ and

$$|\nabla u(t-n)| \leq \kappa M \varepsilon_2^{\rm o}(t) + \delta' \ (n = 1, \cdots, l)$$

and $\varepsilon_1^{\rm o}(t) < \varepsilon_2^{\rm o}(t)$ are satisfied at some $t = t_3$. Then the controller (Eq. (24.39)) comes into operation at $t = t_3$, and so forth.

To give this narrative description a compact mathematical form, introduce a indicator function $s(t)$ defined as $s(t) = m$, $m \in \{1,2\}$ if the mth controller is active at time t. Use Eq. (24.81) to obtain the following decision rule:

$$s(t) = 2 \text{ if } \varepsilon_1^{\rm o}(t) = \varepsilon_2^{\rm o}(t) \tag{24.82}$$

$$s(t) = \begin{cases} 1) \text{ if } s(t-n) = 2 \text{ and } |e(t-n)| \leq \varepsilon_2^{\rm o}(t) \text{ and} \\ \qquad |\nabla u(t-n)| \leq xM\varepsilon_2^{\rm o}(t) + \delta' \text{ for } n = 1, \cdots, l \\ 2) \text{ if } s(t-1) = 1 \text{ and } |e(t)| > 2\varepsilon_1^{\rm o}(t) \\ s(t-1) \text{ otherwise} \end{cases} \tag{24.83}$$

Eqs. (24.39, 24.70–24.80, 24.50, 24.53–24.56, 24.58, and 24.61–24.63) together with Eqs. (24.46–24.48) and Eqs. (24.81–24.83) define the adaptive control algorithm in full. To realize this algorithm, one needs the preliminary calculation of the values of d, M and κ by Eqs. (24.62, 24.68 and 24.69), on the basis of *a priori* knowledge about the sets $\Omega_{\rm a}$ and $\Omega_{\rm b}$.

24.5.4. Finite Convergence Properties

Let t_i ($i = 1,2, \cdots$) be moments of controller switchings. According to Eqs. (24.82) and (24.83), $s(t) = 1\ \forall\, t \in \Delta_i$ with odd i and$s(t) = 2 \forall\, t \in \Delta_i$ with even i where $\Delta_i \overset{\Delta}{=} [t_i, t_{i+1} - 1]$. Exploiting the properties of system $\overline{S}$ which are given by Lemmas 4 and 5 one can conclude that the chosen decision rule guarantees that $\nabla \mathbf{w}(t)$ is uniformly bounded in the norm for all $t \in \Delta_i$. From this fact and Lemma 4 it follows that there exist finite t_2^* , $\boldsymbol{\theta}_2^*$, and ε_2^* satisfying Eq. (24.66) such that $\varepsilon_2^{\mathrm{o}}(t) = \varepsilon_2^*$ and $\boldsymbol{\theta}_2(t) = \boldsymbol{\theta}_2^* \equiv \text{const}\ \forall\, t \geq t_2^*$.

There are three different cases to be examined in order to establish the finite convergence of sequences $\{\varepsilon_1^{\mathrm{o}}(t)\}$, $\{\widetilde{\boldsymbol{\theta}}(t)\}$:

- (Case 1) $\zeta(t)$ satisfies Eq. (24.9)
- (Case 2a) $\zeta(t)$ satisfies Eq. (24.11) and $\zeta(t)$ is bounded
- (Case 2b) $\zeta(t)$ satisfies Eq. (24.11) and $|\zeta(t)| \to \infty$ as $t \to \infty$.

From Lemmas 4 and 5 it follows that the boundedness of $\Theta(t) \overset{\Delta}{=} \Omega_0 \times \Xi(t)$ is guaranteed for Case 1 and Case 2a. If Case 1 or Case 2a take place, then there exist finite

$$t_1^+, t_1^*, t_1^+ \leq t_1^*$$

and ε_1^*, $\widetilde{\boldsymbol{\theta}}^*$ such that

$$\varepsilon_1^{\mathrm{o}}(t) = \varepsilon_1^* \equiv \text{const}\ \forall\ t \in [t_1^+, \infty)$$

and

$$\widetilde{\boldsymbol{\theta}}(t) = \widetilde{\boldsymbol{\theta}}^* \equiv \text{const}\ \forall\ t \in [t_1^*, \infty)$$

where $\varepsilon_1^* < \varepsilon + \delta < \varepsilon_2^*$ for Case 1 and $\varepsilon_1^* = \varepsilon_2^*$ for Case 2a (this result can be established in the same way as in the proof of Lemma 4).

It can be proven that if Case 2b takes place, then, due to Eq. (24.70), the number of violations of the inequality $\rho(t-1) \leq 2\varepsilon_1^{\mathrm{o}}(t-1)$ increases indefinitely as $t \to \infty$. Then, it follows from Eq. (24.75) that there exist finite $t_1^+ \geq t_2^*$ such that

$$\varepsilon_1^{\mathrm{o}}(t) = \varepsilon_2^*\ \ \forall\, t \in [t_1^+, \infty).$$

This yields

$$\widetilde{\boldsymbol{\theta}}(t) = \widetilde{\boldsymbol{\theta}}^* \equiv \text{const}\ \forall\, t \in [t_1^+, \infty).$$

Clearly, if Case 2a or Case 2b take place then due to Eq. (24.82)

$$s(t) = 2\ \forall\ t \in [t_1^+, \infty).$$

Setting $t_1^{\dagger} = t_i$ and employing Lemma 4 obtain

$$|e(t)| < \varepsilon_{\nabla} + \delta \ \forall t \geq t_2^* + 1.$$

It follows from Eq. (24.83) that for Case 1

$$s(t) = 1 \ \forall t \geq t_1^*.$$

Since

$$q_1(t) = -e(t) \ \forall t \geq t_1^* + 1$$

and $\varepsilon_1^* < \varepsilon + \delta$, this leads to

$$|e(t)| < \varepsilon + \delta \ \forall t \geq t_1^* + 1.$$

The results thus established are summarized in the following.

THEOREM 1. Consider the closed-loop system, consisting of the plant (Eq. (24.1)), the controllers (Eqs. (24.39) and (24.50)), the identifiers (Eq. (24.70–24.80)), Eqs. (24.53–24.56, 24.58, and 24.61–24.63) together with the decision rule (Eqs. (24.82) and (24.83)). Let Assumptions 1–3 be valid. Choose $\delta > 0$ and $\delta' > 0$ arbitrarily. Then for $\varepsilon_1^{\circ}(0) = \varepsilon_2^{\circ}(0) = \delta$ and any initial $\boldsymbol{\theta}(0) \in \Omega_o$, $\bar{\varepsilon}(0)$:

(i) $\{\varepsilon_1^{\circ}(t)\}$, $\{\varepsilon_1^{\circ}(t)\}$, $\{\boldsymbol{\theta}(t)\}$, $\{\tilde{\boldsymbol{\theta}}(t)\}$ converge at a finite time t^*

(ii) the number of the controller switchings is finite

(iii) $|e(t)| < \begin{cases} \varepsilon + \delta & \text{for Case 1} \\ \varepsilon_{\nabla} + \delta & \text{for Case 2} \end{cases}$

for all $t \geq t^*$.

It follows from part (iii) of Theorem 1 and Lemmas 1 and 3 and Definition 3 that the designed adaptive control is suboptimal.

24.6. SIMULATION RESULTS

To illustrate the main features and the power of the designed adaptive control algorithm, the results of three simulation experiments are presented here. To this end, $\zeta(t)$ was chosen as $\zeta(t) = \chi(t) + \xi(t)$ where $\chi(t) \equiv 0$ (run 1), $\chi(t) = 0.3$ (run 2), $\chi(t) = \zeta(t-1)$ (run 3) and $\xi(t) \in [-0.\ 4, 0.4]$ is the pseudorandom variable.

The conditions of the experiments:[17] $l = 1$, $a_1 = -0.45$ and $b_1 = 4.0$. The *a priori* sets Ω_a, Ω_b were defined as $\Omega_a = [-0.8, -0.2]$ and $\Omega_b = [2.5, 5.0]$. Then, Eqs. (24.62, 24.68, and 24.69), give $d^2 = 6.61$, $M = 4.6$ and $\kappa = 0.4$. In all runs, the initial estimates are chosen: $a_1(0) = -0.8$ and $b_1(0) = 2.5$.

The results of a 600-step long simulation for $y^0 = 4.0$, $\delta = 0.2$, and $\delta' = 0.2$ are depicted in Figs. 24.4–24.9. It turns out that $\varepsilon_1^{\circ}(t)$ remains less than $\varepsilon_2^{\circ}(t)$ in Case 1

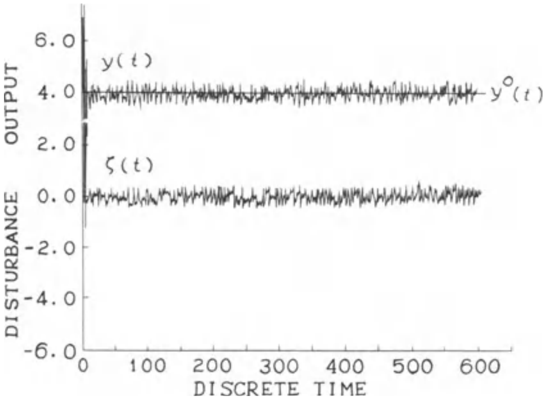


FIGURE 24.4. Plant output and disturbance for the case when the bounds on $\zeta(t)$ are symmetric: $-0.4 \le \zeta(t) \le 0.4$ (run 1).

for $t > 12$ (see Figs. 24.5 and 24.7) and equal to $\varepsilon_2^o(t)$ in Case 2 for $t > 520$ (see Fig. 24.9); herein $\varepsilon_1^o(t) \le 0.4 < \varepsilon + \delta\ (= 0.6)$,

$$\varepsilon_2^o(t) \le 0.7 < E + \delta\ (= 1.0)\ (\text{run } 1)$$

and

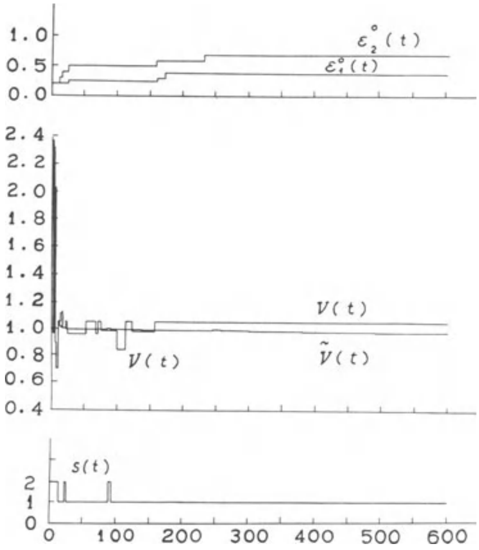


FIGURE 24.5. Parameter estimation and controller switchings for the run 1.

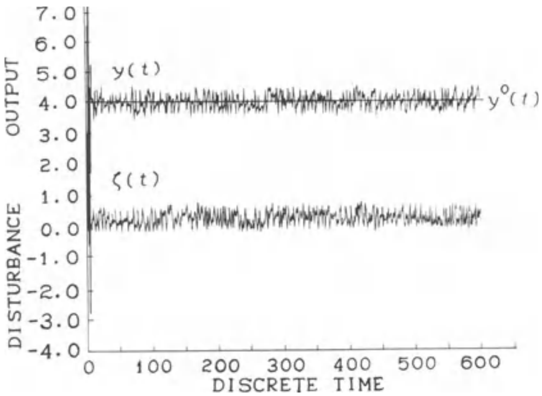


FIGURE 24.6. Plant output and disturbance for the case when the bounds on $\zeta(t)$ are symmetric: $-0.1 \le \zeta(t) \le 0.7$ (run 2).

$$\varepsilon_1^o(t) \le 0.45 < \varepsilon + \delta,$$

$$\varepsilon_2^o(t) \le 0.7 < E + \delta \text{ (run 2)}$$

and

$$\varepsilon_1^o(t),\ \varepsilon_2^o(t) \le 0.5 < \varepsilon_\nabla + \delta\ (= 0.6) \text{ (run 3)}$$

for all $t \le 600$. Figures 24.5, 24.7, and 24.9 show that in all runs, the variables

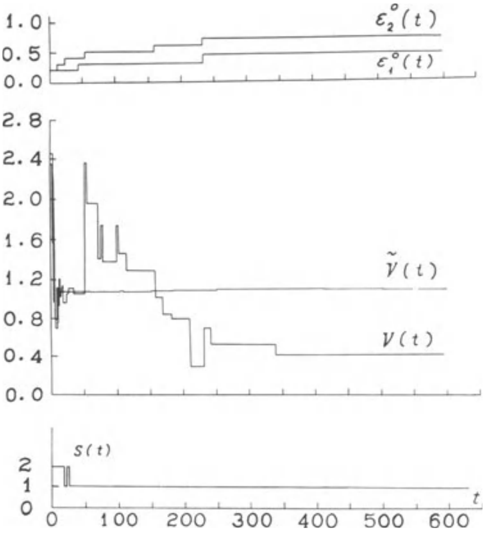


FIGURE 24.7. Parameter estimation and controller switchings for the run 2.

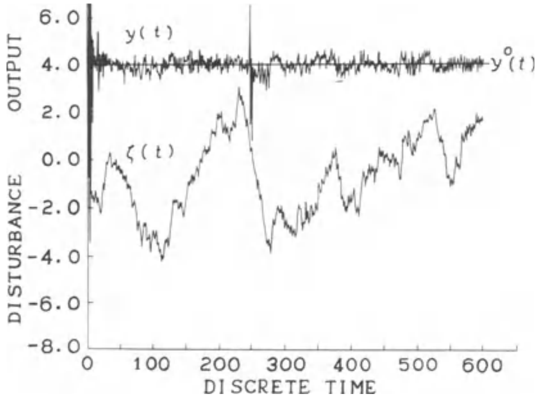


FIGURE 24.8. Plant output and disturbance for the case when $\zeta(t) = \zeta(t-1) + \xi(t)$, where $-0.4 \le \xi(t) \le 0.4$ (run 3).

$$V(t) = \|\boldsymbol{\theta}_0 - \boldsymbol{\theta}(t)\|^2, \qquad \tilde{V}(t) = \|\tilde{\boldsymbol{\theta}}_0 - \tilde{\boldsymbol{\theta}}(t)\|^2$$

are not nonincreasing in t (in contrast to $V(t)$).[8,12] The indicator function $s(t)$, which is depicted in Figs. 24.5, 24.7, and 24.9 demonstrates how the controllers switching occurs. In the end, the controller which is suboptimal becomes active all along,

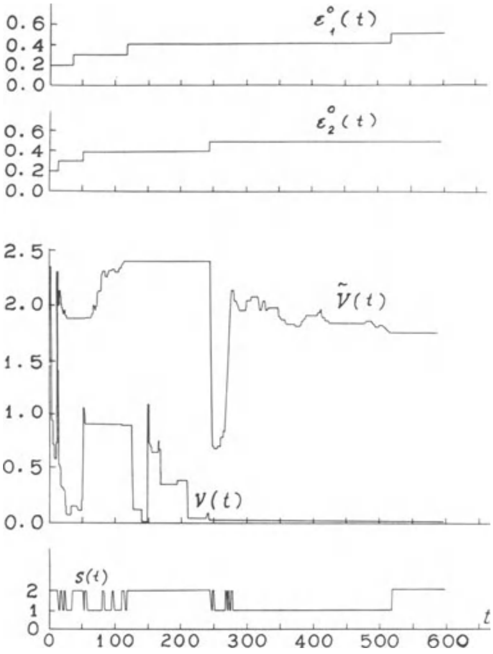


FIGURE 24.9. Parameter estimation and controller switchings for the run 3.

namely, the first controller in Case 1 (Figs. 24.5 and 24.7) and the second controller in Case 2 (Fig. 24.9).

Although $V(t)$ and $\widetilde{V}(t)$ do not go to zero as the parameter estimates converge to their final values, except $V(t)$ for Case 2 (see Fig. 24.9) the proposed adaptive control algorithm copes with different disturbances $\zeta(t)$ both in Case 1 and in Case 2 (see Figs. 24.4, 24.6, and 24.8). A comparison of Figs. 24.4 and 24.6 with Fig. 24.8 shows that no appreciable difference is noticed in the behavior of the plant output $y(t)$, while the disturbances $\zeta(t)$ are different in all these cases.

24.7. CONCLUSIONS

This chapter shows that within one of the bounding approaches it is possible to design the suboptimal adaptive system in the presence of bounded disturbances with unknown parameters. The approach used here relies on the finite convergence property of the recursive projection algorithms for solving of the inequalities. The effectiveness of this approach is demonstrated by simulation runs.

REFERENCES

1. G. C. Goodwin and K. S. Sin, *Adaptive Filtering Prediction and Control*, Prentice Hall, Englewood Cliffs, NJ (1984).
2. K. Forsman and L. Ljung, in: Vol. 2 of *Proc. of the 9th IFAC/IFORS Symposium* (Cs. Bànyàsz and L. Keviczky, eds.) Budapest, Hungary, pp. 1410–1414 (1991).
3. M. Milanese and A. Vicino, in: Vol. 1 of *Proc. of the 9th IFAC/IFORS Symposium* (Cs. Bànyàsz and L. Keviczky, eds.) Budapest, Hungary, pp. 859–867 (1991).
4. S. M. Veres and J. P. Norton, in: Vol. 2 of *Proc. of the 9th IFAC/IFORS Symposium* (Cs. Bànyàsz and L. Keviczky, eds.) Budapest, Hungary, pp. 1038–1043 (1991).
5. E. Walter and H. Piet-Lahanier, in: Vol. 1 of *Proc. of the 9th IFAC/IFORS Symposium* (Cs. Bànyàsz and L. Keviczky, eds.) Budapest, Hungary, pp. 763–768 (1991).
6. B. Egardt, *Stability of Adaptive Controllers: Lecture Notes in Control and Information Sciences*, Springer-Verlag, New York (1979).
7. E. Fogel and Y. F. Huang, *Automatica* **18**, 229 (1982).
8. V. N. Fomin, A. L. Fradkov, and V. A. Yakubovich, *Adaptive Control of Dynamic Plants*, Nauka, Moscow, Russia (1981).
9. G. Kreisselmeier and K. S. Narendra, *IEEE Trans. Autom. Control*, **AC-27**, 1169 (1982).
10. R. Lozano-Leal and R. Ortega, *Automatica* **23**, 247 (1987).
11. J. M. Martin-Sanchez, *IEEE Trans. Autom. Control*, **AC-29**, 461 (1984).
12. R. Ortega and R. Lozano-Leal, *Automatica* **23**, 253 (1987).
13. B. Peterson and K. S. Narendra, *IEEE Trans. Autom. Control*, **AC-27**, 1161 (1982).
14. C. Samson, *Automatica* **19**, 81 (1983).
15. Y. M. Kuntsevich and S. A. Nikitenko, in: Vol. 1 of *Proc. of the 9th IFAC/IFORS Symposium* (Cs. Bànyàsz and L. Keviczky, eds.) Budapest, Hungary, pp. 328–331 (1991).
16. G. M. Bakan and Y. T. Strashko, *Avtomatika i telemekhanika*, **2**, 89 (1980).
17. L. S. Zhiteckij, *Kibernetika i Vychisl. Tekhnika* **60**, 17 (1983).
18. V. A. Bondarko, *Dokl. AN SSSR* **270**, 301 (1983).

19. L. S. Zhiteckij, in: Vol. 1 of *Proc. of the 9th IFAC/IFORS Symposium* (Cs. Bànyàsz and L. Keviczky, eds.) Budapest, Hungary, pp. 585–590 (1991).
20. B. D. Lubachevsky, *Avtomatika i telemekhanika*, **3**, 83 (1974).
21. S. W. Director and R. A. Rohrer, *Introduction to System Theory*, McGraw-Hill, New York (1972).

25

Predictive Self-Tuning Control by Parameter Bounding and Worst-Case Design

S. M. Veres and J. P. Norton

ABSTRACT

The computation of bounds on the parameters of a plant model allows worst-case control synthesis, taking account of the uncertainty in the model. This chapter introduces such a control scheme: predictive bounding control. The scheme contrasts with existing self-tuning control methods which base control synthesis on a nominal plant model. Parameter bounding also permits detection of abrupt plant changes, and adaptive tracking of time-varying plant characteristics by suitable choice of bounds on plant-model output error and plant-parameter increments. Estimation and control are closely integrated, and the control computation can compromise between reducing the model uncertainty and reducing predicted output error. Simulation examples show the excellent performance of predictive bounding control.

25.1. INTRODUCTION

The range of possible techniques for adaptive control has been broadened by the algorithms now available for computing bounds on the parameters of a difference-equation model of a dynamical system.[1–8]

S. M. Veres and J. P. Norton • School of Electronic and Electrical Engineering, University of Birmingham, Edgbaston, Birmingham B15 2TT, United Kingdom.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

The bounds define the set of parameter values giving model-output errors within prescribed bounds, and may be regarded as the result of mapping the uncertainty in the observations directly into uncertainty in the model. Such bounded-parameter models are a natural basis for designing a controller which must meet performance requirements and control constraints expressed as inequalities. Treating the control-design problem in this way reflects classical control-design practice and allows worst-case design. It also makes possible control synthesis without the assumption of certainty equivalence; this factor is important in achieving robust adaptive control based on imprecise parametric models. Moreover, model uncertainty and control performance can be linked through the medium of bounds to open up an approach to dual control.

The next section starts with a reminder of the basic ideas of parameter bounding. Some limitations of traditional self-tuners are then reviewed briefly, to motivate parameter-bound-based self-tuning control. Section 25.3 introduces worst-case control based on parameter bounds. It is followed by a description of predictive bounding control (PBC), which allows on-line bound adaptation and input optimization to exploit any freedom left in the control sequence by the control performance specification. Section 25.5 presents a technique for bounding the parameters of a system undergoing both drift and abrupt changes. Simulation results and conclusions follow.

25.2. PARAMETER BOUNDING AND MOTIVATION FOR ITS USE IN ADAPTIVE CONTROL

Parameter bounding[3,4,9] consists of using observations y_t (assumed scalar) at sampling instants $t = 1, 2, 3,...$ and specified bounds $e_t \in \mathcal{E}_t$ on the output error of the model

$$y_t = f_t(\boldsymbol{\theta}) + e_t \tag{25.1}$$

to infer bounds

$$y_t - f_t(\boldsymbol{\theta}) \in \mathcal{E}_t \tag{25.2}$$

on the unknown model parameters $\boldsymbol{\theta}$. For example, one observation together with plain bounds $/e_t/ \leq \varepsilon$ on error from a model linear in $\boldsymbol{\theta}$ gives a pair of hyperplane bounds on $\boldsymbol{\theta}$. Successive observations yield new bounds which may or may not reduce the size of the feasible parameter set $\mathcal{D}_t$. The essential features of bounding are that there is no reference to an ensemble and there are no preferred values within the feasible set (although a criterion to select a center of the set can be added). The feasible parameter set $\mathcal{D}_t$, which describes all the parameter values consistent with the observations, the model structure and the error bounds, can be the basis for a control computation taking the worst-case plant behavior into account. Bounded-error inputs can be handled by a solution of the errors-in-variables problem.(10) The parameter bounds can also deal with time variation of the plant; bounding of time-varying parameters is discussed by Norton and Mo.(6)

This chapter examines the use of parameter bounding in on-line adaptive control. Existing self-tuning controllers can often perform well with modest levels of disturbance and slowly changing plant dynamics. Minimum-variance self-tuning control(11,12) and its generalizations(13–15) are relatively robust against errors in plant order, but sensitive to some errors in dead-time. Another approach based on closed-loop pole placement(16,17) proves insensitive to dead-time variations but sensitive to model overparametrization. Both approaches can be made to cope with non-minimum-phase models. Robustness against high-frequency noise and unmodeled dynamics is improved by the introduction of observer dynamics and disturbance-rejection filtering, and by careful choice of plant model structure and reference model.

Generalized predictive control (GPC)(18,19) is intended to improve robustness by using a multi-step cost function. It also incorporates integral action (by using controlled autoregressive integrated moving average—CARIMA—models) to prevent steady-state error. GPC requires the specification of cost horizons, a control horizon and input weighting as design parameters, as well as model orders. Special or limiting cases of GPC are dead-beat, GMV and LQG self-tuners and the methods of Peterka,(20) Ydstie(21) and De Keyser and Van Cauwenberghe.(22,23) A detailed discussion of relations of GPC to other methods is given by Clarke and Mohtadi.(19)

A variant, generalized pole-placement control (GPP),(24) is also designed to improve transient response and overall control performance by employing a multi-step cost function. Robust adaptive control has been studied by Middleton *et al.*(25–27) for mixed structured and unstructured uncertainties in models, using relative dead zones for the unstructured uncertainties. They provide a quantitative measure of the unmodeled errors that can be tolerated by the controller.

The well known self-tuning controllers determine the control input from updated estimates of the model parameters. The control synthesis, however, does not make use of estimates of model uncertainty, such as parameter error covariance. There is no way, therefore, to balance short-term performance against longer-term benefit by taking into account the effects of the next control input on model accuracy as well as on short-term output accuracy. In practice, the input may be near-constant for long periods of time, and poor parameter estimates may result from the poor excitation. They may assume values which lead to temporary loss of stability ("bursting"(28)) or poor closed-loop behavior even if some functions of the parameters, e.g., steady-state gain, remain well estimated. Sudden disturbances after quiet periods may elicit insufficient or excessive response because the parameter-estimate updating gain derives from parameter-error covariance estimates which have become unrealistic.

There are at least three ways to prevent such problems. One method is to add an external, persistently exciting signal to the control input.(29–32) On-line input optimization with this aim in a bounding context is described in Section 25.4. A second way is to improve the correction gain of the parameter estimator. This amounts to proper tuning of the underlying model of the noise and plant changes, as described for parametric bounding in Section 25.5. A final possibility is to arrange for the control computation to be conditioned by the uncertainty in the parameters, by making the performance criterion sensitive to the quality of the model. This is done by the worst-case control design of Sections 25.3 and 25.4.

Statistical or least-squares counterparts of most of the techniques suggested below are readily envisaged. Alternatives differing in detail but still based on parameter bounds are also easily devised and may be better matched to particular circumstances. Not all the features discussed are necessary in every application. The point of the chapter is simply to demonstrate what deterministic parametric bounding and worst-case control design offer: a conceptually simple framework for self-tuning control to accommodate a range of practical features.

25.3. WORST-CASE CONTROL BY PARAMETER BOUNDING(33–35)

If we are to apply control inputs which take into account the uncertainty in the model, the quality of the assessment of parameter uncertainty is crucial.(33–35) When the noise and parameter changes can be modeled as the result of linear filtering of white noise with known covariance, the parameters can be treated as state variables. State estimation then provides parameter estimates and the estimated parameter-error covariance. However, if uncertainty is treated statistically, only average behavior can be guaranteed and there remains a possibility of poor closed-loop performance. The parameter-bounding techniques now available provide the means to guarantee control performance deterministically (subject, of course, to validity of the deterministic assumptions about the noise, disturbances and initial uncertainty).

At each sampling instant t the worst-case control (WCC) scheme updates a feasibility (uncertainty) set $\mathcal{D}_t$ for the plant-model parameters. All values within this set give model-output errors within the specified bounds. The set $\mathcal{D}_t$ is the intersection of the m pairs of half-spaces yielded by the upper and lower bounds on model-output error at the most recent m input-output sampling instants. (In the scheme described in Section 25.5, m varies according to the rate of variation of the plant). At time t, a fixed-length control sequence is then computed that is optimal in the worst case over all plant parameter values within $\mathcal{D}_t$, and all future noise and disturbance values within their specified bounds.

Consider the autoregressive moving average exogenous (ARMAX) model

$$y_{t+k} = -\sum_{i=1}^{p} a_i y_{t-i+k} + \sum_{i=1}^{q} b_i u_{t-i+k} + e_{t+k},\ e_{t+k} \in \mathcal{E} \tag{25.3}$$

where a_i and b_i are unknown parameters, possibly time varying but modeled as constant over m sampling intervals, and $\mathcal{E} \equiv [-\delta,\delta]$ with δ specified. The smallest possible value d of the dead time is specified on physical considerations; d can be taken as one by default, as it is here and henceforth for simplicity. An output-error model might be proposed instead, on the grounds that realistic output-error bounds are easier to specify than equation-error bounds. However, to retain linearity in the parameters, model Eq. (25.3) is preferred. (Problems raised by the use of equation-error bounds in parameter bounding by ellipsoids have been examined by Norton).[9] If Eq. (25.3) is valid throughout $1 \le k \le n$, then

$$y_{t+2} = -a_1(-\sum_{i=1}^{p} a_i y_{t-i+1} + \sum_{i=1}^{q} b_i u_{t-i+1} + e_{t+1}) - \sum_{i=2}^{p} a_i y_{t-i+2} + \sum_{i=1}^{q} b_i u_{t-i+2} + e_{t+2}$$

$$= \sum_{i=1}^{p} \alpha_i(2) y_{t-i+1} + \sum_{i=1}^{q} \beta_i(2) u_{t-i+1} + b_1 u_{t+1} + a_1 e_{t+1} + e_{t+2}, \tag{25.4}$$

and so on, to give prediction equations

$$y_{t+k} = \sum_{i=1}^{p} \alpha_i(k) y_{t-i+1} + \sum_{i=1}^{q} \beta_i(k) u_{t-i+1} + \sum_{i=1}^{k-1} \beta_1(i) u_{t+k-i}$$

$$+ \sum_{i=1}^{k-1} \alpha_1(i) e_{t+k-i} + e_{t+k},\ k = 1, \ldots, n \ with\ e_{t+i} \in \mathcal{E},\ 1 \le i \le k. \tag{25.5}$$

The controller is required to minimize a performance index by computing control inputs $u_t^*, u_{t+1}^*, \ldots,$ and u_{t+N-1}^* on the basis of the past inputs and outputs $u_{t-1}, u_{t-2}, \ldots, y_t, y_{t-1}, \ldots,$ for the worst case within an updated feasibility set $\mathcal{D}_t$ for the parameters $\boldsymbol{\theta} = [a_1\ a_2\ \ldots\ a_p\ b_1\ \ldots\ b_q]^T$. Optimizing a sequence of inputs accounts for the indirect influence of u_t on later outputs via its effect on the predicted optimal $u_{t+1}^*, u_{t+2}^*, \ldots, u_{t+N-1}^*$, as well as its direct effect. The idea of optimizing over a finite number of control samples at each update appears in several established adaptive control schemes, e.g., GPC self-tuning, adaptive LQG control[36,37] and model-predictive control.[38]

Denote by $\{y_t^*\}$ the sequence of set points. The control cost function for time $t+k$, computed at time t, is

$$C_k(t) = \sup_{\theta \in \mathcal{D}_t,\, e_{t+i} \in \mathcal{E},\; i=1\ldots k} \{\max(|y^*_{t+k} - y_{t+k}|, |\lambda u_{t+k-1}|)\} \tag{25.6}$$

where λ is a weight to penalize large inputs. The control sequence may also (or alternatively) be confined to a set $\mathcal{U}$, e.g., $/u_{t+i}/\leq \gamma$ at each time $t+i$ to incorporate actuator constraints. The optimal control sequence computed at time t is

$$\{u^*_t(t), \ldots, u^*_{t+N-1}(t)\} = \arg \inf_{u_t, u_{t+1}, \ldots, u_{t+N-1}} \max_{k=d\ldots n} C_k(t) \tag{25.7}$$

of which $u^*_t(t)$ is applied to the plant and the rest of the control sequence is discarded. Such a finite-horizon, worst-case-optimal controller is guaranteed to keep outputs y_{t+k}, $d \leq k \leq n$, within a bounded but unspecified region about the corresponding set-points, as shown in the next section.

In the predictor equations, y_{t+k} is linear in e_{t+i}, $1 \leq i \leq k$, and multinomial in the parameters $\boldsymbol{\theta}$ of the original model. Since the es and $\boldsymbol{\theta}$ are within convex polytopes $\mathcal{E}$ and $\mathcal{D}_t$, the calculation of $C_k(t)$ is the optimization of a multinomial over a convex polytope. The total degree of the multinomial is equal to the time horizon N over which the input is optimized.

The control law defined by Eq. (25.7) will be called explicit worst-case control. In implicit *WCC* considered below, bounds are computed for the parameters in the predictor equations rather than the parameters of the original model. There are significant differences between the two identification problems. First, the implied range of times over which δ and the bounds making up $\mathcal{D}_t$ remain valid is larger for the prediction model. If the model is identified over m steps, then used at time t, it has to be valid for times $t-m+1$ to $t+n$ on the left-hand side; the corresponding input and output samples on the right-hand side range over a longer period for the prediction model. Second, since the predictor equations contain error samples e_{t+k-i} which appear for more than one sample instant $t+k$, the bounds of the $(p+q+2k-2)$-dimensional feasibility set $\mathcal{P}_k(t)$ of the k-step predictor parameters $\boldsymbol{\alpha}(k)$ and $\boldsymbol{\beta}(k)$ are non-linear (reflecting the non-linear relationship between the predictor coefficients and those of the original model), much as in the errors-in-variables problem.[10,39] Recursive updating of polytopes or ellipsoids in that situation is discussed in the references cited.

The control law based on the predictor parameters is

$$\{u^*_t(t), \ldots, u^*_{t+N-1}(t)\} = \arg \inf_{u_t, u_{t+1}, \ldots, u_{t+N-1}} \max_{k=d\ldots n} C^p_k(t) \tag{25.8}$$

where

$$C^p_k(t) = \sup_{\theta^p_k \in \mathcal{P}_k(t),\, e_{t+i} \in \mathcal{E},\; i=1\ldots k} \{\max(|y^*_{t+k} - \hat{y}_{t+k}|, |\lambda u_{t+k-1}|)\},$$

and

$$\hat{y}_{t+k} = \boldsymbol{\phi}_{tk}{}^{\mathrm{T}}\boldsymbol{\theta}_k^p$$

with

$$\boldsymbol{\theta}_k^p \equiv [\alpha_1(k) \ldots \alpha_p(k)\ \ \beta_1(1)\ \ldots \beta_1(k-1)\ \ldots\ \beta_1(k) \ldots \beta_q(k)\, \alpha_1(1) \ldots\ \alpha_1(k-1)]^{\mathrm{T}}$$

confined to its feasibility set $\mathcal{P}_k$ and

$$\boldsymbol{\phi}_{tk} \equiv [\,y_t \cdots y_{t-p+1}\ \ u_{t+k-1} \cdots u_{t-q+1}\ e_{t+k-2} \cdots e_t]^{\mathrm{T}}$$

If the parameters are confined to a polytope, calculation of the optimal input sequence in Eq. (25.8) is reasonably tractable, as shown by the following lemma.

LEMMA 25.1. With $\mathcal{V}_k$ the set of vertices of convex polytope $\mathcal{P}_k(t)$,

$$C_k^p(t) = \max_{\boldsymbol{\theta}_k^p \in \mathcal{V}_k} \{\max(\bar{\boldsymbol{\phi}}_{tk}(1)^{\mathrm{T}}\ \boldsymbol{\theta}_k^p - y^*_{t+k}, y^*_{t+k}\,\bar{\boldsymbol{\phi}}_{tk} - (-1)^{\mathrm{T}}\boldsymbol{\theta}_k^p, |\,\lambda u_{t+k-1}\,|\,)\}$$

where

$$\bar{\boldsymbol{\phi}}_{tk}(1) = [\,y_t \cdots y_{t-p+1}\ \ u_{t+k-1} \cdots u_{t-q+1}\ \bar{e}_1 \cdots \bar{e}_{k-1}]^{\mathrm{T}},$$

$$\bar{\boldsymbol{\phi}}_{tk}(-1) = [\,y_t \cdots y_{t-p+1}\ \ u_{t+k-1} \cdots u_{t-q+1} - \bar{e}_1 \cdots - \bar{e}_{k-1}]^{\mathrm{T}}$$

and

$$\bar{\mathrm{e}}_{\mathrm{i}} = \delta\ \mathrm{sign}(\theta^p_{p+q+k+i-1}).$$

PROOF: From the definition,

$$C_k^p(t) = \sup_{\boldsymbol{\theta}_k^p \in \mathcal{P}_k(t),\ e_{t+i} \in \mathcal{E},\ i=1\ldots k} \{\max(\,|\,y^*_{t+k} - \hat{y}_{t+k}\,|\,, |\,\lambda u_{t+k-1}\,|\,)\}$$

$$= \sup_{\boldsymbol{\theta}_k^p \in \mathcal{P}_k(t)}\ \sup_{e_{t+i} \in \mathcal{E},\ i=1\ldots k} \{\max(\,y^*_{t+k} - \hat{y}_{t+k}, \hat{y}_{t+k} - y^*_{t+k},\ |\,\lambda u_{t+k-1}\,|\,)\}$$

and it is easy to see that

$$\sup_{e_{t+i} \in \mathcal{E},\ i=1\ldots k} \{\,\hat{y}_{t+k} - y^*_{t+k}\} = \bar{\boldsymbol{\phi}}_{tk}(1)^{\mathrm{T}}\boldsymbol{\theta}_k^p - y^*_{t+k},$$

$$\sup_{e_{t+i} \in \mathcal{E},\ i=1\ldots k} \{\,y^*_{t+k} - \hat{y}_{t+k}\} = y^*_{t+k} - \bar{\boldsymbol{\phi}}_{tk}(-1)^{\mathrm{T}}\boldsymbol{\theta}_k^p.$$

Within any one orthant O_j in $\boldsymbol{\theta}$-space, $\bar{\boldsymbol{\phi}}_{tk}(1)$ and $\bar{\boldsymbol{\phi}}_{tk}(-1)$ are fixed. The subset of convex polytope $\mathcal{P}_k$ in O_j is itself a convex polytope, with vertex set $\mathcal{V}_{kj}$ say. A linear form over a convex polytope can have generic extreme values only at the vertices of the polytope, so

$$\sup_{\boldsymbol{\theta}_k^p \in O_j} \{s(\bar{\boldsymbol{\Phi}}_{tk}(s)^{\mathrm{T}}\boldsymbol{\theta}_k^p - y^*_{t+k})\} = \max_{\boldsymbol{\theta}_k^p \in \mathcal{V}_{kj}} \{s(\bar{\boldsymbol{\Phi}}_{tk}(s)^{\mathrm{T}}\boldsymbol{\theta}_k^p - y^*_{t+k})\},\ s = -1, 1$$

Now the supremum over the union of two adjoining orthants cannot be at a point of their common boundary unless that point is also a vertex of the union, since the definition of $\bar{\boldsymbol{\Phi}}_{tk}(1)$ and $\bar{\boldsymbol{\Phi}}_{tk}(-1)$ ensures monotonicity of all terms as the boundary is crossed. The supremum over the union is therefore at a point in the union's vertex set, and by induction the supremum over the whole polytope $\mathcal{P}_k(t)$ is in $\mathcal{V}_k$. □

Since $\bar{\boldsymbol{\Phi}}_{tk}(1)^{\mathrm{T}}\boldsymbol{\theta}_k^p - y^*_{t+k}$ and $y^*_{t+k} - \bar{\boldsymbol{\Phi}}_{tk}(-1)^{\mathrm{T}}\boldsymbol{\theta}_k^p$ are linear forms in $\bar{\mathbf{u}}_t \equiv [u_t\ u_{t+1}\ \dots\ u_{t+N-1}]^{\mathrm{T}}$, $C_k^p(t)$ is piecewise linear over $\mathcal{U}$ and its maximum over $k = d, \dots, n$ can be represented as the maximum of a finite number of linear forms:

$$\hat{C} = \max_{k=1,\dots,n} C_k^p(t) = \max_{i \in \mathcal{J}} L_i(\bar{\mathbf{u}}_t) \tag{25.9}$$

Consider the space of $(\bar{\mathbf{u}}_t, \hat{C})$, in which each linear form $L_i(\bar{\mathbf{u}}_t)$ is a hyperplane. Together with the bounds of $\mathcal{U}$, $\hat{C}(\bar{\mathbf{u}}_t)$ defines a polytope, unbounded upwards and convex downwards in the $\hat{C}$ direction. The polytope may be computed by established polytope-updating algorithms.[6,7] The minimization with respect to $\bar{\mathbf{u}}_t$ amounts to finding the polytope vertex with smallest $\hat{C}$.

The WCC scheme just described provides the rudiments of a controller. Next some other aspects of the controller are considered.

25.4. PREDICTIVE BOUNDING CONTROLLER

This section extends the basic worst-case control scheme to provide for time-varying plant and to allow short-term control performance to be balanced against identification accuracy when finding the control sequence.

A scheme for tracking time-varying plant dynamics, described in detail in Section 25.5, updates the equation-error bound δ and a scalar ρ characterizing the largest possible time variation of $\boldsymbol{\theta}$. The scheme yields a set of acceptable (feasible) values of (δ,ρ) at each update. The worst-case control performance depends on (δ_t,ρ_t) through the set $\mathcal{P}_k(t,\delta_t,\rho_t)$ of feasible k-step predictor-parameter values. Thus one can choose (δ_t,ρ_t) from its feasible set according to its effect on worst-case setpoint-following error.

LEMMA 25.2. A control law exists yielding setpoint-following error bounded by

$$|y^*_{t+k} - y_{t+k}| \le \inf_{u_t,u_{t+1},\dots,u_{t+k-1}} \{ \sup_{\substack{\boldsymbol{\theta}_k^p \in \mathcal{P}_k(t,\delta_t,\rho_t) \\ e_{t+i} \in \mathcal{E}, 0 \le i \le k-2}} |y^*_{t+k} - \boldsymbol{\phi}_{tk}^{\mathrm{T}}\boldsymbol{\theta}_k^p| \} + \delta_t,$$

$k = 1, \dots, N$, where $\{y^*\}$ is the set-point sequence and (δ_t,ρ_t) is within its feasible set.

The lemma follows directly from the definition of $\mathcal{P}_k(t,\delta_t,\rho_t)$ and the assumption that it contains at least one value of $\boldsymbol{\theta}_k^p$ satisfying the prediction equation with the assumed bounds on equation error. Treating δ_t and ρ_t as parameters for the moment, the lemma suggests the control

$$u_t^*(\delta_t,\rho_t) = \arg \inf_{u_t \in \mathcal{U}_t} L_t(u_t,\delta_t,\rho_t) \tag{25.10}$$

where

$$L_t(u_t,\delta_t,\rho_t) = \inf_{u_{t+1},\ldots,u_{t+N-1}} \quad \max_{k=1,2,\ldots,} \quad \sup_{\substack{\boldsymbol{\theta}_k^p \in \mathcal{P}_k(t,\delta_t,\rho_t) \\ e_{t+i} \in \mathcal{E}, 0 \le i \le k-2}} \quad | y_{t+k}^* - \boldsymbol{\phi}_{tk}^{\mathrm{T}} \boldsymbol{\theta}_k^p |$$

The set $\mathcal{U}_t$ of permissible control values is discussed later. For each δ_t in the range $[\delta_{t\min},\delta_{t\max}]$ of feasible values, a minimum feasible $\rho_{\min}(\delta_t)$ gives the smallest parameter variation. PBC uses that $(\delta_t,\rho_{\min}(\delta_t))$ which gives the lowest worst-case setpoint-following error $L_t(u_t,\delta_t,\rho_{\min}(\delta_t))$:

$$u_t^* = \arg \inf_{u_t \in \mathcal{U}_t} L_t(u_t,\delta_t^*,\rho_{t\min}^*)$$

where

$$\delta_t^* \equiv \arg\inf_{\delta_t \in [\delta_{t\min},\delta_{t\max}]} \quad \inf_{u_t \in \mathcal{U}_t} L_t(u_t,\delta_t,\rho_{\min}(\delta_t))$$

and ρ_{tmin}^* is $\rho_{\min}(\delta_t^*)$. In practice, a finite number of pairs $(\delta_t,\rho_{\min}(\delta_t))$ is examined instead of the entire set, and the definition of δ_t^* correspondingly modified.

Next consider the determination of the permissible-control set $\mathcal{U}_t$. Procedures for optimizing the excitation properties of the control sequence (on-line experiment design), and to guarantee closed-loop stability will be described.

As so far defined, PBC recognizes the effects of future control inputs on the output but not their effects on future feasible-parameter sets $\mathcal{P}_k(t+i,\delta,\rho)$ and thence on $L_{t+i}(u_{t+i},\delta,\rho)$. A controller with more foresight can be obtained by relaxing the requirement that $\{u_t, u_{t+1}, \ldots, u_{t+N-1}\}$ should minimize $L_t(u_t,\delta_t,\rho_t)$ and using the resulting freedom in u_t to tighten future parameter bounds. In Section 25.3, the worst-case-optimal control input was computed by searching the vertices of a polytope $\mathcal{U} \cap \bigcap_{i \in \mathcal{J}} \mathcal{L}_i$ in the space of $(\bar{\mathbf{u}}(t),\hat{C})$, where $\bar{\mathbf{u}}(t) \equiv [u_t \; u_{t+1} \; \ldots \; u_{t+N-1}]^{\mathrm{T}}$, $\mathcal{U}$ is the permissible-control set, $\mathcal{L}_i$ is the half space defined by the linear form $L_i(\bar{\mathbf{u}}(t))$, and $\hat{C}$ is the worst-case output error. The best control sequence is at the vertex giving the smallest value $\hat{C}^*$ of $\hat{C}$. For fixed δ and ρ, the same procedure applies to PBC.

If now the worst-case output error is allowed to be a little larger, say $\hat{C} < \hat{C}^* + \varepsilon$ (with ε fixed by the largest acceptable output error), a range of control-sequence values meets this relaxed error requirement. The one with the best excitation properties is found by optimizing over the polytope

$$\mathcal{R}(\varepsilon) = \{(\bar{\mathbf{u}}(t), \hat{C}) \in \mathcal{U} \cap \bigcap_{i \in \mathcal{J}} \mathcal{L}_i \mid \hat{C} < \hat{C}^* + \varepsilon\}.$$

One possible excitation criterion is to make the predictor vectors $\phi_{t,1}$, $\phi_{t-1,1}$, ..., $\phi_{t-p+1,1}$ as nearly orthogonal as possible. For instance,

$$D \equiv \det[\phi_{t,1}\ \phi_{t-1,1} \cdots \phi_{t-p+1,1}]$$

determines the volume of the parallelepiped formed by the 1-step-predictor-parameter bounds found from the prediction errors at times t, $t-1$, ..., $t-p+1$. As D depends linearly on u_t,

$$u_t(\varepsilon) = \arg\max_{\mathcal{U}_t} \{D \mid (\bar{\mathbf{u}}(t), \hat{C}) \in \mathcal{R}(\varepsilon)\} \tag{25.11}$$

requires only a search of the vertices of $\mathcal{R}(\varepsilon)$. This minimization of D is for fixed (δ,ρ). By repeating it, the optimal (δ,ρ) can be found.

An alternative on-line experiment-design technique, again considering the parameter-bounding parallelepiped due to p successive predictor updates, is to make the angles between the vectors $\phi_{t-i,1}$ larger than a lower bound α while keeping their norms above a lower bound μ. Denote by $\mathcal{W}_t$ the set of values of input u_t such that

$$\min_{i=1\ldots p-1} \{Ang(\phi_{t,1}, \phi_{t-i,1})\} \geq \alpha$$

$$\text{and } \|\phi_{t-i,1}\| \geq \mu.$$

If $u_t \in \mathcal{W}_t\ \forall\ t$, no two out of p successive regressor vectors are ever separated by less than α and the distance $2\delta / \|\phi\|$ between any two opposite faces of the parallelepiped is not more than $2\delta/\mu$. The first part of Appendix B briefly discusses the effect of the angles and norms of the vectors $\phi_{t-i,1}$ on the maximum diameter of the feasible parameter set.

The control input found in this way balances control performance with effective excitation, in partial dual control. (Dual control in a stochastic setting is discussed, e.g., by Aström and Wittenmark).[40] Certainty equivalence is replaced by explicit consideration of present and future uncertainty in the model. In the on-line experiment design procedures described above, ε, α and μ provide design parameters to compromise between immediate control performance and the ultimate effect of reduced model uncertainty.

PBC measures performance by setpoint-tracking error, but the control input can also be bounded to guarantee closed-loop stability sample by sample, as follows. In conventional pole-placement self-tuning control,[17,40] any non-minimum-phase factor in the numerator $B(q^{-1})$ of the plant model must be retained in the numerator of the reference model specifying the desired closed-loop transfer function. This difficulty can be sidestepped in a bounding context. Compute $\tilde{u}_t$ such that $B(q^{-1};\boldsymbol{\theta})\tilde{u}_t$ is close to y_t^* for all feasible plant-parameter values $\boldsymbol{\theta}$ and restrict the control law to the form

$$R_t(q^{-1})u_t = \tilde{u}_t - S_t(q^{-1})y_t \tag{25.12}$$

If this is done, the closed-loop behavior is given by

$$(A(q^{-1};\boldsymbol{\theta})R_t(q^{-1}) + B(q^{-1};\boldsymbol{\theta})S_t(q^{-1}))y_t = B(q^{-1};\boldsymbol{\theta})\tilde{u}_t + R_t(q^{-1})e_t \tag{25.13}$$

where the closed-loop characteristic polynomial

$$P_t(q^{-1}) = A(q^{-1};\boldsymbol{\theta})R_t(q^{-1}) + B(q^{-1};\boldsymbol{\theta})S_t(q^{-1})$$

can be bounded to be close to $(1 + 0q^{-1} + 0q^{-2} \ldots)$. With $B(q^{-1};\boldsymbol{\theta}^1)$ $\tilde{u}_t$ close to the setpoint y_t^* for all feasible parameter values, the closed-loop transfer function is acceptable for any plant parameters within the feasible set. At time t, a sequence of $\tilde{u}$s is obtained by minimax optimization:

$$\{\tilde{u}_{t,t}, \tilde{u}_{t+1,t}, \ldots, \tilde{u}_{t+N_b,t}\} = \arg\min \sup_{\boldsymbol{\theta} \in \mathcal{D}_t} \max_{k=0,1,\ldots,N_b} | y^*_{t+k+1} - B(q^{-1};\boldsymbol{\theta})q^{-1}\tilde{u}_{t+k,t} |$$

and $\tilde{u}_t$ is set to $\tilde{u}_{t,t}$. The stabilizing horizon N_b is specified *a priori*. (The choice of N_b is not crucial; simulations show that it is practical to choose N_b greater than the prediction horizon n). To simplify computation of the feasible set of the coefficients of $R_t(q^{-1})$ and $S_t(q^{-1})$, $P_t(q^{-1})$ is confined to a simplex Δ within the stability region. Appendix A shows that if $\mathcal{D}_t$ is a polytope, the feasible set of these coefficients is a polytope.

The set $\mathcal{U}_t^s$ of stabilizing control inputs u_t is then obtained by applying control law Eq. (25.12) over all feasible values of $R_t(q^{-1})$ and $S_t(q^{-1})$. Since the feasible set of $R_t(q^{-1})$ and $S_t(q^{-1})$ is a convex polytope, the stabilizing input set $\mathcal{U}_t^s$ is an interval on the real axis. Overall, the feasible-control set $\mathcal{U}_t$ is $\mathcal{W}_t \cap \mathcal{U}_t^s$.

25.5. ON-LINE BOUNDING OF TIME-VARYING PARAMETERS

A technique to allow for time variation of the system, based on computation of parameter bounds, tunes the model-output error characterization and plant-variation model jointly. Changes can be detected straightforwardly from model-parameter bounds, since a significant change causing the current plant model to become invalid shows as a clash between new and existing bounds (making the updated $\mathcal{D}_t$ empty). The criterion for tuning is the frequency of such changes; the items tuned are scalars δ defining the bounds on model-output error and ρ defining bounds on sample-to-sample variation in the model parameters.

Good tracking relies on the estimator employing suitable prior assumptions on possible parameter changes. Ljung and Gunnarsson(41) survey identification methods for time-varying systems. The most common approach, recursive prediction-error estimation with a scalar forgetting factor, can be far from optimal.(42) Better parameter tracking can be achieved in suitable circumstances by treating the parameters as state variables,(43–48) but only if a state-space model for parameter changes is appropriate and the covariances characterizing the parameter changes and observation errors (including systematic modeling error) are reliable. In practice the covariances usually have to be tuned empirically. Moreover, a simple statistical parameter-change model may be unable to represent abrupt or systematic changes of the plant adequately. This is particularly so when the model has to be simple and the "noise" covers plant dynamics omitted by the model, with significant but unknown structure.

The bounded-error predictor Eq. (25.5) with constant equation-error bound δ gives adequate computed predictor-parameter bounds if the system is time-invariant or varies little enough for a suitably inflated value of δ to give usable parameter bounds. Otherwise, explicit provision for time variation is necessary in computing the predictor-parameter bounds. The worst-case control performance for given future control-input values depends on the predictor-parameter bounds and the prediction-equation error bounds. An adaptive worst-case controller, therefore, must take the time variation of both into account.

One option is to lump the effects of parameter variation on the output into a time-varying equation error, varying δ accordingly. Empirical adjustment of the equation-error bounds could then be based on either the behavior of the parameter bounds or the actual model-output error. (Similar comments can be made of conventional statistical estimators, substituting "covariance" for "bounds"). However, one would expect better performance if parameter variation were distinguished from equation-error variation by making realistic assumptions about how the parameters vary. Adoption of some parameter-variation model is then necessary. The remainder of this section discusses possibilities and describes a scheme which has been found workable in simulations.

Denote by $\mathcal{L}_k(t,\delta)$ the k-step-predictor parameter set (bounded by two hyperplanes) obtained by substituting into the prediction equation for y_t values of plant input and output known at time t, and using a prediction-equation error bound δ:

$$\mathcal{L}_k(t,\delta) = \{\boldsymbol{\theta}_k^p \mid y_t - \bar{\boldsymbol{\Phi}}_{t-k,k}(1)^{\mathrm{T}}\, \boldsymbol{\theta}_k^p \leq \delta,\ \bar{\boldsymbol{\Phi}}_{t-k,k}(-1)^{\mathrm{T}}\, \boldsymbol{\theta}_k^p - y_t \leq \delta\}$$

Possible ways of allowing for parameter variation,(49) all computing parameter bounds from a fixed number m of successive input-output values, include the following.

(i) Assuming that changes in the predictor parameters between times $t-k-m+1$ and t are negligible, and using the bounded set

$$\mathcal{P}_k(t,\delta) \equiv \bigcap_{i=0}^{m-1} \mathcal{L}_k(t-i,\delta)$$

for prediction from time t. This technique has the virtue of simplicity and may be satisfactory for slowly varying systems, but is unsuited to rapidly varying systems.

(ii) Loosening past prediction-error bounds by inflating the equation-error bounds exponentially, yields the feasible predictor-parameter sets

$$\mathcal{L}_k(t-i,\delta\rho^i) = \{\boldsymbol{\theta}_k^p \mid y_{t-i} - \boldsymbol{\Phi}_{t-i-k,k}(1)^{\mathrm{T}}\, \boldsymbol{\theta}_k^p \le \rho^i\delta,$$

$$\boldsymbol{\Phi}_{t-i-k,k}(-1)^{\mathrm{T}}\, \boldsymbol{\theta}_k^p - y_{t-i} \le \rho^i\delta\}, \quad 0 \le i \le m-1$$

with $\rho > 1$. Information older than m steps is discarded for computational economy, incurring little loss if m is suitably chosen. The parameters $\boldsymbol{\theta}_k^p(t)$ of the k-step-ahead predictor for y_{t+k} from time t are then assumed to be in the bounded set

$$\mathcal{P}_k(t,\delta,\rho) \equiv \bigcap_{i=0}^{m-1} \mathcal{L}_k(t-i,\delta\rho^{i+k})$$

The increase in the error bound by a factor ρ^k, to account for the k-step extrapolation of the predictor parameters, can be achieved with no extra computation by using δ larger by that factor when computing $\mathcal{L}_k(t-i,\delta\rho^i)$. A procedure to find acceptable values of δ and ρ is suggested below.

(iii) Assuming that the parameter increments from one sample to the next are confined to a specified bounded convex set: $\boldsymbol{\theta}_k^p(t-k-i) - \boldsymbol{\theta}_k^p(t-k-i-1) \in \mathcal{C}^k(t-k-i)$. The predictor-parameter bounds are computed by alternate time updates, vector-summing the existing bounds and $\mathcal{C}^k(t-k-i)$, and observation updates imposing new bounds

$$y_{t-i} - \boldsymbol{\Phi}_{t-i-k,k}(1)^{\mathrm{T}}\, \boldsymbol{\theta}_k^p \le \delta,\ \boldsymbol{\Phi}_{t-i-k,k}(-1)^{\mathrm{T}}\, \boldsymbol{\theta}_k^p - y_{t-i} \le \delta.$$

At time t, the result of applying this procedure over the preceding m steps is

$$\mathcal{Q}_k(t,\delta) \equiv \mathcal{L}_k(t,\delta) \cap [\mathcal{L}_k(t-1,\delta) + \mathcal{C}^k(t-k)] \cap [\mathcal{L}_k(t-2,\delta) + \mathcal{C}^k(t-k-1) + \mathcal{C}^k(t-k)] \ldots$$

$$\ldots \cap [\mathcal{L}_k(t-i,\delta) + \mathcal{C}^k(t-k-i+1) \ldots + \mathcal{C}^k(t-k)] \ldots$$

$$\ldots \cap [\mathcal{L}_k(t-m,\delta) + \mathcal{C}^k(t-m-k+1) \ldots + \mathcal{C}^k(t-k)].$$

Allowing for evolution of $\boldsymbol{\theta}_k^p$ over the k steps from time $t-k$ to time t, the parameters $\boldsymbol{\theta}_k^p(t)$ of the k-step predictor for y_{t+k} are then in the bounded set $\mathcal{P}_k(t,\delta) \equiv \mathcal{Q}_k(t,\delta) + \mathcal{C}^k(t-k+1) \ldots + \mathcal{C}^k(t)$.

In the absence of background knowledge, the increment set may be taken as a constant polytope $\mathcal{C}^k$. Physical insight may suggest a size and shape for $\mathcal{C}^k$, or $\mathcal{C}^k$ may be tuned in a preliminary off-line parameter-bounding identification exercise. The advantage of such a bounded-parameter-increment model is that different proportional rates of change or correlated changes can be handled so long as $\mathcal{C}^k$ can be specified. However, for simplicity the parameter-increment bounds will henceforth be assumed to be constant, $\mathcal{C}^k \equiv \mathcal{C}^k(t)$, and represented as $\mathcal{C}^k = \rho\mathcal{C}$, $\rho > 0$, where $\mathcal{C}$ is an axis-aligned box in parameter space containing the origin. Thus $\mathcal{C}^k$ is a function of the single scalar parameter ρ.

(iv) Model the parameters as jump processes, either staying unchanged or jumping at time t. It is assumed that jumps are not separated by less than $m_0 \geq dim(\boldsymbol{\theta}^p)$ sampling intervals, and that the sequence $\{\bar{\boldsymbol{\Phi}}_{t-i-k,k},\ 1 \leq i \leq m\}$ is linearly independent, so that a severe enough change causes the feasible parameter set to be empty. An obvious alternative, probabilistic specification of the jump characteristics, conflicts with the motivation of bounding.

With these possibilities in mind, tuning of the equation-error bounds and the parameter time-variation model can be discussed. The dynamics of the parameters are taken to be a mixture of slow drift, which can be accommodated by the equation-error bounds, and abrupt but infrequent changes which cannot. Modeling is done by a combination of (iv) and either (ii) or (iii). The basic idea is to adjust δ and ρ according to their effects on the mean time span over which the model remains valid (i.e., $\mathcal{P}$ or $\mathcal{Q}$ is non-empty) in particular records. In other words, the adjustment mechanism examines the model's age.

Definition 25.1. A model with equation-error bound $\delta > 0$ is said to have age

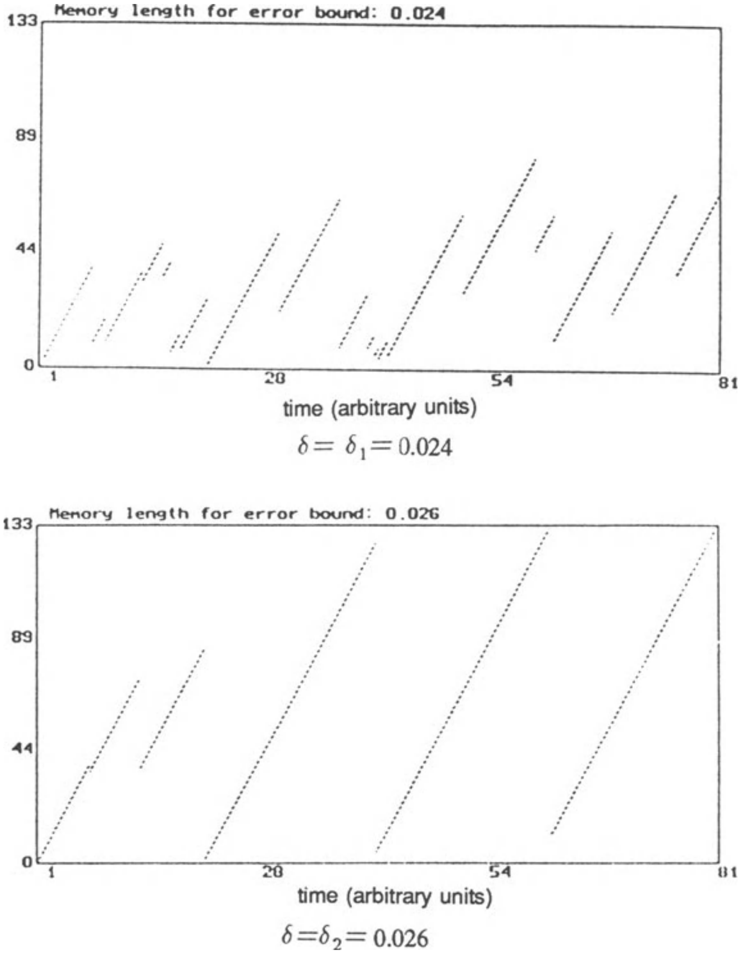
$$m(t,\delta,\rho) = \max\{m > 0 \mid \bigcap_{i=0}^{m-1} \mathcal{L}_{k,t}(t-i,\delta,\rho^i) \neq \emptyset,\ \ 1 \leq k \leq n\}$$

for equation-bound inflating by a factor ρ, or

$$m(t,\delta,\rho) = \max\{m > 0 \mid\ \mathcal{Q}_k(t,\delta) \neq \emptyset,\ \ 1 \leq k \leq n\}$$

for parameter-increment bounding.

Age measures how old the input-output data in the model can become without clashing with the latest data. A clash causes a reduction in age, not necessarily to zero. Typical age behavior over an input-output sequence for fixed δ is shown in Fig. 25.1. A reduction in age ends a generation. Assume that every change in plant dynamics not attributable to drift ends a generation so long as the equation-error bound is within a known range $\delta_{\min} \leq \delta \leq \delta_{\max}$. Appendix B gives conditions under



$\delta = \delta_1 = 0.024$

$\delta = \delta_2 = 0.026$

FIGURE 25.1. Model age for fixed $\delta = \delta_1$ and $\delta = \delta_2$, $\delta_1 < \delta_2$.

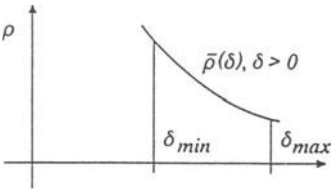


FIGURE 25.2. Least feasible parameter-increment bound as function of equation-error bound for specified number of detected abrupt changes.

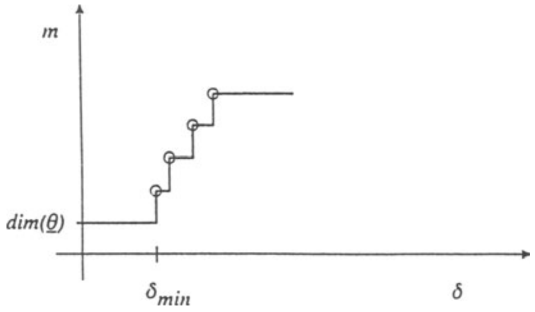


FIGURE 25.3. Variation of age with δ.

which these assumptions are satisfied. The details of such conditions are unimportant and alternative conditions can be developed; their significance is that they allow each abrupt change to be detected by a fall in age.

Assume for the moment that in a given input-output sequence $\{u_t,y_t\}_{t=1,\ldots,P}$, r such changes are known to occur. For a given model structure one can calculate, as a function of δ, the least value $\bar{\rho}(\delta)$ of ρ which makes the number of generations in the given record r. The function $\bar{\rho}(\delta)$ shows the tradeoff between equation-error bounds and parameter-change bounds, as indicated in Fig. 25.2. Clearly, if $\delta_1 \le \delta_2$ then

$$m(t,\delta_1,\rho) \le m(t,\delta_2,\rho) \text{ and } \mathcal{P}_k(t,\delta_1,\rho) \subseteq \mathcal{P}_k(t,\delta_2,\rho)$$

Fig. 25.3 illustrates a typical increase of $m(t,\delta,\rho)$ with δ.

The pair (δ,ρ) is called *feasible* if the number r of generations of $m(t,\delta,\rho)$ in the period considered equals the specified number of abrupt changes. The feasible set of (δ,ρ) consists of all feasible pairs for a given r. The set has extremes δ_{min} and δ_{max}, and has $\bar{\rho}(\delta)$ as its lower boundary for ρ. Having found the feasible set for (δ,ρ) tuning of the time-variation model amounts to choosing δ and ρ. The set can be computed off line or, with sufficient computing power, periodically on line, by determining the number of generations in a given input-output sequence for each of a grid of values of (δ,ρ), as indicated in Appendix C.

25.6. SIMULATIONS

This section compares the performance of the PBC with GPC and pole-placement control (certainty-equivalence controllers based on recursive least-squares estimation). General conclusions cannot be drawn from a few examples, but the simulations will show that PBC has promise. Two examples are considered: a non-minimum-phase system for which it is difficult to obtain acceptable performance by GPC or pole placement, and a time-varying linear system. To measure average performance, long simulations are carried out and density plots for the plant-output errors are examined. Such plots tend to reduce the apparent relative merit of the bounding controller, which considers worst-case rather than average output error.

Example 1: The continuous-time system

$$\begin{aligned} \dot{x}_1(t) &= x_2(t) + u_1(t) \\ \dot{x}_2(t) &= -x_1(t) - 2x_2(t) - 4u_1(t) \\ y(t) &= x_1(t) + 0.04u_2(t) \end{aligned} \tag{25.14}$$

is simulated, where $u_1(t)$ is the control input. The response to step changes in set point are shown in Figs. 25.4–25.6, with input $u_2(t)$ a noise-like disturbance. (Regulation performance was also tested, with $u_2(t)$ a random-step disturbance signal with amplitude $N(0,1)$-distributed; regulation was satisfactory for all the controllers.) The non-minimum-phase transfer function from $u_1(t)$ to $y(t)$ is $(s-2)/(s^2+2s+1)$. With sampling interval 0.4 s, the zero-order-hold-equivalent discrete-time transfer function from $U_1(z^{-1})$ to $Y(z^{-1})$ is

$$(0.1450z^{-1} - 0.3624z^{-2})/(1 - 1.3406z^{-1} + 0.4493z^{-2}).$$

The discrete-time model is of the form

$$y_t + a_1y_{t-1} + a_2y_{t-2} = b_1u_{t-1} + b_2u_{t-2},\ t = 1, 2, 3, \ldots \tag{25.15}$$

and may be identified by least-squares parameter estimation, as the asymptotic information matrix is non-singular. For GPC and pole-placement control, the model parameters are estimated by recursive least-squares with a forgetting factor. Control-input constraints $/u_t/ \leq 100$ are applied in all cases.

The specifications of the controllers, with design parameters chosen carefully, are as follows. For GPC:

- minimum, maximum prediction horizons $N_1 = 1$, $N_2 = 4$, control horizon $N_c = 2$.
- integral action by introducing difference operator into model:

  $$A(q)y_t = B(q)(q-1)u_t + (q-1)e_t$$

  where

  $$A(q) = q^2 + a_1q + a_2, \quad B(q) = b_1q + b_2.$$

- control weight 0.01

- forgetting factor in RLS 1.0.

For the pole-placement controller:

- observer polynomial $A_o(q) = q^2 - 0.02q$
- closed-loop characteristic polynomial $P(q) = q^2 - 0.05q + 1$
- numerator of model $B(q) = b_1q + b_2$, denominator $A(q) = q^2 + a_1q + a_2$.
- estimate $\hat{B}$ used as described by Aström and Wittenmark,[17] discriminating between minimum-phase and non-minimum-phase roots.
- forgetting factor in RLS 1.0.

For PBC:

- prediction horizon $n = 1$
- at each sampling instant t, a polytope feasible parameter set $\mathcal{D}_t$ valid for a short time up to time t is obtained by intersecting the hyperplane parameter bounds for the six immediately preceding sampling instants with the large box

$$\mathcal{B} = \prod_{i=1}^{5} [-10,10].$$

- for robust stability the closed-loop poles are restricted by bounds[51]

$$-d_1 + d_2 + 3d_3 \leq 0.15$$

$$d_1 + d_2 - 3d_3 \leq 0.15$$

$$d_1 - d_2 + \ d_3 \leq 0.05$$

$$-d_1 - d_2 - 3d_3 \leq 0.05$$

on the coefficients of the characteristic polynomial

$$q^3 + d_1q^2 + d_2q + d_3 \equiv (q^2 + a_1q + a_2)(q + r_1) + (b_1q + b_2)(s_1q + s_2),$$

defining a simplex $\mathcal{L}$ say. The control u_t is then constrained by

$$u_t \in \{\tilde{u}_t - r_1u_{t-1} - s_1y_1 - s_2y_{t-1} \mid (d_1,d_2,d_3) \in \mathcal{L}\}$$

where $\tilde{u}_i$ is defined as in Section 25.4.
- stabilizing horizon of PBC, $N_b = 8$.

Figs. 25.4 to 25.6 show runs for GPC, pole placement and PBC, after initial transients have subsided. In the figures, the square-wave set-point sequence and output sequence are superimposed; the control input and disturbance sequences are shown separately. Anticipatory responses to set-point changes are shown by GPC

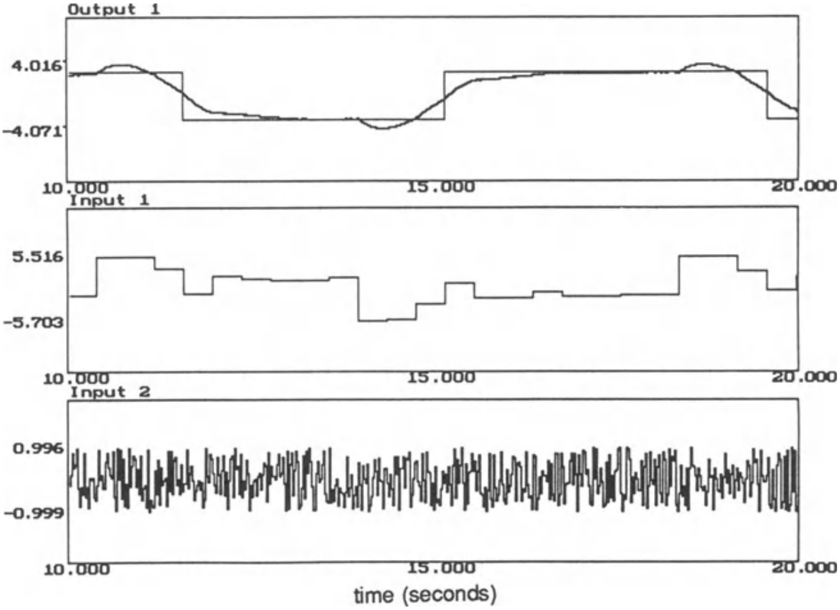


FIGURE 25.4. Set-point and controlled output, control input and disturbance signals for GPC.

and PBC as knowledge of impending changes is exploited; if it is not, the comparison is little changed. Figure 25.7 gives sample means and densities of the output deviation $y^*(t) - y(t)$ from the setpoint in 500-second runs with the u_2 random step disturbance described earlier. The action of PBC, confining the closed-loop poles to acceptable locations then minimizing the worst-case output error with respect to the remaining control freedom, is seen to yield good set-point tracking.

Example 2: A continuous-time system with transfer function $10/(s^2 + a_1 s + 9)$ from $u_1(t)$ to $y(t)$ is simulated, with controller sampling interval 1 s. Parameter a_1 is varied as $6 + \sin(2\pi t/16)$ (as in Fig. 25.8) and then with a step change (as in Fig. 25.9). For realism, a white disturbance is also added as process noise $u_2(t)$ in

$$\dot{x}_1(t) = -9x_1(t) - a_1 x_2(t) + 10u_1(t)$$

$$\dot{x}_2(t) = x_1(t) + Du_2(t)$$

$$y(t) = x_2(t),$$

with $D = 0.04$ and $u_2(t)$ uniformly distributed in $[-1, 1]$.

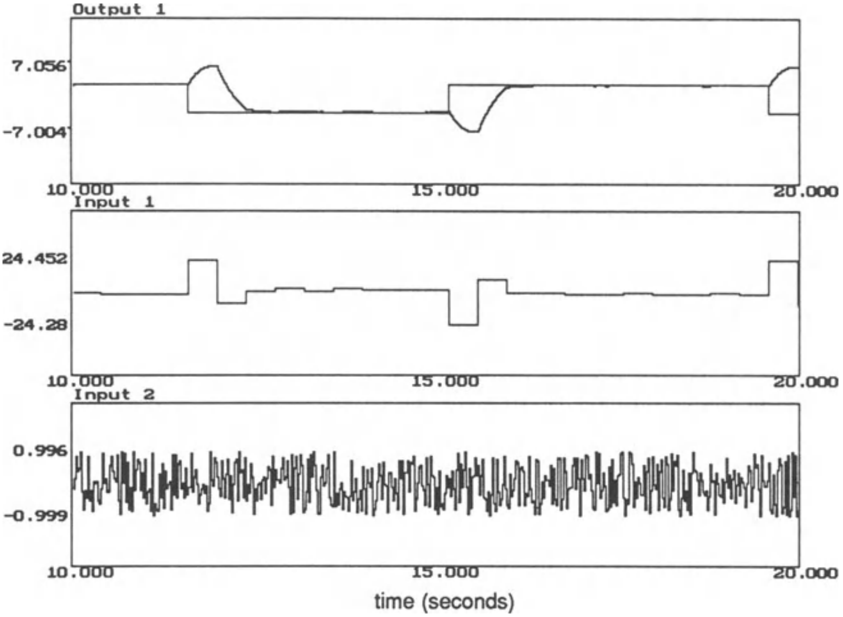


FIGURE 25.5. Set-point and controlled output, control input, and disturbance signals for pole placement.

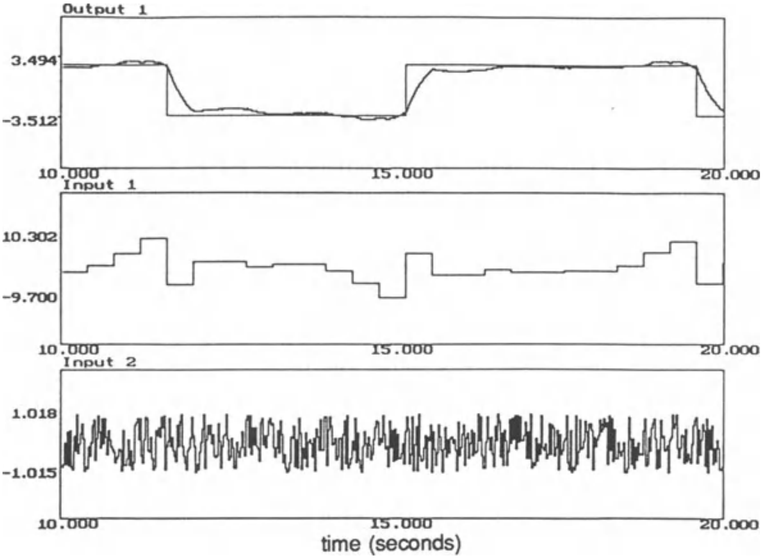


FIGURE 25.6. Set-point and controlled output, control input, and disturbance signals for PBC.

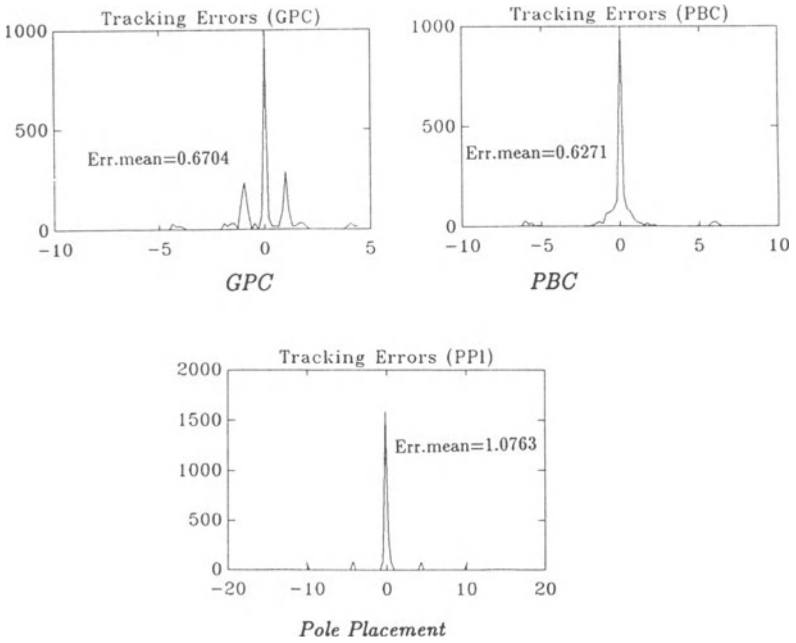


FIGURE 25.7. Sample means and density functions of plant-output error in 500 s runs.

For conciseness, the following figures show responses to set-point changes, noise-like disturbances and the parameter variation; the transient responses are nonetheless readily distinguished.

The model is of the same form as in Example 1 and the same control constraints are applied. The controller specifications are as follows.

For GPC:

- output prediction horizons $N_1 = 1$, $N_2 = 6$, control horizon $N_c = 2$.
- control weight 0.01
- forgetting factor in RLS 0.95.

For pole placement:

- observer polynomial, characteristic polynomial, dead time as in example 1.
- factorization of $\hat{B}(q)$ into minimum- and non-minimum-phase parts as in Aström and Wittenmark (1980), and calculations made accordingly.
- forgetting factor in RLS 0.95.

For PBC:

- prediction horizon $n = 1$

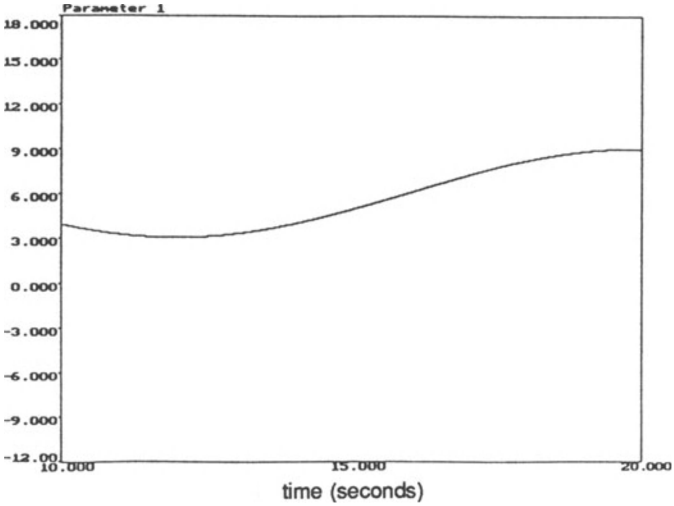


FIGURE 25.8. Smooth variation of parameter a_1.

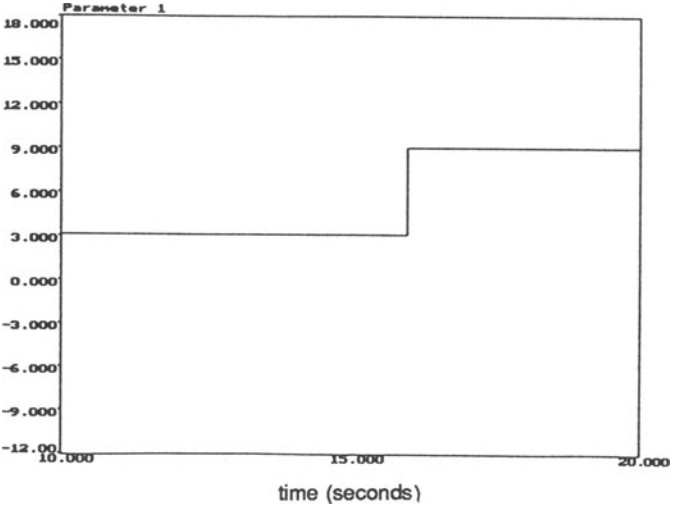


FIGURE 25.9. Step change of parameter a_1.

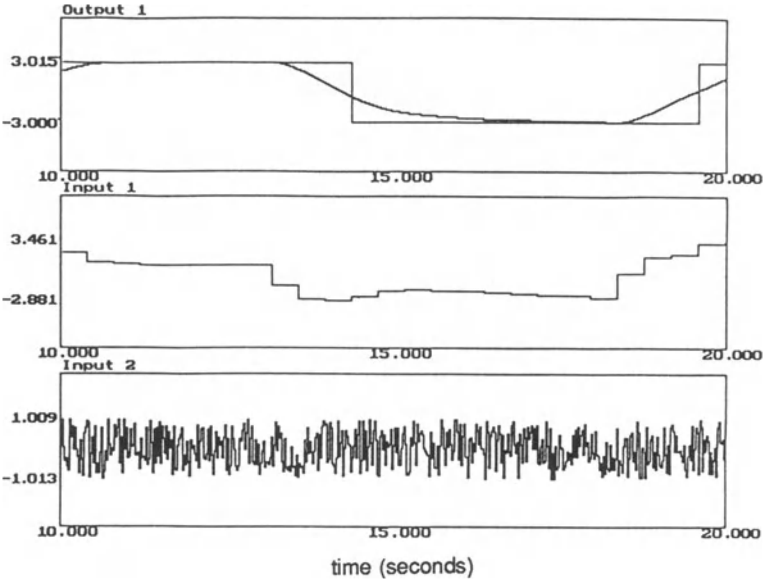


FIGURE 25.10. Set-point and controlled output, control input, and disturbance signals for GPC in Example 2.

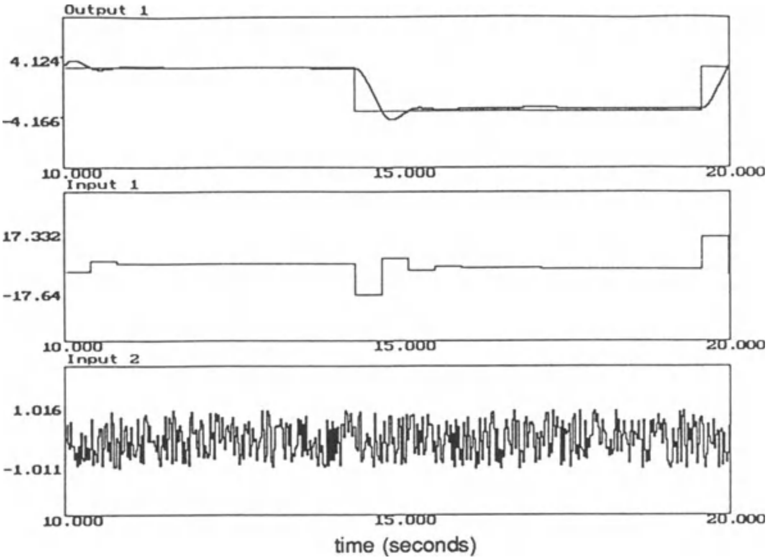


FIGURE 25.11. Set-point and controlled output, control input, and disturbance signals for pole placement in Example 2.

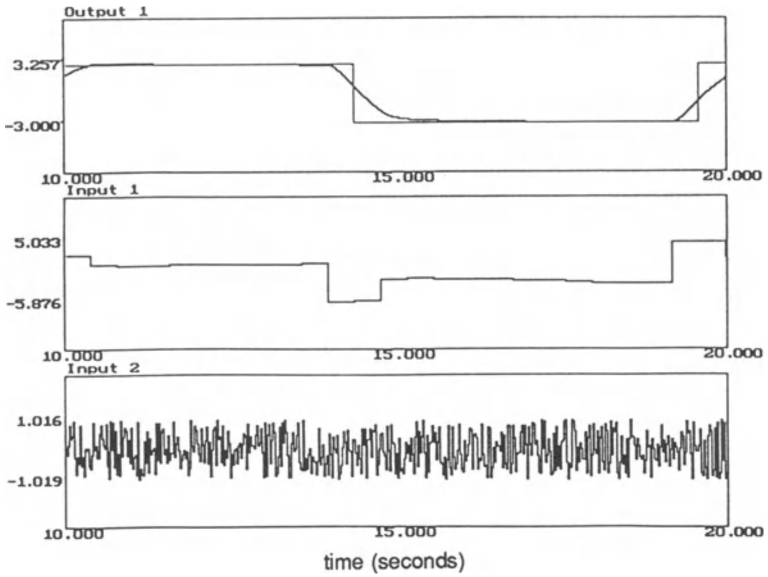


FIGURE 25.12. Set-point and controlled output, control input, and disturbance signals for PBC in Example 2.

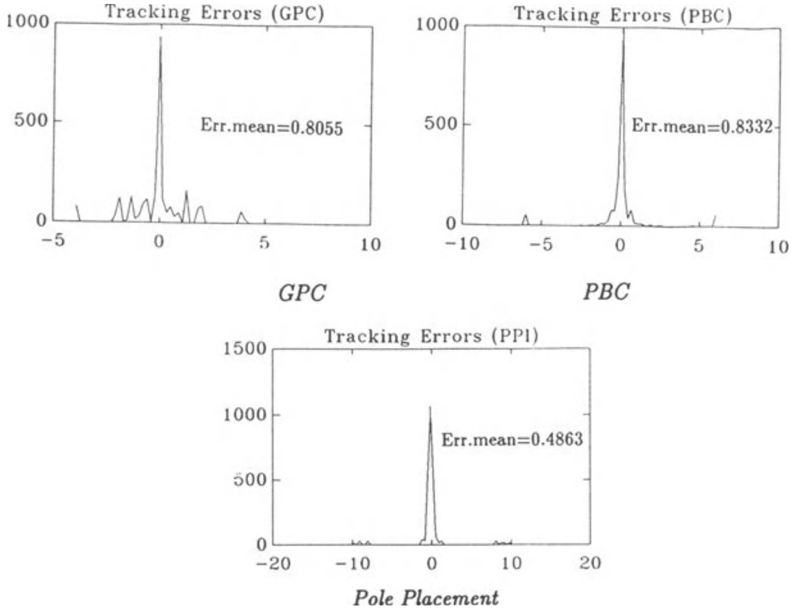


FIGURE 25.13. Sample means and density functions for output errors over 500-sec runs with smooth sinusoidal parameter change in a_1.

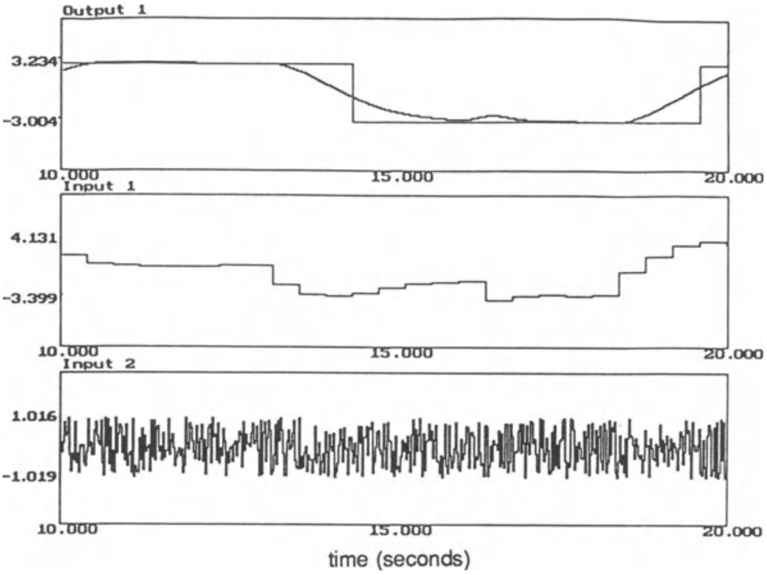


FIGURE 25.14. Set-point and controlled output, control input, and disturbance signals for GPC in Example 2.

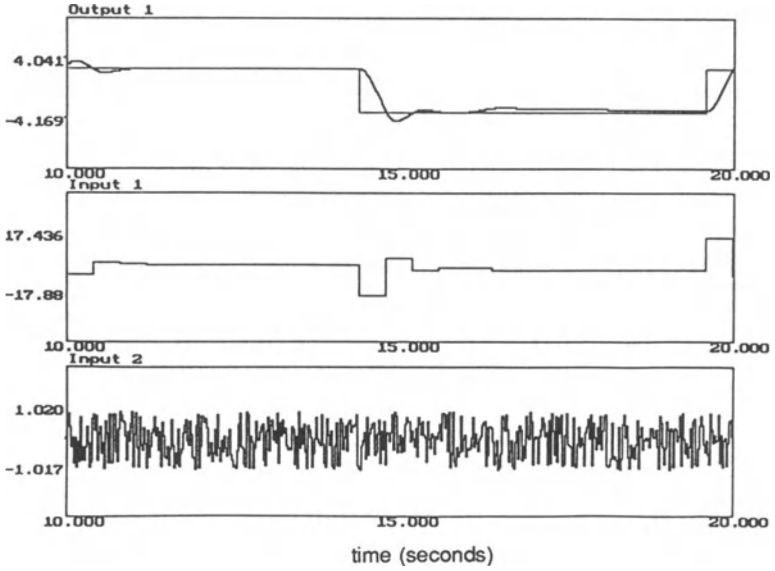


FIGURE 25.15. Set-point and controlled output, control input, and disturbance signals for pole placement in Example 2.

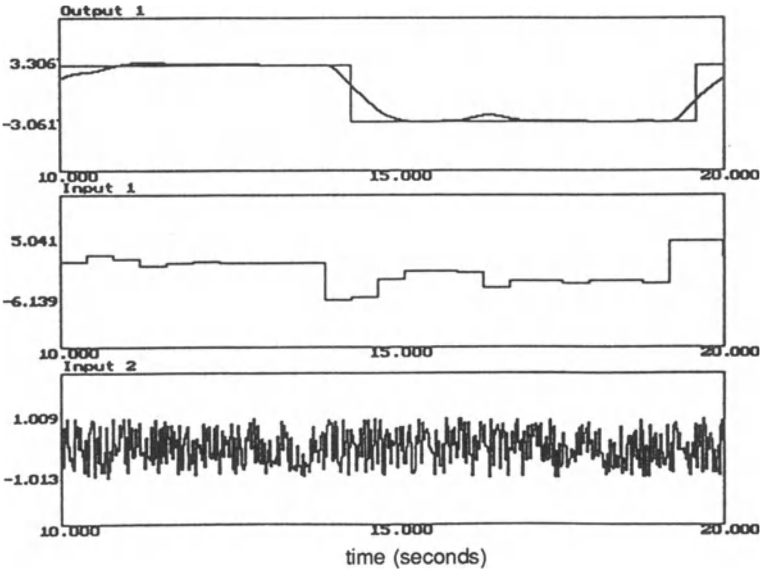


FIGURE 25.16. Set-point and controlled output, control input, and disturbance signals for PBC in Example 2.

- polytope $\mathcal{D}_t$ found as for example 1.
- bound inflation factor 0.95.
- robust closed-loop stability bounds as for example 1.
- stabilizing horizon $N_b = 8$

Figs. 25.10 to 25.12 show the results for the smooth parameter change shown in Fig. 25.8. The figures show similar performance by all three methods, demonstrating that although PBC copes well, it does not always result in significant improvement. Figs. 25.14 to 25.16 show the results for the abrupt parameter change.

25.7. CONCLUSIONS

A new type of adaptive control scheme, predictive bounding control, has been presented. In contrast to existing self-tuning control methods, it accounts for the uncertainty in the plant model at every step. It does so by computing parameter bounds and employing a worst-case performance criterion. Identification and control are closely integrated in PBC. A trade-off can be made between immediate output performance and the longer-term effect of model accuracy on control. An adaptive scheme adjusts the specified bounds on model-output errors and plant-parameter changes according to how long the model parameter bounds remain valid.

The price for the versatility and robustness thus achieved is a considerably higher computational demand than for traditional methods. Nevertheless, the scheme is practicable with present computing resources.

REFERENCES

1. G. Belforte and M. Milanese, in: *Proceedings of the 1st IASTED Symposium on Modelling, Identification and Control*, Davos, Switzerland, pp. 75–79 (1981).
2. E. Fogel and Y. F. Huang, *Automatica* **18**, 229 (1982).
3. J. P. Norton, *Automatica* **23**, 497 (1987).
4. E. Walter and H. Piet-Lahanier, *Math. Comp. Simul.* **32**, 449 (1990); *Preprints from the 12th IMACS World Congress*, Paris, France, pp. 467–472 (1988).
5. M. Milanese and A. Vicino, *Automatica* **27**, 403 (1991).
6. S. H. Mo and J. P. Norton, *Math. Comp. Simul.* **32**, 481 (1990); *Preprints of the 12th IMACS World Congress*, Paris, France, pp. 477–480 (1988).
7. H. Piet-Lahanier and E. Walter, *Math. Comp. Simul.* **32**, 495 (1990); *Preprints of the 12th IMACS World Congress*, Paris, France, pp. 481–483 (1988).
8. S. M. Veres and J. P. Norton, in: *Proceedings of the 9th IASTED Conference on Mathematical Modelling,* Innsbruck, Austria, pp. 367–370 (1990).
9. J. P. Norton, *Int. J. Control* **45**, 375 (1987).
10. S. M. Veres and J. P. Norton, in: *9th IFAC/IFORS Symposium on Identification & System Parameter Estimation*, Budapest, Hungary, pp. 1038–1043 (1991).
11. K. J. Åström and B. Wittenmark, *Automatica* **9**, 185 (1973).
12. D. W. Clarke and P. J. Gawthrop, *Proc. IEE* **122**, 929 (1975).
13. D. W. Clarke and P. J. Gawthrop, *Proc. IEE* **126**, 633 (1979).
14. D. W. Clarke, *Automatica* **20**, 501 (1984).
15. D. W. Clarke, P. P. Kanjilal, and C. Mohtadi, *Int. J. Control* **41**, 1509 (1985).
16. P. E. Wellstead, D. Prager, and P. Zanker, *Proc. IEE* **126**, 781 (1979).
17. K. J. Åström and B. Wittenmark, *Proc. IEE* **127**, 120 (1980).
18. D. W. Clarke, C. Mohtadi, and P. S. Tuffs, *Automatica* **23**, 137; *Automatica* **23**, 149 (1987).
19. D. W. Clarke and C. Mohtadi, *Automatica* **25**, 859 (1989).
20. V. Peterka, *Automatica* **20**, 39 (1984).
21. B. E. Ydstie, in: *Proceedings of the IFAC 9th World Congress*, Budapest, Hungary, pp. 133–138 (1984).
22. R. M. C. De Keyser and A. R. Cauwenberghe, *Journal A* **22**, 167 (1981).
23. R. M. C. De Keyser and A. R. Cauwenberghe, in: *IFAC Symposium on Identification & System Parameter Estimation,* Arlington, VA, pp. 1552–1557 (1982).
24. M. B. Lelić and M. B. Zarrop, *A Generalized Pole-Placement Self-Tuning Controller*, Control Systems Centre Report No. 657, Univ. of Manchester Inst. of Science & Technology, Manchester, U.K. (1986).
25. R. H. Middleton and G. C. Goodwin, *IEEE Trans. Autom. Control* **AC-33**, 150 (1988).
26. R. H. Middleton, G. C. Goodwin, D. J. Hill, and D. Q. Mayne, *IEEE Trans. Autom. Control* **AC-33**, 50 (1988).
27. R. H. Middleton, G. C. Goodwin, and Y. Wang, *Automatica* **25**, 889 (1989).
28. B. D. O. Anderson, *Automatica* **21**, 247 (1985).
29. P. E. Caines and S. Lafortune, *IEEE Trans. Autom. Control* **AC-29**, 312 (1984).

30. H. F. Chen, *SIAM J. Control Optim.* **22**, 758 (1984).
31. H. F. Chen and L. Guo, *Int. J. Control* **43**, 869 (1986).
32. H. F. Chen and L. Guo, *SIAM J. Control Optim.* **25**, 559 (1987).
33. S. M. Veres and J. P. Norton, in: *ITAC '91, IFAC Symposium on Intelligent Tuning and Adaptive Control*, Singapore (1991).
34. S. M. Veres and J. P. Norton, in: *Proceedings Control '91 IEE International Conference*, Edinburgh, Scotland, pp. 1095–1100 (1991).
35. S. M. Veres and J. P. Norton, in: *9th IFAC/IFORS Symposium on Identification & System Parameter Estimation*, Budapest, Hungary, pp. 773–778 (1991).
36. V. Kučera, *Discrete Linear Control: The Polynomial Equation Approach*, Wiley, London, U.K. (1979).
37. M. J. Grimble, in: *Preprints of the 9th IFAC World Congress*, pp. 160–165 (1984).
38. C. E. Garcia, D. M. Prett, and M. Morari, *Automatica* **25**, 335 (1989).
39. S. M. Veres and J.P. Norton, *Identification of errors in variables models by parameter bounding methods, Research Memorandum No. 25*, School of Electronic and Electrical Engineering, University of Birmingham, Birmingham, U.K. (1989).
40. K. J. Åström and B. Wittenmark, *Adaptive Control* Second Ed., Addison-Wesley, Reading, MA (1995).
41. L. Ljung and S. Gunnarsson, *Automatica* **26**, 23 (1990).
42. A. Benveniste, *Int. J. Adapt. Control Signal Proc.* **1**, 3 (1987).
43. J. P. Norton, *Proc. IEE* **122**, 663 (1975).
44. J. P. Norton, *Proc. IEE* **123**, 451 (1976).
45. J. P. Norton, *An Introduction to Identification*, Academic Press, New York (1986).
46. T. Bohlin, in: *System Identification: Advances and Case Studies* (R. K. Mehra and D. G. Lainiotis, eds.), Academic Press, New York (1976).
47. P. C. Young, *Recursive Estimation and Time-Series Analysis*, Springer-Verlag, Berlin, Germany (1984).
48. M. J. Chen and J. P. Norton, *Int. J. Control* **45**, 1387 (1987).
49. J. P. Norton and S. H. Mo, *Math. & Comput. Simul.* **32**, 527 (1990).

APPENDIX A.

Computation of Stabilizing Control Inputs

The aim is to find for all those u_t inputs given by

$$R(q^{-1})u_t = \tilde{u}_t - S(q^{-1})y_t$$

where

$$P(q) = q^{p+q}[R(q^{-1})A(q^{-1};\boldsymbol{\theta}^p) + S(q^{-1})B(q^{-1};\boldsymbol{\theta}^p)]$$

is a stable polynomial for any $\boldsymbol{\theta}^p \in \mathcal{P}_1(t,\delta,\rho)$. Here $degR = degB = q - 1$, $degS = degA = p$.

First, rewrite the polynomial multiplications in matrix form. With vector $f \equiv [r_1,\ldots,r_q,s_0,s_1, \ldots, s_p]^T$composed of the coefficients of $R(q^{-1}) \equiv 1 + r_1q^{-1} + \ldots + r_qq^{-r}$ and $S(q^{-1}) \equiv s_0 + s_1q^{-1} + \ldots + s_pq^{-p}$, the coefficients of $P(q)$ can be expressed as $M(\boldsymbol{\theta}^p)\bar{\mathbf{f}}$

$$M(\boldsymbol{\theta}^p)\bar{\mathbf{f}} = \begin{bmatrix} 1 & 0 & b_1 & & 0 \\ a_1 & 1 & b_2 & & 0 \\ \vdots & & \vdots & & \\ a_p & & 1 & b_r & b_1 \\ & & a_1 & 0 & \\ & & \vdots & & \vdots \\ 0 & \cdots & a_p & 0 & b_r \end{bmatrix} \begin{bmatrix} 1 \\ r_1 \\ \vdots \\ r_q \\ s_0 \\ \vdots \\ s_1 \\ \vdots \\ s_p \end{bmatrix}$$

and $\bar{\mathbf{f}} = \begin{bmatrix} 1 \\ \mathbf{f} \end{bmatrix}$. Define a large region for the coefficients of $(p + q + 1)$-degree stable polynomials, in the form

$$\mathcal{L}_{p+q+1} = \bigcup_{i\in\mathcal{J}(p+q+1)} \mathcal{L}_i^{p+q+1}$$

where $\mathcal{L}_i^{p+q+1}, i\in \mathcal{J}(p + q + 1)$, are polytopes in the space of coefficients of $(p + q + 1)$-degree polynomials. The sets of vectors $\mathbf{f}$ which give stable closed-loop behavior is

$$\{\mathbf{f} \mid \forall\boldsymbol{\theta}\in \mathcal{R}_k(t,\delta,c)\colon M(\boldsymbol{\theta})\bar{\mathbf{f}} \in \mathcal{L}_i^{p+r+1}\} = \{\mathbf{f} \mid \forall\boldsymbol{\theta}_v \in \mathcal{V}_k\colon M(\boldsymbol{\theta}_v)\bar{\mathbf{f}} \in \mathcal{L}_i^{p+r+1}\},$$

where $\mathcal{V}_k$ denotes the finite set of vertices of polytope $\mathcal{P}_k(t,\delta,\rho)$. Relation $M(\boldsymbol{\theta})\bar{f} \in \mathcal{L}_i^{p+r+1}$ can, however, be rewritten in the form of a set of linear inequalities

$$\mathbf{a}_j^T M(\boldsymbol{\theta}v)\bar{\mathbf{f}} \le c_j,\quad j = 1,2, \ldots, \mathcal{H}_i,\ \theta_v \in \mathcal{V}_k$$

where $\mathcal{H}_i$ denotes the number of supporting hyperplanes of $\mathcal{L}_i^{p+r+1}$, which clearly shows that the set of "stabilizing" vectors f (associated with each $\mathcal{L}_i^{p+q+1}$) spans a polytope $\mathcal{T}_i$ for every $i\in\mathcal{J}(p + q + 1)$. The total set of "stabilizing" f vectors is the union of a finite set of polytopes. Finally, the set of stabilizing inputs is the union of intervals

$$\mathcal{U}_t^s = \bigcup_{i\in\mathcal{J}(p+q+1)} [\min_{\mathbf{f}_v\in\mathcal{H}_i} (y_t^* - \mathbf{f}_v^T\phi_t^1),\ \max_{\mathbf{f}_v\in\mathcal{H}_i} (y_t^* - \mathbf{f}_v^T\boldsymbol{\phi}_t^1)],$$

where $\mathcal{H}_i$ is the set of vertices of the convex polytopes $\mathcal{T}_i$, $i\in\mathcal{J}(p + q + 1)$.

APPENDIX B

Influence of Regressor Vectors on Size and Existence of Feasible-Parameter Set

Simple geometry shows that, with equation-error bound δ and bound inflation factor ρ, if the angles between all pairs of the regressor vectors $\phi_t, \dots, \phi_{t-p+1}$ are larger than α, then the largest diameter of the parallelpiped

$$P(t-p+1,t) = \{\boldsymbol{\theta} \mid | y_{t-i} - \boldsymbol{\theta}^{\mathrm{T}}\phi_{t-i} | \le \delta\rho^{i},\ i = 0, 1, 2, \dots, p-1\}$$

is bounded by

$$Diam(P(t-p+1,t)) \le 4\delta\rho^{p(p+1)/2}\,\frac{\cos(\alpha/2)}{\sin(\alpha)}\cdot\frac{1}{\min\limits_{i=o,\dots,p-1}\|\phi_{t-i}\|} \equiv \beta$$

Lemma B. Assume that α and $\mu = \min_{i=o,1,\dots,p-1} \|\phi_{t-i}\|$ are such that the abrupt parameter changes are all greater than 2β, and that ρ and δ are large enough to cover all equation errors, including the effects of parameter drift, in the absence of abrupt changes. If after an abrupt parameter change at time t (or initially, after turning on the bounding controller), the control inputs are selected so that $Ang(\phi_{t+i},\phi_{t+j}) \ge \alpha$, $i = 1,2, \dots, p-1$; $j = i+1, \dots, p$ and $\|\phi_{t+i}\| \ge \mu$, $i = 1, \dots, p$ then every abrupt change, and only an abrupt change, causes the end of a generation.

Proof: A fall in age $m(t,\delta,\rho)$ by time $t+p$ follows straightforwardly from the fact that parallelepiped calculated in the first p steps after each abrupt change does not intersect the parallelepiped formed by the parameter bounds imposed in the last p steps before the change, and hence conflicts with the feasible-parameter set at t. The assumption about ρ and δ ensures that slow parameter drift does not cause a clash of bounds during the time interval between two abrupt changes, so falls in age occur only as a result of abrupt parameter changes. □

APPENDIX C

Example of Bound Tuning

The tuning method for the equation-error bound δ and the scale factor ρ for the parameter increments is demonstrated. Output samples are generated by

$$y_t = a_1 y_{t-1} + 0.95 y_{t-2} + 0.8 u_{t-1} - 0.5 u_{t-2} + e_t + o_t$$

For simplicity only parameter a_1 varies, stepwise between 0.5 and -0.6, and e_t is randomly generated from $[-0.2, 0.2]$. For a single sequence the numbers of generations are as in Fig. C. There is a plateau in the number of generations from

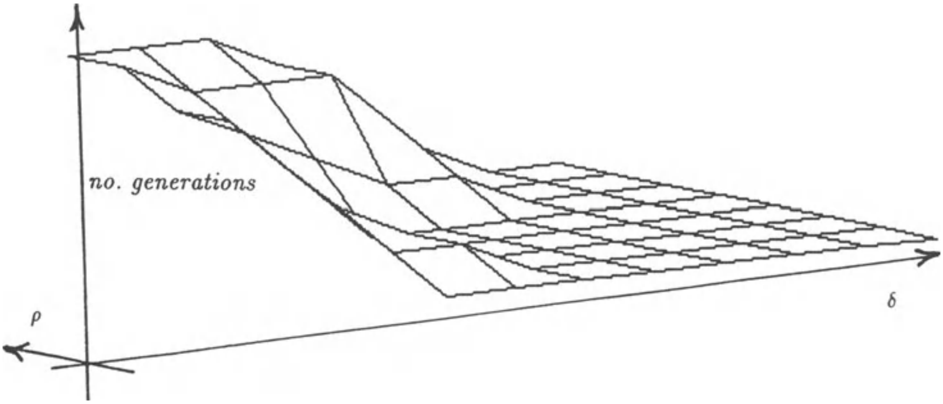


FIGURE 25.C. Number of generations against equation-error bound δ and parameter-increment scale factor ρ.

which both the number of jump changes and the range of feasible bound pairs (δ,ρ) can be inferred.

26

System Identification for H_∞-Robust Control Design

T. J. J. van den Boom and A. A. H. Damen

26.1. INTRODUCTION

In conventional identification techniques a model is proposed which is supposed to be capable of representing the process behavior under study. Parameters are then tuned such that the model outputs correspond according to some criterion for the dominant part of a measured data set. Deviations are thought to be concentrated in some error source in the model, such as output error, prediction error, equation error, and so forth. This artificial error source explains all disturbances acting on the process as well as for all model deviations from the real dynamic behavior of the process. Furthermore, stochastic assumptions have to be proposed concerning the errors leading to the criterion and as a result a "best" model is produced together with some stochastically based range for the parameters and/or dynamic behavior.

For H_∞-robust control design, a best model is required in the sense that a known model error bound can be guaranteed.(1,2) The disturbances should preferably be characterized by filters with norm bounded inputs.

A lot of work has already been done in order to overcome the drawbacks of conventional identification, and to derive bounds for the model error. This has been done, either in a stochastical setting(3,4,5) or in a deterministic setting, with bounded

T. J. J. van den Boom • Department of Electrical Engineering, Delft University of Technology, 2600 GA Delft, The Netherlands. A. A. H. Damen • Department of Electrical Engineering, Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

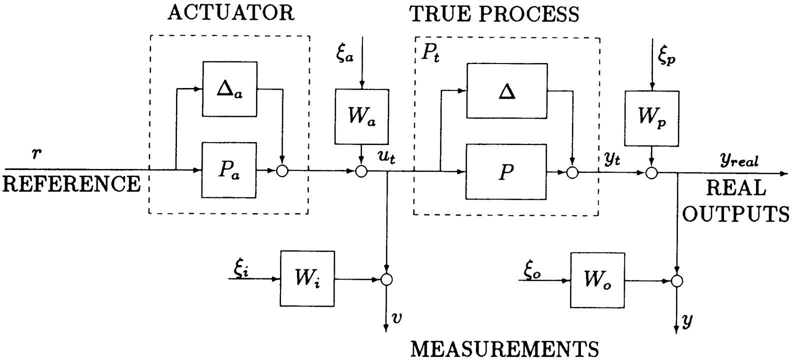


FIGURE 26.1. Detailed set-up.

noise in the time domain,[6,7] or with bounded noise in the frequency domain.[8–14] It is our strong belief that any model identification method should provide explicit descriptions of the accuracy of the model.

A clear and detailed distinction between modeling errors and various disturbances is indispensable. The disturbances should preferably be characterized by filters with norm bounded inputs. As a consequence, detailed information about the process should be acquired in order to arrive at acceptable error bounds. Mere input/output data is far from sufficient.

Fig. 26.1 superficially indicates ideas about details. Extensive preliminary measurements and data processing should provide information about the actuator dynamics $P_a(z)$, bounds for the actuator modeling error $\Delta_a(z)$ (for example,

$$||W_{ac}(z)\Delta_a(z)||_\infty \leq 1,$$

where $W_{ac}(z)$ is an appropriate weighting filter). Furthermore, the disturbances acting on the process should be described as constraints in frequency domain. The filters W_a, W_p, W_i and W_o are chosen such that the error sources ξ_a, ξ_p, ξ_i and ξ_o have discrete Fourier transforms bounded by one:

$$||\xi_a(z)||_\infty \leq 1, \quad ||\xi_s(z)||_\infty \leq 1, \quad ||\xi_i(z)||_\infty \leq 1, \quad ||\xi_o(z)||_\infty \leq 1.$$

In Fig. 26.1, the true process $P_t(z)$ is given in a non-structured additive model error configuration. The aim is to choose a model $P(z)$ in a model set such that the model error $\Delta(z)$ is as small as possible; the $||W_\Delta(z)\Delta(z)||_\infty$ is minimized, with $W_\Delta(z)$ is an appropriate weighting filter.

Since all analysis is in the frequency domain write for the true process P_t and the nominal model P to be selected from the model set $\mathcal{P}$:

$$\min_{P\in\mathcal{P}}\|W_\Delta\Delta\|_\infty = \min_{P\in\mathcal{P}}\|W_\Delta(P_t - P)\|_\infty = \min_{P\in\mathcal{P}}\|W_\Delta(\frac{y_t}{u_t} - P)\|_\infty$$

where

$$y_t \in \tilde{\mathcal{Y}} = \{y - W_o\xi_o - W_p\xi_p \mid \|\xi_o\|_\infty \le 1, \|\xi_p\|_\infty \le 1\}$$

$$u_t \in \tilde{\mathcal{U}} = \{[(P_a + \Delta_a)r + W_a\xi_a \mid \|W_{ac}\Delta_a\|_\infty \le 1, \|\xi_a\|_\infty \le 1]$$

$$\cap\, [v - W_i\xi_i \mid \|\xi_i\|_\infty \le 1]\}$$

This chapter considers a simplified version of above concept. Consider the situation of Fig. 26.2, where the sets of possible inputs and outputs are less complicated.

True input and output signals u_t and y_t are measured in u and y, where noise signals d and e are involved such that $u_t = u - d_t$ and $y_t = y + e_t$. The signal u can stand for either v or r; the other signal is supposed to be unknown. Suppose the control signal r is unknown so that $d = -W_i\xi_i$ and $e = W_p\xi_p + W_o\xi_o$ to get from Fig. 26.1 to Fig. 26.2. Alternatively, think of doing no measurements (r is known) and $u = P_ar$, $d = \Delta_ar + W_a\xi_a$ and again $e = W_p\xi_p + W_o\xi_o$. Both cases arrive at the setup of Fig. 26.2 and give bounds in the frequency domain for the noise signals:

$$|W_d^{-1}d| \le 1 \text{ and} |W_e^{-1}e| \le 1$$

For the filters W_d and W_e we get:

$$\{|W_e| = |W_p| + |W_o|$$

$$\begin{cases} |W_d| = |W_i| & \text{for } u = v \\ |W_d| = |W_{ac}|\,|r| + |W_a| & \text{for } u = P_ar \end{cases}$$

The proposed identification method is executed completely in the frequency-domain. The given data set $u(k), y(k)$ for $k = 1, \ldots, 2N$ consists of samples in the time domain. In order to apply our identification method, the data set has to be transformed to the frequency domain.

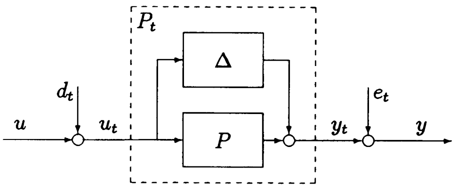


FIGURE 26.2. Basic experimental set-up with additive model error.

26.2. ASSUMPTIONS AND PROBLEM STATEMENT

This chapter is restricted to SISO-systems and only discusses an additive and a multiplicative model error structure. (For MIMO-systems and more general error structures see Ref. 13.) The following remarks concerning the objects of study can be stated:

Plant: The true plant, denoted by $P_t(z)$ is assumed to be linear and time-invariant and stable in the configuration of Fig. 26.2. The plant is excited by the unknown true input signal $u_t(k)$, which results in the unknown true output signal $y_t(k)$. Small non-linear perturbations are accounted for in the additive disturbance $e_t(k)$.

Data set: We do an experiment and measure the input and output in the signals $u(k)$ and $y(k)$ for $k = 1, ..., 2N$, which results in the data set $\{\bar{u}(k), \bar{y}(k)\}$. A discrete Fourier transformation leads to the dataset in the frequency domain:

$$\{\bar{u}(k), \bar{y}(k)\} \overset{\mathcal{F}}{\rightarrow} \{u(z), y(z)\}$$

with

$$z \in \Omega = \{z_1, z_2, \ldots, z_N\}$$

for

$$z_i = e^{j\pi i/N},\ i = 0, \ldots, N-1.$$

In the sequel, we will just consider the models for these frequencies. The model error bounds are computed on just this finite number of frequencies. However, assume that the true process P_t and the optimal model P_{opt} have an impulse response much smaller than the observation interval and, therefore, much smaller than the number of observed frequencies. We can use a simple interpolation technique to find a bound over all z on the unit circle.[13]

Disturbances: The applied input signals are perturbed by additive actuator disturbance d_t, and the true output signal is perturbed by additive output disturbance and measurement noise e_t. Assume the disturbances to belong to the disturbance sets $\tilde{\mathcal{D}}$ and $\tilde{\mathcal{E}}$, respectively. $\tilde{\mathcal{D}}$ and $\tilde{\mathcal{E}}$ consist of all signals with discrete Fourier transforms that are bounded by known disturbance filters $W_d(z)$ and $W_e(z)$:

$$d_t(k) \in \tilde{\mathcal{D}} = \{\tilde{d}(k) \mid |\tilde{d}(z)| < |W_d(z)|, z \in \Omega\}$$

and

$$e_t(k) \in \tilde{\mathcal{E}} = \{\tilde{e}(k) \mid |\tilde{e}(z)| < |W_e(z)|, z \in \Omega\}$$

Model set: The user has to define a model set $\mathcal{P} \subset \mathcal{R}$ with parametrized models $P(\theta,z)$ and parameter vector θ in a set Θ. The true process $P_t(z)$ is not necessarily in this set.

Model error structure: To describe the uncertainty in the model, make use of either the additive or the multiplicative model error structure. The uncertainty $\Delta(z)$ is thought to be additive $\Delta_a(z)$ or multiplicative $\Delta_m(z)$ with respect to the nominal model $P(z)$, so that the true process is described by

(1) Additive: $P_t(z) = P(z) + \Delta_a(z)$
(2) Multiplicative: $P_t(z) = P(z)(I + \Delta_m(z))$

The corresponding configurations are given in Fig. 26.3. The additive and multiplicative model error structures are described in Refs. 1 and 15. Now concentrate on the following model error optimization problem (with either an additive or a multiplicative model error structure).

Find a model $P(z)$ in a given model set $\mathcal{P}$ such that the H_∞-norm of the weighted model error $\Delta(z)$ is minimized, so:

$$\inf_{P \in \mathcal{P}} \|W_\Delta(z)\Delta(z)\|_\infty$$

for some given weighting filter $W_\Delta(z)$ and where $\Delta(z)$ is either the additive model error

$$\Delta_a(z) = P_t(z) - P(z)$$

or the multiplicative model error

$$\Delta_m(z) = (P_t(z) - P(z))P^{-1}(z) = P_t(z)P^{-1}(z) - 1.$$

To make sure that the model error is stable, assume that both the true process $P_t(z)$ and the model $P(z)$ are stable. In the case of a multiplicative model error, the additional assumptions are that the true process $P_t(z)$ and the model $P(z)$ are minimum-phase. Because of the disturbance signals $d(z)$ and $e(z)$, one cannot determine the true process $P_t(z)$ exactly and cannot compute the H_∞-norm of the model error. With the use of the noise sets $\tilde{\mathcal{D}}$ and $\tilde{\mathcal{E}}$ we are able to calculate an upper bound for the H_∞-norm of the model error. Instead of minimizing the H_∞-norm itself we will minimize the upper bound for the H_∞-norm of the model error.

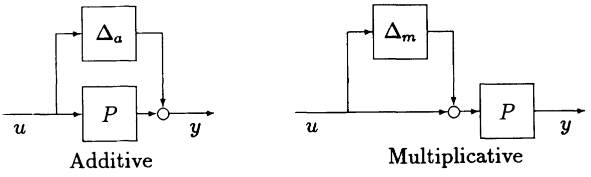


FIGURE 26.3. Model error structures.

It is obvious that the only computational difference between the additive and multiplicative model error is an extra weighting by P^{-1} entering Δ and W_Δ. This difference, however, has a great influence on the minimization, as discussed in Section 26.3.2.

This chapter presents two methods, a two-step and a one-step method. The first step in the two-step identification procedure is to derive uncertainty regions for the system dynamics in the complex frequency plane. The second step finds an approximate model that is optimal in the sense that the upper bound for the weighted model error is minimized. The two-step identification procedure consisting of the derivation of the uncertainty regions and the H_∞ fitting is clear and comprehensible. The main problem is that it does not always lead to the optimal solution because of approximations that are made.

Section 26.4 discusses a one-step identification method based on techniques that rise from robust control theory; it provides the optimal model, given the *a priori* knowledge. It makes use of the concepts of linear fractional transformations and the structured singular value μ.

26.3. TWO-STEP IDENTIFICATION METHOD

26.3.1. Derivation of Uncertainty Regions

Fig. 26.2 shows that

$$u_t(z) = u(z) + d_t(z) \text{ and } y_t(z) = y(z) - e_t(z).$$

However, the signals $d_t(z)$ and $e_t(z)$ are not available. They belong to $\tilde{\mathcal{D}}$ and $\tilde{\mathcal{E}}$, respectively. Therefore, define the following sets:

$$\tilde{\mathcal{U}} = \{\tilde{u}(z) = u(z) + \tilde{d}(z), |\tilde{d}(z)| \le W_d(z), z \in \Omega\}$$

and

$$\tilde{\mathcal{Y}} = \{\tilde{y}(z) = y(z) - \tilde{e}(z), |\tilde{e}(z)| \le W_e(z), z \in \Omega\}$$

Note that $u_t \in \tilde{\mathcal{U}}$ and $y_t \in \tilde{\mathcal{Y}}$.

Consider all signals for one specific frequency, z, and use a simplified notation (i.e., $\tilde{u}$ instead of $\tilde{u}(z)$, $\tilde{y}$ instead of $\tilde{y}(z)$). $\tilde{\mathcal{U}}$ and $\tilde{\mathcal{Y}}$ for one specific frequency are

$$\tilde{\mathcal{U}} = \left\{\tilde{u} = u(1 + \alpha_u e^{j\theta}), 0 \le \theta < 2\pi, 0 \le \alpha_u \le \frac{|W_d|}{|u|}\right\}$$

and

$$\tilde{\mathcal{Y}} = \left\{\tilde{y} = y(1 + \alpha_y e^{j\phi}), 0 \le \phi < 2\pi, 0 \le \alpha_y \le \frac{|W_e|}{|y|}\right\}$$

Fig. 26.4 shows the sets $\tilde{\mathcal{U}}$ and $\tilde{\mathcal{Y}}$ in the complex plane for one frequency sample. For example, take d and e as white Gaussian noises and let the bound be

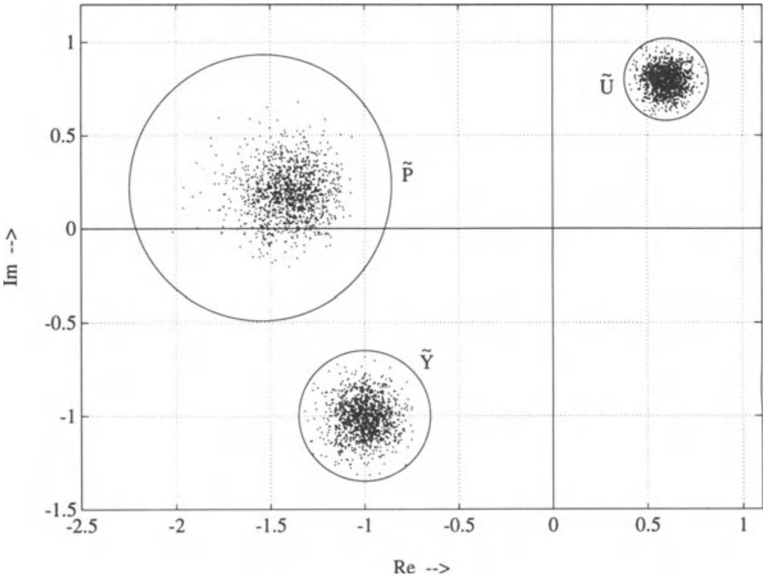


FIGURE 26.4. Sets $\tilde{\mathcal{U}}$, $\tilde{\mathcal{Y}}$, and $\tilde{\mathcal{P}}$ in the complex plane.

given by the 3σ-bound (see Section 26.2.4). One thousand realizations of these disturbances have been presented by points in Fig. 26.4. The point density indicates the probability of the expected signal values.

Now an estimate for the true process $P_t(z) = y_t(z)/u_t(z)$ has to be obtained. Therefore, we define the set of unfalsified systems. This set consists of systems $\tilde{\mathcal{P}} \in \mathcal{R}$ that do not falsify the measured data and the noise bounds. So consider the functions $\tilde{P}(z) = \tilde{y}(z)/\tilde{u}(z)$ for all $\tilde{u}(z) \in \tilde{\mathcal{U}}$ and $\tilde{y}(z) \in \tilde{\mathcal{Y}}$ (assume that $\tilde{u}(z) \neq 0$, for all $z \in \Omega$). This means that we are dealing with a persistently exciting input $u_t(z)$ and a sufficiently small input noise signal $\tilde{d}(z)$. Note that the true process $P_t(z)$ is an element of the set $\tilde{\mathcal{P}}$.

For the example of Fig. 26.4 can derive a representation for the set $\tilde{\mathcal{P}}$, with transfer functions $\tilde{\mathcal{P}}$ by dividing all elements $\tilde{y} \in \tilde{\mathcal{Y}}$ by all elements $\tilde{y} \in \tilde{\mathcal{U}}$. In Fig. 26.4 the set $\tilde{\mathcal{P}}$ is given in the complex plane for one frequency sample. As before, the point density indicates the probability of the expected signal values. This results (for each frequency z) in[10,13]

$$\tilde{\mathcal{P}} = \{\tilde{P} = \frac{yu^*}{uu^* - W_d W_d^*}(1 + \alpha_y e^{j\phi})(1 + \alpha_u e^{j\psi}),$$

$$0 \leq \phi < 2\pi,\ 0 \leq \psi < 2\pi,\ 0 \leq \alpha_y \leq \frac{|W_e|}{|y|},\ 0 \leq \alpha_u \leq \frac{|W_d|}{|u|}\}$$

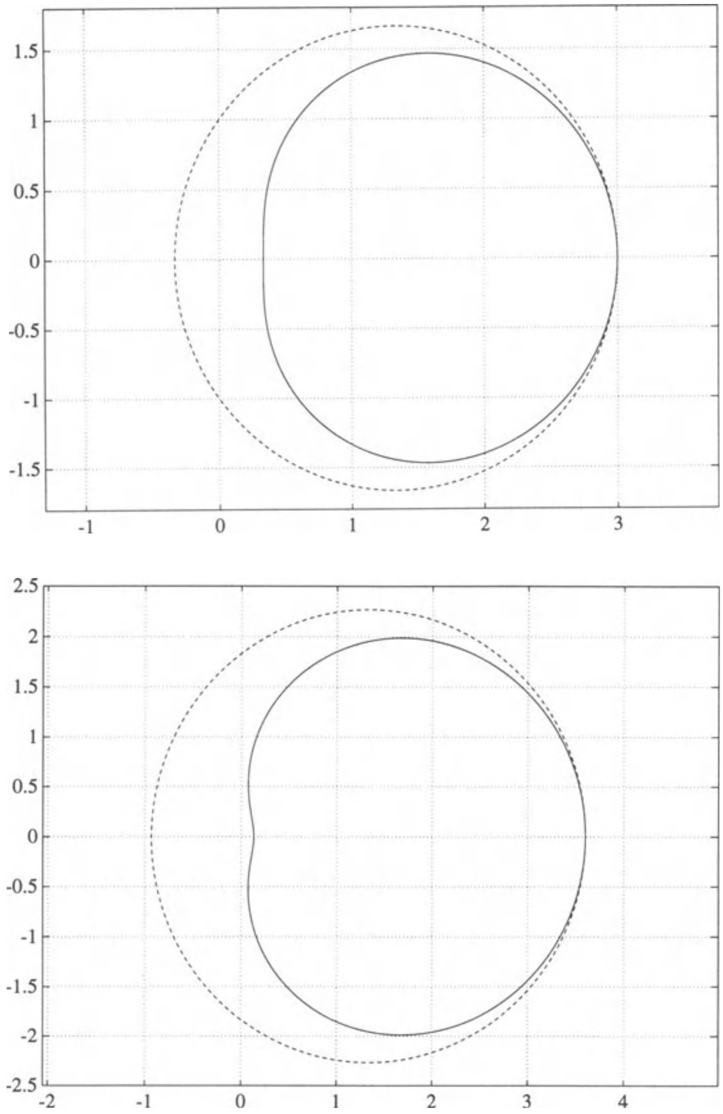
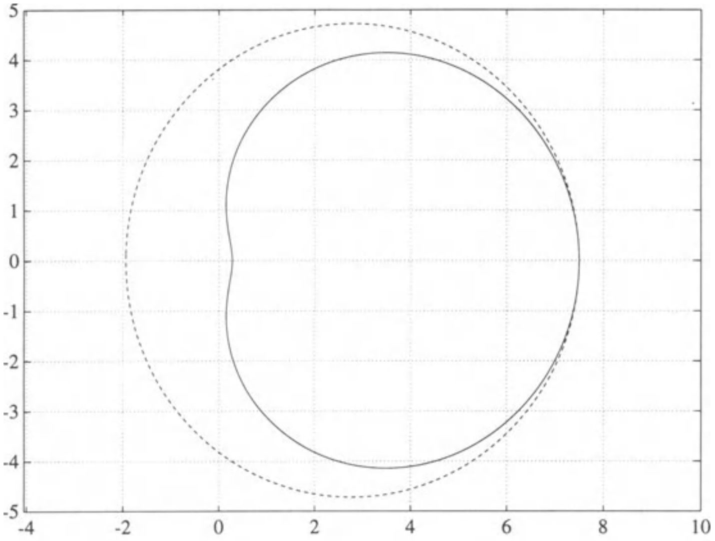


FIGURE 26.5. Sets $\tilde{\mathcal{P}}$ (———), and $\tilde{\mathcal{P}}_c$ (- - -) for various r and s: left upper, $r = 0.5$, $s = 0.5$; right upper, $r = 0.8$, $s = 0.5$; left lower, $r = 0.5$, $s = 0.8$; and right lower, $r = 0.8$, $s = 0.8$.



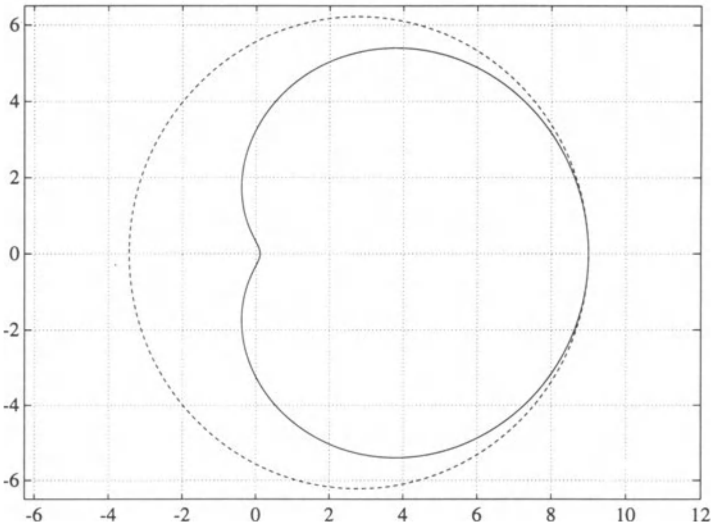


FIGURE 26.5. (Continued)

This is a region which is typically shaped like a bean, as exemplified in Fig. 26.5. The next step is to calculate a boundary function for the region. A simple circular bound (not the smallest)(13) for the set $\tilde{\mathcal{P}}$ can be derived easily. First define:

$$P_c = \frac{yu^*}{uu^* - W_d W_d^*}$$

and

$$r_c = |P_c|\left(\frac{|W_d|}{|u|} + \frac{|W_e|}{|y|} + \frac{|W_d W_e|}{|uy|}\right).$$

The following holds for all ϕ, ψ, α_y and α_u:

$$|\tilde{P} - P_c| = |P_c(1 + \alpha_e^{j\phi})(1 + \alpha_y e^{j\psi}) - P_c| =$$

$$= |P_c(\alpha_y e^{j\phi}) + \alpha_u e^{j\psi} + \alpha_y \alpha_u e^{j(\phi+\psi)})| \le$$

$$\le |P_c|\left(\frac{|W_d|}{|u|} + \frac{|W_e|}{|y|} + \frac{|W_d W_e|}{|uy|}\right) = r_c$$

An enclosing set $\tilde{\mathcal{P}}_c \supseteq \tilde{\mathcal{P}}$ with elements $\tilde{\mathcal{P}}_c$ can be given as

$$\tilde{\mathcal{P}}_c = \{\tilde{P}_c = P_c + \alpha_c e^{j\phi},\ 0 \le \theta < 2\pi,\ 0 \le \alpha_c \le r_c\}$$

The set $\tilde{\mathcal{P}}_c$ encloses the set $\tilde{\mathcal{P}}$ very tightly as long as

$$\frac{|W_d|}{|u|} = s << 1 \text{ and } \frac{|W_e|}{|y|} = r << 1$$

If r and s come closer to one, then the enclosing is less tight. This 'simple' enclosing set is easy to calculate and satisfactory in most cases. Exact expressions for the boundary function of the set $\tilde{\mathcal{P}}$ for specific frequency and the smallest circular enclosing set can be found in Ref. 13.

In Fig. 26.5 the region $\tilde{\mathcal{P}}$ is given for different values of r and s (where $y = 1$ and $u = 1$ are fixed), together with the computed enclosing circle $\tilde{\mathcal{P}}_c$.

26.3.2. H_∞-Fitting

So far we have only derived uncertainty regions for the true process. This section looks for an optimal nominal model in the predefined model set $\mathcal{P}$ and considers the optimization of this parametric model. It minimizes the upper bound of the H_∞-norm of the weighted model error (considering the uncertainty regions).

We define a model set $\mathcal{P}$ with the models $P(\theta,z)$, where $\theta \in \Theta$ is a vector with the model parameters. Of course one can choose many different types of models like ARMA, state space models, finite impulse response models, and so forth.

For additive model error and the multiplicative model error we define the sets of all candidate model errors as

$$\tilde{\Delta}_a = \{\tilde{\Delta}_a(z) \in \mathcal{S} \mid \tilde{\Delta}_a(z) = \tilde{P}(z) - P(z), \tilde{P} \in \tilde{\mathcal{P}}, P \in \mathcal{P}\}$$

and

$$\tilde{\Delta}_m = \left\{\tilde{\Delta}_m(z) \in \mathcal{S} \mid \tilde{\Delta}_m(z) = \frac{\tilde{P}(z) - P(z)}{P(z)}, \tilde{P} \in \tilde{\mathcal{P}}, P \in \mathcal{P}\right\}$$

Since $P_t(z) \in \tilde{\mathcal{P}}$, note that the true model errors belong to the defined model error sets: $\Delta_a(z) \in \tilde{\Delta}_a$ and $\Delta_m(z) \in \tilde{\Delta}_m$. In fact we would like to minimize the H_∞-norm of the true model error ($\|\Delta_a(z)|_\infty$ or $\|\Delta_m(z)\|_\infty$) over all admissible models in the model set $\mathcal{P}$. However, this can only give an upper bound in the presence of input and output noise:

$$\inf_{P\in\mathcal{P}} \|\Delta_a(z)\|_\infty \le \inf_{P\in\mathcal{P}} \sup_{\tilde{P}\in\tilde{\mathcal{P}}} \|\tilde{\Delta}_a(z)\|_\infty = \inf_{P\in\mathcal{P}} \sup_{\tilde{P}\in\tilde{\mathcal{P}}} \| \tilde{P}(z) - P(z)\|_\infty$$

and

$$\inf_{P\in\mathcal{P}} \|\Delta_m(z)\|_\infty \le \inf_{P\in\mathcal{P}} \sup_{\tilde{P}\in\tilde{\mathcal{P}}} \|\tilde{\Delta}_m(z)\|_\infty = \inf_{P\in\mathcal{P}} \sup_{\tilde{P}\in\tilde{\mathcal{P}}} \left\| \frac{\tilde{P}(z) - P(z)}{P(z)} \right\|_\infty$$

The McMillan degree of the model $P(\theta,z)$ is usually fixed, whereas the set $\tilde{\mathcal{P}}$ contains high order systems. So, in a sense, a parametrized model approximation problem has to be solved.

To emphasize specific frequency ranges, we can introduce a (stable and minimum phase) weighting filter $W_\Delta(z)$ and minimize the H_∞-norm of the weighted model error:

$$\inf_{P\in\mathcal{P}} \|W_\Delta(z)\Delta_a(z)\|_\infty \le \inf_{P\in\mathcal{P}} \sup_{\tilde{P}\in\tilde{\mathcal{P}}} \|W_\Delta(z)(\tilde{P}(z) - P(z))\|_\infty$$

or

$$\inf_{P\in\mathcal{P}} \|W_\Delta(z)\Delta_m(z)\|_\infty \le \inf_{P\in\mathcal{P}} \sup_{\tilde{P}\in\tilde{\mathcal{P}}} \|W_\Delta(z)(\tilde{P}(z)P^{-1}(z) - 1)\|_\infty$$

Now the problem of deducing upper bounds of the model error is reduced to a min-max problem. This problem is substantially simplified by the use of the approximate set $\tilde{\mathcal{P}}_c$ at the cost of a little conservatism. Write for the additive model error:

$$\inf_{P\in\mathcal{P}} \sup_{\tilde{P}\in\tilde{\mathcal{P}}} \|W_\Delta(z)(\tilde{P}(z) - P(z))\|_\infty \le$$

$$\inf_{P\in\mathcal{P}} \sup_{\tilde{P}_c\in\tilde{\mathcal{P}}_c} \|W_\Delta(z)(\tilde{P}_c(z) - P(z))\|_\infty =$$

$$\inf_{P \in \mathcal{P}} \| \, | W_\Delta(z) | \, (| \tilde{P}_c(z) - P(z) | + r_c(z)) \, \|_\infty$$

For the multiplicative model error:

$$\inf_{P \in \mathcal{P}} \sup_{\tilde{P} \in \tilde{\mathcal{P}}} \| W_\Delta(z)(\tilde{P}(z) P^{-1}(z) - 1) \|_\infty \leq \inf_{P \in \mathcal{P}} \sup_{\tilde{P}_c \in \tilde{\mathcal{P}}_c} \| W_\Delta(z)(\tilde{P}_c(z) \, P^{-1}(z) - 1)) \|_\infty =$$

$$\inf_{P \in \mathcal{P}} \| \, | W_\Delta(z) | (| P_c(z) P^{-1}(z) - 1 \, | + r_c(z) | P^{-1}(z) |) \|_\infty$$

Consequently, we define the upper bound of the model error for a model with parameter vector θ and for a frequency $z \in \Omega$ as

$$\gamma_{a,\max}(\theta,z) = | P_c(z) - P(\theta,z) \, | + r_c(z)$$

$$\gamma_{m,\max}(\theta,z) = | P_c(z) P^{-1}(\theta,z) - 1 \, | + r_c(z) | P^{-1}(\theta,z) |$$

Now the final problem to solve becomes:

Additive: $$\inf_{P \in \mathcal{P}} \| \, W_\Delta(z) \Delta_a(z) \|_\infty \leq \inf_{\theta \in \Theta} \| W_\Delta(z) \gamma_{a,\max}(\theta,z) \|_\infty$$

Multiplicative: $$\inf_{P \in \mathcal{P}} \| \, W_\Delta(z) \Delta_m(z) \|_\infty \leq \inf_{\theta \in \Theta} \| W_\Delta(z) \gamma_{m,\max}(\theta,z) \|_\infty$$

The problem turns out to be the minimization of the H_∞-norm of a function $W_\Delta(z)\gamma_{a,\max}(\theta,z)$ or $W_\Delta(z)\gamma_{m,\max}(\theta,z)$ over all admissible θ. Note the major drawback of using an H_∞-norm, namely that the cost-criteria $W_\Delta(z)\gamma_{a,\max}(\theta,z)$ and $W_\Delta(z)\gamma_{m,\max}(\theta,z)$ are not differentiable with respect to θ. This means that one cannot directly use a gradient method to search for the minimum. We can solve the problem by using methods which do not need a gradient, e.g., simplex methods and random search based techniques. The problem with these methods, however, is that convergence is not guaranteed if the initial value of θ is far from the optimal value. In that case, we can use estimations from preliminary identifications as initial values.

26.4. ONE-STEP IDENTIFICATION METHOD

The two step identification method, discussed in the preceding section, is based on a graphical approach. It is a straightforward method and transparent in the followed steps. Accuracy, however, is somewhat relaxed by the approximation of the uncertainty set $\tilde{\mathcal{P}}$ by $\tilde{\mathcal{P}}_C$. This approximation can be circumvented by the following one-step method, which uses a more algebraic approach. Start with defining a matrix $Q(z)$ with the true disturbances, scaled by the corresponding weighting filters, on the diagonal:

$$Q(z) = \begin{bmatrix} W_d^{-1}(z)d_t(d) & 0 \\ 0 & W_e^{-1}(z)e_t(z) \end{bmatrix}.$$

Since $|d_t(z)| \le |W_d(z)|$ and $|e_t(z)| \le |W_e(z)|$ a bound for the largest singular value of this scaled disturbance matrix is

$$\sigma_{\max}\{Q(z)\} \le 1 \quad \text{for all } |z| = 1.$$

Notice that (for each $z \in \Omega$)

$$P_t = y_t u_t^{-1} = (y - e_t)(u + d_t)^{-1}.$$

The additive model error becomes

$$\begin{aligned} \Delta_a = P_t - P(\theta) = y_t u_t^{-1} = (y - e_t)(u + d_t)^{-1} - P(\theta) = \\ \left(y - [0 \;\; W_e]Q\begin{bmatrix}1\\1\end{bmatrix}\right)\left(u + [W_d \;\; 0]Q\begin{bmatrix}1\\1\end{bmatrix}\right)^{-1} \\ - P(\theta) = yu^{-1} - P(\theta) \\ + [-yu^{-1}W_d - yW_e]\, Q\left(I - \begin{bmatrix} u^{-1}W_d & 0 \\ u^{-1}W_d & 0 \end{bmatrix} Q\right)^{-1} \begin{bmatrix} u^{-1} \\ u^{-1} \end{bmatrix} \\ = G_{a,1} + G_{a,2}Q(I - G_{a.3}Q)^{-1}G_{a,4} \end{aligned}$$

where G_a is defined as

$$G_a = \begin{bmatrix} G_{a,1} & G_{a,2} \\ G_{a,3} & G_{a,4} \end{bmatrix} = \begin{bmatrix} yu^{-1} - P(\theta) & \begin{bmatrix} -yu^{-1}W_d - yW_e \end{bmatrix} \\ \begin{bmatrix} u^{-1} \\ u^{-1} \end{bmatrix} & \begin{bmatrix} u^{-1}W_d & 0 \\ u^{-1}W_d & 0 \end{bmatrix} \end{bmatrix}.$$

In a shorter notation:

$$\begin{aligned} \Delta_a(z) &= G_{a,1}(z) + G_{a,2}(z)Q(z)(I - G_{a,4}(z)Q(z))^{-1}G_{a,3}(z) \\ &= \mathcal{F}_l(G_a(z), Q(z)) \end{aligned}$$

where $\mathcal{F}_l$ stands for lower linear fractional transformation.

For the multiplicative model error the matrix G_m can be derived in the same way:

$$G_m = \begin{bmatrix} G_{m,1} & G_{m,2} \\ G_{m,3} & G_{m,4} \end{bmatrix} = \begin{bmatrix} yu^{-1}P^{-1}(\theta) - 1 & \begin{bmatrix} -yu^{-1}W_d - yW_e \end{bmatrix} \\ \begin{bmatrix} u^{-1} \\ u^{-1} \end{bmatrix} P^{-1}(\theta) & \begin{bmatrix} u^{-1}W_d & 0 \\ u^{-1}W_d & 0 \end{bmatrix} \end{bmatrix}$$

such that

$$\begin{aligned}\Delta_m(z) &= G_{m,1}(z) + G_{m,2}(z)\tilde{Q}(z)(I - G_{m,4}(z)Q(z))^{-1}G_{m,3}(z)\\ &= \mathcal{F}_l(G_m(z),\tilde{Q}(z)).\end{aligned}$$

Note that the matrices $G_a(z)$ and $G_m(z)$ are built up with known objects: the chosen model $P(\theta, z)$, the data set $\{u(z), y(z)\}$, and the noise bounds $\{W_d(z), W_e(z)\}$. Only the matrix $Q(z)$, which is built up from the true disturbance signals $d_t(z)$ and $e_t(z)$, is unknown. However we do know that it is bounded and that it belongs to a scaled noise set $\tilde{Q}$with elements $\tilde{Q}(z)$ as follows:

$$\tilde{Q} = \{\tilde{Q}(z) \text{ is diagonal and } \bar{\sigma}\{\tilde{Q}(z)\} \leq 1 \text{ for all } z \in \Omega\}$$

Note that matrix $Q(z)$ with the true scaled disturbance signals is in the set $\tilde{Q}$. Consider $\Delta(z)$ to be either $\Delta_a(z)$ or $\Delta_m(z)$ and $G(z)$ is the corresponding G_a or G_m. For every frequency one can bound the magnitude of the model error by

$$|\Delta(z)| = |F_l(G(z),Q(z))| \leq \sup_{\tilde{Q}(z)\in\tilde{Q}} |F_l(G(z),\tilde{Q}(z))|$$

Now we will derive bounds for the magnitude at one specific frequency z. For this specific frequency $G(z)$ and $\tilde{Q}(z)$ are constant complex matrices and will be denoted as G and $\tilde{Q}$. We are looking for the minimum bound γ such that

$$|\Delta| \leq \sup_{\tilde{Q}\in\tilde{Q}} |F_l(G,\tilde{Q})| = \gamma$$

The following result can be used:

$$\sup_{\tilde{Q}\in\tilde{Q}} |F_l(G,\tilde{Q})| = \gamma \ \text{ iff } \mu\left(\begin{bmatrix} \gamma^{-1}G_{11} & \gamma^{-1}G_{12} \\ G_{21} & G_{22} \end{bmatrix}\right) = 1$$

where μ is the structured singular value as defined by Balas *et al.*(16) This value μ is difficult to compute, but we can give an upper bound, which is usually very close to the real value, but somewhat greater, and can be calculated using a convergent algorithm.(16) We can adjust the γ in an iterative way until the approximate μ equals one. This leads to a γ which is very close to the wanted supremum and in any case greater than this supremum. In this way, an upper bound γ for the matrix norm of the model error is derived for all frequencies $z \in \Omega$, which results in a frequency-dependent bound $\gamma(z)$ for a specific model $P(z)$.

By defining a model set with models $P(\theta, z)$ the bound becomes $\gamma(\theta, z)$. Now define the criterion

$$J_\infty(\theta) = \|W_\Delta(z)\gamma(\theta,z)\|_\infty$$

This function is an upper bound for the weighted H_∞-norm of the true model error, so

$$||W_\Delta(z)\Delta(z)||_\infty \leq J_\infty(\theta).$$

Similar to the criterion in the preceding section, $J_\infty(\theta)$ can be minimized using methods that do not need gradients.

26.5. SIMULATION EXAMPLE

This section presents a simulation example. A second order simulation model

$$P_t = \frac{z^2 - 1.1z + 0.24}{z^2 - 1.6z + 0.68} = \frac{(z-0.3)(z+0.8)}{(z-0.8+j0.2)(z+0.8-j0.2)}$$

is excited by an input signal $u(k)$ in a configuration of Fig. 26.2 and output $y(k)$ is measured. A Bode-plot of $P_t(e^{jw})$ is given in Fig. 26.6(a).

Input signal $u(k)$ is generated for 1024 samples. Care has been taken that it is persistently exciting and that the errors due to the discrete Fourier transformation are negligible. The control input and measured output signal are corrupted by additive white Gaussian noise $d(k)$ and $e(k)$, respectively. The 3σ-bounds in the frequency domain provide values for $W_d(z)$ and $W_e(z)$, and so W_d and W_e are constants. This results in the following values for the noise to signal ratios:

$$\frac{|W_d|}{|u(z)|} \leq 0.11 \text{ and } \frac{|W_e|}{|y(z)|} \leq 0.16.$$

We do a simulation experiment and obtain a data-set $\{u(k),y(k)\}$. The computations for the model error bounds are only done on a limited number of frequency points z_i in the frequency set $\Omega = \{z_1,z_2,\ldots,z_{512}\}$ with $z = e^{j\pi i/512}$, $i = 1,..512$. For all frequencies z_i we calculate $P_c(z_i)$ and $r_c(z_i)$, using the simple circular bounds, and we obtain the regions as in Fig. 26.6(b).

First consider an additive model error structure and in which we must optimize the function

$$\gamma_{opt} = \inf_{\theta\in\Theta} ||\, W_\Delta(z)\gamma_{a,\max}(\theta,z)||_\infty$$

where the weighting filter is chosen $W_\Delta(z) = 1$.

In a first run, choose the model set $\mathcal{P}$ consisting of all first order functions

$$P(z) = \frac{\theta_2 z + \theta_3}{z + \theta_1}$$

so $P_t(z)$ is not in the model set $\mathcal{P}$. Use the two-step algorithm to find an optimal $\theta_{opt} = [-0.275\ 0.895\ 0.757]$ where $\gamma_{opt} = 0.936$. In Fig. 26.11(a) the plots of the true

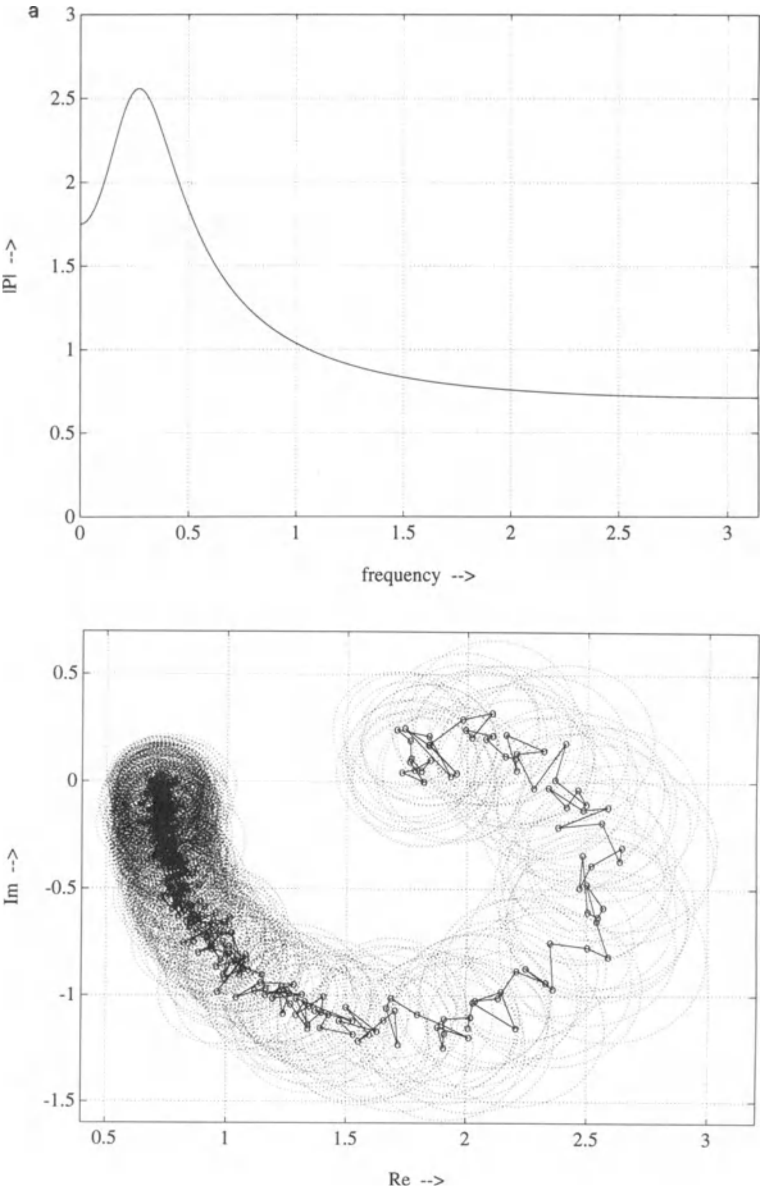


FIGURE 26.6. (a) Bode plot of $P_t(z)$. (b) Uncertainty regions.

process $P_t(z)$, the optimal model $P(z)$ and the region center points $P_c(z)$ are given in the complex plane.

Now define three functions:

$$\gamma_{a,\max}(\theta,z) = |\, P_c(z) - P(\theta,z)| + r_c(z)$$

$$\gamma_{a,\text{med}}(\theta,z) = |\, P_c(z) - P(\theta,z)|$$

$$\gamma_{a,\min}(\theta,z) = \max\,(0, |\, P_c(z) - P(\theta,z)| - r_c(z))$$

The function $\gamma_{a,\max}(\theta,z)$ gives an upper bound for the model error, the function $\gamma_{a,\min}(\theta,z)$ is the minimum distance between $P(z)$ and $\tilde{P}(z)$, and so gives a lower bound. The function $\gamma_{a,\text{med}}(\theta,z)$ gives the distance of $P(z)$ to the center of the region $\tilde{P}(z)$ and is centered between the upper and lower bounds. In Fig. 26.7 the functions $\gamma_{a,\max}(\theta,z)$, $\gamma_{a,\text{med}}(\theta,z)$, $\gamma_{a,\min}(\theta,z)$ and the true model error $|\Delta_a(z)|$ are plotted for the estimated model. In this example the lower bound $\gamma_{a,\min}(\theta,z)$ is larger than zero for nearly all frequencies, which indicates that the nominal model that is found cannot describe the system accurately. Note that these estimates $\gamma_{a,\max}$, $\gamma_{a,\text{med}}$ and $\gamma_{a,\min}$ can always be calculated and be used for defining a weighting filter W_Δ in the next iteration. For a better model for higher frequencies choose a filter that emphasizes the error in the higher frequencies. So $W_\Delta(z)$ is large for higher frequencies, and small for the lower frequencies. Therefore, we define a highpass filter as a weighting filter

$$W_\Delta(z) = \frac{z + 0.16}{z + 0.7}$$

For this choice of weighting filter find $\theta_{opt} = [-0.402\ \ 1.240\ \ 0.190\ ]$ where $\gamma_{opt} = 1.031$. Figs. 26.8 and 26.11(b) give the results for the estimated model. Compare the curves of the unweighted case in Fig. 26.7 with the weighted case in Fig. 26.8.

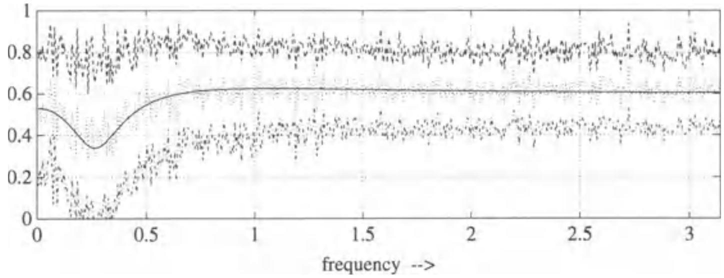


FIGURE 26.7. True additive model error with bounds (1st order, no weight); Δ_t(——), $\gamma_{\max}$(– – –), γ_{med}(···), and $\gamma_{\min}$(– · –).

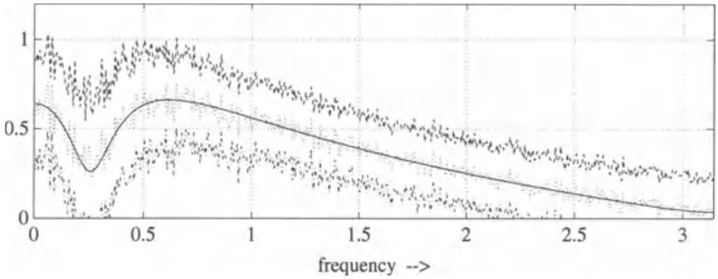


FIGURE 26.8. True additive model error with bounds (1st order, high freq. weight); Δ_t(———), $\gamma_{\max}$(– – –), γ_{med}(···), and $\gamma_{\min}$(– · –).

Note that the model error decreased very much for the higher frequencies, at the cost of a small increase at the lower frequencies.

In a second run, choose the model set $\mathcal{P}$ consisting of all second order functions

$$P(z) = \frac{\theta_3 z^2 + \theta_4 z + \theta_5}{z^2 + \theta_1 z + \theta_2}.$$

Now $P_t(z)$ is in the model set $\mathcal{P}$and, as a weighting filter, $W_\Delta(z) = 1$. Find an optimal $\theta_{opt} = [-1.589\ 0.679\ 1.021\ -1.116\ 0.259]$ where $\gamma_{opt} = 0.536$. Figures 26.9 and 26.11(c) give the results for the estimated model. The lower bound $\gamma_{a,\min}(\theta,z)$ in this example is exactly zero, which indicates that the found nominal model might indeed describe the system exactly.

In a third run, choose the model set $\mathcal{P}$consisting of all third order functions

$$P(z) = \frac{\theta_4 z^3 + \theta_5 z^2 + \theta_6 z + \theta_7}{z^3 + \theta_1 z^2 + \theta_2 z + \theta_3}$$

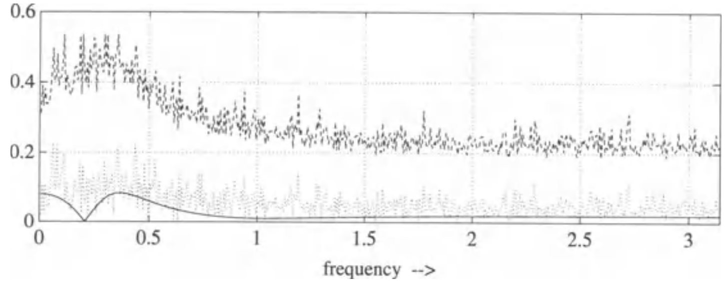


FIGURE 26.9. True additive model error with bounds (2nd order); Δ_t(———), $\gamma_{\max}$(– – –), and γ_{med}(···).

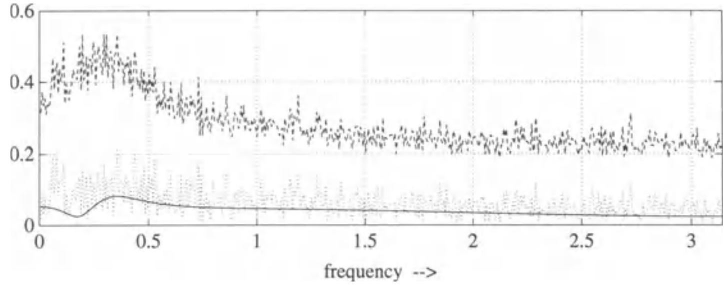


FIGURE 26.10. True additive model error with bounds (3rd order); Δ_t(———), $\gamma_{\max}$(– – –), and γ_{med}(···).

so $P_t(z)$ is in the model set and, as a weighting filter, $W_\Delta(z) = 1$. The result is θ_{opt} = [–1.610 0.703 –0.0101 1.038 –1.157 0.261 0.0063] where $\gamma_{opt} = 0.535$. Figs. 26.10 and 26.11(d) give the results for the estimated model. Also, in this case the lower bound $\gamma_{a,\min}(\theta, z)$ in this example is exactly zero. However, the upper bound for the model error is not decreased much, so it looks as if a second order

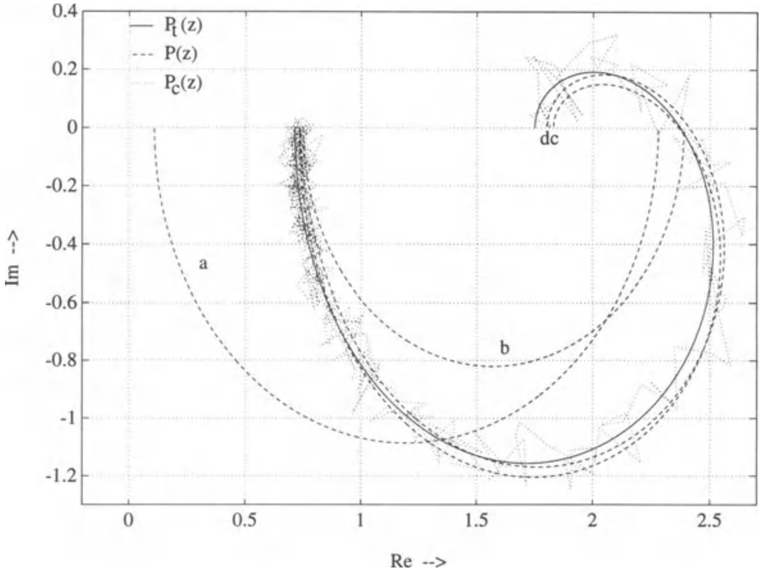


FIGURE 26.11. $P_t(z)$,$P(z)$ and various models $P_c(z)$ in the complex plane: a, additive model error, 1st order model without weighting, b, additive model error; 1st order model with weighting on higher frequencies, c, additive model error; 2nd order model without weighting; and d, additive model error, 3rd order model without weighting.

model will satisfy in this case (as expected). The parameters θ_3 and θ_7 are both nearly zero, which results in a nearly pole-zero cancellation at $z = 0$.

26.6. CONCLUSIONS

This chapter considered two methods for the identification of SISO-systems in terms of a minimum additive and multiplicative error bound. An inherent condition of the problem is that the signal-to-noise ratio is sufficiently small, otherwise the model error bounds become very large, and the proposed methods can possibly fail.

Using the two-step identification method, we calculate uncertainty regions and fit the model in H_∞-norm sense. Minimum, maximum, and medium errors give an indication about the adaptation of the weighting filter $W_\Delta(z)$ in the next iteration, and whether the model can represent the system. The two-step method is easy to understand and it yields a lot of insight. The model error bound for a specific model can be computed analytically.

The one-step identification method uses the structured singular value μ, which has to be computed in an iterative procedure. The one-step identification method, therefore, needs more computation time than the two-step method, but the one-step method generally leads to a smaller model error bound than the two-step method.

REFERENCES

1. J. C. Doyle and G. Stein, *IEEE Trans. Autom. Control* **AC-26**, 4 (1981).
2. J. C. Doyle, B. A. Francis, and A. R. Tannenbaum, *Feedback Control Systems*, MacMillan Publishing Company, New York (1992).
3. G. C. Goodwin and M. E. Salgado, *Int. J. Adapt. Control Signal Proc.* **3**, 333 (1989).
4. M. Gevers, *Proceedings of the 9th IFAC/IFORS Symposium on Identification and System Parameter Estimation,* Budapest, Hungary, pp. 1–10 (1991).
5. Y.-C. Zhu, *Int. J. Control,* **49**, 2241 (1989).
6. B. Wahlberg and L. Ljung, in: *Proceedings of the European Control Conference*, Grenoble, France (1991).
7. A. J. Helmicki, C. A. Jacobsen, and C. N. Nett, *IEEE Trans. Autom. Control* **AC-36**, 1163 (1991).
8. R. O. LaMaire, L. Valavani, M. Athans, and G. Stein, *Automatica* **27**, 23 (1991).
9. G. Gu and P. P. Khargonekar, *Automatica* **2**, 229 (1992).
10. T. J. J. van den Boom, M. H. Klompstra, and A. A. H. Damen, *Proceedings of the 9th IFAC/IFORS Symposium on Identification and System Parameter Estimation,* Budapest, Hungary, pp. 1431–1436 (1991).
11. T. J. J. van den Boom, *System Identification of MIMO-Systems for H_∞ Robust Control: Technical Report, CUED/F-INFENG/TR.88,* Cambridge University, Cambridge, U.K. (1991).
12. T. J. J. van den Boom, in: *Proc. ACC,* Chicago, IL pp. 1248–1252 (1992).
13. T. J. J. van den Boom, *MIMO System Identification for H_∞ Robust Control: A Frequency Domain Approach with Minimum Error Bounds*, Ph.D. thesis, Eindhoven University of Technology, The Netherlands (1993).

14. T. J. J. van den Boom, in: *Proceedings of the 12th IFAC World Congress*, Sydney, Australia (1993).

15. J. B. Cruz Jr., J. S. Freudenberg, and D. P. Looze, *IEEE Trans. Autom. Control* **AC-26**, 66 (1981).

16. G. J. Balas, J. C. Doyle, K. Glover, A. Packard, and R. Smith, μ *Analysis and Synthesis Toolbox (μ-tools): Matlab Functions for the Analysis and Design of Robust Control Systems*, version 1.0a. The MathWorks Inc., Natick, Mass. (1993).

NOTATION

$\mathcal{R}$ = Set of all real rational transfer functions
$\mathcal{S}$ = Set of all stable real rational transfer functions
P_t = rue process transfer ($P_t \in \mathcal{S}$)
u_t, y_t = rue input and output signal
d_t, e_t = True input and output disturbance signal
P_C, r_C = Centerpoint and radius of uncertainty set $\tilde{\mathcal{P}}_C$
$\tilde{\mathcal{P}}$ = Set with all possible process transfers P
$\tilde{\mathcal{P}}_C$ = Enclosing set for $\tilde{P}$ with elements $\tilde{P}_C$
$\mathcal{P}$ = Model set with elements P ($\mathcal{P} \subset \mathcal{S}$)
$\tilde{\mathcal{U}}$ = Set with all possible input signals $\tilde{u}$
$\tilde{\mathcal{Y}}$ = Set with all possible output signals $\tilde{y}$
$\tilde{\mathcal{D}}$ = Set with all possible input noise signals $\tilde{d}$
$\tilde{\mathcal{E}}$ = Set with all possible output noise signals $\tilde{e}$
Ω = Set of unit circle samples $z_i = e^{j\pi i/N}$, $i = 0, \ldots, N-1$

27

Estimation of Mobile Robot Localization: Geometric Approaches

D. Meizel, A. Preciado-Ruiz, and E. Halbwachs

27.1. PRACTICAL PROBLEM POSITION

27.1.1. Introduction

The real device is shown in Fig. 27.1. It is a cart-like wheeled vehicle sketched on Fig. 27.2. It is capable to perform planar displacements and its configuration q (Eq. 27.1) is composed of the 2-D coordinates $(x_c y_c)$ of a characteristic point together with the orientation θ defined in a world coordinate $\mathcal{W}$ (Fig. 27.2).

$$q = (x_x, y_c, \theta)^T \tag{27.1}$$

The purpose of such a vehicle is typically to move from one initial configuration to a goal configuration along a planned path while avoiding unexpected obstacles. The possibility to define displacement missions implies an *a priori* knowledge of the world. At least, it must be possible to designate where the robot should move in such a way that the mission completion can be recognized by the

D. Meizel and E. Halbwachs • Heudiasyc, CNRS, Compiegne Technology University, 60206 Compiegne, France. A. Preciado-Ruiz • ITESM–Campus Toluca, Toluca, Edo. de Mexico, Mexico.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

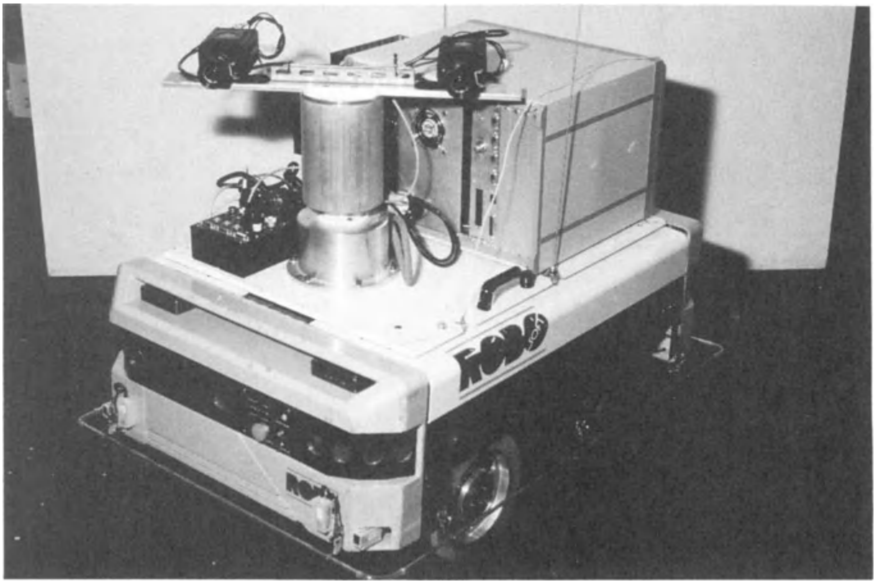


FIGURE 27.1. The real robot.

autonomous vehicle's own means of perception. Performing the localization of this vehicle consists in the evaluation of its configuration q (Eq. 27.1) in a map.

This is realized by using a map where are listed the major obstacles, beacons and landmarks and where the goal and some major passing points can be referenced with respect to sensible elements. Given an initial configuration, and the ultimate and intermediate goals together with the map, a global planner computes a feasible, generally minimum length, and collision-free path along which the vehicle can

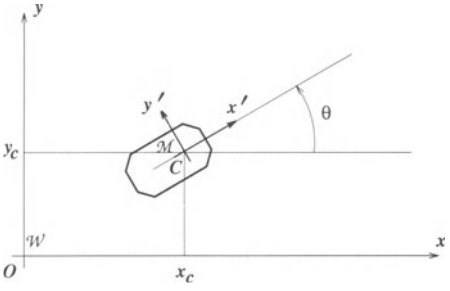


FIGURE 27.2. The configuration space.

move and reach its goal(s). At the execution level, several necessarily non-ideal features of the real world should be taken into account:

- the world map is inaccurate: some elements are missing; some other have disappeared; and the relative positions of beacons and landmarks are not *exactly* described in the map;
- the exact vehicle configuration is not exactly known and its precision should be increased by measurements performed by on-board sensors.

These basic statements have the following consequences:

1. The goal and more generally the reference path should be referenced with respect to sensible elements of the map. This implies that the configuration should be defined at any instant with respect to a local map composed of both the beacons and landmarks set that are expected to be sensed by the robot from its current localization and the *a priori* known obstacles set (that may not be detected as, for instance, a glass-wall not seen by a vision system).[1]
2. Some obstacles are not *a priori* known. The consequence on the robot control architecture is that an obstacle avoidance control level should exist

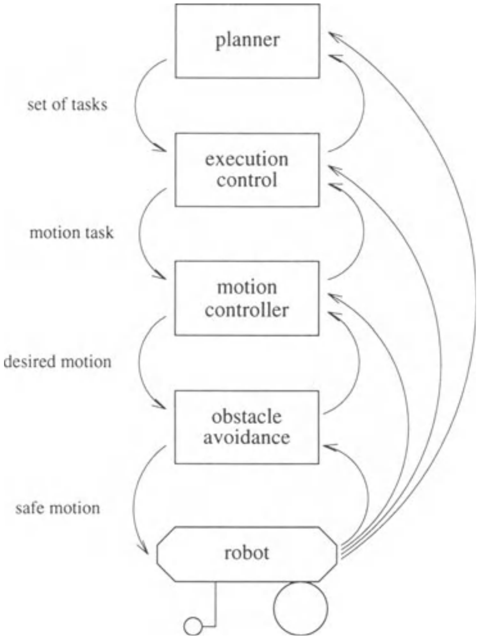


FIGURE 27.3. Different control levels.

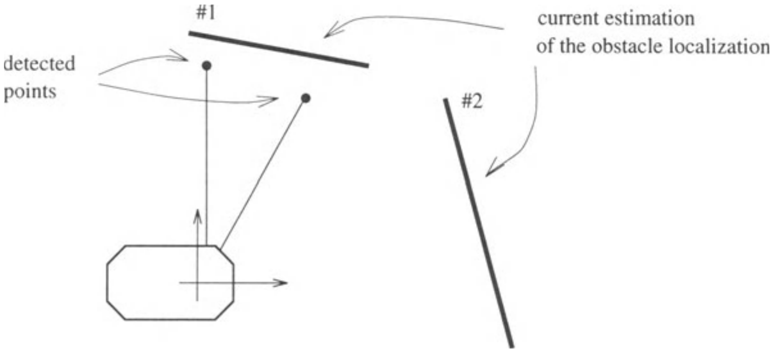


FIGURE 27.4. Matching problem: Does the detected obstacle points match with segment #1 or #2 or is it an outlier?

between the motion control level and the actuators (Fig. 27.3). The obstacle avoidance level does not interpret the measurements in terms of elements of the local map. It just filters the desired motions in such a way as to prevent collision with sensed obstacles, whatever they can be.

3. The existence of obstacles that are not *a priori* known in the map implies that the localization should be capable of discrimination between measurements coming from elements of the map that are usable for the localization and unexpected obstacles that cannot be associated to any landmark or beacon.

Summing up this introductory discussion, it appears that the robot motion should be defined at the planning level in closed loop form by using a configuration estimate defined inside a local map containing both the (intermediate or ultimate) goals, the landmarks, and beacons and the nondetectable obstacles. This closed loop motion is completed by an obstacle avoidance module whose reflexive actions do not imply any interpretation of the on-line measurements. Complementarily, the localization estimation algorithm is thus split in two aspects:

- the former (matching module) consists to associate a measurement to a given landmark or to reject it as unusable for localization purpose (Fig. 27.4),
- the latter (estimation module) treats this matched measurement in order to improve the localization estimation accuracy (Figs. 27.5(a), (b), and (c)).

27.1.2. Basic Localization Principle and Notations

The sequel focuses on the treatment of asynchronous discrete-time telemetric measurements combined with a continuous odometric update of the configuration.

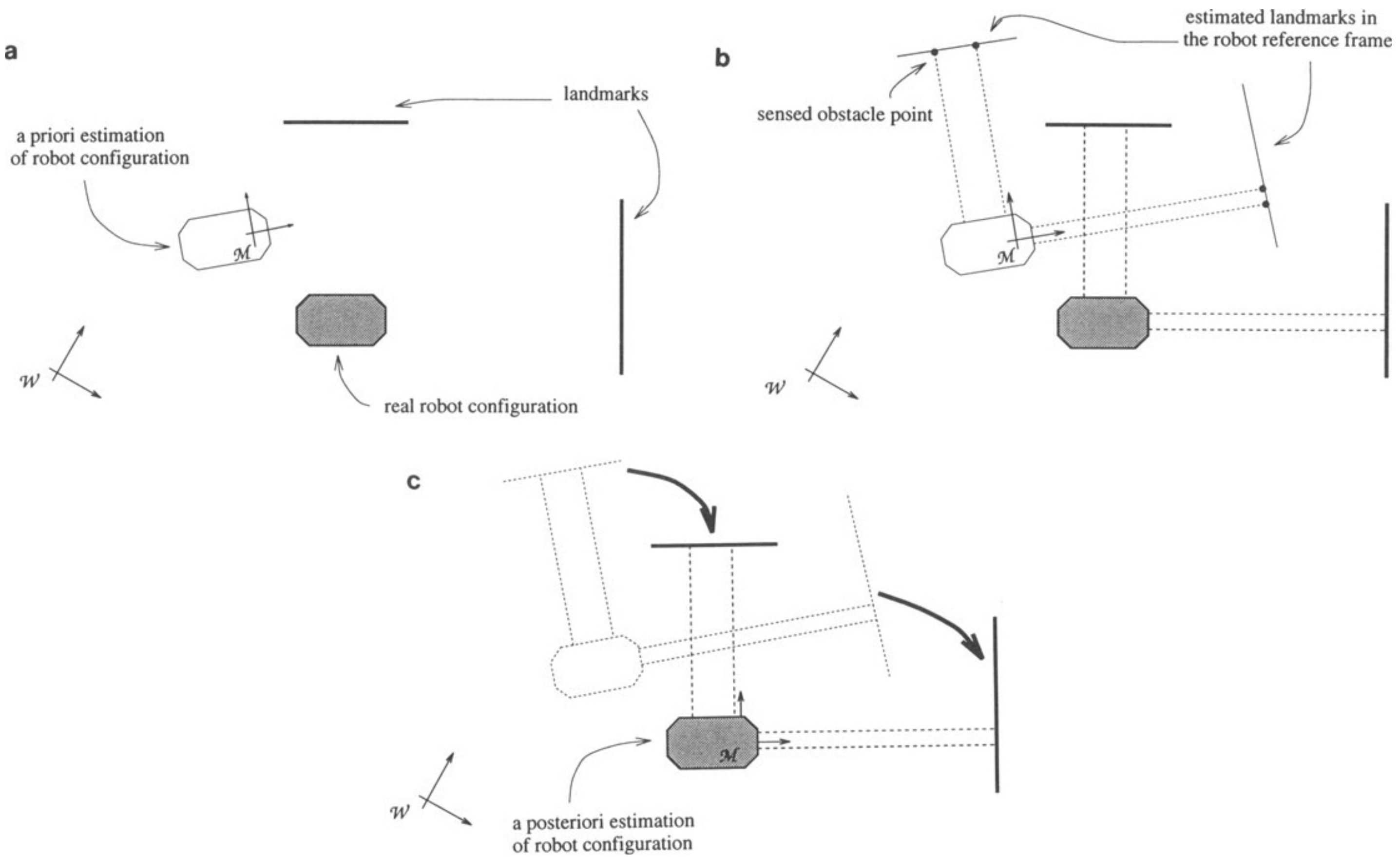


FIGURE 27.5. Exteroceptive measurements.

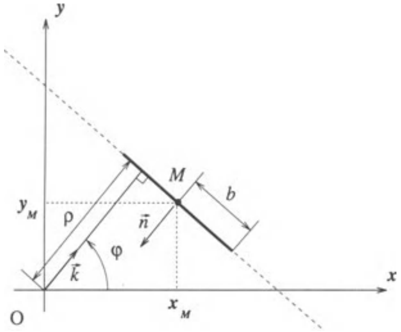


FIGURE 27.6. Two-dimensional line segment.

In addition, the vehicle moves in indoor environment where obstacles and landmarks are modeled as polygonal objects. Maps are then composed of a straight segments primitives, each one defined (Fig. 27.6) by its descriptor $\mathcal{S}$:

$$\mathcal{S} = \{M, b, \varphi, \rho, s\}. \tag{27.2}$$

This descriptor contains the following items:

- center $M(x_M, y_M)$;
- half length b;
- line equation: $x\cos(\varphi) + y\sin(\varphi) - \rho = 0$; and
- external normal vector direction: $\vec{n} = s\vec{k}$, $s = \pm 1$.

A telemeter measures the presence of an obstacle in a given direction by reflection of an acoustic or electromagnetic wave. The configuration of such a sensor is given by (x_t', y_t', α') and the measurement result is (d, α') (see Fig. 27.7). The measurement precision is further analyzed in Section 27.3.1.

The dead reckoning system is tied to the actuators. It integrates elementary translations δs and elementary rotations $\delta\theta$ given by rotary encoders tied to the driving wheels. It updates the configuration of Eq. (27.3) along the motions of the vehicle:

$$\begin{cases} x_c := x_c + \cos(\theta)\delta s \\ y_c := y_c + \sin(\theta)\delta s \\ \theta := \theta + \delta\theta \end{cases} \tag{27.3}$$

T_s denoting the sampling time, the translational (v), and rotational (ω) speeds are estimated as follows:

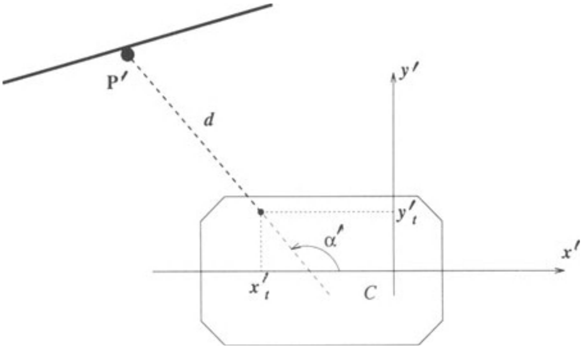


FIGURE 27.7. Telemeter measure.

$$v := \delta s / T_s$$

$$\omega := \delta\theta / T_s$$

The integration of dead reckoning with the telemetric measurement based localization is schematically represented in Fig. 27.8.

From a qualitative point of view matching and estimation are done as follows:

1. Get one (or more) telemetric measurements P' in the mobile frame $\mathcal{M}$(Fig. 27.7).

$$P' = \begin{bmatrix} x' = x'_t + d\cos(\alpha') \\ y' = y'_t + d\sin(\alpha') \end{bmatrix} \tag{27.4}$$

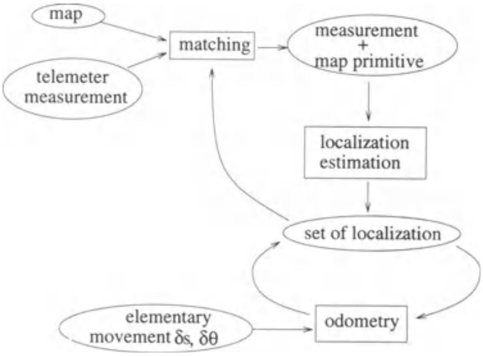


FIGURE 27.8. Exteroceptive localization and dead reckoning cooperation.

2. Update the local map in the mobile frame $\mathcal{M}$ by using the odometry. The result is a list of obstacle descriptors Eq. (27.2) $\mathcal{S}_i'$:

$$\mathcal{S}_i' = \{(M_i', b_i, \varphi_i', \rho_i', s_i);\; i = 1 \ldots n\}$$

3. Associate the measurement P′ with the most appropriated obstacle descriptor or reject it as an outlier.
4. Update the estimation of the matched line segment (φ_i',ρ_i') in the mobile frame $\mathcal{M}$. The error equation (27.5) states that the detected point (x',y') belongs to the ith obstacle:

$$x' \cos(\varphi_i') + y' \sin(\varphi_i') - \rho_i' = 0. \tag{27.5}$$

The result of parameter updating is

$$\{\hat{\varphi}_i', \hat{\rho}_i'\}.$$

Knowing the obstacle equation in the world frame

$$x \cos(\varphi_i) + y \sin(\varphi_i) - \rho_i = 0 \tag{27.6}$$

and the estimation of the obstacle orientation $(\hat{\varphi}')$ in the mobile frame,

$$x' \cos(\hat{\varphi}_i') + y' \sin(\hat{\varphi}_i') - \rho_i' = 0 \tag{27.7}$$

gives the update the vehicle orientation

$$\hat{\theta} = \varphi_i - \hat{\varphi}_i' \tag{27.8}$$

Update the vehicle position (x_c,y_c) by use of the error Eq. (27.9) which represents the fact that the middle M_i of the segment $\mathcal{S}_i$ satisfies the line segment Eq. (27.6) in the world frame.

$$(x_{M_i} - x_c) \cos(\varphi_i) + (y_{M_i} - y_c) \sin(\varphi_i) - \hat{\rho}_i' = 0 \tag{27.9}$$

In the next sections, the measurement uncertainties are taken into account. Extended Kalman filtering (EKF) is the more commonly used solution framework to attack this problem. The major points of this type of solution are first presented in Section 27.2. EKF implies, among other things, the assumption that the sources of error are Gaussian white-noises. This hypothesis is generally forgotten at the execution level and never checked.

Modeling the measurement errors by simply stating the bounds of this error leads to set membership solutions. In this latter class, the use of elliptical algorithms are detailed and discussed in the Section 27.3.

27.2. OUTLINES OF EKF BASED SOLUTIONS

EKF has been used in a sequence of papers[2,3] where the general framework of the solution, based upon a general framework[4] is the following. Consider the state equation (27.3) of the mobile. Assume that noise corrupts the motion. The following discrete time equation describes this by:

$$X_{k+1} = F(X_k, u_k) + v_k \tag{27.10}$$

where

- The state vector $X_k = (x_{c,k}\,;\, y_{c,k}\,;\, \theta_k)^T$ is the localization at the kth sampling time t_k,
- $u_k = (\delta s_k, \delta\theta_k)^T$ is the considered control input,
- v_k is a Gaussian zero-mean white noise ($v_k \sim N(O, Q_k)$).

Along with this definition, observations are noted in the usual form:

$$z_k = h(S_i, X_k) + w_k \tag{27.11}$$

where z_k is the measurement signal, $h(.,\,.)$ is the sensor model, S_i denotes the observed beacon or landmark, and w_k is a zero-mean Gaussian white noise ($w_k \sim N(O, R_k)$) too.

The deterministic part of the model is clearly defined. The covariance matrix R_k of w_k states the sensor precision in a statistic framework but the definition of the covariance matrix Q_k of v_k is not clear. Again, the assumption that noises are white is never checked *a posteriori*.

The localization procedure is then classically the following:

Suppose a matched measurement, for instance, the relative position of an obstacle point $z_k = (x_k^i, y_k^i)^T$ (Eq. (27.4)) is matched to a line segment S_i. The estimation is performed by the following sequence

- Estimation step:
  - — innovation: $v_k^i = z_k - h(S_i, \hat{X}_{k|k-1})$
  - — Kalman gain computation: $\begin{cases} S_k^i = \nabla h(.) P_{k|k-1} \nabla^T h(.) + R_k \\ K_k = P_{k|k-1} \nabla^T h(.) S_k \end{cases}$
  - — estimate actualization: $\hat{X}_{k|k} = \hat{X}_{k|k-1} + K_k v_k^i$
  - — covariance actualization: $P_{k|k} = P_{k|k-1} - K_k S_k^i K_k^T$
- Prediction step:
  - — estimate prediction: $\hat{X}_{k+1|k} = F(\hat{X}_{k|k,u_k})$
  - — covariance prediction: $P_{k+1|k} = \nabla F(.) P_{k|k} \nabla^T F(.) + Q_k$.

This procedure is algorithmically complete provided the covariances matrices (Q_k, R_k) and the initial dispersion matrix $P_{-1,-1}$ are given. Matching can be done in this context by defining a *validation gate* based upon a Mahalanobis distance (Eq. (27.12)) defined as follows:

$$d_k^i = v_k^{iT} S_k^i v_k^i. \tag{27.12}$$

A measurement z_k can be matched with a line segment S_i if the following inequality is satisfied. In this expression, χ_0^2 can be interpreted under the assumption that the innovations are white, as a probability of association between the measurement and its prediction:

$$d_k^i < \chi_0^2. \tag{27.13}$$

From a "pictural" point of view, the choice of a specific value of the threshold χ_0^2 describes an ellipsoid of matchable measurements in the observation space centered around the prediction $h(S_i,\hat{X}_k)$ of the measurement. If several primitives in the local map can be matched with a given measurement, a common rule consists of choosing one that minimizes the criterion:

$$d_k^i + \ln(\det(S_k^i)). \tag{27.14}$$

As a conclusion, this procedure works in numerous cited examples. The tumbling stone of the method is, besides general comments upon the convergence of EKF, a sort of gap existing between the modeling of the sensor and the definition of variance/covariance matrices (Q_k, R_k) (Eqs. (27.10 and 27.12)) which are, in most cases, simple parameters to be tuned in the procedure rather than statistical attributes. Additionally, the Gaussian white noise assumption on the causes of error is only exceptionally addressed.(5) The following section proposes to reconsider this problem from the bounded error point of view. It gives, by using elliptical algorithms, some solutions that are in some ways similar to the one developed from EKF. Additional features are a dead-zone and a measurement consistency test.

27.3. A SET MEMBERSHIP APPROACH TO THE STATIC LOCALIZATION PROBLEM

After having stated the outlines of the classical way to attack the localization problem that consists, in a few words, to model any uncertainty as the realization of a Gaussian white noise, another way to state the lack of precision in the estimation process is presented here.

In the set membership approach, uncertainty in the estimation of a quantity is described by a set of possible values rather than with an accuracy statement (a covariance matrix) of an estimate. Introducing the set membership approach in the localization process has been proposed[(6–8)] and discussed in the wider context of task-directed sensor fusion.[(9)]

Set membership solutions are linked with bounded error characterization of inaccuracy. One advantage of such an approach is that it is not necessary to invoke the law of the large numbers when only few measurements are available. Another nice feature consists in the "natural" definition of the error bounds as shown next.

The presentation of the set membership solution is structured as follows:

Section 27.3.1. analyzes the telemetric measurement process in order to characterize the basic measurement error which is found to be "naturally" bounded. This error statement is then used in the next section dealing with the static localization problem which itself is divided in two subsections: the former (Section 27.3.2) is devoted to the localization estimation and the latter (Section 27.3.3) to the matching problem. Finally, the movement is taken into account in the last section (Section 27.3.4), where measurements and movement are mixed.

27.3.1. Measurement Error Statement

The error-free localization principle has been exposed in Section 27.1.1 (Eqs. (27.4–27.9) in the context of perfect telemetric measurements. Here, we take into account the conic dispersion of the wave emitted by a telemeter, which constitutes a major cause of error. The standard output of such a device is the position (x', y') of a detected obstacle point (Eq. (27.4)) in the mobile robot frame $\mathcal{M}$, whereas the rough measurement (d,α') is composed of both the measured distance d between the sensor and the detected obstacle (Fig. 27.9), and the scanning direction angle α'.

The emission angle γ that defines the aperture of the emitted beam is given by the manufacturer. This information can be used for localization by simply stating that the point (x',y') detected in the mobile frame belongs to the segment $\mathcal{S}$ whose line equation (27.2) is defined in the world frame (Fig. 27.7) as follows:

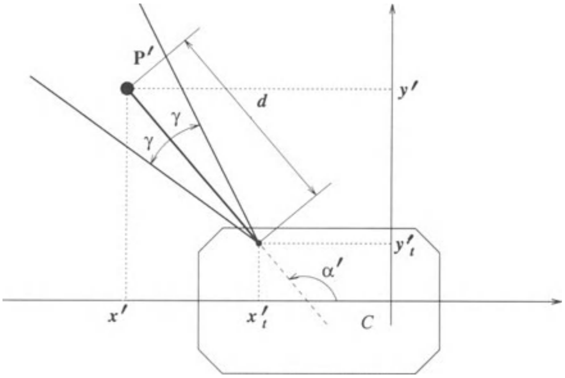


FIGURE 27.9. Telemetric measurement characterization.

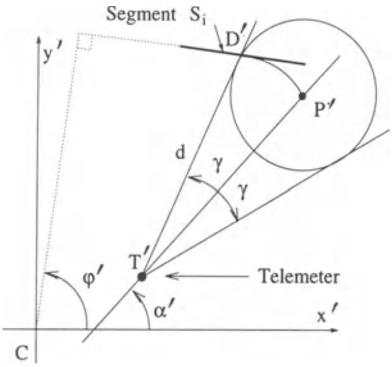


FIGURE 27.10. Sensor model.

$$x_c \cos(\varphi) + y_c \sin(\varphi) + x' \cos(\varphi - \theta) + y' \sin(\varphi - \theta) - \rho' = 0 \qquad (27.15)$$

The measurement error analysis stems from the fact that the beam emitted by the telemeter is conic. The detected distance is the distance of the nearest object in the cone whose normal is such that the reflected beam reaches the receiver/emitter sensor T' (Fig. 27.10).

Without any further information, this distance d between the detected point D' and the sensor T' is interpreted to be the one between the receiver T' and a point $P' = (x',y')$ on the cone axis. Such a point is considered as the standard measurement. It is clear that any real detected point D' lies in the disk of center P' and of radius $\beta(d)(\beta(d) = d\tan(\gamma))$ (Fig. 27.10). With this error assessment, it can be shown that the linear localization equation (27.15) is replaced by the following inequality. It states an admissible strip for the point C:

$$|x_c \cos(\varphi) + y_c \sin(\varphi) + \cos(\varphi - \theta)\, x'$$

$$+ \sin(\varphi - \theta)\, y' - \rho' \,| \le \beta(d) = d \tan(\gamma) \qquad (27.16)$$

In conclusion, the two measurement equations (27.7) and (27.8) are converted into their respective bounded error Eqs. (27.17 and 27.18) where the error bound $\beta(d) = d\tan(\gamma)$ is defined at each measurement:

$$|x'_c \cos(\hat{\varphi}_i') + y' \sin(\hat{\varphi}_i') - \rho_i'| \le \beta(d) \qquad (27.17)$$

and

$$|(x_{M_i} - x_c) \cos(\varphi_i) + (y_{M_i} - y_c) \sin(\varphi_i) - \rho_i'| \le \beta(d). \qquad (27.18)$$

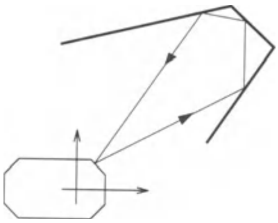


FIGURE 27.11. Multiple reflexion phenomenon.

Another parasitic effect is the multiple reflection phenomenon stemming from specular reflection and represented in Fig. 27.11. This phenomenon is dealt with by the matching module (Section 27.3.3) which should remove data coming from multiple reflections.

27.3.2. The Static Localization Procedure

It simply consists in adapting a standard(10,11) EPC* algorithm to the preceding localization problem (Section 27.1.1) and error Eq. (27.16) that states an admissible band for the point C: Recall the EPC algorithm:

- $\Theta \in \mathcal{R}^p$ is the parameter vector to be estimated;
- $y_k \in \mathcal{R}$, $\Phi_k \in \mathcal{R}^p$ are measurements or known quantities;
- $y_k - \Phi_k^t.\Theta = 0$ is the measurement principle;
- $|y_k - \Phi_k^t.\Theta| \leq \beta_k$ states the measurement error bound;
- $\mathcal{E}_k = \{\Theta \in \mathcal{R}^p;(\Theta - \Theta_k^c)^t P_k^{-1}(\Theta - \Theta_k^c) \leq 1\}$is the feasible parameter set estimate; and
- $\mathcal{E}_0$ defined by (Θ_0^c,P_0^{-1}) is sufficiently large to certainly contain the true parameter vector Θ^*.

The kth feasible domain $\mathcal{E}_k$ is recursively obtained as follow: Iteration k: Measure y_k, Φ_k and compute the two indicators

$$a_k^- = \frac{\Phi_k^T \Theta_{k-1}^c - y_k - \beta_k}{\sqrt{\Phi_k^T P_{k-1} \Phi_k}}$$

and

$$a_k^+ = \frac{y_k - \Phi_k^T \Theta_{k-1}^c - \beta_k}{\sqrt{\Phi_k^T P_{k-1} \Phi_k}}$$

*Elliptical with Parallel Cuts

Test 1: If $a_k^- > 1$ or $a_k^+ > 1$, then $\mathcal{E}_k$ is empty, else continue. Replace a_k^- by $\max(a_k^-, -1)$ and a_k^+ by $\max(a_k^+, -1)$.

Test 2: If $a_k^+ a_k^- \geq 1/p$ (p is the size of vector Θ) then $\mathcal{E}_k = \mathcal{E}_{k-1}$, else

$$\Theta_k^c = \Theta_{k-1}^c + \frac{\sigma_k(a_k^+ - a_k^-)}{2\sqrt{\Phi_k^T P_{k-1} \Phi_k}} P_{k-1}\Phi_k \tag{27.19}$$

and

$$P_k = \delta_k \left(P_{k-1} - \frac{\sigma_k}{\Phi_k^T P_{k-1} \Phi_k} P_{k-1}\Phi_k \Phi_k^T P_{k-1} \right), \tag{27.20}$$

where

$$\delta_k = \frac{p^2}{p^2 - 1}\left(1 - \frac{(a_k^+)^2 + (a_k^-)^2 - \rho_k/p}{2} \right),$$

and

$$\sigma_k = \frac{1}{p+1}\left(p + \frac{2}{(a_k^+ - a_k^-)^2}(1 - a_k^+ a_k^- - \rho_k/2) \right), \tag{27.21}$$

with

$$\rho_k = \sqrt{4(1 - (a_k^-)^2)(1 - (a_k^+)^2) + p^2((a_k^+)^2 - (a_k^-)^2)^2}.$$

When $a_k^+ = a_k^- = a_k$, σ_k as written in Eq. (27.21) is no longer defined. Eqs. (27.19–27.20) then specialize into the centrally symmetric parallel-cut algorithm given by:

$$\Theta_k^c = \Theta_{k-1}^c$$

$$P_k = \frac{p(1 - a_k^2)}{p - 1}\left(P_{k-1} - \frac{1 - pa_k^2}{(1 - a_k^2)\Phi_k^T P_{k-1} \Phi_k} P_{k-1}\Phi_k \Phi_k^T P_{k-1} \right).$$

The adaptation of the error Eq. (27.12) into a linear in parameter form

$$|x' \cos(\varphi_i') + y' \sin(\varphi_i') - \rho_i'| \leq \beta_k$$

can be written as

$$|x' \; u_i' + y' \; v_i' - \rho_i'| \leq \beta_k$$

with the constraints

$$\begin{cases} u_i'^2 + v_i'^2 = 1 \\ \rho_i' \geq 0. \end{cases} \tag{27.22}$$

The algorithm can then be used to estimate the feasible domain of (φ_i', ρ_i') by the statement of

- the parameter vector to be estimated $\Theta = [u_i', vi_i', \rho_i']^T$
- the measurements and known quantities $\Phi_k = [x_k', y_k', -1]^T$; $y_k = 0$

A kth measurement Φ_k results in updating the feasible domain of the segment line equation. From $(\Theta_{k-1}^c, P_{k-1})$ comes (Θ_k^c, P_k). As the admissible components of Θ are constrained by Eq. (27.22), Θ_k^c will be modified to fulfill those constraints (this modification simply consists of multiplying all components of Θ_k^c by the scalar γ_k (Eq. (27.23)) and the matrix P_k by γ_k^2 :

$$\gamma_k = \frac{sign(\Theta_{3,k}^c)}{((\Theta_{1,k}^c)^2 + (\Theta_{2,k}^c)^2)^{1/2}} \tag{27.23}$$

$$\Theta_k^c := \gamma_k . \Theta_k^c$$

$$P_k := \gamma_k^2 . P_k$$

The feasible domain for Θ_k being determined, the initial estimation problem is treated by determining the feasible values of the detected obstacle line segment $\mathcal{S}_i$ characteristics (φ_i', ρ_i') (see Eq. (27.17)). Those feasible domains certainly exist since the center Θ_k^c of $\mathcal{E}_k$ satisfies the constraints of Eq. (27.22).

The possible values of $\mathcal{S}_i$ are thus obtained by intersecting the ellipsoid (Eq. (27.24)) with the cylinder (Eq. (27.25)):

$$\varphi' \in \Phi' : \{\varphi_i' \in [0, 2\pi[;$$

$$\Theta = [\cos \varphi_i' \ \sin \varphi_i' \ \rho_i']^T;$$

$$(\Theta - \Theta_k^c)^T P_k^{-1} (\Theta - \Theta_k^c) \leq 1; \tag{27.24}$$

$$(\Theta_{1,k})^2 + (\Theta_{2,k})^2 = 1\} \tag{27.25}$$

It yields

$$\varphi' \in [\varphi_{i,k}'^c - \varepsilon_{\varphi',k}, \varphi_{i,k}'^c + \varepsilon_{\varphi',k}]$$

The possible values of ρ_i' is the "positive magnitude" of the ellipsoid (Eq (27.24)). It becomes:

$$\rho_i' \in R_i' = [\max(0, \rho_{i,k}'^c - \beta_{\rho_i'}), \rho_{i,k}'^c + \beta_{\rho_i'}]$$

$$\beta_{\rho'_i} = \frac{\| [(P_k)_{3,1}\ (P_k)_{3,2}\ (P_k)_{3,3}]^T \|}{\sqrt{(P_k)_{3,3}}} .$$

This ends the estimation of the relative localization of an obstacle in the mobile frame $\mathcal{M}$. The localization of the robot in the world frame $\mathcal{W}$ is obtained by a change of coordinates.

Computing the position $(x_c y_c)$ can thus be obtained by processing the following error equation:

$$|(x_{M_i} - x_c)\cos\varphi_i + (y_{M_i} - y_c)\sin\varphi_i - \rho'^c_{i,k}| \le \beta_{\rho'_i} .$$

The EPC algorithm can then be used with:

$$\Theta = [(x_c - x_{M_i})\ (y_c - y_{M_i})]^T$$

$$\Phi_k = [\cos\varphi_i\ \ \sin\varphi_i]^T$$

$$y_k = \rho'^c_{i,k}$$

$$\beta_k = \beta_{\rho'_i}$$

It results in updating a feasible ellipse (Eq. (27.26)) parametrized by its center Θ^c_k and its matrix P_k:

$$(\Theta - \Theta^c_k)^T P_k^{-1}(\Theta - \Theta^c_k) \le 1. \tag{27.26}$$

The orientation is obtained by a simple difference (Eq. 27.8). It becomes then the following based upon the φ_i segment relative orientation:

$$\Theta \in [\theta^c_k - \varepsilon_{\theta,k},\ \theta^c_k + \varepsilon_{\theta,k}], \tag{27.27}$$

$$\theta^c_k = \varphi_i - \varphi'^c_{i,k} ,$$

and

$$\varepsilon_{\theta,k} = \varepsilon_{\varphi,k} .$$

This orientation characterization ends the estimation part of the algorithm. It is illustrated by experimental results, where one sees the localization uncertainty decreasing when new significative measurements are added, and where this uncertainty remains constant when almost redundant measurements are treated. Such results obtained from a real static localization experience are shown in the appendix, at the end of this chapter.

This section has dealt with estimation when a measurement, stated by the triple of a landmark segment $\mathcal{S}_i$, a detected obstacle point P', and the precision β, has been obtained. As mentioned in Section 27.1, an initial matching between a given measurement and a given landmark should be done initially.

27.3.3. Matching

Matching a rough telemetric measurement P' (defined in the mobile frame $\mathcal{M}$) with a given line segment primitive (defined in the world frame $\mathcal{W}$) consists in mapping each possible measurement associated to one segment, $\mathcal{S}_i$, represented in the observation space $\mathcal{M}$. Due to localization uncertainty, the various candidate line segment primitives $\mathcal{S}_i$ constitute classes in the observation space in which a new measurement P' should be assigned to give a localization information.

By mimetism to the EKF solution (Section 27.2), this classification is achieved by using a Mahalanobis style distance in such a way that a measurement is either rejected as an outlier or matched to the nearest class according to such a distance.

The definition of such a Mahalanobis distance (Eq. 27.12) is performed by enclosing the possible measurements matchable to a line segment in an elliptic envelope whose characteristics (center and matrix) define the distance. The definition of the dispersion of possible measurements is conceptually represented in Figs. 27.12–27.14. The uncertainty in the localization in $\mathcal{W}$ results in a dispersion of possible telemetric measurements. The possible measurement are centered around possible positions of the obstacle segments in the mobile frame.

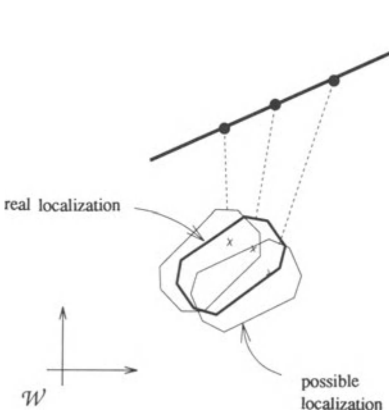


FIGURE 27.12. The uncertainty in the localization in W results in a dispersion of possible telemetric measurements.

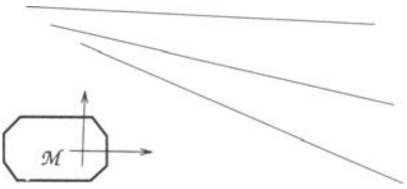


FIGURE 27.13. The possible measurements are centered around possible positions of the obstacle segments in the mobile frame.

This uncertainty stems from

1. The telemetric measurement error analyzed in Section 27.31, which results in the true obstacle point inside a disc centered around the detected point (Fig. 27.14a);
2. The orientation uncertainty (Fig. 27.14(b)); and
3. The position error (Fig. 27.14(c)).

The computation of an ellipse enclosing all this possible sets appears as a difficult operation and the search of a real time solution results in simplifications. A first one consists of replacing the sensor error caused uncertainty domain by its polygonal envelope and the elliptical possible positions set by its rectangular envelope. The problem results in the computation of an elliptical envelope of a set of polygonals which is known(12) to be exactly solvable in real time.

Another rustic solution consists of heuristically selecting four characteristic points somehow representative of the possible measurements set and determining an ellipse containing those points (Fig. 27.15):

From the preceding discussion, the absolute localization uncertainty (Eqs. (27.28 and 27.29)), and a segment characterization (Eq. (27.30)),

$$\theta \in [\theta^c - \varepsilon_\theta, \theta^c + \varepsilon_\theta], \tag{27.28}$$

$$\Theta \in \{(\Theta - \Theta_k^c)^T P_k^{-1}(\Theta - \Theta_k^c) \leq 1;\ \Theta = (x_c, y_c)^T\}, \tag{27.29}$$

and

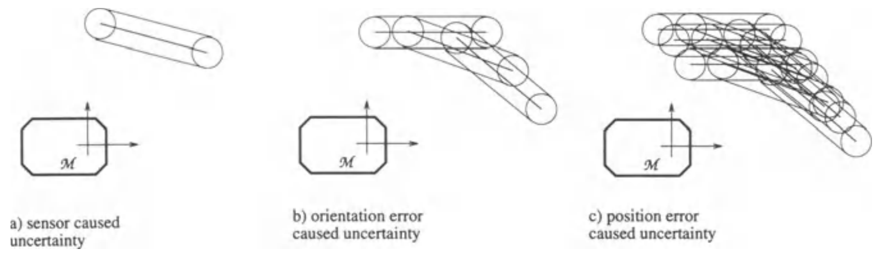

M
a) sensor caused uncertainty
M
b) orientation error caused uncertainty
M
c) position error caused uncertainty

FIGURE 27.14. Orientation and position error caused uncertainty: (a) sensor-caused uncertainty; (b) orientation error-caused uncertainty; and (c) position error-caused uncertainty.

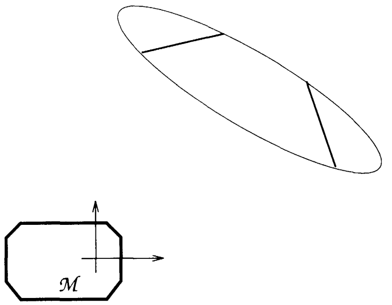


FIGURE 27.15. Ellipse enclosing four heuristic points.

$$(M_i, \varphi_i, \rho_i). \tag{27.30}$$

it is possible to define an ellipse Eq. (27.31) in the mobile frame,

$$[x' - \hat{x}'_{M_i} \quad y' - \hat{y}'_{M_i}] \; \Sigma'_i \begin{bmatrix} x' - \hat{x}'_{M_i} \\ y' - \hat{y}'_{M_i} \end{bmatrix} \le 1 \tag{27.31}$$

that corresponds to the possible detected points which can be associated with $\mathcal{S}_i$. Define now Eq. (27.32) as a Mahalanobis type distance.

$$d^i(P', \mathcal{S}_i) = \| \begin{bmatrix} x' - x'_{M,i} & y' - y'_{M,i} \end{bmatrix}^T \|_{\Sigma'_i} \tag{27.32}$$

it can be used for classification purpose in the same way that in the EKF solution:

1. Find the primitive $\mathcal{S}_{i_o}$ that minimizes

$$d^i(P', \mathcal{S}_i) + \ln(\det(\Sigma'_i))$$

2. If $d^{i_o}(P', \mathcal{S}_{i_o}) < \chi^2_o$, then match P' to $\mathcal{S}_{i_o}$ else reject P' as an outlier.

An additional matching feature is contained in the EPC algorithm (Section 27.3.2, Test 1) and is used to reject incoherent measurements. After having completed the matching and estimation part of a set membership localization algorithm, the next section deals with the vehicle movement and takes into account the fact that measurements are not obtained at the same time they are used for localization.

27.3.4. The Moving Vehicle Localization

27.3.4.1. Uncertainty of the Dead-Reckoning System

This section presents the adaptation of the previous static localization procedure to the case when measurements are taken as the vehicle moves. There are two main differences between the dynamic and the static case:

1. The localization uncertainty drifts when it is only updated by odometry.
2. The localization estimation at time t_k is obtained from the previous one computed at time t_{k-1} and from a sequence of measurements:

$$\{P'(t_{k-1} + \tau_1), \dots, P'(t_{k-1} + \tau_N)\};$$

each one is referenced in the sequence of mobile frame $\mathcal{M}(t_{k-1} + \tau_i)$ bound to the robot configuration sequence

$$\{C(t_{k-1} + \tau_i), \theta(t_{k-1} + \tau_i)\}_{i=1,\dots,N}.$$

Both questions can be answered if it is possible to state how the localization uncertainty increase as the robot moves from one configuration (C_0,θ_0) to another one (C,θ). It is necessary, then, to analyze the dead-reckoning error causes.

Consider a differentially driven vehicle (a cart-like vehicle). The characteristic point C is chosen in the middle of the driving wheel's axis. Let R be the common wheel radius and $2e$ the distance between their centers, an angular deviation $\delta\psi_l$ (resp. $\delta\psi_r$) of the left (resp. right) wheel. Arguing that each wheel rolls without slipping gives the following relation:

$$\begin{cases} \delta s = \dfrac{R}{2}(\delta\psi_r + \delta\psi_l) \\ \delta\theta = \dfrac{R}{2e}(\delta\psi_r - \delta\psi_l) \end{cases}$$

and

$$\begin{cases} \delta x_c = \delta s \cos\theta \\ \delta y_c = \delta s \sin\theta \end{cases}.$$

Consider a deviation $\delta R_l = R_l - R$ (resp. $\delta R_r = R_r - R$) between the left (resp. right) wheel radius and it estimation, and a deviation $2\delta e = 2\hat{e} - 2e$ between the points on the ground where there is rolling without slipping and the wheels center distance $2\hat{e}$. Under the rolling without slipping hypothesis, it becomes:

$$\begin{cases} \delta s = \dfrac{1}{2}(R_r\delta\psi_r + R_l\delta\psi_l) \\ \delta\theta = \dfrac{1}{2e}(R_r\delta\psi_r - R_l\delta\psi_l) \end{cases}.$$

The dead-reckoning system interprets those angular movements as

$$\begin{cases} \hat{\delta s} = \dfrac{1}{2}(R\delta\psi_r + R\delta\psi_l) \\ \hat{\delta\theta} = \dfrac{1}{2\hat{e}}(R\delta\psi_r - R\delta\psi_l) \end{cases} \tag{27.33}$$

and

$$\begin{cases} \delta\hat{x} = \delta\hat{s} \cos \hat{\theta} \\ \delta\hat{y} = \delta\hat{s} \sin \hat{\theta} \end{cases} . \tag{27.34}$$

The deviation between the real and the estimated motion increment is thus:

$$\begin{cases} \delta s - \delta\hat{s} = \frac{1}{2} (\delta R_r \delta\psi_r + \delta R_l \delta\psi_l) \\ \delta\theta - \delta\hat{\theta} \simeq \frac{1}{2\hat{e}} (\delta R_r \delta\psi_r - \delta R_l \delta\psi_l) - \delta\hat{\theta}(\frac{\delta e}{\hat{e}}) \end{cases}$$

This uncertainty can be overvalued by using a total displacement variable ($\delta|\psi_w|$):

$$\delta|\psi_w| = (|\delta\psi_1| + |\delta\psi_r|) \qquad (^\dagger)$$

As the wheel radius deviations δR_l, δR_r and the inter-center distance error δe are naturally bounded, there exists positive constants (p_1, p_2) such that

$$|\delta\theta - \delta\hat{\theta}| \leq p_1 \, \delta|\psi_w|$$

and

$$|\delta s - \delta\hat{s}| \leq p_2 \, \delta|\psi_w|.$$

Consider now a finite displacement from (C_0,θ_0) to $(C,\theta) = (C_0 + \Delta C, \theta_0 + \Delta\theta)$. During this motion, the total displacement is $\Delta|\psi_w|$. The drift in the dead-reckoning estimation is overvalued by the following inequalities:

$$|\Delta\theta - \Delta\hat{\theta} \,| \leq p_1 \, \Delta|\psi_w| \tag{27.35}$$

and

$$||\Delta C - \Delta\hat{C}|| \leq p_2 \, \Delta|\psi_w| + (\varepsilon_{\theta,0} + \varepsilon(\Delta\theta))\Delta\hat{s} \, \Delta|\psi_w| \tag{27.36}$$

Those inequalities end the dead reckoning system error estimation analysis and provide an answer to the initial question.

27.3.4.2. Drift of the Localization Uncertainty Due to Odometry

The analyzed situation is represented on Fig. 27.16. The possible position domain is increased from $\mathcal{E}_{k-1}$ to $\mathcal{E}_k$

$$\mathcal{E}_{k-1} = \{C \in \Re^2; (C - \hat{C}_{k-1})^T P_{k-1}^{-1}(C - \hat{C}_{k-1})\},$$

[†] such a variable is incremented each time a binary encoder tied to the wheels detects an angular motion in any direction

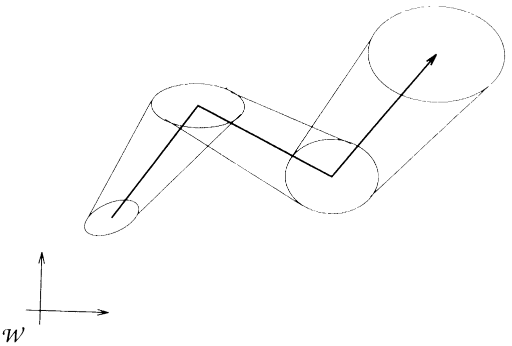


FIGURE 27.16. Ellipsoidal uncertainty increase.

$$\mathcal{E}_k = \{C \in \Re^2; (C - \hat{C}_k)^T P_k^{-1}(C - \hat{C}_k)\},$$

and

$$\hat{C}_k = \hat{C}_{k-1} + \hat{\Delta} C.$$

Modify the (2 × 2) matrix P_k in such a way that the $\mathcal{E}_k$ ellipse principal axes are those of $\mathcal{E}_{k-1}$ augmented by

$$(p_2\Delta|\psi_w| + (\varepsilon_{\theta,k-1}) + (p_1\Delta|\psi_w|).\hat{\Delta} s.\Delta|\psi_w|)),$$

which is a majoration of the position uncertainty drift $\|\Delta C - \hat{\Delta} C\|$ (Eq. (27.36)). The orientation estimation drift is even simpler since it is transformed from

$$\Theta_{k-1} = \{\theta \in [\hat{\theta}_{k-1} - \varepsilon_{\theta,k-1}, \hat{\theta}_{k-1} + \varepsilon_{\theta,k-1}] \subset]-\pi, +\pi] \bmod 2\pi\}$$

to

$$\Theta_k = \{\theta \in [\hat{\theta}_k - \varepsilon_{\theta,k}, \hat{\theta}_k + \varepsilon_{\theta,k}] \subset]-\pi, +\pi] \bmod 2\pi\}$$

with

$$\varepsilon_k(\theta) = \varepsilon_{k-1}(\theta) + p_1\Delta|\psi_w|.$$

The localization uncertainty drift caused by dead reckoning being stated, the next subsection analyzes the way a measurement taken at time t_0 in the frame $\mathcal{M}(t_0)$ bound to the configuration (C_0,θ_0) can be used to determine the localization (C_k,θ_k) at time t_k.

27.3.4.3. Localization by Use of Past Measurements

There are two aspects in this problem: estimation by using a matched measurement and matching a given measurement to a straight segment primitive. The answer to both questions is found by adding to the uncertainty drift due to the dead reckoning with the one of the static localization procedure.

1. Using a measurement $P'(t_{k-1}+\tau)$ matched to a segment $\mathcal{S}_i$ to evaluate the configuration (C_k,θ_k) at time t_k is done by:
   (a) describing in the reference frame $\mathcal{M}(t_k)$ the measurement $P'(t_{k-1}+\tau)$ obtained in $\mathcal{M}(t_{k-1}+\tau)$
   (b) adding the uncertainty $p_2\,\Delta|\psi_w| + (\varepsilon(\theta_{k-1}) + p_1\,\Delta|\psi_w|)\widehat{\Delta s}\Delta|\psi_w|)$ to the initial measurement uncertainty on P' (Eq. (27.36)).
2. Matching a measurement $P'(t_{k-1}+\tau)$ to a segment $\mathcal{S}_i$ is performed by computing the position of $\mathcal{S}_i$ in the reference frame $\mathcal{M}(t_{k-1}+\tau)$ and by using the uncertainty characterization of $(C(t_{k-1}+\tau),\ \theta(t_{k-1}+\tau))$ obtained from the one stated at time t_{k-1} augmented with the dead reckoning uncertainty drift (Eq. (27.35 and 27.36)). The position of the segment $\mathcal{S}_i$ in the mobile frame $\mathcal{M}(t_{k-1}+\tau)$ and the corresponding uncertainty being thus stated, matching can be performed by use of the static matching procedure of Section 27.3.3.

27.4. CONCLUSION

The essentially nice feature in the use of the set membership approach for mobile robots localization lies in the error modeling freedom.

Using statistical EKF based solutions for this problem can be interpreted as a way to perform uncertainty computation, assuming small deviations and linearized progression and observation models around the current estimated localization. This theoretical simplification gives a closed form solution in which error causes are melted in the definition of a state, and measurement noise vector whose covariances become nothing else but a set of tuning parameters.

From an opposite point of view, the geometric and physical description of the measurements error causes described in this chapter has "naturally" led to characterizing error bounds in the measurement equations that constitute the principle of the localization technique. Of course, this problem based error characterization is less straightforward formally as a simple linearization. However, the error analysis is, in our opinion, closer to the basic problem.

From the recursive computation point of view, the bounded error approach has two original features with respect to classical EKF or recursive least squares (RLS) algorithms. The former lies in a model consistency test and the latter in a dead-zone.

Here, the model consistency test is used in the matching module and prevents the use of mismatched couples (measurement, landmark) in addition to a matching procedure that is a copy of the EKF solution matching procedure.

The dead-zone feature of the algorithm has both positive and negative aspects. The negative one consists in the non-increasing accuracy obtained when a large number of measurements are quasi-redundant, the positive one is the same argument considered from another viewpoint: a large number of quasi-redundant measurements result in a small number of localization updates, thus saving computation time.

In conclusion, the authors are now working on an ecumenical localization algorithm. It combines the advantages of the set membership solution and the ones of EKF based solutions to benefit from the law of large numbers when possible, and avoiding to invoke it when scarce measurements are available.

REFERENCES

1. I. Collin, D. Meizel, N. Lefort, and G. Govaert, in: *Proceedings of the IROS '94 Conference*, München, Germany (1994).
2. J. L. Crowley, in: *Proceedings of the IEEE International Conference on Robotics and Automation,* Scottsdale, Arizona, pp. 674–680 (1989).
3. J. J. Leonard and H. Durrant-Whyte, *IEEE Trans. Robotics Autom.* 7, 376 (1991).
4. P. Cheeseman and R. C. Smith, *International J. Robotics Research* 56 (1986).
5. C. Durieu and H. Clergeot, *APII* **25**, 437 (1991).
6. A. Preciado, D. Meizel, A. Segovia, and M. Rombaut, in: *Proceedings of the IEEE International Conference on Robotics and Automation,* Sacramento, California, pp. 2806–2811 (1991).
7. A. Sabater and F. Thomas, in: *Proceedings of the IEEE International Conference on Robotics and Automation,* Sacramento, California, pp. 2718–272 (1991).
8. D. Meizel and A. Preciado, in: *Proceedings of the International Symposium on Intelligent Robotics,* Bangalore, India, pp. 815–824 (1993).
9. G. D. Hager, *Task-Directed Computation of Qualitative Decisions From Sensor Data*, Yale University Department of Computer Science, Yale University, New Haven, CT (1992).
10. E. Walter and H. Piet-Lahanier, *Math. Comput. Simul.* **32**, 449 (1990).
11. R. Bland, D. Goldfarb, and M. Todd, *Oper. Res.* **29**, 1039 (1981).
12. L. Pronzato and E. Walter, in: Vol. 1 of *Proceedings of the IEEE/IFAC/SIAM/ECCA 2nd European Control Conference*, Groningen, The Netherlands, pp. 258–263 (1993).

APPENDIX

A Static Localization Experience

This appendix sums up graphically the steps of the static localization procedure (Section 27.3.2). The initial situation is depicted in Fig. 27.A.1. In all figures, the line segments represent the walls, the real robot is drawn in grey, and the estimated one is represented by dotted lines. The position uncertainty is represented by an ellipse, whereas the orientation uncertainty is represented by an angular sector. Graduations are in mm.

In Fig. 27.A.2, the result of the localization algorithm after processing one single measurement is represented (the point detected by the sonar in the estimated mobile frame is represented by a star). The strip associated with this measurement,

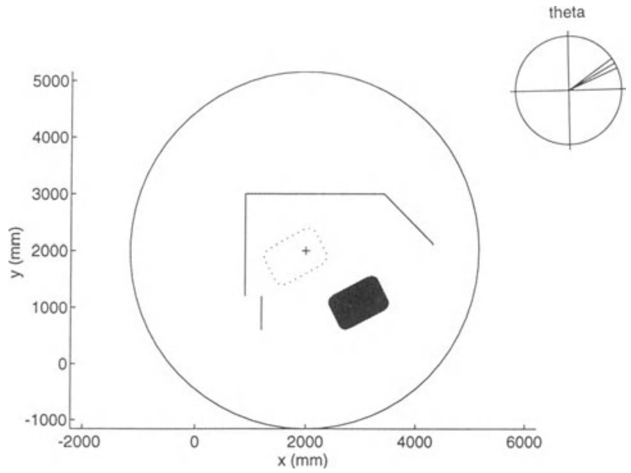


FIGURE 27.A.1. The initial situation.

(which represents the feasible domain for the robot position) is shown. A new ellipse is computed by intersecting the old ellipse and the strip. Uncertainty is reduced in the normal direction to the obstacle.

Note the exploitation of all measurements matched with the first-oblique-segment. In Fig. 27.A.3 there is no real improvement after the first measurement. Indeed, the strips have slightly the same width and they recover nearly exactly the

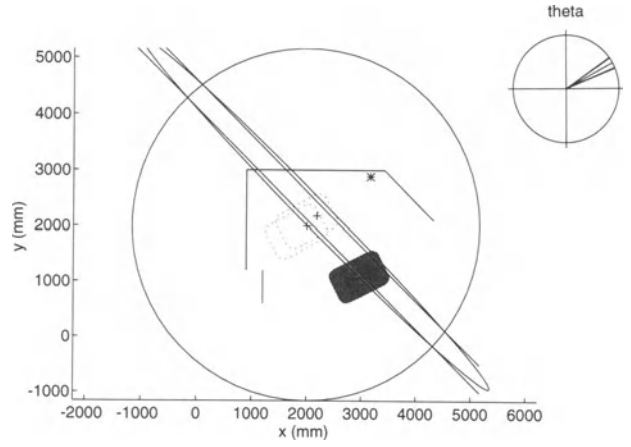


FIGURE 27.A.2. Exploitation of the first measurement.

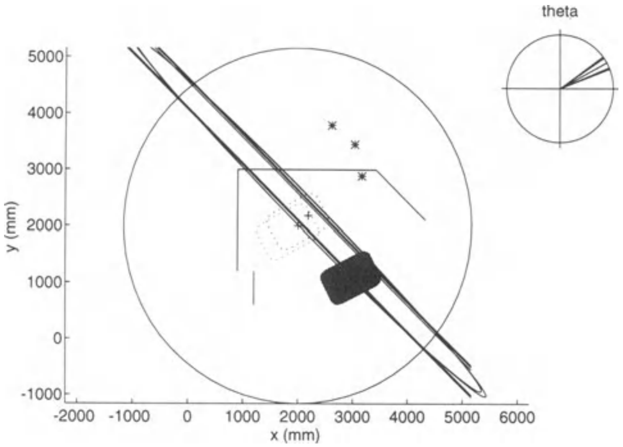


FIGURE 27.A.3. Dead-zone of the localization.

first ellipse. This exhibits the dead-zone characteristics of the bounded-error estimation based localization algorithm.

After exploitation of all measurements matched with the second segment (parallel to the x-axis), (Fig. 27.A.4), uncertainty is reduced in the normal direction to the new obstacle. Here too, the final ellipse is only due to the first measurements. Strips caused by further measurements are wider than this ellipse, and bring no improvement.

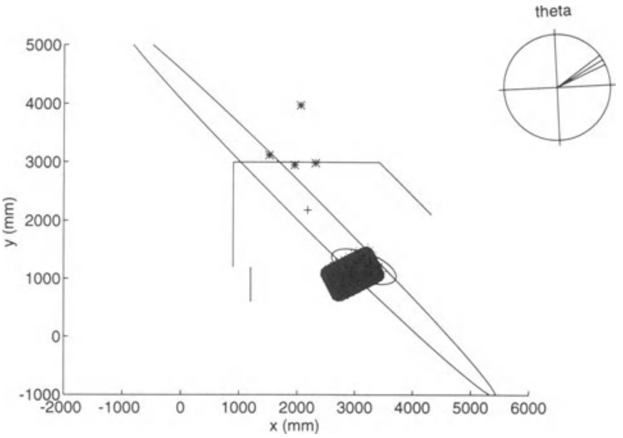


FIGURE 27.A.4. X-axis segment processing.

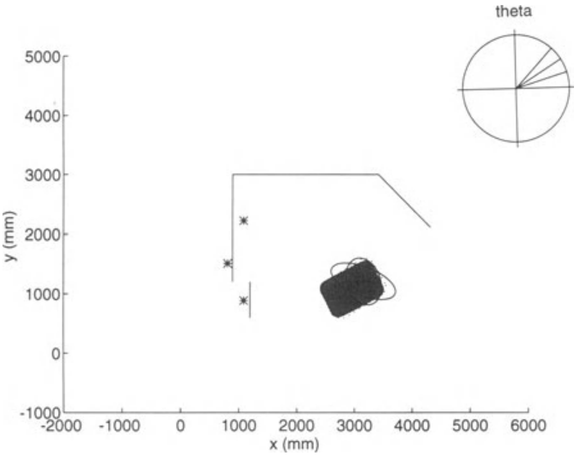


FIGURE 27.A.5. After treatment of y-axis segments.

The exploitation result of all measurements matched to the y-axis segments is represented in Fig. 27.A.5. In Fig. 27.A.6 the exploitation result of all measurements is summed up.

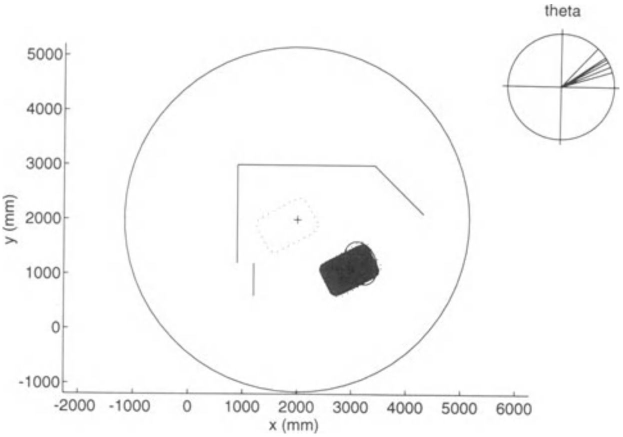


FIGURE 27.A.6. Summary of the initial and final estimations.

28

Improved Image Compression Using Bounded-Error Parameter Estimation Concepts

A. K. Rao

28.1. INTRODUCTION

Classical approaches to parameter estimation yield point estimates of parameters by optimizing some criterion of fit. In contrast, bounded error parameter estimation (BEPE) methods provide sets of parameters which are consistent with the model structure, observation record, and uncertainty constraints. In general, no knowledge of the statistics of the model or observation uncertainty is assumed. The uncertainty, however, is assumed to be constrained in some manner, e.g., with bounded energy or bounded magnitude.(1) BEPE methods seem more appropriate than classical techniques in several situations. If the actual system is only loosely modeled by the chosen model, it appears more reasonable to attempt to optimize the model so as to bound the model mismatch error, rather than to do classical parameter estimation with erroneous assumptions on the statistics of the model mismatch error. In other cases, the statistics of the observation uncertainty may not be known and BEPE techniques may be effective.

In many signal processing applications, point estimates of parameters are required while BEPE actually yields a set of valid parameter estimates. In such

A. K. Rao • COMSAT Labs, Clarksburg, MD 20871.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

cases, the center of the BEPE set can be used as a point estimate for the model parameters. This point estimate is often more effective than the estimate obtained by classical estimation techniques. In other applications, the entire set of parameters is of interest. This chapter, shows how both BEPE based point estimates and parameters sets can be used effectively to enhance standard image compression techniques. Parameter bounding techniques have been used previously for several signal processing applications such as linear prediction of speech,(2) signal restoration,(3) and filter design,(4) but do not appear to have been used for image coding.

Since images are highly non-stationary and difficult to model, BEPE methods may be more effective for estimating time-varying models with possibly large model-mismatch errors. The first application discussed is the use of a specific bounded error estimation technique (the time-varying optimal bounding ellipsoid (OBE) method)(5,6) to improve the efficiency of two-dimensional adaptive differential pulse-coded modulation (ADPCM) coding of images. Conventional algorithms such as least-mean-squares (LMS) or recursive least-squares (RLS) often cannot track the rapidly changing model parameters. Incorrect predictions and large prediction errors result, which require more bits for transmission. The parameter-tracking-bounding ellipsoidal algorithm used here can automatically adapt to small and large changes in model parameters and, as shown later, outperforms conventional slow-adaptation algorithms for some of the images tested.

The other application discussed is in the quantization of discrete cosine transform (DCT) coefficients. The DCT method(7) is used widely for image compression and is now part of the Joint Photographic Experts Group (JPEG), and Motion Pictures Expert Group (MPEG) image and video compression standards. In the DCT coding technique, image blocks are transformed using a two-dimensional transform. The transform causes compaction of the energy in the block into only a few low order transform coefficients. Compression is achieved by coarse quantization of the higher order coefficients. The image block can then be reconstructed by applying the inverse DCT. The unitary nature of the DCT implies that the l^2 norm of the error in the transform domain is identical to the l^2 norm of the error in the spatial domain. However, in several applications such as medical imaging and remote sensing, it is important to specify and control the maximum amount of distortion in the image samples (the l^∞ norm) introduced by this DCT quantization process. This chapter uses bounded error parameter estimation concepts to obtain necessary and sufficient conditions in the transform domain for the distortion in the spatial domain to be bounded. These conditions are easily tested on the DCT coefficients on a block-by-block basis. Unfortunately, the sufficient conditions derived using parameter bounding methods are excessively pessimistic and, therefore, not very useful. However, the necessary conditions can be used to validate the quantization table used for a block to constrain the spatial distortion without an excessive increase in the bit rate. Similar parameter bounding techniques can be applied to other transforms or applications, such as the discrete Fourier transform, wavelet transform, and speech compression.

28.2. ADAPTIVE PARAMETER ESTIMATION

The specific BEPE algorithm considered here is an optimal bounding ellipsoid (OBE) algorithm[8] which is a variant of the original Fogel–Huang minimum volume bounding ellipsoid estimator.[9] Like the Fogel–Huang algorithm, this OBE algorithm obtains a sequence of ellipsoids which upper-bound the feasible parameter set. The difference is that the ellipsoids are not guaranteed to be of minimum volume. Instead a certain upper bound on the size is minimized. The algorithm is computationally simpler and the analysis of its properties for stationary[8] and time-varying[6] models is more tractable. The OBE algorithm estimates the parameters of a linear model of the form:

$$y(t) = \theta^{*T}\phi(t) + w(t), \tag{28.1}$$

where θ^* is the n-dimensional true parameter vector, $\phi(t)$ is the regressor vector of observed data, and $w(t)$ is the observation or modeling uncertainty which is assumed to be upper bounded, i.e.,

$$|w(t)| < \gamma \text{ for all } t.$$

Let the bounding ellipsoid at time instant t–1 be described by

$$E_{t-1} = \{\theta \in \Re^n: [\theta - \theta(t-1)]^T P^{-1}(t-1)[\theta - \theta(t-1)] \leq \sigma^2(t-1)\}$$

where the $\theta(t -- 1)$ is the center of the ellipsoid defined by the ellipsoidal matrix $P^{-1}(t-1)$. A scalar $\sigma^2(t-1)$, along with $P(t-1)$, controls the size of the ellipsoid. Then, the bounding ellipsoid at instant t is[6]

$$E_t = \{\theta \in \Re^n: [\theta - \theta(t)]^T P^{-1}(t)[\theta - \theta(t)] \leq \sigma^2(t)\}, \tag{28.2}$$

and recursively as

$$\theta(t) = \theta(t-1) + \lambda_t P(t)\phi(t)\delta(t) \tag{28.3}$$

$$P^{-1}(t) = (1-\lambda_t)P^{-1}(t-1) + \lambda_t\phi(t)\phi^T(t) \tag{28.4}$$

and

$$\sigma^2(t) = (1-\lambda_t)\sigma^2(t-1) + \lambda_t\gamma^2 - \lambda_t\delta^2(t)\frac{(1-\lambda_t)}{1-\lambda_t+\lambda_t G(t)}\cdot \tag{28.5}$$

The prediction error is

$$\delta(t) = y(t) - \theta^T(t-1)\phi(t) \tag{28.6}$$

and

$$G(t) = \phi^T(t)P(t-1)\phi(t). \tag{28.7}$$

The factor λ_t is a positive time-varying update gain which is chosen to minimize $\sigma^2(t)$ at every sample index t. This has the effect of usually decreasing the size of the bounding ellipsoid from one iteration to the next, though there is no guarantee that the size is minimized. This choice of λ_t has yielded good results experimentally and has simplified the convergence and tracking analysis of the algorithm. The minimization procedure yields the following updating criterion(8)

If

$$\sigma^2(t-1) + \delta^2(t) \le \gamma^2, \text{ then } \lambda_t = 0 \text{ (i.e., no update)} \tag{28.8}$$

Otherwise if $\sigma^2(t-1) + \delta^2(t) > \gamma^2$, then the optimum value of λ_t is non-zero. It can be calculated according to

$$\lambda_t = \min(\alpha, v_t),$$

where

$$v_t = \begin{cases} \alpha & \text{if } \delta^2(t) = 0, \qquad (28.9(a)) \\ \dfrac{1-\beta(t)}{2} & \text{if } G(t) = 1, \qquad (28.9(b)) \\ \dfrac{1}{1-G(t)}\left(1 - \sqrt{\dfrac{G(t)}{1+\beta(t)(G(t)-1)}}\right), & \text{if } 1 + \beta(t)(G(t)-1) > 0. \qquad (28.9(c)) \\ \alpha & \text{if } 1 + \beta(t)(G(t)-1) \le 0, \qquad (28.9(d)) \end{cases}$$

and α is a user chosen upper bound on λ_t satisfying

$$0 < \alpha < 1, \tag{28.10}$$

and

$$\beta(t) = (\gamma^2 - \sigma^2(t-1))/\delta^2(t). \tag{28.11}$$

The initial conditions are chosen to ensure that $\theta^* \in \mathrm{E}_0$. A possible choice is

$$P(0) = \mathbf{I},\ \theta(t) = 0 \text{ and } \sigma^2(0) = 1/\varepsilon^2 \text{ where } \varepsilon << 1.$$

As in other least-squares type algorithms, the update equation for $P^{-1}(t)$ can be manipulated using the matrix inversion lemma to yield a recursive relationship in terms of $P(t)$. For autoregressive with external input (ARX) models, some convergence type properties, such as convergence of the parameter estimates to a ball and boundedness of the prediction error have been derived.(8) The algorithm has been extended to autoregresive moving average (ARMA) models and similar convergence properties have been shown to hold.(10)

Most of the existing parameter bounding algorithms have been developed for the fixed parameter case. As a result, changes in the true parameter may cause the feasible parameter set to vanish. In the case of the OBE, small changes in the parameter can be automatically tracked (in the sense that the center of the bounding ellipsoid moves towards the new parameter). Larger changes can cause the bounding ellipsoid to vanish and the functional $\sigma^2(t)$ is then no longer positive. Thus, monitoring the sign of $\sigma^2(t)$ provides for easy parameter jump detection. In such situations, a rescue procedure can be activated that inflates the size of the previously obtained bounding ellipsoid by an appropriate amount to permit a new valid bounding ellipsoid to be constructed. The inflation is achieved by increasing $\sigma^2(t-1)$ to an amount prescribed by the following algorithm.(5,6)

If

$$1 + \frac{\gamma^2 - \sigma^2(t-1)}{\delta^2(t)}(G(t) - 1) > 0$$

and

$$\lambda_t < \alpha,$$

then

$$\sigma^2(t-1) = \frac{1}{G(t)-1}\left(\delta^2(t) + \gamma^2[G(t) - 1] - \frac{[\gamma(G(t)-1) + |\delta(t)|]^2}{G(t)}\right) + \mathit{offset} \text{ if } G(t) \neq 1$$

and

$$\sigma^2(t-1) = \delta^2(t) + \gamma^2 - 2\gamma|\delta(t)| + \mathit{offset} \text{ if } G(t) = 1.$$

Else

$$\sigma^2(t-1) = \alpha\left(\frac{\delta^2(t)}{1 - \alpha + \alpha G(t)} - \frac{\gamma^2}{1-\alpha}\right) + \mathit{offset}.$$

Offset is a user selectable constant (typical value 1.0) and α is the upper bound on λ. This inflation of $\sigma^2(t-1)$ and, consequently, E_{t-1}, ensures the existence of bounding ellipsoid E_t. The center of E_t gravitates towards the true parameter.

Simulation results with computer generated ARX model data have shown that the OBE algorithm is capable of tracking slow and abrupt parameter variations without activating the rescue procedure. In some runs, after a large parameter change, the bounding ellipsoid vanishes. In such cases the rescue procedure causes remarkably rapid tracking and accurate parameter estimation. Consequently, this time-varying estimator is ideal for model estimation of high resolution images that exhibit rapid spatial variations in intensity.

28.3. ADPCM IMAGE CODING

Most images have a significant amount of redundancy; the pixel intensities are highly correlated horizontally and vertically. Image compression involves the reduction or removal of this redundancy and the use of visual masking techniques to reduce the amount of data which is required to faithfully represent the image. Perhaps the oldest and best established lossy compression technique is differential pulse coded modulation (DPCM).[11] In DPCM, a prediction of the pixel intensity is formed as a linear combination of the pixel intensities in a causal neighborhood of the pixel. The prediction is subtracted from the pixel intensity to obtain a prediction error which is quantized using lesser bits than the original pixel intensity. The quantized prediction error image can thus be transmitted or stored with a lower number of bits than the original image. Typically, compression ratios of 2:1 to 4:1 are obtained. In ADPCM (Adaptive DPCM), the prediction coefficients vary from pixel to pixel. However, to avoid sending the prediction coefficients to the decoder, the encoder and decoder each update the prediction coefficients using the same data (the quantized prediction error and the previously reconstructed image samples). Figure 28.1 shows possible predictor configuration with pixels to the left and top predicting by the current pixel. The prediction operation can be described by

$$\hat{x}(i,j) = a_1x(i,j-1) + a_2x(i-1,j) + a_3x(i-1,j+1) + a_4x(i-1,j-1), \quad (28.12)$$

where $x(m,n)$ represents the reconstructed image sample at line m and column n. The a_k, $k = 1, 2, \ldots, 4$ are the prediction coefficients. Changes in image detail are accommodated by abrupt changes in the coefficients of the linear predictor. For example, in the above case, a coefficient set can model a vertical edge transition,

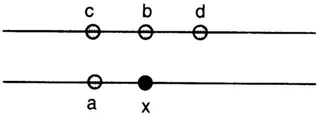


FIGURE 28.1. Predictor configuration showing neighboring pixels a, b, c, and d predicting pixel x.

in which a_2 is unity while the other coefficients are all zero. On the other hand, a horizontal edge is better modeled with $a_1 = 1$ and with all other coefficients zero.

The prediction coefficients can be adapted to changes in the image intensity in a number of ways. In some ADPCM methods, the prediction coefficients are updated using adaptive filtering algorithms such as the least-mean-squares (LMS) or recursive least-squares (RLS) algorithms.[12] Since the OBE algorithm, with the rescue procedure, is extremely effective in tracking slowly varying and abruptly changing parameters it seems particularly appropriate for tracking the prediction coefficients. Block diagrams of typical ADPCM encoder and decoder setups which would incorporate OBE parameter estimation are shown in Figs. 28.2 and 28.3 respectively.

The OBE based parameter adaptation is developed as follows. Assume that the pixel intensities $y(i,j)$ can be modeled by a time-varying linear model of the form

$$y(i,j) = \theta^T(i,j)\phi(i,j) + w(i,j) \tag{28.13}$$

where

$$\theta(i,j) = [a_1,a_2,a_3,a_4,a_5]^T \tag{28.14}$$

and

$$\phi(i,j) = [x(i,j-1), x(i-1,j), x(i-1,j+1), x(i-1,j-1), x(i-2,j)]^T \tag{28.15}$$

The uncertainty term $w(i,j)$ represents the error in modeling the image intensity by this simple moving average model. The goal is to estimate θ^* so that the model

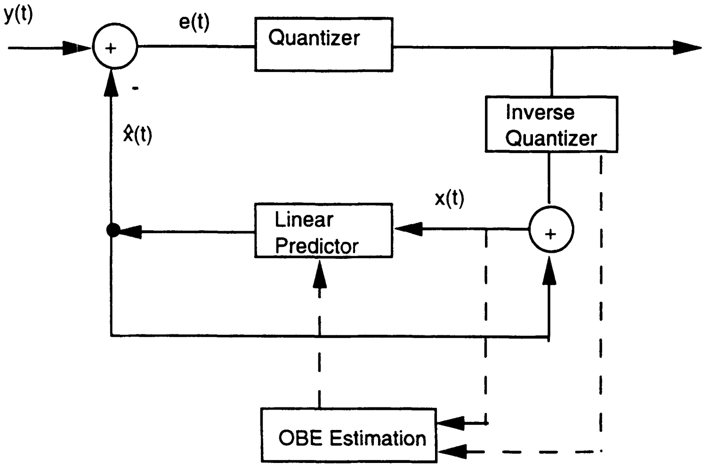


FIGURE 28.2. ADPCM encoder with OBE parameter estimation.

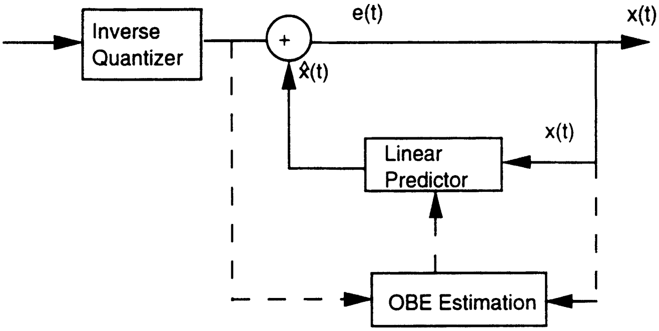


FIGURE 28.3. ADPCM decoder with OBE parameter estimation.

mismatch $w(i,j)$ is kept at a minimum. The pixels in an image are processed from left to right and top to bottom. At the beginning of every new line, the parameter estimates are reset. The update equations for the ADPCM-OBE algorithm are

$$\delta(i,j) = y(i,j) - \theta^T(i,j-1)\phi(i,j) \tag{28.16}$$

$$\bar{\delta}(i,j) = Q[\delta(i,j) \tag{28.17}$$

$$x(i,j) = \bar{\delta}(i,j) + \theta^T(i,j-1)\phi(i,j) \tag{28.18}$$

$$P(i,j) = \frac{1}{1-\lambda_{i,j}}\left[P(i,j-1) - \lambda_{i,j}\frac{P(i,j-1)\phi(i,j)\phi^T(i,j)P(i,j-1)}{1-\lambda_{i,j}+\lambda_{i,j}G(i,j)}\right] \tag{28.19}$$

$$\theta(i,j) = \theta(i,j-1) + \lambda_{i,j}P(i,j)\phi(i,j)\bar{\delta}(i,j) \tag{28.20}$$

$$\sigma^2(i,j) = (1-\lambda_{i,j})\sigma^2(i,j-1) + \lambda_{i,j}\gamma^2 - \lambda_t\bar{\delta}^2(i,j)\frac{(1-\lambda_{i,j})}{1-\lambda_{i,j}+\lambda_{i,j}G(i,j)} \tag{28.21}$$

The equations for calculation of λ, and the rescue procedure are as described in Section 28.2. The algorithm is initialized as

$$P(0,0) = MI, \theta(0,0) = (0.5, 0.25, 0.125, 0.125), \text{ and } \sigma^2(0,0) = \gamma^2,$$

where $M >> 1$, and I is the Identity matrix. The choice of initialization conditions is not critical since the ellipsoid is reinitialized appropriately whenever σ^2 becomes negative. A choice of upper bound $\alpha = 0.5$ has yielded good results.

The OBE based ADPCM scheme has been tested on a number of images and the performance has been compared with the LMS and exponentially weighted RLS

(EWRLS) based ADPCM algorithms.(12) For the EWRLS algorithm, using a forgetting factor $\lambda < 0.95$ can cause divergence due to the lack of persistence of excitation in smooth areas of the image. A possible solution is to perform the updates only if the quantized prediction error is non-zero, i.e., use a dead-zone. Since the input intensities are integer valued, the pixel prediction is rounded to the nearest integer, and consequently the prediction error is also integer valued. Thus a dead-zone of (–0.5, 0.5) is appropriate.

Simulations have been performed for a range of LMS step-sizes, EWLS weighting factors, and OBE bounds on three different images. The results are provided in Table 28.1. A uniform quantizer with a step size of four has been used in the simulations. The zone plate image (shown in Fig. 28.4) is a standard test signal used in the television industry and is considered particularly challenging. Lena is an image of a woman's face and is often used as a benchmark in image compression, while Football is one frame from a fast moving football sequence. An estimate of the compression achievable is obtained by calculating the first order entropy of the prediction error (expressed in bits/pel). The actual bit rates obtained by Huffman coding the prediction errors would be typically only 5–10 percent higher than the entropy estimate. Since the input pel intensities are 8 bit numbers, the compression ratio is given by 8/(Entropy).

The simulation results show that the OBE algorithm is particularly effective for images with significant high frequency content like the zone plate. For this image, the LMS and EWLS algorithms do not perform well. The EWLS algorithm diverged for forgetting factors below 0.95. However, with the use of dead-zoning, smaller forgetting factors can be used and the performance improved significantly.

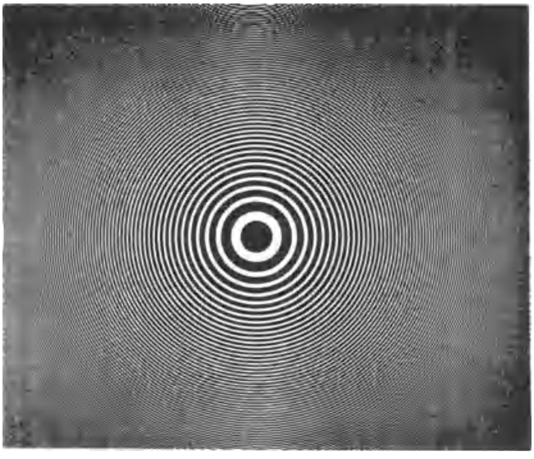


FIGURE 28.4. Zone Plate test image.

TABLE 28.1. ADPCM Coding Results

	LMS			EWLS			Dead-zoned EWLS			OBE		
Image	$\mu \times 10^{-5}$	bits/bel	Peak S/N (dB)	λ	bits/pel	Peak S/N (dB)	λ	bits/pel	Peak S/N (dB)	λ	bits/pel	Peak S/N (dB)
Zone Plate	0.5	3.9	42.9	0.99	4.4	42.8	0.9	1.9	43.2	2	1.5	43.4
	1	3.3	43.0	0.95	2.4	43.1	0.8	1.5	43.4	3	1.7	43.3
	2	3.8	43.0	0.9	Fails	Fails	0.7	1.4	43.5	4	2.1	43.1
Lena	0.5	3.9	42.9	0.99	2.1	43.3	0.9	2.05	43.3	2	2.3	43.3
	1	3.3	43.0	0.9	2.05	43.3	0.8	2.1	43.3	3	2.2	43.3
	2	3.8	43.0	0.8	Fails	Fails	0.7	2.2	43.2	4	2.3	43.2
Football	0.5	2.4	43.6	0.99	1.9	43.3	0.9	1.7	43.3	2	1.8	43.3
	1	2.1	43.2	0.95	1.7	43.3	0.8	1.7	43.4	3	1.8	43.3
	2	2.1	43.3	0.8	1.8	43.3	0.7	1.8	43.3	4	2.0	43.2

The large variance in the parameter estimates due to the use of small forgetting factors is of little significance in ADPCM coding since the objective is to have small prediction errors. For the other images, the OBE algorithm, performs better than the LMS algorithm but does not do any better than the EWLS algorithm. For the Lena and Football images, the performance of the EWLS and the OBE algorithms is relatively insensitive to the variable settings (i.e., the forgetting factor and the noise bound). For the highly non-stationary zone plate image, the compression ratio is significantly affected by the settings used.

28.4. QUANTIZATION OF DCT COEFFICIENTS

The important problem of controlling quantization error in DCT based compression schemes can be approached from a bounded-error parameter-estimation perspective. Specifically, given that a linear relationship exists between the block DCT coefficients and the block image samples, we wish to find the permissible extent of variation in the DCT coefficients (the extent of quantization error) under the constraint that the resulting error in the image sample domain is upper bounded. Suppose a simple description of the permissible set of DCT coefficients can be obtained. The quantization for that block can then be adjusted so that the quantized coefficients remain within the permissible set to ensure that all the errors in the image samples in the block satisfy the given upper bound.

First, focus on the one-dimensional N-point DCT case. The forward DCT transform is defined as[7]

$$F(u) = \frac{2c(u)}{N} \sum_{j=0}^{N-1} f(j) Cos\left[\pi u \frac{2j+1}{2N}\right] \text{ for } u = 0, 1, \ldots, N-1 \tag{28.22}$$

while the inverse DCT transform is defined as

$$f(j) = \sum_{u=0}^{N-1} c(u)F(u) Cos\left[\pi u \frac{2j+1}{2N}\right] \text{ for } j = 0, 1, \ldots, N-1 \tag{28.23}$$

with $c(0) = 1/\sqrt{2}$ and $c(1) = c(2) = \ldots = c(N-1) = 1$.

The inverse DCT can be recast into the now familiar linear parameter model.

$$f(j) = \theta^{*T}\phi(j) \quad \text{for } j = 0, 1, \ldots, N-1 \tag{28.24}$$

where θ^* is the vector of the N DCT coefficients $F(0), F(1), \ldots, F(N-1)$ and $\phi(j)$ contains the cosine terms of the expansion of $f(j)$. Since θ^* is subject to quantization error before the inverse transform, the requirement

$$|f(j) - \theta^T\phi(j)| < \gamma, \quad j = 0, 1, \ldots, N-1 \tag{28.25}$$

is equivalent to determining the set

$$S = \{\theta \in \Re^n: \left|f(j) - \theta^T\phi(j)\right| < \gamma, j = 0, 1, \ldots, N-1\} \qquad (28.26)$$

The set S is thus the feasible parameter set which is actually the intersection of N half spaces in $\Re^N$. Since the DCT is unitary, the $\phi(j), j = 0, 1, \ldots, N-1$ are mutually orthogonal. Thus S is a rotated cuboid in the parameter space with the coordinates of the center of the cuboid being the N unquantized DCT coefficients and the length of each side being equal to 2γ.

At this juncture, it is useful to examine a commonly used DCT coefficient quantization strategy. The quantization of each coefficient is performed by dividing the coefficient by an integer and rounding the result. A base-line quantization table which contains N different divisors for the N coefficients is used. The divisors for the higher order coefficients are usually larger than the others. Depending on the extent of quantization desired, each entry in the quantization table is multiplied by a scaling factor greater than one, and used as the divisor for the corresponding coefficient. The higher order DCT coefficients are thus reduced to zero in case the signal does not have significant high frequency content. If there are high frequencies and a large scaling factor is used, however, then considerable error can occur in the reconstructed image samples.

Given a particular quantization table and scaling factor, ensuring that the quantized DCT coefficients belong to S would essentially require taking the inverse DCT transform of the quantized coefficients and checking that each reconstructed sample in the block of N samples is within γ of the original sample. Alternatively, to choose a scaling factor just small enough to satisfy the sample bound would require computing the inverse DCT for each scaling factor, and stopping when the bound is violated. It is desirable, therefore, to compute bounds on the quantization error in the DCT domain to ensure that the spatial domain constraint Eq. (28.25) is satisfied. The simplest bounds are parameter uncertainty intervals (PUIs)(13) with each interval independently specifying the uncertainty in that parameter. Other bounds, such as the ellipsoidal ones, could be used. However, ensuring that the quantized DCT coefficients are within the ellipsoid would require the same order of complexity as performing the inverse DCT. The PUI approach is equivalent to inscribing a parallelepiped with its faces parallel to the coordinate axes within the cuboid S. This problem can be solved in a variety of ways: solid geometry or convex programming.(14) One method(15) finds the minimal outer-bounding parallelepiped first and then scales its dimensions uniformly to obtain the inner bounding parallelepiped. The outer bounding parallelepiped can be obtained almost trivially by following this technique.(15) The PUI $\Delta \mathrm{F}(u)$ turns out to be proportional to the sum of the absolute values of the elements in the uth row of the forward transform matrix.

$$\Delta F(u) = \gamma \frac{2c(u)}{N} \sum_{j=0}^{N-1} \left| Cos\left(\pi u \frac{2j+1}{2N}\right)\right| \text{ for } u = 0, 1, \ldots, N-1 \qquad (28.27)$$

The inner PUIs can be obtained as

$$\Delta F^{inner}(u) = h\Delta F(u) \quad \text{for } u = 0, 1, \ldots, N-1 \tag{28.28}$$

where

$$h = \frac{\gamma}{\max\limits_{j} \sum\limits_{u=0}^{N-1} \Delta F(u)c(u)\,|\,Cos\left(\pi u \frac{2j+1}{2N}\right)|} \tag{28.29}$$

As an example, the outer and inner PUIs for $N = 4$ are listed in Table 28.2. Unfortunately, for $N > 2$, it is observed that the inner PUIs are very small and excessively pessimistic. As a check, the inner PUIs for N = 4 are also obtained by brute force constrained minimization of the criterion described in Ref. 14 and are very close to the ones listed in Table 28.2. Thus, it appears that inner PUIs do not provide a good means of adjusting the quantization in the DCT domain to constrain the error in the spatial domain. Parameter bounding techniques, however, are still of some use in this application. The bounds provided by the outer PUIs can provide an estimate of the permissible amount of coefficient quantization error. Clearly, quantization settings which causes these outer bounds to be exceeded causes the quantization error bound in the spatial domain to be violated. Thus outer PUIs can be used as a check on the DCT quantization error to improve the fidelity of the reconstructed imagery in critical applications, such as medical imaging and remote sensing.

The outer-bounding technique is easily extended to the two-dimensional case by scanning the rows of the image and DCT coefficient blocks from left to right, and top to bottom and forming vectors. The DCT and inverse DCT equations are slightly different for the two-dimensional case and can be obtained from Ref. 7. In this case

$$\Delta F(u,v) = \gamma \frac{2c(u)c(v)}{N} \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} |\,Cos\left(\pi u \frac{2j+1}{2N}\right)|\;|\,Cos\left(\pi v \frac{2k+1}{2N}\right)|$$

$$u = 0, 1, \ldots, N-1, v = 0, 1, \ldots, N-1$$

TABLE 28.2. Outer and Inner Bounds for the Four Point DCT

Coefficient Index u	Outer Bound ΔF(u)	Inner Bound $\Delta F^{inner}(u)$
0	1.41	0.38
1	1.30	0.35
2	1.41	0.38
3	1.30	0.35

For the standard 8 × 8 DCT case, the calculated outer PUIs for each coefficient range from 6.5 γ to 8.0 γ

28.5. CONCLUSIONS

Two applications of bounded-error parameter-estimation in image compression have been discussed. A time-varying parameter bounding estimator has been used for ADPCM coding of images. The performance of this estimator with one commonly used test image is much better than the standard LMS or EWLS techniques. However, with the use of dead-zoning, much smaller forgetting factors can be used for the EWLS algorithm and its performance can be improved dramatically. For other images, the performance of the OBE estimator is better than the LMS and comparable to the EWLS schemes. Parameter bounding has also been applied to an important application: DCT coefficient quantization. Parameter uncertainty intervals for the DCT coefficients have been obtained to decide which quantizer scaling factor to use for a particular block. It is expected that these uncertainty intervals will be useful in scientific applications where it is important to keep the coding error within bounds.

REFERENCES

1. E. Walter and H. Piet-Lahanier, in: Vol. 1 of *Proceedings of the 1993 IEEE International Symposium on Circuits and Systems*, pp. 774–777, Chicago, IL (1993).
2. J. R. Deller, *IEEE Signal Process. Mag.* **6**, 4 (1989).
3. P. L. Combettes and H. J. Trussel, *IEEE Trans. Acoust., Speech, Signal Process.* **37**, 393 (1989).
4. A. Abo-Taleb and M. M. Fahmy, *IEEE Trans. Circuits Syst.* **31**, 801 (1984).
5. A. K. Rao, *Membership-Set Parameter Estimation via Optimal Bounding Ellipsoids*, Ph.D. Dissertation, University of Notre Dame, South Bend, IN (1990).
6. A. K. Rao and Y. F. Huang, *IEEE Trans. Signal Process.* **41**, 1140 (1993).
7. N. Ahmed, T. Natrajan, and K. R. Rao, Discrete Cosine Transform, *IEEE Trans. Comput.* **C-23**, 90 (1974).
8. S. Dasgupta and Y. F. Huang, *IEEE Trans. Inf. Theory* **33**, 383 (1987).
9. E. Fogel and Y. F. Huang, *Automatica* **18**, 229 (1982).
10. A. K. Rao, Y. F. Huang, and S. Dasgupta, *IEEE Trans. Acoust., Speech, Signal Process.* **38**, 447 (1990).
11. N. Jayant and P. Noll, *Digital Coding of Waveforms*, Prentice-Hall, Englewood Cliffs, NJ (1984).
12. G. C. Goodwin and K. S. Sin, *Adaptive Filtering, Prediction and Control*, Prentice Hall, Englewood Cliffs, NJ (1984).
13. M. Milanese and G. Belforte, *IEEE Trans. Autom. Control* **27**, 408 (April 1982).
14. A. Vicino and M. Milanese, *IEEE Trans. Autom. Control* **36**, 759 (1991).
15. R. Pearson, *SIAM J. Algebraic Discrete Methods* (1988).

29

Applications of OBE Algorithms to Speech Processing

John R. Deller, Jr.

29.1. INTRODUCTION

29.1.1. Application Areas

Many algorithms for identification of speech models are directly or indirectly based on linear predictive coding (LPC) analysis.[†] LPC analysis is tantamount to identification of an autoregressive (AR) model using short-term batch processing of the observations.[1] The LPC model, therefore, is a special case of the discrete-time linear-in-parameters models treated in foregoing chapters. Accordingly, many speech processing tasks represent natural domains for applying bounded-error methods. This chapter discusses the fundamental principles requisite to application of optimal-bounded-ellipsoid (OBE) processing to problems in speech analysis, recognition and coding. The focus is the general problem of LPC identification of speech using OBE methods, including the significant issue of tracking the time-varying parameters of this very dynamic signal. Potential applications of this work in specific speech-processing endeavors include:

[†] The term linear predictive *coding* is often used whether or not coding is the issue. It would be more desirable to use simply "linear prediction" in this work, but this leads to the acronym "LP" which has been used extensively in this book to mean *linear-in-parameters.*

JOHN R. DELLER, JR. • Department of Electrical Engineering, Michigan State University, East Lansing, MI 48824.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

1. General modeling and analysis by predictive methods for spectral (formant) estimation, pitch detection, glottal waveform deconvolution, and pathology detection.[1]
2. Automated recognition of speech in which LPC parameters, or related parameters to which LPC coefficients are converted, are used as features in classifying phones, words, or complete messages in isolated utterances or continuous speech.
3. Speaker recognition, or speaker verification, in which the speaker's identity is determined or verified, respectively, through parametric feature analysis.
4. Compression and synthesis of speech in which LPC parameters are used in strategies which remove redundancy in the acoustic waveform as a means of bandwidth compression or improving storage requirements. Similarly, spectral compression based on LPC analysis can be used for translation of the spectrum for hearing aids.

Whereas OBE, and, more generally, bounded-error, techniques, are usually presented as means for obtaining *set* estimates rather than *point* estimates of the system parameters, this application exploits the OBE processing principally as a means of reducing unnecessary computation for real-time processing. Other benefits obtain from the OBE approach, such as enhanced adaptation and more accurate estimates. The resulting set estimate is not the focus of this work, although its existence may prove useful in some applications.

Although several research groups have studied techniques for temporal selection of data points in LPC analysis,[2–11] the thrust of each of these efforts has been to remove effects of the glottal excitation from the parameter estimate. Such deconvolution measures are useful for some tasks, but LPC parameters obtained without them are adequate or necessary in many applications. This chapter considers temporal data selection of a very different sort in which the estimation goal is the "usual" LPC result obtained without selection. Data selection is not used to eliminate glottal, or any other, effects. Rather, data are chosen and weighted on the basis of their potential to improve upon the existing estimate as a temporal recursion proceeds.

Speech is widely recognized to be a redundant process in terms of its temporal correlations. This redundancy is also manifest in the number of data which are uninformative in the sense of refining the OBE set estimate. OBE identification can reduce the adverse effects of this redundancy by discerning informative data, thereby avoiding the expense of updating at times containing no innovation.

29.1.2. Relationship of OBE Algorithms to the LPC Covariance Method

In speech processing, LPC estimates are ordinarily obtained through batch processing without data weights (called α_k and β_k in Chapter 4).[1] OBE estimates are computed recursively and a number of different weighting strategies are employed as discussed in Chapter 4 (also Refs. 12–14). Further, since this work is motivated by an interest in real-time signal processing, it may be useful to carry out the OBE recursions (see Chapter 4) using an alternative set of computations which are theoretically equivalent, but which can be implemented using contemporary parallel-processing technology. Such an approach is described below. Finally, real-time applications benefit from a more computationally efficient, but suboptimal, test for innovation at each instant. Such a test is also described below.

For a given elemental speech sound (a phone), the speech production system is well modeled by a set of damped resonances, called formants by speech scientists. Accordingly, the AR model is often used to model the speech waveform. This system is excited by a discrete-time unit sample train of appropriate pitch period for "voiced" phones, and by discrete-time white noise for "unvoiced" phones.[†] Let the real, scalar speech sequence be denoted $\{s_t\}$. The "true" model is given by

$$s_t = \sum_{i=1}^{n} a_i s_{t-i} + \omega_t = \mathbf{s}_t^T \mathbf{a}^* + \omega_t \quad \mathbf{s}_t, \mathbf{a}^* \in \mathbb{R}^n \tag{29.1}$$

where a pointwise bound is assumed known on the excitation,

$$|\omega_t| \leq \delta_t \tag{29.2}$$

for all $t > 0$. Comparing with Chapter 4, this model is a special case of the regressor model to which OBE identification can be applied.

The most widely-used algorithms for estimating the parameters $\mathbf{a}^*$ of such an AR model employ the least square error (LSE) criterion as the measure of optimality. Given observations s_t, $t = 1, \ldots, k$, the estimate, say $\mathbf{a}_k$, is sought such that

$$\xi(k) = \sum_{t=1}^{k} w_{k,t} e_{k,t}^2 = \sum_{t=1}^{k} w_{k,t} (s_t - \mathbf{a}_k^T \mathbf{s}_t)^2 \tag{29.3}$$

is minimized, where $\{w_{k,t}\}$ is a set of generally k-dependent (time-varying) error weights. The sequence (on t) $\{e_{k,t}\}$ is the prediction error sequence[‡] associated with parameter estimate $\mathbf{a}_k$. The so-called "covariance" method of LPC analysis seeks an unweighted ($w_{k,t} = 1$, $\forall\, k,t$) solution to this problem through batch processing.[1] No bounding information on the sequence $\{\omega_t\}$ is employed, even if known. On

[†] A voiced phone occurs when the larynx produces a periodic lowpass pulse train which excites the vocal tract. An *unvoiced* phone is excited by turbulence created by forcing air through a constriction in the vocal tract. The difference can be heard in the two sound in the word "is." More details about speech modeling can be found, for example, in Ref. 1.

[‡] The number $e_{k-1,k}$ is the residual associated with $\mathbf{a}_{k-1}$. In Chapter 4, this number is denoted e_k.

the other hand, the OBE method recursively seeks a LSE solution with special weights $\{w_{k,t} = \beta_t^*/\alpha_t^*, \forall k\}$ that are determined using the bounding information. Therefore, when an OBE algorithm is employed over the data range $t \in [1, k]$, the resulting estimate is theoretically equivalent to the covariance estimator on the same data if optimal weights $\{\beta_t^*/\alpha_t^*\}$ were used in the LSE minimization. OBE processing of speech, therefore, can be interpreted as a recursive covariance analysis with a special weighting strategy arising from the bounded-error information.

29.2. ANALYSIS OF STATIONARY SPEECH FRAMES

29.2.1. Introduction

From a real-time speech processing point of view, the central feature of the OBE approach is that it avoids unnecessary updating when a datum is insufficiently informative. This section explores this behavior of the algorithm and reports additional experimental findings of practical importance.

The main database analyzed for presentation in this section consists of three vowel sounds /a/ (as in the standard American pronunciation of "hot"), /i/ ("heed"), and /u/ ("hoot"), spoken by each of three adult male speakers, and two unvoiced sounds /s/ ("see") and /f/ ("fee"), spoken by two adult male speakers. The unvoiced utterances are clipped from renditions of the words "six" and "four." A phonetically diverse group of vowel sounds are included to capture the diversity in behavior of the method in the voiced case, while the unvoiced results are relatively phone-independent and two examples are sufficient. The general findings described here are distilled from several hundred experiments with various voiced and unvoiced sounds.

The volume-minimization measure is used here to determine optimality. For volume and trace OBE algorithms, most weighting strategies are equivalent in the senses of estimates produced, data selected for updating, and ellipsoid volume (or trace) at each step.[13,14] For simplicity, therefore, we employ the algorithm with weights $\alpha_k = 1 \ \forall k$ and optimize over $\beta_k = \lambda_k$ at each step. This OBE algorithm has been called set-membership-weighted recursive least squares (SM-WRLS),[13,15] and has been independently investigated.[16]

29.2.2. Estimation of $\{\delta_t\}$ and Related Issues

Speech analysis involves the important task of estimating the error-bounding sequence $\{\delta_t\}$. This is a key issue in experimental usage of OBE methods as an inappropriately small δ_t (in principle, even at one t) can cause the estimation process to diverge. The lack of *a priori* knowledge of $\{\delta_t\}$ makes this real-world problem fundamentally different from simulation studies described in this volume and elsewhere in the literature.

To understand the importance of $\{\delta_t\}$ estimation, we give a heuristic explanation of how optimal weights are chosen. The level of importance (magnitude of λ_k^*) ascribed an incoming data set at time $t = k$ depends fundamentally upon two quantities. The first is the innovation or residual, $e_{k-1,k}$, which can be expressed as

$$e_{k-1,k} = \omega_k + (\mathbf{a}^* - \mathbf{a}_{k-1})^T \mathbf{s}_k. \tag{29.4}$$

The second factor is the amount of credence placed in the existing estimate, measured in terms of the "size" of the covariance matrix, $\mathbf{C}_{k-1}$, as (inversely) reflected in the scalar $G_k = \mathbf{s}_k^T \mathbf{C}_{k-1}^{-1} \mathbf{s}_k$ (see Chapter 4). A large value of $e_{k-1,k}$ tends to increase λ_k while a small value of G_k (which decreases as the process proceeds) tends to deemphasize the current point. For a fixed G_k, $e_{k-1,k}$ determines the relative information content of the current point. This becomes clear when $e_{k-1,k}$ is interpreted as an estimate of the input ω_k with an error term due to error in the parameter estimate. Since $|\omega_k| \leq \delta_k$, large values of $|e_{k-1,k}|$ (near or exceeding δ_k) are the consequence of an inaccurate set of parameters signaling "correction needed" in $\mathbf{a}_{k-1}$; that is, s_k carries "innovation." The data set at k is more heavily weighted and the confidence in the estimate (ellipsoid size and G_k) correspondingly decreases.

An inappropriately small δ_k can cause the algorithm to diverge. This happens when at time k, $|\omega_k| >> \delta_k$ and the data set is mistaken by the algorithm as "information laden" and heavily weighted. The large weight can cause the current point to have undue influence on the overall estimate, and the effect of any one point is always potentially destructive (causing the estimate to move away from the true parameters). The assumed worth of the point, however, can cause a serious shrinking of the ellipsoid which might be centered on a bad estimate. This shrinking can, in turn, preclude "good" data from further influencing the estimate as the confidence in the estimate is now large. A sufficiently large δ_k, therefore, is essential at each k. On the other hand, it is desirable to keep δ_k as small as possible for a given k, in order to maximize the use of informative points and speed up convergence. Deliberately choosing δ_k to be smaller than is justified, however, does not speed convergence, but instead wastes computation processing points, and worse, may cause the algorithm to diverge. Experimental trials have indeed borne out the need for extreme caution in choosing the sequence $\{\delta_t\}$. Particularly at an early time in the identification when the algorithm is eager to heavily weight informative data, the choice of an incorrect δ_k can doom the estimation.

Deller and Luk[15] explore a number of procedures for estimating $\{\delta_t\}$. In every case, so that relative values are meaningful, the estimation begins by normalizing the speech sequence so that its maximum absolute amplitude is unity. One simple estimate of δ_k is the magnitude value of the speech itself at $t = k$. This method is justified in the deterministic case, where the model includes excitations only at points for which "initial conditions" are small, i.e., when $\omega_k \neq 0$, $s_k \approx \omega_k$. So it is approximately true that $|s_k| \leq \delta_k$ if $|\omega_k| \leq \delta_k$. Of course, δ_k can be made a little larger than "necessary" by using $\delta_k = c|s_k|$, where c is some, possibly time-varying, number greater than unity. As a variation on this idea, the short-term average magnitude of the speech on some small window around (or prior to, or following) k can be used in an attempt to track the envelope of the speech. Other estimates, which are theoretically appropriate in either the voiced or unvoiced case, include a short-term estimate of the standard deviation of the sequence (on t) $\{e_{k-1,t}\}$. This estimator is appropriate because $e_{k-1,k} \approx \omega_k$ for each k. The approximation becomes better as the identification proceeds, assuming that the estimate is correctly converging. When $e_{k-1,k}$ is significantly different from ω_k, it is larger in average square, so that a conservative estimate results. This estimate is obtained by setting λ_t to unity over some short initial window, of duration, say, r, and using the average magnitude of the residual as an estimator. The sequence $\{\delta_t\}$ can then be fixed at some constant, δ, such that

$$\delta = c\frac{1}{r}\sum_{t=n+1}^{n+r} |e_{t-1,t}| \tag{29.5}$$

where c is some constant, usually about six. The constant δ, so estimated, can also be made to fluctuate with the signal level so that more reasonable local bounds are achieved. Each of these techniques is variously successful. The result sometimes depends upon the particular waveform analyzed. In order to discuss the general applicability of the OBE method to speech analysis, the focus here is on one very simple method of estimating $\{\delta_t\}$ which proves to be quite effective generally.

The estimate used at time k is

$$\delta_k = \max\{\gamma^k K, 1\} \times |e_{k-1,k}| \tag{29.6}$$

where K is some constant, typically four, and γ is some number slightly less than unity. The factor in front of $|e_{k-1,k}|$ is used to provide a "margin of error," especially at early ks, for the inaccuracy of the approximation $e_{k-1,k} \approx \omega_k$.

An estimate of $\{\delta_t\}$ must be bounded conservatively from below. Allowing $\{\delta_t\}$ to track the residual (or signal) into the low-level regions of a voiced cycle or at small values of an unvoiced frame, can lead to numerical instability of OBE algorithms with nondecreasing weights (like SM-WRLS), and to divergence generally. With SM-WRLS, such tracking leads to large optimal weights followed by plummeting ellipsoid volumes and spiraling upward weights, eventually leading to complete disintegration of the estimation process. It is therefore necessary to restrict

$$\delta_k \geq \delta_{\min} \tag{29.7}$$

for all k, with $\delta_{\min}$ typically equal to 0.3 (recall normalization of the waveform). The $\delta_{\min}$ precaution prevents numerical instability due to overemphasis of the importance of small values. The small ellipsoid, large weight cycle leading to numerical problems, can persist in general. It is prudent, and necessary therefore, to restrict the size of the weights λ_k^* to $\lambda_{\max}$, typically four. This means that the bounding considerations could deem a point at most twice (or four times in terms of the squared-error minimization) as informative as the nominal contributions of the initial n points more for which λ_k is usually fixed at unity. Furthermore, it is useful to attenuate $\lambda_{\max}$ as time progresses. Points in the long term are unlikely to bring large amounts of new information about a stationary process. Large weights are likely to be indicative of the pending numerical breakdown of the process. Therefore, at a time when the optimization computation occurs and $\lambda_k^* > 0$:

$$\lambda_k^* \leftarrow \min\{\lambda_k^*, \max[\rho^k \lambda_k^*, 1]\}. \tag{29.8}$$

The optimal weight determined from the OBE optimization is replaced by the value on the right side of the arrow: ρ, like γ in Eq. (29.6), is ≈ 1, but < 1. This "interference" with the optimization process has little effect empirically on the process. Examples and further discussion of these phenomena are given below.

In the experiments below, for times k when optimization is carried out: δ_k is set as in Eq. (29.6) with $\gamma = 0.992$ and $K = 2.0$, but is bounded below by $\delta_{\min} = 0.3$; and λ_k^* is modified as in Eq. (29.8) with $\lambda_{\max} = 4$ and $\rho = 0.997$.

29.2.3. Voiced-Case Experiments

Since the development of OBE techniques has been implicitly based upon a model with stochastic excitation, it is difficult to predict *a priori* how the technique might perform in the voiced case. Indeed, nothing in the OBE developments precludes the algorithms' use on "deterministic" waveforms as long as the error-bounding condition is satisfied. The uncertainty in predicting performance is based on the deviation of most voiced waveforms from the over-idealized pulse-driven LPC model.[1] If voiced speech can be modeled exactly as purely AR driven by a periodic pulse train, then the use of n data points (chosen to avoid the excitation time) is sufficient to exactly identify the n parameters. Sometimes n data points are used in identifying a voice frame due to the many nonideal effects, such as glottal coupling and nonminimum-phase effects.[1] The fact that the LPC model is effective, however, indicates that it is substantially correct and that there must be a significant amount of redundant information is a voiced frame of length, say, $20n$. How can a relatively small number of informative points could be selected while retaining enough points to smooth the effects of the model error? The experiments below show that OBE algorithms have the potential to accomplish this task.

Three vowel sounds /a/, /i/, and /u/ comprise the basis for this discussion. Other voiced phones have been analyzed with similar results. The problem is to identify a 14-order LPC model on a 256-point (25.6-ms) frame of speech. In each case, the sound is assumed unknown *a priori*, so *ad hoc* algorithm-parameter settings must be fixed at values general enough for any incoming phones. These are given at the end of Section 29.2.2. The general findings for the experiments are typical:

1. The *average* number of data selected on a frame by OBE processing over all experiments is 28.4% of the total, the high being 34.4% and low 23.8%.
2. In every case, the spectrum of the resulting OBE-based model is acceptable for most practical purposes (relative to the conventional covariance (COV) result[1] and in seven of nine cases, the spectral estimate is very good or excellent.
3. In seven of nine cases the conventional COV method is unable to produce an acceptable spectrum using the same number of points as the OBE method selected on the same frame. This indicates that the bounded-error strategy is indeed making good use of informative data and weighing them to advantage.
4. When a small number of points (equivalent to the OBE selected number) is used by COV, the problem with four of the unacceptable spectra is excessive resonance indication (narrow bandwidths) at one or more of the formants. This same problem occurs (to a lesser extent) in two of the OBE spectra. When OBE "fails," this excessively resonant spectrum is frequently the manifestation. (Such a spectrum might be useful to obtain formant frequencies which seem to be highly accurate.) *Ad hoc* parameter settings which push the total number of selected points below the levels used here (~ 30%) begin to cause this problem in general.
5. OBE ellipsoid volumes are always terminally better than those of COV analysis.

Figs. 29.1(a) through 29.1(e) elaborate on these conclusions with results from specific experiments. They show five spectral plots for representative cases. In each case, the top curve is the speech spectrum based on a 512-point FFT with a Hamming window; the second is the SM-WRLS spectrum; the third is the standard COV spectrum using the entire frame of data; and the fourth and fifth show the results of applying COV (in two slightly different ways to be explained) to the same percentage of data used by SM-WRLS. Each identified spectrum is obtained by computing the 512-point FFT of the sequence $\{1, -a_1, -a_2, \ldots, -a_{14}, 0, 0, 0, \ldots, 0\}$, then taking the inverse magnitude spectrum, where a_i is the ith estimated LPC parameter. In each case, arbitrary offsets have been added to the curves to separate them into the indicated order.

The two /a/ spectra (Figs. 29.1(a) and 29.1(b)) are typical of good SM-WRLS outcomes. In these two experiments SM-WRLS selected 23.8% and 28.9% of the data on the 256-point frames, respectively. The SM-WRLS spectra are nearly identical to that using COV on the entire frame. The two additional curves in each case represent inferior spectral estimates, especially for speaker J (Fig. 29.1(b)). In the first of these additional curves (COV (·· %)) the reduced number of points is

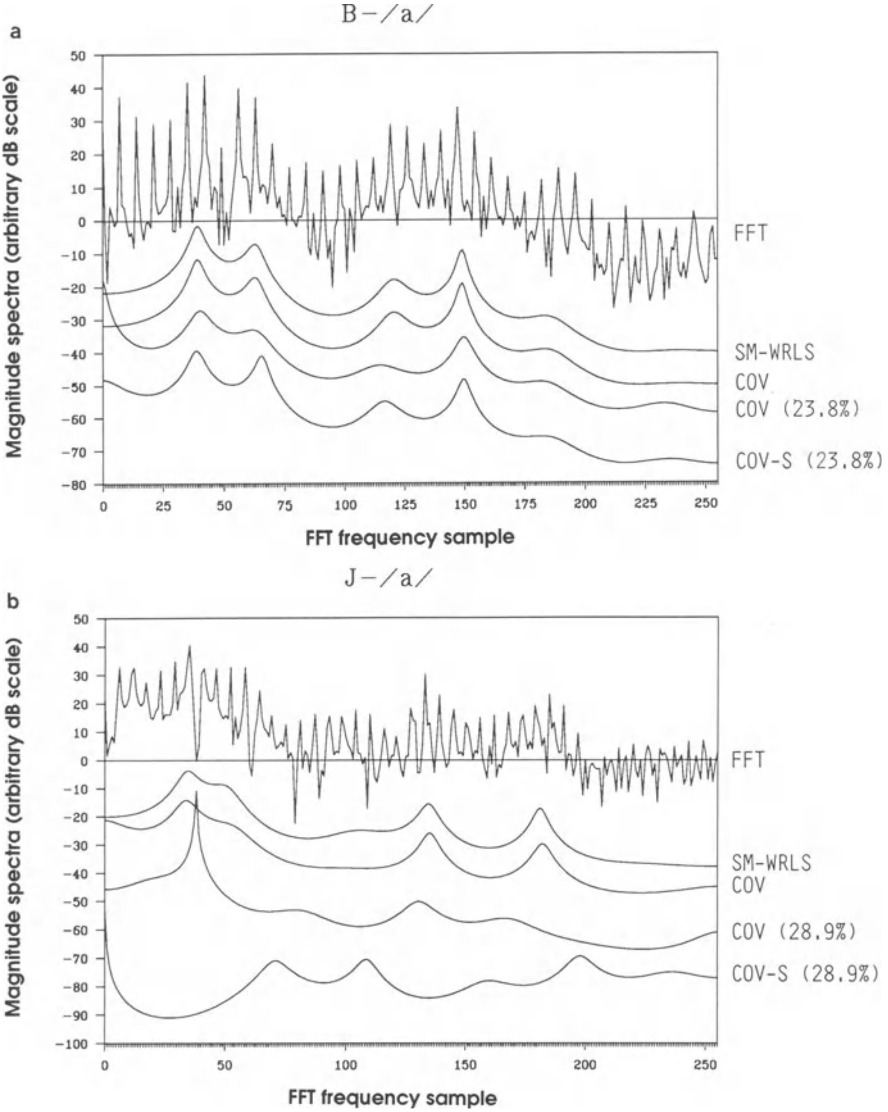


FIGURE 29.1. Spectral results for seven representative experiments. In each case, the top curve is the original speech spectrum based on a 512-point FFT following Hamming windowing. The remaining curves from top to bottom are spectra based on SM-WRLS, COV (over the entire range), COV using the same number of data as SM-WRLS (percentage of data used is shown), and COV using the same-number as SM-WRLS plus point-skipping. (The fifth curve is not shown in all cases.) In the voiced case, the data range includes 256 points; unvoiced, 380; (a) Speaker B, vowel /a/; (b) Speaker J, vowel /a/; (c) Speaker J, vowel /i/; (d) Speaker G, vowel /i/; (e) Speaker J, vowel /u/; (f) Speaker J, fricative /s/; and (g) Speaker J, fricative /f/.

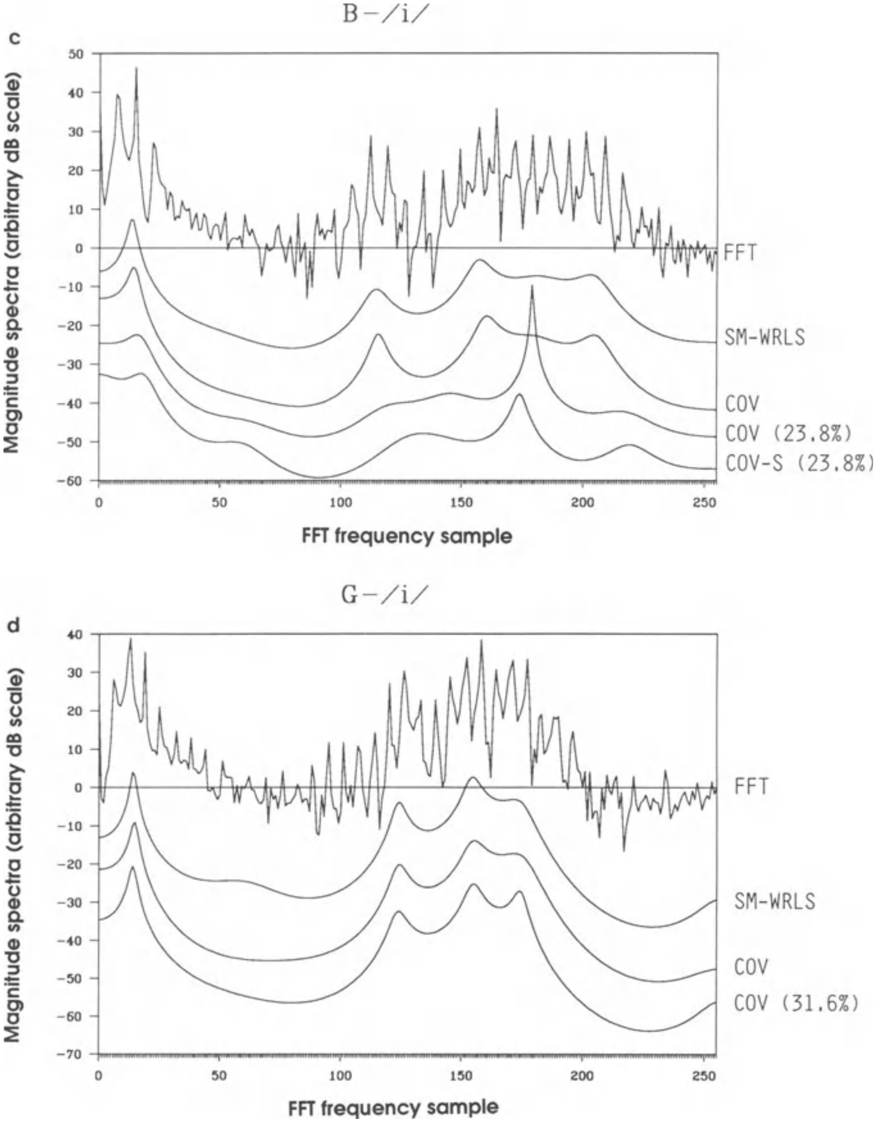
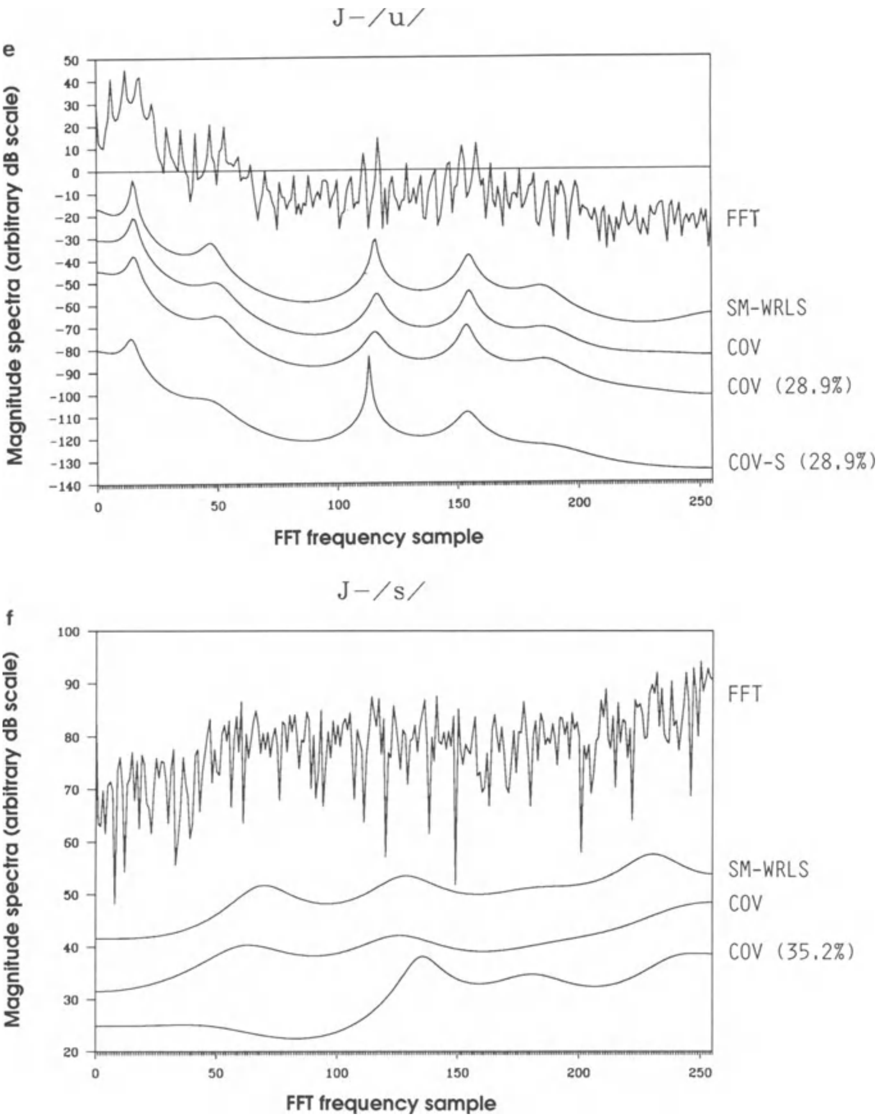


FIGURE 29.1. (Continued)



J−/u/

J−/s/

FIGURE 29.1. (Continued)

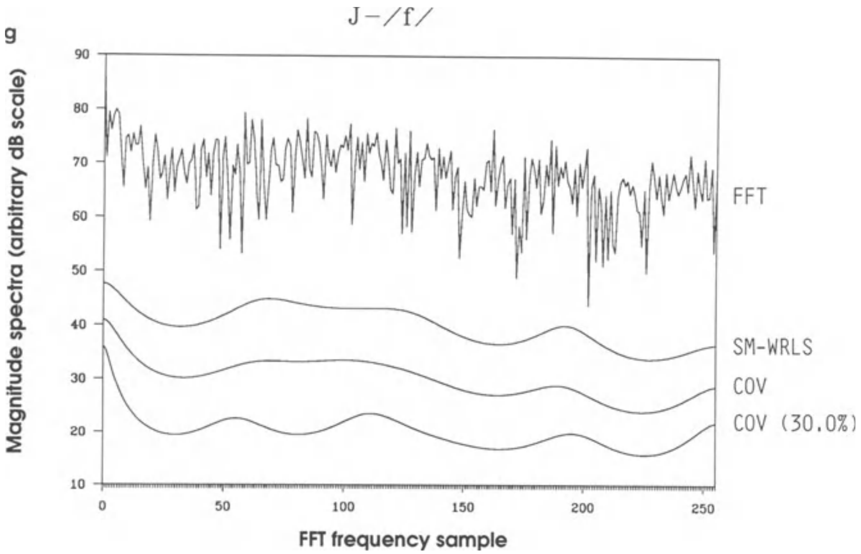


FIGURE 29.1. (Continued)

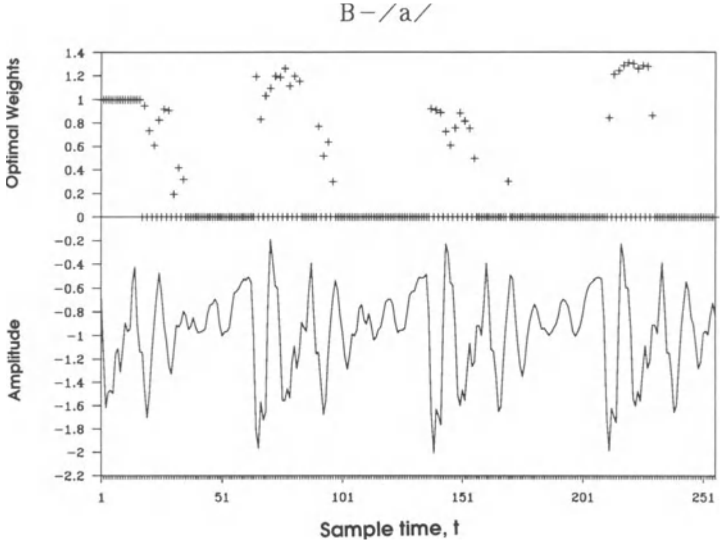


FIGURE 29.2. A typical case of the speech waveform and the optimal weights, $\{\lambda_t^*\}$ (corresponding to the experiment of Fig. 29.1(a); speaker B, vowel /a/).

simply taken from the beginning of the frame. In the second curve (COV-S (·· %)) the reduced number of points is taken from the beginning of the frame, skipping every other point following the initial $n + 1$. The reason for this second test is as follows. Since SM-WRLS tends to select points in clusters, the algorithm is automatically required to discard every other point in order to force a more diverse view of the temporal dynamics. This procedure is of some benefit in a number of experiments, never increases the number of points selected, and never degrades the spectral estimate. The COV-S run is to be sure that the good spectrum in the SM-WRLS case with respect to COV (whole frame) is not an artifact of this "skipping" procedure. Indeed it is seen not to be.

Fig. 29.2 illustrates a typical case of the speech waveform and $\{\lambda_t^*\}$ weights (corresponding to the experiment of Fig. 29.1(a): speaker B, vowel /a/) and Fig. 29.3 shows the log ellipsoid volume plot from SM-WRLS and COV for the same experiment. A good estimate is obtained quickly and small residuals are produced; the optimization process shows little interest in the points on the frame. Evidence of this is seen in the log volume curve in which both SM-WRLS and COV achieve the same small volume in the first cycle. Some incremental improvements in the volume are seen in the second and third cycles for SM-WRLS. Note the *increases* in the ellipsoid in the COV case over the second and third cycles. SM-WRLS never accepts data sets which worsen the volume, whereas COV often does.

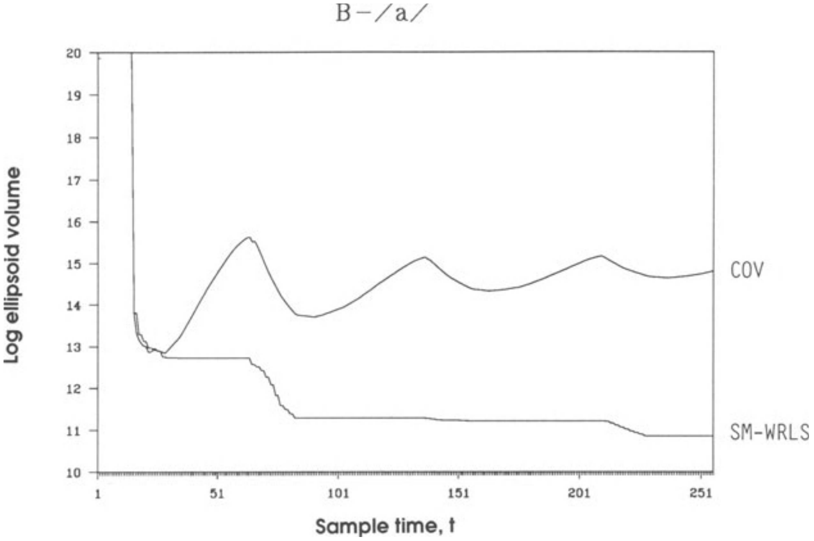


FIGURE 29.3. Log ellipsoid volume plot from SM-WRLS and COV for the experiment of Fig. 29.1(a).

29.2.4. Unvoiced Cases

Typical unvoiced case spectral results are shown in Fig. 29.1(f) and (g) for the phones /s/ from an utterance of "six" and /f/ from "four." SM-WRLS uses 35.2% and 30.0% of 380-point data frames (see below), respectively. In experiments with many sounds, such as unvoiced phones extracted from the words "fuzzy," "church," "shoelace," "three," and "car," acceptable spectra are always produced by SM-WRLS. Again, the COV approach with reduced data almost never produces acceptable results.

The ease of use is much greater and number of nuances of algorithmic behavior much lower in the unvoiced case since the excitation sequence $\{\omega_t\}$ is nominally stationary white noise for which bounds are more easily estimated. Seldom does SM-WRIS converge to an acceptable spectral estimate on frames of 256 points. Since the optimization process typically discards two-thirds or more of the data, there are simply not enough points that SM-WRLS selects to produce a reasonable estimate, in spite of "optimal" weighting. In these unvoiced experiments, therefore, the frame size is extended to 380 points (38 ms). A second relatively minor factor emerging in the unvoiced case experiments is the effect of model size n. The ellipsoid volume is exponential in n.(17) When n is overestimated, this occasionally adversely affects the selection of points (too many unfruitful points taken as the algorithm optimizes over unnecessary dimensions) and increases the tendency toward numerical instability. For this reason, n is reduced to ten in the unvoiced case. This factor is not difficult to deal with in real applications.

29.2.5. Further Discussion and Conclusions

Analysis of stationary frames of speech indicates that many speech data carry redundant information in terms of their ability to refine the set of parameter vectors to which their LPC model must belong. OBE algorithms select only those points which can improve the LPC estimate in this sense, and optimally weigh the incoming data to use the innovation most effectively. The use of OBE analysis of speech presents a challenge: an error-bounding sequence $\{\delta_t\}$ can only be estimated because only the output of the speech system can be measured. Furthermore, there is a need to protect the estimation from numerical instabilities and divergence arising from improper bounds. Some general guiding principles are discussed above for one OBE method, SM-WRLS. With proper care taken in estimating $\{\delta_t\}$ SM-WRLS has shown strong potential for accurate parameter estimates using relatively small numbers of data, even with rather conservative procedures. Because all OBE algorithms are identical in certain important senses,(13,14) one may infer similar optimism about the use of the general class of techniques.

29.3. ENHANCEMENTS OF OBE METHODS FOR REAL-TIME SPEECH PROCESSING

We now turn our attention to some algorithmic enhancements which are necessary and beneficial to speech and other signal processing tasks. Two problems that arise in speech processing require such enhancements. First is the issue of tracking quickly time-varying signals which is not supported by the underlying OBE theory. Second, for OBE-based LPC algorithms to be computationally competitive with existing batch methods, their computational load must be reduced to $O(n)$ floating-point operations (flops) per sample time. Although other useful advantages of OBE over standard batch methods can be demonstrated, speech processing tasks often demand real-time operation. Solutions to the tracking and computational-load-reduction problems are discussed below before beginning the study of speech in earnest. The developments below are easily generalized to the broader regressor model discussed in Chapter 4.

29.3.1. Alternative OBE Recursions

The recursive formulation for OBE processing presented in Chapter 4 can be interpreted as the well-known weighted recursive least squares (WRLS) algorithm with a special set of weights $\{w_{k,k} = \beta_k^*/\alpha_k^*\}$. Since conventional WRLS is based on the matrix inversion lemma (MIL)[18] this underlying algorithm is referred to as MIL-WRLS. Several benefits accrue from the use of a different WRLS algorithm for computing the OBE estimates. This is derived by returning to the fundamental batch solution. The OBE solution is equivalent to the LSE solution of the following overdetermined system of equations[19]

$$\mathbf{Q}_k \begin{bmatrix} \mathbf{s}_1^T \\ \mathbf{s}_2^T \\ \vdots \\ \mathbf{s}_k^T \end{bmatrix} \mathbf{a}_k = \mathbf{Q}_k \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_k \end{bmatrix}, \tag{29.9}$$

denoted

$$\bar{\mathbf{S}}_k \mathbf{a}_k = \bar{\mathbf{s}}_k, \tag{29.10}$$

where $\mathbf{Q}_k$ is a $k \times k$ diagonal matrix with tth diagonal element $\sqrt{\beta_t^*/\alpha_t^*}$. The well-known solution is given by

$$\mathbf{a}_k = [\bar{\mathbf{S}}_k^T \bar{\mathbf{S}}_k]^{-1} \bar{\mathbf{S}}_k \bar{\mathbf{s}}_k. \tag{29.11}$$

The quantity in brackets is called the (weighted) covariance matrix in the signal processing literature. Denote this matrix by $\mathbf{C}_k$, the inverse of matrix $\mathbf{P}_k$ in the MIL-WRLS version of Chapter 4.

One family of techniques which serves as the basis for current systolic-array solutions of this problem is based on the orthogonal triangularization of the $\bar{\mathbf{S}}_k$ matrix by coordinate rotation methods.(20–22) The procedure, in principle, involves the application of a sequence of orthonormal (Givens) operators to both sides of Eq. (29.9) that leaves the system in the form

$$\begin{bmatrix} \mathbf{T}_k \\ \mathbf{0}_{k\times n} \end{bmatrix} \mathbf{a}_k = \begin{bmatrix} \mathbf{d}_{1,k} \\ \mathbf{d}_{2,k} \end{bmatrix}, \tag{29.12}$$

where $\mathbf{T}_k$ is $n \times n$ upper triangular, $\mathbf{d}_{1,k}$ is an n-vector, and $\mathbf{0}_{k\times n}$ denotes the $k \times n$ zero matrix.[†] This procedure amounts to the "QR" decomposition of the matrix $\bar{\mathbf{S}}_k$ discussed in any book on the theory of matrices.(19) The system $\mathbf{T}_k\mathbf{a}_k = \mathbf{d}_{1,k}$ is easily solved using back substitution to obtain the LSE estimate, $\mathbf{a}_k$. By appropriate data handling, this procedure can be carried out sequentially using approximately $(n + 1)/2$ storage locations.(22) Briefly, an $(n + 1) \times (n + 1)$ matrix, say $\mathbf{W}$, is initially filled with zeros. Then for each $t = 1, 2, \ldots, k$, only the bottom row of $\mathbf{W}$ is replaced by the row $\sqrt{\beta_t^*/\alpha_t^*}\ [\mathbf{s}_t^T \mid s_t]$. This "new equation"[‡] is integrated into the upper n rows of $\mathbf{W}$ using an orthonormal operation to leave the result $[\mathbf{T}_k \mid \mathbf{d}_{1,k}]$ in the upper n rows. Since the orthonormal operation consists of a sequence of coordinate rotations, the new equation is "rotated" into the existing set of equations. What remains at the bottom of $\mathbf{W}$ is a row of zeros except for location $(n + 1, n + 1)$ which contains the total squared error $\xi(k)$ of Eq. (29.3). This lowest row is replaced at the next step. When $k \geq n + 1$, the system $[\mathbf{T}_k \mid \mathbf{d}_{1,k}]$ can be solved for $\mathbf{a}_k$ for any k desired. A past equation can also be rotated *out* of the estimate if necessary with a simple modification to the algorithm. This feature is useful for adaptation to time-varying dynamics. Refer to this estimation process as "QR-WRLS."

The OBE considerations are handled through the optimal data weights α_t^* and β_t^*, which, in turn, implies a selection of informative data points for processing. The optimal λ_k, λ_k^*, is found using equations in Chapter 4 for the volume- and trace-minimization algorithms, respectively. In turn, λ_k^* implies optimal values of α_k and β_k. The specific choice of OBE weighting strategy and optimization criterion for this work is described in Section 29.2.

29.3.2. Adaptation for Time-Varying Speech Signals

Speech is a strongly time-varying signal. It is generally assumed that the speech model can remain stationary for short-term frames of 10–20 ms, but a more

[†] For convenience, n equations of the form $\mathbf{0}_{1\times n}^T\mathbf{a}_k = 0$ are appended to the top of system (29.9) prior to beginning the processing. This explains the differences in dimensions between (29.9) (or (29.10)) and (29.12).

[‡] Because of this formulation, the pair $(s_t, \mathbf{s}_t)$ could appropriately be called an *equation* in many contexts in the following. This term is not always satisfactory, however. Whereas the term "datum" is inappropriate to describe $(\mathbf{s}_t)$, and "data" can be misleading, we will frequently refer to this pair as the *data set* at time t. The expression "per t" should be interpreted to mean "per data set."

realistic model would include dynamics that are almost constantly changing. While "stationary" OBE algorithms have been observed to have inherent and fortuitous adaptive capabilities as a result of their optimal weighting strategies, these capabilities are unpredictable. While little is understood about the behavior of OBE algorithms in the presence of time variance, some theoretical results appear in the literature. Rao and Huang(23) recently report the most "practical" of these results and Huang and Deller(24) generalize them. In these papers, the theoretical "tolerance" for time variance at time k is given in the following terms. Suppose the ellipsoid at time $k-1$, $E(k-1)$, is given (see Chapter 4). In order for the true time-varying parameters, say $\mathbf{a}_k^*$, to remain inside the next ellipsoid, $E(k)$, under the usual optimization process, it must be true that that $\mathbf{a}_k^*$ is an element of the set $E(k-1)$ with the ellipsoid scalar κ_{k-1} (called σ_{k-1}^2 in Chapter 4) replaced as

$$\kappa_{k-1} \leftarrow \kappa_{k-1} + \frac{\beta_k^*}{\alpha_k^*} (\delta_k - e_{k-1,k})^2. \tag{29.13}$$

Generally speaking this result gives a modified ellipsoid to which $\mathbf{a}_k^*$ must belong at time $k-1$ if the new ellipsoid is to "capture" it properly in the next step.

The theoretical work of Rao and others does not translate immediately into a practical algorithm, but it suggests several practical ideas. First note that the sequence of bounds $\{\delta_t\}$ can apparently be increased beyond their true minimum values in order to improve the tracking ability of the algorithm. This is apparent from Eq. (29.13). More generally, this work suggests the intuitive notion that increasing the size of the ellipsoid at time $k-1$ prior to analyzing the incoming data set provides some insurance against loss of tracking. The basis for this inflation is to contain the shifting true parameters. At the same time increasing some measure of size of the ellipsoid makes it more likely that incoming data will be selected. The principle of ellipsoid inflation to render explicit and controllable adaptation is discussed in the papers cited above.

There is a limit to the allowable shift in parameters before tracking is lost. This is true regardless of whether the ellipsoid is artificially inflated prior to examining the incoming data, as theoretical results show.(23,24) Accordingly, it is important that a "rescue procedure" prevent disintegration of the OBE algorithm in time-varying environments. The $\{\kappa_t\}$ sequence is a good measure with which to monitor proper operation. An example algorithm employing such a technique is discussed below. Similar ideas are employed in more recent papers.(23,24)

Exponential forgetting: A general OBE algorithm can be designed to be explicitly adaptive within the established framework, by judicious choice of weighting sequence $\{\alpha_t\}$. One choice is to let the weighting effect a conventional forgetting factor,

$$\alpha_t = \alpha, \quad 0 < \alpha < 1 \tag{29.14}$$

for all t. This method has not been found to be effective for adaptation unless the system dynamics are changing slowly.(25) Further, it is not computationally efficient so that it is not a good candidate for real-time processing. Since the focus is on speech processing, this mode of adaptation is not pursued further.

Forgetting by back-rotation: The adaptive OBE algorithms which have been successfully applied to speech do not depend on a fixed scaling factor to expand the ellipsoid volume. These algorithms expand the ellipsoid by (selectively) removing previously accepted influential data sets from the system, either partially or completely. These data relinquish their influence on the ellipsoid to allow it to expand and adapt to the changes in the signal dynamics.

Having obtained an estimate $\mathbf{a}_{k-1}$ with associated covariance matrix $\mathbf{C}_{k-1}$, consider the data set at time k. Before doing so, however, adjust the existing system of equations in order to "downweight" the influence of some or all of the previous data sets. The existing data sets are modified by effectively introducing different minimization weights. Although the recursions are used to downweight the system, the process should be imagined to be "frozen in time" between $k-1$ and k while this downweighting takes place.

The sequences of weights $\{\alpha_t^*\}$ and $\{\beta_t^*\}$ for $t = 1, \ldots, k-1$, imply some set of error minimization weights, say $w_{k-1,1}, w_{k-1,2}, \ldots, w_{k-1,k-1}$, extant in the system at the time $k-1$ (see Eq. (29.3)). Before proceeding to the optimization problem at time k, change (in general, all) weights used at time $k-1$ to a new set, say $w_{k-1,t}^d$, $t = 1, \ldots, k-1$, where the superscript "d" denotes "downdating." The downdated weights are of the form

$$w_{k-1,t}^d = [1 - \varphi_{k-1,t}]w_{k-1,t}, \quad t = 1, 2, \ldots, k-1, \tag{29.15}$$

where $0 \le \varphi_{k-1,t} \le 1 \ \forall k,t$. In effect, this process removes a fraction $\varphi_{k-1,t}$ of the contribution of the data set at time t from the estimate. Not surprisingly, this can be accomplished by treating $(s_t, \mathbf{s}_t)$ as a new data set with "weight" $-\varphi_{k-1,t}w_{k-1,t}$ and incorporating it into the system of equations with the usual recursion.(13,27,28)

The method by which a data set is completely removed (downweighted with factor $\varphi_{k-1,t} = 1$) from the existing system using QR-WRLS has been called back rotation in recent papers. Use this term to refer to either partial or complete removal through QR-WRLS. Now formalize this procedure.

Sequentially modify weights as described above, beginning at time $t = n + 1$. The following (and similar) quantities pertain to the downdated system of equations whose weights have been modified to time t: $\mathbf{C}_{k-1,t}^d$, $\mathbf{T}_{k-1,t}^d$, $\mathbf{d}_{1,k-1,t}^d$, $\mathbf{a}_{k-1,t}^d$, $\kappa_{k-1,t}^d$, where each is similar to familiar quantities in the foregoing discussions. Henceforth omit the second subscripted argument if it is $k-1$. For example, $\mathbf{C}_{k-1,k-1}^d \overset{\text{def}}{=} \mathbf{C}_{k-1}^d$. Following the modification of the tth data set, the downdated equation to be solved (if the solution were desired) is

$$\mathbf{T}^d_{k-1,t}\,\mathbf{a}^d_{k-1,t}=\mathbf{d}^d_{k-1,t}\,. \tag{29.16}$$

The downdated ellipsoid matrix is $\mathbf{C}^d_{k-1,t}\,/\kappa^d_{k-1,t}$, where

$$\mathbf{C}^d_{k-1,t}=[\mathbf{T}^d_{k-1,t}]^T\,\mathbf{T}^d_{k-1,t}\,, \tag{29.17}$$

$$\kappa^d_{k-1,t}=\|\mathbf{d}^d_{1,k-1,t}\|+\tilde{\kappa}^d_{k-1,t}\,, \tag{29.18}$$

with

$$\tilde{\kappa}^d_{k-1,t}\overset{\text{def}}{=}\tilde{\kappa}_{k-1,t-1}-\varphi_{k-1,t}w_{k-1,t}[\delta_t-s_t^2]. \tag{29.19}$$

The quantity $\tilde{\kappa}_{k-1,0}\overset{\text{def}}{=}\kappa_{k-1}$ represents the value of κ updated to include $(s_{k-1},\,\mathbf{s}_{k-1})$. Equations (29.18) and (29.19) follow immediately from the definition of $\kappa_t=\sigma_t^2$ found in Chapter 4 and a basic understanding of the back-rotation process.(13) Following all necessary downdating just prior to time k, the algorithm uses the downdated system to compute the quantity

$$G^d_k\overset{\text{def}}{=}\mathbf{s}^T_k[\mathbf{C}^d_{k-1}]^{-1}\mathbf{s}_k. \tag{29.20}$$

In turn, these downdated numbers are used in place of their "non-downdated" counterparts to check for the existence of, and to compute, the optimal weights for time k. Once the optimization problem is complete, define

$$w_{k,t}=\begin{cases}\alpha^*_k w^d_{k-1,t}, & t=1,2,\ldots,k-1\\ \beta^*_k, & t=k\end{cases} \tag{29.21}$$

for the next iteration.(15)

This process appears to be extraordinarily computationally expensive in general since each past weight is modified at each $k-1$. However, "most" data sets are not included in the estimate ($\beta^*_k=0$) and, therefore, the system need not be downdated at these times. If the data set at time t, for example, is not included in the estimate, then formally $\mathbf{C}^d_{k-1,t+1}=\mathbf{C}_{k-1,t}$, $\mathbf{T}^d_{k-1,t+1}=\mathbf{T}_{k-1,t}$, and so forth, and no computation is required. A similar situation results if a data set, say at time t, is, at some previous time, completely removed by back-rotation so that $w_{k-1,t}=0$. In this case, no computational effort is required to downdate this data set at time $k-1$. Further, in many cases the modification of a particular data set is not desired. If, for example, the data set at t is not to be altered, then $\varphi_{k-1,t}=0$, and no computation is necessary. Finally, note that when the "new" data set at k is rejected ($\beta^*_k=0$), then $\mathbf{T}_k=\mathbf{T}^d_{k-1}$ and $\mathbf{a}_k=\mathbf{a}^d_{k-1}$, and, once again, no computation is actually required.

A wide range of computationally inexpensive adaptation strategies is inherent in the formulation above. For example, L is a constant window length. For all k,

$$\varphi_{k-1,t} = \begin{cases} 1, & t = k - L \\ 0, & \text{other } t \end{cases}. \tag{29.22}$$

The sequence $\{\varphi_{k-1,t}\}$ can, of course, be modified to effect a number of different window shapes. In a generalized version of this strategy some past group of data sets $\mathcal{T}_{k-1}$ is to be "forgotten," and,

$$\varphi_{k-1,t} = \begin{cases} 1, & t \in \mathcal{T}_{k-1} \\ 0, & t \notin \mathcal{T}_{k-1} \end{cases}. \tag{29.23}$$

The first case above corresponds to the use of a sliding window of length L, outside of which all previous data sets are completely removed. Norton and Mo have called this case fixed memory bounding[26] while Deller and Odeh call it simply windowing and suggest an efficient algorithm for implementing it.[25,27,28] The estimate at time k covers the range $t \in [k-L+1, k]$. The windowing technique is made possible by the ability to completely and systematically remove data sets at the trailing edge of the window. Only one back-rotation is required prior to optimizing at time k, and this is only necessary if $w_{k-1,l-L} = \beta^*_{k-L} \neq 0$. Case 2 is a different type of strategy which Deller and Odeh call selective forgetting. This technique selectively removes data sets from the system based on certain user-defined criteria. The selection process can be, for example, to remove (or downweight) only the previously heavily weighted data sets, to remove the data sets that are accepted in regions of abrupt change in the signal dynamics, or to remove the data sets starting from the first data set and proceeding sequentially. Whatever the criteria, a fundamental issue is to detect when adaptation is needed to improve the parameter estimates. This issue is further investigated in the speech processing studies below.

29.3.3. Suboptimal Testing for Innovation in the Data for $O(n)$ Time Processing

29.3.3.1. Complexity of the Basic OBE Algorithm

The experiments below illustrate the excellent spectral estimation, and tracking and adaptation capabilities of the OBE algorithms. From a real-time signal processing and identification point of view, an OBE algorithm has an inherent ability to select only data points which are informative in the sense of refining the feasible set. The fact that typically 70%–95% of the data are rejected by this criterion potentially implies a remarkable savings in computation. Note, however, that this is only true to the extent that the bounded-error preprocessing of an incoming data set is negligibly expensive compared with the inclusion of it in the estimate. This section examines some factors related to this complexity issue and shows how to exploit the selective updating for computational speed gain. Consideration is restricted to the cases in which the ellipsoid volume is minimized, but similar developments pertain to the trace criterion.

The general OBE algorithm can be implemented in a number of different ways. In particular, one can use either MIL-WRLS or QR-WRLS for the basic recursions. It is also possible to add *ad hoc* strategies for adaptation, as discussed above. A careful breakdown of the computational complexity of the OBE algorithm by the tasks of data checking and adaptation by back-rotation is found in Ref. 13. Generally, the average operation count for an adaptive OBE algorithm implemented on a sequential machine is approximated by

$$f_{opt} \sim O(c_1 n^2) + sO(n^2/2) + bO(c_2 n^2) + \rho O(c_3 n^2) \text{ flops per } t \text{ (sample)}, \quad (29.24)$$

where s is unity if the algorithm involves a forgetting sequence $\{\alpha_t\}$ and is zero otherwise; ρ is the average number of data sets accepted per t; b is the average number of back-rotations performed per t; and c_1, c_2 and c_3 are small numbers (all in the range 0.5–2.5) which depend upon whether MIL-WRLS or QR-WRLS is used. The first term is due to the check for information in the incoming data set. The others are attributable to covariance scaling (forgetting), adaptation, and solution update, respectively. The subscript "*opt*" is used to indicate that the "proper" optimization described in Chapter 4 is employed. Apparently, the OBE algorithm, as presently formulated, is an "$O(n^2)$" process. The objective is to reduce the effective complexity to $O(n)$ by reducing the checking cost. This renders the OBE algorithm a desirable alternative to standard RLS-based methods used in many signal processing problems, and in particular, batch LPC methods used in speech analysis. Also, a parallel processing approach achieves the $O(n)$ goal.

Before detailing the methods, some points about the use of the approximation "$O(n^2)$" are necessary. The first concerns a practical matter. The objective is to reduce the computational complexity of the algorithms to an average of $O(n)$ flops per t. Without data buffering, the data flow is still limited by the worst-case $O(n^2)$ computation. However, if a buffer is included, the algorithm may easily be structured to operate in $O(n)$ average time per t. Further, by using interrupt-driven processing of the checking procedure, it may be possible to reduce the average time even further.

Other preliminary points concern algorithmic details. We see from Eq. (29.24), the use of $\alpha_t = 1$ for all t is apparently required in order to avoid an invariant $O(n^2/2)$ flops per t. However, if it is important to include a non-unity $\{\alpha_t\}$, the extra computation can theoretically be avoided by noting the following. A theoretically identical result can be obtained by replacing the sequences $\{\alpha_t^*\}$ and $\{\beta_t^*\}$ by, say, $\{\overline{\alpha}_t^* = 1\}$ and $\{\overline{\beta}_t^* = \beta_t^*/\alpha_t^*\}$.[13] It is possible, therefore, to avoid the extra $O(n^2/2)$ computation by combining the forgetting sequence with the data-weighting sequence in this manner. Henceforth, ignore the $O(n^2/2)$ term, and beware the consequences of including the scaling sequence directly. Accordingly, rewrite Eq. (29.24) as

$$f_{opt} \sim O(c_1 n^2) + bO(c_2 n^2) + \rho O(c_3 n^2) \text{ flops per } t. \tag{29.25}$$

Also note that the $O(n)$ checking procedure to be developed does not depend on the weighting sequence used. Secondly, even if the checking procedure can be made $O(n)$, terms $bO(n^2)$ and $\rho O(n^2)$ (typically $b \approx \rho$) persist in Eq. (29.25). Therefore, to truly achieve $O(n)$ complexity, b and ρ must be $O(1/n)$. For large n, this is not the case. In fact, some experimental evidence suggests, not unexpectedly, that ρ increases with increasing n. For "large" n (conservatively, say, $n > 10$), therefore, the complexity is reduced to $O(\rho n^2)$ by $O(n)$ checking. Neither $O(n)$ nor $O(\rho n^2)$ complexity can be achieved, however, if the checking procedure remains $O(n^2)$. Therefore pursue an $O(n)$ test for innovation.

With an OBE algorithm, the number of computations needed for each n depends on whether the corresponding data set is accepted for processing by the optimization criterion. OBE essentially reverts to WRLS when a data set is accepted. Since most of the time the data set is rejected, for significant complexity reduction, an OBE algorithm must require many fewer than $O(3n^2)$ flops for checking.

Note some of the details of the checking procedure. In principle, the information-checking procedure for the volume algorithms consists of forming a quadratic polynomial, then solving for a positive root. As Favier and Arruda state in Chapter 4, however, it is sufficient to check the zero-order coefficient of the polynomial in either case for negativity to find out if a positive root exists. When the test is successful, then the root-solving and updating proceeds requiring the standard WRLS load, plus a few operations for finding the optimal weight. The most expensive aspect of this information test is the computation of the quantity G_k^d. (For generality, assume downdating is used. If this is not the case, it is merely necessary to drop the subscripts "d" on all quantities.) In the QR-WRLS case, a problem arises because G_k^d depends upon the inverse covariance matrix, $[\mathbf{C}_{k-1}^d]^{-1}$, which is not otherwise used in the process. The following method is suggested to sidestep this problem.[(29)] Recall the definition of G_k^d and note Eq. (29.17), and write

$$G_k^d = \mathbf{s}_k^T[\mathbf{T}_{k-1}^d]^{-1}[\mathbf{T}_{k-1}^d]^{-T}\mathbf{s}_k \overset{\text{def}}{=} [\mathbf{g}_k^d]^T\mathbf{g}_k^d = \|\mathbf{g}_k^d\|^2. \tag{29.26}$$

Since $\mathbf{s}_k = [\mathbf{T}_{k-1}^d]^T\mathbf{g}_k^d$, and the matrix $[\mathbf{T}_{k-1}^d]^T$ is lower triangular, $\mathbf{g}_k^d$ is easily found from the available quantities at time k by forward substitution. The procedure can be repeated to compute the trace quantity if needed.[(13)] The total computational load for computing G_k^d is $O(n^2/2)$ if this modification is used, which is far less than the effort required to invert $\mathbf{C}_{k-1}^d$.

29.3.3.2. Suboptimal Tests for Innovation in the Data and Complexity Reduction

In spite of the simplifications suggested above, the computation of the quantity G_k^d remains of $O(n^2)$ complexity. Clearly, the trick is to try to avoid the computation of these numbers in the information-checking procedure. A simple suboptimal updating rule[13,28] is to include the data set at time k only if

$$|e_{k-1,k}| > \delta_k. \qquad (29.27)$$

This test is applicable to OBE algorithms with any weighting strategy, and may be used for either volume or trace minimization.[†] The rationale for this test is simple. The zero-order coefficients of the volume and trace polynomials are never positive if the test is met. In the volume case, for example, the suboptimal check tests whether the zero-order coefficient is negative even if the term $-\kappa_{k-1}^d G_k^d$ is neglected. This ignored term is always negative and becomes small as k increases if no forgetting is used. For a given set of preceding optimal weights, $\alpha_1^*, \ldots, \alpha_{k-1}^*$ and $\beta_1^*, \ldots, \beta_{k-1}^*$ the suboptimal test *never* fails to accept a data set which would have been accepted by the optimal test if the same previous weights were present.

Equation (29.27) requires only $O(n)$ flops, so that the revised operation count is

$$f_{subopt} \sim O(n) + b' O(c_3 n^2) + \rho' O(c_2 n^2) \text{ flops per } t, \qquad (29.28)$$

where b' is the average number of back-rotations per t under the suboptimal checking policy, and ρ' represents the fraction of the data sets included in the update.

In light of Eq. (29.28), briefly consider the computational loads imposed by the specific adaptation strategies described above. In each case, assume QR-WRLS underlies the process, but the discussion for MIL-WRLS is similar. Of the adaptation methods described above, exponential forgetting is the most expensive computationally. It requires $O(n^2/2)$ flops per t, unless the scalar is combined with the β_t weight as described in Eq. (29.25). In the latter case, the cost of the forgetting factor is negligible, requiring only one real flop per t. Since back-rotation is essentially equivalent to a covariance (or $\mathbf{T}_k$) update[‡] for an incoming data set, each of these rotations takes $O(2.5\, n^2)$ flops. If b back-rotations are performed on the average at each k, then, effectively, the adaptation requires $O\ (2.5bn^2)$ additional operations. Since ρ is usually small, whether a particular adaptation strategy is cost-effective depends on the number b. For windowing, for example, $b \approx \rho$ and the adaptation adds negligibly to the computational load. The cost of selective

[†] Inequality (27) is similar to the test used by Dasgupta and Huang in Ref. 30 to ascertain whether an optimal weight exists in the sense of minimizing κ_k^d. The implications of this similarity are discussed in detail in Ref. 14.

[‡] A *parameter solution* update is *not* required, just the covariance update.

forgetting depends entirely upon the criterion employed for deciding to back-rotate a previous data set, which, in turn, determines the value of b. An example is discussed below.

29.3.3.3. Parallel Hardware Implementations

One of the advantages of the QR-WRLS-based OBE formulation is that it immediately admits a solution by contemporary parallel architectures. This is critical because it reduces the complexity of the optimal algorithm from $O(n^2)$ to $O(n)$. The significant reduction of computational complexity and parallel-hardware implementation of OBE algorithms improve their potential for real-time applications. Odeh and Deller have developed systolic architectures for both nonadaptive(31) and adaptive(25,27) versions of the SM-WRLS algorithm. The complexity of the parallel computation is

$$f_{\substack{opt \\ parallel}} \sim O(3n) + \rho O(11n) \;\text{ flops per } t \tag{29.29}$$

if the optimal checking is implemented, where ρ, as above, is the fraction of the data accepted by the optimization. If suboptimal checking is employed, the average count is

$$f_{\substack{subopt \\ parallel}} \sim O(n) + \rho' O(11n) \;\text{ flops per } t, \tag{29.30}$$

where ρ' likewise indicates the acceptance ratio. When adaptation by back-rotation is added to either strategy, an additional $bO(11n)$ (or $b'O(11n)$) flops per t are required on the average, where b and b', as above, indicate the average number of back-rotations computed per t in the optimal and suboptimal cases. Note that these tallies represent parallel complexities in the sense that they denote the effective number of operations per t, though many processors can be performing this number of operations simultaneously. Accordingly, the parallel complexity indicates the time it takes the parallel architecture to process the data regardless of the total number of operations performed by the individual cells.

Distinct $\{\alpha_t\}$ and $\{\beta_t\}$ sequences can be used in the parallel versions of the algorithms at virtually no computational cost, but at the negligible hardware cost of n multiplication units.

29.4. ADAPTIVE ANALYSIS OF SPEECH SIGNALS

29.4.1. Introduction

This section retreats from the direct use of real speech sequences in order to avoid the nuances of $\{\delta_t\}$ estimation. For simplicity, two "true" AR(2) models whose *time-varying* coefficients are derived using LPC analysis of order two on utterances of the words “four” and “six” by an adult male speaker. While more realistic analysis of speech would involve model orders of 10–14 (e.g., Ref. 1, Ch. 5), this small number of parameters is used here so that the results are easily illustrated. The original speech waveforms are shown in Fig. 29.4.

The “true” models are of the form

$$s_t = a^*_{1,t}s_{t-1} + a^*_{2,t}s_{t-2} + \omega_t. \tag{29.31}$$

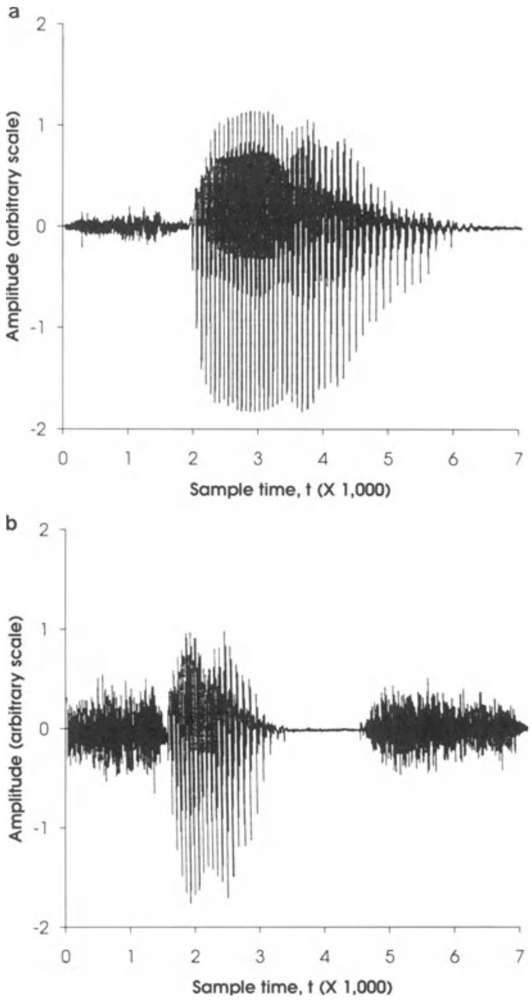


FIGURE 29.4. Digitized speech waveforms used in the adaptive identification experiments: (a) “Four” spoken by an adult male speaker. (b) “Six” spoken by an adult male speaker.

To derive the two sets of "true" parameters, the original speech data are sampled at 10 kHz after 4.7 kHz lowpass filtering, and the algorithm described in Ref. 22 is employed with a forgetting factor 0.996 for adaptation. A 7000-point sequence, $\{s_t\}$, for each case ("four" and "six"), is generated by driving the appropriate set of parameters with an uncorrelated sequence $\{\omega_t\}$ which is uniformly distributed on [–1,1]. Adaptive OBE algorithms with optimal and suboptimal data checking are used to estimate the $\{a^*_{i,t}\}$ parameters. Again, SM-WRLS is the representative OBE method. The estimates are denoted $\{a_{i,t}\}$. Several simulation results are presented. To conserve space, only the result for $a_{1,t}$ is illustrated in each case. Each figure shows two curves, one for the true parameter, the other for the estimate obtained by the algorithm under study.

29.4.2. RLS and OBE Algorithms in Adaptive Speech Processing

In general, the power of the SM-WRLS algorithm is evident when compared with the conventional RLS algorithm.[18] As a basis for further discussion, note Figs. 29.5 and 29.6. The simulation results for the word "four" using the RLS and the SM-WRLS algorithms, respectively, and Figs. 29.7 and 29.8 show the simulation results for the word "six." It is evident that SM-WRLS performs better than RLS in terms of its tracking capability. Critically, this improved performance comes with greatly improved computational efficiency. In this case SM-WRLS uses only 1.86% and 2.16% of the data for the words "four" and "six," respectively, yet yields better parameter estimates almost all the time. SM-WRLS tracks the time-varying parameters faster than RLS. This is manifest in both examples, especially the word "six" (see Figs. 29.7 and 29.8).

The "unmodified" SM-WRLS algorithm (and other OBE algorithms) apparently have adaptive capabilities in its own right. While SM-WRLS is developed

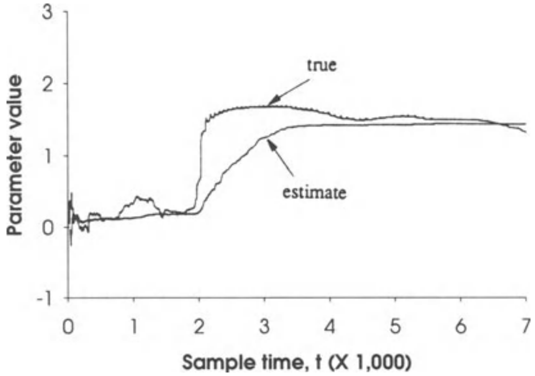


FIGURE 29.5. Result for the parameter $a_{1,t}$ using RLS analysis of the word "four."

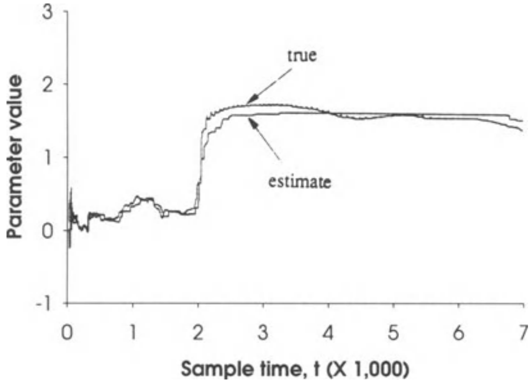


FIGURE 29.6. Result for the parameter $a_{1,t}$ using SM-WRLS analysis of the word "four:" 1.86% of the data is used in the estimation.

under the assumption of stationary system dynamics, it is capable of behaving in this manner because of the special weights used. However, it is not possible to depend upon an OBE algorithm to reliably behave in this adaptive manner, particularly in cases of quickly-varying system dynamics. Each time a new data set is accepted, the ellipsoid volume decreases and the "confidence" in the current estimate increases. In a situation in which the signal is varying rapidly and the parameters are moving away from their current locations, the algorithm accepts incoming data sets to incorporate the new information into the estimate. The ellipsoid volume decreases rapidly, eventually becoming very small. As the parameters continue to move rapidly away from their current locations, they eventually

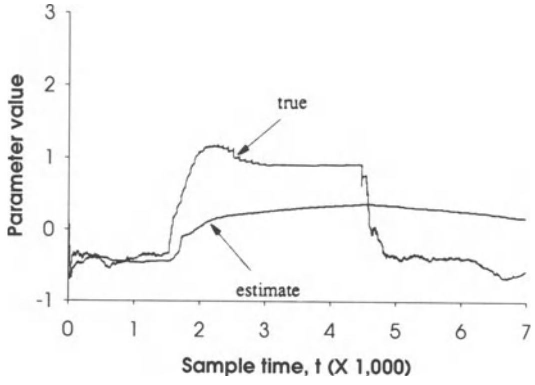


FIGURE 29.7. Result for the parameter $a_{1,t}$ using RLS analysis of the word "six."

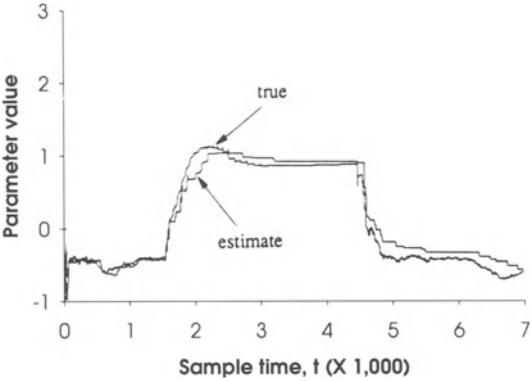


FIGURE 29.8. Result for the parameter $a_{1,t}$ using SM-WRLS analysis of the word "six:" 2.16% of the data is used in the estimation.

move outside the shrinking ellipsoid which becomes an invalid bounding ellipsoid. This condition indicates that a violation of the theory has taken place, and, therefore, the unmodified SM-WRLS algorithm is no longer guaranteed to work properly.

Next, the simulation results of the several variations on the general SM-WRLS algorithm are shown.

29.4.3. Adaptive Algorithms

29.4.3.1. Windowing

Figs. 29.9 and 29.10 show the simulation results of the windowed SM-WRLS algorithm for the words "four" and "six," respectively, using a window of length

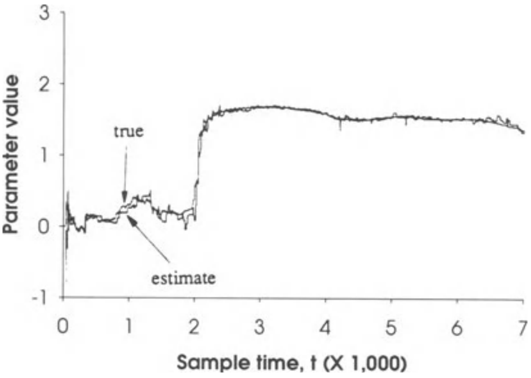


FIGURE 29.9. Result of the windowed SM-WRLS algorithm with L = 1000 for the word "four:" 5.69% of the data is used in the estimation.

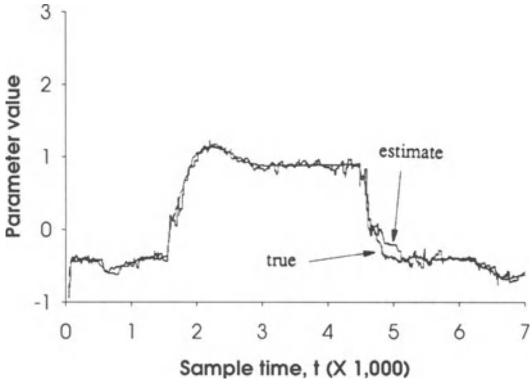


FIGURE 29.10. Result of the windowed SM-WRLS algorithm with L = 1000 for the word "six:" 5.44% of the data is used in the estimation.

1000. This strategy uses only 5.69% and 5.44% of the data for the words "four" and "six," respectively. *Effectively*, however, it uses about twice this many if back-rotation computations are accounted. More data than those with the unmodified SM-WRLS algorithm are used, but more accurate estimates result and the time-varying parameters are tracked more quickly and accurately. This can easily be seen when the parameter dynamics change abruptly near the point 2100 in the word "four" (Fig. 29.9) and near the points 2000 and 4500 in the word "six" (Fig. 29.10).

As an example variation on the windowing procedure, let $\{\varphi_{k-1,t}\}$ of Eq. (29.15) taper linearly from unity at time $t = k - L$ to zero at time $t = k$. This effects a smoother window which gradually forgets the past data by rotating out 0.1% of each of the data sets accepted in the past 1000 recursions. Figures 29.11 and 29.12 show the results. This strategy uses only 6.19% and 4.89% of the data for the words "four" and "six," respectively. Note that this technique uses comparable percentages of the data to those of the windowed strategy and yields smoother estimates. Although the algorithm uses very small percentages of the data, the fraction $\varphi_{k-1,k-L+1} = 0.001$ may not be practical. It means that the algorithm will rotate out each data set that is initially accepted 1000 times, clearly a computational burden. Depending on the nature of the problem, practical values of $\varphi_{k-1,k-L+1}$ may range from 0.002 to 0.01 with an effective window of length 500 to 1000.

29.4.3.2. Selective Forgetting

Selective forgetting chooses data sets to be (partially) removed from the system based on user-defined criteria. The selection criterion is as follows: remove the accepted data sets included at times $k - 1, k - 2, \ldots$, at time k at which it is

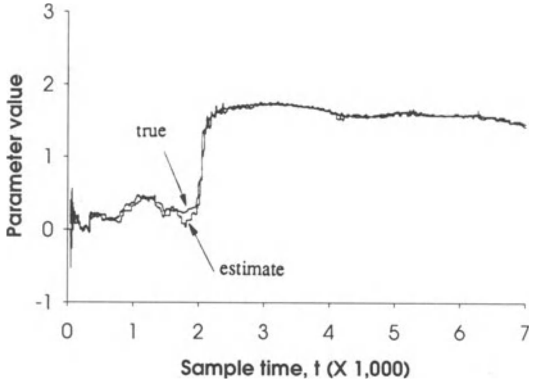


FIGURE 29.11. Result of processing the word "four" using the windowed SM-WRLS algorithm with $L = 1000$ and linear tapering of the window: 6.19% of the data is used in the estimation.

desired to "forget" some of the past; proceed sequentially until some other condition is satisfied. The determination of when to apply the forgetting procedure and when to stop removing data sets is discussed below.

When inspecting the true parameters of the word "four," for example, note that they can be characterized as having slow time variations everywhere except in the region from $t = 2000$ to 2300 where they have fast time variations. The fact that the parameters are changing very slowly in the first 2000 points induces the algorithm to accept some points which, in turn, cause the ellipsoid volume to decrease. Near time $t = 2000$, the ellipsoid volume becomes very small. When the parameters move rapidly away from their current location, they eventually move outside the ellipsoid.

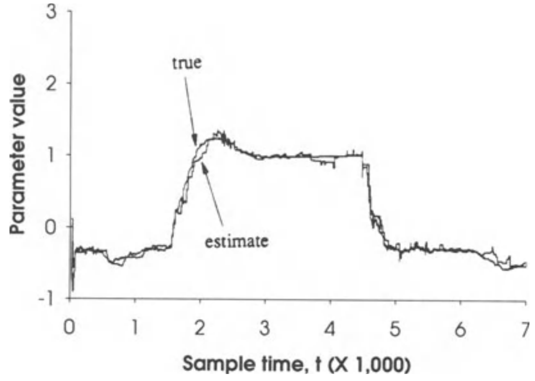
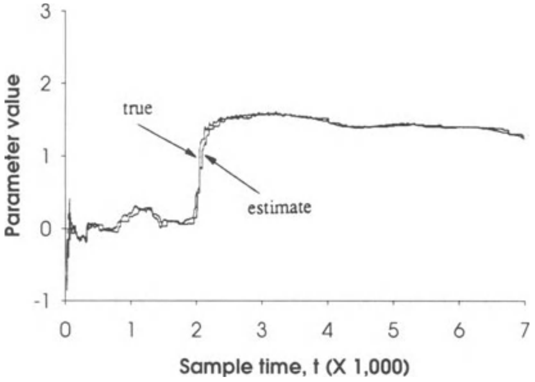


FIGURE 29.12. Result of processing the word "six" using the windowed SM-WRLS algorithm with $L = 1000$ and linear tapering of the window: 4.89% of the data is used in the estimation.



FIGURE' 29.13. Result of processing the word "four" using the selective forgetting SM-WRLS algorithm: 3.6% of the data is employed in the estimation process. Effectively, 4.27% of the data is used when the back-rotations are taken into account.

This condition leads to a negative value of κ_t, a violation of the theory (in particular, the violation of the assumption of stationary dynamics).* A negative κ_k is an effective indicator of need for adaptation at time $t = k$. Use this criterion as the prompt to begin selective forgetting. The algorithm starts rotating out the incorporated past data sets beginning at time $t = k - 1$ until κ_k becomes positive again.

Figs. 29.13 and 29.14 show the simulation results of the selective forgetting strategy described here. When counting the total number of data sets rotated into and out of the system, this strategy effectively uses only 4.27% and 4.41% of the data for the words "four" and "six," respectively. Compared to the windowed and "gradually windowed" adaptive strategies discussed above, the simulation results show that the selective forgetting strategy yields smoother estimates using fewer data.[13]

Carefully note that $\kappa_k > 0$ is only a *necessary* condition for the true parameters to be inside the ellipsoid at time k. The fact that κ goes negative at a particular time does not precisely determine the point at which system dynamics began to change. In fact, $\kappa_k < 0$ indicates a severe breakdown of the process; the "true" parameters have moved well outside of the current ellipsoid. However, it is precisely in cases of *fast*-changing dynamics that this "breakdown" occurs to rapidly result in "$\kappa_k < 0$" being a good locator of changing dynamics which require "immediate" adaptation to preserve the integrity of the process. Figs. 29.6 and 29.8 illustrate cases of slowly-changing dynamics where the theory can be violated without the appearance of negative κ. OBE algorithms seem sufficiently robust to make their own

*$\kappa_k < 0$ indicates an ellipsoid of negative dimensions at time $t = k$.

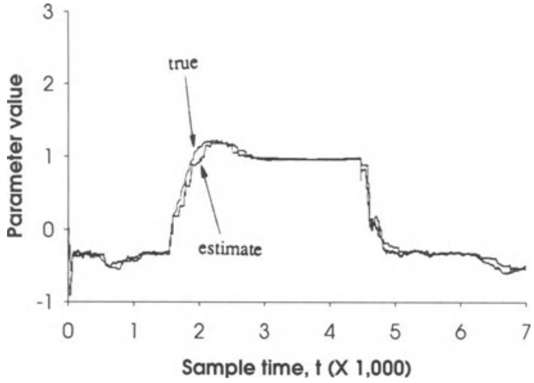


FIGURE 29.14. Result of processing the word "six" using the selective forgetting SM-WRLS algorithm: 2.83% of the data is employed in the estimation process. Effectively, 4.41% of the data is used when the back-rotations are taken into account.

adjustments in such cases. Strict theoretical criteria for the sequential containment of the true parameters in the presence of time-variance are given in papers by Rao and Huang[23] and Huang and Deller.[24]

29.4.4. *Suboptimal Checking*

Figs. 29.15 and 29.16 show the simulation results of the "nonadaptive" SM-WRLS algorithm with suboptimal data selection. In this case, only 1.19% and 1.53% of the data are used or the words "four" and "six," respectively. Compared to the SM-WRLS algorithm (Figs. 29.6 and 29.8), the suboptimal technique uses

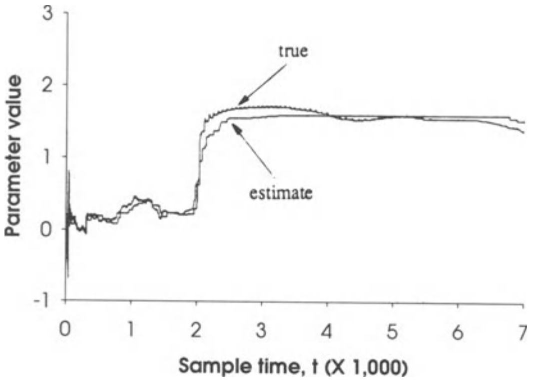


FIGURE 29.15. Result of processing the word "four" using the SM-WRLS algorithm with no adaptation and suboptimal checking: 1.19% of the data is employed in the estimation process.

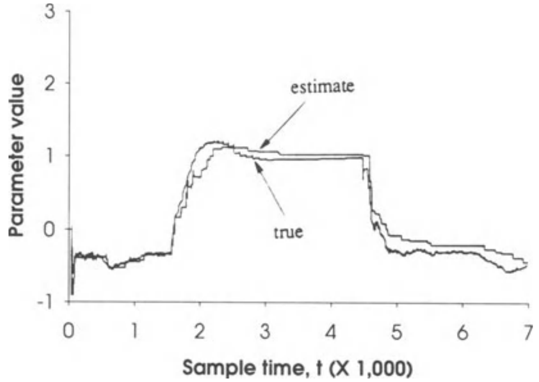


FIGURE 29.16. Result of processing the word "six" using the SM-WRLS algorithm with no adaptation and suboptimal checking: 1.53% of the data is employed in the estimation process.

slightly fewer data but produces comparable estimates. Note that *most* of the data sets (97.6% for the word "four" and 94.4% for the word "six") that are accepted by the suboptimal technique are also accepted by the SM-WRLS algorithm.

29.4.5. Adaptive Algorithm with Suboptimal Checking

The simulation results of the selective forgetting SM-WRLS technique with suboptimal data selection are shown in Figs. 29.17 and 29.18. This strategy effectively uses only 2.16% and 2.76% of the data for the words "four" and "six," respectively.

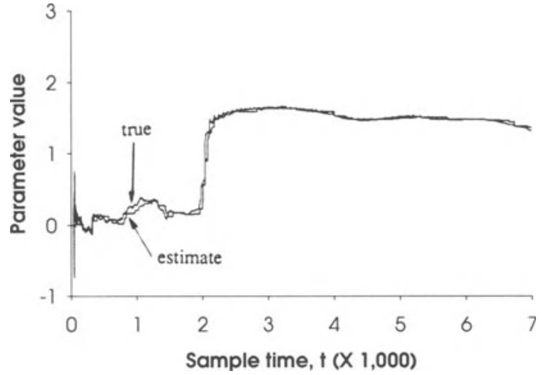


FIGURE 29.17. Result of processing the word "four" using the SM-WRLS algorithm with selective forgetting and suboptimal checking: 1.89% of the data is employed in the estimation process; effectively, 2.16% if the back-rotations are included.

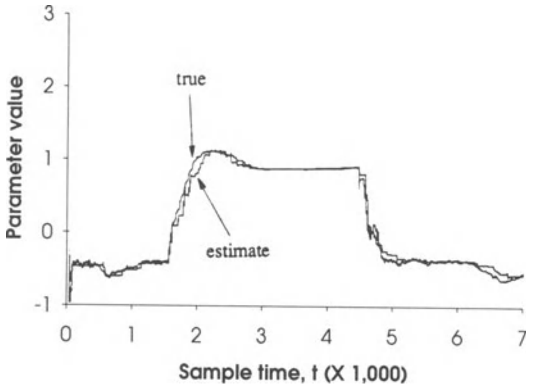


FIGURE 29.18. Result of processing the word "six" using the SM-WRLS algorithm with no adaptation and suboptimal checking: 1.53% of the data is employed in the estimation process; effectively, 2.76% if the back-rotations are included.

Compared to optimal selective forgetting (Figs. 29.13 and 29.14), selective forgetting with suboptimal selection uses fewer data but produces comparable estimates. On the other hand, compared to unmodified SM-WRLS with suboptimal data selection (Figs. 29.15 and 29.16), suboptimal selective forgetting uses more data but produces better estimates.

29.5. CONCLUSIONS

OBE algorithms have strong potential for improving spectral accuracy, tracking ability, and computational load of LPC analysis of speech signals. In turn, LPC identification is at the heart of many important problems in speech compression and coding, recognition and synthesis, as well as in speaker identification and verification. Autoregressive modeling is also central to many other important signal processing applications, for example, image processing and geophysical modeling. The results presented here are immediately applicable. Furthermore, the results are very easily generalized to cover any linear-in-parameters regressor model, including those with complex and/or vector signals. Therefore, although the focus is on the speech-processing application here, the material is of much broader utility.

Speech processing and other real-time signal processing tasks may benefit from the set estimate provided by the OBE method. However, the main focus here has been on the potential to discard uninformative data for speed and accuracy improvement, and for greatly enhanced tracking capability. In this endeavor, the technique has proven very effective in experimental studies. The fundamental issue of finding proper error bounds in real signal applications, however, is not trivial. The significant benefits observed are critically dependent upon these bounds. This issue remains an open area for further theoretical and experimental research. Likewise, the theory of OBE processing does not strictly support the identification of time-varying models, and further theoretical research in explaining the performance of adaptive methods significantly benefits real applications.

Finally, although this work focuses on a simple OBE algorithm, SM-WRLS, certain important results are theoretically guaranteed to be similar with any reasonable weighting strategy. These results include the data selected, and the pointwise ellipsoids and their central estimators. This is a consequence of the unified theory presented and in Chapter 4 and in Refs. 12, 13, and 14. Other OBE algorithms might emerge as advantageous in certain applications, due to practical considerations such as roundoff errors. This issue is also open to further pursuit as these powerful techniques are increasingly applied to real problems.

REFERENCES

1. J. R. Deller, Jr., J. G. Proakis, and J. H. L. Hansen, *Discrete-Time Processing of Speech Signals*, Macmillan, New York (1993).
2. K. Steiglitz and B. Dickinson, *IEEE Trans. Acoust., Speech, Signal Process.* **25**, 34 (1977).
3. M. G. Berouti, D. G. Childers, and A. Paige, in: *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing 1*, Hartford, CT, pp. 33–36 (1977).
4. D. Y. Wong, J. D. Markel, and A. H. Gray, *IEEE Trans. Acoust., Speech, Signal Process.* **27**, 350 (1979).
5. J. R. Deller, Jr., *IEEE Trans. Acoust., Speech, Signal Process.* **29**, 917 (1981).
6. J. N. Larar, Y. A. Alsaka, and D. G. Childers, in: *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing 2*, Tampa, FL, pp. 1089–1092 (1985).
7. A. K. Krishnamurthy, in: *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing 3*, San Diego, CA, pp. 36.3.1–36.3.4 (1984)
8. D. E. Veeneman and S. L. BeMent, *IEEE Trans. Acoust., Speech, Signal Process.* **33**, 369 (1985).
9. A. K. Krishnamurthy and D. G. Childers, *IEEE Trans. Acoust., Speech, Signal Process.* **34**, 730 (1985).
10. Y. Miyoshi, K. Yamamoto, R. Mizoguchi, Y. Masuzo, and O. Kakusho, *IEEE Trans. Acoust., Speech, Signal Process.* **35**, 1233 (1987).
11. G. P. Pichaché, *A Givens Rotation Algorithm for Single Channel Format Tracking and Glottal Waveform Deconvolution*, M.S. Dissertation, Northeastern University, Boston, MA (1988).
12. L. V. R. Arruda and G. Favier, in: *Proceedings of the 9th IFAC/IFORS Symposium on Identification and System Parameter Estimation 2*, Budapest, Hungary, pp. 1027–1032 (1991).
13. J. R. Deller, Jr., M. Nayeri, and S. F. Odeh, *Proc. IEEE* **81**, 813 (1993).
14. J. R. Deller, Jr., M. Nayeri, and M. S. Liu, *Int. J. Autom. Control Signal Process.* **8**, 43 (1994).
15. J. R. Deller, Jr. and T. C. Luk, *Comput. Speech Lang.* **3**, 301 (1989).
16. R. Lozano-Leal and R. Ortega, *Automatica* **23**, 247 (1987).
17. S. M. Veres and J. P. Norton, *Int. J. Control* **50**, 639 (1989).
18. L. Ljung and T. Söderström, *Theory and Practice of Recursive Identification*, MIT Press, Cambridge, MA (1983).
19. G. H. Golub and C. F. van Loan, *Matrix Computations*, 2nd Ed., Johns-Hopkins Univ. Press, Baltimore, MD (1989).

20. W. M. Gentleman and H. T. Kung, in: *Proceedings of the Society of Photoptical Instrumentation Engineers: Real Time Signal Processing IV*, San Diego, CA, pp. 19–26 (1981).
21. J. G. McWhirter, in: *Proceedings of the Society of Photoptical Instrumentation Engineers: Real Time Signal Processing IV*, San Diego, CA, pp. 105–112 (1983).
22. J. R. Deller, Jr. and D. Hsu, *IEEE Trans. Circuits Systems* **34**, 782 (1987).
23. A. K. Rao and Y. F. Huang, *IEEE Trans. Signal Process.* **41**, 1140 (1993).
24. Y. F. Huang and J. R. Deller, Jr., in: *Proceedings of the 39th Annual Allerton Conference on Communications, Control, and Computing*, Monticello, IL, pp. 50–59 (1992).
25. S. F. Odeh, *Algorithms and Architectures for Adaptive Set Membership-based Signal Processing*, Ph.D. Dissertation, Michigan State University, East Lansing, MI (1990).
26. J. P. Norton and S. H. Mo, *Math. Comput. Simul.* **32**, 527 (1990).
27. S. F. Odeh and J. R. Deller, Jr., in: *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing* **5**, Albuquerque, NM, pp. 2419–2422 (1990).
28. J. R. Deller, Jr. and S. F. Odeh, *Proceedings of the 9th IFAC/IFORS Symposium on Identification and System Parameter Estimation* **2**, Budapest, Hungary, pp. 1044–1049 (1991).
29. J. R. Deller, Jr., *IEEE Trans. Acoust., Speech, Signal Process.* **37**, 1432 (1989).
30. S. Dasgupta and Y. F. Huang, *IEEE Trans. Inf. Theory* **33**, 383 (1987).
31. J. R. Deller, Jr., and S. F. Odeh, in: *IEEE International Conference on Acoustics, Speech, and Signal Processing 2*, Glasgow, Scotland, pp. 1067–1070 (1989).

30

Robust Performances Control Design for a High Accuracy Calibration Device

M. Milanese, G. Fiorio, and S. Malan

ABSTRACT

This chapter presents a case study of robust performances control design. The physical plant under examination consists of a platform for calibration of high accuracy accelerometers. It has to assume the properties of an inertial body, despite the vibrations coming from the surrounding ground. Plant modeling and parameter estimation, control system design and robustness analysis of the designed controllers are described and discussed. Besides a simplified model of the plant (the nominal model) perturbations are also considered to take into account parametric and dynamic uncertainties. The procedure followed for estimating model parameters, based on an unknown but bounded approach, is illustrated, and uncertainty intervals of parameter estimates are provided. Bounds of unstructured uncertainty are also derived from results of simulations to evaluate the main effects of the unmodeled dynamics.

The design has been carried on through iterative steps of "nominal" design and robustness analysis. The design has been performed through H_∞ synthesis, based on the nominal model and taking into account the main performance specifications

M. Milanese, G. Fiorio, and S. Malan • Dipartimento di Automatica e Informatica, Politecnico di Torino, 10129 Torino, Italy.

Bounding Approaches to System Identification, edited by M. Milanese *et al.* Plenum Press, New York, 1996.

required for the present case study, i.e. stability, disturbance attenuation and command power limitation. The robustness analysis has been performed using recent techniques able to deal with frequency domain specifications and with mixed non-linear parametric and dynamic perturbation, as required in the present case study.

30.1. INTRODUCTION

The problem originated from a national laboratory, in the realization of a calibration device for high accuracy acceleration transducers. This device requires to work over a platform, whose conditions should be close to those of an inertial body. Unfortunately, the laboratory is located in the neighborhoods of heavy mechanical factories, whose undesired effects is to generate vibrations in the surrounding ground.

In these conditions, it is required to reduce the r.m.s. value of the platform perturbing acceleration in the ratio 1:100 approximately, in order to have sufficiently negligible calibration errors with respect to the accuracy guaranteed for the most sensitive transducers.

Preliminary analysis showed that it is not convenient to use simply passive mechanical elements, such as springs and dampers, to solve the problem of reducing platform perturbing acceleration. Unfeasible parameter values should be required. On the contrary, the use of active elements, such as electromagnetic actuators driven in feedback or in mixed feedback-feedforward control schemes, leads to much more effective disturbance attenuation.

Fig. 30.1 gives a sketch of the plant. A concrete rectangular platform P is supported at each corner by a set of three elements lying on ground: a spring, a damper and an electromagnetic force generator. Another platform B, bearing the calibration device, leans on the first one through similar mechanical elements, but without active generators.

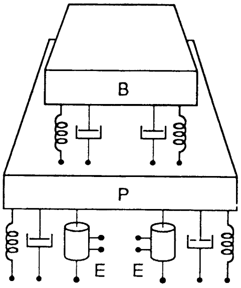


FIGURE 30.1. The plant.

Accuracy is the main concern of the calibration device. The desired accuracy must be guaranteed for the controlled system, despite of the limited and uncertain information available on the plant. Then, a robust (worst-case) approach is taken, based on the typical two steps:

- Identification of the approximate behavior of the system in terms of a *nominal* model and a perturbation model, able to capture the discrepancies between the nominal model and the actual system (described in Sections 30.2 and 30.3).
- Design of a control system which assures acceptable performances according to some given specifications, not only for the nominal model, but for all considered perturbations (described in Section 30.4).

It is well known that the type of perturbation model plays a key role. In the robust control literature, two main types of perturbations have been extensively studied: parametric (real) and dynamic (complex) perturbations. The former account for parametric variations, and the latter for unmodeled dynamics.(1,2)

Recently, some work has been done on mixed types of perturbation.(3,4) Mixed parametric-dynamic perturbations seem able to better capture the physical information on the approximations introduced by the model. Ref. 5 performs a preliminary analysis of this case study and takes into account parametric perturbation only. The results clearly show that in this case the unmodeled dynamics play a key role in robust stability. This chapter performs a more complete study, using a mixed perturbation model.

30.2. THE NOMINAL AND THE PERTURBATION MODELS

The aim of this section is to illustrate the results of model building for a suitable representation of the plant. First, the nominal model is described. Furthermore, perturbations on this model are introduced to take into account its main approximations.

The nominal model is based on the following simplifying hypotheses: 1) The ground, as well as platforms P and B in Fig. 30.1, are considered as rigid bodies; 2) The ground has only vertical motion, and the structure has perfect symmetry at the four corners of the platforms; and 3) Only linear equations are included in the model.

The four electromagnetic actuators at the corners of the lower platform of Fig. 30.1 are driven by the same electric current i, according to the hypothesis that perfect symmetry gives rise to translation motion of the platforms along the vertical axis only. The simplified model is depicted in Fig. 30.2. In this figure, the four identical elements at each corner of the platforms are represented by only one equivalent parameter.

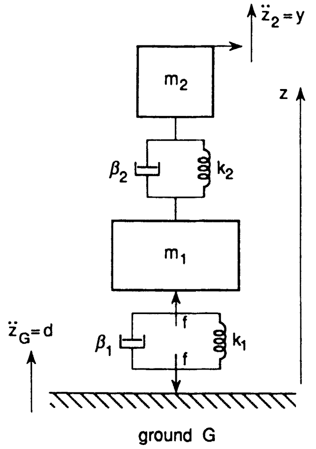


FIGURE 30.2. The simplified model.

In the nominal model, the vertical component of the ground acceleration $\ddot{z}_G$ is considered as the only disturbance, and is denoted by d. The force f in Fig. 30.2 is considered as proportional to the current i, which is the only command variable of the system $f = K_F i$. The vertical component $\ddot{z}_2$ of the upper platform acceleration is the controlled output, denoted by y. Thus, the system can be represented by the single input–single output (SISO) structure of Fig. 30.3, where $y(s)$, $i(s)$ and $d(s)$ represent system output, command variable and disturbance, respectively.

$M(s)$ and $A(s)$ represent output sensor and actuator transfer functions, respectively. The regulator transfer function to be designed is $C(s)$. $G(s)$ and $H(s)$ are the command to output and disturbance to output transfer functions, respectively. Their expressions are:

$$G(s) = N_G(s)D^{-1}(s) \tag{30.1}$$

$$H(s) = N_H(s)D^{-1}(s) \tag{30.2}$$

where

$$N_G(s) = K_F k_1^{-1} s^2 (1 + \beta_2 k_2^{-1} s) \tag{30.3}$$

$$N_H(s) = (1 + \beta_1 k_1^{-1} s)(1 + \beta_2 k_2^{-1} s) \tag{30.4}$$

$$\begin{aligned} D(s) = {} & m_1 m_2 k_1^{-1} k_2^{-1} s^4 + [m_1 \beta_2 + m_2(\beta_1 + \beta_2)] k_1^{-1} k_2^{-1} s^3 + \\ & + [m_1 k_1^{-1} + m_2(k_1^{-1} + k_2^{-1}) + \beta_1 \beta_2 k_1^{-1} k_2^{-1}] s^2 + \\ & + (\beta_1 k_1^{-1} + \beta_2 k_2^{-1}) s + 1 \end{aligned} \tag{30.5}$$

Output sensor is modeled by the transfer function

$$M(s) = K_m s (1 + s\tau)^{-1}. \tag{30.6}$$

Actuator transfer function $A(s)$ is assumed as a constant K_a in the frequency range of interest:

$$A(s) = K_a. \tag{30.7}$$

In order to take into account the approximations introduced by this simplified model, the three main assumptions are briefly discussed.

Consider the linearity assumption. The behavior of the plant in normal operating conditions can actually be considered linear with good approximation, due to the very small displacements and accelerations present in these conditions, and to the fact that input current is controlled to stay within the linearity range of the relation force-current.

On the contrary, while the rigid body assumption appears to be likely for platforms P and B, ground deformability gives parasitic effects which may cause stability problems in the closed loop. Then the deformability of the ground is taken into account by means of an uncertainty represented in term of a multiplicative perturbation Δ as indicated in Fig. 30.3, such that

$$\|W^{-1}(\omega)\Delta(\omega)\|_\infty \le 1, \quad 0 \le \omega \le \infty. \tag{30.8}$$

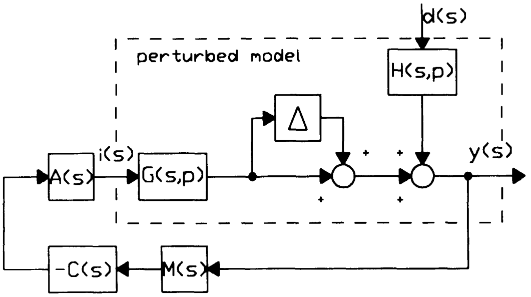


FIGURE 30.3. The control scheme.

The choice of the weighting function $W(\omega)$ is discussed in Section 30.3.

The nonperfect symmetry of the structure gives rise to rotating motion components of the platforms, with the effect of adding two pole-zero couples for each pole of the transfer functions of the nominal model. The pole and the zero of each added couple are very close, and close to the corresponding pole of the nominal model. Simulation has verified that large diasymmetries (up to 10% on each parameter) give transfer functions, which can be recovered with very good approximation by suitable assessments of the parameters of the simplified model of Fig. 30.2.

In fact, it is assumed that parameter vector p is known only to belong to a parameter uncertainty set Π. The identification procedure described in the next section provides parameter estimates with their ranges of variations able to recover this and other sources of parametric perturbations, for example, the ones due to errors in measurements.

30.3. IDENTIFICATION PROCEDURE

The model described in the preceding section contains several parameters whose values have to be known.

Constants K_m, τ, K_F and K_a are given the values from the data sheets of the corresponding components:

$$K_m = 2 \cdot 10^5\ Vs^3m^{-1},\ \tau = 2\ s,\ K_F = 8.7\ NA^{-1},\ K_a = 10.0\ AV^{-1}. \qquad (30.9)$$

The remaining parameters of the model are masses m_1 and m_2, stiffness coefficients k_1 and k_2, and damping coefficients β_1 and β_2.

It might be possible to disassemble the system and to measure these parameters separately. Apart from practical difficulties, this approach is not appropriate because the assumed model is a simplified model, and its parameters have the nature of equivalent parameters, with implicit reference to some physical phenomena neglected in the model, as discussed in the preceding section. For instance, stiffness parameter k_1 is an equivalent parameter to take into account stiffness of the springs sustaining platform P, the ground elasticity, and asymmetry of the structure. It can then be argued, from physical considerations, that parameter m_2 is less affected by neglected dynamics. Consequently, this parameter has been set to the value obtained by direct measurement:

$$m_2 = 440\ kg. \qquad (30.10)$$

The remaining parameters are identifiable from the given experimental conditions, and have been estimated from measurements on the overall system.

The available experimental data are:

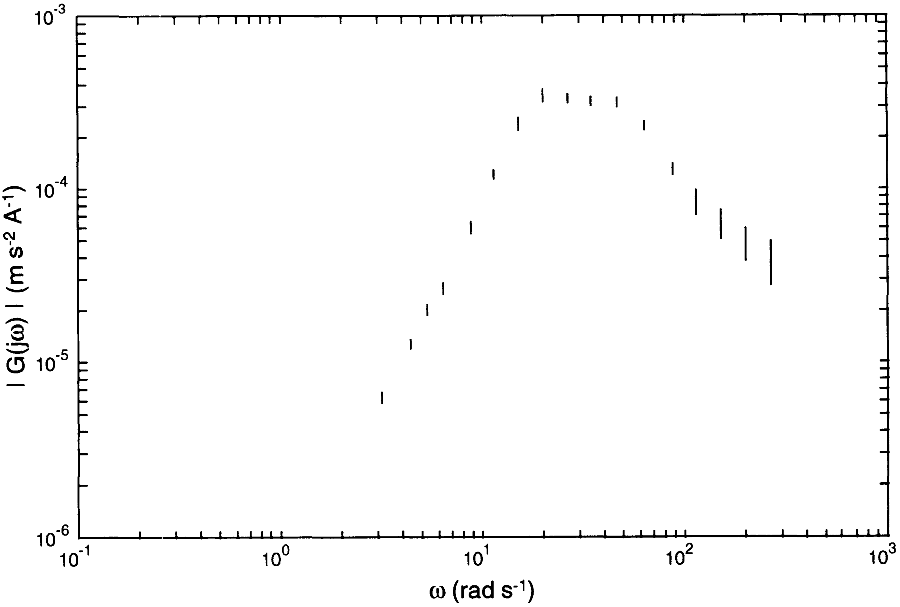


FIGURE 30.4. Samples of $|G(j\omega)|$ with corresponding uncertainty intervals.

- samples of frequency response command to output (magnitude and phase) $G(j\omega)$ with sinusoidal current generator supplying the actuator; input and output measurements have been performed in open loop at about 20 frequencies in the range 0.5 to 40 Hz, and with some different command amplitudes: 0.5, 1, and 2 A r.m.s. values (Figs. 30.4 and 30.5);
- samples of frequency response disturbance to output (only magnitude) $|H(j\omega)|$ deduced from spectral densities of vertical acceleration of both ground and upper platform B in open loop normal operating conditions; frequency range is 0.1 to 9 Hz, with frequency resolution 0.2 Hz; this range of frequency contains more than 95% of both disturbance and output power (Fig. 30.6).

These data can be described by the equation

$$y = F(p) + e, \tag{30.11}$$

where y is the vector containing the samples

$$y = [\,|G(j\omega_1)|, |G(j\omega_2)|, \ldots, \arg G(j\omega_1), \arg G(j\omega_2), \ldots,$$

$$|H(j\omega_1)|, |H(j\omega_2)|, \ldots\,], \tag{30.12}$$

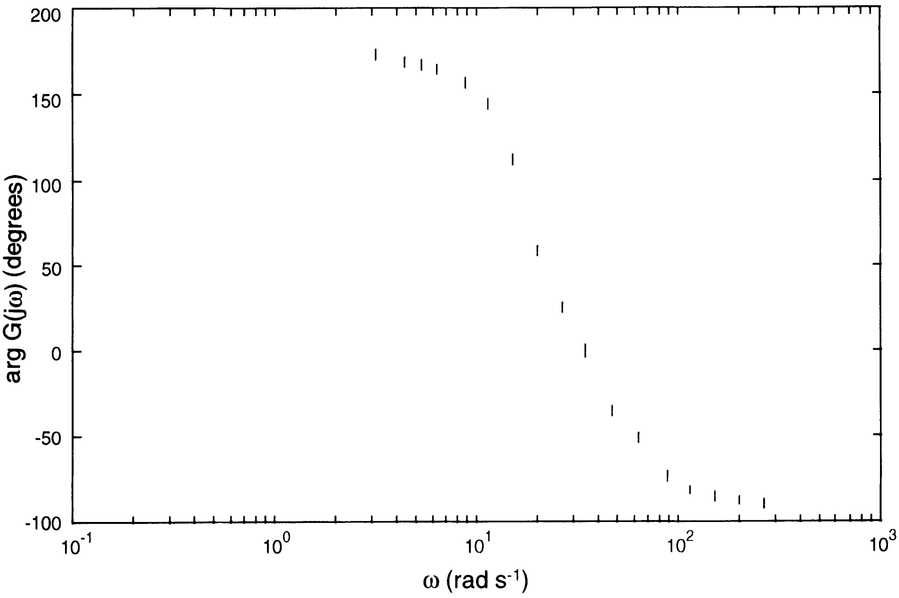


FIGURE 30.5. Samples of arg $G(j\omega)$ with corresponding uncertainty intervals.

p is the vector containing the unknown parameters

$$p = [k_1, k_2, \beta_1, \beta_2, m_1], \tag{30.13}$$

and e is an error term.

Accuracy is the main concern of the calibration problem. Thus, particular attention has been paid to the computation of parameter estimate accuracy, which depends on the knowledge about the error term e and on the estimation algorithm.

A statistical description of e does not appear appropriate, mainly due to the modeling errors discussed above. An unknown but bounded error approach has been adopted, which requires information on measurement error bounds only.(6,7)

Uncertainty intervals v_i are evaluated for each data. They take into account the accuracy of the measurement devices and the effects of unmodeled dynamics, expressed by the bounding function shown in Fig. 30.7. The used values are reported in Figs. 30.4, 30.5, and 30.6. In such a way, the information on e is given as

$$\|e\|_\infty^v \leq 1, \tag{30.14}$$

where $\|e\|_\infty^v = \max_i v_i^{-1}|e_i|$, $v_i > 0$, and v is the vector of error bounds corresponding to the uncertainty intervals on data.

A least square estimate p^0 of the unknown parameters is obtained as

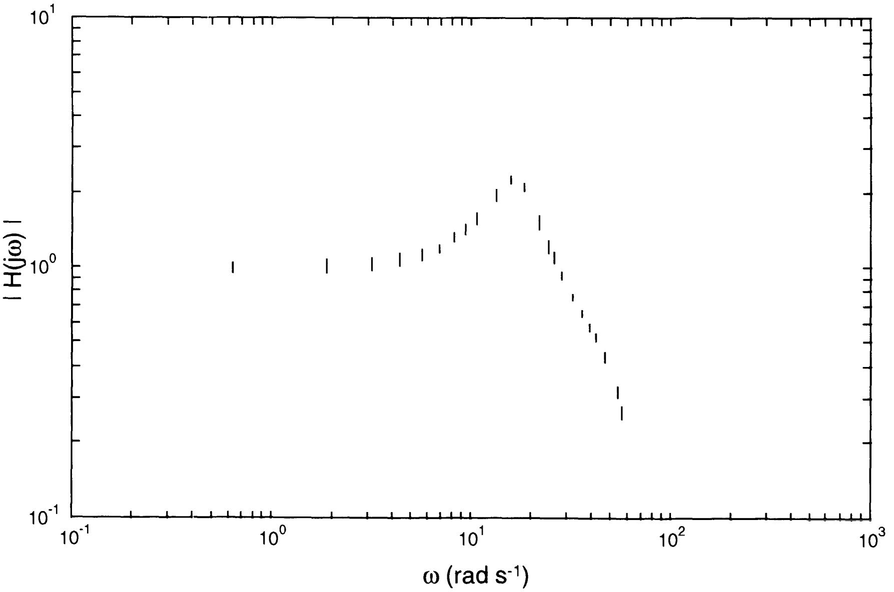


FIGURE 30.6. Samples of $|H(j\omega)|$ with corresponding uncertainty intervals.

$$p^0 := \arg\min_p\{\|y - F(p)\|_2^V\}, \tag{30.15}$$

where $\|\cdot\|_2^V$ denotes a weighted Euclidian norm, and V is a diagonal matrix with elements v_i.

Uncertainty in data induces uncertainty on the parameter estimated values. The maximal range of variation of the i-th component p_i^0, due to possible errors consistent with Eq. (30.14) is indicated as estimate uncertainty interval u_i. The u_is are evaluated by a method proposed in Ref. 8. The method is based on a quasi-linearization technique and gives intervals certainly contained within the exact u_is. Indeed, it gives exact values if $\partial p_i^0/\partial y_i$ has a constant sign for all $y \in Y$. The set Y is defined as

$$Y = \{y : \|y - \hat{y}\|_\infty^v \leq 1\}, \tag{30.16}$$

where $\hat{y}$ denotes actual measurements.

This condition is difficult to be checked with certainty. However, $\partial p^0/\partial y$ is evaluated at several grid points to provide good evidence that the condition is met. This has been accomplished by using the formula:[8]

$$\frac{\partial p^0}{\partial y} = (A^T V A)^{-1} A^T V, \tag{30.17}$$

where

$$A = \left.\frac{\partial F}{\partial p}\right|_{p^0(v)}. \tag{30.18}$$

In turn, $\frac{\partial F}{\partial p}$ is computed as a function of p by means of the symbolic manipulation package DERIVE.

The identification procedure described above gives the following results:

$$\begin{aligned} k_1 &= p_1 = p_1^0 \pm u_1 = (1.4 \pm 0.3)10^6\, Nm^{-1} \\ k_2 &= p_2 = p_2^0 \pm u_2 = (1.0 \pm 0.3)10^6\, Nm^{-1} \\ \beta_1 &= p_3 = p_3^0 \pm u_3 = (4.8 \pm 1.0)10^4\, Nsm^{-1} \\ \beta_2 &= p_4 = p_4^0 \pm u_4 = (1.7 \pm 0.3)10^4\, Nsm^{-1} \\ m_1 &= p_5 = p_5^0 \pm u_5 = (4.2 \pm 0.7)10^3\, kg. \end{aligned} \tag{30.19}$$

Note that other estimators could give smaller estimate uncertainty intervals. In particular, methods to compute minimal uncertainty intervals estimators are proposed in Refs. 9, 10, and 11. However, a least-square estimator has been adopted because it is computationally faster and has better robustness properties with respect to inexact knowledge of uncertainty error bounds v_i.(7)

In order to complete the perturbation model, the weighting function $W(\omega)$ of Eq. (30.8) has to be evaluated. To this extent, effects of ground deformability has been simulated by space discretization of the ground. This discretization introduces several couples of zeros and poles in the transfer function $G(s)$, which are close to the imaginary axis. These low damped zero-pole couples give rise to peaks in the transfer function at relatively high frequencies ($\omega > 1000$ *rad/s*, see Fig. 30.7). Now $W(\omega)$ has to be chosen so that $|G(j\omega,p^o)|(1 \pm |W(\omega)|)$ envelopes these peaks. The methods used for the design and the robustness analysis require $W(\omega)$ to be a rational stable function, whose order affects the complexity of the controller and the computational burden of the analysis. A first order, high pass (for $\omega > 1000$ *rad/s*) function is then chosen:

$$W(\omega) = \frac{1.22 j\omega}{j\omega + 1000}. \tag{30.20}$$

In summary, the perturbed model, which is adopted for the purpose of robust performances regulator design, is the mixed parametric and dynamically perturbed model represented in Fig. 30.3. The forms of $G(s,p)$ and $H(s,p)$ are given by Eqs. (30.1–30.5). Symbols $G(s,p)$ and $H(s,p)$ denote explicitly the dependence of the plant transfer functions on the uncertain parameter vector p.

Parametric uncertainty is represented by the fact that parameter vector p is known only to belong to the parameter uncertainty set Π, defined as

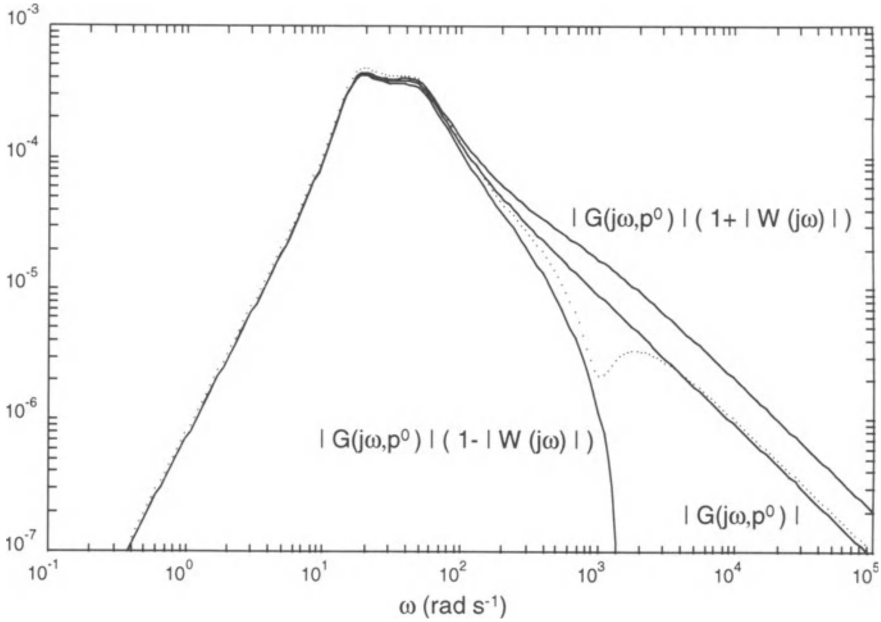


FIGURE 30.7. Bounds of perturbed transfer function $G(s,p^0)[1 + \Delta]$; transfer function of discretized ground model is in dotted lines.

$$\Pi = \{p \in \mathbf{R}^5 : \|p - p^0\|_\infty^u \le 1\}. \tag{30.21}$$

Note that p^0 is the 'nominal' parameter vector, whose components are given by the p_i^0s in Eqs. (30.19), and the vector u has components as the estimates uncertainty intervals u_i in the same equations.

Unstructured uncertainty is represented by the multiplicative dynamic perturbation Δ, known only to belong to the modeling error set, Δ_E, defined as

$$\Delta_E = \{\Delta : \|W^{-1}(\omega)\Delta(\omega)\|_\infty \le 1, \quad 0 \le \omega \le \infty\}, \tag{30.22}$$

where $W(\omega)$ is given by Eq. (30.20).

30.4. ROBUST REGULATOR DESIGN

The feedback control configuration shown in Fig. 30.3 is considered for output regulation of the plant.

The loop transfer function

$$L(s,p,\Delta) = A(s)G(s,p)[1 + \Delta]M(s)C(s) \tag{30.23}$$

is now considered. Symbol $L(s,p,\Delta)$ denotes that both structured and unstructured perturbations affect the loop transfer function.

Furthermore, the sensitivity function $S(s,p,\Delta)$ and its complement $T(s,p,\Delta)$ are considered:

$$S = (1 + L)^{-1}, \quad T = 1 - S = L(1 + L)^{-1}. \tag{30.24}$$

The robust regulation problem requires to design a relatively low order controller $C(s)$ such that the following specifications are met:

1. Closed loop is robustly stable with respect to the allowed structured and unstructured uncertainties, i.e., $\forall p \in \Pi$ and $\forall\ \Delta \in \Delta_E$.
2. Closed loop control guarantees disturbance attenuation of 1:100 in r.m.s. values with respect to open loop operation. This is obtained by imposing:

   $$|S(j\omega,p,\Delta)| \leq U(\omega), \ \ \forall p \in \Pi, \forall\, \Delta{\in}\Delta_E, 0.63 \leq \omega \leq 57\ rad/s, \tag{30.25}$$

   where $U(\omega)$ is the bounding function shown in Fig. 30.8, and [0.63, 57] *rad/s* is the frequency range where the spectral density function $S_d(\omega)$ of the disturbance has more than 95% of its power (see Fig. 30.9).

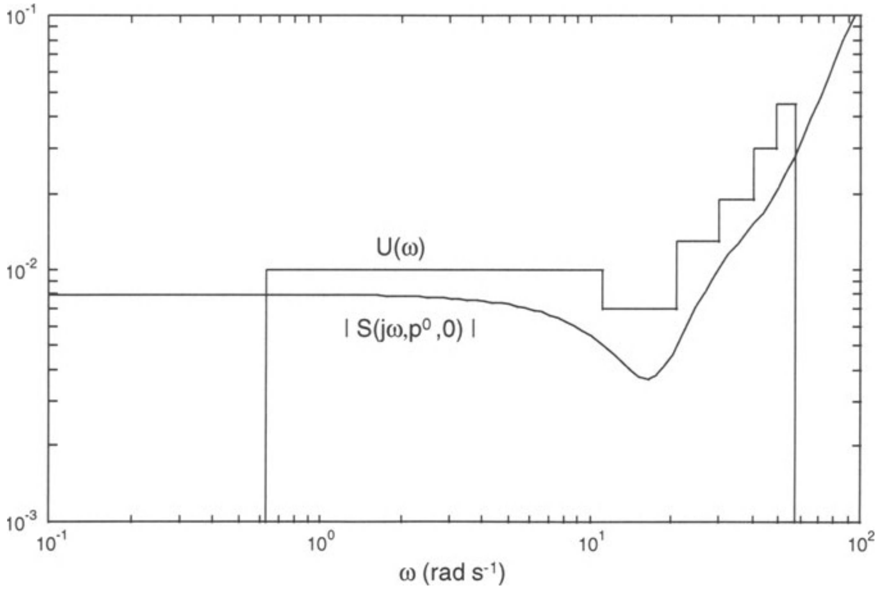


FIGURE 30.8. Nominal sensitivity ($C_K(s) = C_{10}(s)$) and bounding function.

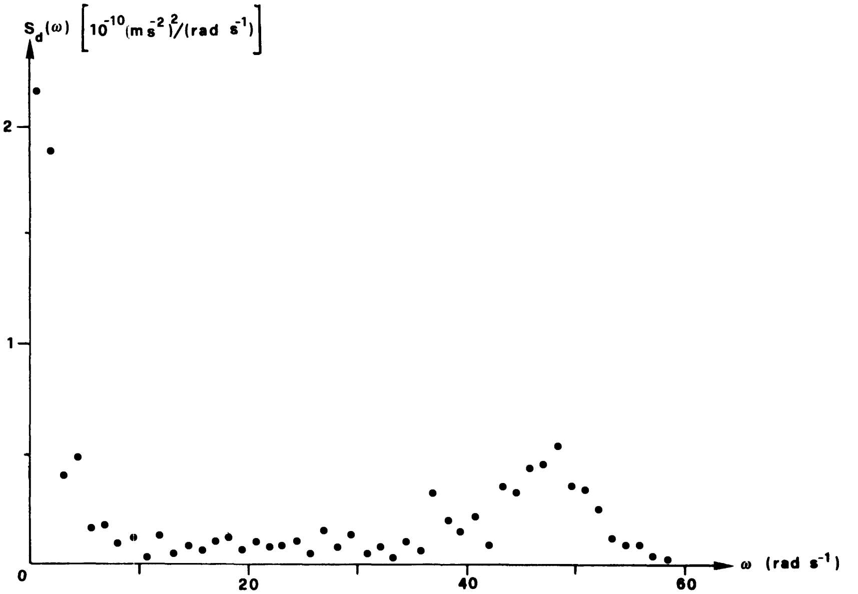


FIGURE 30.9. Spectral density of disturbance.

3. Command amplitude is limited, in order to guarantee that the current in the electromagnetic actuators does not exceed 10 A r.m.s. This is obtained by imposing:

$$|H(j\omega,p)S(j\omega,p,\Delta)M(j\omega)C(j\omega)A(j\omega)| \le 6 \cdot 10^6\, Am^{-1}s^2$$

$$\forall p \in \Pi,\ \forall \Delta \in \Delta_E,\ 0.63 \le \omega \le 57\ rad/s. \qquad (30.26)$$

A systematic approach to a design problem of such a complexity is outside the possibilities offered by the present state of the robust control literature. Consequently a design approach is used based on iterative phases of "nominal" design and robustness analysis, that, in case of failure, gives indications for the successive design phase.

The design phase has been carried on, using the nominal parameters, in the following way: the robust stability requirement with respect to unmodeled dynamics and the given performance specifications are used to suitably define the weighting functions entering in the H_∞ design approach. Through H_∞ synthesis algorithms it is possible to find a controller that satisfies the specifications for the "nominal" parameters, or to understand if some of the specifications have to be relaxed. Note that the controller obtained by H_∞ synthesis may be of higher order than desirable. It is often possible, however, to find a simpler controller satisfying the nominal performance by order reduction techniques. This design phase uses an interactive design software,(12) implemented with the Robust-Control Toolbox of MATLAB™.

The robustness analysis phase has been carried on following the approach proposed in Ref. 4.

First, performance specifications are represented in the frequency domain by functions $F_k(\omega,p)$, $k = 1, 2, 3$. The k-th performance specification is satisfied for all unstructured perturbation $\Delta \in \Delta_E$ if and only if $F_k(\omega,p) > 0$, $\omega \in \Omega_k$. For example, in the case of stability, the corresponding specification inequality can be obtained by the small gain theorem, giving:

$$|W(\omega)T(j\omega,p,0)| < 1. \tag{30.27}$$

A given compensator is said to achieve robust k-performance if

$$F_k(\omega,p) > 0,\ \omega \in \Omega_k,\ \forall p \in \Pi.$$

Note that in this way, robustness is guaranteed versus both parametric and unstructured uncertainty.

A robustness measure for the k-th performance, called performance margin ρ_k^*, is defined as the radius according to l_∞^u norm in parameter space of the maximal ball. It is centered at the nominal parameter p^0, such that the closed loop system preserves the given performance specification for every parameter vector belonging to the maximal ball and for all admissible unstructured perturbations. This measure is a generalization of the widely used concept of stability margin.(13,14,15)

Performance margin ρ_k^* can be computed as

$$\rho_k^* = \min_{p,\omega,\rho} \rho$$

subject to

$$\begin{cases} \rho \geq 0 \\ |p_i - p_i^0| \leq \rho u_i \quad i = 1, \ldots, 4 \\ F_k(\omega,p) \leq 0, \omega \in \Omega_k. \end{cases} \tag{30.28}$$

Note that the k-th performance is robustly achieved if and only if $\rho_k^* \geq 1$.

The optimization problem of Eq. (30.28) may have local minima and the global solution is needed for solving the problem. Global optimization methods based on random search algorithms are not appropriate, since these methods guarantee convergence to the global minimum only in probability. More importantly, they do not give any measure on how far the obtained solution is from the true global minimum. When $F(\omega,p)$ is a polynomial function in ω and p, algorithms able to produce a sequence of upper and lower bounds converging with certainty to global extrema have been proposed, and shown to be able to solve some nontrivial robustness problems.(4,15,16) Indeed, the given specifications can be represented by polynomial inequalities in p and ω. The explicit functional expressions $F_k(\omega,p)$, $k = 1, 2, 3$, corresponding to the considered performances, have been obtained by means of symbolic manipulation package DERIVE™ on a personal computer.

The results reported in this chapter are obtained by use of the algorithm reported in Ref. 17, an improvement over previously cited ones.

Note that in case the controller is found not robustly performing, (i.e., $\rho_k^* < 1$ for some k), the above analysis gives useful indications for the design phase. In particular, if the robustness margin of one performance is less than one, a new controller may be designed by strengthening the corresponding specification and possibly relaxing the specifications with robustness margin greater than one.

Following the described approach, a compensator has been found to satisfy performance specifications for nominal parameter p^o:

$$\tilde{C}(s) = 5\frac{(1+s/0.5)(1+s/11)(1+s/22)^2(1+s/67)(1+s/405)(1+s/10^3)}{s^3(1+s/57)^2(1+s/58.8)(1+s/1065)(1+s/9\cdot 10^{10})}. \qquad (30.29)$$

This transfer function has been found to be reasonably well approximated by the fourth order transfer function:

$$C_{10}(s) = 10\frac{(1+s/0.5)(1+s/50)^2(1+s/1000)}{s^3(1+s/200)}. \qquad (30.30)$$

This controller largely satisfies the given specifications in correspondence to the nominal model: the closed loop is stable with a damping factor of 0.4, the disturbance effect on the output is 1:120 with respect to the open loop, the current in the electromagnetic actuators does not exceed 7 A. As shown in Table 30.1, however, it does not achieve robust disturbance attenuation. This suggests that the gain loop has to be raised.

Indeed, compensator $C_{14}(s) = 1.4C_{10}(s)$ achieves all robust performances. Note that the stability margin for this compensator is near to one. Higher gain may cause stability problems as confirmed, for example, by the robustness analysis of $C_{20}(s) = 2C_{10}(s)$.

TABLE 30.1. Specification Margins and Corresponding Computing Times on a VAX 9000 Computer

	$C_{10}(s)$		$C_{14}(s)$		$C_{20}(s)$	
Specification	ρ^*	CPU Time (sec)	ρ^*	CPU Time (sec)	ρ^*	CPU Time (sec)
1	1.18	128	1.08	80	0.81	104
2	0.63	15	1.46	23	2.16	89
3	3.33	5	3.33	5	3.33	4

Note that the robustness analysis is carried out by an approach able to exploit the nonlinear structure induced by the considered perturbed parameters. It is known that this is a hard problem that requires computationally cumbersome algorithms. Computationally simpler techniques have also been tried, by considering an interval plant formulation.(18,19) This approach leads to conservative results, however. For the present case study, the resulting conservativeness is high enough to prevent the possibility of finding a controller guaranteed to perform robustly by such a simplified analysis.(20) On the other hand, the computational burden of the nonlinear analysis appears to be acceptable (see computing times in Table 30.1) and worth paying, considering the improved results obtained.

30.5. CONCLUSIONS

The presented case study illustrates how some robust identification and control techniques can be applied in dealing with real world problems.

The modeling and identification step has been performed by using physical insight and set membership identification theory, able to account for parametric variations and modeling approximations.

Robustness of the control system with respect to both types of uncertainty is considered, not for stability only, but for other performance specifications, such as disturbance attenuation and command power limitation.

An iterative design strategy has been adopted, based on successive phases of "nominal" design and robustness analysis.

The main conclusion drawn from this case study is that, in facing with real world problems, it is necessary to take approaches with the following features: ability to deal with nonlinear physical parametrizations; accounting for both parametric uncertainty and unmodeled dynamics; and designing and analyzing with respect to different performances.

These requirements clearly lead to difficult problems, but it appears that techniques now exist to solve cases of such a complexity to be of some interest in practical applications.

ACKNOWLEDGMENTS. This work was partially supported by funds of Ministero dell'Università e della Ricerca Scientifica e Tecnologica and of CNR Special Project "Algorithms and Architectures for Identification and Robust and Adaptive Control."

REFERENCES

1. P. Dorato and R. K. Yedavally, eds., *Recent Advances in Robust Control*, IEEE Press (1990).

2. M. Milanese, R. Tempo, and A. Vicino, eds., *Robustness in Identification and Control*, Plenum Press, New York (1989).
3. M. K. H. Fan, A. L. Tits, and J. C. Doyle, *IEEE Trans. on Autom. Control* **AC-36**, 25 (1991).
4. M. Milanese, G. Fiorio, S. Malan, and A. Vicino, in: *Robust Control* (S. P. Bhattacharya and L. Keel, eds.) CRC Press, Boca Raton (1991).
5. G. Fiorio, S. Malan, M. Milanese, and A. Vicino, in: *Proc. 29th IEEE Conference on Decision and Control*, Honolulu (1990).
6. F. C. Schweppe, *Uncertain Dynamic Systems*, Prentice-Hall, Englewood Cliffs, NJ (1973).
7. M. Milanese and A. Vicino, *Automatica* **27**, 997 (1991); M. Milanese and A. Vicino, in: *Bounding Approaches to System Identification* (M. Milanese *et al.* eds.) Plenum Press, New York, Chap. 2 (1996).
8. G. Belforte and M. Milanese in: *Proc. 1st IASTED Symp. Modeling, Identification and Control*, Davos, Switzerland (1981), pp. 222–228.
9. M. Milanese and G. Belforte, *IEEE Trans. on Autom. Control* **AC-27**, 408 (1982).
10. M. Milanese and R. Tempo, *IEEE Trans. on Autom. Control* **AC-30**, 730 (1985).
11. M. Milanese and A. Vicino, *Automatica* **25**, 403 (1991).
12. M. Ferro, *Progetto Assistito da Calcolatore di un Sistema di Controllo Robusto H_∞ sulla Base di Specifiche Classiche*, *Tesi di Laurea*, Politecnico di Torino, Torino, Italy (1992).
13. R. R. E. de Gaston and M. G. Safonov, *IEEE Trans. on Autom. Control* **AC-33**, 156 (1988).
14. D. D. Šiljak, *IEEE Trans. on Autom. Control* **AC-34**, 674 (1989).
15. A. Vicino, A. Tesi, and M. Milanese, *IEEE Trans. on Autom. Control* **AC-35**, 845 (1990).
16. A. Vicino and M. Milanese, in: *Control of Uncertain Systems* (D. Hinrichsen and B. Martensson, eds.) Birkäuser, Boston, MA (1990).
17. M. Malan, M. Milanese, M. Taragna, and J. Garloff, in: *Proc. 31st IEEE Conference on Decision and Control*, Tucson, AZ, pp. 128–133 (1992).
18. H. Chapellat and S. P. Bhattacharyya, *IEEE Trans. on Automatic Control* **AC-34**, 306 (1989).
19. A. Tesi and A. Vicino, *Proc. International Workshop on Robust Control* CRC Press, San Antonio, TX, pp. 403–416 (1991).
20. M. Milanese, M. Taragna, A. Trisoglio, and S. Malan, in: *Robustness of Dynamic Systems with Parametric Uncertainties* (M. Mansour, S. Balemi, and W. Truol, eds.) Birkhauser, Boston, MA pp. 211–218 (1992).

Index

ACKNOWLEDGMENTS

The editors undertook the preparation of this volume as members of the European Union SIMONET network of groups working on topics in identification. The network has as one of its main aims the furtherance of research into bound-based identification.