NASA
Reference
Publication
1138

February 1985

Identification of Dynamic Systems

Theory and Formulation

Richard E. Maine
and Kenneth W. Iliff

(NASA-RP-1138) IDENTIFICATION OF DYNAMIC SYSTEMS, THEORY AND FORMULATION (NASA) 138 p HC A07/MF A01 CSCL 12B
N85-19784
Unclas
H1/66 13589

RECEIVED
NASA STI FACILITY
ACCESS DEPT.

NASA

NASA
Reference
Publication
1138

1985

Identification of Dynamic Systems

Theory and Formulation

Richard E. Maine
and Kenneth W. Iliff

Ames Research Center
Dryden Flight Research Facility
Edwards, California

NASA
National Aeronautics
and Space Administration

Scientific and Technical
Information Branch

PREFACE

The subject of system identification is too broad to be covered completely in one book. This document is restricted to statistical system identification; that is, methods derived from probabilistic mathematical statements of the problem. We will be primarily interested in maximum-likelihood and related estimators. Statistical methods are becoming increasingly important with the proliferation of high-speed, general-purpose digital computers. Problems that were once solved by hand-plotting the data and drawing a line through them are now done by telling a computer to fit the best line through the data (or by some completely different, formerly impractical method). Statistical approaches to system identification are well-suited to computer application.

Automated statistical algorithms can solve more complicated problems more rapidly—and sometimes more accurately—than the older manual methods. There is a danger, however, of the engineer's losing the intuitive feel for the system that arises from long hours of working closely with the data. To use statistical estimation algorithms effectively, the engineer must have not only a good grasp of the system under analysis, but also a thorough understanding of the analytic tools used. The analyst must strive to understand how the system behaves and what characteristics of the data influence the statistical estimators in order to evaluate the validity and meaning of the results.

Our primary aim in this document is to provide the practicing data analyst with the background necessary to make effective use of statistical system identification techniques, particularly maximum-likelihood and related estimators. The intent is to present the theory in a manner that aids intuitive understanding at a concrete level useful in application. Theoretical rigor has not been sacrificed, but we have tried to avoid "elegant" proofs that may require three lines to write, but 3 years of study to comprehend the underlying theory. In particular, such theoretically intriguing subjects as martingales and measure theory are ignored. Several excellent volumes on these subjects are available, including Balakrishnan (1973), Royden (1968), Rudin (1974), and Kushner (1971).

We assume that the reader has a thorough background in linear algebra and calculus (Paige, Swift, and Slobko, 1974; Apostol, 1969; Nering, 1969; and Wilkinson, 1965), including complete familiarity with matrix operations, vector spaces, inner products, norms, gradients, eigenvalues, and related subjects. The reader should be familiar with the concept of function spaces as types of abstract vector spaces (Luenberger, 1969), but does not need expertise in functional analysis. We also assume familiarity with concepts of deterministic dynamic systems (Zadeh and Desoer, 1963; Wiberg, 1971; and Levan, 1983).

Chapter 1 introduces the basic concepts of system identification. Chapter 2 is an introduction to numerical optimization methods, which are important to system identification. Chapter 3 reviews basic concepts from probability theory. The treatment is necessarily abbreviated, and previous familiarity with probability theory is assumed.

Chapters 4-10 present the body of the theory. Chapter 4 defines the concept of an estimator and some of the basic properties of estimators. Chapter 5 discusses estimation as a static problem in which time is not involved. Chapter 6 presents some simple results on stochastic processes. Chapter 7 covers the state estimation problem for dynamic systems with known coefficients. We first pose it as a static estimation problem, drawing on the results from Chapter 5. We then show how a recursive formulation results in a simpler solution process, arriving at the same state estimate. The derivation used for the recursive state estimator (Kalman filter) does not require a background in stochastic processes; only basic probability and the results from Chapter 5 are used.

Chapters 8-10 present the parameter estimation problem for dynamic systems. Each chapter covers one of the basic estimation algorithms. We have considered parameter estimation as a problem in its own right, rather than forcing it into the form of a nonlinear filtering problem. The general nonlinear filtering problem is more difficult than parameter estimation for linear systems, and it requires *ad hoc* approximations for practical implementation. We feel that our approach is more natural and is easier to understand.

Chapter 11 examines the accuracy of the estimates. The emphasis in this chapter is on evaluating the accuracy and analyzing causes of poor accuracy. The chapter also includes brief discussions about the roles of model structure determination and experiment design.

PRECEDING PAGE BLANK NOT FILMED

TABLE OF CONTENTS

	Page
PREFACE	iii
NOMENCLATURE	ix
1.0 INTRODUCTION	1
1.1 SYSTEM IDENTIFICATION	2
1.2 PARAMETER IDENTIFICATION	3
1.3 TYPES OF SYSTEM MODELS	5
1.3.1 Explicit Function	5
1.3.2 State Space	5
1.3.3 Others	7
1.4 PARAMETER ESTIMATION	7
1.5 OTHER APPROACHES	10
2.0 OPTIMIZATION METHODS	11
2.1 ONE-DIMENSIONAL SEARCHES	12
2.2 DIRECT METHODS	12
2.3 GRADIENT METHODS	13
2.4 SECOND ORDER METHODS	15
2.4.1 Newton-Raphson	15
2.4.2 Invariance	16
2.4.3 Singularities	17
2.4.4 Quasi-Newton Methods	18
2.5 SUMS OF SQUARES	18
2.5.1 Linear Case	19
2.5.2 Nonlinear Case	19
2.6 CONVERGENCE IMPROVEMENT	21
3.0 BASIC PRINCIPLES FROM PROBABILITY	23
3.1 PROBABILITY SPACES	23
3.1.1 Probability Triple	23
3.1.2 Conditional Probabilities	23
3.2 SCALAR RANDOM VARIABLES	23
3.2.1 Distribution and Density Functions	23
3.2.2 Expectations and Moments	24
3.3 JOINT RANDOM VARIABLES	24
3.3.1 Distribution and Density Functions	24
3.3.2 Expectations and Moments	24
3.3.3 Marginal and Conditional Distributions	25
3.3.4 Statistical Independence	25
3.4 TRANSFORMATION OF VARIABLES	26
3.5 GAUSSIAN VARIABLES	26
3.5.1 Standard Gaussian Distributions	27
3.5.2 General Gaussian Distributions	27
3.5.3 Properties	30
3.5.4 Central Limit Theorem	33
4.0 STATISTICAL ESTIMATORS	35
4.1 DEFINITION OF AN ESTIMATOR	35
4.2 PROPERTIES OF ESTIMATORS	36
4.2.1 Unbiased Estimators	36
4.2.2 Minimum Variance Estimators	37
4.2.3 Cramer-Rao Inequality (Efficient Estimators)	37
4.2.4 Bayesian Optimal Estimators	39
4.2.5 Asymptotic Properties	39
4.3 COMMON ESTIMATORS	40
4.3.1 *A posteriori* Expected Value	40
4.3.2 Bayesian Minimum Risk	40
4.3.3 Maximum *a posteriori* Probability	41
4.3.4 Maximum Likelihood	42
5.0 THE STATIC ESTIMATION PROBLEM	45
5.1 LINEAR SYSTEMS WITH ADDITIVE GAUSSIAN NOISE	45
5.1.1 Joint Distribution of Z and ξ	45
5.1.2 *A posteriori* Estimators	46
5.1.3 Maximum Likelihood Estimator	48
5.1.4 Comparison of Estimators	49
5.2 PARTITIONING IN ESTIMATION PROBLEMS	50
5.2.1 Measurement Partitioning	50
5.2.2 Application to Linear Gaussian System	52
5.2.3 Parameter Partitioning	53
5.3 LIMITING CASES AND SINGULARITIES	54
5.3.1 Singular P	55
5.3.2 Singular GG^*	55
5.3.3 Singular $CPC^* + GG^*$	56
5.3.4 Infinite P	57
5.3.5 Infinite GG^*	58
5.3.6 Singular $C^*(GG^*)^{-1}C + P^{-1}$	58

PRECEDING PAGE BLANK NOT FILMED

5.4 NONLINEAR SYSTEMS WITH ADDITIVE GAUSSIAN NOISE . . . 58
5.4.1 Joint Distribution of Z and ξ . . . 58
5.4.2 Estimators . . . 59
5.4.3 Computation of the Estimates . . . 60
5.4.4 Singularities . . . 61
5.4.5 Partitioning . . . 61
5.5 MULTIPLICATIVE GAUSSIAN NOISE (ESTIMATION OF VARIANCE) . . . 61
5.6 NON-GAUSSIAN NOISE . . . 64

6.0 STOCHASTIC PROCESSES . . . 69
6.1 DISCRETE TIME . . . 69
6.1.1 Linear Systems Forced by Gaussian White Noise . . . 69
6.1.2 Nonlinear Systems and Non-Gaussian Noise . . . 70
6.2 CONTINUOUS TIME . . . 70
6.2.1 Linear Systems Forced by White Noise . . . 70
6.2.2 Additive White Measurement Noise . . . 72
6.2.3 Nonlinear Systems . . . 72

7.0 STATE ESTIMATION FOR DYNAMIC SYSTEMS . . . 73
7.1 EXPLICIT FORMULATION . . . 73
7.2 RECURSIVE FORMULATION . . . 75
7.2.1 Prediction Step . . . 75
7.2.2 Correction Step . . . 76
7.2.3 Kalman Filter . . . 76
7.2.4 Alternate Forms . . . 77
7.2.5 Innovations . . . 78
7.3 STEADY-STATE FORM . . . 79
7.4 CONTINUOUS TIME . . . 81
7.5 CONTINUOUS/DISCRETE TIME . . . 82
7.6 SMOOTHING . . . 84
7.7 NONLINEAR SYSTEMS AND NON-GAUSSIAN NOISE . . . 86

8.0 OUTPUT ERROR METHOD FOR DYNAMIC SYSTEMS . . . 89
8.1 DERIVATION . . . 90
8.2 INITIAL CONDITIONS . . . 91
8.3 COMPUTATIONS . . . 91
8.3.1 Gauss-Newton Method . . . 91
8.3.2 System Response . . . 92
8.3.3 Finite Difference Response Gradient . . . 93
8.3.4 Analytic Response Gradient . . . 93
8.4 UNKNOWN G . . . 94
8.5 CHARACTERISTICS . . . 95

9.0 FILTER ERROR METHOD FOR DYNAMIC SYSTEMS . . . 97
9.1 DERIVATION . . . 97
9.1.1 Static Derivation . . . 97
9.1.2 Derivation by Recursive Factoring . . . 98
9.1.3 Derivation Using the Innovation . . . 98
9.1.4 Steady-State Form . . . 99
9.1.5 Cost Function Discussion . . . 99
9.2 COMPUTATION . . . 100
9.3 FORMULATION AS A FILTERING PROBLEM . . . 100

10.0 EQUATION ERROR METHOD FOR DYNAMIC SYSTEMS . . . 101
10.1 PROCESS-NOISE APPROACH . . . 101
10.1.1 Derivation . . . 101
10.1.2 Special Case of Filter Error . . . 102
10.1.3 Discussion . . . 103
10.2 GENERAL EQUATION ERROR FORM . . . 104
10.2.1 Discrete State-Equation Error . . . 104
10.2.2 Continuous/Discrete State-Equation Error . . . 104
10.2.3 Observation-Equation Error . . . 106
10.3 COMPUTATION . . . 106
10.4 DISCUSSION . . . 107

11.0 ACCURACY OF THE ESTIMATES . . . 109
11.1 CONFIDENCE REGIONS . . . 110
11.1.1 Random Parameter Vector . . . 110
11.1.2 Nonrandom Parameter Vector . . . 111
11.1.3 Gaussian Approximation . . . 112
11.1.4 Nonstatistical Derivation . . . 113
11.2 ANALYSIS OF THE CONFIDENCE ELLIPSOID . . . 113
11.2.1 Sensitivity . . . 113
11.2.2 Correlation . . . 114
11.2.3 Cramer-Rao Bound . . . 116
11.3 OTHER MEASURES OF ACCURACY . . . 117
11.3.1 Bias . . . 117
11.3.2 Scatter . . . 118
11.3.3 Engineering Judgment . . . 118
11.4 MODEL STRUCTURE DETERMINATION . . . 119
11.5 EXPERIMENT DESIGN . . . 120

12.0 SUMMARY 125

A.0 MATRIX RESULTS 127
A.1 MATRIX INVERSION LEMMAS 127
A.2 MATRIX DIFFERENTIATION 129

REFERENCES 131

NOMENCLATURE

SYMBOLS

It is impractical to list all of the symbols used in this document. The following are symbols of particular significance and those used consistently in large portions of the document. In several specialized situations, the same symbols are used with different meanings not included in this list.

A	stability matrix
B	control matrix
$b(.)$	bias
C	state observation matrix
D	control observation matrix
$E\{.\}$	expected value
e	error vector
$F(.)$	system function
FF^*	process noise covariance matrix
$F_x(.)$	probability distribution function of x
$f(.)$	system state function
GG^*	measurement noise covariance matrix
$g(.)$	system observation function
$h(.)$	equation error function
$J(.)$	cost function
M	Fisher information matrix
m_ξ	prior mean of ξ
$n_i, n(t)$	process noise vector
P	prior covariance of ξ, or covariance of filtered x
$p(x)$	probability density function of x, short notation
$p_x(.)$	probability density function of x, full notation
Q	covariance of predicted x
R	covariance of innovation
t	time
U	system input
$u_i, u(t)$	dynamic system input vector
V_i	concatenated innovation vector
v	innovation vector
x	parameter vector in static models
$x_i, x(t)$	dynamic system state vector
Z	system response
Z_i	concatenated response vector
$z_i, z(t)$	dynamic system response vector
Δ	sample interval
η_i	measurement noise vector
Φ	state transition matrix

PRECEDING PAGE BLANK NOT FILMED

Ψ	input transition matrix
ξ	vector of unknown parameters
Ξ	set of possible parameter values
ω	random noise vector
Ω	probability space
$\tilde{}$	predicted estimate (in filtering contexts)
$\hat{}$	optimum (in optimization contexts), or estimate (in estimation contexts), or filtered estimate (in filtering contexts)
$\bar{}$	smoothed estimate

Subscript ξ indicates dependence on ξ

Abbreviations and acronyms

$\arg\max_x$	value of x that maximizes the following function
corr	correlation
cov	covariance
exp	exponential
ln	natural logarithm
MAP	maximum *a posteriori* probability
MLE	maximum-likelihood estimator
mse	mean-square error
var	variance

Mathematical notation

$f(.)$	the entire function f, as opposed to the value of the function at a particular point
$*$	transpose
∇_x	gradient with respect to the vector x (result is a row vector when the operand is a scalar, or a matrix when the operand is a column vector)
∇^2_x	second gradient with respect to x
Σ	series summation
Π	series product
π	3.14159...
$\cup$	set union
$\cap$	set intersection
$\subset$	subset
$\in$	element of a set
$\{x:c\}$	the set of all x such that condition c holds
$\langle\cdot,\cdot\rangle$	inner product
$\mid$	conditioned on (in probability contexts)
$\lvert\cdot\rvert$	absolute value or determinant
$d\lvert\cdot\rvert$	volume element
t_i^+	right-hand limit at t_i
n-vector	vector with n elements

$x^{(i)}$ ith element of the vector x, or ith row of the matrix x

A lower case subscript generally indicates an element of a sequence

CHAPTER 1

1.0 INTRODUCTION

System identification is broadly defined as the deduction of system characteristics from measured data. It is commonly referred to as an inverse problem because it is the opposite of the problem of computing the response of a system with known characteristics. Gauss (1809, p. 85) refers to "the inverse problem, that is when the true is to be derived from the apparent place." The inverse problem might be phrased as, "Given the answer, what was the question?" Phrased in such general terms, system identification is seen as a simple concept used in everyday life, rather than as an obscure area of mathematics.

> Example 1.0-1 The system is your body, and the characteristic of interest is its mass. You perform an experiment by placing the system on a mechanical transducer in the bathroom which gives as output a position approximately proportional to the system mass and the local gravitational field. Based on previous comparisons with the doctor's scales, you know that your scale consistently reads 2 lb high, so you subtract this figure from the reading. The result is still somewhat higher than expected, so you step off of the scales and then repeat the experiment. The new reading is more "reasonable" and from it you obtain an estimate of the system mass.

This simple example actually includes several important principles of system identification; for instance, the resulting estimates are biased (as defined in Chapter 4).

> Example 1.0-2 The "guess your weight" booth at the fair.

The weight guesser's instrumentation and estimation algorithm are more difficult to describe precisely, but they are used to solve the same system identification problem.

> Example 1.0-3 Newton's deduction of the theory of gravity.

Newton's problem was much more difficult than the first two examples. He had to deduce not just a single number, but also the form of the equations describing the system. Newton was a true expert in system identification (among other things).

As apparent from the above examples, system identification is as much an art as a science. This point is often forgotten by scientists who prove elegant mathematical theorems about a model that doesn't adequately represent the true system to begin with. On the other hand, engineers who reject what they consider to be "ivory tower theory" are foregoing tools that could give definite answers to some questions, and hints to aid in the understanding of others.

System identification is closely tied to control theory, partially by some common methodology, and partially by the use of identified system models for control design. Before you can design a controller for a system, you must have some notion of the equations describing the system.

Another common purpose of system identification is to help gain an understanding of how a system works. Newton's investigations were more along this line. (It is unlikely that he wanted to control the motion of the planets.)

The application of system identification techniques is strongly dependent on the purpose for which the results are intended; radically different system models and identification techniques may be appropriate for different purposes related to the same system. The aircraft control system designer will be unimpressed when given a model based on inputs that cannot be influenced, outputs that cannot be measured, aspects of the system that the designer does not want to control, and a complicated model in a form not amenable to control analysis techniques. The same model might be ideal for the aerodynamicist studying the flow around the vehicle. The first and most important step of any system identification application is to define its purpose.

Following this chapter's overview, this document presents one aspect of the science of system identification—the theory of statistical estimation. The theory's main purpose is to help the engineer understand the system, not to serve as a formula for consistently producing the required results. Therefore, our exposition of the theory, although rigorously defensible, emphasizes intuitive understanding rather than mathematical sophistication. The following comments of Luenberger (1969, p. 2) also apply to the theory of system identification:

> Some readers may look with great expectation toward functional analysis, hoping to discover new powerful techniques that will enable them to solve important problems beyond the reach of simpler mathematical analysis. Such hopes are rarely realized in practice. The primary utility of functional analysis...is its role as a unifying discipline, gathering a number of apparently diverse, specialized mathematical tricks into one or a few geometric principles.

With good intuitive understanding, which arises from such unification, the reader will be better equipped to extend the ideas to other areas where the solutions, although simple, were not formerly obvious.

The literature of the field often uses the terms "system identification," "parameter identification," and "parameter estimation" interchangeably. The following sections define and differentiate these broad terms. The majority of the literature in the field, including most of this document, addresses the field most precisely called parameter estimation.

1.1 SYSTEM IDENTIFICATION

We begin by phrasing the system identification problem in formal mathematical terms. There are three elements essential to a system identification problem: a system, an experiment, and a response. We define these elements here in broad, abstract, set-theoretic terms, before introducing more concrete forms in Section 1.3.

Let U represent some experiment, taken from the set $\mathcal{U}$ of possible experiments on the system. U could represent a discrete event, such as stepping on the scales; or a value, such as a voltage applied. U could also be a vector function of time, such as the motions of the control surfaces while an airplane i flown through a maneuver. In systems terminology, U is the input to the system. (We will use the terms "input," "control," and "experiment" more or less interchangeably.)

Observe the response Z of the system to the experiment. As with U, Z could be represented in many forms including as a discrete event (e.g., "the system blew up") or as a measured time function. It is an element of the set $\mathcal{Z}$ of possible responses. (We also use the terms "output" or "measurement" for Z.)

The abstract system is a map (function) F from the set of possible experiments to the set of possible responses.

$$F: \mathcal{U} \rightarrow \mathcal{Z} \tag{1.1-1}$$

that is

$$Z = F(U) \tag{1.1-2}$$

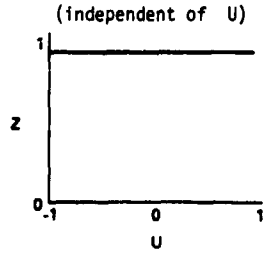
The system identification problem is to reconstruct the function F from a collection of experiments U_i and the corresponding system responses Z_i. This is the purest form of the "black box" identification problem. We are asked to identify the system with no information at all about its internal structure, as if the system were in a black box which we could not see into. Our only information is the inputs and outputs.

An obvious solution is to perform all of the experiments in $\mathcal{U}$ and simply tabulate the responses. This is usually impossible because the set $\mathcal{U}$ is too large (typically, infinite). Also, we may not have complete freedom in selecting the U_i. Furthermore, even if this approach were possible, the tabular format of the result would generally be inconvenient and of little help in understanding the structure of the system.

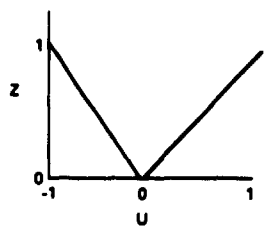
If we cannot perform all of the experiments in $\mathcal{U}$, the system identification problem is impossible without further information. Since we have made no assumptions about the form of F, we cannot be sure of its behavior without checking every point.

Example 1.1-1 The input U and output Z of a system are both represented by real-valued scalar variables. When an input of 1.0 is applied, the output is 1.0. When an input of -1.0 is applied, the output is also 1.0. Without further information we cannot tell which of the following representations (or an infinite number of others) of the system is correct.

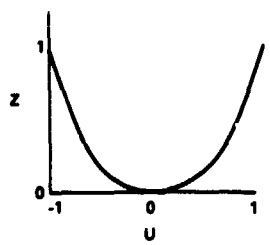
a) $Z = 1$ (independent of U)



b) $Z = |U|$



c) $Z = U^2$



d) The response depends on the time interval between applying U and measuring Z, which we forgot to consider.

Example 1.1-2 The input and output of a system are scalar time functions on the interval $(-\infty,\infty)$. When the input is $\cos(t)$, the output is $\sin(t)$. Without more information we cannot distinguish among

a) $z(t) = \cos(t)$ independent of U

b) $z(t) = \int_0^t U(S)ds$

c) $z(t) = \dot{u}(t)$

d) $z(t) = u\left(t - \frac{3}{2}\pi\right)$

Example 1.1-3 The input and output of a system are integers in the range 1-100. For every input except $U = 37$, we measure the output and find it equal to the input. We have no mathematical basis for drawing any conclusion about the response to the input $U = 37$. We could guess that the output might be $Z = 37$, but there is no mathematical justification for this guess in the problem as formulated.

Our inability to draw any conclusions in the above examples (particularly Example (1.1-3), which seems so obvious intuitively) points out the inadequacy of the pure black-box statement of the system identification problem. We cannot reconstruct the function F without some guidance on choosing a particular function from the infinite number of functions consistent with the results of the experiments performed.

We have seen that the pure black box system identification problem, where absolutely no information is given about the internal structure of the system, is impossible to solve. The information needed to construct the system function F is thus composed of two parts: information which is assumed, and information which is deduced from the experimental data. These two information sources can closely interact. For instance, the experimental data could contradict the assumptions made, requiring a revision of the assumptions, or the data could be used to select one of a set of candidate assumptions (hypotheses). Such interaction tends to obscure the role of the assumption, making it seem as though all of the information was obtained from the experimental data, and thus has a purely objective validity. In fact, this is never the case. Realistically, most of the information used for constructing the system function F will be assumptions based on knowledge of the nature of the physical processes of the system. System identification technology based on experimental data is used only to fill in the relatively small gaps in our knowledge of the system. From this perspective, we recognize system identification as an extremely useful tool for filling in such knowledge gaps, rather than as a panacea which will automatically tell us everything we need to know about a system. The capabilities of some modern techniques may invite the view of system identification as a cure-all, because the underlying assumptions are subtle and seldom explicitly stated.

Example 1.1-4 Return to the problem of example (1.1-3). Seemingly, not much knowledge of the internal behavior of the system is required to deduce that Z will be 37 when U is 37; indeed, many common system identification algorithms would make such a deduction. In fact, the assumptions made are numerous. The specification of the set of possible inputs and outputs already implies many assumptions about the system; for instance, that there are no transient effects, or that such effects are unimportant. The problem statement does not allow for an event such as the system output's oscillating through several values. We have also made an assumption of repeatability. Perhaps the same experiment redone tomorrow would produce different results, depending on some factor not considered. Encompassing all of the other assumptions is the assumption of simplicity. We have applied Occam's Razor and found the simplest system consistent with the data. One can easily imagine useful systems that select specific inputs for special treatment. Nothing in the data has eliminated such systems. We can see that the assumptions play the largest role in solving this problem. Granted the assumption that we want the simplest consistent result, the deduction from the data that $Z = U$ is trivial.

Two general types of assumptions exist. The first consists of restrictions on the allowable forms of the function F. Presumably, such restrictions would reflect the knowledge of what functions are reasonable considering the physics of the system. The second type of assumption is some criterion for selecting a "best" function from those consistent with the experimental results. In the following sections, we will see that these two approaches are combined—restricting the set of functions considered, and then selecting a best choice from this set.

1.2 PARAMETER IDENTIFICATION

For physical systems, information about the general form of the system function F can often be derived from knowledge of the system. Specific numerical values, however, are sometimes prohibitively difficult to compute theoretically without making unacceptable approximations. Therefore, the most widely used area of system identification is the subfield called parameter identification.

In parameter identification, the form of the system function is assumed to be known. This function contains a finite number of parameters, the values of which must be deduced from experimental data.

Let ξ be a vector with the unknown parameters as its elements. Then the system response Z is a known function of the input U and the parameter vector ξ. We can restate this in a more convenient, but completely equivalent way. For each value of the parameter vector ξ, the system response Z is a known function of the input U. (The function can be different for different values of ξ.) We say that the function is parameterized by ξ and write

$$Z = F_\xi(U) \tag{1.2-1}$$

The function $F_\xi(U)$ is referred to as the assumed system model. The subscript notation for F is used purely for convenience to indicate the special role of ξ. The function could be equivalently written as $F(\xi,U)$. The parameter identification problem is then to deduce the value of ξ based on measurement of the responses Z_i to a set of inputs U_i. This problem of identifying the parameter vector ξ is much less ambitious than the system identification problem of constructing the entire F function from experimental data; it is more in line with the amount of information that reasonably can be expected to be obtained from experimental data.

Deducing the value of ξ amounts to solving the following set of simultaneous and generally nonlinear equations.

$$Z_i = F_\xi(U_i) \qquad i = 1,2,\ldots N \tag{1.2-2}$$

where N is the number of experiments performed. Note that the only variable in these equations is the parameter vector ξ. The U_i and Z_i represent the specific input used and response measured for the ith experiment. This is quite different from Equation (1.2-1) which expresses a general relationship among the three variables U, Z, and ξ.

<u>Example 1.2-1</u> In the problem of example (1.1-1), assume we are given that the response is a linear function of the input

$$Z = F_\xi(U) + a_0 + a_1U$$

The parameter vector is $\xi = (a_0,a_1)^*$, the values of a_0 and a_1 being unknown. We were given that $U = -1$ and $U = +1$ both result in $Z = 1$; thus Equation (1.2-2) expands to

$$1 = F_\xi(-1) = a_0 - a_1$$

$$1 = F_\xi(1) = a_0 + a_1$$

This system is easy to solve and gives $a_0 = 1$ and $a_1 = 0$. Thus we have $F(U) = 1$ (independent of U).

<u>Example 1.2-2</u> In the problem of example (1.1-2), assume we know that the system can be represented as

$$\dot{z}(t) = az(t) + bu(t)$$

or, equivalently, expressing Z as an explicit function of U,

$$Z = F_\xi(U): \quad z(t) = \int_{-\infty}^{t} e^{a(t-\tau)}\, bu(\tau)d\tau$$

The unknown parameter vector for this system is $\xi = (a,b)^*$. Since $u(t) = \cos(t)$ resulted in $z(t) = \sin(t)$, Equation (1.2-2) becomes

$$\sin(t) = \int_{-\infty}^{t} e^{a(t-\tau)}\, b\cos(\tau)d\tau$$

for all $t\epsilon(-\infty,\infty)$. This equation is uniquely solved by $a = 0^-$ and $b = -1$.

<u>Example 1.2-3</u> In the problem of Fxample (1.1-3), assume that the system can be represented by a polynomial of order 10 or less.

$$Z = F_\xi(U) = \sum_{n=0}^{10} a_nU^n$$

The unknown parameter vector is $\xi = (a_0,a_1\ldots a_{10})^*$. Using the experimental data described in Example 1.6, Equation (1.2-2) becomes

$$i = \sum_{n=0}^{10} a_n i^n \qquad i = 1,2\ldots36,38,39\ldots100$$

This system of equations is uniquely solved by $a_0 = 0$, $a_1 = 1$, and a_2 through a_{10} all equalling 0.

As with any set of equations, there are three possible results from Equation (1.2-2). First, there can be a unique solution, as in each of the examples above. Second, there could be multiple solutions, in which case either more experiments must be performed or more assumptions would be necessary to restrict the set of allowable solutions or to pick a best solution in some sense. The third possibility is that there could be no solutions, the experimental data being inconsistent with the assumed equations. This situation will require a basic change in our way of thinking about the problem. There will almost never be an exact solution with real data, so the first two possibilities are somewhat academic. The remainder of the document, and Section 1.4 in particular, will address the general situation where Equation (1.2-2) need not have an exact solution. The possibilities of one or more solutions are part of the general case.

Example 1.2-4 In the problem of Example (1.1-1), assume we are given that the response is a quadratic function of the input

$$Z = F_\xi(\pi) = a_0 + a_1U + a_2U^2$$

The parameter vector is $\xi = (a_0,a_1,a_2)^*$. We were given that $U = -1$ and $U = +1$ both result in $Z = 1$. With these data Equation (1.2-2) expands to

$$1 = F_\xi(-1) = a_0 - a_1 + a_2$$

$$1 = F_\xi(1) = a_0 + a_1 + a_2$$

From this information we can deduce that $a_1 = 0$, but a_0 and a_2 are not uniquely determined. The values might be determined by performing the experiment $U = 0$. Alternately, we might decide that the lowest order system consistent with the data available is preferred, giving $a_2 = 0$ and $a_0 = 1$.

Example 1.2-5 In the problem of Example (1.1-1), assume that we are given that the response is a linear function of the input. We were given that $U = -1$ and $U = +1$ both result in $Z = 1$. Suppose that the experiment $U = 0$ is performed and results in $Z = 0.95$. There are then no parameter values consistent with the data.

1.3 TYPES OF SYSTEM MODELS

Although the basic concept of system modeling is quite general, more useful results can be obtained by examining specific types of system models. Clarity of exposition is also improved by using specific models, even when we can obtain the result in a more general context. This section describes some of the broad classes of system model forms which are often used in parameter identification.

1.3.1 Explicit Function

The most basic type of system model is the explicit function. The response Z is written as a known explicit function of the input U and the parameter vector ξ. This type of model corresponds exactly to Equation (1.2-1):

$$Z = F_\xi(U) \tag{1.2-1}$$

In the simplest subset of the explicit function models, the response is a linear function of the parameter vector

$$Z = f(U)\xi \tag{1.3-1}$$

In this equation, $f(U)$ is a matrix which is a known function (nonlinear in general) of the input. This is the type of model used in linear regression. Many systems can be put into this easily analyzed form, even though the systems might appear quite complex at first glance.

A common example of a model linear in its parameters is a finite polynomial expansion of Z in terms of U.

$$Z = \xi_0 + \xi_1U + \xi_2U^2 \ldots + \xi_nU^n \tag{1.3-2}$$

In this case, $f(U)$ is the row vector $(1, U, U^2 \ldots U^n)$. Note that Z is linear in the parameters ξ_i, but not in the input U.

1.3.2 State Space

State-space models are very useful for dynamic systems; that is, systems with responses that are time functions. Wiberg (1971) and Zadeh and Desoer (1963) give general discussions of state-space models. Time can be treated as either a continuous or discretized variable in dynamic models; the theories of discrete- and continuous-time systems are quite different.

The general form for a continuous-time state-space model is

$$x(t_0) = x_0 \tag{1.3-3a}$$

$$\dot{x}(t) = f[x(t),u(t),t,\xi] \tag{1.3-3b}$$

$$z(t) = g[x(t),u(t),t,\xi] \tag{1.3-3c}$$

where f and g are arbitrary known functions. The initial condition x_0 can be known or can be a function of ξ. The variable $x(t)$ is defined as the state of the system at time t. Equation (1.3-3b) is called the state equation, and (1.3-3c) is called the observation equation. The measured system response is z. The state is not considered to be measured; it is an internal system variable. However, $g[x(t),u(t),t,\xi] = x(t)$ is a legitimate observation function, the measurement can be equal to the state if so desired.

Discrete-time state space models are similar to continuous-time models, except that the differential equations are replaced by difference equations. The general form is

$$x(t_0) = x_0 \tag{1.3-4a}$$

$$x(t_{i+1}) = f[x(t_i),u(t_i),t_i,\xi] \qquad i = 0,1,\ldots \tag{1.3-4b}$$

$$z(t_i) = g[x(t_i),u(t_i),t_i,\xi] \qquad i = 1,2,\ldots \tag{1.3-4c}$$

The system variables are defined only at the discrete times t_i.

This document is largely concerned with continuous-time dynamic systems described by differential Equations (1.3-3b). The system response, however, is measured at discrete time points, and the computations are done in a digital computer. Thus, some features of both discrete- and continuous-time systems are pertinent. The system equations are

$$x(t_0) = x_0 \tag{1.3-5a}$$

$$\dot{x}(t) = f[x(t),u(t),t,\xi] \tag{1.3-5b}$$

$$z(t_i) = g[x(t_i),u(t_i),t_i,\xi] \qquad i = 1,2,\ldots \tag{1.3-5c}$$

The response $z(t_i)$ is considered to be defined only at the discrete time points t_i, although the state $x(t)$ is defined in continuous time.

We will see that the theory of parameter identification for continuous-time systems with discrete observations is virtually identical to the theory for discrete-time systems in spite of the superficial differences in the system equation forms. The theory of continuous-time observations requires much deeper mathematical background and will only be outlined in this document. Since practical application of the algorithms developed generally requires a digital computer, the continuous-time theory is of secondary importance.

An important subset of systems described by state space equations is the set of linear dynamic systems. Although the equations are sometimes rewritten in forms convenient for different applications, all linear dynamic system models can be written in the following forms: the continuous-time form is

$$x(t_0) = x_0 \tag{1.3-6a}$$

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{1.3-6b}$$

$$z(t) = Cx(t) + Du(t) \tag{1.3-6c}$$

The matrix A is called the stability matrix, B is called the control matrix, and C and D are called state and control observation matrices, respectively. The discrete-time form is

$$x(t_0) = x_0 \tag{1.3-7a}$$

$$x(t_{i+1}) = \Phi x(t_i) + \Psi u(t_i) \qquad i = 0,1,\ldots \tag{1.3-7b}$$

$$z(t_i) = Cx(t_i) + Du(t_i) \qquad i = 1,2,\ldots \tag{1.3-7c}$$

The matrices Φ and Ψ are called the system transition matrices. The form for continuous systems with discrete observations is identical to Equation (1.3-6), except that the observation is defined only at the discrete time points. In all three forms, A, B, C, D, Φ, and Ψ are matrix functions of the parameter vector ξ. These matrices are functions of time in general, but for notational simplicity, we will not explicitly indicate the time dependence unless it is important to a discussion.

The continuous-time and discrete-time state-equation forms are closely related. In many applications, the discrete-time form of Equation (1.3-7) is used as a discretized approximation to Equation (1.3-6). In this case, the transition matrices Φ and Ψ are related to the A and B matrices by the equations

$$\Phi = \exp(A\Delta) \tag{1.3-8a}$$

$$\Psi = \int_0^{\Delta} \exp(At)dt\, B \tag{1.3-8b}$$

where

$$\Delta = t_{i+1} - t_i \tag{1.3-8c}$$

We discuss this relationship in more detail in Section 7.5. In a similar manner, Equation (1.3-4) is sometimes viewed as an approximation to Equation (1.3-3). Although the principle in the nonlinear case is the same as in the linear case, we cannot write precise expressions for the relationship in such simple closed forms as in the linear case.

Standardized canonical forms of the state-space equations (Wiberg, 1971) play an important role in some approaches to parameter estimation. We will not emphasize canonical forms in this document. The basic theory of parameter identification is the same, whether canonical forms are used or not. In some applications, canonical forms are useful, or even necessary. Such forms, however, destroy any internal relationship between the model structure and the system, retaining only the external response characteristics. Fidelity to the internal as well as to the external system characteristics is a significant aid to engineering judgment and to the incorporation of known facts about the system, both of which play crucial roles in system identification. For instance, we might know the values of many locations of the A matrix in its "natural" form. When the A matrix is transformed to a canonical form, these simple facts generally become unwieldy equations which cannot reasonably be used. When there is little useful knowledge of the internal system structure, the use of canonical forms becomes more appropriate.

1.3.3 Others

Other types of system models are used in various applications. This document will not cover them explicitly, but many of the ideas and results from explicit function and state space models can be applied to other model types.

One of these alternate model classes deserves special mention because of its wide use. This is the class of auto-regressive moving average (ARMA) models and related variants (Hajdasinski, Eykhoff, Damen, and van den Boom, 1982). Discrete-time ARMA models are in the general form

$$z(t_i) + a_1 z(t_{i-1}) + \dots a_n z(t_{i-n}) = b_0 u(t_i) + b_1 u(t_{i-1}) + \dots b_m u(t_{i-m}) \tag{1.3-9}$$

Discrete-time ARMA models can be readily rewritten as linear state space models (Schweppe, 1973), so all of the theory which we will develop for state space models is directly applicable.

1.4 PARAMETER ESTIMATION

The examples in Section 1.2 were carefully chosen to have exact solutions. Real data is seldom so obliging. No matter how careful we have been in selecting the form of the assumed system model, it will not be an exact representation of the system. The experimental data will not be consistent with the assumed model form for any value of the parameter vector ξ. The model may be close, but it will not be exact, if for no other reason than that the measurements of the response will be made with real, and thus imperfect, instruments.

The theoretical development seems to have arrived at a cul-de-sac. The black box system identification problem was not feasible because there were too many solutions consistent with the data. To remove this difficulty, it was necessary to assume a model form and define the problem as parameter identification. With the assumed model, however, there are no solutions consistent with the data.

We need to retain the concept of an assumed model structure in order to reduce the scope of the problem, yet avoid the inflexibility of requiring that the model exactly reproduce the experimental data. We do this by using the assumed model structure, but acknowledging that it is imperfect. The assumed model structure should include the essential characteristics of the true system. The selection of these essential characteristics is the most significant engineering judgment in system analysis. A good example is Gauss' (1809, p. xi) justification that the major axis of a cometary ellipse is not an essential parameter, and that a simplified parabolic model is therefore appropriate:

> There existed, in point of fact, no sufficient reason why it should be taken for granted that the paths of comets are exactly parabolic: on the contrary, it must be regarded as in the highest degree improbable that nature should ever have favored such an hypothesis. Since, nevertheless, it was known, that the phenomena of a heavenly body moving in an ellipse or hyperbola, the major axis of which is very great relatively to the parameter, differs very little near the perihelion from the motion in a parabola of which the vertex is at the same distance from the focus; and that this difference becomes the more inconsiderable the greater the ratio of the axis to the parameter: and since, moreover, experience has shown that between the observed motion and the motion computed in the parabolic orbit, there remained differences scarcely ever greater than those which might safely be attributed to errors of observation (errors quite considerable in most cases): astronomers have thought proper to retain the parabola, and very properly, because there are no means whatever of ascertaining satisfactorily what, if any, are the differences from a parabola.

Chapter 11 discusses some aspects of this selection, including theoretical aids to making such judgments.

Given the assumed model structure, the primary question is how to treat imperfections in the model. We need to determine how to select the value of ξ which makes the mathematical model the "best"

representation of the essential characteristics of the system. We also need to evaluate the error in the determination of ξ due to the unmodeled effects present in the experimental data. These needs introduce several new concepts. One concept is that of a "best" representation as opposed to the correct representation. It is often impossible to define a single correct representation, even in principle, because we have acknowledged the assumed model structure to be imperfect and we have constrained ourselves to work within this structure. Thus ξ does not have a correct value. As Acton (1970) says on this subject,

> A favorite form of lunacy among aeronautical engineers produces countless attempts to decide what differential equation governs the motion of some physical object, such as a helicopter rotor....But arguments about which differential equation represents truth, together with their fitting calculations, are wasted time.

> Example 1.4-1 Estimating the radius of the Earth. The Earth is not a perfect sphere and, thus, does not have a radius. Therefore, the problem of estimating the radius of the Earth has no correct answer. Nonetheless, a representation of the Earth as a sphere is a useful simplification for many purposes.

Even the concept of the "best" representation overstates the meaning of our estimates because there is no universal criterion for defining a single best representation (thus our quotes around "best"). Many system identification methods establish an optimality criterion and use numerical optimization methods to compute the optimal estimates as defined by the criterion; indeed most of this document is devoted to such optimal estimators or approximations to them. To be avoided, however, is the common attitude that optimal (by some criterion) is synonymous with correct, and that any nonoptimal estimator is therefore wrong. Klein (1975) uses the term "adequate model" to suggest that the appropriate judgment on an identified model is whether the model is adequate for its intended purpose.

In addition to these concepts of the correct, best, or adequate values of ξ, we have the somewhat related issue of errors in the determination of ξ caused by the presence of unmodeled effects in the experimental data. Even if a correct value of ξ is defined in principle, it may not be possible to determine this value exactly from the experimental data due to contamination of the data by unmodeled effects.

We can now define the task as to determine the best estimate of ξ obtainable from the data, or perhaps an adequate estimate of ξ, rather than to determine the correct value of ξ. This revised problem is more properly called parameter estimation than parameter identification. (Both terms are often used interchangeably.) Implied subproblems of parameter estimation include the definition of the criteria for best or adequate, and the characterization of potential errors in the estimates.

> Example 1.4-2 Reconsider the problem of example (1.2-5). Although there is no linear model exactly consistent with the data, modeling the output as a constant value of 1 appears a reasonable approximation and agrees exactly with two of the three data points.

One approach to parameter estimation is to minimize the error between the model response and the actual measured response, using a least squares or some similar *ad hoc* criterion. The values of the parameter vector ξ which result in the minimum error are called the best estimates. Gauss (1809, p. 162) introduced this idea:

> Finally, as all our observations, on account of the imperfection of the instruments and of the senses, are only approximations to the truth, an orbit based only on the six absolutely necessary data may still be liable to considerable errors. In order to diminish these as much as possible, and thus to reach the greatest precision attainable, no other method will be given except to accumulate the greatest number of the most perfect observations, and to adjust the elements, not so as to satisfy this or that set of observations with absolute exactness, but so as to agree with all in the best possible manner.

This approach is easy to understand without extensive mathematical background, and it can produce excellent results. It is restricted to deterministic models so that the model response can be calculated.

An alternate approach to parameter estimation introduces probabilistic concepts in order to take advantage of the extensive theory of statistical estimation. We should note that, from Gauss's time, these two approaches have been intimately linked. The sentence immediately following the above exposition in Theoria Motus (Gauss, 1809, p. 162) is

> For which purpose, we will show in the third section how, according to the principles of the calculus of probabilities, such an agreement may be obtained, as will be, if in no one place perfect, yet in all places the strictest possible.

In the statistical approach, all of the effects not included in the deterministic system model are modeled as random noise; the characteristics of the noise and its position in the system equations vary for different applications. The probabilistic treatment solves the perplexing problem of how to examine the effect of the unmodeled portion of the system without first modeling it. The formerly unmodeled portion is modeled probabilistically, which allows description of its general characteristics such as magnitude and frequency content, without requiring a detailed model. Systems such as this, which involve both time and randomness, are referred to as stochastic systems. This document will examine a small part of the extensive theory of stochastic systems, which can be used to define estimates of the unknown parameters and to characterize the properties of these estimates.

Although this document will devote significant time to the treatment of the probabilistic approach, this approach should not be oversold. It is currently popular to disparage model-fitting approaches as nonrigorous and without theoretical basis. Such attitudes ignore two important facts: first, in many of the most common situations, the "sophisticated" probabilistic approach arrives at the same estimation algorithm as the model-fitting approaches. This fact is often obscured by the use of buzz words and unenlightening notation, apparently for fear that the theoretical effort will be considered as wasted. Our view is that such relationships should be emphasized and clearly explained. The two approaches complement each other, and the engineer who understands both is best equipped to handle real world problems. The model-fitting approach gives good intuitive understanding of such problems as modeling error, algorithm convergence, and identifiability, among others. The probabilistic approach contributes quantitative characterization of the properties of the estimates (the accuracy), and an understanding of how these characteristics are affected by various factors.

The second fact ignored by those who disparage model fitting is that the probabilistic approach involves just as many (or more) unjustified *ad hoc* assumptions. Behind the smug front of mathematical rigor and sophistication lie patently ridiculous assumptions about the system. The contaminating noise seldom has any of the characteristics (Gaussian, white, etc.) assumed simply in order to get results in a usable form. More basic is the fact that the contaminating noise is not necessarily random noise at all. It is a composite of all of the otherwise unmodeled portions of the system output, some of which might be "truly" random (deferring the philosophical question of whether truly random events exist), but some of which are certainly deterministic even at the macroscopic level. In light of this consideration, the "rigor" of the probabilistic approach is tarnished from the start, no matter how precise the inner mathematics. Contrary to the impressions often given, the probabilistic approach is not the single correct answer, but is one of the possible avenues that can give useful results, making on the average as many unjustified or blatantly false assumptions as the alternatives. Bayes (1736, p. 9), in an essay reprinted by Barnard (1958), made a classical statement on the role of assumptions in mathematics:

> It is not the business of the Mathematician to dispute whether quantities do in fact ever vary in the manner that is supposed, but only whether the notion of their doing so be intelligible; which being allowed, he has a right to take it for granted, and then see what deductions he can make from that supposition....He is not inquiring how things are in matter of fact, but supposing things to be in a certain way, what are the consequences to be deduced from them; and all that is to be demanded of him is, that his suppositions be intelligible, and his inferences just from the suppositions he makes.

The demands on the applications engineer are somewhat different, and more in line with Bayes' (1736, p. 50) later statement in the same document.

> So far as Mathematics do not tend to make men more sober and rational thinkers, wiser and better men, they are only to be considered as an amusement, which ought not to take us off from serious business.

A few words are necessary in defense of the probabilistic approach, lest the reader decide that it is not worthwhile to pursue. The main issue is the description of deterministic phenomena as random. This disagrees with common modern perceptions of the meaning and use of randomness for physical situations, in which random and deterministic phenomena are considered as quite distinct and well delineated. Our viewpoint owes more to the earlier philosophy of probability theory—that it is a useful tool for studying complicated phenomena which need not be inherently random (if anything is inherently random). Cramer (1946, p. 141) gives a classic exposition of this philosophy:

> [The following is descriptive of]...large and important groups of random experiments. Small variations in the initial state of the observed units, which cannot be detected by our instruments, may produce considerable changes in the final result. The complicated character of the laws of the observed phenomena may render exact calculation practically, if not theoretically, impossible. Uncontrollable action by small disturbing factors may lead to irregular deviations from a presumed "true value".
>
> It is, of course, clear that there is no sharp distinction between these various modes of randomness. Whether we ascribe e.g. the fluctuations observed in the results of a series of shots at a target mainly to small variations in the initial state of the projectile, to the complicated nature of the ballistic laws, or to the action of small disturbing factors, is largely a matter of taste. The essential thing is that, in all cases where one or more of these circumstances are present, an exact prediction of the results of individual experiments becomes impossible, and the irregular fluctuations characteristic of random experiments will appear.
>
> We shall now see that, in cases of this character, there appears amidst all irregularity of fluctuations a certain typical form of regularity that will serve as the basis of the mathematical theory of statistics.

The probabilistic methods allow quantitative analysis of the general behavior of these complicated phenomena, even though we are unable to model the exact behavior.

1.5 OTHER APPROACHES

Our aim in this document is to present a unified viewpoint of the system identification ideas leading to maximum-likelihood estimation of the parameters of dynamic systems, and of the application of these ideas. There are many completely different approaches to identification of dynamic systems.

There are innumerable books and papers in the system identification literature. Eykhoff (1974) and Astrom and Eykhoff (1970) give surveys of the field. However, much of the work in system identification is published outside of the general body of system identification literature. Many techniques have been developed for specific areas of application by researchers oriented more toward the application area than toward general system identification problems. These specialized techniques are part of the larger field of system identification, although they are usually not labeled as such. (Sometimes they are recognizable as special cases or applications of more general results.) In the area most familiar to us, aircraft stability and control derivatives were estimated from flight data long before such estimation was classified as a system identification problem (Doetsch, 1953; Etkin, 1958; Flack, 1959; Greenberg, 1951; Rampy and Berry, 1964; Wolowicz, 1966; and Wolowicz and Holleman, 1958).

We do not even attempt here the monumental task of surveying the large body of system identification techniques. Suffice it to say that other approaches exist, some explicitly labeled as system identification techniques, and some not so labeled. We feel that we are better equipped to make a useful contribution by presenting, in an organized and comprehensible manner, the viewpoint with which we are most familiar. This orientation does not constitute a dismissal of other viewpoints.

We have sometimes been asked to refute claims that, in some specific application, a simple technique such as regression obtained superior results to a "sophisticated" technique bearing impressive-sounding credentials as an optimal nonlinear maximum likelihood estimator. The implication is that simple is somehow synonymous with poor, and sophisticated is synonymous with good, associations that we completely disavow. Indeed, the opposite association seems more often appropriate, and we try to present the maximum likelihood estimator in a simple light. We believe that these methods are all tools to be used when they help do the job. We have used quotations from Gauss several times in this chapter to illustrate his insight into what are still some of the important issues of the day, and we will close the chapter with yet another (Gauss, 1809, p. 108):

> ...we hope, therefore, it will not be disagreeable to the reader, that, besides the solution to be given hereafter, which seems to leave nothing further to be desired, we have thought proper to preserve also the one of which we have made frequent use before the former suggested itself to me. It is always profitable to approach the more difficult problems in several ways, and not to despise the good although preferring the better.

CHAPTER 2

2.0 OPTIMIZATION METHODS

Most of the estimators in this book require the minimization or maximization of a nonlinear function. Sometimes we can write an explicit expression for the minimum or maximum point. In many cases, however, we must use an iterative numerical algorithm to find the solution. Therefore a background in optimization methods is mandatory for appreciation of the various estimators.

Optimization is a major field in its own right and we do not attempt a thorough treatment or even a survey of the field in this chapter. Our purpose is to briefly introduce a few of the optimization techniques most pertinent to parameter estimation. Several of the conclusions we draw about the relative merits of various algorithms are influenced by the general structure of parameter estimation problems and, thus, might not be supportable in a broader context of optimizing arbitrary functions. Numerous books such as Rao (1979), Luenberger (1969), Luenberger (1972), Dixon (1972), and Polak (1971) cover the detailed derivation and analysis of the techniques discussed here and others. These books give more thorough treatments of the optimization methods than we have room for here, but are not oriented specifically to parameter estimation problems. For those involved in the application of estimation theory, and particularly for those who will be writing computer programs for parameter estimation, we strongly recommend reading several of these books. The utility and efficiency of a parameter estimation program depend strongly on its optimization algorithms. The material in this chapter should be sufficient for a general understanding of the problems and the kinds of algorithms used, but not for the details of efficient application.

The basic optimization problem is to find the value of the vector x that gives the smallest or largest value of the scalar-valued function $J(x)$. By convention we will talk about minimization problems; any maximization problem can be made into an equivalent minimization problem by changing the sign of the function. We will follow the widespread practice of calling the function to be minimized a cost function, regardless of whether or not it really has anything to do with monetary cost. To formalize the definition of the problem, a function $J(x)$ is said to have a minimum at $\hat{x}$ if

$$J(\hat{x}) \le J(x) \tag{2.0-1}$$

for all x. This is sometimes called an unconstrained global minimum to distinguish it from local and constrained minima, which are defined below.

Two kinds of side constraints are sometimes placed on the problem. Equality constraints are in the form

$$g_i(x) = 0 \tag{2.0-2}$$

Inequality constraints are in the form

$$h_i(x) \le 0 \tag{2.0-3}$$

The g_i and h_i are scalar-valued functions of x. There can be any number of constraints on a problem. A value of x is called admissible if it satisfies all of the constraints; if a value violates any of the constraints it is inadmissible. The constraints modify the problem statement as follows: $\hat{x}$ is the constrained minimum of $J(x)$ if $\hat{x}$ is admissible and if Equation (2.0-1) holds for all admissible x.

Two crucial questions about any optimization problem are whether a solution exists and whether it is unique. These questions are important in application as well as in theory. A computer program can spend a long time searching for a solution that does not exist. A simple example of an optimization problem with no solution is the unconstrained minimization of $J(x) = x$. A problem can also fail to have a solution because there is no x satisfying the constraints. We will say that a problem that has no solution is ill-posed. A simple problem with a nonunique solution is the unconstrained minimization of $J(x) = (x_1 - x_2)^2$, where x is a 2-vector.

All of the algorithms that we discuss (and most other algorithms) search for a local minimum of the function, rather than the global minimum. A local minimum (also called a relative minimum) is defined as follows: $\hat{x}$ is a local minimum of $J(x)$ if a scalar $\varepsilon > 0$ exists such that

$$J(\hat{x}) \le J(\hat{x} + h) \tag{2.0-4}$$

for all h with $|h| < \varepsilon$. To define a constrained local minimum, we must add the qualifications that $\hat{x}$ and $\hat{x} + h$ satisfy the constraints. The term "extremum" refers to either a local minimum or a local maximum. Figure (2.0-1) illustrates a problem with three local minima, one of which is the global minimum.

Note that if a global minimum exists, even if it is not unique, it is also a local minimum. The converse to this statement is false; the existence of a local minimum does not even imply that a global minimum exists.

We can sometimes prove that a function has only one local minimum point, and that this point is also the global minimum. When we lack such proofs, there is no universal way to guarantee that the local minimum found by an algorithm is the global minimum. A reasonable check for iterative algorithms is to try the algorithm with many different starting values widely distributed within the realm of possible values. If the algorithm consistently converges to the same starting point, that point is probably the global minimum. The cost of such a test, however, is often prohibitively high.

The likelihood of local minima difficulties varies widely depending on the application. In some applications we can prove that there are no local minima except at the unique global minimum. At the other extreme, some applications are plagued by numerous local minima to the extent that most minimization algorithms are

worthless. Most applications lie between these extremes. We can often argue convincingly that a particular answer must be the global minimum, even when rigorous proof is impractical.

The algorithms in this chapter are, with a few exceptions, iterative. Given some starting value x_0, the algorithms give a procedure for computing a new value x_1; then x_2 is computed from x_1, etc. The intent of the iterative algorithms is to create a sequence x_i that converges to the minimum. The starting value can be from an independent estimate of a reasonable answer, or it can come from a special start-up algorithm. The final step of any iterative algorithm is testing convergence. After the algorithm has proceeded for some time, we need to choose among the following alternatives: 1) the algorithm has converged to a value sufficiently close to the true minimum and should therefore be terminated; 2) the algorithm is making acceptable progress toward the solution and should be continued; 3) the algorithm is failing to converge or is converging too slowly to obtain a solution in an acceptable time, and it should therefore be abandoned; or 4) the algorithm is exhibiting behavior that suggests that switching to a different algorithm (or modifying the current one) might be productive. This decision is far from trivial because some algorithms can essentially stall at a point far from any local minimum, making such small changes in x_i that they appear to have converged.

We have briefly mentioned the problems of existence and uniqueness of solutions, local minima, starting values, and convergence tests. These are major issues in practical application, but we will not examine them further here. The references contain considerable discussion of these issues.

A cost function of an N-dimensional x vector can be visualized as a hypersurface in (N + 1)-dimensional space. For illustrating the behavior of the various algorithms, we will use isocline plots of cost functions of two variables. An isocline is the locus of all points in the x-space corresponding to some specified cost function value. The isoclines of positive definite quadratic functions are always ellipses. Furthermore, a quadratic function is completely specified by one of its isoclines and the fact that it is quadratic. Two-dimensional examples are sufficient to illustrate most of the pertinent points of the algorithms.

We will consider unconstrained minimization problems, which illustrate the basic points necessary for our purposes. The references address problems with equality and inequality constraints.

2.1 ONE-DIMENSIONAL SEARCHES

Optimization methodology is strongly influenced by whether or not x is a scalar. Because the optimization problems in this book are generally multi-dimensional, the methods applicable only to scalar x are not directly relevant.

Many of the multi-dimensional optimization algorithms, however, require the solution of one-dimensional subproblems as part of the larger algorithm. Most such subproblems are in the form of minimizing the multi-dimensional cost function with x constrained to a line in the multi-dimensional space. This has the superficial appearance of a multi-dimensional problem, and furthermore one with the added complications of constraints. To clarify the one-dimensional nature of these subproblems, express them as follows: the vector x is restricted to a line defined by

$$x = x_0 + \lambda x_1 \tag{2.1-1}$$

where x_0 and x_1 are fixed vectors, and λ is a scalar variable representing position along the line. Restricted to this line, the cost can be written as a function of λ.

$$g(\lambda) \equiv J(x_0 + \lambda x_1) \tag{2.1-2}$$

The function $g(\lambda)$ is a scalar function of a scalar variable, and one-dimensional minimization algorithms apply directly. Substituting the minimizing value of λ into Equation (2.1-1) then gives the minimizing point along the line in the space of x.

We will not examine the one-dimensional search algorithms in this book. Several of the references have good treatments of the subject. We will note that most of the relevant one-dimensional algorithms involve approximating the function by a low-order polynomial based on the values of the function and its first and second derivatives at one or more points. The minimum point of the polynomial, explicitly evaluate replaces one of the original points, and the process repeats. The distinguishing features of the algorithms are the order of the polynomial, the number of points, and the order of the derivatives of $J(x)$ evaluated. Variants of the algorithms depend on start-up procedures and methods for selecting the point to be replaced.

In some special cases we can solve the one-dimensional minimization problems explicitly by setting the derivative to zero, or by other means, even when we cannot explicitly solve the encompassing multi-dimensional problem. Several of our examples of multi-dimensional algorithms will use explicit solutions of the one-dimensional subproblems to avoid getting bogged down in detail. Real problems seldom will be so conveniently amenable to exact solution of the one-dimensional subproblems, except where the multi-dimensional problem could be directly solved without resort to iterative methods. Iterative one-dimensional searches are usually necessary with any method that involves one-dimensional subproblems. We will encounter one of the rare exceptions in the estimation of variance.

2.2 DIRECT METHODS

Optimization methods that do not require the evaluation of derivatives of the cost function are called direct methods or zero-order methods (because they use up to zeroth order derivatives). These methods use only the cost function values.

Axial iteration, also called the univariate method or coordinate descent, is the basis for many of the direct methods. In this method we search along each of the coordinate directions of the x-space, one at a

time. Starting with the point x_0, fix the values of all but the first coordinate, reducing the problem to one-dimensional minimization. Solve this problem using any one-dimensional algorithm. Call the resulting point x_1. Then fix the first coordinate at the value so determined and do a similar search along the direction of the second coordinate, giving the point x_2. Continue these one-dimensional searches until each of the N coordinate directions has been searched; the final point of this process is x_N.

The point x_N completes the first cycle of minimization. Repeat this cycle starting from the point x_N instead of x_0. Continue repeating the minimization cycle until the process converges (or until you give up, which may well come first).

The performance of the axial iteration algorithm on most problems is unacceptably poor. The algorithm performs well only when the minimum point along each axis is nearly independent of the values of the other coordinates.

> Example 2.2-1 Use axial iteration to minimize $J(x,y) = A(x - y)^2 + B(x + y)^2$ with $A >> B$. The solution is the trivially obvious (0,0), but the problem is good for illustrating the behavior of algorithms in a simple case. Instead of using a one-dimensional search procedure, we will explicitly solve the one-dimensional subproblems. For any fixed y, obtain the minimizing x coordinate value by setting the derivative to zero
>
> $$\frac{\partial}{\partial x} J(x,y) = 2A(x - y) + 2B(x + y) = 0$$
>
> giving
>
> $$x = \frac{A - B}{A + B} y$$
>
> Similarly, for fixed x, the minimizing y value is
>
> $$y = \frac{A - B}{A + B} x$$

We see that for $A >> B$, the values of x and y descend slowly toward the true minimum at (0,0). Figure (2.2-1) illustrates this behavior on an isocline plot. Note that if $A = B$ (the cost function isocline is circular) the exact minimum is obtained in one cycle, but as A/B increases the performance worsens.

Several modifications to the basic axial iteration method are available to improve its performance. Some of these modifications exploit the notion of the pattern direction, the direction from the beginning point $X_{i \times N}$ of a cycle to the end point $x_{(i+1)\times N}$ of the same cycle. Figure (2.2-2) illustrates the pattern direction, which tends to point in the general direction of the minimum. Powell's method is the most powerful of the direct methods that search along pattern directions. See the references for details.

2.3 GRADIENT METHODS

Optimization methods that use the first derivative (gradient) of the cost function are called gradient methods or first order methods. Gradient methods require that the cost function be differentiable; most of the cost functions considered in this book meet this requirement. The gradient methods generally converge in fewer iterations than many of the direct methods because the gradient methods use more information in each iteration. (There are exceptions, particularly when comparing simple-minded gradient methods with the most powerful of the direct methods). The penalty paid for the generally improved performance of the gradient methods compared with the direct methods is the requirement to evaluate the gradient.

We define the gradient of the function $J(x)$ with respect to x to be the row vector. (Some texts define it as a column vector; the difference is inconsequential as long as one is consistent.)

$$\nabla_x J(x) \equiv \left[\frac{\partial}{\partial x_1} \frac{\partial}{\partial x_2} \cdots \frac{\partial}{\partial x_N}\right] J(x) \tag{2.3-1}$$

A reasonable estimate of the computational cost of evaluating the gradient is N times the cost of evaluating the function. This estimate follows from the fact that the gradient can be approximately evaluated by N finite differences

$$\frac{\partial J(x)}{\partial x_i} \approx \frac{[J(x + \varepsilon e_i) - J(x)]}{\varepsilon} \tag{2.3-2}$$

where e_i is the unit vector along the x_i axis and ε is a small number. In special cases, there can be expressions for the gradient that cost significantly less than N function evaluations.

Equation (2.3-2) somewhat obscures the distinction between the gradient methods and the direct methods. We can rewrite any gradient method in a finite difference form that does not explicitly involve gradients. There is, nonetheless, a fairly clear distinction between methods derived from gradient ideas and methods derived from direct search ideas. We will retain this philosophical distinction regardless of whether the gradients are evaluated explicitly or by finite differences.

The method of steepest descent (also called the gradient method) involves a series of one-dimensional searches, as did the axial-iteration method and its variants. In the steepest-descent method, these searches

are along the direction of the negative of the gradient vector, evaluated at the current point. The one-dimensional problem is to find the value of λ that minimizes

$$J_i(\lambda) \equiv J(x_i + \lambda s_i) \tag{2.3-3}$$

where s_i is the search direction given by

$$s_i = -\nabla_x^* J(x)\big|_{x=x_i} \tag{2.3-4}$$

The negative of the gradient is the direction of steepest local descent of the cost function (thus the name of the method). To prove this property, first note that for any vector s we have

$$\frac{d}{d\lambda} J(x + \lambda s) = \langle s, \nabla_x^* J(x)\rangle \tag{2.3-5}$$

We are using the $\langle .,. \rangle$ notation for the inner product

$$\langle x,y\rangle \equiv x^* y \tag{2.3-6}$$

Equation (2.3-5) is a generalization of the definition of the gradient; it applies in spaces where Equation (2.3-1) is not meaningful. We then need only show that, if s is restricted to be a unit vector, Equation (2.3-5) is minimized by choosing s in the direction of $-\nabla_x^* J(x)$. This follows immediately from the Cauchy-Schwartz inequality (Luenberger, 1969) of linear algebra.

<u>Theorem 2.3-1</u> (Cauchy-Schwartz) $\langle x,y\rangle^2 \le |x|^2|y|^2$ with equality if and only if $x = ay$ for some scalar a.

<u>Proof</u> The theorem is trivial if $y = 0$. For $y \neq 0$ examine

$$\langle x + \lambda y, x + \lambda y\rangle = \langle x,x\rangle + \lambda^2\langle y,y\rangle + 2\lambda\langle x,y\rangle \ge 0 \tag{2.3-7}$$

Choose

$$\lambda = -\langle x,y\rangle/\langle y,y\rangle \tag{2.3-8}$$

Substitute into Equation (2.3-7) and rearrange to give

$$\langle x,y\rangle^2 \le \langle x,x\rangle\langle y,y\rangle = |x|^2|y|^2 \tag{2.3-9}$$

Equality holds if and only if $x + \lambda y = 0$ in Equation (2.3-7), which will be true if and only if $x = ay$ (λ will then be $-a$).

On the surface, the steepest descent property of the method seems to imply excellent performance in minimizing the cost function value. The direction of steepest descent, however, is a local property which might point far from the direction of the global minimum. It is thus often a poor choice of search direction. Direct methods such as Powell's often converge more rapidly than steepest descent.

The steepest descent method performs worst in long narrow valleys of the cost function. It is also sensitive to scaling. These two difficulties are closely related; rescaling a problem can easily create long narrow valleys. The following examples illustrate the scaling and valley difficulties:

<u>Example 2.3-1</u> Let the cost function be

$$J(x) = \frac{1}{2}(x_1^2 + x_2^2)$$

The steepest descent method works excellently for this cost function (so does almost every optimization method). The gradient of $J(x)$ is

$$\nabla_x J(x) = (x_1, x_2) = x^*$$

Therefore, from any starting point, the negative of the gradient points exactly at the origin, which is the global minimum. The minimum will be attained exactly (or to the accuracy of the one-dimensional search methods used) in one iteration. Figure (2.3-1) illustrates the algorithm starting from the point $(1,1)^*$.

<u>Example 2.3-2</u> Rescale the preceding example by replacing x_1 by $0.1x_1$. (Perhaps we just redefined the units of x_1 to be millimeters instead of centimeters.) The cost function is then

$$J(x) = \frac{1}{2}(0.01x_1^2 + x_2^2)$$

and the gradient is

$$\nabla_x J(x) = (0.01x_1, x_2)$$

Figure (2.3-2) shows the search direction used by the algorithm starting from the point $(10,1)^*$, which corresponds to the point $(1,1)^*$ in the previous

example. The search direction points almost 90° from the origin. A careless glance at Figure (2.3-2) invites the conclusion that the minimum in the search direction will be on the x axis and thus that the second iteration of the steepest descent algorithm will attain the minimum. It is true that the minimum is close to the x axis, but it is not exactly on the axis; the distinction makes an important difference in the algorithm's performance.

For points $x - \lambda\nabla_x^* J(x)$ along the search direction from any point $(x_1,x_2)^*$, the cost function is

$$g(\lambda) = f(x - \lambda\nabla_x^* J(x)) = \frac{1}{2}\left[0.01x_1^2(1 - 0.01\lambda)^2 + x_2^2(1 - \lambda)^2\right]$$

The minimum of $g(\lambda)$ is at

$$\hat{\lambda} = \frac{(0.01)^2x_1^2 + x_2^2}{(0.01)^3x_1^2 + x_2^2}$$

and thus the minimum point along the search direction is

$$(x_1 - 0.01x_1\hat{\lambda}\ ,\ \ x_2 - x_2\hat{\lambda})^*$$

with $\hat{\lambda}$ defined as above. The following table and Figure (2.3-3) show several iterations of this process starting from the point $(10,1)^*$.

Iteration	x_1	x_2
0	10	1
1	9.899	-.009899
2	4.900	.4900
3	4.851	-.004851
4	2.401	.2401
5	2.377	-.002377
6	1.176	.1176
7	1.165	-.001165

The trend of the algorithm is clear; every two iterations it moves essentially halfway to the solution. Consider the behavior starting from the point $(10,0.1)^*$ instead of $(10,1)^*$:

Iteration	x_1	x_2
0	10	0.1
1	9.802	-.09802
2	9.608	.09608
3	9.418	-.09418
4	9.231	.09231
5	9.048	-.09048
6	8.869	.08869
7	8.694	-.08694

This behavior, plotted in Figure (2.3-4), is abysmal. The algorithm is bouncing back and forth across the valley, making little progress toward the minimum.

Several modifications to the steepest descent method are available to improve its performance. A rescaling step to eliminate valleys caused by scaling yields major improvements for some problems. The method of parallel tangents (PARTAN method) exploits pattern directions similar to those discussed in Section 2.2; searches in such pattern directions are often called acceleration steps. The conjugate gradient method is the most powerful of the modifications to steepest descent. The references discuss these and other gradient algorithms in detail.

2.4 SECOND ORDER METHODS

Optimization methods that use the second derivative (or an approximation to it) of the cost function are called second order methods. These methods require that the first and second derivatives of the cost function exist.

2.4.1 Newton-Raphson

The Newton-Raphson optimization algorithm (also called Newton's method) is the basis for all of the second order methods. The idea of this algorithm is to approximate the cost function by the first three terms of its Taylor series expansion about the current point.

$$J_i(x) \equiv J(x_i) + (x - x_i)^* \nabla_x^* J(x_i) + \frac{1}{2}(x - x_i)^*[\nabla_x^2 J(x_i)](x - x_i) \tag{2.4-1}$$

From a geometric viewpoint, this equation describes the paraboloid that best approximates the function near x_i. Equating the gradient of $J_i(x)$ to zero gives an equation for the minimum point of the approximating function. Taking this gradient, note that $\nabla_x J(x_i)$ and $\nabla_x^2 J(x_i)$ are evaluated at the fixed point x_i and thus are not functions of x.

$$\nabla_x J_i(x) = \nabla_x J(x_i) + (x - x_i)^*[\nabla_x^2 J(x_i)] = 0 \tag{2.4-2}$$

The solution is

$$x = x_i - [\nabla_x^2 J(x_i)]^{-1} \nabla_x^* J(x_i) \tag{2.4-3}$$

If the second gradient of J is positive definite, then Equation (2.4-3) gives the exact unique minimum of the approximating function; it is a reasonable guess at an approximate minimum of the original function. If the second gradient is not positive definite, then the approximating function does not have a unique minimum and the algorithm is likely to perform poorly. The Newton-Raphson algorithm uses Equation (2.4-3) iteratively; the x from this equation is the starting point for the next iteration. The algorithm is

$$x_{x+1} = x_i - [\nabla_x^2 J(x_i)]^{-1} \nabla_x^* J(x_i) \tag{2.4-4}$$

The performance of this algorithm in the close neighborhood of a strict local minimum is unexcelled; this performance represents an ideal toward which other algorithms strive. The Newton-Raphson algorithm attains the exact (except for numerical round-off errors) minimum of any positive-definite quadratic function in a single iteration. Convergence within 5 to 10 iterations is common on some practical nonquadratic problems with several dozen dimensions; direct and gradient methods typically count iterations in hundreds and thousands for such problems and settle for less accurate answers. See the references for analysis of convergence characteristics.

Three negative features of the Newton-Raphson algorithm balance its excellent convergence near the minimum. First is the behavior of the algorithm far from the minimum. If the initial estimate is far from the minimum, the algorithm often converges erratically or even diverges. Such problems are often associated with second gradient matrices that are not positive definite. Because of this problem, it is common to use special start-up procedures to get within the area where Newton-Raphson performs well. One such procedure is to start with a gradient method, switching to Newton-Raphson near the minimum. There are many other start-up procedures, and they play a key role in successful applications of the Newton-Raphson algorithm.

The second negative feature of the Newton-Raphson method is the computational cost and complexity of evaluating the second gradient matrix. The magnitude of this difficulty varies widely among applications. In some special cases the second gradient is little harder to compute than the first gradient; Newton-Raphson, perhaps with a start-up procedure, is a good choice for such applications. If, at the other extreme, you are reduced to finite-difference computation of the second gradient, Davidon-Fletcher-Powell (Section 2.4.4) is probably a more appropriate algorithm. In evaluating the computational burden of Newton-Raphson and other methods, remember that Newton-Raphson requires no one-dimensional searches. Equation (2.4-4) constitutes the entire algorithm. The one-dimensional searches required by most other algorithms can account for a majority of their computational cost.

The third negative feature of the Newton-Raphson algorithm is the necessity to invert the second gradient matrix (or at least to solve the set of linear equations involving the matrix). The computer time required for the inversion is seldom an issue; this time is typically small compared to the time required to evaluate the second gradient. Furthermore, the algorithm converges quickly enough that if one linear system solution per iteration is a large fraction of the total cost, then the total cost must be low, even if the linear system is on the order of 100-by-100. The crucial issue concerning the inversion of the second gradient is the possibility that the matrix could be singular or ill-conditioned. We will discuss singularities in Section 2.4.3.

2.4.2 Invariance

The Newton-Raphson algorithm has far less difficulty with long narrow valleys of the cost function than does the steepest-descent method. This difference is related to an invariance property of the Newton-Raphson algorithm. Invariance of minimization algorithms is a useful concept which many texts mention briefly, if at all. We will therefore elaborate somewhat on the subject.

The examples in the section on steepest descent illustrate a strong link between scaling and narrow valleys. Scaling changes can easily create such valleys. Therefore we can generally state that minimization methods that are sensitive to scaling changes are likely to behave poorly in narrow valleys.

This reasoning suggests a simple criterion for evaluating optimization algorithms: a good optimization algorithm should be invariant under scaling changes. This principle is almost so self-evident as to be unworthy of mention. The user of a program would be justifiably disgruntled if an algorithm that worked in the English Gravitational System (Imperial System) of units failed when applied to the same problem expressed in metric units (or vice versa). Someone trying to duplicate reported results would be perplexed by data published in metric units which could be duplicated only by converting to English Gravitational System units, in which the computation was really done. Nonetheless, many common algorithms, including the steepest descent method, fail to exhibit invariance under scaling.

The criterion is neither necessary nor sufficient. It is easy to construct ridiculous algorithms that are invariant to scale changes (such as the algorithm that always returns the value zero), and scale-sensitive algorithms like the steepest descent method have achieved excellent results in some applications. It is safe to

state, however, that you can usually improve a good scale-sensitive algorithm by making it scale-invariant. An initial step that rescales the problem can effectively make the steepest-descent method scale-invariant (although such a step destroys a different invariance property of the steepest-descent method: invariance under rotation of coordinates). Rescaling a problem can be done manually by the user, or it can be an automatic part of an algorithm; automatic rescaling has the obvious advantage of being easier for the user, and a secondary advantage of allowing dynamic scaling changes as the algorithm proceeds.

We can extend the idea of invariance beyond scale changes. In general, we would like an algorithm to be invariant under the largest possible set of transformations. A justification for this criterion is that almost any complicated minimization problem can be expressed as some transformation (possibly quite complicated) of a simpler problem. We can sometimes use such transformations to simplify the solution of the original problems. Often it is more difficult to do the transformation than to solve the original optimization problem. Even if we cannot do the transformations, we can use the concept to conclude that an optimization algorithm invariant over a large class of transformations is likely to work on a large class of problems.

The Newton-Raphson algorithm is invariant under all invertible linear transformations. This is the widest invariance property that we can usually achieve.

The scale-invariance of the Newton-Raphson algorithm can be partially nullified by poor choice of matrix inversion (or linear system solution) algorithms. We have assumed exact arithmetic in the preceding discussion of scale-invariance. Some matrix inversion routines are sensitive to scaling effects. Inversion based on Cholesky factorization (Wilkinson, 1965, and Acton, 1970) is a good, easily implemented method for [illegible] matrices (the second gradient is always symmetric), and is insensitive to scaling. Alternatively, [illegible] prescale the matrix by using its diagonal elements.

2.4.3 Singularities

The second gradient matrix used in the Newton-Raphson algorithm is positive definite in a region near a strict local minimum. Ideally, the start-up procedure will reach such a region, and the Newton-Raphson algorithm will then converge without needing to contend with singularities. This viewpoint is overly optimistic; singular or ill-conditioned matrices (the difference is largely academic) arise in many situations. In the following discussion, we discount the effects of scaling. Matrices that have large condition numbers because of scaling do not represent intrinsically ill-conditioned problems, and do not require the techniques discussed in this section.

In some situations, the second gradient matrix is exactly singular for all values of x; two columns (and rows) are identical or a column (and corresponding row) is zero. These simple singularities occur regularly even in complex nonlinear problems. They often result from errors in the problem formulation, such as minimizing with respect to a parameter that is irrelevant to the cost function.

In the more general case, the second gradient is singular (or ill-conditioned) at some points but not at others. Whenever we use the term singular in the following discussion, we implicitly mean singular or ill-conditioned. Because of this definition, there will be vaguely defined regions of singularity rather than isolated points. The consequences of singularities are different depending on whether or not they are near the minimum.

Singularities far from the minimum pose no basic theoretical difficulties. There are several practical methods for handling such singularities. One method is to use a gradient algorithm (or any other algorithm unaffected by such singularities) until x is out of the region of singularity. We can also use this method if the second gradient matrix has negative eigenvalues, whether the matrix is ill-conditioned or not. If the matrix has negative eigenvalues, the Newton-Raphson algorithm is likely to behave poorly. (It could even converge to a local maximum.) The second gradient is always positive semi-definite in a region around a local minimum, so negative eigenvalues are only a consideration away from the minimum.

Another method of handling singularities is to add a small positive definite matrix to the second gradient before inversion. We can also use this method to handle negative eigenvalues if the added matrix is large enough. This method is closely related to the previous suggestion of using a gradient algorithm. If the added matrix is a large constant times an identity matrix, the Newton-Raphson algorithm, so modified, gives a small step in the negative gradient direction. For small constants, the algorithm has characteristics between those of steepest descent and Newton-Raphson. The computational cost of this method is high; in essence, we are getting performance like steepest descent while paying the computational cost of Newton-Raphson. Even small additions to the second derivative matrix can dramatically change the convergence behavior of the Newton-Raphson algorithm. We should therefore discontinue this modification when out of the region of singularity. The advantage of this method is its simplicity; excluding the test on when the matrix is ill-conditioned, this modification can be done in two short lines of FORTRAN code.

The last method is to use a pseudo-inverse (rank-deficient solution). Penrose (1955), Aoki (1967), Luenberger (1969), Wilkinson and Reinsch (1971), Moler and Stewart (1973), and Garbow, Boyle, Dongarra, and Moler (1977) discuss pseudo-inverses in detail. The basic idea of the pseudo-inverse method is to ignore the directions in the x-space corresponding to zero eigenvalues (within some tolerance) of the second gradient. In the parameter estimation context, such directions represent parameters, or combinations of parameters, about which the data give little information. Lacking any information to the contrary, the method leaves such parameter combinations unchanged from their initial values.

The pseudo-inverse method does not address the problem of negative eigenvalues, but it is popular in a large class of applications where negative eigenvalues are impossible. The method is easy to implement, being only a rewrite of the matrix-inversion or linear-system-solution subroutine. It also has a useful property absent from the other proposed methods; it does not affect the Newton-Raphson algorithm when the matrix is well-conditioned. Therefore one can freely apply this method without testing whether it is needed. (It is true that condition tests in some form are part of a pseudo-inverse algorithm, but such tests are at a lower level contained within the pseudo-inverse subroutine.)

Singularities near the minimum require special consideration. The excellent convergence of Newton-Raphson near the minimum is the primary reason for using the algorithm. If it significantly slow the convergence near the minimum, there is little argument for using Newton-Raphson. The use of a pseudo-inverse can handle singularities while maintaining the excellent convergence; the pseudo-inverse is thus an appropriate tool for this purpose.

Although pseudo-inverses handle the computational problems, singularities near the minimum also raise theoretical and application issues. Such a singularity indicates that the minimum point is poorly defined. The cost function is essentially flat in at least one direction from the minimum, and the minimum value of the cost function might be attained to machine accuracy by widely separated points. Although the algorithm converges to a minimum point, it might be the wrong minimum point if the minimum is flat. If the only goal is to minimize the cost function, any minimizing point might be acceptable. In the applications of this book, minimizing the cost function is only a means to an end; the desired output is the value of x. If multiple solutions exist, the problem statement is incomplete or faulty.

We strongly advise avoiding the routine use of pseudo-inverses or other computational machinations to "solve" uniqueness problems. If the basic problem statement is faulty, no numerical trick will solve it. The pseudo-inverse works by changing the problem statement of the inversion, adding the stipulation that the inverse have minimum norm. The interpretation of this stipulation is vague in the context of the optimization problem (unless the cost function is quadratic, in which case it specifies the solution nearest the starting point). If this stipulation is a reasonable addition to the problem statement, then the pseudo-inverse is an appropriate tool. This decision can have significant effects. For a nonquadratic cost function, for example, there might be large differences in the solution point, depending on small changes in the starting point, the data, or the algorithm.

The pseudo-inverse can be a good diagnostic tool for getting the information needed to revise the problem statement, but one should not depend upon it to solve the problem autonomously. The analyst's strong point is in formulating the problem; the computer's strength is in crunching numbers to arrive at the solution. A failure in either role will compromise the validity of the solution. This statement is but a rephrasing of the computer cliche "garbage in, garbage out," which has been said many more times than it has been heard

2.4.4 Quasi-Newton Methods

Quasi-Newton methods are intended for problems where explicit evaluation of the second gradient of the cost function is complicated or costly, but the performance of the Newton-Raphson algorithm is desired. These methods form approximations to the second-gradient matrix using the first-gradient values from several iterations. The approximation to the second gradient then substitutes for the exact second gradient in Equation (2.4-4). Some of the methods directly form approximations of the inverse of the second-gradient matrix, avoiding the cost and some of the problems of matrix inversion.

Note that as long as the approximation to the second-gradient matrix is positive definite, Equation (2.4-4) can never converge to any point with a nonzero first gradient. Therefore approximations to the second gradient, no matter how poor, cannot affect the solution point. The approximations can greatly change the speed of convergence and the area of acceptable starting values. Approximations to the first gradient would affect the solution point as well.

The steepest descent method can be considered as the crudest of the quasi-Newton methods, using a constant times the identity matrix as the approximation to the second gradient. The performance of the quasi-Newton methods approaches that of Newton-Raphson as the approximation to the second gradient improves. The Davidon-Fletcher-Powell method (variable metric method) is the most popular quasi-Newton method. See the references for discussions of these methods.

2.5 SUMS OF SQUARES

The algorithms discussed in the previous sections are generally applicable to any minimization problem. By tailoring algorithms to special characteristics of specific problem classes, we can often achieve far better performance than by using the general purpose algorithms.

Many of the cost functions arising in estimation problems have the form of sums of squares. The general sums-of-squares form is

$$J(x) = \sum_{i=1}^{N} [f_i(x)]^* W_i [f_i(x)] \qquad (2.5\text{-}1)$$

The f_i are vector-valued functions of x, and the W_i are weightings. To simplify some of the formulae, we assume that the W_i are symmetric. This assumption does not really restrict the application because we can always substitute $1/2(W_i + W_i^*)$ for a nonsymmetric W_i without changing the function values. In most applications, the W_i are positive semi-definite; this is not a requirement, but we will see that it helps ensure that the stationary points encountered are local minima. The form of Equation (2.5-1) is common enough to merit special study.

The summation sign in Equation (2.5-2) is somewhat superfluous in that any function in the form of Equation (2.5-1) can be rewritten in an equivalent form without the summation sign. This can be done by concatenating the N different $f_i(x)$ vectors into a single, longer $f(x)$ vector and making a corresponding large W matrix with the W_i matrices on diagonal blocks. The only difference is in the notation. We choose the longer notation with the summation sign because it more directly corresponds with the way many parameter estimation problems are naturally phrased.

Several of the algorithms discussed in the previous two sections work well with the form of Equation (2.5-1). For any reasonable f_i functions, Equation (2.5-1) defines a cost function that is well-approximated by quadratics over fairly large regions. Since many of the general minimization schemes are based on quadratic approximations, application of these schemes to Equation (2.5-1) is natural. This statement does not imply that there are never problems minimizing Equation (2.5-1); the problems are sometimes severe, but the odds of success with reasonable effort are much better than they are for arbitrary cost function forms. Although the general methods are usable, we can exploit the problem structure to do better.

2.5.1 Linear Case

If the f_i functions in Equation (2.5-1) are linear, then the cost function is exactly quadratic and we can express the minimum point in closed form. In particular, let the f_i be the arbitrary linear functions

$$f_i(x) \equiv A_i x + b_i \tag{2.5-2}$$

Equation (2.5-1) then becomes

$$J(x) = \sum_{i=1}^{N} [A_i x + b_i]^* W_i [A_i x + b_i] \tag{2.5-3}$$

Equating the gradient of Equation (2.5-3) to zero gives

$$2 \sum_{i=1}^{N} [A_i x + b_i]^* W_i A_i = 0 \tag{2.5-4}$$

Solving for x gives

$$x = -\left[\sum_{i=1}^{N} A_i^* W_i A_i\right]^{-1} \left[\sum_{i=1}^{N} A_i^* W_i b_i\right] \tag{2.5-5}$$

assuming that the inverse exists.

If the inverse exists, then Equation (2.5-5) gives the only stationary point of Equation (2.5-3). This stationary point must be a minimum if all the W_i are positive semi-definite, and it must be a maximum if all the W_i are negative semi-definite. (We leave the straightforward proofs as an exercise.) If the W_i meet neither of these conditions, the stationary point can be a minimum, a maximum, or a saddle point.

If the inverse in Equation (2.5-5) does not exist, then there is a line (at least) of solutions to Equation (2.5-4). All of these points are stationary points of the cost function. Use of a pseudo-inverse will produce the solution with minimum norm, but this is usually a poor idea (see Section 2.4.3).

2.5.2 Nonlinear Case

If the f_i are nonlinear, there is no simple, closed-form solution like Equation (2.5-5). A natural question in such situations, in which there is an easy method to handle linear equations, is whether we can merely linearize the nonlinear equations and use the linear methodology. Such linearization does not give an acceptable closed-form solution to the current problem, but it does form the basis for an iterative method.

Define the linearization of f_i about any point x_j as

$$f_i^{(j)}(x) \equiv A_i^{(j)} x + b_i^{(j)} \tag{2.5-6}$$

where

$$A_i^{(j)} \equiv \nabla_x f_i(x_j) \tag{2.5-7a}$$

$$b_i^{(j)} \equiv f_i(x_j) - A_i^{(j)} x_j \tag{2.5-7b}$$

Equation (2.5-5), with the $A_i^{(j)}$ and $b_i^{(j)}$ substituted for A_i and b_i, gives the stationary point of the cost with the linearized f_i functions. This point is not, in general, a solution to the nonlinear problem. If, however, x_j is close to the solution, then Equation (2.5-5) should give a point closer to the solution, because the linearization will give a good representation of the cost function in the region around x_j.

The iterative algorithm resulting from this concept is as follows: First, choose a starting value x_0. The closer x_0 is to the correct solution, the better the algorithm is likely to work. Then define revised x_j values by

$$x_{j+1} = x_j - \left\{\sum_{i=1}^{N} [\nabla_x f_i \quad]^* W_i [\nabla_x f_i(x_j)]\right\}^{-1} \left\{\sum_{i=1}^{N} [\nabla_x f_i(x_j)]^* W_i f_i(x_j)\right\} \tag{2.5-8}$$

This equation comes from substituting Equation (2.5-7) into Equation (2.5-5) and simplifying. Iterate Equation (2.5-8) until it converges by some criterion, or until you give up. This method is often called quasi-linearization because it is based on linearization not of the cost function itself, but of factors in the cost function.

We made several vague, unsupported statements in the process of deriving this algorithm. We now need to analyze the algorithm's performance and compare it with the performance of the algorithms discussed in the previous sections. This task is greatly simplified by noting that Equation (2.5-8) defines a quasi-Newton algorithm. To show this, we can write the first and second gradients of Equation (2.5-1):

$$\nabla_x J(x) = 2 \sum_{i=1}^{N} [f_i(x)]^* W_i \nabla_x f_i(x) \tag{2.5-9}$$

$$\nabla_x^2 J(x) = 2 \sum_{i=1}^{N} [\nabla_x f_i(x)]^* W_i [\nabla_x f_i(x)] + 2 \sum_{i=1}^{N} [f_i(x)]^* W_i \nabla_x^2 f_i(x) \tag{2.5-10}$$

(We have not previously introduced the definition of the second gradient of a vector, as in the $\nabla_x^2 f_i(x)$ above. The result is technically a tensor, but we will not need to consider it in detail here.) Comparing Equation (2.5-8) with Equations (2.4-4), (2.5-9), and (2.5-10), we see that the only difference between quasi-linearization and Newton-Raphson is that quasi-linearization has dropped the second term in Equation (2.5-10). Quasi-linearization is thus a quasi-Newton method using

$$\nabla_x^2 J(x) \approx 2 \sum_{i=1}^{N} [\nabla_x f_i(x)]^* W_i [\nabla_x f_i(x)] \tag{2.5-11}$$

as an approximation for the second gradient. The algorithm in this form is also known as Gauss-Newton, the term we will adopt in this book.

Near the solution, the neglected term of the second gradient is generally small. Section 5.4.3 outlines this argument as it applies to the parameter estimation problem. Therefore, Gauss-Newton approaches the excellent performance of Newton-Raphson near the solution. Such approximation is the main goal of quasi-Newton methods.

Accurately approximating the performance of Newton-Raphson far from the minimum is not of great concern because Newton-Raphson does not generally perform well in regions far from the minimum. We can even argue that Gauss-Newton sometimes performs better than Newton-Raphson far from the minimum. The worst problems with Newton-Raphson occur when the second gradient matrix has negative eigenvalues; Newton-Raphson can then go in the wrong direction, possibly converging to a local maximum or diverging. If all of the W_i are positive semi-definite (which is usually the case), then the second gradient approximation given by Equation (2.5-11) is positive semi-definite for all x. A positive semi-definite second gradient approximation does not guarantee good behavior, but it surely helps; negative eigenvalues virtually guarantee problems. Thus we can heuristically argue that Gauss-Newton should perform better than Newton-Raphson. We will not attempt a detailed support of this general argument in this book. In several specific cases the improvement of Gauss-Newton over Newton-Raphson is easily demonstrable.

Although Gauss-Newton sometimes performs better than Newton-Raphson far from the solution, it has many of the same basic start-up problems. Both algorithms exhibit their best performance near the minimum. Therefore, we will often need to begin with some other, more stable algorithm, changing to Gauss-Newton as we near the minimum.

The real argument in favor of Gauss-Newton over Newton-Raphson is the lower computational effort and complexity of Gauss-Newton. Any performance improvement is a coincidental side benefit. Equation (2.5-11) involves only first derivatives of $f_i(x)$. These first derivatives are also used in Equation (2.5-9) for the first gradient of the cost. Therefore, after computing the first gradient of J, the only significant computation remaining for the Gauss-Newton approximation is the matrix multiplication in Equation (2.5-11). The computation of the Gauss-Newton approximation for the second gradient can sometimes take less time than the computation of the first gradient, depending on the system dimensions. For complicated f_i functions, evaluation of the $\nabla_x^2 f_i(x)$ in Equation (2.5-10) is a major portion of the computation effort of the full Newton-Raphson algorithm. Gauss-Newton avoids this extra effort, obtaining the performance per iteration of Newton-Raphson (if not better in some areas) with computational effort per iteration comparable to gradient methods.

Considering the cost of the one-dimensional searches required by gradient methods, Gauss-Newton can even be cheaper per iteration than gradient methods. The exact trade-off depends on the relative costs of evaluating the f_i and their gradients, and on the typical number of evaluations required in the one-dimensional searches. Gauss-Newton is at its best when the cost of evaluating the f_i is nearly as much as the cost of evaluating both the f_i and their gradients due to high overhead costs common to both evaluations. This is exactly the case in some aircraft applications, where the overhead consists largely of dimensionalizing the derivatives and building new system matrices at each time point.

The other quasi-Newton methods, such as Davidon-Fletcher-Powell, also approach Newton-Raphson performance without evaluating the second derivatives of the f_i. These methods, however, do require one-dimensional searches. Gauss-Newton stands almost alone in avoiding both second derivative evaluations and one-dimensional searches. This performance is difficult to match in general algorithms that do not take advantage of the special structure of the cost function.

Some analysts (Foster, 1983) introduce one-dimensional line searches into the Gauss-Newton algorithm to improve its performance. The utility of this idea depends on how well the Gauss-Newton method is performing. In most of our experience, Gauss-Newton works well enough that the one-dimensional line searches cannot measurably improve performance; the total computation time can well be larger with the line searches. When the Gauss-Newton algorithm is performing poorly, however, such line searches could help stabilize it.

For cost functions in the form of Equation (2.5-1), the cost/performance ratio of Gauss-Newton is so much better than that of most other algorithms that Gauss-Newton is the clearly preferred algorithm. You may want to modify Gauss-Newton for specific problems, and you will almost surely need to use some special start-up algorithm, but the best methods will be based on Gauss-Newton.

2.6 CONVERGENCE IMPROVEMENT

Second-order methods tend to converge quite rapidly in regions where they work well. There is usually such a region around the minimum point; the size of the region is problem-dependent. The price paid for this region of excellent convergence is that the second-order methods often converge poorly or diverge in regions far from the minimum. Techniques to detect and remedy such convergence problems are an important part of the practical implementation of second-order methods. In this section, we briefly list a few of the many convergence improvement techniques.

Modifications to improve the behavior of second-order methods in regions far from the minimum almost inevitably slow the convergence in the region near the minimum. This reflects a natural trade-off between speed and reliability of convergence. Therefore, effective implementation of convergence-improvement techniques usually includes different treatment of regions far from the minimum and near the minimum.

In regions far from the minimum, the second-order methods are modified or abandoned in favor of more conservative algorithms. In regions near the minimum, there is a transition to the fast second order methods. The means of determining when to make such transitions vary widely. Transitions can be based on a simple iteration count, on adaptive criteria which examine the observed convergence behavior, or on other principles. Transitions can be either gradual or step changes.

Some convergence improvement techniques abandon second-order methods in the regions far from the minimum, adopting gradient methods instead. In our experience, the pure gradient method is too slow for practical use on most parameter estimation problems. Accelerated gradient methods such as PARTAN and conjugate gradient are reasonable possibilities.

Other convergence improvement techniques are modifications of the second-order methods. Many convergence problems relate to ill-conditioned or nonpositive second gradient matrices. This suggests such modifications as adding positive definite matrices to the second gradient or using rank-deficient solutions.

Constraints on the allowable range of estimates or on the change per iteration can also have stabilizing effects. A particularly popular constraint is to fix some of the ordinates at constant values, thus reducing the dimension of the optimization problem; this is a form of axial iteration, and its effectiveness depends on a wise (or lucky) choice of the ordinates to be constrained.

Relaxation methods, which reduce the indicated parameter changes by some fixed percentage, can sometimes stabilize oscillating behavior of the algorithm. Line searches in the indicated direction extend this concept and should be capable of stabilizing almost any problem, at the cost of additional function evaluations.

The above list of convergence improvement techniques is far from complete. It also omits mention of numerous important implementation details. This list serves only to call attention to the area of convergence improvement. See the references for more thorough treatments.

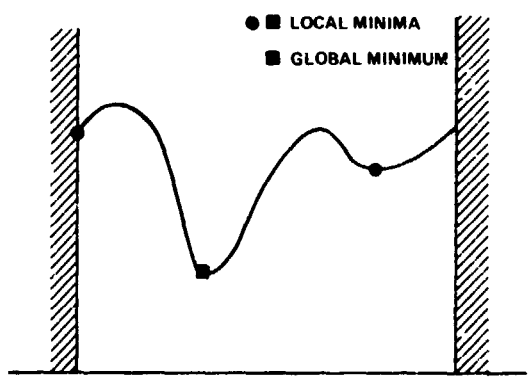


Figure (2.0-1). Illustration of local and global minima.

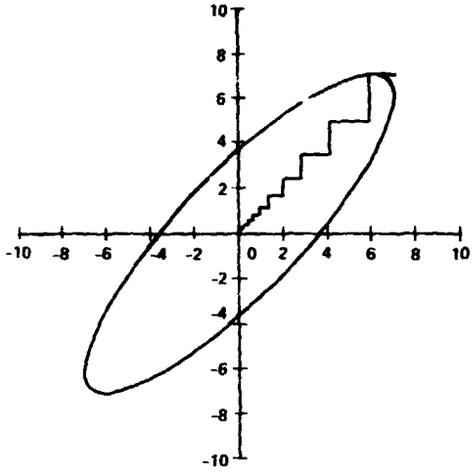


Figure (2.2-1). Behavior of axial iteration.

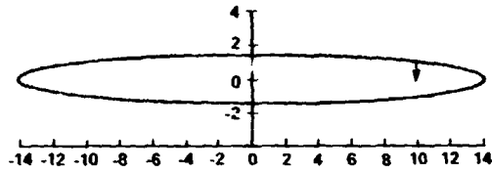


Figure (2.3-2). The gradient direction near a narrow valley.

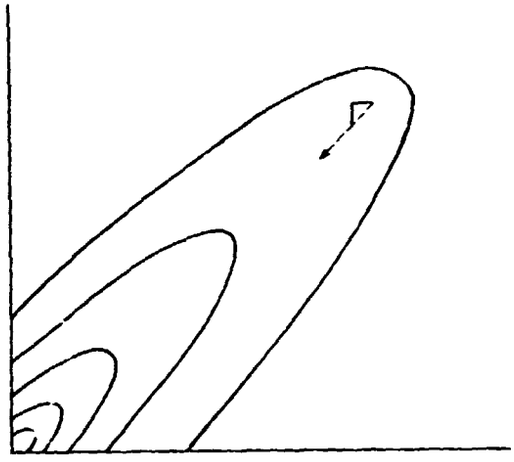


Figure (2.2-2). The pattern direction.

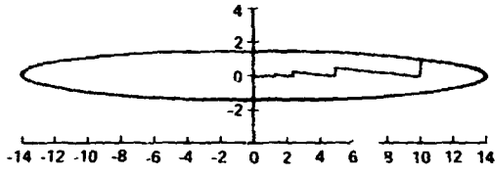


Figure (2.3-3). Behavior of the gradient algorithm in a narrow valley.

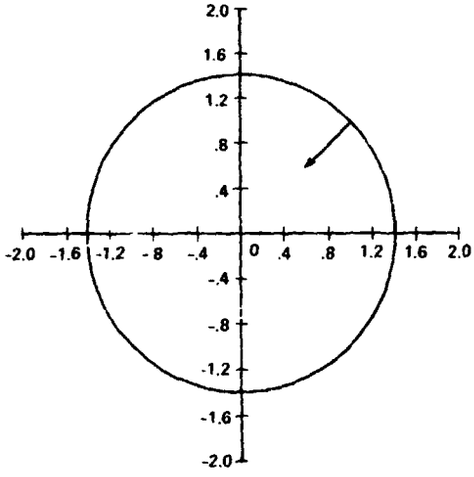


Figure (2.3-1). The gradient direction from a circular isocline.

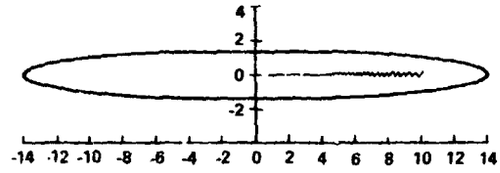


Figure (2.3-4). Worse behavior of the gradient algorithm.

CHAPTER 3

3.0 BASIC PRINCIPLES FROM PROBABILITY

In this chapter we will review some basic definitions and results from probability theory. We presume that the reader has had previous exposure to this material. Our aim here is to review and serve as a reference for those concepts that are used extensively in the following chapters. The treatment, therefore, is quite abbreviated, and devotes little time to motivating the field of study or philosophizing about the results. Proofs of several of the statements are omitted. Some of the other proofs are merely outlined, with some of the more tedious steps omitted. Apostol (1969), Ash (1970), and Papoulis (1965) give more detailed treatment.

3.1 PROBABILITY SPACES

3.1.1 Probability Triple

A probability space is formally defined by three items (Ω,β,P), sometimes called the probability triple. Ω is called the sample space, and the elements ω of Ω are called outcomes or realizations. β is a set of sets defined on Ω, closed under countable set operations (union, intersection, and complement). Each set $B \in \beta$ is called an event. In the current discussion, we will not be concerned with the fine details of the definition of β. β is referred to as the class of measurable sets and is studied in measure theory (Royden, 1968; Rudin, 1974). P is a scalar valued function defined on β, and is called the probability function or probability measure. For each set B in β, the function $P(B)$ defines the probability that ω will be in B. P must satisfy the following axioms:

1) $0 \le P(B) \le 1$ for all $B \in \beta$

2) $P(\Omega) = 1$

3) $P\left(\sum_i B_i\right) = \sum_i P(B_i)$ for all countable sequences of disjoint $B_i \in \beta$

3.1.2 Conditional Probabilities

If A and B are two events and $P(B) \neq 0$, the conditional probability of A given B is defined as

$$P(A|B) = P(A|B)/P(B) \tag{3.1-1}$$

where $A|B$ is the set intersection of the events A and B.

The events A and B are statistically independent if $P(A|B) = P(A)$. Note that this condition is symmetric; that is, if $P(A|B) = P(A)$, then $P(B|A) = P(B)$, provided that $P(A|B)$ and $P(B|A)$ are both defined.

3.2 SCALAR RANDOM VARIABLES

A scalar real-valued function $X(\omega)$ defined on Ω is called a random variable if the set $\{\omega:X(\omega) < x\}$ is in β for all real x.

3.2.1 Distribution and Density Functions

Every random variable has a distribution function defined as follows:

$$F_X(x) = P(\{\omega:X(\omega) < x\}) \tag{3.2-1}$$

It follows directly from the properties of a probability measure that $F_X(x)$ must be a nondecreasing function of x, with $F_X(-\infty) = 0$ and $F_X(\infty) = 1$. By the Lebesque decomposition lemma (Royden, 1968, p. 240; Rudin, 1974, p. 129), any distribution function can always be written as the sum of a differentiable component and a component which is piecewise constant with a countable number of discontinuities. In many cases, we will be concerned with variables with differentiable distribution functions. For such random variables, we define a function, $p_X(s)$, called the probability density function, to be the derivative of the distribution function:

$$p_X(x) = \frac{d}{dx} F_X(x) \tag{3.2-2}$$

We have also the inverse relationship

$$F_X(x) = \int_{-\infty}^{x} p_X(s)ds \tag{3.2-3}$$

A probability density function must be nonnegative, and its integral over the real line must equal 1. For simplicity of notation, we will often shorten $p_X(s)$ to $p(x)$ where the meaning is clear. Where confusion is possible, we will retain the longer notation.

A probability distribution can be defined completely by giving either the distribution function or the density function. We will work mainly with density functions, except when they are not defined.

3.2.2 Expectations and Moments

The expected value of a random variable, X, is defined by

$$E\{X\} = \int_{-\infty}^{\infty} x p_X(x) dx \tag{3.2-4}$$

If X does not have a density function, the precise definition of the expectation is somewhat more technical, involving a Stieltjes integral; Equation (3.2-4) is adequate for the needs of this document. The expected value is also called the expectation or the mean. Any (measurable) function of a random variable is also a random variable and

$$E\{f(X)\} \int_{-\infty}^{\infty} f(x) p_X(x) dx \tag{3.2-5}$$

The expected value of X^n for positive n is called the nth moment of X. Under mild conditions, knowledge of all of the moments of a distribution is sufficient to define the distribution (Papoulis, 1965, p. 158).

The variance of X is defined as

$$\begin{aligned} \mathrm{var}(X) &\equiv E\{(X - E\{X\})^2\} \\ &= E\{X^2\} + E\{X\}^2 - 2E\{X\}E\{X\} \\ &= E\{X^2\} - E\{X\}^2 \end{aligned} \tag{3.2-6}$$

The standard deviation is the square root of the variance.

3.3 JOINT RANDOM VARIABLES

Two random variables defined on the same sample space are called joint random variables.

3.3.1 Distribution and Density Functions

If two random variables, X and Y, are defined on the same sample space, we define a joint distribution function of these variables as

$$F_{X,Y}(x,y) = P(\{\omega : X(\omega) < x,\ Y(\omega) < Y\}) \tag{3.3-1}$$

For absolutely continuous distribution functions, a joint probability density function $p_{X,Y}(x,y)$ is defined by the partial derivative

$$p_{X,Y}(x,y) = \frac{\partial^2}{\partial x \partial y} F_{X,Y}(x,y) \tag{3.3-2}$$

We then have also

$$F_{X,Y}(x,y) = \int_{-\infty}^{x} \int_{-\infty}^{y} p_{X,Y}(s,t) dt\, ds \tag{3.3-3}$$

In a similar manner, joint distributions and densities of N random variables can be defined. As in the scalar case, the joint density function of N random variables must be nonnegative and its integral over the entire space must equal 1.

A random N-vector is the same as N jointly random scalar variables, the only difference being in the terminology.

3.3.2 Expectations and Moments

The expected value of a random vector X is defined as in the scalar case:

$$E\{X\} = \int_{-\infty}^{\infty} x p_X(x) ds \tag{3.3-4}$$

The covariance of X is a matrix defined by

$$\begin{aligned} \mathrm{cov}(X) &= E\{[X - E(X)][X - E(X)]^*\} \\ &= E\{XX^*\} - E\{X\}E\{X\}^* \end{aligned} \tag{3.3-5}$$

The covariance matrix is always symmetric and positive semi-definite. It is positive definite if X has a density function. Higher order moments of random vectors can be defined, but are notationally clumsy and seldom used.

Consider a random vector Y given by

$$Y = AX + b \tag{3.3-6}$$

where A is any deterministic matrix (not necessarily square), and b is an appropriate length deterministic vector. Then the mean and covariance of Y are

$$E\{Y\} = E\{AX + b\} = AE\{X\} + b \tag{3.3-7}$$

$$\begin{aligned} \mathrm{cov}(Y) &= E\{[Y - E(Y)][Y - E(Y)]^*\} \\ &= E\{[AX + b - AE(X) - b][AX + b - AE(X) - b]^*\} \\ &= AE\{[X - E(X)][X - E(X)]^*\} \\ &= A\,\mathrm{cov}(X)A^* \end{aligned} \tag{3.3-8}$$

3.3.3 Marginal and Conditional Distributions

If X and Y are jointly random variables with a joint distribution function given by Equation (3.3-1), then X and Y are also individually random variables, with distribution functions defined as in Equation (3.2-1). The individual distributions of X and Y are called the marginal distributions, and the corresponding density functions are called marginal density functions.

The marginal distributions of X and Y can be derived from the joint distribution. (Note that the converse is false without additional assumptions.) By comparing Equations (3.2-1) and (3.3-1), we obtain

$$F_X(x) = F_{X,Y}(x,\infty) \tag{3.3-9a}$$

and correspondingly

$$F_Y(y) = F_{X,Y}(\infty,y) \tag{3.3-9b}$$

In terms of the density functions, using Equations (3.2-2) and (3.3-3), we obtain

$$p_X(x) = \int_{-\infty}^{\infty} p_{X,Y}(x,y)dy \tag{3.3-10a}$$

$$p_Y(y) = \int_{-\infty}^{\infty} p_{X,Y}(x,y)dx \tag{3.3-10b}$$

The conditional distribution function of X given Y is defined as (see Equation (3.1-1))

$$F_{X|Y}(x|y) = P(\{\omega:X(\omega) < x\}|\{\omega:Y(\omega) < y\}) \tag{3.3-11}$$

and correspondingly for $F_{Y|X}$. The conditional density function, when it exists, can be expressed as

$$p_{X|Y}(x|y) = p_{X,Y}(x,y)/p_Y(y) \tag{3.3-12}$$

Equation (3.3-12) is known as Bayes' rule.

The conditional expectation is defined as

$$E\{X|Y\} = \int_{-\infty}^{\infty} xp_{X|Y}(x|y)dx \tag{3.3-13}$$

assuming that the density function exists. Using Equation (3.3-13), we obtain the useful decomposition

$$E\{f(X,Y)\} = E\{E(f(X,Y)|Y)\} \tag{3.3-14}$$

3.3.4 Statistical Independence

Two random vectors X and Y defined on the same probability space are defined to be independent if

$$F_{X,Y}(x,y) = F_X(x)F_Y(y) \tag{3.3-15}$$

If the joint probability density function exists, we can write this condition as

$$p_{X,Y}(x,y) = p_X(x)p_Y(y) \tag{3.3-16}$$

An immediate corollary, using Equation (3.3-12), is that $p_{X|Y}$ does not depend on y, and $p_{Y|X}$ does not depend on x. If X and Y are independent, then $f(X)$ and $g(Y)$ are independent for any functions f and g.

Two vectors are uncorrelated if

$$E\{XY^*\} = E\{X\}E\{Y^*\} \tag{3.3-17}$$

or equivalently if

$$E\{(X - E\{X\})(Y - E\{Y\})^*\} = 0 \tag{3.3-18}$$

If X and Y are uncorrelated, then the covariance of their sum equals the sum of their covariances.

$$\mathrm{cov}(X + Y) = \mathrm{cov}(X) + \mathrm{cov}(Y) \tag{3.3-19}$$

If two vectors are independent, then they are uncorrelated, but the converse of this statement is false.

3.4 TRANSFORMATION OF VARIABLES

A large part of probability theory is concerned in some manner with the transformation of variables; i.e., characterizing random variables defined as functions of other random variables. We have previously cited limited results on the means and covariances of some transformed variables (Equations (3.2-5), (3.3-7), and (3.3-8)). In this section we seek the entire density function. Our consideration is restricted to variables that have density functions. Let X be a random vector with density function $p_X(x)$ defined on R_n, the Euclidean space of real n-vectors. Then define $Y \in R_m$ by $Y = f(X)$. We seek to derive the density function of Y. There are three cases to consider, depending on whether $m = n$, $m > n$, or $m < n$.

The primary case of interest is when $m = n$. Assume that $f(\cdot)$ is invertible and has continuous partial derivatives. (Technically, this is only required almost everywhere.) Define $g(Y) = f^{-1}(Y)$. Then

$$p_Y(y) = p_X(g(y))|\det(J)| \tag{3.4-1}$$

where J is the Jacobian of the transformation g

$$J_{ij} = \frac{\partial g_i(y)}{\partial y_j} \tag{3.4-2}$$

See Rudin (1974, p. 186) and Apostol (1969, p. 394) for the proof.

<u>Example 3.4-1</u> Let $Y = CX$, with C square and nonsingular. Then $g(y) = C^{-1}y$ and $J = C^{-1}$, giving

$$p_Y(y) = p_X(C^{-1}y)|\det(C^{-1})|$$

as the transformation equation.

If f is not invertible, the distribution of Y is given by a sum of terms similar to Equation (3.4-1).

For the case with $m > n$, the distribution of Y will be concentrated on, at most, an n-dimensional hypersurface in R_m, and will not have a density function in R_m.

The simplest nontrivial case of $m < n$ is when Y consists of a subset of the elements of X. In this case, the density function sought is the density function of the marginal distribution of the pertinent subset of the elements of X. Marginal distributions were discussed in Section 3.3.3. In general, when $m < n$, X can be transformed into a random vector $Z \in R_n$, such that Y is a subset of the elements of Z.

<u>Example 3.4-2</u> Let $X \in R_2$ and $Y = X_1 + X_2$. Define $Z = CX$ where

$$C = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}$$

Then using example 3.4-1,

$$\begin{aligned} p_Z(z) &= p_X(C^{-1})|\det(C^{-1})| \\ &= \frac{1}{2}\, p_X(C^{-1}z) \end{aligned}$$

where

$$C^{-1} = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}$$

Then $Y = Z_1$, so the distribution of Y is the marginal distribution of Z_1, which can be computed from Equation (3.3-10).

3.5 GAUSSIAN VARIABLES

Random variables with Gaussian distributions play a major role in this document and in much of probability theory. We will, therefore, briefly review the definition and some of the salient properties of Gaussian distributions. These distributions are often called normal distributions in the literature.

3.5.1 Standard Gaussian Distributions

All Gaussian distributions derive from the distribution of a standard Gaussian variable with mean 0 and covariance 1. The density function of the standard Gaussian distribution is defined to be

$$p(x) = (2\pi)^{-1/2} \exp\left(-\frac{1}{2} x^2\right) \tag{3.5-1}$$

The distribution function does not have a simple closed-form expression. We will first show that Equation (3.5-1) is a valid density function with mean 0 and covariance 1. The most difficult part is showing that its integral over the real line is 1.

Theorem 3.5-1 Equation (3.5-1) defines a valid probability density function.

Proof The function is obviously nonnegative. There remains only to show that its integral over the real line is 1. Taking advantage of the symmetry about 0, we can reduce this problem to proving that

$$\int_0^\infty \exp\left(-\frac{1}{2} x^2\right) dx = \sqrt{(\pi/2)} \tag{3.5-2}$$

There is no closed-form expression for this integral over any finite range, but for the semi-infinite range of Equation (3.5-2) the following "trick" works. Form the square of the integral:

$$\left[\int_0^\infty \exp\left(-\frac{1}{2} x^2\right) dx\right]^2 = \int_0^\infty \int_0^\infty \exp\left[-\frac{1}{2}(x^2 + y^2)\right] dx\, dy \tag{3.5-3}$$

Then change variables to polar coordinates, substituting r^2 for $x^2 + y^2$ and $r\, dr\, d\theta$ for $dx\, dy$, to get

$$\left[\int_0^\infty \exp\left(-\frac{1}{2} x^2\right) dx\right]^2 = \int_0^{1/2\pi} \int_0^\infty r \exp\left(-\frac{1}{2} r^2\right) dr\, d\theta \tag{3.5-4}$$

The integral in Equation (3.5-4) has a closed-form solution:

$$\int_0^\infty r \exp\left(-\frac{1}{2} r^2\right) dr = -\exp\left(-\frac{1}{2} r^2\right)\Big|_0^\infty = 0 - (-1) = 1 \tag{3.5-5}$$

Thus,

$$\left[\int_0^\infty \exp\left(-\frac{1}{2} x^2\right) dx\right]^2 = \int_0^{1/2\pi} 1\, d\theta = \frac{\pi}{2} \tag{3.5-6}$$

Taking the square root gives Equation (3.5-2), completing the proof.

The mean of the distribution is trivially zero by symmetry. To derive the covariance, note that

$$E\{1 - X^2\} = \int_{-\infty}^\infty (1 - x^2)(2\pi)^{-1/2} \exp\left(-\frac{1}{2} x^2\right) dx = (2\pi)^{-1/2} x \exp\left(-\frac{1}{2} x^2\right)\Big|_{-\infty}^\infty = 0 \tag{3.5-9}$$

Thus,

$$\mathrm{cov}(X) = E\{X^2\} - E\{X\}^2 = 1 - 0 = 1 \tag{3.5-10}$$

This completes our discussion of the scalar standard Gaussian.

We define a standard multivariate Gaussian vector to be the concatenation of n independent standard Gaussian variables. The standard multivariate Gaussian density function is therefore the product of n marginal density functions in the form of Equation (3.5-1).

$$\begin{aligned} p(x) &= \prod_{i=1}^{n} (2\pi)^{-1/2} \exp\left(-\frac{1}{2} x_i^2\right) \\ &= (2\pi)^{-n/2} \exp\left(-\frac{1}{2} x^* x\right) \end{aligned} \tag{3.5-11}$$

The mean of this distribution is 0 and the covariance is an identity matrix.

3.5.2 General Gaussian Distributions

We will define the class of all Gaussian distributions by reference to the standard Gaussian distributions of the previous section. We define a random vector Y to have a Gaussian distribution if Y can be represented in the form

$$Y = AX + m \tag{3.5-12}$$

where X is a standard Gaussian vector, A is a deterministic matrix and m is a deterministic vector. The A matrix need not be square. Note that any deterministic vector is a special case of a Gaussian vector with a zero A matrix.

We have defined the class of Gaussian random variables by a set of operations that can produce such variables. It now remains to determine the forms and properties of these distributions. (This is somewhat backwards from the most common approach, where the forms of the distributions are first defined and Equation (3.5-12) is proven as a result. We find that our approach makes it somewhat easier to handle singular and nonsingular cases consistently without introducing characteristic functions (Papoulis, 1965).

By Equations (3.3-7) and (3.3-8), the Y defined by Equation (3.5-12) has mean m and covariance AA^*. Our first major result will be to show that a Gaussian distribution is uniquely specified by its mean and covariance; that is, if two distributions are both Gaussian and have equal means and covariances, then the two distributions are identical. Note that this does not mean that the A matrices need to be identical; the reason the result is nontrivial is that an infinite number of different A matrices give the same covariance AA^*.

Example 3.5-1 Consider three Gaussian vectors

$$Y_1 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} X_1 , \quad Y_2 = \begin{bmatrix} 0.707 & -0.707 \\ 0.707 & 0.707 \end{bmatrix} X_2 ,$$

and

$$Y_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.866 & 0.5 \end{bmatrix} X_3$$

where X_1 and X_2 are standard Gaussian 2-vectors and X_3 is a standard Gaussian 3-vector. We have

$$\mathrm{cov}(Y_1) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\mathrm{cov}(Y_2) = \begin{bmatrix} 0.707 & -0.707 \\ 0.707 & 0.707 \end{bmatrix}\begin{bmatrix} 0.707 & 0.707 \\ -0.707 & 0.707 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\mathrm{cov}(Y_3) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.866 & 0.5 \end{bmatrix}\begin{bmatrix} 1 & 0 \\ 0.866 & 0 \\ 0.5 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Thus all three Y_i have equal covariance.

The rest of this section is devoted to proving this result in three steps. First, we will consider square, nonsingular A matrices. Second, we will consider general square A matrices. Finally, we will consider nonsquare A matrices. Each of these steps uses the results of the previous step.

Theorem 3.5-2 If Y is a Gaussian n-vector defined by Equation (3.5-12) with a nonsingular A matrix, then the probability density function of Y exists and is given by

$$p(y) = |2\pi\Lambda|^{-1/2} \exp\left[-\frac{1}{2}(y - m)^*\Lambda^{-1}(y - m)\right] \tag{3.5-13}$$

where Λ is the covariance AA^*.

Proof This is a direct application of the transformation of variables formula, Equation (3.4-1).

$$\begin{aligned} p_Y(y) &= p_X[A^{-1}(y - m)]|A^{-1}| \\ &= (2\pi)^{-n/2} \exp\left\{-\frac{1}{2}[A^{-1}(y - m)]^*[A^{-1}(y - m)]\right\}|A|^{-1} \\ &= |2\pi AA^*|^{-1/2} \exp\left[-\frac{1}{2}(y - m)^*(AA^*)^{-1}(y - m)\right] \end{aligned}$$

Substituting Λ for AA^* then gives the desired result.

Note that the density function, Equation (3.5-13), depends only on the mean and covariance, thus proving the uniqueness result for the case restricted to nonsingular matrices. A particular case of interest is where m is 0 and A is unitary. (A unitary matrix is a square one with $AA^* = I$.) In this case, Y has a standard Gaussian distribution.

Theorem 3.5-3 If Y is a Gaussian n-vector defined by Equation (3.5-12) with any square A matrix, then Y can be represented as

$$Y = S\tilde{X} + m \tag{3.5-14}$$

where $\tilde{X}$ is a standard Gaussian n-vector and S is positive semi-definite. Furthermore, the S in this representation is unique and depends only on the covariance of Y.

Proof The uniqueness is easy to prove, and we will do it first. The covariance of the Y given in Equation (3.5-12) is AA^*. The covariance of a Y expressed as in Equation (3.5-14) is SS^*. A necessary (bu not sufficient) condition for Equation (3.5-14) to be a valid representation of Y is therefore, that SS^* equal AA^*. It is an elementary result of linear algebra (Wilkinson, 1965; Dongarra, Moler, Bunch, and Stewart, 1979; and Strang, 1980) that AA^* is always positive semi-definite and that there is one and only one positive semi-definite matrix S satisfying $SS^* = AA^*$. S is called the matrix square root of AA^*. This proves the uniqueness.

The existence proof relies on another result from linear algebra: any square matrix A can be factored as SQ, where S is positive semi-definite and Q is unitary. For nonsingular A, this factorization is easy—S is the matrix square root of AA^* and Q is $S^{-1}A$. A formal proof for general A matrices would be too long a diversion into linear algebra for our current purposes, so we will omit it. This factorization is closely related to, and can be formally derived from, the well-known QR factorization, where Q is unitary and R is upper triangular (Wilkinson, 1965; Dongarra, Moler, Bunch, and Stewart, 1979; and Strang, 1980).

Given the SQ factorization of A, define

$$\tilde{X} = QX \tag{3.5-15}$$

By theorem (3.5-2), $\tilde{X}$ is a standard Gaussian n-vector. Substituting into Equation (3.5-12) gives Equation (3.5-14), completing the proof.

Because the S in the above theorem depends only on the covariance of Y, it immediately follows that the distribution of any Gaussian variable generated by a square A matrix is uniquely specified by the mean and covariance. It remains only to extend this result to rectangular A matrices.

Theorem 3.5-4 The distribution of any Gaussian vector is uniquely defined by its mean and covariance.

Proof We have already shown the result for Gaussian vectors generated by square A matrices. We need only show that a Gaussian vector generated by a rectangular A matrix can be rewritten in terms of a square A matrix. Let A be n-by-m, and consider the two cases, $n > m$ and $n < m$. If $n < m$, define a standard Gaussian n-vector $\tilde{X}$ by augmenting the X vector with $n - m$ independent standard Gaussians. Then define an n-by-n matrix $\tilde{A}$ by augmenting A with $n - m$ rows of zeros. We then have

$$Y = \tilde{A}\tilde{X} + m$$

as desired.

For the case $n < m$, define a random m-vector $\tilde{Y}$ by augmenting Y with $m - n$ zeros. Then

$$\tilde{Y} = \tilde{A}X + \tilde{m}$$

where $\tilde{m}$ and $\tilde{A}$ are obtained by augmenting zeros to m and A. Use Theorem (3.5-3) to rewrite $\tilde{Y}$ as

$$\tilde{Y} = S\tilde{X} + \tilde{m} \tag{3.5-16}$$

Since the last $m - n$ elements of $\tilde{Y}$ are zero, Equation (3.5-16) must be in the form

$$\begin{bmatrix} Y \\ 0 \end{bmatrix} = \begin{bmatrix} \tilde{S} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \tilde{X}_1 \\ \tilde{X}_2 \end{bmatrix} + \begin{bmatrix} m \\ 0 \end{bmatrix}$$

Thus

$$Y = \tilde{S}\tilde{X} + m$$

which is in the required form.

Theorem (3.5-4) is the central result of this approach to Gaussian variables. It makes the practical manipulation of Gaussian variables much easier. Once you have demonstrated that some result is Gaussian, you

need only derive the mean and covariance to specify the distribution completely. This is far easier than manipulating the full density function or distribution function, a process which often requires partial differential equations. If the covariance matrix is nonsingular, then the density function exists and is given by Equation (3.5-13). If the covariance is singular, a density function does not exist (unless you extend the definition of density functions to include components like impulse functions).

Two properties of the Gaussian density function often provide useful computational shortcuts to evaluating the mean and covariance of nonsingular Gaussians. The first property is that the mean of the density function occurs at its maximum. The mean is thus the unique solution of

$$\nabla_y \ln p(y) = 0 \tag{3.5-17}$$

The logarithm in this equation can be removed, but the equation is usually most useful as written. The second property is that the covariance can be expressed as

$$\mathrm{cov}(Y) = -[\nabla_y^2 \ln p(y)]^{-1} \tag{3.5-18}$$

Both of these properties are easy to verify by direct substitution into Equation (3.5-13).

3.5.3 Properties

In this section we derive several useful properties of Gaussian vectors. Most of these properties relate to operations on Gaussian vectors that give Gaussian results. A major reason for the wide use of Gaussian distributions is that many basic operations on Gaussian vectors give Gaussian results, which can be characterized completely by the mean and covariance.

Theorem 3.5-5 If Y is a Gaussian vector with mean m and covariance A, and if Z is given by

$$Z = BY + b$$

then Z is Gaussian with mean $Bm + b$ and covariance BAB^*.

Proof By definition, Y can be expressed as

$$Y = AX + m$$

where X is a standard Gaussian. Substituting Y into the expression for Z gives

$$Z = B(AX + m) + b = BAX + (Bm + b)$$

proving that Z is Gaussian. The mean and covariance expressions for linear operations on any random vector were previously derived in Equations (3.3-7) and (3.3-8).

Several of the properties discussed in this section involve the concept of jointly Gaussian variables. Two or more random vectors are said to be jointly Gaussian if their joint distribution is Gaussian. Note that two vectors can both be Gaussian and yet not be jointly Gaussian.

Example 3.5-2 Let Y be a Gaussian random variable with mean 0 and variance 1. Define Z as

$$Z = \begin{cases} Y & -1 \le Y \le 1 \\ -Y & \text{elsewhere} \end{cases}$$

The random variable Z is Gaussian with mean 0 and variance 1 (apply Equation (3.4-1) to show this), but Y and Z are not jointly Gaussian.

Theorem 3.5-6 Let Y_1 and Y_2 be jointly Gaussian vectors, and let the mean and covariance of the joint distribution be partitioned as

$$\begin{bmatrix} m_1 \\ m_2 \end{bmatrix} \begin{bmatrix} \Lambda_{11} & \Lambda_{12} \\ \Lambda_{21} & \Lambda_{22} \end{bmatrix}$$

Then the marginal distributions of Y_1 and Y_2 are Gaussian with

$$E\{Y_1\} = m_1 \qquad \mathrm{cov}(Y_1) = \Lambda_{11}$$

$$E\{Y_2\} = m_2 \qquad \mathrm{cov}(Y_2) = \Lambda_{22}$$

Proof Apply theorem (3.5-5) with $B = [1 \quad 0]$ and $C = [0 \quad 1]$.

The following two theorems relate to independent Gaussian variables:

Theorem 3.5-7 If Y and Z are two independent Gaussian variables, then Y and Z are jointly Gaussian.

Proof For nonsingular distributions, this proof is easy to do by writing out the product of the density functions. For a more general proof, we can proceed as follows: write Y and Z as

$$Y = A_1X_1 + m_1$$

$$Z = A_2X_2 + m_2$$

where X_1 and X_2 are standard Gaussian vectors. We can always construct the X_1 and X_2 in these equations to be independent, but the following argument avoids the necessity to prove that statement. Define two independent standard Gaussians, $\tilde{X}_1$ and $\tilde{X}_2$, and further define

$$\tilde{Y} = A_1\tilde{X}_1 + m_1$$

$$\tilde{Z} = A_2\tilde{X}_2 + m_2$$

Then $\tilde{Y}$ and $\tilde{Z}$ have the same joint distribution as Y and Z. The concatenation of $\tilde{X}_1$ and $\tilde{X}_2$ is a standard Gaussian vector. Therefore, $\tilde{Y}$ and $\tilde{Z}$ are jointly Gaussian because they can be expressed as

$$\begin{bmatrix}\tilde{Y}\\ \tilde{Z}\end{bmatrix} = \begin{bmatrix}A_1 & 0\\ 0 & A_2\end{bmatrix}\begin{bmatrix}\tilde{X}_1\\ \tilde{X}_2\end{bmatrix} + \begin{bmatrix}m_1\\ m_2\end{bmatrix}$$

Since Y and Z have the same joint distribution as $\tilde{Y}$ and $\tilde{Z}$, Y and Z are also jointly Gaussian.

Theorem 3.5-8 If Y and Z are two uncorrelated jointly Gaussian variables, then Y and Z are independent and Gaussian.

Proof By theorem (3.5-3), we can express

$$\begin{bmatrix}Y\\ Z\end{bmatrix} = SX + m$$

where X is a standard Gaussian vector and S is positive semi-definite. Partition S as

$$\begin{bmatrix}S_{11} & S_{12}\\ S_{21} & S_{22}\end{bmatrix}$$

By the definition of "uncorrelated," we must have $S_{12} = S_{21}^* = 0$. Therefore, partitioning X into X_1 and X_2, and partitioning m into m_1 and m_2, we can write

$$Y = S_{11}X_1 + m_1$$

$$Z = S_{22}X_2 + m_2$$

Since Y and Z are functions of the independent vectors X_1 and X_2, Y and Z are independent and Gaussian.

Since any two independent vectors are uncorrelated, Theorem (3.5-8) proves that independence and lack of correlation are equivalent for Gaussians.

We previously covered marginal distributions of Gaussian vectors. The following theorem considers conditional distributions. We will directly consider only conditional distributions of nonsingular Gaussians. Since the results of the theorem involve inverses, there are obvious difficulties that cannot be circumvented by avoiding the use of probability density functions in the proof.

Theorem 3.5-9 Let Y_1 and Y_2 be jointly Gaussian variables with a nonsingular joint distribution. Partition the mean, covariance, and inverse covariance of the joint distribution as

$$m = \begin{bmatrix}m_1\\ m_2\end{bmatrix}, \quad \Lambda = \begin{bmatrix}\Lambda_{11} & \Lambda_{12}\\ \Lambda_{21} & \Lambda_{22}\end{bmatrix}, \quad \text{and} \quad \Gamma = \Lambda^{-1} = \begin{bmatrix}\Gamma_{11} & \Gamma_{12}\\ \Gamma_{21} & \Gamma_{22}\end{bmatrix}$$

Then the conditional distributions of Y_1 given Y_2, and of Y_2 given Y_1, are Gaussian with means and covariances

$$E\{Y_1|Y_2\} = m_1 + \Lambda_{12}\Lambda_{22}^{-1}(y_2 - m_2) \tag{3.5-18a}$$

$$\mathrm{cov}\{Y_1|Y_2\} = \Lambda_{11} - \Lambda_{12}\Lambda_{22}^{-1}\Lambda_{21} = (\Gamma_{11})^{-1} \tag{3.5-18b}$$

$$E\{Y_2|Y_1\} = m_2 + \Lambda_{21}\Lambda_{11}^{-1}(y_1 - m_1) \tag{3.5-19a}$$

$$\mathrm{cov}\{Y_2|Y_1\} = \Lambda_{22} - \Lambda_{21}\Lambda_{11}^{-1}\Lambda_{12} = (\Gamma_{22})^{-1} \tag{3.5-19b}$$

Proof The joint probability density function of Y_1 and Y_2 is

$$p(y_1,y_2) = c \exp\left\{-\frac{1}{2}\begin{bmatrix} y_1 - m_1 \\ y_2 - m_2 \end{bmatrix}^* \begin{bmatrix} \Gamma_{11} & \Gamma_{12} \\ \Gamma_{21} & \Gamma_{22} \end{bmatrix} \begin{bmatrix} y_1 - m_1 \\ y_2 - m_2 \end{bmatrix}\right\} \tag{3.5-20}$$

where c is a scalar constant, the magnitude of which we will not need to compute. Expanding the exponent, and recognizing that $\Gamma_{21} = \Gamma_{12}^*$, gives

$$p(y_1,y_2) = c \exp\Big[-\frac{1}{2}(y_1 - m_1)^*\Gamma_{11}(y_1 - m_1) - \frac{1}{2}(y_2 - m_2)^*\Gamma_{22}(y_2 - m_2) - (y_1 - m_1)^*\Gamma_{12}(y_2 - m_2)\Big]$$

Completing squares results in

$$p(y_1,y_2) = c \exp\Big\{-\frac{1}{2}[y_1 - m_1 + \Gamma_{11}^{-1}\Gamma_{12}(y_2 - m_2)]^*\Gamma_{11}[y_1 - m_1 + \Gamma_{11}^{-1}\Gamma_{12}(y_2 - m_2)] - \frac{1}{2}(y_2 - m_2)^*(\Gamma_{22} - \Gamma_{21}\Gamma_{11}^{-1}\Gamma_{12})(y_2 - m_2)\Big\} \tag{3.5-21}$$

Integrating this expression with respect to y_1 gives the marginal density function of Y_2. The second term in the exponent does not involve y_1, and we recognize the first term as the exponent in a Gaussian density function with mean $m_1 - \Gamma_{11}^{-1}\Gamma_{12}(y_2 - m_2)$ and covariance Γ_{11}. Its integral with respect to y_1 is therefore a constant independent of y_2. The marginal density function of Y_2 is therefore

$$p(y_2) = c_2 \exp\Big[-\frac{1}{2}(y_2 - m_2)^*(\Gamma_{22} - \Gamma_{21}\Gamma_{11}^{-1}\Gamma_{12})(y_2 - m_2)\Big] \tag{3.5-22}$$

where c_2 is a constant. Note that because we know that Equation (3.5-22) must be a probability density function, we need not compute the value of c_2; this saves us a lot of work. Equation (3.5-22) is an expression for a Gaussian density function with mean m_2 and covariance $(\Gamma_{22} - \Gamma_{21}\Gamma_{11}^{-1}\Gamma_{12})^{-1}$. The partitioned matrix inversion lemma (Appendix A) gives us

$$(\Gamma_{22} - \Gamma_{21}\Gamma_{11}^{-1}\Gamma_{12})^{-1} = \Lambda_{22}$$

thus independently verifying the result of Theorem (3.5-6) on the marginal distribution.

The conditional density of Y_1 given Y_2 is obtained using Bayes' rule, by dividing Equation (3.5-21) by Equation (3.5-22)

$$p(y_1|y_2) = \frac{p(y_1,y_2)}{p(y_2)}$$

$$= c_1 \exp\Big\{-\frac{1}{2}[y_1 - m_1 + \Gamma_{11}^{-1}\Gamma_{12}(y_2 - m_2)]^*\Gamma_{11}[y_1 - m_1 + \Gamma_{11}^{-1}\Gamma_{12}(y_2 - m_2)]\Big\} \tag{3.5-23}$$

where c_1 is a constant. This is an expression for a Gaussian density function with a mean $m_1 - \Gamma_{11}^{-1}\Gamma_{12}(y_2 - m_2)$ and covariance Γ_{11}^{-1}. The partitioned matrix inversion lemma (Appendix A) then gives

$$\Gamma_{11}^{-1} = \Lambda_{11} - \Lambda_{12}\Lambda_{22}^{-1}\Lambda_{21}$$

$$\Gamma_{11}^{-1}\Gamma_{12} = \Lambda_{12}\Lambda_{22}^{-1}$$

Thus the conditional distribution of Y_1 given Y_2 is Gaussian with mean $m_1 + \Lambda_{12}\Lambda_{22}^{-1}(y_2 - m_2)$ and covariance $\Lambda_{11} - \Lambda_{12}\Lambda_{22}^{-1}\Lambda_{21}$, as we desired to prove. The conditional distribution of Y_2 given Y_1 follows by symmetry.

The final result of this section concerns sums of Gaussian variables.

Theorem 3.5-10 If Y_1 and Y_2 are jointly Gaussian random vectors of equal length and their joint distribution has mean and covariance partitioned as

$$m = \begin{bmatrix} m_1 \\ m_2 \end{bmatrix} \qquad \Lambda = \begin{bmatrix} \Lambda_{11} & \Lambda_{12} \\ \Lambda_{21} & \Lambda_{22} \end{bmatrix}$$

Then $Y_1 + Y_2$ is Gaussian with mean $m_1 + m_2$ and covariance $\Lambda_{11} + \Lambda_{22} + \Lambda_{12} + \Lambda_{21}$.

Proof Apply Theorem (3.5-5) with $B = [I \quad I]$ and $b = 0$.

A simple summary of this section is that linear operations on Gaussian variables give Gaussian results. This principle is not generally true for nonlinear operations. Therefore, Gaussian distributions are strongly associated with the analysis of linear systems.

3.5.4 Central Limit Theorem

The Central Limit Theorem is often used as a basis for justifying the assumption that the distribution of some physical quantity is approximately Gaussian.

Theorem 3.5-11 Let $Y_1, Y_2 \ldots$ be a sequence of independent, identically distributed random vectors with finite mean m and covariance Λ. Then the vectors

$$Z_N = \frac{1}{\sqrt{N}} \sum_{i=1}^{N} (Y_i - m)$$

converge in distribution to a Gaussian vector with mean zero and covariance Λ.

Proof See Ash (1970, p. 171) and Apostol (1969, p. 567).

Cramer (1946) discusses several variants on this theorem, where the Y_i need not be independent and identically distributed, but other requirements are placed on the distributions. The general result is that sums of random variables tend to Gaussian limits under fairly broad conditions. The precise conditions will not concern us here. An implication of this theorem is that macroscopic behavior which is the result of the summation of a large number of microscopic events often has a Gaussian distribution. The classic example is Brownian motion. We will illustrate the Central Limit Theorem with a simple example.

Example 3.5-3 Let the distribution of the Y_i in Theorem (3.5-11) be uniform on the interval (-1,1). Then the mean is zero and the covariance is 1/3. Examine the density functions of the first few Z_i.

The first function, Z_1, is equal to Y_1, and thus is uniform on (-1,1). Figure (3.5-1) compares the densities of Z_1 and the Gaussian limit. The Gaussian limit distribution has mean zero and variance 1/3.

For the second function we have

$$Z_2 = \frac{1}{\sqrt{2}} (Y_1 + Y_2)$$

and the density function of Z_2 is given by

$$p(z_2) = \frac{1}{2} (\sqrt{2} - |z|) \quad \text{for} \quad |z| \le \sqrt{2}$$

Figure (3.5-2) compares the density of Z_2 with the Gaussian limit.

The density function of Z_3 is given by

$$p(z_3) = \begin{cases} \frac{3\sqrt{3}}{8} (1 - z^2) & |z| \le \frac{1}{\sqrt{3}} \\ \frac{3\sqrt{3}}{16} (z^2 - 2\sqrt{3}|z| + 3) & \frac{1}{\sqrt{3}} \le |z| \le \sqrt{3} \\ 0 & |z| \ge \sqrt{3} \end{cases}$$

Figure (3.5-3) compares density of Z_3 with the Gaussian limit. By the time N is 3, Z_N is already becoming reasonably close to Gaussian.

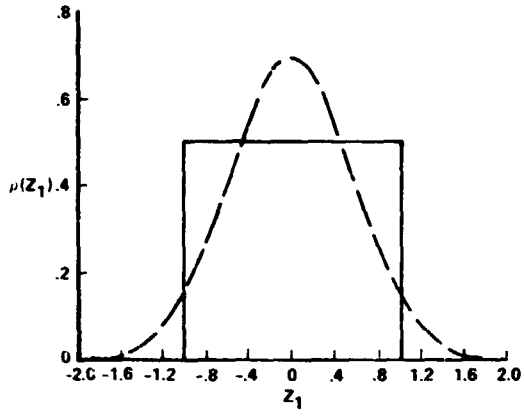


Figure (3.5-1). Density functions of Z_1 and the limit Gaussian.

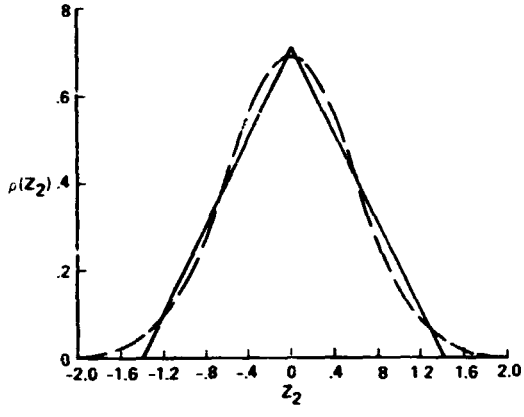


Figure (3.5-2). Density functions of Z_2 and the limit Gaussian.

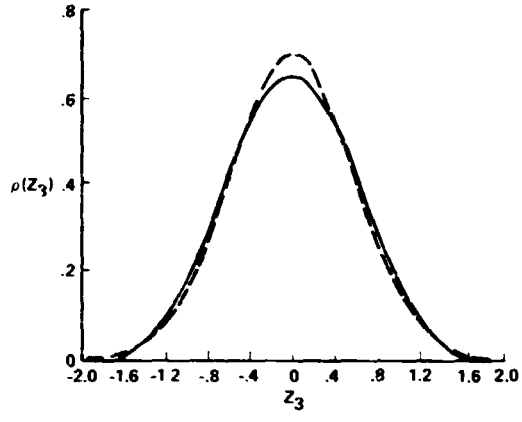


Figure (3.5-3). Density functions of Z_3 and the limit Gaussian.

CHAPTER 4

4.0 STATISTICAL ESTIMATORS

In this chapter, we introduce the concept of an estimator. We then define some basic measures of estimator performance. We use these measures of performance to introduce several common statistical estimators. The definitions in this chapter are general. Subsequent chapters will treat specific forms. For other treatments of this and related material, see Sorenson (1980), Schweppe (1973), Goodwin and Payne (1977), and Eykhoff (1974). These books also cover other estimators that we do not mention here.

4.1 DEFINITION OF AN ESTIMATOR

The concept of estimation is central to our study. The statistical definition of an estimator is as follows:

Perform an experiment (input) U, taken from the set $\mathcal{U}$ of possible experiments on the system. The system response is a random variable:

$$Z = Z(\xi, U, \omega) \tag{4.1-1}$$

where $\xi \in \Xi$ is the true value of the parameter vector and $\omega \in \Omega$ is the random component of the system.

An estimator is any function of Z with range in Ξ. The value of the function is called the estimate $\hat{\xi}$. Thus

$$\hat{\xi} = \hat{\xi}(Z,U) = \hat{\xi}(Z(\xi,U,\omega),U) \tag{4.1-2}$$

This definition is readily generalized to multiple performances of the same experiment or to the performance of more than one experiment. If N experiments U_i are performed, with responses Z_i, then an estimate would be of the form

$$\begin{aligned}\hat{\xi} &= \hat{\xi}(Z_1,\ldots Z_N,U_1\ldots U_N)\\ &= \hat{\xi}(Z(\xi,U_1,\omega_1),\ldots Z(\xi,U_N,\omega_N),U_1\ldots U_N)\end{aligned} \tag{4.1-3}$$

where the ω_i are independent. The N experiments can be regarded as a single "super-experiment" $(U_1\ldots U_N) \in \mathcal{U}\times\mathcal{U}\times\ldots\times\mathcal{U}$, the response to which is the concatenated vector $(Z_1\ldots Z_N) \in \mathcal{Z}\times\mathcal{Z}\times\ldots\times\mathcal{Z}$. The random element is $(\omega_1\ldots\omega_N) \in \Omega\times\Omega\times\ldots\times\Omega$. Equation (4.1-3) is then simply a restatement of Equation (4.1-2) on the larger space.

For simplicity of notation, we will generally omit the dependence on U from Equations (4.1-1) and (4.1-2). For the most part, we will be discussing parameter estimation based on responses to specific, known inputs; therefore, the dependence of the response and the estimate on the input are irrelevant, and merely clutter up the notation. Formally, all of the distributions and expectations may be considered to be implicitly conditioned on U.

Note that the estimate $\hat{\xi}$ is a random variable because it is a function of Z, which is a random variable. When the experiment is actually performed, specific realizations of these random variables will be obtained. The true parameter value ξ is not usually considered to be random, simply unknown.

In some situations, however, it is convenient to define ξ as a random variable instead of as an unknown parameter. The significant difference between these approaches is that a random variable has a probability distribution, which constitutes additional information that can be used in the random-variable approach. Several popular estimators can only be defined using the random-variable approach. These advantages of the random-variable approach are balanced by the necessity to know the probability distribution of ξ. If this distribution is not known, there are no differences, except in terminology, between the random-variable and unknown-parameter approaches.

A third view of ξ involves ideas from information theory. In this context, ξ is considered to be an unknown parameter as above. Even though ξ is not random, it is defined to have a "probability distribution." This probability distribution does not relate to any randomness of ξ, but reflects our knowledge or information about the value of ξ. Distributions with low variance correspond to a high degree of certainty about the value of ξ, and vice versa. The term "probability distribution" is a misnomer in this context. The terms "information distribution" or "information function" more accurately reflect this interpretation.

In the context of information theory, the marginal or prior distribution $p(\xi)$ reflects the information about ξ prior to performing the experiment. A case where there is no prior information can be handled as a limit of prior distributions with less and less information (variance going to infinity). The distribution of the response Z is a function of the value of ξ. When ξ is a random variable, this is called $p(Z|\xi)$, the conditional distribution of z given ξ. We will use the same notation when ξ is not random in order to emphasize the dependence of the distribution on ξ, and for consistency of notation. When $p(\xi)$ is defined, the joint probability density is then

$$p(Z,\xi) = p(Z|\xi)p(\xi) \tag{4.1-4}$$

The marginal probability density of Z is

$$p(Z) = \int p(Z,\xi)d|\xi| \tag{4.1-5}$$

The conditional density of ξ given Z (also called the posterior density) is

$$p(\xi|Z) = \frac{p(Z,\xi)}{p(Z)} = \frac{p(Z|\xi)p(\xi)}{p(Z)} \tag{4.1-6}$$

In the information theory context, the posterior distribution reflects information about the value of ξ after the experiment is performed. It accounts for the information known prior to the experiment, and the information gained by the experiment.

The distinctions among the random variable, unknown parameter, and information theory points of view are largely academic. Although the conventional notations differ, the equations used are equivalent in all three cases. Our presentation uses the probability density notation throughout. We see little benefit in repeating identical derivations, substituting the term "information function" for "likelihood function" and changing notation. We derive the basic equations only once, restricting the distinctions among the three points of view to discussions of applicability and interpretation.

4.2 PROPERTIES OF ESTIMATORS

We can define an infinite number of estimators for a given problem. The definition of an estimator provides no means of evaluating these estimators, some of which can be ridiculously poor. This section will describe some of the properties used to evaluate estimators and to select a good estimator for a particular problem. The properties are all expressed in terms of optimality criteria.

4.2.1 Unbiased Estimators

A bias is a consistent or repeatable error. The parameter estimates from any specific data set will always be imperfect. It is reasonable to hope, however, that the estimate obtained from a large set of maneuvers would be centered around the true value. The errors in the estimates might be thought of as consisting of two components—consistent errors and random errors. Random errors are generally unavoidable. Consistent or average errors might be removable.

Let us restate the above ideas more precisely. The bias b of an estimator $\hat{\xi}(.)$ is defined as

$$b(\xi) = E\{\hat{\xi}|\xi\} - \xi = E\{\hat{\xi}(Z(\xi,\omega))|\xi\} - \xi \tag{4.2-1}$$

The Z in these equations is a random variable, not a specific realization. Note that the bias is a function of the true value. It averages out (by the $E\{.\}$) the random noise effects, but there is no averaging among the different true values. The bias is also a function of the input U, but this dependence is not usually made explicit. All discussions of bias are implicitly referring to some given input.

An unbiased estimator is defined as an estimator for which the bias is identically zero:

$$b(\xi) = 0 \tag{4.2-2}$$

This requirement is quite stringent because it must be met for every value of ξ. Unbiased estimators may not exist for some problems. For other problems, unbiased estimators may exist, but may be too complicated for practical computation. Any estimator that is not unbiased is called biased.

Generally, it is considered desirable for an estimator to be unbiased. This judgment, however, does not apply to all situations. The bias of an estimator measures only the average of its behavior. It is possible for the individual estimates to be so poor that they are ludicrous, yet average out so that the estimator is unbiased. The following example is taken from Ferguson (1967, p. 136).

> Example 4.2-1 A telephone operator has been working for 10 minutes and wonders if he would be missed if he took a 20 minute coffee break. Assume that calls are coming in as a Poisson process with the average rate of λ calls per 10 minutes, λ being unknown. The number Z of calls received in the first 10 minutes has a Poisson distribution with parameter λ.
>
> $$P(Z|\lambda) = \frac{e^{-\lambda}\lambda^Z}{Z!} \qquad Z = 0,1\ldots$$
>
> On the basis of Z, the operator desires to estimate β, the probability of receiving no calls in the next 20 minutes. For a Poisson process, $\beta = e^{-2\lambda}$. If the estimator $\hat{\beta}(Z)$ is to be unbiased, we must have
>
> $$E\{\hat{\beta}(Z(\beta,\omega))|\beta\} = \beta \quad \text{for all} \quad \beta \in (0,1]$$
>
> Thus
>
> $$\sum_{Z=0}^{\infty} \hat{\beta}(Z)\frac{e^{-\lambda}\lambda^Z}{Z!} = \beta = e^{-2\lambda} \quad \text{for all} \quad \lambda \in [0,\infty)$$
>
> Multiply by e^{λ}, giving
>
> $$\sum_{Z=0}^{\infty} \hat{\beta}(Z)\frac{\lambda^Z}{Z!} = e^{-\lambda}$$

Expand the right-hand side as a power series to get

$$\sum_{Z=0}^{\infty} \frac{\hat{\beta}(Z)\lambda^Z}{Z!} = \sum_{Z=0}^{\infty} \frac{(-1)^Z}{Z!} \lambda^Z$$

The convergent power series are equal for all $\lambda \in [0,\infty)$ if the coefficients are identical. Thus $\hat{\beta}(Z) = (-1)^Z$ is the only unbiased estimator of β for this problem. The operator would estimate the probability of missing no calls as +1 if he had received an even number of calls and -1 if he had received an odd number of calls. This estimator is the only unbiased estimator for the problem, but it is a ridiculously poor one. If the estimates are required to lie in the meaningful range of [0,1], then there is no unbiased estimator, but some quite reasonable biased estimators can be easily constructed.

The bias is a useful tool for studying estimators. In general, it is desirable for the bias to be zero, or at least small. However, because the bias measures only the average properties of the estimates, it cannot be used as the sole criterion for evaluating estimators. It is possible for a biased estimator to be clearly superior to all of the unbiased estimators for a problem.

4.2.2 Minimum Variance Estimators

The variance of an estimator is defined as

$$\mathrm{var}(\hat{\xi}) = E\{(\hat{\xi} - E\{\hat{\xi}|\xi\})(\hat{\xi} - E\{\hat{\xi}|\xi\})^*|\xi\} \tag{4.2-3}$$

Note that the variance, like the bias, is a function of the input and the true value. The variance alone is not a reasonable measure for evaluating an estimator. For instance, any constant estimator (one that always returns a constant value, ignoring the data) has zero variance. These are obviously poor estimators in most situations.

A more useful measure is the mean square error:

$$\mathrm{mse}(\hat{\xi}) = E\{(\hat{\xi} - \xi)^*|\xi\} \tag{4.2-4}$$

The mean square error and variance are obviously identical for unbiased estimators ($E\{\hat{\xi}|\xi\} = \xi$). An estimator is uniformly minimum mean-square error if, for every value of ξ, its mean square error is less than or equal to the mean square error of any other estimator. Note that the mean-square error is a symmetric matrix. One symmetric matrix is less than or equal to another if their difference is positive semi-definite. This definition is somewhat academic at this point because such estimators do not exist except in trivial cases. A constant estimator has zero mean-square error when ξ is equal to the constant. (The performance is poor at other values of ξ.) Therefore, in order to be uniformly minimum mean-square error, an estimator would have to have zero mean-square error for every ξ; otherwise, a constant estimator would be better for that ξ.

The concept of minimum mean-square error becomes more useful if the class of estimators allowed is restricted. An estimator is uniformly minimum mean-square error unbiased if it is unbiased and, for every value of ξ, its mean-square error is less than or equal to that of any other unbiased estimator. Such estimators do not exist for every problem, because the requirement must hold for every value of ξ. Estimators optimum in this sense exist for many problems of interest. The mean-square error and the variance are identical for unbiased estimators, so such optimal estimators are also called uniformly minimum variance unbiased estimators. They are also often called simply minimum variance estimators. This term should be regarded as an abbreviation, because it is not meaningful in itself.

4.2.3 Cramer-Rao Inequality (Efficient Estimators)

The Cramer-Rao inequality is one of the central results used to evaluate the performance of estimators. The inequality gives a theoretical limit to the accuracy that is possible, regardless of the estimator used. In a sense, the Cramer-Rao inequality gives a measure of the information content of the data.

Before deriving the Cramer-Rao inequality, let us prove a brief lemma.

Lemma 4.2-1 Let X and Y be two random N-vectors. Then

$$E\{XX^*\} \geq E\{XY^*\}[E\{YY^*\}]^{-1}E\{YX^*\} \tag{4.2-5}$$

assuming that the inverse exists.

Proof The proof is done by completing the square. Let Λ be any nonrandom N-by-N matrix. Then

$$E\{(X - \Lambda Y)(X - \Lambda Y)^*\} \geq 0 \tag{4.2-6}$$

because it is a covariance matrix. Expanding

$$E\{XX^*\} \geq \Lambda E\{YX^*\} + E\{XY^*\}\Lambda^* - \Lambda E\{YY^*\}\Lambda^* \tag{4.2-7}$$

choose

$$\Lambda = E\{XY^*\}[E\{YY^*\}]^{-1} \tag{4.2-8}$$

Then

$$E\{XX^*\} \geq E\{XY^*\}[E\{YY^*\}]^{-1}E\{YX^*\} + E\{XY^*\}[E\{YY^*\}]^{-1}E\{YX^*\} - [E\{XY^*\}[E\{YY^*\}]^{-1}E\{YY^*\}[E\{YY^*\}]^{-1}E\{YX^*\} \tag{4.2-9}$$

or

$$E\{XX^*\} \geq E\{XY^*\}[E\{YY^*\}]^{-1}E\{YX^*\} \tag{4.2-5}$$

completing the lemma.

We now seek to find a bound on $E\{(\hat{\xi} - \xi)(\hat{\xi} - \xi)^*|\xi\}$, the mean square error of the estimate.

<u>Theorem 4.2-2</u> (Cramer-Rao) Assume that the density $p(Z|\xi)$ exists and is smooth enough to allow the operations below. (See Cramér (1946) for details.) This assumption proves adequate for most cases of interest to us. Pitman (1979) discusses some of the cases where $p(Z|\xi)$ is not as smooth as required here. Then

$$E\{(\hat{\xi}(Z) - \xi)(\hat{\xi}(Z) - \xi)|\xi\} \geq [I + \nabla_\xi b(\xi)]M(\xi)^{-1}[I + \nabla_\xi b(\xi)]^* \tag{4.2-10}$$

where

$$M(\xi) = E\{(\nabla_\xi^* \ln p(Z|\xi))(\nabla_\xi \ln p(Z|\xi))|\xi\} \tag{4.2-11}$$

<u>Proof</u> Let X and Y from lemma (4.2-1) be $\hat{\xi}(Z) - \xi$ and $\nabla_\xi^* \ln p(Z|\xi)$, respectively, and let all of the expectations in the lemma be conditioned on ξ. Concentrate first on the term

$$\begin{aligned} E\{XY^*|\xi\} &\equiv E\{(\hat{\xi}(Z) - \xi)(\nabla_\xi \ln p(Z|\xi))|\xi\} \\ &\equiv \int(\hat{\xi}(Z) - \xi)(\nabla_\xi \ln p(Z|\xi))p(Z|\xi)d|Z| \end{aligned} \tag{4.2-12}$$

where $d|Z|$ is the volume element in the space Z. Substituting the relation

$$\nabla_\xi \ln p(Z|\xi) = \frac{\nabla_\xi(Z|\xi)}{p(Z|\xi)} \tag{4.2-13}$$

gives

$$\begin{aligned} E\{XY^*|\xi\} &= \int(\hat{\xi}(Z) - \xi)(\nabla_\xi p(Z|\xi))d|Z| \\ &= \int \hat{\xi}(Z)(\nabla_\xi p(Z|\xi))d|Z| - \int\xi(\nabla_\xi p(Z|\xi))d|Z| \end{aligned} \tag{4.2-14}$$

Now $\hat{\xi}(Z)$ is not a function of ξ. Therefore, assuming sufficient smoothness of $p(Z|\xi)$ as a function of ξ, the first term becomes

$$\begin{aligned} \int\hat{\xi}(Z)\nabla_\xi p(Z|\xi)d|Z| &= \nabla_\xi \int \hat{\xi}(Z)p(Z|\xi)d|Z| \\ &= \nabla_\xi E\{\hat{\xi}(Z)|\xi\} \end{aligned} \tag{4.2-15}$$

Using the definition (Equation (4.2-1)) of the bias, obtain

$$\nabla_\xi E\{\hat{\xi}(Z)|\xi\} = \nabla_\xi[\xi + b(\xi)] = I + \nabla_\xi b(\xi) \tag{4.2-16}$$

In the second term of Equation (4.2-14), ξ is not a function of Z, so

$$\begin{aligned} \int\xi\nabla_\xi p(Z|\xi)d|Z| &= \xi\nabla_\xi\int p(Z|\xi)d|Z| \\ &= \xi\nabla_\xi 1 = 0 \end{aligned} \tag{4.2-17}$$

Using Equations (4.2-16) and (4.2-17) in Equation (4.2-14) gives

$$E\{XY^*|\xi\} = I + \nabla_\xi b(\xi) \tag{4.2-18}$$

Define the Fisher Information matrix

$$M(\xi) \equiv E\{YY^*|\xi\} \equiv E\{(\nabla_\xi^* \ln p(Z|\xi))(\nabla_\xi \ln p(Z|\xi))|\xi\} \tag{4.2-19}$$

They by lemma (4.2-1)

$$E\{(\hat{\xi}(Z) - \xi)(\hat{\xi}(Z) - \xi)^*|\xi\} \geq [I + \nabla_\xi b(\xi)]M(\xi)^{-1}[I + \nabla_\xi b(\xi)]^* \tag{4.2-10}$$

which is the desired result.

Equation (4.2-10) is the Cramer-Rao inequality. Its specialization to unbiased estimators is of particular interest. For an unbiased estimator, $b(\xi)$ is zero so

$$E\{(\hat{\xi}(Z) - \xi)(\hat{\xi}(Z) - \xi)^*|\xi\} \geq M(\xi)^{-1} \tag{4.2-20}$$

This gives us a lower bound, as a function of ξ, on the achievable variance of any unbiased estimator. An unbiased estimator which attains the equality in Equation (4.2-20) is called an efficient estimator. No estimator can achieve a lower variance than an efficient estimator except by introducing a bias in the estimates. In this sense, an efficient estimator makes the most use of the information available in the data.

The above development gives no guarantee that an efficient estimator exists for every problem. When an efficient estimator does exist, it is also a uniformly minimum variance unbiased estimator. It is much easier to check for equality in Equation (4.2-20) than to directly prove that no other unbiased estimator has a smaller variance than a given estimator. The Cramer-Rao inequality is therefore useful as a sufficient (but not necessary) check that an estimator is uniformly minimum variance unbiased.

A useful alternative expression for the information matrix M can be obtained if $p(Z|\xi)$ is sufficiently smooth. Applying Equation (4.2-13) to the definition of M (Equation (4.2-19)) gives

$$M(\xi) = E\left\{\frac{(\nabla_\xi^* p(Z|\xi))(\nabla_\xi p(Z|\xi))}{p(Z|\xi)^2}\bigg|\xi\right\} \tag{4.2-21}$$

Then examine

$$\begin{aligned} E\{\nabla_\xi^2 \ln p(Z|\xi)|\xi\} &= E\left\{\nabla_\xi^* \frac{\nabla_\xi p(Z|\xi)}{p(Z|\xi)}\bigg|\xi\right\} \\ &= E\left\{\frac{\nabla_\xi^2 p(Z|\xi)}{p(Z|\xi)}\bigg|\xi\right\} - E\left\{\frac{(\nabla_\xi^* p(Z|\xi))\nabla_\xi p(Z|\xi)}{p(Z|\xi)^2}\bigg|\xi\right\} \end{aligned} \tag{4.2-22}$$

The second term is equal to $M(\xi)$, as shown in Equation (4.2-21). Evaluate the first term as

$$\int \frac{\nabla_\xi^2 p(Z|\xi)}{p(Z|\xi)} p(Z|\xi)d|Z| = \int \nabla_\xi^2 p(Z|\xi)d|Z| \tag{4.2-23}$$

$$= \nabla_\xi^2 \int p(Z|\xi)d|Z|$$

$$= \nabla_\xi^2 1 = 0 \tag{4.2-23}$$

Thus an alternate expression for the information matrix is

$$M(\xi) = -E\{\nabla_\xi^2 \ln p(Z|\xi)|\xi\} \tag{4.2-24}$$

4.2.4 Bayesian Optimal Estimators

The optimality conditions of the previous sections have been quite restrictive in that they must hold simultaneously for every possible value of ξ. Thus for some problems, no estimators exist that are optimal by these criteria. The Bayesian approach avoids this difficulty by using a single, overall, optimality criterion which averages the errors made for different values of ξ. With this approach, an optimal estimator may be worse than a nonoptimal one for specific values of ξ, but the overall averaged performance of the Bayesian optimal estimator will be better.

The Bayesian approach requires that a loss function (risk function, optimality criterion) be defined as a function of the true value ξ and the estimate $\hat{\xi}$. The most common loss function is a weighted square error

$$J(\xi,\hat{\xi}) = (\xi - \hat{\xi})^* R(\xi - \hat{\xi}) \tag{4.2-25}$$

where R is a weighting matrix. An estimator is considered optimal in the Bayesian sense if it minimizes the *a posteriori* expected value of the loss function:

$$\begin{aligned} E\{J(\xi,\hat{\xi}(Z))|Z\} &= \int J(\xi,\hat{\xi}(Z))p(\xi|Z)d|\xi| \\ &= \frac{\int J(\xi,\hat{\xi}(Z))p(Z|\xi)p(\xi)d|\xi|}{p(Z)} \end{aligned} \tag{4.2-26}$$

An optimal estimator must minimize this expected value for each Z. Since $P(Z)$ is not a function of $\hat{\xi}$, it does not affect the minimization of Equation (4.2-26) with respect to $\hat{\xi}$. Thus a Bayesian optimal estimator also minimizes the expression

$$\int J(\xi,\hat{\xi}(Z))p(Z|\xi)p(\xi)d|\xi| \tag{4.2-27}$$

Note that $p(\xi)$, the probability density of ξ, is required in order to define Bayesian optimality. For this purpose, $p(\xi)$ can be considered simply as a weighting that is part of the loss function, if it cannot appropriately be interpreted as a true probability density or an information function (Section 4.1).

4.2.5 Asymptotic Properties

Asymptotic properties concern the characteristics of the estimates as the amount of data used increases toward infinity. The amount of data used can increase either by repeating experiments or by increasing the time slice analyzed in a single experiment. (The latter is pertinent only for dynamic systems.) Since only a finite amount of data can be used in practice, it is not immediately obvious why there is any interest in asymptotic properties.

This interest arises primarily from considerations of simplicity. It is often simpler to compute asymptotic properties and to construct asymptotically optimal estimators than to do so for finite amounts of data. We can then use the asymptotic results as good approximations to the more difficult finite data results if the amount of data used is large enough. The finite data definitions of unbiased estimators and efficient estimators have direct asymptotic analogues of interest. An estimator is asymptotically unbiased if the bias goes to zero for all ξ as the amount of data goes to infinity. An estimator is asymptotically efficient if it is asymptotically unbiased and if

$$M(\xi)E\{(\hat{\xi} - \xi)(\hat{\xi} - \xi)^*|\xi\} \to I \tag{4.2-28}$$

as the amount of data approaches infinity. Equation (4.2-28) is an asymptotic expression for equality in Equation (4.2-20).

One important asymptotic property has no finite data analogue. This is the notion of consistency. An estimator is consistent if $\hat{\xi} \to \xi$ as the amount of data goes to infinity. For strong consistency, the convergence is required to be with probability one. Note that strong consistency is defined in terms of the convergence of individual realizations of the estimates, unlike the bias, variance, and other properties which are defined in terms of average properties (expected values).

Consistency is a stronger property than asymptotic unbiasedness; that is, all consistent estimators are asymptotically unbiased. This is a basic convergence result—that convergence with probability one implies convergence in distribution (and thus, specifically, convergence in mean). We refer the reader to Lipster and Shiryayev (1977), Cramér (1946), Goodwin and Payne (1977), Zacks (1971), and Mehra and Lainiotis (1976) for this and other results on consistency. Results on consistency tend to involve careful mathematical arguments relating to different types of convergence.

We will not delve deeply into asymptotic properties such as consistency in this book. We generally feel that asymptotic properties, although theoretically intriguing, should be played down in practical application. Application of infinite-time results to finite data is an approximation, one that is sometimes useful, but sometimes gives completely misleading conclusions (see Section 8.2). The inconsistency should be evident in books that spend copious time arguing fine points of distinction between different kinds of convergence and then pass off application to finite data with cursory allusions to using large data samples.

Although we de-emphasize the "rigorous" treatment of asymptotic properties, some asymptotic results are crucial to practical implementation. This is not because of any improved rigor of the asymptotic results, but because the asymptotic results are often simpler, sometimes enough simpler to make the critical difference in usability. This is our primary use of asymptotic results: as simplifying approximations to the finite-time results. Introduction of complicated convergence arguments hides this essential role. The approximations work well in many cases and, as with most approximations, fail in some situations. Our emphasis in asymptotic results will center on justifying when they are appropriate and understanding when they fail.

4.3 COMMON ESTIMATORS

This section will define some of the commonly used general types of estimators. The list is far from complete; we mention only those estimators that will be used in this book. We also present a few general results characterizing the estimators.

4.3.1 *A posteriori* Expected Value

One of the most natural estimates is the *a posteriori* expected value. This estimate is defined as the mean of the posterior distribution.

$$\hat{\xi}(Z) = E\{\xi|Z\} = \int \xi p(\xi|Z)d|\xi| = \frac{\int \xi p(Z|\xi)p(\xi)d|\xi|}{\int p(Z|\xi)p(\xi)d|\xi|} \tag{4.3-1}$$

This estimator requires that $p(\xi)$, the prior density of ξ, be known.

4.3.2 Bayesian Minimum Risk

Bayesian optimality was defined in Section 4.2.4. Any estimator which minimizes the *a posteriori* expected value of the loss function is a Bayesian minimum risk estimator. (In general, there can be more than one such estimator for a given problem.) The prior distribution of ξ must be known to define Bayesian estimators.

Theorem 4.3-1 The *a posteriori* expected value (Section 4.3.1) is the unique Bayesian minimum risk estimator for the loss function

$$J(\xi,\hat{\xi}) = (\xi - \hat{\xi})^*R(\xi - \hat{\xi}) \tag{4.3-2}$$

where R is any positive definite symmetric matrix.

Proof A Bayesian minimum risk estimator must minimize

$$E\{J|Z\} = E\{(\xi - \hat{\xi}(Z))^*R(\xi - \hat{\xi}(Z))|Z\} \tag{4.3-3}$$

Since R is symmetric, the gradient of this function is

$$\nabla_{\hat{\xi}} E\{J|Z\} = -2E\{R(\xi - \hat{\xi}(Z))|Z\}^* \tag{4.3-4}$$

Setting this expression to zero gives

$$0 = R\, E\{\xi - \hat{\xi}(Z)|Z\} = R[E\{\xi|Z\} - \hat{\xi}(Z)] \tag{4.3-5}$$

Therefore

$$\hat{\xi}(Z) = E\{\xi|Z\} \tag{4.3-6}$$

is the unique stationary point of $E\{J|Z\}$. The second gradient is

$$\nabla^2_{\hat{\xi}} E\{J|Z\} = 2R > 0 \tag{4.3-7}$$

so the stationary point is the global minimum.

Theorem (4.3-1) applies only for the quadratic loss function of Equation (4.3-2). The following very similar theorem applies to a much broader class of loss functions, but requires the assumption that $p(\xi|Z)$ is symmetric about its mean. Theorem (4.3-1) makes no assumptions about $p(\xi|Z)$ except that it has finite mean and variance.

<u>Theorem 4.3-2</u> Assume that $p(\xi|Z)$ is symmetric about its mean for each Z; i.e.,

$$p_{\xi|Z}(\bar{\xi}(Z) + \xi|Z) = p_{\xi|Z}(\bar{\xi}(Z) - \xi|Z) \tag{4.3-8}$$

where $\bar{\xi}(Z)$ is the expected value of ξ given Z. Then the *a posteriori* expected value is the unique Bayesian minimum risk estimator for any loss function of the form

$$J(\xi,\hat{\xi}) = J_1(\xi - \hat{\xi}) \tag{4.3-9}$$

where J_1 is symmetric about 0 and is strictly convex.

<u>Proof</u> We need to demonstrate that

$$D(a) \equiv E\{J(\xi,\bar{\xi}(Z) + a|Z\} - E\{J(\xi,\bar{\xi}(Z)|Z\} > 0 \tag{4.3-10}$$

for all $a \neq 0$. Using Equation (4.3-9) and the definition of expectation

$$D(a) = \int P(\xi|Z)[J_1(\xi - \bar{\xi}(Z) - a) - J_1(\xi - \bar{\xi}(Z))]d|\xi| \tag{4.3-11}$$

Because of the symmetry of $p(\xi|Z)$, we can replace the integral in Equation (4.3-11) by an integral over the region

$$S = \{\xi:(\xi - \bar{\xi}(Z),a) \geq 0\} \tag{4.3-12}$$

giving

$$D(a) = \int_S P(\xi|Z)[J_1(\xi - \bar{\xi}(Z) - a) + J_1(\bar{\xi}(z) - \xi - a) - J_1(\xi - \bar{\xi}(Z)) - J_1(\bar{\xi}(Z) - \xi)]d|\xi| \tag{4.3-13}$$

Using the symmetry of J_1 gives

$$D(a) = \int_S P(\xi|Z)[J_1(\xi - \bar{\xi}(Z) - a) + J_1(\xi - \bar{\xi}(Z) + a) - 2J_1(\xi - \bar{\xi}(Z)]d|\xi| \tag{4.3-14}$$

By the strict convexity of J_1

$$J_1(\xi - \bar{\xi}(Z) - a) + J_1(\xi - \bar{\xi}(Z) + a) > 2J_1(\xi - \bar{\xi}(Z)) \tag{4.3-15}$$

for all $a \neq 0$. Therefore $D(a) > 0$ for all $a \neq 0$ as we desired to show.

Note that if J_1 is convex, but not strictly convex, theorem (4.3-2) still holds except for the uniqueness. Theorems (4.3-1) and (4.3-2) are two of the basic results in the theory of estimation. They motivate the use of *a posteriori* expected value estimators.

4.3.3 <u>Maximum *a posteriori* Probability</u>

The maximum *a posteriori* probability (MAP) estimate is defined as the mode of the posterior distribution (i.e., the value of ξ which maximizes the posterior density function). If the distribution is not unimodal, the MAP estimate may not be unique. As with the previously discussed estimators, the prior distribution of ξ must be known in order to define the MAP estimate.

The MAP estimate is equal to the *a posteriori* expected value (and thus to the Bayesian minimum risk for loss functions meeting the conditions of Theorem (4.3-2)) if the posterior distribution is symmetric about its mean and unimodal, since the mode and the mean of such distributions are equal. For nonsymmetric distributions, this equality does not hold.

The MAP estimate is generally much easier to calculate than the *a posteriori* expected value. The *a posteriori* expected value is (from Equation (4.3-1))

$$\hat{\xi}(Z) = \frac{\int \xi p(Z|\xi)p(\xi)d|\xi|}{\int p(Z)|\xi)p(\xi)d|\xi|} \tag{4.3-16}$$

This calculation requires the evaluation of two integrals over Ξ. The MAP estimate requires the maximization of

$$p(\xi|Z) = \frac{p(Z|\xi)p(\xi)}{p(Z)} \tag{4.3-17}$$

with respect to ξ. The $p(Z)$ is not a function of ξ, so the MAP estimate can also be obtained by

$$\hat{\xi}(Z) = \arg\max_{\xi} p(Z|\xi)p(\xi) \tag{4.3-18}$$

The "arg max" notation indicates that $\hat{\xi}$ is the value of ξ that maximizes the density function $p(Z|\xi)p(\xi)$. The maximization in Equation (4.3-18) is generally much simpler than the integrations in Equation (4.3-16).

4.3.4 Maximum Likelihood

The previous estimators have all required that the prior distribution of ξ be known. When ξ is not random or when its distribution is not known, there are far fewer reasonable estimators to choose from. Maximum likelihood estimators are the only type that we will discuss.

The maximum likelihood estimate is defined as the value of ξ which maximizes the likelihood functional $p(Z|\xi)$; in other words,

$$\hat{\xi}(Z) = \arg\max_{\xi} p(Z|\xi) \tag{4.3-19}$$

The maximum likelihood estimator is closely related to the MAP estimator. The MAP estimator maximizes $p(\xi|Z)$; heuristically we could say that the MAP estimator selects the most probable value of ξ, given the data. The maximum likelihood estimator maximizes $p(Z|\xi)$; i.e., it selects the value of ξ which makes the observed data most plausible. Although these may sound like two statements of the same concept, there are crucial differences. One of the most central differences is that maximum likelihood is defined whether or not the prior distribution of ξ is known.

Comparing Equation (4.3-18) with Equation (4.3-19) reveals that the maximum likelihood estimate is identical to the MAP estimate if $p(\xi)$ is a constant. If the parameter space Ξ has finite size, this implies that $p(\xi)$ is the uniform distribution. For infinite Ξ, such as R^n, there are no uniform distributions, so a strict equivalence cannot be established. If we relax our definition of a probability distribution to allow arbitrary density functions which need not integrate to 1 (sometimes called generalized probabilities), the equivalence can be established for any Ξ. Alternately, the uniform distribution for infinite size Ξ can be viewed as a limiting case of distributions with variance going to infinity (less and less prior certainty about the value of ξ).

The maximum likelihood estimator places no preference on any value of ξ over any other value of ξ; the estimate is solely a function of the data. The MAP estimate, on the other hand, considers both the data and the preference defined by the prior distribution.

Maximum likelihood estimators have many interesting properties, which we will cover later. One of the most basic is given by the following theorem:

> Theorem 4.3-3 If an efficient estimator exists for a problem, that estimator is a maximum likelihood estimator.
>
> Proof (This proof requires the use of the full notation for probability density functions to avoid confusion.) Assume that $\hat{\xi}(Z)$ is any efficient estimator. An estimator will be efficient if and only if equality holds in lemma (4.2-1). Equality holds if and only if $X = \Lambda Y$ in Equation (4.2-6). Substituting for Λ from Equation (4.2-8) gives
>
> $$X = E\{XY^*\}E\{YY^*\}^{-1}Y \tag{4.3-20}$$
>
> Substituting for X and Y as in the proof of the Cramer-Rao bound, and using Equations (4.2-18) and (4.2-19) gives
>
> $$\hat{\xi}(Z) - \xi = [I + \nabla_{\xi} b(\xi)]M(\xi)^{-1}\nabla_{\xi}^{*} \ln p_{Z|\xi}(Z|\xi) \tag{4.3-21}$$
>
> Efficient estimators must be unbiased, so $b(\xi)$ is zero and
>
> $$\hat{\xi}(Z) - \xi = M(\xi)^{-1}\nabla_{\xi}^{*} \ln p_{Z|\xi}(Z|\xi) \tag{4.3-22}$$
>
> For an efficient estimator, Equation (4.3-22) must hold for all values of Z and ξ. In particular, for each Z, the equation must hold for $\xi = \hat{\xi}(Z)$. The left-hand side is then zero, so we must have
>
> $$\nabla_{\xi}^{*} \ln p_{Z|\xi}(Z|\xi(Z)) = 0 \tag{4.3-23}$$

The estimate is thus at a stationary point of the likelihood functional. Taking the gradient of Equation (4.3-22)

$$-I = M(\xi)^{-1}\nabla_\xi^2 \ln p_{Z|\xi}(Z|\xi) - M(\xi)^{-1}[\nabla_\xi M(\xi)]M(\xi)^{-1}\nabla_\xi^* \ln p_{Z|\xi}(Z|\xi) \tag{4.3-24}$$

Evaluating this at $\xi = \hat{\xi}(Z)$, and using Equation (4.3-23) gives

$$-I = M(\hat{\xi}(Z))^{-1}\nabla_\xi^2 \ln p_{Z|\xi}(Z|\hat{\xi}(Z)) \tag{4.3-24}$$

Since M is positive definite, the stationary point is a local maximum. In fact, it is the only local maximum, because a local maximum at any point other than $\xi = \hat{\xi}(Z)$ would violate Equation (4.3-22). The requirement for $\int p_{Z|\xi}(Z|\xi)d|Z|$ to be finite implies that $p_{Z|\xi}(Z|\xi) \to 0$ as $|Z| \to \infty$, so that the local maximum will be a global maximum. Therefore $\hat{\xi}(Z)$ is a maximum likelihood estimator.

<u>Corollary</u> All efficient estimators for a problem are equivalent (i.e., if an efficient estimator exists, it is unique).

This theorem and its corollary are not as useful as they might seem at first glance, because efficient estimators do not exist for many problems. Therefore, it is not always true that a maximum likelihood estimator is efficient. The theorem does apply to some simple problems, however, and motivates the more widely applicable asymptotic results which will be discussed later.

Maximum likelihood estimates have the following natural invariance property: let $\hat{\xi}$ be the maximum likelihood estimate of ξ; then $f(\hat{\xi})$ is the maximum likelihood estimate of $f(\xi)$ for any function f. The proof of this statement is trivial if f is invertible. Let $L_\xi(\xi,Z)$ be the likelihood functional of ξ for a given Z. Define

$$x = f(\xi) \tag{4.3-26}$$

Then the likelihood function of x is

$$L_x(x,Z) = L_\xi(f^{-1}(x),Z) \tag{4.3-27}$$

This is the crucial equation. By definition, the left-hand side is maximized by $x = \hat{x}$, and the right-hand side is maximized by $f^{-1}(x) = \hat{\xi}$. Therefore

$$\hat{x} = f(\hat{\xi}) \tag{4.3-28}$$

The extension to noninvertible f is straightforward—simply realize that $f^{-1}(x)$ is a set of values, rather than a single value. The same argument then still holds, regarding $L_x(x,Z)$ as a one-to-many function (set-valued function).

Finally, let us emphasize that, although maximum likelihood estimates are formally identical to MAP estimates with uniform prior distributions, there is a basic theoretical difference in interpretation. Maximum likelihood makes no statements about distributions of ξ, prior or posterior. Stating that a parameter has a uniform prior distribution is drastically different from saying that we have no information about the parameter. Several classic "paradoxes" of probability theory resulted from ignoring this difference. The paradoxes arise in transformations of variable. Let a scalar ξ have a uniform prior distribution, and let f be any continuous invertible function. Then, by Equation (3.4-1), $x = f(\xi)$ has the density function

$$p_x(x) = p_\xi(f^{-1}(x))|f^{-1}(x)| \tag{4.3-29}$$

which is not a uniform distribution on x (unless f is linear). Thus if we say that there is no prior information (uniform distribution) about ξ, then this gives us prior information (nonuniform distribution) about x, and vice versa. This apparent paradox results from equating a uniform distribution with the idea of "no information."

Therefore, although we can formally derive the equations for maximum likelihood estimators by substituting uniform prior distributions in the equations for MAP estimators, we must avoid misinterpretations. Fisher (1921, p. 326) discussed this subject at length:

> There would be no need to emphasize the baseless character of the assumptions made under the titles of inverse probability and BAYES' Theorem in view of the decisive criticism to which they have been exposed....I must indeed plead guilty in my original statement of the Method of Maximum Likelihood (9) to having based my argument upon the principle of inverse probability; in the same paper, it is true, I emphasized the fact that such inverse probabilities were relative only. That is to say, that while one might speak of one value of p as having an inverse probability three times that of another value of p, we might on no account introduce the differential element dp, so as to be able to say that it was three times as probable that p should lie in one rather than the other of two equal elements. Upon consideration, therefore, I perceive that the word probability is wrongly used in such a connection: probability is a ratio of frequencies, and about the frequencies of such values we can know nothing whatever. We must return to the actual fact that one value of p, of the frequency of which we know nothing, would yield the observed result three times as frequently as would another value of p. If we need a word to characterize this relative property of different values of p, I suggest

that we may speak without confusion of the likelihood of one value of p being thrice the likelihood of another, bearing always in mind that likelihood is not here used loosely as a synonym of probability, but simply to express the relative frequencies with which such values of the hypothetical quantity p would in fact yield the observed sample.

CHAPTER 5

5.0 THE STATIC ESTIMATION PROBLEM

In this chapter begins the application of the general types of estimators defined in Chapter 4 to specific problems. The problems discussed in this chapter are static estimation problems; that is, problems where time is not explicitly involved. Subsequent chapters on dynamic systems draw heavily on these static results. Our treatment is far from complete; it is easy to spend an entire book on static estimation alone (Sorenson, 1980). The material presented here was selected largely on the basis of relevance to dynamic systems.

We concentrate primarily on linear systems with additive Gaussian noise, where there are simple, closed-form solutions. We also cover nonlinear systems with additive Gaussian noise, which will prove of major importance in Chapter 8. Non-Gaussian and nonadditive noise are mentioned only briefly, except for the special problem of estimation of variance.

We will initially treat nonsingular problems, where we assume that all relevant distributions have density functions. The understanding and handling of singular and ill-conditioned problems then receive special attention. Singularities and ill-conditioning are crucial issues in practical application, but are insufficiently treated in much of the current literature. We also discuss partitioning of estimation problems, an important technique for simplifying the computational task and treating some singularities.

The general form of a static system model is

$$Z = Z(\xi, U, \omega) \tag{5.0-1}$$

We apply a known specific input U (or a set of inputs) to the system, and measure the response Z. The vector ω is a random vector contaminating the measured system response. We desire to estimate the value of ξ.

The estimators discussed in Chapter 4 require knowledge of the conditional distribution of Z given ξ and U. We assume, for now, that the distribution is nonsingular, with density $p(Z|\xi,U)$. If ξ is considered random, you must know the joint density $p(Z,\xi|U)$. In some simple cases, these densities might be given directly, in which case Equation (5.0-1) is not necessary; the estimators of Chapter 4 apply directly. More typically, $p(Z|\xi,U)$ is a complicated density which is derived from Equation (5.0-1) and $p(\omega|\xi,U)$. It is often reasonable to assume quite simple distributions for ω, independent of ξ and U. In this chapter, we will look at several specific cases.

5.1 LINEAR SYSTEMS WITH ADDITIVE GAUSSIAN NOISE

The simplest and most classic results are obtained for linear static systems with additive Gaussian noise. The system equations are assumed to have the form

$$Z = C(U)\xi + D(U) + G(U)\omega \tag{5.1-1}$$

For any particular U, Z is a linear combination of ξ, ω, and a constant vector. Note that there are no assumptions about linearity with respect to U; the functions C, D, and G can be arbitrarily complicated. Throughout this section, we omit the explicit dependence on U from the notation. Similarly, all distributions and expectations are implicitly understood to be conditioned on U.

The random noise vector ω is assumed to be Gaussian and independent of ξ. By convention, we will define the mean of ω to be 0, and the covariance to be identity. This convention does not limit the generality of Equation (5.1-1), for if ω has a mean m and a finite covariance FF^*, we can define $G_2 = GF$ and $D_2 = D + m$ to obtain

$$Z = C\xi + D_2 + G_2\omega_2 \tag{5.1-2}$$

where ω_2 has zero mean and identity covariance.

When ξ is considered as random, we will assume that its marginal (prior) distribution is Gaussian with mean m_ξ and covariance P.

$$p(\xi) = |2\pi P|^{-1/2} \exp\left\{-\frac{1}{2}(\xi - m_\xi)^* P^{-1}(\xi - m_\xi)\right\} \tag{5.1-3}$$

Equation (5.1-3) assumes that P is nonsingular. We will discuss the implications and handling of singular cases later.

5.1.1 Joint Distribution of Z and ξ

Several distributions which can be derived from Equation (5.1-1) will be required in order to analyze this system. Let us first consider $p(Z|\xi)$, the conditional density of Z given ξ. This distribution is defined whether ξ is random or not. If ξ is given, then Equation (5.1-1) is simply the sum of a constant vector and a constant matrix times a Gaussian vector. Using the properties of Gaussian distributions discussed in Chapter 3, we see that the conditional distribution of Z given ξ is Gaussian with mean and covariance.

$$E\{Z|\xi\} = C\xi + D \tag{5.1-4}$$

$$\mathrm{cov}\{Z|\xi\} = GG^* \tag{5.1-5}$$

Thus, assuming that GG^* is nonsingular,

$$p(Z|\xi) = |2\pi GG^*|^{-1/2} \exp\left\{-\frac{1}{2}(Z - C\xi - D)^*(GG^*)^{-1}(Z - C\xi - D)\right\} \tag{5.1-6}$$

If ξ is random, with marginal density given by Equation (5.1-3), we can also meaningfully define the joint distribution of Z and ξ, the conditional distribution of ξ given Z, and the marginal distribution of Z.

For the marginal distribution of Z, note that Equation (5.1-1) is a linear combination of independent Gaussian vectors. Therefore Z is Gaussian with mean and covariance

$$E(Z) = Cm_\xi + D \tag{5.1-7}$$

$$\mathrm{cov}(Z) = CPC^* + GG^* \tag{5.1-8}$$

For the joint distribution of ξ and Z, we now require the cross-correlation

$$E\{[Z - E(Z)][\xi - E(\xi)]^*\} = CP \tag{5.1-9}$$

The joint distribution of ξ and Z is thus Gaussian with mean and covariance

$$E\begin{bmatrix}\xi \\ \hline Z\end{bmatrix} = \begin{bmatrix} m_\xi \\ \hline Cm_\xi + D\end{bmatrix} \tag{5.1-10}$$

$$\mathrm{cov}\begin{bmatrix}\xi \\ \hline Z\end{bmatrix} = \left[\begin{array}{c|c} P & PC^* \\ \hline CP & CPC^* + GG^*\end{array}\right] \tag{5.1-11}$$

Note that this joint distribution could also be derived by multiplying Equations (5.1-3) and (5.1-6) according to Bayes rule. That derivation arrives at the same results for Equations (5.1-10) and (5.1-11), but is much more tedious.

Finally, we can derive the conditional distribution of ξ given Z (the posterior distribution of ξ) from the joint distribution of ξ and Z. Applying Theorem (3.5-9) to Equations (5.1-10) and (5.1-11), we see that the conditional distribution of ξ given Z is Gaussian with mean and covariance

$$E(\xi|Z) = m_\xi + PC^*(CPC^* + GG^*)^{-1}(Z - Cm_\xi - D) \tag{5.1-12}$$

$$\mathrm{cov}(\xi|Z) = P - PC^*(CPC^* + GG^*)^{-1}CP \tag{5.1-13}$$

Equations (5.1-12) and (5.1-13) assume that $CPC^* + GG^*$ is nonsingular. If this matrix is singular, the problem is ill-posed and should be restated. We will discuss the singular case later.

Assuming that P, GG^*, and $(C^*(GG^*)^{-1}C + P^{-1})$ are nonsingular, we can use the matrix inversion lemmas, (lemmas (A.1-3) and (A.1-4)), to put Equations (5.1-12) and (5.1-13) into forms that will prove intuitively useful.

$$E(\xi|Z) = m_\xi + (C^*(GG^*)^{-1}C + P^{-1})^{-1}C^*(GG^*)^{-1}(Z - Cm_\xi - D) \tag{5.1-14}$$

$$\mathrm{cov}(\xi|Z) = (C^*(GG^*)^{-1}C + P^{-1})^{-1} \tag{5.1-15}$$

We will have much occasion to contrast the form of Equations (5.1-12) and (5.1-13) with the form of Equations (5.1-14) and (5.1-15). We will call Equations (5.1-12) and (5.1-13) the covariance form because they are in terms of the uninverted covariances P and GG^*. Equations (5.1-14) and (5.1-15) are called the information form because they are in terms of the inverses P^{-1} and $(GG^*)^{-1}$, which are related to the amount of information. (The larger the covariance, the less information you have, and vice versa.) Equation (5.1-15) has an interpretation as addition of information: P^{-1} is the amount of prior information about ξ, and $C^*(GG^*)^{-1}C$ is the amount of information in the measurement; the total information after the measurement is the sum of these two terms.

5.1.2 *A Posteriori* Estimators

Let us first examine the three types of estimators that are based on the posterior distribution $p(\xi|Z)$. These three types of estimators are *a posteriori* expected value, maximum *a posteriori* probability, and Bayesian minimum risk.

We previously derived the expression for the *a posteriori* expected value in the process of defining the posterior distribution. Either the covariance or information form can be used. We will use the information form because it ties in with other approaches as will be seen below. The *a posteriori* expected value estimator is thus

$$\hat{\xi} = m_\xi + (C^*GG^*)^{-1}C + P^{-1})^{-1}C^*(GG^*)^{-1}(Z - Cm_\xi - D) \tag{5.1-16}$$

The maximum *a posteriori* probability estimate is equal to the *a posteriori* expected value because the posterior distribution is Gaussian (and thus unimodal and symmetric about its mean). This fact suggests an alternate derivation of Equation (5.1-16) which is quite enlightening. To find the maximum point of the posterior distribution of ξ given Z, write

$$\ln p(\xi|Z) = \ln p(Z|\xi) + \ln p(\xi) - \ln p(Z) \tag{5.1-17}$$

Expanding this equation using Equations (5.1-3) and (5.1-6) gives

$$\ln p(\xi|Z) = -\frac{1}{2}(Z - C\xi - D)^*(GG^*)^{-1}(Z - C\xi - D) - \frac{1}{2}(\xi - m_\xi)^*P^{-1}(\xi - m_\xi) + a(Z) \tag{5.1-18}$$

where $a(Z)$ is a function of Z only. Equation (5.1-18) shows the problem in its "least squares" form. We are attempting to choose ξ to minimize $(\xi - m_\xi)$ and $(Z - C\xi - D)$. The matrices P^{-1} and $(GG^*)^{-1}$ are weightings used in the cost functions. The larger the value of $(GG^*)^{-1}$, the more importance is placed on minimizing $(Z - C\xi - D)$, and vice versa.

Obtain the estimate $\hat{\xi}$ by setting the gradient of Equation (5.1-18) to zero, as suggested by Equation (3.5-17).

$$0 = C^*(GG^*)^{-1}(Z - C\hat{\xi} - D) - P^{-1}(\hat{\xi} - m_\xi) \tag{5.1-19}$$

Write this as

$$0 = C^*(GG^*)^{-1}(Z - Cm_\xi - D) - P^{-1}(\hat{\xi} - m_\xi) - C^*(GG^*)^{-1}C(\hat{\xi} - m_\xi) \tag{5.1-20}$$

and the solution is

$$\hat{\xi} = m_\xi + (C^*(GG^*)^{-1}C + P^{-1})^{-1}C^*(GG^*)^{-1}(Z - Cm_\xi - D) \tag{5.1-21}$$

assuming that the inverses exist. For Gaussian distributions, Equation (3.5-18) gives the covariance as

$$\mathrm{cov}(\xi|Z) = -[\nabla_\xi^2 \ln p(\xi|Z)]^{-1} = (C(GG^*)^{-1}C + P^{-1})^{-1} \tag{5.1-22}$$

Note that the second gradient is negative definite (and the covariance positive definite), verifying that the solution is a maximum of the posterior probability density function. This derivation does not require the use of matrix inversion lemmas, or the expression from Chapter 3 for the Gaussian conditional distribution. For more complicated problems, such as conditional distributions of N jointly Gaussian vectors, the alternate derivation as in Equations (5.1-17) to (5.1-22) is much easier than the straightforward derivation as in Equations (5.1-10) to (5.1-15).

Because of the symmetry of the posterior distribution, the Bayesian optimal estimate is also equal to the *a posteriori* expected value estimate if the Bayes loss function meets the criteria of Theorem (4.3-1).

We will now examine the statistical properties of the estimator given by Equation (5.1-16). Since the estimator is a linear function of Z, the bias is easy to compute.

$$\begin{aligned} b(\xi) &= E\{\hat{\xi}|\xi\} - \xi \\ &= E\{m_\xi + (C^*(GG^*)^{-1}C + P^{-1})^{-1}C^*(GG^*)^{-1}(Z - Cm_\xi - D)|\xi\} - \xi \\ &= m_\xi + (C^*(GG^*)^{-1}C + P^{-1})^{-1}C^*(GG^*)^{-1}[E\{Z|\xi\} - Cm_\xi - D] - \xi \\ &= m_\xi + (C^*(GG^*)^{-1}C + P^{-1})^{-1}C^*(GG^*)^{-1}(C\xi + D - Cm_\xi - D) - \xi \\ &= [I - (C^*(GG^*)^{-1}C + P^{-1})^{-1}C^*(GG^*)^{-1}C](m_\xi - \xi) \end{aligned} \tag{5.1-23}$$

The estimator is biased for all finite nonsingular P and GG^*. The scalar case gives some insight into this bias. If ξ is scalar, the factor in brackets in Equation (5.1-23) lies between 0 and 1. As GG^* decreases and/or P increases, the factor approaches 0, as does the bias. In this case, the estimator obtains less information from the initial guess of ξ (which has large covariance), and more information from the measurement (which has small covariance). If the situation is reversed, GG^* increasing and/or P decreasing, the bias becomes larger. In this case, the estimator shows an increasing predilection to ignore the measured response and to keep the initial guess of ξ.

The variance and mean square error are also easy to compute. The variance of $\hat{\xi}$ follows directly from Equations (5.1-16) and (5.1-5):

$$\begin{aligned} \mathrm{cov}(\hat{\xi}|\xi) &= (C^*(GG^*)^{-1}C + P^{-1})^{-1}C^*(GG^*)^{-1}GG^*(GG^*)^{-1}C(C^*(GG^*)^{-1}C + P^{-1})^{-1} \\ &= (C^*(GG^*)^{-1}C + P^{-1})^{-1}C^*(GG^*)^{-1}C(C^*(GG^*)^{-1}C + P^{-1})^{-1} \end{aligned} \tag{5.1-24}$$

The mean square error is then

$$\mathrm{mse}(\xi) = \mathrm{cov}(\hat{\xi}|\xi) + b(\xi)b(\xi)^* \tag{5.1-25}$$

which is evaluated using Equations (5.1-23) and (5.1-24).

The most obvious question to ask in relation to Equations (5.1-24) and (5.1-25) is how they compare with other estimators and with the Cramer-Rao bound. Let us evaluate the Cramer-Rao bound. The Fisher information matrix (Equation (4.2-19)) is easy to compute using Equation (5.1-6):

$$\begin{aligned} M &= E\{C^*(GG^*)^{-1}(Z - C\xi - D)(Z - C\xi - D)^*(GG^*)^{-1}C\} \\ &= C^*(GG^*)^{-1}GG^*(GG^*)^{-1}C = C^*(GG^*)^{-1}C \end{aligned} \tag{5.1-26}$$

Thus the Cramer-Rao bound for unbiased estimators is

$$\text{mse}(\hat{\xi}|\xi) \geq (C^*(GG^*)^{-1}C)^{-1} \tag{5.1-27}$$

Note that, for some values of ξ, the *a posteriori* expected value estimator has a lower mean-square error than the Cramer-Rao bound for unbiased estimators; naturally, this is because the estimator is biased. To compute the Cramer-Rao bound for an estimator with bias given by Equation (5.1-23), we need to evaluate

$$\begin{aligned} I + \nabla_\xi b(\xi) &= I + (C^*(GG^*)^{-1}C + P^{-1})^{-1}C^*(GG^*)^{-1}C - I \\ &= (C^*(GG^*)^{-1}C + P^{-1})^{-1}C^*(GG^*)^{-1}C \end{aligned} \tag{5.1-28}$$

The Cramer-Rao bound is then (from Equation (4.2-10))

$$\text{mse}(\hat{\xi}|\xi) \geq (C^*(GG^*)^{-1}C + P^{-1})^{-1}C^*(GG^*)^{-1}C(C^*(GG^*)^{-1}C + P^{-1})^{-1} \tag{5.1-29}$$

Note that the estimator does not achieve the Cramer-Rao bound except at the single point $\xi = m_\xi$. At every other point, the second term in Equation (5.1-25) is positive, and the first term is equal to the bound; therefore, the mse is greater than the bound.

For a single observation, we can say in summary that the *a posteriori* estimator is optimal Bayesian for a large class of loss functions, but it is biased and does not achieve the Cramer-Rao lower bound. It remains to investigate the asymptotic properties. The asymptotic behavior of estimators for static systems is defined in terms of N independent repetitions of the experiment, where N approaches infinity. We must first define the application of the *a posteriori* estimator to repeated experiments.

Assume that the system model is given by Equation (5.1-1), with ξ distributed according to Equation (5.1-3). Perform N experiments $U_1 \ldots U_N$. (It does not matter whether the U_i are distinct.) The corresponding system matrices are C_i, D_i, and $G_iG_i^*$, and the measurements are Z_i. The random noise ω_i is an independent, zero-mean, identity covariance, Gaussian vector for each i. The maximum *a posteriori* estimate of ξ is given by

$$\hat{\xi} = m_\xi + \left[\sum_{i=1}^{N} C_i^*(G_iG_i^*)^{-1}C_i + P^{-1}\right]^{-1} \sum_{i=1}^{N} C_i^*(G_iG_i^*)^{-1}(Z_i - C_im_\xi - D_i) \tag{5.1-30}$$

assuming that the inverses exist.

The asymptotic properties are defined for repetition of the same experiment, so we do not need the full generality of Equation (5.1-30). If $U_i = U_j$, $C_i = C_j$, $D_i = D_j$, and $G_i = G_j$ for all i and j, Equation (5.1-30) can be written

$$\hat{\xi} = m_\xi + [NC^*(GG^*)^{-1}C + P^{-1}]^{-1}C^*(GG^*)^{-1} \sum_{i=1}^{N} (Z_i - Cm_\xi - D) \tag{5.1-31}$$

Compute the bias, covariance, and mse of this estimate in the same manner as Equations (5.1-23) to (5.1-25):

$$b(\xi) = [I - (NC^*(GG^*)^{-1}C + P^{-1})^{-1}NC^*(GG^*)^{-1}C](m_\xi - \xi) \tag{5.1-32}$$

$$\text{cov}(\hat{\xi}|\xi) = [NC^*(GG^*)^{-1}C + P^{-1}]^{-1}NC^*(GG^*)^{-1}C[NC^*(GG^*)^{-1}C + P^{-1}]^{-1} \tag{5.1-33}$$

$$\text{mse}(\hat{\xi}|\xi) = \text{cov}(\hat{\xi}|\xi) + b(\xi)b(\xi)^* \tag{5.1-34}$$

The Cramer-Rao bound for unbiased estimators is

$$\text{mse}(\hat{\xi}|\xi) \geq (NC^*(GG^*)^{-1}C)^{-1} \tag{5.1-35}$$

As N increases, Equation (5.1-32) goes to zero, so the estimator is asymptotically unbiased. The effect of increasing N is exactly comparable to increasing $(GG^*)^{-1}$; as we take more and better quality measurements, the estimator depends more heavily on the measurements and less on its initial guess.

The estimator is also asymptotically efficient as defined by Equation (4.2-28) because

$$NC^*(GG^*)^{-1}C \text{ cov}(\hat{\xi}|\xi) \xrightarrow[N]{} I \tag{5.1-36}$$

$$NC^*(GG^*)^{-1}C \; b(\xi)b(\xi)^* \xrightarrow[N]{} 0 \tag{5.1-37}$$

5.1.3 Maximum Likelihood Estimator

The derivation of the expression for the maximum likelihood estimator is similar to the derivation of the maximum *a posteriori* probability estimator done in Equations (5.1-17) to (5.1-22). The only difference is that instead of $\ln p(\xi|Z)$, we maximize

$$\ln p(Z|\xi) = -\frac{1}{2}(Z - C\xi - D)^*(GG^*)^{-1}(Z - C\xi - D) + a(Z) \tag{5.1-38}$$

The only relevant difference between Equation (5.1-38) and Equation (5.1-18) is the inclusion of the term based on the prior distribution of ξ in Equation (5.1-18). (The $a(z)$ are also different, but this is of no consequence at the moment.) The maximum likelihood estimate does not make use of the prior distribution; indeed it does not require that such a distribution exist. We will see that many of the MLE results are equal to the MAP results with the terms from the prior distribution omitted.

Find the maximum point of Equation (5.1-38) by setting the gradient to zero.

$$0 = C^*(GG^*)^{-1}(Z - C\hat{\xi} - D) \tag{5.1-39}$$

The solution, assuming that $C^*(GG^*)^{-1}C$ is nonsingular, is given by

$$\hat{\xi} = (C^*(GG^*)^{-1}C)^{-1}C^*(GG^*)^{-1}(Z - D) \tag{5.1-40}$$

This is the same form as that of the MAP estimate, Equation (5.1-21), with P^{-1} set to zero.

A particularly simple case occurs when $C = I$ and $D = 0$. In this event, Equation (5.1-40) reduces to $\hat{\xi} = Z$.

Note that the expression $(C^*(GG^*)^{-1}C)^{-1}C^*(GG^*)^{-1}$ is a left-inverse of C; that is

$$[(C^*(GG^*)^{-1}C)^{-1}C^*(GG^*)^{-1}]C = I \tag{5.1-41}$$

We can view the estimator given by Equation (5.1-40) as a pseudo-inverse of the system given by Equation (5.1-1). Using both equations, write

$$\begin{aligned}\hat{\xi} &= (C^*(GG^*)^{-1}C)^{-1}C^*(GG^*)^{-1}(C\xi + D + G\omega - D)\\ &= \xi + (C^*(GG^*)^{-1}C)^{-1}C^*(GG^*)^{-1}G\omega\\ &= \xi + (C^*(GG^*)^{-1}C)^{-1}C^*G^{*-1}\omega\end{aligned} \tag{5.1-42}$$

Although we must use Equation (5.1-40) to compute $\hat{\xi}$ because ξ and ω are not known, Equation (5.1-42) is useful in analyzing and understanding the behavior of the estimator. One interesting point is immediately obvious from Equation (5.1-42): the estimate is simply the sum of the true value plus the effect of the contaminating noise ω. For the particular realization $\omega = 0$, the estimate is exactly equal to the true value. This property, which is not shared by the *a posteriori* estimators, is closely related to the bias. Indeed, the bias of the maximum likelihood estimator is immediately evident from Equation (5.1-42).

$$b(\xi) = E\{\hat{\xi}|\xi\} - \xi = 0 \tag{5.1-43}$$

The maximum likelihood estimate is thus unbiased. Note that Equation (5.1-32) for the MAP bias gives the same result if we substitute 0 for P^{-1}.

Since the estimator is unbiased, the covariance and mean square error are equal. Using Equation (5.1-42), they are given by

$$\begin{aligned}\mathrm{cov}(\hat{\xi}|\xi) = \mathrm{mse}(\hat{\xi}|\xi) &= (C^*(GG^*)^{-1}C)^{-1}C^*G^{*-1}G^{-1}C(C^*(GG^*)^{-1}C)^{-1}\\ &= (C^*(GG^*)^{-1}C)^{-1}\end{aligned} \tag{5.1-44}$$

We can also obtain this result from Equations (5.1-33) and (5.1-34) for the MAP estimator by substituting 0 for P^{-1}.

We previously computed the Cramer-Rao bound for unbiased estimators for this problem (Equation 5.1-27)). The mean square error of the maximum likelihood estimator is exactly equal to the Cramer-Rao bound. The maximum likelihood estimator is thus efficient and is, therefore, a minimum variance unbiased estimator. The maximum likelihood estimator is not, in general, Bayesian optimal. Bayesian optimality may not even be defined, since ξ need not be random.

The MLE results for repeated experiments can be obtained from the corresponding MAP equations by substituting zero for P^{-1} and m_ξ. We will not repeat these equations here.

5.1.4 Comparison of Estimators

We have seen that the maximum likelihood estimator is unbiased and efficient, whereas the *a posteriori* estimators are only asymptotically unbiased and efficient. On the other hand, the *a posteriori* estimators are Bayesian optimal for a large class of loss functions. Thus neither estimator emerges as an unchallenged favorite. The reader might reasonably expect some guidance as to which estimator to choose for a given problem.

The roles of the two estimators are actually quite distinct and well-defined. The maximum likelihood estimator does the best possible job (in the sense of minimum mean square error) of estimating the value of ξ based on the measurements alone, without prejudice (bias) from any preconceived guess about the value. The maximum likelihood estimator is thus the obvious choice when we have no prior information. Having no prior information is analogous to having a prior distribution with infinite variance; i.e., $P^{-1} = 0$. In this regard, examine Equation (5.1-16) for the *a posteriori* estimate as P^{-1} goes to zero. The limit is (assuming that $C^*(GG^*)^{-1}C$ is nonsingular)

$$\hat{\xi} = m_\xi + (C^*(GG^*)^{-1}C)^{-1}C^*(GG^*)^{-1}(Z - Cm_\xi - D)$$
$$= m_\xi - (C^*(GG^{**-1}C)^{-1}C^*(GG^*)^{-1}Cm_\xi + (C^*(GG^*)^{-1}C)^{-1}C^*(GG^*)^{-1}(Z - D)$$
$$= (C^*(GG^*)^{-1}C)^{-1}C^*(GG^*)^{-1}(Z - D) \qquad (5.1\text{-}45)$$

which is equal to the maximum likelihood estimate. The maximum likelihood estimate is thus a limiting case of an *a posteriori* estimator as the variance of the prior distribution approaches infinity.

The *a posteriori* estimate combines the information from the measurements with the prior information to obtain the optimal estimate considering both sources. This estimator makes use of more information and thus can obtain more accurate estimates, on the average. With this improved average accuracy comes a bias in favor of the prior estimate. If the prior estimate is good, the *a posteriori* estimate will generally be more accurate than the maximum likelihood estimate. If the prior estimate is poor, the *a posteriori* estimate will be poor. The advantages of the *a posteriori* estimators thus depend heavily on the accuracy of the prior estimate of the value.

The basic criterion in deciding whether to use an MAP or MLE estimator is whether you want estimates based only on the current data or based on both the current data and the prior information. The MLE estimate is based only on the current data, and the MAP estimate is based on both the current data and the prior distribution.

The distinction between the MLE and MAP estimators often becomes blurred in practical application. The estimators are closely related in numerical computation, as well as in theory. An MAP estimate can be an intermediate computational step to obtaining a final MLE estimate, or vice versa. The following paragraphs describe one of these situations; the other situation is discussed in Section 5.2.2.

It is quite common to have a prior guess of the parameters, but to desire an independent verification of the value based on the measurements alone. In this case, the maximum likelihood estimator is the appropriate tool in order to make the estimates independent of the initial guess.

A two-step estimation is often the most appropriate to obtain maximum insight into a problem. First, use the maximum likelihood estimator to obtain the best estimates based on the measurements alone, ignoring any prior information. Then consider the prior information in order to obtain a final best estimate based on both the measurements and the prior information. By this two-step approach, we can see where the information is coming from—the prior distribution, the measurements, or both sources. The two-step approach also allows the freedom to independently choose the methodology for each step. For instance, we might desire to use a maximum likelihood estimator for obtaining the estimates based on the measurements, but use engineering judgment to establish the best compromise between the prior expectations and the maximum likelihood results. This is often the best approach because it may be difficult to completely and accurately characterize the prior information in terms of a specific probability distribution. The prior information often includes heuristic factors such as the engineer's judgment of what would constitute reasonable results.

The theory of sufficient statistics (Ferguson, 1967; Cramer, 1946; and Fisher, 1921) is useful in the two-step approach if we desire to use statistical techniques for both steps. The maximum likelihood estimate and its covariance form a sufficient statistic for this problem. Although we will not go into detail here, if we know the maximum likelihood estimate and its covariance, we know all of the statistically useful information that can be extracted from the data. The specific application is that the *a posteriori* estimates can be written in terms of the maximum likelihood estimate and its covariance instead of as a direct function of the data. The following expression is easy to verify using Equations (5.1-16), (5.1-40), and (5.1-44):

$$\hat{\xi}_a = m_\xi + (Q^{-1} + P^{-1})^{-1}Q^{-1}(\hat{\xi}_{ML} - m_\xi) \qquad (5.1\text{-}46)$$

where $\hat{\xi}_a$ is the *a posteriori* estimate (Equation (5.1-16)), $\hat{\xi}_{ML}$ is the maximum likelihood estimate (Equation (5.1-40)), and Q is the covariance of the maximum likelihood estimate (Equation (5.1-44)). In this form, the relationship between the *a posteriori* estimate and the maximum likelihood estimate is plain. The prior distribution is the only factor which enters into the relationship; it has nothing directly to do with the measured data or even with what experiment was performed.

Equation (5.1-46) is closely related to the measurement-partitioning ideas of the next section. Both relate to combining data from two different sources.

5.2 PARTITIONING IN ESTIMATION PROBLEMS

Partitioning estimation problems has some of the same benefits as partitioning optimization problems. A problem half the size of the original typically takes well less than half the effort to solve. Therefore, we can often come out ahead by partitioning a problem into smaller subproblems. Of course, this trick only works if the solutions to the subproblems can easily be combined to give a solution to the original problem

Two kinds of partitioning applicable to parameter estimation problems are measurement partitioning and parameter partitioning. Both of these schemes permit easy combination of the subproblem solutions in some situations.

5.2.1 Measurement Partitioning

A problem with multiple measurements can often be partitioned into a sequence of subproblems processing the measurements one at a time. The same principle applies to partitioning a vector measurement into a series of scalar (or shorter vector) measurements; the only difference is notational.

The estimators under discussion are all based on $p(Z|\xi)$ or, for *a posteriori* estimators, $p(\xi|Z)$. We will initially consider measurement partitioning as a problem in factoring these density functions.

Let the measurement Z be partitioned into two measurements, Z_1 and Z_2. (Extensions to more than two partitions follow the same principles.) We would like to factor $p(Z|\xi)$ into separate factors dependent on Z_1 and Z_2. By Bayes' rule, we can always write

$$p(Z|\xi) = p(Z_2|Z_1,\xi)p(Z_1|\xi) \tag{5.2-1}$$

This form does not directly achieve the required separation because $p(Z_2|Z_1,\xi)$ involves both Z_1 and Z_2. To achieve the required separation, we introduce the requirement that

$$p(Z_2|Z_1,\xi) = p(Z_2|\xi) \tag{5.2-2}$$

We will call this the Markov criterion.

Heuristically, the Markov criterion assures that $p(Z_1|\xi)$ contains all of the useful information we can extract from Z_1. *Therefore, having computed* $p(Z_1|\xi)$ *at the measured value of* Z_1, *we have no further need* for Z_1. If the Markov criterion does not hold, then there are interactions that require Z_1 and Z_2 to be considered together instead of separately. For systems with additive noise, the Markov criterion implies that the noise in Z_2 is independent of that in Z_1. Note that this does not mean that Z_2 is independent of Z_1.

For systems where the Markov criterion holds, we can substitute Equation (5.2-2) into Equation (5.2-1) to get

$$p(Z|\xi) = p(Z_2|\xi)p(Z_1|\xi) \tag{5.2-3}$$

which is the desired factorization of $p(Z|\xi)$.

When ξ has a prior distribution, the factorization of $p(\xi|Z)$ follows from that of $p(Z|\xi)$.

$$p(\xi|Z) = \frac{p(Z|\xi)p(\xi)}{p(Z)} = \frac{p(Z_2|\xi)p(Z_1|\xi)p(\xi)}{p(Z)} \tag{5.2-4}$$

The mixing of Z_1 and Z_2 in the $p(Z)$ in the denominator is not important, because the denominator is merely a normalizing constant, independent of ξ. It will prove convenient to write Equation (5.2-4) in the form

$$p(\xi|Z) = \frac{p(Z_2|\xi)p(\xi|Z_1)}{p(Z_2|Z_1)} \tag{5.2-5}$$

Let us now consider measurement partition of an MAP estimator for a system with $p(\xi|Z)$ factored as in Equation (5.2-5). The MAP estimate is

$$\hat{\xi} = \arg\max_{\xi} p(Z_2|\xi)p(\xi|Z_1) \tag{5.2-6}$$

This equation is identical in form to Equation (4.3-18), with $p(\xi|Z_1)$ playing the role of the prior distribution. We have, therefore, the following two-step process for obtaining the MAP estimate by measurement partitioning:

First, evaluate the posterior distribution of ξ given Z_1. This is a function of ξ, rather than a single value. Practical application demands that this distribution be easily representable by a few statistics, but we put off such considerations until the next section. Then use this as the prior distribution for an MAP estimator with the measurement Z_2. Provided that the system meets the Markov criterion, the resulting estimate should be identical to that obtained by the unpartitioned MAP estimator.

Measurement partitioning of MLE estimator follows similar lines, except for some issues of interpretation. The MLE estimate for a system factored as in Equation (5.2-3) is

$$\hat{\xi} = \arg\max_{\xi} p(Z_2|\xi)p(Z_1|\xi) \tag{5.2-7}$$

This equation is identical in form to Equation (4.3-18), with $p(Z_1|\xi)$ playing the role of the prior distribution. The two steps of the partitioned MLE estimator are therefore as follows: first, evaluate $p(Z_1|\xi)$ at the measured value of Z_1, giving a function of ξ. Then use this function as the prior density for an MAP estimator with measurement Z_2. Provided that the system meets the Markov criterion, the resulting estimate should be identical to that obtained by the unpartitioned MLE estimator.

The partitioned MLE estimator raises an issue of interpretation of $p(Z_1|\xi)$. It is not a probability density function of ξ. The vector ξ need not even be random. We can avoid the issue of ξ not being random by using information terminology, considering $p(Z_1|\xi)$ *to represent the state of our knowledge of* ξ based on Z_1 instead of being a probability density function of ξ. Alternately, we can simply consider $p(Z_1|\xi)$ to be a function of ξ that arises at an intermediate step of computing the MLE estimate. The process described gives the correct MLE estimate of ξ, regardless of how we choose to interpret the intermediate steps.

The close connection between MAP and MLE estimators is illustrated by the appearance of an MAP estimator as a step in obtaining the MLE estimate with partitioned measurements. The result can be interpreted either as an MAP estimate based on the measurement Z_2 and the prior density $p(Z_1|\xi)$, or as an MLE estimate based on both Z_1 and Z_2.

5.2.2 Application to Linear Gaussian Systems

We now consider the application of measurement partitioning to linear systems with additive Gaussian noise. We will first consider the partitioned MAP estimator, followed by the partitioned MLE estimator.

Let the partitioned system be

$$Z_1 = C_1\xi + D_1 + G_1\omega_1 \tag{5.2-8a}$$

$$Z_2 = C_2\xi + D_2 + G_2\omega_2 \tag{5.2-8b}$$

where ω_1 and ω_2 are independent Gaussian random variables with mean 0 and covariance 1. The Markov criterion requires that ω_1 and ω_2 be independent for measurement partitioning to apply. The prior distribution of ξ is Gaussian with mean m_ξ and covariance P, and is independent of ω_1 and ω_2.

The first step of the partitioned MAP estimator is to compute $p(\xi|Z_1)$. We have previously seen that this is a Gaussian density with mean and covariance given by Equations (5.1-12) and (5.1-13). Denote the mean and covariance of $p(\xi|Z_1)$ by m_1 and P_1. Then, Equations (5.1-12) and (5.1-13) give

$$m_1 = m_\xi + PC_1^*(C_1PC_1^* + G_1G_1^*)^{-1}(Z_1 - C_1m_\xi - D_1) \tag{5.2-9}$$

$$P_1 = P - PC_1^*(C_1PC_1^* + G_1G_1^*)^{-1}C_1P \tag{5.2-10}$$

The second step is to compute the MAP estimate of ξ using the measurement Z_2 and the prior density $p(\xi|Z_1)$. This step is another application of Equation (5.1-12), using m_1 for m_ξ and P_1 for P. The result is

$$\hat{\xi} = m_2 = m_1 + P_1C_2^*(C_2P_1C_2^* + G_2G_2^*)^{-1}(Z_2 - C_2m_1 - D_2) \tag{5.2-11}$$

The $\hat{\xi}$ defined by Equation (5.2-11) is the MAP estimate. It should exactly equal the MAP estimate obtained by direct application of Equation (5.1-12) to the concatenated system. You can consider Equations (5.2-9) through (5.2-11) to be an algebraic rearrangement of the original Equation (5.1-12); indeed, they can be derived in such terms.

Example 5.2-1 Consider a system

$$Z = \xi + \omega$$

where ω is Gaussian with mean 0 and covariance 1, and ξ has a Gaussian prior distribution with mean 0 and covariance 1. We make two independent measurements of Z (i.e., the two samples of ω are independent) and desire the MAP estimate of ξ. Suppose the Z_1 measurement is 2 and the Z_2 measurement is -1.

Without measurement partitioning, we could proceed as follows: write the concatenated system

$$\begin{bmatrix} Z_1 \\ Z_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \xi + \begin{bmatrix} \omega_1 \\ \omega_2 \end{bmatrix}$$

Directly apply Equation (5.1-12) with $m_\xi = 0$, $P = 1$, $C = [1 \;\; 1]^*$, $D = 0$, $G = 1$, and $Z = [2, -1]^*$. The MAP estimate is then

$$\hat{\xi} = [1 \;\; 1] \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}^{-1} Z$$

$$= \frac{1}{3}(Z_1 + Z_2) = \frac{1}{3}$$

Now consider this same problem with measurement partitioning. To get $p(\xi|Z_1)$, apply Equations (5.2-9) and (5.2-10) with $m_\xi = 0$, $P = 1$, $C_1 = 1$, $D_1 = 0$, $G_1 = 1$, and $Z_1 = 2$.

$$m_1 = 1(2)^{-1}Z_1 = \frac{1}{2}Z_1 = 1$$

$$P_1 = 1 - 1(2)^{-1}1 = \frac{1}{2}$$

For the second step, apply Equation (5.2-11) with $m_1 = 1$, $P_1 = 1/2$, $C_2 = 1$, $D_2 = 0$, $G_2 = 1$, and $Z_2 = -1$.

$$\hat{\xi} = 1 + \frac{1}{2}\left(1\,\frac{1}{2}\right)^{-1}(Z_2 - 1) = \frac{1}{3}Z_2 + \frac{2}{3} = \frac{1}{3}$$

We see that the results of the two approaches are identical in this example, as claimed. Note that the partitioning removes the requirement to invert a 2-by-2 matrix, substituting two 1-by-1 inversions.

The computational advantages of using the partitioned form of the MAP estimator vary depending on numerous factors. There are numerous other rearrangements of Equations (5.1-12) and (5.1-13). The information form of Equations (5.1-14) and (5.1-15) is often preferable if the required inverses exist. The information form can also be used in the partitioned estimator, replacing Equations (5.2-9) through (5.2-11) with corresponding information forms. Equation (5.1-30) is another alternative, which is often the most efficient.

There is at least one circumstance in which a partitioned form is mandatory. This is when the data comes in two separate batches and the first batch of data must be discarded (for any of several reasons—perhaps unavailability of enough computer memory) before processing the second batch. Such circumstances occur regularly. Partitioned estimators are also particularly appropriate when you have already computed the estimate based on the first batch of data before receiving the second batch.

Let us now consider the partitioned MLE estimator. The first step is to compute $p(Z_1|\xi)$. Equation (5.1-38) gives a formula for $p(Z_1|\xi)$. It is immediately evident that the logarithm of $p(Z_1|\xi)$ is a quadratic form in ξ. Therefore, although $p(Z_1|\xi)$ need not be interpreted as a probability density function of ξ, it has the algebraic form of a Gaussian density function, except for an irrelevant constant multiplier. Applying Equations (3.5-17) and (3.5-18) gives the mean and covariance of this function as

$$m_1 = P_1C_1^*(G_1G_1^*)^{-1}(Z_1 - D_1) \tag{5.2-12}$$

$$P_1 = -[\nabla_\xi^2 \ln p(Z_1|\xi)]^{-1} = [C_1^*(G_1G_1^*)^{-1}C_1]^{-1} \tag{5.2-13}$$

The second step of the partitioned MLE estimator is identical to the second step of the partitioned MAP estimator. Apply Equation (5.2-11), using the m_1 and P_1 from the first step. For the partitioned MLE estimator, it is most natural (although not required) to use the information form of Equation (5.2-11), which is

$$\hat{\xi} = m_1 + P_2C_2^*(G_2G_2^*)^{-1}(Z_2 - C_2m_1 - D_2) \tag{5.2-14}$$

$$P_2 = [C_2^*(G_2G_2^*)^{-1}C_2 + P_1^{-1}]^{-1} \tag{5.2-15}$$

This form is more parallel to Equations (5.2-12) and (5.2-13).

Example 5.2-2 Consider a maximum likelihood estimator for the problem of Example 5.2-1, ignoring the prior distribution of ξ. To get the MLE estimate for the concatenated system, apply Equation (5.1-40) with $C = [1 \quad 1]^*$, $D = 0$, $G = 1$, and $Z = [2, -1]^*$.

$$\hat{\xi} = (2)^{-1}[1 \quad 1]Z = \frac{1}{2}(Z_1 + Z_2) = \frac{1}{2}$$

Now consider the same problem with measurement partitioning. For the first step, apply Equations (5.2-12) and (5.2-13) with $C_1 = 1$, $D_1 = 0$, $G_1 = 1$, and $Z_1 = 2$.

$$P_1 = [1(1)^{-1}]^{-1} = 1$$

$$m_1 = P_1(1)^{-1}(Z_1 - 0) = Z_1 = 2$$

For the second step, apply Equations (5.2-14) and (5.2-15) with $C_2 = 1$, $D_2 = 0$, $G_2 = 1$, and $Z_2 = -1$.

$$P_2 = [1(1)^{-1} + (1)^{-1}]^{-1} = \frac{1}{2}$$

$$\hat{\xi} = 2 + \frac{1}{2}(1)^{-1}(Z_2 - 2 - 0) = 1 + \frac{1}{2}Z_2 = \frac{1}{2}$$

The partitioned algorithm thus gives the same result as the original unpartitioned algorithm.

There is often confusion on the issue of the bias of the partitioned MLE estimator. This is an MLE estimate of ξ based on both Z_1 and Z_2. It is, therefore, unbiased like all MLE estimators for linear systems with additive Gaussian noise. On the other hand, the last step of the partitioned estimator is an MAP estimate based on Z_2, with a prior distribution described by m_1 and P_1. We have previously shown that MAP estimators are biased. There is no contradiction in these two viewpoints. The estimate is biased based on the measurement Z_2 alone, but unbiased based on Z_1 and Z_2.

Therefore, it is overly simplistic to universally condemn MAP estimators as biased. The bias is not always so clear an issue, but requires you to define exactly on what data you are basing the bias definition. The primary basis for deciding whether to use an MAP or MLE estimator is whether you want estimates based only on the current set of data, or estimates based on the current data and prior information combined. The bias merely reflects this decision; it does not give you independent help in deciding.

5.2.3 Parameter Partitioning

In parameter partitioning, we write the parameter vector ξ as a function of two (or more—the generalizations are obvious) smaller vectors ξ_1 and ξ_2.

$$\xi = f(\xi_1, \xi_2) \tag{5.2-16}$$

The function f must be invertible to obtain ξ_1 and ξ_2 from ξ, or the solution to the partitioned problem will not be unique. The simplest kind of partitions are those in which ξ_1 and ξ_2 are partitions of the ξ vector.

With the parameter ξ partitioned into ξ_1 and ξ_2, we have a partitioned optimization problem. Two possible solution methods apply. The best method, if it can be used, is generally to solve for ξ_1 in terms of ξ_2 (or vice versa) and substitute this relationship into the original problem. Axial iteration is another reasonable method if solutions for ξ_1 and ξ_2 are nearly independent so that few iterations are required.

5.3 LIMITING CASES AND SINGULARITIES

In the previous discussions, we have simply assumed that all of the required matrix inverses exist. We made this assumption to present some of the basic results without getting sidetracked on fine points. We will now take a comprehensive look at all of the singularities and limiting cases, explaining both the circumstances that give rise to the various special cases, and how to handle such cases when they occur.

The reader will recognize that most of the special cases are idealizations which are seldom literally true. We almost never know any value perfectly (zero covariance). Conversely, it is rare to have absolutely no information about the value of a parameter (infinite covariance). There are very few parameters that would not be viewed with suspicion if an estimate of, say, 10^{356} were obtained. These idealizations are useful in practice for two reasons. First, they avoid the necessity to quantify statements such as "virtually perfect" when the difference between virtually perfect and perfect is not of measurable consequence (although one must be careful: sometimes even an extremely small difference can be crucial). Second, numerical problems with finite arithmetic can be alleviated by recognizing essentially singular situations and treating them specially as though they were exactly singular.

We will address two kinds of singularities. The first kind of singularity involves Gaussian distributions with singular covariance matrices. These are perfectly valid probability distributions conforming to the usual definition. The distributions, however, do not have density functions; therefore the maximum *a posteriori* probability and maximum likelihood estimates cannot be defined as we have done. The singularity implies that the probability distribution is entirely concentrated on a subspace of the originally defined probability space. If the problem statement is redefined to include only the subspace, the restricted problem is nonsingular. You can also address this singularity by looking at limits as the covariance approaches the singular matrix, provided that the limits exist.

The second kind of singularity involves Gaussian variables with infinite covariance. Conceptually, the meaning of infinite covariance is easily stated—we have no information about the value of the variable (but we must be careful about generalizing this idea, particularly in nonlinear transformations—see the discussion at the end of Section 4.3.4). Unluckily, infinite covariance Gaussians do not fit within the strict definition of a probability distribution. (They cannot meet axiom 2 in Section 3.1.1.) For current purposes, we need only recognize that an infinite covariance Gaussian distribution can be considered as a limiting case (in some sense that we will not precisely define here) of finite covariance Gaussians. The term "generalized probability distribution" is sometimes used in connection with such limiting arguments. The equations which apply to the infinite covariance case are the limits of the corresponding finite covariance cases, provided that the limits exist. The primary concern in practice is thus how to compute the appropriate limits.

We could avoid several of the singularities by retreating to a higher level of abstraction in the mathematics. The theory can consistently treat Gaussian variables with singular covariances by replacing the concept of a probability density function with the more general concept of a Radon-Nikodym derivative. (A probability density function is a specific case of a Radon-Nikodym derivative.) Although such variables do not have probability density functions, they do have Radon-Nikodym derivatives with respect to appropriate measures. Substituting the more general and more abstract concept of σ-finite measures in place of probability measures allows strict definition of infinite covariance Gaussian variables within the same context.

This level of abstraction requires considerable depth of mathematical background, but changes little in the practical application. We can derive the identical computational methods at a lower level of abstraction. The abstract theory serves to place all of the theoretical results in a common framework. In many senses the general abstract theory is simpler than the more concrete approach; there are fewer exceptions and special cases to consider. In implementing the abstract theory, the same computational issues arise, but the simplified viewpoint can help indicate how to resolve these issues. Simply knowing that the problem does have a well-defined solution is a major aid to finding the solution.

The conceptual simplification gained by the abstract theory requires significantly more background than we assume in this book. Our emphasis will be on the computations required to deal with the singularities, rather than on the abstract theory. Royden (1968), Rudin (1974), and Lipster and Shiryayev (1977) treat such subjects as σ-finite measures and Radon-Nikodym derivatives.

We will consider two general computational methods for treating singularities. The first method is to use alternate forms of the equations which are not affected by the singularity. The covariance form (Equations (5.1-12) and (5.1-13)) and the information form (Equations (5.1-14) and (5.1-15)) of the posterior distribution are equivalent, but have different points of singularity. Therefore, a singularity in one form can often be handled simply by switching to the other form. This simple method fails if a problem statement has singularities in both forms. Also, we may desire to stick with a particular form for other reasons.

The second method is to partition the estimation problem into two parts: the totally singular part and the nonsingular part. This partitioning allows us to use one means of solving the singular part and another means of solving the nonsingular part; we then combine the partial solutions to give the final result.

5.3.1 Singular P

The first case that we will consider is singular P. A singular P matrix indicates that some parameter or linear combination of parameters is known perfectly before the experiment is performed. For instance, we might know that $\xi_1 = 5\xi_2 + 3$, even though ξ_1 and ξ_2 are unknown. In this case, we know the linear combination $\xi_1 - 5\xi_2$ exactly. The singular P matrix creates no problems if we use the covariance form instead of the information form. If we specifically desire to use the information form, we can handle the singularity as follows.

Since P is always symmetric, the range and the null space of P form an orthogonal decomposition of the space Ξ. The singular eigenvectors of P span the null space, and the nonsingular eigenvectors span the range. Use the eigenvectors to decompose the parameter estimation problem into the totally singular subproblem and the totally nonsingular subproblem. This is a parameter partitioning as discussed in Section 5.2. The totally singular subproblem is trivial because we know the exact solution when we start (by definition). Substitute the solution of the singular problem in the original problem and solve the nonsingular subproblem in the normal manner.

A specific implementation of this decomposition is as follows: let X_S be the matrix of orthonormal singular eigenvectors of P, and X_{NS} be the matrix of orthonormal nonsingular eigenvectors. Then define

$$\xi_S = X_S^*\xi \tag{5.3-1a}$$

$$\xi_{NS} = X_{NS}^*\xi \tag{5.3-1b}$$

The covariances of ξ_S and ξ_{NS} are

$$\mathrm{cov}(\xi_S) = X_S^*PX_S = 0 \tag{5.3-2a}$$

$$\mathrm{cov}(\xi_{NS}) = X_{NS}^*PX_{NS} = P_{NS} \tag{5.3-2b}$$

where P_{NS} is nonsingular. Write

$$\xi = X_{NS}\xi_{NS} + X_S\xi_S \tag{5.3-3}$$

Substitute Equation (5.3-3) into the original problem. Use the exactly known value of ξ_S, and restate the problem in terms of ξ_{NS} as the unknown parameter vector. Other decompositions derived from multiplying Equation (5.3-1) by nonsingular transformations can be used if they have advantages for specific situations.

We will henceforth assume that P is nonsingular. It is unimportant whether the original problem statement is nonsingular or we are working with the nonsingular subproblem.

The implementation above is defined in very general terms, which would allow it to be done as an automatic computer subroutine. In practice, we usually know the fact of and reason for the singularity beforehand and can easily handle it more concretely. If an equation gives an exact relationship between two or more variables which we know prior to the experiment, we solve the equation for one variable and remove that variable from the problem by substitution.

> Example 5.3-1 Assume that the output of a system is a known function of the applied force and moment
>
> $$Z = f(F,M)$$
>
> An unknown point force is applied at a known position r referred to the origin. We thus know that
>
> $$M = r \times F$$
>
> If F and M are both considered as unknowns, the P matrix is singular. But this singularity is readily removed by substituting for M in terms of F so that F is the only unknown.
>
> $$Z = f(F, r \times F) = f_2(F)$$

5.3.2 Singular GG*

The treatment of singular GG* is similar in principle to that of singular P. A singular GG* matrix implies that some measurement or combination of measurements is made perfectly (i.e., noise-free). The covariance form does not involve the inverse of GG*, and thus can be used with no difficulty when GG* is singular.

An alternate approach involves a sequential decomposition of the original problem into totally singular (GG* = 0) and nonsingular subproblems. The totally singular subproblem must be handled in the covariance form; the nonsingular subproblem can then be handled in either form. This is a measurement partitioning as described in Section 5.2. Divide the measurement into two portions, called the singular and the nonsingular measurements, Z_S and Z_{NS}. First ignore Z_S and find the posterior distribution of ξ given only Z_{NS}. Then use this result as the distribution prior to Z_S. We specifically implement this decomposition as follows:

For the first step of the decomposition, let X_{NS} be the matrix of nonsingular eigenvectors of GG*. Multiply Equation (5.1-1) on the left by X_{NS}^* giving

$$X_{NS}^{*}Z = X_{NS}^{*}C\xi + X_{NS}^{*}D + X_{NS}^{*}G\omega \tag{5.3-4}$$

Define

$$\left.\begin{aligned} Z_{NS} &= X_{NS}^{*}Z \\ C_{NS} &= X_{NS}^{*}C \\ D_{NS} &= X_{NS}^{*}D \\ G_{NS} &= X_{NS}^{*}G \end{aligned}\right\} \tag{5.3-5}$$

Equation (5.3-4) then becomes

$$Z_{NS} = C_{NS}\xi + D_{NS} + G_{NS}\omega \tag{5.3-6}$$

Note that $G_{NS}G_{NS}^{*}$ is nonsingular. Using the information form for the posterior distribution, the distribution of ξ conditioned on Z_{NS} is

$$m_{NS} = E\{\xi|Z_{NS}\} = m_{\xi} + (C_{NS}^{*}(G_{NS}G_{NS}^{*})^{-1}C_{NS} + P^{-1})^{-1}C_{NS}^{*}(G_{NS}G_{NS}^{*})^{-1}(Z_{NS} - C_{NS}m_{\xi} - D_{NS}) \tag{5.3-7a}$$

$$P_{NS} = \mathrm{cov}\{\xi|Z_{NS}\} = (C_{NS}^{*}(G_{NS}G_{NS}^{*})^{-1}C_{NS} + P^{-1})^{-1} \tag{5.3-7b}$$

For the second step, let X_S be the matrix of singular eigenvectors of GG^*. Corresponding to Equation (5.3-6) is

$$Z_S = C_S\xi + D_S + G_S\omega \tag{5.3-8}$$

where

$$\left.\begin{aligned} Z_S &= X_S^{*}Z \\ C_S &= X_S^{*}C \\ D_S &= X_S^{*}D \\ G_S &= X_S^{*}G = 0 \end{aligned}\right\} \tag{5.3-9}$$

Use Equation (5.3-7) for the prior distribution for this step. Since G_S is 0, we must use the covariance form for the posterior distribution, which reduces to

$$E\{\xi|Z\} = m_{NS} + P_{NS}C_S^{*}(C_SP_{NS}C_S^{*})^{-1}(Z_S - C_Sm_{NS} - D_S) \tag{5.3-10a}$$

$$\mathrm{cov}\{\xi|Z\} = P_{NS} + P_{NS}C_S^{*}(C_SP_{NS}C_S^{*})^{-1}C_SP_{NS} \tag{5.3-10b}$$

Equations (5.3-4), (5.3-6), (5.3-8), and (5.3-10) give an alternate expression for the posterior distribution of ξ given Z which we can use when GG^* is singular. It does require that $C_SP_{NS}C_S^{*}$ be nonsingular. This is a special case of the requirement that $CPC^* + GG^*$ be nonsingular, which we discuss later. It is interesting to note that the covariance (Equation (5.3-10b)) of the estimate is singular. Multiply Equation (5.3-10b) on the right by C_S^{*} and obtain

$$P_{NS}C_S^{*} - P_{NS}C_S^{*}(C_SP_{NS}C_S^{*})^{-1}C_SP_{NS}C_S^{*} = P_{NS}C_S^{*} - P_{NS}C_S^{*} = 0 \tag{5.3-11}$$

Therefore the columns of C_S^{*} are all singular eigenvectors of the covariance of the estimate.

5.3.3 Singular $CPC^* + GG^*$

The next special case that we will consider is when $CPC^* + GG^*$ is singular. Note first that this can happen only when GG^* is also singular, because CPC^* and GG^* are both positive semi-definite, and the sum of two such matrices can be singular only if both terms are singular. Since both GG^* and $CPC^* + GG^*$ are singular, neither the covariance form nor the information form circumvents the singularity. In fact, there is no way to circumvent this singularity. If $CPC^* + GG^*$ is singular, the problem is intrinsically ill-posed. The only solution is to restate the original problem.

If we examine what is implied by a singular $CPC^* + GG^*$, we will be able to see why it necessarily means that the problem is ill-posed, and what kinds of changes in the problem statement are required. Referring to Equation (5.1-8), we see that $CPC^* + GG^*$ is the covariance of the measurement Z. GG^* is the contribution of the measurement noise to this covariance, and CPC^* is the contribution due to the prior variance of ξ. If $CPC^* + GG^*$ is singular, we can exactly predict some part of the measured response. For this to occur, there must be neither measurement noise nor parameter uncertainty affecting that particular part of the response.

Clearly, there are serious mathematical difficulties in saying that we know exactly what the measured value will be before taking the measurement. At best, the measurement can agree with what we predicted, which adds no new information. If, however, there is any disagreement at all, even due to rounding error in the computations, there is an irresolvable contradiction—we said that we knew exactly what the value would be and we were wrong. This is one situation where the difference between almost perfect and perfect is extremely important. As $CPC^* + GG^*$ approaches singularity, the corresponding estimators diverge; we cannot talk about the limiting case because the estimators do not converge to a limit in any meaningful sense.

5.3.4 Infinite P

Up to this point, the special cases considered have all involved singular covariance matrices, corresponding to perfectly known quantities. The remaining special cases all concern limits as eigenvalues of a covariance matrix approach infinity, corresponding to total ignorance of the value of a quantity.

The first such special case to discuss is when an eigenvalue of P approaches infinity. The problem is much easier to discuss in terms of the information matrix P^{-1}. As an eigenvalue of P approaches infinity, the corresponding eigenvalue of P^{-1} approaches zero. At the limit, P^{-1} is singular. To be cautious, we should not speak of P^{-1} being singular but only of the limit as P^{-1} goes to a singularity, as it is not meaningful to say that P^{-1} is singular. Provided that we use the information form everywhere, all of the limits as P^{-1} goes to a singularity are well-behaved and can be evaluated simply by substituting the singular value for P^{-1}. Thus this singularity poses no difficulties in practice, as long as we avoid the use of expressions involving a noninverted P. As previously mentioned, the limit as P^{-1} goes to zero is particularly interesting and results in estimates identical to the maximum likelihood estimates. Using a singular P^{-1} is paramount to saying that there is no prior information about some parameter or set of parameters (or that we choose to discount any such information in order to obtain an independent check). There is no convenient way to decompose the problem so that the covariance form can be used with singular P^{-1} matrices.

The meaning of a singular P^{-1} is most clearly illustrated by some examples using confidence regions. A confidence region is the area where the probability density function (really a generalized probability density function here) is greater than or equal to some given constant. (See Chapter 11 for a more detailed discussion of confidence regions.) Let the parameter vector consist of two elements, ξ_1 and ξ_2. Assume that the prior distribution has mean zero and

$$P^{-1} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$$

The prior confidence regions are given by

$$p(\xi) \geq C_1$$

or equivalently

$$[\xi_1 \quad \xi_2]\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} \xi_1 \\ \xi_2 \end{bmatrix} \geq C_2$$

which reduces to

$$\xi_1^2 \leq C_2$$

where C_1 and C_2 are constants depending on the level of confidence desired. For current purposes, we are interested only in the shape of the confidence region, which is independent of the values of the constants. Figure (5.3-1) is a sketch of the shape. Note that this confidence region is a limiting case of an ellipse with major axis length going to infinity while the minor axis is fixed. This prior distribution gives information about ξ_1, but none about ξ_2.

Now consider a second example, which is identical to the first except that

$$P^{-1} = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

In this case, the prior confidence region is

$$[\xi_1 \quad \xi_2]\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}\begin{bmatrix} \xi_1 \\ \xi_2 \end{bmatrix} \leq C_2$$

or

$$\xi_1^2 + \xi_2^2 - 2\xi_1\xi_2 \leq C_2$$

or

$$(\xi_1 - \xi_2)^2 \leq C_2$$

Figure (5.3-2) is a sketch of the shape of this confidence region. In this case, the difference between ξ_1 and ξ_2 is known with some confidence, but there is no information about the sum $\xi_1 + \xi_2$. The singular eigenvectors of P^{-1} correspond to directions in the parameter space about which there is no prior knowledge.

5.3.5 Infinite GG^*

Corresponding to the case where P^{-1} approaches a singular point is the similar case where $(GG^*)^{-1}$ approaches a singularity. As in the case of singular P^{-1}, there are no computational problems. We can readily evaluate all of the limits simply by substituting the singular matrix for $(GG^*)^{-1}$. The information form avoids the use of a noninverted GG^*. A singular $(GG^*)^{-1}$ matrix would indicate that some measurement or linear combination of measurements had infinite noise variance, which is rather unlikely. The primary use of singular $(GG^*)^{-1}$ matrices in practice is to make the estimator ignore certain measurements if they are worthless or simply unavailable. It is mathematically cleaner to rewrite the system model so that the unused measurements are not included in the observation vector, but it is sometimes more convenient to simply use a singular $(GG^*)^{-1}$ matrix. The two methods give the same result. (Not having a measurement at all is equivalent to having one and ignoring it.) One interesting specific case occurs when $(GG^*)^{-1}$ approaches 0. This method then amounts to ignoring all of the measurements. As might be expected, the *a posteriori* estimate is then the same as the *a priori* estimate.

5.3.6 Singular $C^*(GG^*)^{-1}C + P^{-1}$

The final special case to be discussed is when the $C^*(GG^*)^{-1}C + P^{-1}$ in the information form approaches a singular value. Note that this can occur only if P^{-1} is also approaching a singularity. Therefore, the problem cannot be avoided by using the covariance form. If $C^*(GG^*)^{-1}C + P^{-1}$ is singular, it means that there is no prior information about a parameter or combination of parameters, and that the experiment added no such information. The difficulty, then, is that there is absolutely no basis for estimating the value of the singular parameter or combination. The system is referred to as being unidentifiable when this singularity is present. Identifiability is an important issue in the theory of parameter estimation. The easiest computational solution is to restate the problem, deleting the parameter in question from the list of unknowns. Essentially the same result comes from using a pseudo-inverse in Equation (5.1-14) (but see the discussion in Section 2.4.3 on the blind use of pseudo-inverses to "solve" such problems). Of course, the best alternative is often to examine why the experiment gave no information about the parameter, and to redesign the experiment so that a usable estimate can be obtained.

5.4 NONLINEAR SYSTEMS WITH ADDITIVE GAUSSIAN NOISE

The general form of the system equations for a nonlinear system with additive Gaussian noise is

$$Z = f(\xi,U) + G(U)\omega \tag{5.4-1}$$

As in the case of linear systems, we will define by convention the mean of ω to be zero and the covariance to be identity. If ξ is random, we will assume that it is independent of ω and has the distribution given by Equation (5.1-3).

5.4.1 Joint Distribution of Z and ξ

To define the estimators of Chapter 4, we need to know the distribution $P(Z|\xi,U)$. This distribution is easily derived from Equation (5.4-1). The expressions $f(\xi,U)$ and $G(U)$ are both constants if conditioned on specific values of ξ and U. Therefore we can apply the rules discussed in Chapter 3 for multiplication of Gaussian vectors by constants and addition of constants to Gaussian vectors. Using these rules, we see that the distribution of Z conditioned on ξ and U is Gaussian with mean $f(\xi,U)$ and covariance $G(U)G(U)^*$.

$$p(Z|\xi,U) = |2\pi G(U)G(U)^*|^{-1/2} \exp\left\{-\frac{1}{2}[Z - f(\xi,U)]^*[G(U)G(U)^*]^{-1}[Z - f(\xi,U)]\right\} \tag{5.4-2}$$

This is the obvious nonlinear generalization of Equation (5.1-6); the nonlinearity does not change the basic method of derivation.

If ξ is random, we will need to know the joint distribution $p(Z,\xi|U)$. The joint distribution is computed by Bayes rule

$$p(Z,\xi|U) = p(Z|\xi,U)p(\xi|U) \tag{5.4-3}$$

Using Equations (5.1-3) and (5.4-2) gives

$$p(Z,\xi|U) = [|2\pi P|\ |2\pi GG^*|]^{-1/2} \exp\left\{-\frac{1}{2}[Z - f(\xi,U)]^*[G(U)G(U)^*]^{-1}[Z - f(\xi,U)] - \frac{1}{2}[\xi - m_\xi]^*P^{-1}[\xi - m_\xi]\right\} \tag{5.4-4}$$

Note that $p(Z,\xi|U)$ is not, in general, Gaussian. Although Z conditioned on ξ is Gaussian, and ξ is Gaussian, Z and ξ need not be jointly Gaussian. This is one of the major differences between linear and nonlinear systems with additive Gaussian noise.

Example 5.4-1 Let Z and ξ be scalars, $P = 1$, $m_\xi = 0$, $G(U) = 1$, and $f(\xi,U) = \xi^2$. Then

$$p(Z|\xi,U) = (2\pi)^{-1/2} \exp\left\{-\frac{1}{2}(Z - \xi^2)^2\right\}$$

and

$$p(\xi|U) = (2\pi)^{-1/2} \exp\left\{-\frac{1}{2}\xi^2\right\}$$

This gives

$$p(Z,\xi|U) = (2\pi)^{-1} \exp\left\{-\frac{1}{2}[\xi^2 + (Z - \xi^2)^2]\right\}$$

The general form of a joint Gaussian distribution for two variables Z and ξ is

$$p(Z,\xi) = a \exp\{b\xi^2 + cZ^2 + dZ\xi\}$$

where a, b, c, and d are constants. The joint distribution of Z and ξ cannot be manipulated into this form because a ξ^* term appears in the exponent. Thus Z and ξ are not jointly Gaussian, even though Z conditioned on ξ is Gaussian and ξ is Gaussian.

Given Equation (5.4-4), we can compute the marginal distribution of Z, and the conditional distribution of ξ given Z from the equations

$$p(Z) = \int p(Z,\xi)d\xi \tag{5.4-5}$$

and

$$p(\xi|Z) = \frac{p(Z,\xi)}{p(Z)} \tag{5.4-6}$$

The integral in Equation (5.4-5) is not easy to evaluate in general. Since $p(Z,\xi)$ is not necessarily Gaussian, or any other standard distribution, the only general means of computing $p(Z)$ is to numerically integrate Equation (5.4-5) for a grid of Z values. If ξ and Z are vectors, this can be a quite formidable task. Therefore, we will avoid the use of $p(Z)$ and $P(\xi|Z)$ for nonlinear systems.

5.4.2 Estimators

The *a posteriori* expected value and Bayes optimal estimators are seldom used for nonlinear systems because their computation is difficult. Computation of the expected value requires the numerical integration of Equation (5.4-5) and the evaluation of Equation (5.4-6) to find the conditional distribution, and then the integration of ξ times the conditional distribution. Theorem (4.3-1) says that the Bayes optimal estimator for quadratic loss is equal to the *a posteriori* expected value estimator. The computation of the Bayes optimal estimates requires the same or equivalent multidimensional integrations, so Theorem (4.3-1) does not provide us with a simplified means of computing the estimates.

Since the posterior distribution of ξ need not be symmetric, the MAP estimate is not equal to the *a posteriori* expected value for nonlinear systems. The MAP estimator does not require the use of Equations (5.4-5) and (5.4-6). The MAP estimate is obtained by maximizing Equation (5.4-6) with respect to ξ. Since $p(Z)$ is not a function of ξ, we can equivalently maximize Equation (5.4-4). For general, nonlinear systems, we must do this maximization using numerical optimization techniques.

It is usually convenient to work with the logarithm of Equation (5.4-4). Since standard optimization conventions are phrased in terms of minimization, rather than maximization, we will state the problem as minimizing the negative of the logarithm of the probability density.

$$-\ln p(Z,\xi|U) = \frac{1}{2}[Z - f(\xi,U)]^*(GG^*)^{-1}[Z - f(\xi,U)] + \frac{1}{2}[\xi - m_\xi]^*P^{-1}[\xi - m_\xi] + \frac{1}{2}\ln[|2\pi P|\ |2\pi GG^*|] \tag{5.4-7}$$

Since the last term of Equation (5.4-7) is a constant, it does not affect the optimization. We can therefore define the cost functional to be minimized as

$$J(\xi) = \frac{1}{2}[Z - f(\xi,U)]^*(GG^*)^{-1}[Z - f(\xi,U)] + \frac{1}{2}[\xi - m_\xi]^*P^{-1}[\xi - m_\xi] \tag{5.4-8}$$

We have omitted the dependence of J on Z and U from the notation because it will be evaluated for specific Z and U in application; ξ is the only variable with respect to which we are optimizing. Equation (5.4-8) makes it clear that the MAP estimator is also a least-squares estimator for this problem. The $(GG^*)^{-1}$ and P^{-1} matrices are weightings on the squared measurement error and the squared error in the prior estimate of ξ, respectively.

For the maximum likelihood estimate we maximize Equation (5.4-2) instead of Equation (5.4-4). As in the case of linear systems, the maximum likelihood estimate is equal to the limit of the MAP estimate as P^{-1} goes to zero; i.e., the last term of Equation (5.4-8) is omitted.

For a single measurement, or even for a finite number of measurements, the nonlinear MAP and MLE estimators have none of the optimality properties discussed in Chapter 4. The estimates are neither unbiased, minimum variance, Bayes optimal, or efficient. When there are a large number of measurements, the differences from optimality are usually small enough to ignore for practical purposes The main benefits of the nonlinear MLE and MAP estimators are their relative ease of computation and their links to the intuitively attractive idea of least squares. These links give some reason to suspect that even if some of the assumptions about the noise distribution are questionable, the estimators still make sense from a nonstatistical viewpoint. The final practical judgment of an estimator is based on whether the estimates are adequate for their intended use, rather than on whether they are exactly optimum.

The extension of Equation (5.4-8) to multiple independent experiments is straightforward.

$$J(\xi) = \frac{1}{2}\sum_{i=1}^{N}[Z_i - f(\xi,U_i)]^*(GG^*)^{-1}[Z_i - f(\xi,U_i)] + \frac{1}{2}[\xi - m_\xi]^*P^{-1}[\xi - m_\xi] \tag{5.4-9}$$

where N is the number of experiments performed. The maximum likelihood estimator is obtained by omitting the last term. The asymptotic properties are defined as N goes to infinity. The maximum likelihood estimator can be shown to be asymptotically unbiased and asymptotically efficient (and thus also asymptotically minimum-variance unbiased) under quite general conditions. The estimator is also consistent. The rigorous proofs of these properties (Cramer, 1946), although not extremely difficult, are fairly lengthy and will not be presented here. The only condition required is that

$$\frac{1}{N}\sum_{i=1}^{N}[\nabla_\xi f(\xi,U_i)]^*(GG^*)^{-1}[\nabla_\xi f(\xi,U_i)]$$

converge to a positive definite matrix. Cramer (1946) also proves that the estimates asymptotically approach a Gaussian distribution.

Since the maximum likelihood estimates are asymptotically efficient, the Cramer-Rao inequality (Equation (4.2-20)) gives a good estimate of the covariance of the estimate for large N. Using Equation (4.2-19) for the information matrix gives

$$\begin{aligned}
M(\xi) &= \sum_{i=1}^{N} E\{[\nabla_\xi f(\xi,U_i)]^*(GG^*)^{-1}[Z - f(\xi,U_i)][Z - f(\xi,U_i)]^*(GG^*)^{-1}[\nabla_\xi f(\xi,U_i)]\} \\
&= \sum_{i=1}^{N} [\nabla_\xi f(\xi,U_i)]^*(GG^*)^{-1}E\{[Z - f(\xi,U_i)][Z - f(\xi,U_i)]^*\}(GG^*)^{-1}[\nabla_\xi f(\xi,U_i)] \\
&= \sum_{i=1}^{N} [\nabla_\xi f(\xi,U_i)]^*(GG^*)^{-1}GG^*(GG^*)^{-1}[\nabla_\xi f(\xi,U_i)] \\
&= \sum_{i=1}^{N} [\nabla_\xi f(\xi,U_i)]^*(GG^*)^{-1}[\nabla_\xi f(\xi,U_i)] \qquad (5.4\text{-}10)
\end{aligned}$$

The covariance of the maximum likelihood estimate is thus approximated by

$$\mathrm{cov}(\hat{\xi}|\xi) \approx \left\{\sum_{i=1}^{N}[\nabla_\xi f(\xi,U_i)]^*(GG^*)^{-1}[\nabla_\xi f(\xi,U_i)]\right\}^{-1} \qquad (5.4\text{-}11)$$

When ξ has a prior distribution, the corresponding approximation for the covariance of the posterior distribution of ξ is

$$\mathrm{cov}(\xi|Z) \approx \left\{\sum_{i=1}^{N}[\nabla_\xi f(\xi,U_i)]^*(GG^*)^{-1}[\nabla_\xi f(\xi,U_i)] + P^{-1}\right\}^{-1} \qquad (5.4\text{-}12)$$

5.4.3 Computation of the Estimates

The discussion of the previous section did not address the question of how to compute the MAP and ML estimates. Equation (5.4-9) (without the last term for the MLE) is the cost functional to minimize. Minimization of such nonlinear functions can be a difficult proposition, as discussed in Chapter 2.

Equation (5.4-9) is in the form of a sum of squares. Therefore the Gauss-Newton method is often the best choice of optimization method. Chapter 2 discussed the details of the Gauss-Newton method. The probabilistic background of Equation (5.4-9) allows us to apply the central limit theorem to strengthen one of the arguments used to support the Gauss-Newton method.

For simplicity, assume that all of the U_i are identical. Compare the limiting behavior of the two terms of the second gradient, as expressed by Equation (2.5-10). The term retained by the Gauss-Newton approximation is $N[\nabla_\xi f]^*(GG^*)^{-1}[\nabla_\xi f]$, which grows linearly with N. At the true value of ξ, $Z_i - f(\xi,U_i)$ is a Gaussian random variable with mean 0 and covariance GG^*. Therefore, the omitted term of the second gradient is a sum of independent, identically distributed, random variables with zero mean. By the central limit theorem, the variance of $1/N$ times this term goes to zero as N goes to infinity. Since $1/N$ times the retained term goes to a nonzero constant, the omitted term is small compared to the retained one for large N. This conclusion is still true if the U_i are not identical, as long as f and its gradients are bounded and the first gradient does not converge to zero.

This demonstrates that for large N the omitted term is small compared to the retained term if ξ is at the true value, and, by continuity, if ξ is sufficiently close to the true value. When ξ is far from the true value, the arguments of Chapter 2 apply.

5.4.4 Singularities

The singular cases which arise for nonlinear systems are basically the same as for linear systems and have similar solutions. Limits as P^{-1} or $(GG^*)^{-1}$ approach singular values pose no difficulty. Singular P or GG^* matrices are handled by reducing the problem to a nonsingular subproblem as in the linear case.

The one singularity which merits some additional discussion in the nonlinear case corresponds to singular

$$\sum_{i=1}^{N} C_i^*(GG^*)^{-1}C_i + P^{-1}$$

in the linear case. The equivalent matrix in the nonlinear case, if we use the Gauss-Newton algorithm, is given by

$$\nabla_\xi^2 J(\xi) \approx \sum_{i=1}^{N} [\nabla_\xi f(\xi,U_i)]^*(GG^*)^{-1}[\nabla_\xi f(\xi,U_i)] + P^{-1} \tag{5.4-13}$$

If Equation (5.4-13) is singular at the true value, the system is said to be unidentifiable. We discussed the computational problems of this singularity in Chapter 2. Even if the optimization algorithm correctly finds a unique minimum, Equation (5.4-11) indicates that the covariance of a maximum likelihood estimate would be very large. (The covariance is approximated by the inverse of a nearly singular matrix.) Thus the experimental data contain very little information about the value of some parameter or combination of parameters. Note that the covariance estimate is unrelated to the optimization algorithm; changes to the optimization algorithm might help you find the minimum, but will not change the properties of the resulting estimates. The singularity can be eliminated by using a prior distribution with a positive definite P^{-1}, but in this case, the estimated parameter values will be strongly influenced by the prior distribution, since the experimental data is lacking in information.

As with linear systems, unidentifiability is a serious problem. To obtain usable estimates, it is generally necessary to either reformulate the problem or redesign the experiment. With nonlinear systems, we have the additional difficulty of diagnosing whether identifiability problems are present or not. This difficulty arises because Equation (5.4-13) is a function of ξ and it is necessary to evaluate it at or near the minimum to ascertain whether the system is identifiable. If the system is not identifiable, it may be difficult for the algorithm to approach the (possibly nonunique) minimum because of convergence problems.

5.4.5 Partitioning

In both theory and computation, parameter estimation is much more difficult for nonlinear than for linear systems. Therefore, means of simplifying parameter estimation problems are particularly desirable for nonlinear systems. The partitioning ideas of Section 5.2 have this potential for some problems.

The parameter partitioning ideas of Section 5.2.3 make no linearity assumptions, and thus apply directly to nonlinear problems. We have little more to add to the earlier discussion of parameter partitioning except to say that parameter partitioning is often extremely important in nonlinear systems. It can make the critical difference between a tractable and an intractable problem formulation.

Measurement partitioning, as formulated in Section 5.2.1, is impractical for most nonlinear systems. For general nonlinear systems, the posterior density function $p(\xi|Z_1)$ will not be Gaussian or any other simple form. The practical application of measurement partitioning to linear systems arises directly from the fact that Gaussian distributions are uniquely defined by their mean and covariance.

The only practical method of applying measurement partitioning to nonlinear systems is to approximate the function $p(\xi|Z_1)$ (or $p(Z_1|\xi)$ for MLE estimates) by some simple form described by a few parameters. The obvious approximation in most cases is a Gaussian density function with the same mean and covariance. The exact covariance is difficult to compute, but Equations (5.4-11) and (5.4-12) give good approximations for this purpose.

5.5 MULTIPLICATIVE GAUSSIAN NOISE (ESTIMATION OF VARIANCE)

The previous sections of this chapter have assumed that the G matrix is known. The results are quite different when G is unknown because the noise multiplies G rather than adding to it.

For convenience, we will work directly with GG^* to avoid the necessity of taking matrix square roots. We compute the estimates of G by taking the positive semidefinite, symmetric-matrix square roots of the estimates of GG^*.

The general form of a nonlinear system with unknown G is

$$Z = f(\xi,U) + G(\xi,U)\omega \tag{5.5-1}$$

We will consider N independent measurements Z_i resulting from the experiments U_i. The Z_i are then independent Gaussian vectors with means $f(\xi,U_i)$ and covariances $G(\xi,U_i)G(\xi,U_i)^*$. We will use Equation (5.1-3) for the prior distributions of ξ. Bayes rule (Equation (5.4-3)) then gives us the joint distribution of ξ and the Z_i given the U_i. Equations (5.4-5) and (5.4-6) define the marginal distribution of Z and the posterior distribution of ξ given Z. The latter distributions are cumbersome to evaluate and thus seldom used.

Because of the difficulty of computing the posterior distribution, the *a posteriori* expected value and Bayes optimal estimators are seldom used. We can compute the maximum likelihood estimates minimizing the negative of the logarithm of the likelihood functional. Ignoring irrelevant constant terms, the resulting cost functional is

$$J(\xi) = \frac{1}{2}\sum_{i=1}^{N} \{[Z_i - f(\xi)]^*[G(\xi)G(\xi)^*]^{-1}[Z_i - f(\xi)] + \ln|G(\xi)G(\xi)^*|\} \tag{5.5-2}$$

or equivalently

$$J(\xi) = \frac{1}{2}\,\mathrm{trace}\left\{[G(\xi)G(\xi)^*]^{-1}\sum_{i=1}^{N}[Z_i - f(\xi)][Z_i - f(\xi)]^*\right\} + \frac{1}{2}N\ln|G(\xi)G(\xi)^*| \tag{5.5-3}$$

We have omitted the explicit dependence on U_i from the notation and assume that all of the U_i are identical. (The generalization to different U_i is easy and changes little of essence.) The MAP estimator minimizes a cost functional equal to Equation (5.5-2) plus the extra term $1/2[\xi - m_\xi]^*P^{-1}[\xi - m_\xi]$. The MAP estimate of GG^* is seldom used because the ML estimate is easier to compute and proves quite satisfactory.

We can use numerical methods to minimize Equation (5.5-2) and compute the ML estimates. In most practical problems, the following parameter partitioning greatly simplifies the computation required: assume that the ξ vector can be partitioned into independent vectors ξ_G and ξ_f such that

$$GG^* = GG^*(\xi_G)$$

$$f = f(\xi_f) \tag{5.5-4}$$

The partition ξ_f may be empty, in which case f is a constant (if ξ_G is empty we have a known GG^* matrix, and the problem reduces to that discussed in the previous section). Assume further that the GG^* matrix is completely unknown, except for the restriction that it be positive semidefinite.

Set the gradients of Equation (5.5-2) with respect to GG^* and ξ_f equal to zero in order to find the unconstrained minimum. Using the matrix differentiation results (A.2-5) and (A.2-6) from Appendix A, we get

$$\begin{aligned} 0 &= \nabla_{GG^*}J(\xi_f, GG^*) \\ &= -\frac{1}{2}(GG^*)^{-1}\sum_{i=1}^{N}[Z_i - f(\xi_f)][Z_i - f(\xi_f)]^*(GG^*)^{-1} + \frac{1}{2}N(GG^*)^{-1} \end{aligned} \tag{5.5-5}$$

$$0 = \nabla_{\xi_f}J(\xi_f, GG^*) = -\sum_{i=1}^{N}[Z_i - f(\xi_f)](GG^*)^{-1}[\nabla_{\xi_f}f(\xi_f)] \tag{5.5-6}$$

Equation (5.5-5) gives

$$\hat{GG}^* = \frac{1}{N}\sum_{i=1}^{N}[Z_i - f(\xi_f)][Z_i - f(\xi_f)]^* \tag{5.5-7}$$

which is the familiar sample second moment of the residuals. The estimate of GG^* from Equation (5.5-7) is always positive semidefinite. It is possible for this estimate to be singular, in which case we must use the techniques previously discussed for handling singular GG^* matrices. For a given ξ_f, Equation (5.5-7) is a simple noniterative estimator for GG^*. This closed-form expression is the reason for the partition of ξ into ξ_f and ξ_G.

We can constrain GG^* to be diagonal, in which case the solution is the diagonal elements of Equation (5.5-7). If we place other types of constraints on GG^*, such as knowledge of the values of individual off-diagonal elements, such simple closed-form solutions are not apparent. In practice, such constraints are seldom required.

If ξ_f is empty, Equation (5.5-7) is the solution to the problem. If ξ_f is not empty, we need to combine this subproblem solution with a solution for ξ_f to get a solution of the entire problem. Let us investigate the two methods discussed in Section 5.2.3.

The first method is axial iteration. Axial iteration involves successively estimating ξ_G with fixed ξ_f, and estimating ξ_f with fixed ξ_G. Equation (5.5-5) gives the ξ_G estimate in closed form for fixed ξ_f. To estimate ξ_f with fixed ξ_G, we must minimize Equation (5.5-2) with respect to ξ_f. Unless the system is linear, this minimization requires an iterative method. For fixed G, Equation (5.5-2) is in the form of a sum of squares and the Gauss-Newton method is an appropriate choice (in fact this subproblem is identical to the problem discussed in Section 5.4). We thus have an inner iteration within the outer axial iteration of ξ_f and ξ_G. In such situations, efficiency is often improved by terminating the inner iteration before it converges, inasmuch as the largest changes in the ξ_f estimates occur on the early inner iterations. After these early iterations, more can be gained by revising GG^* to reflect these large changes than by refining ξ_f. Since the estimates of ξ_f and GG^* affect one another, there is no point in obtaining extremely accurate estimates of ξ_f until GG^* is known to a corresponding accuracy. As Gauss (1809, p. 249) said concerning a different problem:

> It then can only be worth while to aim at the highest accuracy, when the final correction is to be given to the orbit to be determined. But as long as it appears probable that new observations will give rise to new corrections, it will be convenient to relax, more or less, as the case may be from extreme precision, if in this way, the length of the computations can be considerably diminished.

Exploiting this concept to its fullest suggests using only one iteration of the Gauss-Newton algorithm for the inner "iteration." In this case the inner iteration is no longer iterative, and the overall algorithm would be as follows:

1. Estimate GG^* using Equation (5.5-7) and the current guess of ξ_f.
2. Use one iteration of the Gauss-Newton algorithm to revise the estimate of ξ_f.
3. Repeat steps 1 and 2 until convergence.

In general, axial iteration is a very poor algorithm, as discussed in Chapter 2. The convergence is often extremely slow. Furthermore, the algorithm can converge to a point that is not a strict local minimum and yet give no hint of a problem. For this particular application, however, the performance of axial iteration borders on spectacular.

Let us consider, for a while, the alternative to axial iteration: substituting Equation (5.5-7) into Equation (5.5-3). This substitution gives

$$J(\xi_f) = \frac{1}{2} N \operatorname{trace}\{I\} + \frac{1}{2} N \ln \left| \frac{1}{N} \sum_{i=1}^{N} [Z_i - f(\xi_f)][Z_i - f(\xi_f)]^* \right| \tag{5.5-8}$$

The first term is irrelevant to the minimization, so we will redefine the cost function as

$$J(\xi_f) = \frac{1}{2} N \ln \left| \frac{1}{N} \sum_{i=1}^{N} [Z_i - f(\xi_f)][Z_i - f(\xi_f)]^* \right| \tag{5.5-9}$$

You may sometimes see this cost function written in the equivalent (for our purposes) form

$$J(\xi_f) = |\hat{G}\hat{G}^*| \tag{5.5-10}$$

Examine the gradient of Equation (5.5-9). Using the matrix differentiation results (A.2-3) and (A.2-6) from Appendix A, we obtain

$$\frac{\partial}{\partial \xi_f^{(j)}} J(\xi_f) = -\operatorname{tr}\left\{ \left[\frac{1}{N} \sum_{i=1}^{N} [Z_i - f(\xi_f)][Z_i - f(\xi_f)]^* \right]^{-1} \sum_{i=1}^{N} \frac{\partial}{\partial \xi_f^{(j)}} f(\xi_f)[Z_i - f(\xi_f)]^* \right\} \tag{5.5-11}$$

This is more compactly expressed as

$$\nabla_{\xi_f} J(\xi_f) = -\sum_{i=1}^{N} [Z_i - f(\xi_f)]^* (\hat{G}\hat{G}^*)^{-1} \left[\nabla_{\xi_f} f(\xi_f) \right] \tag{5.5-12}$$

which is exactly the same as Equation (5.5-6) evaluated at $G = \hat{G}$. Furthermore, the Gauss-Newton method used to solve Equation (5.5-6) is a good method for solving Equation (5.5-12) because

$$\nabla^2_{\xi_f} J(\xi_f) \approx \sum_{i=1}^{N} \left[\nabla_{\xi_f} f(\xi_f) \right]^* (\hat{G}\hat{G}^*)^{-1} \left[\nabla_{\xi_f} f(\xi_f) \right] \tag{5.5-13}$$

Equation (5.5-13) neglects the derivative of $\hat{G}\hat{G}^*$ with respect to ξ_f, but we can easily show that the term so neglected is even smaller than the term containing $\nabla^2 f(\xi_f)$, the omission of which we previously justified.

Therefore, axial iteration is identical to substitution of Equation (5.5-7) as a constraint. It seems likely that we could use this equality to make deductions about the geometry of the cost function and thence about the behavior of various algorithms. (Perhaps there may be some kind of orthogonality property buried here.) Several computer programs, including the Iliff-Maine MMLE3 code (Maine and Iliff, 1980; and Maine, 1981), use axial iteration, or a modification thereof, often with little more justification than that it seems to work well. This is, of course, the final and most important justification, but it is best used as verification of analytical arguments. Although Equations (5.5-12) and (5.5-13) are derived in standard texts, we have not seen the relationship between these equations and axial iteration pursued in the literature. It is plain that this equivalence relates to the excellent performance of axial iteration on this problem. We will leave further inquiry along this line to the reader.

An important special case of Equation (5.5-1) occurs when $f(\xi_f)$ is linear

$$f(\xi_f) = C\xi_f \tag{5.5-14}$$

with invertible C. For linear f, Equation (5.5-6) is solved exactly in a single Gauss-Newton iteration, and the solution is

$$\hat{\xi}_f = (C^*(GG^*)^{-1}C)^{-1}C^*(GG^*)^{-1}\frac{1}{N}\sum_{i=1}^{N} Z_i \tag{5.5-15}$$

If C is invertible, this reduces to

$$\hat{\xi}_f = C^{-1}\frac{1}{N}\sum_{i=1}^{N} Z_i \tag{5.5-16}$$

independent of GG^*. This is, of course, C^{-1} times the sample mean. Substituting Equations (5.5-14) and (5.5-16) into (5.5-15) gives

$$\hat{G}\hat{G}^* = \frac{1}{N}\sum_{i=1}^{N}\left[Z_i - \frac{1}{N}\sum_{i=1}^{N} Z_i\right]\left[Z_i - \frac{1}{N}\sum_{i=1}^{N} Z_i\right]^* \tag{5.5-17}$$

which is the familiar sample variance. Equation (5.5-17) can be manipulated into the alternate form

$$\hat{G}\hat{G} = \frac{1}{N}\left[\sum_{i=1}^{N} Z_i Z_i^* - \frac{1}{N}\left(\sum_{i=1}^{N} Z_i\right)\left(\sum_{i=1}^{N} Z_i^*\right)\right] \tag{5.5-18}$$

Because $\hat{\xi}_f$ is not a function of GG^*, the computation of $\hat{\xi}_f$ and $\hat{G}\hat{G}^*$ does not require iteration for this system model.

In general, the maximum likelihood estimates are asymptotically unbiased and efficient, but they need have no such properties for finite N. For linear invertible systems, the biases are easy to compute. From Equation (5.5-16),

$$E\{\hat{\xi}_f|\xi_f\} = C^{-1}\frac{1}{N}\sum_{i=1}^{N} C\xi_f = \xi_f \tag{5.5-19}$$

This equation shows that $\hat{\xi}_f$ is unbiased for finite N for linear invertible systems. From Equation (5.5-18), using the fact that ΣZ_i is Gaussian with mean $NC\xi_f$ and covariance NGG^*,

$$E\{\hat{G}\hat{G}^*|\xi\} = \frac{1}{N}\left[N(C\xi_f\xi_f^*C^* + GG^*) - \frac{1}{N}(N^2C\xi_f\xi_f^*C^* + NGG^*)\right] = \frac{N-1}{N}GG^* \tag{5.5-20}$$

Thus $\hat{G}\hat{G}^*$ is biased for finite N. Examining Equation (5.5-20), we see that the estimator defined by multiplying the ML estimate by $N/(N-1)$ is unbiased for finite N if $N > 1$. This unbiased estimate is often used instead of the maximum likelihood estimate. For large N, the difference is inconsequential.

In this discussion, we have assumed that both GG^* and ξ_f are unknown. If ξ_f is known, then the maximum likelihood estimator for GG^* is given by Equation (5.5-7) and this estimate is unbiased. The proof is left as an exercise. This result gives insight into the reasons for the bias of the estimator given by Equation (5.5-17). Note that Equations (5.5-17) and (5.5-7) are identical except that the sample mean is used in Equation (5.5-17) in place of the true mean in Equation (5.5-7). This substitution of the sample mean for the true mean has resulted in a bias.

The difference between the estimates from Equations (5.5-17) and (5.5-7) can be written in the form

$$\left[\frac{1}{N}\sum_{i=1}^{N} Z_i - f(\xi_f)\right]\left[\frac{1}{N}\sum_{i=1}^{N} Z_i - f(\xi_f)\right]^* \tag{5.5-21}$$

As this expression shows, the estimate of GG^* using the sample mean is less than or equal to the estimate using the true mean for every realization (i.e., the difference is positive semidefinite), equality occurring only when all of the Z_i are equal to $f(\xi_f)$. This is a stronger property than the bias difference; the bias difference implies only that the expected value using the sample mean is less.

5.6 NON-GAUSSIAN NOISE

Non-Gaussian noise is so general a classification that little can be said beyond the discussion in Chapter 4. The forms and properties of the estimators depend strongly on the types of noise distribution. The same comments apply to Gaussian noise if it is not additive or multiplicative, because the conditional distribution of Z given ξ is then non-Gaussian. In general, we apply the rules for transformation of variables to derive the conditional distribution of Z given ξ. Using this distribution, and the prior distribution of ξ if defined, we can derive the various estimators in principle.

The optimal estimators of Chapter 4 often require considerable computation for non-Gaussian noise. It is often possible to define much simpler estimators which have adequate performance. We will examine one situation where such simplification can occur.

Let the system model be linear with additive noise

$$Z = C\xi + \omega \tag{5.6-1}$$

The distribution of ω must have finite mean and variance independent of ξ, but is otherwise unrestricted. Call the mean m_ω and the variance GG^*. We will restrict ourselves to considering only linear estimators of the form

$$\hat{\xi} = K Z + D \tag{5.6-2}$$

Within this class, we will look for minimum-variance, unbiased estimators. We will require that the variance be minimized only over the class of unbiased linear estimators; there will be no guarantee that a smaller variance cannot be attained by a nonlinear estimator.

The bias of an estimator of the form of Equation (5.6-2) is

$$b(\xi) = E\{\hat{\xi}|\xi\} - \xi = KC\xi - \xi + D - Km_\omega \tag{5.6-3}$$

If the estimator is to be unbiased, we must have

$$D = Km_\omega \tag{5.6-4a}$$

$$KC = I \tag{5.6-4b}$$

The variance of an unbiased estimator of the given form is

$$\mathrm{var}(\hat{\xi}) = KGG^*K^* \tag{5.6-5}$$

Note that the bias and variance of the estimate depend only on the mean and variance of the noise distribution. The exact noise distribution need not even be known. If the noise distribution were Gaussian, a minimum-variance unbiased estimator would exist and be given by

$$\hat{\xi} = (C^*(GG^*)^{-1}C)^{-1}C^*(GG^*)^{-1}(Z_i - m_\omega) \tag{5.6-6}$$

This estimator is linear. Since no unbiased estimator, linear or not, can have a lower variance for the Gaussian case, this estimator is the minimum-variance, unbiased linear estimator for Gaussian noise. Since the bias and variance of a linear estimator depend only on the mean and variance of the noise, this is the minimum-variance, unbiased linear estimator for any noise distribution with the same mean and variance.

The optimality of this estimator can also be easily proven without reference to Gaussian distributions (although the above proof is complete and rigorous). Let

$$A = K - (C^*(GG^*)^{-1}C)^{-1}C^*(GG^*)^{-1} \tag{5.6-7}$$

for any K. Then

$$\begin{aligned} 0 \le AGG^*A^* &= KGG^*K^* + (C^*(GG^*)^{-1}C)^{-1}C^*(GG^*)^{-1}GG^*(GG^*)^{-1}C(C^*(GG^*)^{-1}C)^{-1} \\ &\quad - KGG^*(GG^*)^{-1}C(C^*(GG^*)^{-1}C)^{-1} \\ &\quad - (C^*(GG^*)^{-1}C)^{-1}C^*(GG^*)^{-1}GG^*K^* \\ &= KGG^*K^* + (C^*(GG^*)^{-1}C)^{-1} \\ &\quad - KC(C^*(GG^*)^{-1}C)^{-1} - (C^*(GG^*)^{-1}C)^{-1}C^*K^* \end{aligned} \tag{5.6-8}$$

Using Equation (4.6-4b) as a constraint on K, Equation (5.6-8) becomes

$$0 \le KGG^*K^* - (C^*(GG^*)^{-1}C)^{-1} \tag{5.6-9}$$

or, using Equation (5.6-5)

$$\mathrm{var}(\hat{\xi}) \ge (C^*(GG^*)^{-1}C)^{-1} \tag{5.6-10}$$

Thus no K satisfying Equation (5.6-4b) can achieve a variance lower than that given by Equation (5.6-10). The variance is equal to the minimum if and only if A is zero; that is if

$$K = (C^*(GG^*)^{-1}C)^{-1}C^*(GG^*)^{-1} \tag{5.6-11}$$

Therefore Equation (5.6-6) defines the unique minimum-variance, unbiased linear estimator. We are assuming that GG^* and $C^*(GG^*)^{-1}C$ are nonsingular; Section 5.3 discusses the singular cases.

In summary, if the system is linear with additive noise, and the estimator is required to be linear and unbiased, the results for Gaussian distributions apply to any distribution with the same mean and variance.

The use of optimal nonlinear estimators is seldom justifiable in view of the current state of the art. Although exceptional cases exist, three factors argue against using optimal nonlinear estimators. The first factor is the complexity and corresponding cost of deriving and implementing optimal nonlinear estimators. For some problems, we can construct fairly simple suboptimal nonlinear estimators that give better performance than the linear estimators (often by slightly modifying the linear estimator), but optimal nonlinear estimation is a difficult task.

The second factor is that linear estimators, perhaps slightly modified, often can give quite good estimates, even if they are not exactly optimal. Based on the central limit theorem, several results show that, under fairly general conditions, the linear estimates will approach the optimal nonlinear estimates as the number of samples increases. The precise conditions and proofs of these results are beyond the scope of this book.

The third factor is that we seldom have precise knowledge of the distribution anyway. The errors from inaccurate specification of the distribution are likely to be as large as the errors from using a suboptimal linear estimator. We need to consider this fact in deciding whether an optimal nonlinear estimator is really worth the cost. From Gauss (1809, p. 253)

> The investigation of an orbit having, strictly speaking, the maximum probability, will depend upon a knowledge of...[the probability distribution]; but that depends upon so many vague and doubtful considerations—physiological included—which cannot be subjected to calculation, that it is scarcely, and indeed less than scarcely, possible....

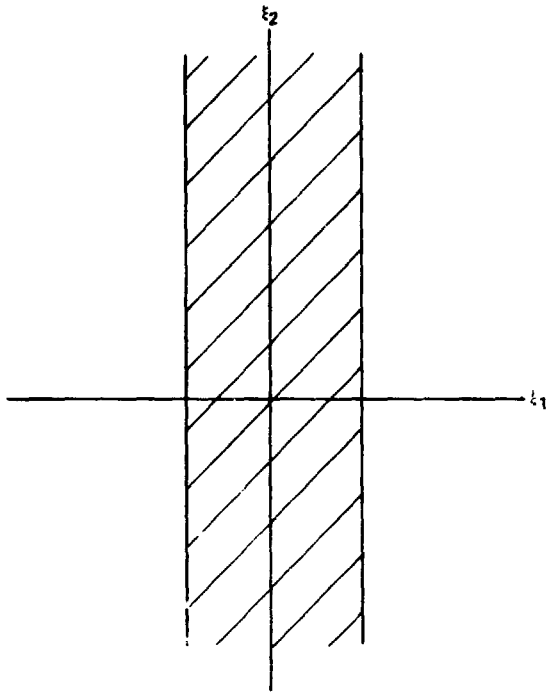


Figure (5.3-1). Confidence region with singular P^{-1}.

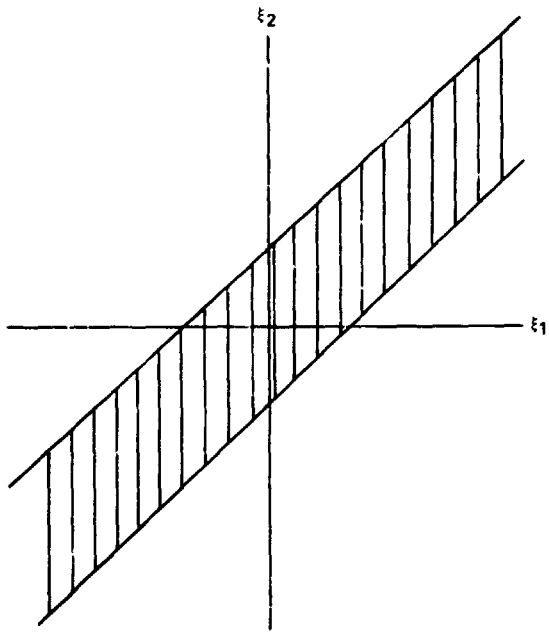


Figure (5.3-2). Confidence region with another singular P^{-1}.

CHAPTER 6

6.0 STOCHASTIC PROCESSES

In simplest terms, a stochastic process is a random variable that is a function of time. Thus stochastic processes are basic to the study of parameter estimation for dynamic systems. A complete and rigorous study of stochastic process theory requires considerable depth of mathematical background, particularly for continuous-time processes. For the purposes of this book, such depth of background is not required. Our approach does not draw heavily on stochastic process theory.

This chapter focuses on the few results that are needed for this document. Astrom (1970), Papoulis (1965), Lipster and Shiryayev (1977), and numerous other books give more complete treatments at varying levels of abstraction. The necessary results in this chapter are largely concerned with continuous-time models. Although we derive a few discrete-time equations in order to examine their continuous-time limits, the chapter can be omitted if you are studying only discrete-time analysis.

6.1 DISCRETE TIME

A discrete-time random process x is simply a collection of random variables x_i, one for each time point, defined on the same probability space. There can be a finite or infinite number of time points. The stochastic process is completely characterized by the joint distributions of all of the x_i. This can be a rather unwieldy means of characterizing the process, however, particularly if the number of time points is infinite.

If the x_i are jointly Gaussian, the process can be characterized by its first and second moments. Non-Gaussian processes are often also analyzed in terms of their first two moments because exact analyses are too complicated. The first two moments of the process x are

$$m(i) = E\{x_i\} \tag{6.1-1}$$

$$R(i,j) = E\{x_i x_j^*\} \tag{6.1-2}$$

The function $R(i,j)$ is called the autocorrelation function of the process.

A process is called stationary if the joint distribution of any collection of the x_i depends only on differences of the i values, not on the absolute time. This is called strict-sense stationarity. A process is stationary to second order or wide-sense stationary if the first moment is constant and the second moments depend only on time differences; i.e., if

$$R(i - k, j - k) = R(i,j) \tag{6.1-3}$$

for all i, j, and k. For Gaussian processes wide-sense stationarity implies strict-sense stationarity. The autocorrelation function of a wide-sense stationary process can be written as a function of one variable, the time difference.

$$R(k) = R(i, i + k) \tag{6.1-4}$$

A process is called white if x_i is independent of x_j for all $i \neq j$. Thus a Gaussian process is white if $R(i,j) = 0$ when $i \neq j$. Any process that is not white is called colored. A white process can be characterized by the distribution of x_i for each i. If a process is both white and stationary, the distribution of x_i is the same as that of x_j for all i and j, and this distribution is sufficient to characterize the process.

6.1.1 Linear Systems Forced by Gaussian White Noise

Our primary interest in this chapter is in the results of passing random signals through dynamic systems. We will first look at the simplest case, stationary white Gaussian noise passing through a linear system. The system equation is

$$x_{i+1} = \Phi x_i + F n_i \qquad i = 0,1,\ldots \tag{6.1-5}$$

where n is a stationary, Gaussian, white process with zero mean and identity covariance. The assumption of zero mean is made solely to simplify the equations. Results for nonzero mean can be obtained by linear superposition of the deterministic response to the mean and the stochastic response to the process with the mean removed. We are also given that x_0 is Gaussian with mean 0 and covariance P_0, and that x_0 is independent of the n_i.

The x_i form a stochastic process generated from the n_i. We desire to examine the properties of the stochastic process x. It is immediately obvious that x is Gaussian because x_i can be written as a linear combination of x_0 and $n_0, n_1, \ldots n_{i-1}$. In fact, the joint distribution of the x_i can be easily derived by explicitly writing this linear relation and using Theorem (3.5-5). We will leave this derivation as an exercise, and pursue instead a derivation using recursion along the lines that will be used in Chapter 7.

Assume we know that x_i has mean 0 and covariance P_i. Then the distribution of x_{i+1} follows immediately from Equation (6.1-5):

$$E\{x_{i+1}\} = \Phi E\{x_i\} + F E\{n_i\} = 0 \tag{6.2-6}$$

PRECEDING PAGE BLANK NOT FILMED

PAGE 68 INTENTIONALLY BLANK

$$E\{x_{i+1}x_{i+1}^*\} = \Phi E\{x_i x_i^*\}\Phi^* + FE\{n_i n_i^*\}F^* + \Phi E\{x_i n_i^*\}F^* + FE\{n_i x_i^*\}\Phi^* = \Phi P_i \Phi^* + FF^* \tag{6.1-7}$$

The cross terms in Equation (6.1-7) drop out because x_i is a function only of x_0 and $n_0, n_1, \ldots n_{i-1}$, all of which are independent of n_i by assumption. We now have a recursive formula for the covariance x_i

$$P_{i+1} = \Phi P_i \Phi^* + FF^* \qquad i = 0,1,\ldots \tag{6.1-8}$$

where P_0 is a given point from which we can start the recursion.

We know that the x_i are jointly Gaussian zero-mean variables with covariances given by the recursion (6.1-8). To complete the characterization of the joint distribution of the x_i, we need only the cross-covariances $E\{x_i x_j^*\}$ for $i \neq j$. Assume without loss of generality that $i > j$. Then x_i can be written as

$$x_i = \Phi^{i-j} x_j + \sum_{k=j}^{i-1} \Phi^{i-1-k} F n_k \tag{6.1-9}$$

Then

$$E\{x_i x_j^*\} = \Phi^{i-j} E\{x_j x_j^*\} + \sum_{k=j}^{i-1} \Phi^{i-1-k} FE\{n_k x_j^*\} = \Phi^{i-j} P \qquad i > j \tag{6.1-10}$$

The cross terms in Equation (6.1-10) are all zero by the same reasoning as used for Equation (6.1-7). For $i < j$, the same derivation (or transposition of the above result) gives

$$E\{x_i x_j^*\} = P_i (\Phi^*)^{j-1} \qquad i < j \tag{6.1-11}$$

This completes the derivation of the joint distribution of the x_i. Note that x is neither stationary nor white (except in special cases).

6.1.2 Nonlinear Systems and Non-Gaussian Noise

If the noise is not Gaussian, analyzing the system becomes much more difficult. Except in special cases, we then have to work with the probability distributions as functions instead of simply using the means and covariances. Similar problems arise for nonlinear systems or nonadditive noise even if the noise is Gaussian, because the distributions of the x_i will not then be Gaussian.

Consider the system

$$x_{i+1} = f(x_i, n_i) \qquad i = 0,1,\ldots \tag{6.1-12}$$

Assume that f has continuous partial derivatives almost everywhere, and can be inverted to obtain n_i (trivial if the noise is additive):

$$n_i = f^{-1}(x_i, x_{i+1}) \tag{6.1-13}$$

The n_i are assumed to be white and independent of x_0, but not necessarily Gaussian. Then the conditional distribution of x_{i+1} given x_i can be obtained from Equation (3.4-1)

$$p_{x_{i+1}|x_i}(x_{i+1}|x_i) = p_{n_i}(f^{-1}(x_i, x_{i+1}))|\det(J)| \tag{6.1-14}$$

where J is the Jacobian of the transformation f^{-1}. The joint distribution of $x_0, \ldots x_N$ can then be obtained from

$$p_x(x_0, \ldots x_N) = p_{x_0}(x_0) \prod_{i=1}^{N} p_{x_i|x_{i-1}}(x_i|x_{i-1}) \tag{6.1-15}$$

Equations (6.1-14) and (6.1-15) are, in general, too unwieldy to work with in practice. Practical work with nonlinear systems or non-Gaussian noise usually involves simplifying approximations.

6.2 CONTINUOUS TIME

We will look at continuous-time stochastic processes by looking at limits of discrete-time processes with the time interval going to 0. The discussion will focus on how to take the limit so that a useful result is obtained. We will not get involved in the intricacies of Ito or Stratanovich calculus (Astrom, 1970; Jazwinski, 1970; and Lipster and Shiryayev, 1977).

6.2.1 Linear Systems Forced by White Noise

Consider a linear continuous-time dynamic system driven by white, zero-mean noise

$$\dot{x}(t) = Ax(t) + F_c n(t) \tag{6.2-1}$$

We would like to look at this system as a limit (in some sense) of the discrete-time systems

$$x(t_i + \Delta) = (I + \Delta A)x(t_i) + \Delta F_c n(t_i) \tag{6.2-2}$$

as Δ, the time interval between samples, goes to zero. Equation (6.2-2) is in the form of Euler's method for approximating the solution of Equation (6.2-1). For the moment we will consider the discrete $n(t_i)$ to be Gaussian. The distribution of the $n(t_i)$ is not particularly important to the end result, but our argument is somewhat easier if the $n(t_i)$ are Gaussian. Equation (6.2-2) corresponds to Equation (6.1-5) with $I + A\Delta$ substituted for Φ, ΔF_c substituted for F, and some changes in notation to make the discrete and continuous notations more similar.

If n were a reasonably behaved deterministic process, we would get Equation (6.2-1) as a limit of Equation (6.2-2) when Δ goes to zero. For the stochastic system, however, the situation is quite different. Substituting $I + \Delta A$ for Φ and ΔF_c for F in Equation (6.1-8) gives

$$P(t_i + \Delta) = (I + \Delta A)P(t_i)(I + \Delta A)^* + \Delta^2 F_c F_c^* \tag{6.2-3}$$

Subtracting $P(t_i)$ and dividing by Δ gives

$$\frac{P(t_i + \Delta) - P(t_i)}{\Delta} = AP(t_i) + P(t_i)A^* + \Delta AP(t_i)A^* + \Delta F_c F_c^* \tag{6.2-4}$$

Thus in the limit

$$\dot{P}(t) = AP(t) + P(t)A \tag{6.2-5}$$

Note that F_c has completely dropped out of Equation (6.2-5). The distribution of x does not depend on the distribution of the forcing noise. In particular, if $P_0 = 0$, then $P(t) = 0$ for all t. The system simply does not respond to the forcing noise.

A model in which the system does not respond to the noise is not very useful. A useful model would be one that gives a finite nonzero covariance. Such a model is achieved by multiplying the noise by $\Delta^{-1/2}$ (and thus its covariance by Δ^{-1}). We rewrite Equation (6.2-2) as

$$x(t_i + \Delta) = (I + \Delta A)x(t_i) + \Delta^{1/2} F_c n(t_i) \tag{6.2-6}$$

The Δ in the $\Delta F_c F_c^*$ term of Equation (6.2-4) then disappears and the limit becomes

$$\dot{P}(t) = AP(t) + P(t)A^* + F_c F_c^* \tag{6.2-7}$$

Note that only a Δ^{-1} behavior of the covariance (or something asymptotic to Δ^{-1}) will give a finite nonzero result in the limit.

We will thus define the continuous-time white-noise process in Equation (6.2-1) as a limit, in some sense, of discrete-time processes with covariances Δ^{-1}. The autocorrelation function of the continuous-time process is

$$R(t,\tau) = E\{n(t)n(\tau)^*\} = \delta(t - \tau) \tag{6.2-8}$$

The impulse function $\delta(s)$ is zero for $x \neq 0$ and infinite for $s = 0$, and its integral over any finite range including the origin is 1. We will not go through the mathematical formalism required to rigorously define the impulse function—suffice it to say that the concept can be defined rigorously.

This model for a continuous-time white-noise process requires further discussion. It is obviously not a faithful representation of any physical process because the variance of $n(t)$ is infinite at every time point. The total power of the process is also infinite. The response of a dynamic system to this process, however, appears well-behaved.

The reasons for this apparently anomalous behavior are most easily understood in the frequency domain. The power spectrum of the process n is flat; there is the same power in every frequency band of the same width. There is finite power in any finite frequency range, but because the process has infinite bandwidth, the total power is infinite. Because any physical system has finite bandwidth, the system response to the noise will be finite. If, on the other hand, we kept the total power of the noise finite as we originally tried to do, the power in any finite frequency band would go to zero as we approached infinite bandwidth; thus, a physical system would have zero response.

The preceding paragraph explains why it is necessary to have infinite power in a meaningful continuous-time white-noise process. It also suggests a rationale for justifying such a model even though any physical noise source must have finite power. We can envision the physical noise as being band limited, but with a band limit much larger than the system band limit. If the noise band limit is large enough, its exact value is unimportant because the system response to inputs at a very high frequency is negligible. Therefore, we can analyze the system with white noise of infinite bandwidth and obtain results that are very good approximations to the finite-bandwidth results. The analysis is much simpler in the infinite-bandwidth white-noise model (even though some fairly abstract mathematics is required to make it rigorous). In summary, continuous-time white-noise is not physically realizable but can give results that are good approximations to physical systems.

6.2.2 Additive White Measurement Noise

We saw in the previous section that continuous-time white noise driving a dynamic system must have infinite power in order to obtain useful results. We will show in this section that the same conclusion applies to continuous-time white measurement noise.

We suppose that noise-corrupted measurements z are made of the system of Equation (6.2-1). The measurement equation is assumed to be linear with additive white noise:

$$z(t) = Cx(t) + G_c n(t) \tag{6.2-9}$$

For convenience, we will assume that the mean of the noise is 0. We then ask what else must be said about $n(t)$ in order to obtain useful results from this model.

Presume that we have measured $z(t)$ over the interval $0 < t < T$, and we want to estimate some characteristic of the system—say, $x(T)$. This is a filtering problem, which we will discuss further in Chapter 7. For current purposes, we will simplify the problem by assuming that $A = 0$ and $F = 0$ in Equation (6.2-1). Thus $x(t)$ is a constant over the interval, and dynamics do not enter the problem. We can consider this a static problem with repeated observations of a random variable, like those situations we covered in Chapter 5.

Let us look at the limit of the discrete-time equivalents to this problem. If samples are taken every Δ seconds, there are $\Delta^{-1}T$ total samples. Equation (5.1-31) is the MAP estimator for the discrete-time problem. The mean square error of the estimate is given by Equations (5.1-32) to (5.1-34). As Δ decreases to 0 and the number of samples increases to infinity, the mean square error decreases to 0. This result would imply that continuous-time estimates are always exact; it is thus not a very useful model. To get a useful model, we must let the covariance of the measurement noise go to infinity like Δ^{-1} as Δ decreases to 0. This argument is very similar to that used in the previous section. If the measurement noise had finite variance, each measurement would give us a finite amount of information, and we would have an infinite amount of information (no uncertainty) when the number of measurements was infinite. Thus the discrete-time equivalent of Equation (6.2-9) is

$$z(t_i) = Cx(t_i) + \Delta^{-1/2} G_c n(t_i) \tag{6.2-10}$$

where $n(t_i)$ has identity covariance.

Because any measurement is made using a physical device with a finite bandwidth, we stop getting much new information as we take samples faster than the response time of the instrument. In fact, the measurement equation is sometimes written as a differential equation for the instrument response instead of in the more idealized form of Equation (6.2-9). We need a noise model with a finite power in the bandwidth of the measurements because this is the frequency range that we are really working in. This argument is essentially the same as the one we used in the discussion of white noise forcing the system. The white noise can again be viewed as an approximation to band-limited noise with a large bandwidth. The lack of fidelity in representing very high-frequency characteristics is not too important, because high frequencies will tend to be filtered out when we operate on the data. (For instance, most operations on continuous-time data will have integrations at some point.) As a consequence of this modeling, we should be dubious of the practical application of any algorithm which results from this analysis and does not filter out high-frequency data in some manner.

We can generalize the conclusions in this and the previous section. Continuous-time white noise with finite variance is generally not a useful concept in any context. We will therefore take as part of the definition of continuous-time white noise that it have infinite covariance. We will use the spectral density rather than the covariance as a meaningful measure of the noise amplitude. White noise with autocorrelation

$$R(t,\tau) = G_c G_c^* \delta(t - \tau) \tag{6.2-11}$$

has spectral density $G_c G_c^*$.

6.2.3 Nonlinear Systems

As with discrete-time nonlinearities, exact analysis of nonlinear continuous-time systems is generally so difficult as to be impossible for most practical intents and purposes. The usual approach is to use a linearization of the system or some other approximation.

Let the system equation be

$$\dot{x}(t) = f(x,t) + g(x,t)n(t) \tag{6.2-12}$$

where n is zero-mean white noise with unity power, spectral density. For compactness of notation, let p represent the distribution of x at time t, given that x was x_0 at time t_0. The evolution of this distribution is described by the following parabolic partial differential equation:

$$\frac{\partial p}{\partial t} = -\sum_{i=1}^{n} \frac{\partial}{\partial x_i}(pf_i) + \frac{1}{2}\sum_{i,j,k=1}^{n} \frac{\partial^2}{\partial x_i \partial x_j}(pg_{ik}g_{jk}) \tag{6.2-13}$$

where n is the length of the x vector. The initial condition for this equation at $t = t_0$ is $p = \delta(x - x_0)$. See Jazwinski (1970) for the derivation of Equation (6.2-13). This equation is called the Fokker-Planck equation or the forward Kolmogorov equation. It is considered one of the basic equations of nonlinear filtering theory. In principle, this equation completely describes the behavior of the system and thus the problem is "solved." In practice, the solution of this multidimensional partial differential equation is usually too formidable to consider seriously.

CHAPTER 7

7.0 STATE ESTIMATION FOR DYNAMIC SYSTEMS

In this chapter, we address the estimation of the state of dynamic systems. The emphasis is on linear dynamic systems with additive Gaussian noise. We will initially develop the theory for discrete-time systems and then extend it to continuous-time and mixed continuous/discrete models.

The general form of a linear discrete-time system model is

$$x_{i+1} = \Phi x_i + \Psi u_i + F n_i \qquad i = 0,1,\ldots \tag{7.0-1a}$$

$$z_i = C x_i + D u_i + G n_i \qquad i = 1,2,\ldots \tag{7.0-1b}$$

The n_i and n_i are assumed to be independent Gaussian noise vectors with zero mean and identity covariance. The noise n is called process noise or state noise; n is called measurement noise. The input vectors, u_i, are assumed to be known exactly. The state of the system at the ith time point is x_i. The initial condition x_0 is a Gaussian random variable with mean m_0 and covariance P_0. (P_0 can be zero, meaning that the initial condition is known exactly.)

In general, the system matrices Φ, Ψ, F, C, D, and G can be functions of time. This chapter will assume that the system is time-invariant in order to simplify the notation. Except for the discussion of steady-state forms in Section 7.3, the results are easily generalized to time-varying systems by adding appropriate time subscripts to the matrices.

The state estimation problem is defined as follows: based on the measurements $z_1, z_2 \ldots z_N$, estimate the state x_M. To shorten the notation, we define

$$Z_N = (z_1, z_2 \ldots z_N)^* \tag{7.0-2}$$

State estimation problems are commonly divided into three classes, depending on the relationship of M and N.

If M is equal to N, the problem is called a filtering problem. Based on all of the measurements taken up to the current time, we desire to estimate the current state. This type of problem is typical of those encountered in real-time applications. It is the most widely treated one, and the one on which we will concentrate.

If M is greater than N, we have a prediction problem. The data are available up to the current time N, and we desire to predict the state at some future time M. We will see that once the filtering problem is solved, the prediction problem is trivial.

If M is less than N, the problem is called a smoothing problem. This type of problem is most commonly encountered in postexperiment batch processing in which all of the data are gathered before processing begins. In this case, the estimate of x_M can be based on all of the data gathered, both before and after time M. By using all values of M from 1 to N - 1, plus the filtered solution for M = N, we can construct the estimated state time history for the interval being processed. This is referred to as fixed-interval smoothing. Smoothing can also be used in a real-time environment where a few time points of delay in obtaining current state estimates is an acceptable price for the improved accuracy gained. For instance, it might be acceptable to gather data up to time N = M + 2 before computing the estimate of x_M. This is called fixed-lag smoothing. A third type of smoothing is fixed-point smoothing; in this case, it is desired to estimate x_M for a particular fixed M in a real-time environment, using new data to improve the estimate.

In all cases, x_N will have a prior distribution derived from Equation (7.0-1a) and the noise distributions. Since Equation (7.0-1) is linear in the noise, and the noise is assumed Gaussian, the prior and posterior distributions of x_N will be Gaussian. Therefore, the *a posteriori* expected value, MAP, and many Bayes' minimum risk estimators will be identical. These are the obvious estimators for a problem with a well-defined prior distribution. The remainder of the chapter assumes the use of these estimators.

7.1 EXPLICIT FORMULATION

By manipulating Equation (7.0-1) into an appropriate form, we can write the state estimation problem as a special case of the static estimation problem studied in Chapter 5. In this section, we will solve the problem by such manipulation; the fact that a dynamic system involved will thus play no special role in the meaning of the estimation problem. We will examine only the filtering problem here.

Our aim is to manipulate the state estimation problem into the form of Equation (5.1-1). The most obvious approach to this problem is to define the ξ of Equation (5.1-1) to be x_N, the vector which we desire to estimate. The observation, Z, would be a concatenation of $z_1,\ldots,z_N$; and the input, U, would be a concatenation of $u_0,\ldots,u_{N-1}$. The noise vector, ω, would then have to be a concatenation of $n_1,\ldots,n_{N-1}, n_1,\ldots n_N$. The problem can indeed be written in this manner. Unfortunately, the prior distribution of x_N is not independent of $n_0,\ldots,n_{N-1}$ (except for the case N = 0); therefore, Equation (5.1-16) is not the correct expression for the MAP estimate of x_N. Of course, we could derive an appropriate expression allowing for the correlation, but we will take an alternate approach which allows the direct use of Equation (5.1-16).

Let the unknown parameter vector be the concatenation of the initial condition and all of the process noise vectors.

$$\xi = [x_0, n_0, n_1, \ldots n_{N-1}]^* \tag{7.1-1}$$

The vector x_N, which we really desire to estimate, can be written as an explicit function of the elements of ξ; in particular, Equation (7.0-1a) expands into

$$x_N = \phi^N x_0 + \sum_{i=0}^{N-1} \phi^{N-1-i}(\Psi u_i + F n_i) \tag{7.1-2}$$

We can compute the MAP estimate of x_N by using the MAP estimates of x_0 and n_i in Equation (7.1-2). Note that we can freely treat the n_i as noise or as unknown parameters with prior distributions without changing the essential nature of the problem. The probability distribution of Z is identical in either case. The only distinction is whether or not we want estimates of the n_i. For this choice of ξ, the remaining items of Equation (5.1-1) must be

$$\begin{aligned} Z &= [z_1, z_2, \ldots, z_N]^* \\ U &= [u_0, u_1, \ldots, u_{N-1}]^* \\ \omega &= [n_1, n_2, \ldots, n_N]^* \end{aligned} \tag{7.1-3}$$

We get an explicit formula for z_i by substituting Equation (7.1-2) into Equation (7.0-1b), giving

$$z_i = C\phi^i x_0 + C \sum_{j=0}^{i-1} \phi^{i-1-j}(\Psi u_j + F n_j) + D u_i + G n_i \tag{7.1-4}$$

which can be written in the form of Equation (5.1-1) with

$$C(U) = \begin{bmatrix} C\phi & CF & 0 & \ldots & 0 & 0 \\ C\phi^2 & C\phi F & CF & \ldots & 0 & 0 \\ \vdots & \vdots & \vdots & \ldots & \vdots & \vdots \\ C\phi^{N-1} & C\phi^{N-2}F & C\phi^{N-3}F & \ldots & CF & 0 \\ C\phi^N & C\phi^{N-1}F & C\phi^{N-2}F & \ldots & C\phi F & CF \end{bmatrix} \tag{7.1-5a}$$

$$D(U) = \begin{bmatrix} C\Psi & D & 0 & \ldots & 0 & 0 \\ C\phi\Psi & C\Psi & D & \ldots & 0 & 0 \\ \vdots & \vdots & \vdots & \ldots & \vdots & \vdots \\ C\phi^{N-1}\Psi & C\phi^{N-2}\Psi & C\phi^{N-3}\Psi & \ldots & D & 0 \\ C\phi^N\Psi & C\phi^{N-1}\Psi & C\phi^{N-2}\Psi & \ldots & C\Psi & D \end{bmatrix} [U] \tag{7.1-5b}$$

$$G(U) = \begin{bmatrix} G & 0 & \ldots & 0 & 0 \\ 0 & G & \ldots & 0 & 0 \\ \vdots & \vdots & \ldots & \vdots & \vdots \\ 0 & 0 & \ldots & G & 0 \\ 0 & 0 & \ldots & 0 & G \end{bmatrix} \tag{7.1-5c}$$

You can easily verify these matrices by substituting them into Equation (5.1-1). The mean and covariance the prior distribution of ξ are

$$m_\xi = \begin{bmatrix} m_0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad P = \begin{bmatrix} P_0 & 0 & 0 & \ldots & 0 \\ 0 & I & 0 & \ldots & 0 \\ 0 & 0 & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ldots & \vdots \\ 0 & 0 & 0 & \ldots & I \end{bmatrix} \tag{7.1-6}$$

The MAP estimate of ξ is then given by Equation (5.1-16). The MAP estimate of x_N, which we seek, is obtained from that of ξ by using Equation (7.1-2).

The filtering problem is thus "solved." This solution, however, is unacceptably cumbersome. If the system state is an ℓ-vector, the inversion of an $(N + 1)\ell$-by-$(N + 1)\ell$ matrix is required in order to estimate x_N. The computational costs become unacceptable after a very few time points. We could investigate whether it is possible to take advantage of the structure of the matrices given in Equation (7.1-5) in order to simplify the computation. We can more readily achieve the same ends, however, by adopting a different approach to solving the problem from the start.

7.2 RECURSIVE FORMULATION

To find a simpler solution to the filtering problem than that derived in the preceding section, we need to take better advantage of the special structure of the problem. The above derivation used the linearity of the problem and the Gaussian assumption on the noise, which are secondary features of the problem structure. The fact that the problem involves a dynamic state-space model is much more basic, but was not used above to any special advantage; the first step in the derivation was to recast the system in the form of a static model. Let us reexamine the problem, making use of the properties of dynamic state-space systems.

The defining property of a state-space model is as follows: the future output is dependent only on the current state and the future input. In other words, provided that the current state of the system is known, knowledge of any previous states, inputs, or outputs, is irrelevant to the prediction of future system behavior; all relevant facts about previous behavior are subsumed in the knowledge of the current state. This is essentially the definition of the state of a system. The probabilistic expression of this idea is

$$p(z_N, z_{N+1} \cdots | x_N) = p(z_N, z_{N+1} \cdots | x_N, x_{N-1} \cdots u_{N-1}, u_{N-2} \cdots Z_{N-1}) \tag{7.2-1}$$

It is this property that allows the system to be described in a recursive form, such as that of Equation (7.0-1). The recursive form involves much less computation than the mathematically equivalent explicit form of Equation (7.1-4).

This reasoning suggests that recursion might be used to some advantage in obtaining a solution to the filtering problem. The estimators under consideration (MAP, etc.) are all defined from the conditional distribution of x_N given Z_N. We will seek a recursive expression for the conditional distribution, and thus for the estimates. We will prove that such an expression exists by deriving it.

In the nature of recursive forms, we start by assuming that the conditional distribution of x_N given z_N is known for some N, and then we attempt to derive an expression for the conditional distribution of x_{N+1} given Z_{N+1}. We recognize this task as similar to the measurement partitioning of Section 5.2.2, in that we want to simplify the solution by processing the measurements one at a time. Equations (5.2-2) and (7.2-1) express similar ideas and give the basis for the simplifications in both cases. (The x_N of Equation (7.2-1) corresponds to the ξ of Equation (5.2-2).)

Our task then is to derive $p(x_{N+1}|Z_{N+1})$. We will divide this task into two steps. First, derive $p(x_{N+1}|Z_N)$ from $p(x_N|Z_N)$. This is called the prediction step, because we are predicting x_{N+1} based on previous information. It is also called the time update because we are updating the estimate to a new time point based on the same data. The second step is to derive $p(x_{N+1}|Z_{N+1})$ from $p(x_{N+1}|Z_N)$. This is called the correction step, because we are correcting the predicted estimate of x_{N+1} based on the new information in z_{N+1}. It is also called the measurement update because we are updating the estimate based on the new measurement.

Since all of the distributions are assumed to be Gaussian, they are completely defined by their means and covariance matrices. Denote the (presumed known) mean and covariance of the distribution $p(x_N|Z_N)$ by $\hat{x}_N$ and P_N, respectively. In general, $\hat{x}_N$ and P_N are functions of Z_N, but we will not encumber the notation with this information. Likewise, denote the mean and covariance of $p(x_{N+1}|Z_N)$ by $\bar{x}_{N+1}$ and Q_{N+1}. The task is thus to derive expressions for $\bar{x}_{N+1}$ and Q_{N+1} in terms of $\hat{x}_N$ and P_N and expressions for $\hat{x}_{N+1}$ and P_{N+1} in terms of $\bar{x}_{N+1}$ and Q_{N+1}.

7.2.1 Prediction Step

The prediction step (time update) is straightforward. For $\bar{x}_{N+1}$, simply take the expected value of Equation (7.0-1a) conditioned on Z_N.

$$E\{x_{N+1}|Z_N\} = \Phi E\{x_N|Z_N\} + \Psi u_N + FE\{n_N|Z_N\} \tag{7.2-2}$$

The quantities $E\{x_{N+1}|Z_N\}$ and $E\{x_N|Z_N\}$ are, by definition, $\bar{x}_{N+1}$ and $\hat{x}_N$, respectively. Z_N is a function of $x_0, n_0, \ldots, n_{N-1}, n_1, \ldots n_N$, and deterministic quantities; n_N is independent of all of these, and therefore independent of Z_N. Thus

$$E\{n_N|Z_N\} = E\{n_N\} = 0 \tag{7.2-3}$$

Substituting this into Equation (7.2-2) gives

$$\bar{x}_{N+1} = \Phi\hat{x}_N + \Psi u_N \tag{7.2-4}$$

In order to evaluate Q_{N+1}, take the covariance of both sides of Equation (7.0-1a). Since the three terms on the right-hand side of the equation are independent, the covariance of their sum is the sum of their covariances.

$$\mathrm{cov}\{x_{N+1}|Z_N\} = \Phi\, \mathrm{cov}\{x_N|Z_N\}\Phi^* + \mathrm{cov}\{\Psi u_N|Z_N\} + F\, \mathrm{cov}\{n_N|Z_N\}F^* \tag{7.2-5}$$

The terms $\mathrm{cov}\{x_{N+1}|Z_N\}$ and $\mathrm{cov}\{x_N|Z_N\}$ are, by definition, Q_{N+1} and P_N, respectively. Ψu_N is deterministic and, thus, has zero covariance. By the independence of n_N and Z_N

$$\mathrm{cov}\{n_N|Z_N\} = \mathrm{cov}\{n_N\} = I \tag{7.2-6}$$

Substituting these relationships into Equation (7.2-5) gives

$$Q_{N+1} = \Phi P_N \Phi^* + FF^* \tag{7.2-7}$$

Equations (7.2-4) and (7.2-7) constitute the results desired for the prediction step (time update) of the filtering problem. They readily generalize to predicting more than one sample ahead. These equations justify our earlier statement that, once the filtering problem is solved, the prediction problem is easy; for suppose we desire to estimate x_M based on Z_N with $M > N$. If we can solve the filtering problem to obtain $\hat{x}_N$, the filtered estimate of x_N, then, by a straightforward extension of Equation (7.2-4),

$$E\{x_M|Z_N\} = \Phi^{M-N}\hat{x}_N + \sum_{i=N}^{M-1} \Phi^{M-1-i}\Psi u_i \tag{7.2-8}$$

is the desired MAP estimate of x_M.

7.2.2 Correction Step

For the correction step (measurement update), assume that we know the mean, $\tilde{x}_{N+1}$, and covariance, Q_{N+1}, of the distribution of x_{N+1} given Z_N. We seek the distribution of x_{N+1} given both Z_N and z_{N+1}. From Equation (7.0-1b)

$$z_{N+1} = Cx_{N+1} + Du_{N+1} + Gn_{N+1} \tag{7.2-9}$$

The distribution of n_{N+1} is Gaussian with zero mean and identity covariance. By the same argument as used for n_N, n_{N+1} is independent of Z_N. Thus, we can say that

$$p(n_{N+1}|Z_N) = p(n_{N+1}) \tag{7.2-10}$$

This trivial-looking statement is the key to the problem, for now everything in the problem is conditioned in Z_N, we know the distributions of x_{N+1} and n_{N+1} conditioned on Z_N, and we seek the distribution of x_{N+1} conditioned on Z_N, and additionally conditioned on z_{N+1}.

This problem is thus exactly in the form of Equation (5.1-1), except that all of the distributions involved are conditioned on Z_N. This amounts to nothing more than restating the problem of Chapter 5 on a different probability space, one conditioned on Z_N. The previous results apply directly to the new probability space. Therefore, from Equations (5.1-14) and (5.1-15)

$$\hat{x}_{N+1} = \tilde{x}_{N+1} + P_{N+1}C^*(GG^*)^{-1}(z_{N+1} - C\tilde{x}_{N+1} - Du_{N+1}) \tag{7.2-11}$$

$$P_{N+1} = (C^*(GG^*)^{-1}C + Q_{N+1}^{-1})^{-1} \tag{7.2-12}$$

In obtaining Equations (7.2-11) and (7.2-12) from Equations (5.1-14) and (5.1-15), we have identified the following quantities:

(5.1-14),(5.1-15)	(7.2-11),(7.2-12)
m_ξ	$\tilde{x}_{N+1}$
P	Q_{N+1}
Z	z_{N+1}
C	C
D	Du_{N+1}
$E\{\xi\|Z\}$	$\hat{x}_{N+1}$
$\mathrm{cov}\{\xi\|Z\}$	P_{N+1}
GG^*	GG^*

This completes the derivation of the correction step (measurement update), which we see to be a direct application of the results from Chapter 5.

7.2.3 Kalman Filter

To complete the recursive solution to the filtering problem, we need only know the solution for some value of N, and we can now propagate that solution to larger N. The solution for $N = 0$ is immediate from the initial problem statement. The distribution of x_0, conditioned on Z_0 (i.e., conditioned on nothing because $Z_1 = (z_1,\ldots,z_1)^*$), is given to be Gaussian with mean m_0 and covariance P_0.

Let us now fit together the pieces derived above to show how to solve the filtering problem:

Step 1: Initialization

$$\text{Define } \hat{x}_0 = m_0$$

$$P_0 \text{ is given}$$

Step 2: Prediction (time update), starting with $i = 0$,

$$\tilde{x}_{i+1} = \Phi\hat{x}_i + \Psi u_i \tag{7.2-13}$$

$$\tilde{z}_{i+1} = C\tilde{x}_{i+1} + Du_{i+1} \tag{7.2-14}$$

$$Q_{i+1} = \Phi P_i \Phi^* + FF^* \tag{7.2-15}$$

Step 3: Correction (measurement update)

$$P_{i+1} = (C^*(GG^*)^{-1}C + Q_{i+1}^{-1})^{-1} \tag{7.2-16}$$

$$\hat{x}_{i+1} = \tilde{x}_{i+1} + P_{i+1}C^*(GG^*)^{-1}(z_{i+1} - \tilde{z}_{i+1}) \tag{7.2-17}$$

We have defined the quantity $\tilde{z}_{i+1}$ by Equation (7.2-14) in order to make the form of Equation (7.2-17) more apparent; $\tilde{z}_{i+1}$ can easily be shown to be $E\{z_{i+1}|Z_i\}$. Repeat the prediction and correction steps for $i = 0, 1,\ldots,N - 1$ in order to obtain $\hat{x}_N$, the MAP estimate of x_N based on $z_1,\ldots,z_N$.

Equations (7.2-13) to (7.2-17) constitute the Kalman filter for discrete-time systems. The recursive form of this filter is particularly suited to real-time applications. Once $\hat{x}_N$ has been computed, it is not necessary, as it was using the methods of Section 7.1, to start from scratch in order to compute $\hat{x}_{N+1}$; we need do only one more prediction step and one more correction step. It is extremely important to note that the computational cost of obtaining $\hat{x}_{N+1}$ from $\hat{x}_N$ is not a function of N. This means that real-time Kalman filters can be implemented using fixed finite resources to run for arbitrarily long time intervals. This was not the case using the methods of Section 7.1, where the estimator started from scratch for each time point, and each new estimate required more computation than the previous estimate. For some applications, it is also important that the P_i and Q_i do not depend on the measurements, and can thus be precomputed. Such precomputation can significantly reduce real-time computational requirements.

None of these advantages should obscure the fact that the Kalman filter obtains the same estimates as were obtained in Section 7.1. The advantages of the Kalman filter lie in the easier computation of the estimates, not in improvements in the accuracy of the estimates.

7.2.4 Alternate Forms

The filter Equations (7.2-13) to (7.2-17) can be algebraically manipulated into several equivalent alternate forms. Although all of the variants are formally equivalent, different ones have computational advantages in different situations. Some of the advantages lie in different points of singularity and different size matrices to invert. We will show a few of the possible alternate forms in this section.

The first variant comes from using Equations (5.1-12) and (5.1-13) (the covariance form) instead of (5.1-14) and (5.1-15) (the information form). Equations (7.2-16) and (7.2-17) then become

$$P_{i+1} = Q_{i+1} - Q_{i+1}C^*(CQ_{i+1}C^* + GG^*)^{-1}CQ_{i+1} \tag{7.2-18}$$

$$\hat{x}_{i+1} = \tilde{x}_{i+1} + Q_{i+1}C^*(CQ_{i+1}C^* + GG^*)^{-1}(z_{i+1} - \tilde{z}_{i+1}) \tag{7.2-19}$$

The covariance form is particularly useful if GG^* or any of the Q_i are singular. The exact conditions under which Q_i can become singular are fairly complicated, but we can draw some simple conclusions from looking at Equation (7.2-15). First, if FF^* is nonsingular, then Q_i can never be singular. Second, a singular P_0 (particularly $P_0 = 0$) is likely to cause problems if FF^* is also singular. The only matrix to invert in Equations (7.2-18) and (7.2-19) is $CQ_{i+1}C^* + GG^*$. If this matrix is singular the problem is ill-posed; the situation is the same as that discussed in Section 5.3.3.

Note that the covariance form involves inversion of an ℓ-by-ℓ matrix, where ℓ is the length of the observation vector. On the other hand, the information form involves inversion of a p-by-p matrix, where p is the length of the state vector. For some systems, the difference between ℓ and p may be significant, resulting in a strong preference for one form or the other.

If G is diagonal (or if GG^* is diagonalizable the system can be rewritten with a diagonal G), Equations (7.2-18) and (7.2-19) can be manipulated into a form that involves no matrix inversions. The key to this manipulation is to consider the system to have ℓ independent scalar observations at each time point instead of a single vector observation of length ℓ. The scalar observations can then be processed one at a time. The Kalman filter partitions the estimation problem by processing the measurements one time-point at a time; with this modification, we extend the same partitioning concept to process one element of the measurement vector at a time. The derivation of the measurement-update Equations (7.2-18) and (7.2-19) applies without change to a system with several independent observations at a time point. We need only apply the measurement-update equation ℓ times with no intervening time updates. We do need a little more complicated notation to keep track of the process, but the equations are basically the same.

Let $C^{(j)}$ and $D^{(j)}$ be the jth rows of the C and D matrices, $G^{(j,j)}$ be the jth diagonal element of G, and $z_{i+1}^{(j)}$ be the jth element of z_{i+1}. Define $\hat{x}_{i+1,j}$ to be the estimate of x_{i+1} after the jth scalar observation at time $i + 1$ has been processed, and define $P_{i+1,j}$ to be the covariance of $\hat{x}_{i+1,j}$. We start the measurement update at each time point with

$$\hat{x}_{i+1,0} = \tilde{x}_{i+1} \tag{7.2-20}$$

$$P_{i+1,0} = Q_{i+1} \tag{7.2-21}$$

Then, for each scalar measurement, we do the update

$$P_{i+1,j+1} = P_{i+1,j} - P_{i+1,j}C^{(j)*}(C^{(j)}P_{i+1,j}C^{(j)*} + G^{(j,j)^2})^{-1}C^{(j)}P_{i+1,j} \tag{7.2-22}$$

$$\hat{x}_{i+1,j+1} = \hat{x}_{i+1,j} + P_{i+1,j}C^{(j)*}(C^{(j)}P_{i+1,j}C^{(j)*} + G^{(j,j)^2})^{-1}(z_{i+1}^{(j+1)} - \hat{z}_{i+1}^{(j+1)}) \tag{7.2-23}$$

where

$$\hat{z}_{i+1}^{(j+1)} = C^{(j+1)}\hat{x}_{i+1,j} + D^{(j+1)}u_{i+1} \tag{7.2-24}$$

Note that the inversions in Equations (7.2-22) and (7.2-23) are scalar inversions rather than matrices. None of these scalars will be 0 unless $CQ_{i+1}C^* + GG^*$ is singular. After processing all ℓ of the scalar measurements for the time point, we have

$$\hat{x}_{i+1} = \hat{x}_{i+1,\ell} \tag{7.2-25}$$

$$P_{i+1} = P_{i+1,\ell} \tag{7.2-26}$$

7.2.5 Innovations

A discussion of the Kalman filter would be incomplete without some mention of the innovations. The innovation at sample point i, also called the residual, is

$$v_i = z_i - \tilde{z}_i \tag{7.2-27}$$

where

$$\tilde{z}_i = E\{z_i|Z_{i-1}\} = C\tilde{x}_i + Du_i \tag{7.2-28}$$

Following the notation for Z_i, we define

$$V_i = [v_1, v_2, \ldots, v_i]^* \tag{7.2-29}$$

Now V_i is a linear function of Z_i. This is shown by Equations (7.2-13) to (7.2-17) and (7.2-27), which give formulae for computing the v_i in terms of the z_i. It may not be immediately obvious that this function is invertible. We will prove invertibility by writing the inverse function; i.e., by expressing Z_i in terms of V_i. Repeating Equations (7.2-13) and (7.2-14):

$$\tilde{x}_{i+1} = \Phi\hat{x}_i + \Psi u_i \tag{7.2-30a}$$

$$\tilde{z}_{i+1} = C\tilde{x}_{i+1} + Du_{i+1} \tag{7.2-30b}$$

Substituting Equation (7.2-27) into Equation (7.2-17) gives

$$\hat{x}_{i+1} = \tilde{x}_{i+1} + P_{i+1}C^*(GG^*)^{-1}v_{i+1} \tag{7.2-30c}$$

Finally, from Equation (7.2-27)

$$z_{i+1} = \tilde{z}_{i+1} + v_{i+1} \tag{7.2-30d}$$

Equation (7.2-30) is called the innovations form of the system. It gives the recursive formula for computing the z_i from the v_i.

Let us examine the distribution of the innovations. The innovations are obviously Gaussian, because they are linear functions of Z, which is Gaussian. Using Equation (3.3-10), it is immediate that the mean of the innovation is 0.

$$\begin{aligned} E\{v_i\} &= E\{z_i - E(z_i|Z_{i-1})\} \\ &= E\{z_i\} - E\{E(z_i|Z_{i-1})\} = 0 \end{aligned} \tag{7.2-31}$$

Derive the covariance matrix of the innovation by writing

$$\begin{aligned} v_i &= Cx_i + Du_i + Gn_i - C\tilde{x}_i - Du_i \\ &= C(x_i - \tilde{x}_i) + Gn_i \end{aligned} \tag{7.2-32}$$

The two terms on the right are independent, so

$$\begin{aligned} \mathrm{cov}(v_i) &= C\,\mathrm{cov}(x_i - \hat{x}_i)C^* + GG^* \\ &= CQ_iC^* + GG^* \end{aligned} \tag{7.2-33}$$

The most interesting property of the innovations is that v_i is independent of v_j for $i \neq j$. To prove this, it is sufficient to show that v_i is independent of V_{i-1}. Let us examine $E\{v_i|V_{i-1}\}$. Since V_{i-1} is obtained from Z_{i-1} by an invertible continuous transformation, conditioning on V_{i-1} is the same as conditioning on Z_{i-1}. (If one is known, so is the other.) Therefore,

$$E\{v_i|V_{i-1}\} = E\{v_i|Z_{i-1}\} = 0 \tag{7.2-34}$$

as shown in Equation (7.2-31). Thus we have

$$E\{v_i|V_{i-1}\} = E\{v_i\} \tag{7.2-35}$$

Comparing this equation with the formula for the Gaussian conditional mean given in Theorem (3.5-9), we see that this can be true only if v_i and V_{i-1} are uncorrelated ($\Lambda_{12} = 0$ in the theorem). Then by Theorem (3.5-8), v_i and V_{i-1} are independent.

The innovation is thus a discrete-time white-noise process (i.e., each time point is independent of all of the others). Thus, the Kalman filter is often called a whitening filter; it creates a white process (V) as a function of a nonwhite process (Z).

7.3 STEADY-STATE FORM

The largest computational cost of the Kalman filter is in the computation of the covariance matrix P_i using Equations (7.2-15) and (7.2-16) (or any of the alternate forms). For a large and important class of problems, we can replace P_i and Q_i by constants P and Q, independent of time. This approach significantly lowers computational cost of the filter.

We will restrict the discussion in this section to time-invariant systems; in only a few special cases do time-invariant filters make sense for time-varying systems.

Equations that a time invariant filter must satisfy are easily derived. Using Equations (7.2-18) and (7.2-15), we can express Q_{i+1} as a function of Q_i.

$$Q_{i+1} = \Phi[Q_i - Q_iC^*(CQ_iC^* + GG^*)^{-1}CQ_i]\Phi^* + FF^* \tag{7.3-1}$$

Thus, for Q_i to equal a constant Q, we must have

$$Q = \Phi[Q - QC^*(CQC^* + GG^*)^{-1}CQ]\Phi^* + FF^* \tag{7.3-2}$$

This is the algebraic matrix Riccati equation for discrete-time systems. (An alternate form can be obtained by using Equation (7.2-16) in place of Equation (7.2-18); the condition can also be written in terms of P instead of Q).

If Q is a scalar, the algebraic Riccati equation is a quadratic equation in Q and the solution is simple. For nonscalar Q, the solution is far more difficult and has been the subject of numerous papers. We will not cover the details of deriving and implementing numerical methods for solving the Riccati equation. The most widely used methods are based on eigenvector decomposition (Potter, 1966; Vaughan, 1970; and Geyser and Lehtinen, 1975). When a unique solution exists, these methods give accurate results with small computational costs.

The derivation of the conditions under which Equation (7.3-2) has an acceptable solution is more complicated than would be appropriate for inclusion in this text. We therefore present the following result without proof:

> Theorem 7.3-1 If all unstable or marginally stable modes of the system are controllable by the process noise and are observable, and if $CFF^*C^* + GG^*$ is invertible, then Equation (7.3-2) has a unique positive semidefinite solution and Q_i converges to this solution for all choices of the initial covariance, P_0.
>
> Proof See Schweppe (1973, p. 142) for a heuristic argument, or Balakrishnan (1981) and Kailath and Ljung (1976) for more rigorous treatments.

The condition on $CFF^*C^* + GG^*$ ensures that the problem is well-posed. Without this condition, the inverse in Equation (7.3-1) may not exist for some initial P_0 (particularly $P_0 = 0$). Some statements of the theorem incorporate the stronger requirement that GG^* be invertible, but the weaker condition is sufficient. Perhaps the most important point to note is that the system is not required to be stable. Although the existence and uniqueness of the solution are easier to prove for stable systems, the more general conditions of Theorem (7.3-1) are important in the estimation and control of unstable systems.

We can achieve a heuristic understanding of the need for the conditions of Theorem (7.3-1) by examining one-dimensional systems, for which we can write the solutions to Equation (7.3-2) explicitly. If the system is one-dimensional, then it is observable if C is nonzero (and G is finite), and it is controllable by the process noise if F is nonzero. We will consider the problem in several cases.

Case 1: $G = 0$. In this case, we must have $C \neq 0$ and $F \neq 0$ in order for the problem to be well-posed. Equation (7.3-1) then reduces to $Q_{i+1} = FF^*$, giving a unique time-invariant covariance satisfying Equation (7.3-2).

Case 2: $G \neq 0$, $C = 0$, $F = 0$. In this case, Equation (7.3-1) becomes $Q_{i+1} = \phi^2 Q_i$. This converges to $Q = 0$ if $|\phi| < 1$ (stable system). If $|\phi| = 1$, Q_i remains at the starting value, and thus the steady state covariance is not unique. If $|\phi| > 1$, the solution diverges or stays at 0, depending on the starting value.

Case 3: $G \neq 0$, $C = 0$, $F \neq 0$. In this case, Equation (7.3-2) reduces to

$$Q = \phi^2 Q + F^2 \tag{7.3-3}$$

For $|\phi| < 1$, this equation has a unique, nonnegative solution

$$Q = \frac{F^2}{1 - \phi^2} \tag{7.3-4}$$

and convergence of Equation (7.3-1) to this solution is easily shown. If $|\phi| \geq 1$, the solution is negative, which is not an admissible covariance, or infinite; in either event, Equation (7.3-1) diverges to infinity.

Case 4: $G \neq 0$, $C \neq 0$, $F = 0$. In this case, Equation (7.3-2) is a quadratic equation with roots zero and $(\phi^2 - 1)G^2/C^2$. If $|\phi| < 1$, the second root is negative, and thus there is a unique nonnegative root. If $|\phi| = 1$, there is a double root at zero, and the solution is still unique. In both of these events, convergence of Equation (7.3-1) to the solution at 0 is easy to show. If $|\phi| > 1$, there are two nonnegative roots, and the system can converge to either one, depending on whether or not the initial covariance is zero.

Case 5: $G \neq 0$, $C \neq 0$, $F \neq 0$. In this case, Equation (7.3-2) is a quadratic equation with roots

$$Q = (1/2)H \pm \sqrt{(1/4)H^2 + F^2G^2/C^2} \tag{7.3-5}$$

where

$$H = F^2 + (\phi^2 - 1)G^2/C^2 \tag{7.3-6}$$

Regardless of the value of ϕ, the square-root term is always larger in magnitude than $(1/2)H$; therefore, there is one positive and one negative root. Convergence of Equation (7.3-1) to the positive root is easy to show.

Let us now summarize the results of these five cases. In all well-posed cases, the covariance converges to a unique value if the system is stable. For unstable or marginally stable systems, a unique converged value is assured if both C and F are nonzero. For one-dimensional systems, there is also a unique convergent solution for $|\phi| = 1$, $G \neq 0$, $C \neq 0$, $F = 0$; this case illustrates that the conditions of Theorem (7.3-1) are not necessary, although they are sufficient.

Heuristically, we can say that observability ($C \neq 0$) prevents the covariance from diverging to infinity for unstable systems. Controllability by the process noise ($F \neq 0$) ensures uniqueness by eliminating the possibility of perfect prediction ($Q = 0$).

An important related question to consider is the stability of the filter. We define the corrected error vector to be

$$\hat{e}_i = x_i - \hat{x}_i \tag{7.3-7}$$

Using Equations (7.0-1), (7.2-15), (7.2-16), and (7.2-19) gives the recursive relationship

$$\hat{e}_{i+1} = (I - KC)\phi\hat{e}_i + (I - KC)Fn_i - KGn_{i+1} \tag{7.3-8}$$

where

$$K = PC^*(GG^*)^{-1} = QC^*(CQC^* + GG^*)^{-1} \tag{7.3-9}$$

We can show that, given the conditions of Theorem (7.3-1), the system of Equation (7.3-8) is stable. This stability implies that, in the absence of new disturbances, (noise) errors in the state estimate will die out with time; furthermore, for bounded disturbances, the errors will always be bounded. A rigorous proof is not presented here.

It is interesting to examine the stability of the one-dimensional example with $G \neq 0$, $C \neq 0$, $F = 0$, and $|\phi| = 1$. We previously noted that Q_i for this case converges to 0 for all initial covariances. Let us examine the steady-state filter. For this case, Equation (7.3-8) reduces to

$$\hat{e}_{i+1} = \hat{e}_i \tag{7.3-10}$$

which is only marginally stable. Recall that this case did not meet the conditions of Theorem (7.3-1), so our stability guarantee does not apply. Although a steady-state filter exists, it does not perform at all like the time-varying filter. The time-varying filter reduces the error to zero asymptotically with time. The steady-state filter has no feedback, and the error remains at its initial value. Balakrishnan (1984) discusses the steady-state filter in more detail.

Two special cases of time-invariant Kalman filters deserve special note. The first case is where F is zero and the system is stable (and GG^* must be invertible to ensure a well-posed problem). In this case, the

steady state Kalman gain K is zero. The Kalman filter simply integrates the state equation, ignoring any available measurements. Since the system is stable and has no disturbances, the error will decay to zero. The same filter is obtained for nonzero F if C is zero or if G is infinite. The error does not then decay to zero, but the output contains no useful information to feed back.

The second special case is where G is zero and C is square and invertible. FF^* must be invertible to ensure a well-posed problem. For this case, the Kalman gain is C^{-1}. The estimator then reduces to

$$\hat{x}_i = C^{-1}(z_i - Du_i) \tag{7.3-11}$$

which ignores all previous information. The current state can be reconstructed exactly from the current measurement, so there is no need to consider past data. This is the antithesis of the case where F is 0 and no information from the current measurement is used. Most realistic systems lie somewhere between these two extremes.

7.4 CONTINUOUS TIME

The form of a linear continuous-time system model is

$$\dot{x}(t) = Ax(t) + Bu(t) + F_c n(t) \tag{7.4-1a}$$

$$z(t) = Cx(t) + Du(t) + G_c \eta(t) \tag{7.4-1b}$$

where n and η are assumed to be zero-mean white-noise processes with unity power spectral density. The input u is assumed to be known exactly. As in the discrete-time analysis, we will simplify the notation by assuming that the system is time invariant. The same derivation applies to time-varying systems by evaluating the matrices at the appropriate time points.

We will analyze Equation (7.4-1) as a limit of the discrete-time systems

$$x(t_i + \Delta) = (I + \Delta A)x(t_i) + \Delta Bu(t_i) + \Delta^{1/2}F_c n(t_i) \tag{7.4-2a}$$

$$z(t_i) = Cx(t_i) + Du(t_i) + \Delta^{-1/2}G_c \eta(t_i) \tag{7.4-2b}$$

where n and η are discrete-time white-noise processes with identity covariances. The reasons for the $\Delta^{1/2}$ factors were discussed in Section 6.2.

The filter for the system of Equation (7.4-2) is obtained by making appropriate substitutions in Equations (7.2-13) to (7.2-17). We need to substitute $(I + \Delta A)$ in place of Φ, ΔB in place of Ψ, $\Delta F_c F_c^*$ in place of FF^*, and $\Delta^{-1}G_c G_c^*$ in place of GG^*. Combining Equations (7.2-13), (7.2-14), and (7.2-17) and making the substitutions gives

$$\hat{x}(t_i + \Delta) = (I + \Delta A)\hat{x}(t_i) + \Delta Bu(t_i) + \Delta P(t_i + \Delta)C^*(G_c G_c^*)^{-1}[z(t_i + \Delta) - C(I + \Delta A)\hat{x}(t_i) - C\Delta Bu(t_i) - Du(t_i + \Delta)] \tag{7.4-3}$$

Subtracting $\hat{x}(t_i)$ and dividing by Δ gives

$$\frac{\hat{x}(t_i + \Delta) - \hat{x}(t_i)}{\Delta} = A\hat{x}(t_i) + Bu(t_i) + P(t_i + \Delta)C^*(G_c G_c)^{-1}[z(t_i + \Delta) - C(I + \Delta A)x(t_i) - C\Delta Bu(t_i) - Du(t_i)] \tag{7.4-4}$$

Taking the limit as $\Delta \to 0$ gives the filter equation

$$\dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + P(t)C^*(G_c G_c^*)^{-1}[z(t) - C\hat{x}(t) - Du(t)] \tag{7.4-5}$$

It remains to find the equation for $P(t)$. First note that Equation (7.2-15) becomes

$$Q(t_i + \Delta) = (I + \Delta A)P(t_i)(I + \Delta A)^* + \Delta F_c F_c^* \tag{7.4-6}$$

and thus

$$\lim_{\Delta \to 0} Q(t_i + \Delta) = P(t_i) \tag{7.4-7}$$

Equation (7.2-18) is a more convenient form for our current purposes than (7.2-16). Make the appropriate substitutions in Equation (7.2-18) to get

$$P(t_i + \Delta) = Q(t_i + \Delta) - Q(t_i + \Delta)C^*(CQ(t_i + \Delta)C^* + \Delta^{-1}G_c G_c^*)^{-1}CQ(t_i + \Delta)$$

Subtract $P(t_i)$ and divide by Δ to give

$$\frac{P(t_i + \Delta) - P(t_i)}{\Delta} = \frac{Q(t_i + \Delta) - P(t_i)}{\Delta} - Q(t_i + \Delta)C^*(\Delta CQ(t_i + \Delta)C^* + G_c G_c^*)^{-1}CQ(t_i + \Delta) \tag{7.4-9}$$

For the first term on the right of Equation (7.4-9), substitute from Equation (7.4-7) to get

$$\frac{Q(t_i + \Delta) - P(t_i)}{\Delta} = AP(t_i) + P(t_i)A^* + \Delta AP(t_i)A^* + F_cF_c^* \tag{7.4-10}$$

Thus in the limit Equation (7.4-9) becomes

$$\dot{P}(t) = AP(t) + P(t)A^* + F_cF_c^* - P(t)C^*(G_cG_c^*)^{-1}CP(t) \tag{7.4-11}$$

Equation (7.4-11) is the continuous-time Ricatti equation. The initial condition for the equation is $P_0 = 0$, the covariance of the initial state. P_0 is assumed to be known. Equations (7.4-5) and (7.4-11) constitute the solution to the continuous-time filtering problem for linear systems with white process and measurement noise. The continuous-time filter requires GG^* to be nonsingular.

One point worth noting about the continuous-time filter is that the innovation $z(t) - \hat{z}(t)$ is a white-noise process with the same power spectral density as the measurement noise. (They are not, however, the same process.) The power spectrum of the innovation can be found by looking at the limit of Equation (7.2-33). Making the appropriate substitutions gives

$$\mathrm{cov}(v(t_i)) = CQ(t_i)C^* + \Delta^{-1}G_cG_c^* \tag{7.4-12}$$

The power spectral density of the innovation is then

$$\lim_{\Delta\to 0} \Delta^{-1}\mathrm{cov}(v(t_i)) = G_cG_c^* \tag{7.4-13}$$

The disappearance of the first term of Equation (7.4-12) in the limit makes the continuous-time filter simpler than the discrete-time one in many ways.

For time-invariant continuous-time systems, we can investigate the possibility that the filter reaches a steady state. As in the discrete-time steady-state filter, this outcome would result in a significant computational advantage. If the steady-state filter exists, it is obvious that the steady-state $P(t)$ must satisfy the equation

$$AP + PA^* + F_cF_c^* - PC^*(G_cG_c^*)^{-1}CP = 0 \tag{7.4-14}$$

obtained by setting $\dot{P}$ to 0 in Equation (7.4-11). The eigenvector decomposition methods referenced after Equation (7.3-2) are also the best practical numerical methods for solving Equation (7.4-14). The following theorem, comparable to Theorem (7.3-1), is not proven here.

> Theorem 7.4-1 If all unstable or neutrally stable modes of the system are controllable by the process noise and are observable, and if $G_cG_c^*$ is invertible, then Equation (7.4-14) has a unique positive semidefinite solution, and $P(t)$ converges to this solution for all choices of the initial covariance P_0.
>
> Proof See Kailath and Lyung (1976), Balakrishnan (1981), or Kalman and Bucy (1961).

7.5 CONTINUOUS/DISCRETE TIME

Many practical applications of filtering involve discrete sampled measurements of systems with continuous-time dynamics. Since this problem has elements of both discrete and continuous time, there is often debate over whether the discrete- or continuous-time filter is more appropriate. In fact, neither of these filters is appropriate because they are both based on models that are not realistic representations of the true system. As Schweppe (1973, p. 206) says,

> Some rather interesting arguments sometimes result when one asks the question, Are the discrete- or the continuous-time results more useful? The answer is, of course, that the question is stupid....neither is superior in all cases.

The appropriate model for a continuous-time dynamic system with discrete-time measurements is a continuous-time model with discrete-time measurements. Although this statement sounds like a tautology, its point has been missed enough to make it worth emphasizing. Some of the confusion may be due to the mistaken impression that such a mixed model could not be analyzed with the available tools. In fact, the derivation of the appropriate filter is trivial, given the pure continuous- and pure discrete-time results. The filter for this class of problems simply involves an appropriate combination of the discrete- and continuous-time filters previously derived. It takes only a few lines to show how the previously derived results fit this problem. We will spend most of this section talking about implementation issues in a little more detail.

Let the system be described by

$$\dot{x}(t) = Ax(t) + Bu(t) + F_cn(t) \tag{7.5-1a}$$

$$z(t_i) = Cx(t_i) + Du(t_i) + Gn(t_i) \qquad i = 1,2,\ldots \tag{7.5-1b}$$

Equation (7.5-1a) is identical to Equation (7.4-1a); and, except for a notation change, Equation (7.5-1b) is identical to Equation (7.0-1b). Note that the observation is only defined at the discrete points t_i, although the state is defined in continuous time.

Between the times of two observations, the analysis of Equation (7.5-1) is identical to that of Equation (7.4-1) with an infinite G matrix or a zero C matrix; either of these conditions is equivalent to having no useful observation. Let $\hat{x}(t_i)$ be the state estimate at time t_i based on the observations up to and including $z(t_i)$. Then the predicted estimate in the interval $(t_i, t_{i+\Delta}]$ is obtained f.

$$\tilde{x}(t_i^+) = \hat{x}(t_i) \tag{7.5-2}$$

$$\dot{\tilde{x}}(t) = A\tilde{x}(t) + Bu(t) \tag{7.5-3}$$

The covariance of the prediction is

$$Q(t_i^+) = P(t_i) \tag{7.5-4}$$

$$\dot{Q}(t) = AQ(t) + Q(t)A^* + F_c F_c^* \tag{7.5-5}$$

Equations (7.5-3) and (7.5-5) are obtained directly by substituting $C = 0$ in Equations (7.4-5) and (7.4-11). The notation has been changed to indicate that, because there is no observation in the interval, these are predicted estimates; whereas, in the pure continuous-time filter, the observations are continuously used and filtered estimates are obtained. Integrate Equations (7.5-3) and (7.5-5) over the interval $(t_i, t_i + \Delta)$ to obtain the predicted estimate $\tilde{x}(t_i + \Delta)$ and its covariance $Q(t_i + \Delta)$.

In practice, although $u(t)$ is defined continuously, it will often be measured (or otherwise known) only at the time points t_i. Furthermore, the integration will likely be done by a digital computer which cannot integrate continuous-time data exactly. Thus Equation (7.5-3) will be integrated numerically. The simplest integration approximation would give

$$\tilde{x}(t_i + \Delta) \cong (I + A)\tilde{x}(t_i^+) + \Delta Bu(t_i) \tag{7.5-6}$$

This approximation may be adequate for some purposes, but it is more often a little too crude. If the A matrix is time-varying, there are several reasonable integration schemes which we will not discuss here; the most common are based on Runge-Kutta algorithms (Acton, 1970). For systems with time-invariant A matrices and constant sample intervals, the transition matrix is by far the most efficient approach. First define

$$\phi = \exp(A\Delta) \tag{7.5-7}$$

$$\Psi = \int_0^\Delta \exp(At)dt\ B \tag{7.5-8}$$

$$\tilde{x}(t_i + \Delta) \cong \phi\tilde{x}(t_i^+) + \Psi u(t_i) \tag{7.5-9}$$

This approximation is the exact solution to Equation (7.5-3) if $u(t)$ holds its value between samples. Wiberg (1971) and Zadeh and Desoer (1963) derive this solution. Moler and Van Loan (1978) discuss various means of numerically evaluating Equations (7.5-7) and (7.5-8). Equation (7.5-9) has an advantage of being in the exact form in which discrete-time systems are usually written (Equation (7.0-1a)).

Equation (7.5-9) introduces about 1/2-sample delay in the modeling of the response to the control input unless the continuous-time $u(t)$ holds its value between samples; this delay is often unacceptable. Figure (7.5-1) shows a sample input signal and the signal as modeled by Equation (7.5-9). A better approximation is usually

$$x(t_i + \Delta) \cong \phi\tilde{x}(t_i^+) + (1/2)\Psi(u(t_i) + u(t_i + \Delta)) \tag{7.5-10}$$

This equation models $u(t)$ between samples as being constant at the average of the two sample values. Figure (7.5-2) illustrates this model. There is little phase lag in the model represented by Equation (7.5-10), and the difference in implementation cost between Equations (7.5-9) and (7.5-10) is negligible. Equation (7.5-10) is probably the most commonly used approximation method with time-invariant A matrices.

The high-frequency content introduced by the jumps in the above models can be removed by modeling $u(t)$ as a linear interpolation between the measured values as illustrated in Figure (7.5-3). This model adds another term to Equation (7.5-10) proportional to $u(t_i + \Delta) - u(t_i)$. In our experience, this degree of fidelity is usually unnecessary, and is not worth the extra cost and complication. There are some applications where the accuracy required might justify this or even more complicated methods, such as higher-order spline fits. (The linear interpolation is a first-order spline.)

If you are using a Runge-Kutta algorithm instead of a transition-matrix algorithm for solving the differential equation, linear interpolation of the input introduces negligible extra cost and is common practice.

Equation (7.5-5) does not involve measured data and thus does not present the problems of interpolating between the measurements. The exact solution of Equation (7.5-5) is

$$Q(t_i + \Delta) = \Phi Q(t_i^+)\Phi^* + \int_0^\Delta \exp(A(\Delta - \tau))F_c F_c^* \exp(A^*(\Delta - \tau))d\tau \tag{7.5-11}$$

as can be verified by substitution. Note that Equation (7.5-11) is exactly in the form of a discrete-time update of the covariance (Equation (7.2-15)) if F is defined as a square root of the integral term. For small Δ, the integral term is well approximated by $\Delta F_c F_c^*$, resulting in

$$Q(t_i + \Delta) \approx \Phi Q(t_i^+)\Phi^* + \Delta F_c F_c^* \tag{7.6-12}$$

The errors in the approximation are usually far smaller than the uncertainty in the value of F_c, and can thus be neglected. This approximation is significantly better than the alternate approximation

$$Q(t_i + \Delta) \approx Q(t_i^+) + \Delta A Q(t_i^+) + \Delta Q(t_i^+)A^* + \Delta F_c F_c^* \tag{7.5-13}$$

obtained by inspection from Equation (7.5-5).

The above discussion has concentrated on propagating the estimate between measurements, i.e., the time update. It remains only to discuss the measurement update for the discrete measurements. We have $\hat{x}(t_i)$ and $Q(t_i)$ at some time point. We need to use these and the measured data at the time point to obtain $\hat{x}(t_i)$ and $P(t_i)$. This is identical to the discrete-time measurement update problem solved by Equations (7.2-16) and (7.2-17). We can also use the alternate forms discussed in Section 7.2.4.

To start the filter, we are given the *a priori* mean $\hat{x}(t_0)$ and covariance $Q(t_0)$ of the state at time t_0. Use Equations (7.2-16) and (7.2-17) (or alternates) to obtain $\hat{x}(t_0)$ and $P(t_0)$. Integrate Equations (7.5-2) to (7.5-5) from t_0^+ to t_1 by some means (most likely Equations (7.5-10) and (7.5-12)) to obtain $\hat{x}(t_1)$ and $Q(t_1)$. This completes one time step of the filter; processing of subsequent time points uses the same procedure.

The solution for the steady-state form of the discrete/continuous filter follows immediately from that of the discrete-time filter, because the equations for the covariance updates are identical for the two filters with the appropriate substitution of F in terms of F_c. Theorem (7.3-1) therefore applies.

We can summarize this section by saying that there is a continuous/discrete-time filter derived from appropriate results in the pure discrete- and pure continuous-time analyses. If the input u holds its value between samples, then the form of the continuous/discrete filter is identical to that of the pure discrete-time filter with an appropriate substitution for the equivalent discrete-time process noise covariance. For more realistic behavior of u, we must adopt approximations if the analysis is done on a digital computer. It is also possible to view the continuous-time filter equations as giving reasonable approximations to the continuous/discrete-time filter in some situations. In any event, we will not go wrong as long as we recognize that we can write the exact filter equations for the continuous/discrete-time system and that we must consider any other equations used as approximations to the exact solution. With this frame of mind we can objectively evaluate the adequacy of the approximations involved for specific problems.

7.6 SMOOTHING

The derivation of optimal smoothers draws heavily on the derivation of the Kalman filter. Starting from the filter results, only a single step is required to compute the smoothed estimates. In this section, we briefly derive the fixed-interval smoother for discrete-time linear systems with additive Gaussian noise. Fixed-interval smoothers are the most widely used. The same general principles apply to deriving fixed-point and fixed-lag smoothers. See Meditch (1969) for derivations and equations for fixed-point and fixed-lag smoothers and for continuous-time forms.

There are alternate computational forms for the fixed-interval smoother; these forms give mathematically equivalent results. We will not discuss computational advantages of the various forms. See Bierman (1977) and Bach and Wingrove (1983) for alternate forms and discussions of their advantages.

Consider the fixed-interval smoothing problem on an interval with N time points. As in the filter derivation, we will concentrate on two time points at a time in order to get a recursive form. It is straightforward to write an explicit formulation for the smoother, like the explicit filter form of Section 7.1, but such a form is impractical.

In the nature of recursive derivations, assume that we have previously computed $\bar{x}_{i+1}$, the smoothed estimate of x_{i+1}, and S_{i+1}, the covariance of x_{i+1} given Z_N. We seek to derive an expression for $\bar{x}_i$ and S_i. Note that this recursion runs backwards in time instead of forwards; a forward recursion will not work, for reasons which we will see later.

The smoothed estimates, $\bar{x}_i$ and $\bar{x}_{i+1}$, are defined by

$$\begin{bmatrix} \bar{x}_i \\ \bar{x}_{i+1} \end{bmatrix} = E\left\{ \begin{matrix} x_i \\ x_{i+1} \end{matrix} \middle| Z_N \right\} \tag{7.6-1}$$

We will use the measurement partitioning ideas of Section 5.2.2, with the measurement Z_N partitioned into Z_i and

$$Z_i = (z_{i+1}, \ldots, z_N) \tag{7.6-2}$$

From the derivation of the Kalman filter, we can write the joint distribution of x_i and x_{i+1} conditioned on Z_i. It is Gaussian with

$$E\left\{\begin{matrix} x_i \\ x_{i+1} \end{matrix}\middle| Z_i\right\} = \begin{Bmatrix} \hat{x}_i \\ \tilde{x}_{i+1} \end{Bmatrix} \tag{7.6-3}$$

$$\mathrm{cov}\left\{\begin{matrix} x_i \\ x_{i+1} \end{matrix}\middle| Z_i\right\} = \begin{bmatrix} P_i & P_i\Phi^* \\ \Phi P_i & Q_{i+1} \end{bmatrix} \tag{7.6-4}$$

We did not previously derive the cross term in the above covariance matrix. To derive the form shown, write

$$\begin{aligned} E\{(x_i - \hat{x}_i)(x_{i+1} - \tilde{x}_{i+1})^*\} &= E\{(x_i - \hat{x}_i)(\Phi x_i + \Psi u_i + Fn_i - \Phi x_i - \Psi u_i)^*\} \\ &= E\{(x_i - \hat{x}_i)(x_i - \hat{x}_i)^*\}\Phi^* + E\{(x_i - x_i)(Fn_i)^*\} \\ &= P_i\Phi^* + 0 \end{aligned} \tag{7.6-5}$$

For the second step of the partitioned algorithm, we consider the measurements $\bar{Z}_i$, using Equations (7.6-3) and (7.6-4) for the prior distribution. The measurements $\bar{Z}_i$ can be written in the form

$$\bar{Z}_i = \bar{C}_i x_{i+1} + \bar{D}_i + \bar{G}_i \bar{n}_i \tag{7.6-6}$$

for some matrices $\bar{C}_i$, $\bar{D}_i$, and $\bar{G}_i$, and some Gaussian, zero-mean, identity-covariance noise vector $\bar{n}_i$. Although we could laboriously write out expressions for the matrices in Equation (7.6-6), this step is unnecessary; we need only know that such a form exists. The important thing about Equation (7.6-6) is that x_i does not appear in it.

Using Equations (7.6-3) and (7.6-4) for the prior distribution and Equation (7.6-6) for the measurement equation, we can now obtain the joint posterior distribution of x_i and x_{i+1} given Z_i. This distribution is Gaussian with mean and covariance given by Equations (5.1-12) and (5.1-13), substituting Equation (7.6-3) for m_ξ, Equation (7.6-4) for P, $\bar{D}_i$ for D, $\bar{G}_i$ for G, and

$$C = [0 | C_i] \tag{7.6-7}$$

By definition (Equation (7.6-1)), the mean of this distribution gives the smoothed estimates $\bar{x}_i$ and $\bar{x}_{i+1}$. Making the substitutions into Equation (5.1-12) and expanding gives

$$\begin{bmatrix} \bar{x}_i \\ \bar{x}_{i+1} \end{bmatrix} = \begin{bmatrix} \hat{x}_i \\ \tilde{x}_{i+1} \end{bmatrix} + \begin{bmatrix} P_i\Phi^*\bar{C}_i \\ Q_{i+1}\bar{C}_i \end{bmatrix} (\bar{C}_i Q_{i+1} \bar{C}_i^* + \bar{G}_i\bar{G}_i^*)^{-1} (\bar{Z}_i - \bar{C}_i\tilde{x}_{i+1} - \bar{D}_i) \tag{7.6-8}$$

We can solve Equation (7.6-8) for $\bar{x}_i$ in terms of $\bar{x}_{i+1}$, which we assume to have been computed in the previous step of the backwards recursion.

$$\bar{x}_i = \hat{x}_i + P_i\Phi^* Q_{i+1}^{-1} (\bar{x}_{i+1} - \tilde{x}_{i+1}) \tag{7.6-9}$$

Equation (7.6-9) is the backwards recursive form sought. Note that the equation does not depend explicitly on the measurements or on the matrices in Equation (7.6-6). That information is all subsumed in $\bar{x}_{i+1}$. The "initial" condition for the recursion is

$$\bar{x}_N = \hat{x}_N \tag{7.6-10}$$

which follows directly from the definitions. We do not have a corresponding known boundary condition at the beginning of the interval, which is why we must propagate the smoothing recursion backwards, instead of forwards.

We can now describe the complete process of computing the smoothed state estimates for a fixed time interval. First propagate the Kalman filter through the entire interval, saving all of the values $\tilde{x}_i$, $\hat{x}_i$, P_i, and Q_i. Then propagate Equation (7.6-9) backwards in time, using the saved values from the filter, and starting from the boundary condition given by Equation (7.6-10).

We can derive a formula for the smoother covariance by substituting appropriately into Equation (5.1-13) to get

$$\begin{bmatrix} S_i & ? \\ ? & S_{i+1} \end{bmatrix} = \begin{bmatrix} P_i & P_i\Phi^* \\ \Phi P_i & Q_{i+1} \end{bmatrix} - \begin{bmatrix} P_i\Phi^*\bar{C}_i \\ Q_{i+1}\bar{C}_i \end{bmatrix} (\bar{C}_i Q_{i+1} \bar{C}_i^* + \bar{G}_i\bar{G}_i^*)^{-1} \begin{bmatrix} P_i\Phi^*\bar{C}_i \\ Q_{i+1}\bar{C}_i \end{bmatrix} \tag{7.6-11}$$

(The off-diagonal blocks are not relevant to this derivation.) We can solve Equation (7.6-11) for S_i in terms of S_{i+1}, giving

$$S_i = P_i - P_i\Phi^* Q_{i+1}^{-1}(Q_{i+1} - S_{i+1})Q_{i+1}^{-1}\Phi P_i \tag{7.6-12}$$

This gives us a backwards recursion for the smoother covariance. The "initial" condition

$$S_N = P_N \tag{7.6-13}$$

follows from the definitions. Note that, as in the recursion for the smoothed estimate, the measurements and the measurement equation matrices have dropped out of Equation (7.6-12). All the necessary data about the future process is subsumed in S_{i+1}. Note also that it is not necessary to compute the smoother covariance S_i in order to compute the smoothed estimates.

7.7 NONLINEAR SYSTEMS AND NON-GAUSSIAN NOISE

Optimal state estimation for nonlinear dynamic systems is substantially more difficult than for linear systems. Only in rare special cases are there tractable exact solutions for optimal filters for nonlinear systems. The same comments apply to systems with non-Gaussian noise.

Practical implementations of filters for nonlinear systems invariably involve approximations. The most common approximations are based on linearizing the system and using the optimal filter for the linearized system. Similarly, non-Gaussian noise is approximated, to first order, by Gaussian noise with the same mean and covariance.

Consider a nonlinear dynamic system with additive noise

$$\dot{x}(t) = f(x(t),u(t)) + n(t) \tag{7.7-1a}$$

$$z(t_i) = g(x(t_i),u(t_i)) + \eta_i \tag{7.7-1b}$$

Assume that we have some nominal estimate, $x_n(t)$, of the state time history. Then the linearization of Equation (7.7-1) about this nominal trajectory is

$$\dot{x}(t) = A(t)x(t) + B(t)u(t) + f_n(t) + n(t) \tag{7.7-2a}$$

$$z(t_i) = C(t_i)x(t_i) + D(t_i) + g_n(t_i) + \eta_i \tag{7.7-2b}$$

where

$$A(t) = \nabla_x f(x_n(t),u(t)) \tag{7.7-3a}$$

$$B(t) = \nabla_u f(x_n(t),u(t)) \tag{7.7-3b}$$

$$C(t) = \nabla_x g(x_n(t),u(t)) \tag{7.7-3c}$$

$$D(t) = \nabla_u g(x_n(t),u(t)) \tag{7.7-3d}$$

$$f_n(t) = f(x_n(t),u(t)) \tag{7.7-4a}$$

$$g_n(t) = g(x_n(t),u(t)) \tag{7.7-4b}$$

For a given nominal trajectory, Equations (7.7-2) to (7.7-4) define a time-varying linear system. The Kalman filter/smoother algorithms derived in previous sections of this chapter give optimal state estimates for this linearized system.

The filter based on this linearized system is called a linearized Kalman filter or an extended Kalman filter (EKF). Its adequacy as an approximation to the optimal filter for the nonlinear system depends on several factors which we will not analyze in depth. It is a reasonable supposition that if the system is nearly linear, then the linearized Kalman filter will be a close approximation to the optimal filter for the system. If, on the other hand, nonlinearities play a major role in defining the characteristic system responses, the reasonableness of the linearized Kalman filter is questionable.

The above description is intended only to introduce the simplest ideas of linearized Kalman filters. Starting from this point, there are numerous extensions, modifications, and nuances of application. Nonlinear filtering is an area of current research. See Bach and Wingrove (1983) and Cox and Bryson (1980) for a few of the many investigations in this field. Schweppe (1973) and Jazwinski (1970) have fairly extensive discussions of nonlinear state estimation.

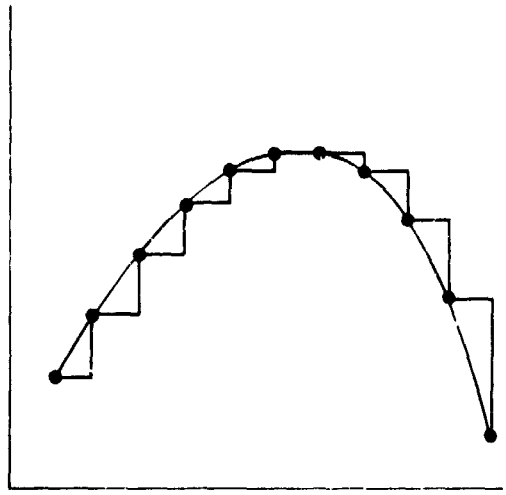


Figure (7.5-1). Hold-last-value input model.

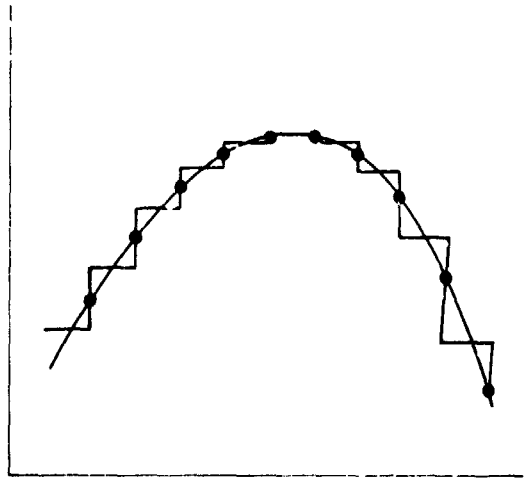


Figure (7.5-2). Average value input model.

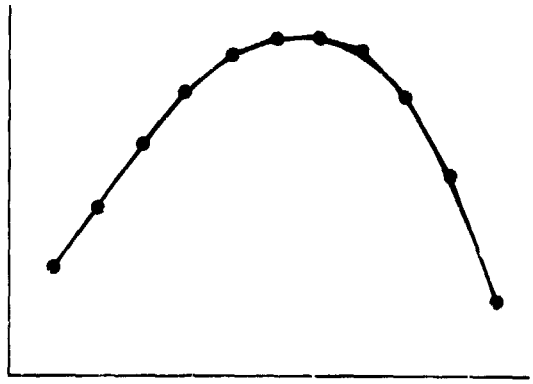


Figure (7.5-3). Linear interpolation input model.

CHAPTER 8

8.0 OUTPUT ERROR METHOD FOR DYNAMIC SYSTEMS

In previous chapters, we have covered the static estimation problem and the estimation of the state of dynamic systems. With this background, we can now begin to address the principle subject of this book, estimation of the parameters of dynamic systems.

Before addressing the more difficult parameter estimation problems posed by more general system models, we will consider the simplified case that leads to the algorithm called output error. The simplification that leads to the output-error method is to omit the process-noise term from the state equation. For this reason, the output-error method is often described by terms like "the no-process-noise algorithm" or "the measurement-noise-only algorithm."

We will first discuss mixed continuous/discrete-time systems, which are most appropriate for the majority of the practical applications. We will follow this discussion by a brief summary of any differences for pure discrete-time systems, which are useful for some applications. The derivation and results are essentially identical. The pure continuous-time results, although similar in expression, involve extra complications. We have never seen an appropriate practical application of the pure continuous-time results; we therefore feel justified in omitting them.

In mixed continuous/discrete time, the most general system model that we will seriously consider is

$$x(t_0) = x_0 \tag{8.0-1a}$$

$$\dot{x}(t) = f[x(t),u(t),\xi] \tag{8.0-1b}$$

$$z(t_i) = g[x(t_i),u(t_i),\xi] + G(\xi)n_i \qquad i = 1,2,\ldots \tag{8.0-1c}$$

The measurement noise n is assumed to be a sequence of independent Gaussian random variables with zero mean and identity covariance. The input u is assumed to be known exactly. The initial condition x_0 can be treated in several ways, as discussed in Section 8.2. In general, the functions f and g can also be explicit functions of t. We omit this from the notation for simplicity. (In any event, explicit time dependence can be put in the notation of Equation (8.0-1) by defining an extra control equal to t.)

The corresponding nonlinear model for pure discrete-time systems is

$$x(t_0) = x_0 \tag{8.0-2a}$$

$$x(t_{i+1}) = f[x(t_i),u(t_i),\xi] \qquad i = 0,1,\ldots \tag{8.0-2b}$$

$$z(t_i) = g[x(t_i),u(t_i),\xi] + G(\xi)n_i \qquad i = 1,2,\ldots \tag{8.0-2c}$$

The assumptions are the same as in the continuous/discrete case.

Although the output-error method applies to nonlinear systems, we will give special attention to the treatment of linear systems. The linear form of Equation (8.0-1) is

$$x(t_0) = x_0 \tag{8.0-3a}$$

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{8.0-3b}$$

$$z(t_i) = Cx(t_i) + Du(t_i) + Gn_i \qquad i = 1,2,\ldots \tag{8.0-3c}$$

The matrices A, B, C, D, and G are functions of ξ; we will not complicate the notation by explicitly indicating this relationship. Of course, x and z are also functions of ξ through their dependence on the system matrices.

In general, the matrices A, B, C, D, and G can also be functions of time. For notational simplicity, we have not explicitly indicated this dependence. In several places, time invariance of the matrices introduces significant computational savings. The text will indicate such situations. Note that ξ cannot be a function of time. Problems with time-varying ξ must be reformulated with a time-invariant ξ in order for the techniques of this chapter to be applicable.

The linear form of Equation (8.0-2) is

$$x(t_0) = x_0 \tag{8.0-4a}$$

$$x(t_{i+1}) = \Phi x(t_i) + \Psi u(t_i) \qquad i = 0,1,\ldots \tag{8.0-4b}$$

$$z(t_i) = Cx(t_i) + Du(t_i) + Gn_i \qquad i = 1,2,\ldots \tag{8.0-4c}$$

The transition matrices Φ and Ψ are functions of ξ, and possibly of time.

For any of the model forms, a prior distribution for ξ may or may not exist, depending on the particular application. When there is no prior distribution, or when you desire to obtain an estimate independent of the

PRECEDING PAGE BLANK NOT FILMED PAGE 88 INTENTIONALLY BLANK

prior distribution, use a maximum-likelihood estimator. When a prior distribution is considered, MAP estimators are appropriate. For the parameter estimation problem, *a posteriori* expected-value estimates and Bayesian optimal estimates are impractical to compute, except in special cases. The posterior distribution of ξ is not, in general, symmetric; thus the *a posteriori* expected value need not equal the MAP estimate.

8.1 DERIVATION

The basic method of derivation for the output-error method is to reduce the problem to the static form of Chapter 5. We will see that the dynamic system makes the models fairly complicated, but not different in any essential way from those of Chapter 5. We first consider the case where G and the initial condition are assumed to be known.

Choose an arbitrary value of ξ. Given the initial condition x_0 and a specified input time-history u, the state equation (8.0-1b) can be solved to give the state as a function of time. We assume that f is sufficiently smooth to guarantee the existence and uniqueness of the solution (Brauer and Noel, 1969). For complicated f functions, the solution may be difficult or impossible to express in closed form, but that aspect is irrelevant to the theory. (The practical implication is that the solution will be obtained using numerical approximation methods.) The important thing to note is that, because of the elimination of the process noise, the solution is deterministic.

For a specified input u, the system state is thus a deterministic function of ξ and time. For consistency with the notation of the filter-error method discussed later, denote this function by $\tilde{x}_\xi(t)$. The ξ subscript emphasizes the dependence on ξ. The dependence on u is not relevant to the current discussion, so the notation ignores this dependence for simplicity. Assuming known G, Equation (8.0-1c) then becomes

$$z(t_i) = g[x_\xi(t_i),u(t_i),\xi] + Gn_i \qquad i = 1,2,\ldots \tag{8.1-1}$$

Equation (8.1-1) is in the form of Equation (5.4-1); it is a static nonlinear model with additive noise. There are multiple experiments, one at each t_i. The estimators of Section 5.4 apply directly. The assumptions adopted have allowed us to solve the system dynamics, leaving an essentially static problem.

The MAP estimate is obtained by minimizing Equation (5.4-9). In the notation of this chapter, this equation becomes

$$J(\xi) = \frac{1}{2}\sum_{i=1}^{N}[z(t_i) - \tilde{z}_\xi(t_i)]^*(GG^*)^{-1}[z(t_i) - \tilde{z}_\xi(t_i)] + \frac{1}{2}(\xi - m_\xi)^*P^{-1}(\xi - m_\xi) \tag{8.1-2}$$

where

$$\tilde{x}_\xi(t_0) = x_0 \tag{8.1-3a}$$

$$\dot{\tilde{x}}_\xi(t) = f[x_\xi(t),u(t),\xi] \tag{8.1-3b}$$

$$\tilde{z}_\xi(t_i) = g[\tilde{x}_\xi(t_i),u(t_i),\xi] \qquad i = 1,2,\ldots \tag{8.1-3c}$$

The quantities m_ξ and P are the mean and covariance of the prior distribution of ξ, as in Chapter 5. For the MLE estimator, omit the last term of Equation (8.1-2), giving

$$J(\xi) = \frac{1}{2}\sum_{i=1}^{N}[z(t_i) - \tilde{z}_\xi(t_i)]^*(GG^*)^{-1}[z(t_i) - \tilde{z}_\xi(t_i)] \tag{8.1-4}$$

Equation (8.1-4) is a quadratic form in the difference between z, the measured response (output), and $\tilde{z}_\xi$, the response computed from the deterministic part of the system model. This motivates the name "output error." The minimization of Equation (8.1-4) is an intuitively plausible estimator defensible even without statistical derivation. The minimizing value of ξ gives the system model that best approximates (in a least-squares sense) the actual system response to the test input. Although this does not necessarily guarantee that the model response and the system response will be similar for other test inputs, the minimizing value of ξ is certainly a plausible estimate.

The estimates that result from minimizing Equation (8.1-4) are sometimes called "least squares" estimates, in reference to the quadratic form of the equation. We prefer to avoid the use of this terminology because it is potentially confusing. Many of the estimators applicable to dynamic systems have a least-squares form, so the term is not definitive. Furthermore, the term "least squares" is most often applied to Equation (8.1-4) to contrast it from other forms labeled "maximum likelihood" (typically the estimators of Section 8.4, which apply to unknown G, or the estimators of Chapter 9, which account for process noise). This contrast is misleading because Equation (8.1-4) describes a completely rigorous, maximum-likelihood estimator for the problem as posed. The differences between Equation (8.1-4) and the estimators of Sections 8.4 and Chapter 9 are differences in the problem statement, not differences in the statistical principles used for solution.

To derive the output-error method for pure discrete-time systems, substitute the discrete-time Equation (8.0-2b) in place of Equation (8.0-1b). The derivation and the result are unchanged except that Equation (8.1-3b) becomes

$$\tilde{x}_\xi(t_{i+1}) = f[\tilde{x}_\xi(t_i),u(t_i),\xi] \qquad i = 0,1,\ldots \tag{8.1-5}$$

8.2 INITIAL CONDITIONS

The above derivation of the output-error method assumed that the initial condition was known exactly. This assumption is seldom strictly true, except when using forms where the initial condition is zero by definition.

The initial condition is typically based on imperfectly measured data. This characteristic suggests treating the initial condition as a random variable with some mean and covariance. Such treatment, however, is incompatible with the output-error method. The output-error method is predicated on a deterministic solution of the state equation. Treatment of a random initial condition requires the more complex filter-error method discussed later.

If the system is stable, then initial condition effects decay to a negligible level in a finite time. If this decay is sufficiently fast and the error in the initial condition is sufficiently small, the initial condition error will have negligible effect on the system response and can be ignored.

If the errors in the initial condition are too large to justify neglecting them, there are several ways to resolve the problem without sacrificing the relative simplicity of the output-error method. One way is to simply improve the initial-condition values. This is sometimes trivially easy if the initial-condition value is computed from the measurement at the first time point of the maneuver (a common practice): change the start time by one sample to avoid an obvious wild point, average the first few data points, or draw a fairing through the noise and use the faired value.

When these methods are inapplicable or insufficient, we can include the initial condition in the list of unknown parameters to estimate. The initial condition is then a deterministic function of ξ. The solution of the state equation is thus still a deterministic function of ξ and time, as required for the output-error method. The equations of Section 5.1 still apply, provided that we substitute

$$\bar{x}_\xi(t_0) = x_0(\xi) \tag{8.2-1}$$

for Equation (8.3-1a).

It is easy to show that the initial-condition estimates have poor asymptotic properties as the time interval increases. The initial-condition information is all near the beginning of the maneuver, and increasing the time interval does not add to this information. Asymptotically, we can and should ignore initial conditions for stable systems. This is one case where asymptotic results are misleading. For real data with finite time intervals we should always carefully consider initial conditions. Thus, we avoid making the mistake of one published paper (which we will leave anonymous) which blithely set the model initial condition to zero in spite of clearly nonzero data. It is not clear whether this was a simple oversight or whether the author thought that asymptotic results justified the practice; in any event, the resulting errors were so egregious as to render the results worthless (except as an object lesson).

8.3 COMPUTATIONS

Equations (8.1-2) and (8.1-3) define the cost function that must be minimized to obtain the MAP estimates (or, in the special case that P^{-1} is zero, the MLE estimates). This is a fairly complicated function of ξ. Therefore we must use an iterative minimization scheme.

It is easy to become overwhelmed by the apparent complexity of J as a function of ξ; $\bar{z}_\xi(t_i)$ is itself a complicated function of ξ, involving the solution of a differential equation. To get J as a function of ξ we must substitute this function for $\bar{z}_\xi(t_i)$ in Equation (8.1-2). You might give up at the thought of evaluating first and second gradients of this function, as required by most iterative optimization methods. The complexity, however, is only apparent. It is crucial to recognize that we do not need to develop a closed-form expression, the development of which would be difficult at best. We are only required to develop a workable procedure for computing the result.

To evaluate the gradients of J, we need only proceed one step at a time; each step is quite simple, involving nothing more complicated than chain-rule differentiation. This step-by-step process follows the advice from *Alice in Wonderland:*

> The White Rabbit put on his spectacles. "Where shall I begin, please your Majesty?" he asked.
>
> "Begin at the beginning," the King said, very gravely, "and go on till you come to the end: then stop."

8.3.1 Gauss-Newton Method

The cost function is in the form of a sum of squares, which makes Gauss-Newton the preferred optimization algorithm. Sections 2.5.2 and 5.4.3 discussed the Gauss-Newton algorithm. To gather together all the important equations, we repeat the basic equations of the Gauss-Newton algorithm in the notation of this chapter. Gauss-Newton is a quasi-Newton algorithm. The full Newton-Raphson algorithm is

$$\hat{\xi}_{L+1} = \hat{\xi}_L - [\nabla_\xi^2 J(\hat{\xi}_L)]^{-1}[\nabla_\xi^* J(\hat{\xi}_L)] \tag{8.3-1}$$

The first gradient is

$$\nabla_\xi J(\xi) = -\sum_{i=1}^{N} [z(t_i) - \bar{z}_\xi(t_i)]^*(GG^*)^{-1}[\nabla_\xi \bar{z}(t_i)] + (\xi - m_\xi)^* P^{-1} \tag{8.3-2}$$

C-2

For the Gauss-Newton algorithm, we approximate the second gradient by

$$\nabla_\xi^2 J(\xi) \approx \sum_{i=1}^{N} [\nabla_\xi \tilde{z}_\xi(t_i)]^*(GG^*)^{-1}[\nabla_\xi \tilde{z}_\xi(t_i)] + P^{-1} \tag{8.3-3}$$

which corresponds to Equation (2.5-11) applied to the cost function of this chapter. Equations (8.3-1) through (8.3-3) are the same, whether the system is in pure discrete time or mixed continuous/discrete time. The only quantities in these equations requiring any discussion are $\tilde{z}_\xi(t_i)$ and $\nabla_\xi \tilde{z}_\xi(t_i)$.

8.3.2 System Response

The methods for computation of the system response depend on whether the system is pure discrete time or mixed continuous/discrete time. The choice of method is also influenced by whether the system is linear or nonlinear.

Computation of the response of discrete-time systems is simply a matter of plugging into the equations. The general equations for a nonlinear system are

$$\tilde{x}_\xi(t_0) = x_0(\xi) \tag{8.3-4a}$$

$$\tilde{x}_\xi(t_{i+1}) = f[\tilde{x}_\xi(t_i), u(t_i), \xi] \qquad i = 0,1,\ldots \tag{8.3-4b}$$

$$\tilde{z}_\xi(t_i) = g[\tilde{x}_\xi(t_i), u(t_i), \xi] \qquad i = 1,2,\ldots \tag{8.3-4c}$$

The more specific equations for a linear discrete-time system are

$$\tilde{x}_\xi(t_0) = x_0(\xi) \tag{8.3-5a}$$

$$\tilde{x}_\xi(t_{i+1}) = \Phi\tilde{x}_\xi(t_i) + \Psi u(t_i) \qquad i = 0,1,\ldots \tag{8.3-5b}$$

$$\tilde{z}_\xi(t_i) = C\tilde{x}_\xi(t_i) + Du(t_i) \qquad i = 1,2,\ldots \tag{8.3-5c}$$

For mixed continuous/discrete-time systems, numerical methods for approximate integration are required. You can use any of numerous numerical methods, but the utility of the more complicated methods is often limited by the available data. It makes little sense to use a high-order method to integrate the system equations between the time points where the input is measured. The errors implicit in interpolating the input measurements are probably larger than the errors in the integration method. For most purposes, a second-order Runge-Kutta algorithm is probably an appropriate choice:

$$\tilde{x}_\xi(t_0) = x_0(\xi) \tag{8.3-6a}$$

$$x_\xi'(t_{i+1}) = \tilde{x}_\xi(t_i) + (t_{i+1} - t_i)f[\tilde{x}_\xi(t_i), u(t_i), \xi] \tag{8.3-6b}$$

$$\tilde{x}_\xi(t_{i+1}) = \tilde{x}_\xi(t_i) + (t_{i+1} - t_i)\,\frac{1}{2}\,\{f[\tilde{x}_\xi(t_i), u(t_i), \xi] + f[x_\xi'(t_{i+1}), u(t_{i+1}), \xi]\} \tag{8.3-6c}$$

$$\tilde{z}_\xi(t_i) = g[\tilde{x}_\xi(t_i), u(t_i), \xi] \tag{8.3-6d}$$

For linear systems, a transition matrix method is more accurate and efficient than Equation (8.3-6).

$$\tilde{x}_\xi(t_0) = x_0(\xi) \tag{8.3-7a}$$

$$\tilde{x}_\xi(t_{i+1}) = \Phi\tilde{x}_\xi(t_i) + \Psi\,\frac{1}{2}\,[u(t_i) + u(t_{i+1})] \qquad i = 0,1,\ldots \tag{8.3-7b}$$

$$\tilde{z}_\xi(t_i) = C\tilde{x}_\xi(t_i) + Du(t_i) \qquad i = 1,2,\ldots \tag{8.3-7c}$$

where

$$\Phi = \exp[A(t_{i+1} - t_i)] \tag{8.3-8}$$

$$\Psi = \int_{t_i}^{t_{i+1}} \exp(A\tau)d\tau\ B \tag{8.3-9}$$

Section 7.5 discusses the form of Equation (8.3-7b). Moler and Van Loan (1978) describe several ways of numerically evaluating Equations (8.3-8) and (8.3-9). In this application, because $t_{i+1} - t_i$ is small compared to the system natural periods, simple series expansion works well.

$$\Phi = I + A\Delta + \frac{(A\Delta)^2}{2!} + \frac{(A\Delta)^3}{3!} + \ldots \tag{8.3-10}$$

$$\Psi = \Delta[I + \frac{A\Delta}{2!} + \frac{(A\Delta)^2}{3!} + \frac{(A\Delta)^3}{4!} + \ldots]B \tag{8.3-11}$$

where

$$\Delta = t_{i+1} - t_i \tag{8.3-12}$$

8.3.3 Finite Difference Response Gradient

It remains to discuss the computation of $\nabla_\xi \tilde{z}_\xi(t_i)$, the gradient of the system response. There are two basic methods for evaluating this gradient: finite-difference differentiation and analytic differentiation. This section discusses the finite difference approach, and the next section discusses the analytic approach.

Finite-difference differentiation is applicable to any model form. The method is easy to describe and equally easy to code. Because it is easy to code, finite-difference differentiation is appropriate for programs where quick results are needed or the production workload is small enough that saving program development time is more important than improving program efficiency. Because it applies with equal ease to all model forms, finite-difference differentiation is also appropriate for programs that must handle nonlinear models, for which analytic differentiation is numerically complicated (Jategaonkar and Plaetschke, 1983).

To use finite-difference differentiation, perturb the first element of the ξ vector by some small amount $\Delta\xi^{(1)}$. Recompute the system response using this perturbed ξ vector, obtaining the perturbed system response $\tilde{z}_p$. The partial derivative of the response with respect to $\xi^{(1)}$ is then approximately

$$\frac{\partial \tilde{z}_\xi(t_i)}{\partial \xi^{(1)}} \approx \frac{\tilde{z}_p(t_i) - \tilde{z}_\xi(t_i)}{\Delta\xi^{(1)}} \tag{8.3-13}$$

Repeat this process, perturbing each element of ξ in turn, to approximate the partial derivatives with respect to each element of ξ. The finite-difference gradient is then the concatenation of the partial derivatives.

$$\nabla_\xi \tilde{z}_\xi(t_i) = \left[\frac{\partial \tilde{z}_\xi(t_i)}{\partial \xi^{(1)}}, \frac{\partial \tilde{z}_\xi(t_i)}{\partial \xi^{(2)}}, \ldots\right] \tag{8.3-14}$$

Selection of the size of the perturbations requires some thought. If the perturbation is too large, Equation (8.3-13) becomes a poor approximation of the partial derivative. If the perturbation is too small, roundoff errors become a problem.

Some people have reported excellent results using simple perturbation-size rules such as setting the perturbation magnitude at 1% of a typical expected magnitude of the corresponding ξ element (assuming that you understand the problem well enough to be able to establish such typical magnitudes). You could alternatively consider percentages of the current iteration estimates (with some special provision for handling zero or essentially zero estimates). Another reasonable rule, after the first iteration, would be to use percentages of the diagonal elements of the second gradient, raised to the -1/2 power. As a final resort (it takes more computer time and is more complex), you could try several perturbation sizes, using the results to gauge the degree of nonlinearity and roundoff error, and adaptively selecting the best perturbation size.

Due to our limited experience with the finite difference approach, we defer making specific recommendations on perturbation sizes, but offer the opinion that the problem is amenable to reasonable solution. A little experimentation should suffice to establish an adequate perturbation-size rule for a specific class of problems. Note that the higher the precision of your computer, the more margin you have between the boundaries of linearity problems and roundoff problems. Those of us with 60- and 64-bit computers (or 32-bit computers in double precision) seldom have serious roundoff problems and can use simple perturbation-size rules with impunity. If you try to get by with single precision on a 32-bit computer, careful perturbation-size selection will be more important.

8.3.4 Analytic Response Gradient

The other approach to computing the gradient of the system response is to analytically differentiate the system equations. For linear systems, this approach is sometimes far more efficient than finite difference differentiation. For nonlinear systems, analytic differentiation is impractically clumsy (partially because you have to redo it for each new nonlinear model form). We will, therefore, restrict our discussion of analytic differentiation to linear systems.

We first consider pure discrete-time linear systems in the form of Equation (8.3-5). It is crucial to recall that we do not need a closed form for the gradient; we only need a method for computing it. A closed-form expression would be formidable, unlike the following equation, which is the almost embarassingly obvious gradient of Equation (8.3-5), obtained by using nothing more complicated than the chain rule:

$$\nabla_\xi \tilde{x}(t_0) = \nabla_\xi x_0(\xi) \tag{8.3-13a}$$

$$\nabla_\xi \tilde{x}(t_{i+1}) = \Phi(\nabla_\xi \tilde{x}(t_i)) + (\nabla_\xi \Phi)\tilde{x}(t_i) + (\nabla_\xi \Psi)u(t_i) \qquad i = 0,1,\ldots \tag{8.3-13b}$$

$$\nabla_\xi \tilde{z}(t_i) = C(\nabla_\xi \tilde{x}(t_i)) + (\nabla_\xi C)\tilde{x}(t_i) + (\nabla_\xi D)u(t_i) \qquad i = 1,2,\ldots \tag{8.3-13c}$$

Equation (8.3-13b) gives a recursive formula for $\nabla_\xi \tilde{x}(t_i)$, with Equation (8.3-13a) as the initial condition. Equation (8.3-13c) expresses $\nabla_\xi \tilde{z}(t_i)$ in terms of the solution of Equation (8.3-13b).

The quantities $\nabla_\xi\Phi$, $\nabla_\xi\Psi$, $\nabla_\xi C$, and $\nabla_\xi D$ in Equation (8.3-13) are gradients of matrices with respect to the vector ξ. The results are vectors, the elements of which are matrices (if you are fond of buzz words, these are third-order tensors). If this starts to sound complicated, you will be pleased to know that the products like $(\nabla_\xi D)u(t_i)$ are ordinary matrices (and indeed sparse matrices—they have lots of zero elements). You can compute the products directly without ever forming the vector of matrices in your program. A program to implement Equation (8.3-13) takes fewer lines than the explanation.

We could write Equation (8.3-13) without using gradients or matrices. Simply replace ∇_ξ by $\partial/\partial\xi^{(j)}$ throughout, and then concatenate the partial derivatives to get the gradient of $\tilde{z}(t_i)$. We than have, at worst, partial derivatives of matrices with respect to scalars; these partial derivatives are matrices. The only difference between writing the equations with partial derivatives or gradients is notational. We choose to use the gradient notation because it is shorter and more consistent with the rest of the book.

Let us look at Equation (8.3-13c) in detail to see how these equations would be implemented in a program, and perhaps to better understand the equations. The left-hand side is a matrix. Each column of the matrix is the partial derivative of $\tilde{z}(t_i)$ with respect to one element of ξ:

$$\nabla_\xi\tilde{z}(t_i) = \left[\frac{\partial}{\partial\xi^{(1)}}\tilde{z}(t_i)\ ,\ \frac{\partial}{\partial\xi^{(2)}}\tilde{z}(t_i)\ \ldots\ \frac{\partial}{\partial\xi^{(p)}}\tilde{z}(t_i)\right] \tag{8.3-14}$$

The quantity $\nabla_\xi\tilde{x}(t_i)$ is a similar matrix, computed from Equation (8.3-13b); thus $C(\nabla_\xi\tilde{x}(t_i))$ is a multiplication of a matrix times a matrix, and this is a calculation we can handle. The quantity $\nabla_\xi C$ is the vector of matrices

$$\nabla_\xi C = \left[\frac{\partial C}{\partial\xi^{(1)}}\ ,\ \frac{\partial C}{\partial\xi^{(2)}}\ \ldots\ \frac{\partial C}{\partial\xi^{(p)}}\right] \tag{8.3-15}$$

and the product $(\nabla_\xi C)\tilde{x}(t_i)$ is

$$(\nabla_\xi C)\tilde{x}(t_i) = \left[\frac{\partial C}{\partial\xi^{(1)}}\tilde{x}(t_i)\ ,\ \frac{\partial C}{\partial\xi^{(2)}}\tilde{x}(t_i)\ \ldots\ \frac{\partial C}{\partial\xi^{(p)}}\tilde{x}(t_i)\right] \tag{8.3-16}$$

(Our notation does not indicate explicitly that this is the intended product formula, but the other conceivable interpretation of the notation is obviously wrong because the dimensions are incompatible. Formal tensor notation would make the intention explicit, but we do not really need to introduce tensor notation here because the correct interpretation is obvious).

In many cases the matrix $\partial C/\partial\xi^{(j)}$ will be sparse. Typically these matrices are either zero or have only one nonzero element. We can take advantage of such sparseness in the computation. If C is not a function of $\xi^{(j)}$ (presumably $\xi^{(j)}$ affects other of the system matrices), then $\partial C/\partial\xi^{(j)}$ is a zero matrix. If only the (k,m) element of C is affected by $\xi^{(j)}$, then $[\partial C/\partial\xi^{(j)}]\tilde{x}(t_i)$ is a vector with $[\partial C^{(k,m)}/\partial\xi^{(j)}]\tilde{x}(t_i)^{(m)}$ in the kth element and zeros elsewhere. If more than one element of C is affected by $\xi^{(j)}$, then the result is a sum of such terms. This approach directly forms $[\partial C/\partial\xi^{(j)}]\tilde{x}(t_i)$, taking advantage of sparseness, instead of forming the full $\partial C/\partial\xi^{(j)}$ matrix and using a general-purpose matrix multiply routine. The terms $(\nabla_\xi D)u(t_i)$, $(\nabla_\xi\Phi)\tilde{x}(t_i)$, and $(\nabla_\xi\Psi)u(t_i)$ are all similar in form to $(\nabla_\xi C)\tilde{x}(t_i)$. The initial condition $\nabla_\xi x_0$ is a zero matrix if x_0 is known; otherwise it has a nonzero element for each unknown element of x_0.

We now know how to evaluate all of the terms in Equation (8.4-13). This is significantly faster than finite differences for some applications. The speed-up is most significant if Φ, Ψ, C, and D are functions of time requiring significant work to evaluate at each point; straighforward finite difference methods would have to reevaluate these matrices for each perturbation.

Gupta and Mehra (1974) discuss a method that is basically a modification of Equation (8.3-13) for computing $\nabla_\xi\tilde{z}(t_i)$. Depending on the number of inputs, states, outputs, and unknown parameters, this method can sometimes save computer time by reducing the length of the gradient vector needed for propagation in Equation (8.4-13).

We now have everything needed to implement the basic Gauss-Newton minimization algorithm. Practical application will typically require some kind of start-up algorithm and methods for handling cases where the algorithm converges slowly or diverges. The Iliff-Maine code, MMLE3 (Maine and Iliff, 1980; and Maine, 1981), incorporates several such modifications. The line-search ideas (Foster, 1983) briefly discussed at the end of Section 2.5.2 also seem appropriate for handling convergence problems. We will not cover the details of such practical issues here.

The discussions of singularities in Section 5.4.4 and of partitioning in Section 5.4.5 apply directly to the problem of this chapter, so we will not repeat them.

8.4 UNKNOWN G

The previous discussion in this chapter has assumed that the G-matrix is known. Equations (8.1-2) and (8.1-4) are derived based on this assumption. For unknown G, the methods of Section 5.5 apply directly. Equation (5.5-2) substitutes for Equation (8.1-4). In the terminology of this chapter, Equation (5.5-2) becomes

$$J(\xi) = \frac{1}{2}\sum_{i=1}^{N}[z(t_i) - \tilde{z}_\xi(t_i)]^*[G(\xi)G(\xi)^*]^{-1}[z(t_i) - \tilde{z}_\xi(t_i)] + \ln|G(\xi)G(\xi)^*| \tag{8.4-1}$$

If G is known, this reduces to Equation (8.1-4) plus a constant.

As discussed in Section 5.5, the best approach to minimizing Equation (8.4-1) is to partition the parameter vector into a part ξ_G affecting G, and a part ξ_f affecting $\tilde{z}$. For each fixed G, the Gauss-Newton equations of Section 8.3 apply to revising the estimate of ξ_f. For each fixed ξ_f, the revised estimate of G is given by Equation (5.5-7), which becomes

$$\hat{G}\hat{G}^* = \frac{1}{N}\sum_{i=1}^{N}[z(t_i) - \tilde{z}(t_i)][z(t_i) - \tilde{z}_\xi(t_i)]^* \tag{8.4-2}$$

in the current notation. Section 5.5 describes the axial iteration method, which alternately applies the Gauss-Newton equations of Section 8.3 for ξ_f and Equation (8.4-2) for G.

The cost function for estimation with unknown G is often written in alternate forms. Although the above form is usually the most useful for computation, the following forms provide some insight into the relations of the estimators with unknown G versus those with fixed G. When G is completely unknown, the minimization of Equation (8.4-1) is equivalent to the minimization of

$$J(\xi) = \left|\frac{1}{N}\sum_{i=1}^{N}[z(t_i) - \tilde{z}_\xi(t_i)][z(t_i) - \tilde{z}_\xi(t_i)]^*\right| \tag{8.4-3}$$

which corresponds to Equation (5.5-9). Section 5.5 derives this equivalence by eliminating G. It is common to restrict G to be diagonal, in which case Equation (8.4-3) becomes

$$J(\xi) = \prod_{j=1}^{\ell}\left\{\frac{1}{N}\sum_{i=1}^{N}[z(t_i)^{(j)} - \tilde{z}_\xi(t_i)^{(j)}]\right\} \tag{8.4-4}$$

This form is a product of the errors in the different signals, instead of the weighted sum-of-the-errors form of Equation (8.1-4).

8.5 CHARACTERISTICS

We have shown that the output error estimator is a direct application of the estimators derived in Section 5.4 for nonlinear static systems. To describe the statistical characteristics of output error estimates, we need only apply the corresponding Section 5.4 results to the particular form of output error.

In most cases, the corresponding static system is nonlinear, even for linear dynamic systems. Therefore, we must use the forms of Section 5.4 instead of the simpler forms of Section 5.1, which apply to linear static systems. In particular, the output error MLE and MAP estimators are both biased for finite time. Asymptotically, they are unbiased and efficient.

From Equation (5.4-11), the covariance of the MLE output error estimate is approximated by

$$\mathrm{cov}(\hat{\xi}|\xi) \approx \left\{\sum_{i=1}^{N}[\nabla_\xi\tilde{z}_\xi(t_i)]^*(GG^*)^{-1}[\nabla_\xi\tilde{z}_\xi(t_i)]\right\}^{-1} \tag{8.5-1}$$

From Equation (5.4-12), the corresponding approximation for the posterior distribution of ξ in an MAP estimator is

$$\mathrm{cov}(\xi|Z) \approx \left\{\sum_{i=1}^{N}[\nabla_\xi\tilde{z}_\xi(t_i)]^*(GG^*)^{-1}[\nabla_\xi\tilde{z}_\xi(t_i)] + P^{-1}\right\}^{-1} \tag{8.5-2}$$

CHAPTER 9

9.0 FILTER ERROR METHOD FOR DYNAMIC SYSTEMS

In this chapter, we consider the parameter estimation problem for dynamic systems with both process and measurement noise. We restrict the consideration to linear systems with additive Gaussian noise, because the exact analysis of more general systems is impractically complicated except in special cases like output error (no process noise).

The easiest way to handle nonlinear systems with both measurement and process noise is usually to linearize the system and apply the linear results. This method does not give exact results for nonlinear systems, but can give adequate approximations in some cases.

In mixed continuous/discrete time, the linear system model is

$$x(t_0) = x_0 \tag{9.0-1a}$$

$$\dot{x}(t) = Ax(t) + Bu(t) + Fn(t) \tag{9.0-1b}$$

$$z(t_i) = Cx(t_i) + Du(t_i) + G\eta_i \qquad i = 1,2,\ldots \tag{9.0-1c}$$

The measurement noise η is assumed to be a sequence of independent Gaussian random variables with zero mean and identity covariance. The process noise n is a zero-mean, white-noise process, independent of the measurement noise, with identity spectral density. The initial condition x_0 is assumed to be a Gaussian random variable, independent of n and η, with mean $\hat{x}_0$ and covariance P_0. As special cases, P_0 can be 0, implying that the initial condition is known exactly; or infinite, implying complete ignorance of the initial condition. The input u is assumed to be known exactly.

As in the case of output error, the system matrices A, B, C, D, F, and G, are functions of ξ and may be functions of time.

The corresponding pure discrete-time model is

$$x(t_0) = x_0 \tag{9.0-2a}$$

$$x(t_{i+1}) = \Phi x(t_i) + \Psi u(t_i) + Fn_i \qquad i = 0,1,\ldots \tag{9.0-2b}$$

$$z(t_i) = Cx(t_i) + Du(t_i) + G\eta_i \qquad i = 1,2,\ldots \tag{9.0-2c}$$

All of the same assumptions apply, except that n is a sequence of independent Gaussian random variables with zero mean and identity covariance.

9.1 DERIVATION

In order to obtain the maximum likelihood estimate of ξ, we need to choose $\hat{\xi}$ to maximize $L(\xi,Z) = p(Z_N|\xi)$ where

$$Z_N = [z(t_1),z(t_2)\ldots z(t_N)]^* \tag{9.1-1}$$

For the MAP estimate, we need to maximize $p(Z_N|\xi)p(\xi)$. In either event, the crucial first step is to find a tractable expression for $p(Z_N|\xi)$. We will discuss three ways of deriving this density function.

9.1.1 Static Derivation

The first means of deriving an expression for $p(Z_N|\xi)$ is to solve the system equations, reducing them to the static form of Equation (5.0-1). This technique, although simple in principle, does not give a tractable solution. We briefly outline the approach here in order to illustrate the principle, before considering the more fruitful approaches of the following sections.

For a pure discrete-time linear system described by Equation (9.0-2), the explicit static expression for $z(t_i)$ is

$$z(t_i) = C\Phi^i x(t_0) + C\sum_{j=0}^{i-1}\Phi^{i-j-1}(\Psi u(t_j) + Fn_j) + Du(t_i) + G\eta_i \tag{9.1-2}$$

This is a nonlinear static model in the general form of Equation (5.5-1). However, the separation of ξ into ξ_G and ξ_f as described by Equation (5.5-4) does not apply. Note that Equation (9.1-2) is a nonlinear function of ξ, even if the matrices are linear functions. In fact, the order of nonlinearity increases with the number of time points. The use of estimators derived directly from Equation (9.1-2) is unacceptably difficult for all but the simplest special cases, and we will not pursue it further.

For mixed continuous/discrete-time systems, similar principles apply, except that the ω of Equation (5.0-1) must be generalized to allow vectors of infinite dimension. The process noise in a mixed continuous/discrete-time system is a function of time, and cannot be written as a finite-dimensional random vector. The material of Chapter 5 covered only finite-dimensional vectors. The Chapter 5 results generalize

PRECEDING PAGE BLANK NOT FILMED

PRECEDING PAGE BLANK NOT FILMED

PAGE 96 INTENTIONALLY BLANK

nicely to infinite-dimensional vector spaces (function spaces), but we will not find that level of abstraction necessary. Application to pure continuous-time systems would require further generalization to allow infinite-dimensional observations.

9.1.2 Derivation by Recursive Factoring

We will now consider a derivation based on factoring $p(Z_N|\xi)$ by means of Bayes rule (Equation (3.3-12)). The derivation applies either to pure discrete-time or mixed continuous/discrete-time systems; the derivation is identical in both cases. For the first step, write

$$p(Z_N|\xi) = p(z(t_N)|Z_{N-1},\xi)p(Z_{N-1}|\xi) \tag{9.1-3}$$

Recursive application of this formula gives

$$p(Z_N|\xi) = \prod_{i=1}^{N} p(z(t_i)|Z_{i-1},\xi) \tag{9.1-4}$$

For any particular ξ, the distribution of $z(t_i)$ given Z_{i-1} is known from the Chapter 7 results; it is Gaussian with mean

$$\begin{aligned}\tilde{z}_\xi(t_i) &\equiv E\{z(t_i)|Z_{i-1},\xi\}\\ &= E\{Cx(t_i) + Du(t_i) + G\eta_i|Z_{i-1},\xi\}\\ &= C\tilde{x}_\xi(t_i) + Du(t_i)\end{aligned} \tag{9.1-5}$$

and covariance

$$\begin{aligned}R_i &\equiv \mathrm{cov}(z(t_i)|Z_{i-1},\xi)\\ &= \mathrm{cov}(Cx(t_i) + Du(t_i) + G\eta_i|Z_{i-1},\xi)\\ &= CQ(t_i)C^* + GG^*\end{aligned} \tag{9.1-6}$$

Note that $\tilde{x}_\xi(t_i)$ and $\tilde{z}_\xi(t_i)$ are functions of ξ because they are obtained from the Kalman filter based on a particular value of ξ; that is, they are conditioned on ξ. We use the ξ subscript notation to emphasize this dependence. R_i is also a function of ξ, although our notation does not explicitly indicate this.

Substituting the appropriate Gaussian density functions characterized by Equations (9.1-5) and (9.1-6) into Equation (9.1-4) gives

$$L(\xi,Z_N) \equiv p(Z_N|\xi) = \prod_{i=1}^{N} |2\pi R_i|^{-1/2} \exp\left\{-\frac{1}{2}[z(t_i) - \tilde{z}_\xi(t_i)]^* R_i^{-1}[z(t_i) - \tilde{z}_\xi(t_i)]\right\} \tag{9.1-7}$$

This is the desired expression for the likelihood functional.

9.1.3 Derivation Using the Innovation

Another derivation involves the properties of the innovation. This derivation also applies either to mixed continuous/discrete-time or to pure discrete-time systems.

We proved in Chapter 7 that the innovations are a sequence of independent, zero-mean Gaussian variables with covariances R_i given by Equation (7.2-33). This proof was done for the pure discrete-time case, but extends directly to mixed continuous/discrete-time systems. The Chapter 7 results assumed that the system matrices were known; thus the results are conditioned on ξ. The conditional probability density function of the innovations is therefore

$$p(V_N|\xi) = \prod_{i=1}^{N} |2\pi R_i|^{-1/2} \exp\left(-\frac{1}{2} v_i^* R_i^{-1} v_i\right) \tag{9.1-8}$$

We also showed in Chapter 7 that the innovations are an invertible linear function of the observations. Furthermore, it is easy to show that the determinant of the Jacobian of the transformation equals 1. (The Jacobian is triangular with 1's on the diagonal). Thus by Equation (3.4-1), we can substitute

$$v_i = z(t_i) - \tilde{z}_\xi(t_i) \tag{9.1-9}$$

into Equation (9.1-8) to give

$$p(Z_N|\xi) = \prod_{i=1}^{N} |2\pi R_i|^{-1/2} \exp\left\{-\frac{1}{2}[z(t_i) - \tilde{z}_\xi(t_i)]^* R_i^{-1}[z(t_i) - \tilde{z}_\xi(t_i)]\right\} \tag{9.1-10}$$

which is identical to Equation (9.1-7). We see that the derivation by Bayes factoring and the derivation using the innovation give the same result.

9.1.4 Steady-State Form

For many applications, we can use the time steady-state Kalman filter in the cost functional, resulting in major computational savings. This usage requires, of course, that the steady-state filter exist. We discussed the criteria for the existence of the steady-state filter in Chapter 7. The most important criterion is obviously that the system be time-invariant. The rest of this section assumes that a steady-state form exists. When a steady-state form exists, two approaches can be taken to justifying its use.

The first justification is that the steady-state form is a good approximation if the time interval is long enough. The time-varying filter gain converges to the steady-state gain with time constants at least as fast as those of the open-loop system, and sometimes significantly faster. Thus, if the maneuver analyzed is long compared to the system time constants, the filter gain would converge to the steady-state gain in a small portion of the maneuver time. We could verify this behavior by computing time-varying gains for representative values of ξ. If the filter gain does converge quickly to the steady-state gain, then the steady-state filter should give a good approximation to the cost functional.

The second possible justification for the use of the steady-state filter involves the choice of the initial state covariance P_0. The time-varying filter requires P_0 to be specified. It is a common practice to set P_0 to zero. This practice arises more from a lack of better ideas than from any real argument that zero is a good value. It is seldom that we know the initial state exactly as implied by the zero covariance. One circumstance which would justify the zero initial covariance would be the case where the initial condition is included in the list of unknown parameters. In this case, the initial covariance is properly zero because the filter is conditioned on the values of the unknown parameters. Any prior information about the initial condition is then reflected in the prior distribution of ξ instead of in P_0. Unless one has a specific need for estimates of the initial condition, there are usually better approaches.

We suggest that the steady-state covariance is often a reasonable value for the initial covariance. In this case, the time-varying and steady-state filters are identical; arguments about the speed of convergence and the length of the data interval are not required. Since the time-varying form requires significantly more computation than the steady-state form, the steady-state form is preferable except where it is clearly and significantly inferior.

If the steady-state filter is used, Equation (9.1-7) becomes

$$L(\xi,Z_N) = \prod_{i=1}^{N} |2\pi R|^{-1/2} \exp\{[z(t_i) - \bar{z}_\xi(t_i)]^* R^{-1}[z(t_i) - \bar{z}_\xi(t_i)]\} \tag{9.1-11}$$

where R is the steady-state covariance of the innovation. In general, R is a function of ξ. The $\bar{z}_\xi(t_i)$ in Equation (9.1-11) comes from the steady-state filter, unlike the $\bar{z}_\xi(t_i)$ in Equation (9.1-7). We use the same notation for both quantities, distinguishing them by context. (The $\bar{z}_\xi(t_i)$ from the steady-state filter is always associated with the steady-state covariance R, whereas the $\bar{z}_\xi(t_i)$ from the time-varying filter is associated with the time-varying covariance R_i.)

9.1.5 Cost Function Discussion

The maximum-likelihood estimate of ξ is obtained by maximizing Equation (9.1-11) (or Equation (9.1-7) if the steady-state form is inappropriate) with respect to ξ.

Because of the exponential in Equation (9.1-11), it is more convenient to work with the logarithm of the likelihood functional, called the log likelihood functional for short. The log likelihood functional is maximized by the same value of ξ that maximizes the likelihood functional because the logarithm is a monotonic increasing function. By convention, most optimization theory is written in terms of minimization instead of maximization. We therefore define the negative of the log likelihood functional to be a cost functional which is to be minimized. We also omit the $\ln(2\pi)$ term from the cost functional, because it does not affect the minimization. The most convenient expression for the cost functional is then

$$J(\xi) = \frac{1}{2}\sum_{i=1}^{N} [z(t_i) - \bar{z}_\xi(t_i)]^* R^{-1}[z(t_i) - \bar{z}_\xi(t_i)] + \frac{1}{2} N \ln|R| \tag{9.1-12}$$

If R is known, then Equation (9.1-12) is in a least-squares form. This is sometimes called a prediction-error form because the quantity being minimized is the square of the one-step-ahead prediction error $z(t_i) - \bar{z}_\xi(t_i)$. The term "filter error" is also used because the quantity minimized is obtained from the Kalman filter.

Note that this form of the likelihood functional involves the Kalman filter—not a smoother. There is sometimes a temptation to replace the filter in this cost function by a smoother, assuming that this will give improved results. The smoother gives better state estimates than the filter, but the problem considered in this chapter is not state estimation. The state estimates are an incidental side-product of the algorithm for estimating the parameter vector ξ. There are ways of deriving and writing the parameter estimation problem which involve smoothers (Cox and Bryson, 1980), but the direct use of a smoother in Equation (9.1-12) is simply incorrect.

For MAP estimates, we modify the cost functional by adding the negative of the logarithm of the prior probability density of ξ. If the prior distribution of ξ is Gaussian with mean m_ξ and covariance W, the cost functional of Equation (9.1-12) becomes (ignoring constant terms)

$$J(\xi) = \frac{1}{2}\sum_{i=1}^{N}[z(t_i) - \hat{z}_\xi(t_i)]^*R^{-1}[z(t_i) - \hat{z}_\xi(t_i)] + \frac{1}{2}N\ \ln|R| + \frac{1}{2}(\xi - m_\xi)^*W^{-1}(\xi - m_\xi) \qquad (9.1\text{-}13)$$

The filter-error forms of Equations (9.1-12) and (9.1-13) are parallel to the output-error forms of Equations (8.1-4) and (8.1-2). When there is no process noise, the steady-state Kalman filter becomes an integration of the system equations, and the innovation covariance R equals the measurement noise covariance GG^*. Thus the output error equations of the previous chapter are special cases of the filter error equations with zero process noise.

9.2 COMPUTATION

The best methods for minimizing Equation (9.1-12) or (9.1-13) are based on the Gauss-Newton algorithm. Because these equations are so similar in form to the output-error equations of Chapter 8, most of the Chapter 8 material on computation applies directly or with only minor modification.

The primary differences between computational methods for filter error and those for output error center on the treatment of the noise covariances, particularly when the covariances are unknown. Maine and Iliff (1981a) discuss the implementation details of the filter-error algorithm. The Iliff-Maine code, MMLE3 (Maine and Iliff, 1980; and Maine, 1981), implements the filter-error algorithm for linear continuous/discrete-time systems.

We generally presume the use of the steady-state filter in the filter-error algorithm. Implementation is significantly more complicated using the time-varying filter.

9.3 FORMULATION AS A FILTERING PROBLEM

An alternative to the direct approach of the previous section is to recast the parameter estimation problem into the form of a filtering problem. The techniques of Chapter 7 then apply.

Suppose we start with the system model

$$x(t_0) = x_0 \qquad (9.3\text{-}1a)$$

$$\dot{x}(t) = A(\xi)x(t) + B(\xi)u(t) + Fn(t) \qquad (9.3\text{-}1b)$$

$$z(t_i) = C(\xi)x(t_i) + D(\xi)u(t_i) + Gn_i \qquad (9.3\text{-}1c)$$

This is the same as Equation (9.0-1), except that here we explicitly indicate the dependence of the matrices on ξ. The problem is to estimate ξ.

In order to apply state estimation techniques to this problem, ξ must be part of the state vector. Therefore, we define an augmented state vector

$$x_a = \begin{bmatrix} x \\ \xi \end{bmatrix} \qquad (9.3\text{-}2)$$

We can combine Equation (9.3-1) with the trivial differential equation

$$\dot{\xi} = 0 \qquad (9.3\text{-}3)$$

to write a system equation with x_a as the state vector. Note that the resulting system is nonlinear in x_a (because it has products of ξ and x), even though Equation (9.3-1) is linear in x.

In principle, we can apply the extended Kalman filter, discussed in Section 7.7, to the problem of estimating x_a. Unfortunately, the nonlinearity in the augmented system is crucial to the system behavior. The adequacy of the extended Kalman filter for this problem has seldom been analyzed in detail. Schweppe (1973, p. 433) says on this subject

> ...the system identification problem has been transformed into a problem which has already been discussed extensively.
>
> The discussions are not terminated at this point for the simple reason that Part IV did not provide any "best" one way to solve a nonlinear state estimation problem. A major conclusion of Part IV was that the best way to proceed depends heavily on the explicit nature of the problem. System identification leads to special types of nonlinear estimation problems, so specialized discussions are needed.
>
> ...the state augmentation approach is not emphasized, as the author feels that it is much more appropriate to approach the system identification problem directly. However, there are special cases where state augmentation works very well.

CHAPTER 10

10.0 EQUATION ERROR METHOD FOR DYNAMIC SYSTEMS

This chapter discusses the equation error approach to parameter estimation for dynamic systems. We will first define a restricted form of equation error, parallel to the treatments of output error and filter error in the previous chapters. This form of equation error is a special case of filter error where there is process noise, but no measurement noise. It therefore stands in counterpoint to output error, which is the special case where there is measurement noise, but no process noise.

We will then extend the definition of equation error to a more general form. Some of the practical applications of equation error do not fit precisely into the overly restrictive form based on process noise only. In its most general forms, the term equation error encompasses output error and filter error, in addition to the forms most commonly associated with the term. The primary distinguishing feature of the methods emphasized in this chapter is their computational simplicity.

10.1 PROCESS-NOISE APPROACH

In this section, we consider equation error in a manner parallel to the previous treatments of output error and filter error. The filter-error method treats systems with both process noise and measurement noise, and output error treats the special case of systems with measurement noise only. Equation error completes this triad of algorithms by treating the special case of systems with process noise only.

The equation-error method applies to nonlinear systems with additive Gaussian process noise. We will restrict the discussion of this section to pure discrete-time models, for which the derivation is straightforward. Mixed continuous/discrete-time models can be handled by converting them to equivalent pure discrete-time models. Equation error does not strictly apply to pure continuous-time models. (The problem becomes ill-posed).

The general form of the nonlinear, discrete-time system model we will consider is

$$x(t_0) = x_0 \tag{10.1-1a}$$

$$x(t_{i+1}) = f[x(t_i),u(t_i),\xi] + Fn_i \qquad i = 0,1,\ldots,N-1 \tag{10.1-1b}$$

$$z(t_i) = g[x(t_i),u(t_i),\xi] \qquad i = 0,1,\ldots,N \tag{10.1-1c}$$

The process noise, n, is a sequence of independent Gaussian random variables with zero mean and identity covariance. The matrix F can be a function of ξ, although the simplified notation ignores this possibility. It will prove convenient to assume that the measurements $z(t_i)$ are defined for $i = 0,\ldots,N$; previous chapters have defined them only for $i = 1,\ldots,N$.

10.1.1 Derivation

The following derivation of the equation-error method closely parallels the derivation of the filter-error method in Section 9.1.3. Both are based primarily on application of the transformation of variables formula, Equation (3.4-1), starting from a process known to be a sequence of independent Gaussian variables.

By assumption, the probability density function of the process noise is

$$p(n_N) = \prod_{i=0}^{N-1} (2\pi)^{-1/2} \exp(n_i^* n_i) \tag{10.1-2}$$

where n_N is the concatenation of the n_i. We further assume that F is invertible for all permissible values of ξ; this assumption is necessary to ensure that the problem is well-posed. We define X_N to be the concatenation of the $x(t_i)$. Then, for each value of ξ, X_N is an invertible linear function of n_N. The inverse function is

$$n_i = F^{-1}[x(t_{i+1}) - \tilde{x}_\xi(t_{i+1})] \tag{10.1-3}$$

where, for convenience and for consistency with the notation of previous chapters, we have defined

$$\tilde{x}_\xi(t_{i+1}) = f[x(t_i),u(t_i),\xi] \tag{10.1-4}$$

The determinant of the Jacobian of the inverse transformation is $|F|^{-N}$ because the inverse transformation matrix is block-triangular with F^{-1} in the diagonal blocks. Direct application of the transformation-of-variables formula, Equation (3.4-1), gives

$$p(X_N|\xi) = \prod_{i=1}^{N} |2\pi FF^*|^{-1/2} \exp\left\{-\tfrac{1}{2}\,[x(t_i) - \tilde{x}_\xi(t_i)]^*(FF^*)^{-1}\,[x(t_i) - \tilde{x}_\xi(t_i)]\right\} \tag{10.1-5}$$

In order to derive a simple expression for $p(Z_N|\xi)$, we require that g be a continuous, invertible function of x for each value of ξ. The invertibility is critical to the simplicity of the equation-error

algorithm. This assumption, combined with the lack of measurement noise, means that we can reconstruct the state vector perfectly, provided that we know ξ. The inverse function gives this reconstruction:

$$\hat{x}_\xi(t_i) = g^{-1}[z(t_i),u(t_i),\xi] \tag{10.1-6}$$

If g is not invertible, a recursive state estimator becomes imbedded in the algorithm and we are again faced with something as complicated as the filter-error algorithm. For invertible g, the transformation-of-variables formula, Equation (3.4-1), gives

$$p(Z_N|\xi) = \prod_{i=1}^{N} \left| \det \frac{\partial g^{-1}[z(t_i),u(t_i),\xi]}{\partial z(t_i)} \right| |2\pi FF^*|^{-1/2} \exp\left\{-\frac{1}{2}[\hat{x}_\xi(t_i) - \tilde{x}_\xi(t_i)]^*(FF^*)^{-1}[\hat{x}_\xi(t_i) - \tilde{x}_\xi(t_i)]\right\} \tag{10.1-7}$$

where $\hat{x}_\xi(t_i)$ is given by Equation (10.1-6), and

$$\tilde{x}_\xi(t_i) = f[\hat{x}_\xi(t_{i-1}),u(t_{i-1}),\xi] \tag{10.1-8}$$

Most practical applications of equation error separate the problems of state reconstruction and parameter estimation. In the context defined above, this is possible when g is not a function of ξ. Then Equation (10.1-6) is also independent of ξ; thus, we can reconstruct the state exactly without knowledge of ξ. Furthermore, the estimates of ξ depend only on the reconstructed state vector and the control vector. There is no direct dependence on the actual measurements $z(t_i)$ or on the exact form of the g-function. This is evident in Equation (10.1-7) because the Jacobian of g^{-1} is independent of ξ and, therefore, irrelevant to the parameter-estimation problem. In many practical applications, the state reconstruction is more complicated than a simple pointwise function as in Equation (10.1-6), but as long as the state reconstruction does not depend on ξ, the details do not matter to the parameter-estimation process.

You will seldom (if ever) see Equation (10.1-7) elsewhere in the form shown here, which includes the factor for the Jacobian of g^{-1}. The usual derivation ignores the measurement equation and starts from the assumption that the state is known exactly, whether by direct measurement or by some reconstruction. We have included the measurement equation only in order to emphasize the parallels between equation error, output error, and filter error. For the rest of this section, we will assume that g is independent of ξ. We will specifically assume that the determinant of the Jacobian of g is 1 (the actual value being irrelevant to the estimator anyway), so that we can write Equation (10.1-7) in a more conventional form as

$$p(Z_N|\xi) = \prod_{i=1}^{N} |2\pi FF^*|^{-1/2} \exp\left\{-\frac{1}{2}[x(t_i) - \tilde{x}_\xi(t_i)]^*(FF^*)^{-1}[x(t_i) - \tilde{x}_\xi(t_i)]\right\} \tag{10.1-9}$$

where

$$\tilde{x}_\xi(t_i) = f[x(t_{i-1}),u(t_{i-1}),\xi] \tag{10.1-10}$$

You can derive slight generalizations, useful in some cases, from Equation (10.1-7).

The maximum-likelihood estimate of ξ is the value that maximizes Equation (10.1-9). As in previous chapters, it is convenient to work in terms of minimizing the negative-log-likelihood functional

$$J(\xi) = \frac{1}{2}\sum_{i=1}^{N} [x(t_i) - \tilde{x}_\xi(t_i)]^*(FF^*)^{-1}[x(t_i) - \tilde{x}_\xi(t_i)] + \frac{1}{2} N \ln|FF^*| \tag{10.1-11}$$

If ξ has a Gaussian prior distribution with mean m_ξ and covariance P, then the MAP estimate minimizes

$$J(\xi) = \frac{1}{2}\sum_{i=1}^{N} [x(t_i) - \tilde{x}_\xi(t_i)]^*(FF^*)^{-1}[x(t_i) - \tilde{x}_\xi(t_i)] + \frac{1}{2} N \ln|FF^*| + \frac{1}{2}(\xi - m_\xi)^*P^{-1}(\xi - m_\xi) \tag{10.1-12}$$

10.1.2 Special Case of Filter Error

For linear systems, we can also derive state-equation error by plugging into the linear filter-error algorithm derived in Chapter 9. Assume that G is 0; FF^* is invertible; C is square, invertible, and known exactly; and D is known exactly. These are the assumptions that mean we have perfect measurements of the state of the system.

The Kalman filter for this case is (repeating Equation (7.3-11))

$$\hat{x}(t_i) = C^{-1}[z(t_i) - Du(t_i)] = x(t_i) \tag{10.1-13}$$

and the covariance, P_i, of this filtered estimate is 0. The one-step-ahead prediction is

$$\tilde{x}(t_{i+1}) = \phi x(t_i) + \gamma u(t_i) \tag{10.1-14}$$

with

$$Q_i = FF^* \tag{10.1-15}$$

From Equation (9.1-6) we have

$$R_i = CFF^*C^* \tag{10.1-16}$$

and thus Equation (9.1-12) becomes

$$J(\xi) = \frac{1}{2}\sum_{i=1}^{N}[z(t_i) - \tilde{z}_\xi(t_i)]^*(CFF^*C^*)^{-1}[z(t_i) - \tilde{z}_\xi(t_i)] + \frac{1}{2}N\,\ln|CFF^*C^*| \tag{10.1-17}$$

Eliminating irrelevant $|C|$ constants, we can redefine the cost function as

$$J(\xi) = \frac{1}{2}\sum_{i=1}^{N}[x(t_i) - \tilde{x}_\xi(t_i)]^*(FF^*)^{-1}[x(t_i) - \tilde{x}_\xi(t_i)] + \frac{1}{2}N\,\ln|FF^*| \tag{10.1-18}$$

which is in the form of Equation (10.1-11). Note that C and D play no role in this estimator, outside of the reconstruction of the state using Equation (10.1-13).

10.1.3 Discussion

The cost function defined by Equation (10.1-11) or (10.1-12) involves a weighted square sum of the error that would be in the state equation, Equation (10.1-1b), if the noise term were omitted. The term "equation error" derives from this fact. This terminology is rather vague, giving little hint as to what equation is meant. The output-error and filter-error methods described in previous chapters could, with equal validity, be categorized as methods involving minimizing the error of some equation. In spite of this potential ambiguity, the use of the term "equation error" is well-established, and the term is unlikely to be misinterpreted. The terms "state-equation error" and "observation-equation error," which we use in the following sections, are more definitive, but not widely used.

The equation-error method is also referred to by several other names. The term "least squares" is sometimes used to define the method, but this terminology is subject to misinterpretation. The large majority of the estimation methods used can be classified as least-squares methods. We suggest using the term "least squares" only to refer to this broad class of methods (as in the statement "equation error is a least squares method"), never to precisely specify a method. The term "linear least squares" is somewhat more definitive (at least for the case in which f is a linear function of ξ) and has been used on occasion. Another term often used is "regression" method (or, more definitively, "linear regression").

The terms "equation error" or "least squares" are often used to contrast this method with maximum-likelihood estimators. Such contrasts are inappropriate and misleading because equation error is a completely rigorous maximum-likelihood estimator for the problem as stated. The differences between equation error, output error, and filter error lie in the problem statements and assumptions, not in the statistical principles used nor in the rigor of the derivation. To disparage equation error on the basis that it is not maximum likelihood because it ignores measurement noise smacks more of snobbery than of honest evaluation. The neglect of measurement noise may, indeed, be a significant flaw for some applications, but this flaw is irrelevant to the issue of whether equation error is maximum likelihood.

A related common misconception is that equation-error estimates are biased, whereas output-error or filter-error estimates are asymptotically unbiased. To the contrary, equation error is asymptotically unbiased for the problem as stated; in many applications, the equation-error estimates are even unbiased for finite time. It is true that equation error is biased in the presence of measurement noise, but output error is likewise biased in the presence of process noise.

The principle illustrated here is universal: any estimator is biased (among other problems) when applied to systems that violate the assumptions used in deriving the estimator. This principle applies to all assumptions, not just to the presence or absence of noise. Because any real system will violate any tractable set of assumptions, all estimators are actually biased. (All of our previous statements that given estimators are unbiased are based on idealized systems meeting the stated assumptions.)

The unqualified statement that a given estimator is biased is, therefore, of little use in evaluating the estimator. More pertinent issues include the questions of which assumptions are most severely violated by the actual system, and how sensitive the estimator is to these violations. The magnitude of the bias is a reasonable means of addressing these questions, but the mere existence of a bias is not.

10.2 GENERAL EQUATION ERROR FORM

Many practical applications of equation error do not fit naturally into the restrictive definition of the previous section, which allows no measurement noise. There are several alternate definitions of equation error that accommodate these applications. These alternate definitions involve apparently disparate statistical assumptions. The unifying theme, which justifies the use of the same terminology and computational tools for these various cases, is the form of the resulting cost function. In some cases, two different viewpoints and corresponding different assumptions about the same application can result in identical computations.

We will, therefore, take the cost-function form as the general defining property of equation-error estimators. This form can arise from several different sets of statistical assumptions. This nonstatistical, result-oriented approach to the definition helps us to avoid unnaturally contorting some problem statements to force them to fit an overly rigid definition, when a more natural problem statemen achieves the same result.

To define a general equation-error estimator, we start with some equation, expressed as a function of the measurements and the unknown parameters, which should ideally (ignoring noise and modeling errors) be satisfied at every measurement time point. We write the equation in the general form

$$h[z(.),u(.),t_i,\xi] = 0 \qquad i = 1,2,\ldots,N \tag{10.2-1}$$

Sections 10.2.1 through 10.2.3 give specific common cases of such equations.

The equation-error estimate based on this equation is then the value of ξ that minimizes the cost function

$$J(\xi) = \frac{1}{2}\sum_{i=1}^{N} h[z(.),u(.),t_i,\xi]^* \cdot h[z(.),u(.),t_i,\xi] \tag{10.2-2}$$

where W is a positive semidefinite weighting matrix. The definition assumes that the minimum exists and is unique.

In order to accommodate prior information and unknown W matrices, we allow the form of Equation (10.2-2) to be extended to

$$J(\xi) = \frac{1}{2}\sum_{i=1}^{N} h[z(.),u(.),t_i,\xi]^* W\, h[z(.),u(.),t_i,\xi] + \frac{1}{2} N \ln|W^{-1}| + \frac{1}{2}(\xi - m_\xi)^* P(\xi - m_\xi) \tag{10.2-3}$$

corresponding to Equation (10.1-12). The above definition is broad enough to include output error and filter error, as well as the equation-error estimators defined in Section 10.1.

The estimators emphasized in this chapter have the particular property that the h dependence on $z(.)$ and $u(\ )$ is restricted to one or two time points. The central statistical assumption that gives this property is that there are perfect (no noise) measurements of the state. This assumption reduces the Kalman filter to the form of Equation (10.1-3), which eliminates the integration of the state equation. With this assumption, Equation (10.1-3) is the obvious optimal filter even for nonlinear state equations. We are also forced to assume that the process noise covariance FF^* is nonsingular; a singular FF^* combined with the perfect state measurements would give an ill-posed problem.

10.2.1 Discrete State-Equation Error

One specific case of the equation-error method is state-equation error. In this case, the specific form of Equation (10.2-1) derives from the state equation, ignoring the process noise. We will first consider state-equation error for discrete-time systems. The discrete-time state equation for a general nonlinear system, ignoring the process noise, is

$$x(t_{i+1}) = f[x(t_i),u(t_i),\xi] \qquad i = 0,1,\ldots,N-1 \tag{10.2-4}$$

The h function based on this equation is

$$h[z(.),u(.),t_i,\xi] = x(t_i) - f[x(t_{i-1}),u(t_{i-1}),\xi] \qquad i = 1,2,\ldots N \tag{10.2-5}$$

This form presumes that the $x(t_i)$ can be reconstructed as a function of the $z(t_i)$ and $u(t_i)$.

We recognize discrete-time state-equation error as the method derived in Section 10.1. Equation (10.1-12) (with Equation (10.1-10)) is a special case of Equation (10.2-3) using Equation (10.2-5) for h and FF^* for W. Section 10.1 discussed the details of the statistical assumptions implicit in this form.

Note also that we can define a state-equation error method whether or not the state measurements are noise-free. The only requisite for a plausible state-equation error method is that we have some estimate of the state to use in Equation (10.2-5). If the measurements are contaminated with noise, then the estimator is not a maximum-likelihood estimator and will be asymptotically biased. There are many practical circumstances, however, where a simple equation-error estimator is preferable to the "optimal" alternatives.

10.2.2 Continuous/Discrete State-Equation Error

For a mixed continuous/discrete-time system with additive process noise, the state equation is

$$\dot{x}(t) = f[x(t),u(t),\xi] + Fn(t) \tag{10.2-6}$$

The h function for a continuous/discrete-time state-equation error method derives from evaluating the state equation at the measurement times t_i and ignoring the process noise:

$$h[z(.),u(.),t_i,\xi] = \dot{x}(t_i) - f[x(t_i),u(t_i),\xi] \tag{10.2-7}$$

The use of this form in an equation-error method presumes that the state $x(t_i)$ can be reconstructed as a function of the $z(t_i)$ and $u(t_i)$. This presumption is identical to that for discrete-time state-equation error, and it implies the same conditions: there must be noise-free measurements of the state, independent of ξ. It is implicit that a known invertible transformation of such measurements is statistically equivalent. As in the discrete-time case, we can define the estimator even when the measurements are noisy, but it will no longer be a maximum-likelihood estimator.

Equation (10.2-7) also presumes that the derivative $\dot{x}(t_i)$ can be reconstructed from the measurements. Neglecting for the moment the statistical implications, note that we can form a plausible equation-error estimator using any reasonable means of approximating a value for $\dot{x}(t_i)$ independently of ξ. The simplest case of this is when the observation vector includes measurements of the state derivatives in addition to the measurements of the states. If such derivative measurements are not directly available, we can always approximate $\dot{x}(t_i)$ by finite-difference differentiation of the state measurements, as in

$$\dot{x}(t_i) = \frac{x(t_{i+1}) - x(t_{i-1})}{t_{i+1} - t_{i-1}} \tag{10.2-8}$$

Both direct measurement and finite-difference approximation are used in practice.

Rigorous statistical treatment is easiest for the case of finite-difference approximations. To arrive at such a form, we write the state equation in integrated form as

$$x(t_{i+1}) = x(t_i) + \int_{t_i}^{t_{i+1}} f[x(t),u(t),\xi]dt + \int_{t_i}^{t_{i+1}} Fn(t)dt \tag{10.2-9}$$

An approximate solution (not necessarily the best approximation) to Equation (10.2-9) is

$$x(t_{i+1}) = x(t_i) + (t_{i+1} - t_i)f[x(t_i),u(t_i),\xi] + F_d n_i \tag{10.2-10}$$

where n_i is a sequence of independent Gaussian variables, and F_d is the equivalent discrete F-matrix. Sections 6.2 and 7.5 discuss such approximations.

Equation (10.2-10) is in the form of a discrete-time state equation. The discrete-time state-equation error method based on this equation uses

$$h[z(.),u(.),t_i,\xi] = x(t_i) - x(t_{i-1}) - (t_i - t_{i-1})f[x(t_{i-1}),u(t_{i-1}),\xi] \tag{10.2-11}$$

Redefining h by dividing by $t_i - t_{i-1}$ gives the form

$$h[z(.),u(.),t_i,\xi] = \dot{x}(t_i) - f[x(t_i),u(t_i),\xi] \tag{10.2-12}$$

where the derivative is obtained from the finite-difference formula

$$\dot{x}(t_i) = \frac{x(t_i) - x(t_{i-1})}{t_i - t_{i-1}} \tag{10.2-13}$$

Other discrete-time approximations of Equation (10.2-9) result in different finite-difference formulae. The central-difference form of Equation (10.2-8) is usually better than the one-sided form of Equation (10.2-13), although Equation (10.2-8) has a lower bandwidth. If the bandwidth of Equation (10.2-8) presents problems, a better approach than Equation (10.2-13) is to use

$$h[z(.),u(.),t_i,\xi] = \dot{x}(t_{i-1/2}) - f[x(t_{i-1/2}),u(t_{i-1/2}),\xi] \tag{10.2-14}$$

where we have used the notation

$$t_{i-1/2} = \frac{1}{2}(t_i + t_{i-1}) \tag{10.2-15}$$

and

$$\dot{x}(t_{i-1/2}) = \frac{x(t_i) - x(t_{i-1})}{t_i - t_{i-1}} \tag{10.2-16}$$

There are several other reasonable finite-difference formulae applicable to this problem.

Rigorous statistical treatment of the case in which direct state derivative measurements are available raises several complications. Furthermore, it is difficult to get a rigorous result in the form typically used—an equation-error method based on $\dot{x}$ measurements substituted into Equation (10.2-7). It is probably best to regard this approach as an equation-error estimator derived from plausible, but ad hoc, reasoning.

We will briefly outline the statistical issues raised by state derivative measurements, without attempting a complete analysis. The first problem is that, for systems with white process noise, the state derivative is infinite at every point in time. (Careful argument is required even to define the derivative.) We could avoid this problem by requiring the process noise to be band-limited, or by other means, but the resulting estimator

will not be in the desired form. A heuristic explanation is that the x measurements contain implicit information about the derivative (from the finite differences), and simple use of the measured derivative ignores this information. A rigorous maximum-likelihood estimator would use both sources of information. This statement assumes that the $\dot{x}$ measurements and the finite-difference derivatives are independent data. It is conceivable that the x "measurements" are obtained as sums of the $\dot{x}$ measurements (for instance, in an inertial navigation unit). Such cases are merely integrated versions of the finite-difference approach, not really comparable to cases of independent $\dot{x}$ measurements.

The lack of a rigorous derivation for the state-equation error method with independently measured state derivatives does not necessarily mean that it is a poor estimator. If the information in the state derivative measurements is much better than the information in the finite-difference state derivatives, we can justify the approach as a good approximation. Furthermore, as expressed in our discussions in Section 1.4, an estimator does not have to be statistically derived to be a good estimator. For some problems, this estimator gives adequate results with low computational costs; when this result occurs, it is sufficient justification in itself.

10.2.3 Observation-Equation Error

Another specific case of the equation-error method is observation-equation error. In this case, the specific form of h comes from the observation equation, ignoring the noise. The equation is the same for pure discrete-time or mixed continuous/discrete-time systems. The observation equation for a system with additive noise is

$$z(t_i) = g[x(t_i),u(t_i),\xi] + Gn_i \tag{10.2-17}$$

The h function based on this equation is

$$h[z(.),u(.),t_i,\xi] = z(t_i) - g[x(t_i),u(t_i),\xi] \tag{10.2-18}$$

As in the case of state-equation error, observation-equation error requires measurements or reconstructions of the state, because $x(t_i)$ appears in the equation. The comments in Section 10.2.1 about noise in the state measurement apply here also. Observation-equation error does not require measurements of the state derivative.

The observation-equation error method also requires that there be some measurements in addition to the states, or the method reduces to triviality. If the states were the only measurements, the observation equation would reduce to

$$z(t_i) = x(t_i) \tag{10.2-19}$$

which has no unknown parameters. There would, therefore, be nothing to estimate.

The observation-equation error method applies only to estimating parameters in the observation equation. Unknown parameters in the state equation do not enter this formulation. In fact, the existence of the state equation is largely irrelevant to the method.

This irrelevance perhaps explains why observation-equation error is usually neglected in discussions of estimators for dynamic systems. The method is essentially a direct application of the static estimators of Chapter 5, taking no advantage of the dynamics of the system (the state equation). From a theoretical viewpoint, it may seem out of place in this chapter.

In practice, the observation-equation-error method is widely used, sometimes contorted to look like a state-equation-error method. The observation-equation-error method is often a competitor to an output-error method. Our treatment of observation-equation error is intended to facilitate a fair evaluation of such choices and to avoid unnecessary contortions into state-equation error forms.

10.3 COMPUTATION

We have previously mentioned that a unifying characteristic of the methods discussed in this chapter is their computational simplicity. We have not, however, given much detail on the computational issues.

Equation (10.2-3), which encompasses all equation-error forms, is in the form of Equation (2.5-1) if the weighting matrix W is known. Therefore, the Gauss-Newton optimization algorithm applies directly. Unknown W matrices can be handled by the method discussed in Sections 5.5 and 8.4.

In the most general definition of equation error, this is nearly the limit of what we can state about computation. The definition of Equation (10.2-3) is general enough to allow output error and filter error as special cases. Both output error and filter error have the special property that the dependence of h on z and u can be cast in a recursive form, significantly lowering the computational costs. Because of this recursive form, the total computational cost is roughly proportional to the number of time points, N. The general definition of equation error also encompasses nonrecursive forms, which could have computational costs proportional to N^2 or higher powers.

The equation-error methods discussed in this chapter have the property that, for each t_i, the dependence of h on $z(.)$ and $u(.)$ is restricted to one or two time points. Therefore, the computational effort for each evaluation of h is independent of N, and the total computational cost is roughly proportional to N. In this regard, state-equation error and output-equation error are comparable to output error and filter error. For a completely general, nonlinear system, the computational cost of state-equation error or output-equation

error is roughly similar to the cost of output error. (General nonlinear models are currently impractical for filter error without using linearized approximations.)

In the large majority of practical applications, however, the f and g functions have special properties which make the computational costs of state-equation error and output-equation error far smaller than the computational costs of output error or filter error.

The first property is that the f and g functions are linear in ξ. This property holds true even for systems described as nonlinear; the nonlinearity meant by the term "nonlinear system" is as a function of x and u—not as a function of ξ. Equation (1.3-2) is a simple example of a static system nonlinear in the input, but linear in the parameters. The output-error method can seldom take advantage of linearity in the parameters, even when the system is also linear in x and u, because the system response is usually a nonlinear function of ξ. (There are some significant exceptions in special cases.)

State-equation error and output-equation error methods, in contrast, can take excellent advantage of linearity in the parameters, even when the system is nonlinear in x and u. In this situation, state-equation error and output-equation error meet the conditions of Section 2.5.1 for the Gauss-Newton algorithm to attain the exact minimum in a single iteration.

This is both a quantitative and a qualitative computational improvement relative to output error. The quantitative improvement is a division of the computational cost by the number of iterations required for the output-error method. The qualitative improvement is the elimination of the issues associated with iterative methods: starting values, convergence-testing criteria, failure to converge, convergence accelerators, multiple local solutions, and other issues. The most commonly cited of these benefits is that there is no need for reasonable starting values. You can evaluate the equations at any arbitrary point (zero is often convenient) without affecting the result.

Another simplifying property of f and g, not quite as universal, but true in the majority of cases, is that each element of ξ affects only one element of f or g. The simplest example of this is a linear system where the unknown parameters are individual elements of the system matrices. With this structure, if we constrain W to be diagonal, Equation (10.2-3) separates into a sum of independent minimization problems with scalar h, one problem for each element of h. If ℓ is the number of elements of the h-vector, we now have ℓ independent functions in the form of Equation (10.2-3), each with scalar h. Each element of ξ affects one and only one of these scalar functions.

This partitioning has the obvious benefit, common to most partitioning algorithms, that the sum of the ℓ-problems with scalar h requires less computation than the unpartitioned vector problem. The outer-product computation of Equation (2.5-11), often the most time-consuming part of the algorithm, is proportional to the square of the number of unknowns and to ℓ. Therefore, if the unknowns are evenly distributed among the ℓ elements of h, the computational cost of the vector problem could be as much as ℓ^3 times the cost of each of the scalar problems. Other portions of the computational cost and overhead will reduce this factor somewhat, but the improvement is still dramatic.

Another benefit of the partitioning is that it allows us to avoid iteration when the noise covariances are unknown. With this partitioning, the minimizing values of ξ are independent of W. The normal role of W is in weighing the importance of fitting the different elements of the h. One value of ξ might fit one element of h best, while another value of ξ fits another element of h best; W establishes how to strike a compromise among these conflicting aims. Since the partitioned problem structure makes the different elements of h independent, W is largely irrelevant. Therefore we can estimate the elements of ξ using any arbitrary value of W (usually an identity matrix). If we want an estimate of W, we can compute it after we estimate the other unknowns.

The combined effect of these computational improvements is to make the computational cost of the state-equation error and output-equation error methods negligible in many applications. It is common for the computational cost of the actual equation-error algorithm to be dwarfed by the overhead costs of obtaining the data, plotting the results, and related computations.

10.4 DISCUSSION

The undebated strong points of the state-equation-error and output-equation-error methods are their simplicity and low computational cost. Most important is that Gauss-Newton gives the exact minimum of the cost function without iteration. Because the methods are noniterative, they require no starting estimates. These methods have been used in many applications, sometimes under different names.

The weaknesses of these methods stem from their assumptions of perfect state measurements. Relatively small amounts of noise in the measurements can cause significant bias errors in the estimates. If a measurement of some state is unavailable, or if an instrument fails, these methods are not directly applicable (though such problems are sometimes handled by state reconstruction algorithms).

State-equation-error and output-equation-error methods can be used with either of two distinct approaches, depending upon the application. The first approach is to accept the problem of measurement-noise sensitivity and to emphasize the computational efficiency of the method. This approach is appropriate when computational cost is a more important consideration than accuracy.

For example, state-equation error and output-equation error methods are popular for obtaining starting values for iterative procedures such as output error. In such applications, the estimates need only be accurate enough to cause the iterative methods to converge (presumably to better estimates).

Another common use for state-equation error and output-error is to select a model from a large number of candidates by estimating the parameters in each candidate model. Once the model form is selected, the rough parameter estimates can be refined by some other method.

The second approach to using state-equation-error or output-equation-error methods is to spend the time and effort necessary to get accurate results from them, which first requires accurate state measurements with low noise levels. In many applications of these methods, most of the work lies in filtering the data and reconstructing estimates of unmeasured states. (A Kalman filter can sometimes be helpful here, provided that the filter does not depend upon the parameters to be estimated. This condition requires a special problem structure.) The total cost of obtaining good estimates from these methods, including the cost of data preprocessing, may be comparable to the cost of more complicated iterative algorithms that require less preprocessing. The trade-off is highly dependent on application variables such as the required accuracy of the estimates, the quality of the available instrumentation, and the existence of independent needs for accurate state measurements.

CHAPTER 11

11.0 ACCURACY OF THE ESTIMATES

Parameter estimates from real systems are, by their nature, imperfect. The accuracy of the estimates is a pervasive issue in the various stages of application, from the problem statement to the evaluation and use of the results.

We introduced the subject of parameter estimation in Section 1.4, using concepts of errors in the estimates and adequacy of the results. The subsequent chapters have largely concentrated on the derivation of algorithms. These derivations are all related to accuracy issues, based on the definitions and discussions in Chapter 4. However, the questions about accuracy have been largely overshadowed by the details of deriving and implementing the algorithms.

In this chapter, we return the emphasis to the critical issue of accuracy. The final judgment of the parameter estimation process for a particular application is based on the accuracy of the results. We examine the evaluation of the accuracy, factors contributing to inaccuracy, and means of improving accuracy. A truly comprehensive treatment of the subject of accuracy is impossible. We restrict our discussion largely to generic issues related to the theory and methodology of parameter estimation.

To make effective use of parameter estimates, we must have some gauge of their accuracy, be it a statistical measure, an intuitive guess, or some other source. If we absolutely cannot distinguish the extremes of accurate versus worthless estimates, we must always consider the possibility that the estimates are worthless, in which case the estimates could not be used in any application in which their validity was important. Therefore, measures of the estimate accuracy are as important as are the estimates themselves. Various means of judging the accuracy of parameter estimates are in current use.

We will group the uses for measures of estimate accuracy into three general classes. The first class of use is in planning the parameter estimation. Predictions of the estimate accuracy can be used to evaluate the adequacy of the proposed experiments and instrumentation system for the parameter estimation on the proposed model. There are limitations to this usage because it involves predicting accuracy before the actual data are obtained. Unexpected problems can always cause degradation of the results compared to the predictions. The accuracy predictions are most useful in identifying experiments that have no hope of success.

The second use is in the parameter estimation process itself. Measures of accuracy can help detect various problems in the estimation, from modeling failures, data problems, program bugs, or other sources. Another facet of this class of use is the comparison of different estimates. The comparisons can be between two different models or methods applied to the same data set, between estimates from independent data sets, or between predictions and estimates from the experimental data. In any of these events, measures of accuracy can help determine which of the conflicting values is best, or whether some compromise between them should be considered. Comparison of the accuracy measures with the differences in the estimates is a means to determine if the differences are significant. The magnitude of the observed differences between the estimates is, in itself, an indicator of accuracy.

The third use of measures of accuracy is for presentation with the final estimates for the user of the results. If the estimates are to be used in a control system design, for instance, knowledge of their accuracy is useful in evaluating the sensitivity of the control system. If the estimates are to be used by an explicit adaptive or learning control system, then it is important that the accuracy evaluation be systematic enough to be automatically implemented. Such immediate use of the estimates precludes the intercession of engineering judgment; the evaluation of the estimates must be entirely automatic. Such control systems must recognize poor results and suitably discount them (or ensure that they never occur—an overly optimistic goal).

The single most critical contributor to getting accurate parameter estimates in practical problems is the analyst's understanding of the physical system and the instrumentation. The most thorough knowledge of parameter estimation theory and the use of the most powerful techniques do not compensate for poor understanding of the system. This statement relates directly to the discussion in Chapter 1 about the "black box" identification problem and the roles of independent knowledge versus system identification. The principles discussed in this chapter, although no substitute for an understanding of the system, are a necessary adjunct to such understanding.

Before proceeding further, we need to review the definition of the term "accuracy" as it applies to real data. A system is never described exactly by the simplified models used for analysis. Regardless of the sophistication of the model, unexplained sources of modeling error will always remain. There is no unique, correct model.

The concept of accuracy is difficult to define precisely if no correct model exists. It is easiest to approach by considering the problem in two parts: estimation and modeling. For analyzing the estimation problem, we assume that the model describes the system exactly. The definition of accuracy is then precise and quantitative. Many results are available in the subject area of estimation accuracy. Sections 11.1 and 11.2 discuss several of them.

The modeling problem addresses the question of whether the form of the model can describe the system adequately for its intended use. There is little guide from the theory in this area. Studies such as those of Gupta, Hall, and Trankle (1978), Fiske and Price (1977), and Akaike (1974), discuss selection of the best model from a set of candidates, but do not consider the more basic issue of defining the candidate models. Section 11.4 considers this point in more detail.

For the most part, the determination of model adequacy is based on engineering judgment and problem-specific analysis relying heavily on the analyst's understanding of the physics of the system. In some cases,

we can test model adequacy by demonstration: if we try the model and it achieves its purpose, it was obviously adequate. Such tests are not always practical, however. This method assumes, of course, that the test was comprehensive. Such assumptions should not be made lightly; they have cost lives when systems encountered untested conditions.

After considering estimation and modeling as separate problems, we need to look at their interactions to complete the discussion of accuracy. We need to consider the estimates that result from a model judged to be adequate, although not exact. As in the modeling problem, this process involves considerable subjective judgment, although we can obtain some quantitative results.

We can examine some specific, postulated sources of modeling error through simulations or analyses that use more complex models than are practical or desirable in the parameter estimation. Such simulations or analyses can include, for example, models of specific, postulated instrumentation errors (Hodge and Bryant, 1978; and Sorensen, 1972). Maine and Iliff (1981b) present some more general, but less rigorous, results.

11.1 CONFIDENCE REGIONS

The concept of a confidence region is central to the analytical study of estimation accuracy. In general terms, a confidence region is a region within which we can be reasonably confident that the true value of ξ lies. Accurate estimates correspond to small confidence regions for a given level of confidence. Note that small confidence regions imply large confidence; in order to avoid this apparent inversion of terminology, the term "uncertainty region" is sometimes used in place of the term "confidence region." The following subsections define confidence regions more precisely.

For continuous, nonsingular estimation problems, the probability of any point estimate's being exactly correct is zero. We need a concept such as the confidence region to make statements with a nonzero confidence. Throughout the discussion of confidence regions, we assume that the system model is correct; that is, we assume that ξ has a true value lying in the parameter space. In later sections we will consider issues relating to modeling error.

11.1.1 Random Parameter Vector

Let us consider first the case in which ξ is a random variable with a known prior distribution. This situation usually implies the use of an MAP estimator.

In this case, ξ has a posterior distribution, and we can define the posterior probability that ξ lies in any fixed region. Although we will use the posterior distribution of ξ as the context for this discussion, we can equally well define prior confidence regions. None of the following development depends upon our working with a posterior distribution. For simplicity of exposition, we will assume that the posterior distribution of ξ has a density function. The posterior probability that ξ lies in a region R is then

$$P(R) = \int_R p(\xi|Z)d|\xi| \tag{11.1-1}$$

We define R to be a confidence region for the confidence level α if $P(R) = \alpha$, and no other region with the same probability is smaller than R. We use the volume of a region as a measure of its size.

> Theorem 11.1 Let R be the set of all points with $p(\xi|Z) \geq c$, where c is a constant. Then R is a confidence region for the confidence level $\alpha = P(R)$.
>
> Proof Let R be as defined above, and let R' be any other region with $P(R') = \alpha$. We need to prove that the volume of R' must be greater than or equal to that of R. We define $T = R \cap R'$, $S = R \cap \bar{R}'$, and $S' = R' \cap \bar{R}$. Then T, S, and S' are disjoint, $R = T \cup S$, and $R' = T \cup S'$. Because $S \subset R$, we must have $p(\xi|Z) \geq c$ everywhere in S. Conversely, $S' \subset \bar{R}$, so $p(\xi|Z) < c$ everywhere in S'. In order for $P(R') = P(R)$, we must have $P(S') = P(S)$. Therefore, the volume of S' must be greater than or equal to that of S. The volume of R' must then be greater than that of R, completing the proof.

It is often convenient to characterize a closed region by its boundary. The boundaries of the confidence regions defined by Theorem 11.1 are isoclines of the posterior density function $p(\xi|Z)$.

We can write the confidence region derived in the above theorem as

$$R = \{x: p_{\xi|Z}(x|Z) \geq c\} \tag{11.1-2}$$

We must use the full notation for the probability density function to avoid confusion in the following manipulations. For consistency with the following section, it is convenient to re-express the confidence region in terms of the density function of the error.

$$e = \xi - \hat{\xi} \tag{11.1-3}$$

The estimate $\hat{\xi}$ is a deterministic function of Z; therefore, Equation (11.1-3) trivially gives

$$p_{\xi|Z}(x|Z) = p_{e|Z}(x - \hat{\xi}|Z) \tag{11.1-4}$$

Substituting this into Equation (11.1-2) gives the expression

$$R = \{x: p_{e|Z}(x - \hat{\xi}|Z) \geq c\} \tag{11.1-5}$$

Substituting $x + \hat{\xi}$ for x in Equation (11.1-5) gives the convenient form

$$R = \{\hat{\xi} + x: p_{e|Z}(x|Z) \geq c] \tag{11.1-6}$$

This form shows the boundaries of the confidence regions to be translated isoclines of the error-density function.

Exact determination of the confidence regions is impractical except in simple cases. One such case occurs when ξ is scalar and $p(\xi|Z)$ is unimodal. An isocline then consists of two points, and the line segment between the two points is the confidence region. In this one-dimensional case, the confidence region is often called a confidence interval.

Another simple case occurs when the posterior density function is in some standard family of density functions expressible in closed form. This is most commonly the family of Gaussian density functions. An isocline of a Gaussian density function with mean m and nonsingular covariance Λ is a set of x values satisfying

$$(x - m)^*\Lambda^{-1}(x - m) = c \tag{11.1-7}$$

This is the equation of an ellipsoid.

For problems not fitting into one of these special cases, we usually must make approximations in the computation of the confidence regions. Section 11.1.3 discusses the most common approximation.

11.1.2 Nonrandom Parameter Vector

When ξ is simply an unknown parameter with no random nature, the development of confidence regions is more oblique, but the result is similar in form to the results of the previous section. The same comments apply when we wish to ignore any prior distribution of ξ and to obtain confidence regions based solely on the current experimental data. These situations usually imply the use of MLE estimators.

In neither of these situations can we meaningfully discuss the probability of ξ lying in a given region. We proceed as follows to develop a substitute concept: the estimate $\hat{\xi}$ is a function of the observation Z, which has a probability distribution conditioned on ξ. Therefore, we can define a probability distribution of $\hat{\xi}$ conditioned on ξ. We will assume that this distribution has a density function $p_{\hat{\xi}|\xi}$.

For a given value of ξ, the isoclines of $p_{\hat{\xi}|\xi}$ define boundaries of confidence regions for $\hat{\xi}$. Let R_1 be such a confidence region, with confidence level α.

$$R_1 = \{x: p_{\hat{\xi}|\xi}(x|\xi) \geq c\} \tag{11.1-8}$$

It is convenient to define R_1 in terms of the error density function $p_{e|\xi}$, using the relation

$$p_{\hat{\xi}|\xi}(x|\xi) = p_{e|\xi}(\xi - x|\xi) \tag{11.1-9}$$

This gives

$$R_1 = \{\xi - x: p_{e|\xi}(x|\xi) > c\} \tag{11.1-10}$$

The estimate $\hat{\xi}$ has probability α of being in R_1. For this chapter, we are more interested in the situation where we know the value of $\hat{\xi}$ and seek to define a confidence region for ξ, which is unknown. We can define such a confidence region for ξ, given $\hat{\xi}$, in two steps, starting with the region R_1.

The first step is to define a region R_2 which is a mirror image of R_1. A point $\xi - x$ in the region R_1 reflects onto the point $\hat{\xi} + x$ in R_2, as shown in Figure (11.1-1). We can thus write R_2 as

$$R_2 = \{\hat{\xi} + x: p_{e|\xi}(x|\xi) \geq c\} \tag{11.1-11}$$

This reflection interchanges ξ and $\hat{\xi}$; therefore, ξ is in R_2 if and only if $\hat{\xi}$ is in R_1. Because there is probability α that $\hat{\xi}$ lies in R_1, there is the same probability α that $\hat{\xi}$ lies in R_2.

To be technically correct, we must be careful about the phrasing of this statement. Because the true value ξ is not random, it makes no sense to say that ξ has probability α of lying in R_2. The randomness is in the construction of the region R_2 because R_2 depends on the estimate $\hat{\xi}$, which depends in turn on the noise-contaminated observations. We can sensibly say that the region R_2, constructed in this manner, has probability α of covering the true value ξ. This concept of a region covering the fixed point ξ replaces the concept of the point ξ lying in a fixed region. The distinction is more important in theory than in practice.

Although we have defined the region R_2 in principle, we cannot construct the region from the data available because R_2 deperds on the value of ξ, which is unknown. Our next step is to construct a region R_3, which approximates R_2, but does not depend on the true value of ξ. We base the approximation on the assumption that $p_{e|\xi}$ is approximately invariant as a function of ξ; that is

$$p_{e|\xi}(x|\xi) \approx p_{e|\xi}(x|\xi + \delta) \tag{11.1-12}$$

This approximation is unlikely to be valid for large values of δ except in simple cases. For small values of δ, the approximation is usually reasonable.

We define the confidence region R_3 by applying this approximation to Equation (11.1-11), using $\hat{\xi} - \xi$ for δ.

$$R_3 = \{\hat{\xi} + x: p_{e|\xi}(x|\hat{\xi}) \geq c\} \tag{11.1-13}$$

The region R_3 depends only on $\hat{\xi}$, $p_{e|\xi}$, and the arbitrary constant c. The function $p_{e|\xi}$ is presumed known from the start, and $\hat{\xi}$ is the estimate computed by the methods described in previous chapters. In principle, we have sufficient information to compute the region R_3. Practical application requires either that $p_{e|\xi}$ be in one of the simple forms described in Section 11.1.1, or that we make further approximations as discussed in Section 11.1.3.

If $\hat{\xi} - \xi$ is small (that is, if the estimate is accurate), then R_3 will likely be a close approximation to R_2. If $\hat{\xi} - \xi$ is large, then the approximation is questionable. The result is that we are unable to define large confidence regions accurately except in special cases. We can tell that the confidence region is large, but its precise size and shape are difficult to determine.

Note that the confidence region for nonrandom parameters, defined by Equation (11.1-13), is almost identical in form to the confidence region for random parameters, defined by Equation (11.1-6). The only difference in the form is what the density functions are conditioned on.

11.1.3 Gaussian Approximation

The previous sections have derived the boundaries of confidence regions for both random and nonrandom parameter vectors in terms of isoclines of probability density functions of the error vector. Except in special cases, the probability density functions are too complicated to allow practical computation of the exact isoclines. Extreme precision in the computation of the confidence regions is seldom necessary; we have already made approximations in the definition of confidence regions for nonrandom parameters In this section, we introduce approximations which allow relatively easy computation of confidence regions.

The central idea of this section is to approximate the pertinent probability density functions by Gaussian density functions. As discussed in Section 11.1.1, the isoclines of Gaussian density functions are ellipsoids, which are easy to compute. We call these "confidence ellipsoids" or "uncertainty ellipsoids." In many cases, we can justify the Gaussian approximation with arguments that the distributions asymptotically approach Gaussians as the amount of data increases. Section 5.4.2 discusses some pertinent asymptotic results.

A Gaussian approximation is defined by its mean and covariance. We will consider appropriate choices for the mean and covariance to make the Gaussian density function a reasonable approximation. An obvious possibility is to set the mean and covariance of the Gaussian approximation to match the mean and covariance of the original density function; we are often forced to settle for approximations to the mean and covariance of the original density function, the exact values being impractical to compute. Another possibility is to use Equations (3.5-17) and (3.5-18). We will illustrate the use of both of these options.

Consider first the case of an MLE estimator. Equation (11.1-13) defines the confidence region. We will use covariance matching to define the Gaussian approximation to $p_{e|\xi}$. The exact mean and covariance of $p_{e|\xi}$ are difficult to compute, but there are asymptotic results which give reasonable approximations.

We use zero as an approximation to the mean of $p_{e|\xi}$; this approximation is based on MLE estimators being asymptotically unbiased. Because MLE estimators are efficient, the Cramer-Rao bound gives an asymptotic approximation for the covariance of $p_{e|\xi}$ as the inverse of the Fisher information matrix $M(\xi)$. We can use either Equation (4.2-19) or (4.2-24) as equivalent expressions for the Fisher information matrix. Equation (5.4-11) gives the particular form of $M(\xi)$ for static nonlinear systems with additive Gaussian noise.

Both $\hat{\xi}$ and $M(\hat{\xi})$ are readily available in practical application. The estimate $\hat{\xi}$ is the primary output of a parameter estimation program, and most MLE parameter-estimation programs compute $M(\hat{\xi})$ or an approximation to it as a by-product of iterative minimization of the cost function.

Now consider the case of an MAP estimator. We need a Gaussian approximation to $p(e|z)$. Equations (3.5-17) and (3.5-18) provide a convenient basis for such an approximation. By Equation (3.5-17), we set the mean of the Gaussian approximation equal to the point at which $p(e|z)$ is a maximum; by definition of the MAP estimator, this point is zero.

We then set the covariance of the Gaussian approximation to

$$\Lambda = [-\nabla_e^2 \ln p(e|z)]^{-1} \tag{11.1-14}$$

evaluated at $\xi = \hat{\xi}$. For static nonlinear systems with additive Gaussian noise, Equation (11.1-14) reduces to the form of Equation (5.4-12), which we could also have obtained by approximate covariance matching arguments. This form for the covariance is the same as that used in the MLE confidence ellipsoid, with the addition of the prior covariance term. As the prior covariance goes to infinity, the confidence ellipsoid for the MAP estimator approaches that for the MLE estimator, as we would anticipate.

Both the MLE and MAP confidence ellipsoids take the form

$$(x - \hat{\xi})^* \Lambda^{-1} (x - \hat{\xi}) = c \tag{11.1-15}$$

where Λ is an approximation to the error-covariance matrix. We have suggested suitable approximations in the above paragraphs, but most approximations to the error covariance are equally acceptable. The choice is usually dictated by what is conveniently available in a given program.

11.1.4 Nonstatistical Derivation

We can alternately derive the confidence ellipsoids for MAP and MLE estimators from a nonstatistical viewpoint. This derivation obtains the same result as the statistical approach and is easier to follow. Comparison of the ideas used in the statistical and nonstatistical derivations reveals the close relationships between the statistical characteristics of the estimates and the numerical problems of computing them. The nonstatistical approach generalizes easily to estimators and models for which precise statistical descriptions are difficult.

The nonstatistical derivation presumes that the estimate is defined as the minimizing point of some cost function. We examine the shape of this cost function as it affects the numerical minimization problem in the area of the minimum. For current purposes, we are not concerned with start-up problems, isolated local minima, and other problems manifested far from the solution point. A relatively flat, ill-defined minimum corresponds to a questionable estimate; the extreme case of this is a function without a discrete local minimum point A steep, well-defined minimum corresponds to a reliable estimate.

With this justification, we define a confidence region to be the set of points with cost-function values less than or equal to some constant. Different values of the constant give different confidence levels. The boundary of such a region is an isocline of the cost function.

We then approximate the cost function in the neighborhood of the minimum by a quadratic Taylor-series expansion about the minimum point.

$$J(\xi) \approx J(\hat{\xi}) + \frac{1}{2}(\xi - \hat{\xi})^*[\nabla_\xi^2 J(\hat{\xi})](\xi - \hat{\xi}) \tag{11.1-16}$$

The isoclines of this quadratic approximation are the confidence ellipsoids.

$$(\xi - \hat{\xi})^*[\nabla_\xi^2 J(\hat{\xi})](\xi - \hat{\xi}) = c \tag{11.1-17}$$

The second gradient of an MLE or MAP cost function is an asymptotic approximation to the appropriate error covariance. Therefore, Equation (11.1-17) gives the same shape confidence ellipsoids as we previously derived on a statistical basis. In practice, the Gauss-Newton or other approximation to the second gradient is usually used.

The constant c determines the size of the confidence ellipsoid. The nonstatistical derivation gives no obvious basis for selecting a value of c. The value $c = 1$ gives the most useful correspondence to the statistical derivation, as we will see in Section 11.2.1.

Figures (11.1-2) and (11.1-3) illustrate the construction of one-dimensional confidence ellipsoids using the nonstatistical definition.

11.2 ANALYSIS OF THE CONFIDENCE ELLIPSOID

The confidence ellipsoid gives a comprehensive picture of the theoretically likely errors in the estimate. It is difficult, however, to display the information content of the ellipsoid on a two-dimensional sheet of paper. In the applications we most commonly work on, there are typically 10 to 30 unknown parameters; that is, the ellipsoid is 10- to 30-dimensional. We can print the covariance matrix which defines the shape of the ellipsoid, but it is difficult to draw useful conclusions from such a presentation format. The problem of meaningful presentation is further compounded when analyzing hundreds of experiments to obtain parameter estimates under a wide variety of conditions.

In the following sections, we discuss simplified statistics that characterize important features of the confidence ellipsoids in ways that are easy to describe and present. The emphasis in these statistics is on reducing the dimensionality of the problem. Many important questions about accuracy reduce to one-dimensional forms, such as the accuracy of the estimate of each element of the parameter vector.

All of the statistics discussed here are functions of the matrix Λ, which defines the shape of the confidence ellipsoid. We have seen above that Λ is an approximation to the error-covariance matrix. These two viewpoints of Λ will provide us with geometrical and statistical interpretations. A third interpretation comes from viewing Λ as the inverse of the second gradient of the cost function. In practice, Λ is usually computed from the Gauss-Newton or other convenient approximation to the second gradient.

These statistics are closely linked to some of the basic sources of estimation errors and difficulties. We will illustrate the discussion with idealized examples of these classes of difficulties. The exact means of overcoming such difficulties depends on the problem, but the first step is to understand the mechanism causing the difficulty. In a surprising number of applications, the major difficulties are cases of the simple idealizations discussed here.

11.2.1 Sensitivity

The sensitivity is the simplest of the statistics relating to the confidence ellipsoid. Although the sensitivity has both a statistical and a nonstatistical interpretation, the use of the statistical interpretation is relatively rare. The term "sensitivity" comes from the nonstatistical interpretation, which we will discuss first.

From the nonstatistical viewpoint, the sensitivity is a measure of how much the cost-function value changes for a given change in a scalar parameter value. The most common definition of the sensitivity with respect to a parameter is the second partial derivative of the cost function with respect to the parameter.

$$S_i = \frac{\partial^2 J(\xi)}{\partial \xi_i^2} \tag{11.2-1}$$

For the purposes of this chapter, we are interested in the sensitivity evaluated at the minimum point of the cost function; we will take this as part of the definition of the sensitivity.

The ξ_i in Equation (11.2-1) can be any scalar function of the ξ vector. In most cases, ξ_i is one of the elements of the ξ vector. For simplicity, we will assume for the rest of this section that ξ_i is the ith element of ξ. Generalizations are straightforward. When ξ_i is the ith element of ξ, the second partial derivative with respect to ξ_i is the ith diagonal element of the second-gradient matrix.

$$S_i = [\nabla_\xi^2 J(\xi)]_{ii} = (\Lambda^{-1})_{ii} \tag{11.2-2}$$

The sensitivity has a simple geometric interpretation based on the confidence ellipsoid. Use the value $c = 1$ in Equation (11.1-17) to define a confidence ellipsoid. Draw a line passing through $\hat{\xi}$ (the center of the ellipsoid) and parallel to the ξ_i axis. The sensitivity with respect to ξ_i is related to the distance, I_i, from the center of the ellipsoid to the intercept of this line and the ellipsoid. We call this distance the insensitivity with respect to ξ_i. Figure (11.2-1) shows the construction of the insensitivities with respect to ξ_1 and ξ_2 on a two-dimensional example. The relationship between the sensitivity and the insensitivity is

$$I_i = (S_i)^{-1/2} = \Lambda^{1/2} \tag{11.2-3}$$

which follows immediately from Equation (11.1-17) for the confidence ellipsoid, and Equation (11.2-1) for the sensitivity.

We can rephrase the geometric interpretation of the insensitivity as follows: the insensitivity with respect to ξ_i is the largest change that we can make in the ith element of $\hat{\xi}$ and still remain within the confidence ellipsoid. All other elements of ξ are constrained to remain equal to their estimates values during this search; that is, the search is constrained to a line parallel to the ξ_i axis passing through $\hat{\xi}$.

From the statistical viewpoint, the insensitivity with respect to ξ_i is an approximation to the standard deviation of e_i, the corresponding component of the error, conditioned on all of the other components of the error. We can see this by recalling the results from Chapter 3 on conditional Gaussian distributions. If the covariance of e is Λ, then the covariance of e_i conditioned on all of the other components is $[(\Lambda^{-1})_{ii}]^{-1}$; therefore, the conditional standard deviation is $[(\Lambda^{-1})_{ii}]^{-1/2}$. From Equations (11.2-2) and (11.2-3), we can see that this expression equals the insensitivity. Note that the conditioning on the other elements in the statistical viewpoint corresponds directly to the constraint on the other elements in the geometric viewpoint.

A sensitivity analysis will detect one of the most obvious kinds of estimation difficulty—parameters which have little or no effect on the system response. If a parameter has no effect on the system response, then it should be obvious that the system response data give no basis for an estimate of the parameter; in statistical terms, the system is unidentifiable. Similarly, if a parameter has little effect on the system response, then there is little basis for an estimate of the parameter; we can expect the estimates to be inaccurate.

Checking for parameters which have no effect on the system response may seem like an academic exercise, considering that practical problems would not be likely to have such irrelevant parameters. In fact, this seemingly trivial difficulty is extremely common in practical applications. It can arise from typographical or other errors in input to computer programs. Perhaps the most common example of this problem is attempting to estimate the effect of an input which is identically zero. The input might either be validly zero, in which case its effect cannot be estimated, or the input signal might have been destroyed or misplaced by sensor or programming problems.

The sensitivity is a reasonable indicator of accuracy only when we are estimating a single parameter, because the estimates of other parameters are never exact, as the sensitivity analysis assumes. The sensitivity analysis ignores all effects of correlation between parameters; we can evaluate the sensitivity with respect to a parameter without even knowing what other parameters are being estimated. When more than one parameter is estimated, the sensitivity gives only a lower bound for the error estimate. The error band is always at least as large as the sensitivity regardless of what other parameters are estimated; correlation effects between parameters can increase, but never decrease, the error band. In other words, high sensitivity is a necessary, but not sufficient, condition for an accurate estimate.

In practice, correlation effects tend to increase the error band so much that the sensitivity is virtually useless as an indicator of accuracy. The sensitivity analysis is usually useful only for detecting the problem of completely irrelevant parameters. The sensitivity will not indicate when the effect of a parameter is indistinguishable from the effects of other parameters, a more common problem.

11.2.2 Correlation

We noted in the previous section that correlations among parameters result in much larger error bands than indicated by the sensitivities alone. The inadequacy of the sensitivity as a measure of estimate accuracy has led to the widespread use of the statistical correlations to indicate accuracy. We will see in this section that the correlations also give an incomplete picture of the accuracy.

The statistical correlation between two error components e_i and e_j is defined to be

$$\mathrm{corr}(e_i,e_j) = E\{e_i e_j\}/\sqrt{[E\{e_i^2\}E\{e_j^2\}]}$$

assuming that the means of e_i and e_j are zero. In terms of Λ, the covariance matrix of e, the correlation is

$$\mathrm{corr}(e_i,e_j) = \Lambda_{ij}/\sqrt{(\Lambda_{ii}\Lambda_{jj})} \tag{11.2-5}$$

Geometrically, the correlations are related to the eccentricity of the confidence ellipsoid. If the sensitivities with respect to all of the unknown parameters are equal (which we can always arrange by a scale change), and if the correlations are all zero, then the confidence ellipsoid is spherical. As the magnitudes of the correlations become larger, the eccentricity of the scaled ellipsoid increases. The magnitude of the correlations can never exceed 1, except through approximations or round-off errors in the computation.

The definition above is for the unconditional, or full correlations. Whenever the term correlation appears without a modifier, it implicitly means the unconditional correlation. We can also define conditional correlations, although they are less commonly used. The definition of the conditional correlation is identical to that of the unconditional correlations, except that the expected values are all conditioned on all of the parameters other than the two under consideration. We can express the conditional correlation of e_i and e_j as

$$\mathrm{cond\ corr}(e_i,e_j) = -\Gamma_{ij}/\sqrt{(\Gamma_{ii}\Gamma_{jj})} \tag{11.2-6}$$

where $\Gamma = \Lambda^{-1}$. This is similar to the expression for the unconditional correlation, the difference being that Γ replaces Λ and the sign is changed.

If there are only two unknowns, the conditional and unconditional correlations are identical. If there are more than two unknowns, the conditional and unconditional correlations can give quite different pictures. Consider the case in which Γ is an N-by-N matrix with 1's on the diagonal and with all of the off-diagonal elements equal to X. As X, the conditional correlation, approaches $-1/(N - 1)$, the full correlation approaches 1. In the limit, when X equals $-1/(N - 1)$, the Γ matrix is singular. Thus, for large N, the full correlations can be quite high even when all of the conditional correlations are low. This same example inverts to show that the converse also is true.

There are three objections to using the correlations, full or conditional, as primary indicators of accuracy. First, although the correlations give information about the shape of the confidence ellipsoid, they completely ignore its size. Figure (11.2-2) shows two confidence ellipsoids. Ellipse A is completely contained within ellipse B and is, therefore, clearly preferable; yet ellipse B has zero correlation and ellipse A has significant correlation. From this example, it is obvious that accurate estimates can have high correlations and poor estimates can have low correlations. To evaluate the accuracy of the estimates, you need information about the sensitivities as well as about the correlations; neither alone is adequate.

As a more concrete example of the interplay between correlation and sensitivity, consider a scalar linear system:

$$z(t_i) = Du(t_i) + H \tag{11.2-7}$$

We wish to estimate D. Both D and the bias H are unknown. The input $u(t_i)$ is an angular position of some control device. Suppose that the input time-history is as shown in Figure (11.2-3). A large portion of the energy in this input is from the steady-state value of 90°; the energy in the pulse is much smaller. This input is highly correlated with a constant bias input. Therefore, the estimate of D will be highly correlated with the estimate of H. (If this point is not obvious, we can choose a few time points on the figure and compute the corresponding covariance matrix.) The sensitivity with respect to D is high; because of the large values of u, small changes in D cause large changes in z.

Now we consider the same system, with the input shown in Figure (11.2-4). Both the correlation and the sensitivity are much lower than they were for the input of Figure (11.2-3). These changes balance each other, resulting in the same accuracy in estimating D. The inputs shown in the two figures are identical, but measured with respect to reference axes rotated by 90°. The choice of reference axis is a matter of convention which should not affect the accuracy; it does, however, affect both the sensitivity and correlation.

This example illustrates that the correlation alone is not a reasonable measure of accuracy. By redefining the reference axis of the input in this example, we can change the correlation at will to any value between -1 and 1.

The second objection to the use of correlations as indicators of accuracy is more serious because it cannot be answered by simply looking at sensitivities and correlations together. In the same way that sensitivities are one-dimensional tools, correlations are two-dimensional tools. The utility of a tool restricted to two-dimensional subspaces is limited. Three simple examples of idealized but realistic situations serve to illustrate the dimensional limitations of the correlations. These examples involve free lateral-directional oscillation of an aircraft.

For the first example, there is a yaw-rate feedback to the rudder and a rudder-to-aileron interconnect. Thus the aileron and rudder signals are both proportional to yaw rate. In this case, the conditional correlations of the aileron, rudder, and yaw-rate derivatives are 1 (or nearly so with imperfect data). Conditioned on the aileron derivatives being known exactly, changes in the rudder derivative estimates can be exactly compensated for by changes in the yaw-rate derivative estimates; thus, the conditional correlation is 1. The

unconditional correlations, however, are easily seen to be only 1/2. Changes in the rudder derivative estimates must be compensated for by some combination of changes in the aileron and yaw-rate derivative estimates. Since there are no constraints on how much of the compensation must come from the aileron and how much from the yaw-rate derivative estimates, the unconditional correlations would be 1/2 (because, on the average, 1/2 of the compensation would come from each source).

For the second example, no feedback is present and there is a neutrally damped, dutch-roll oscillation (or a wing rock). The sideslip, roll-rate, and yaw-rate signals are thus all sinusoids of the same frequency, with different phases and amplitudes. Taken two at a time, these signals have low correlations. The conditional correlations consider only two parameters at a time, and thus the conditional correlations of the derivatives will be low. Nonetheless, the three signals are linearly dependent when all are considered together, because they can all be written as linear combinations of a sine wave and a cosine wave at the dutch-roll frequency. The unconditional correlations of the derivatives will be 1 (or nearly so with imperfect data).

Both of the above examples have three-dimensional correlation problems, which prevent the parameters from being identifiable. The conditional correlations are low in one case, and the unconditional correlations are low in the other. Although neither alone is sufficient, examination of both the conditional and unconditional correlations will always reveal three-dimensional correlation problems.

For the third example, suppose that a wing leveler feeds back bank angle to the aileron, and that a neutrally damped dutch roll is present with the feedback on. There are then four pertinent signals (sideslip, roll rate, yaw rate, and aileron) that are sinusoids with the same frequency and different phases. In this case, both the conditional and the unconditional correlations will be low. Nonetheless, there is a correlation problem which results in unidentifiable parameters. This correlation problem is four-dimensional and cannot be seen using the two-dimensional correlations.

The full and conditional correlations are closely related to the eigenvalues of 2-by-2 submatrices of the Λ and Γ matrices, respectively, normalized to have unity diagonal elements. Specifically, the eigenvalues are 1 plus the correlation and 1 minus the correlation; thus, high correlations correspond to large eigenvalue spreads. Higher-order correlations would be investigated using eigenvalues of larger submatrices. Looked at in this light, the investigation of 2-by-2 submatrices is revealed as an arbitrary choice dictated by its familiarity more than by any objective criterion. The eigenvalues of the full normalized Λ and Γ matrices would seem more appropriate tools. These eigenvalues and the corresponding eigenvectors can provide some information, but they are seldom used. In principle, small eigenvalues of the normalized Γ matrix or large eigenvalues of the normalized Λ matrix indicate correlations among the parameters with significant components in the corresponding eigenvectors. Note that the eigenvalues of the unnormalized Γ and Λ matrices are of little use in studying correlations, because scaling effects tend to dominate.

The last objection to the use of the correlations is the difficulty of presentation. It is impractical to display the estimated correlations graphically in a problem with more than a handful of unknowns. The most common presentation is simply to print the matrix of estimated correlations. This option offers little improvement in comprehensibility over simply printing the Λ matrix. If there are a large number of experiments, it is pointless to print all of the correlation matrices. Such a nongraphical presentation cannot reasonably give a coherent picture of the system analyzed.

11.2.3 Cramer-Rao Bound

The Cramer-Rao bound is the last of the statistics based on the confidence ellipsoid. It proves to be the most useful of these statistics. The Cramer-Rao bound is often referred to by other names, including the standard deviation and the uncertainty level. We will consider both statistical and nonstatistical interpretations of the Cramer-Rao bound.

The Cramer-Rao bound of an estimated scalar parameter is the standard deviation of the error in that parameter. Strictly speaking, the term Cramer-Rao bound applies only to the approximation to the standard deviation obtained from the Cramer-Rao inequality. For the purposes of this section, the properties are similar, regardless of the source of the standard deviation. In terms of the Λ matrix, the Cramer-Rao bound of the ith element of ξ is $(\Lambda_{ii})^{1/2}$.

The Cramer-Rao bound is closely related to the insensitivity. Both are standard deviations of the error, the only difference being that the insensitivity is the conditional standard deviation, whereas the Cramer-Rao bound is unconditional. They are also computationally similar, the difference being in whether the inversion is of the matrix or of the individual element.

The geometric relationship between the Cramer-Rao bound and the insensitivity is particularly revealing. The Cramer-Rao bound on ξ_i is the largest change that you can make in ξ_i and still remain within the confidence ellipsoid. During this search, the other components are free to take any values that keep the point within the confidence ellipsoid. This definition is identical to the geometric definition of the insensitivity, except that the other components are constrained to the estimated values in the definition of insensitivity. This constraint is directly related to the statistical conditioning in the definition of the insensitivity; the Cramer-Rao bound has no such constraints and is an unconditional standard deviation.

The Cramer-Rao bound must always be at least as large as the insensitivity, because releasing a constraint can never make the solution of a maximization problem smaller. This fact relates to our previous statement that correlation effects can increase, but never decrease, the error band defined by the insensitivity. Figure (11.2-5) illustrates the geometric interpretation of the Cramer-Rao bounds and insensitivities in a two-dimensional example.

To prove that the Cramer-Rao bound is the solution to the above optimization problem, we will state and prove a more general result. (The general result is actually easier to prove.)

Theorem 11.2-1 Given a fixed vector x and a positive definite symmetric matrix H, the maximum of x^*y, subject to the constraint that $x^*Hx \leq 1$, is given by $\sqrt{(y^*H^{-1}y)}$.

Proof Since x^*y has no unconstrained local extrema, the solution must lie on the constraint boundary; therefore, the inequality in the constraint can be replaced by an equality. This constrained optimization problem can be restated by the use of Lagrange multipliers (Luenberger, 1969) as the unconstrained minimization of

$$f(x,\lambda) = x^*y - \frac{1}{2}\lambda(x^*Hx - 1) \tag{11.2-8}$$

where λ is the scalar Lagrange multiplier. The maximum is found by setting the gradients to zero as follows:

$$0 = \nabla_x f(x,\lambda) = y - \lambda Hx \tag{11.2-9}$$

$$0 = \frac{\partial}{\partial\lambda} f(x,\lambda) = -\frac{1}{2}(x^*Hx - 1) \tag{11.2-10}$$

From Equation (11.2-9) we have

$$x = \lambda^{-1}H^{-1}y \tag{11.2-11}$$

Substituting this into Equation (11.2-10) gives

$$y^*H^{-1}\lambda^{-1}H\lambda^{-1}H^{-1}y - 1 = 0 \tag{11.2-12}$$

or

$$\lambda^{-2}y^*H^{-1}y = 1 \tag{11.2-13}$$

or

$$\lambda = \sqrt{(y^*H^{-1}y)} \tag{11.2-14}$$

Substituting into Equation (11.2-11) gives

$$x = \frac{H^{-1}y}{\sqrt{(y^*H^{-1}y)}} \tag{11.2-15}$$

and thus

$$x^*y = \frac{y^*H^{-1}y}{\sqrt{(y^*H^{-1}y)}} = \sqrt{(y^*H^{-1}y)}$$

at the solution. This is the result sought.

The specific case of y being a unit vector along the ξ_i axis gives the form claimed for the Cramer-Rao bound of the ξ_i element.

The general form of Theorem (11.2-1) has other applications. The value of any linear combination of the parameters can be expressed as ξ^*y for some fixed y-vector. Thus the general form shows how to evaluate the accuracy of arbitrary linear combinations of parameters. This form applies to many situations where the sum, difference, or other combination of multiple parameters is of interest.

On the basis of this geometric picture, we can think of the Cramer-Rao bounds as insensitivities that are computed accounting for all parameter correlations. The computation and interpretation of the Cramer-Rao bounds are valid in any number of dimensions. In this respect, the Cramer-Rao bounds contrast with the insensitivities, which are one-dimensional tools, and the correlations, which are two-dimensional tools. The Cramer-Rao bounds are thus the best of the theoretical measures of accuracy that can be evaluated for a single experiment.

11.3 OTHER MEASURES OF ACCURACY

The previous sections have discussed the Cramer-Rao bound and other accuracy statistics based on the confidence ellipsoid. Although the Cramer-Rao bound is the best single analytical measure of accuracy, over-reliance on any single source of accuracy data is dangerous. Uncritical use of the Cramer-Rao bound can give extremely misleading results in realistic situations, as discussed by Maine and Iliff (1981b). This section discusses alternate accuracy measures, which can supplement the Cramer-Rao bound.

11.3.1 Bias

The bias of an estimator is occasionally cited as an indicator of accuracy. We do not consider it a useful indicator in most circumstances. This section is limited to a brief exposition of the reasons for this judgment.

Section 4.2.1 defines the bias of an estimator. Bias arises from several sources. Some estimators are intrinsically biased, regardless of the nature of the data. Random noise in the data often causes a bias. The bias from random noise sometimes goes to zero asymptotically for estimators matched to the noise characteristics. Finally, the inevitable modeling errors in analyzing real systems cause all estimators to be biased, even asymptotically. Most discussions of bias refer, implicitly or explicitly, to asymptotic bias. Even for idealized cases with no modeling error, estimators are seldom unbiased for finite time.

There are two reasons why the bias is of minimal use as a measure of accuracy. First, the bias reflects only the consistent errors; it ignores random scatter. As illustrated in Section 4.2.1, it is possible for an estimator to give ludicrous individual estimates which average out to a small or zero bias. This property is intrinsic to the definition of the bias.

Second, the bias is difficult to compute in most cases. If we could compute the bias, we could subtract it from the estimates to obtain revised estimates that were unbiased. (Some estimators use this technique.)

In some cases, it may be practical to compute a bound on the magnitude of the bias from a particular source, even when we cannot compute the actual bias. Although they are rarely used, such bounds can give a reasonable indication of the likely magnitude of the error from some sources. This is the most constructive use of bias information in evaluating accuracy.

In contrast, the often-repeated statements that a given estimator is or is not asymptotically unbiased are of little practical use. Most of the estimators considered in this document are asymptotically unbiased when the assumptions used in the derivation are true. The statement that other estimators are biased under the same conditions amounts to a restatement of the universal principle that estimators are biased in the presence of modeling error. Thus arguments about which of two estimators is biased are silly. These arguments reduce to the issue of what assumptions to use, an issue best addressed directly.

Although quantitative measures of bias may not be available, the analyst should always consider the issue of bias due to modeling error. Bias errors are added to all other types of error in the estimates. Unfortunately, some bias errors are impossible to detect solely by analyzing the data. The estimates can be repeatable with little scatter and appear to be accurate by all other measures, and still have large bias errors. An example of this type of problem is a calibration error in a nonredundant instrument. The only way to avoid such problems is to be meticulous in executing and documenting every step of the application, including modeling, instrumentation, and data handling. No automatic tests exist that adequately substitute for such care.

11.3.2 Scatter

When there are several experiments at the same condition, the scatter of the estimates is an indication of accuracy. We can also evaluate scatter about a smooth fairing of the estimates in a series of experiments with gradually changing conditions. This approach assumes that the parameters change smoothly as a function of experimental condition.

The scatter has a significant advantage over many of the theoretical measures of accuracy discussed below. The scatter measures the actual performance that some of the theoretical measures are trying to predict. Therefore the scatter includes several effects, such as random errors in measuring the experiment conditions, that are ignored in the theoretical predictions. You can gain the most information, of course, by considering both the observed scatter and the theoretical predictions.

An inherent weakness in the use of scatter as a gauge of accuracy is that several data points are required to define it. Depending on the application, this objection can range from inconsequential to insurmountable. A related problem is that the scatter does not show the accuracy of individual points, some of which may be better than others. For instance, if only two conflicting data points are available, the scatter gives no hint as to which is more reliable. Figure (11.3-1) shows estimates of the parameter Cn_p obtained from flight data of a PA-30 aircraft. The scatter is large, showing estimates of both signs.

Figure (11.3-2) shows the same data segregated into rudder and aileron maneuvers. In this case, the scatter makes it evident that the aileron maneuvers result in far more consistent estimates of Cn_p than do the rudder maneuvers. Had there been only one or two aileron and one or two rudder maneuvers available, there would have been no way to deduce from the scatter that the aileron maneuvers were superior for estimating this parameter.

The scatter shares a weakness with most of the theoretical accuracy measures in that it does not account for consistent errors (i.e., biases). Many occurrences can result in small scatter about an incorrect value. The scatter, therefore, should be regarded as a lower bound. The estimates can be worse than is indicated by the scatter, but are seldom better.

Maine and Iliff (1981b) discuss well-documented situations in which the scatter is significantly larger than the Cramer-Rao bounds. In all such cases, we regard the scatter as a more realistic measure of the magnitude of the errors. The Cramer-Rao bound is still a reasonable means of determining which individual experiments are most accurate, but may not give a reasonable magnitude of the error.

In spite of its problems, the data scatter is an easily used tool for evaluating accuracy, and it should always be examined when sufficient data points are available to define it.

11.3.3 Engineering Judgment

Engineering judgment is the oldest measure of estimate reliability. Even with the theoretical accuracy measures now available, the need for judgment remains; the theoretical measures are merely tools which supply more information on which to base the judgment. By definition, the process of applying engineering judgment

cannot be described precisely and quantitatively, or there would be no judgment involved. Algorithms can be devised to search for specific problems, but the engineer still needs to make a final unautomated judgment. Therefore, this section will simply list some of the factors most often considered in making a judgment.

One of the most basic factors in judging the accuracy of the estimates is the anticipated accuracy. The engineer usually has *a priori* knowledge of how accurately one can reasonably expect to be able to estimate the parameters. This knowledge can be based on previous experience, awareness of the relative importance and linear dependence of the parameters, and the quality of experimental data obtained.

Another basic criterion is the reasonability of the estimated parameter values. Before analysis is begun, we usually know the approximate range of values of the parameters. Drastic deviations from this range are reason to suspect the estimates unless we discover the reason for the poor prediction or we independently verify the suspect value.

We have previously mentioned the role of engineering judgment in evaluating model adequacy. The engineer must look for violations of specific assumptions made in deriving the model, and for unexplained problems that may indicate modeling errors. Both the estimator and the theoretical measures of accuracy can be invalidated by modeling errors. The magnitude of the modeling-error effects must be judged.

The engineer judges the quality of the fit of the measured and estimated time histories. The characteristics of this fit can give indications of many problems. Many modeling error problems first become apparent as poor time-history fits. Failed sensors and data processing errors or omissions are among the other classes of problems which can be deduced from the fits.

Finally, engineering judgment is used to assemble and weigh all of the available information about the estimates You must combine the judgmental factors with information from the theoretical tools in order to give a final best estimate of the parameters and of their accuracies.

11.4 MODEL STRUCTURE DETERMINATION

In the previous sections, we have largely assumed that the assumed model form is correct. This is never strictly true in practice. Therefore, we must always consiuer the possible effects of modeling error as a special issue. The tools discussed in Section 11.3 can help in the evaluation of these effects.

In this section, we specifically examine the question of determining the best model structure for parameter estimation. One approach to minimizing the effects of model structure errors is to use a model structure which is close to that of the true system. There are, however, definite limits to this principle. The limitations arise both in how accurate you <u>can</u> make the model and in how accurate you <u>should</u> make it.

In the field of simulation, it is almost axiomatic that the simulation fidelity improves as more detail is added to the model. Practical considerations of cost and the degree of required fidelity dictate the level of detail included in the model. Simulation and system identification are closely related fields, and we might expect that such a basic principle would be common to both. Contrary to this expectation, system identification sometimes obtains better results from a simple than from a detailed model. The use of too detailed a model is probably one of the most common sources of difficulty in the practical application of system identification.

The problems that arise from too detailed a model are best illustrated by a simple example. Presume that Figure (11.4-1) shows experimental data from a system with a scalar input U, and a scalar output Z. The line in the figure is the best linear fit to the data. This line appears to be a reasonable representation of the system.

To investigate possible nonlinear effects, consider the case of polynomial models. It is obvious that the error between the model output and the experimental data will become smaller as the order of the model increases. High-order polynomials include lower-order polynomials as specific cases (we have no requirement that the high-order coefficient be nonzero), so the best second-order fit is at least as good as the best linear fit, and so forth. When the order of the polynomial becomes one less than the number of data points, the model will exactly match the experimental data (unless input values were repeated).

Figure (11.4-2) shows such a perfect match of the data from Figure (11.4-1). Although the data points are matched perfectly, the curve oscillates wildly. The simple linear fit of Figure (11.4-1) is probably a much better representation of the system, even though the model of Figure (11.4-2) is more detailed. We could say that the model of Figure (11.4-2) is fitting the noise instead of the true response.

Essentially, as the model complexity increases, and more unknown parameters are estimated, the problem approaches the black-box system-identification problem where there are no assumptions about the model form. We have previously shown that the pure black-box problem is insoluble. One can deduce only a finite amount of information about the system from a finite amount of experimental data. The engineer provides, in the form of an assumed model structure, the rest of the information required to solve the system-identification problem. As the assumed model structure becomes more general, it provides less information, and thus more of the information must be deduced from the experimental data. Eventually, one reaches a point where the information available is insufficient; the estimation algorithms then perform poorly, giving ridiculous results.

The Cramer-Rao bound gives a statistical basis for estimating whether the experimental data contain sufficient information to reliably estimate the parameters in a model. This and related statistics can be used to determine the number and selection of terms to include in the model (Klein and Batterson, 1983; Gupta, Hall, and Trankle, 1978; and Trankle, Vincent, and Franklin, 1982). The basic principle is to include in the model only those terms that can be accurately estimated from the available experimental data. This process, known as model structure determination, is described in further detail in the cited references. We will restrict our discussion to the general nature and applicability of model structure determination.

Automatic model structure determination is often viewed as a panacea that eliminates the necessity for model selection to be based on engineering judgment and knowledge of the phenomenology of the system. Since we have repeatedly emphasized that pure black-box system identification is impossible, such claims for automatic model determination must be viewed with suspicion.

There is a basic fallacy in the argument that automatic model structure determination can replace engineering judgment in selecting a model. The model structure determination algorithms are not creative; they can only test candidate models suggested by the engineer. In fact, the model structure determination algorithms are a type of parameter estimation in disguise, in which the parameter is an index indicating which model is to be used. In a way, model structure determination is easier than most parameter estimation. At each stage, there are only two possible values for a term, zero or nonzero; whereas most parameter estimation demands that a specific value be picked from the entire real line. This task does not approach the scope of the black-box system-identification problem in which the number of possible models is a high order of infinity.

Engineering judgment is still needed, therefore, to select the types of candidate models to be tested. If the candidate models are not appropriate, the results will be questionable. The very best that could be expected from an automatic algorithm in this circumstance would be rejection of all of the candidates (and not all automatic tests have even that much capability. No automatic algorithm can suggest creative improvements that it has not been specifically programed for.

Consider a system with an actual output of $Z = \sin(U)$. Assume that a polynomial model has been selected by the engineer, and automatic structure determination has been used to determine what order polynomial to use. The task is hopeless in this form. The data can be fit arbitrarily well with a polynomial of a high enough order, but the polynomial form does not describe the essence of the system. In particular, the finite polynomial will not be valid for extrapolating system performance outside of the range of the experimental data.

In the above system, consider three ranges of U-values: $|U| < 0.1$, $|U| < 1.0$, and $|U| < 10.0$. In the range $|U| < 0.1$, the linear polynomial $Z = U$ is a close approximation, as shown in Figure (11.4-3). The extrapolation of this approximation to the range $|U| < 1.0$ introduces noticeable errors, as shown in Figure (11.4-4). Over this range, the approximation $Z = U - U^3/6$ is reasonable. If we expand our view to the range $|U| < 10.0$, as in Figure (1.5-5), then neither the linear nor the third-order polynomial is at all representative of the sine function. It would require at least a seventh-order polynomial to match even the gross characteristics of the sine function over this range; a good match would require a still higher order.

Another problem with automatic model-structure determination is that it gives only a statistical estimate. Like all estimates, it is imperfect. If no better information is available, it is appropriate to use automatic model structure determination as the best guess. If, however, facts about the model structure are deducible from the physics of the system, it is silly to throw away known facts and use imperfect estimates. (This is one of the most basic principles in the entire field of system identification, not just in model structure determination: if a fact is known, use it and save the estimation theory for cases in which it is needed.)

The most basic problem with automatic model structure determination lies in the statement of the problem. The very term "model structure determination" is misleading, because there is seldom a correct model to determine. Even when there is a correct model, it may be far too complicated for practical purposes. The real model structure determination problem is not to determine some nonexistent "correct" model structure, but to determine an adequate model structure. We discussed the idea of adequate models in Section 1.4; the idea of an adequate model structure is an intimate part of the idea of an adequate model.

This basic issue is addressed briefly, if at all, in most of the literature on model structure determination. Many papers generate simulated data with a specified model, and then demonstrate that a proposed model structure determination algorithm can determine the correct model. This approach has little to do with the real issue in model structure determination.

The previous paragraphs have emphasized the numerous problems of automatic model structure determination. That these problems exist does not mean that automatic model-structure determination is worthless, only that the mindless application of it is dangerous. Automatic model structure determination can be a valuable tool when used with an appreciation of its limitations. Most good model structure determination programs allow the engineer to override the statistical decision and force specific terms to be included or omitted. This approach makes good use of both the theory and the judgment, so that the theory is used as a tool to aid the judgment and to warn against some types of poor judgment, but the end responsibility lies with the engineer.

11.5 EXPERIMENT DESIGN

The previous discussion has, for the most part, assumed that a specific set of experimental data has already been gathered. In some cases, this is a valid assumption. In other cases, the opportunity is available to specify the experiments to be performed and the measurements to be taken. This section gives a brief overview of the subject of designing experiments for parameter identification. We leave detailed discussion to works cited in the references.

Methods for experiment design fall into two major categories. The first category is that of methods based on numerical optimization. Such methods choose an input, subject to appropriate constraints, which minimizes the Cramer-Rao bound or some related error estimate. Goodwin (1982) and Plaetschke and Schulz (1979) give theoretical and practical details of some optimization approaches to input design.

Experiment design is often strongly constrained by practical considerations; in the extreme case, the constraints completely specify the input, leaving no latitude for design. In a design based on numerical optimization, the constraints must be expressed mathematically. This derivation of such expressions is sometimes straightforward, as when a control device is limited by a physical stop at a specific position. In other cases, the constraints involve issues such as safety that are difficult to quantify as precise limits.

Slight changes in the form of the constraints can change the entire character of the theoretical optimum input. Because the constraints are one of the major influences in the experiment design, adopting simplified constraint forms solely because they are easy to analyze is often inadvisable. In particular, "soft" constraints in the form of a cost penalty proportional to the square of the input are almost never accurate representations of practical constraints.

Most practical experiment design falls into the second major category, methods based more on heuristic design than on formal optimization of a cost function. Such designs draw heavily on the engineer's understanding of the system. There are several widely applicable rules of thumb to help heuristic experiment design; some of them consider issues such as frequency content, modal excitation, and independence. Plaetschke and Schulz (1979) describe some of these rules, and evaluate inputs based on them.

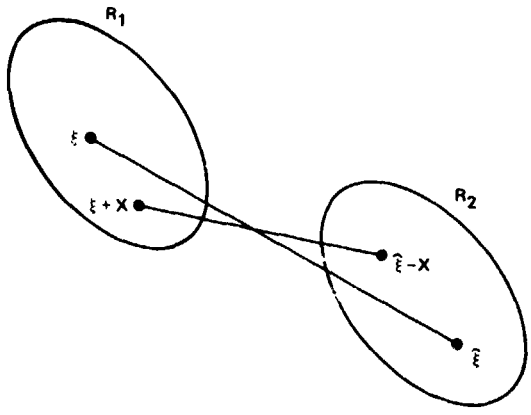


Figure (11.1-1). Construction of R_2.

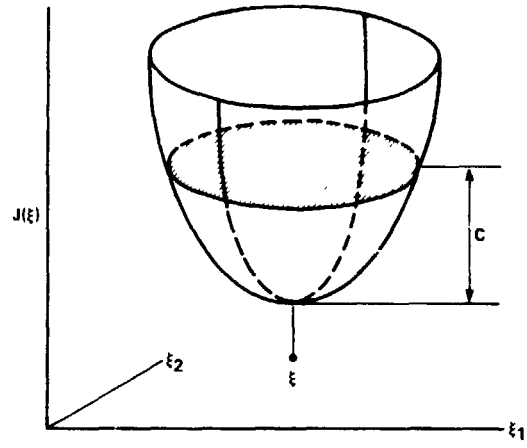


Figure (11.1-3). Construction of two-dimensional confidence ellipsoid.

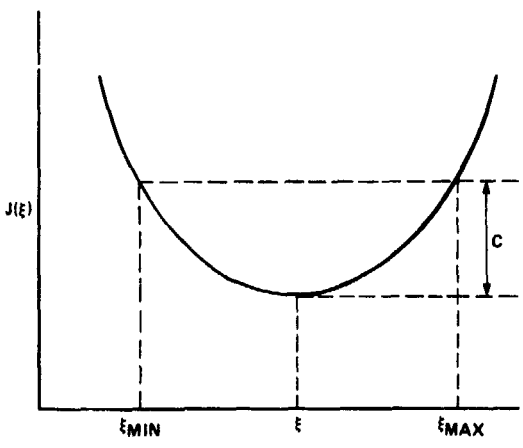


Figure (11.1-2). Construction of one-dimensional confidence ellipsoid.

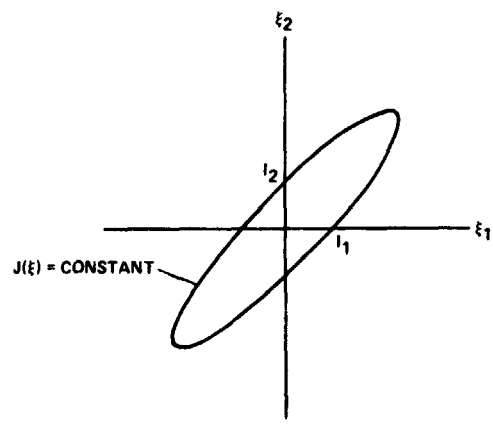


Figure (11.2-1). Geometric interpretation of insensitivity.

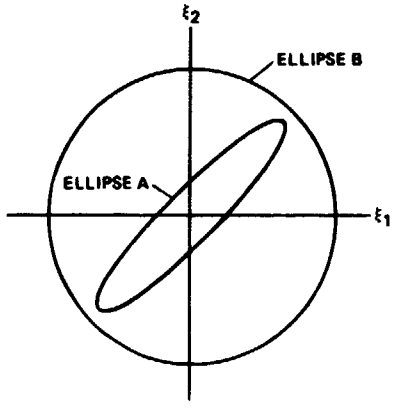


Figure (11.2-2). Correlation and sensitivity.

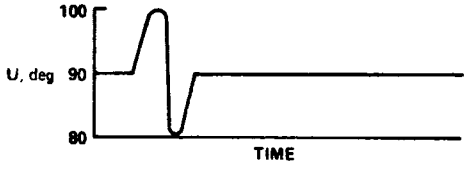


Figure (11.2-3). High correlation and high sensitivity.

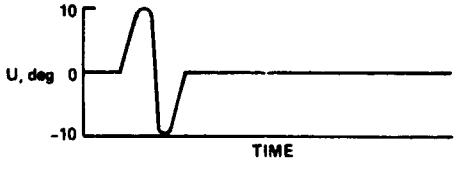


Figure (11.2-4). Low correlation and low sensitivity.

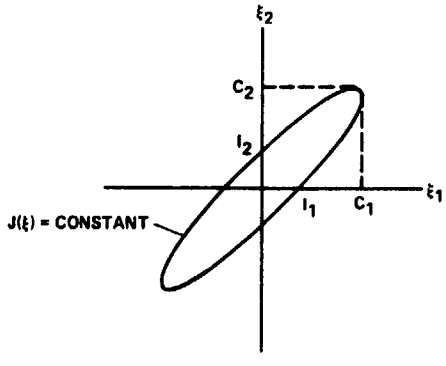


Figure (11.2-5). Cramer-Rao bounds and insensitivities.

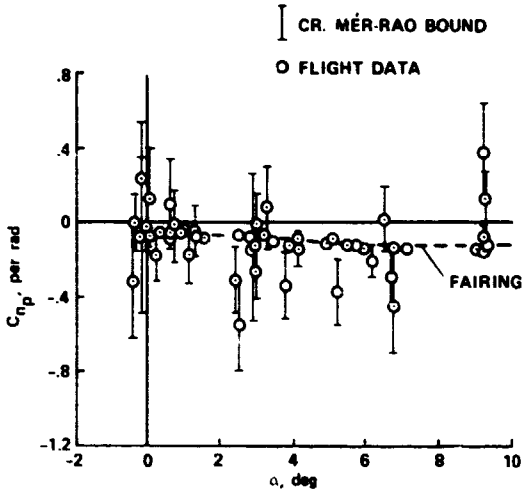


Figure (11.3-1). Estimates of C_{n_p}.

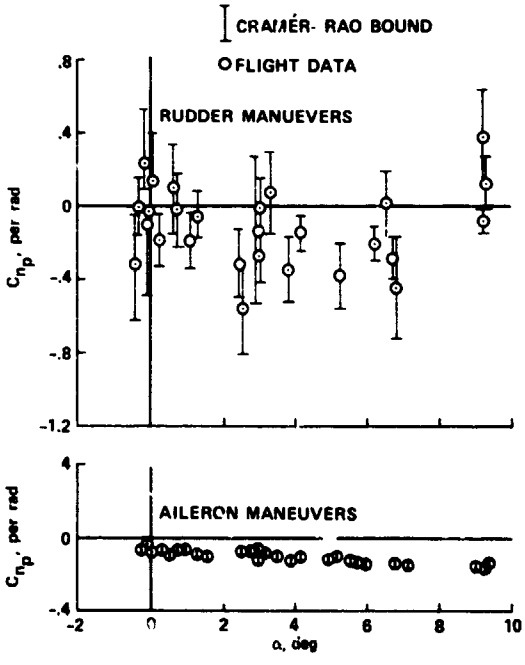


Figure (11.3-2). Estimates of C_{n_p}, segregated by input u.

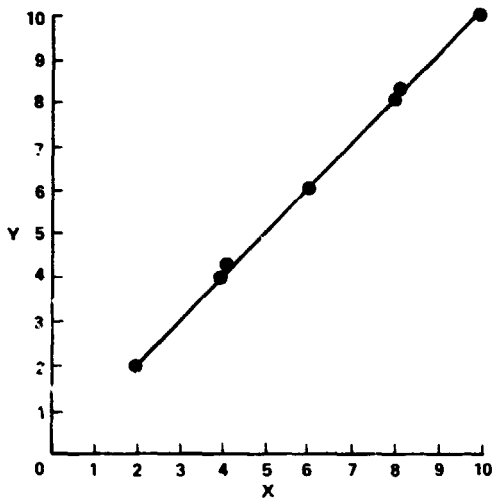


Figure (11.4-1). Best linear fit of noise data.

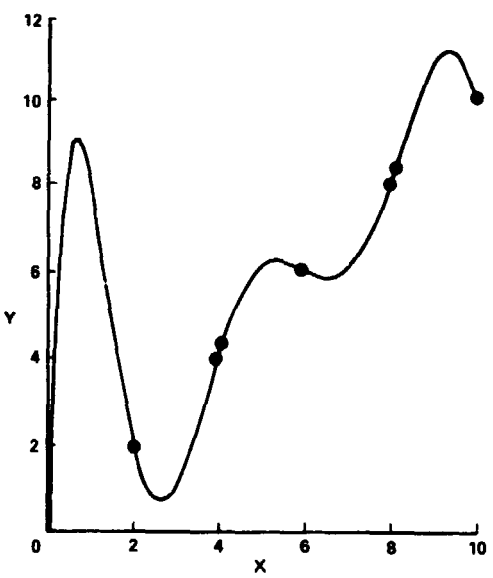


Figure (11.4-2). Exact polynomial match of noise data.

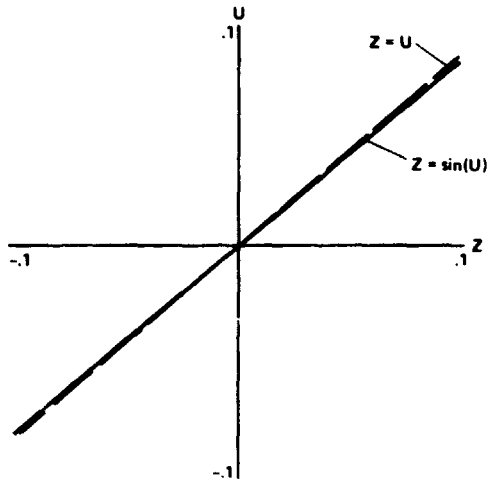


Figure (11.4-3). Z = sin(U) in the range |U| < 0.1.

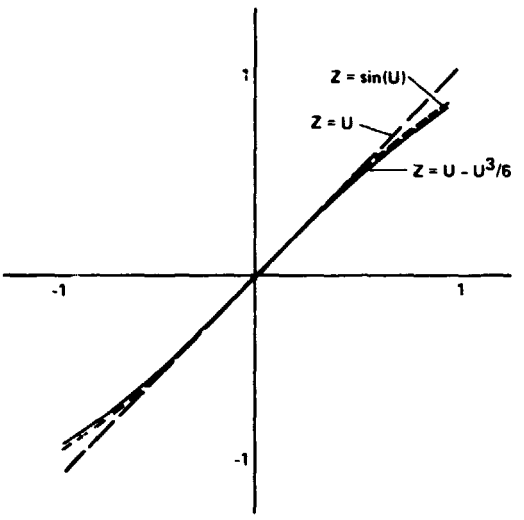


Figure (11.4-4). Z = sin(U) in the range |U| < 1.0.

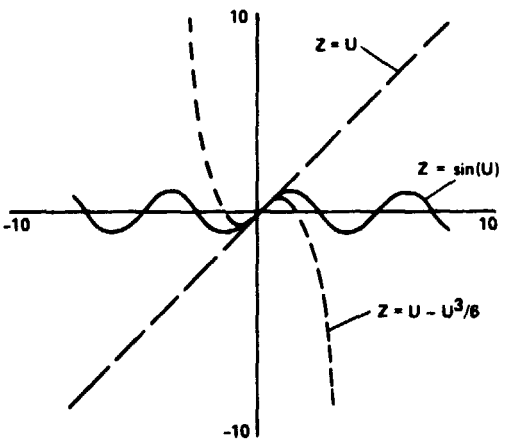


Figure (11.4-5). Z = sin(U) in the range |U| < 10.0.

CHAPTER 12

12.0 SUMMARY

In this document, we have presented the theoretical background of statistical estimators for dynamic systems, with particular emphasis on maximum-likelihood estimators. An understanding of this theoretical background is crucial to the practical application of the estimators; the analyst needs to know the capabilities and limitations of the estimators. There are several examples of artificially complicated problems that succumb to simple approaches, and seemingly trivial questions that have no answers.

A thorough understanding of the system being analyzed is necessary to complement this theoretical background. No amount of theoretical sophistication can compensate for the lack of such understanding. The entire theory rests on the basis of the assumptions made about the system characteristics. The theory can give only limited help in validating or refuting such assumptions.

Errors and unexpected difficulties are inevitable in any substantial parameter estimation project. The eventual success of the project hinges on the analyst's ability to recognize unreasonable results and diagnose their causes. This ability, in turn, requires an understanding of both estimation theory and the system being analyzed. Problems can range from obvious instrumentation failures to subtle modeling inconsistencies and identifiability problems.

Probably the most difficult part of parameter estimation is to straddle the fine line between models too simple to adequately represent the system and models too complicated to be identifiable. There is no conservative position on this issue; excesses in either direction can be fatal. The solution is typically iterative, using diagnostic skills to detect problems and make improvements until an adequate result is obtained. The problem is exacerbated by there being no correct answer.

Neither is there a single correct method to solve parameter estimation problems. Although we have castigated some practices as demonstrably poor, we make no attempt to establish as dogma any particular method. The material of this document is intended more as a set of tools for parameter estimation problems. The selection of the best tools for a particular task is influenced by factors other than the purely theoretical. Better results often come from a crude, but adequate, method that the analyst thoroughly understands than from a sophisticated, but unfamiliar, method. We recommend the attitude expressed by Gauss (1809, p. 108):

> It is always profitable to approach the more difficult problems in several ways, and not to despise the good although preferring the better.

PRECEDING PAGE BLANK NOT FILMED

PAGE 124 INTENTIONALLY BLANK

APPENDIX A

A.0 MATRIX RESULTS

This appendix presents several matrix results used in the body of the book. The derivations are mostly exercises in simple matrix algebra. Various of these results are given in numerous other documents; Goodwin and Payne (1977, appendix E) present most of them.

A.1 MATRIX INVERSION LEMMAS

Consider a square, nonsingular matrix Λ, partitioned as

$$\Lambda = \begin{bmatrix} \Lambda_{11} & \Lambda_{12} \\ \Lambda_{21} & \Lambda_{22} \end{bmatrix} \tag{A.1-1}$$

where Λ_{11} and Λ_{22} are square. Define the inverse of Λ to be Γ, similarly partitioned as

$$\Lambda^{-1} = \Gamma = \begin{bmatrix} \Gamma_{11} & \Gamma_{12} \\ \Gamma_{21} & \Gamma_{22} \end{bmatrix} \tag{A.1-2}$$

where Γ_{11} is the same size as Λ_{11}. We want to express the partitions Γ_{ij} in terms of the Λ_{ij}. To derive such expressions, we need to assume that either Λ_{11} or Λ_{22} is invertible; if both are singular, there is no useful form. Consider first the case where Λ_{11} is invertible.

<u>Lemma A.1-1</u> Given Λ and Γ partitioned as in Equations (A.1-1) and (A.1-2), assume that Λ and Λ_{11} are invertible. Then $(\Lambda_{22} - \Lambda_{21}\Lambda_{11}^{-1}\Lambda_{12})$ is invertible and the partitions of Γ are given by

$$\Gamma_{11} = \Lambda_{11}^{-1} - \Lambda_{11}^{-1}\Lambda_{12}(\Lambda_{22} - \Lambda_{21}\Lambda_{11}^{-1}\Lambda_{12})^{-1}\Lambda_{21}\Lambda_{11}^{-1} \tag{A.1-3}$$

$$\Gamma_{12} = -\Lambda_{11}^{-1}\Lambda_{12}(\Lambda_{22} - \Lambda_{21}\Lambda_{11}^{-1}\Lambda_{12})^{-1} \tag{A.1-4}$$

$$\Gamma_{21} = -(\Lambda_{22} - \Lambda_{21}\Lambda_{11}^{-1}\Lambda_{12})^{-1}\Lambda_{21}\Lambda_{11}^{-1} \tag{A.1-5}$$

$$\Gamma_{22} = (\Lambda_{22} - \Lambda_{21}\Lambda_{11}^{-1}\Lambda_{12})^{-1} \tag{A.1-6}$$

<u>Proof</u> The condition $\Lambda\Gamma = I$ gives the four equations

$$\Lambda_{11}\Gamma_{12} + \Lambda_{12}\Gamma_{22} = 0 \tag{A.1-7}$$

$$\Lambda_{21}\Gamma_{11} + \Lambda_{22}\Gamma_{21} = 0 \tag{A.1-8}$$

$$\Lambda_{11}\Gamma_{11} + \Lambda_{12}\Gamma_{21} = I \tag{A.1-9}$$

$$\Lambda_{21}\Gamma_{12} + \Lambda_{22}\Gamma_{22} = I \tag{A.1-10}$$

and the condition $\Gamma\Lambda = I$ gives the four equations

$$\Gamma_{11}\Lambda_{12} + \Gamma_{12}\Lambda_{22} = 0 \tag{A.1-11}$$

$$\Gamma_{21}\Lambda_{11} + \Gamma_{22}\Lambda_{21} = 0 \tag{A.1-12}$$

$$\Gamma_{11}\Lambda_{11} + \Gamma_{12}\Lambda_{21} = I \tag{A.1-13}$$

$$\Gamma_{21}\Lambda_{12} + \Gamma_{22}\Lambda_{22} = I \tag{A.1-14}$$

Equations (A.1-7) and (A.1-12), respectively, give

$$\Gamma_{12} = -\Lambda_{11}^{-1}\Lambda_{12}\Gamma_{22} \tag{A.1-15}$$

$$\Gamma_{21} = -\Gamma_{22}\Lambda_{21}\Lambda_{11}^{-1} \tag{A.1-16}$$

Substitute Equation (A.1-15) into Equation (A.1-10) and substitute Equation (A.1-16) into Equation (A.1-14) to get

$$(\Lambda_{22} - \Lambda_{21}\Lambda_{11}^{-1}\Lambda_{12})\Gamma_{22} = I \tag{A.1-17}$$

$$\Gamma_{22}(\Lambda_{22} - \Lambda_{21}\Lambda_{11}^{-1}\Lambda_{12}) = I \tag{A.1-18}$$

PRECEDING PAGE BLANK NOT FILMED

PAGE 126 INTENTIONALLY BLANK

By the assumption of invertibility of Λ, the Γ_{ij} exist and satisfy Equations (A.1-7) to (A.1-14). The assumption of invertibility of Λ_{11} then assures, through the above substitutions, that Γ_{22} satisfies Equations (A.1-17) and (A.1-18). Therefore $(\Lambda_{22} - \Lambda_{21}\Lambda_{11}^{-1}\Lambda_{12})$ is invertible and Γ_{22} is given by Equation (A.1-6).

Substituting Equation (A.1-6) into Equations (A.1-15) and (A.1-16) gives Equations (A.1-4) and (A.1-5). Finally, substituting Equation (A.1-5) into Equation (A.1-9) and solving for Γ_{11} gives Equation (A.1-3), completing the proof.

The case where Λ_{22} is nonsingular is simply a permutation of the same lemma.

Lemma A.1-2 Given Λ and Γ partitioned as in Equations (A.1-1) and (A.1-2), assume that Λ and Λ_{22} are invertible. Then $(\Lambda_{11} - \Lambda_{12}\Lambda_{11}^{-1}\Lambda_{21})$ is invertible and the partitions of Γ are given by

$$\Gamma_{11} = (\Lambda_{11} - \Lambda_{12}\Lambda_{22}^{-1}\Lambda_{21})^{-1} \tag{A.1-19}$$

$$\Gamma_{12} = -(\Lambda_{11} - \Lambda_{12}\Lambda_{22}^{-1}\Lambda_{21})^{-1}\Lambda_{12}\Lambda_{22}^{-1} \tag{A.1-20}$$

$$\Gamma_{21} = -\Lambda_{22}^{-1}\Lambda_{21}(\Lambda_{11} - \Lambda_{12}\Lambda_{22}^{-1}\Lambda_{21})^{-1} \tag{A.1-21}$$

$$\Gamma_{22} = \Lambda_{22}^{-1} - \Lambda_{22}^{-1}\Lambda_{21}(\Lambda_{11} - \Lambda_{12}\Lambda_{22}^{-1}\Lambda_{21})^{-1}\Lambda_{12}\Lambda_{22}^{-1} \tag{A.1-22}$$

Proof Define a reordered matrix

$$\Lambda' = \begin{bmatrix} \Lambda_{22} & \Lambda_{21} \\ \Lambda_{12} & \Lambda_{11} \end{bmatrix}$$

The inverse of Λ' is given by the corresponding reordering of Γ.

$$\Gamma' = \begin{bmatrix} \Gamma_{22} & \Gamma_{21} \\ \Gamma_{12} & \Gamma_{11} \end{bmatrix}$$

Then apply the previous lemma to Λ' and Γ'.

When both Λ_{11} and Λ_{22} are invertible, we can combine the above lemmas to obtain two other useful results.

Lemma A.1-3 Assume that two matrices A and C are invertible. Further assume that one of the expressions $(A - BC^{-1}D)$ or $(C - DA^{-1}B)$ is invertible. Then the other expression is also invertible and

$$(A - BC^{-1}D)^{-1} = A^{-1} - A^{-1}B(C - DA^{-1}B)^{-1}DA^{-1} \tag{A.1-23}$$

Proof Define $\Lambda_{11} = A$, $\Lambda_{12} = B$, $\Lambda_{21} = D$, and $\Lambda_{22} = C$. In order to apply Lemmas (A.1-1) and (A.1-2), we first need to show that Λ as defined by Equation (A.1-1) is invertible.

If $(C - DA^{-1}B)$ is invertible, then the Γ_{ij} defined by Equations (A.1-3) to (A.1-6) satisfy Equations (A.1-7) to (A.1-14). Therefore Λ is invertible. Lemma (A.1-2) then gives the invertibility of $(A - BC^{-1}D)$, which is one of the desired results.

Conversely, if we assume that $(A - BC^{-1}D)$ is invertible, then the Γ_{ij} defined by Equations (A.1-19) to (A.1-22) satisfy Equations (A.1-7) to (A.1-14). Therefore Λ is invertible and Lemma (A.1-1) gives the invertibility of the expression $(C - DA^{-1}B)$.

Thus the invertibility of either expression implies invertibility of the other and of Λ. We can now apply both Lemmas (A.1-1) and (A.1-2). Equating the expressions for Γ_{11} given by Equations (A.1-3) and (A.1-19), and putting the result in terms of A, B, C, and D, gives Equation (A.1-23), completing the proof.

Lemma A.1-4 Given A, B, C, and D as in Lemma (A.1-3), with the same invertibility assumptions, then

$$A^{-1}B(C - DA^{-1}B)^{-1} = (A - BC^{-1}D)^{-1}BC^{-1} \tag{A.1-24}$$

Proof The proof is identical to that of Lemma (A.1-3), except that we equate the expressions for Γ_{12} given by Equations (A.1-4) and (A.1-20), giving Equation (A.1-24) as a result.

A.2 MATRIX DIFFERENTIATION

For several of the following results, it is convenient to define the derivative of a scalar with respect to a matrix. If f is a scalar function of the matrix A, we define df/dA to be a matrix with elements equal to the derivatives of f with respect to corresponding elements of A.

$$\left(\frac{df}{dA}\right)^{(i,j)} = \frac{df}{d(A^{(i,j)})} \tag{A.2-1}$$

Two simple relations involving the trace function are useful in manipulating the matrix and vector quantities we work with.

Result A.2-1 If x and y are two vectors of the same length, then

$$x^*y = \mathrm{tr}(yx^*) \tag{A.2-2}$$

Proof Both sides expand to $\sum_i x^{(i)}y^{(i)}$.

Result A.2-2 If A and B are two matrices of the same size, then

$$\sum_{i,j} A^{(i,j)}B^{(i,j)} = \mathrm{tr}(AB^*) \tag{A.2-3}$$

Proof Expand the right side, element by element.

Both of these results are special cases of the same relationship between inner products and outer products. The following result is a particular application of Result (A.2-2).

Result A.2-3 If $f(A)$ is a scalar function of the matrix A, and A is a function of the scalar x, then

$$\frac{df}{dx} = \mathrm{tr}\left(\frac{\partial f(A)}{\partial A}\frac{dA^*}{dx}\right) \tag{A.2-4}$$

Proof Use the chain rule with the individual elements of A to write

$$\frac{df}{dx} = \sum_{i,j} \frac{\partial f}{\partial A^{(i,j)}}\frac{dA^{(i,j)}}{dx} \tag{A.2-5}$$

Equation (A.2-4) then follows from Result (A.2-2) and the definition given by Equation (A.2-1).

Result A.2-4 If the matrix A is a function of x, then

$$\frac{d}{dx}(A^{-1}) = -A^{-1}\left(\frac{dA}{dx}\right)A^{-1} \tag{A.2-6}$$

wherever A is invertible.

Proof By the definition of the inverse

$$AA^{-1} = I \tag{A.2-7}$$

Take the derivative, using the chain rule.

$$\frac{d}{dx}(AA^{-1}) = \frac{dI}{dx} \tag{A.2-8}$$

$$\frac{dA}{dx}A^{-1} + A\frac{d}{dx}(A^{-1}) = 0 \tag{A.2-9}$$

Solving for $d/dx(A^{-1})$ gives Equation (A.2-6), as desired.

Result A.2-5 If A is invertible, and x and y are vectors, then

$$\frac{\partial}{\partial A}(x^*A^{-1}y) = -(A^{-1}yx^*A^{-1})^* \tag{A.2-10}$$

Proof Use result (A.2-4) to get

$$\frac{\partial}{\partial A^{(i,j)}}(x^*A^{-1}y) = -x^*A^{-1}\frac{\partial A}{\partial A^{(i,j)}}A^{-1}y \tag{A.2-11}$$

Now

$$\frac{\partial A}{\partial A^{(i,j)}} = e_i e_j^* \tag{A.2-12}$$

where e_i is a vector with zeros in all but the ith element, which is 1. Therefore

$$\frac{\partial}{\partial A^{(i,j)}} (x^*A^{-1}y) = -x^*A^{-1}e_i e_j^* A^{-1}y = -e_j^* A^{-1}yx^*A^{-1}e_i \tag{A.2-13}$$

which is the (i,j) element of $-(A^{-1}yx^*A^{-1})^*$. The definition of the matrix derivative then gives Equation (A.2-10) as desired.

<u>Result A.2-6</u> If A is invertible, then

$$\frac{\partial}{\partial A} \ln|A| = A^{*-1} \tag{A.2-14}$$

<u>Proof</u> Expanding the determinant by cofactors of the ith row gives

$$\ln|A| = \ln \sum_k A^{(i,k)}(\text{adj } A)^{(k,i)} \tag{A.2-15}$$

Taking the derivative with respect to $A^{(i,j)}$ gives

$$\frac{\partial}{\partial A^{(i,j)}} \ln|A| = \frac{(\text{adj } A)^{(j,i)}}{\sum_k A^{(i,k)}(\text{adj } A)^{(k,i)}} \tag{A.2-16}$$

because $(\text{adj } A)^{(k,i)}$ does not depend on $A^{(j,i)}$. Using Equation (A.2-15) and the expression for a matrix inverse in terms of the matrix of cofactors, we get

$$\frac{\partial}{\partial A^{(i,j)}} \ln|A| = \frac{(\text{adj } A)^{(j,i)}}{|A|} = (A^{-1})^{(j,i)} \tag{A.2-17}$$

Equation (A.2-14) then follows, as desired, from the definition of the derivative with respect to a matrix.

REFERENCES

Acton, Forman S.: Numerical Methods that Work. Harper & Row, New York, 1970.

Akaike, Hirotugu: A New Look at Statistical Model Identification. IEEE Trans. Automat. Contr., Vol. AC-19. No. 6, pp. 716-723, 1974.

Aoki, Masanao: Optimization of Stochastic Systems. Academic Press, New York, 1967.

Apostol, Tom M.: Calculus: Volume II. Xerox College Publishing, Waltham, Mass., 2nd ed., 1969.

Ash, Robert B.: Basic Probability Theory. John Wiley & Sons, Inc., New York, 1970.

Astrom, Karl J.: Introduction to Stochastic Control Theory. Academic Press, New York, 1970.

Astrom, Karl J. and Eykhoff, P.: System Identification—A Survey. Automatica, Vol. 7, pp. 123-162, 1970.

Bach, R. E. and Wingrove, R. C.: Applications of State Estimation in Aircraft Flight Data Analysis. AIAA paper 83-2087, 1983.

Balakrishnan, A. V.: Stochastic Differential Systems I. Filtering and Control—A Function Space Apprach. Lecture Notes in Economics and Mathematical Systems, 84, M. Beckman, G. Goos, and H. P. Kunzi, eds., Springer-Verlag, Berlin, 1973.

Balakrishnan, A. V.: Stochastic Filtering and Control. Optimization Software, Inc., Los Angeles, 1981.

Balakrishnan, A. V.: Kalman Filtering Theory. Optimization Software, Inc., New York, 1984.

Barnard, G. A.: Thomas Bayes Essay Toward Solving a Problem in the Doctrine of Chances. Biometrika, Vol. 45, 1958.

Bayes, Thomas: An Introduction to the Doctrine of Fluxions, and a Defence of the Mathematicians Against the Objections of the Author of the Analyst. John Noon, 1736. (See Barnard, 1958).

Bierman, G. J.: Factorization Methods for Discrete Sequential Estimation. Mathematics in Science and Engineering, Vol. 128, Academic Press, New York, 1977.

Brauer, Fred and Noel, John A.: Qualitative Theory of Ordinary Differential Equations. W. A. Benjamin, New York, 1969.

Cox, A. B. and Bryson, A. E.: Identification by a Combined Smoothing Nonlinear Programming Algorithm. Automatica, Vol. 16, pp. 689-694, 1980.

Cramér, Harald: Mathematical Methods of Statistics. Princeton University Press, Princeton, N.J., 1946.

Dixon, L. C. W.: Nonlinear Optimization. Crane, Russak & Co., New York, 1972.

Doetsch, K. H.: The Time Vector Method for Stability Investigations. A.R.C. R. & M. 2945, 1953.

Dongarra, J. J.; Moler, C. B.; Bunch, J. R.; and Stewart, G. W.: LINPACK User's Guide. SIAM, Philadelphia, 1979.

Etkin, B.: Dynamics of Atmospheric Flight. John Wiley & Sons, Inc., New York, 1958.

Eykhoff, P.: System Identification, Parameter and State Estimation. John Wiley & Sons, London, 1974.

Ferguson, Thomas S.: Mathematical Statistics: A Decision Theoretic Approach. Academic Press, New York, 1967.

Fisher, R. A.: On the Mathematical Foundations of Theoretical Statistics. Phil. Trans. Roy. Soc. London, Vol. 222, pp. 309-368, 1921.

Fiske, P. H. and Price, C. F.: A New Approach to Model Structure Identification. AIAA paper 77-1171, 1977.

Flack, Nelson D.: AFFTC Stability and Control Technique. AFFTC-TN-59-21, Edwards, California, 1959.

Foster, G. W.: The Identification of Aircraft Stability and Control Parameters in Turbulence. RAE TR 83025, 1983.

Garbow, B. S.; Boyle, J. M.; Dongarra, J. J.; and Moler, C. B.: Matrix Eigensystem Routines—EISPACK Guide Extension. Springer-Verlag, Berlin, 1977.

Gauss, Karl Friedrich: Theory of the Motion of the Heavenly Bodies Moving About the Sun in Conic Sections. Translated by Charles Henry Davis, Dover Publications, Inc., New York, 1847. Translated from: Theoria Motus, 1809.

Geyser, Lucille C. and Lehtinen, Bruce: Digital Program for Solving the Linear Stochastic Optimal Control and Estimation Problem. NASA TN D-7820, 1975.

Goodwin, Graham C.: An Overview of the System Identification Problem Experiment Design. Sixth IFAC Symposium on Identification and System Parameter Estimation, Washington, D.C., 1982.

Goodwin, Graham C. and Payne, Robert L.: Dynamic System Identification: Experiment Design and Data Analysis. Academic Press, New York, 1977.

Greenberg, H.: A Survey of Methods for Determining Stability Parameters of an Airplane from Dynamic Flight Measurement. NASA TN-2340, 1951.

Gupta, N. K.; Hall, W. E.; and Trankle, T. L.: Advanced Methods of Model Structure Determination from Test Data. AIAA J. Guidance and Control, Vol. 1, No. 3, 1978.

Gupta, N. K.; and Mehra, R. K.: Computational Aspects of Maximum Likelihood Estimation and Reduction in Sensitivity Function Calculations. IEEE Trans. on Automat. Contr., Vol AC-19, No. 6, pp. 774-783, 1974.

Hajdasinski, A. K.; Eykhoff, P.; Damen, A. A. H.; and van den Boom, A. J. W.: The Choice and Use of Different Model Sets for System Identification. Sixth IFAC Symposium on Identification and System Parameter Estimation, Washington, D.C., 1982.

Hodge, Ward F. and Bryant, Wayne H.: Monte Carlo Analysis of Inaccuracies in Estimated Aircraft Parameters Caused by Unmodeled Flight Instrumentation Errors. NASA TN D-7712, 1975.

Jategaonkar, R. and Plaetschke, E.: Maximum Likelihood Parameter Estimation from Flight Test Data for General Nonlinear Systems. DFVLR-FB 83-14, 1983.

Jazwinski, Andrew H.: Stochastic Processes and Filtering Theory. Academic Press, New York, 1970.

Kailath, T. and Lyung, L.: Asymptotic Behavior of Constant-Coefficient Riccati Differential Equations. IEEE Trans. Automat. Contr., Vol. AC-21, pp. 385-388, 1976.

Kalman, R. E. and Bucy, R. S.: New Results in Linear Filtering and Prediction Theory. Trans. ASME, Series D. Journal of Basic Engineering, Vol. 63, pp. 95-107, 1961.

Klein, Vladislav: On the Adequate Model for Aircraft Parameter Estimation. CIT, Cransfield Report Aero No. 28, 1975.

Klein, Vladislav and Batterson, James G.: Determination of Airplane Model Structure from Flight Data Using Splines and Stepwise Regression. NASA TP-2126, 1983.

Kushner, Harold: Introduction to Stochastic Control. Holt, Rinehart and Winston, Inc., New York, 1971.

Levan, N.: Systems and Signals. Optimization Software, Inc., New York, 1983.

Lipster, R. S. and Shiryayev, A. N.: Statistics of Random Processes I: General Theory. Springer-Verlag, New York, 1977.

Luenberger, David G.: Optimization by Vector Space Methods. John Wiley & Sons, New York, 1969.

Luenberger, David G.: Introduction to Linear and Nonlinear Programming. Addison-Wesley, Reading Mass., 1972.

Maine, Richard E.: Programmer's Manual for MMLE3, A General FORTRAN Program for Maximum Likelihood Parameter Estimation. NASA TP-1690, 1981.

Maine, Richard E. and Iliff, Kenneth W.: User's Manual for MMLE3, A General Fortran Program for Maximum Likelihood Parameter Estimation. NASA TP-1563, 1980.

Maine, Richard E. and Iliff, Kenneth W.: Formulation and Implementation of a Practical Algorithm for Parameter Estimation with Process and Measurement Noise. SIAM J. Appl. Math., Vol. 41, pp. 558-579, 1981(a).

Maine, Richard E. and Iliff, Kenneth W.: The Theory and Practice of Estimating the Accuracy of Dynamic Flight-Determined Coefficients. NASA RP-1077, 1981(b).

Meditch, J. S.: Stochastic Optimal Linear Estimation and Control. McGraw-Hill Book Co., New York, 1969.

Mehra, Raman K. and Lainiotis, Dimitri G. (eds): System Identification: Advances and Case Studies. Academic Press, New York, 1976.

Moler, C. B. and Stewart, G. W.: An Algorithm for Generalized Matrix Eigenvalue Problems. SIAM J. of Numerical Analysis, Vol. 10, pp. 241-256, 1973.

Moler, Cleve; and Van Loan, Charles: Nineteen Dubious Ways to Compute the Exponential of a Matrix. SIAM Review, Vol. 20, No. 4, pp. 801-836, 1978.

Nering, Evar D.: Linear Algebra and Matrix Theory. John Wiley & Sons, Inc., New York, 2nd ed., 1970.

Paige, Lowell J.; Swift, J. Dean; and Slobko, Thomas A.: Elements of Linear Algebra. Xerox College Publishing, Lexington, Mass., 2nd ed., 1974.

Papoulis, Athanasios: Probability, Random Variables, and Stochastic Processes. McGraw-Hill Book Co., New York, 1965.

Penrose, R.: A Generalized Inverse for Matrices. Proc. Cambridge Phil. Soc. 51, pp. 406-413, 1955.

Pitman, E. J. G.: Some Basic Theory for Statistical Inference. Chapman and Hall, London, 1979.

Plaetschke, E. and Schulz, G.: Practical Input Signal Design. AGARD Lecture Series No. 104, 1979.

Polak, E.: Computational Methods in Optimization: A Unified Approach. Academic Press, New York, 1971.

Potter, James E.: Matrix Quadratic Solutions. SIAM J. Appl. Math., Vol. 14, pp. 496-501, 1966.

Rampy, John M. and Berry, Donald T.: Determination of Stability Derivatives from Flight Test Data by Means of High Speed Repetitive Operation Analog Matching. FTC-TDR-64-8, Edwards, Calif., 1964.

Rao, S. S.: Optimization, Theory and Applications. Wiley Eastern Limited, New Delhi, 1979.

Royden, H. L.: Real Analysis. The MacMillan Co., London, 1968.

Rudin, Walter: Real and Complex Analysis. McGraw-Hill Book Co., New York, 1974.

Schweppe, Fred C.: Uncertain Dynamic Systems. Prentice-Hall, Inc., Englewood Cliffs, N.J., 1973.

Sorensen, John A.: Analysis of Instrumentation Error Effects on the Identification Accuracy of Aircraft Parameters. NASA CR-112121, 1972.

Sorensen, Harold W.: Parameter Estimation; Principles and Problems. Marcel Dekker, Inc., New York, 1980.

Strang, Gilbert: Linear Algebra and Its Applications. Academic Press, New York, 1980.

Trankle, T. L.; Vincent, J.H.; and Franklin, S. N.: System Identification of Nonlinear Aerodynamic Models. AGARDograph, The Techniques and Technology of Nonlinear Filtering and Kalman Filtering, 1982.

Vaughan, David R.: A Nonrecursive Algebraic Solution for the Discrete Riccati Equation. IEEE Trans. Automat. Contr., Vol. AC-15, pp. 597-599, 1970.

Wiberg, Donald M.: State Space and Linear Systems. McGraw-Hill Book Co., New York, 1971.

Wilkinson, J. H.: The Algebraic Eigenvalue Problem. Clarendon Press, Oxford, 1965.

Wilkinson, J. H. and Reinsch, C.: Handbook for Automatic Computation. Volume II. Linear Algebra, Part 2. Springer-Verlag, New York, 1971.

Wolowicz, Chester H.: Considerations in the Determination of Stability and Control Derivatives and Dynamic Characteristics from Flight Data. AGARD Rep. 549-Part 1, 1966.

Wolowicz, Chester H. and Holleman, Euclid C.: Stability-Derivative Determination from Flight Data. AGARD Report 224, 1958.

Zacks, Shelemyahu: The Theory of Statistical Inference. John Wiley & Sons, New York, 1971.

Zadeh, Lotfi A. and Desoer, Charles A.: Linear System Theory. McGraw-Hill Book Co., New York, 1963.

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA RP-1138		

4. Title and Subtitle	5. Report Date
Identification of Dynamic Systems – Theory and Formulation	February 1985
	6. Performing Organization Code
7. Author(s)	8. Performing Organization Report No.
Richard E. Maine and Kenneth W. Iliff	H-1255
	10. Work Unit No.
9. Performing Organization Name and Address	
NASA Ames Research Center Dryden Flight Research Facility P.O. Box 273 Edwards, California 93523	11. Contract or Grant No.
	13. Type of Report and Period Covered
12. Sponsoring Agency Name and Address	Reference Publication
National Aeronautics and Space Administration Washington, D.C. 20546	14. Sponsoring Agency Code
	RTOP 505-43-11

15. Supplementary Notes

This document was prepared for the AGARD Flight Mechanics Panel, Flight Test Techniques Group, as part of AGARDograph 300, Flight Test Techniques Series.

16. Abstract

This document addresses the problem of estimating parameters of dynamic systems. The aim is to present the theoretical basis of system identification and parameter estimation in a manner that is complete and rigorous, yet understandable with minimal prerequisites. The document concentrates on maximum likelihood and related estimators. The approach used requires familiarity with calculus, linear algebra, and probability, but does not require knowledge of stochastic processes or functional analysis. No previous background in statistics is assumed.

The treatment emphasizes unification of the various areas in estimation theory and practice. For example, the theory of estimation in dynamic systems is treated as a direct outgrowth of the static system theory.

Topics covered include basic concepts and definitions; numerical optimization methods; probability; statistical estimators; estimation in static systems; stochastic processes; state estimation in dynamic systems; output error, filter error, and equation error methods of parameter estimation in dynamic systems; and the accuracy of the estimates.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
System identification; Parameter estimation; Maximum likelihood; Kalman filters; Optimization; Statistics; Dynamic systems	Unclassified-Unlimited STAR category 66

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	141	A07

For sale by the National Technical Information Service, Springfield, Virginia 22161.

NASA-Langley, 1985