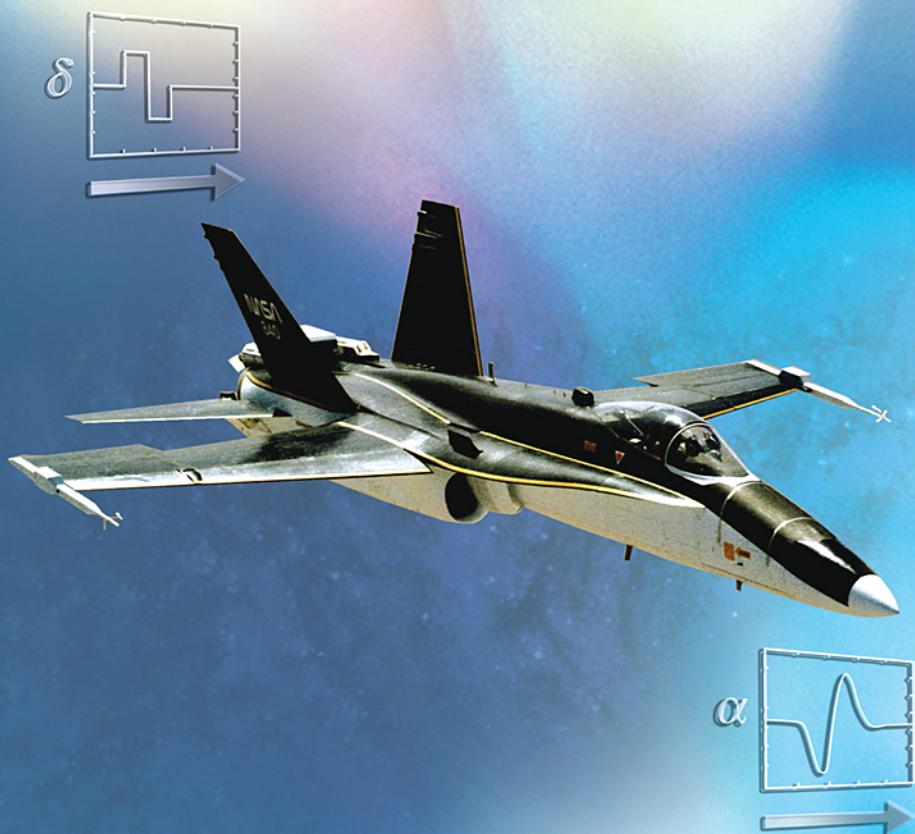


# AIRCRAFT SYSTEM IDENTIFICATION

## THEORY AND PRACTICE

Vladislav Klein  
Eugene A. Morelli



AMERICAN INSTITUTE OF  
AERONAUTICS AND ASTRONAUTICS

AIAA EDUCATION SERIES  
JOSEPH A. SCHETZ  
EDITOR-IN-CHIEF

# Aircraft System Identification

## Theory and Practice

This page intentionally left blank

# Aircraft System Identification

## Theory and Practice

**Vladislav Klein**

George Washington University  
NASA Langley Research Center  
Hampton, Virginia

**Eugene A. Morelli**

NASA Langley Research Center  
Hampton, Virginia



**EDUCATION SERIES**

Joseph A. Schetz

Series Editor-in-Chief  
Virginia Polytechnic Institute and State University  
Blacksburg, Virginia

Published by

American Institute of Aeronautics and Astronautics, Inc.  
1801 Alexander Bell Drive, Reston, VA 20191

MATLAB® is a registered trademark of The Mathworks, Inc.

American Institute of Aeronautics and Astronautics, Inc., Reston, Virginia

1 2 3 4 5

**Library of Congress Cataloging-in-Publication Data**

Klein, Vladislav.

Aircraft system identification : theory and practice / Vladislav Klein,  
Eugene A. Morelli.

p. cm. – (AIAA education series)

ISBN 1-56347-832-3 (alk. paper)

1. Aeronautics—Systems engineering. 2. Airplanes—Design and construction. 3. System  
identification. I. Morelli, Eugene A. II. Title. III. Series.

TL670.K554 2006

629.134'1-dc22

2006047656

Copyright © 2006 by the American Institute of Aeronautics and Astronautics, Inc. All rights reserved.  
Printed in the United States. No part of this publication may be reproduced, distributed, or transmitted,  
in any form or by any means, or stored in a database or retrieval system, without the prior written per-  
mission of the publisher.

Data and information appearing in this book are for informational purposes only. AIAA is not respon-  
sible for any injury or damage resulting from use or reliance, nor does AIAA warrant that use or  
reliance will be free from privately owned rights.

## AIAA Education Series

### Editor-in-Chief

Joseph A. Schetz

*Virginia Polytechnic Institute and State University*

### Editorial Board

Takahira Aoki  
*University of Tokyo*

David K. Holger  
*Iowa State University*

Edward W. Ashford

Rakesh K. Kapania

*Virginia Polytechnic Institute  
and State University*

Karen D. Barker

*The Brahe Corporation*

Brian Landrum

*University of Alabama, Huntsville*

Robert H. Bishop

*University of Texas at Austin*

Tim C. Lieuwen

*Georgia Institute of Technology*

Claudio Bruno

*University of Rome*

Michael Mohaghegh

*The Boeing Company*

Aaron R. Byerley

*U.S. Air Force Academy*

Conrad F. Newberry

*Naval Postgraduate School*

Richard Colgren

*University of Kansas*

Mark A. Price

*Queen's University Belfast*

Kajal K. Gupta

*NASA Dryden Flight  
Research Center*

James M. Rankin

*Ohio University*

Rikard B. Heslehurst

*Australian Defence Force  
Academy*

David K. Schmidt

*University of Colorado,  
Colorado Springs*

David M. Van Wie

*Johns Hopkins University*

This page intentionally left blank

## Foreword

The AIAA Editorial Board is pleased to present *Aircraft System Identification: Theory and Practice* by Vladislav Klein and Eugene Morelli. This important area in the aerospace field is covered in 12 chapters and 4 appendices in almost 500 pages. MATLAB® software is included.

These authors are very well qualified to write on this subject as a result of their long experience in the area, and they have written this book in a manner that will make it of interest to both students and experts in the field. Readers of this book will also likely find the companion volume in this series, *Aircraft and Rotorcraft System Identification: Engineering Methods with Flight-Test Examples* by Mark Tischler and Robert Remple, to be relevant to their needs and interests.

The AIAA Education Series aims to cover a very broad range of topics in the general aerospace field, including basic theory, applications, and design. Information about the complete list of titles can be found on the last pages of this volume. The philosophy of the series is to develop textbooks that can be used in a university setting, instructional materials for continuing education and professional development courses, and also books that can serve as the basis for independent study. Suggestions for new topics or authors are always welcome.

**Joseph A. Schetz**  
Editor-in-Chief  
AIAA Education Series

This page intentionally left blank

## Table of Contents

<b>Preface</b> .....	xiii
<b>Chapter 1. Introduction</b> .....	1
1.1    System Identification Applied to Aircraft. ....	2
1.2    Outline of the Text. ....	4
References .....	6
<b>Chapter 2. Elements of System Theory</b> .....	9
2.1    Mathematical Modeling. ....	9
2.2    System Identification and Parameter Estimation ....	17
2.3    Aircraft System Identification .....	19
2.4    Summary and Concluding Remarks .....	24
References .....	25
<b>Chapter 3. Mathematical Model of an Aircraft</b> .....	27
3.1    Reference Frames and Sign Conventions. ....	28
3.2    Rigid-Body Equations of Motion. ....	31
3.3    Rotational Kinematic Equations .....	36
3.4    Navigation Equations .....	37
3.5    Force Equations in Wind Axes .....	38
3.6    Collected Equations of Motion .....	40
3.7    Output Equations. ....	42
3.8    Aerodynamic Model Equations .....	45
3.9    Simplifying the Equations of Motion .....	60
3.10    Summary and Concluding Remarks .....	71
References .....	72
<b>Chapter 4. Outline of Estimation Theory</b> .....	75
4.1    Properties of Estimators .....	77
4.2    Parameter Estimation .....	79
4.3    State Estimation .....	83

4.4	Summary and Concluding Remarks . . . . .	92
	References . . . . .	94
<b>Chapter 5. Regression Methods . . . . .</b>		<b>95</b>
5.1	Ordinary Least Squares . . . . .	97
5.2	Generalized Least Squares (GLS) . . . . .	132
5.3	Nonlinear Least Squares . . . . .	137
5.4	Model Structure Determination . . . . .	138
5.5	Data Collinearity . . . . .	158
5.6	Data Partitioning . . . . .	174
5.7	Summary and Concluding Remarks . . . . .	176
	References . . . . .	179
<b>Chapter 6. Maximum Likelihood Methods . . . . .</b>		<b>181</b>
6.1	Dynamic System with Process Noise . . . . .	182
6.2	Output-Error Method . . . . .	191
6.3	Computational Aspects . . . . .	195
6.4	Equation-Error Method . . . . .	216
6.5	Summary and Concluding Remarks . . . . .	220
	References . . . . .	221
<b>Chapter 7. Frequency Domain Methods . . . . .</b>		<b>225</b>
7.1	Transforming Measured Data to the Frequency Domain . . . . .	226
7.2	Frequency Response . . . . .	228
7.3	Maximum Likelihood Estimator . . . . .	232
7.4	Output-Error Method . . . . .	237
7.5	Equation-Error Method . . . . .	240
7.6	Complex Linear Regression . . . . .	243
7.7	Low-Order Equivalent System Identification . . . . .	250
7.8	Summary and Concluding Remarks . . . . .	258
	References . . . . .	259
<b>Chapter 8. Real-Time Parameter Estimation . . . . .</b>		<b>261</b>
8.1	Recursive Least Squares . . . . .	264
8.2	Time-Varying Parameters . . . . .	270
8.3	Regularization . . . . .	273
8.4	Frequency-Domain Sequential Least Squares . . . . .	275
8.5	Extended Kalman Filter . . . . .	282
8.6	Summary and Concluding Remarks . . . . .	285
	References . . . . .	287
<b>Chapter 9. Experiment Design . . . . .</b>		<b>289</b>
9.1	Data Acquisition System . . . . .	290

9.2	Instrumentation . . . . .	297
9.3	Input Design . . . . .	299
9.4	Recommendations . . . . .	323
9.5	Open-Loop Parameter Estimation from Closed-Loop Data . . .	327
9.6	Summary and Concluding Remarks . . . . .	329
	References . . . . .	329
<b>Chapter 10. Data Compatibility . . . . .</b>		<b>333</b>
10.1	Kinematic Equations . . . . .	333
10.2	Data Reconstruction . . . . .	336
10.3	Aircraft Instrumentation Errors . . . . .	338
10.4	Model Equations for Data Compatibility Check . . . . .	340
10.5	Instrumentation Error Estimation Methods . . . . .	344
10.6	Summary and Concluding Remarks . . . . .	348
	References . . . . .	349
<b>Chapter 11. Data Analysis . . . . .</b>		<b>351</b>
11.1	Filtering . . . . .	351
11.2	Smoothing . . . . .	352
11.3	Interpolation . . . . .	366
11.4	Numerical Differentiation . . . . .	367
11.5	Signal Comparisons . . . . .	369
11.6	Finite Fourier Transform . . . . .	370
11.7	Power Spectrum Estimation . . . . .	376
11.8	Maneuver Visualization . . . . .	380
11.9	Summary and Concluding Remarks . . . . .	381
	References . . . . .	382
<b>Chapter 12. MATLAB® Software . . . . .</b>		<b>383</b>
12.1	Overview . . . . .	383
12.2	Linear Regression . . . . .	388
12.3	Model Structure Determination . . . . .	390
12.4	Output-Error Parameter Estimation . . . . .	395
12.5	Frequency Domain . . . . .	399
12.6	Real-Time Parameter Estimation . . . . .	403
12.7	Input Design . . . . .	406
12.8	Data Compatibility . . . . .	411
12.9	Data Analysis . . . . .	415
	References . . . . .	422
<b>Appendix A. Mathematical Background . . . . .</b>		<b>423</b>
A.1	Linear Algebra . . . . .	423

A.2	Complex Numbers . . . . .	432
A.3	Calculus . . . . .	434
A.4	Polynomial Splines. . . . .	437
<b>Appendix B. Probability, Statistics, and Random Variables . . . . .</b>		<b>439</b>
B.1	Random Variables . . . . .	439
B.2	Statistics . . . . .	444
B.3	Random Process Theory . . . . .	451
	References. . . . .	454
<b>Appendix C. Reference Information . . . . .</b>		<b>455</b>
C.1	Properties of Air and the Atmosphere . . . . .	455
C.2	Elementary Aerodynamics . . . . .	456
C.3	Mass/Inertia Properties . . . . .	459
C.4	Greek Alphabet and Conversion Factors . . . . .	461
	Reference . . . . .	461
<b>Appendix D. F-16 Nonlinear Simulation . . . . .</b>		<b>463</b>
D.1	F-16 Aircraft. . . . .	463
D.2	Equations of Motion. . . . .	463
D.3	Engine Model . . . . .	468
D.4	Aerodynamic Model. . . . .	469
D.5	Atmosphere Model. . . . .	470
D.6	Mass/Inertia Properties . . . . .	471
D.7	Analysis Tools. . . . .	471
	References. . . . .	476
<b>Index . . . . .</b>		<b>479</b>
<b>Supporting Materials . . . . .</b>		<b>485</b>

## Preface

This book is based on the research and teaching activities of the authors, mostly at the NASA Langley Research Center in Hampton, Virginia. It is intended as a source for researchers and practicing engineers, as well as a textbook for postgraduate and senior-level courses. We assume that the reader has undergraduate-level familiarity with topics such as differential equations, linear algebra, probability, statistics, and aerodynamics. For readers without this preparation, the material in the appendices should provide adequate background information. In writing the book, we sought to provide a comprehensive treatment of both the theoretical underpinnings and the practical application of aircraft modeling based on experimental data, which is also known as aircraft system identification.

The scope of the book is restricted to fixed-wing aircraft, assumed to be rigid bodies. However, the methods discussed are generally applicable and can also be applied to flexible vehicles, rotorcraft, and spacecraft, among many other applications. Methods discussed in the book are used routinely for risk reduction during flight envelope expansion of new aircraft or modified configurations, comparison with results from wind-tunnel tests and analytic methods such as computational fluid dynamics (CFD), control law design and refinement, dynamic analysis, simulation, flying qualities assessments, accident investigations, and other tasks.

Aircraft system identification is an applied engineering discipline, which means that a complete education in this field of study must include practical hands-on experience. To that end, the book includes a software toolbox, written in MATLAB®, which implements most of the methods discussed in the text. The toolbox is called SIDPAC, which is an acronym for System IDentification Programs for AirCraft. Each purchaser of this book can download SIDPAC at <http://www.aiaa.org/publications/supportmaterials>. Chapter 12 provides complete documentation for the software.

SIDPAC can be and should be applied to modeling problems of particular interest to the reader. Example problems using real data from flight tests and wind-tunnel experiments are provided in the text. Calculations and plots for these examples (except for the X-29 examples, due to data restrictions) can be regenerated in full detail using the `sidpac_text_examples.m` script. The reader can study this file to see exactly how each example was done. SIDPAC also includes demonstration scripts, which provide complete examples of how the main tools in SIDPAC can be used to solve aircraft system

identification problems. Each demonstration script has a name that includes the characters `_demo.m`. Each directory of files in SIDPAC includes a file named `Contents.m`, which is a list of the files in the directory, along with short descriptions of each file.

All SIDPAC routines are written in MATLAB<sup>®</sup>, so it is possible for the user to study the implementations and to use the software as a basis for writing additional routines to solve related or specific problems. No special MATLAB<sup>®</sup> toolboxes are necessary to use SIDPAC, just standard MATLAB<sup>®</sup>.

The software package also includes a nonlinear simulation of the F-16 aircraft, written in MATLAB<sup>®</sup> and documented in Appendix D. This software can be used to simulate realistic flight-test maneuvers to provide data for exercising the SIDPAC tools, and for further tool development. The code includes tools to generate linear models from the F-16 nonlinear simulation. The user can run simulated maneuvers using linear F-16 models and then use SIDPAC tools to find estimates of the parameters in the linear models, based on the simulated data. This provides feedback on whether or not the modeling was successful, because (unlike in a real flight-test situation) the correct values of the model parameters will be known.

A companion book, also published by the AIAA, entitled *Aircraft and Rotorcraft System Identification: Engineering Methods with Flight-Test Examples*, by M. B. Tischler and R. K. Remple, is highly recommended for more information on aircraft system identification. Studying both that book and the current one will provide the reader with a very comprehensive knowledge and understanding of the subject, along with a wide variety of practical tools.

The authors are grateful to both the NASA Langley Research Center and George Washington University for financial support and their important roles in our careers and at several stages of the book preparation. The authors also thank their colleagues at NASA, particularly James G. Batterson and Patrick C. Murphy, and many graduate students, for their insights and assistance both with the book preparation and with our research and quest for understanding in this interesting and important area of study. Special thanks to Dr. Ravi Prasanth for his careful reading of the manuscript, which resulted in corrections in several equations and a clearer exposition.

Finally, and most importantly, we wish to extend our deep gratitude to our families, for encouragement and support over the long period of time required to produce this book, and indeed over the course of our lives. It is a pleasure to dedicate this book to all of you.

**Vladislav Klein  
Eugene A. Morelli  
August 2006**

## 1

## Introduction

One of the oldest and most fundamental of all human scientific pursuits is developing mathematical models for physical systems based on imperfect observations or measurements. This activity is known as system identification. This book describes the theory and practice of system identification applied to aircraft.

A good practical definition of system identification is that of Zadeh<sup>1</sup>:

*System identification is the determination, on the basis of observation of input and output, of a system within a specified class of systems to which the system under test is equivalent.*

Implicit in the preceding definition is the practical fact that the mathematical model of a physical system is not unique. In general, the guiding principle for model selection is the parsimony principle, which states that of all models in a specified class that exhibit the desired characteristics, the simplest one should be preferred. There are both theoretical and practical reasons for the parsimony principle, and these will be discussed further throughout the book. The preceding definition also mentions that system identification is based on observations of input and output for the system under test. In practice, these observations are corrupted by measurement noise. This requires the introduction of statistical theory and methods, as will be described in detail. Finally, there must also be some definition of what is meant by the word “equivalent” in the preceding definition. There is more than one way in which a model can be considered equivalent to a system under test. The most common approaches will be presented and explained.

The most important requirement for a mathematical model is that it be useful in some way. Sometimes this means that the model can be used to predict some aspect of the behavior of a physical system, while at other times just the values of parameters in the model provide the desired insight. In any case, the synthesized model must be simple enough to be useful and at the same time complex enough to capture important dependencies and features embodied in the observations or measurements.

## 1.1 System Identification Applied to Aircraft

System identification is one of three general problems in aircraft dynamics and control, which can be understood with reference to Fig. 1.1. The three problems are as follows:

- 1) Simulation: given the input  $u$  and the system  $S$ , find the output  $y$ ;
- 2) Control: given the system  $S$  and the output  $y$ , find the input  $u$ ;
- 3) System identification: given the input  $u$  and output  $y$ , find the system  $S$ .

For many applications, an aircraft can be assumed to be a rigid body, whose motion is governed by the laws of Newtonian physics. System identification can be used to characterize applied forces and moments acting on the aircraft that arise from aerodynamics and propulsion. Typically, thrust forces and moments are obtained from ground tests, so aircraft system identification is applied to model the functional dependence of aerodynamic forces and moments on aircraft motion and control variables.

Modern computational methods and wind-tunnel testing can provide, in many instances, comprehensive data about the aerodynamic characteristics of an aircraft. However, there are still several motivations for identifying aircraft models from flight data, including

- 1) Verifying and interpreting theoretical predictions and wind-tunnel test results (flight results can also be used to help improve ground-based predictive methods);
- 2) Obtaining more accurate and comprehensive mathematical models of aircraft dynamics, for use in designing stability augmentation and flight control systems;
- 3) Developing flight simulators, which require accurate representation of the aircraft in all flight regimes (many aircraft motions and flight conditions simply cannot be duplicated in the wind tunnel nor computed analytically with sufficient accuracy or computational efficiency);
- 4) Expanding the flight envelope for new aircraft, which can include quantifying stability and control impact of aircraft modifications, configuration changes, or special flight conditions;
- 5) Verifying aircraft specification compliance.

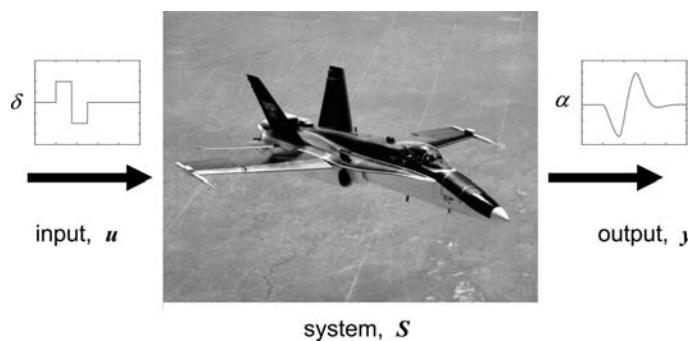


Fig. 1.1 Aircraft dynamics and control.

## INTRODUCTION

3

In the early days of powered flight, only the most basic information about aircraft aerodynamics was obtained from measurements in steady flight. One of the first approaches for obtaining static and dynamic parameters from flight data was given by Milliken<sup>2</sup> in 1947, using frequency response data and a simple semi-graphical method for the analysis. Four years later, Greenberg<sup>3</sup> and Shinbrot<sup>4</sup> established more general and rigorous ways to determine aerodynamic parameters from transient maneuvers. Parameter estimation methods introduced in these reports were based on ordinary and nonlinear least squares.

Dramatic improvement in aircraft aerodynamic modeling techniques came in the late 1960s and early 1970s, because of the availability of digital computers and progress in the new technical discipline known as system identification. There are many papers and books in the technical literature on this subject. Useful starting points would be the survey paper by Åström and Eykhoff<sup>5</sup> and the proceedings from the IFAC Symposia on Identification and System Parameter Estimation. The first of these was held in 1967, and others have followed in three-year intervals. The most relevant textbooks are those by Eykhoff,<sup>6</sup> Goodwin and Payne,<sup>7</sup> Ljung,<sup>8</sup> Söderström and Stoica,<sup>9</sup> Schweppe,<sup>10</sup> Sage and Melsa,<sup>11</sup> Hsia,<sup>12</sup> and Norton.<sup>13</sup> However, none of these textbooks is aimed specifically at aircraft applications, as is this book.

Several authors made substantial contributions in the field of system identification applied to aircraft during the late 1960s and early 1970s, e.g., Taylor et al.,<sup>14</sup> Mehra,<sup>15</sup> Stepner and Mehra,<sup>16</sup> and Gerlach.<sup>17</sup> These contributions were mainly in the area of development and application of various estimation techniques. New challenges to aircraft system identification and parameter estimation were presented by the introduction of highly maneuverable and unstable aircraft. Some of these challenges are addressed by Klein<sup>18</sup> and Klein and Murphy.<sup>19</sup> An extensive bibliography for aircraft parameter estimation was compiled by Iliff and Maine<sup>20</sup> in 1986. Excellent theoretical and practical material on aircraft system identification is given by Maine and Iliff,<sup>21,22</sup> with emphasis on the output-error method. Mulder<sup>23</sup> addressed methods for experiment design, measured data compatibility, and parameter estimation. Broad overviews of aircraft system identification methods can be found in the works by Klein,<sup>24,25</sup> Hamel and Jategaonkar,<sup>26</sup> and the authors of two special issues of the *Journal of Aircraft* on applications of system identification to aircraft.<sup>27,28</sup>

Apart from the contributions of individual authors, the following organizations must be mentioned as well: NASA Dryden Flight Research Center, Langley Research Center, Ames Research Center, and Glenn Research Center, the Army/NASA Rotorcraft Division at NASA Ames Research Center, Deutsches Zentrum für Luft- und Raumfahrt (DLR) in Germany, the Delft University of Technology and the National Aerospace Laboratory (NLR) in the Netherlands, the Royal Aircraft Establishment (RAE) and later the Defense Evaluation and Research Agency (DERA) in the United Kingdom, the Aeronautical Research Laboratory (ARL) in Australia, and the National Research Council (NRC) in Canada. In these establishments, new techniques of system identification have been developed and applied to many different types of aircraft. AIAA and AGARD, and later the NATO Research and Technology Organization (RTO), have also played very constructive roles in the exchange of information and results and in the education of aeronautical engineers and researchers in the area of

## AIRCRAFT SYSTEM IDENTIFICATION

aircraft system identification. References 29–33 are evidence of this, along with numerous papers presented at meetings organized by these agencies.

Based on the development of system identification methodology, it is now possible to determine the structure of aerodynamic model equations and estimate the model parameters involved, along with their confidence intervals, using data from a single flight-test maneuver. If necessary, measurement noise in the output variables can be distinguished from external disturbances to the system caused by wind gusts or modeling errors. The analysis can also include prior knowledge of aircraft aerodynamic model parameters obtained from wind-tunnel measurements and/or previous flight measurements. There are tools for estimating aircraft flying qualities parameters from measured pilot inputs and aircraft responses, and for obtaining more accurate measured data by reconstructing output variables and estimating systematic instrumentation errors, such as biases and scale factor errors. System identification techniques can also be used for experiment design, to maximize information content in the measured data, which leads to more accurate models.

When formulating a system identification problem for aircraft (or any physical system), some general questions must be addressed:

- 1) What are the inputs and outputs?
- 2) How should the data be collected?
- 3) What is a reasonable form for the model to take, given the data and prior knowledge?
- 4) How can the unknown parameters in the model be accurately estimated based on the measured data?
- 5) How good is the identified model?
- 6) How will the results be used?

In the following chapters, theoretical and practical aspects of these questions and their answers for aircraft system identification will be explored in detail.

### 1.2 Outline of the Text

This book presents the requisite theory for the application of system identification methods to aircraft, including MATLAB® software to implement the methods, and some practical examples. An overview of the book is as follows.

In Chapter 2, two elements of system theory, modeling and system identification, are introduced. General mathematical model forms for dynamic systems are briefly reviewed. It is shown that in many practical situations, system identification can be reduced to parameter estimation. This is followed by an overview of aircraft system identification methodology.

Chapter 3 presents aircraft equations of motion and various forms of aircraft mathematical models, with emphasis on the aerodynamic model equations. The form of the aerodynamic model depends on the functions used to model the aerodynamic forces and moments. These functions are typically polynomials, polynomial splines, or indicial functions, computed using aircraft states and controls. Simplified equations of motion are derived, where the simplifications are

## INTRODUCTION

5

brought about by analytical approximation methods or by substituting measured quantities into the nonlinear equations.

In Chapter 4, theory related to parameter estimation and state estimation is outlined. In addition to the linear and nonlinear behavior of a dynamic system, a distinction is made between models that are linear or nonlinear in their parameters. Three models for uncertainties in the parameters and the measurements are introduced. These models then lead to different estimation techniques, distinguished by their optimality criterion.

Chapter 5 deals with various forms of regression as a relatively simple technique for model structure determination and parameter estimation. Least-squares methods and various graphical and analytical diagnostic tools are introduced. The use of stepwise regression and orthogonal function theory for model structure determination is explained and demonstrated. Data partitioning is discussed as a data-handling method, along with techniques for problems of near-linear dependence among model terms, known as data collinearity.

A second group of techniques often used in aircraft parameter estimation is based on maximum likelihood. Two methods are covered in Chapter 6. In the first method, the dynamic system is considered stochastic, due to the presence of noise in the dynamic equations, also called process noise. The process noise can be considered as an input that is not measured, e.g., turbulence. The second method considers the dynamic system as deterministic with measureable inputs. In both cases, the estimation problem is nonlinear in the parameters, regardless of whether the dynamic system model is linear or nonlinear in the states and controls. A method for accurately characterizing parameter estimate uncertainty in the context of aircraft problems is introduced and explained.

The previously mentioned methods for time-domain data, linear regression and maximum likelihood, are formulated in the frequency domain in Chapter 7. As an example application, this chapter includes closed-loop dynamic modeling in the frequency domain, which is useful for flying qualities work. Another application example uses indicial function model forms to identify unsteady aerodynamic effects in the frequency domain.

All the techniques covered through Chapter 7 describe methods based on having all measured data available. These are called batch methods or off-line methods. In Chapter 8, techniques are developed for updating model parameter estimates in real time, as the measured data become available. Previous estimates can be updated at each time step using recursive formulations and the current measured data, without reprocessing old data. Such formulations are called recursive or on-line. Alternatively, least-squares parameter estimation in the time and frequency domains can be applied repeatedly to the most recent data. This approach is called sequential least squares. Five techniques for real-time parameter estimation are discussed: recursive least squares, extended Kalman filter, sequential least squares, modified sequential least squares, and sequential least squares in the frequency domain.

Chapter 9 presents information on instrumentation, data collection, and experiment design. Flight-test instrumentation requirements for aircraft system identification are given, with recommendations and rules of thumb. Input design for system identification flight-test maneuvers is discussed and illustrated, along with recommendations for conducting an experimental flight-test program.

For high-performance, highly augmented aircraft or for stringent experimental conditions, it is more difficult to design inputs for system identification flight-test maneuvers in a heuristic way. Chapter 9 outlines various approaches for such situations, including optimal input design. The chapter also includes a discussion of open-loop parameter estimation when the data are collected with the aircraft operating under closed-loop control.

In Chapter 10, measured data from aircraft sensors is used with the kinematic equations for rigid-body motion to estimate instrumentation systematic errors. After correcting the data for estimated systematic errors, the data should include only random errors. The result of this procedure, called data compatibility analysis, is more accurate data. Several methods for data compatibility analysis are explained. Chapter 10 also discusses corrections for sensor position and alignment that must be applied to the raw measurements.

Some miscellaneous but important data analysis methods are explained in Chapter 11. Most of the techniques discussed could be classified as data processing. The chapter includes material on filtering, smoothing, practical computing methods for the finite Fourier transform and power spectrum estimation, and data visualization.

Computer programs that implement the system identification techniques discussed in the book are described in Chapter 12. The programs are written as MATLAB® m-files<sup>34</sup> and are included with this book. Operation of the programs is explained, and demonstration examples are included for the main routines.

The appendices contain background and supplemental information that is helpful in achieving a complete understanding of the material in the main body of the text. The appendices include mathematical background material (Appendix A); notes on probability, statistics, and random variables (Appendix B); reference information, including material on properties of the atmosphere, basic aerodynamics, aircraft geometry, and mass properties (Appendix C); and a non-linear F-16 simulation in MATLAB® that can be used to simulate flight testing (Appendix D).

Throughout the book, real flight-test data, wind-tunnel data, and the F-16 non-linear simulation are used to demonstrate various applications of system identification methodology. All examples are from work of the authors on aircraft system identification problems at NASA Langley Research Center.

## References

<sup>1</sup>Zadeh, L. A., "From Circuit Theory to System Theory," *Proceedings of the IRE*, Vol. 50, May 1962, pp. 856–865.

<sup>2</sup>Milliken, W. F., Jr., "Progress in Stability and Control Research," *Journal of the Aeronautic Sciences*, Vol. 14, September 1947, pp. 494–519.

<sup>3</sup>Greenberg, H., "A Survey of Methods for Determining Stability Parameters of an Airplane from Dynamic Flight Measurements," NASA TN 2340, 1951.

<sup>4</sup>Shinbrot, M., "A Least Squares Curve Fitting Method with Application of the Calculation of Stability Coefficients from Transient-Response Data," NACA TN 2341, 1951.

<sup>5</sup>Åström, K. J., and Eykhoff, P., "System Identification—A Survey," *Automatica*, Vol. 7, March 1971, pp. 123–162.

## INTRODUCTION

7

<sup>6</sup>Eykhoff, P., *System Identification, Parameter and State Estimation*, Wiley, New York, 1974.

<sup>7</sup>Goodwin, G. C., and Payne, R. L., *Dynamic System Identification: Experiment Design and Data Analysis*, Academic International Press, New York, 1977.

<sup>8</sup>Ljung, L., *System Identification, Theory for the User*, 2nd ed., Prentice-Hall, Upper Saddle River, NJ, 1999.

<sup>9</sup>Söderström, T., and Stoica, P., *System Identification*, Prentice-Hall, Upper Saddle River, NJ, 1989.

<sup>10</sup>Schweppe, F. C., *Uncertain Dynamic Systems*, Prentice-Hall, Upper Saddle River, NJ, 1973.

<sup>11</sup>Sage, A. P., and Melsa, J. L., *System Identification*, Academic International Press, New York, 1971.

<sup>12</sup>Hsia, T. C., *System Identification*, Lexington Books, Lexington, MA, 1977.

<sup>13</sup>Norton, J. P., *An Introduction to Identification*, Academic International Press, London, 1986.

<sup>14</sup>Taylor, L. W., Iliff, K. W., and Powers, B. G., "A Comparison of Newton-Raphson and Other Methods for Determining Stability Derivatives from Flight Data," AIAA Paper 69-315, 1969.

<sup>15</sup>Mehra, R. K., "Maximum Likelihood Identification of Aircraft Parameters," *Proceedings of the Joint Automatic Control Conference*, Atlanta, GA, Paper 18-C, June 1970, pp. 442-444.

<sup>16</sup>Stepner, D. E., and Mehra, R. K., "Maximum Likelihood Identification and Optimal Input Design for Identifying Aircraft Stability and Control Derivatives," NASA CR-2200, 1973.

<sup>17</sup>Gerlach, O. H., "The Determination of Stability Derivatives and Performance Characteristics from Dynamic Maneuvers," Society of Automotive Engineers, Paper 700236, 1970.

<sup>18</sup>Klein, V., "Application of System Identification to High Performance Aircraft," *Proceedings of the 32nd IEEE Conference on Decision and Control*, San Antonio, TX, December 1993, pp. 2253-2259.

<sup>19</sup>Klein, V., and Murphy, P. C., "Aerodynamic Parameters of High Performance Aircraft Estimated from Wind Tunnel and Flight Test Data," *System Identification for Integrated Aircraft Development and Flight Testing*, NATO Res. and Techn. Org., RTO-MP-11, Paper 18, May 1999.

<sup>20</sup>Iliff, K. W., and Maine, R. E., "Bibliography for Aircraft Parameter Estimation," NASA TM 86804, 1986.

<sup>21</sup>Maine, R. E., and Iliff, K. W., "Identification of Dynamic Systems, Theory and Formulation," NASA RP 1138, 1985.

<sup>22</sup>Maine, R. E., and Iliff, K. W., "Application of Parameter Estimation to Aircraft Stability and Control, The Output Error Approach," NASA RP 1168, 1986.

<sup>23</sup>Mulder, J. A., "Design and Evaluation of Dynamic Flight Test Manoeuvres," Delft Univ. of Technology, Dept. of Aerospace Engineering, Report LR-497, Delft, The Netherlands, 1986.

<sup>24</sup>Klein, V., "Estimation of Aircraft Aerodynamic Parameters from Flight Data," *Progress in Aerospace Sciences*, Vol. 26, No. 1, 1989, pp. 1-77.

<sup>25</sup>Klein, V., "Identification Evaluation Methods," *Parameter Identification*, AGARD-LS-104, Paper 2, 1972b.

<sup>26</sup>Hamel, P. G., and Jategaonkar, R., "Evolution of Flight Vehicle System Identification," *Journal of Aircraft*, Vol. 33, Jan.-Feb., 1996, pp. 9-28.

## AIRCRAFT SYSTEM IDENTIFICATION

<sup>27</sup>*Journal of Aircraft*, Vol. 41, No. 4, July–Aug., 2004.

<sup>28</sup>*Journal of Aircraft*, Vol. 42, No. 1, Jan.–Feb., 2005.

<sup>29</sup>AGARD-CP-172, *Methods for Aircraft State and Parameter Identification*, 1975.

<sup>30</sup>*Parameter Identification*, AGARD-LS-104, 1979.

<sup>31</sup>*Dynamic Stability Parameters*, AGARD-LS-114, 1981.

<sup>32</sup>*Rotorcraft System Identification*, AGARD-LS-178, 1991.

<sup>33</sup>*System Identification for Integrated Aircraft Development and Flight Testing*, NATO Res. and Techn. Org., RTO-MP-11, May 1999.

<sup>34</sup>*Getting Started with MATLAB, Version 7*, The MathWorks, Natick, MA, 2005.

**2****Elements of System Theory**

The content of this book is closely related to system theory, a scientific discipline devoted to the study of mathematical properties of various physical systems. System theory has become an extensive and rapidly growing field that overlaps and interrelates with other fields of study, such as information theory, signal theory, stability and control theory, and others. There are many problems and areas covered by system theory, including mathematical modeling, system identification, dynamic system analysis, control system synthesis, and optimization, among others. In this chapter, two elements of system theory, mathematical modeling and system identification, will be discussed, with an emphasis on application to aircraft.

## **2.1 Mathematical Modeling**

Mathematical modeling is the process of developing an adequate mathematical representation of some aspects of a physical system. Mathematical models can take various forms. One form that often results from direct application of physical laws is a set of differential equations relating input to output. An equivalent and more preferable form of the model is the state-space representation, which relates three variables: input, output, and state. The input  $u$ , output  $y$ , and state  $x$  can be vector or scalar quantities.

The input excites the system and can usually be specified by the experimenter. It is therefore an external disturbance that can be directly measured. The input must be distinguished from unmeasured disturbances, which are observed only through their influence on the system response. In some cases, the experimenter may not have the capability to specify the input, so that inputs for normal operation of the system must be used.

The output is an observable signal indicating the system response to the input and disturbances. Typically, the output at any time  $t$  is a function of the current state and input. The state is a variable that completely specifies the status of the system at any given time. The state is not unique. For example, a change in coordinate system would give a different but equivalent state. For deterministic physical systems, knowledge of the state at time  $t_0$ , combined with knowledge of the system dynamics and inputs from time  $t_0$  to time  $t \geq t_0$ , is sufficient to compute the state at time  $t$ . The state at time  $t$  reflects what has happened to

the system up to time  $t$ , and can be used at time  $t$  in lieu of the system history. Mathematically, this can be expressed as

$$\mathbf{x}(t) = \varphi\{\mathbf{x}(t_0), \mathbf{u}_{[t_0, t]}\} \quad (2.1)$$

$$\mathbf{y}(t) = \mathbf{h}[\mathbf{x}(t), \mathbf{u}(t)] \quad (2.2)$$

where  $\mathbf{u}_{[t_0, t]}$  denotes the input  $\mathbf{u}$  over the time interval  $[t_0, t]$ . Equations (2.1) and (2.2) can represent the solution of model equations for the behavior of a dynamic system, as will be shown.

The principal categorizations of the various models for representing a physical system are linear or nonlinear, time invariant or time varying, continuous time or discrete time, and deterministic or stochastic. These models belong to the class of parametric models. They are finite dimensional, and can be implemented as state-space equations, differential equations, and transfer functions, among other forms. Nonparametric models, on the other hand, do not require explicit specification of the system dimension. They are inherently infinite dimensional, and can be implemented in the form of impulse or step responses, frequency responses, correlation functions, or spectral densities, among other forms.

A short review of parametric and nonparametric models often used in the mathematical representation of physical systems is given next. The models are categorized as linear or nonlinear.

### 2.1.1 Linear Models

The state-space continuous-time representation of a linear, time-invariant deterministic system can be formulated as

$$\dot{\mathbf{x}}(t) = \mathbf{Ax}(t) + \mathbf{Bu}(t) \quad \mathbf{x}(0) = \mathbf{x}_0 \quad (2.3)$$

$$\mathbf{y}(t) = \mathbf{Cx}(t) + \mathbf{Du}(t) \quad (2.4)$$

where  $\mathbf{A}$  is the stability (or system) matrix,  $\mathbf{B}$  is the control (or input) matrix,  $\mathbf{C}$  and  $\mathbf{D}$  are output transformation matrices, and  $\mathbf{x}_0$  is a vector of initial conditions for the state. The time-invariant adjective applies to the matrices  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$ , and  $\mathbf{D}$ , which contain constant elements. In general, the input  $\mathbf{u} \in \mathbb{R}^{n_u}$ , output  $\mathbf{y} \in \mathbb{R}^{n_o}$ , and state  $\mathbf{x} \in \mathbb{R}^{n_s}$  vary with time. The time scale is defined so that the initial time  $t_0$  is zero. This will be done throughout the book.

The state equation for a scalar system with zero input (also called an autonomous or homogeneous system) reduces to

$$\dot{x}(t) = ax(t) \quad x(0) = x_0 \quad (2.5)$$

The solution can be found by separation of variables,

$$\int_{x_0}^{x(t)} \frac{dx}{x} = \int_0^t a dt \quad (2.6)$$

$$x(t) = e^{at}x_0 \quad (2.7)$$

## ELEMENTS OF SYSTEM THEORY

11

The homogeneous form of vector equation (2.3) is

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) \quad \mathbf{x}(0) = \mathbf{x}_0 \quad (2.8)$$

By analogy to the scalar case, the proposed solution is

$$\mathbf{x}(t) = e^{\mathbf{A}t} \mathbf{x}_0 \quad (2.9)$$

where the matrix  $e^{\mathbf{A}t}$  is defined by the infinite series

$$e^{\mathbf{A}t} = \mathbf{I} + \mathbf{A}t + \frac{\mathbf{A}^2 t^2}{2!} + \frac{\mathbf{A}^3 t^3}{3!} + \dots \quad (2.10)$$

The time derivative of  $e^{\mathbf{A}t}$  is

$$\frac{d}{dt}(e^{\mathbf{A}t}) = \mathbf{A} + \mathbf{A}^2 t + \frac{\mathbf{A}^3 t^2}{2!} + \dots = \mathbf{A} e^{\mathbf{A}t} \quad (2.11)$$

It follows that the proposed solution in Eq. (2.9) satisfies Eq. (2.8). Since at the initial time  $t = 0$ , the state vector must equal  $\mathbf{x}_0$ , Eq. (2.9) also satisfies the initial condition, and therefore is the unique solution of the homogeneous Eq. (2.8).

Using Eq. (2.9) and assuming the state vector contains  $n_s$  elements, the solutions of Eq. (2.8) for the  $n_s$  initial condition vectors

$$\mathbf{x}(0) = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad \mathbf{x}(0) = \begin{bmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad \mathbf{x}(0) = \begin{bmatrix} 0 \\ 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix}, \quad \dots, \quad \mathbf{x}(0) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} \quad (2.12)$$

are the columns of the matrix  $e^{\mathbf{A}t}$ . The matrix  $e^{\mathbf{A}t}$  is therefore analogous to the impulse response function for a scalar ordinary differential equation. Each solution for the initial conditions in Eq. (2.12) is a vector function of time, and the collection of these  $n_s$  vectors make up the  $n_s \times n_s$  matrix  $e^{\mathbf{A}t}$ . The matrix  $e^{\mathbf{A}t}$  is one example of a general type of matrix called a state transition matrix, since multiplying the state at initial time  $t = 0$  by the state transition matrix  $e^{\mathbf{A}t} = e^{\mathbf{A}(t-0)}$  brings about a transition to the state at time  $t$ .

For the general linear state equation (2.3), the forcing function  $\mathbf{B}\mathbf{u}(t)$  can be considered a sequence of impulses, the response to which can be computed using the impulse response matrix  $e^{\mathbf{A}t}$  inside a convolution integral to implement the superposition property of the linear system,

$$\mathbf{x}(t) = e^{\mathbf{A}t} \mathbf{x}_0 + \int_0^t e^{\mathbf{A}(t-\tau)} \mathbf{B} \mathbf{u}(\tau) d\tau \quad (2.13)$$

Equations (2.13) and (2.4) are equivalent in form to Eqs. (2.1) and (2.2). The first term on the right-hand side of Eq. (2.13) is the free response due to the initial

conditions, and the convolution integral term is the forced response due to the input  $\mathbf{u}(t)$ . Note that the convolution integral uses  $e^{A(t-\tau)}$  to compute the contribution to  $\mathbf{x}(t)$  due to an impulse  $\mathbf{B}\mathbf{u}(\tau)$  at time  $\tau \leq t$ . Substituting Eq. (2.13) into Eq. (2.4), the output can be written as

$$\mathbf{y}(t) = \mathbf{C} \left[ e^{At} \mathbf{x}_0 + \int_0^t e^{A(t-\tau)} \mathbf{B} \mathbf{u}(\tau) d\tau \right] + \mathbf{D} \mathbf{u}(t) \quad (2.14)$$

Now define a matrix  $\mathbf{G}(t)$  such that

$$\mathbf{G}(t) = \mathbf{C} e^{At} \mathbf{B} + \mathbf{D} \delta(t) \quad (2.15)$$

where  $\delta(t)$  is the impulse function or Dirac delta function, with the properties

$$\delta(t) = \begin{cases} 0 & \text{for } t \neq 0 \\ \infty & \text{for } t = 0 \end{cases} \quad \text{and} \quad \int_{-\infty}^{\infty} \delta(t) dt = 1 \quad (2.16)$$

Then, the expression for the output in Eq. (2.14) can be written as

$$\mathbf{y}(t) = \mathbf{C} e^{At} \mathbf{x}_0 + \int_0^t \mathbf{G}(t-\tau) \mathbf{u}(\tau) d\tau \quad (2.17)$$

where  $\mathbf{G}(t)$  is called a weighting function matrix.

When the convolution integral in Eq. (2.13) is taken over a small time step  $\Delta t$  for which the input vector  $\mathbf{u}(t)$  can be considered constant,

$$\begin{aligned} \mathbf{x}(\Delta t) &= e^{A \Delta t} \mathbf{x}(0) + e^{A \Delta t} \int_0^{\Delta t} e^{-A \tau} d\tau \mathbf{B} \mathbf{u}(0) \\ &= e^{A \Delta t} \mathbf{x}(0) + e^{A \Delta t} [\mathbf{A}^{-1} (\mathbf{I} - e^{-A \Delta t})] \mathbf{B} \mathbf{u}(0) \\ &= e^{A \Delta t} \mathbf{x}(0) + [\mathbf{A}^{-1} (e^{A \Delta t} - \mathbf{I})] \mathbf{B} \mathbf{u}(0) \end{aligned} \quad (2.18)$$

The same calculation for a single time step  $\Delta t$  can be based at any other starting time  $t$ , instead of at  $t = 0$ , as in Eq. (2.18). In discrete-time notation,

$$\mathbf{x}(i) = e^{A \Delta t} \mathbf{x}(i-1) + [\mathbf{A}^{-1} (e^{A \Delta t} - \mathbf{I})] \mathbf{B} \mathbf{u}(i-1) \quad (2.19)$$

where  $\mathbf{x}(i) \equiv \mathbf{x}(i\Delta t)$ , etc., for nonnegative integers  $i$ . Equivalently,

$$\mathbf{x}(i) = \boldsymbol{\Phi} \mathbf{x}(i-1) + \boldsymbol{\Gamma} \mathbf{u}(i-1) \quad (2.20)$$

where

$$\boldsymbol{\Phi} \equiv e^{A \Delta t} \quad \boldsymbol{\Gamma} \equiv [\mathbf{A}^{-1} (e^{A \Delta t} - \mathbf{I})] \mathbf{B} \quad (2.21)$$

## ELEMENTS OF SYSTEM THEORY

13

Equation (2.21) shows the connection between the continuous-time system and control matrices and the discrete form of these matrices.

Returning to the homogeneous linear vector differential equation (2.8), consider the candidate solution

$$\mathbf{x}(t) = \boldsymbol{\xi} e^{\lambda t} \quad (2.22)$$

where  $\lambda$  is a scalar and  $\boldsymbol{\xi}$  is a vector. Substituting into Eq. (2.8) gives

$$\lambda \boldsymbol{\xi} e^{\lambda t} = \mathbf{A} \boldsymbol{\xi} e^{\lambda t} \quad (2.23)$$

or

$$(\lambda \mathbf{I} - \mathbf{A}) \boldsymbol{\xi} = \mathbf{0} \quad (2.24)$$

A nonzero solution to this set of algebraic equations exists if the determinant of the coefficient matrix equals zero,

$$|\lambda \mathbf{I} - \mathbf{A}| = 0 \quad (2.25)$$

The determinant in Eq. (2.25) is called the characteristic determinant, and the roots of Eq. (2.25),  $\lambda_1, \lambda_2, \dots, \lambda_{n_s}$  for an  $n_s$ -dimensional state vector  $\mathbf{x}(t)$ , are called the eigenvalues of the  $n_s \times n_s$  matrix  $\mathbf{A}$ . If the eigenvalues are distinct, each eigenvalue  $\lambda_i$ ,  $i = 1, 2, \dots, n_s$ , has a corresponding eigenvector  $\boldsymbol{\xi}_i$ , which is found by solving Eq. (2.24) with  $\lambda = \lambda_i$ . The solution corresponding to each real eigenvalue or to each complex pair of eigenvalues is called a mode. The solution of the homogeneous equation (2.8) is then a sum of the modal components,

$$\mathbf{x}(t) = c_1 \boldsymbol{\xi}_1 e^{\lambda_1 t} + c_2 \boldsymbol{\xi}_2 e^{\lambda_2 t} + \dots + c_{n_s} \boldsymbol{\xi}_{n_s} e^{\lambda_{n_s} t} \quad (2.26)$$

with the values of the scalars  $c_i$ ,  $i = 1, 2, \dots, n_s$ , determined by the requirement that the initial condition  $\mathbf{x}(0) = \mathbf{x}_0$  be satisfied. This means that each  $c_i$ ,  $i = 1, 2, \dots, n_s$ , is equal to the projection of the initial condition vector  $\mathbf{x}_0$  along the direction of the corresponding eigenvector.

If the forcing function  $\mathbf{B} \mathbf{u}(t)$  is again considered as a sequence of impulses, a similar analysis applies for the forcing function  $\mathbf{B} \mathbf{u}(\tau)$  inside the convolution integral of Eq. (2.13), so that the participation of each mode in the forced solution depends on the projection of the forcing function along the eigenvectors.

It follows from the preceding discussion and Eq. (2.26) that the eigenvalues and eigenvectors are important properties of the linear dynamic system, providing information about response and stability. In general, the eigenvalues characterize the type of response associated with each mode, and the eigenvectors define the characteristic directions in state space for exciting particular modal responses.

Applying the Laplace transform (see Appendix A) to Eqs. (2.3) and (2.4) for  $\mathbf{x}(0) = \mathbf{0}$  gives

$$s\tilde{\mathbf{x}}(s) = \mathbf{A}\tilde{\mathbf{x}}(s) + \mathbf{B}\tilde{\mathbf{u}}(s) \quad (2.27)$$

$$\tilde{\mathbf{y}}(s) = \mathbf{C}\tilde{\mathbf{x}}(s) + \mathbf{D}\tilde{\mathbf{u}}(s) \quad (2.28)$$

where  $s$  is a complex variable. The transformed state is

$$\tilde{\mathbf{x}}(s) = \int_0^{\infty} \mathbf{x}(t) e^{-st} dt \quad (2.29)$$

and similarly for  $\tilde{\mathbf{y}}(s)$  and  $\tilde{\mathbf{u}}(s)$ . From Eqs. (2.15) and (2.17), and the fact that convolution in the time domain is equivalent to multiplication in the Laplace domain (see Appendix A),

$$\tilde{\mathbf{y}}(s) = [\mathbf{C}(sI - \mathbf{A})^{-1} \mathbf{B} + \mathbf{D}] \tilde{\mathbf{u}}(s) = \mathbf{G}(s) \tilde{\mathbf{u}}(s) \quad (2.30)$$

Matrix  $\mathbf{G}(s)$  is the transfer function matrix with elements

$$[G_{jk}] = \frac{\text{num}_{jk}(s)}{\text{den}(s)} \quad \text{for } \begin{cases} j = 1, 2, \dots, n_o \\ k = 1, 2, \dots, n_i \end{cases} \quad (2.31)$$

where  $n_o$  and  $n_i$  are the number of outputs and inputs, respectively. The quantities  $\text{num}_{jk}(s)$  and  $\text{den}(s)$  are both polynomials in  $s$ . The numerator polynomial  $\text{num}_{jk}(s)$  corresponds to the transfer function of the  $j$ th output to the  $k$ th input, and the denominator polynomial  $\text{den}(s)$  is the characteristic polynomial in  $s$ , which is the same for all elements of the  $\mathbf{G}(s)$  matrix.

Setting  $s = j\omega$ , where  $j = \sqrt{-1}$  and  $\omega$  is the angular frequency, the Laplace transform becomes the Fourier transform,

$$\begin{aligned} \tilde{\mathbf{y}}(j\omega) &= \int_0^{\infty} \mathbf{y}(t) e^{-j\omega t} dt \\ \tilde{\mathbf{u}}(j\omega) &= \int_0^{\infty} \mathbf{u}(t) e^{-j\omega t} dt \end{aligned} \quad (2.32)$$

and the transfer function matrix becomes the frequency response matrix  $\mathbf{G}(j\omega)$ . This quantity can be determined experimentally, as a function of frequency. The result is a nonparametric model in the frequency domain. This model type is discussed in Chapter 7.

Elements of the matrices  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$ , and  $\mathbf{D}$  in Eqs. (2.3) and (2.4) are constant model parameters. These parameters do not depend on input  $\mathbf{u}$ , state  $\mathbf{x}$ , their derivatives, or time  $t$ . When the model parameters are time varying, Eqs. (2.3)

## ELEMENTS OF SYSTEM THEORY

15

and (2.4) change to

$$\dot{\mathbf{x}}(t) = \mathbf{A}(t)\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t) \quad \mathbf{x}(t_0) = \mathbf{x}_0 \quad (2.33)$$

$$\mathbf{y}(t) = \mathbf{C}(t)\mathbf{x}(t) + \mathbf{D}(t)\mathbf{u}(t) \quad (2.34)$$

where the time scale cannot be changed so that  $t_0 = 0$ , because the model depends explicitly on time. These equations represent a time-varying system. The solution is

$$\mathbf{y}(t) = \mathbf{C}(t)\Phi(t, t_0)\mathbf{x}(t_0) + \int_{t_0}^t \mathbf{C}(t)\Phi(t, \tau)\mathbf{B}(\tau)\mathbf{u}(\tau)d\tau + \mathbf{D}(t)\mathbf{u}(t) \quad (2.35)$$

In this case, the state transition matrix  $\Phi$  is a function of two variables: the time of application of the cause  $\tau$ , and the time of observation of the effect  $t$ . The solution given by Eq. (2.35) involves a superposition integral and not a convolution integral as in Eq. (2.14). More detailed discussion of the state-space representation of a dynamic system can be found in Ref. 1.

Adding uncertain input disturbances to Eq. (2.33), the previously deterministic model is changed to a stochastic model,

$$\dot{\mathbf{x}}(t) = \mathbf{A}(t)\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t) + \mathbf{B}_w(t)\mathbf{w}(t) \quad (2.36)$$

$$\mathbf{y}(t) = \mathbf{C}(t)\mathbf{x}(t) + \mathbf{D}(t)\mathbf{u}(t) \quad (2.37)$$

where the vector  $\mathbf{w}(t)$  is usually called process noise and  $\mathbf{B}_w(t)$  is the control matrix associated with  $\mathbf{w}(t)$ . In many applications, it is assumed that  $\mathbf{w}(t)$  is a white noise process specified by its mean and covariance matrix:

$$E[\mathbf{w}(t)] = \mathbf{0}$$

$$E[\mathbf{w}(t_i)\mathbf{w}^T(t_j)] = \mathbf{Q}(t_i)\delta(t_i - t_j)$$

where  $E[ ]$  is the expectation operator (see Appendix B) and  $\delta(t_i - t_j)$  is the Dirac delta function. To complete the modeling of the stochastic system represented by Eqs. (2.36) and (2.37), it is necessary to specify the vector of initial conditions and define the interrelation between  $\mathbf{x}(t_0)$  and  $\mathbf{w}(t)$ , usually as

$$E[\mathbf{x}(t_0)] = \bar{\mathbf{x}}_0$$

$$E\{[\mathbf{x}(t_0) - \bar{\mathbf{x}}_0][\mathbf{x}(t_0) - \bar{\mathbf{x}}_0]^T\} = \mathbf{P}_0$$

$$E[\mathbf{x}(t_0)\mathbf{w}^T(t)] = \mathbf{0}$$

where  $\mathbf{P}_0$  is a constant  $n_s \times n_s$  error covariance matrix for the initial state vector. The symbol  $\bar{\mathbf{x}}_0$  is used because the initial condition is the expected value of the stochastic quantity  $\mathbf{x}(t_0)$ .

Using the continuous-time formulation of a stochastic system creates some problems with the properties of  $\mathbf{w}(t)$ . For a fixed length of time,  $\mathbf{w}(t)$  is a zero

mean random vector with infinite power, because the white noise power spectrum is theoretically a constant over an infinite frequency range. This means that  $\mathbf{w}(t)$  does not exist in any physical sense.

To avoid this difficulty, it is preferable to formulate the model for a stochastic system in discrete-time form. The signals in this model are sampled at  $t_0 + i\Delta t$ ,  $i = 0, 1, 2, \dots$ , with a constant time increment  $\Delta t$  between samples. Then, using the shorthand notation

$$\mathbf{x}(i) \equiv \mathbf{x}(t_0 + i\Delta t) \quad i = 0, 1, 2, \dots \quad (2.38)$$

and similarly for  $\mathbf{u}(i)$  and  $\mathbf{w}(i)$ , the discrete-time stochastic state-space model is given by

$$\mathbf{x}(i) = \Phi(i-1)\mathbf{x}(i-1) + \Gamma(i-1)\mathbf{u}(i-1) + \Gamma_w(i-1)\mathbf{w}(i-1) \quad (2.39)$$

$$\mathbf{y}(i) = \mathbf{C}(i)\mathbf{x}(i) + \mathbf{D}(i)\mathbf{u}(i) \quad i = 1, 2, \dots \quad (2.40)$$

with

$$E[\mathbf{x}(0)] = \bar{\mathbf{x}}_0$$

$$E[\mathbf{w}(i)] = \mathbf{0}$$

$$E[\mathbf{w}(i)\mathbf{w}^T(j)] = (\Delta t)^{-1}\mathbf{Q}(i)\delta_{ij}$$

where  $\delta_{ij}$  equals 1 for  $i = j$ , and zero for  $i \neq j$ .

The solution to Eqs. (2.39) and (2.40) is discussed in Chapter 4 in connection with state estimation.

### 2.1.2 Nonlinear Models

Most real-world systems are nonlinear. If these systems operate over a restricted range of conditions, then linear models can be used to approximate the nonlinear behavior. When such an approximation is not possible, a suitable nonlinear model must be postulated. For a stochastic, time-varying system, the model equations take the form

$$\dot{\mathbf{x}}(t) = \mathbf{f}[\mathbf{x}(t), \mathbf{u}(t), \mathbf{w}(t), t] \quad (2.41)$$

$$\mathbf{y}(t) = \mathbf{h}[\mathbf{x}(t), \mathbf{u}(t), t] \quad (2.42)$$

As for a linear system, these equations must be augmented with a model for  $\mathbf{x}(t_0)$  and  $\mathbf{w}(t)$ . In the case of a deterministic, time-invariant system, Eqs. (2.41) and (2.42) are simplified to

$$\dot{\mathbf{x}}(t) = \mathbf{f}[\mathbf{x}(t), \mathbf{u}(t)] \quad \mathbf{x}(0) = \mathbf{x}_0 \quad (2.43)$$

$$\mathbf{y}(t) = \mathbf{h}[\mathbf{x}(t), \mathbf{u}(t)] \quad (2.44)$$

In general, the solution of nonlinear differential equations must be computed using numerical methods such as fourth-order Runge-Kutta (e.g., see Appendix D).

## ELEMENTS OF SYSTEM THEORY

17

For a wide class of nonlinear systems, a general nonparametric representation is the Volterra series.<sup>2</sup> The general form of the Volterra series for a scalar system can be expressed as

$$y(t) = \sum_{n=1}^{\infty} y_n(t) \quad (2.45)$$

where  $y_n(t)$  is the  $n$ th functional defined as

$$\begin{aligned} y_1(t) &= \int_0^t g_1(\tau_1) u(t - \tau_1) d\tau_1 \\ y_2(t) &= \int_0^t \int_0^t g_2(\tau_1, \tau_2) u(t - \tau_1) u(t - \tau_2) d\tau_1 d\tau_2 \\ &\vdots \\ y_n(t) &= \int_0^t \int_0^t \cdots \int_0^t g_n(\tau_1, \tau_2, \dots, \tau_n) u(t - \tau_1) \cdots u(t - \tau_n) d\tau_1 \cdots d\tau_n \end{aligned} \quad (2.46)$$

The terms

$$g_1(\tau_1), g_2(\tau_1, \tau_2), \dots, g_n(\tau_1, \tau_2, \dots, \tau_n)$$

are called impulse responses or kernels. The description using the Volterra series is therefore a direct generalization of the model for a linear system using a convolution integral. Practical use of the Volterra series for nonlinear modeling requires truncating the infinite series, usually after the first two or three terms.

More about mathematical models for dynamic systems can be found in Refs. 1–3.

### 2.2 System Identification and Parameter Estimation

System identification is the determination of a mathematical model from measured input-output data. The system identification problem is characterized by the selected inputs and outputs, the class of models from which the model will be chosen, and a criterion for equivalence of the model and the physical system. This is consistent with the definition of system identification quoted in Chapter 1.

System identification is preceded by an experiment where the inputs typically are specified, taking into account *a priori* knowledge about the system and the purpose of the identification. The class of models entering system identification procedures is also selected based on *a priori* knowledge and the purpose of the identification, but also on data from the experiment.

It is common practice to express the equivalence of the model and the physical system in terms of a scalar cost function that quantifies the equivalence of the observed physical system output  $z$  and the model output  $y$ ,

$$J = J(z, y) \quad (2.47)$$

Normally, the cost function  $J$  consists of a weighted sum of squared differences between  $z$  and  $y$ .

The relation between the variables  $z$  and  $y$  is expressed by the measurement equation,

$$z = y + \nu \quad (2.48)$$

where  $\nu$  is the measurement error, also called the measurement noise, which is assumed to be random. The measurement noise cannot be measured directly, so its properties are assumed. Measurement noise properties can also be estimated from measured data using filtering and smoothing techniques explained in Chapter 11.

Finding the best model based on the cost function in Eq. (2.47) could lead to the investigation of a large number of model candidates. To simplify the problem, system identification is changed to an optimization problem that requires finding a model  $\mathcal{M}$ , selected from a class of models, such that  $\mathcal{M}$  minimizes the cost function. In many situations, a class of models can be limited to the models  $\mathcal{M}^*$  that have the same structure (mathematical form), but are distinguished by different values of the parameters in the model,  $\theta$ ,

$$\mathcal{M}^* = \{\mathcal{M}(\theta)\} \quad (2.49)$$

Data collected from the experiment is denoted by

$$\mathbf{Z}_N = [z(1), \mathbf{u}(1), z(2), \mathbf{u}(2), \dots, z(N), \mathbf{u}(N)] \quad (2.50)$$

where  $N$  is the number of data points. Once the model structure is selected, system identification becomes the selection of a value for  $\theta$ , based on the information in  $\mathbf{Z}_N$  that minimizes a scalar cost function

$$J = J[\mathbf{Z}_N, \mathbf{Y}_N(\theta)] \quad (2.51)$$

where  $\mathbf{Y}_N(\theta)$  represents the model outputs,

$$\mathbf{Y}_N(\theta) = [y(1), y(2), \dots, y(N)] \quad (2.52)$$

which depend on the parameter vector  $\theta$ . Thus, system identification is reduced to model parameter estimation. Such a formulation makes it possible to exploit methods of statistical inference, mainly estimation theory. The selection of an estimation method will be influenced by assumptions made for measurement noise and whether the parameters are assumed to be random variables or unknown constants. Three principal estimation techniques will be explained in Chapter 4. Important references for the topic of parameter estimation are the books by Sorenson,<sup>3</sup> Eykhoff,<sup>4</sup> Ljung,<sup>5</sup> and Schweppe.<sup>6</sup>

Results from the parameter estimation process should include parameter estimates and their properties, including error bounds, and information for testing

## ELEMENTS OF SYSTEM THEORY

19

various statistical hypotheses. Some additional questions might arise during the identification process, such as the following:

- 1) Is the information content in the data sufficient to distinguish among different models?
- 2) Is the model formulated so that unique parameter values can be found?
- 3) Do the estimated parameters have physically realistic values and small error bounds?

These questions are related to the concept of identifiability, which is central to the identification procedure. Identifiability is discussed in a rigorous way in Ref. 5. In this book, it will be mentioned mainly in connection with experiment design, model structure determination, and in examples.

### 2.3 Aircraft System Identification

When system identification is applied to an aircraft, the equations governing the aircraft dynamic motion are postulated and an experiment is designed to obtain measurements of input and output variables. The equations of motion for an aircraft come from the translational and rotational forms of Newton's second law of motion. Chapter 3 contains a detailed derivation of these equations, which describe the translational motion of the aircraft center of gravity and the rotational motion about the center of gravity. In vector form, the equations are

$$m\dot{V} + \boldsymbol{\omega} \times mV = \mathbf{F}_G(\boldsymbol{\zeta}) + \mathbf{F}_T + \mathbf{F}_A(V, \boldsymbol{\omega}, \mathbf{u}, \boldsymbol{\theta}) \quad (2.53)$$

$$\mathbf{I}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{I}\boldsymbol{\omega} = \mathbf{M}_T + \mathbf{M}_A(V, \boldsymbol{\omega}, \mathbf{u}, \boldsymbol{\theta}) \quad (2.54)$$

where  $m$  is the aircraft mass,  $\mathbf{I}$  is the inertia tensor,  $\boldsymbol{\zeta}$  are Euler angles indicating the attitude of the aircraft relative to fixed earth axes,  $V$  and  $\boldsymbol{\omega}$  are translational and angular velocity vectors for the aircraft motion, and  $\mathbf{u}$  is the control vector. Applied forces in these equations come from gravity ( $\mathbf{F}_G$ ), thrust ( $\mathbf{F}_T$ ), and aerodynamics ( $\mathbf{F}_A$ ). Applied moments are generated by thrust ( $\mathbf{M}_T$ ) and aerodynamics ( $\mathbf{M}_A$ ). The gravity force is modeled by adding kinematic differential equations to describe the aircraft attitude relative to earth axes, assuming a constant gravitational acceleration vector. Usually, the applied forces and moments due to thrust are modeled using results from engine tests done on the ground and the geometry of the engine installation. Aircraft system identification then reduces to the determination of a model structure for the aerodynamic forces ( $\mathbf{F}_A$ ) and moments ( $\mathbf{M}_A$ ), and the estimation of unknown parameters in those model structures, based on measured data. The quantity  $\boldsymbol{\theta}$  is a vector of parameters that in the current formulation specifies aerodynamic characteristics of the aircraft. The model structures identified for  $\mathbf{F}_A$  and  $\mathbf{M}_A$  are referred to as the aerodynamic model equations.

The dynamic equations (2.53) and (2.54) are augmented with output equations that specify the connection of aircraft states and controls to measured outputs, along with measurement equations describing the measurement process. The

complete set of all these equations can be written as

$$\dot{\mathbf{x}}(t) = \mathbf{f}[\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\theta}] \quad \mathbf{x}(0) = \mathbf{x}_0 \quad (2.55)$$

$$\mathbf{y}(t) = \mathbf{h}[\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\theta}] \quad (2.56)$$

$$z(i) = \mathbf{y}(i) + \mathbf{v}(i) \quad i = 1, 2, \dots, N \quad (2.57)$$

where  $\mathbf{x}$  is composed of  $\mathbf{V}$ ,  $\boldsymbol{\omega}$ , and  $\boldsymbol{\zeta}$ . The control vector  $\mathbf{u}$  is generally composed of throttle position and control surface deflections. Elements of the output vector  $\mathbf{y}$  are aircraft response variables, which usually include state variables. Discrete measured outputs  $z(i)$  are corrupted by measurement noise  $\mathbf{v}(i)$ .

Aircraft system identification can be defined as the determination, from input and output measurements, of a model structure for  $\mathbf{F}_A$  and  $\mathbf{M}_A$ , and the estimation of the unknown parameters  $\boldsymbol{\theta}$  contained in these model structures. In many practical applications, the structure of the models for the aerodynamic forces and moments is assumed to be known, and the system identification problem reduces to parameter estimation. The most common situation is that the aerodynamic forces and moments depend linearly on current values of the states and controls, leading to linear time-invariant aerodynamic model formulations. Chapter 3 details the various forms of the models for aerodynamic forces and moments.

In the literature, the process properly known as parameter estimation is often referred to as parameter identification, or an abbreviation of that term, PID. It is also common to see the term “system identification” applied to a process that is really parameter estimation.

Aircraft system identification includes model postulation, experiment design, data compatibility analysis, model structure determination, parameter and state estimation, collinearity diagnostics, and model validation. These steps are necessary to identify a mathematical description of the functional dependence of the applied aerodynamic forces and moments on aircraft motion and control variables. A block diagram depicting the general approach to aircraft system identification is shown in Fig. 2.1. Each of the steps in the procedure will be described briefly.

**Model postulation.** Model postulation is based on *a priori* knowledge about the aircraft dynamics and aerodynamics. The postulated model influences the type of flight-test maneuver used for system identification. It is common practice to express the aerodynamic forces and moments in terms of linear expansions, polynomials, or polynomial spline functions in the states and controls, with time-invariant parameters quantifying the contribution of each to the total aerodynamic force or moment. In recent years, this formulation has been extended to cases with unsteady aerodynamic effects modeled by indicial functions or additional state equations. Formulations of the aerodynamic model and various forms of the equations of motion are covered in Chapter 3.

**Experiment design.** Experiment design includes selection of an instrumentation system, and specification of the aircraft configuration, flight conditions, and

## ELEMENTS OF SYSTEM THEORY

21

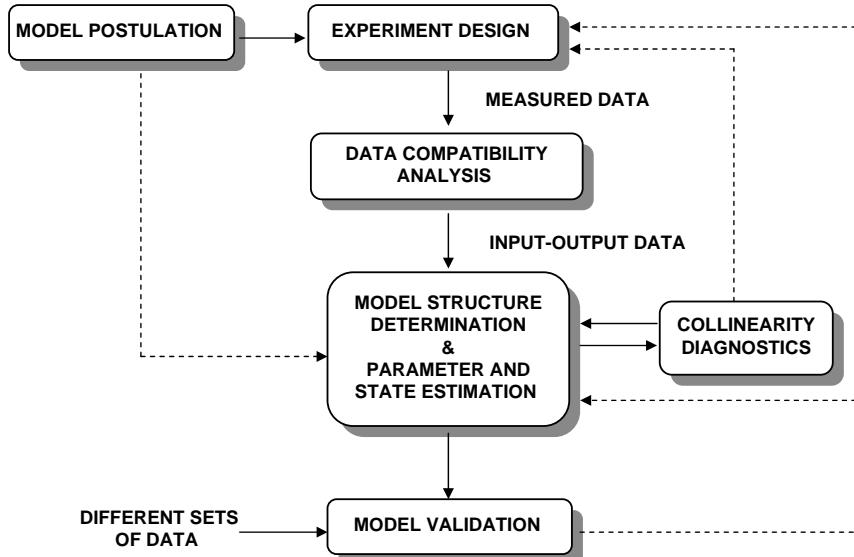


Fig. 2.1 Block diagram of aircraft system identification.

maneuvers for system identification. The instrumentation system is primarily required to measure input and output variables at regular sampling intervals during the maneuver. Input variables are throttle position and control surface deflections for open-loop or bare-airframe modeling. The output variables include quantities specifying the magnitude and direction of the air-relative velocity (airspeed, angle of attack, and sideslip angle), angular velocities, translational and angular accelerations, and Euler attitude angles. In addition to these variables, quantities defining flight conditions and configuration are also recorded.

An important aspect of the experiment design is the selection of input forms for the flight maneuvers. The input influences aircraft response, which in turn influences the accuracy of the system identification from flight measurements. Attempts to obtain parameter estimates with high accuracy in the most efficient manner has led researchers to the development of optimized inputs for aircraft parameter estimation. This and other experiment design issues are discussed in Chapter 9.

**Data compatibility analysis.** In practice, measured aircraft response data can contain systematic errors, even after careful instrumentation and experimental procedure. To verify data accuracy, data compatibility analysis can be applied to measured aircraft responses. Data compatibility analysis includes aircraft state estimation based on known rigid-body kinematics and available sensor measurements, estimation of systematic instrumentation errors, and a comparison of reconstructed responses with measured responses.

The state equations for the data compatibility analysis are kinematic relationships among the measured aircraft responses, and the model parameters are

constant biases and scale factor errors for the sensors. The estimation techniques employed are similar to those used in estimation of aircraft aerodynamic parameters. Data compatibility analysis is described in Chapter 10.

***Model structure determination.*** Model structure determination in aircraft system identification means selecting a specific form for the model from a class of models, based on measured data. For example, this might involve choosing an appropriate polynomial expansion in the aircraft motion and control variables to model a component of aerodynamic force acting on the aircraft, from the class of all possible polynomial models of order two or less. The model should be parsimonious to retain good prediction capability, while still adequately representing the physical phenomena. An adequate model is a model that fits the data well, facilitates the successful estimation of unknown parameters associated with model terms whose existence can be substantiated, and has good prediction capabilities.

Several techniques for selection of an aerodynamic model have been developed, and two of these are described in Chapter 5. One of the techniques, step-wise regression, has been used extensively in practice. In this technique, the determination of a model proceeds in three steps: postulation of terms that might enter the model, selection of an adequate model based on statistical metrics, and validation of the selected model. The other technique generates multivariate orthogonal modeling functions from the data to facilitate model structure determination. The orthogonality of the modeling functions make it possible to automate the first two of the three steps listed earlier for model structure determination. Retained orthogonal functions can be decomposed without error into ordinary polynomial terms for the final model form.

***Parameter and state estimation.*** Four items are needed for implementation of aircraft system identification: an informative experiment, measured input-output data, a mathematical model of the aircraft being tested, and an estimation technique. Parameter and state estimation constitute a principal part of the aircraft system identification procedure.

Currently, two methods—equation-error and output-error—are used for most aircraft parameter estimation. The equation-error method is based on linear regression using the ordinary least-squares principle. The unknown aerodynamic parameters are estimated by minimizing the sum of squared differences between measured and modeled aerodynamic forces and moments. Linear regression constitutes a linear estimation problem, meaning that the model output is linearly dependent on the model parameters. This simplifies the optimization required to find parameter estimates to the solution of an overdetermined set of linear equations, which can be found using well-known techniques from linear algebra.

In the output-error method, the unknown parameters are obtained by minimizing the sum of weighted square differences between the measured aircraft outputs and model outputs. The estimation problem is nonlinear because the unknown parameters appear in the equations of motion, which are integrated to compute the states. Outputs are computed from the states, controls, and parameters, using the output equations. Iterative nonlinear optimization techniques are required to solve this nonlinear estimation problem.

## ELEMENTS OF SYSTEM THEORY

23

Theoretically, either the equation-error or the output-error method can be a maximum likelihood estimator, which means that the cost function optimization used for computing the unknown parameters is equivalent to maximizing the probability density of the outcome from the experiment. In addition, both equation-error and output-error parameter estimation can be considered special cases of a more general approach based on Bayes's rule. These methods will be addressed in Chapters 4, 5, and 6.

Parameter estimation for linear dynamic systems based on maximum likelihood and the least-squares principle can also be formulated in the frequency domain. For this case, the measured data are first transformed from the time domain to the frequency domain using the Fourier transform. Parameter estimation methods can be applied to transformed input and output data, frequency response curves, or power spectral densities. The last two forms can also be used in nonparametric estimation methods. Both parameter estimation and nonparametric estimation in the frequency domain are covered in Chapter 7.

Sometimes during flight testing, the aircraft is subjected to random external disturbances, e.g., turbulence. In this case, the model becomes stochastic, and the states must be estimated, in addition to model parameters. The Kalman filter is predominantly used to estimate states in the case of a linear dynamic model. For a nonlinear dynamic model, the extended Kalman filter can be applied to estimate the states. The extended Kalman filter can also serve as a technique for obtaining simultaneous estimates of states and parameters, regardless of whether the dynamic model is linear or nonlinear. Chapters 4, 6, 7, and 8 contain material related to the use of the Kalman filter in aircraft system identification.

An overview of various estimation methods is given in Chapter 4. More detailed development of the methods and their practical application are discussed in Chapters 5, 6, and 7. Chapter 8 is concerned with real-time implementations of the estimation algorithms.

***Collinearity diagnostics.*** In almost all practical applications of linear regression, the model terms are correlated to some extent. Usually, the levels of correlation are low, and therefore not problematic. However, in some situations, the model terms are almost linearly related. When this happens, the problem of data collinearity exists, and inferences about the model based on the data can be misleading, or completely wrong.

The ability to diagnose data collinearity is important to users of linear regression or other parameter estimation techniques. Such a diagnostic consists of two basic steps: 1) detecting the presence of collinearity among the model terms, and 2) assessing the extent to which these relationships have adversely affected estimated parameters. Then, diagnostic information can aid in deciding what corrective actions are necessary and worthwhile. Data collinearity diagnostics are discussed in Chapter 5.

***Model validation.*** Model validation is the last step in the identification process, and should be applied regardless of how the model was found. The identified model must demonstrate that its parameters have physically reasonable values and acceptable accuracy, and that the model has good prediction

capability on comparable maneuvers. Flight-determined parameter estimates should be compared with any available information about the aircraft aerodynamics, which can include theoretical predictions, wind-tunnel measurements, or estimates from previous flight measurements using different maneuvers and/or different estimation techniques. During these comparisons, the limitations and accuracy of theoretical calculations, wind-tunnel measurements, and the flight results must be taken into consideration.

Prediction capability of an identified model is checked on data not used in the identification process. The measured input for the prediction data is applied to the identified model to compute predicted responses, which are then compared with measured values. The differences between predicted values from the model and measured values should be random in nature, indicating that all deterministic components in the measured output have been represented by the identified model. Examples of model validation are presented throughout the book.

## 2.4 Summary and Concluding Remarks

This chapter briefly outlined two elements of system theory that are relevant for this book—mathematical modeling and system identification.

An aircraft is a nonlinear dynamic system, so the mathematical model for the dynamic motion consists of nonlinear differential equations. When the motion of the aircraft is restricted to small perturbations, the aircraft can be modeled as a linear dynamic system. In this chapter, state-space and transfer function mathematical models for continuous-time linear dynamic systems were developed, along with the relationship between these two model forms, and between the continuous-time and discrete-time forms of linear state-space dynamic models. In practical cases, the outputs from any of these models for given inputs and initial conditions are found numerically, because the inputs are measured time series that are not easily characterized analytically.

Once the decision has been made to model aircraft dynamic motion with a particular class of model, e.g., a continuous-time linear state-space model, the next tasks are to determine the structure of that model type, including deciding which inputs, outputs, and states will be included, and determining the number and role of unknown parameters to be estimated, followed by estimating the unknown model parameters based on measured data. These steps, called model structure determination and parameter estimation, respectively, are important parts of system identification. System identification also includes other tasks that support and interact with model structure determination and parameter estimation. These tasks include design of the experiment, data compatibility analysis, collinearity diagnostics, and model validation. Brief descriptions of these aspects of system identification were included in the chapter.

The intent of this chapter is to give some background on the mathematical model forms used to characterize dynamic systems such as an aircraft, and to give a general overview of system identification applied to aircraft. Subsequent chapters will fill in details of the theory and application. The next chapter begins this process by explaining in detail the mathematical model forms used for aircraft system identification.

## ELEMENTS OF SYSTEM THEORY

25

**References**

- <sup>1</sup>DeRusso, P. M., Roy, R. J., and Close, C. M., *State Variables for Engineers*, Wiley, New York, 1965.
- <sup>2</sup>Unbehauen, H., and Rao, G. P., *Identification of Continuous Systems*, North-Holland, New York, 1987.
- <sup>3</sup>Sorenson, H. W., *Parameter Estimation: Principles and Problems*, Marcel-Decker, New York, 1980.
- <sup>4</sup>Eykhoff, P., *System Identification, Parameter and State Estimation*, Wiley, New York, 1974.
- <sup>5</sup>Ljung, L., *System Identification, Theory for the User*, 2nd ed., Prentice-Hall, Upper Saddle River, NJ, 1999.
- <sup>6</sup>Scheppe, F. C., *Uncertain Dynamic Systems*, Prentice-Hall, Upper Saddle River, NJ, 1973.

This page intentionally left blank

**3****Mathematical Model of an Aircraft**

Aircraft system identification is mainly concerned with providing a mathematical description for the aerodynamic forces and moments in terms of relevant measureable quantities such as control surface deflections, aircraft angular velocities, airspeed or Mach number, and the orientation of the aircraft to the relative wind. Aerodynamic parameters quantify the functional dependence of the aerodynamic forces and moments on measureable quantities, when the mathematical model is parametric.

Estimation of aerodynamic parameters from flight-test data requires that a mathematical model of the aircraft be postulated. The mathematical model includes both the aircraft equations of motion and the equations for aerodynamic forces and moments, known as the aerodynamic model equations. In this chapter, the equations of motion will be formulated as ordinary differential equations for the aircraft states, with algebraic equations for the measured outputs. The aerodynamic model equations will be developed first using linear terms, polynomials, and polynomial splines with time-invariant parameters, then generalized to include time-dependent terms representing unsteady aerodynamic effects.

Continuous-time differential equations are used almost exclusively for aircraft system identification. The main reasons are used the form of the aircraft equations of motion (which will be derived in this chapter) is known, and the parameters appearing in the continuous-time differential equations have physical significance for aircraft stability and control. It is therefore of interest to estimate the values of these parameters and their associated uncertainties. Furthermore, results from wind-tunnel tests and analytic computations are typically given as values of the physical parameters, and these values are often used as *a priori* information or for comparison with results from flight data analysis. Although it is possible to use discrete-time equations for flight data analysis, the parameters in discrete-time models are not the same as the physical parameters, and this introduces additional complexity.

The development given here is for conventional airplanes. Some fairly straightforward modifications can be made to model aircraft that vary from a conventional airplane in areas such as the controls available and the aircraft configuration. Throughout the book, the assumption that the aircraft is a rigid body is adopted. There is a vast literature concerning modeling the dynamics of flexible vehicles, including methods to address the important problems of flutter

and structural divergence. However, in many practical applications of aircraft system identification, the aircraft can be usefully approximated as a rigid body.

### 3.1 Reference Frames and Sign Conventions

Before developing the aircraft equations of motion, it is necessary to define reference frames and sign conventions. The reference frames required for studying aircraft dynamics and system identification are defined next. All reference frames are right handed and with mutually orthogonal axes.

#### 3.1.1 Reference Frames

*Inertial axes.* The origin of this reference frame is fixed or moving with a constant velocity relative to the distant stars, and the orientation is arbitrary and fixed. Newton's laws apply in an inertial reference frame defined this way.

*Earth axes*  $Ox_Ey_Ez_E$ . Origin is at an arbitrary point on the earth surface, with positive  $Ox_E$  axis pointing toward geographic north, positive  $Oy_E$  axis pointing east, and positive  $Oz_E$  axis pointing to the center of the earth. Earth axes are fixed with respect to the Earth. For most aircraft system identification work, Earth axes are assumed to be inertial axes, which is equivalent to ignoring the motion of the earth relative to the distant stars.

*Vehicle-carried earth axes*  $Ox_Vy_Vz_V$ . Origin is at the aircraft center of gravity (c.g.), orientation of the axes is parallel to Earth axes. The center of gravity is the point about which the aircraft would balance if suspended by a cable. This reference frame is used to conveniently show the rotational orientation of the aircraft relative to Earth axes.

*Body axes*  $Oxyz$ . Origin is at the aircraft c.g., with positive  $Ox$  axis pointing forward through the nose of the aircraft, positive  $Oy$  axis out the right wing, and positive  $Oz$  axis through the underside. The  $Oxz$  plane is usually a plane of symmetry for the aircraft. Body axes are fixed with respect to the aircraft body (see Fig. 3.1). It follows that the position vector between specified points on a rigid body is constant in body axes. Many of the variables associated with aircraft motion are referenced to body axes.

*Stability axes*  $Ox_sy_sz_s$ . Stability axes are a type of body axes, so they are fixed with respect to the aircraft. The orientation of stability axes is related to a reference flight condition, usually defined at the start of a maneuver. Origin is at the aircraft c.g., with positive  $Ox_s$  axis forward and aligned with the projection of the velocity vector of the aircraft c.g. through the air (also called the air-relative velocity) onto the  $Oxz$  plane in body axes. The positive  $Oy_s$  axis points out the right wing, and the positive  $Oz_s$  axis is directed through the underside.

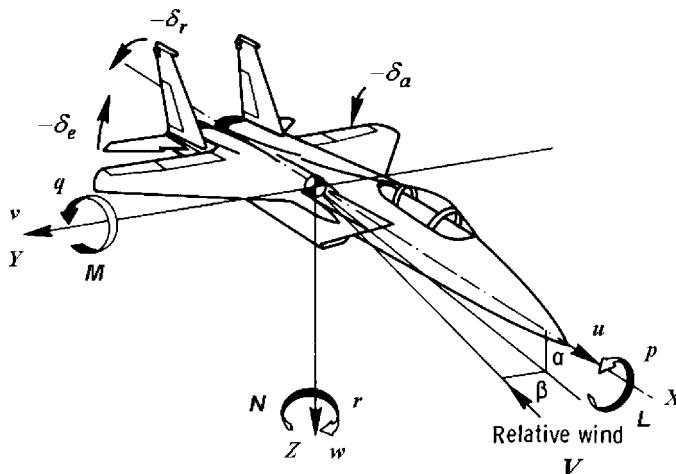


Fig. 3.1 Airplane notation and sign conventions:  $u, v, w$  = body-axis components of aircraft velocity relative to Earth axes;  $p, q, r$  = body-axis components of aircraft angular velocity;  $X, Y, Z$  = body-axis components of aerodynamic force acting on the aircraft; and  $L, M, N$  = body-axis components of aerodynamic moment acting on the aircraft.

**Wind axes  $Ox_w, y_w, z_w$ .** Origin is at the aircraft c.g., with positive  $Ox_w$  axis forward and aligned with the air-relative velocity vector, positive  $Oy_w$  axis out the right side of the aircraft, and positive  $Oz_w$  axis through the underside in the  $Oxz$  plane in body axes. The origin of the wind axes traces out the trajectory of the aircraft through the air. The  $Ox_w$  wind axis is always tangent to the air-relative trajectory, so wind axes are not fixed with respect to the aircraft body.

A typical jet fighter aircraft is shown in Fig. 3.1. Components of the translational velocity vector, angular velocity vector, applied aerodynamic force vector, and applied aerodynamic moment vector are expressed in components along the body axes. The notation shown is standard.

### 3.1.2 Sign Conventions

For angular velocities or applied moments, the sign convention follows the right-hand rule. If the right-hand thumb is pointed in the direction of a positive axis, the fingers curl in the direction of positive rotation. Angular velocities or applied moments about the  $x$ ,  $y$ , and  $z$  body axes are described with the adjectives roll, pitch, and yaw, respectively.

Control surfaces are hinged surfaces that can be rotated about a hinge line to change the applied aerodynamic forces and moments on an aircraft. Conventional airplanes have elevator  $\delta_e$ , aileron  $\delta_a$ , and rudder  $\delta_r$  control surfaces (see Fig. 3.2). These controls are intended primarily to produce moments about the body  $y$  (pitch) axis,  $x$  (roll) axis, and  $z$  (yaw) axis, respectively. A variety of

other control surfaces may appear instead of or in addition to these three basic controls, such as flaps, strakes, canards, speed brakes, stabilators, and spoilers. Some aircraft also have the capability to change the line of action of the thrust, which is called thrust vectoring. Thrust vectoring controls are fundamentally different from aerodynamic control surfaces in that the change in the line of action of the thrust is the source of the applied force and moment change, rather than aerodynamics.

Individual control surface deflections also follow the right-hand rule. Some control surfaces, like the ailerons, are deflected simultaneously in an asymmetric manner, which means that the individual aileron control surfaces on each wing move in opposite directions. This requires another way to define a positive aileron deflection. Although there is no universal standard, in this text the aileron deflection is defined as one-half the right aileron deflection minus the left aileron deflection,

$$\delta_a \equiv \frac{1}{2}(\delta_{aR} - \delta_{aL}) \quad (3.1)$$

Figure 3.2 illustrates asymmetric aileron deflection. Other asymmetric control surface deflections are defined similarly. Pairs of individual control surfaces can also be deflected symmetrically, which means the individual control surfaces are deflected together in the same direction with the same magnitude. The sign convention for control surface deflections is usually such that a positive control surface deflection produces a negative aerodynamic moment on the aircraft.

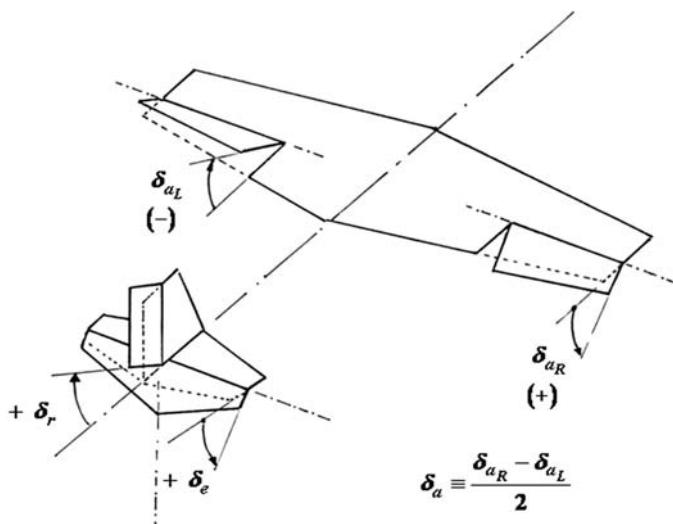


Fig. 3.2 Control surface sign conventions.

### 3.2 Rigid-Body Equations of Motion

A general formulation of aircraft flight dynamics would consider the dynamics of an elastic vehicle with varying mass density and moving component subsystems, subject to aerodynamic, propulsive, and gravitational forces, flying in nonstationary air. Common simplifying assumptions (used for most of this text) are as follows:

- 1) The aircraft is a rigid body with fixed mass distribution and constant mass.
- 2) The air is at rest relative to the earth (no steady wind, gusts, or wind shears).
- 3) The earth is fixed in inertial space.
- 4) Flight in the earth's atmosphere is close to the earth's surface (on an astronomical scale), so the earth's surface can be approximated as flat.
- 5) Gravity is uniform, so that the aircraft c.g. and the center of mass are coincident, and gravitational forces do not change with altitude.

The assumption that the aircraft is a rigid body means that dynamic effects due to fuel slosh, structural deformations, and relative motion of control surfaces, are assumed negligible. The general motion of an aircraft can then be described by Newton's second law of motion in translational and rotational forms:

$$\mathbf{F} = \frac{d}{dt}(m\mathbf{V}) \quad (3.2)$$

$$\mathbf{M} = \frac{d}{dt}(\mathbf{I}\boldsymbol{\omega}) \quad (3.3)$$

where  $\mathbf{F}$  is the applied force,  $m\mathbf{V}$  is the linear momentum,  $m$  is the mass,  $\mathbf{V}$  is the translational velocity,  $\mathbf{M}$  is the applied moment about the c.g.,  $\mathbf{I}\boldsymbol{\omega}$  is the angular momentum about the c.g.,  $\boldsymbol{\omega}$  is the angular velocity, and  $\mathbf{I}$  is the inertia matrix. Equations (3.2) and (3.3) are vector equations describing the translational motion of the c.g. and the rotational motion about the c.g., respectively. Each vector equation represents three scalar equations for the vector components, giving a total of six scalar equations for six degrees of freedom for the aircraft motion.

Equations (3.2) and (3.3) are valid in an inertial reference frame, but it is convenient to express the individual quantities in terms of their components in the body axes, which will generally be translating and rotating relative to inertial axes. The reasons for this are that most measurements are made in the body-axis system, and the inertia matrix  $\mathbf{I}$  is constant in body axes, but would be a function of time in inertial axes. Body-axis components of the quantities in Eqs. (3.2) and (3.3) are

$$\mathbf{F} = \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix} \quad \mathbf{V} = \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (3.4)$$

$$\mathbf{M} = \begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} \quad \mathbf{I} = \begin{bmatrix} I_x & -I_{xy} & -I_{xz} \\ -I_{yx} & I_y & -I_{yz} \\ -I_{zx} & -I_{zy} & I_z \end{bmatrix} \quad \boldsymbol{\omega} = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (3.5)$$

where

$$\begin{aligned}
 I_x &\equiv \int_{\text{Volume}} x^2 dm & I_y &\equiv \int_{\text{Volume}} y^2 dm & I_z &\equiv \int_{\text{Volume}} z^2 dm \\
 I_{xy} &\equiv \int_{\text{Volume}} xy dm = I_{yx} & I_{yz} &\equiv \int_{\text{Volume}} yz dm = I_{zy} \\
 I_{xz} &\equiv \int_{\text{Volume}} xz dm = I_{zx}
 \end{aligned} \tag{3.6}$$

Sign conventions for the components of  $\mathbf{F}$ ,  $\mathbf{V}$ ,  $\mathbf{M}$ , and  $\boldsymbol{\omega}$  are consistent with sign conventions for the body axes and the right-hand rule. The components of the angular velocity vector  $\boldsymbol{\omega}$  are roll rate  $p$  (positive right wing down), pitch rate  $q$  (positive nose up), and yaw rate  $r$  (positive nose right).

The quantities  $x$ ,  $y$ , and  $z$  inside the integrals in Eqs. (3.6) are body-axis coordinates of mass elements  $dm$ , which together compose the aircraft. Many texts contain derivations of the expressions in Eqs. (3.6), including Etkin,<sup>1</sup> Etkin and Reid,<sup>2</sup> McRuer et al.,<sup>3</sup> and Roskam.<sup>4</sup> These references are also resources for the derivation of the rigid-body equations of motion given next.

From the definitions in Eqs. (3.6), it is clear that for a rigid body with symmetry relative to the  $Oxz$  plane in body axes, the inertia matrix  $\mathbf{I}$  is symmetric, and  $I_{xy} = I_{yx} = I_{yz} = I_{zy} = 0$ . The inertia matrix then reduces to

$$\mathbf{I} = \begin{bmatrix} I_x & 0 & -I_{xz} \\ 0 & I_y & 0 \\ -I_{xz} & 0 & I_z \end{bmatrix} \tag{3.7}$$

so that

$$\mathbf{I}\boldsymbol{\omega} = \begin{bmatrix} I_x p - I_{xz} r \\ I_y q \\ -I_{xz} p + I_z r \end{bmatrix} \tag{3.8}$$

Note that translational velocity  $\mathbf{V}$  and angular velocity  $\boldsymbol{\omega}$  represent the aircraft motion relative to inertial axes, but expressed in body-axis components. All aircraft motion relative to body axes is zero by definition of the body axes.

For rotating axis systems like the body axes, the derivative operator applied to vectors has two parts—one that accounts for the rate of change of the vector components expressed in the rotating system, and one that accounts for the axis system rotation<sup>2</sup>:

$$\frac{d}{dt}(\cdot) = \frac{\delta}{\delta t}(\cdot) + \boldsymbol{\omega} \times (\cdot) \tag{3.9}$$

## MATHEMATICAL MODEL OF AN AIRCRAFT

33

Combining Eqs. (3.2), (3.3), and (3.9) with the rigid body and constant mass assumptions, and using a dot superscript notation for  $\delta/\delta t$ ,

$$\mathbf{F} = m\dot{\mathbf{V}} + \boldsymbol{\omega} \times m\mathbf{V} \quad (3.10)$$

$$\mathbf{M} = \mathbf{I}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{I}\boldsymbol{\omega} \quad (3.11)$$

Equations (3.10) and (3.11) are the vector forms of the equations of motion written in body axes. Substituting Eqs. (3.4), (3.5), (3.7), and (3.8) into Eqs. (3.10) and (3.11) gives the body-axis component form of the equations:

*Force equations:*

$$F_x = m(\dot{u} + qw - rv) \quad (3.12a)$$

$$F_y = m(\dot{v} + ru - pw) \quad (3.12b)$$

$$F_z = m(\dot{w} + pv - qu) \quad (3.12c)$$

*Moment equations:*

$$M_x = \dot{p}I_x - \dot{r}I_{xz} + qr(I_z - I_y) - qpI_{xz} \quad (3.13a)$$

$$M_y = \dot{q}I_y + pr(I_x - I_z) + (p^2 - r^2)I_{xz} \quad (3.13b)$$

$$M_z = \dot{r}I_z - \dot{p}I_{xz} + pq(I_y - I_x) + qrI_{xz} \quad (3.13c)$$

For airplanes, the applied forces and moments on the left sides of the preceding equations arise from aerodynamics, gravity, and propulsion. Since gravity acts through the c.g., and the gravity field is assumed uniform, there is no gravity moment acting on the airplane. Equations (3.10) and (3.11) can therefore be written as

$$\mathbf{F}_A + \mathbf{F}_T + \mathbf{F}_G = m\dot{\mathbf{V}} + \boldsymbol{\omega} \times m\mathbf{V} \quad (3.14)$$

$$\mathbf{M}_A + \mathbf{M}_T = \mathbf{I}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{I}\boldsymbol{\omega} \quad (3.15)$$

**Aerodynamics.** Aerodynamic forces and moments acting on the aircraft result from the relative motion of the air and the aircraft. Components of the aerodynamic forces and moments can be expressed in terms of nondimensional coefficients:

$$\mathbf{F}_A = \bar{q}S \begin{bmatrix} C_X \\ C_Y \\ C_Z \end{bmatrix} \quad (3.16)$$

$$\mathbf{M}_A = \bar{q}S \begin{bmatrix} bC_l \\ \bar{c}C_m \\ bC_n \end{bmatrix} \quad (3.17)$$

where  $\bar{q} = (1/2)\rho V^2$  is the dynamic pressure,  $V$  is the magnitude of the air-relative velocity (also called the airspeed),  $\rho$  is the air density,  $S$  is the wing reference area,  $b$  is the wing span, and  $\bar{c}$  is the mean aerodynamic chord of the wing (see Appendix C).

In general, the nondimensional aerodynamic force and moment coefficients depend nonlinearly on the aircraft translational and angular velocity vector components and the control surface deflections, plus possibly their time derivatives, and/or other nondimensional quantities, such as Mach number and Reynolds number. A discussion of how this dependence can be characterized mathematically is given later in the chapter.

**Gravity.** Aircraft weight is assumed constant in both magnitude and direction relative to earth axes, acting along earth axis  $O_{ZE}$ . The components of the aircraft weight along the body axes change with orientation of the aircraft relative to earth axes. Gravity components in body axes therefore depend on the aircraft orientation relative to earth axes, and this dependence can be described based on the relative orientation of the body axes to vehicle-carried earth axes. Aircraft orientation with respect to vehicle-carried earth axes can be described in many ways, but the most common method is using Euler angles.

Figure 3.3 shows how the orientation of one right-handed coordinate system can be defined relative to another. The sequence for rotating vehicle-carried Earth axes into alignment with body axes is a yaw angle rotation  $\psi$  about the  $O_{ZV}$  axis, followed by a pitch angle rotation  $\theta$  about an intermediate  $y$  axis, completed by a roll angle rotation  $\phi$  about the  $Ox$  body axis.

Components of the gravity vector in body axes can be found by means of a product of three rotation matrices (see Appendix A):

$$\mathbf{F}_G = \begin{bmatrix} g_x \\ g_y \\ g_z \end{bmatrix}_B = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}_V$$

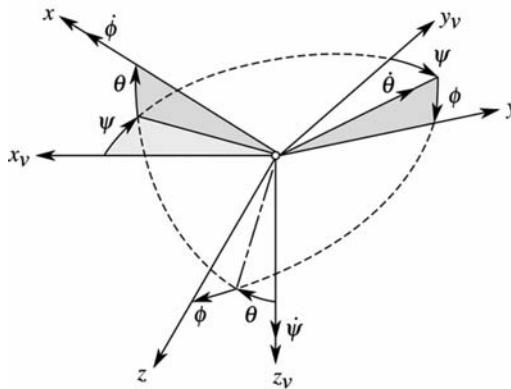
$$\begin{bmatrix} g_x \\ g_y \\ g_z \end{bmatrix}_B = \begin{bmatrix} -g\sin\theta \\ g\sin\phi\cos\theta \\ g\cos\phi\cos\theta \end{bmatrix}$$

so

$$\mathbf{F}_G = m \begin{bmatrix} g_x \\ g_y \\ g_z \end{bmatrix}_B = \begin{bmatrix} -mg\sin\theta \\ mg\sin\phi\cos\theta \\ mg\cos\phi\cos\theta \end{bmatrix} \quad (3.18)$$

## MATHEMATICAL MODEL OF AN AIRCRAFT

35



**Fig. 3.3 Rotation from vehicle-carried Earth axes to body axes.**

**Propulsion.** Assuming the thrust from the propulsion system acts along the  $x$  body axis and through the c.g., the thrust appears only as an applied force along the  $x$  body axis,

$$\mathbf{F}_T = \begin{bmatrix} T \\ 0 \\ 0 \end{bmatrix} \quad (3.19a)$$

When the line of action of the thrust is not directed along the aircraft longitudinal body axis and through the c.g., thrust terms appear in other applied force and moment body-axis components. This would be the case for aircraft equipped with thrust vectoring or engines mounted below the wings and below the c.g., for example.

Sometimes it is necessary to account for the effect of rotating mass in the propulsion system, e.g., propellers or rotors of jet engines. The gyroscopic terms associated with the rotating mass must be considered an applied moment, because the equations of motion have been formulated assuming the aircraft is a rigid body with no internal moving parts.

Assuming the thrust acts along the  $x$  body axis of the aircraft, the angular momentum of the rotating mass in body axes is

$$\mathbf{h}_p = [I_p \Omega_p \quad 0 \quad 0]^T \quad (3.19b)$$

where  $I_p$  is the inertia of the rotating mass and  $\Omega_p$  is the angular velocity. If the angular velocity of the rotating mass is constant, then  $I_p \Omega_p = 0$ , and the gyroscopic moment from the rotating mass in the propulsion system is given

by [cf. Eq. (3.9)]

$$\mathbf{M}_T = \frac{d}{dt}(\mathbf{h}_p) = \boldsymbol{\omega} \times \mathbf{h}_p = \begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix} \begin{bmatrix} I_p \Omega_p \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ I_p \Omega_p r \\ -I_p \Omega_p q \end{bmatrix} \quad (3.19c)$$

The components in Eq. (3.19c) are added on the left side of the moment equations (3.13).

**Applied forces and moments.** Combining the expressions for the aerodynamic forces and moments with Eqs. (3.18) and (3.19), the body-axis components of the applied forces and moments are

$$F_x = \bar{q}SC_X - mg \sin \theta + T \quad (3.20a)$$

$$F_y = \bar{q}SC_Y + mg \cos \theta \sin \phi \quad (3.20b)$$

$$F_z = \bar{q}SC_Z + mg \cos \theta \cos \phi \quad (3.20c)$$

$$M_x = \bar{q}SbC_l \quad (3.21a)$$

$$M_y = \bar{q}S\bar{c}C_m + I_p\Omega_p r \quad (3.21b)$$

$$M_z = \bar{q}SbC_n - I_p\Omega_p q \quad (3.21c)$$

Substituting the preceding expressions into the dynamic equations (3.12)–(3.13) gives

*Force equations:*

$$m\dot{u} = m(rv - qw) + \bar{q}SC_X - mg \sin \theta + T \quad (3.22a)$$

$$m\dot{v} = m(pw - ru) + \bar{q}SC_Y + mg \cos \theta \sin \phi \quad (3.22b)$$

$$m\dot{w} = m(qu - pv) + \bar{q}SC_Z + mg \cos \theta \cos \phi \quad (3.22c)$$

*Moment equations:*

$$\dot{p}I_x - \dot{r}I_{xz} = \bar{q}SbC_l - qr(I_z - I_y) + qpI_{xz} \quad (3.23a)$$

$$\dot{q}I_y = \bar{q}S\bar{c}C_m - pr(I_x - I_z) - (p^2 - r^2)I_{xz} + I_p\Omega_p r \quad (3.23b)$$

$$\dot{r}I_z - \dot{p}I_{xz} = \bar{q}SbC_n - pq(I_y - I_x) - qrI_{xz} - I_p\Omega_p q \quad (3.23c)$$

### 3.3 Rotational Kinematic Equations

The rotational kinematic equations relate the rate of change of the Euler angles to the body-axis components of angular velocity. The relationships can be found by applying the technique in Appendix A for expressing vector components in a

## MATHEMATICAL MODEL OF AN AIRCRAFT

37

rotated axis system,

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & \sin \phi \\ 0 & -\sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & \sin \phi \\ 0 & -\sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \quad (3.24)$$

or

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\sin \theta \\ 0 & \cos \phi & \sin \phi \cos \theta \\ 0 & -\sin \phi & \cos \phi \cos \theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (3.25)$$

Inverting the last relationship gives differential equations for the Euler angles, which describe the rotational kinematics:

$$\dot{\phi} = p + \tan \theta (q \sin \phi + r \cos \phi) \quad (3.26a)$$

$$\dot{\theta} = q \cos \phi - r \sin \phi \quad (3.26b)$$

$$\dot{\psi} = \frac{q \sin \phi + r \cos \phi}{\cos \theta} \quad (3.26c)$$

Equations (3.22), (3.23), and (3.26) are coupled nonlinear first-order differential equations for nine aircraft states: three translational velocity components  $u$ ,  $v$ , and  $w$ ; three angular velocity components  $p$ ,  $q$ , and  $r$ ; and three Euler angles  $\phi$ ,  $\theta$ , and  $\psi$ .

### 3.4 Navigation Equations

The navigation equations are written by expressing the aircraft velocity vector in earth axes, starting with body-axis components,

$$\begin{bmatrix} \dot{x}_E \\ \dot{y}_E \\ \dot{z}_E \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & \sin \phi \\ 0 & -\sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix} \begin{bmatrix} \cos \psi & \sin \psi & 0 \\ -\sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (3.27)$$

Introducing  $h = \text{altitude (height above the ground)} = -z_E$ ,

$$\begin{aligned}\dot{x}_E &= u \cos \psi \cos \theta + v (\cos \psi \sin \theta \sin \phi - \sin \psi \cos \phi) \\ &\quad + w (\cos \psi \sin \theta \cos \phi + \sin \psi \sin \phi)\end{aligned}\quad (3.28a)$$

$$\begin{aligned}\dot{y}_E &= u \sin \psi \cos \theta + v (\sin \psi \sin \theta \sin \phi + \cos \psi \cos \phi) \\ &\quad + w (\sin \psi \sin \theta \cos \phi - \cos \psi \sin \phi)\end{aligned}\quad (3.28b)$$

$$\dot{h} = u \sin \theta - v \cos \theta \sin \phi - w \cos \theta \cos \phi \quad (3.28c)$$

### 3.5 Force Equations in Wind Axes

Aircraft sensors measure airspeed  $V \equiv |V|$ , angle of attack  $\alpha$ , and sideslip angle  $\beta$ , rather than body-axis velocities  $u$ ,  $v$ , and  $w$ . In addition, the nondimensional aerodynamic force and moment coefficients are generally characterized as functions of  $\alpha$ ,  $\beta$ , and Mach number  $V/a$ , where  $a$  is the speed of sound in air. Therefore, it is often useful to write the force equations in terms of  $V$ ,  $\alpha$ , and  $\beta$  instead of  $u$ ,  $v$ , and  $w$ . To do this, first note the definition of  $\alpha$  and  $\beta$  shown in Figs. 3.1 and 3.4. The sequence of rotating the wind axes into alignment with body axes is a negative  $\beta$  rotation about the  $Oz_w$  axis, followed by a positive  $\alpha$  rotation about the  $Oy$  body axis. These two aerodynamic angles are defined as

$$\alpha = \tan^{-1}\left(\frac{w}{u}\right) \quad (3.29)$$

$$\beta = \sin^{-1}\left(\frac{v}{V}\right) \quad (3.30)$$

and the airspeed  $V$  is given by

$$V \equiv |V| = \sqrt{u^2 + v^2 + w^2} \quad (3.31)$$

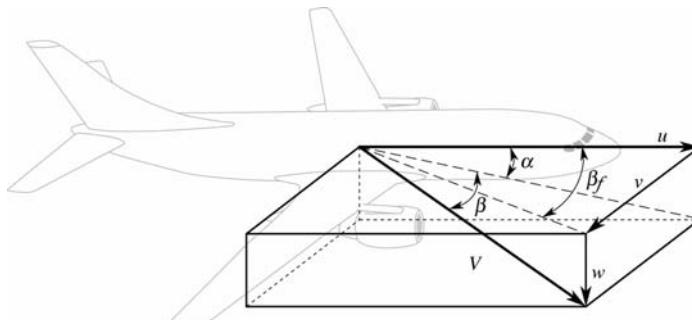


Fig. 3.4 Aerodynamic angle definitions.

## MATHEMATICAL MODEL OF AN AIRCRAFT

39

Body-axis velocity components are related to  $V$ ,  $\alpha$ , and  $\beta$  by

$$u = V \cos \alpha \cos \beta \quad (3.32a)$$

$$v = V \sin \beta \quad (3.32b)$$

$$w = V \sin \alpha \cos \beta \quad (3.32c)$$

Differentiating Eqs. (3.29)–(3.31) with respect to time gives

$$\dot{V} = \frac{1}{V}(u\dot{u} + v\dot{v} + w\dot{w}) \quad (3.33a)$$

$$\dot{\alpha} = \left( \frac{u\dot{w} - w\dot{u}}{u^2 + w^2} \right) \quad (3.33b)$$

$$\begin{aligned} \dot{\beta} &= \left( \frac{V\dot{v} - v\dot{V}}{V^2} \right) \left[ \frac{1}{\sqrt{1 - (v/V)^2}} \right] = \left[ \frac{V\dot{v} - v\dot{V}}{V\sqrt{u^2 + w^2}} \right] \\ &= \frac{(u^2 + v^2 + w^2)\dot{v} - v(u\dot{u} + v\dot{v} + w\dot{w})}{V^2\sqrt{u^2 + w^2}} \\ &= \frac{(u^2 + w^2)\dot{v} - v(u\dot{u} + w\dot{w})}{V^2\sqrt{u^2 + w^2}} \end{aligned} \quad (3.33c)$$

Substituting in Eqs. (3.33) for  $\dot{u}$ ,  $\dot{v}$ , and  $\dot{w}$  from Eqs. (3.22), and for  $u$ ,  $v$ , and  $w$  from Eqs. (3.32), gives the force equations written in terms of  $V$ ,  $\alpha$ , and  $\beta$ :

$$\begin{aligned} \dot{V} &= -\frac{\bar{q}S}{m} C_{D_W} + \frac{T}{m} \cos \alpha \cos \beta \\ &\quad + g(\cos \phi \cos \theta \sin \alpha \cos \beta + \sin \phi \cos \theta \sin \beta - \sin \theta \cos \alpha \cos \beta) \end{aligned} \quad (3.34a)$$

$$\begin{aligned} \dot{\alpha} &= -\frac{\bar{q}S}{mV \cos \beta} C_L + q - \tan \beta(p \cos \alpha + r \sin \alpha) \\ &\quad + \frac{g}{V \cos \beta} (\cos \phi \cos \theta \cos \alpha + \sin \theta \sin \alpha) - \frac{T \sin \alpha}{mV \cos \beta} \end{aligned} \quad (3.34b)$$

$$\begin{aligned} \dot{\beta} &= \frac{\bar{q}S}{mV} C_{Y_W} + p \sin \alpha - r \cos \alpha + \frac{g}{V} \cos \beta \sin \phi \cos \theta \\ &\quad + \frac{\sin \beta}{V} \left( g \cos \alpha \sin \theta - g \sin \alpha \cos \phi \cos \theta + \frac{T \cos \alpha}{m} \right) \end{aligned} \quad (3.34c)$$

where

$$C_L = -C_Z \cos \alpha + C_X \sin \alpha \quad (3.35a)$$

$$C_D = -C_X \cos \alpha - C_Z \sin \alpha \quad (3.35b)$$

$$\begin{aligned} C_{D_W} &= -C_X \cos \alpha \cos \beta - C_Y \sin \beta - C_Z \sin \alpha \cos \beta \\ &= C_D \cos \beta - C_Y \sin \beta \end{aligned} \quad (3.35c)$$

$$\begin{aligned} C_{Y_W} &= -C_X \cos \alpha \sin \beta + C_Y \cos \beta - C_Z \sin \alpha \sin \beta \\ &= C_Y \cos \beta + C_D \sin \beta \end{aligned} \quad (3.35d)$$

The nondimensional coefficients on the left sides of Eqs. (3.35) are obtained from body-axes components by rotation through  $\alpha$  and  $\beta$ . Positive lift coefficient  $C_L$  and drag coefficient  $C_D$  are directed along the  $-z_S$  and  $-x_S$  stability axes, respectively, whereas positive wind-axes drag coefficient  $C_{D_W}$  and side force coefficient  $C_{Y_W}$  are directed along the  $-x_W$  and  $+y_W$  wind axes, respectively.

### 3.6 Collected Equations of Motion

The equations of motion in body axes, containing force and moment dynamic equations, and the rotational kinematic equations, are

*Force equations:*

$$\dot{u} = rv - qw + \frac{\bar{q}S}{m} C_X - g \sin \theta + \frac{T}{m} \quad (3.36a)$$

$$\dot{v} = pw - ru + \frac{\bar{q}S}{m} C_Y + g \cos \theta \sin \phi \quad (3.36b)$$

$$\dot{w} = qu - pv + \frac{\bar{q}S}{m} C_Z + g \cos \theta \cos \phi \quad (3.36c)$$

*Moment equations:*

$$\dot{p} - \frac{I_{xz}}{I_x} \dot{r} = \frac{\bar{q}Sb}{I_x} C_l - \frac{(I_z - I_y)}{I_x} qr + \frac{I_{xz}}{I_x} qp \quad (3.37a)$$

$$\dot{q} = \frac{\bar{q}S\bar{c}}{I_y} C_m - \frac{(I_x - I_z)}{I_y} pr - \frac{I_{xz}}{I_y} (p^2 - r^2) + \frac{I_p}{I_y} \Omega_p r \quad (3.37b)$$

$$\dot{r} - \frac{I_{xz}}{I_z} \dot{p} = \frac{\bar{q}Sb}{I_z} C_n - \frac{(I_y - I_x)}{I_z} pq - \frac{I_{xz}}{I_z} qr - \frac{I_p}{I_z} \Omega_p q \quad (3.37c)$$

*Kinematic equations:*

$$\dot{\phi} = p + \tan \theta (q \sin \phi + r \cos \phi) \quad (3.38a)$$

$$\dot{\theta} = q \cos \phi - r \sin \phi \quad (3.38b)$$

$$\dot{\psi} = \frac{q \sin \phi + r \cos \phi}{\cos \theta} \quad (3.38c)$$

The force equations can be written in wind axes as

## MATHEMATICAL MODEL OF AN AIRCRAFT

41

*Wind-axes force equations:*

$$\begin{aligned}\dot{V} = & -\frac{\bar{q}S}{m}C_{D_W} + \frac{T}{m} \cos \alpha \cos \beta + g(\cos \phi \cos \theta \sin \alpha \cos \beta \\ & + \sin \phi \cos \theta \sin \beta - \sin \theta \cos \alpha \cos \beta)\end{aligned}\quad (3.39a)$$

$$\begin{aligned}\dot{\alpha} = & -\frac{\bar{q}S}{mV \cos \beta}C_L + q - \tan \beta(p \cos \alpha + r \sin \alpha) - \frac{T \sin \alpha}{mV \cos \beta} \\ & + \frac{g}{V \cos \beta}(\cos \phi \cos \theta \cos \alpha + \sin \theta \sin \alpha)\end{aligned}\quad (3.39b)$$

$$\begin{aligned}\dot{\beta} = & \frac{\bar{q}S}{mV}C_{Y_W} + p \sin \alpha - r \cos \alpha + \frac{g}{V} \cos \beta \sin \phi \cos \theta \\ & + \frac{\sin \beta}{V} \left( g \cos \alpha \sin \theta - g \sin \alpha \cos \phi \cos \theta + \frac{T \cos \alpha}{m} \right)\end{aligned}\quad (3.39c)$$

where

$$C_L = -C_Z \cos \alpha + C_X \sin \alpha \quad (3.40a)$$

$$C_D = -C_X \cos \alpha - C_Z \sin \alpha \quad (3.40b)$$

and

$$C_{D_W} = C_D \cos \beta - C_Y \sin \beta \quad (3.40c)$$

$$C_{Y_W} = C_Y \cos \beta + C_D \sin \beta \quad (3.40d)$$

The moment equations (3.37) can be rearranged in state-space form as

*State-space moment equations:*

$$\dot{p} = (c_1 r + c_2 p - c_4 I_p \Omega_p)q + \bar{q}Sb(c_3 C_l + c_4 C_n) \quad (3.41a)$$

$$\dot{q} = (c_5 p + c_7 I_p \Omega_p)r - c_6(p^2 - r^2) + c_7 \bar{q}S \bar{c} C_m \quad (3.41b)$$

$$\dot{r} = (c_8 p - c_2 r - c_9 I_p \Omega_p)q + \bar{q}Sb(c_9 C_n + c_4 C_l) \quad (3.41c)$$

where  $c_1, c_2, \dots, c_9$  = inertia constants dependent on body-axis moments of inertia,

$$\begin{aligned}c_1 &= [(I_y - I_z)I_z - I_{xz}^2]/\Gamma & \Gamma &= I_x I_z - I_{xz}^2 \\ c_2 &= [(I_x - I_y + I_z)I_{xz}]/\Gamma & c_3 &= I_z/\Gamma \\ c_4 &= I_{xz}/\Gamma & c_5 &= (I_z - I_x)/I_y \\ c_6 &= I_{xz}/I_y & c_7 &= 1/I_y \\ c_8 &= [(I_x - I_y)I_x - I_{xz}^2]/\Gamma & c_9 &= I_x/\Gamma\end{aligned}\quad (3.42)$$

The navigation equations can be written in terms of  $V, \alpha$ , and  $\beta$ , by substituting Eqs. (3.32) into Eqs. (3.28),

*Navigation equations:*

$$\begin{aligned}\dot{x}_E = & V \cos \alpha \cos \beta \cos \psi \cos \theta + V \sin \beta (\cos \psi \sin \theta \sin \phi - \sin \psi \cos \phi) \\ & + V \sin \alpha \cos \beta (\cos \psi \sin \theta \cos \phi + \sin \psi \sin \phi)\end{aligned}\quad (3.43a)$$

$$\begin{aligned}\dot{y}_E = & V \cos \alpha \cos \beta \sin \psi \cos \theta + V \sin \beta (\sin \psi \sin \theta \sin \phi + \cos \psi \cos \phi) \\ & + V \sin \alpha \cos \beta (\sin \psi \sin \theta \cos \phi - \cos \psi \sin \phi)\end{aligned}\quad (3.43b)$$

$$\dot{h} = V \cos \alpha \cos \beta \sin \theta - V \sin \beta \cos \theta \sin \phi - V \sin \alpha \cos \beta \cos \theta \cos \phi\quad (3.43c)$$

The equations in this section were developed under the following assumptions:

- 1) The earth is fixed in inertial space (i.e., earth axes are an inertial reference frame).
- 2) The aircraft is a rigid body.
- 3) Aircraft mass and mass distribution are constant.
- 4) The aircraft is symmetric about the  $Oxz$  plane in body axes.
- 5) The atmosphere is fixed relative to earth axes.
- 6) The earth has negligible curvature ("flat earth").
- 7) Gravitational acceleration is constant in magnitude and direction.
- 8) Thrust is directed along the  $x$  body axis and through the aircraft c.g.

The equations of motion can be expressed in the general form of a nonlinear first-order vector differential equation for the aircraft state,

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \quad (3.44)$$

where  $\mathbf{x}$  is a vector of state variables  $u, v, w, p, q, r, \phi, \theta, \psi, x_E, y_E, h$ , or  $V, \beta, \alpha, p, q, r, \phi, \theta, \psi, x_E, y_E, h$ , and  $\mathbf{u}$  is a vector of input variables that usually is composed of throttle position and control surface deflections. The input variables are not explicitly shown in the preceding collected force and moment equations, but are included implicitly, because they influence the thrust and the aerodynamic forces and moments acting upon the aircraft.

### 3.7 Output Equations

Output variables for the mathematical model of an aircraft are the measured aircraft responses  $V, \alpha, \beta, p, q, r, \phi, \theta, \psi, h, a_x, a_y, a_z, \dot{p}, \dot{q}$ , and  $\dot{r}$ . The quantities  $x_E$  and  $y_E$ , which define aircraft  $x$  and  $y$  position relative to earth axes, are not included, because these position coordinates are not relevant to aircraft dynamics. Altitude has an indirect effect on aircraft dynamics, which is manifested through changes in the air density, and consequently the dynamic pressure. Neither altitude  $h$  nor heading angle  $\psi$  affects aircraft dynamics directly, but these quantities are important piloting parameters that are usually measured. In addition, these measurements can be used in data compatibility analysis to

## MATHEMATICAL MODEL OF AN AIRCRAFT

43

check other measurements, as discussed in Chapter 10. For these reasons, the altitude  $h$  and heading angle  $\psi$  are retained as aircraft states in the development given here.

The output equations specify the analytical connection between the output variables and the aircraft states, state derivatives, and controls. For specifying the output equations, the following assumptions are made:

- 1) Instrument calibrations are known and have been applied to the raw measurements.
- 2) Measurements are corrected to the aircraft c.g., and for misalignment with the body axes.
- 3) Instrumentation dynamics are negligible.

Chapter 10 contains a detailed discussion of the steps that need to be taken with the instrumentation and raw measurements so that the preceding assumptions are applicable.

In general, the output equations take the following form:

$$\mathbf{y} = \mathbf{h}(\mathbf{x}, \dot{\mathbf{x}}, \mathbf{u}) \quad (3.45)$$

The precise form of the output equations in (3.45) depends on which aircraft responses are included as outputs, the aircraft instrumentation, and how the equations of motion are formulated. For example, if the force equations are written in body axes [cf. Eqs. (3.36)], the output variables are  $V$ ,  $\alpha$ , and  $\beta$ , then the output equations are

$$V = \sqrt{u^2 + v^2 + w^2} \quad (3.46a)$$

$$\alpha = \tan^{-1}\left(\frac{w}{u}\right) \quad (3.46b)$$

$$\beta = \sin^{-1}\left(\frac{v}{V}\right) \quad (3.46c)$$

These relationships, which were introduced earlier as Eqs. (3.29)–(3.31), show a nonlinear dependence of the output variables on the states. On the other hand, if the force equations are written in wind axes [cf. Eqs. (3.39)], the outputs  $V$ ,  $\alpha$ , and  $\beta$  are also states, so the output equations are simply

$$V = V \quad (3.47a)$$

$$\alpha = \alpha \quad (3.47b)$$

$$\beta = \beta \quad (3.47c)$$

which indicate a linear relationship between the outputs and the states. Similar output equations could be written for  $p$ ,  $q$ ,  $r$ ,  $\phi$ ,  $\theta$ ,  $\psi$ ,  $\dot{p}$ ,  $\dot{q}$ , and  $\dot{r}$ , because all of these quantities are either states or state derivatives, which are available directly from the equations of motion.

Airspeed  $V$  is usually obtained from dynamic pressure measurements. The aerodynamic angles  $\alpha$  and  $\beta$  are normally measured using wind vanes mounted on the aircraft. These vanes directly measure angle of attack  $\alpha$  in the  $Oxz$  plane, and flank angle  $\beta_f$  in the  $Oxy$  plane (see Fig. 3.4). The flank angle  $\beta_f$  can be expressed as

$$\beta_f = \tan^{-1} \left( \frac{v}{u} \right) \quad (3.48)$$

The sideslip angle  $\beta$  is defined as the angle between the velocity vector and its projection onto the  $Oxz$  plane. Sideslip angle is related to flank angle by

$$\begin{aligned} \tan \beta_f &= \frac{v}{u} = \frac{V \sin \beta}{V \cos \alpha \cos \beta} = \frac{\tan \beta}{\cos \alpha} \\ \beta &= \tan^{-1} (\tan \beta_f \cos \alpha) \end{aligned} \quad (3.49)$$

For small  $\beta$  and  $\beta_f$  in radians,

$$\beta \approx \beta_f \cos \alpha \quad (3.50a)$$

If angle of attack  $\alpha$  is also small,

$$\beta \approx \beta_f \quad (3.50b)$$

Accelerometers measure the translational acceleration due to applied forces, excluding gravity. In vector notation, the output equation for the translational acceleration is

$$\mathbf{a} = \dot{\mathbf{V}} + \boldsymbol{\omega} \times \mathbf{V} - \frac{\mathbf{F}_G}{m} = \frac{1}{m} (\mathbf{F}_A + \mathbf{F}_T) \quad (3.51)$$

which follows from Eq. (3.14). In scalar form,

$$a_x = \dot{u} - rv + qw + g \sin \theta \quad (3.52a)$$

$$a_y = \dot{v} - pw + ru - g \cos \theta \sin \phi \quad (3.52b)$$

$$a_z = \dot{w} - qu + pv - g \cos \theta \cos \phi \quad (3.52c)$$

The accelerometer output equations are nonlinear in the states and linear in the state derivatives. From Eq. (3.51), the accelerometer outputs can be related to the applied forces,

$$\mathbf{a} = \frac{(\mathbf{F}_A + \mathbf{F}_T)}{m} \quad (3.53)$$

## MATHEMATICAL MODEL OF AN AIRCRAFT

45

or in scalar form [cf. Eqs. (3.16) and (3.19a)],

$$a_x = \frac{1}{m}(\bar{q}SC_X + T) \quad (3.54a)$$

$$a_y = \frac{1}{m}(\bar{q}SC_Y) \quad (3.54b)$$

$$a_z = \frac{1}{m}(\bar{q}SC_Z) \quad (3.54c)$$

### 3.8 Aerodynamic Model Equations

Modeling the aircraft aerodynamics raises the fundamental question of what the mathematical structure of the model should be. Although a complicated model structure can be justified for accurate description of the aerodynamic forces and moments, it is not always clear what the relationship between model complexity and information in the measured data should be. If too many model parameters are sought for a limited amount of data, reduced accuracy of estimated parameters can be expected, or the attempts to estimate all the parameters in the model might fail.

In general, the nondimensional aerodynamic force and moment coefficients depend nonlinearly on present and past values of airspeed, angles of incidence of the air-relative velocity with respect to the aircraft body, aircraft rigid-body rotation rates, air-relative linear and angular accelerations, control surface deflections, and other nondimensional quantities. The functional dependencies can be quite complicated, so a variety of experiments are used to determine an adequate characterization.

There are also analytic methods to find the aerodynamic functional dependencies, such as computational fluid dynamics (CFD), which involves solving the partial differential equations governing the motion of the air about the aircraft; panel methods and strip theory, which involve dicing the aircraft into sections that approximate two-dimensional airfoil sections, then summing up lift, drag, and pitching moment from each section to get resultant forces and moments on the airplane; and U.S. Air Force DATCOM, which is an extensive set of rules of thumb for aerodynamic dependencies, based on experience and airplane geometry. Although these methods work well in some cases, typically for low angles of attack and low rotational rates, the best aerodynamic predictions are obtained using experimental methods. The experimental methods include wind-tunnel tests (static tests, forced oscillation tests, rotary balance tests, and spin tests), as well as flight-test measurements in steady and maneuvering flight.

Based on dimensional analysis, the nondimensional aerodynamic force and moment coefficients for rigid aircraft can be characterized as a function of nondimensional quantities as follows:

$$C_i = C_i\left(\alpha, \beta, \delta, \frac{\Omega l}{V}, \frac{\dot{\Omega}l^2}{V^2}, \frac{\dot{V}l}{V^2}, \frac{\omega l}{V}, \frac{\rho Vl}{\mu}, \frac{V^2}{lg}, \frac{V}{a}, \frac{m}{\rho l^3}, \frac{I}{\rho l^5}, \frac{tV}{l}\right)$$

for  $i = D, Y, L, l, m, n$ , where

$\delta$  = control surface deflections, deg or rad

$\Omega$  = stability axis rotation rate, rad/s

$l$  = characteristic length, ft or m

$\omega$  = oscillation frequency, rad/s

$\omega l/V \equiv Str$  = Strouhal number (unsteady oscillatory flow effects)

$\rho Vl/\mu = Vl/\nu \equiv Re$  = Reynolds number (fluid inertial forces/viscous forces)

$\nu \equiv \mu/\rho$  = kinematic viscosity,  $\text{ft}^2/\text{s}$  or  $\text{m}^2/\text{s}$

$V^2/gl \equiv Fr$  = Froude number (inertial forces/gravitational forces)

$V/a \equiv M$  = Mach number (fluid compressibility effects)

$a$  = speed of sound, ft/s or m/s

$t$  = time, s

Common simplifications are that the airplane mass and inertia are significantly larger than the surrounding air mass and inertia, fluid properties change slowly, and Froude number effects are small. In addition, the flow is often assumed to be quasi-steady, which means that the flowfield adjusts instantaneously to changes. One exception to this is the retention of Strouhal number effects, also called reduced frequency effects. These assumptions reduce the preceding relationship to

$$C_i = C_i \left( \alpha, \beta, \delta, \frac{\Omega l}{V}, \frac{\omega l}{V}, \frac{\rho Vl}{\mu}, \frac{V}{a} \right)$$

for  $i = D, Y, L, l, m, n$ .

In wind-tunnel testing, experimental characterization of the functional dependencies in the preceding equation is usually broken down as follows:

$$C_i = f_1 \left( \alpha, \beta, \delta, \frac{V}{a} \right) \text{ (static wind-tunnel tests)}$$

$$+ f_2 \left( \frac{\rho Vl}{\mu} \right) \text{ (Reynolds number corrections)}$$

$$+ f_3 \left( \frac{\Omega l}{V} \right) \text{ (rotary balance tests)}$$

$$+ f_4 \left( \frac{\omega l}{V} \right) \text{ (forced oscillation tests)}$$

for  $i = D, Y, L, l, m, n$ .

This decomposition assumes that the effects are separable and that superposition can be used—assumptions normally associated with linear and uncoupled functional dependencies. These assumptions are not always valid.

For full-scale aircraft flight testing, Reynolds number effects are not relevant, since this nondimensional quantity changes only slightly for the flight

## MATHEMATICAL MODEL OF AN AIRCRAFT

47

experiment. The effects of rotary motion and forced oscillation are usually modeled as a function of the body-axis angular rates, air incidence angles, and their first-time derivatives. For a full-scale conventional airplane in quasi-steady flow at low Mach number, the functional form for the nondimensional force and moment coefficients becomes

$$C_i = C_i \left( \frac{V}{V_o}, \alpha, \beta, \frac{pb}{2V}, \frac{q\bar{c}}{2V}, \frac{rb}{2V}, \frac{\dot{\alpha}\bar{c}}{2V}, \frac{\dot{\beta}b}{2V}, \delta \right)$$

for  $i = D, Y, L, l, m, n$ , where  $V_o$  is the airspeed at a reference condition, and  $\delta$  represents all the aircraft controls.

The variables inside the parentheses of the preceding equation are sometimes called independent variables, although in flight these variables cannot be changed independently. In general, the functional dependencies of the nondimensional force and moment coefficients on the independent variables are nonlinear. The airspeed and angular rates must be nondimensionalized as just shown, for dimensional consistency. The nondimensional quantities are also denoted by

$$\hat{V} = \frac{V}{V_o} \quad \hat{p} = \frac{pb}{2V} \quad \hat{q} = \frac{q\bar{c}}{2V} \quad \hat{r} = \frac{rb}{2V} \quad \hat{\alpha} = \frac{\dot{\alpha}\bar{c}}{2V} \quad \hat{\beta} = \frac{\dot{\beta}b}{2V}$$

In general, aerodynamic forces and moments are functionals of the state variables  $u, v, w, p, q, r$  or  $V, \alpha, \beta, p, q, r$ , and the control variables, denoted collectively by  $\delta$ . For a conventional, tailed airplane,  $\delta$  includes elevator, aileron, and rudder deflections that can change during the maneuver, as well as flap deflections and power settings, which generally are constant throughout a maneuver.

Considering only the angle of attack dependence of the aerodynamic force  $F_A$ ,

$$F_A(t) = F_A[\alpha(\tau)] \quad -\infty < \tau \leq t$$

The aerodynamic force depends on the instantaneous value of  $\alpha$ , but also on its entire history. In the majority of practical applications, the dependence on past values of  $\alpha$  can be neglected by considering the flow to be quasi-steady. This assumption presumes that the flow reaches a steady state instantaneously, so that dependence on the history of the explanatory variable is neglected. The aerodynamic model is then converted from a functional that depends on the entire history of  $\alpha$  into a function that depends only on the current value  $\alpha(t)$ . In practice, the aerodynamic forces and moments depend on multiple states and controls, not just  $\alpha$ , which makes the modeling much more complex.

In the following, a linear aerodynamic model for quasi-steady flow will be considered first. This approach leads to the concept of stability and control derivatives, also called aerodynamic derivatives. This concept will be extended to nonlinear aerodynamic models, formulated as polynomials or polynomial splines. Finally, the problem of modeling aerodynamics in unsteady flow conditions will be briefly addressed. In this case, the dependence of the aerodynamic forces and moments on past values of the explanatory variables must be restored.

### 3.8.1 Quasi-Steady Flow

*Linear model.* The form of aerodynamic model equations based on quasi-steady flow is given by a linear Taylor series expansion of the aerodynamic forces and moments about a reference condition, e.g.,

$$X = X_o + X_u \Delta u + X_v \Delta v + \cdots + X_\delta \Delta \delta \quad (3.55)$$

and similarly for  $Y$ ,  $Z$ ,  $L$ ,  $M$ , and  $N$ . The expansion can also be done in terms of  $V$ ,  $\alpha$ ,  $\beta$  instead of  $u$ ,  $v$ ,  $w$ . In Eq. (3.55),  $X_o$  represents the force component at the reference condition, which is usually specified as steady symmetric flight with  $v = p = q = r = 0$ . The quantities

$$X_u = \left. \frac{\partial X}{\partial u} \right|_o, \quad X_v = \left. \frac{\partial X}{\partial v} \right|_o, \quad \dots$$

are the partial derivatives of  $X$  with respect to  $u$ ,  $v$ ,  $\dots$ ,  $\delta$ , evaluated at the reference condition. Equation (3.55) is a Taylor series approximation, truncated to retain only the linear terms. The approximation is only valid for small perturbations in the explanatory states and controls, relative to their values at the reference condition. For these small perturbations, the omitted higher-order terms involving multiplication of small perturbation quantities are negligible compared with the first-order terms. The model parameters  $X_u$ ,  $X_v$ ,  $\dots$ ,  $X_\delta$ , etc., are constants associated with the reference condition. Strictly speaking, the reference flight condition should be specified by reference values of all the independent variables shown in the lengthy functional dependence derived from dimensional analysis, shown earlier. In practice, a reference flight condition of straight and level trimmed flight is specified by reference angle of attack  $\alpha_o$ , Mach number  $M_o$  or airspeed  $V_o$ , altitude  $h_o$ , mass  $m_o$ , and inertia properties  $I_o$ . For a steady turn, add bank angle  $\phi_o$ ; for steady climbing or descending flight, add path angle  $\gamma_o$ .

Based on vehicle symmetry and experience, the model equations like (3.55) can be simplified by neglecting 1) dependence of symmetric (longitudinal) forces and moment  $X$ ,  $Z$ ,  $M$ , on asymmetric (lateral) variables  $v$ ,  $p$ ,  $r$ ; and 2) dependence of asymmetric (lateral) force and moments  $Y$ ,  $L$ ,  $N$ , on symmetric (longitudinal) variables  $u$ ,  $w$ ,  $q$ .

The simplified equations are then augmented by adding two terms  $Z_{\dot{w}} \Delta \dot{w}$  and  $M_{\dot{w}} \Delta \dot{w}$ , or equivalently  $Z_{\dot{\alpha}} \Delta \dot{\alpha}$  and  $M_{\dot{\alpha}} \Delta \dot{\alpha}$ . These terms are necessary for obtaining closer correlation between predicted and observed aircraft longitudinal motion, as first mentioned in Ref. 5. For aircraft with a conventional wing and tail configuration, these terms can be approximated by the change in the vertical aerodynamic force and pitching moment of the tail, resulting from the lag in the downwash from the wing.<sup>1</sup> In general, the parameters  $Z_{\dot{w}}$  and  $M_{\dot{w}}$ , or  $Z_{\dot{\alpha}}$  and  $M_{\dot{\alpha}}$ , can be determined on the basis of unsteady aerodynamic modeling, as explained as follows.

Simplified aerodynamic models can be further augmented by adding three terms:  $Y_{\dot{v}} \Delta \dot{v}$ ,  $L_{\dot{v}} \Delta \dot{v}$ , and  $N_{\dot{v}} \Delta \dot{v}$ , or equivalently  $Y_{\dot{\beta}} \Delta \dot{\beta}$ ,  $L_{\dot{\beta}} \Delta \dot{\beta}$ , and  $N_{\dot{\beta}} \Delta \dot{\beta}$ . This time, the coefficients associated with each term can be determined only on the

## MATHEMATICAL MODEL OF AN AIRCRAFT

49

basis of unsteady aerodynamics. The augmentation by  $\dot{v}$  or  $\dot{\beta}$  terms will not be considered in equations describing quasi-steady flow.

To maintain consistency in the aerodynamic model equations,  $Z_{\dot{w}}$  and  $M_{\dot{w}}$  are interpreted as

$$Z_{\dot{w}} = \frac{\partial Z}{\partial \dot{w}} \Big|_o \quad \text{and} \quad M_{\dot{w}} = \frac{\partial M}{\partial \dot{w}} \Big|_o \quad (3.56)$$

and similarly for  $Z_{\dot{\alpha}}$  and  $M_{\dot{\alpha}}$ . This interpretation is mathematically incorrect, since neither  $\dot{w}$  nor  $\dot{\alpha}$  can be varied independently of  $w$  or  $\alpha$ . However, use of the term “derivative” for the parameters  $Z_{\dot{w}}$  and  $M_{\dot{w}}$ , or  $Z_{\dot{\alpha}}$  and  $M_{\dot{\alpha}}$ , is fully embedded in the flight dynamics literature. It will also appear in this book.

Considering the simplifications and term additions just mentioned, the linear aerodynamic model equations can be written as follows:

*Longitudinal:*

$$X = X_o + X_u \Delta u + X_w \Delta w + X_q q + X_\delta \Delta \delta \quad (3.57a)$$

$$Z = Z_o + Z_u \Delta u + Z_w \Delta w + Z_{\dot{w}} \Delta \dot{w} + Z_q q + Z_\delta \Delta \delta \quad (3.57b)$$

$$M = M_o + M_u \Delta u + M_w \Delta w + M_{\dot{w}} \Delta \dot{w} + M_q q + M_\delta \Delta \delta \quad (3.57c)$$

*Lateral:*

$$Y = Y_o + Y_v \Delta v + Y_p p + Y_r r + Y_\delta \Delta \delta \quad (3.58a)$$

$$L = L_o + L_v \Delta v + L_p p + L_r r + L_\delta \Delta \delta \quad (3.58b)$$

$$N = N_o + N_v \Delta v + N_p p + N_r r + N_\delta \Delta \delta \quad (3.58c)$$

The angular rate perturbations do not have the  $\Delta$  notation, because the angular rates at the assumed reference condition are zero, so the angular rates and their perturbation values are identical. For state variables  $V$ ,  $\alpha$ ,  $\beta$ ,  $p$ ,  $q$ ,  $r$ , and the aerodynamic force components in terms of lift  $L$  and drag  $D$ , the equations change as follows:

*Longitudinal:*

$$D = D_o + D_V \Delta V + D_\alpha \Delta \alpha + D_q q + D_\delta \Delta \delta \quad (3.59a)$$

$$L = L_o + L_V \Delta V + L_\alpha \Delta \alpha + L_{\dot{\alpha}} \Delta \dot{\alpha} + L_q q + L_\delta \Delta \delta \quad (3.59b)$$

$$M = M_o + M_V \Delta V + M_\alpha \Delta \alpha + M_{\dot{\alpha}} \Delta \dot{\alpha} + M_q q + M_\delta \Delta \delta \quad (3.59c)$$

*Lateral:*

$$Y = Y_o + Y_\beta \Delta \beta + Y_p p + Y_r r + Y_\delta \Delta \delta \quad (3.60a)$$

$$L = L_o + L_\beta \Delta \beta + L_p p + L_r r + L_\delta \Delta \delta \quad (3.60b)$$

$$N = N_o + N_\beta \Delta \beta + N_p p + N_r r + N_\delta \Delta \delta \quad (3.60c)$$

The model parameters representing partial derivatives in Eqs. (3.57)–(3.60) are called dimensional derivatives. These can be expressed in terms of partial derivatives of the nondimensional aerodynamic coefficients. For example, in Eq. (3.59b),

$$\begin{aligned} L_V &= \frac{\partial}{\partial V} \left( \frac{1}{2} \rho V^2 S C_L \right) \Big|_o = 2\rho_o V_o S C_{L_o} + \frac{1}{2} \rho_o V_o^2 S \frac{\partial C_L}{\partial V} \Big|_o \\ L_\alpha &= \frac{1}{2} \rho_o V_o^2 S \frac{\partial C_L}{\partial \alpha} \Big|_o \quad L_{\dot{\alpha}} = \frac{1}{2} \rho_o V_o^2 S \frac{\partial C_L}{\partial \dot{\alpha}} \Big|_o \\ L_q &= \frac{1}{2} \rho_o V_o^2 S \frac{\partial C_L}{\partial q} \Big|_o \quad L_\delta = \frac{1}{2} \rho_o V_o^2 S \frac{\partial C_L}{\partial \delta} \Big|_o \end{aligned} \quad (3.61)$$

and similarly for the remaining derivatives in Eqs. (3.57)–(3.60). From Eq. (3.61) it is apparent that the dimensional derivatives depend on the reference airspeed  $V_o$  and the air density  $\rho_o$ , which changes with atmospheric conditions and altitude.

For system identification applied to aircraft, it is more convenient to use non-dimensional derivatives of the nondimensional aerodynamic force and moment coefficients  $C_D$  or  $C_X, C_Y, C_L$  or  $C_Z, C_l, C_m$ , and  $C_n$ . This removes the known dependence on the airspeed and air density (dynamic pressure), and normalizes the partial derivatives. These derivatives are obtained from the following relationships:

$$\begin{aligned} C_X &= C_X(u, w, q, \delta) \\ C_a &= C_a(u, w, \dot{w}, q, \delta) \quad \text{for } a = Z \text{ or } m \end{aligned} \quad (3.62)$$

or

$$\begin{aligned} C_D &= C_D(V, \alpha, q, \delta) \\ C_a &= C_a(V, \alpha, \dot{\alpha}, q, \delta) \quad \text{for } a = L \text{ or } m \end{aligned} \quad (3.63)$$

and

$$C_a = C_a(\beta, p, r, \delta) \quad \text{for } a = Y, l, \text{ or } n \quad (3.64)$$

Then

$$\begin{aligned} C_X &= C_{X_o} + C_{X_u} \frac{\Delta u}{u_o} + C_{X_w} \frac{\Delta w}{u_o} + C_{X_q} \frac{q \bar{c}}{2u_o} + C_{X_\delta} \Delta \delta \\ C_a &= C_{a_o} + C_{a_u} \frac{\Delta u}{u_o} + C_{a_w} \frac{\Delta w}{u_o} + C_{a_{\dot{w}}} \left[ \frac{\dot{w}}{u_o} \right] \frac{\bar{c}}{2u_o} + C_{a_q} \frac{q \bar{c}}{2u_o} + C_{a_\delta} \Delta \delta \\ &\quad \text{for } a = Z \text{ or } m \end{aligned} \quad (3.65)$$

## MATHEMATICAL MODEL OF AN AIRCRAFT

51

or

$$C_D = C_{D_o} + C_{D_V} \frac{\Delta V}{V_o} + C_{D_\alpha} \Delta \alpha + C_{D_q} \frac{q\bar{c}}{2V_o} + C_{D_\delta} \Delta \delta$$

$$C_a = C_{a_o} + C_{a_V} \frac{\Delta V}{V_o} + C_{a_\alpha} \Delta \alpha + C_{a_\dot{\alpha}} \frac{\dot{\alpha}\bar{c}}{2V_o} + C_{a_q} \frac{q\bar{c}}{2V_o} + C_{a_\delta} \Delta \delta$$

for  $a = L$  or  $m$  (3.66)

and

$$C_a = C_{a_o} + C_{a_\beta} \Delta \beta + C_{a_p} \frac{pb}{2V_o} + C_{a_r} \frac{rb}{2V_o} + C_{a_\delta} \Delta \delta \quad \text{for } a = Y, l, \text{ or } n \quad (3.67)$$

In Eqs. (3.65) and (3.66),

$$C_{a_u} = u_o \frac{\partial C_a}{\partial u} \Big|_o \quad C_{a_w} = u_o \frac{\partial C_a}{\partial w} \Big|_o \quad C_{a_{\dot{w}}} = \frac{2u_o^2}{\bar{c}} \frac{\partial C_a}{\partial \dot{w}} \Big|_o$$

$$C_{a_q} = \frac{2u_o}{\bar{c}} \frac{\partial C_a}{\partial q} \Big|_o \quad C_{a_\delta} = \frac{\partial C_a}{\partial \delta} \Big|_o \quad \text{for } a = X, Z, \text{ or } m \quad (3.68)$$

$$C_{a_V} = V_o \frac{\partial C_a}{\partial V} \Big|_o \quad C_{a_\alpha} = \frac{\partial C_a}{\partial \alpha} \Big|_o \quad C_{a_{\dot{\alpha}}} = \frac{2V_o}{\bar{c}} \frac{\partial C_a}{\partial \dot{\alpha}} \Big|_o$$

$$C_{a_q} = \frac{2V_o}{\bar{c}} \frac{\partial C_a}{\partial q} \Big|_o \quad C_{a_\delta} = \frac{\partial C_a}{\partial \delta} \Big|_o \quad \text{for } a = D, L, \text{ or } m \quad (3.69)$$

Similarly, in Eq. (3.67)

$$C_{a_\beta} = \frac{\partial C_a}{\partial \beta} \Big|_o \quad C_{a_p} = \frac{2V_o}{b} \frac{\partial C_a}{\partial p} \Big|_o \quad C_{a_r} = \frac{2V_o}{b} \frac{\partial C_a}{\partial r} \Big|_o \quad C_{a_\delta} = \frac{\partial C_a}{\partial \delta} \Big|_o$$

for  $a = Y, l, \text{ or } n$  (3.70)

The quantities defined in Eqs. (3.68)–(3.70) are called nondimensional stability and control derivatives. Stability derivatives involve partial derivatives with respect to states; control derivatives involve partial derivatives with respect to controls. The stability derivatives are further divided into static stability derivatives for derivatives associated with air-relative velocity quantities ( $u, v, w, V, \alpha, \beta$ ), dynamic stability derivatives for derivatives associated with angular rates ( $p, q, r$ ), and derivatives associated with unsteady aerodynamics ( $\dot{w}, \dot{\alpha}$ ).

Note that the angular rates are nondimensionalized using  $u_o$  in Eq. (3.65) and  $V_o$  in Eq. (3.66). For maneuvers at low values of trim angle of attack, the difference is negligible, but at high angles of attack, the result is a slightly different definition of the parameters associated with  $\dot{\alpha}$  and  $q$  [see Eq. (3.32a)].

At very high speeds, e.g., hypersonic flight, a problem occurs with the non-dimensionalization of the angular rates. In that case,  $V_o$  and  $u_o$  are very large,

making the nondimensionalized rates very small. This can cause numerical problems in the parameter estimation, because of low sensitivity for these parameters (see Chapter 6). One solution is to identify dimensional parameters such as  $(\partial C_m / \partial q)|_o$ , then nondimensionalize the results using Eqs. (3.68) and (3.69) after the parameter estimation is finished.

If Eq. (3.66) is used for the model equations, an identifiability problem arises with the estimation of the  $\dot{\alpha}$  and  $q$  derivatives.<sup>5,6</sup> The problem is that for typical aircraft motion, the time histories for  $\dot{\alpha}$  and  $q$  are very similar, which leads to an indeterminacy in the modeling (see Chapter 5). To avoid this problem, the  $\dot{\alpha}$  and  $q$  terms can be lumped together, and a single equivalent derivative is then identified. The model equations would be

$$C_D = C_{D_o} + C_{D_V} \frac{\Delta V}{V_o} + C_{D_\alpha} \Delta \alpha + C_{D_q} \frac{q \bar{c}}{2V_o} + C_{D_\delta} \Delta \delta \quad (3.71)$$

$$C_a = C_{a_o} + C_{a_V} \frac{\Delta V}{V_o} + C_{a_\alpha} \Delta \alpha + \bar{C}_{a_q} \frac{q \bar{c}}{2V_o} + C_{a_\delta} \Delta \delta \quad \text{for } a = L \text{ or } m$$

where

$$\bar{C}_{a_q} = C_{a_{\dot{\alpha}}} + C_{a_q} \quad \text{for } a = L \text{ or } m \quad (3.72)$$

**Nonlinear model.** In many practical situations, the aerodynamic models in Eqs. (3.66) and (3.67) are good representations of the aerodynamic forces and moments. However, for large amplitudes or rapid excursions from the reference flight condition, it is necessary to extend the linear models by adding nonlinear terms. Taking the lift coefficient dependence on angle of attack and pitch rate as an example,  $C_L = C_L(\alpha, q)$ , and the Taylor series expansion can be written as

$$C_L = C_{L_o} + \frac{\partial C_L}{\partial \alpha} \Delta \alpha + \frac{\partial C_L}{\partial q} q + \frac{1}{2} \left[ \frac{\partial^2 C_L}{\partial \alpha^2} (\Delta \alpha)^2 + 2 \frac{\partial^2 C_L}{\partial \alpha \partial q} \Delta \alpha q + \frac{\partial^2 C_L}{\partial q^2} q^2 \right] + \dots \quad (3.73)$$

Introducing nonlinear aerodynamic derivatives,

$$C_L = C_{L_o} + C_{L_\alpha} \Delta \alpha + C_{L_q} \frac{q \bar{c}}{2V_o} + \frac{1}{2} \left[ C_{L_{\alpha^2}} (\Delta \alpha)^2 + 2 C_{L_{\alpha q}} \left( \Delta \alpha \frac{q \bar{c}}{2V_o} \right) + C_{L_{q^2}} \left( \frac{q \bar{c}}{2V_o} \right)^2 \right] + \dots \quad (3.74)$$

where the nonlinear derivatives are defined in a manner analogous to Eq. (3.69).

Another way to include nonlinear effects is to combine the static terms and treat the dynamic stability derivatives and control derivatives as functions of important explanatory variables, e.g., angle of attack. In the previous example,

## MATHEMATICAL MODEL OF AN AIRCRAFT

53

the nonlinear model would be

$$C_L = C_{L_o}(\alpha) + C_{Lq}(\alpha) \frac{q\bar{c}}{2V_o} \quad (3.75)$$

This approach is sometimes preferable in nonlinear modeling, because the nonlinear dependencies are partitioned differently and simplified. The idea can be generalized to all the aerodynamic coefficients. Common assumptions made are:

- 1) For subsonic flight, airspeed changes do not affect the aerodynamic coefficients.
- 2) The  $\dot{\alpha}$  contributions to  $C_L$  and  $C_m$  are included in the  $q$  terms.
- 3) The dependence of longitudinal and lateral coefficients on the states and controls are

$$\begin{aligned} C_a &= C_a(\alpha, \beta, q, \delta) \quad \text{for } a = D, L, \text{ or } m \\ C_a &= C_a(\alpha, \beta, p, r, \delta) \quad \text{for } a = Y, l, \text{ or } n \end{aligned} \quad (3.76)$$

- 4) The aerodynamic coefficients are modeled as the sum of a static term that includes nonlinear angle of attack and sideslip angle dependencies, and dynamic and control terms that are linear in  $p$ ,  $q$ ,  $r$ , and  $\delta$ . The second group of terms usually involves derivatives that depend nonlinearly on angle of attack, and sometimes also sideslip angle or Mach number.

Under the preceding assumptions, the aerodynamic model equations can be written as

$$C_a = C_{a_o}(\alpha, \beta)_{q=\delta=0} + \bar{C}_{aq}(\alpha) \frac{q\bar{c}}{2V_o} + C_{a\delta}(\alpha)\delta \quad \text{for } a = D, L, \text{ or } m \quad (3.77)$$

$$C_a = C_{a_o}(\alpha, \beta)_{p=r=\delta=0} + C_{ap}(\alpha) \frac{pb}{2V_o} + C_{ar}(\alpha) \frac{rb}{2V_o} + C_{a\delta}(\alpha)\delta$$

for  $a = Y, l, \text{ or } n$  (3.78)

These expressions for the aerodynamic coefficients are similar to those used in wind-tunnel testing. In that case, the functions shown would be implemented as tables of measured values. Note that the explanatory variables shown are typically not perturbations, because the nonlinear model formulation can be used to model the aerodynamic coefficients over the entire physical range of the explanatory variables. When this is not true, then perturbation quantities must again be used, and the model validity is localized.

The first terms on the right side of Eqs. (3.77) and (3.78) represent the static part, with controls fixed at zero deflection. The remaining terms represent contributions of dynamic stability derivatives, control derivatives, and their dependence on angle of attack.

Equations (3.77) and (3.78) are fairly general formulations of the aerodynamic forces and moments. The functional dependencies shown are based on wind-tunnel and flight-testing experience. In each particular case, however, the aerodynamic model equations should reflect any available *a priori* knowledge based on wind-tunnel experiments and/or theoretical aerodynamic calculations.

Each of the stability and control derivative functions in Eqs. (3.77) and (3.78) can be approximated by either polynomials or polynomial splines in the explanatory variables. For example,

$$C_{m_o}(\alpha, \beta)_{q=\delta=0} = \theta_{m_{o0}} + \theta_{m_{o1}}\alpha + \theta_{m_{o2}}\beta + \theta_{m_{o3}}\alpha\beta + \theta_{m_{o4}}\alpha^2$$

$$C_{l_p}(\alpha) = \theta_{l_{p0}} + \theta_{l_{p1}}\alpha \quad (3.79)$$

For large-amplitude maneuvers and flight at high angles of attack, the behavior of aerodynamic coefficients over different ranges of angle of attack may be different and totally unrelated. In these cases, the polynomial approximation for some aerodynamic nonlinearities can be inadequate. The choices then are to add more terms to the model or identify separate models for partitions of the independent variable space. Practically, the second option usually works better, because it is easier to solve several smaller and simpler subproblems compared to one large problem with a complicated model structure.

Polynomials can follow a curve in one interval but depart from that curve or oscillate elsewhere. Even if a high-order polynomial approximates the aerodynamic function sufficiently, the increase in the number of terms needed can lead to inaccurate parameter estimates from measured data.

To avoid the disadvantages of the polynomial representation, spline functions can be used. Splines remove some difficulties of polynomials because they are defined only on selected intervals, and low-order terms defined on limited intervals can approximate nonlinearities quite well.

Spline functions are defined as piecewise polynomials of a given degree<sup>7,8</sup> (see Appendix A). When continuity restrictions are included, the function values and derivatives agree at the points where the piecewise polynomials join. These points are called knots, and are defined by their location in the independent variable space.

As examples of using splines in formulating aerodynamic models, the lift coefficient  $C_L$  and the yawing moment coefficient  $C_n$  are considered. In the first case,

$$C_L = C_{L_o}(\alpha)_{q=\delta=0} + \bar{C}_{L_q}(\alpha) \frac{q\bar{c}}{2V_o} + C_{L_\delta}(\alpha)\delta \quad (3.80)$$

and the terms on the right side of Eq. (3.80) are approximated by

$$C_{L_o}(\alpha) = C_{L_o}(0) + C_{L_\alpha}\alpha + \sum_{i=1}^k D_{\alpha_i}(\alpha - \alpha_i)_+ \quad (3.81a)$$

$$\bar{C}_{L_q}(\alpha) = \bar{C}_{L_q}(0) + \bar{C}_{L_{q\alpha}}\alpha + \bar{C}_{L_{q\alpha^2}}\alpha^2 + \sum_{i=1}^k D_{q_i}(\alpha - \alpha_i)_+^2 \quad (3.81b)$$

$$C_{L_\delta}(\alpha) = C_{L_\delta}(0) + \sum_{i=1}^k D_{\delta_i}(\alpha - \alpha_i)_+^0 \quad (3.81c)$$

## MATHEMATICAL MODEL OF AN AIRCRAFT

55

where

$$(\alpha - \alpha_i)_+^m = \begin{cases} (\alpha - \alpha_i)^m & \alpha \geq \alpha_i \\ 0 & \alpha < \alpha_i \end{cases} \quad (3.82)$$

and the  $\alpha_i$  are constant values of  $\alpha$ , which are the knots, and the  $D_{\alpha_i}$ ,  $D_{q_i}$ , and  $D_{\delta_i}$  are constant parameters that quantify each spline contribution, in a manner similar to stability and control derivatives.

Equations (3.81) indicate that  $C_{L_o}(\alpha)$  is approximated by piecewise linear polynomials (first-degree splines),  $C_{L_q}(\alpha)$  by piecewise quadratic polynomials (second-degree splines), and  $C_{L_\delta}(\alpha)$  by piecewise constants (zero-degree splines). These spline types are sketched in Fig. 3.5 for the same knots.

In the second example, the yawing moment coefficient dependence on  $\alpha$  and  $\beta$  only is considered. Using a two-dimensional spline,  $C_n(\alpha, \beta)$  can be

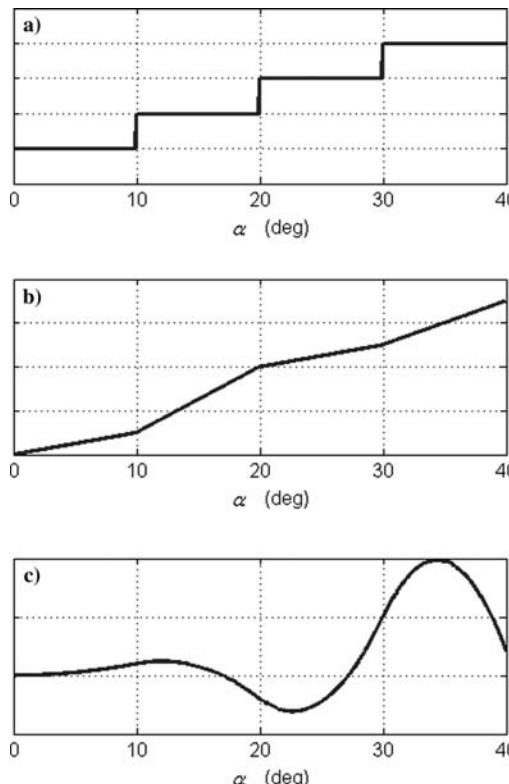


Fig. 3.5 Polynomial splines: a) zero degree, b) first degree, and c) second degree.

approximated as

$$C_n(\alpha, \beta) = C_{no} + C_{n\beta}\beta + \sum_{i=1}^k (A_{0i} + A_{1i}\beta)(\alpha - \alpha_i)_+^0 + \sum_{j=1}^l B_{0j}\beta \left(1 - \frac{\beta_j}{|\beta|}\right)_+ + \sum_{i=1}^k \sum_{j=1}^l D_{ij}\beta \left(1 - \frac{\beta_j}{|\beta|}\right)_+ (\alpha - \alpha_i)_+^0 \quad (3.83)$$

where

$$\beta \left(1 - \frac{\beta_j}{|\beta|}\right)_+ = \begin{cases} 0 & |\beta| \leq \beta_j \\ \beta - \beta_j & \beta > \beta_j \\ \beta + \beta_j & \beta < -\beta_j \end{cases} \quad (3.84)$$

and  $A_0$ ,  $A_{1i}$ ,  $B_{0j}$ , and  $D_{ij}$  are constant parameters. The knots  $\beta_j$  are always positive values, and  $C_n$  is an odd function of  $\beta$ . Equation (3.83) is of the same general form as Eq. (3.79), where the polynomials have been replaced by splines.

### 3.8.2 Unsteady Flow

In the preceding section, it was shown how the aerodynamic forces and moments acting on an aircraft in arbitrary motion can be approximated by linear terms, polynomials, or polynomial splines. It was assumed that the parameters appearing in these approximations were time invariant and that only the current values of the explanatory variables were required. These assumptions, however, have been questioned many times, based on studies of unsteady aerodynamics that go back to the 1920s.<sup>9</sup> A formulation of linear unsteady aerodynamics in the aircraft longitudinal equations in terms of indicial functions was introduced by Tobak.<sup>10</sup> Later, Tobak and Schiff<sup>11</sup> expressed the aerodynamic coefficients as functionals of the state and input variables. This very general approach includes linear unsteady aerodynamics as a special case.

Using results from Ref. 11, aircraft aerodynamic characteristics can be formulated as

$$C_a(t) = C_a(\infty) + \int_0^t C_{a\xi_1}[t - \tau; \xi(\tau)]^T \frac{d}{d\tau} \xi_1(\tau) d\tau + \frac{l}{V} \int_0^t C_{a\xi_2}[t - \tau; \xi(\tau)]^T \frac{d}{d\tau} \xi_2(\tau) d\tau \quad (3.85)$$

where

$$a = D, L, m, Y, l, \text{ or } n$$

$C_a(t)$  = aerodynamic force or moment coefficient

$C_a(\infty)$  = steady-state value of the aerodynamic force or moment coefficient

$C_{a\xi}(t)$  = vector of indicial functions; each element of the vector is the response of  $C_a$  to a unit step in an element of  $\xi$

$$\xi_1 = [\alpha \ \beta]^T; \quad \xi_2 = [p \ q \ r]^T$$

$$\xi = [\xi_1^T \ \xi_2^T]^T = [\alpha \ \beta \ p \ q \ r]^T$$

$$l = \text{characteristic length, } l = \bar{c}/2 \text{ or } l = b/2$$

## MATHEMATICAL MODEL OF AN AIRCRAFT

57

The indicial functions approach steady-state values as the argument  $(t - \tau)$  increases. To indicate this property, each indicial function can be expressed as

$$C_{a\xi_j}[t - \tau; \xi(\tau)] = C_{a\xi_j}[\infty; \xi(\tau)] - F_{a\xi_j}[t - \tau; \xi(\tau)] \quad (3.86)$$

where

$C_{a\xi_j}[\infty; \xi(\tau)]$  = steady-state rate of change of the coefficient  $C_a$  with respect to  $\xi_j$ , with the remaining variables in  $\xi$  fixed at the instantaneous values  $\xi(\tau)$

$F_{a\xi_j}[t - \tau; \xi(\tau)]$  = the deficiency function, which approaches zero as  $(t - \tau) \rightarrow \infty$

When Eq. (3.86) is substituted into Eq. (3.85), the terms involving the steady-state parameters can be integrated, and Eq. (3.85) becomes

$$\begin{aligned} C_a(t) &= C_a[\infty; \xi(t)] - \int_0^t F_{a\xi_1}[t - \tau; \xi(\tau)]^T \frac{d}{d\tau} \xi_1(\tau) d\tau \\ &\quad - \frac{l}{V} \int_0^t F_{a\xi_2}[t - \tau; \xi(\tau)]^T \frac{d}{d\tau} \xi_2(\tau) d\tau \end{aligned} \quad (3.87)$$

where

$C_a[\infty; \xi(t)]$  = the total aerodynamic coefficient for steady flow with  $\xi$  fixed at the instantaneous values  $\xi(t)$

$F_{a\xi_1}[t - \tau; \xi(\tau)]$  and  $F_{a\xi_2}[t - \tau; \xi(\tau)]$  = vector deficiency functions

Further simplification of Eq. (3.87) is achieved by assuming that the coefficients  $C_a$  are linearly dependent on the motion rates  $\xi_2$ . Applying a Taylor series expansion about  $\xi_2 = \mathbf{0}$  to the terms in Eq. (3.87), and keeping only the linear terms, results in

$$\begin{aligned} C_a(t) &= C_a[\infty; \xi_1(t), \mathbf{0}] - \int_0^t F_{a\xi_1}[t - \tau; \xi_1(\tau), \mathbf{0}]^T \frac{d}{d\tau} \xi_1(\tau) d\tau \\ &\quad - \frac{l}{V} C_{a\xi_2}[\infty; \xi_1(\tau), \mathbf{0}]^T \xi_2 \end{aligned} \quad (3.88)$$

or, using different notation,

$$\begin{aligned} C_a(t) &= C_a[\infty; \alpha(t), \beta(t)] + \frac{l}{V} C_{ap}[\infty; \alpha(t), \beta(t)] p(t) \\ &\quad + \frac{l}{V} C_{aq}[\infty; \alpha(t), \beta(t)] q(t) + \frac{l}{V} C_{ar}[\infty; \alpha(t), \beta(t)] r(t) \\ &\quad - \int_0^t F_{a\alpha}[t - \tau; \alpha(\tau), \beta(\tau)] \dot{\alpha}(\tau) d\tau \\ &\quad - \int_0^t F_{a\beta}[t - \tau; \alpha(\tau), \beta(\tau)] \dot{\beta}(\tau) d\tau \end{aligned} \quad (3.89)$$

If the indicial responses are only functions of elapsed time, and the steady flow coefficients are assumed to be independent of  $\alpha(t)$  and  $\beta(t)$ , Eq. (3.89) is simplified as

$$\begin{aligned} C_a(t) = & C_{a_0}(\infty) + C_{a_\alpha}(\infty)\alpha(t) + C_{a_\beta}(\infty)\beta(t) \\ & + \frac{l}{V} C_{ap}(\infty) p(t) + \frac{l}{V} C_{aq}(\infty) q(t) + \frac{l}{V} C_{ar}(\infty) r(t) \\ & - \int_0^t F_{a\alpha}(t-\tau) \dot{\alpha}(\tau) d\tau - \int_0^t F_{a\beta}(t-\tau) \dot{\beta}(\tau) d\tau \end{aligned} \quad (3.90)$$

where  $C_{a\alpha}(\infty)$ ,  $C_{a\beta}(\infty)$ , ...,  $C_{ar}(\infty)$  are constants representing the classical stability derivatives.

As an example, the equation for the pitching moment coefficient in terms of indicial functions is considered. Assuming linear aerodynamics with  $C_{m_0}(\infty) = 0$ , and neglecting the effects of  $\dot{q}(t)$  and lateral variables, the resulting equation follows from Eq. (3.90) as

$$C_m(t) = C_{m\alpha}(\infty)\alpha(t) + \frac{\bar{c}}{2V} C_{mq}(\infty)q(t) - \int_0^t F_{m\alpha}(t-\tau) \dot{\alpha}(\tau) d\tau \quad (3.91)$$

For fixed controls, the pitching moment is formulated as

$$C_m(t) = C_{m\alpha}\alpha(t) + \frac{\bar{c}}{2V} C_{mq}q(t) + \frac{\bar{c}}{2V} C_{m\dot{\alpha}}\dot{\alpha}(t) \quad (3.92)$$

It is therefore expected that the integral in Eq. (3.91) should be a counterpart of the term  $(\bar{c}/2V) C_{m\dot{\alpha}}\dot{\alpha}(t)$ . If the unsteady effect on  $C_m(t)$  is neglected, the integral in Eq. (3.91) can be reduced to a constant  $\dot{\alpha}$  term,

$$\int_0^t F_{m\alpha}(t-\tau) \dot{\alpha}(\tau) d\tau = \dot{\alpha}(t) \int_0^t F_{m\alpha}(\tau) d\tau \quad (3.93)$$

as demonstrated in Ref. 10 or 11. It follows that the counterpart of  $C_{m\dot{\alpha}}$  is proportional to the area under the deficiency function curve, i.e.,

$$C_{m\dot{\alpha}} = -\frac{2V}{\bar{c}} \int_0^\infty F_{m\alpha}(\tau) d\tau \quad (3.94)$$

When indicial functions are used in aerodynamic model equations, it is not clear, either from theory or experiment, what the analytical form of the indicial functions should be. In general, the indicial function represents a sum of two contributions. The first is a noncirculatory part that decays rapidly, and the second is a circulatory part that approaches the steady-state value of the indicial function with increased time.<sup>11</sup> Jones<sup>12</sup> developed approximation formulas for the lift indicial function for angle of attack on an elliptical wing with finite

## MATHEMATICAL MODEL OF AN AIRCRAFT

59

aspect ratio as

$$C_{L\alpha}(t) = a_1(1 - c_1 e^{-b_1 t}) \quad (3.95)$$

which leads to a deficiency function of the form

$$F_{L\alpha}(t) = a e^{-b_1 t} \quad (3.96)$$

For nonlinear aerodynamics, the deficiency function in Eq. (3.96) was generalized in Ref. 13 as

$$F_{L\alpha}(t; \alpha) = h(t; \alpha) a(\alpha) \quad (3.97)$$

where

$$h(t; \alpha) = \sum_{j=0}^m c_j e^{-b_j(\alpha)t} \quad (3.98)$$

and  $a(\alpha)$  and  $b(\alpha)$  are polynomials in  $\alpha$ .

The modeling of unsteady aerodynamics becomes more complicated for a wing/tail configuration, as discussed in Refs. 14–16.

Goman et al.<sup>17</sup> proposed a different approach to modeling unsteady aerodynamics using a concept of internal state variables. This approach retains the state-space formulation of aircraft dynamics,

$$\dot{\mathbf{x}} = \mathbf{f}[\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\theta}] \quad \mathbf{x}(0) = \mathbf{x}_o \quad (3.99)$$

by augmenting the aircraft states with the additional state variable  $\eta(t)$ . Then, the aerodynamic coefficients are formulated as

$$C_a(t) = C_a[\xi(t), \eta(t)] \quad (3.100)$$

where

$$\dot{\eta} = g[\eta(t), \xi(t), \dot{\xi}(t)] \quad (3.101)$$

and

$$\xi(t) = \begin{bmatrix} \mathbf{x}(t) \\ \mathbf{u}(t) \end{bmatrix} \quad (3.102)$$

Goman and Khrabrov<sup>18</sup> gave an example of Eq. (3.101) for aircraft longitudinal dynamics. There, the internal state variable represents the vortex burst point location along the chord of a triangular wing. This location is described by

$$T_1 \dot{\eta} + \eta = \eta_o(\alpha - T_\alpha \dot{\alpha}); \quad |\eta| \leq 1 \quad (3.103)$$

where

$\eta_o$  = vortex burst point location under steady conditions

$T_1$  = time constant in the vortical flow development

$T_\alpha$  = time lag in the vortical flow development caused by the angle of attack rate of change

Klein and Noderer<sup>19</sup> showed that in certain cases of linear aerodynamics, the formulations using either indicial functions or internal state variables lead to identical models.

The first example of aerodynamic model equations with unsteady terms used in aircraft parameter estimation was given by Goman et al.<sup>17</sup> and later by Goman and Khrabov.<sup>18</sup> Klein and Murphy,<sup>13</sup> Abramov et al.,<sup>20</sup> and Murphy and Klein<sup>21</sup> gave examples using wind-tunnel data. Fishenberg<sup>22</sup> reported some results obtained from flight data.

### 3.9 Simplifying the Equations of Motion

The aircraft equations of motion in general form are a set of coupled nonlinear differential equations. However, for many applications, these equations can be simplified. The most commonly used simplification is obtained by linearizing the equations about a reference condition. When the reference condition is selected as steady, wings-level flight with no sideslip, the linearized equations decouple into two independent sets: one describing the longitudinal motion in the plane of symmetry ( $V, \alpha, q, \theta$ ), and the other describing the lateral motion out of the plane of symmetry ( $\beta, p, r, \phi$ ).

Linearized equations have been extensively and successfully used in stability and control analysis and also in system identification. There are three main reasons for the wide use of linearized models:

- 1) Many common flight motions can be described by small changes in linear and angular velocities from a reference condition.
- 2) The main aerodynamic effects are very well described by linear functions of state and control variables in many cases.
- 3) Practical stability analysis and control system design are based on linear dynamic models.

For some flight motions, such as spins, flight near the stall, and maneuvers involving large changes in amplitudes and/or high angular rates, linear models are usually inadequate. In these cases, nonlinear models must be used.

#### 3.9.1 Linearization

The nonlinear equations of motion can be linearized by applying small-disturbance theory. Each variable is assumed to be composed of two parts—a constant component associated with the steady reference condition, and a perturbation associated with the linear model. In the following development, the reference condition is chosen as steady, wings-level flight with no sideslip. Other reference flight conditions, such as a steady turn, can also be used, but the steady, wings-level flight condition is the most common. Two axis systems

## MATHEMATICAL MODEL OF AN AIRCRAFT

61

are used for the model equations: 1) wind axes for the force equations and 2) body axes for the moment equations.

Steady values for each variable are denoted by subscript  $o$ , and perturbations are denoted by the prefix  $\Delta$ ,

$$\begin{aligned}
 V &= V_o + \Delta V & \alpha &= \alpha_o + \Delta \alpha & \beta &= \beta_o + \Delta \beta \\
 p &= p_o + \Delta p & q &= q_o + \Delta q & r &= r_o + \Delta r \\
 \phi &= \phi_o + \Delta \phi & \theta &= \theta_o + \Delta \theta & \psi &= \psi_o + \Delta \psi \\
 C_{Dw} &= C_{Dw_o} + \Delta C_{Dw} & C_{Yw} &= C_{Yw_o} + \Delta C_{Yw} & C_L &= C_{Lo} + \Delta C_L \\
 C_l &= C_{lo} + \Delta C_l & C_m &= C_{mo} + \Delta C_m & C_n &= C_{no} + \Delta C_n \\
 \delta &= \delta_o + \Delta \delta
 \end{aligned} \tag{3.104}$$

In cases where the steady value is zero, the perturbation and the value of the total variable are the same, so the  $\Delta$  prefix will be dropped. Similarly, time derivatives do not have the  $\Delta$  prefix, because the time derivative of the total variable is the same as the time derivative of its perturbation component.

For steady, wings-level flight with no sideslip at a reference condition,

$$\beta_o = p_o = q_o = r_o = \phi_o = 0 \tag{3.105}$$

and all the perturbations are zero. The equations of motion (3.39) and (3.37) then reduce to

$$0 = -\frac{\bar{q}S}{m}C_{D_o} - g \sin \gamma_o + \frac{T_o}{m} \cos \alpha_o \tag{3.106a}$$

$$0 = -\frac{\bar{q}S}{mV_o}C_{L_o} + \frac{g}{V_o} \cos \gamma_o - \frac{T_o \sin \alpha_o}{mV_o} \tag{3.106b}$$

$$0 = C_{Y_o} \tag{3.106c}$$

$$0 = C_{l_o} = C_{m_o} = C_{n_o} \tag{3.106d}$$

where  $\gamma_o$  is the steady flight-path angle

$$\gamma_o = \theta_o - \alpha_o \tag{3.107}$$

and the kinematic equations (3.38) reduce to zero on both sides. Next, small perturbation theory is applied, by replacing all the variables in the nonlinear equations (3.39), (3.37), and (3.38) with expansions from Eq. (3.104), then subtracting steady-state equations (3.106), applying the small-angle approximations  $\sin x \approx x$ ,  $\cos x \approx 1$ , and  $\tan x \approx x$ , for  $x$  in radians, and dropping terms that are second order and higher in the perturbation quantities. In this case, all the angular perturbations are small angles. The following relationships, which come from the small-angle approximations, are useful in deriving the linearized equations:

$$\cos(\phi_o + \Delta \phi) \approx \cos \phi_o - \sin \phi_o \Delta \phi \tag{3.108a}$$

$$\sin(\phi_o + \Delta \phi) \approx \sin \phi_o + \cos \phi_o \Delta \phi \tag{3.108b}$$

$$\tan(\theta_o + \Delta \theta) \approx \tan \theta_o + \Delta \theta \tag{3.108c}$$

The airspeed dependence is also simplified by assuming that  $V \approx V_o$  and  $\bar{q} \approx \bar{q}_o$ , which are good assumptions for airspeeds associated with fixed-wing aircraft, where  $V_o \gg \Delta V$ . The gyroscopic terms in the moment equations from the rotating mass of the propulsion system are usually dropped, because these terms are small for small angular rate perturbations. Similarly, the terms multiplied by  $\sin \beta/V$  in Eq. (3.39c) are also dropped, because their contribution is usually small. The resulting linearized equations are

$$\dot{V} = -\frac{\bar{q}_o S}{m} \Delta C_D - g \cos \gamma_o (\Delta \theta - \Delta \alpha) - \frac{T_o \sin \alpha_o}{m} \Delta \alpha + \frac{\cos \alpha_o}{m} \Delta T \quad (3.109a)$$

$$\dot{\alpha} = -\frac{\bar{q}_o S}{m V_o} \Delta C_L + q - \frac{g \sin \gamma_o}{V_o} (\Delta \theta - \Delta \alpha) - \frac{T_o \cos \alpha_o}{m V_o} \Delta \alpha - \frac{\sin \alpha_o}{m V_o} \Delta T \quad (3.109b)$$

$$\dot{\beta} = \frac{\bar{q}_o S}{m V_o} \Delta C_Y + p \sin \alpha_o - r \cos \alpha_o + \frac{g \cos \theta_o}{V_o} \phi \quad (3.109c)$$

$$\dot{p} - \frac{I_{xz}}{I_x} \dot{r} = \frac{\bar{q}_o S b}{I_x} \Delta C_l \quad (3.109d)$$

$$\dot{q} = \frac{\bar{q}_o S c}{I_y} \Delta C_m \quad (3.109e)$$

$$\dot{r} - \frac{I_{xz}}{I_z} \dot{p} = \frac{\bar{q}_o S b}{I_z} \Delta C_n \quad (3.109f)$$

$$\dot{\phi} = p + \tan \theta_o r \quad (3.109g)$$

$$\dot{\theta} = q \quad (3.109h)$$

$$\dot{\psi} = \sec \theta_o r \quad (3.109i)$$

The perturbations in the nondimensional aerodynamic force and moment coefficients  $\Delta C_D$ ,  $\Delta C_L$ ,  $\Delta C_Y$ ,  $\Delta C_l$ ,  $\Delta C_m$ , and  $\Delta C_n$  are replaced by the linear expansions in Eqs. (3.66) and (3.67), excluding the constant terms with subscript  $o$ , which were subtracted with the trim equations (3.106).

Linear expansions in terms of propulsion derivatives for  $\Delta T$  can be found in Etkin<sup>1</sup> for various propulsion types. However, flight testing for aerodynamic parameter estimation is usually done at constant power setting for each maneuver, which means the  $\Delta T$  term is zero. To estimate the effects of thrust or power level on aerodynamic parameters, several maneuvers can be run at different power settings to collect the required flight data. In this sense, the power setting is treated like a flight condition parameter, similar to Mach number. The power setting can affect the aerodynamic force and moment coefficients, due to changes in the flowfield caused by flow entrainment from jet exhaust plumes or from propeller slipstreams washing over the aircraft.

## MATHEMATICAL MODEL OF AN AIRCRAFT

63

Assuming constant power setting,  $\Delta T = 0$ , and substituting linear expansions for the force and moment coefficient perturbations, Eqs. (3.109) can be divided into two decoupled subsets describing the longitudinal and lateral motion:

*Longitudinal equations:*

$$\begin{aligned}\dot{V} = & -\frac{\bar{q}_o S}{m} \left( C_{D_V} \frac{\Delta V}{V_o} + C_{D_\alpha} \Delta \alpha + C_{D_q} \frac{q \bar{c}}{2V_o} + C_{D_\delta} \Delta \delta \right) \Delta C_D \\ & - g \cos \gamma_o (\Delta \theta - \Delta \alpha) - \frac{T_o \sin \alpha_o}{m} \Delta \alpha\end{aligned}\quad (3.110a)$$

$$\begin{aligned}\dot{\alpha} = & -\frac{\bar{q}_o S}{m V_o} \left( C_{L_V} \frac{\Delta V}{V_o} + C_{L_\alpha} \Delta \alpha + C_{L_\alpha} \frac{\dot{\alpha} \bar{c}}{2V_o} + C_{L_q} \frac{q \bar{c}}{2V_o} + C_{L_\delta} \Delta \delta \right) \\ & + q - \frac{g \sin \gamma_o}{V_o} (\Delta \theta - \Delta \alpha) - \frac{T_o \cos \alpha_o}{m V_o} \Delta \alpha\end{aligned}\quad (3.110b)$$

$$\dot{q} = \frac{\bar{q}_o S \bar{c}}{I_y} \left( C_{m_V} \frac{\Delta V}{V_o} + C_{m_\alpha} \Delta \alpha + C_{m_\alpha} \frac{\dot{\alpha} \bar{c}}{2V_o} + C_{m_q} \frac{q \bar{c}}{2V_o} + C_{m_\delta} \Delta \delta \right) \quad (3.110c)$$

$$\dot{\theta} = q \quad (3.110d)$$

*Lateral equations:*

$$\begin{aligned}\dot{\beta} = & \frac{\bar{q}_o S}{m V_o} \left( C_{Y_\beta} \beta + C_{Y_p} \frac{pb}{2V_o} + C_{Y_r} \frac{rb}{2V_o} + C_{Y_\delta} \delta \right) \\ & + p \sin \alpha_o - r \cos \alpha_o + \frac{g \cos \theta_o}{V_o} \phi\end{aligned}\quad (3.111a)$$

$$\dot{p} - \frac{I_{xz}}{I_x} \dot{r} = \frac{\bar{q}_o S b}{I_x} \left( C_{l_\beta} \beta + C_{l_p} \frac{pb}{2V_o} + C_{l_r} \frac{rb}{2V_o} + C_{l_\delta} \delta \right) \quad (3.111b)$$

$$\dot{r} - \frac{I_{xz}}{I_z} \dot{p} = \frac{\bar{q}_o S b}{I_z} \left( C_{n_\beta} \beta + C_{n_p} \frac{pb}{2V_o} + C_{n_r} \frac{rb}{2V_o} + C_{n_\delta} \delta \right) \quad (3.111c)$$

$$\dot{\phi} = p + \tan \theta_o r \quad (3.111d)$$

$$\dot{\psi} = \sec \theta_o r \quad (3.111e)$$

The longitudinal variables  $\Delta V$ ,  $\Delta \alpha$ ,  $q$ , and  $\Delta \theta$  appear only in the longitudinal equations, and the lateral variables  $\Delta \beta$ ,  $p$ ,  $r$ ,  $\Delta \phi$ , and  $\Delta \psi$  appear only in the lateral equations. In addition, the longitudinal controls are assumed to affect only longitudinal forces and moments, and similarly for the lateral controls. Therefore, these two sets of equations are decoupled, and can be solved and analyzed separately. This simplification is important for aircraft system identification problems, as well as aircraft dynamic analysis and control system design.

The linear model states  $\Delta V$ ,  $\Delta\alpha$ ,  $\Delta\beta$ ,  $p$ ,  $q$ ,  $r$ ,  $\Delta\phi$ ,  $\Delta\theta$ , and  $\Delta\psi$  can also be outputs, so the linear output equations are simply

$$\begin{aligned}\Delta V &= \Delta V \\ \Delta\alpha &= \Delta\alpha \\ &\vdots \\ \Delta\psi &= \Delta\psi\end{aligned}\tag{3.112}$$

For translational accelerations in  $g$  units, the linear output equations follow from Eqs. (3.54), with constant power setting ( $\Delta T = 0$ ),

$$\Delta a_x = \frac{\bar{q}_o S}{mg} \left( 2C_{X_o} \frac{\Delta V}{V_o} + \Delta C_X \right) \tag{3.113a}$$

$$\Delta a_y = \frac{\bar{q} S}{mg} \Delta C_Y \tag{3.113b}$$

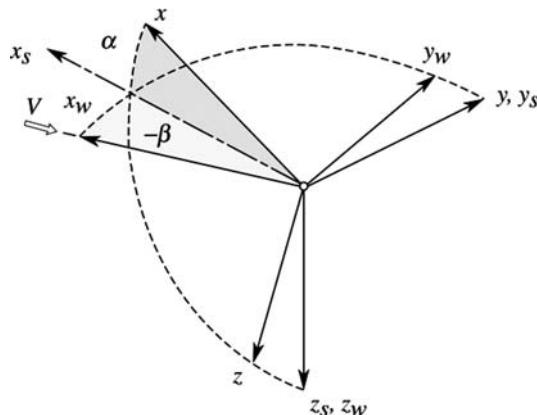
$$\Delta a_z = \frac{\bar{q}_o S}{mg} \left( 2C_{Z_o} \frac{\Delta V}{V_o} + \Delta C_Z \right) \tag{3.113c}$$

The terms involving  $C_{X_o}$  and  $C_{Z_o}$  come from the dependence of the translational accelerations on  $\bar{q}$ . A similar term does not appear in the  $\Delta a_y$  equation because  $C_{Y_o} = 0$ .

Equations (3.40a) and (3.40b) can be solved for  $C_X$  and  $C_Z$  in terms of  $C_D$  and  $C_L$ , or the expressions for  $C_X$  and  $C_Z$  in terms of  $C_D$  and  $C_L$  can be determined by trigonometry from Fig. 3.6 to be

$$C_X = -C_D \cos \alpha + C_L \sin \alpha \tag{3.114a}$$

$$C_Z = -C_L \cos \alpha - C_D \sin \alpha \tag{3.114b}$$



**Fig. 3.6 Rotation from wind axes to stability axes and body axes.**

## MATHEMATICAL MODEL OF AN AIRCRAFT

65

The perturbations  $\Delta C_X$  and  $\Delta C_Z$  are obtained from Eq. (3.114) as

$$\Delta C_X = (C_{D_o} \sin \alpha_o + C_{L_o} \cos \alpha_o) \Delta \alpha - \cos \alpha_o \Delta C_D + \sin \alpha_o \Delta C_L \quad (3.115a)$$

$$\Delta C_Z = (C_{L_o} \sin \alpha_o - C_{D_o} \cos \alpha_o) \Delta \alpha - \cos \alpha_o \Delta C_L - \sin \alpha_o \Delta C_D \quad (3.115b)$$

When output equations (3.113a) and (3.113c) are used with the longitudinal equations (3.110), the expressions for  $\Delta a_x$  and  $\Delta a_z$  in Eqs. (3.113a) and (3.113c) must be modified by substituting for  $\Delta C_X$  and  $\Delta C_Z$  from Eqs. (3.115), and replacing  $\Delta C_D$  and  $\Delta C_L$  with their linear expansions, specified in Eq. (3.66).

The linearized output equation for lateral acceleration measured in  $g$  units comes from combining Eq. (3.113b) and Eq. (3.67),

$$\Delta a_y = \frac{\bar{q}S}{mg} \left( C_{Y\beta} \beta + C_{Yp} \frac{pb}{2V_o} + C_{Yr} \frac{rb}{2V_o} + C_{Y\delta} \delta \right) \quad (3.116)$$

Further simplification of the linearized longitudinal equations can be achieved using the short-period approximation, which is a low-order linear dynamic model that accurately describes the short-period response of the aircraft. Most longitudinal flight maneuvers of interest for stability and control purposes occur over relatively short time periods, and therefore involve the short-period response.

The short-period approximation is obtained by assuming  $\Delta V = 0$ , and dropping the drag force equation. The short-period dynamic equations can be further simplified in practice by introducing the additional assumptions: 1)  $\gamma_o = 0$  for level flight at the reference condition, 2)  $[(T_o \cos \alpha_o)/mV_o] \Delta \alpha$  is relatively small and can be neglected, 3)  $C_L = C_L(\alpha, q, \delta)$ , and 4)  $C_m = C_m(\alpha, \dot{\alpha}, q, \delta)$ .

The short-period linear model equations then have the form

$$\dot{\alpha} = q - \frac{\bar{q}_o S}{mV_o} \left( C_{L\alpha} \Delta \alpha + C_{Lq} \frac{q \bar{c}}{2V_o} + C_{L\delta} \Delta \delta \right) \quad (3.117a)$$

$$\dot{q} = \frac{\bar{q}_o S \bar{c}}{I_y} \left( C_{m\alpha} \Delta \alpha + C_{mq} \frac{\dot{\alpha} \bar{c}}{2V_o} + C_{m\delta} \frac{q \bar{c}}{2V_o} + C_{m\dot{\alpha}} \Delta \delta \right) \quad (3.117b)$$

From Eq. (3.117a),  $\dot{\alpha}$  is a linear combination of  $\Delta \alpha$ ,  $q$ , and  $\Delta \delta$ . Substituting Eq. (3.117a) into Eq. (3.117b) yields

$$\dot{q} = \frac{\bar{q}_o S \bar{c}}{I_y} \left( C'_{m\alpha} \Delta \alpha + C'_{mq} \frac{q \bar{c}}{2V_o} + C'_{m\delta} \Delta \delta \right) \quad (3.118)$$

where

$$C'_{m\alpha} = C_{m\alpha} - \frac{\rho S \bar{c}}{4m} C_{m\dot{\alpha}} C_{L\alpha} \quad (3.119a)$$

$$C'_{mq} = C_{mq} + C_{m\dot{\alpha}} \left( 1 - \frac{\rho S \bar{c}}{4m} C_{Lq} \right) \quad (3.119b)$$

$$C'_{m\delta} = C_{m\delta} - \frac{\rho S \bar{c}}{4m} C_{m\dot{\alpha}} C_{L\delta} \quad (3.119c)$$

The linearized output equations for the states of the short-period model have the form [cf. Eq. (3.112)]

$$\begin{aligned} \Delta\alpha &= \Delta\alpha \\ q &= q \end{aligned} \quad (3.120)$$

Since the drag equation has been dropped, the  $\Delta a_x$  output equation is also omitted. For the vertical acceleration output, the linearized equation comes from Eq. (3.113c) with  $\Delta V = 0$ ,

$$\Delta a_z = \frac{\bar{q}_o S}{mg} \Delta C_Z \quad (3.121)$$

Dropping the  $\Delta C_D$  term from Eq. (3.115b) and substituting into Eq. (3.121) gives the linearized output equation for the vertical acceleration in the short-period approximation,

$$\Delta a_z = \frac{\bar{q}_o S}{mg} [(C_{L_o} \sin \alpha_o - C_{D_o} \cos \alpha_o) \Delta \alpha - \cos \alpha_o \Delta C_L] \quad (3.122)$$

Substituting the expression for  $\Delta C_L$  from inside the parentheses on the right side of Eq. (3.117a),

$$\begin{aligned} \Delta a_z &= \frac{\bar{q}_o S}{mg} \left[ (C_{L_o} \sin \alpha_o - C_{D_o} \cos \alpha_o) \Delta \alpha \right. \\ &\quad \left. - \cos \alpha_o \left( C_{L\alpha} \Delta \alpha + C_{Lq} \frac{q \bar{c}}{2V_o} + C_{L\delta} \Delta \delta \right) \right] \end{aligned} \quad (3.123)$$

For low values of trim angle of attack  $\alpha_o$ , the first  $\Delta \alpha$  term is small and  $\cos \alpha_o \approx 1$ , so that

$$\Delta a_z \approx -\frac{\bar{q}_o S}{mg} \left( C_{L\alpha} \Delta \alpha + C_{Lq} \frac{q \bar{c}}{2V_o} + C_{L\delta} \Delta \delta \right) \quad (3.124)$$

## MATHEMATICAL MODEL OF AN AIRCRAFT

67

Equations (3.117a), (3.118), and (3.124) are the linear equations for the short-period approximation,

$$\dot{\alpha} = q - \frac{\bar{q}_o S}{m V_o} \left( C_{L\alpha} \Delta\alpha + C_{Lq} \frac{q\bar{c}}{2V_o} + C_{L\delta} \Delta\delta \right) \quad (3.125a)$$

$$\dot{q} = \frac{\bar{q}_o S \bar{c}}{I_y} \left( C'_{m\alpha} \Delta\alpha + C'_{mq} \frac{q\bar{c}}{2V_o} + C'_{m\delta} \Delta\delta \right) \quad (3.125b)$$

$$\Delta a_z = -\frac{\bar{q}_o S}{mg} \left( C_{L\alpha} \Delta\alpha + C_{Lq} \frac{q\bar{c}}{2V_o} + C_{L\delta} \Delta\delta \right) \quad (3.125c)$$

These equations are sometimes written in terms of dimensional stability and control derivatives,

$$\dot{\alpha} = -L_\alpha \Delta\alpha + (1 - L_q)q - L_\delta \Delta\delta \quad (3.126a)$$

$$\dot{q} = M_\alpha \Delta\alpha + M_q q + M_\delta \Delta\delta \quad (3.126b)$$

$$\Delta a_z = -\frac{V_o}{g} (L_\alpha \Delta\alpha + L_q q + L_\delta \Delta\delta) \quad (3.126c)$$

where

$$L_\alpha = \frac{\bar{q}_o S}{m V_o} C_{L\alpha} \quad L_q = \frac{\bar{q}_o S \bar{c}}{2m V_o^2} C_{Lq} \quad L_\delta = \frac{\bar{q}_o S}{m V_o} C_{L\delta} \quad (3.127a)$$

$$M_\alpha = \frac{\bar{q}_o S \bar{c}}{I_y} C'_{m\alpha} \quad M_q = \frac{\bar{q}_o S \bar{c}^2}{2V_o I_y} C'_{mq} \quad M_\delta = \frac{\bar{q}_o S \bar{c}}{I_y} C'_{m\delta} \quad (3.127b)$$

At low angles of attack,  $-L = Z \cos \alpha \approx Z$ , so that Eqs. (3.126) can also be written as

$$\dot{\alpha} = Z_\alpha \Delta\alpha + (1 + Z_q)q + Z_\delta \Delta\delta \quad (3.128a)$$

$$\dot{q} = M_\alpha \Delta\alpha + M_q q + M_\delta \Delta\delta \quad (3.128b)$$

$$\Delta a_z = \frac{V_o}{g} (Z_\alpha \Delta\alpha + Z_q q + Z_\delta \Delta\delta) \quad (3.128c)$$

where

$$Z_\alpha = \frac{\bar{q}_o S}{m V_o} C_{Z\alpha} \quad Z_q = \frac{\bar{q}_o S \bar{c}}{2m V_o^2} C_{Zq} \quad Z_\delta = \frac{\bar{q}_o S}{m V_o} C_{Z\delta} \quad (3.129a)$$

$$M_\alpha = \frac{\bar{q}_o S \bar{c}}{I_y} C'_{m\alpha} \quad M_q = \frac{\bar{q}_o S \bar{c}^2}{2V_o I_y} C'_{mq} \quad M_\delta = \frac{\bar{q}_o S \bar{c}}{I_y} C'_{m\delta} \quad (3.129b)$$

Using dimensional stability and control derivatives, the lateral linearized equations (3.111) and (3.116) can be written as

$$\dot{\beta} = Y_\beta \beta + (Y_p + \sin \alpha_o)p + (Y_r - \cos \alpha_o)r + \frac{g \cos \theta_o}{V_o} \phi + Y_\delta \delta \quad (3.130a)$$

$$\dot{p} - \frac{I_{xz}}{I_x} \dot{r} = L_\beta \beta + L_p p + L_r r + L_\delta \delta \quad (3.130b)$$

$$\dot{r} - \frac{I_{xz}}{I_z} \dot{p} = N_\beta \beta + N_p p + N_r r + N_\delta \delta \quad (3.130c)$$

$$\dot{\phi} = p + \tan \theta_o r \quad (3.130d)$$

$$\dot{\psi} = \sec \theta_o r \quad (3.130e)$$

$$\Delta a_y = \frac{V_o}{g} (Y_\beta \beta + Y_p p + Y_r r + Y_\delta \delta) \quad (3.130f)$$

where

$$Y_\beta = \frac{\bar{q}_o S}{m V_o} C_{Y_\beta} \quad Y_p = \frac{\bar{q}_o S b}{2 m V_o^2} C_{Y_p} \quad Y_r = \frac{\bar{q}_o S b}{2 m V_o^2} C_{Y_r} \quad Y_\delta = \frac{\bar{q}_o S}{m V_o} C_{Y_\delta} \quad (3.131a)$$

$$L_\beta = \frac{\bar{q}_o S b}{I_x} C_{l_\beta} \quad L_p = \frac{\bar{q}_o S b^2}{2 V_o I_x} C_{l_p} \quad L_r = \frac{\bar{q}_o S b^2}{2 V_o I_x} C_{l_r} \quad L_\delta = \frac{\bar{q}_o S b}{I_x} C_{l_\delta} \quad (3.131b)$$

$$N_\beta = \frac{\bar{q}_o S b}{I_z} C_{n_\beta} \quad N_p = \frac{\bar{q}_o S b^2}{2 V_o I_z} C_{n_p} \quad N_r = \frac{\bar{q}_o S b^2}{2 V_o I_z} C_{n_r} \quad N_\delta = \frac{\bar{q}_o S b}{I_z} C_{n_\delta} \quad (3.131c)$$

Linearized dynamic equations for translational motion of the aircraft can also be developed in body axes, using the same procedure, starting from the nonlinear equations (3.36). The result is

$$\Delta \dot{u} = -w_o q + \frac{\bar{q} S}{m} \Delta C_X - g \cos \theta_o \Delta \theta + \frac{\Delta T}{m} \quad (3.132a)$$

$$\Delta \dot{v} = w_o p - u_o r + \frac{\bar{q} S}{m} \Delta C_Y + g \cos \theta_o \phi \quad (3.132b)$$

$$\Delta \dot{w} = u_o q + \frac{\bar{q} S}{m} \Delta C_Z - g \sin \theta_o \Delta \theta \quad (3.132c)$$

## MATHEMATICAL MODEL OF AN AIRCRAFT

69

The linear output equations for air-data quantities normally measured on aircraft (i.e.,  $V$ ,  $\alpha$ , and  $\beta$ ) follow from Eqs. (3.46) with  $u_o \approx V_o$ ,

$$\Delta V = \Delta u + w_o \frac{\Delta w}{u_o} \quad (3.133a)$$

$$\Delta \alpha = \Delta w / u_o \quad (3.133b)$$

$$\Delta \beta = \Delta v / u_o \quad (3.133c)$$

The linear output equations for the accelerometers for  $\Delta T = 0$  are similar to those given in Eqs. (3.113),

$$\Delta a_x = \frac{\bar{q}_o S}{mg} \left\{ 2C_{X_o} \left[ \frac{\Delta u}{u_o} + \left( \frac{w_o}{u_o} \right) \frac{\Delta u}{u_o} \right] + \Delta C_X \right\} \quad (3.134a)$$

$$\Delta a_y = \frac{\bar{q}_o S}{mg} \Delta C_Y \quad (3.134b)$$

$$\Delta a_z = \frac{\bar{q}_o S}{mg} \left\{ 2C_{Z_o} \left[ \frac{\Delta u}{u_o} + \left( \frac{w_o}{u_o} \right) \frac{\Delta u}{u_o} \right] + \Delta C_Z \right\} \quad (3.134c)$$

Note that if the body axes used to derive Eqs. (3.132) are rotated into stability axes, then  $w_o = 0$ ,  $u = V$ ,  $\Delta \gamma = \Delta \theta$ ,  $\alpha_o = 0$ ,  $C_Z = -C_L$ , and  $C_X = -C_D$ . Making these substitutions in Eqs. (3.132) and using the approximations in Eqs. (3.133), results in the translational linear dynamic equations in stability axes, Eqs. (3.109a)–(3.109c).

### 3.9.2 Substituting Measured Values

Data used for system identification applied to aircraft come mostly from maneuvers that excite the longitudinal short-period dynamics or the lateral modes associated with body-axis roll, spiral motion, and lateral oscillations. For these maneuvers, the airspeed is expected to be constant, and the longitudinal and lateral motions are decoupled. In practice, the airspeed exhibits some changes from the reference value, and there are variations in the lateral quantities during a longitudinal maneuver, and vice versa. To keep the equations relevant to the performed maneuvers in a simple form, the dynamic pressure  $\bar{q}$ , airspeed  $V$ , and the variables that would cause coupling between the longitudinal and lateral motion can be replaced by their measured values. For longitudinal motion, dropping the drag force and kinematic equations, the longitudinal equations (3.39b) and (3.37b) are simplified by using measured values from the experiment,  $\bar{q}_E$ ,  $T_E$ ,  $V_E$ ,  $\alpha_E$ ,  $\theta_E$ ,  $\beta_E$ ,  $p_E$ ,  $r_E$ , and  $\phi_E$ , where  $\alpha_E$  and  $\theta_E$  are used only in the gravity and thrust terms. The

result is

$$\begin{aligned}\dot{\alpha} = & -\frac{\bar{q}_E S}{mV_E \cos \beta_E} C_L + q - \tan \beta_E (p_E \cos \alpha_E + r_E \sin \alpha_E) - \frac{T_E \sin \alpha_E}{mV_E \cos \beta_E} \\ & + \frac{g}{V_E \cos \beta_E} (\cos \phi_E \cos \theta_E \cos \alpha_E + \sin \theta_E \sin \alpha_E)\end{aligned}\quad (3.135a)$$

$$\dot{q} = \frac{\bar{q}_E S \bar{c}}{I_y} C_m - \frac{(I_x - I_z)}{I_y} p_E r_E - \frac{I_{xz}}{I_y} (p_E^2 - r_E^2) + \frac{I_p}{I_y} \Omega_p r_E \quad (3.135b)$$

Assuming measured sideslip angle  $\beta_E$  is small, the simplified longitudinal equations have the form

$$\begin{aligned}\dot{\alpha} = & -\frac{\bar{q}_E S}{mV_E} C_L + q - (p_E \cos \alpha_E + r_E \sin \alpha_E) \beta_E - \frac{T_E \sin \alpha_E}{mV_E} \\ & + \frac{g}{V_E} (\cos \phi_E \cos \theta_E \cos \alpha_E + \sin \theta_E \sin \alpha_E)\end{aligned}\quad (3.136a)$$

$$\dot{q} = \frac{\bar{q}_E S \bar{c}}{I_y} C_m - \frac{(I_x - I_z)}{I_y} p_E r_E - \frac{I_{xz}}{I_y} (p_E^2 - r_E^2) + \frac{I_p}{I_y} \Omega_p r_E \quad (3.136b)$$

Using measured values removes nonlinearities and also decouples the longitudinal equations from the lateral equations. Linear expansions from Eq. (3.71) are substituted for  $C_L$  and  $C_m$ . Note that this expansion must retain the constant terms  $C_{L_0}$  and  $C_{m_0}$ , because the measured quantities include steady trim values. Further simplifications to the equations can be made by dropping individual terms with small relative magnitudes for the specific maneuver being studied.

Using measured values, the linearized lateral equations are

$$\begin{aligned}\dot{\beta} = & \frac{\bar{q}_E S}{mV_E \cos \beta_E} C_{Y_W} + p \sin \alpha_E - r \cos \alpha_E + \frac{g}{V_E} \cos \beta_E \sin \phi_E \cos \theta_E \\ & + \frac{\sin \beta_E}{V_E} \left( g \cos \alpha_E \sin \theta_E - g \sin \alpha_E \cos \phi_E \cos \theta_E + \frac{T_E \cos \alpha_E}{m} \right)\end{aligned}\quad (3.137a)$$

$$\dot{p} - \frac{I_{xz}}{I_x} \dot{r} = \frac{\bar{q}_E S b}{I_x} C_l - \frac{(I_z - I_y)}{I_x} q_E r + \frac{I_{xz}}{I_x} q_E p \quad (3.137b)$$

$$\dot{r} - \frac{I_{xz}}{I_z} \dot{p} = \frac{\bar{q}_E S b}{I_z} C_n - \frac{(I_y - I_x)}{I_z} q_E p - \frac{I_{xz}}{I_z} q_E r - \frac{I_p}{I_z} \Omega_p q_E \quad (3.137c)$$

$$\dot{\phi} = p + \tan \theta_E (q_E \sin \phi_E + r \cos \phi_E) \quad (3.137d)$$

$$\dot{\psi} = \sec \theta_E r \quad (3.137e)$$

# MATHEMATICAL MODEL OF AN AIRCRAFT

71

Assuming sideslip angle  $\beta$  is small, so that  $\cos \beta \approx 1$ ,  $C_{Y_W} \approx C_Y$ , and  $\sin \beta/V_E \approx 0$ , the lateral equations are simplified as

$$\dot{\beta} = \frac{\bar{q}_E S}{mV_E} C_Y + p \sin \alpha_E - r \cos \alpha_E + \frac{g}{V_E} \sin \phi_E \cos \theta_E \quad (3.138a)$$

$$\dot{p} - \frac{I_{xz}}{I_x} \dot{r} = \frac{\bar{q}_E S b}{I_x} C_l - \frac{(I_z - I_y)}{I_x} q_{ER} + \frac{I_{xz}}{I_x} q_{EP} \quad (3.138b)$$

$$\dot{r} - \frac{I_{xz}}{I_z} \dot{p} = \frac{\bar{q}_E S b}{I_z} C_n - \frac{(I_y - I_x)}{I_z} q_{EP} - \frac{I_{xz}}{I_z} q_{ER} - \frac{I_p}{I_z} \Omega_p q_E \quad (3.138c)$$

$$\dot{\phi} = p + \tan \theta_E (q_E \sin \phi_E + r \cos \phi_E) \quad (3.138d)$$

$$\dot{\psi} = \sec \theta_E r \quad (3.138e)$$

The equations linearized in this way can be used for larger-amplitude maneuvers or significantly coupled maneuvers, because none of the nonlinear terms were dropped due to multiplications of small perturbation quantities. However, when measured values are substituted into the equations of motion, the accuracy of the solution will be affected by systematic and random errors in the measured quantities.

## 3.10 Summary and Concluding Remarks

In this chapter, the rigid-body equations of motion for an airplane were derived. The resulting coupled nonlinear ordinary differential equations include terms representing the aerodynamic forces and moments acting on the vehicle. Since the rigid-body equations of motion, also called the Euler equations, are well known, the focus of aircraft system identification is typically on obtaining good models for the dependencies of the aerodynamic forces and moments on aircraft states and controls. The equations developed for this purpose are called aerodynamic model equations. Several mathematical constructs were introduced for the aerodynamic model equations, including linear expansions, multivariate polynomials, polynomial splines, and indicial functions. Parameters in these mathematical models are the unknowns to be estimated based on measured data. Each model form has implications in terms of the relationships that can be described and the experimentation and measurements required for successful modeling. These issues will be explored further in the following chapters.

The full nonlinear equations of motion were then simplified using two methods: linearization for small motions about a reference condition, and substitution of measured values in the nonlinear equations to linearize them. The resulting sets of equations are linear with fewer equations and fewer model parameters, which simplifies the analysis.

The next chapter covers the theory required for estimating the unknown parameters in postulated models of the forms developed in this chapter. Chapters 5–8 discuss practical application of this theory to aircraft system identification problems.

## References

- <sup>1</sup>Etkin, B., *Dynamics of Atmospheric Flight—Stability and Control*, 2nd ed., Wiley, New York, 1982.
- <sup>2</sup>Etkin, B., and Reid, L. D., *Dynamics of Flight—Stability and Control*, 3rd ed., Wiley, New York, 1996.
- <sup>3</sup>McRuer, D., Ashkenas, I., and Graham, D., *Aircraft Dynamics and Automatic Control*, Princeton Univ. Press, Princeton, NJ, 1973.
- <sup>4</sup>Roskam, J., *Airplane Flight Dynamics and Automatic Flight Controls*, Pts. 1 and 2, Roskam Aviation and Engineering, Lawrence, KS, 1979.
- <sup>5</sup>Greenberg, H., "Determination of Stability Derivatives from Flight Data," *Journal of the Aeronautical Sciences*, Vol. 16, No. 1, 1949, p. 62.
- <sup>6</sup>Greenberg, H., "A Survey of Methods for Determining Stability Parameters of an Airplane from Dynamic Flight Measurements," NASA TN 2340, 1951.
- <sup>7</sup>Schumacher, L. L., *Spline Function Basic Theory*, Wiley, New York, 1981.
- <sup>8</sup>Klein, V., and Batterson, J. G., "Determination of Airplane Model Structure from Flight Data Using Splines and Stepwise Regression," NASA TP-2126, 1983.
- <sup>9</sup>Cowley, W. L., and Glauert, H., "The Effect of the Lag of the Downwash on the Longitudinal Stability of an Airplane and on the Rotary Derivative," Reports and Memoranda No. 718, February 1921.
- <sup>10</sup>Tobak, M., "On the Use of the Indicial Function Concept in the Analysis of Unsteady Motion of Wing and Wing-Tail Combinations," NACA Rep. 1188, 1954.
- <sup>11</sup>Tobak, M., and Schiff, L. B., "On the Formulation of the Aerodynamic Characteristics in Aircraft Dynamics," NASA TR R-456, 1976.
- <sup>12</sup>Jones, R. T., "The Unsteady Lift of a Wing of Finite Aspect Ratio," NACA Rep. 681, 1939.
- <sup>13</sup>Klein, V., and Murphy, P. C., "Estimation of Aircraft Nonlinear Unsteady Parameters from Wind Tunnel Data," NASA TM-1998-208969, 1998.
- <sup>14</sup>Jones, R. T., and Fehlner, L. F., "Transient Effects of the Wing Wake on the Horizontal Tail," NASA TN 771, 1940.
- <sup>15</sup>Klein, V., "Modeling of Longitudinal Unsteady Aerodynamics of a Wing-Tail Combination," NASA CR-1999-209547, 1999.
- <sup>16</sup>Khrabrov, A., Vinogradov, Y., and Abramov, N., "Mathematical Modelling of Aircraft Unsteady Aerodynamics at High Incidence with Account of Wing-Tail Interaction," *AIAA Atmospheric Flight Mechanics Conference*, AIAA Paper 2004-5278, 2004.
- <sup>17</sup>Goman, M. G., Stolyarov, G. J., Tartyshnikov, S. L., Usokev, S. P. and Khrabrov, A. N., "Mathematical Description of Aerodynamic Forces and Moments at Nonstationary Flow Regimes with a Nonunique Structure," *Proceedings of TsAGI*, Issue 2195, Moscow, Russia, 1983 (in Russian).
- <sup>18</sup>Goman, M. G., and Khrabrov, A. N., "State-Space Representation of Aerodynamic Characteristics of an Aircraft at High Angles of Attack," AIAA Paper 92-4651, 1992.
- <sup>19</sup>Klein, V., and Noderer, K. D., "Modeling of Aircraft Unsteady Aerodynamic Characteristics, Part 1—Postulated Models," NASA TM 109120, 1994.
- <sup>20</sup>Abramov, N., Goman, M., and Khrabrov, A., "Aircraft Dynamics at High Incidence Flight with Account of Unsteady Aerodynamic Effects," *AIAA Atmospheric Flight Mechanics Conference*, AIAA Paper 2004-5274, 2004.

## MATHEMATICAL MODEL OF AN AIRCRAFT

73

<sup>21</sup>Murphy, P. C., and Klein, V., "Estimation of Aircraft Unsteady Aerodynamic Parameters from Dynamic Wind Tunnel Testing," AIAA Paper 2001-4016, 2001.

<sup>22</sup>Fishenberg, D., "Identification of an Unsteady Aerodynamic Stall Model from Flight Test Data," AIAA Paper 95-3438, 1995.

This page intentionally left blank

## 4

## Outline of Estimation Theory

The parameter estimation process consists of finding values of unknown model parameters  $\boldsymbol{\theta}$  in an assumed model structure, based on noisy measurements  $z$ . An estimator is a function of the random variable  $z$  that produces an estimate  $\hat{\boldsymbol{\theta}}$  of the unknown parameters  $\boldsymbol{\theta}$ . Since the estimator computes  $\hat{\boldsymbol{\theta}}$  based on noisy measurements  $z$ ,  $\hat{\boldsymbol{\theta}}$  is a random variable.

Parameter estimation requires specification of the following:

- 1) A model structure with unknown parameters  $\boldsymbol{\theta}$  to be estimated;
- 2) Observations, or measurements,  $z$ ;
- 3) A mathematical model for the measurement process;
- 4) Assumptions about the uncertainty in the model parameters  $\boldsymbol{\theta}$  and the measurement noise  $\boldsymbol{v}$ .

In Chapter 2, a distinction was made between linear and nonlinear dynamic systems, based on the relation between state time derivatives and the state and control variables. For parameter estimation, however, the relation between the measured outputs and model parameters is of much greater importance.

A model is called linear in the parameters if the output  $y$  is given by

$$\mathbf{y} = \mathbf{H}\boldsymbol{\theta} \quad (4.1)$$

where the matrix  $\mathbf{H}$  is assumed to be known. Then the measurement equation can be expressed as

$$\mathbf{z} = \mathbf{H}\boldsymbol{\theta} + \boldsymbol{v} \quad (4.2)$$

A model that is nonlinear in the parameters has a measurement equation of the form

$$\mathbf{z} = \mathbf{h}(\boldsymbol{\theta}) + \boldsymbol{v} \quad (4.3)$$

where the form of the function  $\mathbf{h}(\boldsymbol{\theta})$  is assumed to be known.

In general, there are  $n_o$  measured outputs, and a vector of measurements is taken at each sample  $i$ , where  $i = 1, 2, \dots, N$ , and  $N$  is the number of sampled data points. In this chapter, a single measured output is assumed, so  $n_o = 1$ , and  $\mathbf{z}$  is a vector composed of  $N$  scalar measurements. The extension to multiple measured outputs is discussed in Chapters 6 and 7.

The notion of a linear or nonlinear dynamic system has no connection to the model being linear or nonlinear in the parameters, as can be seen from the following example.

### Example 4.1

Consider a linear, first-order, scalar system described as

$$\dot{x} + \theta x = u \quad (4.4)$$

with output equation

$$y = \dot{x} = -\theta x + u \quad (4.5)$$

and measurement equation

$$z = y + v = -\theta x + u + v \quad (4.6)$$

The output is linear in the parameter  $\theta$ . If instead the output equation is

$$y = x = \int_0^t e^{-\theta(t-\tau)} u(\tau) d\tau \quad (4.7)$$

so that

$$z = \int_0^t e^{-\theta(t-\tau)} u(\tau) d\tau + v \quad (4.8)$$

then the output is nonlinear in the parameter  $\theta$ . In this simple example, the same dynamic system exhibits measurement equations that are linear or nonlinear in the parameter, depending on how the model output is defined.  $\square$

The notion of linearity and nonlinearity in the parameters will be further clarified when the various parameter estimation methods are introduced. Models for uncertainty in  $\theta$  and  $v$  will be specified by probability density functions  $p(\theta)$  and  $p(v)$ , respectively.

Knowledge about the estimated parameters can be expressed in terms of the probability density function  $p(\hat{\theta}|z)$ , which represents the probability density of the parameter estimate  $\hat{\theta}$ , given the measurements  $z$ . This probability density function would be the most complete information that could be derived by applying statistical techniques. In practice, however, it can be quite difficult to find the solution to the estimation problem in the form of a probability density function, because the expected value and higher-order moments for a random variable vector would have to be estimated. For that reason, the solution is reduced

## OUTLINE OF ESTIMATION THEORY

77

from the probability density function  $p(\hat{\boldsymbol{\theta}}|z)$  to its most significant properties, which are as follows:

$$\text{expected value: } E(\hat{\boldsymbol{\theta}} | z)$$

$$\text{covariance: } E\left\{ \left[ \hat{\boldsymbol{\theta}} - E(\hat{\boldsymbol{\theta}} | z) \right] \left[ \hat{\boldsymbol{\theta}} - E(\hat{\boldsymbol{\theta}} | z) \right]^T \right\}$$

$$\text{bias: } E(\hat{\boldsymbol{\theta}} | z) - E(\boldsymbol{\theta} | z) \quad (4.9)$$

During the estimation process, an attempt is made to obtain a good estimate of  $\boldsymbol{\theta}$ . To achieve this, it is necessary to define what is meant by “good.” The main properties that are used to characterize the quality of an estimator are defined next. This is followed by the development of estimators for a system described by Eq. (4.2) or (4.3), and by three different models of the uncertainty in  $\boldsymbol{\theta}$  and  $\boldsymbol{v}$ . Then state estimation procedures for dynamic systems will be introduced and described.

#### 4.1 Properties of Estimators

The following definitions of estimator properties are presented without development or proof. Rigorous treatment of these definitions can be found in Refs. 1 and 2.

##### Definition 4.1

An estimator is linear if  $\hat{\boldsymbol{\theta}}$  is obtained as a linear function of measurements. If  $\hat{\boldsymbol{\theta}}$  is obtained as a nonlinear function of measurements, the estimator is nonlinear.

##### Definition 4.2

An estimator is unbiased if the expected value of  $\hat{\boldsymbol{\theta}}$  is equal to the expected value of  $\boldsymbol{\theta}$  for different sample sizes, i.e.,

$$E(\hat{\boldsymbol{\theta}}) = E(\boldsymbol{\theta}) \quad \text{for each } N \text{ and all } \boldsymbol{\theta} \quad (4.10)$$

##### Definition 4.3

An estimator is called a minimum mean square error estimator if it minimizes the mean square error (MSE):

$$\text{MSE} = E\left[ (\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})^T (\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}) \right] \quad (4.11)$$

Note that the MSE for an estimate  $\hat{\boldsymbol{\theta}}$  is equal to the trace of the corresponding error covariance matrix,

$$\text{MSE} = E[(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})^T(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})] = \text{Tr}\left\{E[(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})^T]\right\} \quad (4.12)$$

In general, the MSE includes both the variance (random) error and the squared bias (systematic) error,

$$\text{MSE} = \text{variance} + (\text{bias})^2$$

For an unbiased estimate  $\hat{\boldsymbol{\theta}}$ , a minimum mean square error estimator is a minimum variance estimator.

#### Definition 4.4

An estimator is called a best linear unbiased estimator of  $\boldsymbol{\theta}$  if it has minimum MSE among the class of unbiased estimators that are linear functions of the measurements.

#### Definition 4.5

The Fisher information matrix  $\mathbf{M}$  is defined as

$$\mathbf{M} \equiv E\left[\left(\frac{\partial \ell_n \mathbb{L}}{\partial \boldsymbol{\theta}}\right)\left(\frac{\partial \ell_n \mathbb{L}}{\partial \boldsymbol{\theta}}\right)^T\right] = -E\left(\frac{\partial^2 \ell_n \mathbb{L}}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T}\right) \quad (4.13)$$

where  $\mathbb{L}$  is the likelihood function, which is equal to the probability density function of  $\mathbf{z}$  given  $\boldsymbol{\theta}$ , i.e.,

$$\mathbb{L}(\mathbf{z}; \boldsymbol{\theta}) \equiv p(\mathbf{z} | \boldsymbol{\theta}) \quad (4.14)$$

In Eq. (4.13), the first equality is a definition; the second equality is derived in Appendix B.

The likelihood function is regarded as a function of the unknown parameter vector  $\boldsymbol{\theta}$ , with  $\mathbf{z}$  denoting the measurements. Then an unbiased estimator is called efficient if the covariance matrix equals the inverse of the Fisher information matrix,

$$\text{Cov}(\hat{\boldsymbol{\theta}}) = E[(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})^T] = \mathbf{M}^{-1} \quad (4.15)$$

The matrix  $\mathbf{M}^{-1}$  is known as the Cramér-Rao lower bound, and the expression

$$\text{Cov}(\hat{\boldsymbol{\theta}}) \geq \mathbf{M}^{-1} \quad (4.16)$$

## OUTLINE OF ESTIMATION THEORY

79

is the Cramér-Rao inequality for an unbiased estimator  $\hat{\boldsymbol{\theta}}$  (see Appendix B). This inequality indicates that any unbiased estimator can have a covariance matrix no smaller than  $\mathbf{M}^{-1}$ . If the Cramér-Rao inequality becomes an equality as  $N \rightarrow \infty$ , the estimator is called asymptotically efficient.

The Cramér-Rao inequality for a constant unknown parameter vector  $\boldsymbol{\theta}$  and an unbiased estimator will be discussed further in Chapter 6. The Cramér-Rao inequality for a random parameter vector  $\boldsymbol{\theta}$  and/or a biased estimator is covered in Ref. 2.

### Definition 4.6

Let  $\hat{\boldsymbol{\theta}}(N)$  be an estimate based on  $N$  samples. Then an estimator is called consistent if, for increasing  $N$ ,  $\hat{\boldsymbol{\theta}}(N)$  converges to the true value  $\boldsymbol{\theta}$ ,

$$\lim_{N \rightarrow \infty} \hat{\boldsymbol{\theta}}(N) = \boldsymbol{\theta} \quad (4.17)$$

## 4.2 Parameter Estimation

In this section, the cost functions to be optimized for various estimators are introduced. Linear and nonlinear models for the observations as functions of the model parameters will be considered, together with three different models for the uncertainties in  $\boldsymbol{\theta}$  and  $\boldsymbol{v}$ . Applications of the estimation theories presented here will be described in later chapters.

Measurement equations that are linear and nonlinear in the parameters have already been introduced as

$$\mathbf{z} = \mathbf{H}\boldsymbol{\theta} + \boldsymbol{v} \quad (4.18)$$

and

$$\mathbf{z} = \mathbf{h}(\boldsymbol{\theta}) + \boldsymbol{v} \quad (4.19)$$

Three models for the uncertainties in the parameters and the measurements will be considered. They are designated according to Schweppe<sup>3</sup> as the Bayesian model, the Fisher model, and the least-squares model, formed as follows.

*Bayesian model:*

- 1)  $\boldsymbol{\theta}$  is a vector of random variables with probability density  $p(\boldsymbol{\theta})$ .
- 2)  $\boldsymbol{v}$  is a random vector with probability density  $p(\boldsymbol{v})$ .

*Fisher model:*

- 1)  $\boldsymbol{\theta}$  is a vector of unknown constant parameters.
- 2)  $\boldsymbol{v}$  is a random vector with probability density  $p(\boldsymbol{v})$ .

*Least-squares model:*

- 1)  $\boldsymbol{\theta}$  is a vector of unknown constant parameters.
- 2)  $\boldsymbol{v}$  is a random vector of measurement noise.

For measurement equations that are nonlinear in the parameters, Eq. (4.18) in all three models is replaced by Eq. (4.19).

#### 4.2.1 Estimator for the Bayesian Model

The development of an estimator for the Bayesian model follows from the Bayesian estimation theory explained, for example, in Refs. 3 and 4. The probability densities  $p(\boldsymbol{\theta})$  and  $p(\boldsymbol{v})$  are assumed to be known *a priori*. The conditional density of parameter vector  $\boldsymbol{\theta}$ , given the observation  $\mathbf{z}$ , designated by  $p(\boldsymbol{\theta}|\mathbf{z})$ , is sometimes called the *a posteriori* probability density. This probability density is related to the *a priori* probability densities by Bayes's rule (see Appendix B):

$$p(\boldsymbol{\theta}|\mathbf{z}) = \frac{p(\mathbf{z}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{z})} \quad (4.20)$$

There are several ways to form an estimator, as reviewed in Refs. 2 and 3. In this section, only the estimator that selects  $\hat{\boldsymbol{\theta}}$  as a value that maximizes the conditional probability density  $p(\boldsymbol{\theta}|\mathbf{z})$  is considered. If the vectors  $\boldsymbol{\theta}$  and  $\boldsymbol{v}$  have Gaussian distributions and are independent, their mean values and variances are

$$E(\boldsymbol{\theta}) = \boldsymbol{\theta}_p, \quad \text{Cov}(\boldsymbol{\theta}) = \boldsymbol{\Sigma}_p \quad (4.21)$$

$$E(\boldsymbol{v}) = \mathbf{0}, \quad \text{Cov}(\boldsymbol{v}) = \mathbf{R} \quad (4.22)$$

which can also be stated as (see Appendix B)

$$\begin{aligned} \boldsymbol{\theta} &\text{ is } \mathcal{N}(\boldsymbol{\theta}_p, \boldsymbol{\Sigma}_p) \\ \boldsymbol{v} &\text{ is } \mathcal{N}(\mathbf{0}, \mathbf{R}) \end{aligned} \quad (4.23)$$

where  $\boldsymbol{\theta}$  and  $\boldsymbol{v}$  are independent. Under these assumptions,

$$p(\boldsymbol{\theta}) = [(2\pi)^{n_p} |\boldsymbol{\Sigma}_p|]^{-\frac{1}{2}} \exp\left[-\frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\theta}_p)^T \boldsymbol{\Sigma}_p^{-1} (\boldsymbol{\theta} - \boldsymbol{\theta}_p)\right] \quad (4.24)$$

where  $n_p$  is the number of unknown parameters. For the model of Eq. (4.18),

$$p(\mathbf{z}|\boldsymbol{\theta}) = [(2\pi)^N |\mathbf{R}|]^{-\frac{1}{2}} \exp\left[-\frac{1}{2}(\mathbf{z} - \mathbf{H}\boldsymbol{\theta})^T \mathbf{R}^{-1} (\mathbf{z} - \mathbf{H}\boldsymbol{\theta})\right] \quad (4.25)$$

## OUTLINE OF ESTIMATION THEORY

81

The probability density function  $p(\boldsymbol{\theta}|z)$  is obtained from Eq. (4.20) as

$$p(\boldsymbol{\theta}|z) = \left[ \frac{1}{p(z)} \right] [(2\pi)^{N+n_p} |\mathbf{R}| |\boldsymbol{\Sigma}_p|]^{-\frac{1}{2}} \cdot \exp \left[ -\frac{1}{2}(z - \mathbf{H}\boldsymbol{\theta})^T \mathbf{R}^{-1} (z - \mathbf{H}\boldsymbol{\theta}) - \frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\theta}_p)^T \boldsymbol{\Sigma}_p^{-1} (\boldsymbol{\theta} - \boldsymbol{\theta}_p) \right] \quad (4.26)$$

Thus, the most probable estimate

$$\hat{\boldsymbol{\theta}} = \max_{\boldsymbol{\theta}} p(\boldsymbol{\theta}|z) \quad (4.27)$$

minimizes

$$J(\boldsymbol{\theta}) = \frac{1}{2}(z - \mathbf{H}\boldsymbol{\theta})^T \mathbf{R}^{-1} (z - \mathbf{H}\boldsymbol{\theta}) + \frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\theta}_p)^T \boldsymbol{\Sigma}_p^{-1} (\boldsymbol{\theta} - \boldsymbol{\theta}_p) \quad (4.28)$$

The quantity  $J(\boldsymbol{\theta})$  is usually called the cost function. The probability density  $p(z)$  does not depend on  $\boldsymbol{\theta}$ , and therefore has no influence on the cost function for parameter estimation, as follows from Eqs. (4.26) and (4.27).

### 4.2.2 Estimator for the Fisher Model

An estimator for the Fisher model is based on the Fisher<sup>5</sup> estimation theory, using the concept of a likelihood function,

$$\mathbb{L}(z; \boldsymbol{\theta}) = p(z | \boldsymbol{\theta}) \quad (4.29)$$

Because  $\boldsymbol{\theta}$  is now assumed to be a vector of unknown constants, and not a random variable, the probability density function  $p(\boldsymbol{\theta})$  is not defined, and Bayes's rule does not hold.

The most common estimator for the Fisher model is the maximum likelihood (ML) estimator, which is equal to the value of  $\boldsymbol{\theta}$  that maximizes  $\mathbb{L}(z; \boldsymbol{\theta})$  for given  $z$ . In the case of a Gaussian  $p(z)$ , where  $\mathbf{v}$  is  $\mathbb{N}(\mathbf{0}, \mathbf{R})$ , the likelihood function takes the form

$$\mathbb{L}(z; \boldsymbol{\theta}) = [(2\pi)^N |\mathbf{R}|]^{-\frac{1}{2}} \exp \left[ -\frac{1}{2}(z - \mathbf{H}\boldsymbol{\theta})^T \mathbf{R}^{-1} (z - \mathbf{H}\boldsymbol{\theta}) \right] \quad (4.30)$$

Then the maximum likelihood estimate

$$\hat{\boldsymbol{\theta}} = \max_{\boldsymbol{\theta}} \mathbb{L}(z; \boldsymbol{\theta}) \quad (4.31)$$

minimizes

$$J(\boldsymbol{\theta}) = \frac{1}{2}(z - \mathbf{H}\boldsymbol{\theta})^T \mathbf{R}^{-1}(z - \mathbf{H}\boldsymbol{\theta}) \quad (4.32)$$

or, in the case of nonlinear observations,

$$J(\boldsymbol{\theta}) = \frac{1}{2}[z - \mathbf{h}(\boldsymbol{\theta})]^T \mathbf{R}^{-1}[z - \mathbf{h}(\boldsymbol{\theta})] \quad (4.33)$$

The development of ML parameter estimators for a dynamic system, with direct application to aircraft parameter estimation, is covered in Chapter 6.

#### 4.2.3 Estimator for the Least-Squares Model

In specifying the form of the least-squares model, no uncertainty models for  $\boldsymbol{\theta}$  and  $\nu$  are used, i.e., there are no probability statements concerning  $\boldsymbol{\theta}$  and  $\nu$ . An estimate for the least-squares model can be obtained by the reasoning that, given  $z$ , the “best” estimate of  $\boldsymbol{\theta}$  comes from minimizing the weighted sum of squared differences between the measured outputs and the model outputs,

$$J(\boldsymbol{\theta}) = \frac{1}{2}(z - \mathbf{H}\boldsymbol{\theta})^T \mathbf{R}^{-1}(z - \mathbf{H}\boldsymbol{\theta}) \quad (4.34)$$

where  $\mathbf{R}^{-1}$  is now a positive definite weighting matrix, chosen by judgment. Optimization of the preceding  $J(\boldsymbol{\theta})$  leads to the well-known weighted least-squares (WLS) estimator. In the special case where  $\mathbf{R} = \mathbf{I}$ , the ordinary least-squares (OLS) estimator is obtained, with cost function

$$J(\boldsymbol{\theta}) = \frac{1}{2}(z - \mathbf{H}\boldsymbol{\theta})^T(z - \mathbf{H}\boldsymbol{\theta}) \quad (4.35)$$

Considering the entire set of measured data  $z(i)$ ,  $i = 1, 2, \dots, N$ , the OLS estimator for a scalar measurement is obtained by minimizing

$$J(\boldsymbol{\theta}) = \frac{1}{2} \sum_{i=1}^N [z(i) - \mathbf{H}(i)\boldsymbol{\theta}]^2 \quad (4.36)$$

which is the same as Eq. (4.35). For a nonlinear observation model, the minimization of

$$J(\boldsymbol{\theta}) = \frac{1}{2} \sum_{i=1}^N [z(i) - \mathbf{h}(i, \boldsymbol{\theta})]^2 \quad (4.37)$$

leads to the nonlinear least-squares estimator. The application of various least-squares estimators to aircraft system identification problems is covered in Chapter 5.

## OUTLINE OF ESTIMATION THEORY

83

### 4.3 State Estimation

For a deterministic dynamic system, the time histories of state variables are obtained by integrating the state equations for given input and initial conditions. For a stochastic system, however, there are random variables in the dynamic equations, so the time histories of the state variables must be estimated using a statistical method. There are many possible formulations of the state estimation problem. In this section, state estimation algorithms are outlined using probability density functions describing the state and measured variables. This probabilistic formulation is usually called the Bayesian approach to state estimation, and has many similarities to the development of parameter estimation using the Bayesian model for the uncertainties, discussed in the preceding section.

Because of technical difficulties associated with continuous-time white noise, mentioned in Chapter 2, the development of state estimation algorithms begins with a discrete-time system. The discrete-time results are then extended to the case of a continuous-time dynamic system model with discrete-time measurements, which corresponds to the practical aircraft problem. One of the resulting linear estimation formulas is known as the Kalman filter. This filter is a widely-used algorithm for state estimation.

#### 4.3.1 Linear State Estimator

Recalling Eqs. (2.39) and (2.40), the discrete-time linear stochastic system can be modeled as

$$\mathbf{x}(i) = \Phi(i-1)\mathbf{x}(i-1) + \Gamma(i-1)\mathbf{u}(i-1) + \Gamma_w(i-1)\mathbf{w}(i-1) \quad (4.38)$$

$$\mathbf{y}(i) = \mathbf{C}(i)\mathbf{x}(i) + \mathbf{D}(i)\mathbf{u}(i) \quad (4.39)$$

$$\mathbf{z}(i) = \mathbf{y}(i) + \mathbf{v}(i) \quad i = 1, 2, \dots, N \quad (4.40)$$

The matrices  $\Phi(i)$ ,  $\Gamma(i)$ ,  $\Gamma_w(i)$ ,  $\mathbf{C}(i)$ , and  $\mathbf{D}(i)$  are assumed to be known functions of time  $t_i = i\Delta t$ , and  $\mathbf{w}(i)$  and  $\mathbf{v}(i)$  are white noise sequences. The Bayesian model for uncertainties in  $\mathbf{x}(0)$ ,  $\mathbf{w}(i)$ , and  $\mathbf{v}(i)$  is given as

$$\begin{aligned} E[\mathbf{x}(0)] &= \bar{\mathbf{x}}_0 & E\{[\mathbf{x}(0) - \bar{\mathbf{x}}_0][\mathbf{x}(0) - \bar{\mathbf{x}}_0]^T\} &= \mathbf{P}_0 \\ E[\mathbf{w}(i)] &= \mathbf{0} & E[\mathbf{w}(i)\mathbf{w}^T(j)] &= \mathbf{Q}(i)\delta_{ij} \\ E[\mathbf{v}(i)] &= \mathbf{0} & E[\mathbf{v}(i)\mathbf{v}^T(j)] &= \mathbf{R}(i)\delta_{ij} \end{aligned} \quad (4.41)$$

where  $\mathbf{x}(0)$ ,  $\mathbf{w}(i)$ , and  $\mathbf{v}(i)$  are uncorrelated.

In addition, the following notation is used:

- 1)  $\mathbf{Z}_i = [z(1) \ z(2) \ \cdots \ z(i)]^T$  contains all measurements up to and including time  $t_i = i\Delta t$ .
- 2)  $\hat{\mathbf{x}}(i_1 | i_2)$  is the “best” estimate of the state  $\mathbf{x}(i_1)$  using measurements  $\mathbf{Z}_{i_2} = [z(1) \ z(2) \ \cdots \ z(i_2)]^T$ .
- 3)  $\mathbf{e}_x(i_1 | i_2)$  is the error in the estimate  $\hat{\mathbf{x}}(i_1 | i_2)$ , i.e.,

$$\mathbf{e}_x(i_1 | i_2) = \mathbf{x}(i_1) - \hat{\mathbf{x}}(i_1 | i_2) \quad (4.42)$$

The state error covariance matrix, sometimes referred to as the state covariance matrix, is

$$\mathbf{P}(i_1 | i_2) = E[\mathbf{e}_x(i_1 | i_2)\mathbf{e}_x^T(i_1 | i_2)] \quad (4.43)$$

assuming that  $\hat{\mathbf{x}}(i_1 | i_2)$  is an unbiased estimate.

For the system specified by Eqs. (4.38)–(4.41), with measurements  $\mathbf{Z}_{i_2} = [z(1) z(2) \cdots z(i_2)]^T$ , three types of state estimation can be considered:

- 1) Prediction:  $i_1 > i_2$
- 2) Filtering:  $i_1 = i_2$
- 3) Smoothing:  $1 \leq i_1 < i_2$

Of particular interest are unbiased and consistent estimators that minimize the state estimation error in a well-defined statistical sense.

*Prediction.* A one-step prediction is formulated as the calculation of an estimate of  $\mathbf{x}(i)$ , given the measured data  $\mathbf{Z}_{i-1}$ . From Eq. (4.38),

$$\mathbf{x}(i) = \Phi(i-1)\mathbf{x}(i-1) + \Gamma(i-1)\mathbf{u}(i-1) + \Gamma_w(i-1)\mathbf{w}(i-1) \quad (4.44)$$

Assuming that  $\hat{\mathbf{x}}(i-1 | i-1)$  is available, the best estimate of  $\Phi(i-1)\mathbf{x}(i-1)$  in Eq. (4.44) is  $\Phi(i-1)\hat{\mathbf{x}}(i-1 | i-1)$ . Because  $\mathbf{w}(i-1)$  represents a random white sequence, the best prediction of  $\mathbf{w}(i-1)$  that can be made from the measurements  $\mathbf{Z}_{i-1}$  is the expected value  $E[\mathbf{w}(i-1)] = \mathbf{0}$ , and  $\Gamma(i-1)\mathbf{u}(i-1)$  is deterministic. Therefore,

$$\hat{\mathbf{x}}(i | i-1) = \Phi(i-1)\hat{\mathbf{x}}(i-1 | i-1) + \Gamma(i-1)\mathbf{u}(i-1) \quad (4.45)$$

The one-step prediction of the estimation error at time  $i$  follows from Eqs. (4.42), (4.44), and (4.45) as

$$\mathbf{e}_x(i | i-1) = \Phi(i-1)\mathbf{e}_x(i-1 | i-1) + \Gamma_w(i-1)\mathbf{w}(i-1) \quad (4.46)$$

The state covariance matrix of the one-step prediction is given by

$$\begin{aligned} \mathbf{P}(i | i-1) &= E[\mathbf{e}_x(i | i-1)\mathbf{e}_x^T(i | i-1)] \\ &= \Phi(i-1)\mathbf{P}(i-1 | i-1)\Phi^T(i-1) \\ &\quad + \Gamma_w(i-1)E[\mathbf{w}(i-1)\mathbf{e}_x^T(i-1 | i-1)]\Phi^T(i-1) \\ &\quad + \Phi(i-1)E[\mathbf{e}_x(i-1 | i-1)\mathbf{w}^T(i-1)]\Gamma_w^T(i-1) \\ &\quad + \Gamma_w(i-1)\mathbf{Q}(i-1)\Gamma_w^T(i-1) \end{aligned}$$

The quantities  $E[\mathbf{w}(i-1)\mathbf{e}_x^T(i-1 | i-1)]$  and  $E[\mathbf{e}_x(i-1 | i-1)\mathbf{w}^T(i-1)]$  equal zero, because  $\mathbf{w}$  is a white sequence, and  $\mathbf{w}(i-1)$  affects  $\mathbf{e}_x(i | i-1)$  by

## OUTLINE OF ESTIMATION THEORY

85

Eq. (4.46), but not  $e_x(i - 1 | i - 1)$ . Therefore,

$$\mathbf{P}(i | i - 1) = \Phi(i - 1) \mathbf{P}(i - 1 | i - 1) \Phi(i - 1) + \Gamma_w(i - 1) \mathbf{Q}(i - 1) \Gamma_w^T(i - 1) \quad (4.47)$$

*Filtering.* The Bayesian approach to filtering requires an update of the knowledge about the state of the dynamic system from the probability density  $p(\mathbf{x})$  to the conditional probability density  $p(\mathbf{x} | z)$ , using the measurements  $z$ . The updating procedure can be repeated every time a new measurement vector is available.

It is therefore necessary to find a recursive relationship that changes  $p[\mathbf{x}(i - 1) | \mathbf{Z}_{i-1}]$  into  $p[\mathbf{x}(i) | \mathbf{Z}_i]$ . The development of this relationship, which is shown, for example, in Refs. 6 and 7, results in the *a posteriori* density function

$$p[\mathbf{x}(i) | \mathbf{Z}_i] = \frac{p[z(i) | \mathbf{x}(i)] p[\mathbf{x}(i) | \mathbf{Z}_{i-1}]}{p[z(i) | \mathbf{Z}_{i-1}]} \quad (4.48)$$

Expressions for filtering can be obtained from Eq. (4.48) in different ways. Smith<sup>4</sup> and Sorenson<sup>7</sup> use a Gaussian assumption in the model for uncertainties. This assumption also means that all  $\hat{\mathbf{x}}(i)$  and  $z(i)$  will be Gaussian, as will be all conditional probability densities. In this case, all probability densities involved can be fully described by their mean value and covariance matrix.

Using the Gaussian assumption, the conditional probability density  $p[\mathbf{x}(i) | \mathbf{Z}_i]$  takes the form

$$p[\mathbf{x}(i) | \mathbf{Z}_i] = [(2\pi)^{n_s} |\mathbf{P}(i)|]^{-\frac{1}{2}} \cdot \exp\left\{-\frac{1}{2} [\mathbf{x}(i) - \hat{\mathbf{x}}(i)]^T \mathbf{P}^{-1}(i) [\mathbf{x}(i) - \hat{\mathbf{x}}(i)]\right\} \quad (4.49)$$

with mean value given by

$$\hat{\mathbf{x}}(i | i) = \hat{\mathbf{x}}(i | i - 1) + \mathbf{K}(i)[z(i) - \mathbf{C}(i)\hat{\mathbf{x}}(i | i - 1) - \mathbf{D}(i)\mathbf{u}(i)] \quad (4.50a)$$

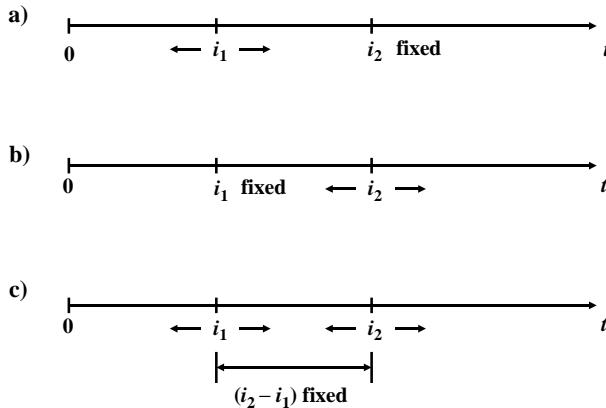
and state covariance matrix

$$\mathbf{P}(i | i) = [\mathbf{I} - \mathbf{K}(i)\mathbf{C}(i)]\mathbf{P}(i | i - 1) \quad (4.50b)$$

where

$$\mathbf{K}(i) = \mathbf{P}(i | i - 1)\mathbf{C}^T(i)[\mathbf{C}(i)\mathbf{P}(i | i - 1)\mathbf{C}^T(i) + \mathbf{R}(i)]^{-1} \quad (4.50c)$$

The algorithm for sequential estimation is completed by Eqs. (4.45) and (4.47) for propagating the state estimate from one sample time to the next, along with



**Fig. 4.1** Three types of discrete-time smoothing: a) fixed interval, b) fixed point, and c) fixed lag.

the initial conditions

$$\begin{aligned}\hat{x}(0 | 0) &= \bar{x}_0 = E[x(0)] \\ P(0 | 0) &= P_0 = E\{[x(0) - \bar{x}_0][x(0) - \bar{x}_0]^T\}\end{aligned}\quad (4.50d)$$

The development of Eqs. (4.50) can be found in Ref. 4, 7, or 8.

**Smoothing.** As stated previously, the smoothing process produces an optimal state estimate  $\hat{x}(i_1 | i_2)$  for  $1 \leq i_1 < i_2$ , using measurements  $Z_{i_2} = [z(1) z(2) \cdots z(i_2)]^T$ . Three types of smoothing can be defined, as shown in Fig. 4.1. They are called fixed-interval, fixed-point, and fixed-lag smoothing. In fixed-interval smoothing,  $i_2$  is fixed and  $i_1$  varies from 1 to  $i_2$ . For fixed-point smoothing,  $i_1$  is fixed while  $i_2$  increases. Finally, in fixed-lag smoothing, both  $i_1$  and  $i_2$  vary, but the interval between  $i_1$  and  $i_2$  remains fixed.

There are many algorithms for optimal smoothing.<sup>8,9</sup> A fixed-interval smoother will be introduced in Chapter 10 in connection with estimation of systematic instrumentation errors and flight-path reconstruction.

### 4.3.2 Kalman Filter

Equations (4.50) together with Eqs. (4.45) and (4.47) are complete recursive relationships for computing the optimal estimate of the state  $\hat{x}$  and its covariance matrix  $P$ , at each time step. The equations are summarized as follows.

## OUTLINE OF ESTIMATION THEORY

87

*Initial conditions:*

$$\begin{aligned}\hat{x}(0 | 0) &= \bar{x}_0 \\ P(0 | 0) &= P_0\end{aligned}\quad (4.51a)$$

*Prediction:*

$$\hat{x}(i | i - 1) = \Phi(i - 1)\hat{x}(i - 1 | i - 1) + \Gamma(i - 1)u(i - 1) \quad (4.51b)$$

$$P(i | i - 1) = \Phi(i - 1)P(i - 1 | i - 1)\Phi^T(i - 1) + \Gamma_w(i - 1)Q(i - 1)\Gamma_w^T(i - 1) \quad (4.51c)$$

*Measurement update:*

$$\hat{x}(i | i) = \hat{x}(i | i - 1) + K(i)[z(i) - C(i)\hat{x}(i | i - 1) - D(i)u(i)] \quad (4.51d)$$

$$P(i | i) = [I - K(i)C(i)]P(i | i - 1) \quad (4.51e)$$

$$K(i) = P(i | i - 1)C^T(i)[C(i)P(i | i - 1)C^T(i) + R(i)]^{-1} \quad (4.51f)$$

Equations (4.51) are identical to the Kalman filter algorithm originally developed in Ref. 10, using a different approach. In Eq. (4.51d), the second term on the right represents a correction to the propagated estimate  $\hat{x}(i | i - 1)$ , based on the innovations, defined as

$$v(i) = z(i) - C(i)\hat{x}(i | i - 1) - D(i)u(i) \quad (4.52)$$

The estimator (4.51d) can therefore be viewed as a linear feedback system with time-varying gain  $K(i)$ . The block diagram of the system described by Eqs. (4.51) is shown in Fig. 4.2.

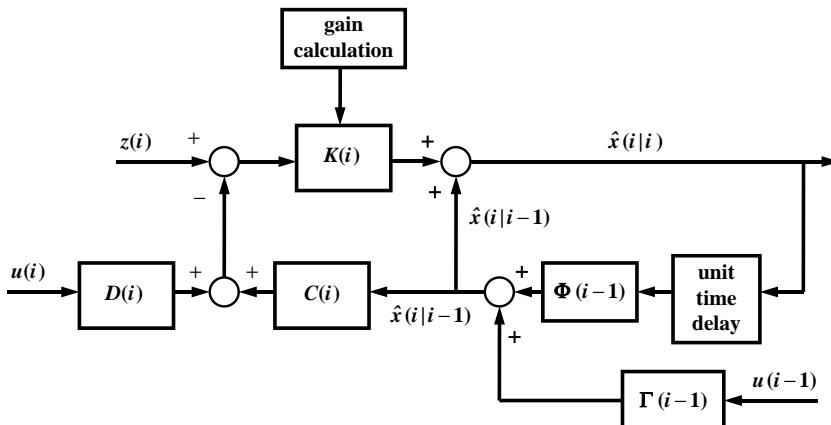
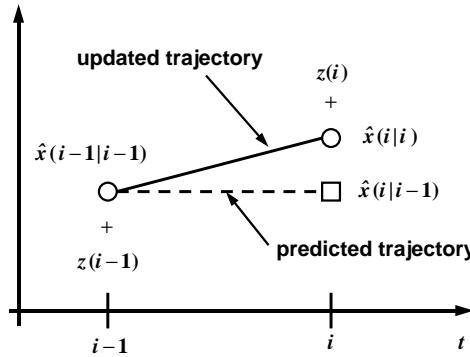


Fig. 4.2 Block diagram of the discrete-time Kalman filter.



**Fig. 4.3** Two steps in the discrete-time Kalman filter.

The matrix difference equation (4.51e) for  $\mathbf{P}(i | i)$  is called a Riccati equation. Because  $\mathbf{P}(i | i)$  represents the error covariance matrix for the state estimate, the Kalman filter provides an error analysis that indicates how well the state estimation is being performed. In Eq. (4.51e), the second term on the right shows a change in the uncertainty of the state estimate, due to process noise and as a result of the noisy measurement, cf. Eq. (4.51f). Because Eqs. (4.51e) and (4.51f) do not depend on the state estimates, they can be computed before the measurement is taken.

The two steps in the state estimation, i.e., the prediction and measurement update, are visualized for a simple scalar system in Fig. 4.3. The various quantities involved in the discrete optimal filter equations (4.51) are shown for a single time step.

**Steady-state Kalman filter.** So far, the Kalman filter equations have been presented for a time-varying discrete model specified by Eqs. (4.38) and (4.39). The filter equations can be simplified when the dynamic system is time invariant, i.e., the matrices  $\Phi$ ,  $\Gamma$ ,  $\Gamma_w$ ,  $C$ ,  $D$ ,  $Q$ , and  $R$  are constant. Further simplification is possible when the estimator becomes time invariant as well. This can happen if the state covariance matrix  $\mathbf{P}(i | i)$  converges to a constant matrix for every  $i$ ,

$$\mathbf{P}(i | i) = \mathbf{P}(i - 1 | i - 1) \equiv \mathbf{P}_\infty \quad (4.53)$$

Then  $\mathbf{P}(i | i - 1) \equiv \mathbf{P}$  will also be a constant matrix which is, in general, different from  $\mathbf{P}_\infty$ , and the filter equations for the steady-state estimator are

$$\hat{\mathbf{x}}(i | i - 1) = \Phi \hat{\mathbf{x}}(i - 1 | i - 1) + \Gamma \mathbf{u}(i - 1) \quad (4.54a)$$

$$\mathbf{P} = \Phi \mathbf{P}_\infty \Phi^T + \Gamma_w \mathbf{Q} \Gamma_w^T \quad (4.54b)$$

## OUTLINE OF ESTIMATION THEORY

89

$$\hat{x}(i | i) = \hat{x}(i | i - 1) + K[z(i) - C\hat{x}(i | i - 1) - Du(i)] \quad (4.54c)$$

$$P_\infty = P - KCP \quad (4.54d)$$

$$K = PC^T [CPC^T + R]^{-1} \quad (4.54e)$$

and the initial conditions are given by Eq. (4.51a). The filter represented by Eqs. (4.54) is known as a steady-state filter. It can be easily implemented once  $P_\infty$  has been determined. The matrix  $P_\infty$  can be obtained by solving the matrix difference equation (4.51e) repeatedly until it converges to a steady-state value. The other possibility is to find a positive definite solution from the three algebraic equations (4.54b), (4.54d), and (4.54e), or equivalently from the combined equation

$$P = \Phi [P - PC^T (CPC^T + R)^{-1} CP] \Phi^T + \Gamma_w Q \Gamma_w^T \quad (4.55)$$

Conditions for the existence of a solution to the preceding equation are discussed in Ref. 3 and a method for the solution is discussed in Refs. 3 and 11. More about the steady-state Kalman filter can be found in Ref. 9.

### 4.3.3 Continuous-Discrete Kalman Filter

For some state estimation problems, it is preferable to use the state equation in continuous-time form. As shown in Chapter 3, this is the form taken by the aircraft linearized equations of motion. When combined with a discrete measurement equation, the combined continuous-discrete model equations corresponding to Eqs. (4.38) and (4.40) are

$$\dot{x}(t) = A(t)x(t) + B(t)u(t) + B_w(t)w(t) \quad (4.56)$$

$$z(i) = C(i)x(i) + D(i)u(i) + v(i) \quad i = 1, 2, \dots, N \quad (4.57)$$

where

$$\begin{aligned} E[x(0)] &= \bar{x}_0 & E\{[x(0) - \bar{x}_0][x(0) - \bar{x}_0]^T\} &= P_0 \\ E[w(t)] &= \mathbf{0} & E[w(t_i)w^T(t_j)] &= Q(t_i)\delta(t_i - t_j) \\ E[v(i)] &= \mathbf{0} & E[v(i)v^T(j)] &= R(i)\delta_{ij} \end{aligned} \quad (4.58)$$

and  $x(0)$ ,  $w(t)$ , and  $v(i)$  are uncorrelated.

To develop the prediction equations, it is assumed that at time,  $t_i = i\Delta t$  the Kalman filter has a state estimate  $\hat{x}(i-1 | i-1)$  with covariance  $P(i-1 | i-1)$ . As in the previous discrete case, a one-step ahead prediction is formulated as the calculation of an estimate  $\hat{x}(t | i-1)$ , given the measured data  $Z_{i-1}$ . The equation for the predicted state estimates can be obtained by

taking conditional expectations on both sides of Eq. (4.56), which gives

$$\frac{d}{dt}[\hat{x}(t | i - 1)] = \mathbf{A}(t)\hat{x}(t | i - 1) + \mathbf{B}(t)\mathbf{u}(t) \quad (4.59)$$

for  $(i - 1)\Delta t \leq t \leq i\Delta t$ .

The state covariance matrix of the one-step ahead prediction can be obtained by limiting arguments applied to Eq. (4.51c) (cf. Ref. 8) or by using covariance equations from Ref. 6. The resulting differential equation is

$$\frac{d}{dt}[\mathbf{P}(t | i - 1)] = \mathbf{A}(t)\mathbf{P}(t | i - 1) + \mathbf{P}(t | i - 1)\mathbf{A}^T(t) + \mathbf{B}_w(t)\mathbf{Q}(t)\mathbf{B}_w^T(t) \quad (4.60)$$

The measurement update equations in the continuous-discrete Kalman filter are identical to those given in Eqs. (4.51d)–(4.51f). The Kalman filter equations for the continuous-discrete system specified by Eqs. (4.56)–(4.58) are summarized as follows.

*Initial conditions:*

$$\begin{aligned} \hat{x}(0 | 0) &= \bar{x}_0 \\ \mathbf{P}(0 | 0) &= \mathbf{P}_0 \end{aligned} \quad (4.61a)$$

*Prediction:*

$$\frac{d}{dt}[\hat{x}(t | i - 1)] = \mathbf{A}(t)\hat{x}(t | i - 1) + \mathbf{B}(t)\mathbf{u}(t) \quad (4.61b)$$

$$\frac{d}{dt}[\mathbf{P}(t | i - 1)] = \mathbf{A}(t)\mathbf{P}(t | i - 1) + \mathbf{P}(t | i - 1)\mathbf{A}^T(t) + \mathbf{B}_w(t)\mathbf{Q}(t)\mathbf{B}_w^T(t) \quad (4.61c)$$

*Measurement update:*

$$\hat{x}(i | i) = \hat{x}(i | i - 1) + \mathbf{K}(i)[z(i) - \mathbf{C}(i)\hat{x}(i | i - 1) - \mathbf{D}(i)\mathbf{u}(i)] \quad (4.61d)$$

$$\mathbf{P}(i | i) = [\mathbf{I} - \mathbf{K}(i)\mathbf{C}(i)]\mathbf{P}(i | i - 1) \quad (4.61e)$$

$$\mathbf{K}(i) = \mathbf{P}(i | i - 1)\mathbf{C}^T(i)[\mathbf{C}(i)\mathbf{P}(i | i - 1)\mathbf{C}^T(i) + \mathbf{R}(i)]^{-1} \quad (4.61f)$$

#### 4.3.4 Nonlinear State Estimator

There are several model forms for nonlinear dynamic systems, expressing various degrees of complexity. In the following discussion, a discrete-time nonlinear model is considered in the form

$$\mathbf{x}(i) = \mathbf{f}[\mathbf{x}(i - 1), \mathbf{u}(i - 1), i - 1] + \boldsymbol{\Gamma}_w(i - 1)\mathbf{w}(i - 1) \quad (4.62a)$$

$$\mathbf{y}(i) = \mathbf{h}[\mathbf{x}(i), \mathbf{u}(i), i] \quad (4.62b)$$

$$\mathbf{z}(i) = \mathbf{h}[\mathbf{x}(i), \mathbf{u}(i), i] + \mathbf{v}(i) \quad i = 1, 2, \dots, N \quad (4.62c)$$

## OUTLINE OF ESTIMATION THEORY

91

$$\begin{aligned} E[\mathbf{x}(0)] &= \bar{\mathbf{x}}_0 & E\{[\mathbf{x}(0) - \bar{\mathbf{x}}_0][\mathbf{x}(0) - \bar{\mathbf{x}}_0]^T\} &= \mathbf{P}_0 \\ E[\mathbf{w}(i)] &= \mathbf{0} & E[\mathbf{w}(i)\mathbf{w}^T(j)] &= \mathbf{Q}(i)\delta_{ij} \\ E[\mathbf{v}(i)] &= \mathbf{0} & E[\mathbf{v}(i)\mathbf{v}^T(j)] &= \mathbf{R}(i)\delta_{ij} \end{aligned} \quad (4.62d)$$

where  $\mathbf{x}(0)$ ,  $\mathbf{w}(i)$ , and  $\mathbf{v}(i)$  are uncorrelated, and  $f$  and  $\mathbf{h}$  are vector functions of the state vector  $\mathbf{x}(i)$ , control vector  $\mathbf{u}(i)$ , and time index  $i$ . The same Bayesian model for the uncertainties that was stated earlier for a linear system is considered.

As in the linear case, the state estimator uses measurements to update the knowledge of the state from the *a priori* probability density  $p(\mathbf{x})$ , to the *a posteriori* probability density  $p(\mathbf{x}|z)$ . The recursive relation between these two probability densities has already been presented as

$$p[\mathbf{x}(i) | \mathbf{Z}_i] = \frac{p[z(i) | \mathbf{x}(i)]p[\mathbf{x}(i) | \mathbf{Z}_{i-1}]}{p[z(i) | \mathbf{Z}_{i-1}]} \quad (4.63)$$

Computing probability densities when the state and output equations are nonlinear is difficult in practice, because nonlinear functions of Gaussian probability densities are not Gaussian, so expected values and higher-order moments must be computed to characterize the probability densities.

To avoid this difficulty, several approximate solutions to the nonlinear filtering problem have been developed.<sup>3,8,12</sup> Most of these algorithms are based on linearization of the nonlinear model equations. They differ, however, by the selection of the point about which the linearization is done.

As an example, the filter equations will be presented for the model specified by Eqs. (4.62). The nonlinear terms  $f[\mathbf{x}(i-1), \mathbf{u}(i-1), i-1]$  and  $\mathbf{h}[\mathbf{x}(i), \mathbf{u}(i), i]$  will be expanded about the most recent available estimates of the state. The linearization yields

$$\begin{aligned} f[\mathbf{x}(i-1), \mathbf{u}(i-1), i-1] &= f[\hat{\mathbf{x}}(i-1 | i-1), \mathbf{u}(i-1), i-1] \\ &\quad + \Phi(i-1)[\mathbf{x}(i-1) - \hat{\mathbf{x}}(i-1 | i-1)] \end{aligned} \quad (4.64)$$

$$\mathbf{h}[\mathbf{x}(i), \mathbf{u}(i), i] = \mathbf{h}[\hat{\mathbf{x}}(i | i-1), \mathbf{u}(i), i] + \mathbf{C}(i)[\mathbf{x}(i) - \hat{\mathbf{x}}(i | i-1)] \quad (4.65)$$

where

$$\Phi(i-1) = \left. \frac{\partial f[\mathbf{x}, \mathbf{u}(i-1), i-1]}{\partial \mathbf{x}} \right|_{\mathbf{x}=\hat{\mathbf{x}}(i-1 | i-1)} \quad (4.66a)$$

$$\mathbf{C}(i) = \left. \frac{\partial \mathbf{h}[\mathbf{x}, \mathbf{u}(i), i]}{\partial \mathbf{x}} \right|_{\mathbf{x}=\hat{\mathbf{x}}(i | i-1)} \quad (4.66b)$$

Substituting Eqs. (4.64) and (4.65) into Eqs. (4.62), the linearized model takes the form

$$\mathbf{x}(i) = \Phi(i-1)\mathbf{x}(i-1) + \mathbf{v}(i-1) + \Gamma_w(i-1)\mathbf{w}(i-1) \quad (4.67)$$

$$\mathbf{z}(i) = \mathbf{C}(i)\mathbf{x}(i) + \mathbf{b}(i) + \mathbf{v}(i) \quad (4.68)$$

where  $v(i - 1)$  and  $b(i)$  are known at the  $i$ th time step,

$$v(i - 1) = f[\hat{x}(i - 1 | i - 1), u(i - 1), i - 1] - \Phi(i - 1)\hat{x}(i - 1 | i - 1) \quad (4.69)$$

$$b(i) = h[\hat{x}(i | i - 1), u(i), i] - C(i)\hat{x}(i | i - 1) \quad (4.70)$$

For the filter equations, assuming that the best estimate of  $x(i - 1)$  for  $Z_{i-1}$ ,  $\hat{x}(i - 1 | i - 1)$ , is known, then the best estimate of  $\Phi(i - 1)x(i - 1)$  is  $\Phi(i - 1)\hat{x}(i - 1 | i - 1)$ . Therefore, the prediction equation for the state is

$$\begin{aligned} \hat{x}(i | i - 1) &= \Phi(i - 1)\hat{x}(i - 1 | i - 1) + v(i - 1) \\ &= f[\hat{x}(i - 1 | i - 1), u(i - 1), i - 1] \end{aligned} \quad (4.71a)$$

Similarly, for the output prediction,

$$\hat{y}(i | i - 1) = h[\hat{x}(i | i - 1), u(i), i] \quad (4.71b)$$

The remaining prediction equation for the state covariance matrix is given by Eq. (4.51c) as

$$\begin{aligned} P(i | i - 1) &= \Phi(i - 1)P(i - 1 | i - 1)\Phi^T(i - 1) \\ &\quad + \Gamma_w(i - 1)Q(i - 1)\Gamma_w^T(i - 1) \end{aligned} \quad (4.71c)$$

The measurement update equations are obtained from Eqs. (4.51d)–(4.51f) after only small modification,

$$\hat{x}(i | i) = \hat{x}(i | i - 1) + K(i)[z(i) - \hat{y}(i | i - 1)] \quad (4.71d)$$

$$P(i | i) = [I - K(i)C(i)]P(i | i - 1) \quad (4.71e)$$

$$K(i) = P(i | i - 1)C^T(i)[C(i)P(i | i - 1)C^T(i) + R(i)]^{-1} \quad (4.71f)$$

From Eqs. (4.71f) and (4.66b), the matrix  $K(i)$  is dependent on the estimator  $\hat{x}(i | i - 1)$ . This means that the gain matrix cannot be precomputed, so the complete set of filter equations (4.71) must be solved simultaneously. Equations (4.71) are referred to as the extended Kalman filter. Application of this filter to state and parameter estimation is discussed in Chapters 8 and 10. State estimation algorithms for a nonlinear system using the Fisher model or the least-squares model for the uncertainties can also be developed. More on these possibilities is available in Ref. 3.

#### 4.4 Summary and Concluding Remarks

This chapter contains a brief introduction to parameter and state estimation theory for dynamic systems. Parameter estimation was recognized as a process for finding values of unknown system parameters  $\theta$ , based on noisy

## OUTLINE OF ESTIMATION THEORY

93

measurements  $z$ . Because the parameter estimates depend on the random variable  $z$ , estimated parameters are expressed in terms of the most important properties of the probability density function  $p(\boldsymbol{\theta} | z)$ , i.e., the mean and covariance. To describe the quality of the estimates of  $\boldsymbol{\theta}$ , estimator properties were defined.

Two measurement equations relating the measurements  $z$ , measurement errors  $\boldsymbol{v}$ , and parameters  $\boldsymbol{\theta}$ , were considered:

$$\begin{aligned} z &= \mathbf{H}\boldsymbol{\theta} + \boldsymbol{v} \\ z &= \mathbf{h}(\boldsymbol{\theta}) + \boldsymbol{v} \end{aligned}$$

where the matrix  $\mathbf{H}$  and the form of the function  $\mathbf{h}(\boldsymbol{\theta})$  are assumed to be completely known. The first equation is linear in the parameters, and the second is nonlinear in the parameters, leading to linear and nonlinear estimation problems, respectively.

The other distinction between measurement equations is based on the mathematical models for the uncertainties in the parameters and measurements. The models considered were the Bayesian model, Fisher model, and least-squares model.

In the first model, the unknown parameters are assumed to be random variables. These parameters are estimated using Bayes's rule, which assumes the existence of an *a priori* probability density for the parameters and measurements. From this information, Bayes's rule provides the *a posteriori* probability density for the parameters, which is equal to the conditional probability density for the parameters, given the measurements. Despite the generality of the Bayes estimator, the method has not found wide application in aircraft parameter estimation. The main reason is the difficulty in making an explicit statement about the form of the *a priori* probability density for the parameters. There are, however, Bayes-like methods that combine *a priori* information about the parameters with measured data. These methods will be described in Chapters 5 and 6 in connection with the linear regression and maximum likelihood techniques.

In the Fisher model, the parameters are assumed to be unknown constants, and the probability density function of the measurement noise is specified. This estimator is based on maximization of a likelihood function, which is equal to the conditional probability density of the measurements, given the parameters. Detailed treatment of the maximum likelihood estimator and its application to aircraft parameter estimation is given in Chapter 6.

In the least-squares model, the parameters are assumed to be unknown constants, and the measurement noise is assumed to be random. An estimator is obtained by application of the least-squares principle. The least-squares model is a basis for the regression analysis to be discussed in Chapter 5, along with its practical application to aircraft parameter estimation problems.

The three uncertainty models have corresponding formulations in the frequency domain (see Chapter 7) and as real-time estimators (see Chapter 8). Broader explanations of the parameter estimation problem can be found in other textbooks, e.g., Refs. 2 and 13.

The discussion of state estimation was based on the Bayesian approach to the problem. Three estimation techniques—prediction, filtering, and smoothing—were covered, with special emphasis on the Kalman filter for a linear discrete-time system and a continuous-time dynamic system with discrete-time measurements. The steady-state version of the Kalman filter and an extension to a nonlinear discrete-time dynamic system were also introduced. Application of the Kalman filter in the maximum likelihood parameter estimation algorithm for a stochastic system is covered in Chapter 6. The nonlinear filtering problem appears in Chapters 8 and 10, in the context of recursive time-varying parameter estimation and simultaneous estimation of states and parameters.

There have been numerous books and papers published on optimal filtering, optimal smoothing, and specifically on the Kalman filter. Recommended books are those by Schweppe,<sup>3</sup> Gelb,<sup>8</sup> Jazwinski,<sup>14</sup> and Minkler and Minkler<sup>9</sup> and the paper by Sorenson.<sup>7</sup> The presentation here was intended to provide background information and theory for the next four chapters, which are concerned with application of the theory to practical aircraft system identification problems.

## References

- <sup>1</sup>Goodwin, G. C., and Payne, R. L., *Dynamic System Identification: Experiment Design and Data Analysis*, Academic International Press, New York, 1977.
- <sup>2</sup>Sorenson, H. W., *Parameter Estimation: Principles and Problems*, Marcel-Decker, New York, 1980.
- <sup>3</sup>Schweppe, F. C., *Uncertain Dynamic Systems*, Prentice-Hall, Upper Saddle River, NJ, 1973.
- <sup>4</sup>Smith, G. L., "On the Theory and Methods of Statistical Inference," NASA TR R-251, 1967.
- <sup>5</sup>Fisher, R. A., "On an Absolute Criterion for Fitting Frequency Curves", *Messenger of Mathematics*, Vol. 41, 1912, pp. 155–160.
- <sup>6</sup>Bryson, A. E., and Ho, Y. C., *Applied Optimal Control*, Hemisphere, Washington, DC, 1975.
- <sup>7</sup>Sorenson, H. W., "Comparison of Kalman, Bayesian, and Maximum Likelihood Estimation Techniques," *Theory and Application of Kalman Filtering*, AGARD AG-139, 1970, pp. 121–142.
- <sup>8</sup>Gelb, A. (ed.), *Applied Optimal Estimation*, MIT Press, Cambridge, MA, 1974.
- <sup>9</sup>Minkler, G., and Minkler, J., *Theory and Application of Kalman Filtering*, Magellan Book Co., Baltimore, MD, 1993.
- <sup>10</sup>Kalman, R. E., "A New Approach to Linear Filtering and Prediction Problems," *Transactions of the ASME*, Series D, Vol. 82, 1960, pp. 35–45.
- <sup>11</sup>Vaughan, D. R., "A Nonrecursive Algebraic Solution for the Discrete Riccati Equation," *IEEE Transactions on Automatic Control*, Vol. AC-15, 1970, pp. 597–599.
- <sup>12</sup>*Advances in the Techniques and Technology of the Application of Nonlinear Filters and Kalman Filter*, AGARD-AG-256, 1982.
- <sup>13</sup>Mendel, J. M., *Discrete Techniques of Parameter Estimation, The Equation Error Formulation*, Marcel Dekker, New York, 1973.
- <sup>14</sup>Jazwinski, A. H., *Stochastic Processes and Filtering Theory*, Academic International Press, New York, 1970.

## 5

## Regression Methods

In the context of system identification applied to aircraft, regression refers to a statistical technique for modeling and investigating relationships among measured variables. An example of a regression is the following model relating the nondimensional pitching moment coefficient  $C_m$ , to angle of attack  $\alpha$  and Mach number  $M$ , for data collected during a wind-tunnel test:

$$C_m = C_{m_0} + C_{m\alpha}\alpha + C_{mM}M + C_{m\alpha M}\alpha M + v_m \quad (5.1)$$

In this example,  $\alpha$  and  $M$  are variables that are set to selected values for each experimental test point, and measurements of  $\alpha$ ,  $M$ , and  $C_m$  are made. The pitching moment coefficient  $C_m$  is assumed to depend on  $\alpha$  and  $M$  in the manner postulated by the form of the model shown in Eq. (5.1), where  $C_{m_0}$ ,  $C_{m\alpha}$ ,  $C_{mM}$ , and  $C_{m\alpha M}$  are constant model parameters to be determined. Accordingly,  $\alpha$  and  $M$  are called the independent variables, and  $C_m$  is called the dependent variable or the response variable. There may be other variables that could affect  $C_m$ , but these are held fixed during the test, insofar as it is known what the influential variables might be.

Since the independent variables  $\alpha$  and  $M$  are set by the experimenter for each data point, it is assumed that they can be measured without error. The dependent variable  $C_m$  is subject to random measurement errors, and is therefore a random variable. The random error term  $v_m$  includes random effects of unknown influences and random measurement errors in the dependent variable.

The model in Eq. (5.1) is linear in the parameters, although the quantities that the parameters multiply are both linear and nonlinear functions of the independent variables. The term “linear regression” refers to the linearity of the equation for the model output [e.g., Eq. (5.1)] with respect to the model parameters, not with respect to the independent variables. As shown in Eq. (5.1), a linear regression problem can have modeling functions that are linear or nonlinear functions of the independent variables.

In the preceding wind-tunnel example, measured values of  $C_m$  are obtained directly by nondimensionalizing the pitching moment measured by a strain-gauge balance installed between the model and the mounting system in the wind tunnel. For flight-test data, the measured values of nondimensional force and moment coefficients cannot be obtained in this way, but must instead

be computed from measurements of the aircraft translational and rotational motion, along with the geometric and mass/inertia properties of the aircraft, and the equations of motion. Values of  $C_m$  are therefore computed from other measurements, rather than measured directly, as in the wind-tunnel experiment. For the case of pitching moment coefficient  $C_m$ , the measured values are computed from Eq. (3.37b),

$$C_m = \frac{1}{\bar{q}S\bar{c}} [I_y\dot{q} + (I_x - I_z)pr + I_{xz}(p^2 - r^2) - I_p\Omega_p r] \quad (5.2)$$

where the pitch rate derivative  $\dot{q}$  is normally not measured directly, but rather is obtained using a smoothed numerical derivative of measured pitch rate  $q$  (see Chapter 11).

Data are collected for independent variables and response variables at each sample time, and the modeling problem based on flight-test data is then similar to that for the wind-tunnel experiment. In the case of flight-test data analysis, control surface deflections and motion variables such as nondimensional pitch rate can also enter into the model for  $C_m$ . Wind-tunnel tests can also include these independent variables, but the essential difference is that in flight testing, more than one of the independent variables change at the same time, in a manner dictated by the aircraft dynamics. As a result, these variables cannot be varied independently in flight, as is done in wind-tunnel experiments.

For example, changing the angle of attack on an aircraft flying in still air requires a control surface deflection, whereas in a wind-tunnel experiment the angle of attack can be changed at will with the control surface deflections held constant. It follows that the independent variables in a flight test are not really independent; however, this terminology is commonly used in practice, and will be employed here also. This issue complicates experiment design for flight testing, but the fundamental modeling problem of finding a mathematical relationship among measured variables is the same.

For flight-test data, modeling the values of  $C_m$  obtained from the pitching moment equation in a least-squares sense will minimize the squared error in the pitching moment equation when an identified model is substituted for  $C_m$  [cf. Eq. (5.2)]. For this reason, the method is also called equation error.

The modeling example in Eq. (5.1) can be generalized to the following model form for relating independent variables to a dependent variable,

$$y = \theta_o + \sum_{j=1}^n \theta_j \xi_j \quad (5.3)$$

where  $y$  is the dependent variable; the  $\xi_j$  are linear or nonlinear functions of the  $m$  independent variables  $x_1, x_2, \dots, x_m$ ; and the model parameters  $\theta_o, \theta_1, \theta_2, \dots, \theta_n$  are constants that quantify the influence of each term on the dependent variable  $y$ . The notation has been simplified by omitting the explicit dependence of the modeling functions  $\xi_j$  on the independent variables, i.e.,  $\xi_j \equiv \xi_j(x_1, x_2, \dots, x_m)$ . The parameter  $\theta_o$  can be assumed to multiply the constant 1, and models the bias in the dependent variable. The measured values of the dependent variable

## REGRESSION METHODS

97

are corrupted by random measurement noise, so that

$$z(i) = \theta_0 + \sum_{j=1}^n \theta_j \xi_j(i) + v(i) \quad i = 1, 2, \dots, N \quad (5.4)$$

where  $z(i)$  are the output measurements,  $N$  is the number of data points, and the  $\xi_j(i)$  depend on the  $m$  independent variables  $x_1, x_2, \dots, x_m$  at the  $i$ th data point. Equation (5.4) is called the regression equation, and  $\xi_j, j = 1, 2, \dots, n$ , are called the regressors. It is assumed that the independent variables  $x_1, x_2, \dots, x_m$  are measured without error, and the mathematical forms of the dependence of regressor functions  $\xi_j$  on the independent variables are postulated and therefore known. As a result, the regressors are assumed to be known without error. The quantity  $v$  is the measurement error. Sometimes the dependent variable  $y$  is also called the output variable, and the independent variables  $x_k, k = 1, 2, \dots, m$ , are called input variables.

Following the material in Chapter 4, the least-squares model will be assumed initially, wherein the model parameters are assumed to be unknown constants, and the output measurements are corrupted by a vector of random noise. The next task then is to find an estimate  $\hat{\boldsymbol{\theta}}$  of the parameter vector  $\boldsymbol{\theta}^T = [\theta_0 \ \theta_1 \ \dots \ \theta_n]$ , based on the least-squares principle. The solution to this problem will be discussed in detail in this chapter, along with methods for characterizing the accuracy of the estimated parameters.

There is also practical interest in choosing an appropriate mathematical model structure for the regression equation, i.e., deciding what terms should appear in model equations like Eq. (5.1), or its more general form, Eq. (5.4). In the foregoing discussion, it was assumed that the postulated model structure was adequate to characterize the measured data. In practice, an adequate model structure is not always known *a priori* and therefore must be identified from the measured data using a model structure determination procedure. Approaches to diagnosing model structure deficiencies and solving the model structure determination problem are discussed in this chapter.

Additional problems related to accurate parameter estimation can be caused by near-linear dependencies among the regressors. Methods for detecting and assessing these dependencies must be used, along with new methods for obtaining accurate parameter estimates. The last part of the chapter is concerned with this problem.

This chapter contains essential information for using linear regression in aircraft system identification. Comprehensive treatments of linear regression can be found in Refs. 1–3.

### 5.1 Ordinary Least Squares

The general form of the model equation (5.3) and the regression equation (5.4) can be written using vector and matrix notation as

$$\mathbf{y} = \mathbf{X}\boldsymbol{\theta} \quad (5.5)$$

and

$$\mathbf{z} = \mathbf{X}\boldsymbol{\theta} + \boldsymbol{\nu} \quad (5.6)$$

where

$$\mathbf{z} = [z(1) \ z(2) \ \cdots \ z(N)]^T = N \times 1 \text{ vector}$$

$$\boldsymbol{\theta} = [\theta_0 \ \theta_1 \ \cdots \ \theta_n]^T = n_p \times 1 \text{ vector of unknown parameters, } n_p = n + 1$$

$$\mathbf{X} = [1 \ \boldsymbol{\xi}_1 \ \cdots \ \boldsymbol{\xi}_n] = N \times n_p \text{ matrix of vectors of ones and regressors}$$

$$\boldsymbol{\nu} = [\nu(1) \ \nu(2) \ \cdots \ \nu(N)]^T = N \times 1 \text{ vector of measurement errors}$$

The regressor vectors  $\boldsymbol{\xi}_j, j = 1, 2, \dots, n$ , are known postulated functions of the vectors of independent variables. Usually, at least some of the regressors are equal to the independent variables themselves.

Regression equation (5.6) is equivalent to measurement equation (4.2), with  $\mathbf{H} = \mathbf{X}$ . For the least-squares model, there are no probability statements regarding  $\boldsymbol{\theta}$  or  $\boldsymbol{\nu}$ , but  $\boldsymbol{\nu}$  is assumed to be zero mean and uncorrelated, with constant variance,

$$E(\boldsymbol{\nu}) = \mathbf{0} \quad E(\boldsymbol{\nu}\boldsymbol{\nu}^T) = \sigma^2 \mathbf{I} \quad (5.7)$$

As discussed in Chapter 4, the best estimator of  $\boldsymbol{\theta}$  in a least-squares sense comes from minimizing the sum of squared differences between the measurements and the model,

$$J(\boldsymbol{\theta}) = \frac{1}{2}(\mathbf{z} - \mathbf{X}\boldsymbol{\theta})^T(\mathbf{z} - \mathbf{X}\boldsymbol{\theta}) \quad (5.8)$$

The parameter estimate  $\hat{\boldsymbol{\theta}}$  that minimizes the cost function  $J(\boldsymbol{\theta})$  must satisfy

$$\frac{\partial J}{\partial \boldsymbol{\theta}} = -\mathbf{X}^T \mathbf{z} + \mathbf{X}^T \mathbf{X} \hat{\boldsymbol{\theta}} = \mathbf{0} \quad (5.9a)$$

or

$$\mathbf{X}^T \mathbf{X} \hat{\boldsymbol{\theta}} = \mathbf{X}^T \mathbf{z} \quad (5.9b)$$

or

$$\mathbf{X}^T (\mathbf{z} - \mathbf{X} \hat{\boldsymbol{\theta}}) = \mathbf{0} \quad (5.9c)$$

The  $n_p = n + 1$  equations represented in Eqs. (5.9) are called the normal equations. The solution of these equations for the unknown parameter vector  $\boldsymbol{\theta}$  gives the formula for the least-squares estimator, also called the ordinary

## REGRESSION METHODS

99

least-squares estimator,

$$\hat{\boldsymbol{\theta}} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{z} \quad (5.10)$$

The  $n_p \times n_p$  matrix  $\mathbf{X}^T \mathbf{X}$  is always symmetric. If the regressor vectors that make up the columns of  $\mathbf{X}$  are linearly independent, then  $\mathbf{X}^T \mathbf{X}$  is positive definite, the eigenvalues of  $\mathbf{X}^T \mathbf{X}$  are positive real numbers, the associated eigenvectors are mutually orthogonal, and  $(\mathbf{X}^T \mathbf{X})^{-1}$  exists (see Appendix A). Note also from Eq. (5.9a) that the second gradient of the cost function with respect to the parameter vector is  $\mathbf{X}^T \mathbf{X}$ , which is positive definite, indicating a minimum rather than a maximum.

The covariance matrix of the parameter estimate  $\hat{\boldsymbol{\theta}}$ , also known as the covariance matrix of the estimation error  $\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}$ , is

$$\begin{aligned} \text{Cov}(\hat{\boldsymbol{\theta}}) &\equiv E[(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})^T] = E\{(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T (\mathbf{z} - \mathbf{y})(\mathbf{z} - \mathbf{y})^T \mathbf{X} (\mathbf{X}^T \mathbf{X})^{-1}\} \\ &= (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T E(\mathbf{v} \mathbf{v}^T) \mathbf{X} (\mathbf{X}^T \mathbf{X})^{-1} \end{aligned} \quad (5.11)$$

where the true parameter vector  $\boldsymbol{\theta}$  is related to the true output  $\mathbf{y}$  by  $\boldsymbol{\theta} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$ . Assuming the measurement errors are uncorrelated and have constant variance  $\sigma^2$ ,  $E(\mathbf{v} \mathbf{v}^T) = \sigma^2 \mathbf{I}$  [cf. Eq. (5.7)]. The expression for the covariance matrix of  $\hat{\boldsymbol{\theta}}$  is then simplified to

$$\text{Cov}(\hat{\boldsymbol{\theta}}) = E[(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})^T] = \sigma^2 (\mathbf{X}^T \mathbf{X})^{-1} \quad (5.12)$$

Note that the matrix  $(\mathbf{X}^T \mathbf{X})^{-1}$  was also required to compute the least-squares parameter estimate in Eq. (5.10). Defining the matrix  $\mathcal{D}$  as

$$\mathcal{D} \equiv (\mathbf{X}^T \mathbf{X})^{-1} = [d_{jk}] \quad j, k = 1, 2, \dots, n_p \quad (5.13)$$

the variance of the  $j$ th estimated parameter in the parameter vector  $\hat{\boldsymbol{\theta}}$  is the  $j$ th diagonal element of the covariance matrix, or

$$\text{Var}(\hat{\theta}_j) = \sigma^2 d_{jj} \equiv s^2(\hat{\theta}_j) \quad j = 1, 2, \dots, n_p \quad (5.14)$$

and the covariance between two estimated parameters  $\hat{\theta}_j$  and  $\hat{\theta}_k$  is

$$\text{Cov}(\hat{\theta}_j, \hat{\theta}_k) = \sigma^2 d_{jk} \quad j, k = 1, 2, \dots, n_p \quad (5.15)$$

The correlation coefficient  $r_{jk}$  is defined as

$$r_{jk} \equiv \frac{d_{jk}}{\sqrt{d_{jj} d_{kk}}} = \frac{\text{Cov}(\hat{\theta}_j, \hat{\theta}_k)}{\sqrt{\text{Var}(\hat{\theta}_j) \text{Var}(\hat{\theta}_k)}} \quad j, k = 1, 2, \dots, n_p \quad (5.16a)$$

$$-1 \leq r_{jk} \leq 1 \quad j, k = 1, 2, \dots, n_p \quad (5.16b)$$

The correlation coefficient  $r_{jk}$  is a measure of the pair-wise correlation between parameter estimates  $\hat{\theta}_j$  and  $\hat{\theta}_k$ . A value of  $r_{jk} = 1$  means that estimated parameters  $\hat{\theta}_j$  and  $\hat{\theta}_k$  are linearly related or, equivalently, that their corresponding regressors are linearly dependent. When  $r_{jk} = -1$ , the same statements apply, except that the linear relationships include a negative sign. Arranging all the values of  $r_{jk}$  in an  $n_p \times n_p$  matrix forms the parameter correlation matrix  $\text{Corr}(\hat{\theta})$ , which is symmetric with ones on the main diagonal,

$$\text{Corr}(\hat{\theta}) = [r_{jk}] \quad j, k = 1, 2, \dots, n_p \quad (5.17)$$

The parameter correlation matrix can also be computed by the following matrix multiplication,

$$\text{Corr}(\hat{\theta}) = \left[ \begin{array}{cccc} \frac{1}{s(\hat{\theta}_1)} & 0 & \dots & 0 \\ 0 & \frac{1}{s(\hat{\theta}_2)} & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & \frac{1}{s(\hat{\theta}_{n_p})} \end{array} \right] \text{Cov}(\hat{\theta}) \left[ \begin{array}{cccc} \frac{1}{s(\hat{\theta}_1)} & 0 & \dots & 0 \\ 0 & \frac{1}{s(\hat{\theta}_2)} & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & \frac{1}{s(\hat{\theta}_{n_p})} \end{array} \right] \quad (5.18)$$

Using Eqs. (5.5) and (5.10), the estimated dependent variable vector  $\hat{y}$ , based on the vector of measured outputs  $z$  and the regressor matrix  $X$ , is given by

$$\hat{y} = X\hat{\theta} = X(X^T X)^{-1} X^T z = Kz \quad (5.19)$$

where

$$K = X(X^T X)^{-1} X^T \quad (5.20)$$

is the  $N \times N$  prediction matrix that maps the measured outputs to the estimated outputs. The differences between the measured values  $z$  and the estimated values  $\hat{y}$  are the residuals, which form the vector  $v$ ,

$$\begin{aligned} v &= z - \hat{y} = z - X\hat{\theta} \\ &= z - X(X^T X)^{-1} X^T z \\ &= (I - K)z \end{aligned} \quad (5.21)$$

The prediction matrix and the residuals will be discussed further in Sec. 5.2.4.

## REGRESSION METHODS

101

The calculation of the parameter covariance matrix from Eq. (5.12) requires the measurement error variance  $\sigma^2$ , which was assumed constant for all the data points when deriving Eq. (5.12). In practice,  $\sigma^2$  is usually not known *a priori* and therefore must be estimated from the measured data. An unbiased estimate of  $\sigma^2$  can be computed from repeated measurements at the same independent variable settings using

$$\hat{\sigma}^2 = \frac{1}{(n_r - 1)} \sum_{i=1}^{n_r} [z_r(i) - \bar{z}_r]^2 \equiv s^2 \quad (5.22)$$

where  $n_r$  is the number of repeated measurements  $z_r$ , and  $\bar{z}_r$  is the mean value of the  $n_r$  repeated measurements,

$$\bar{z}_r = \frac{1}{n_r} \sum_{i=1}^{n_r} z_r(i) \quad (5.23)$$

For time series from flight data, where it is unlikely that the same independent variable settings will be exactly repeated,  $\sigma^2$  can be estimated independently using smoothing methods (see Chapter 11). Assuming the model structure is adequate, an unbiased estimate for  $\sigma^2$  can also be obtained based on the residuals,

$$\hat{\sigma}^2 = \frac{\mathbf{v}^T \mathbf{v}}{(N - n_p)} = \frac{\sum_{i=1}^N [z(i) - \hat{y}(i)]^2}{(N - n_p)} \equiv s^2 \quad (5.24)$$

The square root of  $s^2$  is sometimes called the fit error, which indicates how close the estimates  $\hat{y}(i)$  are to the measured values  $z(i)$ , using the same units as the measured and estimated values, rather than their squares. Note that the estimator in Eq. (5.24) depends on the model, because of the appearance of  $\hat{y}(i)$  in the calculation. Consequently, when Eq. (5.24) is used, the estimate of  $\sigma^2$  depends on the postulated model structure and associated parameter estimates, instead of being based solely on the measured data.

Another metric that quantifies the closeness of  $\hat{y}(i)$  to  $z(i)$  is the coefficient of determination,  $R^2$ . The definition of  $R^2$  follows from partitioning the total sum of squared variations in the measured output  $z$  about its mean value into the sum of squared variations of the estimate  $\hat{y}$  about the same mean value, plus the sum of squared variations of the measurement  $z$  about the estimate  $\hat{y}$ . These quantities are called the total sum of squares  $SS_T$ , the regression sum of squares  $SS_R$ , and the residual sum of squares  $SS_E$ , respectively. The three sums are defined by

$$SS_T \equiv \sum_{i=1}^N [z(i) - \bar{z}]^2 = \mathbf{z}^T \mathbf{z} - N\bar{z}^2 \quad (5.25)$$

where

$$\bar{z} = \frac{1}{N} \sum_{i=1}^N z(i) \quad (5.26)$$

$$SS_R \equiv \sum_{i=1}^N [\hat{y}(i) - \bar{z}]^2 \quad (5.27)$$

$$\begin{aligned} SS_E &\equiv \sum_{i=1}^N [z(i) - \hat{y}(i)]^2 = (\mathbf{z} - \mathbf{X}\hat{\boldsymbol{\theta}})^T(\mathbf{z} - \mathbf{X}\hat{\boldsymbol{\theta}}) \\ &= \mathbf{z}^T\mathbf{z} - 2\hat{\boldsymbol{\theta}}^T\mathbf{X}^T\mathbf{z} + \hat{\boldsymbol{\theta}}^T\mathbf{X}^T\mathbf{X}\hat{\boldsymbol{\theta}} \\ &= \mathbf{z}^T\mathbf{z} - 2\hat{\boldsymbol{\theta}}^T\mathbf{X}^T\mathbf{z} + \hat{\boldsymbol{\theta}}^T\mathbf{X}^T\mathbf{z} \\ &= \mathbf{z}^T\mathbf{z} - \hat{\boldsymbol{\theta}}^T\mathbf{X}^T\mathbf{z} \end{aligned} \quad (5.28)$$

where the third line in Eq. (5.28) follows from the normal equation (5.9b).

The relationship among the three sums  $SS_T$ ,  $SS_R$ , and  $SS_E$  is

$$SS_T = SS_R + SS_E \quad (5.29)$$

which is derived in Appendix B. From Eqs. (5.29), (5.25), and (5.28),

$$SS_R = SS_T - SS_E = \hat{\boldsymbol{\theta}}^T\mathbf{X}^T\mathbf{z} - N\bar{z}^2 \quad (5.30)$$

In other words, Eq. (5.29) says that the total sum of squares of the measurements about the mean can be partitioned into the sum of squares of the model about the same mean, plus the sum of squares of the measurements about the model. For good models,  $SS_E$  will include only noise, and  $SS_R$  will be large relative to  $SS_E$ .

The coefficient of determination  $R^2$  represents the proportion of the variation in the measured output that is explained by the model,

$$R^2 = \frac{SS_R}{SS_T} = 1 - \frac{SS_E}{SS_T} = \frac{\hat{\boldsymbol{\theta}}^T\mathbf{X}^T\mathbf{z} - N\bar{z}^2}{\mathbf{z}^T\mathbf{z} - N\bar{z}^2} \quad (5.31)$$

Values of  $R^2$  vary from 0 to 1, where 1 represents a perfect fit to the data; however,  $R^2$  is usually expressed as a percentage. Further information on the coefficient of determination will be given in Sec. 5.4.

### 5.1.1 Properties of the Least-Squares Estimator

The least-squares estimator was developed under the assumptions of a postulated model that is linear in the parameters, deterministic regressors, and white measurement noise with constant variance. Under these assumptions, and using

## REGRESSION METHODS

103

the definitions in Chapter 4, the properties of the least-squares estimates are as follows:

- 1) The least-squares estimator is unbiased:

$$\begin{aligned}
 E(\hat{\boldsymbol{\theta}}) &= E[(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{z}] = E[(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T (\mathbf{X}\boldsymbol{\theta} + \boldsymbol{\nu})] \\
 &= E(\boldsymbol{\theta}) + (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T E(\boldsymbol{\nu}) \\
 &= \boldsymbol{\theta}
 \end{aligned} \tag{5.32}$$

- 2) The least-squares estimator is a minimum variance and efficient estimator. This means that  $\hat{\boldsymbol{\theta}}$  is the best unbiased estimate that is a linear function of the measurements, and the parameter accuracy is given by the parameter variance from Eq. (5.14).
- 3) The least-squares estimator is consistent. The practical meaning of this property is that as the number of data points increases, the parameter estimates will converge to their true values.
- 4) If the measurement noise is assumed Gaussian, where  $\boldsymbol{\nu}$  is  $\mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})$ , then the least-squares model for the measurement process becomes the Fisher model. It follows from Chapter 4 that in this situation the least-squares estimator is a maximum likelihood estimator.

In practical aircraft problems, the assumptions stated here are often violated, so that these theoretical properties do not hold. Section 5.1.7 and the following material in this chapter describe approaches used in practice to account for violation of the assumptions in the theory.

### 5.1.2 Confidence Intervals

When estimating the model parameters in a linear regression problem, it is of interest to know the quality of the parameter estimates obtained. The quality of the parameter estimates can be evaluated in terms of confidence intervals for the model parameter estimates, the estimated output, and the prediction of outputs for new data that were not used to identify the model. Procedures used to quantify these confidence intervals require an additional assumption that the measurement errors are normally distributed, i.e.,

$$\boldsymbol{\nu} \text{ is } \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})$$

Then, from the linearity of the least-squares estimator with respect to the measurements [cf. Eq. (5.10)], along with the assumption that the regressors are deterministic, it follows that

$$\begin{aligned}
 \hat{\boldsymbol{\theta}} &\text{ is } \mathcal{N}(\boldsymbol{\theta}, \sigma^2 (\mathbf{X}^T \mathbf{X})^{-1}) \\
 \hat{\mathbf{y}} &\text{ is } \mathcal{N}(\mathbf{X}\boldsymbol{\theta}, \sigma^2 \mathbf{K})
 \end{aligned} \tag{5.33}$$

Since any linear function of a Gaussian random variable is also Gaussian (see Appendix B). The parameter estimates from repeated experiments should

therefore have a Gaussian distribution about the true value, and similarly for the estimated outputs.

*Confidence interval on estimated parameters.* The normal distribution of the estimated parameter vector  $\hat{\theta}$  implies that each element  $\hat{\theta}_j$  is also normally distributed, with mean value  $\theta_j$  and variance  $\sigma^2 d_{jj}$ , where  $d_{jj}$  is the  $j$ th diagonal element of the  $(X^T X)^{-1}$  matrix. Consequently, each of the statistics

$$t = \frac{\hat{\theta}_j - \theta_j}{\sqrt{\hat{\sigma}^2 d_{jj}}} = \frac{\hat{\theta}_j - \theta_j}{s(\hat{\theta}_j)} \quad j = 1, 2, \dots, n_p \quad (5.34)$$

has the  $t$ -distribution with  $N - n_p$  degrees of freedom (see Appendix B). A common value selected for the confidence level is 95%, so that the fraction of the two-sided  $t$ -distribution excluded is  $\alpha = 0.05$ . Accordingly, a  $100(1 - \alpha)$  percent confidence interval for the parameter  $\hat{\theta}_j$  is

$$\hat{\theta}_j - t(\alpha/2, N - n_p)s(\hat{\theta}_j) \leq \theta_j \leq \hat{\theta}_j + t(\alpha/2, N - n_p)s(\hat{\theta}_j) \quad j = 1, 2, \dots, n_p \quad (5.35)$$

where  $\alpha = 0.05$  for a 95% confidence level. From the table of  $t$ -distribution values, e.g., in Ref. 3,  $t(\alpha/2, N - n_p) \approx 1.96$  for  $N - n_p > 100$ , and  $\alpha = 0.05$ . Then the confidence interval can be written as

$$\theta_j = \hat{\theta}_j \pm 1.96 s(\hat{\theta}_j) \quad j = 1, 2, \dots, n_p \quad (5.36)$$

In practical flight-test applications,  $N - n_p \gg 100$ , and the  $t$ -distribution approaches the normal distribution, so that a confidence interval corresponding to two standard deviations is often used:

$$\theta_j = \hat{\theta}_j \pm 2s(\hat{\theta}_j) \quad j = 1, 2, \dots, n_p \quad (5.37)$$

For normally distributed measurement errors, this can be interpreted as a 95% probability that the true value of the parameter  $\theta_j$  will be within the interval  $[\hat{\theta}_j - 2s(\hat{\theta}_j), \hat{\theta}_j + 2s(\hat{\theta}_j)]$ .

*Confidence interval on estimated output.* The confidence interval for the estimated output can be constructed at each data point. At the  $i$ th data point, the vector of regressors is

$$\mathbf{x}^T(i) = [1 \quad \xi_1(i) \quad \xi_2(i) \quad \cdots \quad \xi_n(i)] \quad (5.38)$$

The quantity  $\mathbf{x}^T(i)$  is the  $i$ th row of the regressor matrix  $X$ . The regressors  $\xi_1(i), \xi_2(i), \dots, \xi_n(i)$  are functions of the measured independent variables

## REGRESSION METHODS

105

(typically aircraft states and controls) at the  $i$ th data point, although this is not shown in the notation.

The estimated output at that point is

$$\hat{y}(i) = \mathbf{x}^T(i)\hat{\boldsymbol{\theta}} \quad (5.39)$$

and the variance of  $\hat{y}(i)$  is

$$\begin{aligned} \text{Var}[\hat{y}(i)] &= E\{[\hat{y}(i) - y(i)][\hat{y}(i) - y(i)]^T\} \\ &= E\{[\mathbf{x}^T(i)\hat{\boldsymbol{\theta}} - \mathbf{x}^T(i)\boldsymbol{\theta}][\mathbf{x}^T(i)\hat{\boldsymbol{\theta}} - \mathbf{x}^T(i)\boldsymbol{\theta}]^T\} \\ &= \mathbf{x}^T(i)E[(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})^T]\mathbf{x}(i) \\ &= \sigma^2 \mathbf{x}^T(i)(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{x}(i) \\ &\equiv s^2[\hat{y}(i)] \end{aligned} \quad (5.40)$$

The  $100(1 - \alpha)$  percent confidence interval for the output estimate at the  $i$ th data point is

$$y(i) = \hat{y}(i) \pm t(\alpha/2, N - n_p) s[\hat{y}(i)] \quad (5.41)$$

where  $s[\hat{y}(i)]$  is computed from Eq. (5.40), with  $\sigma^2$  replaced by its estimate from Eq. (5.24) or from an independent estimate, as described earlier.

*Confidence interval on predicted output.* The quantity  $\hat{y}(i)$  can be interpreted in two ways. First, it is the estimate of the output at the  $i$ th data point. In this context,  $\hat{y}(i)$  is sometimes called the fitted value to the measurement  $z(i)$ . The variance of  $\hat{y}(i)$  given by Eq. (5.40) reflects the variation of  $\hat{y}$  at  $\mathbf{x}(i)$  for repeated regressions with the same sample size and the same regressors.

In the second interpretation,  $\hat{y}(i)$  is the predicted value of  $y$  at  $\mathbf{x}(i)$ . For assessment of  $\hat{y}(i)$  as a predictor, consider  $z(i)$  to be a new single measurement at  $\mathbf{x}(i)$ . This measurement is assumed to be independent of  $\hat{y}(i)$ . The prediction error variance is

$$\begin{aligned} \text{Var}[z(i) - \hat{y}(i)] &= E\{[z(i) - \hat{y}(i)][z(i) - \hat{y}(i)]\} \\ &= \sigma^2 E\{[v(i) - [\hat{y}(i) - y(i)]][v(i) - [\hat{y}(i) - y(i)]]\} \\ &= E[v(i)v(i)] + E\{[\hat{y}(i) - y(i)][\hat{y}(i) - y(i)]\} \\ &= \sigma^2 [1 + \mathbf{x}^T(i)(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{x}(i)] \\ &\equiv s^2[z(i) - \hat{y}(i)] \end{aligned} \quad (5.42)$$

Note that the variance of  $z(i) - \hat{y}(i)$  is the sum of the measurement error variance  $\sigma^2$  and the variance of the estimated output  $\hat{y}(i)$  from Eq. (5.40). Using Eq. (5.42)

with an estimated measurement error variance  $\hat{\sigma}^2$ , the prediction interval is given by

$$y(i) = \hat{y}(i) \pm t(\alpha/2, N - n_p) s[z(i) - \hat{y}(i)] \quad (5.43)$$

### 5.1.3 Hypothesis Testing

In the analysis of the linear regression problem up to this point, it has been assumed that the regressors necessary for an adequate model were known. In practice, this is often not true, so there must be some tests for adding or deleting terms from a proposed model structure, based on statistics that can be computed from the measured data. Three of these tests will be outlined, and later used in the selection of an adequate model structure from measured data. In the following development, the Fisher model

$$\begin{aligned} z &= \mathbf{X}\boldsymbol{\theta} + \boldsymbol{\nu} \\ \boldsymbol{\nu} &\text{ is } \mathbb{N}(\mathbf{0}, \sigma^2 \mathbf{I}) \end{aligned} \quad (5.44)$$

with  $n$  regressors and  $n_p = n + 1$  parameters will be considered. This is identical to a least-squares model with weighting matrix chosen to be the same as for a Fisher model with white Gaussian measurement noise of constant variance.

*Test for significance of the regression.* The objective of the test for the significance of the regression is to determine if there is a linear relationship between the measured dependent variable vector  $z$  and any of a set of candidate regressor vectors  $\xi_j, j = 1, 2, \dots, n$ . The bias term in the model, which has a regressor vector consisting of a vector of ones, is included in the model by default, because an estimate of the bias term is always useful, even if it turns out to be zero.

The related hypotheses, called the null hypothesis  $H_0$ , and the alternative hypothesis  $H_1$ , are stated as

$$\begin{aligned} H_0: \theta_1 &= \theta_2 = \dots = \theta_n = 0 \\ H_1: \theta_j &\neq 0 \text{ for at least one } j \end{aligned} \quad (5.45)$$

Since the constant term is always assumed to be included in the model, it is not included in  $H_0$  or  $H_1$ . If the null hypothesis is rejected, then at least one of the regressors  $\xi_j, j = 1, 2, \dots, n$ , significantly contributes to modeling the variation in the dependent variable. If the null hypothesis  $H_0$  is true, then the sum of squares due to the regression  $SS_R$  divided by its degrees of freedom will be an estimate of the fit error variance, using only the bias term in the model. The residual sum of squares  $SS_E$  divided by its degrees of freedom is also an estimate of the fit error variance. The relevant test statistic for  $H_0$  is therefore the  $F$ -statistic for the ratio of squared random variables

## REGRESSION METHODS

107

(see e.g., Ref. 3),

$$F_0 = \frac{SS_R/n}{SS_E/(N - n_p)} = \frac{\hat{\boldsymbol{\theta}}^T \mathbf{X}^T \mathbf{z} - N\bar{z}^2}{ns^2} \quad (5.46)$$

The statistic  $F_0$  is a random variable that follows the  $F$ -distribution (see Appendix B), with the number of degrees of freedom  $n$  for the numerator and  $N - n_p$  for the denominator. The null hypothesis is rejected if  $F_0 > F(\alpha; n, N - n_p)$ , where  $F(\alpha; n, N - n_p)$  can be found from tabulated values of the  $F$ -distribution for a selected significance level  $\alpha$ ,  $0 \leq \alpha \leq 1$ . If the null hypothesis is rejected at the selected significance level  $\alpha$ , that is equivalent to the statement that not all of the model parameters are zero, with  $100(1 - \alpha)$  percent confidence. The situation is also sometimes described by saying that  $F_0$  is statistically significant at the  $100 \alpha$  percent level. If the null hypothesis is rejected, then one or more of the model terms is explaining some of the variation in the dependent variable, which causes the numerator of  $F_0$  to be larger than it would be if none of the model terms were significant. Note that the selected confidence level  $(1 - \alpha)$  and the statistical inference are necessary because of the uncertainty associated with chance variations introduced by the measurement noise. For aircraft system identification, it is typical to choose  $\alpha = 0.05$ , which corresponds to a 95% confidence level.

*Test for significance of a subset of model parameters.* The statistical significance testing described earlier can also be used to investigate the contribution of a subset of one or more regressors to modeling the variation in the dependent variable. For this investigation, the parameter vector is partitioned as follows:

$$\boldsymbol{\theta} = \begin{bmatrix} \boldsymbol{\theta}_1 \\ \boldsymbol{\theta}_2 \end{bmatrix} \quad (5.47)$$

where  $\boldsymbol{\theta}_1$  is a  $p \times 1$  vector of model parameters that includes the bias term, and  $\boldsymbol{\theta}_2$  is a  $q \times 1$  vector of unknown parameters associated with terms being considered for inclusion in the model, where  $q < n$  and  $p + q = n_p$ . For the partitioned parameter vector, the regression equations changes to

$$\mathbf{z} = \mathbf{X}_1 \boldsymbol{\theta}_1 + \mathbf{X}_2 \boldsymbol{\theta}_2 + \boldsymbol{\nu} \quad (5.48)$$

To determine if the subset of regressors in the columns of  $\mathbf{X}_2$  contribute significantly to the model, the hypotheses are

$$\begin{aligned} H_0: \boldsymbol{\theta}_2 &= \mathbf{0} \\ H_1: \boldsymbol{\theta}_2 &\neq \mathbf{0} \end{aligned} \quad (5.49)$$

For the full model, it is known that

$$\begin{aligned}\hat{\boldsymbol{\theta}} &= (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{z} \\ SS_R(\hat{\boldsymbol{\theta}}) &= \hat{\boldsymbol{\theta}}^T \mathbf{X}^T \mathbf{z} - N\bar{z}^2 \\ SS_E(\hat{\boldsymbol{\theta}}) &= \mathbf{z}^T \mathbf{z} - \hat{\boldsymbol{\theta}}^T \mathbf{X}^T \mathbf{z}\end{aligned}\quad (5.50)$$

To find the contribution of the  $\mathbf{X}_2 \boldsymbol{\theta}_2$  terms to the model, it is necessary to estimate the model parameters assuming that the null hypothesis is true. The reduced model is

$$\mathbf{z} = \mathbf{X}_1 \boldsymbol{\theta}_1 + \boldsymbol{\nu} \quad (5.51)$$

with the solution

$$\begin{aligned}\hat{\boldsymbol{\theta}}_1 &= (\mathbf{X}_1^T \mathbf{X}_1)^{-1} \mathbf{X}_1^T \mathbf{z} \\ SS_R(\hat{\boldsymbol{\theta}}_1) &= \hat{\boldsymbol{\theta}}_1^T \mathbf{X}_1^T \mathbf{z} - N\bar{z}^2 \\ SS_E(\hat{\boldsymbol{\theta}}_1) &= \mathbf{z}^T \mathbf{z} - \hat{\boldsymbol{\theta}}_1^T \mathbf{X}_1^T \mathbf{z}\end{aligned}\quad (5.52)$$

Then the regression sum of squares due to  $\mathbf{X}_2 \boldsymbol{\theta}_2$  given that  $\mathbf{X}_1 \boldsymbol{\theta}_1$  is already in the model, denoted by  $SS_R(\hat{\boldsymbol{\theta}}_2 | \hat{\boldsymbol{\theta}}_1)$ , follows from the relation

$$SS_R(\hat{\boldsymbol{\theta}}_2 | \hat{\boldsymbol{\theta}}_1) = SS_R(\hat{\boldsymbol{\theta}}) - SS_R(\hat{\boldsymbol{\theta}}_1) \quad (5.53)$$

Using Eqs. (5.50) and (5.52),

$$\begin{aligned}SS_R(\hat{\boldsymbol{\theta}}_2 | \hat{\boldsymbol{\theta}}_1) &= [\hat{\boldsymbol{\theta}}^T \mathbf{X}^T \mathbf{z} - \hat{\boldsymbol{\theta}}_1^T \mathbf{X}_1^T \mathbf{z}] = [\hat{\boldsymbol{\theta}}^T \mathbf{X}^T - \hat{\boldsymbol{\theta}}_1^T \mathbf{X}_1^T] \mathbf{z} \\ &= \mathbf{z}^T [\mathbf{X}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T - \mathbf{X}_1(\mathbf{X}_1^T \mathbf{X}_1)^{-1} \mathbf{X}_1^T] \mathbf{z}\end{aligned}\quad (5.54)$$

The associated  $F$  statistic for testing the null hypothesis (5.49) is

$$F_0 = \frac{SS_R(\hat{\boldsymbol{\theta}}_2 | \hat{\boldsymbol{\theta}}_1)}{qs^2} \quad (5.55)$$

If  $F_0 > F(\alpha; q, N - n_p)$ , the null hypothesis is rejected, so at least one of the parameters in  $\boldsymbol{\theta}_2$  is not zero. Therefore, at least one of the regressors in  $\mathbf{X}_2$  contributes significantly to the regression model. Significance testing using the statistic (5.55) is sometimes called a partial  $F$ -test, because it is concerned with the contribution of the regressors in  $\mathbf{X}_2$  given that the other regressors in  $\mathbf{X}_1$  are already in the model.

## REGRESSION METHODS

109

*Test for significance of a single model parameter.* If the parameter vector  $\theta_2$  in the preceding development is reduced to a scalar, then the significance of an individual regressor to the regression model can be tested by the hypotheses

$$\begin{aligned} H_0: \theta_2 &= 0 \\ H_1: \theta_2 &\neq 0 \end{aligned} \quad (5.56)$$

where  $X_2$  is now a single vector regressor and  $\theta_2$  is a scalar. Then the  $F$  statistic has one degree of freedom in the numerator, and  $N - n_p$  degrees of freedom in the denominator. Defining a new dependent variable  $z_1$  as the original dependent variable adjusted for the terms  $X_1 \hat{\theta}_1$ ,

$$z_1 \equiv z - X_1 \hat{\theta}_1 \quad (5.57)$$

the regression equation is

$$z_1 = X_2 \theta_2 + \nu \quad (5.58)$$

The partial  $F$  statistic is

$$F_0 = \frac{\hat{\theta}_2 X_2^T z_1}{s^2} = \frac{(X_2^T X_2) \hat{\theta}_2^2}{s^2} = \frac{\hat{\theta}_2^2}{s^2(\hat{\theta}_2)} \quad (5.59)$$

where the second equality follows from the normal equation (5.9b), and the last equality comes from Eq. (5.14). Equation (5.59) shows that the partial  $F$  statistic in this case is identical to the square of the  $t$  statistic for parameter  $\theta_2$  with the true value of  $\theta_2$  equal to zero (in accordance with the null hypothesis), and  $N - n_p$  degrees of freedom [cf. Eq. (5.34)]. The distribution of the squared  $t$  statistic is the same as the  $F$  distribution, so that equivalent significance testing can be done using the  $t$  statistic

$$t_0 = \frac{\hat{\theta}_2}{s(\hat{\theta}_2)} \quad (5.60)$$

The null hypothesis is rejected if  $|t_0| > t(\alpha/2, N - n_p)$ . This can be interpreted as a test for  $\hat{\theta}_2$  being statistically different from zero. From Eqs. (5.59) and (5.60),  $t_0^2 = F_0$ .

#### 5.1.4 Analysis of Residuals

The residuals  $\nu(i)$ ,  $i = 1, 2, \dots, N$ , are defined as the difference between the measured output  $z(i)$  and the estimated output  $\hat{y}(i)$

$$\nu(i) = z(i) - \hat{y}(i) \quad i = 1, 2, \dots, N \quad (5.61)$$

The residuals can be interpreted as samples of the measurement errors  $v(i)$ . Any departure from the underlying assumptions on the errors should be seen in the residuals. Analysis of the residuals is an effective method for discovering various types of model deficiencies. Unfortunately, it is not simple to do this, as can be seen from the following development.

Returning to Eq. (5.21), the vector of residuals can be expressed as

$$\mathbf{v} = (\mathbf{I} - \mathbf{K})\mathbf{z} \quad (5.62a)$$

$$\mathbf{K} = \mathbf{X}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \quad (5.62b)$$

where the prediction matrix  $\mathbf{K}$  is symmetric and idempotent, which means that  $\mathbf{K} = \mathbf{K}^T$  and  $\mathbf{K}\mathbf{K} = \mathbf{K}$ . From Eqs. (5.62) and (5.21), the expected value of the residuals is zero,

$$E(\mathbf{v}) = E(\mathbf{z}) - \mathbf{K}E(\mathbf{z}) = \mathbf{X}\boldsymbol{\theta} - \mathbf{X}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{X}\boldsymbol{\theta} = \mathbf{0} \quad (5.63)$$

and the covariance matrix is

$$\begin{aligned} \text{Cov}(\mathbf{v}) &= E(\mathbf{v}\mathbf{v}^T) = E[(\mathbf{I} - \mathbf{K})\mathbf{z}\mathbf{z}^T(\mathbf{I} - \mathbf{K})^T] \\ &= E[(\mathbf{X}\boldsymbol{\theta} + \mathbf{v} - \mathbf{K}\mathbf{X}\boldsymbol{\theta} - \mathbf{K}\mathbf{v})(\mathbf{X}\boldsymbol{\theta} + \mathbf{v} - \mathbf{K}\mathbf{X}\boldsymbol{\theta} - \mathbf{K}\mathbf{v})^T] \\ &= E[(\mathbf{v} - \mathbf{K}\mathbf{v})(\mathbf{v} - \mathbf{K}\mathbf{v})^T] = (\mathbf{I} - \mathbf{K})E(\mathbf{v}\mathbf{v}^T)(\mathbf{I} - \mathbf{K})^T \\ &= \sigma^2(\mathbf{I} - 2\mathbf{K} + \mathbf{K}^2) \\ &= \sigma^2(\mathbf{I} - \mathbf{K}) \end{aligned} \quad (5.64)$$

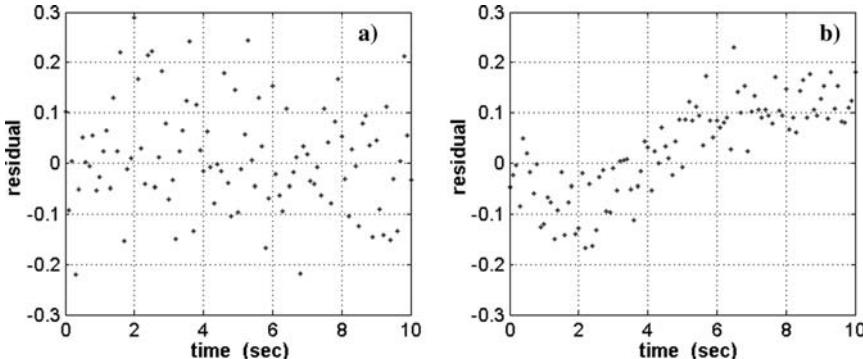
For the  $i$ th residual  $v(i)$ ,

$$\text{Var}[v(i)] = \sigma^2(1 - \kappa_{ii}) \quad (5.65)$$

$$\text{Cov}[v(i), v(j)] = -\sigma^2(\kappa_{ij}) \quad (5.66)$$

where  $\mathbf{K} = [\kappa_{ij}]$  for  $i, j = 1, 2, \dots, N$ . Equations (5.65) and (5.66) indicate that the residuals have different variances and they are correlated. However, it is shown in Ref. 4 that for large  $N$  the average values of the elements of the  $\mathbf{K}$  matrix approach zero, so that  $\text{Cov}(\mathbf{v})$  approaches  $\sigma^2\mathbf{I}$ .

In practice, the analysis of residuals involves the use of various types of plots. The residuals are plotted against time order, because in many cases, including aircraft parameter estimation, the time sequence in which the data were collected is known. The ideal appearance of this type of residual plot is sketched in Fig. 5.1a. The residuals in this figure form a random pattern around zero. On the other hand, the plot in Fig. 5.1b shows some deterministic components in the residuals, which might be the result of a deficiency in the model structure, also called deterministic modeling error.



**Fig. 5.1** Residual plots: a) random (satisfactory) and b) deterministic component remaining (unsatisfactory).

The whiteness of the residuals can be checked by estimating the autocorrelation function of the residuals,

$$\hat{\mathcal{R}}_{vv}(k) = \frac{1}{(N-k)} \sum_{i=1}^{N-k} v(i)v(i+k) \quad k = 0, 1, 2, \dots, N-1 \quad (5.67a)$$

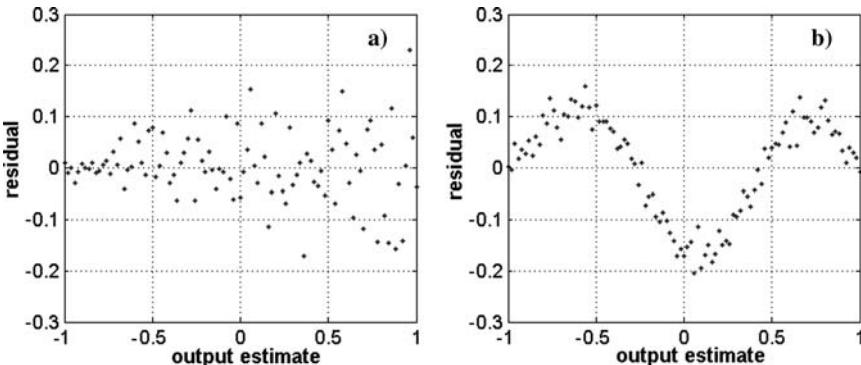
This calculation of  $\hat{\mathcal{R}}_{vv}(k)$  gets very inaccurate for large lag values  $k$ , which make  $(N-k)$  small. Instead, the following expression can be used

$$\hat{\mathcal{R}}_{vv}(k) = \frac{1}{N} \sum_{i=1}^{N-k} v(i)v(i+k) \quad k = 0, 1, 2, \dots, N-1 \quad (5.67b)$$

Theoretically, the  $\hat{\mathcal{R}}_{vv}(k)$  values from Eq. (5.67b) are biased (see Ref. 5). However, in the important practical cases where  $N$  is large relative to  $k$ , the last two expressions give nearly the same result. In practice, for large values of  $k$  relative to  $N$ , Eq. (5.67b) gives more accurate estimates, in spite of the fact that these estimates are biased. Consequently, Eq. (5.67b) is used to compute  $\hat{\mathcal{R}}_{vv}(k)$  for residuals.

If the residuals are completely uncorrelated, then it should be that  $\hat{\mathcal{R}}_{vv}(k) = 0, k \neq 0$ . In practice, even for an adequate model structure, the values of  $\hat{\mathcal{R}}_{vv}(k), k \neq 0$ , are never exactly zero, but instead vary slightly around zero. However, zero should be within  $\pm 2$  standard errors of the residual autocorrelation estimate. The standard error for the residual autocorrelation estimate can be approximated by (see Ref. 6),

$$s[\hat{\mathcal{R}}_{vv}(k \neq 0)] \approx \frac{\hat{\mathcal{R}}_{vv}(0)}{\sqrt{N}} \quad (5.68)$$



**Fig. 5.2 Residual plots: a) variance of residuals not constant and b) unmodeled nonlinearity.**

A plot of residuals  $v(i)$  against the corresponding estimated values  $\hat{y}(i)$  can be useful for detecting several common types of model inadequacy. Two of them are demonstrated in Fig. 5.2. The pattern in Fig. 5.2a indicates that the variance of the residual is not constant. Figure 5.2b shows some nonlinearity not captured by the model.

Another type of residual plot can be used for detecting the deviation of the probability distribution of the residuals from a Gaussian distribution. Small differences between the assumed normal distribution and the existing distribution do not substantially affect the model. However, large differences can be serious for confidence and prediction intervals, and for hypothesis testing.

A simple method for checking the normality assumption on the residuals is to plot the residuals on the abscissa in ascending order, accounting for the signs, against the following quantity on the ordinate:

$$\Phi^{-1}[P(i)] \quad i = 1, 2, \dots, N \quad (5.69a)$$

where

$$P(i) = \left( i - \frac{1}{2} \right) / N$$

$$\Phi(z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^z e^{-u^2/2} du \quad (5.69b)$$

The quantity  $\Phi(z)$  gives the cumulative probability distribution of a variable  $z$  that is normally distributed with zero mean and unit variance, i.e.,  $z \sim \mathcal{N}(0, 1)$ . The symbol  $\Phi^{-1}$  represents the inverse of  $\Phi$ , so that the expression (5.69a) returns the  $z$  value corresponding to a given value of cumulative probability  $P(i)$ . The cumulative probability must lie in the interval  $[0, 1]$ .

The plot of sorted residuals versus  $\Phi^{-1}[P(i)]$  should form approximately a straight line, if the residuals are in fact normally distributed. The function in Eq. (5.69a) straightens the cumulative probability density curve for ordered normally distributed residuals from an “S” shape into a straight line, using the inverse of the ideal normal cumulative probability distribution (see Fig. B.2, Appendix B). Deviations from a normal distribution will make the plot deviate from a straight line. The degree to which the line is straight is usually determined visually. Points near the middle of the plot are more important than the points at either edge. The mean value of the residuals can be determined by picking the abscissa value corresponding to an ordinate of 0 (the mean value of the standardized  $z$  variable), and the standard error can be obtained from the difference between the abscissa values at ordinate values of 1 and 0, which is the slope of the line.

Examples of various residual plots resulting from flight data analysis will be given in Sec. 5.1.7.

### 5.1.5 Standardized Regressors

For some purposes, it is advantageous to work with scaled versions of the regressors and response variable. For example, the assessment of data collinearity discussed later in this chapter uses scaled regressors. There are several scaling techniques; however, only the one known as unit length scaling will be introduced here. The equations for the transformed regressors and response variable are

$$\xi_j^*(i) = \frac{\xi_j(i) - \bar{\xi}_j}{\sqrt{S_{jj}}} \quad i = 1, 2, \dots, N \\ j = 1, 2, \dots, n \quad (5.70)$$

$$z^*(i) = \frac{z(i) - \bar{z}}{\sqrt{S_{zz}}} \quad i = 1, 2, \dots, N \quad (5.71)$$

where

$$S_{jj} = \sum_{i=1}^N [\xi_j(i) - \bar{\xi}_j]^2 \quad (5.72)$$

is the centered sum of squares for the regressor  $\xi_j$ , and

$$S_{zz} = \sum_{i=1}^N [z(i) - \bar{z}]^2 \quad (5.73)$$

is the centered sum of squares for the response  $z$ . With this scaling, each new regressor  $\xi_j^*$  has mean value

$$\bar{\xi}_j^* = 0 \quad j = 1, 2, \dots, n$$

and length

$$||\xi_j^*|| = \sqrt{\sum_{i=1}^N [\xi_j^*(i)]^2} = 1 \quad j = 1, 2, \dots, n$$

The regression model with scaled regressors takes the form

$$z^*(i) = \theta_1^* \xi_1^*(i) + \theta_2^* \xi_2^*(i) + \dots + \theta_n^* \xi_n^*(i) + \nu(i), \quad i = 1, 2, \dots, N \quad (5.74)$$

where the constant bias term is omitted because the regressors and the response variable are now centered, with their mean values removed. Using vector and matrix notation as before, the least-squares estimator is

$$\hat{\boldsymbol{\theta}}^* = (\mathbf{X}^{*T} \mathbf{X}^*)^{-1} \mathbf{X}^{*T} z^* \quad (5.75a)$$

with covariance matrix

$$\text{Cov}(\hat{\boldsymbol{\theta}}^*) = (\sigma^*)^2 (\mathbf{X}^{*T} \mathbf{X}^*)^{-1} \quad (5.75b)$$

where  $\mathbf{X}^* = [\xi_1^* \quad \xi_2^* \quad \dots \quad \xi_n^*]$  and  $(\sigma^*)^2$  is the error variance for  $z^*$ . The  $\mathbf{X}^{*T} \mathbf{X}^*$  matrix then takes the form of a correlation matrix,

$$\mathbf{X}^{*T} \mathbf{X}^* = \begin{bmatrix} 1 & r_{12} & r_{13} & \dots & r_{1n} \\ r_{21} & 1 & r_{23} & \dots & r_{2n} \\ \vdots & & & & \vdots \\ r_{n1} & r_{n2} & r_{n3} & \dots & 1 \end{bmatrix} \quad (5.76)$$

where

$$r_{jk} = \frac{\sum_{i=1}^N [\xi_j^*(i) - \bar{\xi}_j][\xi_k^*(i) - \bar{\xi}_k]}{\sqrt{S_{jj} S_{kk}}} \quad j, k = 1, 2, \dots, n \quad (5.77)$$

is the simple pair-wise correlation between  $\xi_j$  and  $\xi_k$  [cf. Eq. (5.16a)]. Note that the correlation expression (5.16a) is for the regressors in their original form, whereas Eq. (5.76) is an analogous expression for the centered and scaled versions of the regressors. Similarly,

$$\mathbf{X}^{*T} z = [r_{1z} \quad r_{2z} \quad \dots \quad r_{nz}]^T \quad (5.78)$$

where

$$r_{jz} = \frac{\sum_{i=1}^N [\xi_j^*(i) - \bar{\xi}_j][z(i) - \bar{z}]}{\sqrt{S_{jj} S_{zz}}} \quad j = 1, 2, \dots, n \quad (5.79)$$

is the simple correlation between  $\xi_j^*$  and  $z$ . The parameters in the vector  $\boldsymbol{\theta}^*$  are usually called standardized parameters. They are dimensionless and are related to the original parameters by

$$\theta_j = \theta_j^* \sqrt{\frac{S_{zz}}{S_{jj}}} \quad j = 1, 2, \dots, n \quad (5.80)$$

$$\hat{\theta}_0 = \bar{z} - \sum_{j=1}^n \hat{\theta}_j^* \bar{\xi}_j \quad (5.81)$$

### 5.1.6 Multivariate Orthogonal Functions

Typically, once the experimental data are collected, polynomials in the independent variables are used to model the functional dependence of the response variables on the independent variables. Model parameters are estimated from the measured data using ordinary least-squares linear regression, as described earlier. As noted in Chapter 3, this modeling corresponds to a local multivariate Taylor series expansion in the independent variables. One difficulty that can arise with this approach is that polynomial terms can be highly correlated, because of similar values in restricted ranges. For example, linear and cubic functions near zero have a similar form. In addition, it often happens in aircraft flight testing that some independent variables are highly correlated because of how the aircraft normally flies, or because of high gain feedback control. For example, the pitch rate and time derivative of the angle of attack are often quite similar when the dominant aircraft response is longitudinal short-period motion. For aircraft with active pitch rate feedback to stabilize the aircraft in pitch, or to improve the short-period damping, movements of the longitudinal pitch control surface, typically the elevator, are highly correlated with pitch rate.

Correlation among regressors in a proposed model structure causes problems in parameter estimation, simply because it is difficult for the parameter estimation algorithm to accurately assign parameter values for the regressors in the model when some of the regressors are similar. Stated another way, the parameter estimator cannot determine which model term should be used to model a particular variation in the response variable if there are several model terms that could fill the same role nearly as well. This problem manifests itself mathematically in poor conditioning of the  $X^T X$  matrix, which affects the parameter estimates and error bounds through inaccuracies in the matrix inversion [cf. Eqs. (5.10) and (5.12)].

There are methods that can be used successfully in practice to address this problem. Some of these will be discussed later in Sec. 5.5 on data collinearity. This section describes a method for solving the problem by generating multivariate orthogonal modeling functions directly from the measured data for the independent variables. The orthogonalization is equivalent to isolating the unique variations in each of the original model regressors, and separately assigning that unique variation to individual orthogonal functions. This process results in a complete decorrelation of the modeling functions. The orthogonal functions

are generated in a manner that allows each of them to be decomposed without ambiguity into an expansion of ordinary multivariate polynomials in the independent variables. Therefore, a model identified using multivariate orthogonal functions can be subsequently converted to a multivariate ordinary polynomial expansion in the independent variables. This latter form of the model provides physical insight into the functional dependencies.

*Generating multivariate orthogonal functions.* Ref. 7 describes a procedure for using measured data for the independent variables to generate multivariate orthogonal modeling functions  $\mathbf{p}_j, j = 0, 1, 2, \dots, n$ , which have the following important orthogonality property:

$$\mathbf{p}_i^T \mathbf{p}_j = 0 \quad i \neq j, \quad i, j = 0, 1, 2, \dots, n \quad (5.82)$$

It is also possible to generate multivariate orthogonal functions by first generating ordinary multivariate polynomials in the independent variables, then orthogonalizing these functions using a Gram-Schmidt orthogonalization procedure. This approach is described in Ref. 8, which is the basis for the material presented here.

The process begins by choosing one of the ordinary multivariate polynomial regressors as the first orthogonal function. Typically, the vector of ones associated with the bias term in the model is chosen as the first orthogonal function,

$$\mathbf{p}_0 = \mathbf{1} \quad (5.83)$$

In general, any of the regressors can be chosen as the first orthogonal function, without any change in the procedure. To generate the next orthogonal function, an ordinary multivariate polynomial function is made orthogonal to the preceding orthogonal function(s). Define the  $j$ th orthogonal function  $\mathbf{p}_j$  as

$$\mathbf{p}_j = \xi_j - \sum_{k=0}^{j-1} \gamma_{kj} \mathbf{p}_k \quad j = 1, 2, \dots, n \quad (5.84)$$

where  $\xi_j$  is the  $j$ th ordinary multivariate regressor vector. The  $\gamma_{kj}$  for  $k = 0, 1, \dots, j-1$  are scalars determined by multiplying both sides of Eq. (5.84) by  $\mathbf{p}_k^T$ , invoking the mutual orthogonality of the  $\mathbf{p}_k$ ,  $k = 0, 1, \dots, j$ , and solving for  $\gamma_{kj}$ ,

$$\gamma_{kj} = \frac{\mathbf{p}_k^T \xi_j}{\mathbf{p}_k^T \mathbf{p}_k} \quad k = 0, 1, \dots, j-1 \quad j = 1, 2, \dots, n \quad (5.85)$$

The same process can be implemented in sequence for each ordinary multivariate regressor  $\xi_j, j = 1, 2, \dots, n$ . The total number of ordinary multivariate regressors used as raw material for generating the multivariate orthogonal functions, including the bias term, is  $n + 1$ . It can be seen from Eqs. (5.83–5.85) that

## REGRESSION METHODS

117

each orthogonal function can be expressed in terms of a linear expansion of the original regressors. The orthogonal functions are generated sequentially by orthogonalizing the original regressors with respect to the orthogonal functions already computed, so that each orthogonal function can be considered an orthogonalized version of an original regressor.

Typically, the original regressors will be ordinary multivariate polynomials in the independent variables, but this is not a requirement. The original regressors can in fact be arbitrary functions of the independent variables, such as splines or arbitrary nonlinear functions, including discontinuous functions.

The orthogonalization process described earlier can be repeated to generate orthogonal functions for multivariate polynomials of arbitrary order in the independent variables, subject only to limitations related to the information contained in the data. For example, it is not possible to generate an orthogonal function corresponding to  $\alpha^2$  if there are only two distinct values of angle of attack in the measured data. This is analogous to the requirement that at least three data points are needed to identify a quadratic model, which has three parameters. The same limit also applies to the orthogonal function corresponding to any cross term, such as  $\alpha^2 \delta_e$  for this example.

If the  $p_j$  vectors and the  $\xi_j$  vectors are arranged as columns of matrices  $P$  and  $X$ , respectively, and the  $\gamma_{kj}$  are elements in the  $(k+1)$ th row and  $(j+1)$ th column of an upper triangular matrix  $G$  with ones on the diagonal,

$$G = \begin{bmatrix} 1 & \gamma_{01} & \gamma_{02} & \cdots & \gamma_{0n} \\ 0 & 1 & \gamma_{12} & \cdots & \gamma_{1n} \\ 0 & 0 & 1 & \cdots & \gamma_{2n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix} \quad (5.86)$$

Then

$$X = PG \quad (5.87)$$

which leads to

$$P = XG^{-1} \quad (5.88)$$

The columns of  $G^{-1}$  contain the coefficients for expansion of each column of  $P$  (i.e., each multivariate orthogonal function) in terms of the original regressors contained in the columns of  $X$ . Equation (5.88) can be used to express each multivariate orthogonal function in terms of the original regressors. The manner in which the orthogonal functions are generated allows them to be decomposed without ambiguity into an expansion of the original regressors, which have physical meaning.

For all flight-test data sets and many wind-tunnel data sets, the measured values of the independent are not uniformly spaced over an interval. This makes it impossible to use standard orthogonal polynomials described in many

textbooks. The method described here is simple, and works regardless of the spacing of the independent variable measurements.

Because of the sequential nature of the method used to generate orthogonal functions from the original regressors, the particular orthogonal functions generated depend on the order in which the original regressors are used in the orthogonalization. This is most apparent for the case of two original regressors that are highly correlated. The orthogonal function generated for the first of these original regressors will have essentially all of the unique character that can be used for modeling the dependent variable. The orthogonal function for the second original regressor will then be close to the zero vector, because the previously generated orthogonal function will be able to account for nearly all of the useful content of the second original regressor. In effect, nearly all of the useful content of the second correlated regressor will be removed during the orthogonalization with Eqs. (5.84) and (5.85).

The order in which the orthogonal functions are generated also affects the number of original regressors required to represent each orthogonal function using Eq. (5.88). In general, the later in the order that a particular orthogonal function is generated, the more ordinary regressors that will be required for its expansion. To achieve the most compact final model in terms of the original regressors, the best approach is to present the most important original regressors first in line for orthogonalization.

*Ordinary least squares using multivariate orthogonal functions.* Once the ordinary multivariate polynomial regressors postulated for the linear regression model have been orthogonalized, Eq. (5.6) can be rewritten as

$$\mathbf{z} = \mathbf{P}\mathbf{a} + \boldsymbol{\nu} \quad (5.89)$$

where

$$\mathbf{z} = [z(1) \ z(2) \ \cdots \ z(N)]^T = N \times 1 \text{ vector}$$

$$\mathbf{a} = [a_0 \ a_1 \ \cdots \ a_n]^T = n_p \times 1 \text{ vector of unknown parameters, } n_p = n + 1$$

$$\mathbf{P} = [\mathbf{p}_0 \ \mathbf{p}_1 \ \cdots \ \mathbf{p}_n] = N \times n_p \text{ matrix of mutually orthogonal regressors}$$

$$\boldsymbol{\nu} = [\nu(1) \ \nu(2) \ \cdots \ \nu(N)]^T = N \times 1 \text{ vector of measurement errors}$$

The  $a_j, j = 0, 1, 2, \dots, n$  are constant model parameters to be determined. These parameters are different, in general, from  $\boldsymbol{\theta} = [\theta_0 \ \theta_1 \ \cdots \ \theta_n]^T$ , because the modeling functions are different. The parameter vector estimate that minimizes the ordinary least-squares cost function

$$J(\mathbf{a}) = \frac{1}{2}(\mathbf{z} - \mathbf{P}\mathbf{a})^T(\mathbf{z} - \mathbf{P}\mathbf{a}) \quad (5.90)$$

is computed using Eq. (5.10),

$$\hat{\mathbf{a}} = (\mathbf{P}^T \mathbf{P})^{-1} \mathbf{P}^T \mathbf{z} \quad (5.91)$$

## REGRESSION METHODS

119

The estimated parameter covariance matrix is computed by Eq. (5.12),

$$\text{Cov}(\hat{\mathbf{a}}) = E[(\hat{\mathbf{a}} - \mathbf{a})(\hat{\mathbf{a}} - \mathbf{a})^T] = \sigma^2 (\mathbf{P}^T \mathbf{P})^{-1} \quad (5.92)$$

and the error variance  $\sigma^2$  can be estimated from the residuals,

$$\mathbf{v} = \mathbf{z} - \mathbf{P}\hat{\mathbf{a}} \quad (5.93)$$

$$\hat{\sigma}^2 = \frac{1}{(N - n_p)} [(\mathbf{z} - \mathbf{P}\hat{\mathbf{a}})^T (\mathbf{z} - \mathbf{P}\hat{\mathbf{a}})] = \frac{\mathbf{v}^T \mathbf{v}}{(N - n_p)} \quad (5.94)$$

or  $\sigma^2$  can be estimated independently, as discussed earlier. Parameter standard errors are computed as the square root of the diagonal elements of the  $\text{Cov}(\hat{\mathbf{a}})$  matrix from Eq. (5.92).

All of this looks the same as before, until the fact that the modeling functions are orthogonal is introduced. When the modeling functions are mutually orthogonal,  $\mathbf{P}^T \mathbf{P}$  is a diagonal matrix with the inner product of the orthogonal functions on the main diagonal. This decouples the normal equations, so that Eq. (5.91) becomes

$$\hat{a}_j = (\mathbf{p}_j^T \mathbf{z}) / (\mathbf{p}_j^T \mathbf{p}_j) \quad j = 0, 1, 2, \dots, n \quad (5.95)$$

Using Eqs. (5.82) and (5.95) in Eq. (5.90),

$$J(\hat{\mathbf{a}}) = \frac{1}{2} \left[ \mathbf{z}^T \mathbf{z} - \sum_{j=0}^n \hat{a}_j^2 (\mathbf{p}_j^T \mathbf{p}_j) \right] \quad (5.96)$$

or, using Eq. (5.95),

$$J(\hat{\mathbf{a}}) = \frac{1}{2} \left[ \mathbf{z}^T \mathbf{z} - \sum_{j=0}^n (\mathbf{p}_j^T \mathbf{z})^2 / (\mathbf{p}_j^T \mathbf{p}_j) \right] \quad (5.97)$$

Equation (5.95) shows that the estimated parameter  $\hat{a}_j$  associated with orthogonal function  $\mathbf{p}_j$  depends only on  $\mathbf{p}_j$  and  $\mathbf{z}$ , so that  $\hat{a}_j$  can be computed independently of the other orthogonal modeling functions. The least-squares modeling problem is therefore decoupled, which solves the problems associated with correlated regressors. From Eq. (5.97), the reduction in the cost resulting from including the term  $a_j \mathbf{p}_j$  in the model depends only on the dependent variable data  $\mathbf{z}$  and the added orthogonal modeling function  $\mathbf{p}_j$ . This makes it possible to evaluate each orthogonal modeling function in terms of its ability to reduce the least-squares model fit to the data, regardless of which other orthogonal modeling functions are already present in the model. When the modeling functions are instead polynomials in the independent variables (or any other nonorthogonal function set), the least-squares problem is coupled, and iterative analysis is required to find the subset of modeling functions for an adequate model structure.

Typically, a statistical modeling metric such as the predicted squared error (PSE) is used to select which orthogonal functions should be included in the model. More on this metric can be found in Sec. 5.4 on model structure determination.

After the model structure is determined using the multivariate orthogonal modeling functions, the estimated model output is

$$\hat{y} = \mathbf{P}\hat{a} \quad (5.98)$$

where the  $\mathbf{P}$  matrix now includes only the orthogonal functions selected in the model structure determination. Then, each retained orthogonal modeling function can be decomposed without error into an expansion of original regressors in the independent variables, using the columns of  $\mathbf{G}^{-1}$  in Eq. (5.88) corresponding to the retained orthogonal functions. Combining like terms from this final step puts the final model in the form of a multivariate Taylor series expansion when the original regressors are ordinary multivariate polynomials. Aircraft dynamics and control analyses are often conducted with the assumption of this form for the dependence of the nondimensional aerodynamic force and moment coefficients on independent variables such as angle of attack and sideslip angle. This final form of the model also allows straightforward analytic differentiation for partial derivatives of the response variable with respect to the independent variables, which is useful for linearization and dynamic analysis.

### 5.1.7 Application to Aircraft

In applying linear regression to aircraft parameter estimation, the regressors  $\xi_j, j = 1, 2, \dots, n$ , in Eq. (5.4) or (5.6) are computed from direct measurements of the aircraft state and control variables. As noted earlier, often at least some of the regressors are simply the aircraft states and controls themselves. Those regressors are linear functions of the independent variables. Other regressors are postulated based on experience or are selected from a pool of candidate regressors using statistical model structure determination techniques, which will be described later.

The vector of ones for the bias term in the model is always included as a regressor. If the modeling is to be referenced to values of the independent variables that are different from zero, then the independent variables should have their constant part removed before calculating the regressors. There are two reasons for this. First, the bias term in the model is already being used to account for the constant part of the dependent variable, so any other regressor with a constant part will be correlated with the bias term. Regressor correlation causes problems with the parameter estimation, as described earlier. Second, removing the constant part from the independent variables will automatically remove the bias in any regressors that are multivariate polynomial regressors. These regressors are commonly used to approximate the dependent variable locally via a multivariable Taylor series expansion [cf. Eq. (3.73)]. If the biases are not removed from the measured independent variables, this is equivalent to assuming that the associated multivariate Taylor series expansion is being

## REGRESSION METHODS

121

made about a reference value of zero for every independent variable. The reference values for any multivariate Taylor series expansion must be carefully noted, because the parameter values depend on what these reference values are, and the reference values are important when comparing results with information from other sources.

For aerodynamic parameter estimation, nondimensional aerodynamic force and moment coefficients are used as the dependent variable in linear regressions. A separate linear regression problem is solved for each force or moment coefficient, corresponding to minimizing the equation error in each individual equation of motion for the six degrees of freedom of the aircraft. As explained earlier, values for the aerodynamic force and moment coefficients cannot be measured directly in flight, but instead must be computed from other measurements and the equations of motion. The necessary equations follow from Eqs. (3.54), (3.40), and (3.37):

$$C_X \equiv -C_A = \frac{1}{\bar{q}S}(ma_x - T) \quad (5.99a)$$

$$C_Y = \frac{ma_y}{\bar{q}S} \quad (5.99b)$$

$$C_Z = -C_N = \frac{ma_z}{\bar{q}S} \quad (5.99c)$$

$$C_L = -C_Z \cos \alpha + C_X \sin \alpha \quad (5.99d)$$

$$C_D = -C_X \cos \alpha - C_Z \sin \alpha \quad (5.99e)$$

$$C_l = \frac{1}{\bar{q}Sb} [I_x \dot{p} - I_{xz}(pq + \dot{r}) + (I_z - I_y)qr] \quad (5.100a)$$

$$C_m = \frac{1}{\bar{q}Sc} [I_y \dot{q} + (I_x - I_z)pr + I_{xz}(p^2 - r^2) - I_p \Omega_p r] \quad (5.100b)$$

$$C_n = \frac{1}{\bar{q}Sb} [I_z \dot{r} - I_{xz}(\dot{p} - qr) + (I_y - I_x)pq + I_p \Omega_p q] \quad (5.100c)$$

Equations (5.99) and (5.100) show that the quantities required to calculate the nondimensional force and moment coefficients are translational and angular accelerations, angular velocities, thrust, mass/inertia properties, dynamic pressure, and reference geometry. Note that errors in any of these measured quantities show up as errors in the dependent variables of the regression problems. This makes it important to have good instrumentation with small systematic and random errors when using the linear regression approach. Data compatibility analysis described in Chapter 10 can be used to obtain a consistent data set with small systematic instrumentation errors.

Reference geometry and mass properties appear in Eqs. (5.99) and (5.100), so these must be determined accurately as well. Relevant information is included in Appendix C.

Thrust is usually not measured directly in flight. Often the thrust is obtained by interpolating ground test data or using a model identified from ground test data. Alternatively, it is possible to identify stability and control derivatives that quantify the combined effect of aerodynamics and thrust. In this case, the subtraction of the thrust term in Eq. (5.99a) is omitted, and the identified model characterizes the combined effect of aerodynamics and thrust. This approach might be used when the interest is more in accurate simulation as opposed to accurately characterizing the aerodynamics. Obviously, the same method can be used for other force and moment coefficients when the thrust is not directed along the  $x$  body axis and through the c.g. However, if the effects of aerodynamics and thrust are to be accurately separated, a good measurement or estimate of the thrust magnitude and direction is required.

This issue is not to be confused with thrust-induced aerodynamic effects, where local airflow over the aircraft is affected by the operation of the engine, resulting in changes in the apparent aerodynamic characteristics of the aircraft. The aerodynamic modeling then typically includes propulsion variables such as throttle position or power level, but the same principles discussed earlier regarding separating aerodynamic and thrust effects apply.

For the lift coefficient  $C_L$  and drag coefficient  $C_D$ , the measured angle of attack is also required [cf. Eqs. (5.99d) and (5.99e)]. These coefficients might be used instead of  $C_X$  and  $C_Z$  when there is a desire to compare parameter estimation results with wind-tunnel values, for example. However, each calculation for  $C_L$  and  $C_D$  involves both translational accelerations  $a_x$  and  $a_z$ , as well as measured angle of attack and thrust. Using all these measurements to compute  $C_L$  and  $C_D$  can lead to multiple error sources and significantly increased noise levels. Because of this,  $C_X$  (or axial force coefficient  $C_A = -C_X$ ) and  $C_Z$  (or normal force coefficient  $C_N = -C_Z$ ) are usually preferred for aerodynamic parameter estimation, although this is not a universal practice.

In Chapter 3, it was seen that the applied forces and moments appearing in the equations of motion act at the aircraft c.g. However, it is sometimes necessary to compute nondimensional aerodynamic coefficients at a location different from the aircraft c.g. This happens when comparing flight results with wind-tunnel data, where the forces and moments were measured in the wind tunnel at a reference location different from the c.g. of the aircraft in flight. The force coefficients are unaltered for a change in reference point; the moment coefficients at the reference point can be computed from

$$\begin{bmatrix} C_l \\ C_m \\ C_n \end{bmatrix}_{\text{ref}} = \begin{bmatrix} C_l \\ C_m \\ C_n \end{bmatrix}_{\text{c.g.}} + \begin{bmatrix} 1/b & 0 & 0 \\ 0 & 1/\bar{c} & 0 \\ 0 & 0 & 1/b \end{bmatrix} \left\{ \begin{bmatrix} (x_{\text{cg}} - x_{\text{ref}}) \\ (y_{\text{cg}} - y_{\text{ref}}) \\ (z_{\text{cg}} - z_{\text{ref}}) \end{bmatrix} \times \begin{bmatrix} C_X \\ C_Y \\ C_Z \end{bmatrix} \right\} \quad (5.101)$$

## REGRESSION METHODS

123

where  $[x_{cg} \ y_{cg} \ z_{cg}]$  and  $[x_{ref} \ y_{ref} \ z_{ref}]$  are the coordinates of the aircraft c.g. and the reference point, respectively. If this conversion is done before the modeling begins, then the estimated aerodynamic model parameters will be associated with the reference point, rather than with the aircraft c.g., which can simplify comparisons with wind-tunnel data.

Finally, the angular accelerations  $\dot{p}$ ,  $\dot{q}$ , and  $\dot{r}$  are usually not measured directly. Instead, they are obtained by a smoothed numerical differentiation of the angular rates. Effective algorithms for obtaining accurate smoothed derivatives of measured data are presented in Chapter 11.

When applying linear regression using flight-test data, the regressors are assembled from measured data, which are noisy. This violates the assumption made in the linear regression analysis that the regressors are deterministic. The result is that the estimated parameters are biased and inefficient, as discussed in Refs. 2 and 3. The extent to which this occurs increases with increasing noise levels on the measurements used to assemble the regressors.

Linear regression can also be applied to the linearized state-space aircraft equations of motion, such as Eqs. (3.126a), (3.126b), and (3.130a–3.130c). In this case, the state derivative terms on the left sides of the equations are considered the dependent variable, and the perturbation states and controls are the regressors. The estimated parameters are the dimensional stability and control derivatives. A similar approach can be used with the linearized output equations (3.126c) and (3.130f).

This technique can also be used with transfer function models and measured data transformed into the frequency domain (see Chapter 7). In both the state-space and transfer function models, the dimensional model parameters combine the nondimensional aerodynamic stability and control derivatives with dynamic pressure, aircraft reference geometry, and mass/inertia properties [cf. Eqs. (3.127) and (3.131)]. Consequently, the dimensional parameters can vary throughout the maneuver as the dynamic pressure and mass/inertia properties change. This introduces some inaccuracy in the estimates of these parameters, because the parameter estimation algorithms assume that the model parameters are constants throughout the maneuver. The problem is avoided by using nondimensional aerodynamic coefficients as the dependent variable, as described earlier.

### Example 5.1

In this example, linear regression is applied to aircraft flight-test data to estimate nondimensional stability and control derivatives. The test aircraft was the NASA Twin Otter aircraft, which is a twin-engine turboprop commuter aircraft, shown in Fig. 5.3.

Flight-test data were collected for two lateral maneuvers initiated from the same steady trim condition, using rudder and aileron deflections. The flight control system was unaugmented, so the pilot commands were directly implemented at the control surfaces through the control linkage. Measured flight data from run 1 were intended for aerodynamic parameter estimation; data from run 2 were for model validation. The basic aircraft characteristics



**Fig. 5.3 NASA Twin Otter aircraft.**

and flight condition are specified as follows:

$$\begin{array}{lll} \bar{c} = 6.5 \text{ ft} & I_x = 20,900 \text{ slug}\cdot\text{ft}^2 & V_o = 238 \text{ ft/s} \\ b = 65 \text{ ft} & I_y = 24,261 \text{ slug}\cdot\text{ft}^2 & \bar{q}_o = 56.6 \text{ psf} \\ S = 422.5 \text{ ft}^2 & I_z = 38,469 \text{ slug}\cdot\text{ft}^2 & g = 32.17 \text{ ft/s}^2 \\ m = 340 \text{ slugs} & I_{xz} = 1,128 \text{ slug}\cdot\text{ft}^2 & \end{array}$$

The input and output variables were sampled at intervals of 0.02 s, corresponding to a 50-Hz sampling rate. Figure 5.4 shows measured data for run 1, which is a lateral maneuver implemented by a series of rudder pulses, followed by an aileron doublet.

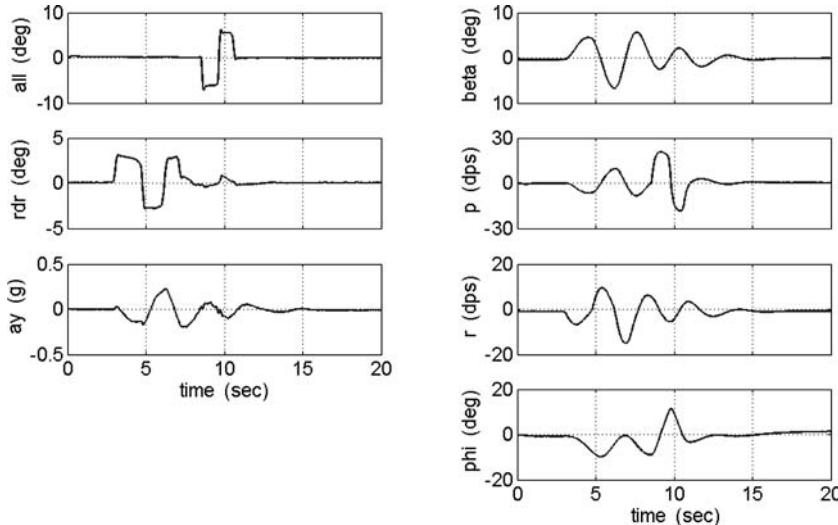
The regression equations for lateral aerodynamic force and moment coefficients were

$$C_Y(i) = C_{Yo} + C_{Y\beta}\beta(i) + C_{Yr} \frac{b}{2V_o} r(i) + C_{Y\delta_r} \delta_r(i) + v_Y(i) \quad (5.102a)$$

$$\begin{aligned} C_l(i) = C_{lo} + C_{l\beta}\beta(i) + C_{lp} \frac{b}{2V_o} p(i) + C_{lr} \frac{b}{2V_o} r(i) \\ + C_{l\delta_a}(i) + C_{l\delta_r} \delta_r(i) + v_l(i) \end{aligned} \quad (5.102b)$$

## REGRESSION METHODS

125



**Fig. 5.4 Measured input and output variables for lateral maneuver, run 1.**

$$\begin{aligned}
 C_n(i) = & C_{no} + C_{n\beta} \beta(i) + C_{np} \frac{b}{2V_o} p(i) + C_{nr} \frac{b}{2V_o} r(i) \\
 & + C_{n_{\delta_a}} \delta_a(i) + C_{n_{\delta_r}} \delta_r(i) + v_n(i)
 \end{aligned} \tag{5.102c}$$

for  $i = 1, 2, \dots, N$ . The error terms are assumed to be zero mean with constant variance, i.e.,  $E[v_n(i)] = 0$  and  $\text{Var}[v_Y(i)] = E[v_Y^2(i)] = \sigma_Y^2$ , etc. The dependent variable values on the left sides of the preceding equations were computed from Eqs. (5.99b), (5.100a), and (5.100c), respectively. The angular accelerations  $\dot{p}$  and  $\dot{r}$  in Eqs. (5.100a) and (5.100c) were obtained by smoothed local numerical differentiation of the measured angular velocities  $p$  and  $r$ , as described in Chapter 11.

The least-squares estimate of the aerodynamic parameters in the preceding equations is given by Eq. (5.10),

$$\hat{\boldsymbol{\theta}} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{z}$$

For the yawing moment coefficient  $C_n$ ,

$$\begin{aligned}
 \boldsymbol{\theta} &= [C_{no} \ C_{n\beta} \ C_{np} \ C_{nr} \ C_{n_{\delta_a}} \ C_{n_{\delta_r}}]^T \\
 \mathbf{z} &= [C_n(1) \ C_n(2) \ \dots \ C_n(N)]^T
 \end{aligned}$$

**Table 5.1 Least-squares parameter estimation results, aerodynamic yawing moment coefficient, run 1**

Parameter	$\hat{\theta}$	$s(\hat{\theta})$	$ t_0 $	$100[s(\hat{\theta})/ \hat{\theta} ]$
$C_{n\beta}$	$8.54 \times 10^{-2}$	$3.58 \times 10^{-4}$	238.9	0.4
$C_{np}$	$-5.15 \times 10^{-2}$	$1.43 \times 10^{-3}$	35.9	2.8
$C_{nr}$	$-1.98 \times 10^{-1}$	$1.30 \times 10^{-3}$	151.8	0.7
$C_{n\delta_a}$	$2.34 \times 10^{-3}$	$5.00 \times 10^{-4}$	4.7	21.4
$C_{n\delta_r}$	$-1.31 \times 10^{-1}$	$5.97 \times 10^{-4}$	218.5	0.5
$C_{n_o}$	$-4.60 \times 10^{-4}$	$7.42 \times 10^{-6}$	62.0	1.6
$s = \hat{\sigma}$	$2.25 \times 10^{-4}$	—	—	—
$R^2, \%$	99.6	—	—	—

$$\mathbf{X} = \begin{bmatrix} 1 & \beta(1) & \frac{b}{2V_o} p(1) & \frac{b}{2V_o} r(1) & \delta_a(1) & \delta_r(1) \\ 1 & \beta(1) & \frac{b}{2V_o} p(2) & \frac{b}{2V_o} r(2) & \delta_a(2) & \delta_r(2) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \beta(N) & \frac{b}{2V_o} p(N) & \frac{b}{2V_o} r(N) & \delta_a(N) & \delta_r(N) \end{bmatrix}$$

and similarly for  $C_Y$  and  $C_l$ .

The results for the yawing moment coefficient are summarized in Table 5.1, including parameter estimates, standard errors,  $t$  statistics, fit error, and coefficient of determination. The parameter estimates were computed from Eq. (5.10). Standard errors for the parameter estimates come from the square root of the diagonal elements of the covariance matrix computed using Eq. (5.12), with the fit error estimated by Eq. (5.24). The  $t$  statistic for the addition of each single model term is computed from Eq. (5.60). The coefficient of determination comes from Eq. (5.31), with Eq. (5.26). Pair-wise correlations for the estimated parameters are obtained from Eq. (5.18), and shown in Table 5.2. Results for  $C_Y$  and  $C_l$  can be computed in the same way.

In Table 5.1, note that the  $|t_0|$  values are very high for all parameters except  $C_{n\delta_a}$ . The estimate of  $C_{n\delta_a}$  is also close to zero. This parameter quantifies the effect of aileron on the yawing moment. Airplanes are designed so that the ailerons affect primarily rolling moment, and produce as little yawing moment as possible. Because of this,  $C_{n\delta_a}$  is normally a weak parameter, i.e., a parameter with relatively small magnitude. Based on this information, it might be that the  $C_{n\delta_a} \delta_a$  term is not necessary in the model. Section 5.4 gives more detail on methods that can be used to address this issue of model structure determination, using statistical metrics based on the measured data.

## REGRESSION METHODS

127

**Table 5.2 Parameter correlation matrix, aerodynamic yawing moment coefficient, run 1**

	$C_{n\beta}$	$C_{np}$	$C_{nr}$	$C_{n\delta_a}$	$C_{n\delta_r}$	$C_{no}$
$C_{n\beta}$	1	0.82	0.15	0.67	-0.02	0.03
$C_{np}$	—	1	0.30	0.89	-0.18	-0.02
$C_{nr}$	—	—	1	0.29	0.79	-0.21
$C_{n\delta_a}$	—	—	—	1	0.16	-0.01
$C_{n\delta_r}$	—	—	—	—	1	-0.28
$C_{no}$	—	—	—	—	—	1

The last column of Table 5.1 gives the standard error as a percentage of the parameter estimate. The row associated with  $C_{n\delta_a}$  demonstrates that this metric must be interpreted carefully, because a small parameter estimate will lead to a large percentage error, even for a very small standard error. This is the case for  $C_{n\delta_a}$  in Table 5.1.

Table 5.2 shows pair-wise correlations for terms in the linear model structure of Eq. (5.102c). Although there are some moderately high correlations, none has absolute value that exceeds 0.90. The relatively high correlation between  $pb/2V$  and  $\delta_a$  may be partially responsible for the low significance of  $C_{n\delta_a}$ . Generally,  $C_{np}$  and  $C_{n\delta_a}$  are both weak parameters, meaning that only a relatively small part of the variation in the response variable ( $C_n$  in this case) can be explained by the associated model terms. When  $pb/2V$  and  $\delta_a$  have high correlation, one of the associated parameters can exhibit low significance, because the other has already accounted for the relatively small part of the variation that can be explained mostly by either.

A comparison between the measured yawing moment coefficient and the identified model is shown in Fig. 5.5. The average 95% confidence interval on the mean response  $\hat{C}_n$ , computed from Eqs. (5.40) and (5.41), is approximately  $\pm 3.0 \times 10^{-5}$ . These interval limits about the estimated output are not plotted, because the interval is so small that the associated lines are not distinguishable.

Residuals are plotted against time and  $\hat{C}_n$  in Fig. 5.6. The dashed lines in Fig. 5.6 represent the average 95% confidence interval for prediction, which was  $\pm 4.5 \times 10^{-4}$ , computed from Eqs. (5.42) and (5.43). The residuals have a random character and lie mostly within the 95% confidence bounds, which is indicative of a good model.

Figure 5.7 shows the cumulative probability plot computed using the ordered residuals and Eqs. (5.69). A straight line indicates that the residuals are normally distributed. Only small deviations from a straight line are evident.

In Fig. 5.8, autocorrelation estimates computed from Eq. (5.67b) are plotted along with the estimated 95% confidence interval, which is  $\pm 2\hat{R}_{vv}(0)/\sqrt{N}$ , from Eq. (5.68).

If the residual sequence were white, then the autocorrelation estimates for nonzero lags would lie almost completely inside the 95% confidence interval shown, and the autocorrelation function would resemble a delta function at

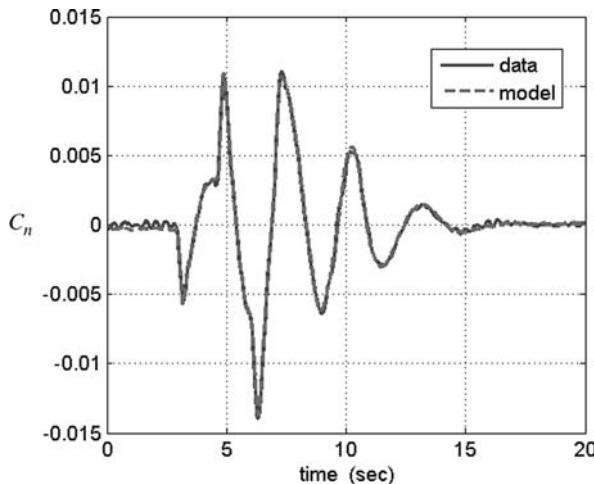


Fig. 5.5 Equation-error model fit to yawing moment coefficient for lateral maneuver, run 1.

zero lag. Since this is not true, the residual sequence is colored. This has a significant effect on the parameter standard errors, as will be shown in Example 5.2.

The statistical metrics and parameter estimation results shown indicate that the linear model was adequate to characterize the measured data and allowed

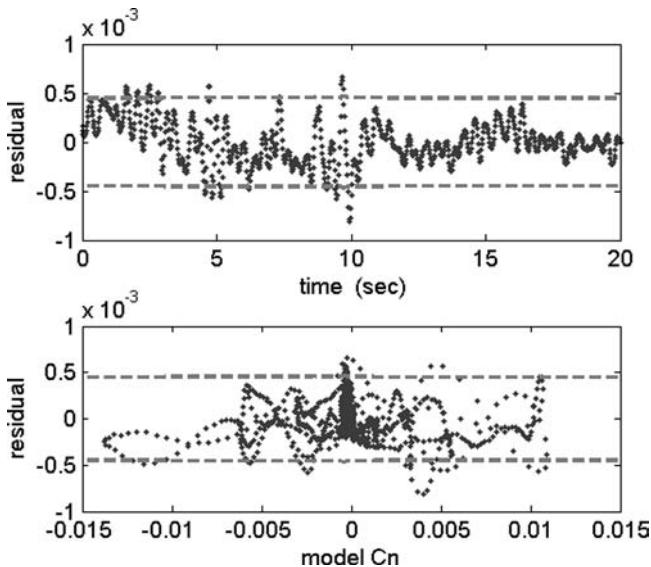


Fig. 5.6 Yawing moment coefficient residuals for lateral maneuver, run 1.

## REGRESSION METHODS

129

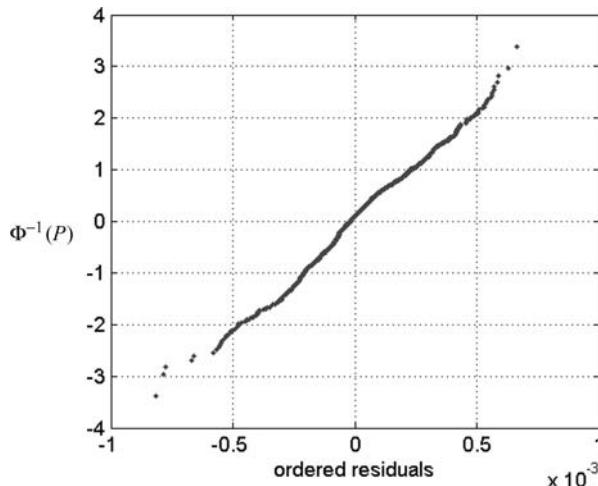


Fig. 5.7 Diagnostic plot for yawing moment coefficient residuals.

accurate calculation of the model parameters. The coefficient of determination was very high ( $R^2 = 99.6\%$ , from Table 5.1), and the model fit to the data was excellent. Residuals showed little deterministic content, with magnitudes that fell mostly within the 95% confidence interval for response prediction. All of this is indicative of a good model.

The model identified for  $C_n$  from run 1 was validated using data from run 2. Measured data for run 2 appear in Fig. 5.9. Note that the polarities of the

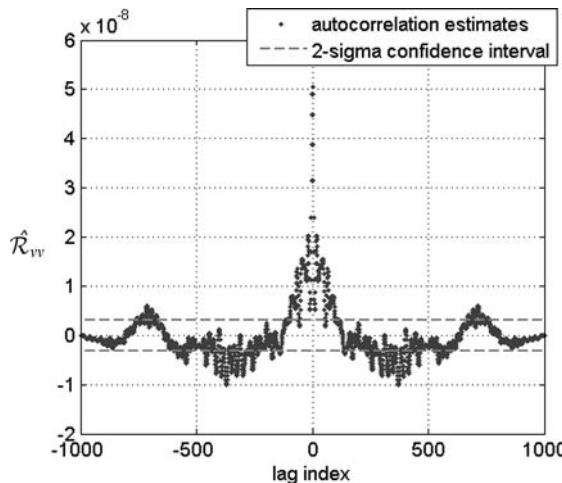


Fig. 5.8 Autocorrelation plot for yawing moment coefficient residuals for lateral maneuver, run 1.

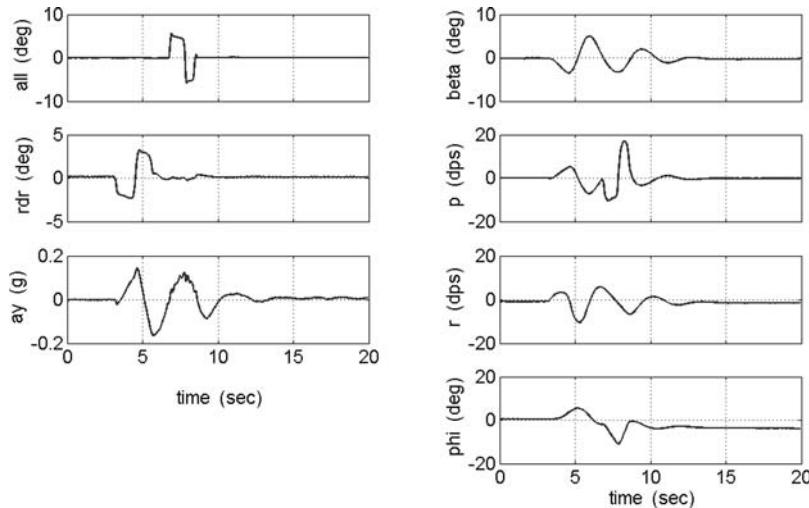


Fig. 5.9 Measured input and output variables for lateral maneuver, run 2.

initial rudder input and the aileron doublet are reversed compared with run 1. This maneuver is a simple doublet sequence at the same flight condition as specified earlier for run 1. Parameter estimates from run 1, listed in Table 5.1, were used with regressor data from run 2 to predict yawing moment coefficient for run 2.

Figure 5.10 shows the comparison of measured  $C_n$  from run 2 with the predicted values. The residuals for this prediction case are shown in Fig. 5.11, where the dashed lines mark the average 95% confidence interval for the prediction, computed using the modeling data of run 1. These are the same confidence intervals plotted in Fig. 5.6. The residuals for the prediction case are mostly within the 95% confidence bounds for the prediction, as expected.

The results shown here indicate that the linear model structure given earlier is adequate for characterizing the aerodynamic yawing moment coefficient, and that the identified model predicts well for a different maneuver at the same flight condition. Good prediction capability makes the model useful, and gives confidence that the identified model is a good characterization of the aircraft dynamics.

Chapter 11 includes a description of a global smoothing method that uses Fourier analysis to separate deterministic signal from random noise. This method can be used to isolate the random component of the dependent variable, which can then be compared to residuals from the modeling. If the model structure used in the analysis was adequate, then the deterministic part of the dependent variable would be described by the identified model, and the residuals should be the same as the random component found with the global Fourier smoother. The global Fourier smoother is a nonparametric method that does not depend on any assumed model, only on the measured data. Therefore, the global Fourier smoother produces a model-independent estimate of the random

## REGRESSION METHODS

131

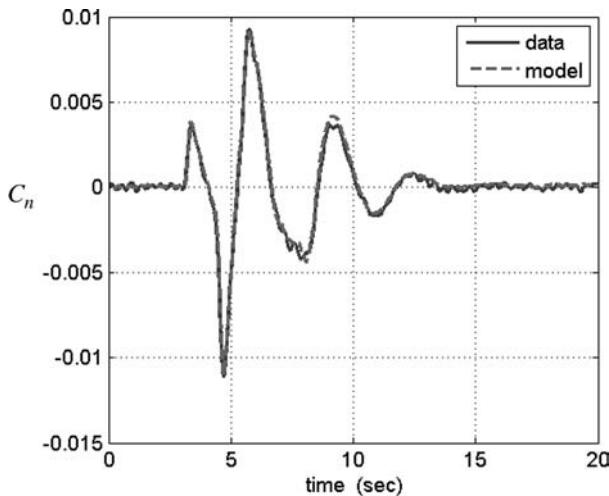


Fig. 5.10 Model prediction of yawing moment coefficient for lateral maneuver, run 2.

component of the dependent variable measurement. This random component cannot be modeled with deterministic regressors.

Figure 5.12 shows a comparison of the residuals from Fig. 5.6, shown in the upper plot, with the random component of measured  $C_n$  obtained from the global

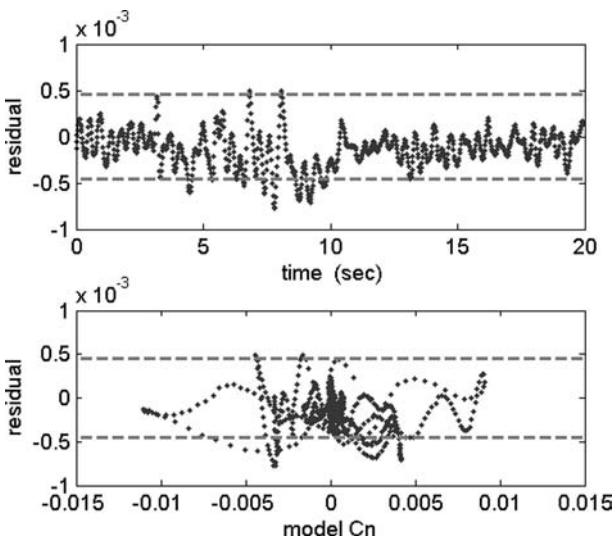
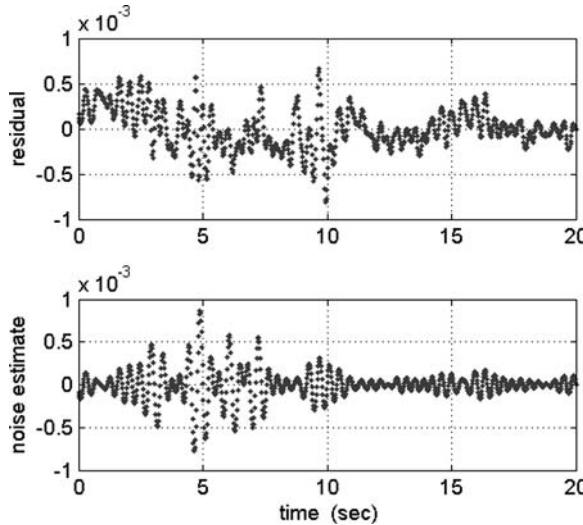


Fig. 5.11 Yawing moment coefficient prediction residuals for lateral maneuver, run 2.



**Fig. 5.12 Yawing moment coefficient residuals for lateral maneuver, run 1.**

Fourier smoother, shown in the lower plot. The magnitudes are similar. It follows that any unmodeled components in the residuals of the upper plot that are not also in the lower plot must have amplitudes within the noise level, and cannot be extracted because the signal-to-noise ratio is approximately 1 or less. This general approach can be used to assess adequacy of any assumed model structure.  $\square$

## 5.2 Generalized Least Squares (GLS)

In the previous development for ordinary least-squares linear regression, it was assumed that the measurement errors had zero mean, and were uncorrelated with equal variance. In many practical cases, the assumptions of uncorrelated measurement errors and homogeneous variances are not valid. Thus, it is necessary to make the least-squares model more general by assuming that

$$z = X\theta + \nu \quad (5.103a)$$

$$E(\nu) = \mathbf{0} \quad \text{Cov}(\nu) = E(\nu\nu^T) \equiv V \quad (5.103b)$$

The  $N \times N$  noise covariance matrix  $V$  is nonsingular and positive definite. The least-squares estimator for this modified model is obtained by minimizing

$$J_{\text{GLS}}(\theta) = \frac{1}{2}(z - X\theta)^T V^{-1} (z - X\theta) \quad (5.104)$$

## REGRESSION METHODS

133

The minimizing parameter vector is computed from

$$\hat{\boldsymbol{\theta}}_{\text{GLS}} = (\mathbf{X}^T \mathbf{V}^{-1} \mathbf{X})^{-1} \mathbf{X}^T \mathbf{V}^{-1} \mathbf{z} \quad (5.105)$$

which is called the generalized least-squares estimator. Using calculations similar to those shown earlier for ordinary least squares, it can be shown that  $\hat{\boldsymbol{\theta}}_{\text{GLS}}$  is asymptotically unbiased,  $E(\hat{\boldsymbol{\theta}}_{\text{GLS}}) = \boldsymbol{\theta}$ , with covariance matrix given by

$$\text{Cov}(\hat{\boldsymbol{\theta}}_{\text{GLS}}) = E[(\hat{\boldsymbol{\theta}}_{\text{GLS}} - \boldsymbol{\theta})(\hat{\boldsymbol{\theta}}_{\text{GLS}} - \boldsymbol{\theta})^T] = (\mathbf{X}^T \mathbf{V}^{-1} \mathbf{X})^{-1} \quad (5.106)$$

Furthermore, under the assumptions (5.103), it can be shown that  $\hat{\boldsymbol{\theta}}_{\text{GLS}}$  is the best linear estimator of  $\boldsymbol{\theta}$  (see Ref. 3).

If the measurement errors for the dependent variable are uncorrelated, but with different variances, then  $\mathbf{V}$  is a diagonal matrix with unequal elements on the diagonal. Introducing  $\mathbf{W} = \mathbf{V}^{-1}$ , the elements of  $\mathbf{W}$  are weights for each equation in the regression problem, and the parameter estimation procedure is called weighted least squares. The expressions for the parameter estimates and covariance matrix in this case are the same as for the generalized least squares, with  $\mathbf{V}^{-1}$  now being a diagonal matrix [cf. Eqs. (5.105) and (5.106), respectively]. This approach is equivalent to applying a weighting of  $\sqrt{w_{ii}}$  to both sides of the  $i$ th regression equation, where  $w_{ii}$  is the  $i$ th diagonal element of  $\mathbf{W}$ . The idea is that the weightings should be chosen to scale the measurement noise of each equation to approximately the same magnitude. Once that is achieved, the ordinary least-squares analysis and solution apply.

For flight-test data analysis, measured data for the linear regression problem come from several recorded time series, as in Example 5.1. Physically, the same instrumentation system is being used to collect each data sample, under similar conditions and at nearly the same time, so there is generally no justification for introducing unequal weightings to model heterogeneous variances. It is true that the residuals from a particular analysis usually exhibit varying magnitudes (cf. Fig. 5.6), but this is the result of model structure deficiencies, rather than a change in the measurement error variance from point to point. Accordingly, the expression (5.10) for ordinary least-squares parameter estimation is used for flight-test data analysis, which is equivalent to a constant weighting for each data point in the least-squares problem.

On the other hand, the residuals from flight-test data analysis are often significantly correlated with their adjacent neighbors, because the data are collected sequentially in time from a maneuvering aircraft. For static experiments, such as a wind-tunnel test, randomization of the test conditions is used to ensure that the residuals are uncorrelated. This cannot be done in a flight test, so there must be some correction for the correlated residuals. Correlated residuals have a significant effect on the estimated parameter covariance matrix.

The parameter covariance matrix for correlated residuals is derived starting with Eq. (5.11),

$$\text{Cov}(\hat{\boldsymbol{\theta}}) = E[(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})^T] = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T E(\mathbf{v} \mathbf{v}^T) \mathbf{X} (\mathbf{X}^T \mathbf{X})^{-1} \quad (5.107)$$

When  $\nu$  is a zero mean, weakly stationary random process,

$$E(\mathbf{v}\mathbf{v}^T) = E[\nu(i)\nu(j)] = \mathcal{R}_{\nu\nu}(i-j) = \mathcal{R}_{\nu\nu}(j-i) \quad i, j = 1, 2, \dots, N \quad (5.108)$$

where  $\mathcal{R}_{\nu\nu}(i-j)$  is the autocorrelation matrix for the residuals. The estimated parameter covariance matrix for correlated residuals can be computed by substituting for  $E(\mathbf{v}\mathbf{v}^T)$  from Eq. (5.108) into Eq. (5.107), using an estimate of  $\mathcal{R}_{\nu\nu}(i-j)$  from Eq. (5.67b),

$$\hat{\mathcal{R}}_{\nu\nu}(k) = \frac{1}{N} \sum_{i=1}^{N-k} \nu(i)\nu(i+k) = \hat{\mathcal{R}}_{\nu\nu}(-k) \quad k = 0, 1, 2, \dots, r \quad (5.109)$$

The index  $k$  represents the time separation of the residuals in the summation, and  $r$  is the maximum time index difference. Since only proximate residuals are significantly correlated, the value of  $r$  can be relatively small, which reduces the required computations. Combining the last three equations,

$$\text{Cov}(\hat{\boldsymbol{\theta}}) = (\mathbf{X}^T \mathbf{X})^{-1} \left[ \sum_{i=1}^N \mathbf{x}(i) \sum_{j=1}^N \hat{\mathcal{R}}_{\nu\nu}(i-j) \mathbf{x}^T(j) \right] (\mathbf{X}^T \mathbf{X})^{-1} \quad (5.110)$$

where  $\mathbf{x}^T(i)$  is the  $i$ th row of the  $\mathbf{X}$  matrix, containing the measured regressors at the  $i$ th data point, and  $\hat{\mathcal{R}}_{\nu\nu}(i-j)$  is replaced by its estimate from Eq. (5.109). Note that if the residuals are uncorrelated, then

$$E(\mathbf{v}\mathbf{v}^T) = \hat{\mathcal{R}}_{\nu\nu}(0)\mathbf{I} = \sigma^2 \mathbf{I} \quad (5.111)$$

and Eq. (5.107) reduces to the ordinary least-squares expression

$$\text{Cov}(\hat{\boldsymbol{\theta}}) = \sigma^2 (\mathbf{X}^T \mathbf{X})^{-1} \quad (5.112)$$

Equations (5.110) with (5.109) represent a postprocessing of the residuals from an ordinary least-squares solution to account for the effect of correlated residuals on the parameter covariance matrix. The standard errors for the estimated parameters, corrected for correlated residuals, are found as the square root of the diagonal elements of the covariance matrix calculated from Eq. (5.110).

In cases where generalized least squares is appropriate, but ordinary least squares is used instead, the parameter estimates are still unbiased, i.e.,  $E(\hat{\boldsymbol{\theta}}) = E(\hat{\boldsymbol{\theta}}_{\text{GLS}}) = \boldsymbol{\theta}$ . However, the misspecification of  $E(\mathbf{v}\mathbf{v}^T)$  results in a loss of efficiency in estimating  $\boldsymbol{\theta}$ , and a biased  $\text{Cov}(\hat{\boldsymbol{\theta}})$ . These problems are discussed in Refs. 1 and 9. In Ref. 10, investigation of bias in the parameter covariance matrix found that there is a tendency to underestimate the estimated parameter variances when the ordinary least-squares expression is used in cases where the residuals do not satisfy the assumptions made in the ordinary least-squares solution. Ref. 11 is the source for the corrected expression (5.110), which has

been found effective in characterizing estimated parameter accuracy for simulation and flight-test cases with many different colored noise sequences.

The following example demonstrates the covariance matrix correction for colored residuals applied to the analysis of flight-test data.

### Example 5.2

Returning to the analysis in Example 5.1, the same model parameters for the yawing moment coefficient are estimated using the same measured data from run 1 [cf. Eq. (5.102c) and Fig. 5.4]. Ordinary least-squares parameter estimates and the associated standard error estimates based on the white noise residual assumption appear in Table 5.3, columns 2 and 3, respectively. These columns are repeated from Table 5.1. Column 4 of Table 5.3 contains the estimated parameter standard errors using Eq. (5.110), which accounts for colored residuals. Column 5 of Table 5.3 shows the ratio of the standard errors corrected for colored residuals to those from the ordinary least-squares calculation.

In this case, which is typical, the standard errors computed based on the white residual assumption are smaller by roughly a factor of 3, compared with the standard errors calculated accounting for colored residuals.

The standard errors computed using the colored residual expression (5.110) are consistent with the scatter in parameter estimates from repeated maneuvers at the same flight condition. The conventional standard error calculation gives optimistic values for the estimated parameter accuracy when the residuals are colored.

Each repeated maneuver has its own specific noise sequences, corresponding to its particular realization of the random processes. The scatter in the parameter estimates from repeated maneuvers comes from the influence of the different noise sequences, which of course carries over into the parameter estimates that are based on the measured data. As discussed earlier, the parameter estimates

**Table 5.3 Least-squares estimated parameters and error bounds, aerodynamic yawing moment coefficient, run 1**

Parameter	$\hat{\theta}$	$s(\hat{\theta})$ white residual Eq. (5.112)	$s(\hat{\theta})_{\text{corr}}$ colored residual Eq. (5.110)	$\frac{s(\hat{\theta})_{\text{corr}}}{s(\hat{\theta})}$
$C_{n\beta}$	$8.54 \times 10^{-2}$	$3.58 \times 10^{-4}$	$1.16 \times 10^{-3}$	3.2
$C_{np}$	$-5.15 \times 10^{-2}$	$1.43 \times 10^{-3}$	$4.01 \times 10^{-3}$	2.8
$C_{nr}$	$-1.98 \times 10^{-1}$	$1.30 \times 10^{-3}$	$3.66 \times 10^{-3}$	2.8
$C_{n\delta_a}$	$2.34 \times 10^{-3}$	$5.00 \times 10^{-4}$	$1.27 \times 10^{-3}$	2.5
$C_{n\delta_r}$	$-1.31 \times 10^{-1}$	$5.97 \times 10^{-4}$	$1.88 \times 10^{-3}$	3.1
$C_{n_o}$	$-4.60 \times 10^{-4}$	$7.42 \times 10^{-6}$	$2.71 \times 10^{-5}$	3.7
$s = \hat{\sigma}$	$2.25 \times 10^{-4}$	—	—	—
$R^2, \%$	99.6	—	—	—

are random variables, characterized by expected values and standard deviations when the measurement noise is assumed to be Gaussian. The goal of the analysis is to estimate mean values and confidence intervals for the parameters so that the true values of the parameters lie inside the confidence intervals centered on the estimated mean values. True values of the parameters are likely to be close to the mean value of parameter estimates from repeated maneuvers. All of this assumes that an adequate model structure has been chosen for the repeated maneuvers, and that this model structure is held constant for the analysis of each maneuver.

The best approach would be to always run repeated maneuvers at every flight condition, but this is often not possible, due to economic and other practical constraints. Instead, one or two maneuvers at the same flight condition are usually available for analysis, and the colored noise correction is applied to compute a confidence interval that accurately represents the scatter in parameter estimates that would have occurred with numerous repeated maneuvers.

Column 5 of Table 5.3 shows that the factor by which the ordinary least-squares calculation is optimistic varies slightly with the parameter. This reflects differences in the information content of the regressors. The values for these factors also change with the coloring of the residual sequence. Figure 5.13 shows the power spectrum of the residual sequence shown in Fig. 5.6 for the yawing moment coefficient model. Most of the noise power is concentrated in the frequency range [0,4] Hz. This is a typical moderate coloring of the residuals.  $\square$

It is common practice to apply a constant correction factor (typically 5 or 10) to the standard errors computed from the ordinary least-squares calculation to

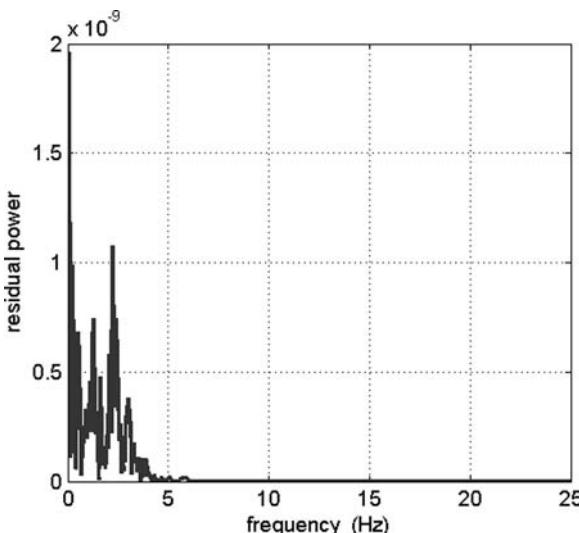


Fig. 5.13 Power spectrum of yawing moment coefficient residuals for lateral maneuver, run 1.

## REGRESSION METHODS

137

account for colored residuals. This practice is a rough approximation, however, and can be improved by using the colored residual formula (5.110) to compute the estimated parameter covariance matrix and standard errors. The correction for colored residuals can be run as a postprocessing of the residuals, after the model parameter estimates are computed using ordinary least squares.

Ideally, the generalized least-squares solution in Eqs. (5.105) and (5.106) should be used to refine the parameter estimates, using the residual sequence from ordinary least squares to estimate the weighting matrix  $V^{-1}$  from Eqs. (5.103b) and (5.108). Then, a new residual sequence could be obtained, and the process repeated until the parameter estimates converge. Unfortunately, this procedure involves inverting an  $N \times N$  weighting matrix  $V$ , which can be very large and ill-conditioned for data from typical flight-test maneuvers. Consequently, computing  $V^{-1}$  takes a relatively long time, and the result can be inaccurate, which in turn causes convergence problems with the generalized least-squares parameter estimates. Because of this, practical modeling results are found using the ordinary least-squares solution with error bounds corrected for colored residuals.

### 5.3 Nonlinear Least Squares

In some modeling problems, the relationship between the regressors and the response variable is nonlinear in the parameters. For this case, the least-squares model was formulated in Chapter 4 as

$$z = h(\boldsymbol{\theta}) + \nu \quad (5.113)$$

which is equivalent to a nonlinear regression model,

$$z(i) = f[\mathbf{x}(i), \boldsymbol{\theta}] + \nu(i) \quad i = 1, 2, \dots, N \quad (5.114)$$

where  $\mathbf{x}^T(i)$  is a row vector of regressors computed from measured data at the  $i$ th data point, and  $f$  is a nonlinear function of  $\mathbf{x}(i)$  and the parameters in the vector  $\boldsymbol{\theta}$ .

As before, the least-squares estimator can be obtained by minimizing the sum of squared errors,

$$J(\boldsymbol{\theta}) = \frac{1}{2} \sum_{i=1}^N \{z(i) - f[\mathbf{x}(i), \boldsymbol{\theta}]\}^2 \quad (5.115)$$

The minimum of the preceding cost function is found by satisfying the normal equations,

$$\left. \frac{\partial J}{\partial \boldsymbol{\theta}} \right|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}} = - \sum_{i=1}^N \{z(i) - f[\mathbf{x}(i), \hat{\boldsymbol{\theta}}]\} \left. \frac{\partial f[\mathbf{x}(i), \boldsymbol{\theta}]}{\partial \boldsymbol{\theta}} \right|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}} = \mathbf{0} \quad (5.116)$$

where  $\partial J / \partial \boldsymbol{\theta}$  is a row vector containing the partial derivatives of the nonlinear scalar function  $J(\boldsymbol{\theta})$  with respect to the elements of  $\boldsymbol{\theta}$ , and  $\partial f[\mathbf{x}(i), \boldsymbol{\theta}] / \partial \boldsymbol{\theta}$  is a row vector of output sensitivities to changes in the model parameters.

Equation (5.116) is a set of nonlinear algebraic equations. This means that  $\hat{\theta}$  cannot be obtained by simple matrix algebra, as in the case of a model that is linear in the parameters. Instead, an iterative nonlinear optimization technique must be used. There are many different numerical methods to solve this nonlinear minimization problem (see Ref. 12). Some of these methods will be explained in Chapter 6.

## 5.4 Model Structure Determination

Up to this point, the data analysis techniques for linear regression problems were concerned with parameter estimation for an assumed model structure in the regression equation (5.4). The model structure refers to the number and form of the model terms in the regression equation. When the model structure is assumed known and fixed, only the constant model parameter values and their standard errors need to be estimated from the measured data. This can be done using the parameter estimation techniques discussed earlier.

Assuming a model structure also implicitly assumes that each of the terms in the model makes a significant contribution to modeling the variation in the measured response  $z$ . Choice of an adequate model structure depends heavily on how the experiment was conducted. However, in many practical cases it is not clear exactly what the model structure should be. From theory, previous experiments, or knowledge of the physical system to be modeled, an analyst can specify candidate regressors that might be considered for the model. Then the task is to select a subset of the candidate regressors that best model the response variable based on the measured data. This procedure is called model structure determination.

In aircraft applications of system identification, the estimation of stability and control derivatives has become a standard procedure for small perturbation flight-test maneuvers where the aerodynamics can be described using regressors that are linear in the independent variables. Interest in near- and post-stall flight regimes, and in dynamics of rapid, large-amplitude maneuvers has created a need to extend aerodynamic modeling into flight regimes where nonlinear aerodynamic effects can be pronounced. Nonlinear terms can also be required to extend the validity of models to large ranges of the independent variables. These applications introduce the problem of determining how complex the model should be.

In Chapter 3, several nonlinear forms for the aerodynamic model were discussed, including indicial functions, splines, and multivariate polynomials in a Taylor series expansion. Splines and multivariate polynomials have been the most frequently used modeling functions for characterizing nonlinearity, because of their relatively simple form and physical interpretation. Adding these types of terms to the linear Taylor series terms that include stability and control derivatives also seems like a natural extension to model more complicated dependencies.

In determining the model structure, there are two conflicting objectives. On one hand, a model with many regressors might be desired so that the model can describe nearly all of the variation in the dependent variable. On the other hand, one would like to have a model with as few regressors as possible,

because the variance of the prediction  $\hat{y}$  increases as the number of regressors increases. If too many parameter estimates are sought from a given set of measured data, reduced accuracy of the estimated parameters can be expected, or, in some cases, attempts to obtain reasonable estimates of all of the model parameters can fail. More informative data, which often means a greater volume of data, are needed to support accurate estimation of more model parameters.

Several algorithms for model structure determination based on measured data have been developed. Often, different methods will select different subsets of candidate regressors as the best model. There is no guarantee that any selected procedure will give the one best model, because there might be several equally good models for characterizing a particular data set, especially when nonlinearity is involved. Extensive treatments of model structure determination are given in Refs. 1–3 and 13.

In the following sections, the properties of a reduced linear regression model will be discussed first, followed by the development of stepwise regression for model structure determination. This approach has been successfully applied to flight-test data for many aircraft problems. Several criteria for selecting the best model will be introduced, along with a method for using multivariate orthogonal functions to determine model structure. Then, nonlinear model structure determination will be demonstrated in an example.

#### 5.4.1 Properties of Reduced Models

A model with one or more regressors removed relative to another model will be called a reduced model. The properties of a reduced model can be determined by comparison with the properties of a complete model with  $n_p = n + 1$  regressors,

$$z = \mathbf{X}\boldsymbol{\theta} + \boldsymbol{\nu} \quad (5.117)$$

This model can be written as

$$z = \mathbf{X}_1\boldsymbol{\theta}_1 + \mathbf{X}_2\boldsymbol{\theta}_2 + \boldsymbol{\nu} \quad (5.118)$$

where the  $N \times n_p$  matrix  $\mathbf{X}$  has been partitioned into the  $N \times p$  matrix  $\mathbf{X}_1$  and the  $N \times q$  matrix  $\mathbf{X}_2$ , with  $n_p = p + q$ . The parameter vector is similarly partitioned into  $\boldsymbol{\theta}_1$  and  $\boldsymbol{\theta}_2$ . Equation (5.118) is the same as Eq. (5.48), which was introduced to investigate the significance of  $\mathbf{X}_2\boldsymbol{\theta}_2$  in the regression model.

If the model of Eq. (5.117) is correct, then the properties of  $\hat{\boldsymbol{\theta}}$  and  $\hat{\sigma}^2$  as estimates of  $\boldsymbol{\theta}$  and  $\sigma^2$  are well known. Properties of the ordinary least-squares estimates of  $\boldsymbol{\theta}_1$  and  $\sigma^2$  in the reduced model,

$$z = \mathbf{X}_1\boldsymbol{\theta}_1 + \boldsymbol{\nu} \quad (5.119)$$

have been investigated by many authors. Their findings are summarized in Refs. 3 and 13. The results motivate the use of reduced models rather than the complete model for the reasons outlined next.

Deleting regressors associated with parameters that have small numerical values and large standard errors will result in higher accuracy for the parameter estimates of the retained regressors. Deleting regressors from the model also potentially introduces bias in the estimates of  $\theta_1$  in the reduced model. However, the variance of the biased estimates  $\tilde{\theta}_1$  using the reduced model will be smaller than the variance of the unbiased estimates  $\hat{\theta}_1$ , which is the estimate of  $\theta_1$  using the full model structure. The mean squared error of the biased estimates will also be smaller than the variance of the unbiased estimates, that is

$$\begin{aligned} \text{Var}(\tilde{\theta}_1) &\leq \text{Var}(\hat{\theta}_1) \\ \text{MSE}(\tilde{\theta}_1) &\leq \text{Var}(\hat{\theta}_1) \end{aligned} \quad (5.120a)$$

where

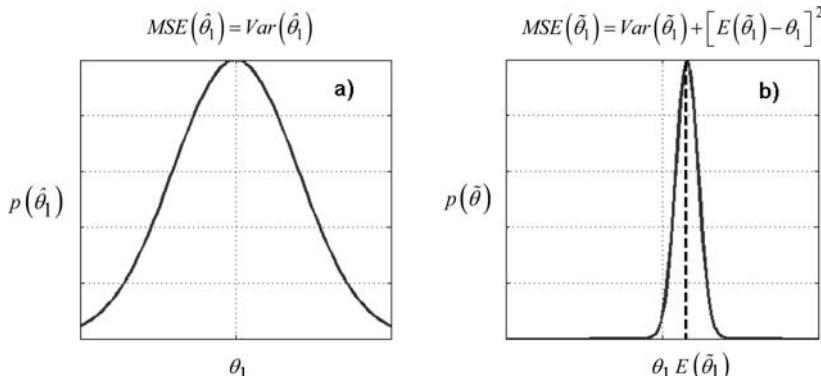
$$\text{MSE}(\tilde{\theta}_1) = \text{Var}(\tilde{\theta}_1) + [E(\tilde{\theta}_1) - \theta_1]^2 \quad (5.120b)$$

Note that the mean squared error of a parameter estimate is composed of the random error variance plus the squared bias error (see Ref. 9). When the parameter estimate is unbiased, as for the full model structure, then the mean squared error equals the random error variance.

The situation is illustrated in Fig. 5.14 by presenting probability densities of a single estimated parameter  $\theta_1$  from a complete model,  $p(\hat{\theta}_1)$ , and from a reduced model,  $p(\tilde{\theta}_1)$ . The parameter estimate  $\hat{\theta}_1$  is unbiased, whereas the parameter estimate  $\tilde{\theta}_1$  is biased by  $E(\tilde{\theta}_1) - \theta_1$ .

If the reduced model is used for prediction, then the predicted value of the response at the  $i$ th data point is

$$\tilde{y}_1(i) = \mathbf{x}_1^T(i)\tilde{\theta}_1 \quad (5.121)$$



**Fig. 5.14** Probability density of a parameter estimate from a) a complete model and b) a reduced model.

with mean value

$$E[\tilde{y}_1(i)] = \mathbf{x}_1^T(i)\boldsymbol{\theta}_1 + \mathbf{x}_1^T(i)\mathbf{A}\boldsymbol{\theta}_2 \quad (5.122)$$

where

$$\mathbf{A} = (\mathbf{X}_1^T\mathbf{X}_1)^{-1}\mathbf{X}_1^T\mathbf{X}_2 \quad (5.123)$$

and prediction interval variance is

$$\text{Var}[z(i) - \tilde{y}_1(i)] = \sigma^2[1 + \mathbf{x}_1^T(i)(\mathbf{X}_1^T\mathbf{X}_1)^{-1}\mathbf{x}_1(i)] \quad (5.124)$$

The mean squared error for the prediction is given by

$$\begin{aligned} \text{MSE}[z(i) - \tilde{y}_1(i)] &= E\{[z(i) - \tilde{y}_1(i)]^2\} \\ &= \sigma^2[1 + \mathbf{x}_1^T(i)(\mathbf{X}_1^T\mathbf{X}_1)^{-1}\mathbf{x}_1(i)] \\ &\quad + [\mathbf{x}_1^T(i)\mathbf{A}\boldsymbol{\theta}_2 - \mathbf{x}_2^T(i)\boldsymbol{\theta}_2]^2 \end{aligned} \quad (5.125)$$

Detailed development of Eq. (5.125) is given in Ref. 13. Combining the last two equations, the prediction mean squared error can also be expressed as

$$\text{MSE}(z - \tilde{y}_1) = \text{Var}(z - \tilde{y}_1) + [E(\tilde{y}_1) - y]^2 \quad (5.126)$$

Similarly to the properties of parameter estimates from a reduced model shown previously,

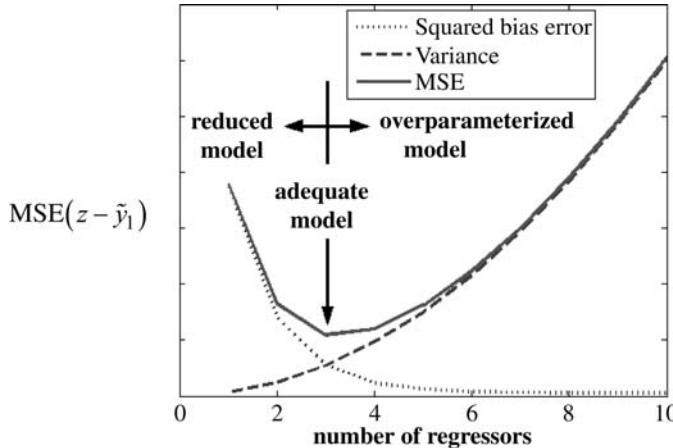
$$\begin{aligned} \text{Var}(z - \tilde{y}_1) &\leq \text{Var}(z - \hat{y}) \\ \text{MSE}(z - \tilde{y}_1) &\leq \text{Var}(z - \hat{y}) \end{aligned} \quad (5.127)$$

As the number of regressors retained in the model increases,  $\text{Var}(z - \tilde{y}_1)$  will increase from its minimum value, but the bias error will decrease. This means that there will be some minimum of  $\text{MSE}(z - \tilde{y}_1)$  for a certain number of regressors in the model. Figure 5.15 demonstrates this by showing  $\text{MSE}(z - \tilde{y}_1)$  and its components plotted against number of regressors in the model.

The material in this section provides theoretical substantiation for the parsimony principle stated earlier and discussed further in Sec. 5.4.3. Models with too many terms (or too few terms) can lead to reduced accuracy for estimated model parameters and degraded prediction capability. Too many terms increase the variance error, whereas too few terms increase the bias error.

#### 5.4.2 Stepwise Regression

This section explains computational techniques that can be used to evaluate subsets of a pool of candidate regressors for inclusion in a linear regression model, by adding or deleting regressors one at a time. The procedures can be



**Fig. 5.15 Variation of prediction mean square error with number of regressors in the model.**

classified into three basic categories: forward selection, backward elimination, and stepwise regression, which is a combination of the first two.

**Forward selection.** This technique begins with the model that has no regressors, only the intercept or bias term  $\theta_0$ . Then one regressor at a time is added until all candidate regressors are in the model or until some selection criterion is satisfied. The first regressor selected for entry into the regression equation is the one that has the highest simple correlation with the dependent variable, adjusted for the mean value. Equation (5.79) is used to calculate the correlation coefficients for each regressor with the dependent variable, adjusted for the mean value. The regressor with the highest absolute value of correlation is also the regressor that yields the largest value of the partial  $F$  statistic for testing the significance of a single added regressor [cf. Eq. (5.59)]. This regressor enters if the partial  $F$  statistic exceeds a preselected value, usually called  $F$ -to-enter, or  $F_{in}$ . The second regressor selected for entry into the model is the one with the largest correlation with  $z - \theta_0 - \theta_1 \xi_1$ , which is the measured dependent variable adjusted for the effect of the mean and the first regressor in the model. This process continues until all candidate regressors are in the model, or no remaining candidate regressors pass the  $F_{in}$  criterion for entry into the model. In general, a regressor  $\xi_j$  is added to a model with  $p$  terms if [cf. Eq. (5.55)]

$$F_0 = \frac{SS_R(\hat{\boldsymbol{\theta}}_{p+j}) - SS_R(\hat{\boldsymbol{\theta}}_p)}{s^2} = \frac{SS_R(\hat{\theta}_j | \hat{\boldsymbol{\theta}}_p)}{s^2} > F_{in} \quad (5.128)$$

where  $SS_R(\hat{\boldsymbol{\theta}}_p)$  is the regression sum of squares using the  $p$  terms already in the model, and  $SS_R(\hat{\boldsymbol{\theta}}_{p+j})$  is the regression sum of squares obtained by adding the  $j$ th regressor to the original  $p$  terms. The fit error variance  $s^2$  is computed in the usual

## REGRESSION METHODS

143

way, assuming the  $j$ th regressor is included in the model [cf. Eq. (5.24)],

$$s^2 \equiv \hat{\sigma}^2 = \frac{\mathbf{v}^T \mathbf{v}}{(N - p - 1)} = \frac{\sum_{i=1}^N [z(i) - \hat{y}(i)]^2}{(N - p - 1)} \quad (5.129)$$

The value of  $F_{\text{in}}$  is  $F(\alpha; 1, N - p - 1)$ , where  $\alpha$  is the selected significance level.

**Backward elimination.** This technique begins with all of the candidate regressors included in the regression equation. Unnecessary regressors are then removed one at a time. At each step, the regressor with the smallest partial  $F$ -ratio, computed from the current regression, is eliminated if

$$F_0 = \min_j \frac{SS_R(\hat{\boldsymbol{\theta}}_p) - SS_R(\hat{\boldsymbol{\theta}}_{p-j})}{s^2} < F_{\text{out}} \quad (5.130)$$

where  $F_{\text{out}}$  is  $F(\alpha; 1, N - p)$ , and  $SS_R(\hat{\boldsymbol{\theta}}_{p-j})$  is the regression sum of squares obtained by removing the  $j$ th regressor from the original  $p$  terms. Backward elimination is the converse of forward selection, in the sense that forward selection starts with no regressors in the model and adds terms, whereas backward elimination starts with every candidate regressor in the model and removes terms.

**Stepwise regression.** The stepwise regression method described here is a combination of forward selection and backward elimination, proposed in Ref. 14. It modifies the forward selection by adding backward elimination to reassess the regressors entered into the model. A regressor added at an early stage may become redundant because of its relationship with regressors added subsequently to the model.

Stepwise regression starts by constructing a pool of candidate regressors, and modeling the dependent variable using only the intercept or bias term, as for forward selection. Then the first regressor for the model is chosen as the one with the highest absolute value of correlation with  $z$ , adjusted for the mean value. The correlations are computed from Eq. (5.79), as in the case of forward selection. If  $\xi_1$  is selected as the first regressor in the model, then the model

$$z(i) = \theta_0 + \theta_1 \xi_1(i) + v(i) \quad i = 1, 2, \dots, N \quad (5.131)$$

is used with ordinary least-squares parameter estimation to fit the measured data. The partial  $F$  statistic is computed as

$$F_0 = \frac{SS_R(\theta_1)}{s^2} \quad (5.132)$$

and compared with  $F_{\text{in}}$ . If  $F_0 > F_{\text{in}}$ , the model (5.131) is accepted as the model at the first step.

Next, new regressors are constructed from the remaining regressors in the pool of candidate regressors, by removing the part of each remaining candidate regressor vector that is in the direction of terms already in the model. This can be done by modeling each regressor in the remaining pool of regressors with  $\xi_1$  and a vector of ones (for the bias term), and treating the residuals as the new pool of regressors. This is the equivalent of removing any variation in the remaining candidate regressors that is included in terms already in the model. For example, regressor  $\xi_2$  is modeled as

$$\xi_2(i) = a_0 + a_1 \xi_1(i) + v_2(i) \quad i = 1, 2, \dots, N \quad (5.133)$$

so that the new regressor  $v_2$  is given by

$$v_2(i) = \xi_2(i) - \hat{a}_0 - \hat{a}_1 \xi_1(i) \quad i = 1, 2, \dots, N \quad (5.134)$$

with the parameter estimates  $\hat{a}_0$  and  $\hat{a}_1$  computed from ordinary least-squares parameter estimation. The other remaining regressors in the pool of candidate regressors are similarly modified, where each remaining regressor will have its own parameter values analogous to  $\hat{a}_0$  and  $\hat{a}_1$  in Eq. (5.133). Note that this procedure orthogonalizes the remaining candidate regressors with respect to the regressors currently in the model [cf. Eq. (5.9c)]. In the same way, a new dependent variable is computed from

$$v_2(i) = z(i) - \hat{\theta}_0 - \hat{\theta}_1 \xi_1(i) \quad i = 1, 2, \dots, N \quad (5.135)$$

For the next step, a new set of correlations involving the new regressors and the new dependent variable is computed, and the process is repeated. Each stage removes the effects of model terms already chosen, so that the focus of the model structure determination is always on the model terms necessary to characterize the variation in the dependent variable that has not yet been modeled. This is helpful, because the magnitude of the early dominant terms can dwarf later terms that might be necessary for modeling smaller magnitude variations in the dependent variable. Note that only the partial correlation calculation uses quantities corrected for terms already in the model; all other calculations use the original candidate regressors and dependent variable.

At every step, the regressors incorporated into the model at previous stages and the new regressor just entering the model are examined. This is done for the regression model in the form of Eq. (5.4), but with only the selected regressors included in the model. For this part of the analysis, the selected regressors and the dependent variable are in their original form, not the conditioned form used for the forward selection step described earlier. This is different from orthogonal function modeling, where the regressors in the model are mutually orthogonal, and the least-squares parameter estimation problem is decoupled. The partial  $F$  criterion from Eq. (5.130) for backward elimination is

evaluated for each regressor in the model and compared with the value of  $F_{\text{out}}$ . Any regressor that provides a nonsignificant contribution, evidenced by a small value of  $F_0$ , is removed from the model. A regressor that may have been the best single regressor to enter at an early stage may, at a later stage, be redundant because of a linear or nearly linear relationship with other regressors in the model. The process of selecting and removing regressors continues until no more regressors are admitted and no more are rejected.

The preselected values of  $F_{\text{in}}$  and  $F_{\text{out}}$  depend on the number of data points  $N$ , the current number of regressors  $p$ , and the selected confidence level  $1 - \alpha$ . For most practical aircraft problems,  $N \gg 100$  and  $p < 10$ , so the effect of  $p$  on values of  $F(\alpha; 1, N - p)$  is small. Therefore, for 95% confidence,  $F(0.05; 1, N - p)$  can be taken as a constant equal to 4. Some analysts prefer to choose  $F_{\text{in}} > F_{\text{out}}$ , which means it will be relatively more difficult to add a regressor than to remove one.

#### 5.4.3 Statistical Metrics and Stopping Rules

The subset of regressors selected for the model by stepwise regression can be determined according to stopping rules that apply to statistical metrics computed from the measured data. Using only the partial  $F$  statistic with  $F_{\text{in}}$  and  $F_{\text{out}}$  can sometimes be too restrictive. A large preselected value of  $F_{\text{in}}$  can terminate the selection procedure before all influential regressors are included in the model. On the other hand, if  $F_{\text{out}}$  is chosen small, some of the regressors with limited influence can be retained in the model. For these reasons, several additional metrics have been proposed to find an adequate model from a given set of experimental data. Optimal values of these metrics provide a guide for selection of a model with good fit to the data and good prediction capability. Practical experience has shown that it is useful to consider more than one statistical metric when selecting the model structure, because each statistical metric has its own peculiarities. A model structure based on a consensus among several good statistical metrics improves confidence in the model selection, and this approach has been found to give excellent results.

As mentioned earlier, there are contradictory arguments regarding the appropriate number of parameters in the model. A rule commonly used in choosing a model that would be a good predictor is known as the principle of parsimony.<sup>15</sup> This principle can be stated as follows:

*Given two models fitted to the same data with nearly equal residual variances, choose the model with the fewest parameters.*

Based on this principle, the objective is to include only model terms that significantly decrease the residual variance. The criterion for judging whether or not a given decrease in the residual variance is significant is some type of comparison with the variance of the measurement noise on the dependent variable. Generally speaking, if an additional model term does not characterize an effect with magnitude significantly greater than the noise level, then that term might very well be modeling noise, which is spurious and nonrepeatable. It follows that a model with too many parameters is a poor predictor. Some

statistical metrics address this problem directly by quantifying the prediction error of an identified model.

Several stopping rules for stepwise regression applied to aircraft aerodynamic model structure determination have been proposed and tested by Klein et al.<sup>16</sup> The selection of these stopping rules was influenced by the relationship between the number of parameters in the model  $p$ , and the number of data points  $N$ . For aircraft flight-test data analysis, usually  $N \gg 100$  and  $p < 10$ . For that reason, the stopping rules that are sensitive to the difference  $N - p$  were not considered. Note that in this section, as in the last section,  $p$  is used to denote the current number of parameters in the model, which may be different from the number of parameters in the final model,  $n_p$ .

The following statistical metrics have been used successfully for aircraft aerodynamic model structure determination.

*Coefficient of determination,  $R^2$ .* The coefficient of determination was introduced in Sec. 5.1 [cf. Eq. (5.31)],

$$R^2 = \frac{SS_R}{SS_T} = 1 - \frac{SS_E}{SS_T} = \frac{\hat{\theta}^T X^T z - N\bar{z}^2}{z^T z - N\bar{z}^2} \quad (5.136)$$

where  $0 \leq R^2 \leq 1$ . Often,  $R^2$  is given as a percentage. Adding a regressor to the model will always increase  $R^2$ . However, the more influential the added term is, the greater the change in  $R^2$ . This means that in stepwise regression, the rise in  $R^2$  at the beginning can be large. After all influential terms are in the model, the changes in  $R^2$  with additional terms will be small. An adequate model is achieved when  $R^2$  is not substantially changed by adding a new term to the model. If an added model term increases  $R^2$  by less than 0.5 percent, that term is probably not significant, and should be omitted.

*F statistic.* The  $F$  statistic for testing significance of a regression is given by Eq. (5.46) as

$$\begin{aligned} F_0 &= \frac{SS_R/n}{SSE/(N-p)} \\ &= \frac{\hat{\theta}^T X^T z - N\bar{z}^2}{ns^2} \end{aligned} \quad (5.137)$$

where  $n$  is the number of regressors in the current model, excluding the bias term, so  $n = p - 1$ . Using Eqs. (5.136) and (5.137), the relationship between  $F_0$  and  $R^2$  is

$$F_0 = \frac{(N-p)SS_R}{(p-1)SS_E} = \frac{(N-p)}{(p-1)} \frac{SS_R}{SS_T - SS_R} = \frac{(N-p)}{(p-1)} \frac{R^2}{1-R^2} \quad (5.138)$$

## REGRESSION METHODS

147

This relationship indicates that the changes in  $F_0$  at the beginning of the model selection process can be quite pronounced. Later, however, when less influential terms are added into model, the increase in  $R^2/(1 - R^2)$  could be smaller than the decrease in  $(N - p)/(p - 1)$ . As a consequence,  $F_0$  values could reach a maximum value and then gradually decrease with more terms included in the model. Based on that behavior of  $F_0$ , it was suggested by Hall et al.<sup>17</sup> that  $F_0 = F_{0\max}$  be used as a stopping rule for the model that provides the best fit to the data with the smallest number of parameters.

*Predicted sum of squares, PRESS.* The PRESS statistic proposed in Ref. 18 is defined as the sum of  $N$  residuals:

$$\text{PRESS} = \sum_{i=1}^N \{z(i) - \hat{y}[i|\mathbf{x}(1), \mathbf{x}(2), \dots, \mathbf{x}(i-1), \mathbf{x}(i+1), \dots, \mathbf{x}(N)]\}^2 \quad (5.139)$$

Equation (5.139) shows that PRESS uses each possible subset of  $N - 1$  observations as the parameter estimation data set, and every observation in turn as a single prediction point. For practical computation, PRESS can be reformulated<sup>18</sup> as

$$\text{PRESS} = \sum_{i=1}^N \left[ \frac{v(i)}{(1 - k_{ii})} \right]^2 \quad (5.140)$$

which means that the PRESS residuals are just the ordinary residuals normalized by the diagonal elements of the prediction matrix,

$$k_{ii} = \mathbf{x}^T(i)(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{x}(i) \quad (5.141)$$

where  $\mathbf{x}^T(i)$  is the  $i$ th row of the  $\mathbf{X}$  matrix. The model associated with the minimum PRESS will be a good predictor, so minimum PRESS is used as a stopping rule.

The usefulness of minimum PRESS as a stopping rule in model structure determination can be limited in cases where  $N \gg p$ , which is typical of flight-test data analysis. The behavior of the PRESS statistic for an increasing number of data points  $N$  can be examined from the limit of  $k_{ii}$  as  $N \rightarrow \infty$ . This limit can be formulated as

$$\lim_{N \rightarrow \infty} k_{ii} = \lim_{N \rightarrow \infty} \mathbf{x}^T(i)(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{x}(i) = 0 \quad (5.142)$$

where the limit of the scalar  $k_{ii}$  is zero, because as more data are collected, the elements of the  $\mathbf{X}^T \mathbf{X}$  matrix become arbitrarily large, whereas each single row of data  $\mathbf{x}^T(i)$  remains fixed. From Eqs. (5.140) and (5.142), it is apparent that PRESS approaches the residual sum of squares  $SS_E$  as the number of data points  $N$  increases. In this situation, PRESS can only decrease as regression terms are added to the model. To preserve the ability of PRESS to reach a

minimum as  $p$  is changed, it was recommended in Ref. 16 that a reduced number of data points (e.g., every tenth point) be used in computing PRESS, to keep  $N$  low. This recommendation has no substantiation, of course. Research needs to be done to find an optimal selection of data points for the PRESS stopping rule.

**Predicted square error, PSE.** Another statistical metric that can be used for model structure determination is the predicted squared error (PSE) defined by

$$\begin{aligned} \text{PSE} &\equiv \frac{1}{N}(z - \hat{y})^T(z - \hat{y}) + \sigma_{\max}^2 \frac{p}{N} \\ &= \text{MSFE} + \sigma_{\max}^2 \frac{p}{N} \end{aligned} \quad (5.143)$$

where MSFE is the mean squared fit error for the modeling data,

$$\text{MSFE} \equiv \frac{1}{N}(z - \hat{y})^T(z - \hat{y}) = \frac{1}{N}(\mathbf{v}^T \mathbf{v}) \quad (5.144)$$

The MSFE is proportional to the ordinary least-squares cost function given in Eq. (5.8). The quantity  $\sigma_{\max}^2$  is a constant to be discussed later,  $p$  is the number of terms in the current model, and  $N$  is the number of data points. The PSE in Eq. (5.143) depends on the MSFE,  $(\mathbf{v}^T \mathbf{v})/N$ , and a term proportional to the number of terms in the model,  $p$ . The MSFE decreases with each additional term added to the model, so that minimizing MSFE alone would provide no protection against overfitting the data by adding too many terms to the model. The overfit penalty term  $\sigma_{\max}^2 p/N$  in the PSE increases with each added model term, because  $p$  increases. This term prevents overfitting the data with too many model terms, which is detrimental to model prediction accuracy, as discussed previously.

The constant  $\sigma_{\max}^2$  is the *a priori* upper-bound estimate of the squared error between future data and the model, i.e., the upper-bound mean squared error for prediction cases. The upper bound is used in the model overfit penalty term to account for the fact that PSE is calculated when the model structure is not correct, i.e., during the model structure determination stage. Using the upper bound is conservative in the sense that model complexity will be minimized as a result of using an upper bound for this constant in the penalty term. Because of this, the value of PSE computed from Eq. (5.143) for a particular model structure tends to overestimate actual prediction errors on new data. Therefore, the PSE metric conservatively estimates the squared error for prediction cases.

A simple estimate of  $\sigma_{\max}^2$  that is independent of the model structure can be obtained by considering  $\sigma_{\max}^2$  to be the residual variance estimate corresponding to a constant model equal to the mean of the measured response values,

$$\sigma_{\max}^2 = \frac{1}{N} \sum_{i=1}^N [z(i) - \bar{z}]^2 \quad (5.145)$$

## REGRESSION METHODS

149

where

$$\bar{z} = \frac{1}{N} \sum_{i=1}^N z(i) \quad (5.146)$$

For wind-tunnel testing, repeated runs at the same test conditions are often available. If  $\hat{\sigma}^2$  is the measurement error variance estimated from measurements of the dependent variable for repeated runs at the same test conditions [cf. Eq. (5.22)], then  $\sigma_{\max}^2$  can be estimated as

$$\sigma_{\max}^2 = 25\hat{\sigma}^2 \quad (5.147)$$

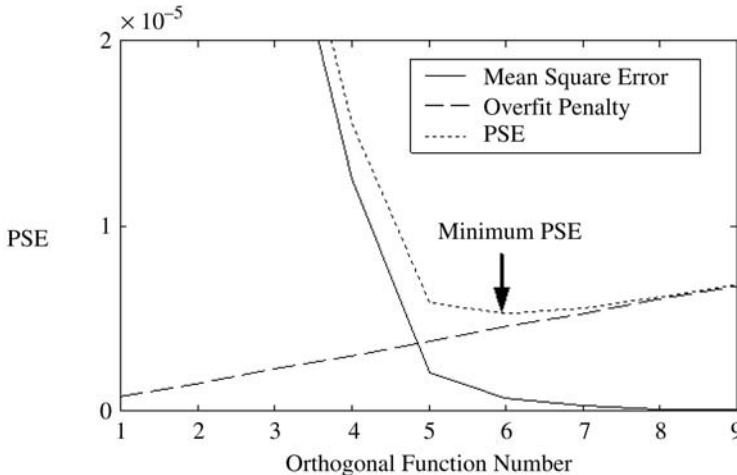
If the residuals were Gaussian, Eq. (5.147) would correspond to conservatively placing the maximum output variance at 25 times the estimated value, corresponding to a  $5\hat{\sigma}$  maximum deviation. However, the estimate  $\hat{\sigma}^2$  may not be very good, because of relatively few repeated runs available, or errors in the independent variable settings, or drift errors when duplicating test conditions for the repeat runs. The  $5\hat{\sigma}$  value has been found to give accurate models in model identification algorithm tests. In Ref. 19, the model structure determined using PSE with multivariate orthogonal functions was found to be virtually the same for  $\sigma_{\max}^2$  in the range

$$9\sigma_o^2 \leq \sigma_{\max}^2 \leq 100\sigma_o^2 \quad (5.148)$$

This happens because the plot of mean squared fit error versus added modeling functions is typically very flat in the region of minimum PSE.

When multivariate orthogonal modeling functions are used in model structure determination, the effectiveness in reducing the MSFE for each individual orthogonal function can be computed independently of which other orthogonal functions are already selected for the model [cf. Eqs. (5.96) and (5.97)]. Introducing orthogonal modeling functions into the model in order of most effective to least effective in reducing the mean squared fit error (quantified by  $(\mathbf{p}_j^T \mathbf{z})^2 / (\mathbf{p}_j^T \mathbf{p}_j)$  for the  $j$ th orthogonal modeling function  $\mathbf{p}_j$ ) means that the MSFE will decrease monotonically as orthogonal model terms are added, while the overfit penalty term will increase monotonically. The PSE metric will then always have a single global minimum value.

Figure 5.16 depicts this graphically, using actual modeling results from Ref. 19. The figure shows that after the first six modeling functions, the added model complexity associated with an additional orthogonal modeling function is not justified by the associated reduction in mean squared fit error. This point is marked by a minimum PSE, which defines an adequate model structure with good predictive capability. Note that Fig. 5.16 is analogous to Fig. 5.15; however, the values shown in Fig. 5.16 are components of the PSE metric, computed from measured wind-tunnel data. Therefore, Fig. 5.16 is a practical realization of the conceptual plot in Fig. 5.15. Barron<sup>20</sup> provides further justifying



**Fig. 5.16 Model structure determination using orthogonal functions and PSE.**

statistical arguments and analysis for the form of PSE given in Eq. (5.143), including justification for its use in modeling problems.

Using orthogonal functions to model the response variable makes it possible to evaluate the merit of including each modeling function individually, using the predicted squared error PSE, because of the properties of orthogonal functions and the resultant decoupling of the associated least-squares parameter estimation problem. Since the goal is to select a model structure with minimum PSE, and the PSE always has a single global minimum for orthogonal modeling functions, the model structure determination becomes a well-defined and straightforward process that can be automated. After the model structure is identified in terms of orthogonal functions, the retained orthogonal functions can be decomposed into the original regressors from which they were generated [cf. Eq. (5.88)]. This decomposition is done to express the model in physically meaningful terms.

**Other statistical metrics.** In addition to the metrics listed earlier and their associated stopping rules, at each step of the model selection procedure, the values of the fit error and  $F_0$  or  $t_0$  statistic for each model parameter can be computed. The estimated fit error

$$s = \sqrt{\frac{SS_E}{N-p}} = \sqrt{\frac{\sum_{i=1}^N [z(i) - \hat{y}(i)]^2}{N-p}} \quad (5.149)$$

should be compared with an unbiased estimate of the standard deviation of the measurement error on the dependent variable. One method for obtaining such an estimate is given in Ref. 21. In this approach, the random measurement

error is separated from the deterministic part of the measured dependent variable using an optimal Fourier smoother (see Chapter 11).

The  $F_0$  and  $t_0$  statistics for each estimated parameter are obtained from Eqs. (5.59) and (5.60)

$$F_0 = \frac{\hat{\theta}_j^2}{s^2(\hat{\theta}_j)} \quad (5.150)$$

$$t_0 = \frac{\hat{\theta}_j}{s(\hat{\theta}_j)} \quad (5.151)$$

These expressions for the  $F_0$  and  $t_0$  statistics represent the inverse of the relative parameter variance and the inverse of the relative parameter standard error, respectively. Both of these statistics should be large for the terms in an adequate model. The forms shown in Eqs. (5.150) and (5.151) make it clear that these metrics can be interpreted as tests for the value of an estimated parameter being statistically different from zero.

### Example 5.3

This example demonstrates the use of stepwise regression for model structure determination and parameter estimation. The application is the vertical aerodynamic force coefficient  $C_Z$  from a piloted nonlinear simulation of the F-16 aircraft. Appendix D gives a full description of the F-16 simulation, which runs in MATLAB® and is included in the software associated with this text. There is no automatic feedback control system included in the F-16 simulation, so the pilot stick and rudder commands move the control surfaces directly. Flaps and spoilers are fixed, which means that the elevator is the only movable longitudinal control surface. The c.g. is located at a forward position ( $x_{c,g.} = 0.25\bar{c}$ ) to achieve open-loop static stability, which allows the F-16 to be flown without an automatic feedback control system for stability augmentation.

Figure 5.17 shows measured time series from a piloted large-amplitude longitudinal maneuver, where the pilot applied small-amplitude longitudinal stick perturbations on top of a steady pull-up and push-over. This produces data that cover large ranges of angle of attack, pitch rate, and elevator deflection. Figure 5.18 shows cross plots of angle of attack, nondimensional pitch rate, and elevator deflection, indicating the effectiveness of this maneuver in covering a large portion of the independent variable space. These cross plots can also be interpreted as a graphical indication of the pair-wise correlation between independent variables. If two independent variables were highly correlated, then their cross plot would approximate a straight line. The independent variables plotted in Fig. 5.18 show very low pair-wise correlations.

Each output variable from the nonlinear simulation was corrupted by white Gaussian noise with magnitude typical of flight-test instrumentation. The resulting signal-to-noise ratio was approximately 30 for the measured response variable  $C_Z$  shown in Fig. 5.19. This signal-to-noise ratio is relatively high, and is more than adequate for good modeling results. For small-perturbation

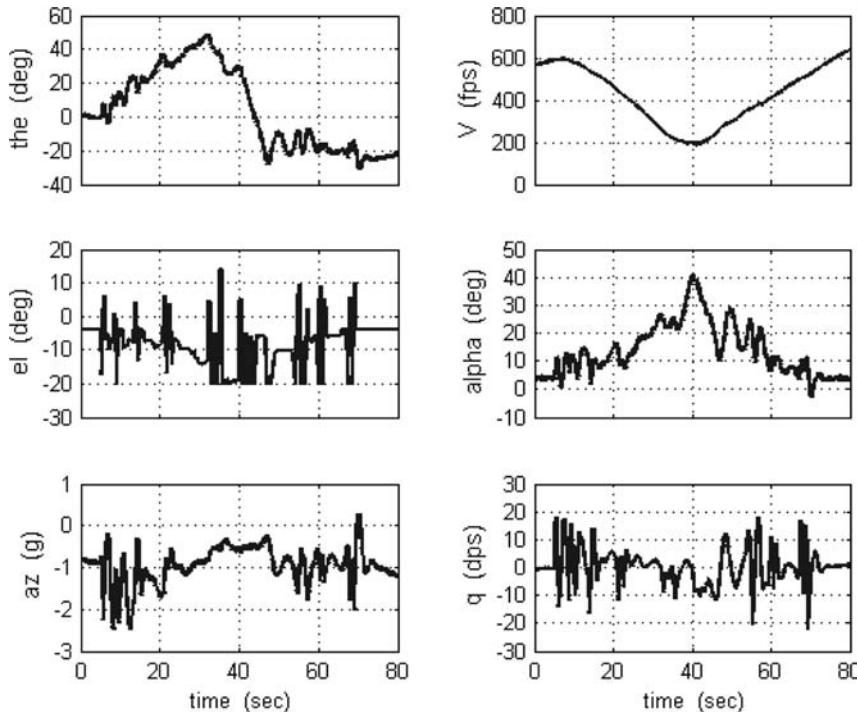


Fig. 5.17 Input and output variables for a large-amplitude longitudinal maneuver.

maneuvers, the signal amplitudes would be much lower, while the noise levels would remain approximately the same, resulting in lower signal-to-noise ratios.

The  $C_Z$  coefficient shown in Fig. 5.19 was modeled using the stepwise regression method described earlier. Table 5.4 shows parameter estimates and statistical metrics for each step in the model structure determination. At the start, partial correlations  $r$  of each regressor with the response variable  $C_Z$  are computed. As each regressor is brought into the model, a value for the corresponding parameter estimate and  $F_0$  statistic is computed.

At step 2, the linear  $C_{Zq}$  term is chosen rather than the nonlinear  $C_{Zq^2}$  term, although the partial correlation for the latter term is slightly higher. In borderline cases like this, the linear term should generally be included first in the model, so that the nonlinear terms are used only to model variations that cannot be characterized with linear terms. This corresponds to adding terms of increasing complexity to the Taylor series after the simpler linear terms are already in the model. Including the linear terms first, and using the nonlinear terms to model the remaining variation, has been called modified stepwise regression.

In the software implementation of the stepwise regression method that accompanies this text, regressors are moved in and out of the model via manual instructions from the analyst. This makes it possible to effectively handle borderline cases such as the one demonstrated in step 2 of Table 5.4, and to easily implement

## REGRESSION METHODS

153

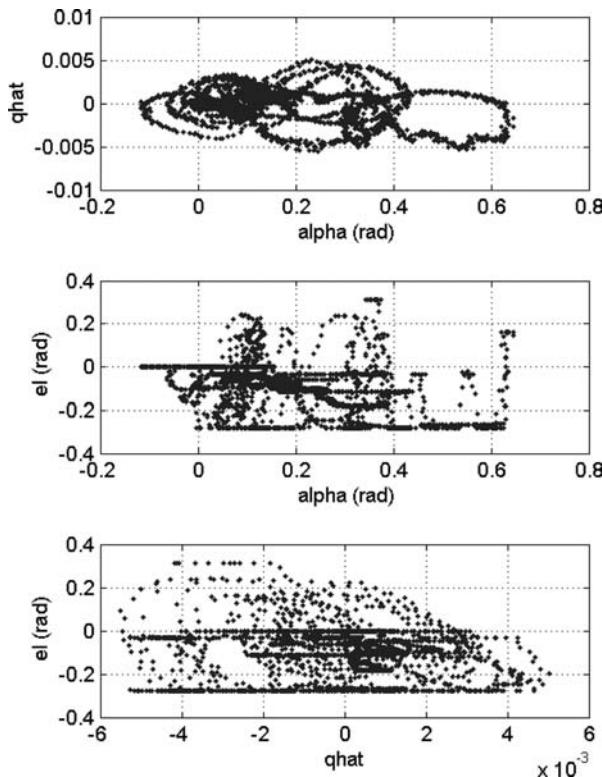


Fig. 5.18 Independent variable cross plots for a large-amplitude longitudinal maneuver.

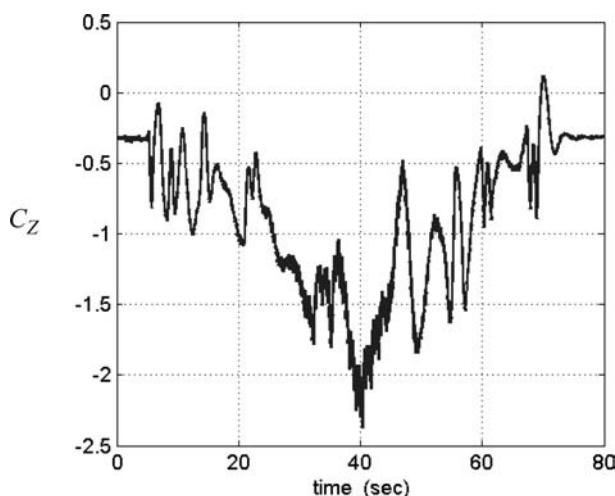


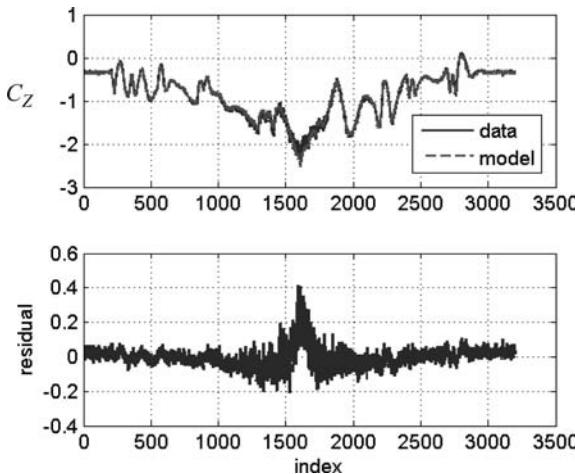
Fig. 5.19 Body-axis Z force coefficient for a large-amplitude longitudinal maneuver.

Table 5.4 Stepwise regression results, body-axis Z force coefficient for a large-amplitude longitudinal maneuver

Step	1		2		3		4		5	
	$\hat{\theta}$	$F_0$	$r$	$\hat{\theta}$	$F_0$	$r$	$\hat{\theta}$	$F_0$	$r$	$\hat{\theta}$
$C_{ZM}$	0	0.725	0	0.020	0	0.000	0	0.100	0	0.008
$C_{Z\alpha}$	0	0.973	-3.13	1.2e5	-3.21	1.6e5	-3.35	2.0e5	-4.17	7.0e4
$C_{Z\dot{\alpha}}$	0	0.031	0	0.305	-27.0	1.4e3	-32.3	2.8e3	-28.1	4.0e3
$C_{Z\delta_e}$	0	0.166	0	0.114	0	0.322	-0.42	1.5e3	-0.43	3.2e3
$C_{Z\alpha^2}$	0	0.838	0	0.306	0	0.303	0	0.485	1.34	3.0e3
PRESS	852.0	23.0			16.0		10.9			5.6
PSE	0.266	0.00735			0.00524		0.00372			0.00216
$R^2$ , %	0.0	97.3			98.1		98.7			99.3

## REGRESSION METHODS

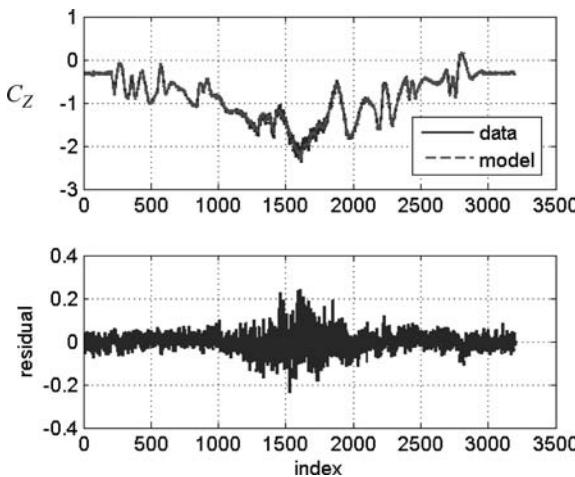
155



**Fig. 5.20** Vertical force coefficient modeling results after stepwise regression step 4 in Table 5.4.

modified stepwise regression, if desired. The analyst is therefore involved in each decision, and can apply specialized knowledge of the physical situation or experiment to the modeling problem.

Figure 5.20 shows the model fit and residual at step 4, where the model includes the strong linear terms in  $\alpha$ ,  $q\bar{c}/2V$ , and  $\delta_e$ , but not the weak linear term in Mach number,  $M$ . The residual in the lower plot of Fig. 5.20 clearly indicates some remaining deterministic content. The residual becomes random after adding the nonlinear  $\alpha^2$  term in step 5, see Fig. 5.21.



**Fig. 5.21** Vertical force coefficient modeling results after stepwise regression step 5 in Table 5.4.

Statistical modeling metrics are shown at the bottom of Table 5.4 for each step in the model structure determination. As discussed earlier, the  $R^2$  metric jumps to a large value after the initial regressor, then increases in small increments thereafter. The final value of  $R^2$  is 99.3%, which is indicative of a good model. Both the PRESS and PSE metrics decreased for each added regressor, and the corresponding  $F_0$  values were very high, indicating that each added regressor should be retained in the model.

The partial correlation for the Mach number regressor was never the highest among the candidate regressors not yet in the model, so Mach number was not selected for inclusion in the model. If the Mach number regressor is added at step 5 (not shown in Table 5.4), the PSE increases, and  $R^2$  remains essentially unchanged, but the PRESS metric decreases. The value of  $F_0$  for the Mach number regressor in the model is around 26, which is higher than  $F_{in}$ , but much lower than  $F_0$  for the other regressors. These mixed messages from the statistical metrics are not uncommon but can usually be resolved by using a majority consensus of the various metrics, as in this case.

Typically, there are more candidate regressors in the candidate pool than shown in this example. However, given a larger pool of candidate regressors, the stepwise regression can be used to select an adequate model in a manner similar to that shown here. Sometimes there is some trial and error involved in assembling a good pool of candidate regressors. The stepwise regression method can only select regressors available in the candidate pool, so the modeling functions needed for an adequate model must be present in the candidate pool.

One method for identifying a good set of candidate regressors is to use the orthogonal function modeling technique described earlier. This technique considers all multivariate polynomial combinations of the independent variables (up to a selected maximum order), then orthogonalizes those functions and ranks them according to their ability to reduce the mean square fit error. The ranked list of orthogonal modeling functions provides a good set of candidate regressors for stepwise regression.

Note also from Table 5.4 that the parameter estimates for terms already in the model change as each new regressor is added. This is because the regressors are not mutually orthogonal. If the stepwise regression were done using orthogonal modeling functions, each model parameter estimate would remain unchanged as other orthogonal modeling terms were swapped in and out of the model.

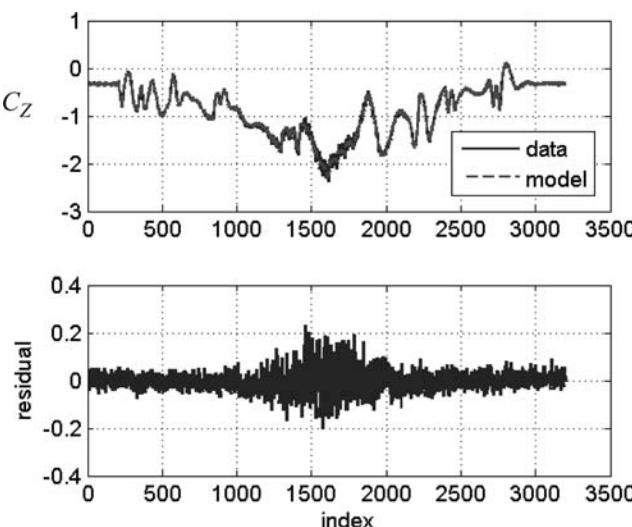
Table 5.5 contains the final parameter estimation results obtained from the stepwise regression, including standard errors corrected for colored residuals. The model parameters are estimated accurately, with large  $t_0$  statistics, and the model has a high  $R^2$  value. This example shows that a simple nonlinear term augmentation to a linear model can be used to effectively model data from a large-amplitude maneuver.

The nonlinearity highlighted in Fig. 5.20 can be characterized using other modeling functions, such as a first-order spline. Assuming a first-order spline is to be used, there is still the question of where the knot should be located. Stepwise regression can be used to help the analyst make this decision. If the pool of candidate regressors includes first-order splines with various knot locations, the best one will be indicated during the stepwise regression by the highest partial correlation after all significant linear terms are in the model.

**Table 5.5 Stepwise regression parameter estimation results, body-axis Z aerodynamic force coefficient, large-amplitude longitudinal maneuver**

Parameter	$\hat{\theta}$	$s(\hat{\theta})$	$ t_0 $	$100[s(\hat{\theta})/ \hat{\theta} ]$
$C_{Z_M}$	0	—	—	—
$C_{Z_\alpha}$	-4.166	0.0497	83.9	1.2
$C_{Z_q}$	-28.13	0.9822	28.6	3.5
$C_{Z_{\delta_e}}$	-0.4333	0.0123	35.3	2.8
$C_{Z_{\alpha^2}}$	1.344	0.0823	16.3	6.1
$C_{Z_o}$	-0.3229	0.0034	95.0	1.1
$s = \hat{\sigma}$	0.0418	—	—	—
$R^2, \%$	99.3	—	—	—

Figure 5.22 shows the model fit to  $C_Z$  for this case, where a first-order spline term in angle of attack with knot location at 25 deg was selected using stepwise regression. The associated parameter estimation results appear in Table 5.5. Comparison of these results with Fig. 5.21 and Table 5.5 demonstrate that in this case the nonlinearity in  $C_Z$  can be modeled equally well with a nonlinear  $\alpha^2$  term or a first-order spline term. Note that parameter estimates for model terms common to the models in Tables 5.5 and 5.6 (e.g.,  $C_{Z_\alpha}$ ) are slightly different, because the regressors in the models are different and not mutually orthogonal. Because of this, the task of characterizing variations in the response variable is shared differently among the regressors in each model.  $\square$

**Fig. 5.22 Vertical force coefficient modeling results using a linear spline term in angle of attack to model the nonlinearity.**

**Table 5.6 Stepwise regression parameter estimation results, body-axis Z aerodynamic force coefficient, large-amplitude longitudinal maneuver**

Parameter	$\hat{\theta}$	$s(\hat{\theta})$	$ t_0 $	$100[s(\hat{\theta})/ \hat{\theta} ]$
$C_{Z_M}$	0	—	—	—
$C_{Z_\alpha}$	-3.537	0.0160	220.7	0.5
$C_{Z_q}$	-29.45	0.8241	35.7	2.8
$C_{Z_{\delta_e}}$	-0.4400	0.0113	38.8	2.6
$C_{Z_{[\alpha=25(\pi/180)]^+}}$	1.681	0.0732	23.0	4.4
$C_{Z_o}$	-0.3348	0.0028	119.7	0.8
$s = \hat{\sigma}$	0.0400	—	—	—
$R^2, \%$	99.4	—	—	—

## 5.5 Data Collinearity

If a regressor in a linear regression model (5.6) is equal to a linear combination of one or more of the other regressors, then all the involved regressors are said to be linearly dependent. In that case, the parameter estimation routine cannot assign specific values to the parameters for those terms, because many different weighted combinations of the linearly dependent regressors could be used equally well to model the same variation in the dependent variable. This makes the parameter estimation problem ill-conditioned. In practice, the regressors might be almost linearly dependent, but not exactly so. But the fundamental difficulty remains the same, with its severity becoming worse as regressors get closer to being perfectly correlated, i.e., as  $|r_{jk}|$ ,  $j, k \in (1, 2, \dots, n)$ , from Eq. (5.77) approaches 1. Any situation where regressors are correlated at a high enough level to cause problems in the parameter estimation is called data collinearity. Operationally, when data collinearity is present, the parameter estimation routines will produce inaccurate parameter estimates with large variances, or in severe cases the parameter estimation routine may fail.

This section discusses methods for assessing data collinearity and some approaches for getting good parameter estimation results when data collinearity exists. For the following discussion and analysis, it is convenient to use standardized regressors introduced in Sec. 5.1.5.

As shown in Sec. 5.1.5, the matrix  $X^{*T}X^*$  is the  $n \times n$  matrix of pair-wise regressor correlation coefficients. Denoting  $\xi_j^*$ ,  $j = 1, 2, \dots, n$ , as the vectors of centered and scaled regressors that are the columns of  $X^*$ , then  $X^* = [\xi_1^* \ \xi_2^* \ \dots \ \xi_n^*]$ . If  $\xi_j^{*T}\xi_k^* = 0$ ,  $j \neq k$ , the regressors are orthogonal, and the  $X^{*T}X^*$  matrix is a diagonal matrix equal to the identity matrix. The vectors  $\xi_1^*, \xi_2^*, \dots, \xi_n^*$  are linearly dependent if there is a set of constants  $c_j$ , not all zero, such that

$$\sum_{j=1}^n c_j \xi_j^* = \mathbf{0} \quad (5.152)$$

In that case, the rank of  $X^{*T}X^*$  is less than  $n$ ,  $\det(X^{*T}X^*) = 0$ , and  $(X^{*T}X^*)^{-1}$  does not exist.

In many applications of linear regression, Eq. (5.152) is only approximately true. This indicates near-linear dependency among the columns of  $X^*$ , and the problem of data collinearity exists. In such a case, the  $X^{*T}X^*$  matrix is called ill conditioned. This causes computational problems, and reduces the accuracy of the parameter estimates, because of the use of the matrix  $(X^{*T}X^*)^{-1}$  in both the parameter estimation and estimated parameter covariance calculations [cf. Eq. (5.75)].

Collinearity is a data problem, and is not specific to any particular data analysis or modeling method. There are at least three different sources of collinearity: 1) design of the experiment, 2) constraints, and 3) model specification.

If the experiment is conducted in a manner such that the data for two or more of the regressors are mostly changed proportionally, then collinearity can occur. The problem can also arise when the changes in the regressors are not sufficient to excite the response variable above the noise level. The key point is that the experiment must involve significant, independent changes in any regressor for which an accurate, independent estimate of influence (i.e., parameter estimate) is sought (see Chapter 9).

Constraints in the data could be caused by an inherent property or mode of operation for the system being tested. For example, an aircraft control system can deflect various control surfaces in proportion to measured body-axis angular rates, for stability augmentation, or can deflect several control surfaces proportionally for improved control authority. This causes near-linear dependency among the regressors, and data collinearity occurs. The regressors are never perfectly correlated for the aircraft problem, because the control system has some delay between measuring the angular rates and implementing the control surface deflections, and there is also some noise on all the measured signals, which is different for each signal. However, the absolute value of pairwise regressor correlations can exceed 0.9 for highly augmented aircraft such as jet fighters.

Collinearity can also be caused by specifying a model with terms that are not warranted. For example, if  $\xi_1^*$  is small, then the regressors  $(\xi_1^*)^2$  and  $\xi_1^*\xi_2^*$  might have very little influence on the response variable, since these regressors will be close to zero. This shows up as a data collinearity, where  $\xi_1^2$  and  $\xi_1\xi_2$  are roughly the same regressor, namely a zero vector.

### 5.5.1 *Detection and Assessment of Data Collinearity*

Many procedures have been developed to detect collinearity. Some of these are discussed in Refs. 3 and 22. In this section, only three methods will be considered: 1) examination of the regressor correlation matrix and its inverse, 2) eigensystem analysis or singular value decomposition, and 3) parameter variance decomposition. The use of each method will be demonstrated in an example.

*Examination of the regressor correlation matrix.* The simplest and most straightforward procedure for assessing collinearity is to examine the regressor correlation matrix. A high correlation coefficient between two regressors can point to a possible collinearity problem. A good practical rule of thumb is that collinearity might cause problems in the parameter estimation for any pairwise regressor correlation with absolute value greater than 0.9.

The absence of high pair-wise correlation, however, cannot be viewed as a guarantee of no problem. The correlation matrix is unable to reveal the presence of correlation among more than two regressors. This was demonstrated by Goldberg.<sup>10</sup>

The aforementioned shortcoming in using the regressor correlation matrix  $\mathbf{X}^* \mathbf{X}^*$  as a diagnostic measure of collinearity limits the usefulness of its inverse in a similar way. The diagonal elements of  $(\mathbf{X}^* \mathbf{X}^*)^{-1}$  are often called the variance inflation factors, VIF, which can be expressed as

$$\text{VIF}_j = \frac{1}{1 - R_j^2} \quad (5.153)$$

for the  $j$ th regressor, where  $R_j^2$  is the coefficient of multiple determination for the regressor  $\xi_j$  modeled as a linear function of the other regressors in the model.<sup>23</sup> The quantity  $\text{VIF}_j$  is the  $j$ th diagonal element of  $(\mathbf{X}^{*T} \mathbf{X}^*)^{-1}$ . The name “variance inflation factor” comes from a relationship with the  $j$ th parameter variance. As shown in Ref. 23, this relationship is

$$\text{Var}(\theta_j) = \frac{\sigma^2}{\xi_j^{*T} \xi_j^*} \text{VIF}_j \quad (5.154)$$

The diagnostic value of  $\text{VIF}_j$  follows from Eq. (5.153). For data with no collinearity at all (i.e., orthogonal regressors), the variance inflation factors  $\text{VIF}_j$  are equal to 1, since  $R_j^2 = 0$  for every  $j$ . Large values of  $\text{VIF}_j$  indicate an  $R_j^2$  near unity, which points to collinearity. The weakness of this diagnostic measure is its inability to distinguish among several coexisting near-dependencies, and the lack of a specific value for  $\text{VIF}_j$  that can be considered an acceptable upper bound for low data collinearity. According to Myers,<sup>2</sup> a  $\text{VIF}_j$  larger than 10 indicates a serious problem with collinearity.

*Eigensystem analysis and singular value decomposition.* The matrix  $\mathbf{X}^T \mathbf{X}$  can be decomposed as follows:

$$\mathbf{X}^T \mathbf{X} = \mathbf{T} \Lambda \mathbf{T}^T \quad (5.155)$$

where  $\Lambda$  is an  $n \times n$  diagonal matrix whose diagonal elements are the eigenvalues  $\lambda_j$ ,  $j = 1, 2, \dots, n$ , of  $\mathbf{X}^T \mathbf{X}$ , and  $\mathbf{T}$  is an  $n \times n$  orthonormal matrix whose columns are the eigenvectors of  $\mathbf{X}^T \mathbf{X}$ . The  $N \times n$  matrix  $\mathbf{X}$  considered here can be scaled and/or centered. For diagnostic purposes, Belsley et al.<sup>22</sup> recommend scaling the columns of  $\mathbf{X}$  to unit length. The matrix  $\mathbf{X}$  should not be centered if the role of the intercept term in near-linear dependencies is being investigated.

One or more small eigenvalues imply that there are near-linear dependencies among the columns of  $\mathbf{X}$ . The severity of these dependencies is indicated by how small the eigenvalues are, compared with the maximum eigenvalue. Near-linear

## REGRESSION METHODS

161

dependencies among the columns of the data matrix  $X$  can be measured by the condition index

$$\frac{\lambda_{\max}}{\lambda_j} \geq 1 \quad j = 1, 2, \dots, n \quad (5.156)$$

The largest value  $\lambda_{\max}/\lambda_{\min}$  is known as the condition number of the  $X^T X$  matrix. According to Ref. 22, any condition index from Eq. (5.156) in the range of 100–1000 indicates moderate to strong collinearity. However, experience from the analysis of flight data using linear regression has revealed that in some cases the parameter estimates could be affected by data collinearity even if the condition number is less than 100.

Belsley et al.<sup>22</sup> recommend an approach using singular-value decomposition for diagnosing collinearity. This approach is based on a stable numerical decomposition of the matrix  $X$  as

$$X = U D T^T \quad (5.157)$$

where  $U$  is an  $N \times n$  matrix with mutually orthonormal columns, and  $T$  is an  $n \times n$  matrix with mutually orthonormal columns, so that  $U^T U = T^T T = I$ . The matrix  $D$  is an  $n \times n$  diagonal matrix with nonnegative elements on the diagonal,  $\mu_j$ ,  $j = 1, 2, \dots, n$ , which are called the singular values of  $X$ . The singular-value decomposition, SVD, is closely related to the concept of eigenvalues and eigenvectors, since from Eqs. (5.155) and (5.157),

$$X^T X = T D^2 T^T = T \Lambda T^T \quad (5.158)$$

The diagonal elements of  $D^2$  are therefore the eigenvalues of  $X^T X$  and the columns of  $T$  in Eq. (5.157) are the eigenvectors of  $X^T X$  associated with the  $n$  nonzero eigenvalues. The severity of ill-conditioning in the  $X^T X$  matrix is indicated by how small the singular values are relative to the maximum singular value. Therefore, a condition index for the matrix  $X$  is proposed as

$$\frac{\mu_{\max}}{\mu_j} = \sqrt{\frac{\lambda_{\max}}{\lambda_j}} \quad j = 1, 2, \dots, n \quad (5.159)$$

The largest value  $\mu_{\max}/\mu_{\min}$  is the condition number of the  $X$  matrix.

The SVD of the matrix  $X$  provides similar information to that given by the eigensystem of  $X^T X$ . However, the SVD is generally preferred, because of greater numerical stability in its computing algorithm.

**Parameter variance decomposition.** An approach using parameter variance decomposition for detecting collinearity was proposed in Ref. 22.

It follows from the covariance matrix of parameter estimates  $\hat{\boldsymbol{\theta}}$ , which is obtained as

$$\text{Cov}(\hat{\boldsymbol{\theta}}) = \sigma^2 (\mathbf{X}^T \mathbf{X})^{-1} = \sigma^2 \mathbf{T} \Lambda^{-1} \mathbf{T}^T \quad (5.160)$$

where the last equality uses the eigensystem decomposition from Eq. (5.158). The variance of each parameter is then equal to

$$\text{Var}(\hat{\theta}_k) = \sigma^2 \sum_{j=1}^n \frac{t_{kj}^2}{\lambda_j} = \sigma^2 \sum_{j=1}^n \frac{t_{kj}^2}{\mu_j^2} \quad (5.161)$$

where  $t_{kj}$  is the  $k$ th element of the  $j$ th eigenvector associated with  $\lambda_j$ , and the  $j$ th eigenvector is the  $j$ th column of  $\mathbf{T}$ . The expression (5.161) shows that the variance for each estimated parameter can be decomposed into a sum of components, each corresponding to one of the  $n$  eigenvalues  $\lambda_j, j = 1, 2, \dots, n$ . For each term in the summation of Eq. (5.161), the eigenvalue appears in the denominator so that one or more small eigenvalues can substantially increase the variance of  $\hat{\theta}_k$ . A high proportion of the variance for two or more estimated parameters from the same small eigenvalue can provide evidence that the corresponding near-dependency is causing problems. Introducing

$$\phi_{kj} \equiv \frac{t_{kj}}{\lambda_j} \quad \text{and} \quad \phi_k \equiv \sum_{j=1}^n \frac{t_{kj}}{\lambda_j} = \sum_{j=1}^n \phi_{kj} \quad (5.162)$$

The  $k, j$  variance proportion  $\pi_{kj}$  is defined as the proportion of the variance of the  $k$ th estimated parameter, associated with the  $j$ th component of the decomposition in Eq. (5.161),

$$\pi_{kj} \equiv \frac{\phi_{kj}}{\phi_k} \quad j = 1, 2, \dots, n \quad (5.163)$$

Since two or more regressors are required to create near-dependency, two or more variances will be adversely affected by high variance proportions associated with each small singular value. Variance proportions greater than 0.5 are recommended guidance for possible collinearity problems. The variance proportions complement the other diagnostics in assessing the effect of data collinearity on the estimated parameter variances.

The total diagnosis for data collinearity should involve the condition indices to assess the severity of a particular dependency, the variance proportions to indicate which regressors are involved in the dependency and to what extent, and the VIF to aid in determining the damage to individual parameter estimates and variances.

### 5.5.2 Adverse Effects of Data Collinearity

The presence of collinearity in the data for a linear regression problem results in various unwanted properties of the least-squares parameter estimates and variances. The increased variance in the estimates is apparent from Eqs. (5.153), (5.154), and (5.161). Collinearity also tends to produce least-squares estimates that are large in absolute value. This can be demonstrated using Eq. (4.11) for the mean squared error in the vector  $\hat{\boldsymbol{\theta}}$  with no bias error from model structure misspecification,

$$\text{MSE}(\hat{\boldsymbol{\theta}}) = E[(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})^T(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})] \quad (5.164)$$

which can also be expressed as

$$\begin{aligned} \text{MSE}(\hat{\boldsymbol{\theta}}) &= \sum_{j=1}^n E[(\hat{\theta}_j - \theta_j)^2] \\ &= \sigma^2 \text{Tr}[(\mathbf{X}^T \mathbf{X})^{-1}] = \sigma^2 \text{Tr}[(\mathbf{T} \Lambda \mathbf{T}^T)^{-1}] \\ &= \sigma^2 \sum_{j=1}^n \lambda_j^{-1} \end{aligned} \quad (5.165)$$

Equation (5.165) shows that the distance from  $\hat{\boldsymbol{\theta}}$  to  $\boldsymbol{\theta}$  may be large if at least one of the  $\lambda_j$  is small.

Belsley et al.<sup>22</sup> address the problem of decreased accuracy of parameters estimated from collinear data, in connection with insufficient information in the data. For this case, the linear regression model can be formulated as

$$\begin{aligned} z &= \mathbf{X}\boldsymbol{\theta} + \boldsymbol{\nu} \\ &= \mathbf{X}_1\boldsymbol{\theta}_1 + \mathbf{X}_2\boldsymbol{\theta}_2 + \boldsymbol{\nu} \end{aligned} \quad (5.166)$$

where  $\mathbf{X}$  has been partitioned into two matrices  $\mathbf{X}_1$  and  $\mathbf{X}_2$ , with dimensions  $N \times n_1$  and  $N \times n_2$ , respectively. The matrix  $\mathbf{X}_2$  contains regressors involved in  $n_2$  near-dependencies with regressors in  $\mathbf{X}_1$ . Modeling the regressors in  $\mathbf{X}_2$  as a linear combination of the regressors in  $\mathbf{X}_1$  gives

$$\mathbf{X}_2 = \mathbf{X}_1\mathbf{C} + \mathbf{V}_2 \quad (5.167)$$

where  $\mathbf{C}$  is a matrix of parameters with each column containing the parameters for the model of the corresponding column of  $\mathbf{X}_2$  in terms of  $\mathbf{X}_1$ , and  $\mathbf{V}_2$  is the associated matrix of residuals. Substituting Eq. (5.167) into (5.166) results in the regression model

$$z = \mathbf{X}_1(\boldsymbol{\theta}_1 + \mathbf{C}\boldsymbol{\theta}_2) + \mathbf{V}_2\boldsymbol{\theta}_2 + \boldsymbol{\nu} \quad (5.168)$$

Since  $\mathbf{X}_1^T \mathbf{V}_2 = \mathbf{V}_2^T \mathbf{X}_1 = \mathbf{0}$  from the normal equations associated with Eq. (5.167), it is possible to estimate  $\boldsymbol{\theta}_1 + \mathbf{C}\boldsymbol{\theta}_2$  and  $\boldsymbol{\theta}_2$  using separate regressions of  $z$  as a linear function of  $\mathbf{X}_1$  and  $\mathbf{V}_2$ , respectively. Further, because of the near-linear

dependency between  $X_1$  and  $X_2$ ,  $\det(V_2^T V_2) \ll \det(X_1^T X_1)$ , which means that most of the information in the data will be applied to the estimation of  $\boldsymbol{\theta}_1 + \mathbf{C} \boldsymbol{\theta}_2$ , and only a small amount will be available for the estimation of  $\boldsymbol{\theta}_2$ . Thus, collinearity can create a situation where there is insufficient information in the data for highly accurate estimation of all model parameters.

### 5.5.3 Mixed Estimator

As discussed previously, the ordinary least-squares technique provides an unbiased linear estimator that has minimum variance in the class of unbiased linear estimators. However, there is no guarantee that the variance will be small. It was seen in the last section that the application of ordinary least squares to a set of data with collinearity problems can result in large estimated parameters with large variances. Figure 5.14 showed that biased parameter estimates can have smaller variance than unbiased parameter estimates, so that for small bias errors, the mean squared error for biased parameter estimates can be smaller than the variance of parameter estimates from an unbiased estimator. This possibility has inspired the development of various biased parameter estimation techniques. Some of these techniques are reviewed in Ref. 13. In the following, only one biased parameter estimation technique, the mixed estimator, will be described and applied to experimental data.

The mixed estimator is developed in Ref. 9 as a Bayes-like technique that augments the measured data with prior information about the parameters. For the linear regression model

$$\begin{aligned} z &= \mathbf{X}\boldsymbol{\theta} + \mathbf{v} \\ E(\mathbf{v}) &= \mathbf{0} \quad \text{and} \quad E(\mathbf{v}\mathbf{v}^T) = \sigma^2 \mathbf{I} \end{aligned} \quad (5.169)$$

it is assumed that  $m < n_p$  prior constraints on the elements of  $\boldsymbol{\theta}$  are available. These constraints are formulated as

$$\mathbf{d} = \mathbf{B}\boldsymbol{\theta} + \mathbf{s} \quad (5.170)$$

In Eq. (5.170),  $\mathbf{B}$  is an  $m \times n_p$  matrix with known constant elements,  $\mathbf{d}$  is an  $m \times 1$  vector of values that can be specified, and  $\mathbf{s}$  is a random vector with

$$E(\mathbf{s}) = \mathbf{0} \quad E(\mathbf{v}\mathbf{s}^T) = \mathbf{0} \quad E(\mathbf{s}\mathbf{s}^T) = \sigma^2 \mathbf{V} \quad (5.171)$$

where  $\mathbf{V}$  is a known matrix.

Combining Eqs. (5.169–5.171), the mixed model is given as

$$\begin{aligned} \begin{bmatrix} z \\ \mathbf{d} \end{bmatrix} &= \begin{bmatrix} \mathbf{X} \\ \mathbf{B} \end{bmatrix} \boldsymbol{\theta} + \begin{bmatrix} \mathbf{v} \\ \mathbf{s} \end{bmatrix} \\ E\left\{\begin{bmatrix} \mathbf{v} \\ \mathbf{s} \end{bmatrix}\right\} &= \mathbf{0} \quad \text{and} \quad E\left\{\begin{bmatrix} \mathbf{v} \\ \mathbf{s} \end{bmatrix} [\mathbf{v}^T \quad \mathbf{s}^T]\right\} = \sigma^2 \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{V} \end{bmatrix} \end{aligned} \quad (5.172)$$

## REGRESSION METHODS

165

The least-squares cost function to be minimized is

$$\begin{aligned} J_{\text{ME}} &= \frac{1}{2} \begin{bmatrix} (z - X\theta)^T \\ (d - B\theta)^T \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & V \end{bmatrix}^{-1} \begin{bmatrix} (z - X\theta) \\ (d - B\theta) \end{bmatrix} \\ &= \frac{1}{2}(z - X\theta)^T(z - X\theta) + \frac{1}{2}(d - B\theta)^T V^{-1}(d - B\theta) \\ &= \frac{1}{2}(z - X\theta)^T(z - X\theta) + \frac{1}{2}(\theta - B^{-1}d)^T B^T V^{-1} B (\theta - B^{-1}d) \quad (5.173) \end{aligned}$$

which is the form of the cost function for the Bayesian estimator [cf. Eq. (4.28) with  $\theta_p = B^{-1}d$  and  $\Sigma_p^{-1} = B^T V^{-1} B$ ]. Applying ordinary least-squares parameter estimation to the cost function in Eq. (5.173) results in the mixed estimator

$$\hat{\theta}_{\text{ME}} = (X^T X + B^T V^{-1} B)^{-1} (X^T z + B^T V^{-1} d) \quad (5.174)$$

with covariance matrix

$$\text{Cov}(\hat{\theta}_{\text{ME}}) = \sigma^2 (X^T X + B^T V^{-1} B)^{-1} \quad (5.175)$$

where  $\sigma^2$  can be estimated using the methods described earlier.

Introducing the augmented variables,

$$z_a \equiv \begin{bmatrix} z \\ d \end{bmatrix} \quad X_a \equiv \begin{bmatrix} X \\ B \end{bmatrix} \quad v_a \equiv \begin{bmatrix} v \\ s \end{bmatrix} \quad (5.176)$$

the mixed model can also be written as

$$\begin{aligned} z_a &= X_a \theta + v_a \\ E(v_a) &= \mathbf{0} \quad \text{and} \quad E(v_a v_a^T) = \sigma^2 \begin{bmatrix} I & \mathbf{0} \\ \mathbf{0} & V \end{bmatrix} \equiv \sigma^2 V_a \quad (5.177) \end{aligned}$$

For the model given in Eq. (5.177), the mixed estimator can be expressed in the form

$$\hat{\theta}_{\text{ME}} = (X_a^T V_a^{-1} X_a)^{-1} X_a^T V_a^{-1} z_a \quad (5.178)$$

with covariance matrix

$$\text{Cov}(\hat{\theta}_{\text{ME}}) = \sigma^2 (X_a^T V_a^{-1} X_a)^{-1} \quad (5.179)$$

The quantity  $\hat{\theta}_{\text{ME}}$  computed from Eq. (5.174) or (5.178) is an unbiased linear estimator of  $\theta$ , because the constraint equations (5.170) were assumed to hold exactly.

In practical applications of mixed estimation, the *a priori* constraints on the parameters are usually not known exactly. In this case, Eq. (5.170) is changed to

$$\mathbf{d} = \mathbf{B}\boldsymbol{\theta} + \mathbf{b} + \mathbf{s} \quad (5.180)$$

where  $\mathbf{b}$  is an unknown vector. The mixed estimate associated with the condition (5.180) will be denoted  $\tilde{\boldsymbol{\theta}}_{ME}$ . The expected value of  $\tilde{\boldsymbol{\theta}}_{ME}$  is obtained by substituting from Eqs. (5.169) and (5.180) into Eq. (5.174),

$$\begin{aligned} E(\tilde{\boldsymbol{\theta}}_{ME}) &= (X^T X + \mathbf{B}^T \mathbf{V}^{-1} \mathbf{B})^{-1} E(X^T X \boldsymbol{\theta} + X^T \mathbf{v} + \mathbf{B}^T \mathbf{V}^{-1} \mathbf{B} \boldsymbol{\theta} \\ &\quad + \mathbf{B}^T \mathbf{V}^{-1} \mathbf{b} + \mathbf{B}^T \mathbf{V}^{-1} \mathbf{s}) \\ &= (X^T X + \mathbf{B}^T \mathbf{V}^{-1} \mathbf{B})^{-1} \{ (X^T X + \mathbf{B}^T \mathbf{V}^{-1} \mathbf{B}) E(\boldsymbol{\theta}) + X^T E(\mathbf{v}) \\ &\quad + \mathbf{B}^T \mathbf{V}^{-1} \mathbf{b} + \mathbf{B}^T \mathbf{V}^{-1} E(\mathbf{s}) \} \\ &= \boldsymbol{\theta} + \mathbf{M}^{-1} \mathbf{B}^T \mathbf{V}^{-1} \mathbf{b} \end{aligned} \quad (5.181)$$

where

$$\mathbf{M} \equiv X^T X + \mathbf{B}^T \mathbf{V}^{-1} \mathbf{B} \quad (5.182)$$

From Eq. (5.181), the estimate  $\tilde{\boldsymbol{\theta}}_{ME}$  is biased by the quantity  $\mathbf{M}^{-1} \mathbf{B}^T \mathbf{V}^{-1} \mathbf{b}$ . The covariance matrix for the mixed estimator  $\tilde{\boldsymbol{\theta}}_{ME}$  is

$$\text{Cov}(\tilde{\boldsymbol{\theta}}_{ME}) = \sigma^2 (X^T X + \mathbf{B}^T \mathbf{V}^{-1} \mathbf{B})^{-1} = \sigma^2 \mathbf{M}^{-1} \quad (5.183)$$

The difference between the covariance of the ordinary least-squares estimator and the mixed estimator is

$$\text{Cov}(\hat{\boldsymbol{\theta}}) - \text{Cov}(\tilde{\boldsymbol{\theta}}_{ME}) = \sigma^2 (X^T X)^{-1} - \sigma^2 (X^T X + \mathbf{B}^T \mathbf{V}^{-1} \mathbf{B})^{-1} \quad (5.184)$$

As shown in Ref. 9, the right side of Eq. (5.184) is a nonnegative definite matrix. Adding *a priori* information to ordinary least-squares regression therefore results in a reduction in the estimated parameter variances, compared with the ordinary least squares. Considering the mean squared error of the estimated parameter vector as a measure of the accuracy of an estimator, then for the mixed estimator,

$$\text{MSE}(\tilde{\boldsymbol{\theta}}_{ME}) = \sigma^2 \text{Tr}(\mathbf{M}^{-1}) + (\boldsymbol{\theta} - \mathbf{M}^{-1} \mathbf{B}^T \mathbf{V}^{-1} \mathbf{b})^T (\boldsymbol{\theta} - \mathbf{M}^{-1} \mathbf{B}^T \mathbf{V}^{-1} \mathbf{b}) \quad (5.185)$$

Thus, more accurate *a priori* values of  $\mathbf{d}$  (i.e., smaller  $\mathbf{b}$ ) will result in a more accurate estimate  $\tilde{\boldsymbol{\theta}}_{ME}$ .

## REGRESSION METHODS

167

The constraints on the parameters given by Eq. (5.170) can take several forms. The most common are the following:

1) A separate *a priori* estimate of  $\boldsymbol{\theta}$  exists. If this estimate is called  $\boldsymbol{\theta}_p$ , with covariance matrix  $\boldsymbol{\Sigma}_p$ , then Eq. (5.170) becomes

$$\boldsymbol{\theta}_p = \boldsymbol{\theta} + \boldsymbol{s} \quad (5.186a)$$

and the expression for the mixed estimator is

$$\tilde{\boldsymbol{\theta}}_{ME} = (\mathbf{X}^T \mathbf{X} + \boldsymbol{\Sigma}_p^{-1})^{-1} (\mathbf{X}^T \mathbf{z} + \boldsymbol{\Sigma}_p^{-1} \boldsymbol{\theta}_p) \quad (5.186b)$$

with covariance matrix

$$\text{Cov}(\tilde{\boldsymbol{\theta}}_{ME}) = \sigma^2 (\mathbf{X}^T \mathbf{X} + \boldsymbol{\Sigma}_p^{-1})^{-1} \quad (5.186c)$$

Equations (5.186) can be used to estimate parameters and error bounds for combined data from the experiment that generated  $\mathbf{X}$  and  $\mathbf{z}$ , and a previous experiment, where information from the previous experiment is included through the *a priori* parameter estimate  $\boldsymbol{\theta}_p$  and covariance matrix  $\boldsymbol{\Sigma}_p$ . This idea can be used repeatedly to estimate parameters based on combined data from more than one flight maneuver and/or wind-tunnel test.

2) The *a priori* information is given as a statement that particular parameters lie in a certain range ( $d_{\min}, d_{\max}$ ). For a parameter  $\theta_j$ , the *a priori* value can be placed at the center of the range, with the appropriate uncertainty,

$$\theta_{pj} = \frac{1}{2}(d_{\min_j} + d_{\max_j}) = \theta_j + s_j \quad j = 1, 2, \dots, n_p \quad (5.187)$$

The situation is then identical to having *a priori* parameter estimates, as in the preceding case.

3) Setting  $\mathbf{d} = \mathbf{0}$ ,  $\mathbf{B} = \mathbf{I}$ , and  $E(\mathbf{ss}^T) = \sigma^2 \mathbf{V} = k_R^{-1} \mathbf{I}$ , the mixed estimator becomes a ridge estimator,

$$\tilde{\boldsymbol{\theta}}_R = (\mathbf{X}^T \mathbf{X} + k_R \mathbf{I})^{-1} \mathbf{X}^T \mathbf{z} \quad (5.188)$$

where  $k_R$  is a scalar biasing parameter. Methods for choosing  $k_R$  and an extension that allows separate biasing parameters for each regressor are described in Ref. 3. The ridge regression is frequently applied to collinear data when *a priori* values for the parameters are unavailable.

### Example 5.4

This example is based on the material in Ref. 24. The test vehicle is the X-29 aircraft, which is a single-engine, single-seat fighter-type research aircraft with forward-swept wings. Figure 5.23 is a photograph of the X-29 in flight.



Fig. 5.23 X-29 aircraft.

The aircraft has relaxed static longitudinal stability in subsonic and transonic regimes, and near-neutral stability in supersonic regimes. Because of this, the aircraft has a full-time automatic flight control system for stability augmentation. For longitudinal control, deflections of canard, wing flaps (flaperons), and fuselage strakes were used. Lateral control was provided by the rudder and asymmetric deflection of the flaperons.

In addition to pilot stick and rudder inputs, the aircraft responses could also be excited using the concept of a remotely augmented vehicle (RAV). The RAV arrangement employed a ground computer to augment commands from the onboard control system. This capability could be used to introduce a command to the pilot controls (pitch stick, roll stick, or rudder pedals), or to control surfaces (symmetric flaperons, strakes, canard, rudder, or asymmetric flaperons). The RAV commands, usually a pulse or doublet, were summed with the already existing commands to independently move pilot controls or control surfaces. A detailed description of the aircraft and its control system can be found in Ref. 25.

For this example, the following two sets of data were used to estimate aircraft parameters: 1) longitudinal maneuvers implemented by a pilot, at Mach numbers ranging from 0.5 to 1.4, and 2) longitudinal maneuvers implemented by computer-generated inputs using the RAV system, at Mach numbers from 0.6 to 1.3.

During the data analysis, several problems associated with the inherent instability, high augmentation, and sometimes insufficient excitation of the aircraft responses had to be addressed. The main issues were 1) parameter estimation for an aircraft that is open-loop unstable, 2) adverse effects of data collinearity on parameter identifiability and accuracy, and 3) insufficient information content in the data.

To address these problems, the RAV system was used in the experiment, and both linear regression and mixed estimation were used in the analysis.

## REGRESSION METHODS

169

Figures 5.24 and 5.25 show measurements of the pitch stick  $\eta$ , and longitudinal control surface deflections of the canard  $\delta_c$ , symmetric flaperons  $\delta_f$ , and strake  $\delta_s$ , along with the longitudinal response variables, for a piloted pitch stick input maneuver and a maneuver using a sequence of computer-generated RAV inputs.

For the pilot pitch stick input maneuver in Fig. 5.24, all control surface deflections have similar forms, suggesting that data collinearity exists. In general, regressor time series must be dissimilar in form for good parameter estimation, because when any regressor can be scaled to approximately match another, there

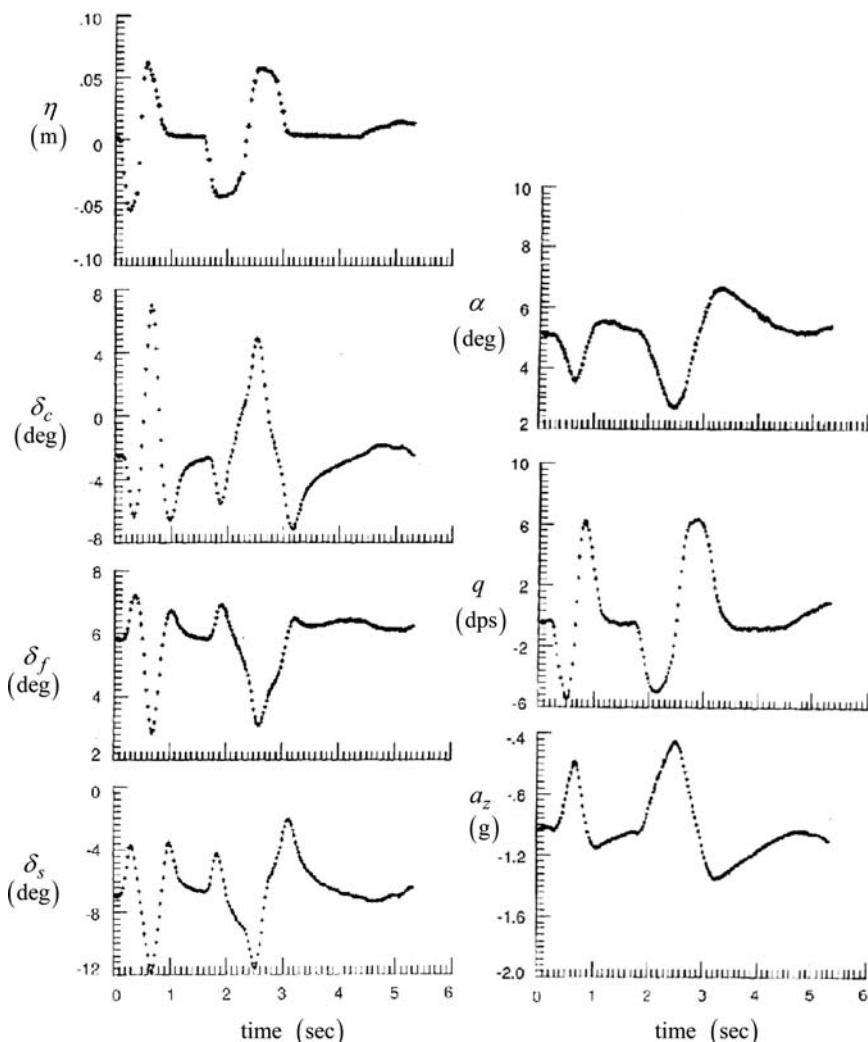


Fig. 5.24 Pilot pitch stick input maneuver for the X-29 aircraft.

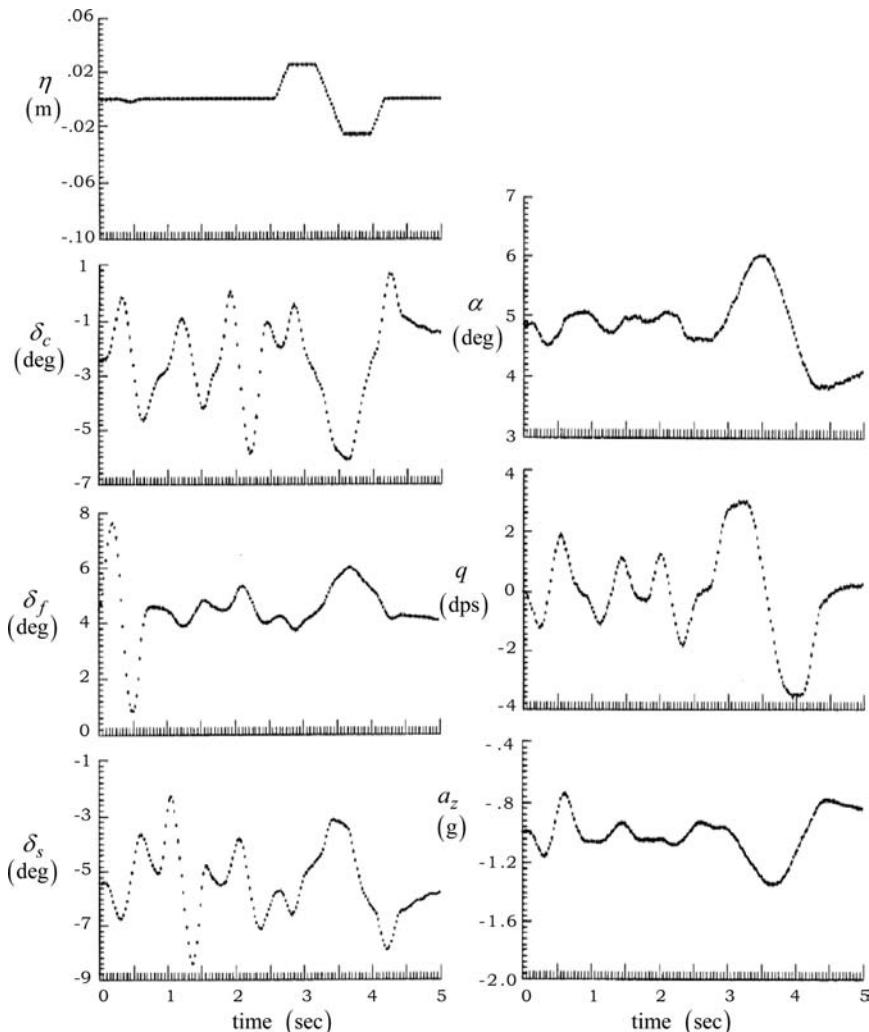


Fig. 5.25 RAV input sequence maneuver for the X-29 aircraft.

is an indeterminacy in how variations in the dependent variable can be modeled, and data collinearity exists.

Figure 5.25 demonstrates the change in data collinearity that results from replacing the pilot pitch stick input with a computer-generated sequence of doublets on the symmetric flaperons, strake, canard, and pitch stick. For both maneuvers, data collinearity was assessed by examining pair-wise correlations between regressors, and the corresponding parameter variance proportions.

## REGRESSION METHODS

171

**Table 5.7 Correlation matrix for the regressors, pilot pitch stick input**

	$\alpha$	$q\bar{c}/2V_o$	$\delta_s$	$\delta_f$	$\delta_c$
$\alpha$	1.000	0.151	0.753	0.711	-0.795
$q\bar{c}/2V_o$	—	1.000	0.318	-0.341	-0.980
$\delta_s$	—	—	1.000	0.643	-0.928
$\delta_f$	—	—	—	1.000	-0.844
$\delta_c$	—	—	—	—	1.000

The aerodynamic model equations for the vertical force coefficient  $C_Z$  and the pitching moment coefficient  $C_m$  were postulated as

$$C_Z = C_{Z_0} + C_{Z_\alpha} \Delta \alpha + C_{Z_q} \frac{q\bar{c}}{2V_o} + C_{Z_{\delta_s}} \Delta \delta_s + C_{Z_{\delta_f}} \Delta \delta_f + C_{Z_{\delta_c}} \Delta \delta_c$$

$$C_m = C_{m_0} + C_{m_\alpha} \Delta \alpha + C_{m_q} \frac{q\bar{c}}{2V_o} + C_{m_{\delta_s}} \Delta \delta_s + C_{m_{\delta_f}} \Delta \delta_f + C_{m_{\delta_c}} \Delta \delta_c$$

where  $\Delta$  indicates perturbation relative to the trim value.

Correlation matrices for the two sets of data are given in Tables 5.7 and 5.8, respectively. The results for the pilot input show high correlation between  $q\bar{c}/2V_o$  and  $\delta_c$  and between  $\delta_c$  and  $\delta_s$ . High correlation is defined as a pair-wise correlation with absolute value greater than or equal to 0.9. The RAV data do not exhibit any high correlations between regressors and generally have lower pair-wise correlations overall.

Condition indices and parameter variance proportions for the two data sets are given in Tables 5.9 and 5.10. The maximum condition index (i.e., the condition number) for the pilot input data is 174, whereas the maximum condition index for the RAV input sequence is 14, indicating a reduced spread of eigenvalues when the RAV system was used. The variance proportions in Table 5.9 for the largest condition index show strong collinearity among the bias, strake, and canard. The same quantities in Table 5.10 indicate only a moderate collinearity between the bias and canard.

**Table 5.8 Correlation matrix for the regressors, RAV input sequence**

	$\alpha$	$q\bar{c}/2V_o$	$\delta_s$	$\delta_f$	$\delta_c$
$\alpha$	1.000	0.275	0.673	0.328	-0.772
$q\bar{c}/2V_o$	—	1.000	0.345	-0.375	-0.089
$\delta_s$	—	—	1.000	0.098	-0.578
$\delta_f$	—	—	—	1.000	-0.303
$\delta_c$	—	—	—	—	1.000

**Table 5.9 Collinearity diagnostics for the pilot pitch stick input**

Eigenvalue	Condition index	Variance proportions (scaled regressors)					
		1	$\alpha$	$q\bar{c}/2V_o$	$\delta_s$	$\delta_f$	$\delta_c$
3.310	1	0.0000	0.0092	0.4806	0.0017	0.0006	0.0001
1.278	3	0.0099	0.0001	0.0397	0.0720	0.0058	0.0001
1.033	3	0.0001	0.2572	0.1604	0.0000	0.0342	0.0002
0.261	13	0.0504	0.1140	0.0991	0.0762	0.1887	0.0245
0.098	34	0.1220	0.6194	0.2143	0.0041	0.7123	0.0548
0.019	174	0.8176	0.0001	0.0061	0.8460	0.0585	0.9204

Table 5.11 contains identifiability diagnostics for estimating pitching moment coefficient parameters from each data set. The table lists the increment in coefficient of determination  $\Delta R^2$ ,  $t$  statistic, and variance inflation factor for each regressor. The values of  $\Delta R^2$  represent the amount of variation in the pitching moment coefficient that was modeled by the individual terms. The  $t$  statistic can be considered a measure of significance of the individual parameters. Analysis of the pilot input data revealed that  $C_{m_{\delta_c}} \delta_c$  is the most influential term in the pitching moment equation, with a limited possibility for accurate estimates of parameters  $C_{m_{\delta_s}}$  and  $C_{m_{\delta_f}}$ , and that the significance of the  $C_{m_q} q\bar{c}/2V_o$  term is almost zero. The RAV experiment improved the identifiability of parameters  $C_{m_{\alpha}}$ ,  $C_{m_{\delta_s}}$ , and  $C_{m_{\delta_f}}$ , and showed  $C_{m_{\delta_c}} \delta_c$  as still the dominant term. The chance for accurate estimation of  $C_{m_q}$  remains small.

Estimates of two parameters  $C_{m_{\alpha}}$  and  $C_{m_{\delta_c}}$ , which contribute the most to the pitching moment model, are plotted against Mach number in Fig. 5.26, along with wind-tunnel results. The estimates were obtained from data generated by pilot pitch stick inputs using ordinary least squares and mixed estimation, and from RAV inputs using only ordinary least squares. The wind-tunnel values for the strake effectiveness were used as *a priori* values in the mixed estimation.

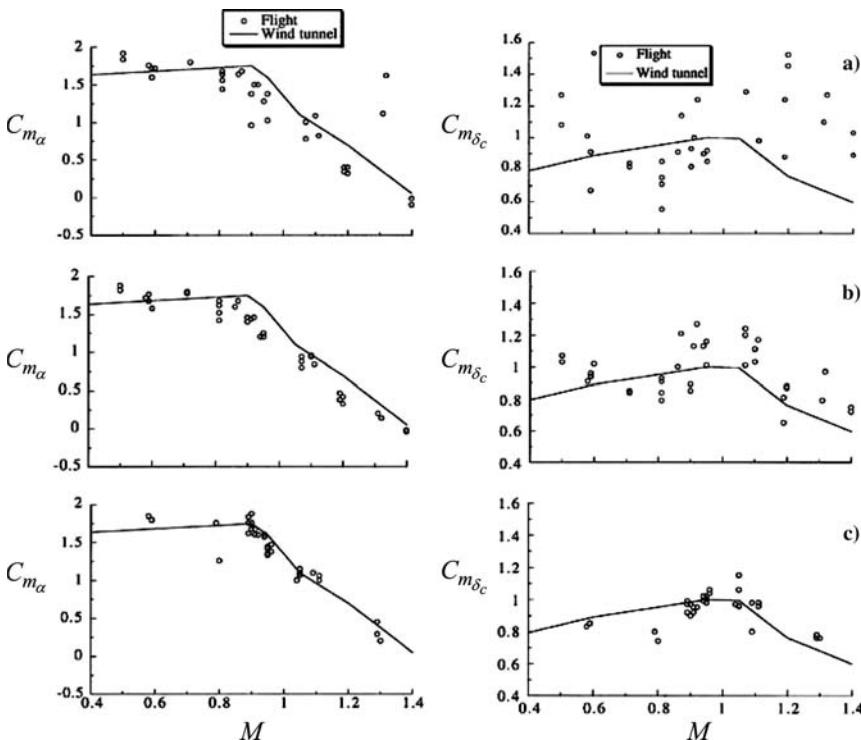
**Table 5.10 Collinearity diagnostics for the RAV input sequence**

Eigenvalue	Condition index	Variance proportions (scaled regressors)					
		1	$\alpha$	$q\bar{c}/2V_o$	$\delta_s$	$\delta_f$	$\delta_c$
2.532	1	0.0003	0.0017	0.8594	0.0003	0.0000	0.0018
1.434	2	0.0905	0.0002	0.0049	0.0008	0.0132	0.1901
1.003	3	0.0156	0.2543	0.0098	0.1481	0.1685	0.2565
0.474	5	0.2173	0.0355	0.0171	0.0178	0.4886	0.0172
0.380	7	0.0460	0.6463	0.0220	0.4605	0.0094	0.0506
0.177	14	0.6302	0.0621	0.0867	0.3724	0.4720	0.7147

**Table 5.11 Identifiability diagnostics, for the pilot pitch stick input, Fig. 5.24, and for the RAV input sequence, Fig. 5.25**

Parameter	Pilot input			RAV input sequence		
	$\Delta R^2$	$ t_0 $	VIF	$\Delta R^2$	$ t_0 $	VIF
$C_{m\alpha}$	5.1	28.0	3.3	9.1	64.8	3.6
$C_{mq}$	0.0	0.0	3.2	1.1	13.7	1.6
$C_{m\delta_s}$	1.0	9.2	31.8	4.5	42.7	2.7
$C_{m\delta_f}$	0.9	9.2	14.0	8.1	46.4	1.5
$C_{m\delta_c}$	91.5	12.3	14.9	75.7	99.1	2.0

The accuracies of the *a priori* values were determined from repeated tests in two different wind tunnels. As can be seen from Fig. 5.26, the scatter in the parameter estimates was reduced by either applying the mixed estimator to pilot input data or using data from the RAV experiment.  $\square$



**Fig. 5.26 Pitching moment parameters estimated from flight data: a) pilot input using ordinary least squares, b) pilot input using mixed estimation, and c) RAV input sequence using ordinary least squares.**

## 5.6 Data Partitioning

In most practical applications of system identification to aircraft problems, parameter estimation methods are applied to measured data from small-amplitude maneuvers executed about a trimmed flight condition. For these maneuvers, a linear aerodynamic model can be assumed. However, results from this analysis are only valid locally, that is, near the flight condition where the maneuver was performed. Information about the aerodynamic characteristics over a wider range of conditions can be obtained by analyzing maneuvers involving large variations in angle of attack, sideslip angle, and control deflections.

Analysis of large-amplitude maneuvers requires postulation of a model that might involve a relatively large number of parameters, e.g., including higher-order terms in a multivariate Taylor series expansion. Very often the increased model complexity cannot be supported by the information content in the data. This can result in parameter estimates with low accuracy, or the parameter estimation might fail.

To overcome these problems, a procedure known as data partitioning can be used (cf. Ref. 26). The idea is to divide the data points from a maneuver or set of maneuvers covering a large range of some important independent variables into partitions, where each partition contains the data points with values of important independent variables that lie within small ranges. This converts a modeling problem that might require a complicated nonlinear model structure and many model parameters into a series of simpler problems that require only linear model structures and just a few model parameters for each of the simpler problems.

Data partitioning with respect to a single variable, e.g., angle of attack  $\alpha$ , involves dividing the measured data into  $m$  subsets, where each subset contains the data points that satisfy the condition

$$\alpha_k < \alpha \leq \alpha_{k+1} \quad k = 1, 2, \dots, m \quad (5.189)$$

and the boundary values for the angle of attack partitioning,  $\alpha_k$ ,  $k = 1, 2, \dots, m + 1$ , are selected by the analyst. The median value of  $\alpha$  in the  $k$ th subset is

$$\bar{\alpha}_k = \frac{\alpha_k + \alpha_{k+1}}{2} \quad k = 1, 2, \dots, m \quad (5.190)$$

The interval  $\Delta\alpha_k = |\alpha_{k+1} - \alpha_k|$  should be small enough so that the assumption

$$\alpha \approx \bar{\alpha}_k = \text{constant} \quad (5.191)$$

can be made. In this way, dependence on  $\alpha$  is simplified or removed for each data subset, so that the modeling problem for each individual subset is simplified. The model for each subset is

$$z_k = X_k \boldsymbol{\theta}_k + \boldsymbol{v}_k \quad k = 1, 2, \dots, m \quad (5.192)$$

## REGRESSION METHODS

175

where the subscript  $k$  indicates that the data points are selected according to the value of  $\alpha$  using Eq. (5.189), and the parameter estimates apply only for that restricted range of  $\alpha$ . All of the  $\theta_k$ ,  $k = 1, 2, \dots, m$ , considered as a function of median  $\alpha$ ,  $\bar{\alpha}_k$ , together give a global view of the aerodynamic characteristics for data covering a large range in angle of attack.

In the following example, an aircraft performed many lateral maneuvers over a wide range of angles of attack  $\alpha$ . For these maneuvers taken together, the lateral coefficients might depend on  $\alpha$  in a nonlinear way, i.e.,

$$C_n = C_n(\alpha, \beta, p, r, \delta_a, \delta_r) \quad (5.193)$$

and similarly for  $C_l$  and  $C_Y$ . To simplify the modeling and parameter estimation, the measured data are partitioned into subsets with respect to  $\alpha$ . For each subset, the model is changed to

$$C_n(\bar{\alpha}_k) = C_n(\beta, p, r, \delta_a, \delta_r)_{\alpha_k < \alpha \leq \alpha_{k+1}} \quad k = 1, 2, \dots, m \quad (5.194)$$

and similarly for  $C_l$  and  $C_Y$ . In general, each subset can include data from more than one section of time during the maneuver.

Another consideration in the data partitioning process is that each partition must contain a sufficient number of data points and that the data for the regressors must exhibit substantial variability, to allow accurate parameter estimation for each partition. Model structure determination might still be necessary for each partition, since large variations may have occurred in variables other than the one on which the partitioning was based. Some trial and error is usually necessary to select appropriate boundaries for the partitions. In addition to Ref. 26, examples of data partitioning and subsequent model formulation and parameter estimation can be found in Refs. 16 and 27.

### Example 5.5

Measured data from 56 low-speed lateral maneuvers of the X-29 aircraft, described earlier in Example 5.4, were assembled into one set with 50,201 data points. The data were then partitioned into 41 one-degree  $\alpha$  subsets, and 1 three-degree  $\alpha$  subset. The distribution of data points in these subsets is shown in Fig. 5.27.

Half of the lateral maneuvers were analyzed as individual maneuvers using stepwise regression. The possibility of data collinearity in the measured data was investigated by procedures explained in Example 5.4. Application of stepwise regression to the partitioned data resulted in model structures for the lateral aerodynamic coefficients and least-squares estimates of parameters for each data partition. For data subsets with  $\alpha \leq 40$  deg, models with linear stability and control derivatives were adequate. For data at  $\alpha > 40$  deg, models for the lateral force and yawing moment coefficients included some of the nonlinear terms  $\beta^2$ ,  $\beta^3$ ,  $\alpha\beta$ ,  $\beta\delta_c$ , or  $\delta_r^3$ . However, the parameter estimates for these additional terms did not provide any consistent comprehensive information

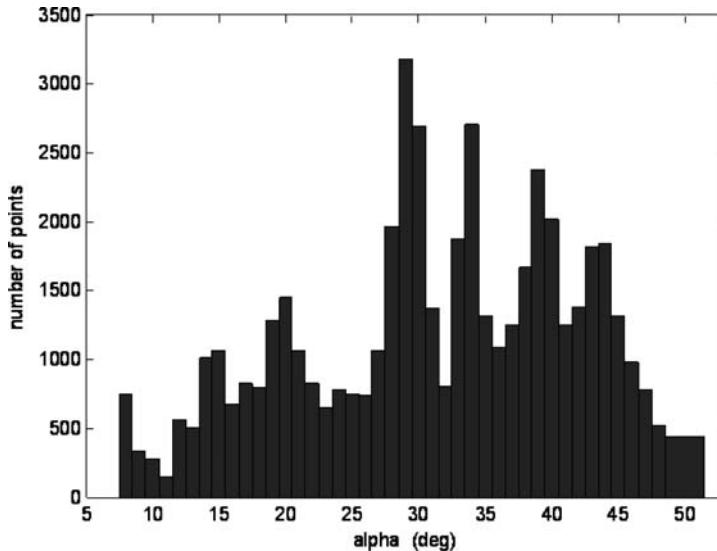


Fig. 5.27 Data partitioning for 56 lateral maneuvers of the X-29 aircraft.

about aerodynamic nonlinearities or effects of longitudinal variables on the lateral aerodynamic coefficients.

The estimated parameters (stability and control derivatives) from the 42 data partitions were fitted by quadratic polynomial splines in angle of attack. A comparison of estimates for the directional static stability derivative  $C_{n\beta}$  from partitioned data and single maneuvers is shown in Fig. 5.28. In addition, the 95% confidence limits ( $\pm 2$ -sigma limits) for prediction using the fitted quadratic spline model, computed from Eq. (5.43), are shown as dashed lines. Most of the estimates from single maneuver results lie within the prediction intervals shown. Similar behavior was observed for the other lateral aerodynamic parameters. It follows that the single maneuver results were consistent with those obtained using partitioned data. This suggests that the modeling and the data analysis were done properly for the many maneuvers involved in the analysis, and that combining the data from different maneuvers using data partitioning produces good parameter estimation results. □

## 5.7 Summary and Concluding Remarks

The techniques discussed in this chapter belong to a group of methods known as regression analysis, which is probably the most frequently used of all data analysis approaches. This chapter was concerned in particular with linear regression, in which the model equation relates a dependent variable to a sum of model terms called regressors, and each regressor is multiplied by an unknown constant parameter to be determined from measured data. The main

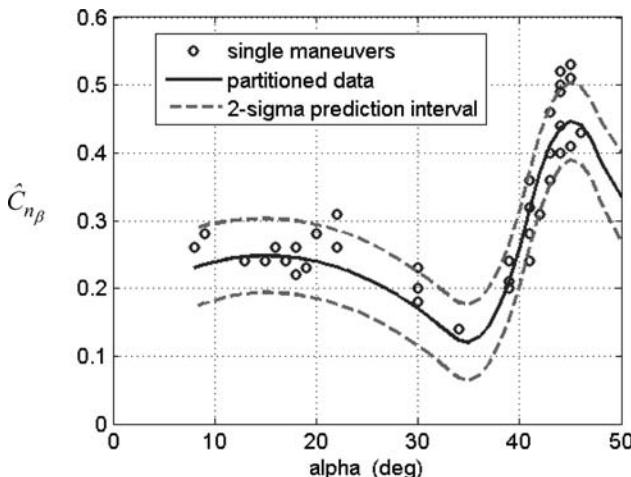


Fig. 5.28 Directional static stability parameter for the X-29 aircraft estimated from partitioned data and single maneuvers.

topics covered were linear regression, model structure determination, and data collinearity.

Details were given for the calculations required to estimate unknown parameters in a linear regression model using the least-squares principle. The resulting parameter estimates are unbiased, efficient, and consistent. The calculations also provide standard errors of the estimated parameters as a measure of their accuracy, along with measures of the quality of the model fit to the data, such as fit error and the coefficient of determination. By assuming a normal distribution for measurement errors, it was possible to construct confidence intervals for the estimated parameters, estimated output, and predicted output.

Least-squares parameter estimation using linear regression assumes that the form of the model is known. In practice, however, it is often unclear what terms should be included in the model. This uncertainty may result in a reduced model or in a model with too many terms, neither of which will be a good predictor for other similar data. Therefore, there is a need to find an adequate model that fits the data well and is a good predictor. Statistical metrics and model structure determination techniques were introduced for identifying which model terms are significant, based on the measured data, and should therefore be retained in the model. This makes it possible to identify an adequate model structure based on measured data. Two methods that have been found useful for this purpose, stepwise regression and orthogonal function modeling, were discussed in detail.

Historically, linear regression was developed for the situation where the variables that will be used to formulate the model, called the independent variables, are set to selected values by the experimenter, and therefore are assumed to be known without error. The response variable is measured directly, and assumed to be corrupted by random noise, which is the combination of measurement

errors and unknown influences. This situation corresponds to a typical wind-tunnel experiment, and linear regression methods are used for this problem on a routine basis.

For flight-test data, the situation is different in some important respects. Dependent variables, typically nondimensional aerodynamic force and moment coefficients, are not measured directly, but rather are derived from other measurements. Independent variables can no longer be set independently, because the values of these quantities change dynamically as the aircraft flies. The independent variables are typically aircraft states and control surface deflections. Although measurements of control surface deflections typically have very low noise levels and can be assumed to be measured without error, aircraft state measurements are usually noisy. This violates the assumption that independent variables are measured without error, with the result that the least-squares parameter estimation results are biased and inefficient. Although these effects can be mitigated with good instrumentation and careful data handling, this is considered the main disadvantage of the linear regression method when applied to flight data.

There are, however, significant advantages to using linear regression methods for aerodynamic modeling based on flight data. The modeling can be done using individual equations, one at a time, so that the complete problem of estimating aerodynamic model parameters for an aircraft can be done by solving a series of smaller problems. Linear regression model parameters are usually stability and control derivatives that multiply aircraft states or controls; however, model parameters can also multiply nonlinear modeling terms, such as polynomials or polynomial splines. This allows the modeling to be easily extended to nonlinear dependencies. Least-squares parameter estimation is relatively simple, and the solution does not require iteration. In addition, the linear regression approach can be used for unstable aircraft that must be tested with an automatic feedback control system active. Chapter 7 shows that the same least-squares parameter estimation equations developed in this chapter can be applied to frequency-domain data as well. Because of the model structure determination tools available for linear regression methods, and the capability to easily incorporate nonlinear functions as modeling terms, the linear regression approach is very general and useful for modeling complicated nonlinear aerodynamic dependencies, which are common for maneuvers that include large ranges or amplitudes of the flow angles, high angular rates, or unsteady aerodynamic effects.

Residuals from linear regression models based on flight data are typically correlated in time, or colored. The theory assumes that the residuals are uncorrelated in time, or white. For experiments such as wind-tunnel tests, the test points are randomized to enforce the white residuals assumption. For flight data, this is not possible because the data points are collected sequentially as the aircraft flies. The mismatch between the assumption and reality when using flight data causes inaccurate calculated values of parameter standard errors. For that reason, the theory must be modified to account for colored residuals, by reformulating the expression for the parameter covariance matrix estimate to include the autocorrelation function of the residuals. Parameter estimates are virtually unaffected by the residual coloring, so the modified parameter covariance matrix and

## REGRESSION METHODS

179

standard errors are computed after the parameters are estimated by ordinary least squares.

Another practical problem, which is not limited to linear regression modeling, is data collinearity, where model terms have similar forms, quantified by high-correlation coefficients. This can occur because of high-gain automatic feedback control, control surfaces that move simultaneously for increased control authority, and also the natural relationship among various quantities for an aircraft in flight. Procedures for detection and assessment of collinearity in linear regression were discussed in this chapter. These include evaluation of the regressor correlation matrix and its inverse, eigensystem analysis or singular value decomposition, and parameter variance decomposition.

One way of dealing with data collinearity is to use different estimation techniques from ordinary least squares. One of these techniques is the mixed estimator. It is a Bayes-like method that is applied to measured data augmented by prior information. The parameter estimates are biased but can have lower mean squared error than parameter estimates from ordinary least squares, when high data collinearity exists. An example demonstrated that this approach to dealing with data collinearity can be useful for estimating parameters of a highly augmented aircraft from flight data.

Finally, data partitioning was introduced as a data-handling strategy that can be used with linear regression methods to identify a model from data covering a wide range of important independent variables, such as angle of attack. Examples presented throughout the chapter demonstrated this and other modeling techniques.

Linear regression methods are often applied initially to measured flight data, because of their simplicity and generality. The next chapter describes additional practical methods for aircraft system identification, based on the principle of maximum likelihood.

## References

- <sup>1</sup>Draper, N. R., and Smith, H., *Applied Regression Analysis*, 2nd ed., Wiley, New York, 1981.
- <sup>2</sup>Myers, R. H., *Classical and Modern Regression with Applications*, Duxbury Press, Boston, MA, 1986.
- <sup>3</sup>Montgomery, D. C., Peck, E. A., and Vining, G. G., *Introduction to Linear Regression Analysis*, 3rd ed., Wiley, New York, 2001.
- <sup>4</sup>Chatterjee, S., and Hadi, A. S., *Sensitivity Analysis in Linear Regression*, Wiley, New York, 1988.
- <sup>5</sup>Bendat, J. S., and Piersol, A. G., *Random Data Analysis and Measurement Procedures*, 2nd ed., Wiley, New York, 1986.
- <sup>6</sup>Box, G. E. P., and Jenkins, G. M., *Time Series Analysis: Forecasting and Control*, Holden-Day, San Francisco, 1976.
- <sup>7</sup>Morelli, E. A., "Global Nonlinear Aerodynamic Modeling Using Multivariate Orthogonal Functions," *Journal of Aircraft*, Vol. 32, No. 2, 1995, pp. 270–277.
- <sup>8</sup>Morelli, E. A., and DeLoach, R., "Wind Tunnel Database Development Using Modern Experiment Design and Multivariate Orthogonal Functions," AIAA Paper 2003-0653, 2003.

- <sup>9</sup>Toutenburg, H., *Prior Information in Linear Models*, Wiley, London, 1982.
- <sup>10</sup>Goldberg, A. S., *Estimation Theory*, Wiley, London, 1964.
- <sup>11</sup>Morelli, E. A., and Klein, V., "Accuracy of Aerodynamic Model Parameters Estimated from Flight Test Data," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 1, 1997, pp. 74–80.
- <sup>12</sup>Speedy, C. B., Brown, R. F., and Goodwin, G. C., *Control Theory, Identification, and Optimal Control*, Oliver and Boyd, Edinburgh, 1970.
- <sup>13</sup>Hocking, R. R., "The Analysis and Selection of Variables in Linear Regression," *Biometrics*, Vol. 32, March 1976, pp. 1–49.
- <sup>14</sup>Efroymson, M. A., "Multiple Regression Analysis," *Mathematical Methods for Digital Computer*, edited by A. Ralston and H.S. Wilf, Wiley, New York, 1960.
- <sup>15</sup>Kashyap, R. L., "A Bayesian Comparison of Different Classes of Dynamic Models Using Empirical Data," *IEEE Transactions on Automatic Control*, Vol. AC-22, No. 5, October 1977, pp. 715–727.
- <sup>16</sup>Klein, V., Batterson, J. G., and Murphy, P. C., "Determination of Airplane Model Structure from Flight Data by Using Modified Stepwise Regression," NASA TP-1916, 1981.
- <sup>17</sup>Hall, W. E., Gupta, N. K., and Smith, R. G., "Identification of Aircraft Stability and Control Coefficients for the High Angle-of-Attack Regime," Systems Control, Inc., Engineering TR 2, Palo Alto, CA, 1974.
- <sup>18</sup>Allen, D. M., "The Prediction Sum of Squares as a Criterion for Selecting Predictor Variables," Univ. of Kentucky, TR 23, August 1971.
- <sup>19</sup>Morelli, E. A., and DeLoach, R., "Response Surface Modeling Using Multivariate Orthogonal Functions," AIAA Paper 2001-0168, 2001.
- <sup>20</sup>Barron, A. R., "Predicted Squared Error: A Criterion for Automatic Model Selection," *Self-Organizing Methods in Modeling*, edited by S.J. Farlow, Marcel Dekker, New York, 1984, pp. 87–104.
- <sup>21</sup>Morelli, E. A., "Estimating Noise Characteristics from Flight Test Data Using Optimal Fourier Smoothing," *Journal of Aircraft*, Vol. 32, No. 4, 1995, pp. 689–695.
- <sup>22</sup>Belsley, D. A., Kuh, E., and Welsh, R. E., *Regression Diagnostics: Identifying Influential Data and Sources of Collinearity*, Wiley, New York, 1980.
- <sup>23</sup>Theil, H., *Principles of Econometrics*, Wiley, New York, 1971.
- <sup>24</sup>Klein, V., and Murphy, P. C., "Aerodynamic Parameters of High Performance Aircraft Estimated from Wind Tunnel and Flight Test Data," *System Identification for Integrated Aircraft Development and Flight Testing*, NATO Res. and Techn. Org., March 1999 RTO-MP-11, Paper 18.
- <sup>25</sup>Gera, J., "Dynamic and Controls Flight Testing of the X-29A Airplane," NASA TM-86803, 1986.
- <sup>26</sup>Batterson, J. G., "Estimation of Airplane Stability and Control Derivatives from Large Amplitude Longitudinal Maneuvers," NASA TM 83185, 1981.
- <sup>27</sup>Klein, V., Ratvasky, T. R., and Cobleigh, B. R., "Aerodynamic Parameters of High-Angle-of-Attack Research Vehicle (HARV) Estimated From Flight Data," NASA TM-102692, 1990.

## 6

## Maximum Likelihood Methods

In Chapter 4, the maximum likelihood estimator was developed for the Fisher model, formulated as

$$z = \mathbf{H}\boldsymbol{\theta} + \mathbf{v} \quad \text{or} \quad z = \mathbf{h}(\boldsymbol{\theta}) + \mathbf{v} \quad (6.1)$$

where  $\boldsymbol{\theta}$  is a vector of unknown constant parameters and  $\mathbf{v}$  is  $\mathbb{N}(\mathbf{0}, \mathbf{R})$ .

It was shown that for a linear measurement equation  $z = \mathbf{H}\boldsymbol{\theta} + \mathbf{v}$ , the maximum likelihood estimator reduces to a least-squares estimator with weighting equal to the inverse of the noise covariance matrix. This chapter starts with the development of the maximum likelihood estimator for a stochastic dynamic system described by differential equations with process noise. In this case, the measurements are a nonlinear function of the parameters, as shown in Chapter 4. The general form of the relevant measurement equation is therefore  $z = \mathbf{h}(\boldsymbol{\theta}) + \mathbf{v}$ .

In general, model parameter estimates are found by maximizing a likelihood function, which involves minimizing the weighted least-squares difference between measured outputs and model outputs. The solution combines a state estimator represented by a Kalman filter and a nonlinear parameter estimator. The state estimator is necessary because the presence of process noise in the dynamic equations means that the states are random variables. A nonlinear parameter estimator is required because of the nonlinear connection between model parameters and model outputs mentioned above. Deterministic inputs are still assumed to be measured without error, as in Chapter 5. Because the maximum likelihood estimator includes a Kalman filter to estimate the states, and the outputs are computed from the resulting state estimates, this algorithm for parameter estimation is often called the filter-error method.

The combined estimation of states and parameters constitutes a difficult nonlinear estimation problem. For minimization of the cost function, the Newton-Raphson method is adopted, although other nonlinear optimization methods could also be used. For practical implementation, three simplifications are made:

- 1) The Newton-Raphson optimization scheme is replaced by a simplified version known as the Gauss-Newton method or the modified Newton-Raphson method.

- 2) For linear dynamic models, the general Kalman filter is replaced by its steady-state form.
- 3) The unknowns are divided into three subsets: states, model parameters, and noise covariance matrix elements.

Presentation of the filter-error estimation algorithm and its various forms is followed by a summary of the properties of the maximum likelihood estimates.

To develop the technique further for practical application, the additional assumption of no process noise is made. When the process noise is neglected, the states can be computed deterministically by direct numerical integration. The maximum likelihood cost function again involves weighted squared differences between measured and computed outputs. The resulting maximum likelihood estimator is known as the output-error method. This method is a generalization of the nonlinear least-squares estimator presented in Chapter 5.

In terms of experimentation, flight testing is done preferably on days with calm air, and the maneuvers are designed so that an assumed aerodynamic model structure (usually linear) will be adequate to characterize the data. This eliminates the need to include process noise in the model, so that the simpler output-error method can be used in the data analysis and modeling.

Because the output-error method is one of the most frequently used techniques for aircraft parameter estimation, attention is given to computational aspects of the solution. These include computing the sensitivities, inverting the Fisher information matrix, and computing the accuracy of the parameter estimates. Practical application of the output-error method is demonstrated with examples.

Finally, it is shown that assuming the state variables are measured without error leads to an equation-error formulation of the maximum likelihood estimator that is identical to linear regression.

## 6.1 Dynamic System with Process Noise

A wide range of practical system identification problems can be characterized by discrete-time measurements made on a continuous-time dynamic system. Chapter 3 showed that the aircraft dynamic modeling problem is in this category.

A stochastic forcing term is sometimes added to the linear dynamic equations to model gusts or to account for inadequacy of the linear model structure. The model equations are then

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{B}_w\mathbf{w}(t) \quad (6.2a)$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \quad (6.2b)$$

$$\mathbf{z}(i) = \mathbf{y}(i) + \mathbf{v}(i) \quad i = 1, 2, \dots, N \quad (6.2c)$$

where

$$E[\mathbf{x}(0)] = \bar{\mathbf{x}}_0 \quad E\{[\mathbf{x}(0) - \bar{\mathbf{x}}_0][\mathbf{x}(0) - \bar{\mathbf{x}}_0]^T\} = \mathbf{P}_0 \quad (6.2d)$$

## MAXIMUM LIKELIHOOD METHODS

183

The random vectors  $\mathbf{w}(t)$  and  $\mathbf{v}(i)$  are assumed to be white with

$$\begin{aligned} E[\mathbf{w}(t)] &= \mathbf{0} & E[\mathbf{w}(t_i)\mathbf{w}^T(t_j)] &= \mathbf{Q}(t_i)\delta(t_i - t_j) \\ E[\mathbf{v}(i)] &= \mathbf{0} & E[\mathbf{v}(i)\mathbf{v}^T(j)] &= \mathbf{R}(i)\delta_{ij} \end{aligned} \quad (6.2e)$$

Elements of the vector of unknown parameters  $\boldsymbol{\theta}$ , in general, appear in the elements of the  $\mathbf{A}, \mathbf{B}, \mathbf{B}_w, \mathbf{C}, \mathbf{D}, \mathbf{P}_0, \mathbf{Q}$ , and  $\mathbf{R}$  matrices, and the initial state vector  $\bar{\mathbf{x}}_0$ . The  $\mathbf{P}_0, \mathbf{Q}$ , and  $\mathbf{R}$  matrices are typically diagonal. The state  $\mathbf{x}$  is a vector of random variables because of the stochastic forcing term  $\mathbf{B}_w\mathbf{w}(t)$  in the dynamic equations.

Following the development in Chapter 4, the likelihood function for a sequence of measurements  $\mathbf{Z}_N = [\mathbf{z}(1) \ \mathbf{z}(2) \ \dots \ \mathbf{z}(N)]^T$  will be denoted by  $\mathbb{L}[\mathbf{Z}_N; \boldsymbol{\theta}]$ . By successive applications of Bayes's rule (see Appendix B), the expression for the likelihood function is

$$\begin{aligned} \mathbb{L}[\mathbf{Z}_N; \boldsymbol{\theta}] &= \mathbb{L}[\mathbf{z}(1), \mathbf{z}(2), \dots, \mathbf{z}(N); \boldsymbol{\theta}] \\ &= \mathbb{L}[\mathbf{z}(N) | \mathbf{Z}_{N-1}; \boldsymbol{\theta}] \mathbb{L}[\mathbf{Z}_{N-1}; \boldsymbol{\theta}] \\ &= \mathbb{L}[\mathbf{z}(N) | \mathbf{Z}_{N-1}; \boldsymbol{\theta}] \mathbb{L}[\mathbf{z}(N-1) | \mathbf{Z}_{N-2}; \boldsymbol{\theta}] \mathbb{L}[\mathbf{Z}_{N-2}; \boldsymbol{\theta}] \\ &= \vdots \\ &= \prod_{i=1}^N \mathbb{L}[\mathbf{z}(i) | \mathbf{Z}_{i-1}; \boldsymbol{\theta}] \end{aligned} \quad (6.3)$$

For computational purposes, it is advantageous to minimize the negative logarithm of the likelihood function, rather than to maximize the likelihood function. This is permissible because the logarithm is a monotonic function. The maximum likelihood estimator can be expressed in the form

$$\begin{aligned} \hat{\boldsymbol{\theta}} &= \max_{\boldsymbol{\theta}} \mathbb{L}[\mathbf{Z}_N; \boldsymbol{\theta}] \\ &= \max_{\boldsymbol{\theta}} \prod_{i=1}^N \mathbb{L}[\mathbf{z}(i) | \mathbf{Z}_{i-1}; \boldsymbol{\theta}] \\ &= \min_{\boldsymbol{\theta}} \sum_{i=1}^N -\ell_n\{\mathbb{L}[\mathbf{z}(i) | \mathbf{Z}_{i-1}; \boldsymbol{\theta}]\} \end{aligned} \quad (6.4)$$

If  $\mathbf{w}(t)$  and  $\mathbf{v}(i)$  are independent and normally distributed, then  $\mathbf{z}(i)$  will also have these properties. Then,

$$\mathbb{L}[\mathbf{z}(i) | \mathbf{Z}_{i-1}; \boldsymbol{\theta}] = \mathbb{L}[\mathbf{z}(i); \boldsymbol{\theta}] \quad (6.5)$$

will be uniquely determined by the mean and covariance. By definition,

$$\begin{aligned} E[z(i); \boldsymbol{\theta}] &\equiv \hat{y}(i | i - 1) \\ \text{Cov}[z(i); \boldsymbol{\theta}] &\equiv E\{[z(i) - \hat{y}(i | i - 1)][z(i) - \hat{y}(i | i - 1)]^T\} \\ &= E[\mathbf{v}(i)\mathbf{v}^T(i)] \equiv \mathbf{B}(i) \end{aligned} \quad (6.6)$$

where

$$\mathbf{v}(i) = z(i) - \hat{y}(i | i - 1) \quad (6.7)$$

is the  $(n_o \times 1)$  vector of innovations and  $\mathbf{B}(i)$  is the innovation covariance matrix.

As the sampling rate increases, the probability density of the innovations approach a Gaussian distribution.<sup>1</sup> Thus, for a sufficiently high sampling rate, the likelihood function can be written as

$$\mathbb{L}[z(i); \boldsymbol{\theta}] = (2\pi)^{-n_o/2} |\mathbf{B}(i)|^{-\frac{1}{2}} \exp\left[-\frac{1}{2} \mathbf{v}^T(i) \mathbf{B}^{-1}(i) \mathbf{v}(i)\right] \quad (6.8)$$

with the negative log-likelihood function for all of the measured data equal to

$$-\ell_n[\mathbb{L}(\mathbf{Z}_N; \boldsymbol{\theta})] = \frac{1}{2} \sum_{i=1}^N [\mathbf{v}^T(i) \mathbf{B}^{-1}(i) \mathbf{v}(i) + \ell_n|\mathbf{B}(i)|] + \frac{Nn_o}{2} \ell_n(2\pi) \quad (6.9)$$

where  $n_o$  is the number of output variables. The term on the far right is a constant that has no effect on the optimization problem, so it can be dropped. This leaves

$$-\ell_n[\mathbb{L}(\mathbf{Z}_N; \boldsymbol{\theta})] = \frac{1}{2} \sum_{i=1}^N [\mathbf{v}^T(i) \mathbf{B}^{-1}(i) \mathbf{v}(i) + \ell_n|\mathbf{B}(i)|] \quad (6.10)$$

The problem of determining the negative log-likelihood function is thus reduced to finding the mean and covariance of the innovations  $\mathbf{v}(i)$ . These two statistics can be obtained from a Kalman filter. As shown in Sec. 4.4, the Kalman filter recursively processes measurements one at a time. At each point, the filter produces minimum variance estimates of the state and output vector, based on all data measured up to that point. The filtering is a two-step procedure, consisting of prediction and measurement update. The corresponding sets of equations are given next and are based on Eqs. (4.61) for a linear, time-varying system. The equations are modified for constant parameters, which results in a time-invariant system.

*Initial conditions:*

$$\begin{aligned} \mathbf{x}(0) &= E[\mathbf{x}(0)] = \bar{\mathbf{x}}_0 \\ \mathbf{P}(0) &= E\{[\mathbf{x}(0) - \bar{\mathbf{x}}_0][\mathbf{x}(0) - \bar{\mathbf{x}}_0]^T\} = \mathbf{P}_0 \end{aligned} \quad (6.11a)$$

*Prediction:*

$$\frac{d}{dt}[\hat{x}(t | i - 1)] = A\hat{x}(t | i - 1) + Bu(t) \quad (6.11b)$$

$$\frac{d}{dt}[P(t | i - 1)] = AP(t | i - 1) + P(t | i - 1)A^T + B_w Q B_w^T$$

for  $(i - 1)\Delta t \leq t \leq i\Delta t \quad (6.11c)$

*Measurement update:*

$$\hat{x}(i | i) = \hat{x}(i | i - 1) + K(i) v(i) \quad (6.11d)$$

$$K(i) = P(i | i - 1) C^T B^{-1}(i | i - 1) \quad (6.11e)$$

$$P(i | i) = [I - K(i)C]P(i | i - 1) \quad (6.11f)$$

where

$$B(i | i - 1) = CP(i | i - 1)C^T + R \quad (6.11g)$$

$$v(i) = z(i) - C\hat{x}(i | i - 1) - Du(i) \quad (6.11h)$$

### 6.1.1 Optimization Algorithm

Maximum likelihood parameter estimates are obtained by minimizing the negative log-likelihood function developed above,

$$J(\theta) = \frac{1}{2} \sum_{i=1}^N v^T(i) B^{-1}(i) v(i) + \frac{1}{2} \sum_{i=1}^N |B(i)| \quad (6.12)$$

subject to the constraints imposed by Eqs. (6.11).

There are several optimization techniques that could be applied to this non-linear optimization problem. A comparison of different methods in Ref. 2 showed that the Newton-Raphson scheme has a very good convergence rate. This approach requires first- and second-order gradients of the cost function, which appear in the Taylor series expansion of  $J(\theta)$ . Assuming the vector  $\theta$  can be expressed as a small perturbation  $\Delta\theta$  from a nominal parameter estimate  $\theta_o$ ,

$$J(\theta_o + \Delta\theta) = J(\theta_o) + \Delta\theta^T \left. \frac{\partial J}{\partial \theta} \right|_{\theta=\theta_o} + \Delta\theta^T \left. \frac{\partial^2 J}{\partial \theta \partial \theta^T} \right|_{\theta=\theta_o} \Delta\theta + \dots \quad (6.13)$$

where

$\Delta\theta$  = vector of changes in  $\theta$

$\partial J / \partial \theta$  = vector of gradients  $\partial J / \partial \theta_j$ ,  $j = 1, 2, \dots, n_p$

$\partial^2 J / \partial \theta \partial \theta^T$  = second-order gradient matrix, called the Hessian matrix, with elements  $\partial^2 J / \partial \theta_j \partial \theta_k$ ,  $j, k = 1, 2, \dots, n_p$

Using the second-order expansion in Eq. (6.13) as an approximation for  $J(\boldsymbol{\theta}_o + \Delta\boldsymbol{\theta})$ ,

$$J(\boldsymbol{\theta}_o + \Delta\boldsymbol{\theta}) \approx J(\boldsymbol{\theta}_o) + \Delta\boldsymbol{\theta}^T \frac{\partial J}{\partial \boldsymbol{\theta}} \Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}_o} + \Delta\boldsymbol{\theta}^T \frac{\partial^2 J}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} \Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}_o} \Delta\boldsymbol{\theta} \quad (6.14)$$

The necessary condition for  $J(\boldsymbol{\theta}_o + \Delta\boldsymbol{\theta})$  to be a minimum is

$$\frac{\partial}{\partial \boldsymbol{\theta}} [J(\boldsymbol{\theta}_o + \Delta\boldsymbol{\theta})] = \mathbf{0} \quad (6.15)$$

Combining the last two equations,

$$\frac{\partial}{\partial \boldsymbol{\theta}} [J(\boldsymbol{\theta}_o + \Delta\boldsymbol{\theta})] = \frac{\partial J}{\partial \boldsymbol{\theta}} \Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}_o} + \frac{\partial^2 J}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} \Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}_o} \Delta\boldsymbol{\theta} = \mathbf{0} \quad (6.16)$$

The solution of the last equation provides an estimate for the vector of parameter changes,

$$\Delta\hat{\boldsymbol{\theta}} = - \left[ \frac{\partial^2 J}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} \Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}_o} \right]^{-1} \frac{\partial J}{\partial \boldsymbol{\theta}} \Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}_o} \quad (6.17)$$

assuming that the Hessian matrix is nonsingular. Since the first- and second-order gradients of the cost function are computed at a nominal value of the parameters  $\boldsymbol{\theta}_o$ , the updated parameter estimate  $\hat{\boldsymbol{\theta}}$  is computed from

$$\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}_o + \Delta\hat{\boldsymbol{\theta}} \quad (6.18)$$

Because of the approximation to  $J(\boldsymbol{\theta})$  in Eq. (6.14), it is necessary to repeat the estimation procedure by taking the estimated parameter vector  $\hat{\boldsymbol{\theta}}$  as the new nominal value  $\boldsymbol{\theta}_o$ , i.e., set  $\boldsymbol{\theta}_o = \hat{\boldsymbol{\theta}}$  for the next iteration. The reason for this iteration is that the output depends nonlinearly on the parameters, so the cost dependence on the parameters is more complicated than quadratic. In the linear regression problem of Chapter 5, the output depended linearly on the parameters, so that the cost was a quadratic function of the parameters, and Eq. (6.14) in that case was exact, not an approximation. The parameter estimates could then be obtained in one iteration, corresponding to the solution of the normal equations [cf. Eq. (5.10), which is equivalent to Eq. (6.17)]. In the present case, repeated quadratic approximations to the nonlinear dependence of the cost on the parameters are used to iteratively arrive at the solution. The iterative process is completed when selected convergence criteria are satisfied. Convergence criteria that have been found useful in practice are given in Sec. 6.4.

A complete set of expressions for the first- and second-order gradients in Eq. (6.17) can be found in Refs. 3 and 4. A block diagram of the computing algorithm for parameter estimation based on Eqs. (6.11), (6.17), and (6.18) is given in

Fig. 6.1. As mentioned earlier, this algorithm for parameter estimation is called the filter-error method, because the maximum likelihood estimator shown in Fig. 6.1 includes a Kalman filter for output estimation.

Minimization of the negative log-likelihood function for the filter-error problem is generally a very difficult optimization problem. The second-order gradients in the Hessian matrix required for the Newton-Raphson algorithm are computationally expensive to obtain, and are susceptible to numerical error. Because of this, practical applications require the use of a simplified approach known as the modified Newton-Raphson method. Details of this simplified approach to nonlinear optimization appear later in this chapter.

The estimation algorithm described earlier can be generalized to a nonlinear dynamic system with process noise and nonzero initial conditions.<sup>5</sup> The system can be described by

$$\dot{\mathbf{x}}(t) = \mathbf{f}[\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\theta}] + \mathbf{B}_w \mathbf{w}(t) \quad (6.19a)$$

$$\mathbf{y}(t) = \mathbf{h}[\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\theta}] \quad (6.19b)$$

$$\mathbf{z}(i) = \mathbf{y}(i) + \mathbf{v}(i) \quad i = 1, 2, \dots, N \quad (6.19c)$$

where

$$E[\mathbf{x}(0)] = \bar{\mathbf{x}}_0 \quad \text{and} \quad E\{[\mathbf{x}(0) - \bar{\mathbf{x}}_0][\mathbf{x}(0) - \bar{\mathbf{x}}_0]^T\} = \mathbf{P}_0 \quad (6.19d)$$

The unknown parameters appear in the nonlinear vector functions  $\mathbf{f}$  and  $\mathbf{h}$ , as well as in the matrices  $\mathbf{B}_w$ ,  $\mathbf{P}_0$ ,  $\mathbf{Q}$ , and  $\mathbf{R}$ , and in the initial state vector  $\bar{\mathbf{x}}_0$ . The Kalman filter is replaced by the extended Kalman filter outlined in Sec. 4.3.4.

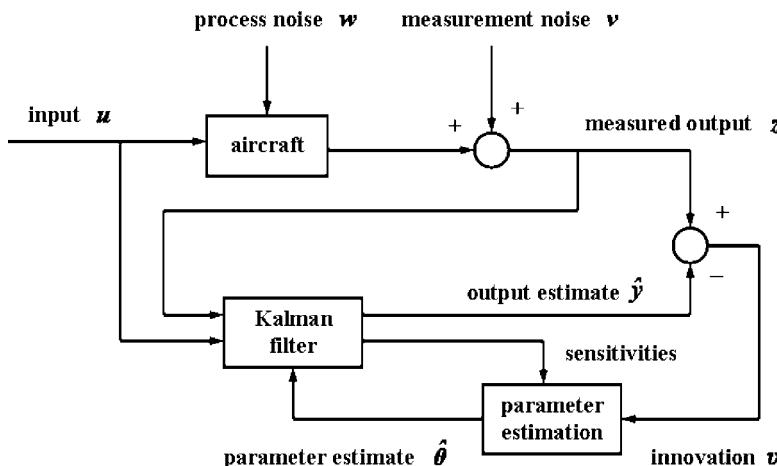


Fig. 6.1 Block diagram for filter-error parameter estimation.

### 6.1.2 Simplified Algorithms

The maximum likelihood estimation algorithm for a time-invariant stochastic linear dynamic system can be simplified by replacing the Kalman filter equations (6.11) with their steady-state forms. The equations follow from Eqs. (4.61) as

*Initial conditions:*

$$E[\mathbf{x}(0)] = \bar{\mathbf{x}}_0 \quad E\{[\mathbf{x}(0) - \bar{\mathbf{x}}_0][\mathbf{x}(0) - \bar{\mathbf{x}}_0]^T\} = \mathbf{P}_0 \quad (6.20a)$$

*Prediction:*

$$\frac{d}{dt}[\hat{\mathbf{x}}(t | i-1)] = \mathbf{A}\hat{\mathbf{x}}(t | i-1) + \mathbf{B}\mathbf{u}(t) \\ \text{for } (i-1)\Delta t \leq t \leq i\Delta t \quad (6.20b)$$

*Measurement update:*

$$\hat{\mathbf{x}}(i | i) = \hat{\mathbf{x}}(i | i-1) + \mathbf{K}\mathbf{v}(i) \quad (6.20c)$$

$$\mathbf{K} = \mathbf{P}\mathbf{C}^T\mathbf{B}^{-1} \quad (6.20d)$$

where

$$\mathbf{B} = \mathbf{C}\mathbf{P}\mathbf{C}^T + \mathbf{R} \quad (6.20e)$$

$$\mathbf{v}(i) = \mathbf{z}(i) - \mathbf{C}\hat{\mathbf{x}}(i | i-1) - \mathbf{D}\mathbf{u}(i) \quad (6.20f)$$

As suggested in Ref. 5, the steady-state discrete-time form of the Riccati equation can be solved for  $\mathbf{P}$ . Equation (4.55) is one of the forms of the Riccati equation,

$$\mathbf{P} = \Phi[\mathbf{P} - \mathbf{P}\mathbf{C}^T(\mathbf{C}\mathbf{P}\mathbf{C}^T + \mathbf{R})^{-1}\mathbf{C}\mathbf{P}]\Phi^T + \Gamma_w\mathbf{Q}\Gamma_w^T \quad (6.20g)$$

For the continuous-discrete form of the filter equations,  $\Phi$  and  $\Gamma_w$  are obtained from Eq. (2.21) as

$$\Phi \equiv e^{A\Delta t} \quad \Gamma_w \equiv [A^{-1}(e^{A\Delta t} - I)]\mathbf{B}_w \quad (6.20h)$$

In the simplified formulation, the unknown parameters are elements of matrices  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{B}_w$ ,  $\mathbf{C}$ ,  $\mathbf{D}$ ,  $\mathbf{Q}$ , and  $\mathbf{R}$ . The steady-state filter brings simplification to the equation for the first-order gradients of the cost function, because the partial derivatives of the covariance matrix  $\mathbf{P}$  with respect to the parameters are obtained from a linear matrix equation.<sup>5</sup> Despite this simplification, the estimation algorithm can still have difficulties, because the elements of both noise covariance matrices  $\mathbf{Q}$  and  $\mathbf{R}$  are treated as unknowns. These matrices appear indirectly in the likelihood function through  $\mathbf{B}$ , which is a complicated function of  $\mathbf{A}$ ,  $\mathbf{B}_w$ ,  $\mathbf{C}$ ,  $\mathbf{Q}$ , and  $\mathbf{R}$ . In addition, the convergence of the algorithm can be a problem, as illustrated in Ref. 5 on a scalar case.

In the general form given here, this estimation algorithm is very difficult to use in practice. Consequently, there have been only a few applications of this

algorithm to flight data. In Ref. 6, the problem of estimating the noise covariance matrices was not addressed, and in Ref. 7, only the process noise covariance matrix, and not the measurement noise covariance matrix, was estimated.

Further simplification to the algorithm was suggested by Mehra.<sup>8</sup> The vector of unknown parameters was recast to include elements of  $\mathbf{K}$  and  $\mathbf{B}$ , rather than elements of the  $\mathbf{Q}$  and  $\mathbf{R}$  noise covariance matrices. The gradient of the cost function with respect to  $\mathbf{B}$  can be computed by writing the cost function as

$$\begin{aligned} J(\boldsymbol{\theta}) &= \frac{1}{2} \sum_{i=1}^N \mathbf{v}^T(i) \mathbf{B}^{-1} \mathbf{v}(i) + \frac{N}{2} \ell_n |\mathbf{B}| \\ &= \frac{1}{2} \text{Tr} \left[ \mathbf{B}^{-1} \sum_{i=1}^N \mathbf{v}(i) \mathbf{v}^T(i) \right] + \frac{N}{2} \ell_n |\mathbf{B}| \end{aligned} \quad (6.21)$$

The scalar cost can then be differentiated with respect to the matrix  $\mathbf{B}$  (see Appendix A) to obtain

$$\frac{\partial J}{\partial \mathbf{B}} = -\frac{1}{2} \mathbf{B}^{-1} \sum_{i=1}^N \mathbf{v}(i) \mathbf{v}^T(i) \mathbf{B}^{-1} + \frac{N}{2} \mathbf{B}^{-1} \quad (6.22)$$

Setting the gradient equal to zero and solving for  $\mathbf{B}$  results in the estimator for the covariance matrix of the innovations,

$$\hat{\mathbf{B}} = \frac{1}{N} \sum_{i=1}^N \mathbf{v}(i) \mathbf{v}^T(i) \quad (6.23)$$

Then, for a given estimate  $\hat{\mathbf{B}}$ , the cost function becomes

$$J(\boldsymbol{\theta}) = \frac{1}{2} \sum_{i=1}^N \mathbf{v}^T(i) \hat{\mathbf{B}}^{-1} \mathbf{v}(i) \quad (6.24)$$

This cost is minimized with respect to the unknown parameters in matrices  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$ ,  $\mathbf{D}$ , and  $\mathbf{K}$  using a nonlinear optimization technique, such as the Newton-Raphson method described earlier. Once the estimates of the unknown parameters are obtained, a new sequence of residuals is computed, from which the estimate  $\hat{\mathbf{B}}$  can be updated using Eq. (6.23). The procedure continues until selected convergence criteria for the entire set of unknown parameters are satisfied. The unknowns in the problem are estimated in subsets, where the parameters contained in  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$ ,  $\mathbf{D}$ , and  $\mathbf{K}$  are held constant while the parameters in  $\mathbf{B}$  are estimated, and then the parameters contained in  $\mathbf{B}$  are held constant while the parameters in  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$ ,  $\mathbf{D}$ , and  $\mathbf{K}$  are estimated using a nonlinear optimizer. This approach to estimating the full set of unknown parameters is often referred to as a relaxation technique.

The estimate of the measurement noise covariance matrix can be easily calculated from Eqs. (6.20d) and (6.20e) as

$$\hat{\mathbf{R}} = (\mathbf{I} - \hat{\mathbf{C}} \hat{\mathbf{K}}) \hat{\mathbf{B}} \quad (6.25)$$

The computation of  $\mathbf{B}_w$  and  $\mathbf{Q}$  from Eqs. (6.20c), (6.20e), and (6.20f) is much more difficult. In general, these equations do not have a unique solution. To find one, it is necessary to impose further constraints on  $\mathbf{B}_w$  and  $\mathbf{Q}$  (see Refs. 8 and 9).

Estimating  $\mathbf{B}$  and  $\mathbf{K}$  directly simplifies the estimation problem considerably. Unfortunately, there are only a few references where this estimation algorithm was applied to flight data. In these examples, identifiability problems were detected, and the algorithm resulted in poor estimates of  $\mathbf{K}$  (see Refs. 10 and 11). Maine and Iliff<sup>5</sup> also point out a theoretical problem associated with the estimation of  $\mathbf{K}$ . Because of the requirement that  $\mathbf{P} \geq \mathbf{0}$ , elements of  $\mathbf{K}$  cannot take on arbitrary values, but instead must be constrained to lie within certain boundaries. This makes the estimation problem into a constrained nonlinear optimization.

Because of the difficulties with the two simplified versions of the general estimator, a new formulation was proposed in Ref. 5 and reiterated in Ref. 12. It takes advantage of the previous approaches by specifying the unknowns as elements in the matrices  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{B}_w$ ,  $\mathbf{C}$ ,  $\mathbf{D}$ ,  $\mathbf{Q}$ , and  $\mathbf{B}$ . The algorithm proceeds as follows:

- 1) Using nominal values of the unknown parameters, assemble the system and covariance matrices  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{B}_w$ ,  $\mathbf{C}$ ,  $\mathbf{D}$ ,  $\mathbf{Q}$ , and  $\mathbf{R}$ , and compute the innovations  $\mathbf{v}(i)$  for  $i = 1, 2, \dots, N$ .
- 2) Compute  $\hat{\mathbf{B}}$  from Eq. (6.23).
- 3) Find estimates for the unknown parameters in  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{B}_w$ ,  $\mathbf{C}$ ,  $\mathbf{D}$ , and  $\mathbf{Q}$  using a nonlinear optimization technique to minimize the cost in Eq. (6.24).
- 4) For  $\mathbf{B} = \hat{\mathbf{B}}$ , compute  $\hat{\mathbf{K}}$  and  $\hat{\mathbf{R}}$  from Eqs. (6.20e) and (6.25).
- 5) Update the nominal values of the unknown parameters.
- 6) Repeat steps 1–5 until convergence criteria are satisfied.

The filter-error method just detailed is the most general estimation method used in practical aircraft system identification, in that it allows for the existence of both process noise and measurement noise in the model. The main practical problem with this method is that there are a large number of parameters to be estimated. This often leads to difficulties with identifiability and insufficient information in the data for accurate parameter estimation.

### 6.1.3 Properties of Maximum Likelihood Parameter Estimates

The accuracy of the estimated parameters is related to the properties of the estimates. These have been discussed in numerous references, e.g. Refs. 13 and 14. Using the definitions in Sec. 4.1, the properties of the maximum likelihood estimates can be summarized as follows:

- 1) Maximum likelihood estimates of dynamic system parameters are asymptotically unbiased,

$$E(\hat{\boldsymbol{\theta}}) \rightarrow \boldsymbol{\theta} \text{ as } N \rightarrow \infty$$

This property states that the mean of the distribution for the random vector  $\hat{\boldsymbol{\theta}}$  approaches the true parameter vector  $\boldsymbol{\theta}$  as the number of data points increases.

- 2) Maximum likelihood estimates are consistent,

$$\hat{\boldsymbol{\theta}} \rightarrow \boldsymbol{\theta} \text{ as } N \rightarrow \infty$$

The estimate  $\hat{\boldsymbol{\theta}}$  approaches the true value  $\boldsymbol{\theta}$  as the number of data points increases.

- 3) Maximum likelihood estimates are asymptotically efficient,

$$\text{Cov}(\hat{\boldsymbol{\theta}}) \rightarrow \mathbf{M}^{-1} \text{ for } N \rightarrow \infty$$

where  $\mathbf{M}$  is the Fisher information matrix defined as [cf. Eq. (4.13)]

$$\mathbf{M} \equiv E \left\{ \left[ \frac{\partial \ell_n \mathbb{L}(\mathbf{Z}_N; \boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right] \left[ \frac{\partial \ell_n \mathbb{L}(\mathbf{Z}_N; \boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right]^T \right\} = -E \left[ \frac{\partial^2 \ell_n \mathbb{L}(\mathbf{Z}_N; \boldsymbol{\theta})}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} \right]$$

This property means that the main diagonal elements of the inverse information matrix provide the lower bounds on the parameter variances, called the Cramér-Rao bounds. Thus, the diagonal elements of  $\mathbf{M}^{-1}$  represent the achievable accuracy for the estimated parameters. In practice, this achievable accuracy can be closely approached for values of  $N$  associated with typical flight-test maneuver lengths and sampling rates.

- 4) Maximum likelihood estimates are asymptotically normal; i.e., the distribution of the estimates asymptotically approaches a normal distribution with mean  $\boldsymbol{\theta}$  and variance  $\mathbf{M}^{-1}$ , so that  $\hat{\boldsymbol{\theta}}$  is  $\mathcal{N}(\boldsymbol{\theta}, \mathbf{M}^{-1})$ .

## 6.2 Output-Error Method

The maximum likelihood parameter estimation method can be substantially simplified when applied to a deterministic linear dynamic system. In this case, there is no process noise, and the dynamic system can be described by

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \quad \mathbf{x}(0) = \mathbf{x}_o \quad (6.26a)$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \quad (6.26b)$$

$$z(i) = \mathbf{y}(i) + \boldsymbol{\nu}(i) \quad i = 1, 2, \dots, N \quad (6.26c)$$

$$\boldsymbol{\nu} \text{ is } \mathcal{N}(\mathbf{0}, \mathbf{R}) \quad (6.26d)$$

$$\text{Cov}[\boldsymbol{\nu}(i)] = E[\boldsymbol{\nu}(i)\boldsymbol{\nu}^T(j)] = \mathbf{R}\delta_{ij} \quad (6.26e)$$

Because there is no process noise,  $\mathbf{Q}$  is zero, and the state equations (6.26a) are deterministic. The Kalman gain  $\mathbf{K}$  is also zero, and the Kalman filter is replaced

by a simple integration of the state equations. The innovations become the output errors or residuals,

$$\mathbf{v}(i) = \mathbf{z}(i) - \mathbf{y}(i) = \mathbf{z}(i) - \mathbf{Cx}(i) - \mathbf{Du}(i) \quad i = 1, 2, \dots, N \quad (6.27)$$

The negative log-likelihood function takes the form

$$-\ell n \mathbb{L}(\mathbf{Z}_N; \boldsymbol{\theta}) = \frac{1}{2} \sum_{i=1}^N \mathbf{v}^T(i) \mathbf{R}^{-1} \mathbf{v}(i) + \frac{N}{2} \ell n |\mathbf{R}| + \frac{Nn_o}{2} \ell n(2\pi) \quad (6.28)$$

For the linear dynamic system specified by Eqs. (6.26), the unknown parameters are elements of matrices  $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ , and  $\mathbf{R}$ , and initial condition vector  $\mathbf{x}_o$ . Optimizing the right side of Eq. (6.28) with respect to  $\mathbf{R}$  is done by differentiating with respect to  $\mathbf{R}$ , setting the result equal to zero, and solving for  $\mathbf{R}$ , which gives [cf. Eq. (6.23)],

$$\hat{\mathbf{R}} = \frac{1}{N} \sum_{i=1}^N \mathbf{v}(i) \mathbf{v}^T(i) \quad (6.29)$$

Usually only the diagonal elements of the  $\mathbf{R}$  matrix are estimated from Eq. (6.29), enforcing an assumption that the measurement noise sequences for the  $n_o$  measured outputs are uncorrelated with one another. This assumption is a good one in practice, and a diagonal  $\hat{\mathbf{R}}$  simplifies the calculations. Retaining the full  $\mathbf{R}$  matrix could be done without conceptual difficulty, but the small difference in the results usually does not warrant the extra computation involved. For a given  $\hat{\mathbf{R}}$ , the negative log-likelihood cost function  $J(\boldsymbol{\theta})$  becomes

$$\begin{aligned} J(\boldsymbol{\theta}) &= \frac{1}{2} \sum_{i=1}^N \mathbf{v}(i) \hat{\mathbf{R}}^{-1} \mathbf{v}^T(i) \\ &= \frac{1}{2} \sum_{i=1}^N [\mathbf{z}(i) - \mathbf{y}(i)] \hat{\mathbf{R}}^{-1} [\mathbf{z}(i) - \mathbf{y}(i)]^T \end{aligned} \quad (6.30)$$

where the last two terms in Eq. (6.28) are dropped because they do not depend on the unknown model parameters  $\boldsymbol{\theta}$ . Since the innovations  $\mathbf{v}(i)$  in the cost function are output errors, this approach is called the output-error method.

The negative log-likelihood cost function is minimized using a relaxation technique, by computing  $\hat{\mathbf{R}}$  from Eq. (6.29) for a given fixed  $\boldsymbol{\theta}$ , then fixing  $\mathbf{R} = \hat{\mathbf{R}}$ , and minimizing the cost in Eq. (6.30) with respect to  $\boldsymbol{\theta}$ . The idea is that optimization with respect to the complete set of unknown parameters in  $\boldsymbol{\theta}$  and  $\mathbf{R}$  is more well conditioned if  $\boldsymbol{\theta}$  and  $\mathbf{R}$  are adjusted alternately, with one being allowed to vary while the other is held constant. The two steps are repeated until convergence criteria are satisfied. There is no general proof that this sequence will converge, but extensive practical experience has shown that the sequence does in fact converge.

## MAXIMUM LIKELIHOOD METHODS

193

Optimizing the cost in Eq. (6.30) can be done using the Newton-Raphson method, as discussed earlier. The gradient of the cost function is obtained as

$$\begin{aligned}\frac{\partial J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} &= \sum_{i=1}^N \frac{\partial \mathbf{v}^T(i)}{\partial \boldsymbol{\theta}} \hat{\mathbf{R}}^{-1} \mathbf{v}(i) \\ &= - \sum_{i=1}^N \frac{\partial \mathbf{y}^T(i)}{\partial \boldsymbol{\theta}} \hat{\mathbf{R}}^{-1} \mathbf{v}(i)\end{aligned}\quad (6.31)$$

which is a vector with elements

$$\frac{\partial J(\boldsymbol{\theta})}{\partial \theta_j} = - \sum_{i=1}^N \frac{\partial \mathbf{y}^T(i)}{\partial \theta_j} \hat{\mathbf{R}}^{-1} \mathbf{v}(i) \quad j = 1, 2, \dots, n_p \quad (6.32)$$

The second equality in Eq. (6.31) follows from Eq. (6.27) and the fact that the measurements  $\mathbf{z}(i)$  do not depend on the model parameters  $\boldsymbol{\theta}$ . The elements of the second-order gradient matrix are

$$\begin{aligned}\frac{\partial^2 J(\boldsymbol{\theta})}{\partial \theta_j \partial \theta_k} &= \sum_{i=1}^N \frac{\partial \mathbf{y}^T(i)}{\partial \theta_j} \hat{\mathbf{R}}^{-1} \frac{\partial \mathbf{y}(i)}{\partial \theta_k} - \sum_{i=1}^N \frac{\partial^2 \mathbf{y}(i)}{\partial \theta_j \partial \theta_k} \hat{\mathbf{R}}^{-1} \mathbf{v}(i) \\ &\quad j, k = 1, 2, \dots, n_p\end{aligned}\quad (6.33)$$

If the second-order partial derivative term in Eq. (6.33) is neglected, the resulting optimization algorithm is called Gauss-Newton or modified Newton-Raphson. This simplification is made for practical reasons, because the second-order gradient is computationally expensive to obtain and susceptible to numerical error because of the higher-order differentiation. Since the second-order gradient term is multiplied by the residual  $\mathbf{v}(i)$ , the approximation gets better as the estimated parameter vector approaches the solution, and is very good near the solution.

Using the approximate second-order gradient matrix, the estimate for the parameter vector change is [cf. Eq. (6.17)]

$$\Delta \hat{\boldsymbol{\theta}} = \left[ \sum_{i=1}^N \frac{\partial \mathbf{y}^T(i)}{\partial \boldsymbol{\theta}} \hat{\mathbf{R}}^{-1} \frac{\partial \mathbf{y}(i)}{\partial \boldsymbol{\theta}} \right]_{\boldsymbol{\theta}=\boldsymbol{\theta}_o}^{-1} \left[ \sum_{i=1}^N \frac{\partial \mathbf{y}^T(i)}{\partial \boldsymbol{\theta}} \hat{\mathbf{R}}^{-1} \mathbf{v}(i) \right]_{\boldsymbol{\theta}=\boldsymbol{\theta}_o} \quad (6.34)$$

Elements of the  $n_o \times n_p$  matrix  $\partial \mathbf{y} / \partial \boldsymbol{\theta}$  are called output sensitivities. The output sensitivities quantify the change in the outputs due to changes in the parameters. Because  $\hat{\mathbf{R}}^{-1}$  is typically diagonal, Eq. (6.34) shows that the output sensitivities must be linearly independent and nonzero for good matrix inversion and a reasonable  $\Delta \hat{\boldsymbol{\theta}}$ . When the output sensitivities are linearly independent and nonzero, each model parameter has a unique and significant impact on the model outputs, so that minimizing the output error will be a

well-conditioned problem leading to accurate values for the unknown parameters.

Using the approximation for the second-order gradient of the cost function, and assuming a given constant  $\mathbf{R} = \hat{\mathbf{R}}$ , the Fisher information matrix is simplified to

$$\mathbf{M} \equiv -E\left[\frac{\partial^2 \ell/\ln \mathbb{L}(\mathbf{Z}_N; \boldsymbol{\theta})}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T}\right] = \sum_{i=1}^N \frac{\partial \mathbf{y}^T(i)}{\partial \boldsymbol{\theta}} \hat{\mathbf{R}}^{-1} \frac{\partial \mathbf{y}(i)}{\partial \boldsymbol{\theta}} \quad (6.35)$$

The maximum likelihood parameter estimator can be expressed as

$$\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}_o - \mathbf{M}_{\boldsymbol{\theta}=\boldsymbol{\theta}_o}^{-1} \left[ \frac{\partial J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right]_{\boldsymbol{\theta}=\boldsymbol{\theta}_o} \quad (6.36)$$

and the parameter covariance matrix satisfies

$$\text{Cov}(\hat{\boldsymbol{\theta}}) \geq \mathbf{M}_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}}^{-1} \quad (6.37)$$

Equation (6.37) is the Cramér-Rao inequality, indicating the lower bound for the parameter covariance matrix (see Appendix B). A block diagram of the output-error method is shown in Fig. 6.2.

Equation (6.34) for the parameter vector estimate update can also be derived by replacing the model output in the cost function of Eq. (6.30) with a Taylor series in  $\boldsymbol{\theta}$ , expanded about a nominal parameter vector value  $\boldsymbol{\theta}_o$ , and truncated

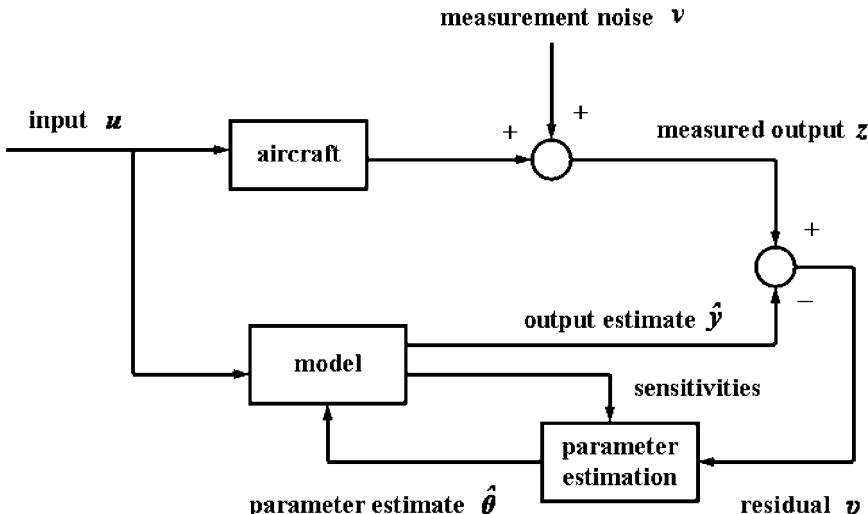


Fig. 6.2 Block diagram for output-error parameter estimation.

after the linear term, i.e.,

$$\mathbf{y}(i) \approx \mathbf{y}(i)\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}_o} + \frac{\partial \mathbf{y}(i)}{\partial \boldsymbol{\theta}}\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}_o} (\boldsymbol{\theta} - \boldsymbol{\theta}_o) \quad i = 1, 2, \dots, N \quad (6.38)$$

Equation (6.38) is a linear approximation for  $\mathbf{y}(i)$  in the neighborhood of a nominal starting value for the parameter vector,  $\boldsymbol{\theta}_o$ . Substituting this linear approximation into Eq. (6.30), setting the gradient  $\partial J / \partial \boldsymbol{\theta}$  equal to zero, and solving for  $\Delta \boldsymbol{\theta} \equiv \boldsymbol{\theta} - \boldsymbol{\theta}_o$  results in the modified Newton-Raphson parameter vector update expression in Eq. (6.34).

Since the model output was assumed to be a nonlinear function of the parameter vector, due to the time integration involved in solving Eqs. (6.26), it follows that the dynamic model can in fact be an arbitrary nonlinear function of the parameter vector. Consequently, the output-error method can be used for arbitrary nonlinear models. In particular, the full nonlinear aircraft equations of motion can be used as the dynamic model equations, without any change to the output-error cost formulation or the nonlinear optimization. Similarly, the nonlinear least-squares problem formulated in Chapter 5 can be solved using the same nonlinear optimization technique described here, among others.

### 6.3 Computational Aspects

Aspects of computing maximum likelihood estimates are presented here in connection with the output-error algorithm. Most of the material is also applicable to the more general case where the dynamic system model includes process noise.

As pointed out in Ref. 15, practical application of the maximum likelihood estimator to flight data sometimes leads to difficulties. For example, the shape of the log-likelihood function can be far from quadratic, as assumed by many nonlinear optimization methods, and can have multiple minima or discontinuities. The surface of the log-likelihood function is also ragged, because of noise in the measurements. These properties can prevent finding the global maximum of the likelihood function, which is necessary to achieve unbiased and efficient parameter estimates. Furthermore, the information matrix can be singular or nearly singular, creating problems with its inversion.

The parameter estimation method for the output-error method using modified Newton-Raphson optimization explained earlier can be summarized as

$$\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}_o + \Delta \hat{\boldsymbol{\theta}} \quad (6.39a)$$

$$\Delta \hat{\boldsymbol{\theta}} = -M_{\boldsymbol{\theta}=\boldsymbol{\theta}_o}^{-1} g_{\boldsymbol{\theta}=\boldsymbol{\theta}_o} \quad (6.39b)$$

$$\text{Cov}(\hat{\boldsymbol{\theta}}) \geq M_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}}^{-1} \quad (6.39c)$$

where

$$\mathbf{M}_{\boldsymbol{\theta}=\boldsymbol{\theta}_o} = \sum_{i=1}^N [\mathbf{S}^T(i) \hat{\mathbf{R}}^{-1} \mathbf{S}(i)]_{\boldsymbol{\theta}=\boldsymbol{\theta}_o} \quad (6.40a)$$

$$\mathbf{g}_{\boldsymbol{\theta}=\boldsymbol{\theta}_o} = \sum_{i=1}^N [\mathbf{S}^T(i) \hat{\mathbf{R}}^{-1} \mathbf{v}(i)]_{\boldsymbol{\theta}=\boldsymbol{\theta}_o} \quad (6.40b)$$

$$\mathbf{S}(i) = [s_{jk}(i)] = \begin{bmatrix} \frac{\partial y_j(i, \boldsymbol{\theta})}{\partial \theta_k} \end{bmatrix} \quad \begin{array}{l} j = 1, 2, \dots, n_o \\ k = 1, 2, \dots, n_p \end{array} \quad (6.40c)$$

$$\mathbf{v}(i) = z(i) - \hat{y}(i, \boldsymbol{\theta}) \quad (6.40d)$$

The information matrix  $\mathbf{M}$  has dimensions  $n_p \times n_p$ , and the sensitivity matrix  $\mathbf{S}(i)$  has dimensions  $n_o \times n_p$ . To arrive at the global maximum of the likelihood function using the modified Newton-Raphson method, the nominal parameter vector estimate  $\boldsymbol{\theta}_o$  must be near the value of  $\boldsymbol{\theta}$  corresponding to the minimum of the cost function  $J(\boldsymbol{\theta})$  for a given  $\hat{\mathbf{R}}$ . The reason for this is simply that the modified Newton-Raphson method is based on the assumption that  $\boldsymbol{\theta}_o$  is close to the solution. If the starting value  $\boldsymbol{\theta}_o$  is not in the same valley of the cost function as the global solution, then the result will be either convergence to a local minimum, or divergence, the latter being the more common result for flight-test data analysis.

Fortunately, excellent starting values for  $\boldsymbol{\theta}_o$  can be obtained using the linear regression methods described in Chapter 5. Alternatively, Sec. 6.3.1 includes a description of a method that can be used to eliminate the need to compute starting values  $\boldsymbol{\theta}_o$ . When there is some doubt as to the model structure, model structure determination methods discussed in Chapter 5 can be used to identify both an adequate model structure and good starting values for the parameters.

An initial estimate of  $\mathbf{R}$  can be obtained using the starting values  $\boldsymbol{\theta}_o$  and Eq. (6.29), or  $\hat{\mathbf{R}}$  can be initially set to the identity matrix,  $\hat{\mathbf{R}} = \mathbf{I}$ . The updates of  $\hat{\mathbf{R}}$  using Eq. (6-29) should occur after each optimization of  $J(\boldsymbol{\theta})$  in Eq. (6.30) satisfies the convergence criteria. The combined optimization is finished when convergence criteria for both  $\hat{\boldsymbol{\theta}}$  and  $\hat{\mathbf{R}}$  are satisfied. Typical convergence criteria involve one or more of the following:

- 1) Absolute value of the elements of  $\Delta\boldsymbol{\theta}$  are sufficiently small.
- 2) Changes in the cost  $J(\hat{\boldsymbol{\theta}})$  are sufficiently small for consecutive iterations.
- 3) Absolute values of the elements of the cost gradient  $\mathbf{g}$  are sufficiently close to zero.
- 4) Changes in the elements of  $\hat{\mathbf{R}}$  are sufficiently small.

The following convergence criteria have been found to work well for aircraft parameter estimation problems:

$$|(\hat{\theta}_j)_k - (\hat{\theta}_j)_{k-1}| < 1.0 \times 10^{-5} \quad \forall j, \quad j = 1, 2, \dots, n_p \quad (6.41a)$$

or alternatively,

$$\frac{\|\hat{\boldsymbol{\theta}}_k - \hat{\boldsymbol{\theta}}_{k-1}\|}{\|\hat{\boldsymbol{\theta}}_{k-1}\|} < 0.001 \quad (6.41b)$$

## MAXIMUM LIKELIHOOD METHODS

197

where  $k$  denotes the  $k$ th iteration and  $\| \cdot \|$  indicates the Euclidean norm, or the square root of the sum of squares of the vector elements. At the same time,

$$\left| \frac{J(\hat{\boldsymbol{\theta}}_k) - J(\hat{\boldsymbol{\theta}}_{k-1})}{J(\hat{\boldsymbol{\theta}}_{k-1})} \right| < 0.001 \quad (6.41c)$$

$$\left| \left( \frac{\partial J(\boldsymbol{\theta})}{\partial \theta_j} \right)_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}_k} \right| < 0.05 \quad \forall j, \quad j = 1, 2, \dots, n_p \quad (6.41d)$$

$$\left| \frac{(\hat{r}_{jj})_k - (\hat{r}_{jj})_{k-1}}{(\hat{r}_{jj})_{k-1}} \right| < 0.05 \quad \forall j, \quad j = 1, 2, \dots, n_o \quad (6.41e)$$

where  $\hat{r}_{jj}$  denotes the estimate of the  $j$ th diagonal element of  $\hat{\mathbf{R}}$ .

In some cases, varying the magnitude of  $\Delta\hat{\boldsymbol{\theta}}$  in the direction computed from Eq. (6.39b) might provide an additional decrease in the cost function for each iteration. This can be done by multiplying the expression for  $\Delta\hat{\boldsymbol{\theta}}$  by a constant  $c$ ,

$$\Delta\hat{\boldsymbol{\theta}} = -c M_{\boldsymbol{\theta}=\boldsymbol{\theta}_o}^{-1} \mathbf{g}_{\boldsymbol{\theta}=\boldsymbol{\theta}_o} \quad (6.42)$$

This approach is discussed in Ref. 16. Speedy et al.<sup>3</sup> give the formula for computing  $c$  using quadratic interpolation. However, this approach is computationally expensive because it requires three computed values of the cost for each iteration. For practical aircraft parameter estimation problems, this refinement of the modified Newton-Raphson algorithm is probably not worth the trouble, because modern computers are so fast that the difference in convergence time is small.

### 6.3.1 Computing Sensitivities

Two methods can be used to compute the sensitivities  $\partial\mathbf{y}/\partial\theta_j, j = 1, 2, \dots, n_p$ , appearing in the equations of the preceding section. The methods can be called the analytical approach and the numerical approach.

In the analytical approach, equations for the output sensitivities  $\partial\mathbf{y}/\partial\theta_j, j = 1, 2, \dots, n_p$ , are generated by taking partial derivatives of the output equation with respect to the unknown parameters  $\theta_j, j = 1, 2, \dots, n_p$ , and solving the resulting equations. For the linear dynamic system in Eqs. (6.26), the result is

$$\frac{\partial\mathbf{y}}{\partial\theta_j} = \mathbf{C} \frac{\partial\mathbf{x}}{\partial\theta_j} + \frac{\partial\mathbf{C}}{\partial\theta_j} \mathbf{x} + \frac{\partial\mathbf{D}}{\partial\theta_j} \mathbf{u} \quad j = 1, 2, \dots, n_p \quad (6.43)$$

The preceding equations are called the output sensitivity equations. Since  $\mathbf{C}$  and  $\mathbf{D}$  are constant matrices containing unknown parameters,  $\partial\mathbf{C}/\partial\theta_j$  and  $\partial\mathbf{D}/\partial\theta_j$  are also constant matrices. The state  $\mathbf{x}$  is computed from the state equations for the dynamic system

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \quad \mathbf{x}(0) = \mathbf{x}_o \quad (6.44)$$

The state sensitivities  $\partial \mathbf{x} / \partial \theta_j, j = 1, 2, \dots, n_p$ , are computed by solving the state sensitivity equations, which are obtained by differentiating the state equation (6.44) with respect to the unknown parameters,

$$\frac{d}{dt} \left( \frac{\partial \mathbf{x}}{\partial \theta_j} \right) = \mathbf{A} \frac{\partial \mathbf{x}}{\partial \theta_j} + \frac{\partial \mathbf{A}}{\partial \theta_j} \mathbf{x} + \frac{\partial \mathbf{B}}{\partial \theta_j} \mathbf{u} \quad \frac{\partial \mathbf{x}(0)}{\partial \theta_j} = \mathbf{0} \quad j = 1, 2, \dots, n_p \quad (6.45)$$

On the left side of Eq. (6.45), the order of differentiation with respect to time and  $\theta_j$  was switched, which can be done under the assumption that  $\mathbf{x}$  is analytic. The initial conditions for the state sensitivity equations are zero when the initial condition  $\mathbf{x}_0$  is known, as assumed here. The initial state vector  $\mathbf{x}_0$  can be obtained from locally smoothed initial measurements (see Chapter 11). Equations (6.44) and (6.45) are integrated numerically, using a Runge-Kutta method, for example.

For  $n_s$  states, there are  $n_s n_p$  differential equations to be solved for the state sensitivities, plus  $n_s$  for the dynamic system equations, for a total of  $n_s(n_p + 1)$  differential equations to solve. A scheme for reducing the sensitivity calculations is presented in Ref. 17.

Alternatively, discrete formulas can be used with the state transition matrix. The discrete formulas follow from Eqs. (2.20) and (2.21) as

$$\mathbf{x}(i) = \Phi \mathbf{x}(i-1) + \Gamma \frac{[\mathbf{u}(i) + \mathbf{u}(i-1)]}{2} \quad \mathbf{x}(0) = \mathbf{x}_0 \quad (6.46a)$$

where the quantity  $[\mathbf{u}(i) + \mathbf{u}(i-1)]/2$  is an averaging of the input values at the endpoints of the time interval, which improves the accuracy of the solution for the discrete-time dynamic equation. The state sensitivity equations are

$$\begin{aligned} \frac{\partial \mathbf{x}(i)}{\partial \theta_j} &= \Phi \frac{\partial \mathbf{x}(i-1)}{\partial \theta_j} + \Phi \frac{\partial \mathbf{A}}{\partial \theta_j} \frac{[\mathbf{x}(i) + \mathbf{x}(i-1)]}{2} \\ &\quad + \mathbf{A}^{-1} (e^{A\Delta t} - \mathbf{I}) \left[ \frac{\partial \mathbf{A}}{\partial \theta_j} \mathbf{B} + \frac{\partial \mathbf{B}}{\partial \theta_j} \right] \frac{[\mathbf{u}(i) + \mathbf{u}(i-1)]}{2} \end{aligned} \quad (6.46b)$$

$$\frac{\partial \mathbf{x}(0)}{\partial \theta_j} = \mathbf{0}$$

where

$$\Phi \equiv e^{A \Delta t} \quad \Gamma \equiv \int_0^{\Delta t} e^{A \tau} d\tau \mathbf{B} = [\mathbf{A}^{-1} (e^{A\Delta t} - \mathbf{I})] \mathbf{B} \quad (6.46c)$$

The forcing functions for the state sensitivities in Eq. (6.45) are the states and controls. Taylor and Iliff<sup>18</sup> recommend computing the state sensitivities initially using measured states. This has the effect of generating sensitivities that are close enough to the final sensitivities that the parameter change computed from Eq. (6.34) using these initial sensitivities will put the estimated parameter vector in close proximity to the solution, regardless of the starting point. Therefore, the starting value  $\boldsymbol{\theta}_0$  can be the zero vector, as long as the initial sensitivities are computed using measured states in the state sensitivity equations. This approach works well in practice, and avoids the need to assemble a good

## MAXIMUM LIKELIHOOD METHODS

199

initial parameter vector estimate from equation-error methods or wind-tunnel results. The price for this convenience is some additional programming to make the first iteration of the state sensitivity calculations different from all subsequent calculations.

The numerical approach to computing output sensitivities uses an approximation to the definition of numerical partial derivatives,

$$\frac{\partial \mathbf{y}}{\partial \theta_j} = \frac{\mathbf{y}(\boldsymbol{\theta}_o + \delta\boldsymbol{\theta}_j) - \mathbf{y}(\boldsymbol{\theta}_o)}{|\delta\boldsymbol{\theta}_j|} \quad j = 1, 2, \dots, n_p \quad (6.47a)$$

or

$$\frac{\partial \mathbf{y}}{\partial \theta_j} = \frac{\mathbf{y}(\boldsymbol{\theta}_o + \delta\boldsymbol{\theta}_j) - \mathbf{y}(\boldsymbol{\theta}_o - \delta\boldsymbol{\theta}_j)}{2|\delta\boldsymbol{\theta}_j|} \quad j = 1, 2, \dots, n_p \quad (6.47b)$$

where  $\delta\boldsymbol{\theta}_j$  denotes a vector with all zero elements except for the  $j$ th element, which contains the perturbation for parameter  $\theta_j$ , and  $|\delta\boldsymbol{\theta}_j|$  is the scalar magnitude of  $\delta\boldsymbol{\theta}_j$ . The size of the perturbation for each parameter can be chosen to reflect the relative significance of each parameter's influence on the outputs, but in practice using perturbations that are fixed at 1% of the nominal parameter value works well for many applications. If the nominal parameter value is near zero, then the perturbation can be set to 0.01. The output  $\mathbf{y}$  is computed by numerically solving the dynamic equations, using the nominal and perturbed values of the parameter vector.

Equation (6.47a) implements forward finite differences and Eq. (6.47b) implements central finite differences. Forward finite differences requires  $(n_p + 1)$  solutions of the dynamic equations, whereas central finite differences requires  $2n_p$  solutions. The central finite difference approach is much more accurate, since its error in approximating the partial derivatives is  $\mathcal{O}(|\delta\boldsymbol{\theta}_j|^2)$ , whereas the errors for forward finite differences are  $\mathcal{O}(|\delta\boldsymbol{\theta}_j|)$ . Note that the sensitivities must be calculated for every iteration of the modified Newton-Raphson method, so the difference in the amount of computation can be large. However, the central finite difference calculation is preferred, because the number of iterations of Eqs. (6.39) will generally be lower with higher accuracy sensitivities.

When the dynamic system model is nonlinear,

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}, \mathbf{u}, \boldsymbol{\theta}) \quad \mathbf{x}(0) = \mathbf{x}_o \quad (6.48a)$$

$$\mathbf{y} = \mathbf{h}(\mathbf{x}, \mathbf{u}, \boldsymbol{\theta}) \quad (6.48b)$$

the resulting sensitivity equations are

$$\frac{d}{dt} \left( \frac{\partial \mathbf{x}}{\partial \theta_j} \right) = \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \frac{\partial \mathbf{x}}{\partial \theta_j} + \frac{\partial \mathbf{f}}{\partial \theta_j} \quad \frac{\partial \mathbf{x}(0)}{\partial \theta_j} = \mathbf{0} \quad j = 1, 2, \dots, n_p \quad (6.49a)$$

$$\frac{\partial \mathbf{y}}{\partial \theta_j} = \frac{\partial \mathbf{h}}{\partial \mathbf{x}} \frac{\partial \mathbf{x}}{\partial \theta_j} + \frac{\partial \mathbf{h}}{\partial \theta_j} \quad j = 1, 2, \dots, n_p \quad (6.49b)$$

In this case, the numerical approach is used more often, because the analytic derivatives of the nonlinear functions  $f$  and  $h$  can be complicated and are different for every problem.

A generalization of the finite difference method is the local surface approximation method of Ref. 19. In this technique, the output sensitivities are computed using the slope information from a local surface approximation of  $y(\theta)$ . The approximations are made near  $y(\theta_o)$ . According to Murphy,<sup>19</sup> this approach requires less computational effort than either a finite difference method or an integration of state and output sensitivity equations.

### Example 6.1

Flight-test data from Example 5.1, run 1, are used for this example. The data are from a lateral maneuver of the NASA Twin Otter aircraft. Figure 5.4 shows the measured inputs and outputs. In Chapter 5, these data were analyzed using linear regression. In this example, the output-error method will be used to obtain parameter estimates from the same data.

In the output-error method, the equations of motion are used together, so the lateral aerodynamic model parameters are estimated all at once. This contrasts with the equation-error method, where a separate analysis is done for each dynamic equation to estimate the parameters associated with each lateral aerodynamic coefficient  $C_Y$ ,  $C_l$ , and  $C_n$ .

The maneuver shown in Fig. 5.4 is a small-perturbation maneuver at low angle of attack, so a linearized dynamic model can be used, with the aerodynamic model equations given by Eq. (5.102). This results in the following dynamic model equations [cf. Eqs. (3.111) and (3.116)],

$$\begin{aligned} \dot{\beta} = & \frac{\bar{q}_o S}{m V_o} \left( C_{Y\beta} \beta + C_{Yr} \frac{rb}{2V_o} + C_{Y\delta_r} \delta_r \right) \\ & + p \sin \alpha_o - r \cos \alpha_o + \frac{g \cos \theta_o}{V_o} \phi + b_\beta \end{aligned} \quad (6.50a)$$

$$I_x \dot{p} - I_{xz} \dot{r} = \bar{q}_o S b \left( C_{l\beta} \beta + C_{lp} \frac{pb}{2V_o} + C_{lr} \frac{rb}{2V_o} + C_{l\delta_a} \delta_a + C_{l\delta_r} \delta_r \right) + b_p \quad (6.50b)$$

$$I_z \dot{r} - I_{xz} \dot{p} = \bar{q}_o S b \left( C_{n\beta} \beta + C_{np} \frac{pb}{2V_o} + C_{nr} \frac{rb}{2V_o} + C_{n\delta_a} \delta_a + C_{n\delta_r} \delta_r \right) + b_r \quad (6.50c)$$

$$\dot{\phi} = p + \tan \theta_o r + b_\phi \quad (6.50d)$$

$$a_y = \frac{\bar{q}_o S}{mg} \left( C_{Y\beta} \beta + C_{Yr} \frac{rb}{2V_o} + C_{Y\delta_r} \delta_r \right) + b_{ay} \quad (6.51)$$

Equations (6.50) are linear dynamic equations for the lateral states  $\beta$ ,  $p$ ,  $r$ , and  $\phi$ , which are also measured outputs, and Eq. (6.51) is an output equation for lateral acceleration. Perturbation quantities are used in these linearized equations of motion, and the aerodynamic model structure is the same as for Example 5.1.

## MAXIMUM LIKELIHOOD METHODS

201

Values with subscript  $o$  are constants equal to measured trim values at the start of the maneuver, or mean values computed from the measured data. In this example, mean values are used. Mass and inertia quantities are also mean values, although, for simplicity, this is not shown in the notation.

Note that all of the dynamic equations (6.50) include a bias parameter. This is to account for any measurement biases not removed by data compatibility analysis (see Chapter 10) or biases that result from using perturbation quantities. For example, if the initial measured value of rudder deflection is biased or noisy, then using that initial value to generate the perturbation quantity  $\delta_r$  will artificially induce a bias in  $\delta_r$ . Estimating a bias parameter in each dynamic equation will remove effects like this. Without the bias parameters for each dynamic equation, integration of the dynamic equations can result in a drift in the computed state time histories. The bias term in the output equation for lateral acceleration is introduced to account for biases from inaccurate trim values (as described earlier) and bias in the measured perturbation output  $a_y$ .

Equations (6.50) and (6.51) can be written in state-space form as

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & I_{xx} & -I_{xz} & 0 \\ 0 & -I_{xz} & I_{zz} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{\beta} \\ \dot{p} \\ \dot{r} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} k_1 C_{Y\beta} & \sin \alpha_o & (k_1 C_{Yr} k_2 - \cos \alpha_o) & \frac{g \cos \theta_o}{V_o} \\ k_3 C_{l\beta} & k_3 C_{lp} k_2 & k_3 C_{lr} k_2 & 0 \\ k_3 C_{n\beta} & k_3 C_{np} k_2 & k_3 C_{nr} k_2 & 0 \\ 0 & 1 & \tan \theta_o & 0 \end{bmatrix} \begin{bmatrix} \beta \\ p \\ r \\ \phi \end{bmatrix} + \begin{bmatrix} 0 & k_1 C_{Y\delta_r} & b_\beta \\ k_3 C_{l\delta_a} & k_3 C_{l\delta_r} & b_p \\ k_3 C_{n\delta_a} & k_3 C_{n\delta_r} & b_r \\ 0 & 0 & b_\phi \end{bmatrix} \begin{bmatrix} \delta_a \\ \delta_r \\ 1 \end{bmatrix} \quad (6.52a)$$

$$\begin{bmatrix} \beta \\ p \\ r \\ \phi \\ a_y \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ k_4 C_{Y\beta} & 0 & k_4 C_{Yr} k_2 & 0 \end{bmatrix} \begin{bmatrix} \beta \\ p \\ r \\ \phi \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & k_4 C_{Y\delta_r} & b_{a_y} \end{bmatrix} \begin{bmatrix} \delta_a \\ \delta_r \\ 1 \end{bmatrix} \quad (6.52b)$$

where

$$k_1 = \frac{\bar{q}_o S}{m V_o} \quad k_2 = \frac{b}{2 V_o} \quad k_3 = \bar{q}_o S b \quad k_4 = \frac{\bar{q}_o S}{mg}$$

Equations (6.50b) and (6.50c) [equivalently, the second and third rows of Eq. (6.52a)] must be combined to produce equations with only a single state derivative on the left-hand side. This is the form required for numerical integration techniques such as Runge-Kutta methods. The required calculations are simple, and were done in Chapter 3 to arrive at Eqs. (3.41). Equations (6.50b)

and (6.50c) are transformed as follows before numerical integration:

$$\dot{p} = c_3 L + c_4 N + b'_p \quad (6.53a)$$

$$\dot{r} = c_4 L + c_9 N + b'_r \quad (6.53b)$$

where

$$L = \bar{q}_o Sb \left( C_{l\beta} \beta + C_{lp} \frac{pb}{2V_o} + C_{lr} \frac{rb}{2V_o} + C_{l\delta_a} \delta_a + C_{l\delta_r} \delta_r \right)$$

$$N = \bar{q}_o Sb \left( C_{n\beta} \beta + C_{np} \frac{pb}{2V_o} + C_{nr} \frac{rb}{2V_o} + C_{n\delta_a} \delta_a + C_{n\delta_r} \delta_r \right)$$

$$\Gamma = I_x I_z - I_{xz}^2 \quad c_3 = I_z / \Gamma \quad c_4 = I_{xz} / \Gamma \quad c_9 = I_x / \Gamma$$

$$b'_p = c_3 b_p + c_4 b_r$$

$$b'_r = c_4 b_p + c_9 b_r$$

The bias parameters exist only to correct for biases in the measured perturbation quantities. Because of this, it is acceptable to redefine the bias terms as just shown. Such parameters, sometimes called *nuisance* parameters, must be estimated along with the others, although their estimates are not of direct interest for the problem.

Modified Newton-Raphson optimization, [cf. Eqs. (6.39)–(6.40)], was used to minimize the cost function in Eq. (6.30), with the measurement noise covariance matrix estimate computed from Eq. (6.29). Output sensitivities were computed numerically using central finite differences [cf. Eq. (6.47b)]. The vector of parameters to be estimated was

$$\boldsymbol{\theta} = \begin{bmatrix} C_{Y\beta} & C_{Yr} & C_{Y\delta_r} & b_\beta \\ C_{l\beta} & C_{lp} & C_{lr} & C_{l\delta_a} & C_{l\delta_r} & b'_p \\ C_{n\beta} & C_{np} & C_{nr} & C_{n\delta_a} & C_{n\delta_r} & b'_r & b_\phi & b_{ay} \end{bmatrix}^T$$

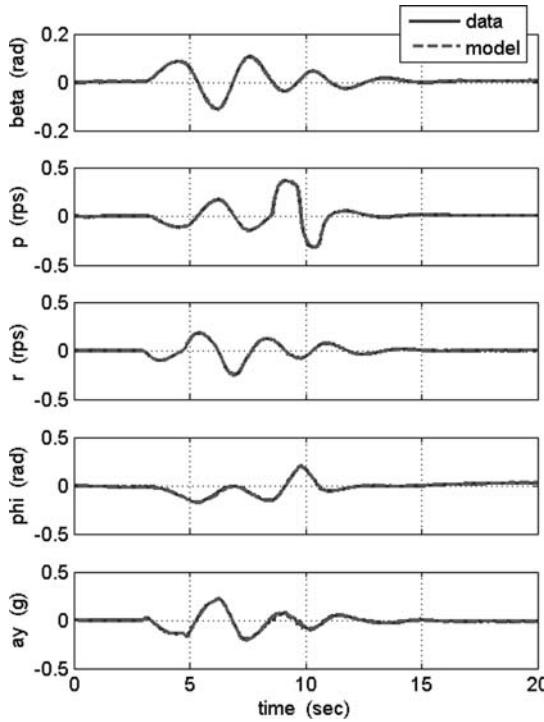
Figure 6.3 shows the model fit to the measured outputs  $\beta$ ,  $p$ ,  $r$ ,  $\phi$ , and  $a_y$ . The plots show an excellent model fit to the measured data, using the assumed model structure. The solution took 41 modified Newton-Raphson iterations, and 5 updates of  $\hat{\mathbf{R}}$ . Criteria given in Eq. (6.41) were used to define convergence.

Parameter estimation results from the output-error method are shown in Table 6.1, together with results obtained using the equation-error method demonstrated in Example 5.1. The estimated parameter covariance matrix for the output-error method was computed from Eq. (6.39c) as the Cramér-Rao lower bound,  $\text{Cov}(\hat{\boldsymbol{\theta}}) = \mathbf{M}_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}}^{-1}$ . In this calculation, the residuals are assumed to be white. The next example in this chapter demonstrates the changes to the estimated parameter errors when the residual coloring is taken into account.

It is also possible to use the measured inputs and outputs directly in the output-error method, rather than using perturbation quantities. The only

## MAXIMUM LIKELIHOOD METHODS

203



**Fig. 6.3 Output-error model fit to measured perturbation outputs for lateral maneuver, run 1.**

**Table 6.1 Parameter estimation results for lateral maneuver, run 1**

Parameter	Output error		Equation error	
	$\hat{\theta}$	$s(\hat{\theta})$	$\hat{\theta}$	$s(\hat{\theta})$
$C_{Y\beta}$	$-8.66 \times 10^{-1}$	$2.80 \times 10^{-3}$	$-8.45 \times 10^{-1}$	$7.41 \times 10^{-3}$
$C_{Yr}$	$9.31 \times 10^{-1}$	$1.65 \times 10^{-2}$	$9.63 \times 10^{-1}$	$5.53 \times 10^{-2}$
$C_{Y\delta_r}$	$3.75 \times 10^{-1}$	$7.98 \times 10^{-3}$	$3.43 \times 10^{-1}$	$2.89 \times 10^{-2}$
$C_{l\beta}$	$-1.19 \times 10^{-1}$	$5.60 \times 10^{-4}$	$-1.05 \times 10^{-1}$	$2.54 \times 10^{-3}$
$C_{l_p}$	$-5.84 \times 10^{-1}$	$2.39 \times 10^{-3}$	$-5.28 \times 10^{-1}$	$9.93 \times 10^{-3}$
$C_{l_r}$	$1.88 \times 10^{-1}$	$1.71 \times 10^{-3}$	$1.93 \times 10^{-1}$	$8.33 \times 10^{-3}$
$C_{l\delta_a}$	$-2.28 \times 10^{-1}$	$8.43 \times 10^{-4}$	$-2.12 \times 10^{-1}$	$3.39 \times 10^{-3}$
$C_{l\delta_r}$	$3.84 \times 10^{-2}$	$7.12 \times 10^{-4}$	$2.99 \times 10^{-2}$	$3.86 \times 10^{-3}$
$C_{n\beta}$	$8.65 \times 10^{-2}$	$2.07 \times 10^{-4}$	$8.54 \times 10^{-2}$	$3.58 \times 10^{-4}$
$C_{np}$	$-6.39 \times 10^{-2}$	$9.49 \times 10^{-4}$	$-5.15 \times 10^{-2}$	$1.43 \times 10^{-3}$
$C_{nr}$	$-1.92 \times 10^{-1}$	$6.07 \times 10^{-4}$	$-1.98 \times 10^{-1}$	$1.30 \times 10^{-3}$
$C_{n\delta_a}$	$-2.73 \times 10^{-3}$	$3.50 \times 10^{-4}$	$2.34 \times 10^{-3}$	$5.00 \times 10^{-4}$
$C_{n\delta_r}$	$-1.36 \times 10^{-1}$	$2.66 \times 10^{-4}$	$-1.31 \times 10^{-1}$	$5.97 \times 10^{-4}$

difference is that the bias parameters then also take on the task of estimating the constants that would have been subtracted to produce perturbation quantities. For example, in Eq. (6.50a), the bias term  $b_{\beta}$  would now also include the term  $-(\bar{q}_o S/mV_o)C_{Y_{\delta_r}} \delta_{r_0}$  (among others), where  $\delta_{r_0}$  is trim value of the rudder. These terms were already removed when perturbation quantities were used in the analysis. Since the biases are nuisance parameters anyway, their role can be changed in this way without any effect on the useful results.

Using measured inputs and outputs in the analysis puts the burden of estimating the trim value to be removed into the bias term estimate. The disadvantage is that all of the trim value estimates for each equation are lumped into one parameter, so it is not clear what term was effectively subtracted for each variable. The advantage is that the analysis is done using the measured physical magnitudes of the measured inputs and outputs, and it is not necessary to determine the trim value that should be subtracted for the perturbations, since those values are estimated by the bias term. There is not much difference either way for maneuvers at low angles of attack where most of the trim quantities are small. However, the issue increases in prominence as the trim angle of attack for the maneuver increases and the trim values become larger, for both longitudinal and lateral maneuvers.

Figure 6.4 shows measured and predicted outputs for a different lateral maneuver at a similar flight condition, run 2. This is the same prediction maneuver used in Example 5.1. For the prediction case, the bias parameters

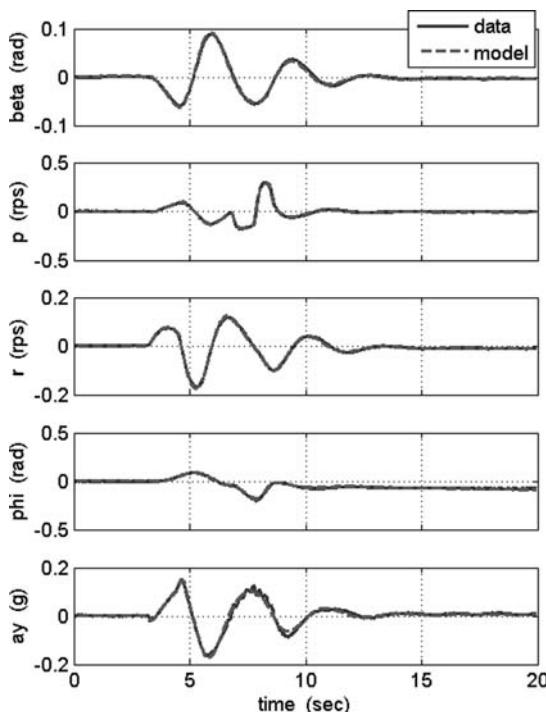


Fig. 6.4 Model prediction for lateral maneuver, run 2.

$b_{\dot{\beta}}$ ,  $b'_{\dot{p}}$ ,  $b'_{\dot{r}}$ ,  $b_{\dot{b}}$ , and  $b_{a_y}$  were re-estimated using output-error parameter estimation and the prediction data, to account for small bias errors in the measurements. All other model parameters were held fixed at the estimated values given in Table 6.1. The plots show that the identified model is a good predictor for a maneuver with different inputs at the same flight condition.  $\square$

### 6.3.2 Nearly Singular Information Matrix

The problem of a nearly singular information matrix, also called an ill-conditioned information matrix, has already been discussed in connection with properties of the  $X^T X$  matrix for linear regression. In that case, the sensitivities of the model output to the parameters were just the columns of the  $X$  matrix. For the case where the output is a vector nonlinear function of the parameters, the analog of the  $X$  matrix is the matrix of output sensitivities  $\partial \mathbf{y} / \partial \boldsymbol{\theta}$ , which are weighted in the information matrix using the noise covariance matrix [cf. Eq. (6.40)]. This arrangement includes contributions to the information matrix from each element of the output vector, using signal-to-noise ratio for the modeling problem, roughly speaking.

A common reason for an ill-conditioned information matrix is having too many unknown model parameters, also known as overparameterization. When this happens, the information content in the data can be too low for the number of estimated parameters required, so that the estimator cannot produce results with sufficient accuracy, or the estimation process can fail completely. Another cause is misspecification of the model, where changes in more than one model parameter produce nearly equivalent changes in the outputs, or perhaps one or more model parameters have little or no effect on the outputs. A third cause is insufficient information content in the data, where there is so little movement in the outputs that it appears that one or more parameters have little or no effect on the outputs; i.e., the corresponding output sensitivities are near zero. Another form of this problem occurs when a model parameter is associated with a quantity that is constant or nearly constant, which causes the associated parameter to be confounded with the bias parameter. The latter problem could also be classified as model misspecification.

There are two main consequences of a nearly singular information matrix:

- 1) The information matrix  $\mathbf{M}$  can be negative definite, resulting in a cost increase  $J(\hat{\boldsymbol{\theta}}_{k+1}) > J(\hat{\boldsymbol{\theta}}_k)$  for the modified Newton-Raphson step.
- 2) The modified Newton-Raphson algorithm step size  $\Delta \hat{\boldsymbol{\theta}}$  in one or more directions can be large.

The second problem can be seen from the decomposition of the inverse  $\mathbf{M}$  matrix as

$$\mathbf{M}^{-1} = \sum_{j=1}^{np} \frac{1}{\lambda_j} \mathbf{t}_j \mathbf{t}_j^T \quad (6.54)$$

as already discussed in Sec. 5.5.1. In Eq. (6.54),  $\lambda_j$  are the eigenvalues of  $\mathbf{M}$  corresponding to the eigenvectors  $\mathbf{t}_j$ . Then the modified Newton-Raphson

step in the direction of  $\mathbf{t}_j$  is found by combining the previous expression with Eq. (6.39b),

$$\Delta \hat{\boldsymbol{\theta}}_{\lambda_j} = -\frac{1}{\lambda_j} (\mathbf{t}_j^T \mathbf{g}) \mathbf{t}_j \quad (6.55)$$

This step will be large for small  $\lambda_j$ , in a direction  $\mathbf{t}_j$  for which little information is available. The parameter step  $\Delta \hat{\boldsymbol{\theta}}$  is therefore dominated by bad information, resulting in a parameter vector update that does not approach the minimum of the cost function.

There are several techniques for dealing with an ill-conditioned information matrix. Three of them associated with the modified Newton-Raphson method will be described here.

*Rank deficient method.* The rank deficient method is based on the reduced-order inverse of  $\mathbf{M}$  using the singular value decomposition (SVD),

$$\mathbf{M} = \mathbf{U} \mathbf{D} \mathbf{T}^T \quad (6.56)$$

discussed in Chapter 5. The formula for the reduced-order inverse is

$$\begin{aligned} \mathbf{M}^{-1} &= \mathbf{T} \begin{bmatrix} 1/\mu_1 & & & \\ & 1/\mu_2 & & 0 \\ & & \ddots & \\ & & & 1/\mu_{n_p-m} \\ & 0 & & 0 \\ & & & \ddots \\ & & & 0 \end{bmatrix} \mathbf{U}^T \\ &= \sum_{j=1}^{n_p-m} \frac{1}{\mu_j} \mathbf{t}_j \mathbf{u}_j^T \end{aligned} \quad (6.57)$$

where  $\mathbf{t}_j$  and  $\mathbf{u}_j$  are the  $j$ th columns of the  $\mathbf{T}$  and  $\mathbf{U}$  matrices, respectively, and the singular values are arranged from largest to smallest in magnitude,

$$\mu_1 > \mu_2 > \dots > \mu_{n_p-m} > \mu_{n_p-m+1} > \dots > \mu_{n_p} \quad (6.58)$$

The terms associated with the smallest  $m$  singular values are dropped in computing the matrix inverse. The criterion is to drop any singular value for which

$$\frac{\mu_j}{\mu_{\max}} < N\varepsilon \quad (6.59)$$

## MAXIMUM LIKELIHOOD METHODS

207

where  $N$  is the number of data points, and  $\varepsilon$  is the computing machine precision. This implements the reduced-order matrix inverse and gives the inverse of  $\mathbf{M}$  with rank reduced from  $n_p$  to  $n_p - m$ . The rank deficient method is, in principle, similar to principal component regression for reducing the adverse effects of data collinearity.<sup>20,21</sup>

*Levenberg-Marquardt method.* The Levenberg-Marquardt<sup>22</sup> method augments the information matrix to improve its conditioning and thereby produce a more reasonable inverse. The formula for the matrix inverse is

$$\mathbf{M}^{-1} = (\mathbf{M}_o + k\mathbf{A})^{-1} \quad (6.60)$$

where  $\mathbf{M}_o$  is the original information matrix, which is presumably ill conditioned,  $k$  is a positive nonzero scalar parameter, and  $\mathbf{A}$  is a positive definite matrix. Usually  $\mathbf{A}$  is taken as the identity matrix, so

$$\mathbf{M}^{-1} = (\mathbf{M}_o + k\mathbf{I})^{-1} \quad (6.61)$$

The scalar  $k$  can be obtained by an iterative procedure proposed by Theil.<sup>21</sup> The initial value for  $k$  is recommended as  $k = 0.01$ . As  $k$  increases, the Levenberg-Marquardt inverse of the information matrix causes the modified Newton-Raphson step to follow the cost gradient vector more closely [cf. Eq. (6.39b)]. The Levenberg-Marquardt method resembles the ridge regression method mentioned in Sec. 5.5.3.

*Bayes-like method.* The Bayes-like estimation method improves the conditioning of the  $\mathbf{M}$  matrix by combining the measured data with prior estimates of some or all of the unknown parameters in the model. The cost function for the parameter estimation is similar to that for the Bayesian estimator [cf. Eq. (4.28)],

$$J(\boldsymbol{\theta}) = \frac{1}{2} \sum_{i=1}^N \mathbf{v}^T(i) \mathbf{R}^{-1} \mathbf{v}(i) + \frac{1}{2} (\boldsymbol{\theta} - \boldsymbol{\theta}_p)^T \boldsymbol{\Sigma}_p^{-1} (\boldsymbol{\theta} - \boldsymbol{\theta}_p) \quad (6.62)$$

where  $\boldsymbol{\theta}_p$  is the vector of prior parameter values, and  $\boldsymbol{\Sigma}_p$  is a positive semidefinite covariance matrix that reflects the confidence in the parameter values  $\boldsymbol{\theta}_p$ .

Using the modified Newton-Raphson minimization technique, the estimate of  $\Delta\boldsymbol{\theta}$  is computed from

$$\Delta\hat{\boldsymbol{\theta}} = -(\mathbf{M}_{\boldsymbol{\theta}=\boldsymbol{\theta}_o} + \boldsymbol{\Sigma}_p^{-1})^{-1} [\mathbf{g} + \boldsymbol{\Sigma}_p^{-1}(\boldsymbol{\theta} - \boldsymbol{\theta}_p)]_{\boldsymbol{\theta}=\boldsymbol{\theta}_o} \quad (6.63)$$

This estimator is similar to the mixed estimator introduced in Sec. 5.5.3, although its development followed a different path. The  $\boldsymbol{\Sigma}_p$  matrix is usually diagonal, with

each diagonal element representing the variance of the corresponding element of  $\boldsymbol{\theta}_p$ .

From Eq. (6.63), it can be seen that prior values can help particularly with parameters for which there is little or no information from the data. In a sense, the prior information can fill information deficiencies in the measured data, and thereby regularize the matrix inversion. This is a good example of a general principle in modeling, which is that information about the model parameters is either supplied to the estimator by some mechanism like prior estimates and their variances, or extracted from the measured data.

The estimates obtained from all three techniques just mentioned will be biased; however, in practical cases, the bias is small in comparison with the inaccuracy that would be introduced if  $\mathbf{M}$  was nearly singular and nothing was done to regularize the estimation.

In some cases, the information matrix  $\mathbf{M}$  is nearly singular only at certain points in the progression toward a solution using the modified Newton-Raphson method. In those cases, the methods described earlier can be used temporarily to get past the difficult points, and then omitted as the sequence of parameter estimates  $\boldsymbol{\theta}_k$  approaches the solution.

The rank deficiency method described earlier is easy to use in this way. To implement the approach, the inverse of the information matrix is always calculated using the SVD (which is a superior method for calculating the inverse anyway<sup>23</sup>), and the singular value ratios are checked against the criterion in Eq. (6.59). If any singular values are too small, their corresponding terms are dropped from the inverse using Eq. (6.57), and the modified Newton-Raphson method proceeds in the usual way. This approach is a simple method for addressing the near-singularity of the information matrix and works well in practice.

### 6.3.3 Accuracy of Parameter Estimates

In the last chapter, the focus was on linear unbiased estimators and their accuracy. The accuracy of a parameter estimate was determined by the computed variance or standard error. With the assumption of Gaussian measurement noise, it was possible to construct a confidence interval for each parameter or a confidence region for more than one parameter.

For a nonlinear estimator, the evaluation of parameter accuracy is much more difficult. As in the linear estimation problem, for a finite number of data samples, the parameter estimates are biased. Because of the nonlinear dependence of the model outputs on the parameters, the parameter covariance matrix represents only an approximation to the lower bounds for the parameter covariances. Furthermore, normality of the measurement noise does not guarantee normal distribution of the parameter estimates, because the parameter estimates are a nonlinear function of the measurements. Nevertheless, the usual statement regarding accuracy of the parameters for a nonlinear maximum likelihood estimator quotes the Cramér-Rao lower bound on the parameter variances, which are obtained from the main diagonal of the inverse information matrix [cf. Eq. (6.39c)].

Practical experience with aircraft parameter estimation reveals large discrepancies between the Cramér-Rao lower bound and the ensemble variance of

parameter estimates obtained from analysis of repeated maneuvers at the same flight condition.<sup>24–26</sup> On the other hand, for simulated data, the Cramér-Rao lower bound has been found to be a very good estimator for the variance of an ensemble of parameter estimates.

Several approaches have been developed to explain the discrepancy: 1) developing a technique for determining bias and mean square error for an arbitrary parameter estimator, 2) changing the assumptions about the measurement noise, and 3) computing confidence intervals by considering the likelihood function for the model, rather than its quadratic approximation.

A technique for determining accuracy for a general parameter estimator was proposed by Gupta.<sup>27</sup> The technique is based on higher-order sensitivity analysis of an estimator for a general dynamic system. First- and second-order approximations of the expectation  $E(\mathbf{Z}_N; \boldsymbol{\theta})$  resulted in expressions for biases and covariance. When applied to a maximum likelihood estimator, it was found that, to first order, the bias and covariance are obtained as

$$\begin{aligned} \mathbf{b} &\equiv E(\hat{\boldsymbol{\theta}}) - \boldsymbol{\theta} = \mathbf{0} \\ E[(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})^T] &= \left[ \sum_{i=1}^N \frac{\partial \mathbf{v}^T(i)}{\partial \boldsymbol{\theta}} \mathbf{R}^{-1} \frac{\partial \mathbf{v}(i)}{\partial \boldsymbol{\theta}} \right]^{-1} \end{aligned} \quad (6.64)$$

These results are identical to those obtained using the modified Newton-Raphson algorithm. For better assessment of the parameter accuracy, second-order expressions for the biases and covariance have to be computed. The expressions for these quantities become very complex, requiring computation of the second-order gradient  $\partial^2 \mathbf{v}(i)/\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T$ . The sensitivity analysis provides a very general tool that can, in theory, be used in the error analysis for any estimator, including the effects of any misspecification of the postulated model. Unfortunately, the complexity of the analysis makes this approach impractical for the multi-dimensional dynamic systems used to model aircraft dynamics.

In the development of the maximum likelihood estimator, it was assumed that the measurement noise  $\mathbf{v}(i)$  came from a random white sequence that was zero mean and Gaussian. This means that the power of the noise should be evenly spread over the frequency range  $[0, f_N]$  Hz, where  $f_N = 1/(2\Delta t)$  is the Nyquist frequency (see Chapter 9). The relation between the one-sided power spectral density of the noise vector and the noise covariance matrix is therefore

$$\mathbf{R} = \text{PSD } f_N = \frac{\text{PSD}}{2\Delta t} \quad (6.65)$$

where **PSD** is a diagonal matrix with the power spectral densities of the elements of the noise vector on the main diagonal. From before, the Cramér-Rao inequality for the parameter covariance matrix of the output-error estimator is

$$\text{Cov}(\hat{\boldsymbol{\theta}}) \geq \left[ \sum_{i=1}^N \mathbf{S}^T(i) \mathbf{R}^{-1} \mathbf{S}(i) \right]^{-1} \quad (6.66)$$

or, in terms of the power spectral density,

$$\text{Cov}(\hat{\theta}) \geq \frac{1}{2\Delta t} \left[ \sum_{i=1}^N S^T(i) (\text{PSD})^{-1} S(i) \right]^{-1} \quad (6.67)$$

In many practical cases, analysis of the residuals from output-error estimation indicates that the residuals are colored. This means that the power spectral densities of the residuals show more power concentrated in a particular frequency band, compared with the relatively constant power out to the Nyquist frequency that is characteristic of white noise. For aircraft parameter estimation, this concentration of noise power occurs in the low frequency band corresponding to the rigid-body aircraft dynamics. The phenomenon is usually caused by inadequacies in the proposed model structure. For example, it could be that the true model parameters are not really constant, or that the data include some nonlinear effects not accounted for by a proposed linear model structure.

The theoretical development of the maximum likelihood estimator assumes that the residuals are white, because the power spectral densities of the residuals are not known *a priori*, and using the assumption of white residuals makes the theoretical development more straightforward. The assumption of white residuals is equivalent to assuming that the proposed model structure is completely adequate to characterize the measured data. In simulation, this assumption is perfectly true, because the same model structure is used both to generate the data and to analyze it, and the added noise sequence is typically white and Gaussian. For this case, the application matches the theoretical development, and the usual calculation of the parameter variances provides an excellent measure of the parameter accuracy.

To handle the colored residuals encountered in practice, Balakrishnan and Maine<sup>28</sup> suggested approximating the colored noise with band-limited white noise of the same power. The approximation is depicted graphically in Fig. 6.5 for one element of the residual vector, where the one-sided power spectral density of wide-band white noise is included for comparison.

For the power spectral density of band-limited white noise with one-sided bandwidth  $B$ , the analog of Eq. (6.65) is

$$R = \text{PSD} \cdot B \quad (6.68a)$$

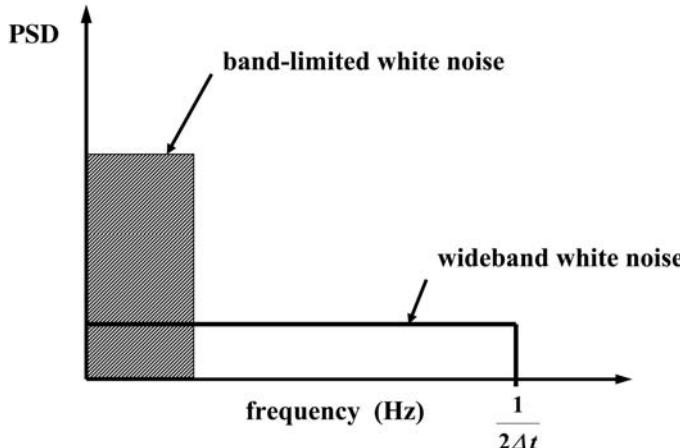
or

$$\frac{R}{2\Delta t B} = \frac{\text{PSD}}{2\Delta t} \quad (6.68b)$$

Thus, Eq. (6.67) changes to

$$\text{Cov}(\hat{\theta}) \geq \frac{1}{2\Delta t B} \left[ \sum_{i=1}^N S^T(i) R^{-1} S(i) \right]^{-1} \quad (6.69)$$

which reduces to Eq. (6.66) when  $B = f_N = 1/(2\Delta t)$ . Expression (6.69) means that for colored noise, the Cramér-Rao bounds can be underestimated by a factor of  $1/(2\Delta t B) = f_N/B$ .



**Fig. 6.5** Approximation of colored noise with band-limited white noise.

For example, if the bandwidth of the rigid-body dynamics and the colored residuals is 1 Hz, and the sampling rate is 50 Hz, then  $f_N = 25$  Hz, and the Cramér-Rao bounds for the parameter covariance matrix elements would be too low by a factor of 25. This translates into parameter standard error bounds that are too low by a factor of 5. These results provide an explanation for the discrepancies between Cramér-Rao bound estimates computed from simulation versus flight data, and between Cramér-Rao bounds computed from a single flight maneuver versus an ensemble variance estimate computed from repeated maneuvers. More details can be found in Refs. 25 and 28.

Further improvement in the Cramér-Rao bound estimate can be obtained by replacing the white residual assumption by assuming the residuals are colored, and including an estimate of the residual coloring in the expression for the Cramér-Rao bound estimate. This approach has already been mentioned in Sec. 5.2 for the least-squares estimator. In Ref. 29, the approach was generalized for the output-error method, resulting in an expression for the parameter covariance matrix for colored residuals,

$$\text{Cov}(\hat{\boldsymbol{\theta}}) \geq \mathbf{M}^{-1} \left[ \sum_{i=1}^N \mathbf{S}^T(i) \mathbf{R}^{-1} \sum_{j=1}^N \mathcal{R}_{vv}(i-j) \mathbf{R}^{-1} \mathbf{S}(j) \right] \mathbf{M}^{-1} \quad (6.70)$$

where  $\mathcal{R}_{vv}(i-j)$  is the autocorrelation matrix for the output residual vector. This quantity can be estimated using

$$\hat{\mathcal{R}}_{vv}(k) = \frac{1}{N} \sum_{i=1}^{N-k} \mathbf{v}(i) \mathbf{v}^T(i+k) = \hat{\mathcal{R}}_{vv}(-k) \quad k = 0, 1, 2, \dots, r \quad (6.71)$$

The estimate in Eq. (6.71) usually only includes the diagonal terms, since the off-diagonal terms are close to zero for the same reasons given in the discussion

of estimating the noise covariance matrix  $\mathbf{R}$  from Eq. (6.29). The value of  $r$  in Eq. (6.71) determines the maximum time difference used in the computation of the residual autocorrelation estimate. Small values of  $r$  correspond to computing residual coloring at low frequency. Since the residual coloring that occurs in practice is at low frequencies associated with the rigid-body dynamic modes, the value of  $r$  need not run all the way up to  $N - 1$ , but instead can be stopped at about  $N/5$  for typical flight-test maneuvers and sample rates. This reduces the calculations required and has no discernable effect on the quality of the results.

The values for  $\mathbf{S}$ ,  $\mathbf{M}$ , and  $\mathbf{R}$  in Eq. (6.70) are from the conventional maximum likelihood estimation calculations, so they are the same as the quantities appearing in Eqs. (6.29), (6.40), and (6.66). Consequently, Eqs. (6.70) and (6.71) can be used as a residual postprocessor after conventional maximum likelihood parameter estimation. The residual postprocessing corrects the parameter covariance matrix to account for colored residuals. Parameter variances computed from Eqs. (6.70) and (6.71) have been shown to be consistent with ensemble estimates of the variance of parameter estimates from repeated maneuvers, using simulated data with many different realistic colored noise sequences, as well as flight data.<sup>26,29</sup>

For a linear unbiased estimate, the parameter confidence interval can be easily obtained, and is symmetric about the value of the estimated parameter. The confidence level associated with this interval expresses the probability of finding the parameter true value within the interval. The confidence interval for a single parameter can be extended to a parameter confidence region for multiple parameters, defined by a confidence ellipsoid.<sup>25</sup>

For nonlinear parameter estimation, the confidence ellipsoid is only an approximation to the confidence region. This region could have an irregular shape quite different from an ellipsoid. Furthermore, because the parameter estimates in a nonlinear case do not have Gaussian distribution, the confidence level based on the  $F$ -distribution is again an approximation.

A method for finding confidence intervals for parameter estimates obtained by maximum likelihood estimation was presented in Ref. 19 for both linear and nonlinear estimation cases. The construction of the confidence interval for a one-dimensional case is shown in Fig. 6.6. The confidence interval for a linear estimate is given by the interval for which the change in the maximum likelihood cost function is a constant selected value  $\Delta J_L$ ,

$$\Delta J_L = n_p F(\alpha; n_p, N - n_p) \quad (6.72)$$

where  $F(\alpha; n_p, N - n_p)$  is the value of the  $F$ -distribution with confidence level  $\alpha$ . For the nonlinear estimate,

$$\Delta J_{NL} = n_p F(\alpha; n_p, N - n_p) \left[ 1 - \frac{N(n_p + 2)}{(N - n_p)n_p} N_\phi \right] \quad (6.73)$$

where  $N_\phi$  is the nondimensional intrinsic nonlinearity measure introduced by Beale.<sup>30</sup>

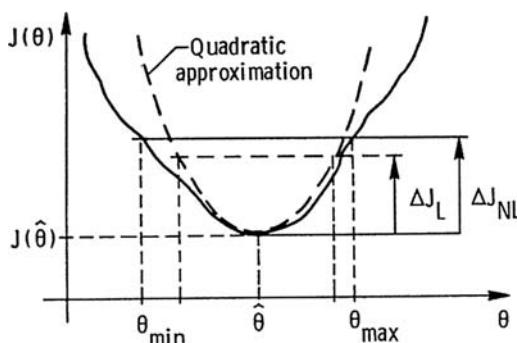


Fig. 6.6 Confidence interval determination by cost increments.<sup>19</sup>

For computing the limits of the confidence interval  $\hat{\theta}_{\min}$  and  $\hat{\theta}_{\max}$ , the search algorithm developed by Mereau and Raymond<sup>31</sup> can be used. This algorithm finds the contour boundaries of the cost function by testing a series of randomly selected points in and around the confidence region. Through many iterations, the limits of the confidence region can be determined, and confidence intervals for each of the parameters are obtained. This method has not been widely used, probably because the search algorithm is computationally very demanding, even for a model with a small number of parameters.

### Example 6.2

In this example, data from Example 6.1 are used to demonstrate the estimated parameter covariance corrections for colored residuals. The method is similar to that shown in Example 5.2 for linear regression, with the main difference being that the number of model outputs for the regression case was one, whereas the number of model outputs for the output-error case is generally more than one.

Figure 6.7 shows the model fit to the measured output data, with model parameters estimated using output-error method. The same model structure as before was used [cf. Eqs. (6.52) and (6.53)]. In this case, however, the output-error parameter estimation was done using the measured input-output data directly; i.e., the measured time series were not converted to perturbations about initial trim values. This can be seen by comparing the initial values shown in Figs. 6.3 and 6.7, particularly for  $a_y$ . The output-error parameter estimation results are identical to those found in Example 6.1, except for the bias parameters.

Figure 6.8 shows the output-error residuals, which are the difference between the data and model time series shown in Fig. 6.7. The output-error residuals are obviously colored, because they do not resemble random sequences. Table 6.2 contains the output-error parameter estimation results, including estimated

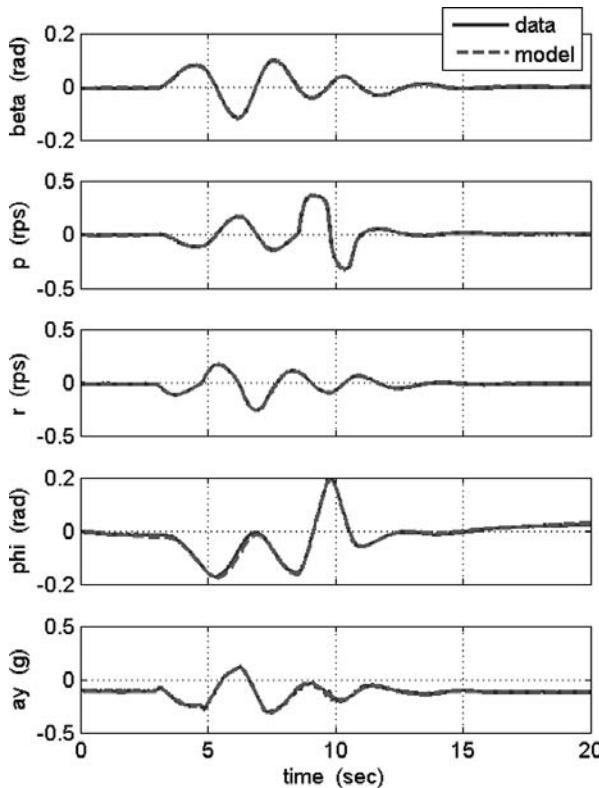


Fig. 6.7 Output-error model fit to measured outputs for lateral maneuver, run 1.

parameter standard errors using both the white residual assumption [Eq. (6.66)], and accounting for colored residuals [Eq. (6.70)]. In Fig. 6.9, equation-error parameter estimates from Examples 5.1 and 5.2 are plotted along with output-error parameter estimates based on the same data, using the same model structure. The error bounds shown on all the parameter estimates are 95% confidence intervals, accounting for colored residuals. The figure shows that the equation-error and output-error parameter estimates are in good agreement.

The equation-error estimates are biased as a result of the regressors being noisy, (and not deterministic, as assumed in the theory), as well as from any deterministic model structure error, which should be very small for this perturbation maneuver at low angle of attack. The bias in the output-error parameter estimates due to deterministic model structure error should likewise be small. Estimates from either method could be biased because of the finite number of measurements, since in theory the estimation methods are only asymptotically unbiased. More accurate parameter estimates can be obtained by averaging the estimates from repeated maneuvers at the same flight condition.

## MAXIMUM LIKELIHOOD METHODS

215

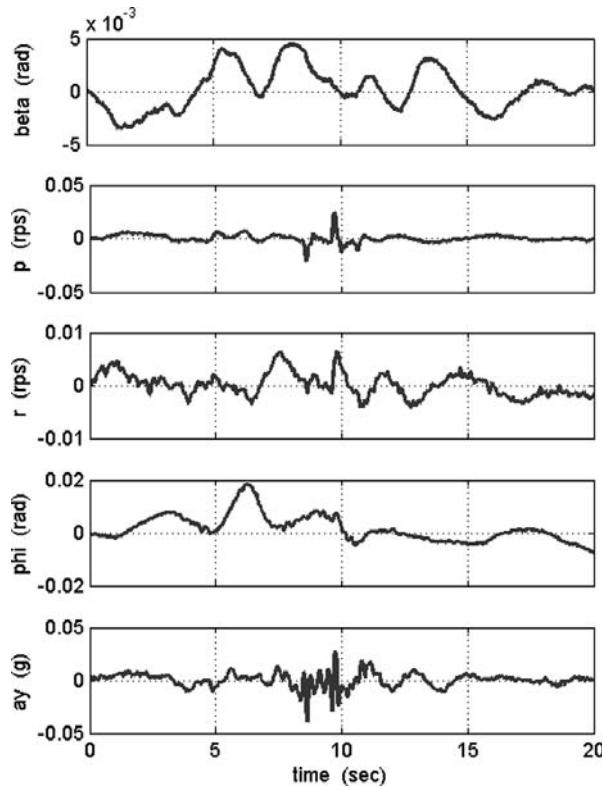


Fig. 6.8 Output-error residuals for lateral maneuver, run 1.

Table 6.2 Output-error parameter estimation results for lateral maneuver, run 1

Parameter	$\hat{\theta}$	$s(\hat{\theta})$ white residual Eq. (6.66)	$s(\hat{\theta})_{\text{corr}}$ colored residual Eq. (6.70)	$\frac{s(\hat{\theta})_{\text{corr}}}{s(\hat{\theta})}$
$C_{Y\beta}$	$-8.66 \times 10^{-1}$	$2.80 \times 10^{-3}$	$8.82 \times 10^{-3}$	3.15
$C_{Y_r}$	$9.31 \times 10^{-1}$	$1.65 \times 10^{-2}$	$6.04 \times 10^{-2}$	3.65
$C_{Y\delta_r}$	$3.75 \times 10^{-1}$	$7.98 \times 10^{-3}$	$3.15 \times 10^{-2}$	3.95
$C_{l\beta}$	$-1.19 \times 10^{-1}$	$5.60 \times 10^{-4}$	$1.89 \times 10^{-3}$	3.37
$C_{l_p}$	$-5.84 \times 10^{-1}$	$2.39 \times 10^{-3}$	$8.23 \times 10^{-3}$	3.44
$C_{l_r}$	$1.88 \times 10^{-1}$	$1.71 \times 10^{-3}$	$8.89 \times 10^{-3}$	5.19
$C_{l\delta_a}$	$-2.28 \times 10^{-1}$	$8.43 \times 10^{-4}$	$3.03 \times 10^{-3}$	3.60
$C_{l\delta_r}$	$3.84 \times 10^{-2}$	$7.12 \times 10^{-4}$	$6.03 \times 10^{-3}$	8.47
$C_{n\beta}$	$8.65 \times 10^{-2}$	$2.07 \times 10^{-4}$	$1.53 \times 10^{-3}$	7.36
$C_{n_p}$	$-6.39 \times 10^{-2}$	$9.49 \times 10^{-4}$	$6.41 \times 10^{-3}$	6.76
$C_{n_r}$	$-1.92 \times 10^{-1}$	$6.07 \times 10^{-4}$	$4.06 \times 10^{-3}$	6.68
$C_{n\delta_a}$	$-2.73 \times 10^{-3}$	$3.50 \times 10^{-4}$	$2.23 \times 10^{-3}$	6.35
$C_{n\delta_r}$	$-1.36 \times 10^{-1}$	$2.66 \times 10^{-4}$	$1.86 \times 10^{-3}$	7.00

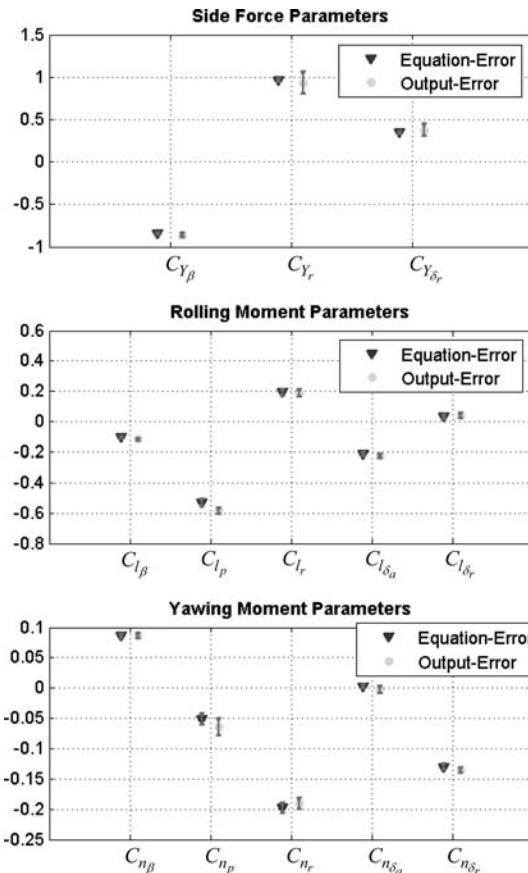


Fig. 6.9 Comparison of parameter estimates for lateral maneuver, run 1.

#### 6.4 Equation-Error Method

Another variation of the maximum likelihood parameter estimation method is obtained by assuming that 1) the dynamic model is in linear state-space form with process noise, 2) the model parameters to be estimated are dimensional stability and control derivatives, and 3) all the states are measured without error, in addition to the inputs.

Under the preceding assumptions, the model equations are formed as

$$\dot{\mathbf{x}}(t) = \mathbf{Ax}(t) + \mathbf{Bu}(t) \quad \mathbf{x}(0) = \mathbf{x}_o \quad (6.74a)$$

$$\mathbf{z}(i) = \mathbf{Ax}(i) + \mathbf{Bu}(i) + \boldsymbol{\nu}(i) \quad i = 1, 2, \dots, N \quad (6.74b)$$

Note that the equation error  $\boldsymbol{\nu}(i)$  can include measurement errors for the state derivative, as well as process noise that may be present in the state equations.

## MAXIMUM LIKELIHOOD METHODS

217

Typically, the state derivatives are not measured directly but instead are found from smoothed numerical differentiation of the measured states (see Chapter 11). If the equation errors  $\mathbf{v}(i)$  are assumed to be Gaussian with

$$E[\mathbf{v}(i)] = \mathbf{0} \quad E[\mathbf{v}(i)\mathbf{v}^T(j)] = \mathbf{R}\delta_{ij} \quad (6.75)$$

then the negative log-likelihood function has the form

$$\begin{aligned} -\ell_n \mathbb{L}(\mathbf{Z}_N; \boldsymbol{\theta}) &= \frac{1}{2} \sum_{i=1}^N [\mathbf{z}(i) - \mathbf{Ax}(i) - \mathbf{Bu}(i)]^T \mathbf{R}^{-1} [\mathbf{z}(i) - \mathbf{Ax}(i) - \mathbf{Bu}(i)] \\ &\quad + \frac{N}{2} \ln |\mathbf{R}| + \frac{Nn_o}{2} \ln(2\pi) \end{aligned} \quad (6.76)$$

For known  $\hat{\mathbf{R}}$ , the cost function to be minimized is

$$J(\boldsymbol{\theta}) = \frac{1}{2} \sum_{i=1}^N [\mathbf{z}(i) - \mathbf{Ax}(i) - \mathbf{Bu}(i)]^T \hat{\mathbf{R}}^{-1} [\mathbf{z}(i) - \mathbf{Ax}(i) - \mathbf{Bu}(i)] \quad (6.77)$$

The measurement equation (6.74b) and the cost function in Eq. (6.77) indicate that the parameter estimation has been simplified to that of linear regression, where the innovations are the equation errors,

$$\mathbf{v}(i) = \mathbf{z}(i) - \mathbf{Ax}(i) - \mathbf{Bu}(i) \quad i = 1, 2, \dots, N \quad (6.78)$$

When the noise covariance matrix estimate  $\hat{\mathbf{R}}$  is diagonal, as is the usual case, minimizing the cost function in Eq. (6.77) can be done by minimizing the cost associated with each individual row in the state-space model. Therefore, the unknown parameters in matrices  $\mathbf{A}$  and  $\mathbf{B}$  can be estimated in subsets corresponding to their individual equation in the state-space model; e.g., the lift parameters can be estimated from the  $\dot{x}$  equation, the pitching moment parameters from the  $\dot{q}$  equation, and so on.

It is just as simple to use output equations instead of dynamic equations for this approach. That is, Eq. (6.74b) can be changed to use measured output equations in the state-space formulation as follows:

$$\mathbf{z}(i) = \mathbf{Cx}(i) + \mathbf{Du}(i) + \mathbf{v}(i) \quad i = 1, 2, \dots, N \quad (6.79)$$

Elements in the relevant rows of the  $\mathbf{C}$  and  $\mathbf{D}$  matrices must contain model parameters to be estimated. Consequently, the measurement equations for translational accelerometers can be used in this way. A block diagram of the equation-error method is shown in Fig. 6.10.

In any practical situation, the measured states and inputs are corrupted by errors. Regression of the elements of  $\dot{x}$  or  $z$  on measured  $x$  and  $u$  will then result in biased estimates<sup>32</sup> of the unknown model parameters in  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$ , and  $\mathbf{D}$ .

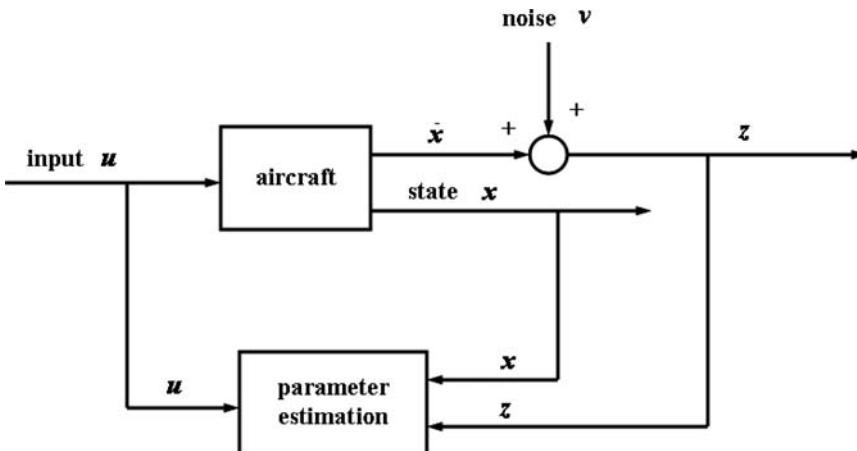


Fig. 6.10 Block diagram for equation-error parameter estimation.

### Example 6.3

Measured flight-test data for this example comes from a longitudinal perturbation maneuver of the NASA Twin Otter aircraft, pictured in Fig. 5.3. Figure 6.11 shows measured input-output data for the maneuver. The equation-error method described in the previous section was used to estimate dimensional stability and control derivatives.

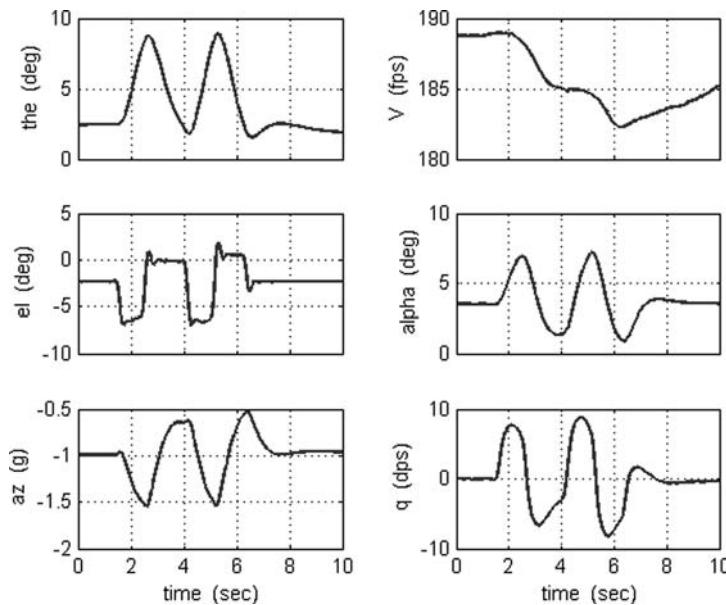


Fig. 6.11 Measured input and output variables for a longitudinal maneuver.

## MAXIMUM LIKELIHOOD METHODS

219

The linearized longitudinal equations in state-space form are [cf. Eqs. (3.128)],

$$\dot{\alpha} = Z_\alpha \alpha + q + Z_{\delta_e} \delta_e \quad (6.80a)$$

$$\dot{q} = M_\alpha \alpha + M_q q + M_{\delta_e} \delta_e \quad (6.80b)$$

$$a_z = \frac{V_o}{g} (Z_\alpha \alpha + Z_{\delta_e} \delta_e) \quad (6.80c)$$

where all states and controls are perturbation quantities, which is consistent with the linear model structure. The maneuver was executed at a low trim angle of attack, so the lift force is nearly along the  $z$  body axis, which allows use of the form shown for the  $\dot{\alpha}$  equation. At low angles of attack,  $Z_q \approx 0$ , so the associated term does not appear in Eqs. (6.80a) and (6.80c). The stepwise regression technique described in Chapter 5 was used to verify that the  $Z_q$  term should not be included in the model structure.

The dynamic equation for angle of attack  $\alpha$  is typically dominated by two terms: the inertial term  $q$  and the  $Z_\alpha \alpha$  term. The output equation for  $z$  body-axis acceleration  $a_z$  does not include the inertial term, so all of the information in the  $a_z$  measurement can be used for aerodynamic parameter estimation. In addition, the  $a_z$  measurement is directly applicable to estimating the dimensional parameters associated with  $Z$ , whereas using the dynamic equation for  $\alpha$  involves an additional approximation, namely,  $-L = Z \cos \alpha \approx Z$  [see the discussion following Eqs. (3.126)]. As a result, the output equation for  $a_z$  usually gives better estimates for the aerodynamic parameters associated with the body-axis force  $Z$ . The output equation for  $a_z$ , Eq. (6.80c), was used to estimate the  $Z$  aerodynamic force parameters, and the dynamic equation for pitch rate  $q$ , Eq. (6.80b), was used to estimate the pitching moment parameters. This also means that only  $\dot{q}$  needed to be computed numerically, and not  $\dot{\alpha}$ . The state derivative  $\dot{q}$  was computed using smoothed local numerical differentiation described in Chapter 11.

Table 6.3 gives the equation-error estimates of the dimensional stability and control derivatives found by minimizing the cost function of Eq. (6.77) twice—once for the  $Z$  force parameters using Eq. (6.80c) and once for the pitching moment parameters using Eq. (6.80b). Parameter estimates and

**Table 6.3** Equation-error dimensional parameter estimation results for a longitudinal maneuver

Parameter	$\hat{\theta}$	$s(\hat{\theta})$	$ t_0 $	$100[s(\hat{\theta})]/ \hat{\theta} $
$Z_\alpha$	-1.589	0.026	61.5	1.6
$Z_{\delta_e}$	-0.038	0.016	2.4	41.2
$Z_o$	-0.406	0.014	29.9	3.4
$M_\alpha$	-5.245	0.050	104.8	1.0
$M_q$	-2.598	0.047	55.4	1.8
$M_{\delta_e}$	-7.852	0.082	95.6	1.0
$M_o$	-0.003	0.006	0.6	165.1

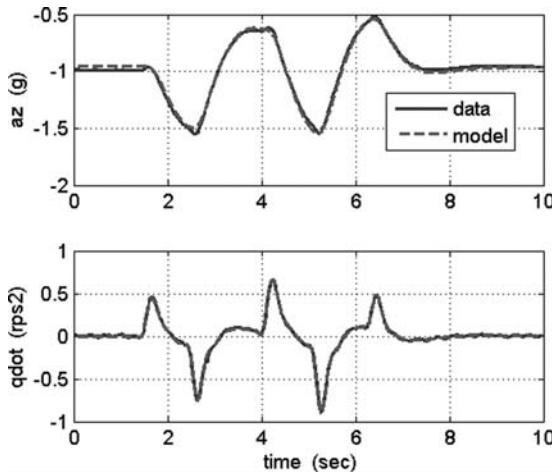


Fig. 6.12 Equation-error model fit to  $a_z$  and  $\dot{q}$  for a longitudinal maneuver.

associated error bounds were found in the usual manner for linear regression problems, as described in Chapter 5. Estimated parameter error bounds were corrected for colored residuals in linear regression problems, as in Chapter 5. The measured outputs to be fit by the models were  $a_z$  and  $\dot{q}$ , respectively. Figure 6.12 shows the model fit to the data, using estimated parameters from Table 6.3. The parameter estimates given in Table 6.3 are good, useful estimates and also provide excellent starting values for the iterative output-error method. The dimensional derivative estimates in Table 6.3 can be nondimensionalized using Eqs. (3.129), if necessary.  $\square$

Finally, the expressions in Eq. (3.129) show that the estimated dimensional parameters include the dynamic pressure  $\bar{q} = \rho V^2/2$  and airspeed  $V$ . Because of this, changes in airspeed and altitude must be relatively small during the maneuver used for estimating dimensional stability and control derivatives. This applies to any method where dimensional stability and control derivatives are being estimated. The reason is that the model parameters are assumed to be constants in the analysis, but dimensional parameters will vary with airspeed and altitude changes, because the dimensional derivatives include dynamic pressure and airspeed. Similar statements apply for the mass/inertia properties, but these quantities rarely vary significantly over the time period typical of a stability and control maneuver. A good rule of thumb is that dimensional parameters can be used when the airspeed variation is less than 10% of the mean value. Maximum allowable variation in altitude can be defined as the altitude change that induces a similar change in the magnitude of the dynamic pressure.

## 6.5 Summary and Concluding Remarks

The maximum likelihood method for parameter estimation is based on a relatively simple idea: assuming the outcome of an experiment depends on unknown

parameters, find the values of those parameters that make the measured values the most likely to occur. This concept is realized by maximization of a likelihood function, which equals the probability density function for measured values, given the parameters.

In aircraft applications, the measured data are time series of input and output variables for a nonlinear dynamic system with process noise. It is usually assumed that the inputs are measured without errors, and the measured outputs are corrupted by measurement noise that is Gaussian, white, and zero mean. The cost function for parameter estimation includes innovations weighted by their covariance matrix. The innovations are the differences between measured and computed outputs, where the computed outputs are based on states estimated using an extended Kalman filter.

The parameter estimation as stated represents a difficult optimization problem. Several optimization schemes are available, but the modified Newton-Raphson method is usually used in practice. A simplification of the estimation problem is achieved when the dynamic system model is linear. Then the extended Kalman filter is replaced by a Kalman filter, or its simplified version, the steady-state Kalman filter. The estimation includes both the state of the dynamic system and the model parameters. This approach is called the filter-error method.

The resulting parameter estimates have favorable statistical properties: they are asymptotically unbiased, efficient, consistent, and normally distributed. Despite the generality of the estimation problem, this full version of the maximum likelihood estimator is rarely used in practice because of the complexity of the algorithm and possible identifiability problems caused by low information content relative to the number of unknowns to be estimated. Good treatments of the maximum likelihood method with examples can be found in reports by Stepner and Mehra,<sup>4</sup> Hall et al.,<sup>33</sup> Jategaonkar and Plaetschke,<sup>12,34</sup> and Maine and Iliff.<sup>5</sup>

Substantial simplification of the algorithm for the maximum likelihood method results from assuming there is no process noise. Then the state estimation reduces to integration of the state equations, which are deterministic. The maximum likelihood method becomes the output-error method, which is, in principle, identical to a nonlinear regression. Excellent explanation of the output-error method with many examples is given by Maine and Iliff.<sup>35</sup> Further, if the states are also assumed to be measured without errors, the maximum likelihood method reduces to a linear regression.

Because the output-error method is widely used in aircraft parameter estimation, some practical aspects of computing the estimates were presented. These include computing sensitivities, inverting a nearly singular information matrix, and estimating parameter accuracy. Several examples were included to demonstrate applications to aircraft modeling problems.

## References

<sup>1</sup>Kailath, T., "A General Likelihood-Ratio Formula for Random Signals in Gaussian Noise," *IEEE Transactions on Information Theory*, Vol. IT-15, No. 3, 1969, pp. 350–361.

<sup>2</sup>Taylor, L. W., Iliff, K. W., and Powers, B. G., "A Comparison of Newton-Raphson and Other Methods for Determining Stability Derivatives from Flight Data," AIAA Paper 69-315, March 1969.

<sup>3</sup>Speedy, C. B., Brown, R. F., and Goodwin, G. C., *Control Theory, Identification, and Optimal Control*, Oliver and Boyd, Edinburgh, 1970.

<sup>4</sup>Stepner, D. E., and Mehra, R. K., "Maximum Likelihood Identification and Optimal Input Design for Identifying Aircraft Stability and Control Derivatives," NASA CR-2200, 1973.

<sup>5</sup>Maine, R. E., and Iliff, K. W., "Formulation and Implementation of a Practical Algorithm for Parameter Estimation with Process and Measurement Noise," *SIAM Journal of Applied Mathematics*, Vol. 41, No. 3, 1981, pp. 558–579.

<sup>6</sup>Schultz, G., "Maximum Likelihood Identification Using Kalman Filter–Least Squares Estimation: A Comparison for the Estimation of Stability Derivatives Considering Gust Disturbances," European Space Agency Technical Translation, ESA TT-258, 1976.

<sup>7</sup>Iliff, K. W., "Identification and Stochastic Control with Application to Flight Control in Turbulence," School of Engineering and Applied Sciences, Univ. of California, UCLA-ENG-7340, Los Angeles, CA, 1973.

<sup>8</sup>Mehra, R. K., "Identification of Stochastic Linear Dynamic System Using Kalman Filter Representation," *AIAA Journal*, Vol. 9, No. 4, January 1971.

<sup>9</sup>Mehra, R. K., "On the Identification of Variances and Adaptive Kalman Filtering," *IEEE Transactions on Automatic Control*, Vol. AC-15, No. 2, 1970, pp. 175–184.

<sup>10</sup>Bach, R. E., Jr., "A User's Manual for AMES (A Parameter Estimation Program), NASA Ames Research Center," Rept. NAS2-7397, Moffett Field, CA, 1974.

<sup>11</sup>Klein, V., "Maximum Likelihood Method for Estimating Airplane Stability and Control Parameters from Flight Data in Frequency Domain," NASA TP-1637, 1980.

<sup>12</sup>Jategaonkar, R., and Plaetschke, E., "Maximum Likelihood Estimation of Parameters in Linear Systems with Process and Measurement Noise," DFVLR Forschungsbericht 87-20, 1987.

<sup>13</sup>Åström, K. J., and Eykhoff, P., "System Identification—A Survey," *Automatica*, Vol. 7, March 1971, pp. 123–162.

<sup>14</sup>Eykhoff, P., *System Identification, Parameter and State Estimation*, Wiley, New York, 1974.

<sup>15</sup>Iliff, K. W., and Taylor, L. W., "Determination of Stability Derivatives from Flight Data Using a Newton-Raphson Minimization Technique," NASA TN D-6579, 1971.

<sup>16</sup>Myers, R. H., *Classical and Modern Regression with Applications*, Duxbury Press, Boston, MA, 1986.

<sup>17</sup>Gupta, N. K., and Mehra, R. K., "Computational Aspects of Maximum Likelihood Estimation and Reduction in Sensitivity Function Calculations," *IEEE Transactions on Automatic Control*, Vol. AC-19, No. 6, 1974, pp. 774–783.

<sup>18</sup>Taylor, L. W., and Iliff, K. W., "Systems Identification Using a Modified Newton-Raphson Method: A FORTRAN Program," NASA TN D-6734, 1972.

<sup>19</sup>Murphy, P. C., "A Methodology for Airplane Parameter Estimation and Confidence Interval Determination in Nonlinear Estimation Problems," NASA RP-1153, 1986.

<sup>20</sup>Montgomery, D. C., Peck, E. A., and Vining, G. G., *Introduction to Linear Regression Analysis*, 3rd ed., Wiley, New York, 2001.

<sup>21</sup>Theil, H., *Principles of Econometrics*, Wiley, New York, 1971.

<sup>22</sup>Marquardt, D. W., "An Algorithm for Least Squares Estimation of Nonlinear Parameters," *SIAM Journal of Numerical Analysis*, Vol. 11, No. 2, 1963, pp. 431–441.

## MAXIMUM LIKELIHOOD METHODS

223

<sup>23</sup>Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. R., *Numerical Recipes in FORTRAN: The Art of Scientific Computing*, 2nd ed., Cambridge Univ. Press, New York, 1992.

<sup>24</sup>Klein, V., "Determination of Stability and Control Parameters of a Light Airplane from Flight Data Using Two Estimation Methods," NASA TP-1306, 1979.

<sup>25</sup>Maine, R. E., and Iliff, K. W., "The Theory and Practice of Estimating the Accuracy of Dynamic Flight-Determined Coefficients," NASA RP 1077, 1981.

<sup>26</sup>Morelli, E. A., and Klein, V., "Determining the Accuracy of Maximum Likelihood Parameter Estimates with Colored Residuals," NASA CR 194893, 1994.

<sup>27</sup>Gupta, N. K., "Bias and Mean Square Error Properties of General Estimators," *Proceedings of the 12th Conference on Decision and Control*, December 1976, pp. 624–628.

<sup>28</sup>Balakrishnan, A. V., and Maine, R. E., "Improvements in Aircraft Extraction Programs," NASA CR-145090, 1976.

<sup>29</sup>Morelli, E. A., and Klein, V., "Accuracy of Aerodynamic Model Parameters Estimated from Flight Test Data," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 1, 1997, pp. 74–80.

<sup>30</sup>Beale, E. M. L., "Confidence Regions in Non-Linear Estimation," *Journal of the Royal Statistical Society, Series B*, Vol. 22, No. 1, 1960, pp. 41–88.

<sup>31</sup>Mereau, P., and Raymond, J., "Computation of the Intervals of Uncertainties about the Parameters Found for Identification," NASA TM-76978, 1982.

<sup>32</sup>Klein, V., "Identification Evaluation Methods," *Parameter Identification*, AGARD-LS-104, Paper 2, 1979.

<sup>33</sup>Hall, W. E., Gupta, N. K., and Smith, R. G., "Identification of Aircraft Stability and Control Coefficients for the High Angle-of-Attack Regime," Systems Control, Inc., Engineering TR 2, Palo Alto, CA, 1974.

<sup>34</sup>Jategaonkar, R., and Plaetschke, E., "Estimation of Aircraft Parameters Using Filter Error Methods and Extended Kalman Filter," DFVLR Forschungsbericht 88-15, 1988.

<sup>35</sup>Maine, R. E., and Iliff, K. W., "Application of Parameter Estimation to Aircraft Stability and Control, The Output Error Approach," NASA RP 1168, 1986.

This page intentionally left blank

## 7

## Frequency Domain Methods

Many methods for data analysis and modeling can be formulated in the frequency domain. Frequency domain analysis has certain advantages, including physical insight in terms of frequency content, direct applicability to control system design, and a smaller number of data points for parameter estimation, among others. The basis for frequency domain methods is the finite Fourier transform, which is the mechanism for transforming time-domain data to the frequency domain. Any errors in the transformation from time to frequency domain affect the accuracy of the data in the frequency domain, which in turn impacts data analysis and modeling results. This chapter begins by presenting a method for accurately evaluating the finite Fourier transform for sampled time-domain data, and continues with a discussion of spectral densities and frequency response computed from measured input-output data.

For parametric modeling in the frequency domain, two different models for uncertainty are considered, the Fisher model and the Least Squares model. The Bayesian model is not discussed because of difficulties in formulating probability densities for complex random variables, and the consequent very limited use of this model in aircraft parameter estimation. The two models for frequency domain analysis have the following forms:

### *Fisher Model:*

- 1)  $\tilde{z} = \tilde{h}(\theta) + \tilde{v}$
- 2)  $\theta$  is a vector of unknown constant real parameters.
- 3)  $\tilde{v}$  is a complex random vector with probability density  $p(\tilde{v})$ .

### *Least-Squares Model:*

- 1)  $\tilde{z} = \tilde{H}\theta + \tilde{v}$
- 2)  $\theta$  is a vector of unknown constant real parameters.
- 3)  $\tilde{v}$  is a complex random vector of measurement noise.

In the measurement equations,  $\tilde{h}(\theta)$  is a known complex vector function and  $\tilde{H}$  is a known complex matrix. Both  $\tilde{h}(\theta)$  and  $\tilde{H}$  are normally functions of the transformed states and controls,  $\tilde{x}$  and  $\tilde{u}$ , respectively. The measurement vector  $\tilde{z}$  is the Fourier transform of the time-domain vector  $z$ . Properties of the complex random

sequence  $\tilde{\nu}$  are presented in Ref. 1. These properties were developed from the assumed properties of the real random sequence  $\nu$ .

The two models just introduced lead to maximum likelihood estimators for a dynamic system and to an ordinary least-squares estimator. In the development of maximum likelihood estimators, a linear dynamic system with process noise is considered first. Then, in a manner similar to the time-domain approach, estimation algorithms for simplified models with no process noise or no measurement noise are presented. For the least-squares model, a general form of complex linear regression is described. Applications of frequency domain techniques to parameter estimation from flight and water-tunnel data are demonstrated in examples.

## 7.1 Transforming Measured Data to the Frequency Domain

The transformation of measured data from the time domain to the frequency domain is based on the Fourier integral. For a given continuous function in the time domain  $x(t)$ , the corresponding continuous function in the frequency domain  $\tilde{x}(\omega)$  is obtained as

$$\mathcal{F}[x(t)] \equiv \tilde{x}(\omega) = \int_{-\infty}^{\infty} x(t) e^{-j\omega t} dt \quad (7.1)$$

where  $j = \sqrt{-1}$  and  $\omega$  is the angular frequency in radians per second. The quantity  $\tilde{x}(\omega)$  defined in the preceding equation is called the Fourier transform of  $x(t)$ . The  $\tilde{x}(\omega)$  function can be transformed back to the time domain using the inverse Fourier transform,

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \tilde{x}(\omega) e^{j\omega t} d\omega \quad (7.2)$$

The functions  $x(t)$  and  $\tilde{x}(\omega)$  are related by Eqs. (7.1) and (7.2), which are known as the Fourier transform pair. Transformation properties are discussed in detail in many references (e.g., Bendat and Piersol<sup>2</sup>).

The Fourier transform of a continuous scalar time function  $x(t)$  on a finite time interval  $[0, T]$  is called the finite Fourier transform, defined by

$$\tilde{x}(\omega) \equiv \int_0^T x(t) e^{-j\omega t} dt \quad (7.3)$$

or

$$\tilde{x}(f) \equiv \int_0^T x(t) e^{-j2\pi ft} dt \quad (7.4)$$

where

$$\omega = 2\pi f \quad (7.5)$$

The preceding relationships show that the finite Fourier transform  $\tilde{x}(\omega)$  can be interpreted as a coefficient in an expansion of  $x(t)$  in terms of the basis function  $e^{-j\omega t} = \cos \omega t - j \sin \omega t$ , for each frequency  $\omega$ .

## FREQUENCY DOMAIN METHODS

227

When the time function  $x(t)$  is sampled at discrete, evenly spaced time intervals  $\Delta t$ , the finite Fourier transform can be approximated by

$$\tilde{x}(f) \approx \Delta t \sum_{i=0}^{N-1} x(i) e^{-j2\pi fi \Delta t} \quad (7.6)$$

where the time length of the measured data is  $T = (N - 1)\Delta t$  and

$$\begin{aligned} \Delta t &= T/(N - 1) \\ t_i &= i\Delta t \quad x(i) \equiv x(t_i) = x(i\Delta t) \quad i = 0, 1, 2, \dots, N - 1 \end{aligned} \quad (7.7)$$

Note that the total number of data points in the time domain is  $N$ , and the time index  $i$  starts at 0.

For a conventional finite Fourier transform, the frequencies are chosen as

$$f_k = \frac{k}{N\Delta t} \quad k = 0, 1, 2, \dots, N - 1 \quad (7.8)$$

or

$$\omega_k = 2\pi f_k = 2\pi \frac{k}{N\Delta t} \quad k = 0, 1, 2, \dots, N - 1 \quad (7.9)$$

Using the discrete frequencies defined in Eqs. (7.8) and (7.9), the approximation to the finite Fourier transform in Eq. (7.6) becomes

$$\tilde{x}(k) = \Delta t \sum_{i=0}^{N-1} x(i) e^{-j(2\pi k/N)i} \quad k = 0, 1, 2, \dots, N - 1 \quad (7.10)$$

and the inverse is

$$x(i) = \frac{1}{N\Delta t} \sum_{k=0}^{N-1} \tilde{x}(k) e^{j(2\pi k/N)i} \quad i = 0, 1, 2, \dots, N - 1 \quad (7.11)$$

Often Eqs. (7.10) and (7.11) are written for the normalized sampling interval  $\Delta t = 1$ . The result is called the discrete Fourier transform, defined by

$$X(k) \equiv \sum_{i=0}^{N-1} x(i) e^{-j(2\pi k/N)i} \quad k = 0, 1, 2, \dots, N - 1 \quad (7.12)$$

which has the inverse

$$x(i) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j(2\pi k/N)i} \quad i = 0, 1, 2, \dots, N - 1 \quad (7.13)$$

Equations (7.12) and (7.13) are called the discrete Fourier transform (DFT) pair.

In general, the DFT coefficients are complex numbers, with real and imaginary parts. If  $x(i)$ ,  $i = 0, 1, 2, \dots, N - 1$ , is a sequence of real numbers, as is the case for flight-test data, it can be seen from Eq. (7.12) that

$$X(N - k) = X^*(k) \quad (7.14)$$

where  $X^*(k)$  is the complex conjugate of  $X(k)$ . It follows that the transformed data have conjugate symmetry about the value of  $k$  corresponding to the Nyquist frequency  $1/(2\Delta t)$ . Because of this, it suffices to compute the finite Fourier transform for only the first  $M$  frequencies defined in Eq. (7.9), where

$$M = \begin{cases} N/2 + 1 & \text{for } N \text{ even} \\ (N + 1)/2 & \text{for } N \text{ odd} \end{cases} \quad (7.15)$$

The values given for  $M$  in Eq. (7.15) represent the fundamental limitation that frequencies contained in a sampled time history must fall within the frequency band  $[0, f_N]$ , where  $f_N = 1/(2\Delta t)$  is the Nyquist frequency, defined as half the sampling frequency (see Chapter 9).

Direct computation of the DFT using Eq. (7.12) is not economical computationally. A numerically efficient method for computing the DFT is the fast Fourier transform (FFT) algorithm, proposed by Cooley and Tukey.<sup>3</sup> Bendat and Piersol<sup>2</sup> give a more general form of this algorithm, which is used as the basis for many Fourier transform computations. Ref. 4 describes a method based on the chirp  $z$ -transform that can be used to compute the finite Fourier transform defined in Eq. (7.3) with very high accuracy at arbitrarily selected frequencies within the frequency band  $[0, f_N]$ . This method is described in detail in Chapter 11.

Often, data from more than one maneuver or data record must be combined for a single analysis. In the frequency domain, this can be done by simply adding the Fourier transforms from individual maneuvers at corresponding frequencies. Each maneuver to be combined must be transformed into the frequency domain using the same frequencies. This is easily done when the frequencies for the transformation can be selected arbitrarily, as in the procedure described in Chapter 11.

If frequency-domain data from two maneuvers are to be combined to represent the situation where the associated time series are concatenated in time, the task can be done in the frequency domain by multiplying the frequency domain data from the second maneuver by  $e^{-j\omega T_1}$ , where  $T_1$  is the time length of the first maneuver, and then adding the frequency-domain data at corresponding frequencies  $\omega$ . This is equivalent to a pure time shift of the data in the second maneuver by  $T_1$  seconds.

## 7.2 Frequency Response

In Sec. 2.1, the nonparametric relation between the input and output vectors of a linear, time-invariant system was formed as

$$\mathbf{y}(t) = \int_0^t \mathbf{G}(t - \tau) \mathbf{u}(\tau) d\tau \quad (7.16)$$

## FREQUENCY DOMAIN METHODS

229

assuming zero initial conditions. The Laplace transform of the preceding equation is

$$\tilde{y}(s) = G(s)\tilde{u}(s) \quad (7.17)$$

where  $G(s)$  is the transfer function matrix. Replacing the Laplace transform variable  $s$  with  $j\omega$  results in the frequency response matrix  $G(j\omega)$  with elements

$$G_{jk}(j\omega) = \frac{\tilde{y}_j(j\omega)}{\tilde{u}_k(j\omega)} \quad j = 1, 2, \dots, n_o \quad k = 1, 2, \dots, n_i \quad (7.18)$$

Individual frequency response functions can be determined experimentally from measured input-output data as

$$\hat{G}_{jk}(j\omega) = \frac{\tilde{z}_j(j\omega)}{\tilde{u}_k(j\omega)} \quad j = 1, 2, \dots, n_o \quad k = 1, 2, \dots, n_i \quad (7.19)$$

where  $\tilde{z}_j(j\omega)$  and  $\tilde{u}_k(j\omega)$  are finite Fourier transforms of the relevant measured input and output. The frequency response matrix is therefore composed of elements that are frequency response functions for a scalar input and a scalar output. Simplifying the notation by dropping the subscripts indicating the particular input and output, an individual frequency response function can be estimated from

$$\hat{G}(j\omega) = \frac{\tilde{z}(j\omega)}{\tilde{u}(j\omega)} \quad (7.20)$$

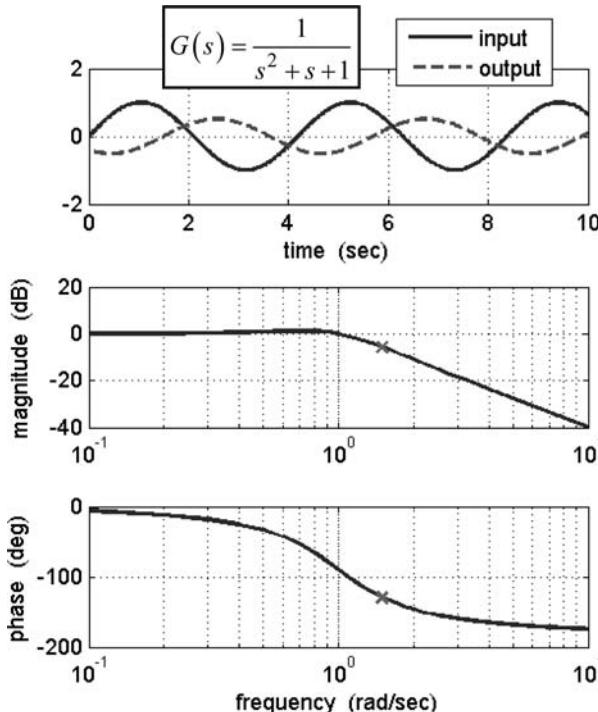
For each selected frequency, the frequency response function is, in general, a complex number,

$$\begin{aligned} G(j\omega) &= \operatorname{Re}[G(j\omega)] + j\operatorname{Im}[G(j\omega)] \\ &= R(\omega)e^{j\phi(\omega)} \end{aligned} \quad (7.21)$$

where

$$\begin{aligned} R(\omega) &= \sqrt{\{\operatorname{Re}[G(j\omega)]\}^2 + \{\operatorname{Im}[G(j\omega)]\}^2} \\ \phi(\omega) &= \tan^{-1} \left\{ \frac{\operatorname{Im}[G(j\omega)]}{\operatorname{Re}[G(j\omega)]} \right\} \end{aligned} \quad (7.22)$$

A collection of frequency response functions for many different frequencies is called the frequency response, which can be presented graphically, usually by displaying the amplitude ratio  $R(\omega)$  and the phase angle  $\phi(\omega)$  between the input and output, as a function of frequency. Such plots are called frequency response curves. Plotting  $20\log_{10} R(\omega)$  and phase  $\phi(\omega)$  versus  $\log_{10} \omega$  results in a Bode plot, shown in the lower portion Fig. 7.1 for a simple transfer function.



**Fig. 7.1 Frequency response for a linear system.**

In the special case where the input is a simple harmonic function, the steady-state output of a linear, time-invariant system is also a simple harmonic function, with the same frequency but modified amplitude and a phase shift. An example can be seen in Fig. 7.1, where the input to the transfer function is a pure sine wave with unit amplitude and frequency  $\omega = 1.5 \text{ rad/s}$ . The steady-state amplitude ratio and phase angle are estimated as

$$\begin{aligned}\hat{R}(\omega) &= |z|/|u| \\ \hat{\phi}(\omega) &= \tau\omega\end{aligned}\tag{7.23}$$

where  $|u|$  and  $|z|$  are the steady-state input and output amplitudes, respectively, and  $\tau$  is the time delay from input to output, which can be evaluated at a zero crossing in the time-domain, for example.

Frequency response functions are often estimated from spectral densities.<sup>2,5–8</sup> If the input and output variables for a linear, time-invariant dynamic system are expressed in terms of correlation functions, as is done for stochastic systems, Eq. (7.16) takes the form

$$\mathcal{R}_{yu}(\tau) = \int_0^\infty G(t)\mathcal{R}_{uu}(t + \tau) dt = \mathcal{R}_{uy}(-\tau)\tag{7.24}$$

## FREQUENCY DOMAIN METHODS

231

where the correlation functions are given by

$$\mathcal{R}_{yu}(\tau) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T y(t)u(t + \tau) dt \quad (7.25a)$$

$$\mathcal{R}_{uu}(\tau) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T u(t)u(t + \tau) dt \quad (7.25b)$$

Applying the Fourier transform to the correlation functions results in spectral densities (see Appendix B),

$$S_{yu}(j\omega) = \int_0^\infty \mathcal{R}_{yu}(\tau) e^{-j\omega\tau} d\tau \quad (7.26a)$$

$$S_{uu}(j\omega) = \int_0^\infty \mathcal{R}_{uu}(\tau) e^{-j\omega\tau} d\tau \quad (7.26b)$$

Since convolution in the time domain is equivalent to multiplication in the frequency domain (see Appendix A), the Fourier transform of Eq. (7.24) is

$$S_{yu}(j\omega) = G(j\omega)S_{uu}(j\omega) \quad (7.27)$$

Combining Eqs. (7.25) and (7.26), the spectral densities can also be given as

$$S_{yu}(j\omega) = \tilde{y}(j\omega)\tilde{u}^*(j\omega) \quad (7.28a)$$

$$S_{uu}(j\omega) = \tilde{u}(j\omega)\tilde{u}^*(j\omega) \quad (7.28b)$$

where  $\tilde{u}^*(j\omega)$  is the complex conjugate of  $\tilde{u}(j\omega)$ . Therefore, by applying the finite Fourier transform to measured input and output data, an estimate of the frequency response function can also be obtained from Eqs. (7.27) and (7.28) as

$$\hat{G}(j\omega) = \frac{\hat{S}_{yu}(j\omega)}{\hat{S}_{uu}(j\omega)} = \frac{\tilde{y}(j\omega)\tilde{u}^*(j\omega)}{\tilde{u}(j\omega)\tilde{u}^*(j\omega)} \quad (7.29)$$

Chapter 11 includes a discussion of some practical methods for computing the spectral densities required in Eq. (7.29) to calculate an empirical estimate of the frequency response function  $\hat{G}(j\omega)$ .

The coherence between the input and output is a real-valued quantity defined in terms of the spectral densities,

$$\gamma^2(\omega) = \frac{|S_{yu}(j\omega)|^2}{S_{uu}(j\omega)S_{yy}(j\omega)} \quad 0 \leq \gamma^2(\omega) \leq 1 \quad (7.30)$$

An estimated value for the coherence  $\gamma^2(\omega)$  is obtained using estimates of the spectral densities. The coherence is a measure of linearity between the input and output, so  $\gamma^2(\omega) = 1$  indicates that the input and output are related perfectly by a linear dynamic system. The coherence is often used as a metric for evaluating the

adequacy of a linear model for characterizing frequency domain data. The coherence can be less than 1 for any of the following reasons: 1) measurement noise is present in the data, 2) the dynamic system is nonlinear, and/or 3) other inputs or disturbances influence the output. More information on the coherence and its extension for a general multivariable system can be found in Ref. 2.

### 7.3 Maximum Likelihood Estimator

To develop the maximum likelihood estimator in the frequency domain, it is necessary to postulate a model for the dynamic system, and then transform this model into the frequency domain. The next step is to formulate the likelihood function of the innovations, and then to maximize the likelihood function with respect to the unknown parameters. When the estimation problem is nonlinear, the modified Newton-Raphson algorithm is usually selected for the optimization, as in the time-domain case.

Model equations for a linear continuous-time dynamic system with process noise and discrete-time measurements are considered,

$$\dot{\mathbf{x}}(t) = \mathbf{Ax}(t) + \mathbf{Bu}(t) + \mathbf{B}_w \mathbf{w}(t) \quad (7.31a)$$

$$z(i) = \mathbf{Cx}(i) + \mathbf{Du}(i) + \mathbf{v}(i) \quad i = 1, 2, \dots, N \quad (7.31b)$$

$$E[\mathbf{x}(0)] = \mathbf{0} \quad E[\mathbf{x}(0)\mathbf{x}^T(0)] = \mathbf{P}_0 \quad (7.31c)$$

where  $\mathbf{w}(t)$  and  $\mathbf{v}(i)$  are assumed to be independent white Gaussian noise sequences with

$$\begin{aligned} E[\mathbf{w}(t)] &= \mathbf{0} & E[\mathbf{w}(t_i) \mathbf{w}^T(t_j)] &= \mathbf{Q} \delta(t_i - t_j) \\ E[\mathbf{v}(i)] &= \mathbf{0} & E[\mathbf{v}(i)\mathbf{v}^T(j)] &= \mathbf{R} \delta_{ij} \end{aligned} \quad (7.31d)$$

In the general case, the unknown parameters will occur in the matrices  $\mathbf{A}, \mathbf{B}, \mathbf{B}_w, \mathbf{C}, \mathbf{D}, \mathbf{P}_0, \mathbf{Q}$ , and  $\mathbf{R}$ . As discussed in Sec. 6.1, estimation of all of these parameters can be extremely difficult because of the algorithm complexity and possible identifiability problems. For these reasons, the problem will be simplified by using a steady-state Kalman filter and estimating the unknown parameters in this formulation. Returning to Eqs. (6.6) and (6.20), the filter equations take the form

$$\frac{d}{dt}[\hat{\mathbf{x}}(t|i-1)] = \mathbf{A}\hat{\mathbf{x}}(t|i-1) + \mathbf{B}\mathbf{u}(t) \quad (7.32a)$$

for  $(i-1)\Delta t \leq t \leq i\Delta t$ ,

$$\hat{\mathbf{x}}(i|i) = \hat{\mathbf{x}}(i|i-1) + \mathbf{K}\mathbf{v}(i) \quad (7.32b)$$

where the innovations

$$\mathbf{v}(i) = z(i) - \mathbf{C}\hat{\mathbf{x}}(i|i-1) - \mathbf{D}\mathbf{u}(i) \quad (7.32c)$$

## FREQUENCY DOMAIN METHODS

233

form a sequence of independent Gaussian vectors with

$$E[\mathbf{v}(i)] = \mathbf{0} \quad E[\mathbf{v}(i)\mathbf{v}^T(j)] = \mathcal{B}\delta_{ij} \quad (7.32d)$$

Based on the suggestion of Mehra,<sup>9</sup> the unknown parameters are selected as the elements of the matrices  $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}, \mathbf{K}$ , and  $\mathcal{B}$ .

In the next step, the time functions in Eqs. (7.32) are written in the form of their Fourier series expansions. The Fourier series components of  $\mathbf{x}(i)$  are defined by Eq. (7.10) as

$$\tilde{\mathbf{x}}(k) = \Delta t \sum_{i=0}^{N-1} \mathbf{x}(i) e^{-j(2\pi k/N)i} \quad k = 0, 1, 2, \dots, N-1 \quad (7.33)$$

Similar expressions apply for  $\mathbf{z}(i)$ ,  $\mathbf{u}(i)$ , and  $\mathbf{v}(i)$ . As stated in Ref. 1, the Fourier series expansion of random variables holds in the mean-squared sense, so that

$$E\left[ \left| \mathbf{v}(i) - \frac{1}{N\Delta t} \sum_{k=0}^{N-1} \tilde{\mathbf{v}}(k) e^{j(2\pi k/N)i} \right|^2 \right] = \mathbf{0} \quad (7.34)$$

and similarly for the other transformed quantities. The transform in Eq. (7.33) represents an Euler approximation to the finite Fourier integral, which can be replaced by a high-accuracy calculation mentioned earlier and described in Chapter 11. Now Eqs. (7.32) transformed into the frequency domain become

$$j\omega_k \tilde{\mathbf{x}}(k) = \mathbf{A}\tilde{\mathbf{x}}(k) + \mathbf{B}\tilde{\mathbf{u}}(k) + \mathbf{K}\tilde{\mathbf{v}}(k) \quad (7.35a)$$

or

$$\tilde{\mathbf{x}}(k) = (j\omega_k I - \mathbf{A})^{-1} \mathbf{B}\tilde{\mathbf{u}}(k) + (j\omega_k I - \mathbf{A})^{-1} \mathbf{K}\tilde{\mathbf{v}}(k) \quad (7.35b)$$

$$\tilde{\mathbf{z}}(k) = \mathbf{C}\tilde{\mathbf{x}}(k) + \mathbf{D}\tilde{\mathbf{u}}(k) + \tilde{\mathbf{v}}(k) \quad k = 0, 1, 2, \dots, N-1 \quad (7.35c)$$

where  $\omega_k$  is the  $k$ th frequency, and  $\omega_k = 2\pi k/N$  when the transform in Eq. (7.33) is used. The transformed innovations  $\tilde{\mathbf{v}}(k)$  are uncorrelated Gaussian random variables,<sup>1</sup> with

$$E[\tilde{\mathbf{v}}(k)] = \mathbf{0} \quad E[\tilde{\mathbf{v}}(k)\tilde{\mathbf{v}}^\dagger(k)] = \frac{\mathbf{S}_{vv}}{N} \quad (7.35d)$$

where  $\mathbf{S}_{vv}$  is a real diagonal matrix with diagonal elements equal to the power spectral densities of the elements of  $\mathbf{v}(i)$ , and  $\tilde{\mathbf{v}}^\dagger(k)$  is the complex conjugate transpose of  $\tilde{\mathbf{v}}(k)$ .

The left side of Eq. (7.35a) requires that the bias and trend be removed from  $\mathbf{x}(t)$ , so that  $\mathbf{x}(T) = \mathbf{x}(0) = 0$  before the Fourier transformation, since

$$\begin{aligned}\mathcal{F}[\dot{\mathbf{x}}(t)] &= \int_0^T \dot{\mathbf{x}}(t) e^{-j\omega t} dt = j\omega \int_0^T \dot{\mathbf{x}}(t) e^{-j\omega t} dt + \mathbf{x}(T)e^{-j\omega T} - \mathbf{x}(0) \\ &= j\omega \tilde{\mathbf{x}}(\omega) + \mathbf{x}(T)e^{-j\omega T} - \mathbf{x}(0)\end{aligned}\quad (7.36)$$

It is also possible to eliminate the terms associated with the endpoints using a modulating function approach based on harmonically-related sinusoids.<sup>10</sup>

From Eqs. (7.35),

$$\begin{aligned}\tilde{\mathbf{z}}(k) &= [\mathbf{C}(j\omega_k \mathbf{I} - \mathbf{A})^{-1} \mathbf{B} + \mathbf{D}] \tilde{\mathbf{u}}(k) + [\mathbf{C}(j\omega_k \mathbf{I} - \mathbf{A})^{-1} \mathbf{K} + \mathbf{I}] \tilde{\mathbf{v}}(k) \\ &= \mathbf{G}_1(k, \boldsymbol{\theta}) \tilde{\mathbf{u}}(k) + \mathbf{G}_2(k, \boldsymbol{\theta}) \tilde{\mathbf{v}}(k)\end{aligned}\quad (7.37)$$

where  $\mathbf{G}_1$  and  $\mathbf{G}_2$  are the system transfer function matrices defined as

$$\mathbf{G}_1(k, \boldsymbol{\theta}) = \mathbf{C}(j\omega_k \mathbf{I} - \mathbf{A})^{-1} \mathbf{B} + \mathbf{D} \quad (7.38a)$$

$$\mathbf{G}_2(k, \boldsymbol{\theta}) = \mathbf{C}(j\omega_k \mathbf{I} - \mathbf{A})^{-1} \mathbf{K} + \mathbf{I} \quad (7.38b)$$

and  $\boldsymbol{\theta}$  is the vector of unknown parameters in matrices  $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}, \mathbf{K}$ , and  $\mathbf{S}_{vv}$ . Since the Kalman filter representation in Eqs. (7.32) is invertible,<sup>11</sup>  $\mathbf{G}_2$  is nonsingular, and the innovations can be expressed as

$$\tilde{\mathbf{v}}(k) = \mathbf{G}_2^{-1} \tilde{\mathbf{z}}(k) - \mathbf{G}_2^{-1} \mathbf{G}_1 \tilde{\mathbf{u}}(k) \quad (7.39)$$

To develop the likelihood function, the quantity  $\mathbf{V}_N$  is introduced, which consists of all innovations up to and including the frequency for which  $k = N - 1$ ,

$$\mathbf{V}_N \equiv [\tilde{\mathbf{v}}(0) \quad \tilde{\mathbf{v}}(1) \quad \cdots \quad \tilde{\mathbf{v}}(N-1)]^T \quad (7.40)$$

Assuming that  $\tilde{\mathbf{v}}(k)$  has a probability density  $p(\tilde{\mathbf{v}})$ , then it follows from the definition of probability densities that

$$p(\mathbf{V}_N) = p[\tilde{\mathbf{v}}(N-1)|\mathbf{V}_{N-1}] p(\mathbf{V}_{N-1}) \quad (7.41)$$

Repeated use of Eq. (7.41) gives the expression for the likelihood function,

$$\begin{aligned}\mathbb{L}(\mathbf{V}_N; \boldsymbol{\theta}) &= p(\mathbf{V}_N | \boldsymbol{\theta}) \\ &= p[\tilde{\mathbf{v}}(N-1)|\mathbf{V}_{N-1}] p[\tilde{\mathbf{v}}(N-2)|\mathbf{V}_{N-2}] \cdots p[\tilde{\mathbf{v}}(1)|\mathbf{V}_1] p[\tilde{\mathbf{v}}(0)]\end{aligned}\quad (7.42)$$

Since the distribution of  $\tilde{\mathbf{v}}(k)$  is Gaussian, the distribution of  $\tilde{\mathbf{v}}(k)$  given  $\mathbf{V}_k$  is also Gaussian. The probability density for this complex Gaussian vector is

$$p[\tilde{\mathbf{v}}(k)|\mathbf{V}_k] = \frac{N^{no}}{\pi^{no} |\mathbf{S}_{vv}|} \exp[-N \tilde{\mathbf{v}}^\dagger(k) \mathbf{S}_{vv}^{-1} \tilde{\mathbf{v}}(k)] \quad (7.43)$$

## FREQUENCY DOMAIN METHODS

235

See Refs. 1 and 12 for more details. The power spectral density  $S_{vv}$  appears in Eq. (7.43) because of the definitions in Eq. (7.35d). Combining Eqs. (7.42) and (7.43), the negative logarithm of  $\mathbb{L}(V_N; \boldsymbol{\theta})$  is

$$-\ell_n[\mathbb{L}(V_N; \boldsymbol{\theta})] \equiv J(\boldsymbol{\theta}) = N \sum_{k=0}^{N-1} \tilde{\mathbf{v}}^\dagger(k) S_{vv}^{-1} \tilde{\mathbf{v}}(k) + N \ell_n|S_{vv}| + N n_o \ell_n \left( \frac{\pi}{N} \right) \quad (7.44)$$

Excluding the constant term that is unaffected by  $\boldsymbol{\theta}$ , the cost function to be minimized is

$$J(\boldsymbol{\theta}) = N \sum_{k=0}^{N-1} \tilde{\mathbf{v}}^\dagger(k) S_{vv}^{-1} \tilde{\mathbf{v}}(k) + N \ell_n|S_{vv}| \quad (7.45)$$

The maximum likelihood parameter vector estimate is the value of  $\boldsymbol{\theta}$  that minimizes the cost function in Eq. (7.45). Optimizing  $J(\boldsymbol{\theta})$  for the parameters in  $S_{vv}$  gives

$$\hat{S}_{vv} = \sum_{k=0}^{N-1} \tilde{\mathbf{v}}(k, \hat{\boldsymbol{\theta}}) \tilde{\mathbf{v}}^\dagger(k, \hat{\boldsymbol{\theta}}) = \sum_{k=0}^{N-1} [\tilde{\mathbf{z}}(k) - \tilde{\mathbf{y}}(k, \hat{\boldsymbol{\theta}})] [\tilde{\mathbf{z}}(k) - \tilde{\mathbf{y}}(k, \hat{\boldsymbol{\theta}})]^\dagger \quad (7.46)$$

where  $\hat{\boldsymbol{\theta}}$  is an estimate of the vector of parameters  $\boldsymbol{\theta}$ , which appear in the dynamic system matrices. The initial estimate of  $S_{vv}$  is made with  $\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}_o$ , where  $\boldsymbol{\theta}_o$  is a nominal starting value of  $\boldsymbol{\theta}$ .

The estimates of the remaining parameters are found using the modified Newton-Raphson technique for nonlinear optimization. For a given  $\hat{S}_{vv}$ , the cost function

$$J(\boldsymbol{\theta}) = N \sum_{k=0}^{N-1} \tilde{\mathbf{v}}^\dagger(k) \hat{S}_{vv}^{-1} \tilde{\mathbf{v}}(k) \quad (7.47)$$

is minimized by iteratively computing

$$\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}_o + \Delta\hat{\boldsymbol{\theta}} \quad (7.48)$$

where the parameter vector update  $\Delta\hat{\boldsymbol{\theta}}$  is given by

$$\Delta\hat{\boldsymbol{\theta}} = -\mathbf{M}_{\boldsymbol{\theta}=\boldsymbol{\theta}_o}^{-1} \left[ \frac{\partial J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right]_{\boldsymbol{\theta}=\boldsymbol{\theta}_o} \quad (7.49)$$

and  $\mathbf{M}$  is the Fisher information matrix,

$$\mathbf{M} = -E \left[ \frac{\partial^2 \ell_n \mathbb{L}(V_N; \boldsymbol{\theta})}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} \right] = E \left[ \frac{\partial^2 J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} \right] \quad (7.50)$$

Because  $\Delta\hat{\theta}$  is a vector with only real elements, and the log-likelihood function is real, the expressions for the first- and second-order gradients of  $J(\theta)$  are also real,

$$\begin{aligned}\frac{\partial J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} &= 2N\text{Re} \left[ \sum_{k=0}^{N-1} \tilde{\mathbf{v}}^\dagger(k) \hat{\mathbf{S}}_{vv}^{-1} \frac{\partial \tilde{\mathbf{v}}(k)}{\partial \boldsymbol{\theta}} \right] \\ &= -2N\text{Re} \left[ \sum_{k=0}^{N-1} \tilde{\mathbf{v}}^\dagger(k) \hat{\mathbf{S}}_{vv}^{-1} \frac{\partial \tilde{\mathbf{y}}(k)}{\partial \boldsymbol{\theta}} \right]\end{aligned}\quad (7.51)$$

$$\begin{aligned}\frac{\partial^2 J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} &= 2N\text{Re} \left[ \sum_{k=0}^{N-1} \frac{\partial \tilde{\mathbf{v}}^\dagger(k)}{\partial \boldsymbol{\theta}} \hat{\mathbf{S}}_{vv}^{-1} \frac{\partial \tilde{\mathbf{v}}(k)}{\partial \boldsymbol{\theta}} \right] \\ &= 2N\text{Re} \left[ \sum_{k=0}^{N-1} \frac{\partial \tilde{\mathbf{y}}^\dagger(k)}{\partial \boldsymbol{\theta}} \hat{\mathbf{S}}_{vv}^{-1} \frac{\partial \tilde{\mathbf{y}}(k)}{\partial \boldsymbol{\theta}} \right]\end{aligned}\quad (7.52)$$

where the expression in Eq. (7.52) is the simplified second gradient used in the modified Newton-Raphson method.

Expressions for the elements of both cost gradients were developed by Klein<sup>1</sup> in the form

$$\begin{aligned}\frac{\partial J(\boldsymbol{\theta})}{\partial \theta_j} &= -2\text{Re} \left\{ \sum_{k=0}^{N-1} \text{Tr} \left[ \frac{\partial \mathbf{G}_1^\dagger}{\partial \theta_j} (\mathbf{G}_2^\dagger)^{-1} \hat{\mathbf{S}}_{vv}^{-1} \mathbf{G}_2^{-1} (\hat{\mathbf{S}}_{zu} - \mathbf{G}_1 \hat{\mathbf{S}}_{uu}) \right] \right. \\ &\quad \left. - \text{Tr} \left[ \frac{\partial \mathbf{G}_2^\dagger}{\partial \theta_j} (\mathbf{G}_2^\dagger)^{-1} \right] \right\}\end{aligned}\quad (7.53)$$

$$\begin{aligned}M_{ij} = \frac{\partial^2 J(\boldsymbol{\theta})}{\partial \theta_i \partial \theta_j} &= 2\text{Re} \left\{ \sum_{k=0}^{N-1} \text{Tr} \left[ \frac{\partial \mathbf{G}_1^\dagger}{\partial \theta_j} (\mathbf{G}_2^\dagger)^{-1} \hat{\mathbf{S}}_{vv}^{-1} \mathbf{G}_2^{-1} \frac{\partial \mathbf{G}_1}{\partial \theta_i} \hat{\mathbf{S}}_{uu} \right] \right. \\ &\quad \left. + \text{Tr} \left[ \frac{\partial \mathbf{G}_2^\dagger}{\partial \theta_j} (\mathbf{G}_2^\dagger)^{-1} \mathbf{G}_2^{-1} \frac{\partial \mathbf{G}_2}{\partial \theta_i} \right] \right\}\end{aligned}\quad (7.54)$$

where

$$\hat{\mathbf{S}}_{uu}(k) = \tilde{\mathbf{u}}(k) \tilde{\mathbf{u}}^\dagger(k) \quad (7.55a)$$

$$\hat{\mathbf{S}}_{zu}(k) = \tilde{\mathbf{z}}(k) \tilde{\mathbf{u}}^\dagger(k) \quad (7.55b)$$

are the estimates of the input spectral density matrix and cross spectral density matrix, respectively.

After estimating  $\mathbf{S}_{vv}$  from Eq. (7.46), the vector of remaining unknown parameters  $\boldsymbol{\theta}$  is composed of elements of the  $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ , and  $\mathbf{K}$  matrices. The final estimated parameters for the model of Eqs. (7.35) have the properties given in Sec. 6.1.3.

## 7.4 Output-Error Method

As in the time-domain analysis, the maximum likelihood estimator in the frequency domain is substantially simplified when there is no process noise. In that case, the dynamic system is deterministic,

$$j\omega_k \tilde{x}(k) = A\tilde{x}(k) + B\tilde{u}(k) \quad (7.56a)$$

$$\tilde{y}(k) = C\tilde{x}(k) + D\tilde{u}(k) \quad (7.56b)$$

$$\tilde{z}(k) = G(k, \boldsymbol{\theta})\tilde{u}(k) + \tilde{\nu}(k) \quad (7.56c)$$

$$E[\tilde{\nu}(k)] = \mathbf{0} \quad E[\tilde{\nu}(k)\tilde{\nu}^\dagger(k)] = \frac{S_{\nu\nu}}{N} \quad (7.56d)$$

where

$$G(k, \boldsymbol{\theta}) \equiv G_1(k, \boldsymbol{\theta}) = C(j\omega I - A)^{-1}B + D \quad (7.57a)$$

$$\omega_k = 2\pi k/T \quad k = 0, 1, 2, \dots, N-1 \quad (7.57b)$$

and  $S_{\nu\nu}$  is the spectral density of  $\nu(i)$ . For these model equations, the innovations are reduced to output errors or residuals,

$$\tilde{\nu}(k, \boldsymbol{\theta}) = \tilde{z}(k) - \tilde{y}(k, \boldsymbol{\theta}) = \tilde{z}(k) - G(k, \boldsymbol{\theta})\tilde{u}(k) \quad (7.58)$$

The output-error cost function is the negative log-likelihood function, excluding the constant term,

$$J(\boldsymbol{\theta}) = N \sum_{k=0}^{N-1} \tilde{\nu}^\dagger(k, \boldsymbol{\theta}) S_{\nu\nu}^{-1} \tilde{\nu}(k, \boldsymbol{\theta}) + N \ell_n |S_{\nu\nu}| \quad (7.59)$$

and the parameter estimation process is similar to that presented earlier for the more general model equations. The estimates for the parameters in  $S_{\nu\nu}$  are obtained from

$$\hat{S}_{\nu\nu} = \sum_{k=0}^{N-1} \tilde{\nu}(k, \hat{\boldsymbol{\theta}}) \tilde{\nu}^\dagger(k, \hat{\boldsymbol{\theta}}) = \sum_{k=0}^{N-1} [\tilde{z}(k) - \tilde{y}(k, \hat{\boldsymbol{\theta}})] [\tilde{z}(k) - \tilde{y}(k, \hat{\boldsymbol{\theta}})]^\dagger \quad (7.60)$$

To estimate the unknown parameters in matrices  $A$ ,  $B$ ,  $C$ , and  $D$ , the modified Newton-Raphson algorithm is used again. The elements of the gradient  $\partial J(\boldsymbol{\theta})/\partial \boldsymbol{\theta}$  and the information matrix  $M$  are obtained by simplifying Eqs. (7.53) and (7.54), using  $G_2 = I$  and  $G_1 = G$ ,

$$\frac{\partial J(\boldsymbol{\theta})}{\partial \theta_j} = -2\text{Re} \left\{ \sum_{k=0}^{N-1} \text{Tr} \left[ \frac{\partial G^\dagger}{\partial \theta_j} \hat{S}_{\nu\nu}^{-1} (\hat{S}_{zu} - G \hat{S}_{uu}) \right] \right\} \quad (7.61)$$

$$M_{ij} = \frac{\partial^2 J(\boldsymbol{\theta})}{\partial \theta_i \partial \theta_j} = 2\text{Re} \left\{ \sum_{k=0}^{N-1} \text{Tr} \left[ \frac{\partial G^\dagger}{\partial \theta_j} \hat{S}_{\nu\nu}^{-1} \frac{\partial G}{\partial \theta_i} \hat{S}_{uu} \right] \right\} \quad (7.62)$$

where

$$\hat{S}_{uu}(k) = \tilde{u}(k)\tilde{u}^\dagger(k) \quad (7.63a)$$

$$\hat{S}_{zu}(k) = \tilde{z}(k)\tilde{u}^\dagger(k) \quad (7.63b)$$

The expressions for the gradient of the negative log-likelihood function and the information matrix can also be derived from the output-error cost function,

$$J(\boldsymbol{\theta}) = N \sum_{k=0}^{N-1} \tilde{\mathbf{v}}^\dagger(k, \boldsymbol{\theta}) \hat{S}_{vv}^{-1} \tilde{\mathbf{v}}(k, \boldsymbol{\theta}) \quad (7.64)$$

as

$$\frac{\partial J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = -2N \operatorname{Re} \left[ \sum_{k=0}^{N-1} \mathbf{S}^\dagger(k) \hat{S}_{vv}^{-1} \tilde{\mathbf{v}}(k, \boldsymbol{\theta}) \right] \quad (7.65)$$

$$\mathbf{M}(\boldsymbol{\theta}) = \frac{\partial^2 J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} = 2N \operatorname{Re} \left[ \sum_{k=0}^{N-1} \mathbf{S}^\dagger(k) \hat{S}_{vv}^{-1} \mathbf{S}(k) \right] \quad (7.66)$$

where  $\mathbf{S}(k)$  is the  $n_o \times n_p$  output sensitivity matrix, given by

$$\mathbf{S}(k) = \frac{\partial \mathbf{G}(k, \boldsymbol{\theta}) \tilde{u}(k)}{\partial \boldsymbol{\theta}} \quad (7.67)$$

The modified Newton-Raphson step is given by Eqs. (7.48) and (7.49), as before.

For a state-space model,  $\mathbf{G}(k, \boldsymbol{\theta})$  is computed from Eq. (7.57a); however,  $\mathbf{G}(k, \boldsymbol{\theta})$  can also be a matrix of transfer functions with transfer function coefficients as the unknown parameters in  $\boldsymbol{\theta}$ .

The output-error parameter estimation algorithm can be easily modified for measured data in the form of frequency response curves. In this case, the cost function takes the form

$$J(\boldsymbol{\theta}) = N \sum_{k=0}^{N-1} \tilde{u}^\dagger(k) [\mathbf{G}_E(k) - \mathbf{G}(k, \boldsymbol{\theta})]^\dagger \hat{S}_{vv}^{-1} [\mathbf{G}_E(k) - \mathbf{G}(k, \boldsymbol{\theta})] \tilde{u}(k) \quad (7.68)$$

where  $\mathbf{G}(k, \boldsymbol{\theta})$  is a matrix with transfer functions as elements, and  $\mathbf{G}_E(k)$  is a matrix of experimentally-determined frequency responses, which can be found using spectral estimates, as described earlier and in Chapter 11.

Cost functions in Eq. (7.64) or (7.68) can be minimized with respect to unknown parameters in  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$ , and  $\mathbf{D}$ , or with respect to transfer function coefficients in the elements of  $\mathbf{G}(k, \boldsymbol{\theta})$ . The parameter estimates are obtained from Eqs. (7.48), (7.49), and (7.65)–(7.67). Spectral densities of the residuals are estimated from Eq. (7.60).

## FREQUENCY DOMAIN METHODS

239

For a dynamic system with a single input, the output-error cost function with measured frequency response curves is defined as

$$J(\boldsymbol{\theta}) = N \sum_{k=0}^{N-1} \tilde{u}^*(k) [\mathbf{G}_E(k) - \mathbf{G}(k, \boldsymbol{\theta})]^\dagger \mathbf{S}_{vv}^{-1} [\mathbf{G}_E(k) - \mathbf{G}(k, \boldsymbol{\theta})] \tilde{u}(k) \quad (7.69)$$

In this formulation, the scalar variable  $\tilde{u}(k)$  can be interpreted as a weighting function quantifying the importance of the measured data in the frequency domain according to the harmonic content of the input.

Another approach is to formulate the cost function without the input weighting,

$$J(\boldsymbol{\theta}) = N \sum_{k=0}^{N-1} [\mathbf{G}_E(k) - \mathbf{G}(k, \boldsymbol{\theta})]^\dagger \mathbf{S}_{vv}^{-1} [\mathbf{G}_E(k) - \mathbf{G}(k, \boldsymbol{\theta})] \quad (7.70)$$

When the cost functions in Eqs. (7.69) and (7.70) are used for a single-input, single-output case, the factor  $N\mathbf{S}_{vv}^{-1}$  can be omitted from the cost without affecting the parameter estimation results.

A variation of Eq. (7.70) is to introduce relative weighting on each term (i.e., for each frequency), according to the coherence. The idea is to weight the frequency-domain data with higher coherence more heavily in the cost function. The coherence is a measure of the linear relationship between inputs and outputs, so that data with high coherence are good in the sense that they are compatible with a linear model structure and not overly corrupted by noise.

Sometimes the cost function is partitioned into separate parts, quantifying the model fit to the magnitude and phase for a Bode plot. To implement this for a single-input, single-output case, the cost function is

$$\begin{aligned} J(\boldsymbol{\theta}) = & \sum_{k=0}^{N-1} [20 \log_{10} |\mathbf{G}_E(k)| - 20 \log_{10} |\mathbf{G}(k, \boldsymbol{\theta})|]^2 \\ & + w \sum_{k=0}^{N-1} \{\arg[\mathbf{G}_E(k)] - \arg[\mathbf{G}(k, \boldsymbol{\theta})]\}^2 \end{aligned} \quad (7.71)$$

where typically the weighting  $w$  is taken as 0.0175, to balance the contributions of the magnitude and phase error components of the cost function. The cost formulation in Eq. (7.71), with coherence weighting, is the basis of the commercially available FORTRAN software called CIFER®, which has been used extensively in practice, particularly for rotorcraft and V/STOL aircraft.

Although the summations over the frequency index  $k$  in the preceding expressions are written to include all frequencies that would be calculated by the discrete Fourier transform, it is not necessary to include all these frequencies. Using selected frequencies would only change the summation indices to correspond to the selected frequencies. Arbitrary frequencies can be used when the transform is done using the high-accuracy Fourier transform algorithm described in Chapter 11. For selected frequencies, there is a difference in interpretation of the results, because the model and estimate of  $\mathbf{S}_{vv}$  will apply only for signal components at the selected frequencies and will not include all of the power in the

time-domain data. Transforming the time-domain data using a small frequency band automatically discards some of the power in the time-domain data. In most cases, this is equivalent to a zero-lag, low-pass filtering operation, which improves the convergence and accuracy of the parameter estimation.

Finally, the weighting matrix  $S_{vv}$  in the cost function is the spectral density of the residuals, which ideally should be estimated as a function of frequency index  $k$  [cf. Eq. (7.60)]. However, in most practical cases, the frequencies used for the analysis correspond to the bandwidth of the dynamic system modes, which is also the bandwidth where most of the residual power resides when the residuals are colored. Therefore, using a constant estimate of  $S_{vv}$  over the frequencies used in the analysis [cf. Eq. (7.60) and Fig. 5.12] is a good representation of the spectral density of the residual power in the frequency domain. Consequently, the noise assumptions in the theory match well with reality, and there is no need to correct the estimated error bounds for colored residuals when the parameter estimation is done in the frequency domain.

## 7.5 Equation-Error Method

Further simplification of the parameter estimation can be made by assuming that all state variables are measured without errors. Including this assumption results in the equation-error method. The dynamic model equation in the frequency domain is

$$j\omega_k \tilde{x}(\omega_k) = A\tilde{x}(\omega_k) + B\tilde{u}(\omega_k) + B_w \tilde{w}(\omega_k) \quad (7.72)$$

where process noise is included, but there is no measurement noise because both the inputs and states are assumed to be measured without errors. Considering the state derivatives as the measured outputs at discrete frequencies  $\omega_k = 2\pi k/T$ ,

$$\tilde{z}(k) = j\omega_k \tilde{x}(k) \quad (7.73)$$

$$\omega_k = 2\pi k/T \quad k = 0, 1, 2, \dots, N-1 \quad (7.74)$$

The model equation then becomes

$$\tilde{z}(k) = A\tilde{x}(k) + B\tilde{u}(k) + \tilde{v}(k) \quad k = 0, 1, 2, \dots, N-1 \quad (7.75)$$

where  $\tilde{v}(k) \equiv B_w \tilde{w}(k)$  are the equation errors, which are Gaussian with

$$E[\tilde{v}(k)] = \mathbf{0} \quad E[\tilde{v}(k)\tilde{v}^\dagger(k)] = \frac{S_{vv}}{N} \quad (7.76)$$

The cost function for equation-error parameter estimation is

$$J(\boldsymbol{\theta}) = N \sum_{k=0}^{N-1} [\tilde{z}(k) - A\tilde{x}(k) - B\tilde{u}(k)]^\dagger S_{vv}^{-1} [\tilde{z}(k) - A\tilde{x}(k) - B\tilde{u}(k)] \quad (7.77)$$

where the innovations

$$\tilde{v}(k) = \tilde{z}(k) - A\tilde{x}(k) - B\tilde{u}(k) \quad (7.78)$$

are equation errors.

## FREQUENCY DOMAIN METHODS

241

Parameter estimates are obtained by minimizing the cost in Eq. (7.77) with respect to the unknown parameters in matrices  $\mathbf{A}$  and  $\mathbf{B}$ . The minimization is applied separately to each state equation, corresponding to each row in the vector equation (7.72). This assumes that the errors in the equations are mutually uncorrelated, so that  $S_{vv}$  is a diagonal matrix. This is generally a very good assumption in practice. Since the equation error is minimized for one equation at a time, the weighting  $NS_{vv}^{-1}$  can be omitted from the least-squares optimization; however, this weighting must be included to compute the estimated parameter error bounds properly. As in the time-domain case, the equation-error method can be used to estimate dimensional or nondimensional aerodynamic parameters, by using the appropriate model equations.

The equation for the measured outputs of the dynamic system,

$$\tilde{\mathbf{z}}(k) = \mathbf{C}\tilde{\mathbf{x}}(k) + \mathbf{D}\tilde{\mathbf{u}}(k) + \tilde{\mathbf{v}}(k) \quad k = 0, 1, 2, \dots, N - 1 \quad (7.79)$$

can also be used to estimate aerodynamic parameters appearing in the  $\mathbf{C}$  and  $\mathbf{D}$  matrices. In this case, the actual measured outputs of the dynamic system are used for  $\tilde{\mathbf{z}}(k)$ , and

$$\tilde{\mathbf{v}}(k) = \tilde{\mathbf{z}}(k) - \mathbf{C}\tilde{\mathbf{x}}(k) - \mathbf{D}\tilde{\mathbf{u}}(k) \quad (7.80)$$

For example, output equations for measured accelerometer outputs can be used to estimate aerodynamic force parameters.

When the model is formulated in terms of a transfer function,

$$\begin{aligned} \tilde{\mathbf{z}}(k) &= G(k, \boldsymbol{\theta})\tilde{\mathbf{u}}(k) + \tilde{\mathbf{v}}(k) \\ &= \frac{\text{num}(k, \boldsymbol{\theta})}{\text{den}(k, \boldsymbol{\theta})}\tilde{\mathbf{u}}(k) + \tilde{\mathbf{v}}(k) \\ &= \frac{\text{num}_1(k, \boldsymbol{\theta})}{(j\omega_k)^{n_s} + \text{den}_1(k, \boldsymbol{\theta})}\tilde{\mathbf{u}}(k) + \tilde{\mathbf{v}}(k) \end{aligned} \quad (7.81)$$

where  $\text{den}_1(k)$  and  $\text{num}_1(k)$  are obtained by dividing  $\text{den}(k, \boldsymbol{\theta})$  and  $\text{num}(k, \boldsymbol{\theta})$  by the coefficient of the highest-order term,  $(j\omega_k)^{n_s}$ , in  $\text{den}(k, \boldsymbol{\theta})$ . The least-squares cost function is

$$J(\boldsymbol{\theta}) = N \sum_{k=0}^{N-1} \tilde{\mathbf{v}}^*(k) S_{vv}^{-1} \tilde{\mathbf{v}}(k) \quad (7.82)$$

where

$$\tilde{\mathbf{v}}(k) = (j\omega)^{n_s} \tilde{\mathbf{z}}(k) + \text{den}_1(k, \boldsymbol{\theta}) \tilde{\mathbf{z}}(k) - \text{num}_1(k, \boldsymbol{\theta}) \tilde{\mathbf{u}}(k) \quad (7.83)$$

In Eq. (7.83), the model parameters are transfer function coefficients, which are estimated as the values that minimize the cost function (7.82).

The main disadvantage of using the cost function in Eq. (7.82) with Eq. (7.83) is the appearance of factors of  $j\omega_k$  for each power of  $s$  in the numerator and

denominator of the transfer function, corresponding to each time derivative. This causes frequency-domain data at higher frequencies to be given increased weighting in the least-squares problem, because factors like  $(j\omega_k)^2$  are larger for high frequencies than for low frequencies.

To fix this problem, the residuals in the cost function can be normalized by factors of  $(j\omega_k)$ . One approach that works well in practice is to divide the residuals by a factor of  $(j\omega_k)^{n_s-1}$ , where  $n_s$  is the order of the denominator of the transfer function, which is also the number of states in the dynamic system model. This effectively removes the high-frequency weighting, which is an artifact of the cost formulation, and therefore has no physical substantiation. The revised cost function is still

$$J(\boldsymbol{\theta}) = N \sum_{k=0}^{N-1} \tilde{v}^*(k) S_{vv}^{-1} \tilde{v}(k) \quad (7.84)$$

but the residuals are changed to

$$\begin{aligned} \tilde{v}(k) &= (j\omega_k) \tilde{z}(k) + [\text{den}_1(k, \boldsymbol{\theta}) / (j\omega_k)^{n_s-1}] \tilde{z}(k) \\ &\quad - [\text{num}_1(k, \boldsymbol{\theta}) / (j\omega_k)^{n_s-1}] \tilde{u}(k) \end{aligned} \quad (7.85)$$

The preceding modification to the cost function is equivalent to scaling the  $k$ th equation to be solved in the least-squares sense by the factor  $1/(j\omega_k)^{n_s-1}$ . This is a weighted least-squares formulation, as discussed in Chapter 5.

As in the case of the output-error method, selected frequencies can be used to generate the finite Fourier transforms for equation-error parameter estimation in the frequency domain. This results in different limits for the summations in the preceding expressions, changes the number of frequencies to  $M \leq N$ , and makes the results applicable for only the frequencies included. Specifying frequencies near the modal frequencies of the physical system results in good signal-to-noise ratios and an automatic filtering effect, which produces better parameter estimates.

The usable range of frequencies for modeling is limited on the low end to

$$\omega_{\min} = \frac{2\pi}{(T/2)} = 4\pi/T \text{ rad/s} \quad (7.86)$$

where  $T$  is the time length of the maneuver in seconds. This allows at least two full sinusoid waveforms over the length of the maneuver for each frequency  $\omega_k$ . The frequency limit at the high end is theoretically the Nyquist frequency, although the practical limit is lower (see Chapter 9).

For both the state-space and transfer function model forms, the resulting parameter estimation is a least-squares linear regression problem with complex numbers. The next section gives the solution for this problem.

The frequency-domain output-error and equation-error methods presented here are based on the material in Refs. 1, 13, and 14. Example applications of the methods are given in Refs. 1 and 13 to 18.

## 7.6 Complex Linear Regression

Linear regression in the frequency domain follows the same approach used in Sec. 5.1 for time-domain data. The general form of the regression equation is

$$\tilde{\mathbf{z}} = \tilde{\mathbf{X}}\boldsymbol{\theta} + \tilde{\mathbf{v}} \quad (7.87)$$

where

$\tilde{\mathbf{z}} = N \times 1$  vector of transformed dependent variable measurements

$\boldsymbol{\theta} = n_p \times 1$  vector of unknown parameters

$\tilde{\mathbf{X}} = N \times n_p$  matrix of vectors of ones and transformed regressors,  $n_p = n + 1$

$\tilde{\mathbf{v}} = N \times 1$  vector of complex measurement errors

and the properties of  $\tilde{\mathbf{v}}$  are

$$E(\tilde{\mathbf{v}}) = \mathbf{0} \quad E(\tilde{\mathbf{v}}\tilde{\mathbf{v}}^\dagger) = \sigma^2 \mathbf{I} \quad (7.88)$$

where the quantity  $\sigma^2$  is a real number representing the squared magnitude of the complex residual.

The least-squares estimate of  $\boldsymbol{\theta}$  given  $\tilde{\mathbf{z}}$  follows from the minimization of

$$J(\boldsymbol{\theta}) = (\tilde{\mathbf{z}} - \tilde{\mathbf{X}}\boldsymbol{\theta})^\dagger (\tilde{\mathbf{z}} - \tilde{\mathbf{X}}\boldsymbol{\theta}) = \tilde{\mathbf{v}}^\dagger \tilde{\mathbf{v}} = |\tilde{\mathbf{v}}|^2 \quad (7.89)$$

which results in the least-squares estimate

$$\hat{\boldsymbol{\theta}} = [\tilde{\mathbf{X}}^\dagger \tilde{\mathbf{X}}]^{-1} \tilde{\mathbf{X}}^\dagger \tilde{\mathbf{z}} \quad (7.90)$$

The covariance matrix of the parameter estimates is

$$\text{Cov}(\hat{\boldsymbol{\theta}}) = E[(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})^\dagger] = \sigma^2 [\tilde{\mathbf{X}}^\dagger \tilde{\mathbf{X}}]^{-1} \quad (7.91)$$

In the last two expressions,  $\tilde{\mathbf{X}}^\dagger \tilde{\mathbf{X}}$  is a Hermitian matrix that is nonsingular and positive definite.

The vector of estimated dependent variables is

$$\hat{\mathbf{y}} = \tilde{\mathbf{X}}\hat{\boldsymbol{\theta}} \quad (7.92)$$

and the unbiased estimate of the fit error variance  $\sigma^2$  is

$$s^2 \equiv \hat{\sigma}^2 = \frac{\tilde{\mathbf{v}}^\dagger \tilde{\mathbf{v}}}{N - n_p} \quad (7.93)$$

where

$$\tilde{\mathbf{v}} = \tilde{\mathbf{z}} - \hat{\mathbf{y}} \quad (7.94)$$

In the preceding development, there was no assumption as to whether the parameter vector was real or complex, so it could be either. In practical aircraft parameter estimation problems, the parameter vector is real, so the expressions for the parameter estimate and covariance matrix are simplified to

$$\hat{\boldsymbol{\theta}} = \left[ \operatorname{Re}(\tilde{\mathbf{X}}^\dagger \tilde{\mathbf{X}}) \right]^{-1} \operatorname{Re}(\tilde{\mathbf{X}}^\dagger \tilde{\mathbf{z}}) \quad (7.95)$$

$$\operatorname{Cov}(\hat{\boldsymbol{\theta}}) = \sigma^2 \left[ \operatorname{Re}(\tilde{\mathbf{X}}^\dagger \tilde{\mathbf{X}}) \right]^{-1} \quad (7.96)$$

and the fit error estimate in Eq. (7.93) remains the same because  $\tilde{\mathbf{v}}^\dagger \tilde{\mathbf{v}}$  is always real.

If  $\tilde{\mathbf{v}}$  is  $\mathbb{N}(\mathbf{0}, \sigma^2 \mathbf{I})$ , the likelihood function of  $\tilde{\mathbf{z}}$  is

$$p(\tilde{\mathbf{z}}) = \frac{1}{\pi^N \sigma^{2N}} \exp \left[ -\frac{|\tilde{\mathbf{z}} - \tilde{\mathbf{X}} \hat{\boldsymbol{\theta}}|^2}{\sigma^2} \right] \quad (7.97)$$

as developed by Klein<sup>1</sup> and Miller.<sup>12</sup> The denominator of the exponent in Eq. (7.97) is  $\sigma^2$ , rather than  $2\sigma^2$  as in the case of a real random variable. This follows from the assumption

$$p(\tilde{\mathbf{z}}) = p[\operatorname{Re}(\tilde{\mathbf{z}}), \operatorname{Im}(\tilde{\mathbf{z}})] = p[\operatorname{Re}(\tilde{\mathbf{z}})] p[\operatorname{Im}(\tilde{\mathbf{z}})]$$

where

$$\tilde{\mathbf{z}} = \operatorname{Re}(\tilde{\mathbf{z}}) + j \operatorname{Im}(\tilde{\mathbf{z}})$$

Maximization of  $p(\tilde{\mathbf{z}})$  with respect to  $\boldsymbol{\theta}$  gives the estimator in Eq. (7.95), which means that the maximum likelihood estimator is identical to the least-squares estimator. Properties of the least-squares estimates are identical to those summarized in Sec. 5.1.1.

Note that the expressions in Eqs. (7.93), (7.95), and (7.96) for complex data are identical to the expressions developed for real data in Chapter 5, Eqs. (5.24), (5.10), and (5.12), since the conjugate transpose and transpose operations are identical operations when applied to real numbers. Therefore, if the equations in this section are used, the same least-squares parameter estimation equations can be used regardless of whether the data are real or complex.

### Example 7.1

In this example, the same flight data used in Examples 5.1 and 6.1 to demonstrate equation-error and output-error parameter estimation techniques in the time domain will now be used to demonstrate the same methods in the frequency domain. The flight data are from a lateral maneuver of the NASA Twin Otter (see Figs. 5.3 and 5.4).

## FREQUENCY DOMAIN METHODS

245

**Table 7.1 Least-squares parameter estimation results, aerodynamic yawing moment coefficient, run 1**

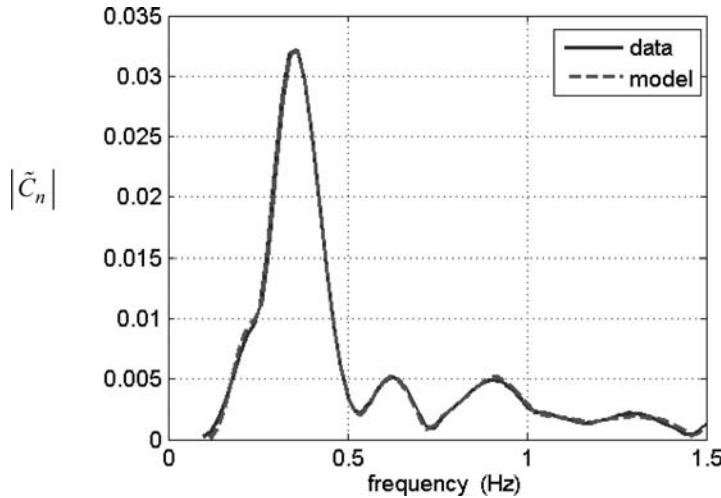
Parameter	$\hat{\theta}$	$s(\hat{\theta})$	$ t_0 $	$100[s(\hat{\theta})/ \theta ]$
$C_{n\beta}$	$8.51 \times 10^{-2}$	$8.29 \times 10^{-4}$	102.5	1.0
$C_{np}$	$-5.52 \times 10^{-2}$	$3.47 \times 10^{-3}$	15.9	6.3
$C_{nr}$	$-2.00 \times 10^{-1}$	$2.92 \times 10^{-3}$	68.7	1.5
$C_{n\delta_a}$	$3.98 \times 10^{-4}$	$1.23 \times 10^{-3}$	0.3	309.4
$C_{n\delta_r}$	$-1.32 \times 10^{-1}$	$1.36 \times 10^{-3}$	96.8	1.0
$C_{no}$	$-1.46 \times 10^{-5}$	$4.06 \times 10^{-5}$	0.4	277.8
$s = \hat{\sigma}$	$1.17 \times 10^{-7}$	—	—	—
$R^2, \%$	99.9	—	—	—

All relevant time series were transformed into the frequency domain using the high-accuracy finite Fourier transform described in Chapter 11, for the frequency vector  $f = [0.10, 0.12, 0.14, \dots, 1.5]^T$  Hz. This frequency range includes all the significant spectral components in the measured input-output data for the maneuver. The same model equation (5.102c) was used for least-squares estimation of yawing moment coefficient parameters in the frequency domain. Equations (7.95) and (7.96) with Eq. (7.93) were used to compute the estimated parameters and covariance matrix.

Table 7.1 contains the parameter estimation results using equation-error linear regression in the frequency domain for the yawing moment coefficient  $C_n$ . The results compare favorably with the analogous time-domain results given in Table 5.1, including the indication that the term associated with the  $C_{n\delta_a}$  parameter is not significant and probably should be omitted from the model structure. Figure 7.2 shows the model fit to measured  $C_n$  in the frequency domain. Note that the plot was made using magnitudes of the complex numbers, which include both real and imaginary parts. The model fit can also be examined by plotting real and imaginary parts separately. Equations (7.95), (7.96), and (7.93) account for both real and imaginary parts in the least-squares modeling.

For output-error parameter estimation in the frequency domain, the cost gradient and information matrix were computed from Eqs. (7.65) and (7.66), and the parameter estimate updates were found from Eq. (7.49). Output sensitivities were computed numerically using central finite differences based on Eq. (7.67), although the sensitivities could also have been computed analytically. Note that analytical computation of the output sensitivities involves only algebraic equations in the frequency domain. Similarly, the outputs are computed algebraically in the frequency domain [cf. Eqs. (7.56c) and (7.57a)]. Estimates of the spectral densities of the residuals were obtained from Eq. (7.60).

In Table 7.2, time-domain output-error results from Example 6.1 are listed along with parameter estimation results using output-error in the frequency domain. The parameter estimation results are in agreement, given the estimated error bounds. The frequency-domain results again indicate that the  $C_{n\delta_a}$  term is insignificant. Figure 7.3 shows the model fit to measured outputs in the frequency domain.



**Fig. 7.2 Frequency-domain equation-error model fit to yawing moment coefficient for lateral maneuver, run 1.**

Because of the close agreement between output-error parameter estimates in the time domain and frequency domain, the frequency-domain model has excellent prediction capability similar to that seen in Example 6.1 for the time-domain model. However, computing predicted outputs in the frequency domain using a frequency-domain model does not require a re-estimation of bias parameters, as in the time-domain case of Example 6.1. Modeling in the frequency domain

**Table 7.2 Output-error parameter estimation results for lateral maneuver, run 1**

Parameter	Time-domain		Frequency-domain	
	$\hat{\theta}$	$s(\hat{\theta})$	$\hat{\theta}$	$s(\hat{\theta})$
$C_{Y\beta}$	$-8.66 \times 10^{-1}$	$2.80 \times 10^{-3}$	$-8.73 \times 10^{-1}$	$7.10 \times 10^{-3}$
$C_{Yr}$	$9.31 \times 10^{-1}$	$1.65 \times 10^{-2}$	$9.53 \times 10^{-1}$	$4.29 \times 10^{-2}$
$C_{Y\delta_r}$	$3.75 \times 10^{-1}$	$7.98 \times 10^{-3}$	$3.40 \times 10^{-1}$	$2.04 \times 10^{-2}$
$C_{l\beta}$	$-1.19 \times 10^{-1}$	$5.60 \times 10^{-4}$	$-1.18 \times 10^{-1}$	$9.61 \times 10^{-4}$
$C_{lp}$	$-5.84 \times 10^{-1}$	$2.39 \times 10^{-3}$	$-5.92 \times 10^{-1}$	$4.26 \times 10^{-3}$
$C_{lr}$	$1.88 \times 10^{-1}$	$1.71 \times 10^{-3}$	$1.75 \times 10^{-1}$	$3.06 \times 10^{-3}$
$C_{l\delta_a}$	$-2.28 \times 10^{-1}$	$8.43 \times 10^{-4}$	$-2.33 \times 10^{-1}$	$1.57 \times 10^{-3}$
$C_{l\delta_r}$	$3.84 \times 10^{-2}$	$7.12 \times 10^{-4}$	$2.33 \times 10^{-2}$	$1.45 \times 10^{-3}$
$C_{n\beta}$	$8.65 \times 10^{-2}$	$2.07 \times 10^{-4}$	$8.77 \times 10^{-2}$	$5.93 \times 10^{-4}$
$C_{np}$	$-6.39 \times 10^{-2}$	$9.49 \times 10^{-4}$	$-5.47 \times 10^{-2}$	$2.65 \times 10^{-3}$
$C_{nr}$	$-1.92 \times 10^{-1}$	$6.07 \times 10^{-4}$	$-1.87 \times 10^{-1}$	$1.87 \times 10^{-3}$
$C_{n\delta_a}$	$-2.73 \times 10^{-3}$	$3.50 \times 10^{-4}$	$7.30 \times 10^{-5}$	$9.73 \times 10^{-4}$
$C_{n\delta_r}$	$-1.36 \times 10^{-1}$	$2.66 \times 10^{-4}$	$-1.36 \times 10^{-1}$	$8.54 \times 10^{-4}$

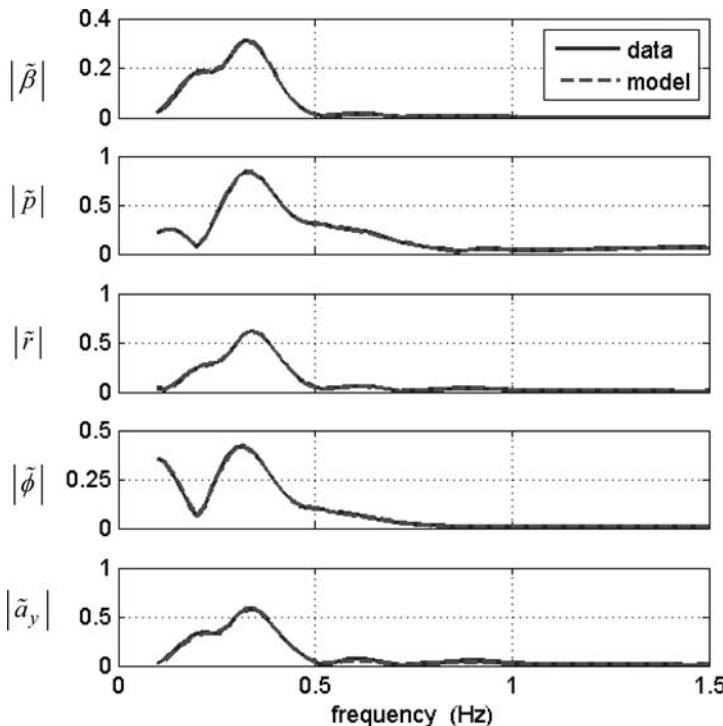


Fig. 7.3 Frequency-domain output-error model fit to measured perturbation outputs for lateral maneuver, run 1.

omits the biases altogether because the biases and trends are removed before Fourier transformation. However, if time-domain matches of predicted model output to measured output are desired using a model identified in the frequency domain, then the bias terms must be estimated in the time domain using the same method shown in Example 6.1 for the time-domain prediction case.  $\square$

### Example 7.2

Frequency domain methods can also be applied to data from forced oscillation tests conducted in a wind tunnel or water tunnel.<sup>17,18</sup> This example uses data from Ref. 18, where a 2.5% model of an F-16XL was oscillated in pitch in a water tunnel. Nondimensional normal force coefficient was computed from measurements of the hydrodynamic normal force on the model and the dynamic pressure. The pitch angle, which was the same as the angle of attack, was commanded and measured during the test. The configuration of the model was fixed, with control surface deflections at zero. A Schroeder sweep (see Chapter 9) was used to command the pitch angle. This input was selected to excite the dynamic system using a uniform power spectrum for many different frequencies in the range [0.008, 0.2] Hz. Figure 7.4

shows the time series of angle of attack and normal force coefficient for an experimental run. The experiment shown involves oscillations in angle of attack of approximately  $\pm 2.5$  deg about the mean angle of attack  $\alpha_o \approx 35$  deg.

The frequencies were chosen to exhibit unsteady aerodynamic effects, so the postulated model had the form

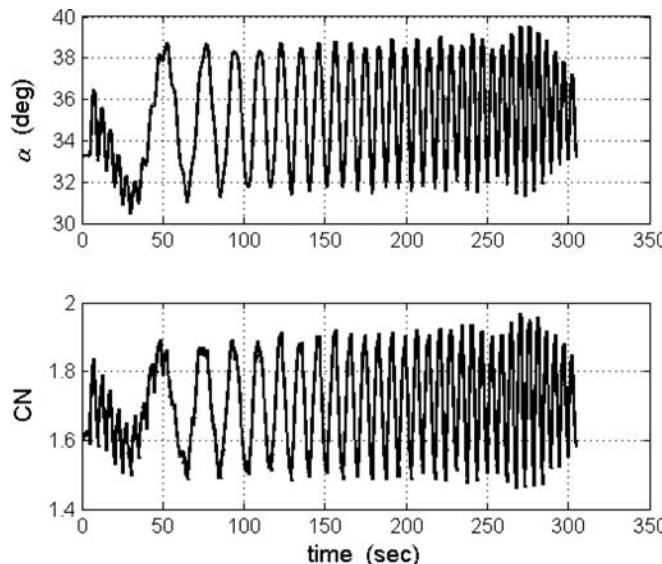
$$C_N(t) = C_N(\infty) + \int_0^t C_{N\alpha}(t-\tau)\dot{\alpha}(\tau)d\tau + \frac{\bar{c}}{2V} \int_0^t C_{Nq}(t-\tau)\dot{q}(\tau)d\tau$$

where  $C_{N\alpha}(t)$  and  $C_{Nq}(t)$  are indicial functions, and  $C_N(\infty)$  is the steady-state normal force coefficient at  $\alpha = q = 0$ . Two assumptions were adopted to simplify the model:

- 1) The effect of  $\dot{q}(t)$  on the normal force coefficient was neglected.
- 2) The indicial function  $C_{N\alpha}(t)$  was modeled using a simple exponential decay function,  $C_{N\alpha}(t) = C_{N\alpha}(\infty) - a_1 e^{-b_1 t}$ , where  $C_{N\alpha}(\infty)$  is the static  $\alpha$  derivative in steady-flow conditions, and  $a_1$  and  $b_1$  are unknown constant parameters.

The simplified model then had the form

$$C_N(t) = C_N(\infty) + C_{N\alpha}(\infty)\alpha(t) - a_1 \int_0^t e^{-b_1(t-\tau)}\dot{\alpha}(\tau)d\tau + \frac{\bar{c}}{2V} C_{Nq}(\infty)q(t) \quad (7.98)$$



**Fig. 7.4** Schroeder sweep in angle of attack and the normal force coefficient response for an F-16XL 2.5% model.

## FREQUENCY DOMAIN METHODS

249

where the integral characterizes the unsteady aerodynamic effect on the normal force coefficient, and the other terms are the usual linear steady-flow terms. Removing the constant term  $C_N(\infty)$ , noting that  $q(t) = \dot{\alpha}(t)$  for one degree-of-freedom pitching motion, and taking Laplace transforms of both sides results in

$$\tilde{C}_N = C_{N\alpha}(\infty)\tilde{\alpha} - \frac{a_1 s}{s + b_1} \tilde{\alpha} + \frac{\bar{c}}{2V} C_{Nq}(\infty) s \tilde{\alpha}$$

Solving for the transfer function of normal force coefficient to angle of attack,

$$\frac{\tilde{C}_N}{\tilde{\alpha}} = \frac{As^2 + Bs + C}{s + b_1}$$

where

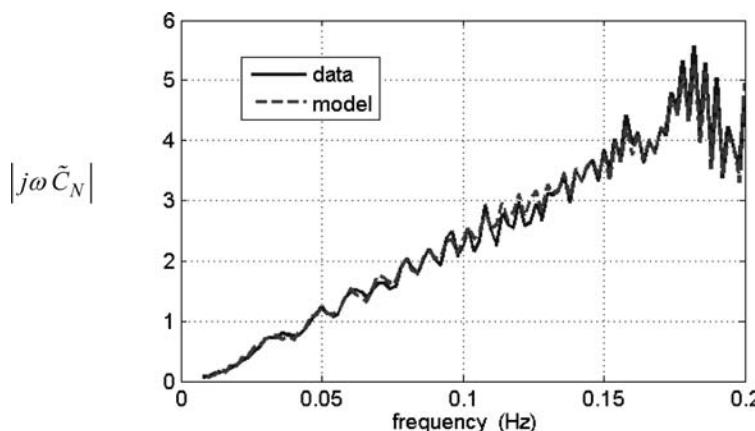
$$A = \frac{\bar{c}}{2V} C_{Nq}(\infty)$$

$$B = C_{N\alpha}(\infty) - a_1 + b_1 \frac{\bar{c}}{2V} C_{Nq}(\infty)$$

$$C = b_1 C_{N\alpha}(\infty)$$

Model parameters  $A, B, C$ , and  $b_1$  can be estimated by applying equation error or output error in the frequency domain to the transfer function model. If  $C_{N\alpha}(\infty)$  is known from static tests, then  $a_1$  can also be computed, resulting in estimates for all of the unknown parameters in the simplified model equation (7.98).

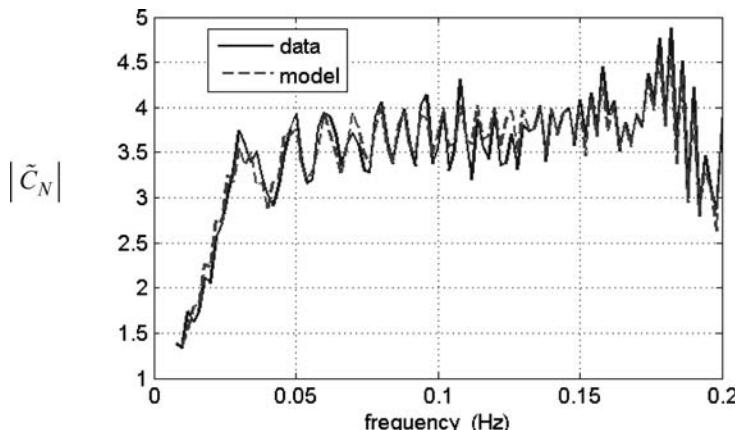
Figure 7.5 shows the model fit to the data in the frequency domain using the equation-error approach with no frequency weighting [cf. Eqs. (7.82) and (7.83)].



**Fig. 7.5** Frequency-domain equation-error model fit to normal force coefficient for an F-16XL 2.5% model.

**Table 7.3 Parameter estimation results for Schroeder sweep forced oscillation on an F-16XL 2.5% model**

Parameter	Frequency-domain equation error		Frequency-domain output error	
	$\hat{\theta}$	$s(\hat{\theta})$	$\hat{\theta}$	$s(\hat{\theta})$
A	$8.58 \times 10^{-1}$	$3.32 \times 10^{-2}$	$8.32 \times 10^{-1}$	$3.44 \times 10^{-2}$
B	$3.22 \times 10^0$	$3.92 \times 10^{-2}$	$3.21 \times 10^0$	$3.17 \times 10^{-2}$
C	$1.83 \times 10^{-1}$	$7.20 \times 10^{-2}$	$1.95 \times 10^{-1}$	$2.80 \times 10^{-2}$
$b_1$	$1.46 \times 10^{-1}$	$2.69 \times 10^{-2}$	$1.55 \times 10^{-1}$	$1.19 \times 10^{-2}$


**Fig. 7.6 Frequency-domain output-error model fit to normal force coefficient for an F-16XL 2.5% model.**

Note the increase in response amplitude with increasing frequency, which is the result of omitting the frequency weighting.

Columns 2 and 3 of Table 7.3 give the estimated parameters and standard errors for this case. Columns 4 and 5 of Table 7.3 contain the analogous results using output-error parameter estimation in the frequency domain. Figure 7.6 shows the output-error model fit to the measured data in the frequency domain.

The parameter results from the two methods are in statistical agreement, given the estimated error bounds.  $\square$

## 7.7 Low-Order Equivalent System Identification

An important application of parameter estimation in the frequency domain is identifying low-order equivalent system (LOES) models from measured data.

A LOES model characterizes the closed-loop dynamic response of the airframe and control system, as it appears to the pilot. Consequently, the inputs for LOES modeling are pilot stick and rudder pedal deflections or forces, and the outputs are aircraft response variables, such as angle of attack or roll rate. LOES models identified from flight-test data are useful in many applications, including control law design validation, simulation, flying qualities research, aircraft development, and aircraft specification compliance.

The LOES model has the same form as the model for an open-loop unaugmented airplane with classical dynamic modes (see Chapter 3), except the inputs are pilot controls with equivalent time delays, instead of control surface deflections. The equivalent time delay is a pure time delay on the control input, introduced to account for delay resulting from the control system implementation (e.g., sampling delay) and the phase lag at high frequency from control system dynamics and various nonlinearities, such as control surface rate limiting. Control system dynamics can include significant dynamics attributable to the control law implementation, artificial feel systems, sensors, filters, and actuators. The complexity of these control systems results from the desire for improved performance and control over expanded flight envelopes. The LOES model attempts to approximate the complete high-order closed-loop aircraft response using a low-order classical linear dynamic model with an equivalent time delay on the input. If this approximation can be done well, the overall closed-loop response can be interpreted more readily in terms of the equivalent low-order model. In addition, there are military specifications<sup>19</sup> that relate values of the LOES model parameters to pilot opinions of the aircraft flying qualities.<sup>20–22</sup> This connection can be used to help interpret and evaluate pilot opinions of aircraft flying qualities.

Since the model structure for LOES modeling is fixed *a priori* to correspond to classical linear aircraft dynamic response with an input time delay, the problem reduces to parameter estimation based on measured data. For the short-period longitudinal dynamic mode, the closed-loop pitch rate response to longitudinal stick deflection is modeled in transfer function form as

$$\frac{\tilde{q}}{\tilde{\eta}_e} = \frac{K_{\theta}(s + 1/T_{\theta_2})e^{-\tau s}}{(s^2 + 2\zeta_{sp}\omega_{sp}s + \omega_{sp}^2)} = \frac{(b_1 s + b_0)e^{-\tau s}}{s^2 + a_1 s + a_0} \quad (7.99)$$

where the relationships among the different parameters can be obtained by direct comparison of the coefficients of powers of  $s$ . The symbol  $\eta_e$  denotes longitudinal stick deflection, and  $\tau$  is the equivalent time delay in seconds. The first model parameterization in Eq. (7.99) is used in flying qualities work, and the second is a generic transfer function form.

Based on the dimensional linear equations for short-period motion derived in Chapter 3,

$$\begin{bmatrix} \dot{\alpha} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} -L_\alpha & 1 - L_q \\ M_\alpha & M_q \end{bmatrix} \begin{bmatrix} \alpha \\ q \end{bmatrix} + \begin{bmatrix} L_{\eta_e} \\ M_{\eta_e} \end{bmatrix} \eta_e(t - \tau) \quad (7.100)$$

where the input is now longitudinal stick deflection, with an equivalent time delay. Assuming  $L_q \approx 0$  for low angles of attack, and  $L_{\eta_e} \approx 0$ , then taking Laplace transforms,

$$\begin{bmatrix} s + L_\alpha & -1 \\ -M_\alpha & s - M_q \end{bmatrix} \begin{bmatrix} \tilde{\alpha} \\ \tilde{q} \end{bmatrix} = \begin{bmatrix} 0 \\ M_{\eta_e} \end{bmatrix} \tilde{\eta}_e e^{-s\tau} \quad (7.101)$$

$$\frac{\tilde{q}}{\tilde{\eta}_e} = \frac{M_{\eta_e}(s + L_\alpha)e^{-s\tau}}{s^2 + (L_\alpha - M_q)s - (M_q L_\alpha + M_\alpha)} \quad (7.102)$$

The transfer function in Eq. (7.102) has the same form as Eq. (7.99), but different parameterization.

Other LOES models can be derived in a similar way for other longitudinal and lateral responses to pilot inputs. Note that the dimensional stability and control derivatives in Eq. (7.102) are equivalent derivatives that include both bare airframe and closed-loop control effects. It follows that these dimensional derivatives are not, in general, the same as those discussed in Chapter 3.

Any of the  $\tilde{q}/\tilde{\eta}_e$  models given earlier could also be expressed in the time domain. For example, the time-domain form of the second version of the transfer function in Eq. (7.99) is

$$\ddot{q}(t) + a_1 \dot{q}(t) + a_0 q(t) = b_1 \eta_e(t - \tau) + b_0 \eta_e(t - \tau) \quad (7.103)$$

Estimating the equivalent time delay parameter  $\tau$  in the time domain is problematic because flight-test data are collected at regular sampling intervals  $\Delta t$ , so interpolation of the measured input data is required to implement a value of  $\tau$  that is not equal to an integer number of sampling intervals. However, interpolation is a smoothing operation, which is a modification of the data. This of course changes the original problem (because the data are changed for every different time lag), causing difficulty in convergence. If values of  $\tau$  are restricted to integer multiples of  $\Delta t$ , resolution of the  $\tau$  estimate can be coarse, and convergence problems can also occur. These problems can be avoided by analyzing the data in the frequency domain, because the time delay parameter appears as an ordinary real-valued model parameter in the frequency domain.

LOES models of the aircraft dynamic response are usually identified in the frequency domain over a frequency band corresponding to typical pilot inputs, [0.1, 10] rad/s. Refs. 5–8 give details of a method that has been used extensively to identify LOES models based on spectral densities estimated from measured data. This method employs output-error in the frequency domain using the cost function formulation in Eq. (7.71) for transfer function parameters.

It is also possible to identify LOES models using Fourier transforms of the measured time-domain data, and applying equation-error or output-error parameter estimation in the frequency domain, as described earlier. This approach avoids estimating spectral densities from measured time-domain data. In fact, the LOES modeling problem is a fairly straightforward application of the techniques described earlier for equation-error or output-error parameter estimation in the frequency domain, as the following example demonstrates.

**Example 7.3**

The test aircraft for this example is the Tu-144LL supersonic transport aircraft, shown in Fig. 7.7. Measured input-output data for a longitudinal multistep 2-1-1 maneuver (see Chapter 9) are shown in Fig. 7.8. The pilot applied the multistep input to the longitudinal control,  $\eta_e$ , which consisted of fore and aft movements of the control wheel in the cockpit. The goal is to identify an accurate LOES model from the measured data, using the model form given in Eq. (7.99), which characterizes the closed-loop pitch rate response for the longitudinal short-period mode. Figure 7.9 shows the single-input, single-output data to be used for LOES identification.

The method used to identify the LOES model, described in Ref. 15, is a two-step approach using equation-error and output-error formulations in the frequency domain to estimate the parameters. Parameter estimates from the equation-error method are used as starting values for the output-error method.

For the equation-error formulation, the cost function is given by Eqs. (7.82) and (7.83) for the LOES model,

$$J(\boldsymbol{\theta}) = M \sum_{k=0}^{M-1} \tilde{v}^*(k) S_{vv}^{-1} \tilde{v}(k) \quad (7.104)$$

where

$$\begin{aligned} \tilde{v}(k) &= (j\omega_k)^2 \tilde{q}(k) - a_1(j\omega_k) \tilde{q}(k) - a_0 \tilde{q}(k) \\ &\quad + b_1(j\omega_k) \tilde{\eta}_e(k) e^{-j\omega_k \tau} + b_0 \tilde{\eta}_e(k) e^{-j\omega_k \tau} \end{aligned} \quad (7.105)$$



Fig. 7.7 Tu-144LL supersonic transport aircraft.

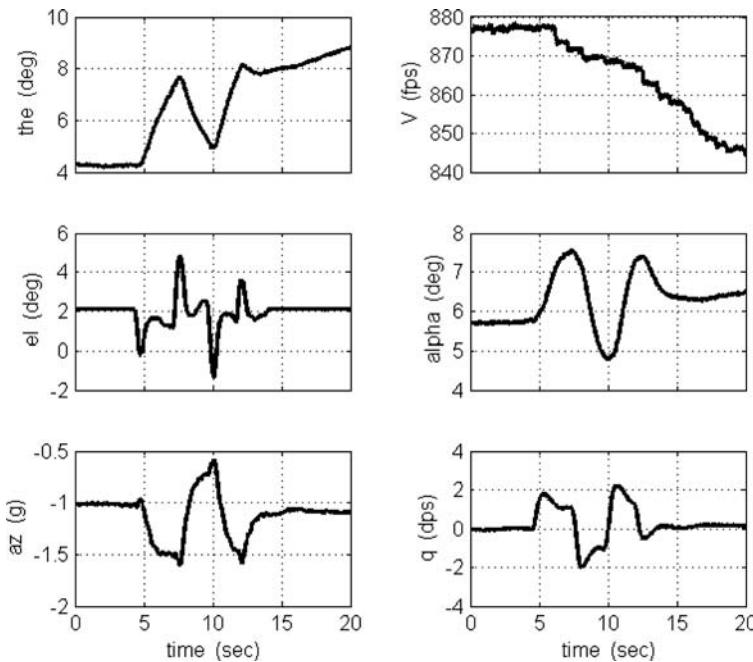


Fig. 7.8 Measured input-output data for a longitudinal maneuver.

Accurate Fourier transforms were computed using the method of Ref. 4, described in Chapter 11, for frequencies evenly spaced at  $\Delta\omega = 0.1 \text{ rad/s}$  in the interval  $[0.6, 10] \text{ rad/s}$ . These selected frequencies are the  $\omega_k$ , not the integral multiples of the lowest frequency given in Eq. (7.74). The number of data points

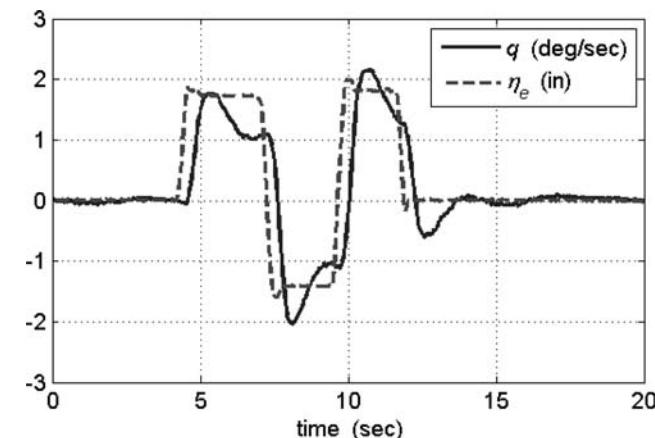


Fig. 7.9 Measured input-output data for longitudinal LOES modeling.

## FREQUENCY DOMAIN METHODS

255

in the frequency domain was 95, so  $M = 95$  in Eq. (7.104). The lower bound of the frequency band for the data analysis corresponds to the 20-second time length of the data,  $\omega_{\min} = 2\pi(2/20) \approx 0.6 \text{ rad/s}$  from Eq. (7.86). The upper bound of the frequency band corresponds to the highest frequency used for LOES modeling, 10 rad/s. The frequency weighting shown in Eq. (7.85) was not included but could have been. In this problem, the order of the transfer function is low ( $n_s = 2$ ), and the selected frequencies  $\omega_k$  are low, so the frequency weighting can be omitted.

Parameter estimation results from the equation-error (EE) method were used as starting values for the output-error (OE) method, which minimized the cost function given in Eq. (7.64):

$$J_{\text{OE}}(\boldsymbol{\theta}) = M \sum_{k=0}^{M-1} \tilde{\mathbf{v}}^*(k, \boldsymbol{\theta}) S_{\mathbf{vv}}^{-1} \tilde{\mathbf{v}}(k, \boldsymbol{\theta}) \quad (7.106)$$

where

$$\tilde{\mathbf{v}}(k, \boldsymbol{\theta}) = \tilde{q}(k) - \frac{[j\omega_k b_1 + b_0]}{[(j\omega_k)^2 + j\omega_k a_1 + a_0]} \tilde{\eta}_e(k) e^{-j\omega_k \tau} \quad (7.107)$$

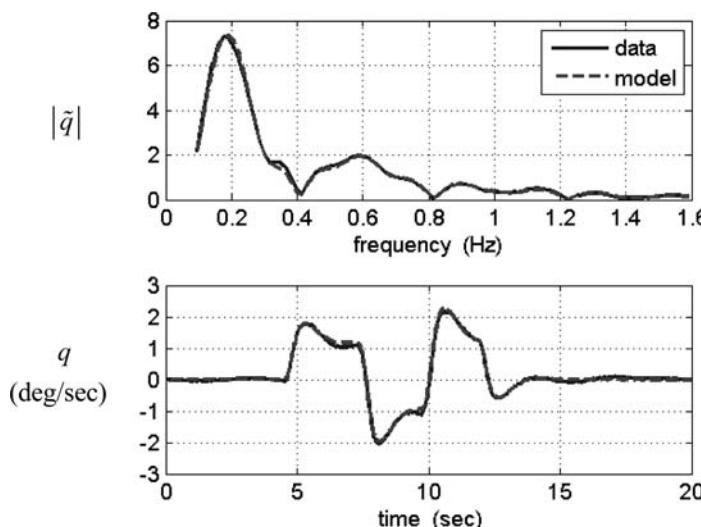
Final parameter estimates were from the output-error formulation, because of known favorable asymptotic properties (see Chapter 6). The equation-error method is much more robust to starting values of the parameters, because the model output depends linearly on all model parameters except the equivalent time delay  $\tau$ . The term  $\tilde{\eta}_e(k) e^{-j\omega_k \tau}$  makes the equation-error parameter estimation a nonlinear optimization problem, as is the case for the output-error parameter estimation. Because of this, the same optimization routine, implementing the modified Newton-Raphson technique, can be used to solve either formulation of the parameter estimation problem. The difference is only in how the model is formulated. Using an EE/OE sequence gives parameter estimation results with good asymptotic properties, without requiring good starting values for the parameters. Estimated parameter standard errors do not require correction, because the analysis in the frequency domain automatically accounts for colored residuals, as mentioned previously.

The time-domain data for this example were measured at sampling intervals  $\Delta t = 0.03125 \text{ s}$  (32 Hz), resulting in 641 data points for the 20-second maneuver. The 95 data points in the frequency domain represent a significant reduction in the number of data points to be processed. In addition, the model equations in the frequency domain are always algebraic, so that there is no need for a numerical integration, such as fourth-order Runge-Kutta. Both of these factors make the computations go very quickly in the frequency domain, even for the nonlinear optimization that must be done for both equation-error and output-error parameter estimation to identify LOES models. Another advantage of frequency-domain techniques is that the dynamic system can be unstable, and the techniques work just the same. In contrast, time-domain output-error methods must integrate differential equations for each iteration of the modified Newton-Raphson technique. This is at best difficult and often not possible for an unstable dynamic system.

**Table 7.4 Tu-144LL longitudinal short-period LOES modeling results,  
 $\alpha_o = 5.8 \text{ deg}$ ,  $M_o = 0.88$**

Parameter	Maneuver 20_4d			
	Equation error		Output error	
	$\hat{\theta}$	$s(\hat{\theta})$	$\hat{\theta}$	$s(\hat{\theta})$
$b_1$	3.916	0.088	3.788	0.168
$b_0$	4.921	0.360	5.181	0.249
$a_1$	3.795	0.097	3.709	0.148
$a_0$	7.128	0.376	7.396	0.258
$\tau$	0.270	0.003	0.268	0.008
$1/T_{\theta_2}$	1.26	—	1.37	—
$\zeta_{sp}$	0.71	—	0.68	—
$\omega_{sp}$	2.67	—	2.72	—

Table 7.4 contains parameter estimation results from the equation-error and output-error parameter estimation in the frequency domain. Figure 7.10 shows that the LOES model fit to the data in the time and frequency domains is very good. The time-domain plot in the lower part of Fig. 7.10 shows that the LOES cannot completely match some high-order response effects, because the low order of the model does not have enough parameters (or model complexity) to



**Fig. 7.10 LOES output-error model fit to pitch rate data for a longitudinal maneuver.**

## FREQUENCY DOMAIN METHODS

257

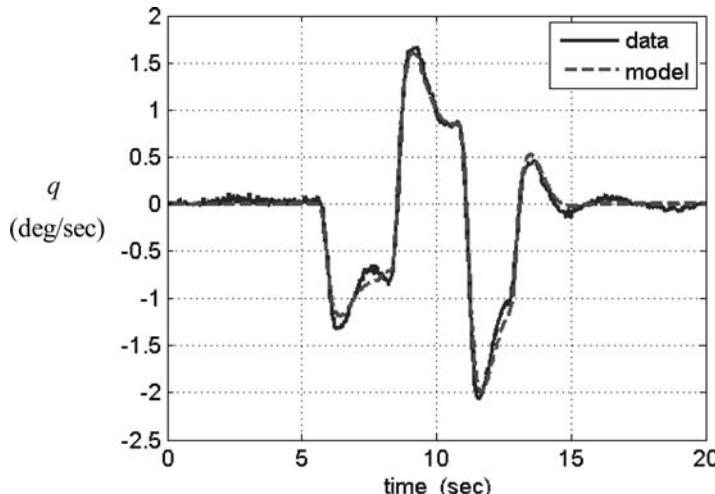


Fig. 7.11 Pitch rate prediction for a longitudinal doublet maneuver.

capture all the dynamics in the real (high-order) physical system. However, the low-order model characterizes the important features of the response very well and has the advantage of easy interpretation, since the LOES model is only second order.

Measured data from a similar maneuver at the same flight condition are shown in Fig. 7.11. The measured longitudinal pilot input from this maneuver was applied to the LOES model identified from the data shown in Fig. 7.9 to compute the pitch rate response for the prediction maneuver. Estimated parameters from the EE/OE sequence (i.e., parameters in column 4 of Table 7.4) were used for the LOES model. Figure 7.11 shows that the match of the model prediction to measured data is very good, and of similar quality to the model fit to the identification data shown in Fig. 7.10. This gives confidence that the identified LOES model has captured the low-order closed-loop short-period pitch rate dynamics of the Tu-144LL at this flight condition. Note that the polarity of the input was reversed for the prediction maneuver in Fig. 7.11, compared to the identification maneuver in Fig. 7.10. This is a check on prediction capability and linearity, since a linear system will respond similarly in either direction and is not sensitive to input form.

The methods described in this example can also be used for multi-input, multi-output models, or for multiple maneuvers by combining the frequency-domain data from each maneuver in the manner described at the end of Sec. 7.1.

The maneuver used for the modeling in this example was a relatively short multistep maneuver. This was possible because Fourier transforms were used directly, and not experimentally derived frequency responses from spectral estimates. Accurate spectral estimates generally require more data, so methods that use experimentally-derived frequency responses will require more data, such as

repeated multistep maneuvers or, more commonly, frequency sweeps (see Chapter 9).  $\square$

## 7.8 Summary and Concluding Remarks

The modeling approaches applied previously to time-domain data were adapted and applied to data transformed into the frequency domain. After a discussion of the data transformation process and the concept of frequency response, the maximum likelihood estimator was developed for frequency-domain data. Following a path similar to Chapter 6 for time-domain data, the output-error and equation-error methods in the frequency domain were developed by making simplifying assumptions in the general problem formulation and then solving the resulting optimization problems. Three aircraft application examples were included to demonstrate the techniques.

Modeling in the frequency domain has several practical advantages, including applicability to unstable dynamic systems, and generally faster computation, as a result of a reduced number of data points in the frequency domain and the fact that the solution to linear differential equations transformed into the frequency domain involves only algebraic operations. It can also be argued that modeling in the frequency domain represents a more robust solution that is more closely aligned with the underlying dynamic system response, because the ordinary or weighted least-squares fitting is applied to spectral components, rather than to individual measured data points, as in time-domain methods. The chapter included an explanation of why error bounds for estimated model parameters computed from frequency-domain data do not need correction for colored residuals.

As with any modeling approach, working in the frequency domain has disadvantages as well. Generally, the transformation into the frequency domain omits all biases and linear trends in the time-domain data, so this information is lost, and the frequency-domain model does not capture the corresponding effects. However, this fact can be considered an advantage when bias and linear trends are not of interest, because the frequency-domain analysis then results in fewer parameters to estimate. Frequency-domain data are generally more difficult to interpret directly, so there is some disconnection with the physical system when working with frequency-domain data. Using spectral estimates can alleviate this problem, because the Bode plot can be readily interpreted in terms of physical characteristics of the dynamic system. The output-error approach in the frequency domain is limited to linear dynamic systems, fundamentally because the finite Fourier transform is a linear operator. On the other hand, the equation-error method based on linear regression can use arbitrarily nonlinear terms in the model, because the nonlinear terms can be generated in the time domain before transformation into the frequency domain.

In general, it is advantageous to have both time-domain and frequency-domain techniques available, so that the appropriate tool for the problem at hand can be applied. These two classes of methods are based on different forms of the data, and differ in the formulations of their respective optimization problems. Consequently, successful comparison between time-domain results and frequency-domain results can increase confidence in the models obtained.

## References

- <sup>1</sup>Klein, V., "Maximum Likelihood Method for Estimating Airplane Stability and Control Parameters from Flight Data in Frequency Domain," NASA TP-1637, 1980.
- <sup>2</sup>Bendat, J. S., and Piersol, A. G., *Random Data Analysis and Measurement Procedures*, 2nd ed., Wiley, New York, 1986.
- <sup>3</sup>Cooley, J. W., and Tukey, J. W., "An Algorithm for the Machine Calculation of Complex Fourier Series," *Mathematics and Computation*, Vol. 19, No. 90, 1965, pp. 297–301.
- <sup>4</sup>Morelli, E. A., "High Accuracy Evaluation of the Finite Fourier Transform Using Sampled Data," NASA TM 110340, 1977.
- <sup>5</sup>Otnes, R. K., and Enochson, L., *Applied Time Series Analysis, Basic Techniques*, Vol. 1, Wiley, New York, 1978.
- <sup>6</sup>Tischler, M. B., "Frequency-Response Identification of the XV-15 Tilt-Rotor Aircraft Dynamics," NASA TM 89428, 1987.
- <sup>7</sup>Tischler, M. B., and Cauffman, M. G., "Comprehensive Identification from Frequency Responses, Vol. 1—Class Notes," NASA CP 10149, 1994.
- <sup>8</sup>Tischler, M. B., and Cauffman, M. G., "Comprehensive Identification from Frequency Responses, Vol. 2—User's Manual," NASA CP 10150, 1994.
- <sup>9</sup>Mehra, R. K., "Identification of Stochastic Linear Dynamic System Using Kalman Filter Representation," *AIAA Journal*, Vol. 9, No. 4, 1971, pp. 28–31.
- <sup>10</sup>Pearson, A. E., "Aerodynamic Parameter Estimation Via Fourier Modulating Function Techniques," NASA CR 4654, 1995.
- <sup>11</sup>Åström, K. J., *Introduction to Stochastic Control*, Academic International Press, New York, 1970.
- <sup>12</sup>Miller, K. S., *Complex Stochastic Processes—An Introduction to Theory and Application*, Addison Wesley Longman, Reading, MA, 1974.
- <sup>13</sup>Klein, V., "Aircraft Parameter Estimation in Frequency Domain," *AIAA Atmospheric Flight Mechanics Conference*, AIAA Paper 78-1344, 1978.
- <sup>14</sup>Klein, V., and Keskar, D. A., "Frequency Domain Identification of a Linear System Using Maximum Likelihood Estimation," *5th IFAC Symposium on Identification and System Parameter Estimation*, Pergamon Press, Vol. 2, 1979, pp. 1039–1046.
- <sup>15</sup>Morelli, E. A., "Identification of Low Order Equivalent System Models from Flight Test Data," NASA TM-2000-210117, 2000.
- <sup>16</sup>Morelli, E. A., "Low-Order Equivalent System Identification for the Tu-144LL Supersonic Transport Aircraft," *Journal of Guidance, Control, and Dynamics*, Vol. 26, No. 2, 2003, pp. 354–362.
- <sup>17</sup>Klein, V., Murphy, P. C., Curry, T. J., and Brandon, J. M., "Analysis of Wind Tunnel Longitudinal Static and Oscillatory Data of the F-16XL Aircraft," NASA/TM-97-206276, 1997.
- <sup>18</sup>Murphy, P. C., and Klein, V., "Validation of Methodology for Estimating Aircraft Unsteady Aerodynamic Parameters from Dynamic Wind Tunnel Tests," *AIAA Atmospheric Flight Mechanics Conference*, AIAA Paper 2003-5397, 2003.
- <sup>19</sup>Military Standard—*Flying Qualities of Piloted Aircraft*, MIL-STD-1797A, Jan. 1990.
- <sup>20</sup>Cooper, G. E., and Harper, R. P. Jr., "The Use of Pilot Rating in the Evaluation of Aircraft Handling Qualities," NASA TN D-5153, 1969.

<sup>21</sup>Hodgkinson, J., LaManna, W. J., and Heyde, J. L., "Handling Qualities of Aircraft with Stability and Control Augmentation Systems—A Fundamental Approach," *Aeronautical Journal*, February 1976, pp. 75–81.

<sup>22</sup>Mitchell, D. G., and Hoh, R. H., "Low-Order Approaches to High-Order Systems: Problems and Promises," *Journal of Guidance, Control, and Dynamics*, Vol. 5, No. 5, 1982, pp. 482–489.

## 8

## Real-Time Parameter Estimation

The techniques presented in Chapters 5–7 apply to a complete set of data that is available after the experiment is completed. Such methods are called batch processing, postflight data analysis, or off-line parameter estimation.

It is also possible to derive parameter estimation algorithms that can be used in real time, giving interim results as the experiment is being conducted. One of the methods to do this is a recursive formulation of ordinary least squares, where parameter estimates are calculated at each sample time when new measured data are available. This is called recursive least squares. The recursive nature of the calculation avoids reprocessing of old data, making the procedure efficient for real-time operation. The extended Kalman filter, which involves augmentation of recursive least squares with information about the noise processes and the dynamic system, can also be used for real-time parameter estimation. Another approach is to use batch methods on recent stretches of data to approximate time-varying parameter estimation. This is usually referred to as sequential least squares. Real-time parameter estimation can also be formulated in the frequency domain, using a recursive finite Fourier transform to provide data for a least-squares solution. All of these methods would be classified as on-line processing or real-time parameter estimation. The short list of methods just given is by no means exhaustive, but all of them have been applied to aircraft system identification problems. Each of these methods will be described in this chapter.

An important application of real-time parameter estimation is characterizing changing aircraft dynamics for real-time control law reconfiguration. One approach for satisfying this requirement is to assume the dynamic model has a linear structure with time-varying parameters. The time variation of the parameters accounts for changes in the flight condition, fuel burn, aircraft configuration changes, or various types of failures, wear, or damage. Allowing the stability and control derivatives in a linear model structure to vary in time can also be used to locally approximate nonlinear aerodynamics. The task is then to estimate time-varying model parameters from measured data in real time, so that adaptive control logic can make the necessary changes to the control law to achieve stability and performance goals. Real-time parameter estimation techniques can also be used for in-flight monitoring of parameter estimates for stability and control testing, flight envelope expansion, or safety monitoring.

The main advantages of real-time parameter estimation can be summarized as follows:

- 1) Calculation of parameter estimates in real time, without having to process the entire data set as additional measurements are added;
- 2) Ability to track time-varying parameters;
- 3) Indication of model structure inadequacy and/or identifiability problems through time variations in the parameter estimates and error bounds;
- 4) Availability of interim parameter estimation results, which can be helpful in evaluating the effectiveness of various input forms and determining appropriate time lengths for excitation. This feature has also proven useful in accident investigations.

The main disadvantages of real-time parameter estimation are as follows:

- 1) The model structure must be fixed. Generally, the approach is to use a fixed linear model structure, but allow the parameters to vary in time. This is an alternative method to account for nonlinearity.
- 2) Many of the methods have a problem called covariance windup, where the estimated parameter variances continually decrease as time goes on, regardless of whether or not there is excitation to the dynamic system. This leads to inaccurate (i.e., optimistic) error bounds for the parameter estimates.
- 3) There is a trade-off between rapid response to time variation in the model parameters and smooth time histories for the parameter estimates. This is generally related to data memory, or how the most recent data are treated relative to older data in terms of weighting in the parameter estimation calculations.
- 4) Periods of low or no excitation combined with a finite data memory can lead to numerical problems for some methods.
- 5) Data processing tasks such as data compatibility analysis and smoothing (see Chapters 10 and 11) usually cannot be done in real time.

Because of the aforementioned disadvantages, real-time parameter estimation often involves some engineering judgment or iterative adjustment of values used in the algorithm, sometimes called tuning parameters. A good example is deciding how long into the past the data memory will extend. The adjustments are typically made in simulation or with similar flight-test data for which parameter estimates are known from batch methods.

There are many parameter estimation methods, but the requirement of being simple enough to be implemented in real time aboard an aircraft narrows the field. In particular, any method that iterates through the data must be eliminated, which means that equation-error formulations are used. As mentioned earlier, the general approach is to assume a linear model structure with model parameters that are allowed to vary with time.

To estimate dimensional parameters, Eqs. (3.126) and (3.130) can be used:

$$q - \dot{\alpha} = L_\alpha \Delta \alpha + L_q q + L_\delta \Delta \delta + L_o \quad (8.1a)$$

$$\dot{q} = M_\alpha \Delta \alpha + M_q q + M_\delta \Delta \delta + M_o \quad (8.1b)$$

$$-\frac{g}{V} \Delta a_z = L_\alpha \Delta \alpha + L_q q + L_\delta \Delta \delta + L_o \quad (8.1c)$$

## REAL-TIME PARAMETER ESTIMATION

263

$$\dot{\beta} - p \sin \alpha + r \cos \alpha - \frac{g \cos \theta}{V} \phi = Y_\beta \beta + Y_p p + Y_r r + Y_\delta \delta + Y_o \quad (8.2a)$$

$$\dot{p} - \frac{I_{xz}}{I_x} \dot{r} = L_\beta \beta + L_p p + L_r r + L_\delta \delta + L_o \quad (8.2b)$$

$$\dot{r} - \frac{I_{xz}}{I_z} \dot{p} = N_\beta \beta + N_p p + N_r r + N_\delta \delta + N_o \quad (8.2c)$$

$$\frac{g}{V} \Delta a_y = Y_\beta \beta + Y_p p + Y_r r + Y_\delta \delta + Y_o \quad (8.2d)$$

Nondimensional parameters can be estimated using the nondimensional forms of the preceding equations, which are Eqs. (3.125), (3.111), and (3.116),

$$\frac{mV}{\bar{q}S}(q - \dot{\alpha}) = C_{L\alpha} \Delta \alpha + C_{Lq} \frac{q\bar{c}}{2V} + C_{L\delta} \Delta \delta + C_{Lo} \quad (8.3a)$$

$$\frac{I_y}{\bar{q}Sc} \dot{q} = C'_{m\alpha} \Delta \alpha + C'_{mq} \frac{q\bar{c}}{2V} + C'_{m\delta} \Delta \delta + C'_{mo} \quad (8.3b)$$

$$-\frac{mg}{\bar{q}S} \Delta a_z = C_{L\alpha} \Delta \alpha + C_{Lq} \frac{q\bar{c}}{2V} + C_{L\delta} \Delta \delta + C_{Lo} \quad (8.3c)$$

$$\begin{aligned} \frac{mV}{\bar{q}S} \left( \dot{\beta} - p \sin \alpha + r \cos \alpha - \frac{g \cos \theta}{V} \phi \right) \\ = C_{Y\beta} \beta + C_{Yp} \frac{pb}{2V} + C_{Yr} \frac{rb}{2V} + C_{Y\delta} \delta + C_{Yo} \end{aligned} \quad (8.4a)$$

$$\frac{1}{\bar{q}Sb} (I_x \dot{p} - I_{xz} \dot{r}) = C_{l\beta} \beta + C_{lp} \frac{pb}{2V} + C_{lr} \frac{rb}{2V} + C_{l\delta} \delta + C_{lo} \quad (8.4b)$$

$$\frac{1}{\bar{q}Sb} (I_z \dot{r} - I_{xz} \dot{p}) = C_{n\beta} \beta + C_{np} \frac{pb}{2V} + C_{nr} \frac{rb}{2V} + C_{n\delta} \delta + C_{no} \quad (8.4c)$$

$$\frac{mg}{\bar{q}S} \Delta a_y = C_{Y\beta} \beta + C_{Yp} \frac{pb}{2V} + C_{Yr} \frac{rb}{2V} + C_{Y\delta} \delta + C_{Yo} \quad (8.4d)$$

It is also possible to compute values for the nondimensional aerodynamic force and moment coefficients based on measurements and the nonlinear moment equations, then set up a linear regression problem, as was done in Chapter 5. Ignoring the effects of rotating mass in the propulsion system, the equations are [cf. Eqs. (5.99) and (5.100)]

$$C_X = \frac{1}{\bar{q}S} (ma_x - T) = C_{X\alpha} \Delta \alpha + C_{Xq} \frac{q\bar{c}}{2V} + C_{X\delta} \Delta \delta + C_{Xo} \quad (8.5a)$$

$$C_Z = \frac{ma_z}{\bar{q}S} = C_{Z\alpha} \Delta \alpha + C_{Zq} \frac{q\bar{c}}{2V} + C_{Z\delta} \Delta \delta + C_{Zo} \quad (8.5b)$$

$$C_D = -C_X \cos \alpha - C_Z \sin \alpha = -\frac{(ma_x - T)}{\bar{q}S} \cos \alpha - \frac{ma_z}{\bar{q}S} \sin \alpha$$

$$= C_{D\alpha} \Delta \alpha + C_{Dq} \frac{q\bar{c}}{2V} + C_{D\delta} \Delta \delta + C_{Do}$$
(8.5c)

$$C_L = -C_Z \cos \alpha + C_X \sin \alpha = -\frac{ma_z}{\bar{q}S} \cos \alpha + \frac{(ma_x - T)}{\bar{q}S} \sin \alpha$$

$$= C_{L\alpha} \Delta \alpha + C_{Lq} \frac{q\bar{c}}{2V} + C_{L\delta} \Delta \delta + C_{Lo}$$
(8.5d)

$$C_m = \frac{1}{\bar{q}Sc} [I_y \dot{q} + (I_x - I_z) pr + I_{xz}(p^2 - r^2)]$$

$$= C_{m\alpha} \Delta \alpha + C_{mq} \frac{q\bar{c}}{2V} + C_{m\delta} \Delta \delta + C_{mo}$$
(8.5e)

$$C_Y = \frac{ma_y}{\bar{q}S} = C_{Y\beta} \beta + C_{Yp} \frac{pb}{2V} + C_{Yr} \frac{rb}{2V} + C_{Y\delta} \delta + C_{Yo}$$
(8.6a)

$$C_l = \frac{1}{\bar{q}Sb} [I_x \dot{p} - I_{xz}(pq + \dot{r}) + (I_z - I_y) qr]$$

$$= C_{l\beta} \beta + C_{lp} \frac{pb}{2V} + C_{lr} \frac{rb}{2V} + C_{l\delta} \delta + C_{lo}$$
(8.6b)

$$C_n = \frac{1}{\bar{q}Sb} [I_z \dot{r} - I_{xz}(\dot{p} - qr) + (I_y - I_x) pq]$$

$$= C_{n\beta} \beta + C_{np} \frac{pb}{2V} + C_{nr} \frac{rb}{2V} + C_{n\delta} \delta + C_{no}$$
(8.6c)

The rest of the chapter describes methods for real-time estimation of model parameters in the preceding equations.

## 8.1 Recursive Least Squares

In Chapter 5, the linear measurement equation was expressed as

$$z = \mathbf{X}\boldsymbol{\theta} + \mathbf{v} \quad (8.7)$$

where

$$E(\mathbf{v}) = \mathbf{0} \quad E(\mathbf{v}\mathbf{v}^T) = \sigma^2 \mathbf{I} \quad (8.8)$$

The ordinary least-squares solution based on  $k$  measurements followed from minimization of the cost function

$$J(\boldsymbol{\theta}) = \frac{1}{2} \sum_{i=1}^k [z(i) - \mathbf{x}^T(i)\boldsymbol{\theta}]^2 \quad (8.9)$$

## REAL-TIME PARAMETER ESTIMATION

265

as

$$\boldsymbol{\theta}(k) = \mathcal{D}(k) \mathbf{X}_k^T \mathbf{Z}_k \quad (8.10a)$$

$$\text{Cov}[\hat{\boldsymbol{\theta}}(k)] = \sigma^2 \mathcal{D}(k) \quad (8.10b)$$

where

$$\mathcal{D}(k) = [\mathbf{X}_k^T \mathbf{X}_k]^{-1} \quad (8.11a)$$

$$\mathbf{X}_k^T = [\mathbf{x}(1) \quad \mathbf{x}(2) \quad \cdots \quad \mathbf{x}(k)] \quad (8.11b)$$

$$\mathbf{Z}_k = [z(1) \quad z(2) \quad \cdots \quad z(k)]^T \quad (8.11c)$$

When a new measurement  $z(k+1)$  is available, the ordinary least-squares solution is

$$\begin{aligned} \boldsymbol{\theta}(k+1) &= \mathcal{D}(k+1) \mathbf{X}_{k+1}^T \mathbf{Z}_{k+1} \\ &= \mathcal{D}(k+1) [\mathbf{X}_k^T \mathbf{Z}_k + \mathbf{x}(k+1) z(k+1)] \end{aligned} \quad (8.12)$$

where

$$\begin{aligned} \mathcal{D}(k+1) &= [\mathbf{X}_{k+1}^T \mathbf{X}_{k+1}]^{-1} \\ &= \left\{ \left[ \begin{array}{c} \mathbf{X}_k \\ \mathbf{x}^T(k+1) \end{array} \right]^T \left[ \begin{array}{c} \mathbf{X}_k \\ \mathbf{x}^T(k+1) \end{array} \right] \right\}^{-1} \\ &= [\mathbf{X}_k^T \mathbf{X}_k + \mathbf{x}(k+1) \mathbf{x}^T(k+1)]^{-1} \\ &= [\mathcal{D}^{-1}(k) + \mathbf{x}(k+1) \mathbf{x}^T(k+1)]^{-1} \end{aligned} \quad (8.13)$$

Applying the matrix inversion lemma (see Appendix A) to the last expression results in

$$\begin{aligned} \mathcal{D}(k+1) &= \mathcal{D}(k) - \mathcal{D}(k) \mathbf{x}(k+1) \\ &\quad \cdot [1 + \mathbf{x}^T(k+1) \mathcal{D}(k) \mathbf{x}(k+1)]^{-1} \mathbf{x}^T(k+1) \mathcal{D}(k) \end{aligned} \quad (8.14)$$

Equation (8.14) represents a direct update of the matrix  $\mathcal{D} = [\mathbf{X}^T \mathbf{X}]^{-1}$ . Using this expression, the inversion of a  $p \times p$  matrix in Eq. (8.13) has been replaced by the inversion of a scalar in Eq. (8.14), which is a simple division.

Substituting Eq. (8.14) into Eq. (8.12) and using Eq. (8.10),

$$\begin{aligned} \hat{\boldsymbol{\theta}}(k+1) &= \hat{\boldsymbol{\theta}}(k) + \mathcal{D}(k) \mathbf{x}(k+1) z(k+1) \\ &\quad - \mathcal{D}(k) \mathbf{x}(k+1) [1 + \mathbf{x}^T(k+1) \mathcal{D}(k) \mathbf{x}(k+1)]^{-1} \\ &\quad \cdot [\mathbf{x}^T(k+1) \hat{\boldsymbol{\theta}}(k) + \mathbf{x}^T(k+1) \mathcal{D}(k) \mathbf{x}(k+1) z(k+1)] \end{aligned}$$

Rearranging the last two terms,

$$\begin{aligned}\hat{\theta}(k+1) = & \hat{\theta}(k) - \mathcal{D}(k)\mathbf{x}(k+1)[1 + \mathbf{x}^T(k+1)\mathcal{D}(k)\mathbf{x}(k+1)]^{-1}\mathbf{x}^T(k+1)\hat{\theta}(k) \\ & + \mathcal{D}(k)\mathbf{x}(k+1)[1 + \mathbf{x}^T(k+1)\mathcal{D}(k)\mathbf{x}(k+1)]^{-1}\{[1 + \mathbf{x}^T(k+1) \\ & \cdot \mathcal{D}(k)\mathbf{x}(k+1) - \mathbf{x}^T(k+1)\mathcal{D}(k)\mathbf{x}(k+1)]z(k+1)\}\end{aligned}$$

or

$$\begin{aligned}\hat{\theta}(k+1) = & \hat{\theta}(k) \\ & + \mathcal{D}(k)\mathbf{x}(k+1)[1 + \mathbf{x}^T(k+1)\mathcal{D}(k)\mathbf{x}(k+1)]^{-1} \\ & \cdot [z(k+1) - \mathbf{x}^T(k+1)\hat{\theta}(k)]\end{aligned}\quad (8.15)$$

Therefore, the recursive least-squares estimate can be computed from the following equations:

$$\hat{\theta}(k+1) = \hat{\theta}(k) + \mathbf{K}(k+1)[z(k+1) - \mathbf{x}^T(k+1)\hat{\theta}(k)] \quad (8.16a)$$

where  $\mathbf{K}(k+1)$  is a time-varying matrix computed as

$$\mathbf{K}(k+1) = \mathcal{D}(k)\mathbf{x}(k+1)[1 + \mathbf{x}^T(k+1)\mathcal{D}(k)\mathbf{x}(k+1)]^{-1} \quad (8.16b)$$

and  $\mathcal{D}(k)$  can be computed recursively from Eq. (8.14),

$$\begin{aligned}\mathcal{D}(k+1) = & \mathcal{D}(k) \\ & - \mathcal{D}(k)\mathbf{x}(k+1)[1 + \mathbf{x}^T(k+1)\mathcal{D}(k)\mathbf{x}(k+1)]^{-1}\mathbf{x}^T(k+1)\mathcal{D}(k)\end{aligned}\quad (8.16c)$$

To use the recursive formulas (8.16), starting values  $\hat{\theta}(0)$  and  $\mathcal{D}(0)$  must be specified. These quantities can be estimated using batch processing in Eqs. (8.9)–(8.11) with an initial data record, or they can be specified using any prior information or parameter estimation results. Recursive least squares can be used to continue a previous analysis by setting  $\hat{\theta}(0)$  and  $\mathcal{D}(0)$  equal to the parameter vector estimate and dispersion matrix obtained from a previous analysis. When there is no prior information about the parameter estimates,  $\hat{\theta}(0)$  can be set to the zero vector, and  $\mathcal{D}(0)$  can be chosen as  $\mathcal{D}(0) = c\mathbf{I}$ , where  $c$  is a large number, e.g.,  $c = 10^6$ . These choices start the algorithm with the statement that the (arbitrary) starting values for the parameters have very large variances, which is equivalent to stating that there is no *a priori* information about the parameter estimates or their accuracy. Another way to interpret this is that the initial information matrix  $\mathbf{M}(0) = \mathcal{D}^{-1}(0) = c^{-1}\mathbf{I}$  will be close to the zero matrix, indicating no information content at the outset.

## REAL-TIME PARAMETER ESTIMATION

267

Equations (8.16) constitute the recursive form of the ordinary least-squares solution in Eqs. (8.10). The matrix  $\mathcal{D}$  must be converted to the parameter covariance matrix [cf. Eqs. (5.12) and (5.13)],

$$\text{Cov}[\hat{\boldsymbol{\theta}}(k)] = \sigma^2 [X_k^T X_k]^{-1} = \sigma^2 \mathcal{D}(k) \equiv \boldsymbol{\Sigma}(k) \quad (8.17)$$

Substituting Eq. (8.17) into (8.16), the recursive least-squares algorithm becomes

$$\hat{\boldsymbol{\theta}}(k+1) = \hat{\boldsymbol{\theta}}(k) + \mathbf{K}(k+1)[z(k+1) - \mathbf{x}^T(k+1)\hat{\boldsymbol{\theta}}(k)] \quad (8.18a)$$

$$\mathbf{K}(k+1) = \boldsymbol{\Sigma}(k)\mathbf{x}(k+1)[\sigma^2 + \mathbf{x}^T(k+1)\boldsymbol{\Sigma}(k)\mathbf{x}(k+1)]^{-1} \quad (8.18b)$$

$$\begin{aligned} \boldsymbol{\Sigma}(k+1) &= \boldsymbol{\Sigma}(k) - \boldsymbol{\Sigma}(k)\mathbf{x}(k+1) \\ &\cdot [\sigma^2 + \mathbf{x}^T(k+1)\boldsymbol{\Sigma}(k)\mathbf{x}(k+1)]^{-1}\mathbf{x}^T(k+1)\boldsymbol{\Sigma}(k) \end{aligned} \quad (8.18c)$$

The recursive computation for the parameter covariance matrix requires the fit error variance  $\sigma^2$ . Usually,  $\sigma^2$  is not known, so it must be replaced by its estimate  $s^2 \equiv \hat{\sigma}^2$ , computed from prior data analysis. The fit error variance can also be estimated recursively using

$$s^2(k+1) = \frac{1}{(k+1)}[ks^2(k) + v^2(k+1)] \quad k \leq 5n_p \quad (8.19a)$$

$$s^2(k+1) = \frac{1}{(k+1-n_p)}[ks^2(k) + v^2(k+1)] \quad k = 5n_p \quad (8.19b)$$

$$s^2(k+1) = \frac{1}{(k+1-n_p)}[(k-n_p)s^2(k) + v^2(k+1)] \quad k > 5n_p \quad (8.19c)$$

where

$$v(k+1) = z(k+1) - \mathbf{x}^T(k+1)\hat{\boldsymbol{\theta}}(k+1) \quad (8.19d)$$

The initial estimates from Eq. (8.19a) are biased because the estimate is not adjusted for the number of parameters in the model,  $n_p$ . The transition value  $5n_p$  is approximate.

The recursive least-squares algorithm given here is equivalent to the batch algorithm for ordinary least squares presented in Chapter 5. This can be inferred from the derivation shown. Results from recursive least squares at the end of a data record should match ordinary least-squares batch processing results based on the same data.

**Example 8.1**

In this example, the flight data used in Chapter 5 to demonstrate batch least-squares parameter estimation is used again with the same model structure

to demonstrate recursive least squares. The measured data, shown in Fig. 5.4, are from a lateral maneuver of the NASA Twin Otter aircraft (see Fig. 5.3). Recursive least squares implemented by Eqs. (8.16) was used to estimate stability and control derivatives associated with the nondimensional yawing moment coefficient. Figure 8.1 shows the time histories of the estimated parameters using recursive least squares. The marks at the right of each plot indicate the parameter estimates from batch least squares, calculated in Chapter 5 (cf. Table 5.1).

The plots in Fig. 8.1 show that the final values of the recursive least-squares parameter estimates match the batch least-squares estimates, as they should. Because of this, using the final parameter estimates shown in Fig. 8.1 produces the same excellent match to the yawing moment coefficient data that was shown in Fig. 5.5.

The recursive least-squares algorithm reveals how the parameter estimates improve and stabilize as information is added from the measurements. Figures 8.2 and 8.3 show how the control derivatives stabilize and approach their final accurate values as information comes into the estimator from the control surface deflection measurements.  $\square$

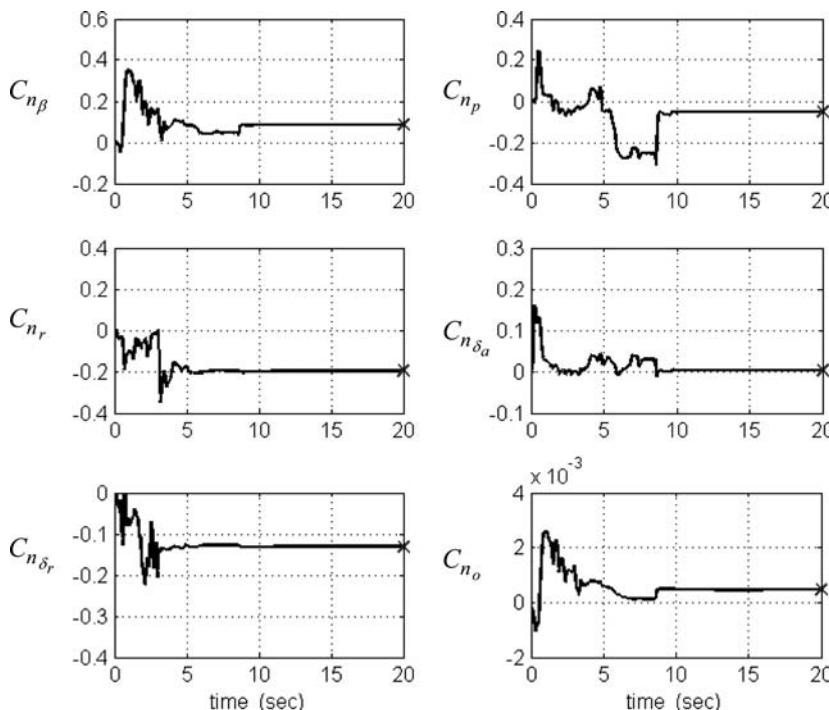


Fig. 8.1 Recursive least-squares parameter estimates for a lateral maneuver.

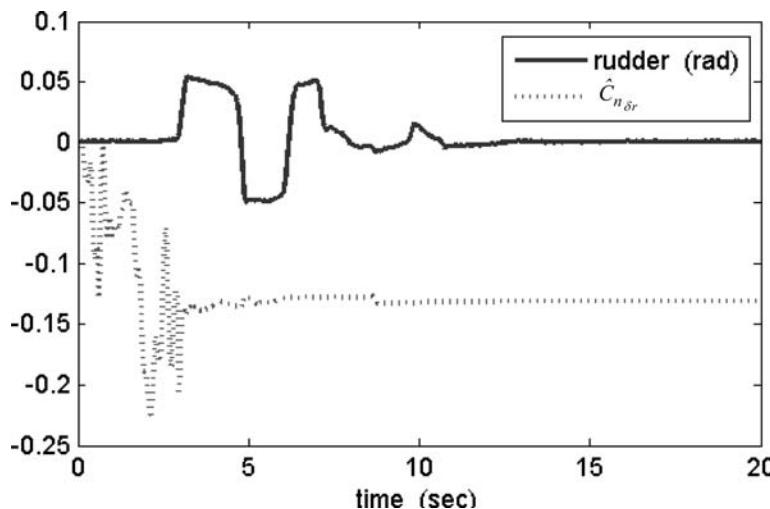


Fig. 8.2 Rudder control derivative recursive estimate convergence.

None of the plots in Figs. 8.1–8.3 show standard error estimates for the recursive parameter estimates. The reason is related to the fact that the covariance matrix calculation requires an estimate of the fit error variance  $s^2 = \hat{\sigma}^2$  [cf. Eq. (8.17)]. The interim fit error variance estimates from Eqs. (8.19) are inaccurate, because they must use current model parameters to calculate each

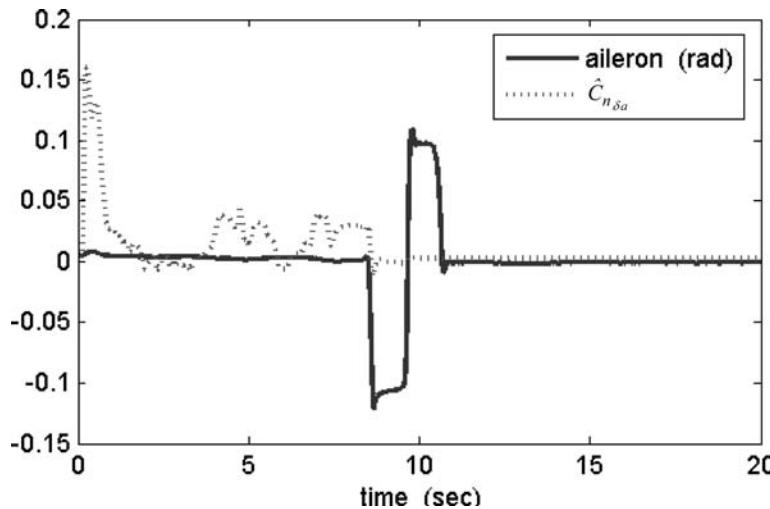


Fig. 8.3 Aileron control derivative recursive estimate convergence.

residual, and the early parameter estimates are inaccurate. Consequently, the fit error estimate varies significantly as time progresses, particularly at the beginning when the estimator has been given little information. Including inaccurate  $s^2$  in the calculation of the estimated parameter covariance matrix leads to inaccurate values for the estimated parameter standard errors. Note that the parameter estimates are unaffected by this issue, because ordinary least squares assumes equal weighting of the equations for each data point (see Chapter 5).

A better approach is to use the estimate  $s^2(N)$  at the end of the maneuver to compute the parameter covariance matrix. However, this approach delays the computation of the parameter standard errors until the end of maneuver when the  $s^2$  estimate is improved. Another approach is to use an *a priori* estimate of  $s^2$ , but this is difficult because  $s^2$  typically includes both wideband noise and narrowband deterministic modeling errors, which are maneuver dependent. There is also the issue of corrections for colored residuals, which has not been addressed at all. All of this highlights the weakness of recursive least squares in the time domain on the issue of obtaining accurate values for estimated parameter uncertainties.

## 8.2 Time-Varying Parameters

The recursive least-squares algorithm described in the last section assumes that the model parameters are constants, so this algorithm is not directly applicable when the parameters are changing with time. As pointed out earlier, model parameters can vary with time in aerospace applications. In such cases, use of the ordinary least-squares estimator can result in estimates that are much different from the true values, with pronounced variations in the parameter estimates as the algorithm proceeds. Several modifications to the recursive least-squares algorithm have been proposed to estimate time-varying parameters. Three of these, the exponentially weighted least squares, the Kalman filter, and sequential least squares, will be described briefly.

### 8.2.1 Exponentially Weighted Least Squares

In this algorithm, old data in the estimation process are gradually devalued and eventually discarded, according to an introduced weighting. The cost function is formulated as

$$J(\boldsymbol{\theta}) = \frac{1}{2} \sum_{i=k-m}^k \lambda^{k-i} [z(i) - \mathbf{x}^T(i)\boldsymbol{\theta}]^2 \quad 0 < \lambda < 1 \quad (8.20)$$

where  $m$  is the number of past values included in the data window weighted by powers of  $\lambda$ , which is often called the forgetting factor. The estimation algorithm that minimizes the cost in Eq. (8.20) can be developed in a manner similar to the

## REAL-TIME PARAMETER ESTIMATION

271

recursive least squares in the last section. The resulting equations are

$$\hat{\boldsymbol{\theta}}(k+1) = \hat{\boldsymbol{\theta}}(k) + \mathbf{K}(k+1)[z(k+1) - \mathbf{x}^T(k+1)\hat{\boldsymbol{\theta}}(k)] \quad (8.21a)$$

$$\mathbf{K}(k+1) = \mathbf{D}(k)\mathbf{x}(k+1)[\lambda + \mathbf{x}^T(k+1)\mathbf{D}(k)\mathbf{x}(k+1)]^{-1} \quad (8.21b)$$

$$\begin{aligned} \mathbf{D}(k+1) &= \frac{1}{\lambda} \{ \mathbf{D}(k) - \mathbf{D}(k)\mathbf{x}(k+1) \\ &\cdot [\lambda + \mathbf{x}^T(k+1)\mathbf{D}(k)\mathbf{x}(k+1)]^{-1}\mathbf{x}^T(k+1)\mathbf{D}(k) \} \end{aligned} \quad (8.21c)$$

For  $\lambda = 1$  and  $m$  corresponding to the entire data record, Eqs. (8.21) revert to recursive least squares. With  $\lambda \ll 1$ , a large weighting is placed on recent data by rapidly fading out older data. Thus, the selection of  $\lambda$  is a compromise between fast adaptation to parameter changes and reduced parameter accuracy due to truncation of the data. Typical values of  $\lambda$  are chosen in the range  $0.9 \leq \lambda < 1.0$ . Further discussion of the algorithm can be found in Refs. 1 to 4.

### 8.2.2 Kalman Filter

In this approach, it is assumed that the parameter variations in time can be modeled by a stochastic linear difference equation,

$$\boldsymbol{\theta}(k) = \Phi(k-1)\boldsymbol{\theta}(k-1) + \mathbf{w}(k-1) \quad (8.22a)$$

with measurement equation

$$z(k) = \mathbf{x}^T(k)\boldsymbol{\theta}(k) + \nu(k) \quad (8.22b)$$

where  $\Phi(k-1)$  is assumed to be known for all  $k$ , and

$$\begin{aligned} E[\mathbf{w}(k)\mathbf{w}^T(l)] &= \mathbf{Q}(k)\delta_{kl} \\ E[\nu(k)\nu(l)] &= \sigma^2(k)\delta_{kl} \end{aligned} \quad (8.22c)$$

For the model specified by Eqs. (8.22), the best linear unbiased estimate of  $\boldsymbol{\theta}$  based on past measurements can be obtained from Kalman filter equations (4.51), realizing that in Eqs. (8.22),  $\Phi(k-1)$  is the state transition matrix and  $\mathbf{x}^T(k)$  relates the state  $\boldsymbol{\theta}(k)$  to the output. Starting with Eqs. (4.51), and replacing  $\mathbf{x}$  with  $\boldsymbol{\theta}$ ,  $\mathbf{C}(k)$  with  $\mathbf{x}^T(k)$ , and  $\Gamma_w(k-1)$  with  $\mathbf{I}$ , the new set of equations takes the form

$$\hat{\boldsymbol{\theta}}(k|k-1) = \Phi(k-1)\hat{\boldsymbol{\theta}}(k-1|k-1) \quad (8.23a)$$

$$\mathbf{P}(k|k-1) = \Phi(k-1)\mathbf{P}(k-1|k-1)\Phi^T(k-1) + \mathbf{Q}(k-1) \quad (8.23b)$$

$$\hat{\boldsymbol{\theta}}(k|k) = \hat{\boldsymbol{\theta}}(k|k-1) + \mathbf{K}(k)[z(k) - \mathbf{x}^T(k)\hat{\boldsymbol{\theta}}(k|k-1)] \quad (8.23c)$$

$$\mathbf{P}(k|k) = [\mathbf{I} - \mathbf{K}(k)\mathbf{x}^T(k)]\mathbf{P}(k|k-1) \quad (8.23d)$$

$$\mathbf{K}(k) = \mathbf{P}(k|k-1)\mathbf{x}(k)[\mathbf{x}^T(k)\mathbf{P}(k|k-1)\mathbf{x}(k) + \sigma^2(k)]^{-1} \quad (8.23e)$$

The state covariance matrix  $\mathbf{P}$  in the preceding equations is the same as the parameter covariance matrix  $\Sigma$  in Eqs. (8.18), since the state vector of the Kalman filter in this case is the parameter vector  $\boldsymbol{\theta}$ .

Since the resulting algorithm is rather complicated for practical application, simplified equations have been suggested.<sup>4</sup> Simplification is achieved by setting  $\Phi(k) = \mathbf{I}$  in Eq. (8.22a), so that the state equation is

$$\hat{\boldsymbol{\theta}}(k) = \hat{\boldsymbol{\theta}}(k-1) + \mathbf{w}(k-1) \quad (8.24)$$

This state equation is equivalent to specifying that the dynamics of the estimated parameters exhibit a random walk behavior. Furthermore, it is assumed that  $\mathbf{w}(k)$  and  $v(k)$  are stationary random sequences, so that the process noise covariance matrix  $\mathbf{Q}$  and the measurement noise variance  $\sigma^2$  are constant. The new set of Kalman filter equations is then

$$\hat{\boldsymbol{\theta}}(k|k-1) = \hat{\boldsymbol{\theta}}(k-1|k-1) \equiv \hat{\boldsymbol{\theta}}(k-1) \quad (8.25a)$$

$$\mathbf{P}(k|k-1) = \mathbf{P}(k-1|k-1) + \mathbf{Q} \quad (8.25b)$$

$$\hat{\boldsymbol{\theta}}(k) = \hat{\boldsymbol{\theta}}(k-1) + \mathbf{K}(k)[z(k) - \mathbf{x}^T(k)\hat{\boldsymbol{\theta}}(k-1)] \quad (8.25c)$$

$$\mathbf{P}(k|k) = [\mathbf{I} - \mathbf{K}(k)\mathbf{x}^T(k)]\mathbf{P}(k|k-1) \quad (8.25d)$$

$$\mathbf{K}(k) = \mathbf{P}(k|k-1)\mathbf{x}(k)[\mathbf{x}^T(k)\mathbf{P}(k|k-1)\mathbf{x}(k) + \sigma^2(k)]^{-1} \quad (8.25e)$$

When  $\mathbf{w}(k) = \mathbf{0}$  for all  $k$ ,  $\mathbf{Q} = \mathbf{0}$ , and the state model is

$$\boldsymbol{\theta}(k) = \boldsymbol{\theta}(k-1) \quad (8.26)$$

which indicates that the parameters are constants. In that case, Eq. (8.25b) becomes  $\mathbf{P}(k|k-1) = \mathbf{P}(k-1|k-1) \equiv \mathbf{P}(k-1)$ , and the algorithm becomes identical to the recursive least-squares algorithm of Eqs. (8.18).

### 8.2.3 Sequential Least Squares

Ordinary least squares can be applied repeatedly to recent measured data to generate a sequence of parameter estimation results. This is the equivalent of approximating time-varying parameters with parameter estimates that are piecewise constant with respect to time. The ordinary least-squares solution applied to the cost function in Eq. (8.20) results in

$$\hat{\boldsymbol{\theta}}(k) = [\mathbf{M}_\lambda(k)]^{-1}\mathbf{S}_\lambda(k) \quad (8.27)$$

$$\text{Cov}[\hat{\boldsymbol{\theta}}(k)] = \sigma^2[\mathbf{M}_\lambda(k)]^{-1} \quad (8.28)$$

where

$$\begin{aligned}\mathbf{M}_\lambda(k) &= \sum_{i=k-m}^k \lambda^{k-i} \mathbf{x}(i) \mathbf{x}^T(i) \\ \mathbf{S}_\lambda(k) &= \sum_{i=k-m}^k \lambda^{k-i} \mathbf{x}(i) z(i)\end{aligned}\quad (8.29)$$

and the fit error variance estimate  $s^2 = \hat{\sigma}^2$  can be computed using Eqs. (8.19), or from prior data.

The quantities  $\mathbf{M}_\lambda(k)$  and  $\mathbf{S}_\lambda(k)$  can be updated recursively using

$$\begin{aligned}\mathbf{M}_\lambda(k) &= \lambda \mathbf{M}_\lambda(k-1) + \mathbf{x}(k) \mathbf{x}^T(k) \\ \mathbf{S}_\lambda(k) &= \lambda \mathbf{S}_\lambda(k-1) + \mathbf{x}(k) z(k)\end{aligned}\quad (8.30)$$

In the preceding expressions, the data window is not limited, but repeated multiplications by the forgetting factor  $\lambda$  make the influence of the older data approach zero.

The sequential estimates and covariance matrix can be computed from Eqs. (8.27) and (8.28) at any sample time  $k$ , using the latest updated values of  $\mathbf{M}_\lambda(k)$  and  $\mathbf{S}_\lambda(k)$  from Eq. (8.30) and  $s^2(k) = \hat{\sigma}^2$  from Eqs. (8.19). However, the parameter estimate update calculations are typically done at a slower rate, such as 1 or 2 Hz. This saves computation time, while also providing a good approximation to the time variation in the parameters. The matrix inversion  $[\mathbf{M}_\lambda(k)]^{-1}$  in Eqs. (8.27) and (8.28) can be done efficiently using Cholesky factorization or singular value decomposition.<sup>5</sup>

### 8.3 Regularization

Aircraft flight often includes extended periods of steady conditions. If such periods are at least as long as the data windowing implemented by the forgetting factor  $\lambda$  in Eq. (8.20), then the information in the regressors is mainly noise. This obviously has very detrimental effects on the accuracy of the estimated parameters. To address the problem, prior information about the parameters can be introduced to regularize the information matrix, and thereby numerically stabilize the parameter estimation. This is the same singular information matrix issue discussed in Chapter 5. The discussion here is modified slightly for the real-time parameter estimation problem, and is based on the work presented by Ward et al.<sup>6</sup>

To include prior information in the parameter estimation process, the cost in Eq. (8.20) can be augmented with constraint equations of the form

$$\mathbf{C}(t) = \mathbf{L}\boldsymbol{\theta}(t) \quad (8.31)$$

where  $\mathbf{C}(t)$  is a time-varying vector of constraint values, and  $\mathbf{L}$  is a constant matrix. If the number of constraint equations is  $n_c$ , then  $\mathbf{L}$  is  $n_c \times n_p$ , and  $\mathbf{C}(t)$

is an  $n_c \times 1$  vector. This general form of the constraint equations can be used to incorporate *a priori* parameter estimates (spatial constraints), or to limit time variations in the real-time parameter estimates (temporal constraints), or to enforce relationships among the parameters (physical constraints).<sup>6</sup> The augmented cost takes the form

$$\begin{aligned} J[\boldsymbol{\theta}(k)] = & \frac{1}{2} \sum_{i=k-m}^k \lambda^{k-i} [z(i) - \mathbf{x}^T(i)\boldsymbol{\theta}(k)]^2 \\ & + \frac{\gamma}{2} [\mathbf{C}(k) - \mathbf{L}\boldsymbol{\theta}(k)]^T \mathbf{W} [\mathbf{C}(k) - \mathbf{L}\boldsymbol{\theta}(k)] \end{aligned} \quad (8.32)$$

where  $\mathbf{W}$  is a diagonal weighting matrix and  $\gamma$  is proportional to the area under the windowing function applied to the residuals in the cost function,

$$\gamma = \sum_{i=k-m}^k \lambda^{k-i} \quad (8.33)$$

The constant  $\gamma$  is included in the constraint term of the cost so that the weighting implemented by the matrix  $\mathbf{W}$  will not be affected by the choice of  $\lambda$ . The diagonal elements of the weighting matrix  $\mathbf{W}$  quantify the relative influence of the prior information in each constraint equation, compared with recent measured data. Weighting matrix elements are usually chosen using simulation data or flight-test data for which batch estimates of the parameters are known.

Taking the partial derivative of the cost with respect to  $\boldsymbol{\theta}(k)$ , and setting the result equal to zero gives the normal equations,

$$\begin{aligned} & \sum_{i=k-m}^k \lambda^{k-i} \mathbf{x}(i) \mathbf{x}^T(i) \boldsymbol{\theta}(k) + \gamma \mathbf{L}^T \mathbf{W} \mathbf{L} \boldsymbol{\theta}(k) \\ & - \sum_{i=k-m}^k \lambda^{k-i} \mathbf{x}(i) z(i) - \gamma \mathbf{L}^T \mathbf{W} \mathbf{C}(k) = \mathbf{0} \end{aligned} \quad (8.34)$$

Solving these equations for  $\boldsymbol{\theta}(k)$  gives the constrained least-squares estimate,

$$\hat{\boldsymbol{\theta}}(k) = [\mathbf{M}_\lambda(k) + \gamma \mathbf{L}^T \mathbf{W} \mathbf{L}]^{-1} [\mathbf{S}_\lambda(k) + \gamma \mathbf{L}^T \mathbf{W} \mathbf{C}(k)] \quad (8.35)$$

where

$$\begin{aligned} \mathbf{M}_\lambda(k) &= \sum_{i=k-m}^k \lambda^{k-i} \mathbf{x}(i) \mathbf{x}^T(i) \\ \mathbf{S}_\lambda(k) &= \sum_{i=k-m}^k \lambda^{k-i} \mathbf{x}(i) z(i) \end{aligned} \quad (8.36)$$

Note in Eq. (8.35) that the prior information in the constraints leads to a regularization term for the matrix to be inverted. The term due to the prior information in the rightmost brackets of Eq. (8.35) also influences the parameter estimates, unless there is significant information in the recent data, in which case the  $S_\lambda(k)$  term dominates. The selection of weighting matrix  $W$  determines the information level for transition between the estimates being influenced by the prior information as opposed to recent measured data.

The quantities  $M_\lambda(k)$  and  $S_\lambda(k)$  can be updated recursively as shown previously in Eq. (8.30). With these recursive updates, real-time parameter estimation can be done using Eq. (8.35) to estimate the parameter vector at selected times, which is just the sequential least-squares approach described earlier. This use of sequential least squares with prior information to regularize the parameter estimation has been called modified sequential least squares.<sup>6</sup>

## 8.4 Frequency-Domain Sequential Least Squares

Sequential least squares can also be implemented in the frequency domain. This approach has some practical advantages, as will be discussed later. As in the time-domain sequential least squares, parameter estimation is repeated at short intervals to produce piecewise-constant estimates for time-varying model parameters in a linear model structure.

### 8.4.1 Equation Error in the Frequency Domain

The finite Fourier transform of a signal  $x(t)$  is defined by (cf. Chapter 7)

$$\tilde{x}(\omega) \equiv \int_0^T x(t)e^{-j\omega t} dt \quad (8.37)$$

which can be approximated by

$$\tilde{x}(\omega) \approx \Delta t \sum_{i=0}^{N-1} x(i)e^{-j\omega i\Delta t} \quad (8.38)$$

where  $x(i) \equiv x(i\Delta t)$ , and  $\Delta t$  is the sampling interval. The summation in Eq. (8.38) is defined as the discrete Fourier transform, as discussed in Chapter 7,

$$X(\omega) \equiv \sum_{i=0}^{N-1} x(i)e^{-j\omega i\Delta t} \quad (8.39)$$

so that the finite Fourier transform approximation in Eq. (8.38) can be written as

$$\tilde{x}(\omega) \approx \Delta t X(\omega) \quad (8.40)$$

Chapter 11 details some fairly straightforward corrections that can be made to remove the inaccuracy resulting from the fact that Eq. (8.40) is a simple Euler approximation to the finite Fourier transform of Eq. (8.37). However, if the sampling rate is much higher than the frequencies of interest (as is often the case for flight data analysis), then the corrections are relatively small and can be safely ignored for real-time parameter estimation, to reduce computations.

Applying the Fourier transform to Eq. (8.1b), for example, gives

$$j\omega_k \tilde{q}(k) = M_\alpha \tilde{\alpha}(k) + M_q \tilde{q}(k) + M_\delta \tilde{\delta}(k) \quad k = 1, 2, \dots, M \quad (8.41)$$

where the  $\Delta$  notation indicating perturbation quantities has been dropped. The Fourier transforms are done for  $M$  frequencies of interest  $\omega_k$ ,  $k = 1, 2, \dots, M$ . The index  $k$  has been used to denote dependence of the Fourier transform values on the frequencies  $\omega_k$ .

The least-squares cost function to be minimized is

$$J(\boldsymbol{\theta}) = \frac{1}{2} \sum_{k=1}^M \left| j\omega_k \tilde{q}(k) - M_\alpha \tilde{\alpha}(k) - M_q \tilde{q}(k) - M_\delta \tilde{\delta}(k) \right|^2 \quad (8.42)$$

where the vertical lines denote the modulus of the complex number enclosed. The symbol  $\tilde{q}(k)$  denotes the Fourier transform of the measured pitch rate for frequency  $\omega_k$ , and similarly for the other terms. This is the equation-error method in the frequency domain, described in Sec. 7.5. Similar cost expressions can be written for other individual state and output equations given earlier in this chapter.

As in Chapter 7, estimation of the unknown parameters can be formulated as a standard least-squares regression problem with complex data. For this example,

$$\tilde{z} = \tilde{X}\boldsymbol{\theta} + \tilde{\nu} \quad (8.43)$$

where

$$\tilde{z} \equiv \begin{bmatrix} j\omega_1 \tilde{q}(1) \\ j\omega_2 \tilde{q}(2) \\ \vdots \\ j\omega_M \tilde{q}(M) \end{bmatrix} \quad (8.44)$$

$$\tilde{\mathbf{X}} \equiv \begin{bmatrix} \tilde{\alpha}(1) & \tilde{q}(1) & \tilde{\delta}(1) \\ \tilde{\alpha}(2) & \tilde{q}(2) & \tilde{\delta}(2) \\ \vdots & \vdots & \vdots \\ \tilde{\alpha}(M) & \tilde{q}(M) & \tilde{\delta}(M) \end{bmatrix} \quad (8.45)$$

$$\boldsymbol{\theta} = \begin{bmatrix} M_\alpha \\ M_q \\ M_\delta \end{bmatrix} \quad (8.46)$$

and  $\tilde{\mathbf{v}}$  represents the complex equation error in the frequency domain. The least-squares cost function is

$$J(\boldsymbol{\theta}) = \frac{1}{2}(\tilde{\mathbf{z}} - \tilde{\mathbf{X}}\boldsymbol{\theta})^\dagger(\tilde{\mathbf{z}} - \tilde{\mathbf{X}}\boldsymbol{\theta}) \quad (8.47)$$

which is the same as the cost in Eq. (8.42) when Eqs. (8.44–8.46) are used. The parameter vector estimate that minimizes this cost function is computed from (cf. Sec. 7.6),

$$\hat{\boldsymbol{\theta}} = [\text{Re}(\tilde{\mathbf{X}}^\dagger \tilde{\mathbf{X}})]^{-1} \text{Re}(\tilde{\mathbf{X}}^\dagger \tilde{\mathbf{z}}) \quad (8.48)$$

with estimated parameter covariance matrix

$$\text{Cov}(\hat{\boldsymbol{\theta}}) \equiv E[(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})^T] = \sigma^2 [\text{Re}(\tilde{\mathbf{X}}^\dagger \tilde{\mathbf{X}})]^{-1} \quad (8.49)$$

The equation-error variance  $\sigma^2$  can be estimated from the residuals,

$$s^2 \equiv \hat{\sigma}^2 = \frac{1}{(M - n_p)} [(\tilde{\mathbf{z}} - \tilde{\mathbf{X}}\hat{\boldsymbol{\theta}})^\dagger(\tilde{\mathbf{z}} - \tilde{\mathbf{X}}\hat{\boldsymbol{\theta}})] \quad (8.50)$$

where  $n_p$  is the number of elements in parameter vector  $\boldsymbol{\theta}$ , so  $n_p = 3$  for this example. Parameter standard errors are computed as the square root of the diagonal elements of the  $\text{Cov}(\hat{\boldsymbol{\theta}})$  matrix from Eq. (8.49), using  $\hat{\sigma}^2$  from Eq. (8.50).

To implement sequential least-squares parameter estimation in the frequency domain, the preceding parameter estimation equations are applied to frequency-domain data at selected regular time intervals. The frequency-domain data must be available at any time, so the Fourier transforms are computed using a recursive Fourier transform, which is described next.

### 8.4.2 Recursive Fourier Transform

For a given frequency  $\omega$ , the discrete Fourier transform in Eq. (8.39) at sample time  $i \Delta t$  is related to the discrete Fourier transform at time  $(i - 1)\Delta t$  by

$$X_i(\omega) = X_{i-1}(\omega) + x(i)e^{-j\omega i \Delta t} \quad (8.51)$$

where

$$e^{-j\omega i \Delta t} = e^{-j\omega \Delta t} e^{-j\omega(i-1)\Delta t} \quad (8.52)$$

The quantity  $e^{-j\omega \Delta t}$  is constant for a given frequency  $\omega$  and constant sampling interval  $\Delta t$ . It follows that the discrete Fourier transform can be computed for a given frequency at each time step using one addition in Eq. (8.51) and two multiplications—one in Eq. (8.52) using the stored constant  $e^{-j\omega \Delta t}$  for frequency  $\omega$ , and one in Eq. (8.51). There is no need to store the time-domain data in memory when computing the discrete Fourier transform in this way, because each sampled data point is processed immediately. Time-domain data from all preceding maneuvers can be used in all subsequent analysis by simply continuing the recursive calculation of the Fourier transform. In this sense, the recursive Fourier transform acts as memory for the information in the data. More data from more maneuvers improve the quality of the data in the frequency domain without increasing memory requirements to store it. Furthermore, the Fourier transform is available at any time  $i \Delta t$ . The approximation to the finite Fourier transform is completed using Eq. (8.40).

The recursive computation of the Fourier transform does not use a fast Fourier transform (FFT) algorithm, and therefore would be comparatively slow, if the entire frequency band up to the Nyquist frequency (see Chapter 9) were of interest. However, rigid-body dynamics of piloted aircraft lie in the rather narrow frequency band of approximately [0.01, 2.0] Hz. Since the frequency band is limited, it is efficient to compute the discrete Fourier transform using Eqs. (8.51) and (8.52), which are a recursive formulation of Eq. (8.39), for selected frequencies  $\omega_k$ ,  $k = 1, 2, \dots, M$ . With this approach, it is possible to select closely spaced fixed frequencies for the Fourier transform and the subsequent data analysis.

Excluding zero frequency removes trim values and measurement biases, so it is not necessary to estimate bias parameters. Using a limited frequency band for the Fourier transformation confines the data analysis to the frequency band where the dynamics reside, and automatically filters wideband measurement noise or structural response outside the frequency band of interest. These automatic filtering features are important for real-time applications, where data compatibility analysis (see Chapter 10) and filtering (see Chapter 11) would require additional computational resources that may not be available.

In past work on fighter aircraft short-period modeling, frequency spacing of 0.04 Hz on the interval [0.1, 1.5] Hz was found to be adequate, giving 36 evenly spaced frequencies for each transformed time-domain signal.<sup>7</sup> Finer frequency spacing requires slightly more computation, but was found to have little effect on the results. When the frequency spacing is very coarse, there is a danger of omitting important frequency components, and this can lead to

## REAL-TIME PARAMETER ESTIMATION

279

inaccurate parameter estimates. In general, a good rule of thumb is to use frequencies evenly spaced at 0.04 Hz over the bandwidth for the dynamic system. For good results, the bandwidth should be limited to the frequency range where the signal components in the frequency domain are at least twice the amplitude of the wideband noise level. However, the algorithm is robust to these design choices, so the selections to be made are not difficult.

For airplane dynamic modeling, the number of time-domain signals to be transformed is usually low (9 or less—more if there are many control surfaces), so this approach requires a small amount of computer memory. Since the data analysis is done in the frequency domain, the memory required is fixed and independent of the time length of the flight maneuvers. Each state and control requires memory for  $M$  complex numbers to hold the current values of its Fourier transform for  $M$  frequencies.

The recursive Fourier transform update need not be done for every sampled time point. Skipping some time points effectively decimates the data before Fourier transformation. This saves computation, and does not adversely impact the frequency-domain data, provided that aliasing is avoided (see Chapter 9).

The states and controls in the linear equations are perturbation quantities, not the measured quantities themselves. Therefore, it is first necessary to remove the trim values, which would more generally be called the constant offsets or biases in cases where there is no distinct trim condition. The biases can be removed using a high-pass filter on all of the signals before the recursive Fourier transform is applied. If the bias in each signal is not removed in this way before the recursive Fourier transform, then the relatively large spectral component at zero frequency due to the bias will spill over to neighboring frequencies. This pollutes the frequency content at the lower frequencies, which is detrimental to the modeling results.

As discussed in Chapter 7, the standard errors computed from the covariance matrix in Eq. (8.49) already include the corrections for colored residuals. These values therefore are a good representation of the error in the estimated parameters. Having high-quality error measures is important for problems such as failure detection and control law reconfiguration. As in the case of time-domain sequential least squares, the parameter estimation and covariance calculations in Eqs. (8.48) and (8.49) can be done at any time, but are usually done at 1 or 2 Hz, to save computations. Linearized aircraft dynamic characteristics rarely change faster than this, except in cases of strong nonlinearity, failures, or rapid maneuvering.

The fit error variance estimate from Eq. (8.50) is well conditioned and appropriate for each covariance matrix calculation. This means that the error bounds computed at any time are an appropriate representation of the parameter accuracy, based on the information available.

Finally, the recursive Fourier transform in Eqs. (8.51) and (8.52) represents a data information memory for as long as the running sum is incremented. It follows that when the aircraft dynamics change, the older data should be discounted in some way, as was done in the time domain using the forgetting factor. If this is not done, then the speed of response for the real-time parameter estimator is progressively degraded, as new information has to overwhelm an increasingly long memory. It is possible to trigger a reduction in the magnitudes of the Fourier transforms computed from past data based on an event, such as a

detected failure or a significant increment in the information content of the measured data. In the latter case, the detected increase in recent data information content would presumably support an update to the parameter estimates. In practical cases, the criteria and discounting schemes for data forgetting depend on the problem and the aircraft.

### Example 8.2

Returning again to the data from the lateral maneuver on the NASA Twin Otter aircraft, the yawing moment parameters will now be estimated using sequential least squares in the frequency domain. The measured data were plotted in Fig. 5.4. The same model equation (5.102c) was used, but the data were transformed into the frequency domain using the recursive Fourier transform of Eqs. (8.51) and (8.52) for frequencies spaced evenly at 0.04 Hz over the frequency range [0.1, 1.5] Hz, which corresponds to a spacing of 0.25 rad/s over the range [0.63, 9.42] rad/s.

Figure 8.4 shows the sequential parameter estimates, including the estimated 95% confidence interval ( $\pm 2$  standard errors), shown as a vertical bar on each estimate. Parameter estimation calculations in Eqs. (8.48)–(8.50) were done at

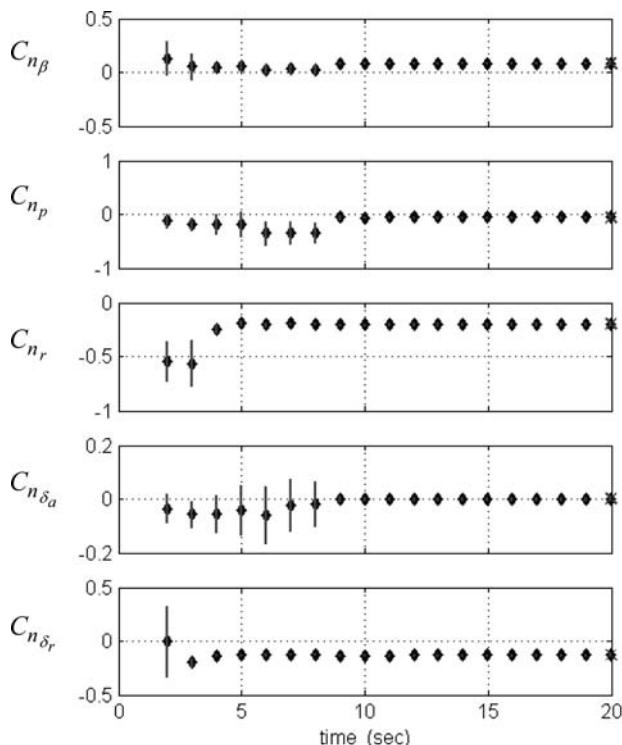


Fig. 8.4 Sequential least-squares parameter estimates in the frequency domain for a lateral maneuver.

## REAL-TIME PARAMETER ESTIMATION

281

1 Hz, and the recursive Fourier transforms were updated with each measurement, at 50 Hz. The marker at the right of each plot indicates the batch least-squares parameter estimate using time-domain data. After 9 s, the sequential parameter estimates change very little, and the confidence intervals are so small that the symbols representing the parameter estimates obscure them.

The parameter estimates shown were based only on the measured data—the algorithm was started with no *a priori* information about the parameters or their uncertainties. Final values of the sequential parameter estimates in the frequency domain matched the parameter estimates from batch least squares in the time domain, as expected.

Estimated standard errors shown in Fig. 8.4 accurately reflect the quality of each sequential estimate, including the effect of colored residuals. This comes about because the modeling is done in the frequency domain, where the fit error variance estimate is well defined and the residual coloring is properly incorporated in the parameter covariance matrix calculation. The parameter estimates have larger uncertainties at the beginning of the maneuver, due to low information. The uncertainties decrease as information becomes available to the estimator, and the estimated uncertainties do not change when there is no excitation (i.e., no additional information) at the end of the maneuver. This is an accurate characterization of the quality of the parameter estimates.

Note that the  $C_{n_p}$  parameter has some uncertainty until the aileron begins to move. This happens because the initial rudder movement excites the roll rate somewhat, but not as much as the aileron. Once the aileron moves, the roll rate is excited, and the estimator can accurately determine both  $C_{n_p}$  and  $C_{n_{\delta_a}}$ . The other parameters are already accurately determined by this time, because of excitation from the rudder.

Figure 8.5 for sequential least squares in the frequency domain shows behavior similar to that shown in Fig. 8.3 for recursive least squares in the time domain.

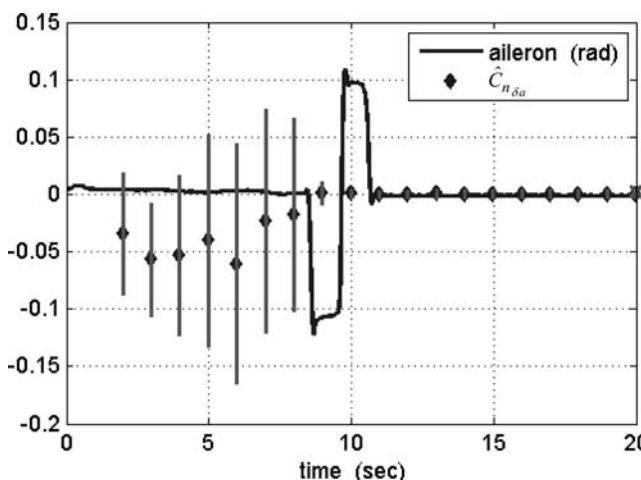


Fig. 8.5 Aileron control derivative estimate convergence using sequential least squares in the frequency domain.

Again, the aileron derivative estimate does not converge to an accurate and steady value until the aileron is moved, providing the necessary information.  $\square$

### 8.5 Extended Kalman Filter

In Chapter 4, the extended Kalman filter was introduced as a state estimation algorithm for nonlinear dynamic systems. The same algorithm can also be used to estimate both the states and parameters in a dynamic system model. To demonstrate this, the following model equations are considered:

$$\dot{\mathbf{x}} = \mathbf{A}(\boldsymbol{\theta})\mathbf{x} + \mathbf{B}(\boldsymbol{\theta})\mathbf{u} + \mathbf{w} \quad (8.53a)$$

$$\mathbf{z}(i) = \mathbf{C}(\boldsymbol{\theta})\mathbf{x}(i) + \mathbf{N}(i) \quad i = 1, 2, \dots, N \quad (8.53b)$$

$$E[\mathbf{x}(0)] = \bar{\mathbf{x}}_0 \quad E\{[\mathbf{x}(0) - \bar{\mathbf{x}}_0][\mathbf{x}(0) - \bar{\mathbf{x}}_0]^T\} = \mathbf{P}_{x_0} \quad (8.53c)$$

The noise sequences  $\mathbf{w}$  and  $\mathbf{v}$  are zero-mean white Gaussian noise sequences with covariance matrices  $\mathbf{Q}$  and  $\mathbf{R}$ , respectively, i.e.,

$$\mathbf{v} \text{ is } \mathcal{N}(\mathbf{0}, \mathbf{R}) \quad \text{Cov}[\mathbf{v}(i)] = E[\mathbf{v}(i)\mathbf{v}^T(j)] = \mathbf{R}\delta_{ij} \quad (8.53d)$$

$$\mathbf{w} \text{ is } \mathcal{N}(\mathbf{0}, \mathbf{Q}) \quad \text{Cov}[\mathbf{w}(t)] = E[\mathbf{w}(t_i)\mathbf{w}^T(t_j)] = \mathbf{Q}\delta(t_i - t_j) \quad (8.53e)$$

If the state vector is augmented with the parameters, the resulting augmented state vector is given by

$$\mathbf{x}_a = \begin{bmatrix} \mathbf{x} \\ \boldsymbol{\theta} \end{bmatrix} \quad (8.54)$$

The augmented state vector with both the dynamic system states and the unknown model parameters will be considered as the state vector for the extended Kalman filter. The unknown parameters  $\boldsymbol{\theta}$  are contained in the dynamic system matrices, and are assumed to be constant. Therefore, the dynamics of the estimated parameters are governed by

$$\dot{\boldsymbol{\theta}} = \mathbf{0} \quad (8.55)$$

The system equations for the augmented state are then

$$\dot{\mathbf{x}}_a = \mathbf{A}_a \mathbf{x}_a + \mathbf{B}_a \mathbf{u} + \mathbf{w}_a \quad (8.56a)$$

$$\mathbf{z}(i) = \mathbf{C}_a \mathbf{x}_a(i) + \mathbf{N}(i) \quad i = 1, 2, \dots, N \quad (8.56b)$$

$$E[\mathbf{x}_a(0)] = \bar{\mathbf{x}}_{a0} \quad E\{[\mathbf{x}_a(0) - \bar{\mathbf{x}}_{a0}][\mathbf{x}_a(0) - \bar{\mathbf{x}}_{a0}]^T\} = \mathbf{P}_{a0} \quad (8.56c)$$

where

$$\begin{aligned} \mathbf{A}_a &\equiv \begin{bmatrix} \mathbf{A}(\boldsymbol{\theta}) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} & \mathbf{B}_a &\equiv \begin{bmatrix} \mathbf{B}(\boldsymbol{\theta}) \\ \mathbf{0} \end{bmatrix} & \mathbf{w}_a &\equiv \begin{bmatrix} \mathbf{w} \\ \mathbf{0} \end{bmatrix} & \bar{\mathbf{x}}_{a0} &\equiv \begin{bmatrix} \bar{\mathbf{x}}_0 \\ \boldsymbol{\theta}_0 \end{bmatrix} \\ \mathbf{C}_a &\equiv [\mathbf{C}(\boldsymbol{\theta}) \quad \mathbf{0}] \end{aligned} \quad (8.56d)$$

## REAL-TIME PARAMETER ESTIMATION

283

The dynamic system for the augmented state is no longer linear, because the system matrices depend on elements of the augmented state. This nonlinearity is present regardless of the linearity of the original dynamic system. Thus, the system described by Eqs. (8.56) can be presented in a more concise form as

$$\dot{\hat{x}}_a = f(\hat{x}_a, \mathbf{u}) + \mathbf{w}_a \quad (8.57a)$$

$$z(i) = h[\hat{x}_a(i)] + \mathbf{v}(i) \quad i = 1, 2, \dots, N \quad (8.57b)$$

with the initial conditions given by Eq. (8.56c).

An extended Kalman filter algorithm for the system given in Eqs. (8.57) is obtained from the general equations (4.71) developed in Chapter 4 for a continuous nonlinear dynamic system with discrete measurements. The equations for state propagation and measurement update are as follows:

*Initial conditions:*

$$\hat{x}_a(0) = \bar{x}_{a0} \quad (8.58a)$$

$$\mathbf{P}_a(0) = \mathbf{P}_{a0} \quad (8.58a)$$

$$\mathbf{P}_{a0} \equiv \begin{bmatrix} \mathbf{P}_{x0} & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_{\theta_0} \end{bmatrix} \quad \bar{x}_{a0} \equiv \begin{bmatrix} \bar{x}_0 \\ \boldsymbol{\theta}_0 \end{bmatrix} \quad (8.58b)$$

$$E[\mathbf{x}(0)] = \bar{x}_0 \quad E\{[\mathbf{x}(0) - \bar{x}_0][\mathbf{x}(0) - \bar{x}_0]^T\} = \mathbf{P}_{x0} \quad (8.58c)$$

$$E[\boldsymbol{\theta}(0)] = \boldsymbol{\theta}_0 \quad E\{[\boldsymbol{\theta}(0) - \boldsymbol{\theta}_0][\boldsymbol{\theta}(0) - \boldsymbol{\theta}_0]^T\} = \mathbf{P}_{\theta_0} \quad (8.58d)$$

*Prediction:*

$$\frac{d}{dt}[\hat{x}_a(t|k-1)] = \mathbf{A}_a \hat{x}_a(t|k-1) + \mathbf{B}_a \mathbf{u}(t) \quad (8.58e)$$

$$\frac{d}{dt}[\mathbf{P}_a(t|k-1)] = \mathbf{A}_a \mathbf{P}_a(t|k-1) \mathbf{A}_a^T + \mathbf{Q}_a \quad (8.58f)$$

for  $(k-1)\Delta t \leq t \leq k\Delta t$ ,

$$\begin{aligned} \mathbf{A}_a &\equiv \begin{bmatrix} \mathbf{A}(\hat{\boldsymbol{\theta}}) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} = \left. \frac{\partial f}{\partial \mathbf{x}_a} \right|_{\mathbf{x}_a=\hat{\mathbf{x}}_a} \\ \mathbf{B}_a &\equiv \begin{bmatrix} \mathbf{B}(\hat{\boldsymbol{\theta}}) \\ \mathbf{0} \end{bmatrix} = \left. \frac{\partial f}{\partial \mathbf{u}} \right|_{\mathbf{x}_a=\hat{\mathbf{x}}_a} \quad \mathbf{Q}_a \equiv \begin{bmatrix} \mathbf{Q} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \end{aligned} \quad (8.58g)$$

*Measurement update:*

$$\mathbf{K}_a(k) = \mathbf{P}_a(k|k-1) \mathbf{C}_a^T [\mathbf{C}_a \mathbf{P}_a(k|k-1) \mathbf{C}_a^T + \mathbf{R}]^{-1} \quad (8.58h)$$

$$\hat{\mathbf{x}}_a(k|k) = \hat{\mathbf{x}}_a(k|k-1) + \mathbf{K}_a(k) [\mathbf{z}(k) - \mathbf{h}[\hat{\mathbf{x}}_a(k|k-1)]] \quad (8.58i)$$

$$\mathbf{P}_a(k|k) = [\mathbf{I} - \mathbf{K}_a(k) \mathbf{C}_a] \mathbf{P}_a(k|k-1) \quad (8.58j)$$

$$\mathbf{C}_a \equiv [\mathbf{C}(\hat{\boldsymbol{\theta}}) \quad \mathbf{0}] = \frac{\partial \mathbf{h}}{\partial \mathbf{x}_a} \Big|_{\mathbf{x}_a=\hat{\mathbf{x}}_a(k|k-1)} \quad \mathbf{K}_a \equiv \begin{bmatrix} \mathbf{K}_x \\ \mathbf{K}_{\boldsymbol{\theta}} \end{bmatrix} \quad (8.58k)$$

The augmented state covariance matrix has four partitions corresponding to the original state error covariance, the estimated parameter error covariance, and their cross variance,

$$\mathbf{P}_a \equiv \begin{bmatrix} \mathbf{P}_x & \mathbf{P}_{x\theta} \\ \mathbf{P}_{\theta x} & \mathbf{P}_{\theta} \end{bmatrix} \quad (8.59)$$

The expressions given here are the standard extension of the Kalman filter to a nonlinear dynamic system, which is called the extended Kalman filter, as discussed in Chapter 4. The only difference between the equations shown here and the Kalman filter equations for a linear dynamic system are in the definitions of the augmented matrices given by Eqs. (8.54), (8.58b), (8.58g), (8.58k), and (8.59).

The extended Kalman filter produces time histories of the parameter estimates, since the parameters are elements of the augmented state vector. As a result, the extended Kalman filter is a recursive algorithm that is applicable to real-time parameter estimation. Despite the dynamic equation (8.55), the parameter estimate time histories can exhibit time variations, due to the measurement updates. Equation (8.55) simply specifies that the estimated parameters cannot change during the propagation of the augmented state from one sample time to the next. Therefore, the extended Kalman filter can track parameter variations, along with the states of the dynamic system. Because of the inherent feedback in the algorithm [cf. Eq. (8.58i)], the extended Kalman filter can be applied to an unstable system for which the use of the output-error method might be difficult or impossible. The measurement updates continually correct the state estimates and prevent divergence. An arbitrary nonlinear dynamic model can be used, because the system matrices are computed by partial differentiation of the nonlinear functions  $\mathbf{f}(\mathbf{x}_a, \mathbf{u})$  and  $\mathbf{h}(\mathbf{x}_a)$ , [cf. Eqs. (8.58g) and (8.58k)].

The extended Kalman filter also has some disadvantages. As pointed out in Chapter 4, the Kalman gain matrix cannot be computed in advance, as for a linear filter. In addition, the parameter estimates are generally correlated with the state estimates. This correlation can decrease the accuracy of the parameter estimates. Values for  $\mathbf{Q}$  and  $\mathbf{R}$  must be chosen, and it is difficult to find a method to justify these choices. Consequently, the values are usually treated as tuning parameters, and are selected using simulation cases or flight-test data for

which batch parameter estimates are known. Estimates of the initial states and their variances can be obtained from initial measurements. However, the initial values  $\mathbf{x}_{a_0}$  and  $\mathbf{P}_{a_0}$  must also include initial estimates of the parameters and their variances. Parameter estimation results from the extended Kalman filter can be sensitive to the choices made for these quantities. Further problems can appear in convergence of the parameter estimates, due to Eq. (8.55).<sup>8</sup> This problem can be ameliorated by assuming that the parameter dynamics are driven by a fictitious random noise  $\mathbf{w}_f$ , as in Sec. 8.2.2, so that Eq. (8.55) would be replaced by

$$\dot{\boldsymbol{\theta}} = \mathbf{w}_f \quad (8.60)$$

Applications of the extended Kalman filter to aircraft state and parameter estimation is covered, e.g., in Refs. 8 and 9.

## 8.6 Summary and Concluding Remarks

This chapter presented methods that can be used in real time to estimate parameters for aircraft dynamic models, based on measured data. The methods were recursive least squares, sequential least squares, modified sequential least squares, extended Kalman filter, and sequential least squares in the frequency domain. The first technique is a recursive formulation of batch least-squares linear regression. The extended Kalman filter provides simultaneous estimates of state variables and model parameters. This technique is a nonlinear estimation problem, even for a linear dynamic system model. Sequential least squares uses batch methods on recent measured data. Regularization methods were introduced to address some of the problems described earlier. Sequential least squares in the frequency domain has some practical advantages, including robustness to measurement noise and biases, accurate parameter estimates and error bounds, low computational requirements, and efficient storage of data information content. For all the methods, there is the issue of how far back in time the data record should extend to form the basis for parameter estimation.

Two main problems must be addressed to achieve accurate real-time parameter estimation: 1) discrimination between signal and noise for tracking time-varying parameters, 2) data information content for parameter identifiability.

It is difficult to design a parameter estimation technique that is insensitive to noise but still responds rapidly to sudden changes in the system dynamics, mainly because it takes a fairly long data record to distinguish noise from a sudden change in the dynamics. This problem can be handled in the time domain by using recursive least squares and a forgetting factor to decrease the influence of older data, or by using sequential batch least squares with short data records and including various constraints in the cost function for parameter estimation. If an extended Kalman filtering approach is used, discriminating signal from noise is implemented through weighting matrices that represent assumed measurement and process noise covariance. For all of these time domain

methods, some adjustment of one or more tuning parameters must be done in simulation. Standard errors for the model parameter estimates, which are important both for failure detection and adaptive or reconfigurable control, cannot be accurately computed using recursive time-domain methods.

In the context of aircraft flight, lack of information content in the data can be problematic, because normal flight operations include extended periods where the control and state variables are fairly constant. During these times, the dynamic content of the signals are at or below the (relatively constant) noise level. In this circumstance, a time-domain regression method will give very inaccurate parameter estimates unless the estimation is regularized by including a term in the cost function that penalizes movement of the parameters away from *a priori* values (e.g., values from wind-tunnel tests), and/or a term that penalizes time variation of the parameter estimates. Tuning parameters must also be adjusted for this approach, because the magnitude of the penalty term(s) must be balanced properly relative to the least-squares part of the cost function associated with the measured data. Another approach is to implement a very long memory, but this has the disadvantage that new data are combined with old data, resulting in parameter estimates that are some weighted average. Adaptation to new data is consequently slower.

The recursive Fourier transform, discussed in connection with least-squares parameter estimation in the frequency domain, efficiently implements a long data memory. To make this method, and others, responsive to sudden changes in the aircraft dynamics, there must be some method for deciding how much of the past data information content should be forgotten. Actually implementing the memory loss in any of the methods is very straightforward—the difficult question is how much to forget and when.

Another problem that falls in the category of poor data information content is data collinearity due to the control system. Many control laws move more than one control surface at the same time in a nearly proportional way, or move control surfaces nearly in proportion to state variables. When states and controls are nearly proportional to one another, data collinearity exists, and it is difficult to identify individual stability and control derivatives from the measured data alone, as discussed in Chapter 5. When the proportionality is perfect, the task becomes impossible, and there must be some regularization or assumptions included to make the parameter estimation tractable. The data collinearity problem appears often when real-time parameter estimation is attempted on aircraft operating normally, as opposed to being flight tested specifically for parameter estimation. The next chapter discusses design methods for inputs that can be used to excite the aircraft response enough so that real-time parameter estimation methods will provide useful results. Some data information augmentation of this type is required for accurate parameter estimates at times when the aircraft is in a steady flight condition and/or has a feedback control system operating.

Real-time parameter estimation can improve efficiency and effectiveness of flight testing for stability and control or flight envelope expansion. In addition, many types of indirect reconfigurable control and failure detection require real-time parameter estimation. There are also uses for real-time parameter estimation in the areas of real-time safety monitoring and accident investigation.

## References

- <sup>1</sup>Goodwin, G. C., and Payne, R. L., *Dynamic System Identification: Experiment Design and Data Analysis*, Academic International Press, New York, 1977.
- <sup>2</sup>Norton, J. P., *An Introduction to Identification*, Academic International Press, London, 1986.
- <sup>3</sup>Young, P. C., *Recursive Estimation and Time-Series Analysis*, Springer-Verlag, New York, 1984.
- <sup>4</sup>Hsia, T. C., *System Identification*, Lexington Books, D.C. Heath and Company, Lexington, MA, 1977.
- <sup>5</sup>Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. R., *Numerical Recipes in FORTRAN: The Art of Scientific Computing*, 2nd ed., Cambridge Univ. Press, New York, 1992.
- <sup>6</sup>Ward, D. G., Monaco, J. F., and Bodson, M., "Development and Flight Testing of a Parameter Identification Algorithm for Reconfigurable Control," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 6, 1998, pp. 948–956.
- <sup>7</sup>Morelli, E. A., "Real-Time Parameter Estimation in the Frequency Domain," *Journal of Guidance, Control, and Dynamics*, Vol. 23, No. 5, 2000, pp. 812–818.
- <sup>8</sup>Jategaonkar, R., and Plaetschke, E., "Estimation of Aircraft Parameters Using Filter Error Methods and Extended Kalman Filter," DFVLR Forschungsbericht 88-15, 1988.
- <sup>9</sup>Garcia-Velo, J., and Walker, B., "Aerodynamic Parameter Estimation for High-Performance Aircraft Using Extended Kalman Filter," AIAA Paper 95-3500, 1995.

This page intentionally left blank

## 9

## Experiment Design

Experiment design involves determining which physical quantities will be measured, how those quantities will be measured, what the test conditions will be, and how the system being studied will be excited. For aircraft system identification, this translates into specifying the instrumentation and data acquisition system, selecting the aircraft configurations and flight conditions, and designing inputs for the maneuvers. These issues are the topics of this chapter.

The goal of experiment design is to maximize the information content in the data, subject to practical constraints. Some examples of practical constraints are

- 1) limits on input and/or output amplitudes, e.g., to ensure that a linear model structure can be used to estimate parameters from the measured data;
- 2) limited resolution or range for the sensors or data acquisition system;
- 3) hardware or telemetry limitations restricting the rate at which data can be measured or the number of physical quantities that can be measured at an acceptable rate;
- 4) limited time available for each maneuver and/or for the overall experimental investigation;
- 5) sensor limitations, characteristics, or availability;
- 6) limitations on how the aircraft can be excited, e.g., control surface rate or position limits, or the requirement for a continuously operating feedback control system when the aircraft is open-loop unstable.

As discussed in Chapter 3, inputs for modeling aircraft open-loop or bare-airframe dynamics are the control surface deflections. Outputs are air-relative velocity data ( $V, \alpha, \beta$ ), body-axis angular velocities ( $p, q, r$ ), Euler attitude angles ( $\phi, \theta, \psi$ ), translational accelerations ( $a_x, a_y, a_z$ ), and sometimes body-axis angular accelerations ( $\dot{p}, \dot{q}, \dot{r}$ ). Modeling is usually done using some subset of the inputs and outputs just listed, as in the case of linear models for longitudinal and lateral motion described in Chapter 3. For closed-loop modeling, where the modeling includes both the bare-airframe dynamics and the control system, the inputs are one or more of the pilot controls: longitudinal stick deflection, lateral stick deflection, and rudder pedal deflection, or the corresponding forces. Typically, throttle is not moved dynamically, but rather remains constant throughout a maneuver, so the throttle position is treated more like part of the

description of the flight-test condition. Outputs for closed-loop modeling are again subsets of the aforementioned quantities.

### 9.1 Data Acquisition System

The data acquisition system records time series of input and output variables, along with quantities that define the flight condition and aircraft configuration, such as outside air temperature and static pressure to calculate air density and pressure altitude, power level or throttle position, fuel consumption for estimating aircraft weight and inertia characteristics, landing gear position, flap settings, and other aircraft configuration variables.

Because of high computational requirements, modern system identification methods are implemented on digital computers. Therefore, the measurements of continuous-time signals associated with the aircraft must be converted to digital form. Analog signals from the sensors are passed through analog anti-aliasing filters, then possibly scaled to a proper voltage range for digitization using analog-to-digital (A/D) conversion electronics. The output of the A/D conversion is a digital count that is converted to engineering units using laboratory calibrations. Important aspects of the data acquisition system for system identification are discussed later. The discussion here does not include every aspect of aircraft data acquisition system design—only the most critical aspects for system identification. Ref. 1 provides more information on this topic.

**Sampling rate.** The sampling rate is the rate at which the physical quantities will be measured or sampled, usually expressed in samples/second or hz. Ideally, all measured signals should be sampled at the same constant rate, but it is not uncommon for measured quantities to be sampled at different rates, for practical reasons. When several sampling rates are used for the measured data, interpolation can be used to convert the lower sample rate data to the highest sampling rate. Interpolation is discussed in Chapter 11. Another approach is to thin the high-rate data to match the sample rate of the low-rate data. This is discussed later. In any case, all data should be converted to the same sampling rate before analysis.

The choice of sampling rate is influenced by Shannon's sampling theorem<sup>2</sup> but also by practical considerations. In theory, the discrete samples of a continuous signal sampled at a frequency  $f_s$  capture frequency content up to and including the frequency  $f_s/2$ , which is known as the Nyquist frequency  $f_N$ ,

$$f_N = f_s/2 \quad (9.1)$$

Figure 9.1 shows an illustration of this fact, using a continuous sine wave signal. The frequency of the continuous signal is 1 Hz, and the sampling frequency is 2 Hz. Note that if the continuous signal was shifted in time by the equivalent of 90 deg of phase angle, then the samples taken at the times shown would be all zeros. For this reason, sampling at the theoretical minimum rate of twice the Nyquist frequency is not acceptable for capturing frequency content up to and including the Nyquist frequency. In addition, it is clear from

## EXPERIMENT DESIGN

291

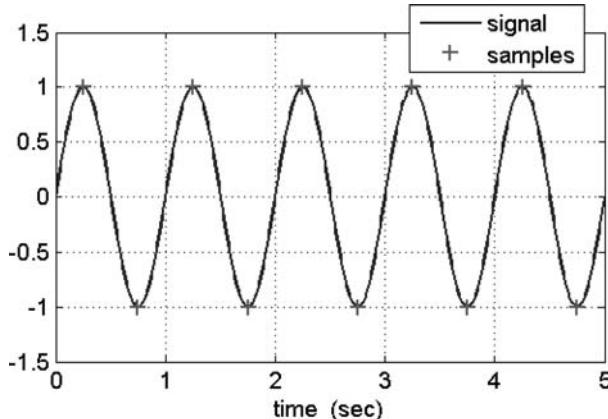


Fig. 9.1 Theoretical minimum sampling rate.

Fig. 9.1 that the theoretical minimum sampling rate gives a very rough representation of the continuous signal. To obtain good results in practice, it is necessary to sample at a rate much higher than the theoretical minimum rate. If  $f_{\max}$  is the maximum frequency of interest in the continuous-time signals from the dynamic system, a good rule of thumb for selecting the sampling rate  $f_s$  is

$$f_s = 25f_{\max} \quad (9.2)$$

For many aircraft, the frequencies of the rigid-body dynamic modes are below 2 Hz, which would put the sampling rate at 50 Hz.

If the aircraft is a scale model, the frequencies of the dynamic modes scale according to the inverse square root of the model geometric scale,

$$f_{\text{model}} = \frac{1}{\sqrt{s}} f_{\text{aircraft}} \quad (9.3)$$

where  $s$  is the model scale.<sup>3</sup> Therefore, for a 1/16 scale model of a full-scale aircraft with rigid-body dynamics at less than 1 Hz, the natural frequencies of the scale model would lie below 4 Hz, requiring a sampling rate of 100 Hz.

**Presampling data conditioning.** When the continuous signal being sampled has components at frequencies higher than the Nyquist frequency, a phenomenon called aliasing occurs. Aliasing is illustrated in Fig. 9.2 for a simple sinusoidal continuous signal. The result of aliasing is that frequency content above the Nyquist frequency  $f_N$  is falsely attributed (aliased) to lower frequencies by the sampling process. The frequencies above  $f_N$  that are aliased to the frequency  $f$  in the range  $0 \leq f \leq f_N$  are

$$2nf_N \pm f \quad n = 1, 2, \dots \quad (9.4)$$

The manner in which the high frequencies fold downward relative to the Nyquist frequency is illustrated by the diagram in Fig. 9.3. Aliasing pollutes the true

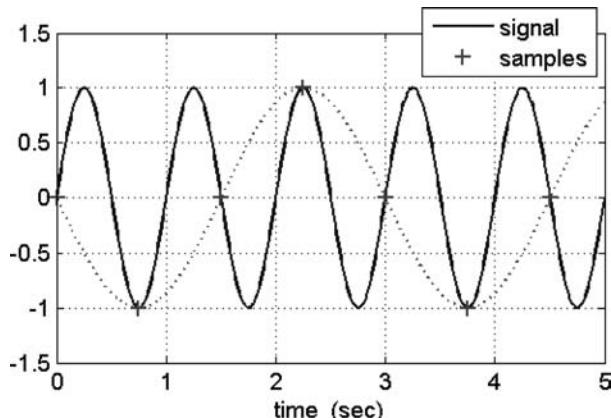


Fig. 9.2 Aliasing.

frequency content at lower frequencies with frequency content folded down from frequencies above  $f_N$ .

Aliasing is a serious problem that must be avoided. The solution is to use analog low-pass filters to remove high-frequency components before sampling. Low-pass filters used in this way are called antialiasing filters or pre-sampling filters. The filters must be analog to work on the signals before sampling. Aliasing cannot be repaired with postflight data analysis or digital filtering, because once the aliasing occurs in the sampling process, the frequencies

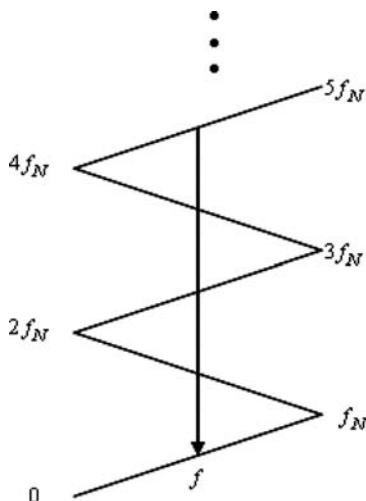
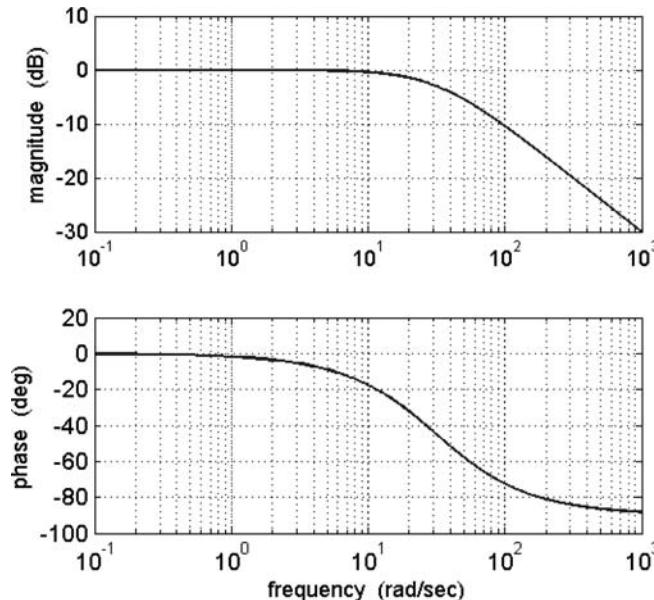


Fig. 9.3 Frequency folding.

## EXPERIMENT DESIGN

293



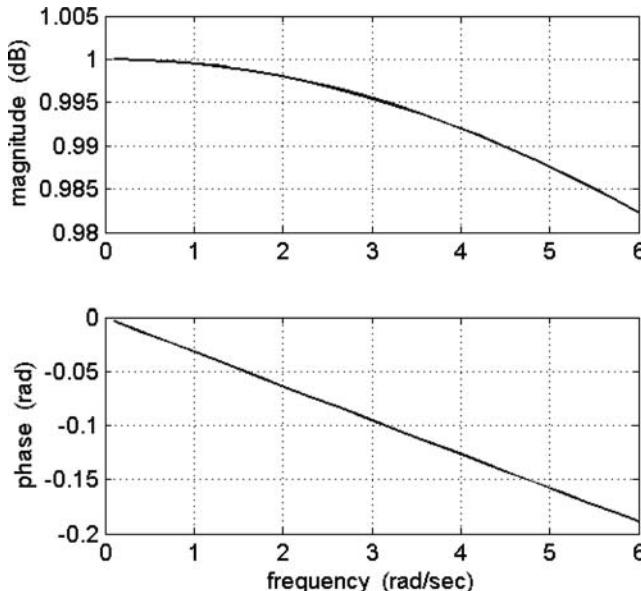
**Fig. 9.4** Frequency response for a typical analog antialiasing filter:  $\tilde{y}(s)/\tilde{u}(s) = 31.4/(s + 31.4)$ .

above the Nyquist frequency have been commingled with the true low-frequency content, and there is no way to reliably separate the components.

Antialias filtering can be done with a simple first-order low-pass filter. Figure 9.4 shows a Bode plot of the frequency response for a typical antialiasing filter with break frequency at 5 Hz (31.4 rad/s). Low-frequency signal content, which includes the frequency components of interest, is passed through essentially unmodified, and high-frequency content that could be aliased is removed before the sampling occurs. Figure 9.5 shows the amplitude and phase modification for the passband of the same antialiasing filter, plotted using a linear frequency scale. The magnitudes are modified only slightly, and the phase change is nearly linear with frequency. This corresponds to a constant time delay  $\tau$  at all frequencies, since for phase angle  $\phi$

$$\begin{aligned}\phi &= -\omega\tau \approx \frac{d\phi}{d\omega}\omega \\ \tau &\approx -\frac{d\phi}{d\omega}\end{aligned}\tag{9.5}$$

All sampled signals should have the same antialiasing filtering, so that the small time delay from antialias filtering before sampling is the same for all measured signals at each frequency.



**Fig. 9.5** Magnitude and phase modification for the passband of the analog antialiasing filter:  $\tilde{y}(s)/\tilde{u}(s) = 31.4/(s + 31.4)$ .

The break frequency of the filter must be high enough so that the frequency content of interest is not modified by the antialias filtering, but low enough so that high-frequency attenuation is sufficient to remove unwanted high-frequency components. A good rule of thumb is to set the break frequency for the anti-aliasing filters  $f_a$  at five times the highest expected frequency of interest,

$$f_a = 5f_{\max} \quad (9.6)$$

which makes the relationships among the important frequencies as follows:

$$f_s = 5f_a = 25f_{\max} \quad (9.7)$$

If it is unclear what the highest frequency of interest  $f_{\max}$  might be, it is best to err on the side of a higher sampling rate. A higher sampling rate moves the Nyquist frequency higher and therefore also raises the level for frequencies that can be aliased. Higher sampling rates also have the advantage of providing more samples than are really needed and therefore represent protection against problems such as power spikes and data dropouts. In addition, higher sampling rates reduce the possible time skews among the measured signals.

Thinning high sampling rate data to a lower sampling rate, which is also called decimating the data, is equivalent to sampling the data at a lower sampling rate. Decimation therefore lowers the Nyquist frequency, and the same potential for

## EXPERIMENT DESIGN

295

aliasing exists. However, if the original high-rate data collection was done properly, it is only necessary to remove frequency content above the reduced Nyquist frequency associated with the decimation to avoid aliasing.

For example, decimating 50 Hz data to 25 Hz by dropping every other data point reduces the effective Nyquist frequency from 25 Hz to 12.5 Hz. To avoid aliasing from the decimation, there should be no significant components above 12.5 Hz in the original 50 Hz sampled data. Zero phase-shift digital filtering or smoothing, discussed in Chapter 11, can be used before decimation to remove frequency content above 12.5 Hz in the 50 Hz sampled data. The digital filtering or smoothing is done before decimation, for the same reasons cited earlier for the analog antialias filtering.

Implicit in all of the modeling techniques discussed elsewhere in this book is the assumption that all measured physical quantities have been sampled simultaneously. That is, it is assumed that there are negligible differences between the instants when the first and last physical quantities are measured for each data sample taken. In practice, there is a small time difference, but the maximum time difference should be negligible (usually less than a millisecond) for modern data acquisition systems intended for collecting flight-test data. However, in some cases, such as low-budget flight-test programs and accident analysis, the physical quantities may not be sampled at nearly the same time. In that situation, it is necessary to interpolate or time shift the data so that the data are brought to the state of nearly simultaneous sampling of the physical quantities. This is important, because if it is not done, then the time skews that remain can be interpreted as phase lags due to the dynamic response of the physical system, which they are not.

In Ref. 4, the effect of time skews in the recorded data on parameter estimates from output error was studied using flight-test data from five different aircraft. The results showed that parameters related to high-frequency quantities, such as body-axis angular rates, were more sensitive to time skews in the measured data than parameters associated with lower-frequency quantities, such as angle of attack and sideslip angle. This makes sense, because a given magnitude of time skew represents a larger phase angle change for a high-frequency signal, compared with a lower-frequency signal.

For the full-scale aircraft flight data studied in Ref. 4, time skews greater than about 40 ms caused significant inaccuracies in parameter estimation results. Note that if the sampling rate of the data acquisition system is 50 Hz and the system is operating properly, the relative time skews of the measured data can be no greater than the sampling interval, which is 20 ms. This built-in protection against large time skews in the data is another reason to prefer higher sampling rates.

*Sensor range and resolution.* The range of each sensor should include values larger than the largest expected value of the physical quantity to be measured, to allow for unusual motions and to avoid sensor saturation. Physical values greater than the maximum of the sensor range would produce the same maximum sensor reading, for example, resulting in a loss of information.

For typical modern data acquisition systems, the A/D conversion might use a 14-bit binary word. In this case, the sensor resolution can be computed from

$$\text{resolution} = \text{range}/2^{14} = \frac{(\text{max sensor reading} - \text{min sensor reading})}{16,384} \quad (9.8)$$

For 14-bit A/D conversion, the resolution is 0.006 percent of the full range of the sensor. The resolution represents a lower limit on the accuracy of the measurements.

The preceding equation shows that there is a tradeoff between range and resolution, for given A/D conversion hardware. Of course, more bits in the A/D conversion improves the resolution for a given sensor range, and therefore ameliorates the problem. The range and resolution must be selected carefully for each measured physical quantity. The intent of many flight-test programs is to collect data for linear model identification over a relatively large portion of the flight envelope. This requires good resolution to accurately quantify the small changes in physical quantities associated with small perturbation maneuvers, and good range so that the sensors can be used over a large flight envelope. Table 9.1 is an example from Ref. 5 of instrumentation characteristics that have worked well for a general aviation airplane.

The sensors themselves are also dynamic systems, so their characteristics should be such that the dynamics of the sensor do not interact with the dynamics of the aircraft. This is achieved by building sensors with relatively high natural

**Table 9.1 Instrumentation characteristics for a general aviation airplane**

Measured quantity	Transducer	Working range	Resolution	rms measurement error
$a_x, g$	Accelerometer	[−1,1]	0.001	0.0046
$a_y, g$	Accelerometer	[−1,1]	0.001	0.0050
$a_z, g$	Accelerometer	[−3,6]	0.001	0.0050
$p, \text{deg/s}$	Rate gyro	[−102,102]	0.12	0.20
$q, \text{deg/s}$	Rate gyro	[−29,29]	0.032	0.19
$r, \text{deg/s}$	Rate gyro	[−29,29]	0.034	0.080
$\phi, \text{deg}$	Vertical gyro	[−90,90]	0.10	0.077
$\theta, \text{deg}$	Vertical gyro	[−87,87]	0.098	0.092
$\alpha, \text{deg}$	Flow vane	[−12,27]	0.029	0.027
$\beta, \text{deg}$	Flow vane	[−29,32]	0.018	0.019
$\delta_{ar}, \text{deg}$	Potentiometer	[−23,10]	0.002	0.019
$\delta_{al}, \text{deg}$	Potentiometer	[−10,25]	0.002	0.0061
$\delta_s, \text{deg}$	Potentiometer	[−16,3]	0.010	0.0037
$\delta_r, \text{deg}$	Potentiometer	[−31,28]	0.011	0.0091
$V, \text{m/s}$	Pressure transducer	[0,75]	0.037	0.89
$h, \text{m}$	Altimeter	[−150,2900]	—	—
$T, ^\circ\text{C}$	Thermometer	[−18,38]	—	—

## EXPERIMENT DESIGN

297

Table 9.2 Dynamic characteristics of transducers

Measured quantity	Transducer	Natural frequency, Hz	Damping ratio	Time delay, s
$a_x, g$	Accelerometer	402	1.58	0.0012
$a_y, g$	Accelerometer	216	1.10	0.0016
$a_z, g$	Accelerometer	921	1.58	0.0005
$p, \text{deg/s}$	Rate gyro	27	0.64	0.0075
$q, \text{deg/s}$	Rate gyro	27	0.64	0.0075
$r, \text{deg/s}$	Rate gyro	27	0.64	0.0075
$\alpha, \text{deg}$	Flow vane	23 <sup>a</sup>	0.085 <sup>a</sup>	0.0012
$\beta, \text{deg}$	Flow vane	23 <sup>a</sup>	0.085 <sup>a</sup>	0.0012

<sup>a</sup>At  $V = 164 \text{ ft/s}$ 

frequencies, high damping ratios, and small time delays. Table 9.2 gives the dynamic characteristics of the general aviation aircraft instrumentation in Table 9.1.

## 9.2 Instrumentation

Flight-test instrumentation is constantly evolving, so any description of current hardware would quickly become outdated. However, important characteristics of the flight instrumentation can be discussed in general, without reference to any particular sensors, and that is the approach taken here. The discussion is organized according to the physical quantities to be measured.

**Air-Relative Velocity.** Accurate air-relative velocity data, which is composed of measurements of angle of attack, sideslip angle, and airspeed, are perhaps the most difficult to obtain. Part of the reason is that local flows about the aircraft influence measured values of these quantities, while the desired quantities for rigid-body dynamic modeling are the values for the aircraft at the c.g. Therefore, no matter where the sensors for these quantities are placed, there will be some corrections involved. Calibration is also more difficult, because air must be flowing over the sensor at the time of the calibrations, and the calibration is really only valid when the sensor is installed on the aircraft. Complicating matters further is the fact that these data are arguably the most important physical quantities in terms of modeling the aerodynamic forces and moments acting on the aircraft. Consequently, the quest for accurate air-relative velocity data is continual.

Sensors for air-relative velocity data can be calibrated very accurately in the wind tunnel, preferably installed on the aircraft or on parts of the aircraft that are nearby (i.e., the wing or nose cone). The best location for the sensors is on a nose boom, but this location is impractical for aircraft with a propeller mounted on the nose. Flow angle sensors are often located on wing tip booms,

which should extend 2–3 chord lengths forward of the leading edge of the wing. Local flow at the wingtips in the outboard direction usually biases the sideslip angle measurement, so it is more accurate to use averaged sideslip measurements from sensors on both wing tips. It is possible to use sensors near the wing or fuselage, including sensors based on pressure, but this generally requires careful and extensive calibration for good accuracy.

The position of the flow angle sensors relative to the aircraft c.g. must be known accurately. This information is necessary to implement corrections for angle of attack and sideslip angle readings resulting from angular motion of the aircraft. Usually, the c.g. and the sensor positions are specified relative to a single fixed reference point, and the sensor position relative to the c.g. is computed from that information. Details of these corrections appear in Chapter 10.

*Angular velocity.* Aircraft angular velocity components are usually measured using rate gyros attached to the aircraft and aligned with the body axes. These sensors are among the most reliable and accurate of all aircraft flight instrumentation. Response is very linear, and typical instrumentation errors consist of small biases that are very repeatable among different maneuvers. In theory, the location of these sensors can be anywhere on the aircraft, without the need for any position correction to the c.g., because angular rates are the same at any point on a rigid body. In practice, however, the aircraft is not rigid, and it is necessary to locate the sensors away from the nodes of any significant structural response modes, which would cause rotational motion that the sensors will pick up. Often, the rate gyros are packaged with the translational accelerometers.

*Translational acceleration.* Translational accelerometers should be located close to the aircraft c.g., and aligned with the body axes. This keeps the corrections small when the measurements are corrected to the aircraft c.g. The corrections come from accelerations due to rotational aircraft motion about the c.g., in conjunction with the position offset of the accelerometers from the c.g. Translational accelerometers have excellent linearity and usually only a small bias error. The main negative aspect of these sensors is that their frequency response is excellent, so in addition to rigid-body motion, they also pick up structural response and engine vibrations. This can make the signals quite noisy, and also can cause problems with high-frequency responses folding down to the range of rigid-body frequencies, if the antialiasing filters are not designed and implemented properly. The position corrections for these sensors involve body-axis angular accelerations (see Chapter 10), so some additional noise is introduced as a result of the position corrections, since the angular acceleration signals are typically noisy, regardless of whether they are obtained from smoothed numerical derivatives of the angular rates or directly from a sensor. The relatively high noise levels for accelerometer signals also make it more difficult to filter or smooth the data, because large random noise components can mask the deterministic signal content.

## EXPERIMENT DESIGN

299

*Rotational acceleration.* Sensors for rotational acceleration are not common, but occasionally are included in flight-test instrumentation. Most of the sensors available at the time of this writing have relatively high noise levels and/or lags. Because of this, measured angular accelerations are not used often for aircraft system identification. As sensor technologies evolve, this situation may change. Measurements from these sensors can be easily incorporated into the modeling methods. Good angular acceleration measurements would improve parameter estimation results by providing additional information content to the data for output-error modeling, and obviating the need to numerically differentiate the angular rates for instrumentation error corrections and equation-error modeling.

*Euler attitude angles.* Euler angles are usually measured using integrating gyros or magnetometers. The Euler angles have secondary importance in system identification for aircraft, because the aerodynamic forces and moments do not depend on aircraft orientation relative to earth axes. However, as discussed in Chapter 3, the Euler angles are needed to include the gravity terms in the equations of motion. The heading angle appears only in the kinematic equations and not in any of the dynamic equations, so it is the least important of the Euler angles for dynamic modeling (but of course the most important for guidance). All of the Euler angles are useful in data compatibility analysis, for estimating instrumentation errors on rate gyro measurements, as discussed in Chapter 10.

*Control surface deflections and pilot controls.* Sensors that measure control surface deflections and pilot control deflections are typically some type of potentiometer, which produce a voltage proportional to rotational or linear motion. These sensors are very reliable and linear, and have very low noise levels. The latter characteristic is important, because the modeling methods assume that known deterministic inputs can be measured without error.

### 9.3 Input Design

In designing inputs for dynamic systems in general, and aircraft in particular, there are two general approaches. The first is to design the input assuming no *a priori* knowledge of how the dynamic system behaves. The goal in that case is to excite the system over a broad frequency range, with nearly constant power for all frequencies. Input designs that employ this approach include frequency sweeps and impulse inputs. The second approach is to use *a priori* knowledge about the dynamic system response and tailor the input accordingly. Optimal input designs are in this category, along with square-wave inputs that are designed to excite the dynamic system at or near the *a priori* estimates of the natural frequencies for the dynamic modes.

### 9.3.1 Maneuver Definition

In Chapter 3, a common aerodynamic model structure was shown to be a locally valid approximation of the global functional dependencies, typically a linear multivariate Taylor series in the states and controls. The point about which this expansion is made is typically a trimmed flight condition, such as steady, level flight. The coefficients of the Taylor series expansion are the aerodynamic model parameters to be estimated. These parameters are therefore a function of flight condition for the test.

Consequently, one aspect of maneuver definition is selecting the flight conditions where maneuvers will be executed. For the nondimensional aerodynamic force and moment coefficients, the relevant aspects of the flight condition are typically trim angle of attack, Mach number, aircraft configuration, altitude, and power level. Flight conditions for testing are chosen based on the goals of the particular investigation, weighed against resource limitations and other practical constraints.

The other aspect of the maneuver definition is specifying the excitation. To define the excitation for a maneuver, the following must be specified:

- 1) Maneuver time length,
- 2) Control surfaces or pilot inputs to be moved,
- 3) Input forms (e.g., square wave, frequency sweep, etc.), which includes the input amplitudes.

Input design involves selecting the preceding quantities in a way that provides adequate information content in the data for accurate modeling. The exact meaning of this will be explored in detail.

### 9.3.2 Input Design Objective

The objective of input design for dynamic model identification is to excite the dynamic system so that the data contain sufficient information for accurate modeling, subject to the practical constraints of the experiment. This section describes data information content and practical constraints mathematically, and provides methods for achieving the desired results in practice.

*Data information content.* For a single measured quantity, data information content can be quantified by the signal-to-noise ratio. Specifically, the part of the measured quantity that is called the signal is assumed to be deterministic, whereas the noise is random, with incoherent phase and amplitude variations that can be readily recognized visually in both the time and frequency domains. It is possible to separate the deterministic signal and random noise components of a measured signal using an optimal Fourier smoothing method (see Chapter 11), so that signal-to-noise ratio can be computed using root-mean-square values. Signal-to-noise ratio can also be estimated visually from plots of the data. For measured aircraft responses, the signal-to-noise ratio should be 10 or more for good modeling results. Usable results can be obtained with signal-to-noise ratios as low as 3.

## EXPERIMENT DESIGN

301

In the frequency domain, the coherence is a measure of signal-to-noise ratio for the frequency response, assuming a linear relationship between the input and output (see Chapter 7). Coherence values greater than approximately 0.8 are usually required for good linear modeling results.

Both signal-to-noise ratio and coherence could be classified as nonparametric measures of data information content, because they have no connection to any mathematical model with parameters. For most aircraft system identification, the end result is an identified parametric model, so mathematical descriptions of data information content for parametric models are important.

For the simple case of a single-input, single-output model with one model parameter, data information content is quantified by the sensitivity of the model output to changes in the parameter, called the output sensitivity, which was discussed in Chapter 6. The best input for a parameter estimation experiment is the input that maximizes the squared output sensitivity over the test time  $T$ . This can be expressed as

$$u^* = \max_{u \in U} \sum_{i=1}^N \left[ \frac{\partial y(i)}{\partial \theta} \right]^2 = \min_{u \in U} \sum_{i=1}^N \left\{ \left[ \frac{\partial y(i)}{\partial \theta} \right]^2 \right\}^{-1} \quad (9.9)$$

where  $u^*$  is the scalar optimal input waveform over the test time  $[0, T]$ ,  $U$  is the set of admissible inputs, and the summation over  $N$  time points approximates a time integral with  $T = N \Delta t$  and sampling interval  $\Delta t$ .

The optimization in Eq. (9.9) is equivalent to maximizing the signal-to-noise ratio for a parameterized model. The noise does not appear in this scalar case, because the input has no influence on it and there is only a single output, so it is not necessary to account for different noise levels on different outputs. High output sensitivity means that small changes in the model parameter will cause large changes in the model output. Consequently, small changes in the model parameter have a large effect on how closely the model output matches the measured output. In this circumstance, parameter estimation routines will be able to accurately locate the parameter value that results in the best match of the model output to the measured output. At the other extreme, if the output sensitivity is low, the parameter estimation routines can adjust the parameter considerably without changing the model output very much, so that a range of parameters may be indistinguishable in terms of the model output match to the measured output. The result is an inaccurate parameter estimate with large uncertainty.

For aircraft dynamic models, there are often multiple outputs and multiple model parameters. The information content in the data is then quantified by a matrix, called the information matrix, which was discussed in Chapters 5 and 6. For a parameter vector  $\boldsymbol{\theta}$  of length  $n_p$ , the  $n_p \times n_p$  information matrix  $\mathbf{M}$  is

$$\mathbf{M} = \sum_{i=1}^N \left[ \frac{\partial \mathbf{y}(i)}{\partial \boldsymbol{\theta}} \right]^T \mathbf{R}^{-1} \left[ \frac{\partial \mathbf{y}(i)}{\partial \boldsymbol{\theta}} \right] \quad (9.10)$$

which is Eq. (6.35).

When the  $\mathbf{R}$  matrix is diagonal, as in the usual case, the diagonal elements of  $\mathbf{R}^{-1}$  introduce a scaling of the output sensitivities according to the inverse of the individual output measurement noise variances. If the sensitivities of the outputs to the parameters are interpreted as the “signal” for parametric modeling purposes, then the information matrix is a discrete-time sum of multiple-output, multiple-parameter signal-to-noise ratios. It follows that the information matrix can be loosely interpreted as signal-to-noise ratio for multiple-output parameterized models. If the sensitivities are large relative to the noise levels and are uncorrelated with one another, then the output dependence on the parameters is strong and distinct for each parameter. Parameter values can then be estimated with high accuracy when adjusting the parameters so that model outputs match measured outputs.

As discussed in Chapter 6, the partial derivatives in the  $n_o \times n_p$  sensitivity matrix  $\partial\mathbf{y}(i)/\partial\boldsymbol{\theta}$  are the output sensitivities, which can be found using finite differences or by solving the sensitivity equations.

For a linear dynamic system of the form

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A}(\boldsymbol{\theta})\mathbf{x}(t) + \mathbf{B}(\boldsymbol{\theta})\mathbf{u}(t) & \mathbf{x}(0) &= \mathbf{x}_0 \\ \mathbf{y}(t) &= \mathbf{C}(\boldsymbol{\theta})\mathbf{x}(t) + \mathbf{D}(\boldsymbol{\theta})\mathbf{u}(t)\end{aligned}\quad (9.11)$$

with discrete measurements

$$\begin{aligned}\mathbf{z}(i) &= \mathbf{y}(i) + \boldsymbol{\nu}(i) & i &= 1, 2, \dots, N \\ E[\boldsymbol{\nu}(i)\boldsymbol{\nu}^T(j)] &= \mathbf{R}\delta_{ij}\end{aligned}\quad (9.12)$$

the sensitivity equations for each parameter  $\theta_j$ , developed in Chapter 6, are

$$\begin{aligned}\frac{d}{dt}\left(\frac{\partial\mathbf{x}}{\partial\theta_j}\right) &= \mathbf{A}\frac{\partial\mathbf{x}}{\partial\theta_j} + \frac{\partial\mathbf{A}}{\partial\theta_j}\mathbf{x} + \frac{\partial\mathbf{B}}{\partial\theta_j}\mathbf{u}, & \frac{\partial\mathbf{x}(0)}{\partial\theta_j} &= 0 \\ \frac{\partial\mathbf{y}}{\partial\theta_j} &= \mathbf{C}\frac{\partial\mathbf{x}}{\partial\theta_j} + \frac{\partial\mathbf{C}}{\partial\theta_j}\mathbf{x} + \frac{\partial\mathbf{D}}{\partial\theta_j}\mathbf{u} & j &= 1, 2, \dots, n_p\end{aligned}\quad (9.13)$$

Note that it is necessary to have *a priori* values for the model parameters to solve the dynamic and sensitivity equations (9.11) and (9.13), respectively. The information matrix depends on the input  $\mathbf{u}$ , which influences the sensitivities both directly as a forcing function in the sensitivity equations and indirectly as an influence on the states, which also force the sensitivity equations.

In Appendix A, it is shown that the inverse of the information matrix is the theoretical lower limit for estimated parameter covariances, computed using an asymptotically unbiased and efficient parameter estimation algorithm. This theoretical lower limit is called the Cramér-Rao lower bound or the dispersion matrix, and is given by

$$\boldsymbol{\Sigma} = \mathbf{M}^{-1} = \left\{ \sum_{i=1}^N \left[ \frac{\partial\mathbf{y}(i)}{\partial\boldsymbol{\theta}} \right]^T \mathbf{R}^{-1} \left[ \frac{\partial\mathbf{y}(i)}{\partial\boldsymbol{\theta}} \right] \right\}^{-1} \leq \text{Cov}(\boldsymbol{\theta}) \quad (9.14)$$

## EXPERIMENT DESIGN

303

Since the information matrix  $M$  depends on the input sequence  $u$ , the theoretical lower bounds on the parameter covariances also depend on the input sequence. The dependence of the Cramér-Rao bounds on the input is nonlinear, regardless of whether or not the system equations (9.11) are linear, because of the nonlinear character of Eq. (9.14) and the time integrations required to solve Eqs. (9.11) and (9.13). Similarly, the Cramér-Rao bounds depend nonlinearly on the states.

As discussed in Chapter 6, the diagonal elements of the dispersion matrix are the theoretical minimum values of the individual parameter variances. The Cramér-Rao lower bounds for the parameter standard errors are the square root of the diagonal elements of the dispersion matrix,

$$s(\hat{\theta}_j) = \sqrt{s_{jj}} \quad j = 1, 2, \dots, n_p \quad (9.15)$$

where  $s_{jk}$  are the matrix elements of the dispersion matrix  $\Sigma$ ,

$$\Sigma = [s_{jk}] \quad j, k = 1, 2, \dots, n_p \quad (9.16)$$

The  $j$ th diagonal element of the dispersion matrix corresponds to the  $j$ th parameter in the sensitivity equations (9.13).

The Cramér-Rao bounds are independent of the parameter estimation algorithm used to extract parameter estimates and standard errors from the data, since the Cramér-Rao bounds represent a theoretical lower bound. Thus, the merit of an input design for aircraft parameter estimation can be determined by examining the Cramér-Rao bounds, since these depend only on the information matrix and not on the parameter estimation algorithm. In other words, input designs are evaluated based only on the information content in the data, which is calculable before any parameter estimation is done.

Computation of the Cramér-Rao bounds requires an *a priori* dynamic system and measurement model. Consequently, a model complete with parameter values is necessary to design an experiment that will produce data for estimating the model parameters. This has been called the circularity problem.<sup>6</sup> For aircraft, the problem is mitigated by use of parameter estimates obtained from aerodynamic calculations, wind-tunnel experiments, or previous flight tests. In practice, most input design methods require very little *a priori* information, and some require none at all, as discussed later. Only optimal input design techniques require calculation of the information matrix using an *a priori* model. However, the information matrix and the associated calculations are useful in evaluating any input design for parameter estimation, regardless of how that input was designed.

Knowledge of how the information matrix is defined is also helpful for insight into practical input design. For example, the calculation of the information matrix and Cramér-Rao bounds implicitly include the time length of the flight-test maneuver. Since the discrete approximation of a time integral is part of the information matrix calculation, it is clear that longer maneuvers will provide more information in the data and lower Cramér-Rao bounds, assuming that the input continuously excites the system.

*Practical constraints.* There are many practical constraints that restrict the manner in which flight testing for aircraft parameter estimation can be conducted, which in turn limits the information content in the data from flight-test maneuvers.

Most models identified from flight-test data would be called local models, because their range of validity is limited to a relatively small range of important explanatory variables, such as angle of attack and sideslip angle. During the experiment, aircraft response must be limited to perturbations about the initial flight condition, to retain the validity of the local model structure. Under these conditions, the model parameters can be considered constant throughout the maneuver. If the model to be identified is a linear dynamic model, then the output perturbations must also be limited to satisfy the small perturbation requirement necessary for decoupling longitudinal and lateral dynamic equations.

Constraints for model structure validity are best implemented in terms of amplitude constraints on selected aircraft response variables. Typical values for the perturbation amplitude constraints are  $\pm 5$  deg in angle of attack or sideslip angle,  $\pm 20$  deg/s in body-axis angular rates, and  $\pm 0.1$  g to  $\pm 0.3$  g in translational accelerations. In addition, constraints may be required on aircraft attitude angles for flight-test operational considerations, such as flight safety and maintaining the data downlink to the ground. These constraints can be represented mathematically as

$$|y_k(t)| \leq \xi_k \quad \forall t \quad k \in (1, 2, \dots, n_o) \quad (9.17)$$

where  $\xi_k$  is the constant amplitude constraint for the  $k$ th output, and  $n_o$  is the number of outputs.

Input amplitudes are limited by mechanical stops, flight control software limiters, or model structure validity. These constraints are specified by

$$|u_j(t)| \leq \mu_j \quad \forall t \quad j \in (1, 2, \dots, n_i) \quad (9.18)$$

where  $\mu_j$  is the constant amplitude constraint for the  $j$ th input, and  $n_i$  is the number of inputs.

Sometimes the preceding constraints are implemented indirectly using a constraint on the input energy,

$$\int_0^T \mathbf{u}(t)^T \mathbf{u}(t) dt \leq E \quad (9.19)$$

where  $E$  is a selected input energy. The input energy constraint has been introduced mostly to simplify formulations for input optimization using variational calculus. In practice, it is difficult to choose an appropriate value for the input energy  $E$ , since neither the pilot nor control system has any such energy limitations. Use of the input energy constraint as an indirect limit on output amplitudes is imprecise, because the output amplitudes depend on the frequency content of the input, which is not captured by the input energy calculated from Eq. (9.19). Figure 9.6 shows three square wave inputs with very different time lengths and amplitudes, but the same input energy calculated from Eq. (9.19). Further discussion of this issue can be found in Ref. 7.

## EXPERIMENT DESIGN

305

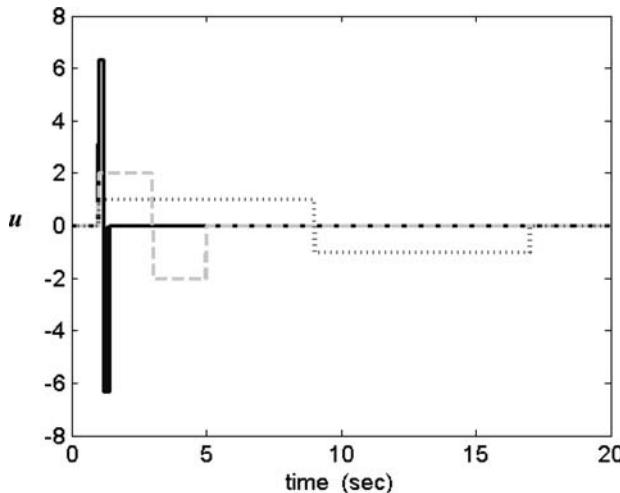


Fig. 9.6 Constant energy square waves with very different forms.

The expression for the information matrix in Eq. (9.10) is a discrete approximation to a time integral. In general, longer maneuvers (or multiple maneuvers) provide more information, as long as the input continues to excite the system within the output amplitude constraints. In many practical situations, such as subscale model tests, and flight conditions that cannot be sustained for a long time (e.g., high angles of attack or hypersonic flight), the length of individual maneuvers is limited. In addition, finite resources always put a limit on total flight-test time available. Equations (9.13) and (9.14) indicate that large output amplitudes over long time periods will produce the most accurate parameter estimates. However, practical constraints act to limit both the maneuver time and the allowable output amplitudes.

In addition to the foregoing, there are practical considerations related to executing the flight-test maneuver. One such consideration is that the allowable frequencies for the input will be limited at the high end, due to limited instrument dynamic response, reduced dynamic system response to high-frequency inputs, and the high-frequency limitations of the pilot and control system. Inputs must also avoid structural resonance frequencies, to maintain the validity of rigid-body modeling and for safety. Low-frequency inputs can cause the aircraft to drift away from the flight condition selected for the maneuver.

Another practical limitation on input forms might be called implementation distortion. This refers to the practical fact that a designed input may be distorted when actually implemented, owing to the inherent variability of the pilot, or to the actions of the feedback control system. In many cases, the distortion of the input form can actually be helpful. This will be discussed in further detail later.

Finally, in addition to the input amplitude limits described earlier, there are limits on the rate that input amplitudes can be changed. For piloted inputs, this

is the result of human pilot capability. Useful pilot inputs are generally in the frequency range of [0.1, 10] rad/s or [0.016, 1.6] Hz. If the input is implemented by a computerized system, the constraint comes from the control surface actuator rate limits. These limits become important if the desired input is a sharp-edged form, such as a square wave.

### 9.3.3 Single-Input Design

There are many different input forms that can be used for flight-test maneuvers. The main types will be discussed next. Single-input design is discussed first, and then extended to multiple-input design.

All of the input designs discussed here are for aerodynamic model parameter estimation. They are designed as some type of balanced perturbation about the trim condition, so that the flight condition remains essentially unchanged, and the model parameters can be considered constant throughout the maneuver. The inputs shown are excitations starting from the trim value, which is defined as zero for this discussion. In practice, the input forms would be added to the trim value of the control, which may be nonzero.

**Impulse.** Perhaps the simplest of the inputs used for aircraft system identification is the impulse input. This is simply a spike or sudden bump, often called a stick rap. Sometimes the impulse input is two sided to help return the aircraft to the initial condition. Figure 9.7 shows a two-sided impulse input and its power spectrum. The power in the impulse is wideband, but low amplitude.

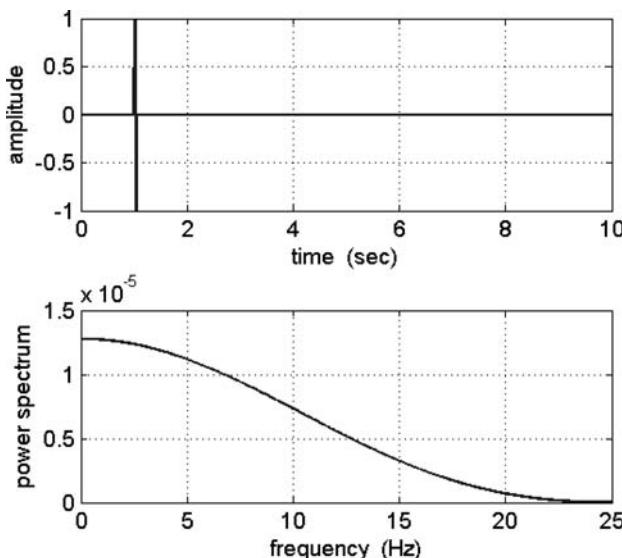


Fig. 9.7 Impulse input.

## EXPERIMENT DESIGN

307

Theoretically, the impulse could be used to collect modeling data when there is no *a priori* information about the aircraft dynamics, but this is often impractical because of low input energy. Impulse inputs are best suited for prediction cases.

**Frequency sweeps.** A more commonly used input type when there is little or no *a priori* information about the dynamic system is the frequency sweep, shown in Fig. 9.8a, along with its power spectrum. The idea behind this input is to apply a continuous sinusoid with the frequency increasing in time, so that

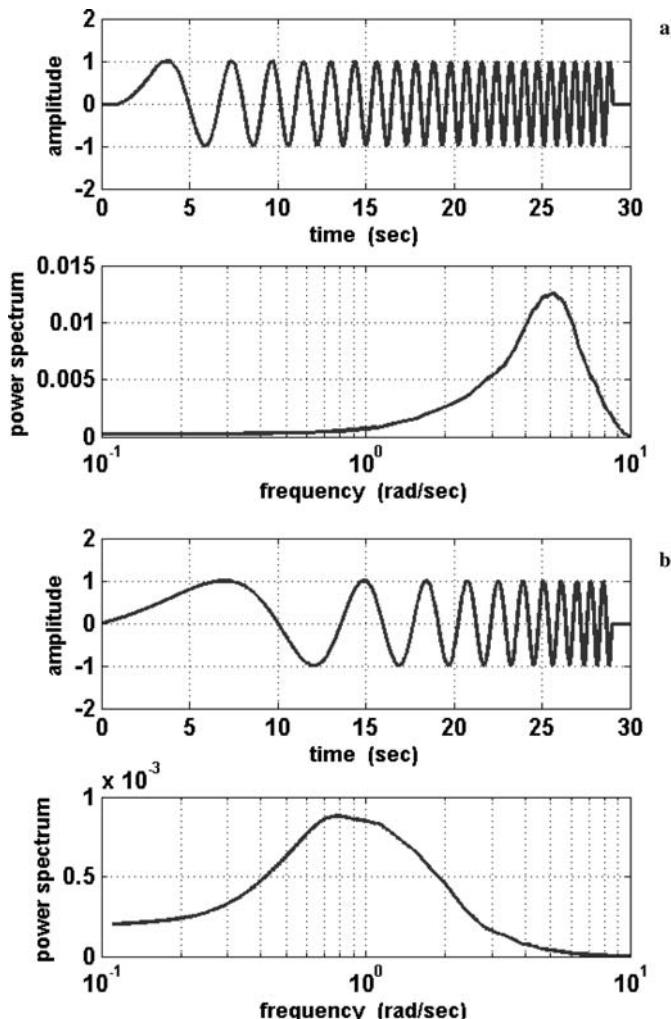


Fig. 9.8 Frequency sweep inputs: a) linear, and b) logarithmic.

the frequency content of the input covers a frequency band of interest. The linear frequency sweep can be described mathematically by

$$u(i) = \sin[\varphi(i)] \quad i = 0, 1, 2, \dots, N - 1 \quad (9.20a)$$

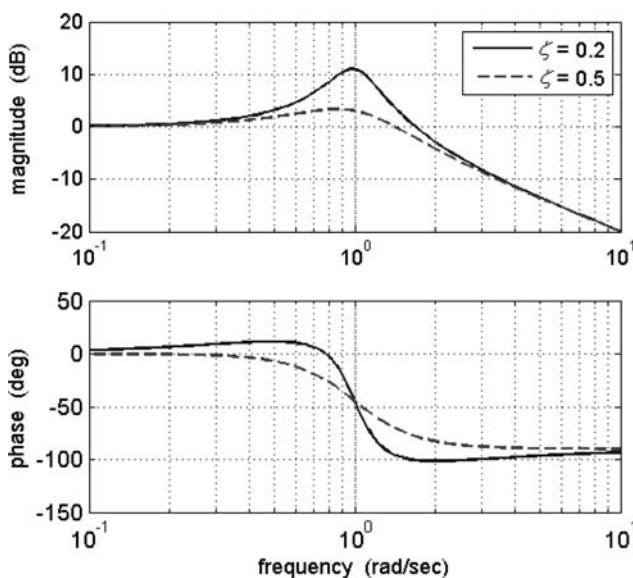
$$\varphi(i) = \omega_0 t(i) + \frac{1}{2}(\omega_1 - \omega_0) \frac{[t(i)]^2}{T} \quad i = 0, 1, 2, \dots, N - 1 \quad (9.20b)$$

where  $t(i) = i\Delta t$ ,  $T$  is the maneuver time,  $T = (N - 1)\Delta t$ , and  $[\omega_0, \omega_1]$  rad/s is the frequency band. The sine function is used so that the input is a perturbation that begins and ends at zero. Low frequencies are applied first, to allow time for the dynamic system response to the lower frequency input.

Frequency sweeps are often used to collect data for generating a Bode plot, shown in Fig. 9.9. The abscissa of the Bode plot is a logarithmic scale, so that using the linear frequency sweep defined in Eq. (9.20) gives sparse frequency content at the lower frequencies. This situation can be remedied by using the logarithmic frequency sweep, which can be implemented using Eq. (9.20a) with

$$\varphi(i) = \omega_0 t(i) + c_2(\omega_1 - \omega_0) \left[ \frac{T}{c_1} e^{c_1 t(i)/T} - t(i) \right] \quad i = 0, 1, 2, \dots, N - 1 \quad (9.21)$$

where  $c_1 = 4$  and  $c_2 = 0.0187$  have been found to work well in practice.<sup>8</sup>



**Fig. 9.9** Frequency response for the second-order system:  $\tilde{y}(s)/\tilde{u}(s) = (s + 1)/(s^2 + 2\xi + 1)$ .

## EXPERIMENT DESIGN

309

This varies the sweep frequency on a logarithmic scale. Figure 9.8b shows a logarithmic frequency sweep and its power spectrum.

The frequency sweep input should contain several complete cycles of many frequencies across a frequency band of interest and therefore can require a relatively long time if low-frequency excitation is included. For a typical pilot input frequency band between 0.1 and 10 rad/s, each frequency sweep input might require 60–90 seconds.

The practical experience of researchers at NASA Ames on rotorcraft and V/STOL aircraft is very extensive in frequency sweep flight-testing and subsequent modeling in the frequency domain. References 8–11 contain detailed design guidelines for frequency sweep inputs, along with many practical examples.

It is not important for frequency sweeps to have constant input amplitude, exact input shape, exact frequency progression, exact repeatability, or sustained high-frequency inputs. At high frequencies, the required cycles are collected relatively quickly, so the corresponding dwell time required is small.

Modeling results actually improve if the inputs for each run have some variety in shape and frequency, because this enhances the information content in the data. Consequently, computerized frequency sweeps have been found to be inferior to piloted frequency sweeps, because the computer does not introduce variations in the input from run to run. Random noise can be added to a computer implementation of the frequency sweep to simulate human pilot variations, but this represents a good deal of complexity to do something that a pilot does naturally. In any case, there must be control inputs applied to suppress off-axis response (e.g., roll response during longitudinal sweeps) and to maintain the aircraft flight condition during the frequency sweep. A good pilot can usually do this.

There are several problems with frequency sweep inputs that sometimes render them unusable for aircraft system identification. One problem is that the sweeps generally take a long time to implement, which rules them out for very short flight-test times, such as drop model testing and flight testing at high angles of attack. Sometimes it is difficult to maintain flight condition in the low-frequency part of the sweep at the start of the maneuver. To handle this, the low-frequency input amplitudes must be reduced considerably, which lowers signal-to-noise ratio. Frequency sweeps are applied to one input at a time, so they are not efficient for a modeling problem with more than one input. For many aircraft, both the longitudinal and lateral dynamic modeling are multiple-input problems. Finally, the sweep frequencies might inadvertently pass through a structural resonance frequency, which can compromise flight safety.

The main advantage of frequency sweep inputs is comprehensive coverage of the frequency band, so that models identified from frequency sweeps typically have very good prediction capability.

**Multisines.** The multisine input is a sum of sinusoids with various frequencies, amplitudes, and phase angles. The frequencies are chosen to cover a frequency band of interest, similar to frequency sweeps, and the amplitudes are chosen to achieve a specific power distribution over the frequency band. Phase

angles can be chosen arbitrarily, and are sometimes set to random values in the interval  $[-\pi, \pi]$  rad.

One type of multisine input that has found use in aircraft system identification is composed of a set of summed harmonic sinusoids with individual phase lags. The input takes the form

$$u(i) = \sum_{k=1}^M A_k \cos\left(\frac{2\pi k t(i)}{T} + \phi_k\right) \quad i = 0, 1, 2, \dots, N-1 \quad (9.22)$$

where  $M$  is the total number of available harmonically related frequencies,  $T$  is the time length of the excitation, and the  $\phi_k$  are phase angles for each of the harmonic components. The  $\phi_k$  can be chosen to produce a low peak factor (PF), defined by

$$\text{PF}(\mathbf{u}) \equiv \frac{[\max(\mathbf{u}) - \min(\mathbf{u})]/2}{\sqrt{(\mathbf{u}^T \mathbf{u})/N}} \quad (9.23)$$

where  $\mathbf{u} = [u_0, u_1, \dots, u_{N-1}]^T$ , or

$$\text{PF}(\mathbf{u}) = \frac{[\max(\mathbf{u}) - \min(\mathbf{u})]/2}{\text{rms}(\mathbf{u})} = \frac{\|\mathbf{u}\|_\infty}{\|\mathbf{u}\|_2} \quad (9.24)$$

The last equality only holds when  $\mathbf{u}$  oscillates symmetrically about zero. In the literature, the quantity  $\|\mathbf{u}\|_\infty/\|\mathbf{u}\|_2$  is called the crest factor. A single sinusoidal component from the summation in Eq. (9.22) has  $\text{PF} = \sqrt{2}$ , so the relative peak factor (RPF) defined by

$$\text{RPF}(\mathbf{u}) = \frac{[\max(\mathbf{u}) - \min(\mathbf{u})]}{2\sqrt{2} \text{rms}(\mathbf{u})} = \frac{\text{PF}(\mathbf{u})}{\sqrt{2}} \quad (9.25)$$

quantifies the peak factor of  $\mathbf{u}$  relative to the peak factor of a single sinusoid. For a single sinusoid, RPF equals 1.

The relative peak factor is a measure of efficiency of an input for parameter estimation purposes, in terms of the amplitude range of the input divided by a measure of the input energy. Lower relative peak factors are more desirable for parameter estimation, where the objective is to excite the system with a variety of frequencies without driving it too far away from the nominal operating point. Inputs with low peak factors are efficient in the sense of providing good input energy over a selected frequency band with low amplitudes in the time domain.

Schroeder<sup>12</sup> has shown that a phase-shifted sum of sinusoids, commonly called the Schroeder sweep, provides an input with good frequency content and low peak factor. For a uniform power spectrum, the Schroeder sweep

## EXPERIMENT DESIGN

311

input results from Eq. (9.22) with

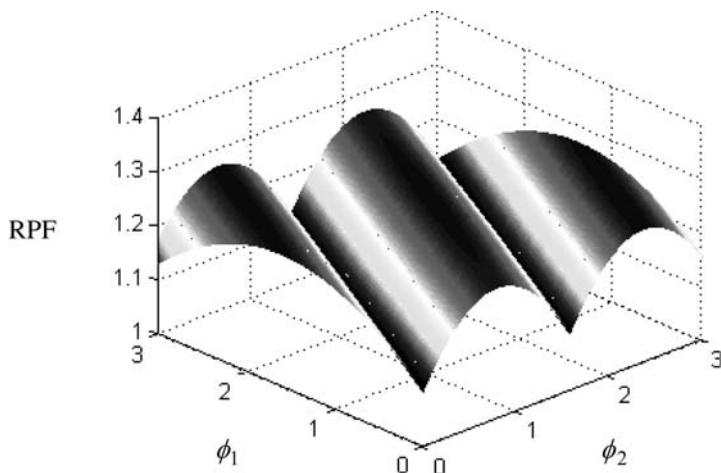
$$\begin{aligned} A_k &= \sqrt{P/M} \\ \phi_1 &= 0 \\ \phi_k &= \phi_{k-1} - \frac{\pi k^2}{M} \quad k = 2, 3, \dots, M \end{aligned} \quad (9.26)$$

where  $P$  is the total desired input power.

Comparisons of the Schroeder sweep with linear and logarithmic frequency sweep inputs have indicated that the Schroeder sweep is generally the superior input for dynamic modeling in the frequency domain.<sup>13</sup> The Schroeder sweep has been used successfully in practical aircraft system identification problems.<sup>14,15</sup>

Using the Schroeder sweep phase angles from Eq. (9.26) results in an input with low relative peak factor, but the input often does not begin and end at zero, as required for a perturbation input. However, the initial phase angle  $\phi_1$  is arbitrary [set to zero in Eq. (9.26)], so  $\phi_1$  can be adjusted to achieve an input that begins and ends at zero. This works because the component sinusoids in Eq. (9.22) are harmonics on the time interval  $[0, T]$ .

It is possible to lower the peak factor of the Schroeder sweep input by starting the phase angles at the Schroeder values from Eq. (9.26), then using a simplex optimization algorithm<sup>16</sup> to minimize the peak factor. The optimization problem is nonconvex, as indicated in Fig. 9.10, which shows the variation of relative peak factor with phase angle choices for a multisine input with two component frequencies, i.e.,  $M = 2$  in Eq. (9.22). Figure 9.10 also shows that there



**Fig. 9.10** Relative peak factors for a two-component multisine input.

are several phase angle solutions that are equally good in terms of low RPF, or nearly so. This approach to optimizing multisine inputs based on minimum RPF is described in Ref. 17 for both single-input and multiple-input cases.

**Doublets and multisteps.** Doublet inputs are two-sided pulses, as shown in Fig. 9.11. This input could be viewed as a square wave approximation to a sine wave. For the doublet shown in Fig. 9.11, the dominant frequency would be 0.5 Hz, corresponding to the frequency of a sine wave with the same period. The power spectrum of the doublet shown in Fig. 9.11 is shifted slightly low in frequency, because only a single isolated doublet is used, rather than a train of repeated doublets. The timing of the pulses for a doublet is chosen so that the dominant frequency in the input is at or close to the expected natural frequency of the dynamic system.

As an example, assume that the dynamic system can be described by a transfer function with a second-order denominator and a first-order numerator, as shown in the frequency response of Fig. 9.9. This situation exists for the transfer function of pitch rate to elevator deflection using the short-period approximation (see Chapter 3). A frequency sweep for this dynamic system would be designed to contain a wide spectrum of frequencies that includes the expected natural frequency of the system. In contrast, the logic for a doublet input would proceed roughly as follows. The static gain is an easy parameter to estimate, and almost any input will suffice to provide information for that parameter. The system is assumed to be second order, which means that at low and high frequencies relative to the natural frequency, the phase and amplitude characteristics are known, because all second-order systems look the same (except for the static

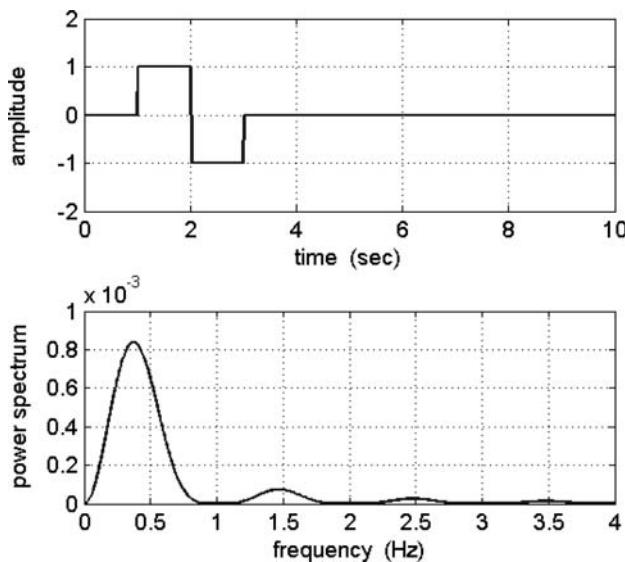


Fig. 9.11 Doublet input.

## EXPERIMENT DESIGN

313

gain) at these frequencies. The natural frequency and damping will be determined most accurately using an input containing frequencies in a band around where the natural frequency of the dynamic system is expected to be. This input does the most good in terms of maximizing information in the data because this is the frequency range where the dynamic responses of second-order systems differ according to the values of the damping and natural frequency. An input with frequencies in the range of the natural frequency of the system therefore collects the required information in a short length of test time.

The lower plot in Fig. 9.11 shows the power spectrum of the doublet input, with the dominant frequency evident. For square wave inputs such as the doublet, it is straightforward to compute the frequency spectrum analytically, as follows.

Consider the doublet to be composed of two individual pulses, with amplitudes of opposite sign. Each pulse can be considered the sum of two step functions. For a pulse  $u_k$  with amplitude  $A_k$  on the interval  $[t_{1k}, t_{2k}]$ , the Fourier transform is

$$\tilde{u}_k(\omega) = \int_0^\infty A_k[1(t - t_{1k}) - 1(t - t_{2k})]e^{-j\omega t} dt \quad (9.27)$$

$$= A_k \left( -\frac{1}{j\omega} e^{-j\omega t} \Big|_{t_{1k}}^\infty + \frac{1}{j\omega} e^{-j\omega t} \Big|_{t_{2k}}^\infty \right)$$

$$\tilde{u}_k(\omega) = \frac{A_k}{j\omega} (e^{-j\omega t_{1k}} - e^{-j\omega t_{2k}}) \quad (9.28a)$$

where  $1(t - t_{1k})$  indicates a step function that initiates at time  $t_{1k}$ .

The Fourier transform for a square-wave input is the sum of terms like Eq. (9.28) for each pulse, at each nonzero frequency  $\omega$ . For  $\omega = 0$ ,

$$\tilde{u}_k(0) = A_k(t_{2k} - t_{1k}) \quad (9.28b)$$

If there are  $m$  pulses, then the Fourier transform for the square-wave input is

$$\begin{aligned} \tilde{u}(\omega) &= \frac{1}{j\omega} \sum_{k=1}^m A_k (e^{-j\omega t_{1k}} - e^{-j\omega t_{2k}}) \\ \tilde{u}(0) &= \sum_{k=1}^m A_k (t_{2k} - t_{1k}) \end{aligned} \quad (9.29)$$

The input power spectrum is computed from the squared magnitude of the Fourier transform for the input, as discussed in Chapter 7,

$$G_{uu}(\omega) = \tilde{u}(\omega)\tilde{u}^*(\omega) \quad (9.30)$$

Note from Fig. 9.11 that the square edges of the doublet input produce a wider frequency spectrum than the single spike that would be obtained with a pure sine wave at the same basic frequency. The square-wave form therefore serves a

useful purpose for inputs intended for system identification, in that the spectrum of the input is broadened, which is a hedge against a poor *a priori* estimate of the modal frequency. This consideration is the reason that a single-frequency sinusoid makes a poor input for system identification.

The amplitude of the doublet input is chosen so that the amplitude of the dynamic response output is large enough for good signal-to-noise ratio, but not so large that the model structure assumption is violated or the model parameters can no longer be considered constant. This can generally be achieved by enforcing the output amplitude constraints mentioned earlier.

Taking the preceding concepts further, more pulses of different widths can be added. A common input for aircraft system identification is called the 3-2-1-1.<sup>18,19</sup> This input consists of alternating pulses with widths in the ratio 3-2-1-1. Figure 9.12 shows an example of this input, along with its power spectrum, computed analytically in the manner described earlier. The width of the 2 pulse is selected to correspond to half the period of the expected natural frequency of the dominant dynamic mode. The 3 and 1 pulses then bracket that frequency on either side, resulting in a relatively wideband input. This input has been called a "poor man's frequency sweep" because of the increasing frequency of the square waves that is analogous to the increasing frequency of a sinusoidal frequency sweep. Comparing Figs. 9.11 and 9.12, the 3-2-1-1 is seen to have much richer frequency content than a doublet.

The 3-2-1-1 input is sometimes difficult to use because the 3 pulse is long and tends to drive the aircraft off flight condition, in a manner similar to what can happen with a frequency sweep. To address this, a 2-1-1 input can be used instead. Figure 9.13 shows this input and its associated power spectrum.

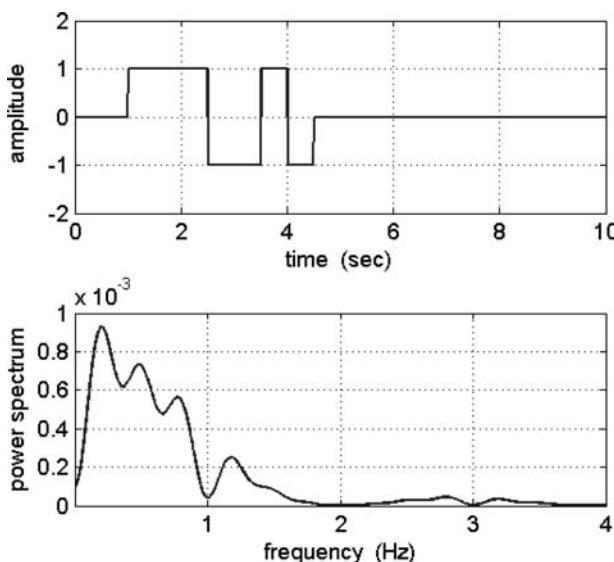


Fig. 9.12 3-2-1-1 input.

## EXPERIMENT DESIGN

315

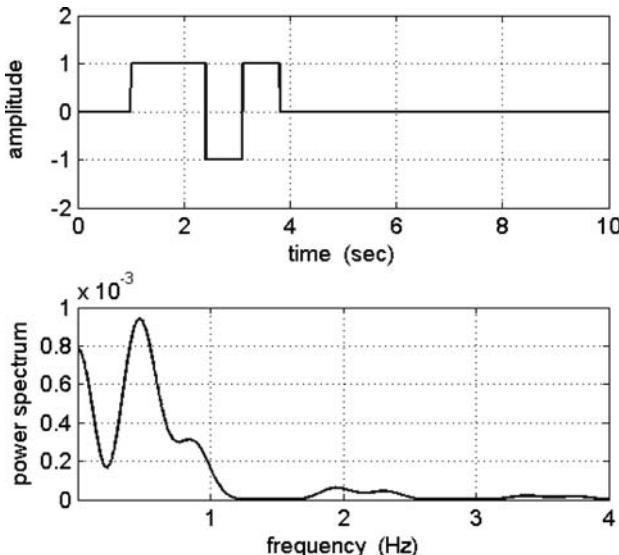


Fig. 9.13 2-1-1 input.

Pulse widths are selected in the ratio 2-1-1. The pulse widths are chosen so that the associated frequencies bracket the expected natural frequency of the dynamic mode to be identified. The following choice for the 1 pulse width has been found to work well:

$$1 \text{ pulse width} = \frac{0.7}{2f_n} \quad (9.31)$$

where  $f_n$  is the expected natural frequency in hz for the dominant mode.

Using Eqs. (9.29) and (9.30), it is possible to investigate the power spectrum of arbitrary square waves with arbitrary amplitudes, or to design a square-wave input that closely approximates an arbitrary input power spectrum, which was done in Ref. 19. Allowing different input amplitudes for each pulse expands the possibilities for input spectra using a multistep input form. The more difficult problem is deciding what the input power spectrum should be. Approximating the chosen input spectra using square-wave inputs is relatively straightforward.

Another practical approach to input design is using a sequence of doublets with different duration. This provides rich frequency content using the simple doublet input form. Figure 9.14 shows a sequence of doublets with different basic frequencies, chosen to bracket the expected natural frequency of the dynamic mode to be modeled. The power spectrum plot in Fig. 9.14 shows that this approach can provide more input power over a widened frequency band, compared with a simple doublet input (cf. Fig. 9.11).

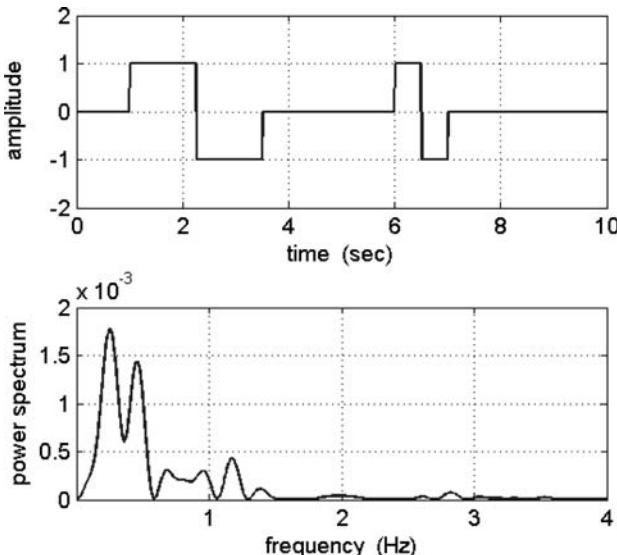


Fig. 9.14 Compound doublet input.

**Pilot inputs.** A good test pilot can very efficiently excite the dynamic response of the aircraft. The key to success in this approach is to make sure the pilot fully understands what is desired and why. Once this is achieved, the pilot's ability to fly the aircraft precisely, and to sense and control the aircraft response, can result in excellent inputs for aircraft system identification.

In addition, the adaptability of a human pilot is a great advantage in flight testing for aircraft system identification. For example, if an input designed prior to flight is based on wind-tunnel data, it is frequently true that the amplitude and/or frequency content are not quite right for the real aircraft in flight. A pilot can make adjustments for this, and many other eventualities that cannot be predicted prior to flight. This issue is particularly important because most of the time and money necessary to do flight tests are spent getting the aircraft in the air, for tasks such as installing instrumentation, checkout, and safety reviews, and not on the operational costs of the aircraft in flight, such as fuel. Once airborne, the best approach is to do some type of testing, even if it's not optimal or exactly what was planned. A pilot who understands the goals of the experiment can adapt the flight-test plan in flight and collect useful flight data.

If the goal is closed-loop modeling, then pilot stick and rudder pedal inputs must be used. However, when the model being identified is the open-loop or bare-airframe model, implementing inputs from the cockpit has limitations, in that sometimes closed-loop feedback control systems move control surfaces in proportion to aircraft states, or move more than one surface in a proportional manner, resulting in data collinearity (see Chapter 5). Often the feedback control system cannot be turned off, because the aircraft is open-loop unstable.

## EXPERIMENT DESIGN

317

There are several approaches to solving this problem, including computerized movement of individual surfaces (discussed later), which approaches the problem by modifying the experiment, and identifying equivalent derivatives or using other data sources to augment the flight data (see Chapter 5), which approaches the problem through the modeling process.

Pilots have successfully implemented impulses, frequency sweeps, doublets, and multistep inputs on many different aircraft. As mentioned earlier, distortions in the desired input form by the human pilot implementation usually help the identification by augmenting the range of frequencies and amplitudes applied to the dynamic system. For multisine inputs, a computerized implementation is required.

**Other input types.** Some specialized inputs that are useful for specific purposes include the longitudinal push-over/pull-up maneuver (essentially a low-frequency longitudinal doublet), which is used for lift and drag performance characterization over a relatively large range of angle of attack, and the windup turn (a turn with increasing aft stick to steadily decrease the turn radius), which is used for data system checkout and data compatibility analysis.

There are other types of inputs for system identification that have been discussed in the literature, such as white noise inputs and pseudo-random binary sequences. The latter input could be described as a long multistep input with constant amplitude and randomly varying pulse widths. Neither of these input types has been widely used in aircraft system identification, mainly because their efficiency is low, and the cost of flight time is very high.

### 9.3.4 Multiple-Input Design

Many practical aircraft modeling problems involve multiple inputs. The most common is aircraft lateral dynamic modeling, which involves rudder and aileron inputs, or equivalent yawing and rolling moment controls. Modern aircraft have many control effectors for both longitudinal and lateral control, so the multiple-input problem appears often.

In the multiple-input case, there are three considerations that are not present for the single input case: relative effectiveness, coordination, and correlation. The discussion here is focused on inputs for bare-airframe model identification, which are typically control surface deflections. The material also applies for closed-loop lateral modeling, which involves lateral stick and rudder pedal inputs.

**Relative effectiveness.** If an input is moved very little or not at all, there is no information on the effect of that input on the aircraft response, and corresponding model parameters cannot be estimated. A control with large effectiveness need not be moved as much as a control with lower effectiveness, to produce the same magnitude change in the aerodynamic forces and moments. Generally, the appropriate input amplitudes for a multiple-input maneuver design are estimated based on *a priori* information from wind-tunnel tests or previous flight-test experience on the same or similar aircraft. For pilot inputs, the test pilot

can adjust the input amplitudes in flight to achieve the desired magnitudes of variation in the response variables.

**Coordination.** Multiple inputs must be coordinated to maximize data information content and to make sure the aircraft responses do not exceed the limits for model structure validity. As a simple example, if two rolling moment control effectors are moved sequentially in the same way, it is easy to exceed roll rate and roll angle limits. In addition, coordination of the inputs can improve the maneuver by taking advantage of dynamic coupling and the natural aircraft motion. The most common example is the case of lateral modeling using a linear model structure and doublet inputs on the rudder and aileron. Applying the rudder doublet first gets the Dutch roll motion of the aircraft started early in the maneuver, so that more data can be collected for this relatively slower dynamic mode. When the aileron doublet is applied subsequently, the roll mode is excited, with some additional excitation of the Dutch roll mode. Since the roll mode is relatively fast, it can be excited later in the maneuver, because less time is required to characterize it. Sequencing the rudder and aileron doublets in this way, with a spacing near the period of the Dutch roll mode, produces a much better maneuver than is obtained when applying the aileron doublet first, followed by the rudder doublet.

**Correlation.** Input correlation refers to how similar the waveforms are for multiple inputs. If all the input waveforms look the same, then any algorithm trying to assign values for the control effectiveness of each individual control will fail, because it is impossible to determine which of the multiple inputs, moved in the same manner, was responsible for the changes in the aerodynamic forces and moments. For the common case of linear aerodynamic models, the preceding statements also apply when the input waveforms are scaled versions of one another. Input forms that are completely decorrelated will give the most accurate control effectiveness estimates, all other things being equal.

It is common for an automatic control system to move two control surfaces in a proportional manner, bringing about nearly exact linear correlation between control surfaces. This is usually done to improve control authority. In this case, the modeling must be done by introducing *a priori* information, such as fixing the effectiveness of all but one of the correlated control surfaces to *a priori* values, or defining a fixed ratio for the control effectiveness of all of the correlated control surfaces relative to the remaining one. Another approach is to estimate a combined control effectiveness, which essentially treats all of the correlated control surfaces as if they were a single control surface. However, if the control system is ever changed so that the scheduled movements of the correlated control surfaces are different (not uncommon in development programs), then the results for the combined control effectiveness become useless.

Mathematically, the correlation between two input waveform vectors  $\mathbf{u}_1$  and  $\mathbf{u}_2$  can be quantified by the pair-wise correlation coefficient for scaled and centered regressors, introduced in Chapter 5, Eq. (5.77), which is a normalized inner product over the length of the maneuver.

## EXPERIMENT DESIGN

319

When the experimenter has complete control of the input forms, multiple inputs can be completely decorrelated by skewing the inputs in time or by using orthogonal forms. Figure 9.15 shows doublet inputs skewed in time, which makes their inner product zero by Eq. (5.77). Figure 9.16 shows multi-sine inputs with different harmonic frequencies, whose inner product is zero since

$$\sum_{i=0}^{N-1} \cos\left[2\pi k_1 \frac{t(i)}{T} + \phi_1\right] \cos\left[2\pi k_2 \frac{t(i)}{T} + \phi_2\right] = 0 \quad (9.32)$$

for integers  $k_1 \neq k_2$  and any constants  $\phi_1$  and  $\phi_2$ .

Figure 9.17 shows orthogonal square waves that also have inner products equal to zero by Eq. (5.77). These square-wave analogs of harmonic sinusoids are sometimes called Walsh functions.

It is more efficient in terms of minimizing maneuver time to move multiple control surfaces simultaneously using orthogonal forms or inputs with low pair-wise correlations. When a feedback control system is operating, desired input forms become distorted by the feedback control. This makes the resulting control surface deflections nonorthogonal, but usually if the inputs for parameter estimation are moving simultaneously and are mutually orthogonal, it takes an extremely high-gain feedback control system to cause the control surface deflections to be correlated to the point where the modeling results are compromised. If time skew is being used to implement input orthogonality, high-gain feedback

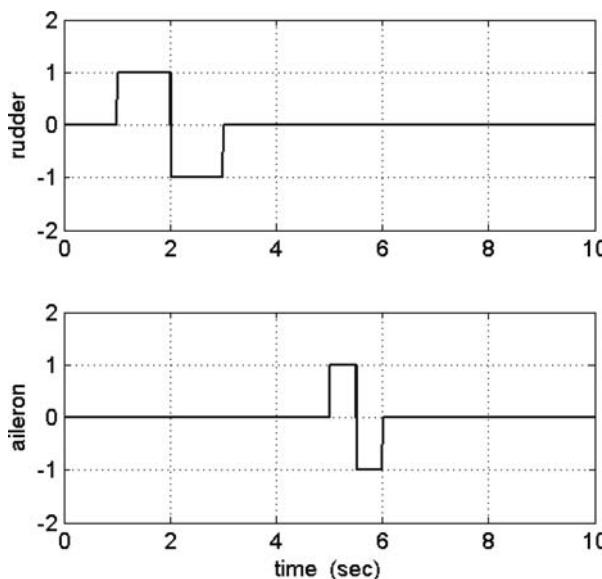


Fig. 9.15 Time-skewed doublet inputs.

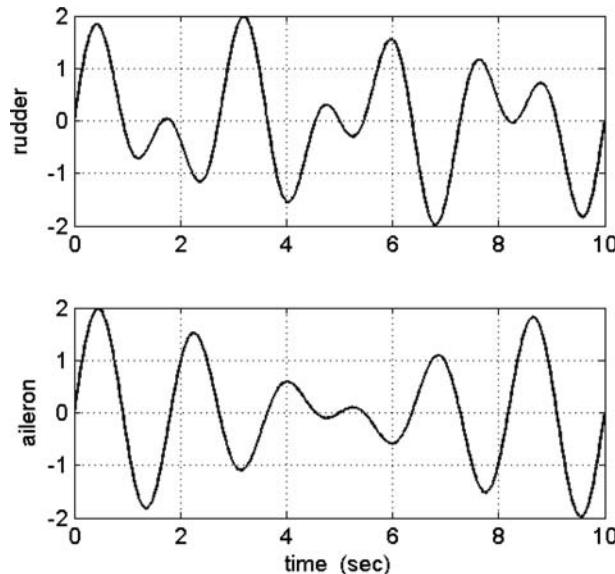


Fig. 9.16 Orthogonal multisine inputs.

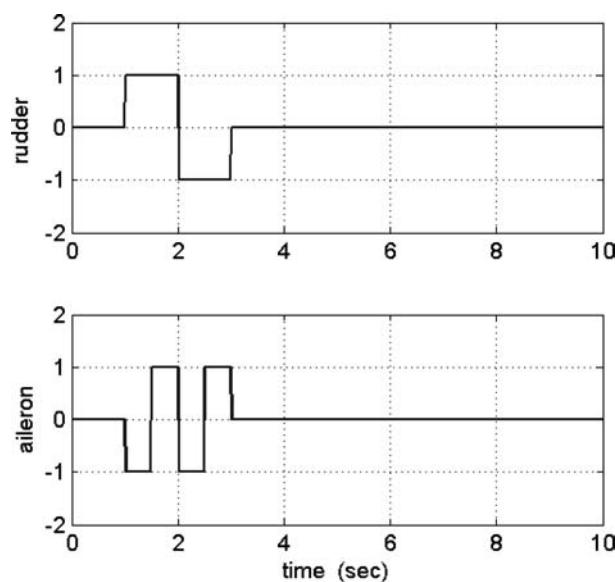


Fig. 9.17 Orthogonal square-wave inputs.

## EXPERIMENT DESIGN

321

can ruin the orthogonality completely, resulting in correlated states and/or controls.

**Optimal inputs.** For a single-input, single-output system with one parameter and a fixed maneuver time  $T$ , an optimal input for model parameter estimation can be found by optimizing a scalar criterion, as indicated in Eq. (9.9). The optimization is applied to quantities that define the input form. For example, if an optimal square-wave input is desired, then the square-wave switching times and amplitudes are quantities that can be adjusted by the optimization. Similarly, a multisine input could be optimized by adjusting amplitudes, frequencies, and phase shifts for the sinusoidal components. This optimal input design problem is the simplest case and is fairly straightforward to solve.

As noted earlier, the input optimization depends on the *a priori* model, which is needed to compute the output sensitivities  $\partial \mathbf{y}(t) / \partial \boldsymbol{\theta}$  and the noise covariance matrix  $\mathbf{R}$ . The input optimization is also subject to constraints on the input and output, the most significant being the amplitude constraints. Although the input amplitude constraint is relatively easy to implement, the output amplitude constraint is not, because the output amplitude depends on the amplitude and frequency content of the input acting through the dynamic system, which is characterized only approximately by the *a priori* model.

The input optimization problem gets much more complex for multiple outputs and more than one model parameter, which is the typical situation for aircraft problems. This requires the use of the information matrix [cf. Eq. (9.10)], so that now a choice must be made as to what function of the information matrix should be used as a scalar optimization criteria. There is not a clear choice—some of the options and their associated implications are discussed by Mehra<sup>20</sup> and Mehra and Gupta.<sup>21</sup> However, it is common to use some scalar function of the information matrix inverse or dispersion matrix [cf. Eq. (9.14)]. Another approach is to minimize one or more selected parameter covariances from the  $\Sigma$  matrix.<sup>22</sup>

There have been many approaches to optimal input design for aircraft parameter estimation, including optimal control formulations using variational calculus,<sup>20,21,23–25</sup> a multidimensional optimization approach using multisine input forms,<sup>6</sup> a suboptimal solution for minimum time to achieve specified parameter error bounds using Walsh functions,<sup>26</sup> and a globally optimal square-wave solution using dynamic programming.<sup>7,27,28</sup> The last technique balances the theoretical objective of exciting the system as much as possible with the constraints and requirements of the practical flight-test situation, such as multiple inputs, constraints on input and output amplitudes, limited flight-test time, avoidance of structural resonance frequencies in the input, testing with active feedback control, implementation by either a pilot or computerized system, and robustness to errors in the *a priori* model. This technique has also been demonstrated in flight.<sup>27–29</sup>

Optimal input design methods in the frequency domain have been developed by Mehra,<sup>20</sup> Mehra and Gupta,<sup>21</sup> and Gupta and Hall.<sup>25</sup> These methods have not been popular in practice, due to relative complexity of the required calculations,

dependence on *a priori* models, practical implementation issues, and the fact that simple inputs such as frequency sweeps and multisteps provide excellent data for modeling in the frequency domain.

Optimal inputs represent a different approach to maneuver design, compared with using wideband heuristic inputs such as frequency sweeps. Specifically, the optimized input applies energy in the vicinity of the break frequencies of the dominant dynamic modes. As discussed earlier in connection with Fig. 9.9, large-scale and important changes in magnitude and phase of a modal response occur near its break frequency, so it is most efficient to concentrate the excitation energy around that frequency when collecting modeling data. The input optimization routine knows approximately where the break frequencies are from an *a priori* model.

In contrast, a heuristic wideband input like a frequency sweep requires little *a priori* information about the system, because the input power is applied approximately uniformly across the entire frequency range of interest. The tradeoff is that when heuristic wideband inputs are used, the maneuver times must be relatively longer and/or maneuvers must be repeated to achieve equivalent modeling accuracy. In general, heuristic wideband inputs are less efficient than optimal inputs, but more robust to deficiencies in *a priori* knowledge of the system being tested.

***Input comparisons.*** It is easy to find works in the literature that compare the effectiveness of various inputs designed for aircraft parameter estimation, particularly for optimal input designs. Unfortunately, making a fair comparison of these inputs is very difficult. This section attempts to explain why.

As discussed previously, input designs for parametric modeling of dynamic systems can be evaluated using the Cramér-Rao bounds from Eq. (9.14). The expression for the information matrix is a discrete approximation to a time integral over the maneuver duration. Therefore, when the effectiveness of various input designs are compared using some function of the dispersion matrix  $\Sigma$  as the criterion for comparison, the input designs being compared should have the same maneuver duration.

Practical considerations impose amplitude constraints on the inputs and outputs for flight-test maneuvers, and  $\Sigma$  depends nonlinearly on the input and output amplitudes. Consequently, for a fair comparison of input forms, maximum input and output amplitudes should also be the same. This approach contrasts with comparisons presented in many studies in the literature, where input comparisons were based on constant input energy. A very wide variety of inputs with varying duration and input amplitudes can have the same constant input energy, as was demonstrated in Fig. 9.6. If the maneuver duration and maximum input and output amplitudes are not the same for the inputs being compared, it is possible to arrange matters so that almost any chosen input form will appear to be the best, using a criterion that depends on  $\Sigma$ . Consequently, the most fair comparison of input forms is achieved when each input has the same time length, and the same input and output amplitude constraints. Then, the input that produces the best measure of data information content, e.g., minimum  $\text{Tr}(\Sigma)$ , can be considered the best for aircraft parameter estimation.

## 9.4 Recommendations

In this section, recommendations are given for successfully planning and executing flight experiments for aircraft system identification. The viewpoint encompasses the entire project, rather than specific details such as flight instrumentation, data acquisition system, and specific input design methods, which were covered previously.

### 9.4.1 Flight-Test Planning

The first and most important step in flight-test planning is to decide on the objective(s) of the investigation. This decision influences all subsequent decisions, and therefore should be made first, in consultation with management and those providing the funding. The most important issues are the following:

- 1) Define what constitutes a successful outcome of the experimentation and modeling. All parties involved should agree on this. Be clear on what the objectives are. For example, is the requirement 10% accuracy on all stability and control derivatives throughout the flight envelope, or is 2% accuracy required only on  $\hat{C}_{m\alpha}$  in approach? The requirements drive how the flight testing and data analysis will be done. In general, higher accuracy requires more information, which means more maneuvers or longer maneuvers will be required to reduce error bounds, or else the signal-to-noise ratios must be enhanced with better maneuvers and/or better instrumentation.
- 2) If possible, build iteration into the plan. If everything that could happen were known *a priori*, there would be no need to conduct the experiment. It is not practical to decide on the entire test program *a priori*. To some, this has the look of making the problem open-ended, but the iterative approach is necessary from a technical standpoint. Here is a fictional (but realistic) exchange that illustrates the problem:

Manager: Tell me how much flight time you will need for the project, so I can enter that number into my spreadsheet and compute the total cost for my budget planning.

Engineer: That depends on what we find.

Manager: Don't be difficult. I want your complete list of maneuvers for the project on my desk by the end of the week.

- 3) In light of the last item, spend no more than one-quarter of total resources on the initial flight tests. Unfortunately, the time when the most effective experiment can be designed is near the end of the program, when the most is known. The best arrangement is to conduct an initial test flight, then stop flight-test operations and analyze the data all the way through to obtaining model parameter estimates and error bounds. This makes it possible to identify any data recording or instrumentation problems, and to make sure that all the necessary information for successful dynamic modeling is being collected in an acceptable form. Any problems discovered can be corrected at a time when most of the flight-test time is still in

the future. Based on the preliminary results, the flight-test plan should be reformulated as necessary before continuing. It is often surprising how much the information from one flight can change the flight-test plan. If iteration is impossible for some reason, another approach is to use some form of in-flight adaptive system identification.<sup>30</sup>

- 4) Make sure the objectives can be achieved with the available resources. Limited resources always have an effect on how flight experiments are conducted, and on the results obtained. When choices must be made, it is usually best to cut back on the number of objectives and get good results for fewer objectives than to try to get everything and obtain mediocre results. Sometimes resource limitations affect how many different flight conditions can be studied. In this case, the flight conditions to be studied must be prioritized by importance to the goals of the investigation. If the goal is flight envelope expansion, then the sequence of flight conditions might be defined by increasing trim angle of attack or Mach number.

#### **9.4.2 Data Collection and Input Design**

A general principle of system identification is that the identified model can only capture behavior that is exhibited by the system and embodied in the measured data. For example, the phugoid (long-period) dynamics cannot be modeled based on data from a short-duration maneuver. Similarly, if the effect of a particular control surface is to be modeled, that control surface must be moved during the testing, and the movement must be sufficiently different from simultaneous variations in other explanatory variables. The same principle applies to all explanatory variables. The fact that the explanatory variables cannot be moved independently for a flying airplane is part of what makes effective experiment design for aircraft system identification a challenging task.

If it is known that some surfaces will always be moved in a specified way (such as inboard flaps that always move symmetrically with the same amplitude), then these surfaces can be treated as a single control surface for modeling purposes. This simplifies the analysis, since only one combined effectiveness parameter will be estimated, and reduces flight-test and instrumentation requirements, since there is only one effective surface being moved. In general, control surfaces should be tested in the same way as they will be used. For example, if surfaces will always be deflected together symmetrically or asymmetrically, test them that way, rather than moving each control surface individually. This ensures that any interaction effects are properly characterized. The negative aspect of this approach is that if the scheduled movement of the surfaces is changed, more testing will be required, where the surfaces will have to be moved according to the new schedule or individually, to characterize the control effectiveness properly.

System identification can be viewed as processing information from measured data to produce model parameter estimates and associated error bounds. For a fixed amount of information from a given maneuver or set of maneuvers, as more model parameter estimates are requested, the parameter estimates get less accurate. In a sense, the fixed amount of information is spread more thinly over the estimation of more model parameters. Because of this, more complicated

## EXPERIMENT DESIGN

325

model structures with more model parameters require more information to get accurate model parameter estimates. This additional information comes from more or better maneuvers with information content applicable to the model being identified.

At each flight condition, it is advantageous to repeat each maneuver at least once (i.e., two maneuvers total), but preferable to repeat each maneuver three or four times (four or five maneuvers total). Repeats of a maneuver are like insurance from the viewpoint of the data analyst. This is true because of data dropouts, bad trims, turbulence, etc., which affect the data, and may render some maneuvers unusable for the intended purpose. More important, significant improvements in accuracy and confidence in the results come with averaging from repeated maneuvers. Variations in the input forms for the repeated maneuvers are desirable, because these differences excite the dynamic system in a more diverse manner, resulting in a more robust identified model. Inputs can be varied by simply reversing input polarity, or by using variations of the amplitudes and frequencies of the input form. When a pilot implements the inputs, these variations come automatically with the inherent variability of the human pilot.

To set up for a maneuver, the aircraft should be flown in an accurate reference flight condition, with the data collection system turned on, for at least 2 s before initiating the maneuver. This enforces a steady initial flight condition and provides data for accurate estimation of the initial condition. When the excitation is complete, the pilot should allow 5 s of free response with initial trim controls held, before taking control of the aircraft. This is done to make sure that the entire free response is recorded.

Unless there are severe practical limits on the time available for each maneuver, a good rule of thumb is to excite the aircraft for a time length of roughly 5 times the period of the dominant dynamic mode. In most cases, it is adequate to use a fixed maneuver time of 10–20 s for parameter estimation maneuvers, unless the modeling will include slower dynamics such as the phugoid mode or spiral mode, in which case a much longer maneuver time will be needed.

Table 9.3 contains a list of important issues in designing inputs for aircraft system identification flight-test maneuvers, along with a brief explanation of why each is important.

If the aircraft has a direct connection from the pilot stick and rudder controls to the control surfaces and there is no feedback control, then the pilot can very effectively implement the required inputs for flight-test maneuvers intended for collecting modeling data. This situation is in fact preferable for system identification testing, because of the richness associated with the variability in the input implementation by the pilot, and the fact that a pilot can easily change amplitudes, frequencies, and waveforms in real time during the flight.

It is not always necessary that the aircraft have the full complement of sensors listed in Table 9.1. For example, if the objective is to identify a LOES model to characterize the longitudinal flying qualities associated with the short-period response, only measurements of the longitudinal stick deflection and pitch rate are required. On the other hand, all the values listed in Table 9.1, plus mass and inertia properties, and aircraft reference geometry, would be required to estimate longitudinal and lateral nondimensional stability and control derivatives. In general, the quantities that need to be measured are determined by the objectives of the investigation.

**Table 9.3 Summary of important considerations for flight-test maneuvers**

Desirable features	Reasons
Time-skewed inputs or orthogonal inputs	Input decorrelation for accurate control effectiveness estimates, improved excitation
All controls moving at once	Shorten required flight-test time
Low peak factors for the inputs	Maintain flight condition, collect data consistent with model structure assumptions, good data information content
Lots of control activity	Accurate control effectiveness estimates, persistent excitation for good data information content
Double-sided inputs	Maintain desired flight condition, collect data consistent with model structure assumptions, accurate Fourier transforms
Perturbation inputs begin and end at zero (2 s at start, 5 s at end)	Taylor series modeling, maintain desired flight condition, good initial condition estimate, full dynamic free response at the end, accurate Fourier transforms
Repeated maneuvers with slightly different inputs	Improves modeling accuracy and prediction capability, provides insurance against poorly executed maneuvers

#### **9.4.3 When to Optimize Inputs**

Designing optimal inputs involves a much larger investment of time and resources than the alternative heuristic methods. However, there are situations when this extra effort is justified. One of these is when the maneuver time is extremely limited for practical reasons. This can occur for unpowered drop model tests, or for specialized tests that are necessarily of short duration because the desired flight condition cannot be maintained for very long, such as flight at high angles of attack and hypersonic flight. Most of the heuristic input design methods do not include an effective means for extending the design to multiple inputs, whereas the most recent and practical optimal input design methods can handle this with only a corresponding increase in computational effort.

The cynical view of optimal input design is that it can only be done effectively when it isn't really needed. Most input optimization techniques for parametric model identification require an *a priori* model, and the input optimization gets better as the *a priori* model improves. However, an optimized square-wave input form has enough frequency spillover so that the *a priori* model need not be very accurate, and the input optimization can improve parameter estimation results.<sup>27–29</sup> An input design can be optimized so that the data information content for a fixed maneuver time is maximized for the estimation of a particular

## EXPERIMENT DESIGN

327

model parameter or subset of model parameters that might be of special interest. To save flight-test time, optimized inputs can be designed for minimum maneuver time required to achieve specific accuracy goals for selected model parameters.

A type of input optimization that is independent of an *a priori* model might be called a nonparametric frequency domain method, where the input is designed to match a chosen input spectrum in a least-squares sense.<sup>19</sup> However, the only methods available to give guidance as to what the optimal input spectrum should be, depend (again), on *a priori* parametric models.<sup>20</sup> Otherwise, the best approach is a flat input spectrum over a range of frequencies thought to include the natural frequencies of the dominant dynamic modes.

Frequency sweeps, multisines, doublets, and multistep inputs have been used successfully for aircraft system identification flight-testing for many years. In general, optimal input designs, such as the square-wave input design described in Ref. 7, can produce maneuvers with more information content per unit of flight-test time than heuristic input designs. The price for this is a requirement for *a priori* information, typically in the form of wind-tunnel data or aerodynamic calculations such as computational fluid dynamics (CFD), and increased computational resources for the input optimization. Refs. 27–29 have shown that practical obstacles to using an optimal square-wave input in flight have been overcome. However, in general, the recommendation is to use heuristic inputs with wideband frequency content, except in cases with special requirements, such as multiple inputs, individual parameter accuracy goals, or short maneuver times, which require the enhanced information content provided by an optimized input.

## 9.5 Open-Loop Parameter Estimation from Closed-Loop Data

Some aircraft are designed so that the airframe is unstable without feedback control. In this case, the aircraft is said to be open-loop unstable. This is normally done to improve maneuverability or to reduce trim drag, and means that the aircraft would not be controllable by the pilot if the control surfaces were moved exclusively by pilot stick and rudder pedal inputs. Therefore, an open-loop unstable aircraft must have an automatic feedback control system operating continuously for the aircraft to be flown in a stable and controllable manner.

The impact of automatic feedback control in terms of aircraft system identification is that any input implemented by the pilot or introduced directly at the control surface actuator by a computerized system will be distorted by the feedback control. When conducting flight tests for aircraft dynamic modeling, the objective is to excite the natural aircraft motion as much as possible, within the practical constraints of the flight test. The objective of the feedback control is generally stabilization and disturbance rejection. Excitation initiated by the pilot or a computerized system to implement a system identification maneuver is seen by the feedback control system as a disturbance that should be damped and eliminated. To do this, the feedback system moves the control surfaces in a manner that mutes the natural dynamic response of the aircraft. The result is that the perturbation input form is distorted and the natural dynamic response is subdued. It follows that feedback control works against the objectives of the

system identification experiment. Therefore, when system identification maneuvers are flown, it is advantageous to reduce feedback control gains as much as possible without risking departure.

Note that the bare-airframe dynamic response is not altered by feedback control. The feedback control modifies the aircraft closed-loop response to pilot inputs by altering the control surface deflections, but cannot change the bare-airframe dynamics.

Many modern aircraft, including those that are open-loop unstable, have multiple control surfaces with redundant functions. For example, an aircraft with a canard and all-moving horizontal tail has two control surfaces that can change the aerodynamic pitching moment. Such aircraft normally have an automatic feedback control system. Often, the feedback control system moves the multiple control surfaces so that their motion is highly correlated with one another, or with state variables. To continue the earlier example, the control system on an aircraft with a canard and all-moving horizontal tail will typically move the canard and horizontal tail in a proportional way, with the ratio scheduled as a function of angle of attack. This is done to distribute the aerodynamic loads and reduce trim drag. Consequently, for a small-perturbation flight-test maneuver, the ratio of the canard and horizontal tail movements is practically a fixed number, and the data for the two control surfaces are highly collinear, leading to severe problems in the modeling (cf. Chapter 5). If the same aircraft also has relaxed longitudinal static stability, then both control surfaces may also be highly correlated with the pitch rate, for example, due to the stability augmentation function of the control system. This was seen in Example 5.4 for longitudinal maneuvers on the X-29 aircraft.

This problem can be solved through the modeling process, by estimating equivalent derivatives, or by using biased estimators, possibly with the flight data augmented by information from other sources, as discussed in Chapter 5. However, it is preferable to address the problem by applying orthogonal perturbations for system identification directly at the control surface actuators using a computerized system. The perturbations can be added to the control surface actuator command coming from the control system. These perturbations are sometimes called independent surface inputs. The effect of the added perturbations is to decorrelate the resulting control surface motion from other explanatory variables, and thereby allow accurate determination of the control surface effectiveness and other model parameters. For multiple inputs, time-skewed or otherwise orthogonal perturbation inputs must be applied to at least  $n_i - 1$  of the  $n_i$  inputs to decorrelate all the inputs.

Outside of the input distortion and data collinearity caused by the feedback control, data analysis and modeling required to estimate open-loop or bare-airframe model parameters when the maneuvers are performed with a feedback control system operating are exactly the same as described throughout this book. It is therefore possible, and indeed not any more difficult, to estimate open-loop model parameters from data measured with the aircraft operating under closed-loop feedback control. The problem is getting enough excitation past the feedback control system so that accurate modeling can be done based on measured data from the experiment.

## 9.6 Summary and Concluding Remarks

An essential part of aircraft system identification is design of the experiment. When this is done well, the data analysis and modeling normally proceed without much trouble. On the other hand, if a significant mistake is made in the experimentation, often there are no postflight data processing remedies, and the data can be practically useless for modeling purposes.

This chapter laid out important considerations in experiment design for aircraft system identification, including the characteristics of the data acquisition system and flight instrumentation, and the design of excitation inputs. Heuristic and optimal design approaches were presented for both single- and multiple-input design cases. Optimal input design for parameter estimation maneuvers involves optimizing a scalar measure of signal-to-noise ratios for the modeling problem, or an input efficiency metric, subject to practical constraints of the flight-test environment. Heuristic input design approaches generally implement some form of wideband input, with frequency content selected to include or be concentrated near the expected modal frequencies of the aircraft.

The chapter summarized general recommendations for effectively conducting flight-test programs for aircraft dynamic modeling. Practical problems related to experimentation for aircraft parameter estimation were discussed, along with suggested solutions. These problems included data collinearity and input distortion resulting from active feedback control, and estimation of bare-airframe model parameters from data collected under closed-loop feedback control.

## References

- <sup>1</sup>*Basic Principles of Flight Test Instrumentation Engineering*, AGARDograph No. 160, Vol. 1, 1974.
- <sup>2</sup>Shannon, C. E., "Communication in the Presence of Noise," *Proceedings of the IRE*, Vol. 37, No. 1, 1949, pp. 10–21.
- <sup>3</sup>Phillips, W. H., "Effects of Model Scale on Flight Characteristics and Design Parameters," *Journal of Aircraft*, Vol. 31, No. 2, March–April, 1993, pp. 454–457.
- <sup>4</sup>Steers, S. T., and Iliff, K. W., "Effects of Time-Shifted Data on Flight-Determined Stability and Control Derivatives," NASA TN D-7830, 1975.
- <sup>5</sup>Klein, V., "Estimation of Aircraft Aerodynamic Parameters from Flight Data," *Progress in Aerospace Sciences*, Vol. 26, No. 1, 1989, pp. 1–77.
- <sup>6</sup>Mulder, J. A., "Design and Evaluation of Dynamic Flight Test Manoeuvres," Delft University of Technology, Dept. of Aerospace Engineering, Rept. LR-497, Delft, The Netherlands, 1986.
- <sup>7</sup>Morelli, E. A., "Practical Input Optimization for Aircraft Parameter Estimation Experiments," Sc.D. Dissertation, Joint Institute for Advancement of Flight Sciences, George Washington Univ., Hampton, VA, 1990; republished as NASA CR 191462, May 1993.
- <sup>8</sup>Tischler, M. B., "Frequency-Response Identification of the XV-15 Tilt-Rotor Aircraft Dynamics," NASA TM 89428, 1987.
- <sup>9</sup>Tischler, M. B., and Cauffman, M. G., "Frequency-Response Method for Rotorcraft System Identification with Applications to the BO 105 Helicopter," *American Helicopter Society 46th Annual Forum*, May 1990, pp. 99–137.

- <sup>10</sup>Tischler, M. B., Fletcher, J. W., Diekmann, V. L., Williams, R. A., and Cason, R. W., "Demonstration of Frequency-Sweep Testing Technique Using a Bell 214-ST Helicopter," NASA TM 89422, 1987.
- <sup>11</sup>Williams, J. N., Ham, J. A., and Tischler, M. B., "Flight Test Manual, Rotorcraft Frequency Domain Flight Testing," U.S. Army Aviation Technical Test Center, AQTD Project No. 93-14, Edwards AFB, CA, 1995.
- <sup>12</sup>Schroeder, M. R., "Synthesis of Low-Peak-Factor Signals and Binary Sequences with Low Autocorrelation," *IEEE Transactions on Information Theory*, Vol. IT-18, No. 1, 1970, pp. 85-89.
- <sup>13</sup>Young, P., and Patton, R. J., "Comparison of Test Signals for Aircraft Frequency Domain Identification," *Journal of Guidance, Control, and Dynamics*, Vol. 13, No. 3, 1990, pp. 430-438.
- <sup>14</sup>Bosworth, J. T., and Burken, J. J., "Tailored Excitation for Multivariable Stability-Margin Measurement Applied to the X-31A Nonlinear Simulation," AIAA Paper 94-3361, 1997.
- <sup>15</sup>Klein, V., and Murphy, P. C., "Aerodynamic Parameters of High Performance Aircraft Estimated from Wind Tunnel and Flight Test Data," *System Identification for Integrated Aircraft Development and Flight Testing*, RTO-MP-11, Paper 18, 1999.
- <sup>16</sup>Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. R., *Numerical Recipes in FORTRAN: The Art of Scientific Computing*, 2nd ed., Cambridge Univ. Press, New York, 1992.
- <sup>17</sup>Morelli, E. A., "Multiple Input Design for Real-Time Parameter Estimation in the Frequency Domain," *13th IFAC Symposium on System Identification*, Paper REG-360, 2003.
- <sup>18</sup>Koehler, R., and Wilhelm, K., "Auslegung von Eingangssignalen für die Kennwertermittlung" ("Design of Input Signals for Identification"), DFVLR Institut für Flugmechanik, IB 154-77/40, Brunswick, Germany, 1977 (in German).
- <sup>19</sup>Proskawetz, K.-O., "Optimierung stufenförmiger Eingangssignale im Frequenzbereich für die Parameteridentifizierung" ("Multi-Step Input Signals for Parameter Estimation, Optimized in the Frequency Domain"), *Zeitschrift für Flugwissenschaften und Weltraumforschung*, Vol. 9, No. 6, 1985, pp. 362-370 (in German).
- <sup>20</sup>Mehra, R. K., "Optimal Input Signals for Parameter Estimation in Dynamic Systems—Survey and New Results," *IEEE Transactions on Automatic Control*, Vol. AC-19, No. 6, 1974, pp. 753-768.
- <sup>21</sup>Mehra, R. K., and Gupta, N. K., "Status of Input Design for Aircraft Parameter Identification," *Methods for Aircraft State and Parameter Identification*, AGARD-CP-172, Paper 12, 1975.
- <sup>22</sup>Wells, W. R., and Ramachandran, S., "Input Design for Minimum Correlation Between Aerodynamic Parameters," *Fifth Symposium on Nonlinear Estimation*, 1974.
- <sup>23</sup>Reid, D. B., "Optimal Inputs for System Identification," Dept. of Aeronautics and Astronautics, Stanford Univ., SUDAAR No. 440, Stanford, CA, 1972.
- <sup>24</sup>Stepner, D. E., and Mehra, R. K., "Maximum Likelihood Identification and Optimal Input Design for Identifying Aircraft Stability and Control Derivatives," NASA CR-2200, 1973.
- <sup>25</sup>Gupta, N. K., and Hall, W. E., Jr., "Input Design for Identification of Aircraft Stability and Control Derivatives," NASA CR-2493, 1975.

## EXPERIMENT DESIGN

331

- <sup>26</sup>Chen, R. T. N., "Input Design for Aircraft Parameter Identification: Using Time-Optimal Control Formulation," *Methods for Aircraft State and Parameter Identification*, AGARD-CP-172, Paper 13, 1975.
- <sup>27</sup>Morelli, E. A., "Advances in Experiment Design for High Performance Aircraft," *AGARD System Identification Specialists Meeting*, Paper 8, 1998.
- <sup>28</sup>Morelli, E. A., "Flight Test of Optimal Inputs and Comparison with Conventional Inputs," *Journal of Aircraft*, Vol. 36, No. 2, 1999, pp. 389–397.
- <sup>29</sup>Cobleigh, B. R., "Design of Optimal Inputs for Parameter Estimation Flight Experiments with Application to the X-31 Drop Model," M.S. Thesis, Joint Institute for Advancement of Flight Sciences, George Washington Univ., Hampton, VA, 1991.
- <sup>30</sup>Morelli, E. A., "In-Flight System Identification," AIAA Paper 98-4261, 1998.

This page intentionally left blank

**10****Data Compatibility**

Instrumentation on a research aircraft includes sensors that measure accelerations, rates, and positions associated with the translational motion of the c.g. and the rotational motion about the c.g., as well as the magnitude and orientation of the air-relative velocity. Kinematic relationships among these quantities can be used to check that the measurements are mutually consistent. An analysis of this type is called a data compatibility analysis or a data consistency check.

If all the measurements were perfect, the data compatibility analysis would show that the kinematic relationships are perfectly satisfied by the sensor measurements. In practice, each sensor measurement has both systematic and random errors. The kinematic relationships are used as a tool to quantify these instrumentation errors and correct the measured data from the sensors for systematic errors. This is an important task, which results in a kinematically consistent data set with improved accuracy. A consistent and accurate data set is a prerequisite for the model structure determination and parameter estimation stages of the system identification process.

Measurements of control surface deflections, power settings, and pilot inputs are notably absent from the foregoing discussion. These measurements cannot be checked against others, because there are no kinematic relationships between these measurements and others in the data set. Consequently, the discussion that follows only applies for measurements associated with the rigid-body motion of the aircraft, which can be checked for compatibility with other measurements.

### **10.1 Kinematic Equations**

The kinematic equations are composed of the translational equations of motion in body axes, the rotational kinematic equations, and the navigation equations in body axes. These equations were derived in Chapter 3. The translational equations of motion in body axes are

$$\dot{u} = rv - qw + \frac{\bar{q}SC_X}{m} - g \sin \theta + \frac{T}{m} \quad (10.1a)$$

$$\dot{v} = pw - ru + \frac{\bar{q}SC_Y}{m} + g \cos \theta \sin \phi \quad (10.1b)$$

$$\dot{w} = qu - pv + \frac{\bar{q}SC_Z}{m} + g \cos \theta \cos \phi \quad (10.1c)$$

The specific applied forces in  $g$  units can be replaced by the translational accelerometer measurements (cf. Sec. 3.7)

$$a_x = \frac{1}{mg}(\bar{q}SC_X + T) \quad (10.2a)$$

$$a_y = \frac{1}{mg}(\bar{q}SC_Y) \quad (10.2b)$$

$$a_z = \frac{1}{mg}(\bar{q}SC_Z) \quad (10.2c)$$

Combining Eqs. (10.1) and (10.2),

$$\dot{u} = rv - qw - g \sin \theta + ga_x \quad (10.3a)$$

$$\dot{v} = pw - ru + g \cos \theta \sin \phi + ga_y \quad (10.3b)$$

$$\dot{w} = qu - pv + g \cos \theta \cos \phi + ga_z \quad (10.3c)$$

In this form, the translational equations of motion relate only kinematic quantities. The rotational kinematic equations and the navigation equations in body axes are used in the forms derived in Chapter 3. The complete set of kinematic equations is:

*Translational kinematics:*

$$\dot{u} = rv - qw - g \sin \theta + ga_x \quad (10.4a)$$

$$\dot{v} = pw - ru + g \cos \theta \sin \phi + ga_y \quad (10.4b)$$

$$\dot{w} = qu - pv + g \cos \theta \cos \phi + ga_z \quad (10.4c)$$

*Rotational kinematics:*

$$\dot{\phi} = p + \tan \theta(q \sin \phi + r \cos \phi) \quad (10.5a)$$

$$\dot{\theta} = q \cos \phi - r \sin \phi \quad (10.5b)$$

$$\dot{\psi} = \frac{q \sin \phi + r \cos \phi}{\cos \theta} \quad (10.5c)$$

## DATA COMPATIBILITY

335

*Position kinematics:*

$$\begin{aligned}\dot{x}_E = & u \cos \psi \cos \theta + v(\cos \psi \sin \theta \sin \phi - \sin \psi \cos \phi) \\ & + w(\cos \psi \sin \theta \cos \phi + \sin \psi \sin \phi)\end{aligned}\quad (10.6a)$$

$$\begin{aligned}\dot{y}_E = & u \sin \psi \cos \theta + v(\sin \psi \sin \theta \sin \phi + \cos \psi \cos \phi) \\ & + w(\sin \psi \sin \theta \cos \phi - \cos \psi \sin \phi)\end{aligned}\quad (10.6b)$$

$$\dot{h} = u \sin \theta - v \cos \theta \sin \phi - w \cos \theta \cos \phi \quad (10.6c)$$

The kinematic equations are coupled, nonlinear state differential equations for data compatibility analysis. However, the position kinematic states  $x_E$  and  $y_E$  do not appear in the other kinematic equations. Altitude is the only position state that has any bearing on the aircraft stability and control (because air density varies with altitude), so the kinematic equations associated with  $x_E$  and  $y_E$  are generally not used.

Output equations for data compatibility analysis define the relationship between the kinematic states and aircraft responses that are directly measured. For the translational kinematics, measured outputs are related to the states by Eqs. (3.46),

$$V = \sqrt{u^2 + v^2 + w^2} \quad (10.7a)$$

$$\alpha = \tan^{-1}\left(\frac{w}{u}\right) \quad (10.7b)$$

$$\beta = \sin^{-1}\left(\frac{v}{\sqrt{u^2 + v^2 + w^2}}\right) \quad (10.7c)$$

The Euler angles and altitude are also measured outputs, so their output equations are simply

$$\phi = \phi \quad (10.7d)$$

$$\theta = \theta \quad (10.7e)$$

$$\psi = \psi \quad (10.7f)$$

$$h = h \quad (10.7g)$$

For the kinematic data consistency check, the body-axis angular rates,  $p$ ,  $q$ , and  $r$ , and the measured acceleration at the c.g.,  $a_x$ ,  $a_y$ , and  $a_z$ , are considered inputs in the state equations. The states are  $u$ ,  $v$ ,  $w$ ,  $\phi$ ,  $\theta$ ,  $\psi$ , and  $h$ , and the outputs are  $V$ ,  $\alpha$ ,  $\beta$ ,  $\phi$ ,  $\theta$ ,  $\psi$ , and  $h$ .

The general idea is that a subset of the measurements is used as inputs to the kinematic equations, which are solved and used with the output equations to generate reconstructed values for a different subset of the measurements. If all the measurements are compatible, then the reconstructed outputs will match the measurements of the same output quantities, except for random measurement noise. If the reconstructed outputs do not match the measured outputs, then instrumentation error parameters are introduced into the equations, and values

for these parameters are estimated using optimization techniques, such as those discussed in Chapters 6 and 8 in conjunction with aerodynamic parameter estimation.

Measured accelerations and angular rates include random measurement noise, so that the kinematic differential equations are in fact stochastic. In some formulations, to be explained in more detail later, this fact is ignored, and the differential equations are solved as if they were deterministic. As long as the measurement noise on the accelerations and angular rates is relatively small and zero mean, the numerical solution of the kinematic equations is not affected significantly by this process noise. It is also possible to use filtering or smoothing techniques (see Chapter 11) to remove random noise from the measured acceleration and angular rate signals before using them as inputs to the kinematic equations.

## 10.2 Data Reconstruction

Solving the kinematic equations numerically using measured values of the inputs  $a_x$ ,  $a_y$ ,  $a_z$ ,  $p$ ,  $q$ ,  $r$  and initial conditions for the states  $u$ ,  $v$ ,  $w$ ,  $\phi$ ,  $\theta$ ,  $\psi$ , and  $h$ , and using the output equations, results in reconstructed time series for the outputs  $V$ ,  $\alpha$ ,  $\beta$ ,  $\phi$ ,  $\theta$ ,  $\psi$ , and  $h$ . The initial conditions can be obtained from smoothed measured values at the initial time. Outputs from this process are called reconstructed outputs, because they have been calculated from other measurements and kinematic relationships. This process is sometimes called flight-path reconstruction.

The reconstructed outputs are sometimes used in lieu of sensor measurements in cases where the sensors are either impractical or simply not available. A good example is the angle of attack on a hypersonic vehicle. Any sort of vane or probe mounted on a vehicle of this type would melt immediately because of aerodynamic heating. Angle of attack can instead be reconstructed from acceleration and angular rate measurements and the kinematic equations.

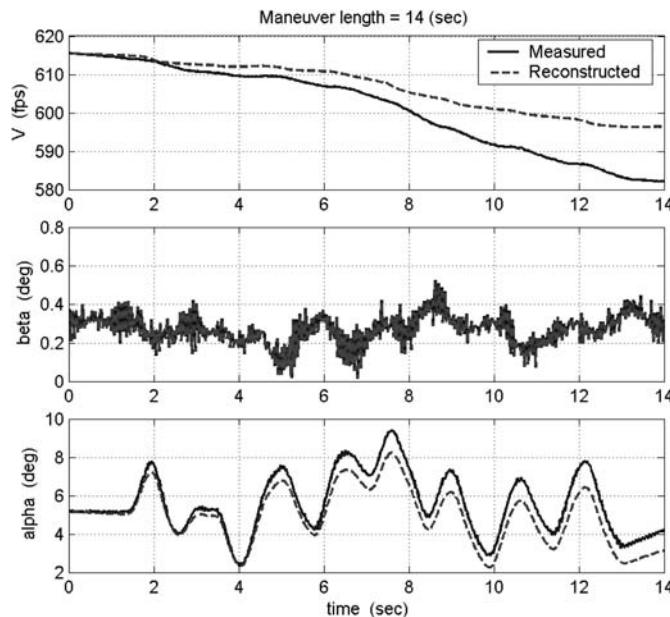
The main problem with this approach is that the kinematic equations are unstable relative to the inputs  $a_x$ ,  $a_y$ ,  $a_z$ ,  $p$ ,  $q$ , and  $r$ . For example, a bias in the  $a_z$  measurement would cause the  $w$  state to drift steadily over time, impacting the reconstructed values of the air-relative velocity data. An example of this is shown in Fig. 10.1 using flight data from a longitudinal maneuver on the NASA F-18 High Alpha Research Vehicle (HARV).

The solid lines in Fig. 10.1 show measured outputs and the dashed lines represent reconstructed outputs. In practice, it is common for the measured kinematic inputs  $a_x$ ,  $a_y$ ,  $a_z$ ,  $p$ ,  $q$ , and  $r$  to contain bias errors, causing the type of drift in the reconstructed outputs shown in Fig. 10.1. If the bias errors in the measured outputs are ignored, then the drift can be estimated and removed by imposing the condition that the reconstructed outputs match the measured outputs at the endpoints of the time series.

Another approach is to use a complementary filter, where the output is reconstructed as described earlier for the high-frequency response, and the low-frequency response is computed from *a priori* aerodynamic parameter information. Figure 10.2 shows a block diagram of a complementary filter for an output estimate  $\hat{y}$ , where the subscript  $c$  indicates a value calculated from

## DATA COMPATIBILITY

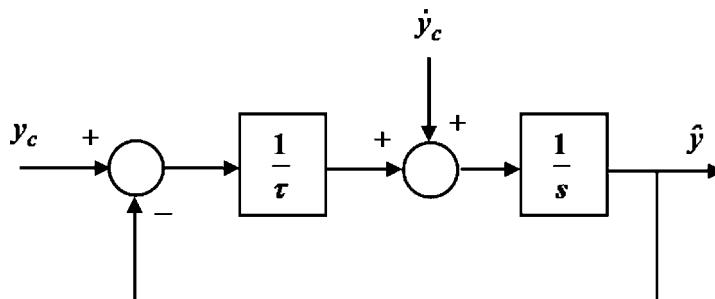
337



**Fig. 10.1** Data compatibility analysis for the NASA F-18 HARV.

other measurements. Based on Fig. 10.2, the estimated output  $\hat{y}$  can be expressed as

$$\hat{y} = \frac{1}{s + 1/\tau} s y_c + \frac{1}{\tau s + 1} y_c \quad (10.8)$$



$$\hat{y} = \begin{cases} \text{filtered } y_c & \text{for } \omega < 1/\tau \\ \text{integrated } \dot{y}_c & \text{for } \omega > 1/\tau \end{cases}$$

**Fig. 10.2** Complementary filter.

The estimated output  $\hat{y}$  is computed by integrating  $\dot{y}_c$  for frequencies  $\omega \gg 1/\tau$ , whereas for  $\omega \ll 1/\tau$ ,  $\hat{y}$  tracks  $y_c$ . At frequencies near  $1/\tau$ , the estimated output  $\hat{y}$  is a combination of the two components.

For example, the equations of a complementary filter for the sideslip angle  $\beta$  are

$$\dot{\beta}_c \approx \frac{g}{V} (a_y + \sin \phi \cos \theta) + p \sin \alpha - r \cos \alpha \quad (10.9a)$$

$$\beta_c = \frac{1}{C_{Y_\beta}} \left( \frac{mg}{\bar{q}S} a_y - C_{Y_r} \frac{rb}{2V} - C_{Y_\delta} \Delta \delta \right) \quad (10.9b)$$

where the first equation comes from Eq. (3.39c), assuming the sideslip angle is small, and the second comes from Eqs. (3.54b) and (3.67), with the assumption  $C_{Y_p} \approx 0$ . Of course, the low-frequency  $\hat{\beta}$  will depend on the accuracy of the *a priori* estimates of the aerodynamic parameters used in Eq. (10.9b).

Another problem with using inertial measurements to reconstruct data such as angle of attack and sideslip angle is that the reconstruction assumes the atmosphere is fixed relative to earth axes. When there are gusts or winds, this assumption is not valid, and the reconstructed data have errors. Gusts and winds are usually not measured or at best are measured at locations and times near the location and time of the flight. If flight testing is done on calm days, this type of error in the reconstructed data is small, and the instrumentation systematic errors can be accurately estimated.

## 10.3 Aircraft Instrumentation Errors

Three types of corrections are made to measurements from aircraft instrumentation: ground calibration, sensor alignment and position corrections, and systematic instrumentation error corrections. The first two types of corrections are made as part of the data reduction and do not change unless new sensor hardware is installed on the aircraft. Systematic instrumentation error corrections require analysis and computation that is specific to each maneuver.

### 10.3.1 Calibration

Calibrations are available from the sensor manufacturer and can also be obtained from laboratory testing on the ground. Calibration for the rate gyros can be done using a rate table, where known rates are applied to a surface on which the sensors are mounted. Euler angle sensors and accelerometers can be tested by mounting them on a surface with known angular position. The accelerometers should measure the negative gravity vector, which is the specific force applied by the surface to the sensor. For an extended range, the sensitivity axis of the accelerometer is aligned with the radius of a rate table to experience the normal acceleration of uniform circular motion.

The same general idea can be used as a check of angle of attack and pitch angle measurements with translational accelerometer measurements  $a_x$  and  $a_z$ ,

using data from steady, wings-level flight:

$$\alpha = \theta = \sin^{-1}(a_x) = \cos^{-1}(a_z)$$

which follows from Eqs. (10.4a) and (10.4c).

Flow angle sensors are best calibrated in a wind tunnel, preferably mounted in a manner that mimics the installation on the aircraft. Such calibrations provide local flow corrections, also called upwash corrections for angle of attack and sidewash corrections for sideslip angle. There are also several flight-test methods that can be used to calibrate air data sensors.<sup>1</sup>

### 10.3.2 Sensor Alignment and Position Errors

The sensitive axes of the rate gyros and accelerometers should be aligned with the body axes of the aircraft, so that the components of the angular velocity and translational acceleration are assigned to the proper component. If this alignment cannot be done for some reason, corrections can be applied, as long as the angular displacements from the proper alignment are known.<sup>2</sup> The corrections involve expressing a vector in a rotated axis system, which is discussed in Appendix A. For angle of attack, sideslip angle, and the Euler angles, any sensor misalignment is a constant bias error that can be measured on the ground and removed from the raw data as part of the data reduction.

The equations of motion derived in Chapter 3 assume that the aircraft responses are measured at the c.g. However, the c.g. is not a fixed location on the aircraft, because of changes in loading and fuel state. This is not a problem for measured Euler angles  $\phi$ ,  $\theta$ ,  $\psi$ , angular rates  $p$ ,  $q$ ,  $r$ , and angular accelerations  $\dot{p}$ ,  $\dot{q}$ ,  $\dot{r}$ , because these quantities are independent of the sensor position on a rigid body.

For flow angle data, the variation in measured values due to sensor position is the result of the aircraft rotational motion about the c.g. The rotation produces additional air-relative velocity at any sensor location displaced from the c.g. If  $[x_\alpha \ y_\alpha \ z_\alpha]^T$  and  $[x_\beta \ y_\beta \ z_\beta]^T$  denote the position vectors of the angle of attack and sideslip angle sensors relative to the c.g. in body axes, then the measured values from the air data sensors are

$$\alpha_E = \tan^{-1}\left(\frac{w - qx_\alpha + py_\alpha}{u}\right) \quad (10.10a)$$

$$\beta_E = \sin^{-1}\left(\frac{v + rx_\beta - pz_\beta}{\sqrt{u^2 + v^2 + w^2}}\right) \quad (10.10b)$$

where the subscript  $E$  denotes the measured value from the experiment. For small angular rates and small aerodynamic angles, Eqs. (10.10) can be written as

$$\alpha = \alpha_E + \frac{qx_\alpha}{V} - \frac{py_\alpha}{V} \quad (10.11a)$$

$$\beta = \beta_E - \frac{rx_\beta}{V} + \frac{pz_\beta}{V} \quad (10.11b)$$

Similar corrections could be specified for the airspeed measurement, but typically these corrections are small relative to nominal values of airspeed, so the corrections can be omitted.

For translational accelerometers mounted at a location different from the c.g., rotational motion about the c.g. results in tangential and centripetal accelerations that are picked up by accelerometers. For accelerometers located at a location relative to the c.g. defined by  $[x_a \ y_a \ z_a]^T$  in body axes, the position corrections are

$$g \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix}_{\text{c.g.}} = g \begin{bmatrix} a_{xE} \\ a_{yE} \\ a_{zE} \end{bmatrix} + \begin{bmatrix} (q^2 + r^2) & -(pq - \dot{r}) & -(pr + \dot{q}) \\ -(pq + \dot{r}) & (p^2 + r^2) & -(qr - \dot{p}) \\ -(pr - \dot{q}) & -(qr + \dot{p}) & (p^2 + q^2) \end{bmatrix} \begin{bmatrix} x_a \\ y_a \\ z_a \end{bmatrix} \quad (10.12)$$

The position vectors  $[x_\alpha \ y_\alpha \ z_\alpha]^T$ ,  $[x_\beta \ y_\beta \ z_\beta]^T$ , and  $[x_a \ y_a \ z_a]^T$  are found using the current aircraft c.g. location and the known position of the sensors. The units for the position vector elements must be compatible with the correction equations, which usually means the units are feet or meters. The definitive reference for the instrumentation error corrections described here, along with many other types and variations, is Ref. 2.

Sensor position corrections in Eqs. (10.10)–(10.12) require values for the body-axis angular rates and angular accelerations. The body-axis angular rates are measured, and the body-axis angular accelerations can be obtained from measurements or from smoothed differentiation of the angular rates (see Chapter 11). All of these quantities can be noisy, which results in an increased noise level on the corrected accelerations and air-relative velocity data, due to the position corrections. This problem can be solved by applying smoothing or zero-phase filtering techniques to the angular rates and accelerations before using them for position corrections (see Chapter 11).

### 10.3.3 Sensor Dynamics

Dynamic characteristics of sensors are provided by the manufacturer or can be evaluated from dynamic tests in the laboratory. Because the natural frequencies of sensors in aircraft instrumentation systems are usually very high relative to the frequencies associated with the quantities being measured, the sensor dynamics can be approximated by a small time delay or simply neglected.<sup>3</sup>

## 10.4 Model Equations for Data Compatibility Check

All of the errors discussed so far do not vary over the course of a flight-test program unless some instrumentation hardware is replaced, modified, or moved. In contrast, systematic instrumentation errors can change with the maneuver type and the flight condition. Consequently, these errors are often estimated for each individual maneuver or each type of maneuver.<sup>3</sup>

The measurement equation model for aircraft sensors with typical systematic instrumentation errors is

$$z(i) = (1 + \lambda)y(i) + b + v(i) \quad i = 1, 2, \dots, N \quad (10.13)$$

## DATA COMPATIBILITY

341

where  $z(i)$  is the  $i$ th measured output from the sensor,  $y(i)$  is the true output,  $v(i)$  is random measurement noise,  $\lambda$  is a constant scale factor error parameter, and  $b$  is a constant bias error parameter. If the instrumentation error parameters  $\lambda$  and  $b$  are zero, the measured output equals the true output, corrupted only by random measurement noise. Figure 10.3 shows that the instrumentation error model in Eq. (10.13) constitutes a simple model of the relationship between the measured output from the sensor and the true output. Practical experience with flight test instrumentation has shown that this simple model is adequate for most systematic instrumentation errors. More sophisticated measurement equation models could be used instead, if the situation warranted, without any major changes in the methods used to estimate the instrumentation error parameters.

In practice, not all sensors require both the scale factor and bias error parameters. Table 10.1 shows the typical instrumentation error parameters for various sensors. Constant bias terms are omitted for heading angle  $\psi$  and altitude  $h$ , because the reference values of these variables can be selected arbitrarily. In some cases, the initial conditions for the states in the kinematic equations are also included as unknown parameters.

Combining the information in Table 10.1 with Eqs. (10.4)–(10.6) and (10.13), the state-space form of the kinematic equations used for data compatibility analysis is

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \\ \dot{h} \end{bmatrix} = \begin{bmatrix} 0 & r_E - b_r & -(q_E - b_q) & 0 \\ -(r_E - b_r) & 0 & p_E - b_p & 0 \\ q_E - b_q & -(p_E - b_p) & 0 & 0 \\ \sin \theta & -\cos \theta \sin \phi & -\cos \theta \cos \phi & 0 \end{bmatrix} \begin{bmatrix} u \\ v \\ w \\ h \end{bmatrix} + \begin{bmatrix} -g \sin \theta + g a_{x_E} - b_{a_x} \\ g \cos \theta \sin \phi + g a_{y_E} - b_{a_y} \\ g \cos \theta \cos \phi + g a_{z_E} - b_{a_z} \\ 0 \end{bmatrix} + \begin{bmatrix} 0 & w & -v & 0 \\ -w & 0 & u & 0 \\ v & -u & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} v_p \\ v_q \\ v_r \\ 0 \end{bmatrix} + \begin{bmatrix} v_{a_x} \\ v_{a_y} \\ v_{a_z} \\ 0 \end{bmatrix} \quad (10.14a)$$

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \tan \theta \sin \phi & \tan \theta \cos \phi \\ 0 & \cos \phi & -\sin \phi \\ 0 & \frac{\sin \phi}{\cos \theta} & \frac{\cos \phi}{\cos \theta} \end{bmatrix} \begin{bmatrix} p_E - b_p + v_p \\ q_E - b_q + v_q \\ r_E - b_r + v_r \end{bmatrix} \quad (10.14b)$$

where the subscript  $E$  again indicates measured values from the experiment. For the equations in this section, it is assumed that the measured values have been

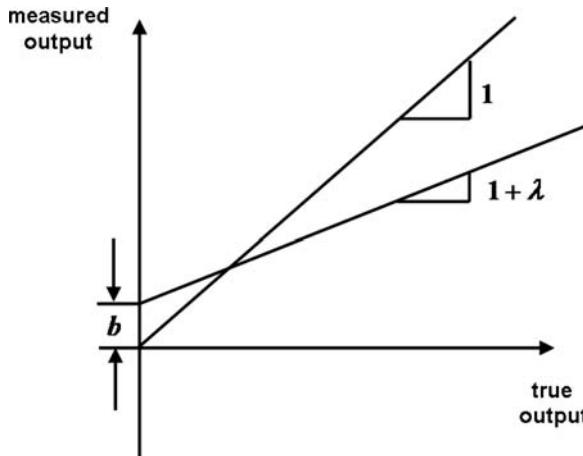


Fig. 10.3 Instrumentation error model.

corrected to the vehicle c.g. using the methods described earlier. The measured output equations are

$$V_E(i) = (1 + \lambda_V)\sqrt{u^2(i) + v^2(i) + w^2(i)} + b_V + v_V(i) \quad (10.15a)$$

$$\beta_{f_E}(i) = \left(1 + \lambda_{\beta_f}\right) \tan^{-1} \left[ \frac{v(i)}{u(i)} \right] + b_{\beta_f} + v_{\beta_f}(i) \quad (10.15b)$$

Table 10.1 Typical instrumentation errors

Sensor	Variable	Bias error	Scale factor error
Translational accelerometer (specific applied forces)	$a_x, a_y, a_z$	X	—
Rotational accelerometer (body-axis angular accelerations)	$\dot{p}, \dot{q}, \dot{r}$	X	—
Rate gyro (body-axis angular rates)	$p, q, r$	X	—
Airflow angle vane (angle of attack, sideslip angle)	$\alpha, \beta$	X	X
Dynamic pressure sensor (airspeed)	$V$	X	X
Integrating gyro (Euler angles)	$\phi, \theta, \psi$	X	X
Pressure altimeter (altitude)	$h$	—	X

## DATA COMPATIBILITY

343

$$\alpha_E(i) = (1 + \lambda_\alpha) \tan^{-1} \left[ \frac{w(i)}{u(i)} \right] + b_\alpha + v_\alpha(i) \quad (10.15c)$$

$$\phi_E(i) = (1 + \lambda_\phi) \phi(i) + b_\phi + v_\phi(i) \quad (10.15d)$$

$$\theta_E(i) = (1 + \lambda_\theta) \theta(i) + b_\theta + v_\theta(i) \quad (10.15e)$$

$$\psi_E(i) = (1 + \lambda_\psi) \psi(i) + v_\psi(i) \quad (10.15f)$$

$$h_E(i) = (1 + \lambda_h) h(i) + v_h(i) \quad (10.15g)$$

The expression for the measured sideslip angle assumes a vane sensor, so that the angle measured is actually the flank angle, as discussed in Chapter 3. The sideslip angle to be used for aerodynamic modeling is computed from Eq. (3.49),

$$\beta = \tan^{-1}(\beta_f \cos \alpha) \quad (10.16)$$

State equations (10.14) represent a nonlinear stochastic system where the measurement noise in the acceleration and angular rate signals acts as process noise. State and parameter estimation, either simultaneous or separate, would be an extremely difficult task.

A simplification is achieved by assuming that the translational accelerations and angular rates have no random measurement errors when used as inputs to the kinematic differential equations. Then Eqs. (10.14) can be written as

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \\ \dot{h} \end{bmatrix} = \begin{bmatrix} 0 & r_E - b_r & -(q_E - b_q) & 0 \\ -(r_E - b_r) & 0 & p_E - b_p & 0 \\ q_E - b_q & -(p_E - b_p) & 0 & 0 \\ \sin \theta & -\cos \theta \sin \phi & -\cos \theta \cos \phi & 0 \end{bmatrix} \begin{bmatrix} u \\ v \\ w \\ h \end{bmatrix} + \begin{bmatrix} -g \sin \theta + g a_{x_E} - b_{ax} \\ g \cos \theta \sin \phi + g a_{y_E} - b_{ay} \\ g \cos \theta \cos \phi + g a_{z_E} - b_{az} \\ 0 \end{bmatrix} \quad (10.17a)$$

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \tan \theta \sin \phi & \tan \theta \cos \phi \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} p_E - b_p \\ q_E - b_q \\ r_E - b_r \end{bmatrix} \quad (10.17b)$$

Further simplification can be achieved by separating the kinematic differential equations and the measured output equations into longitudinal and lateral subsets when the flight-test maneuver contains mainly longitudinal or lateral motion. However, maneuvers used for kinematic consistency checks are often larger-amplitude maneuvers such as windup turns and push-over/pull-up maneuvers, and in any case, the kinematic equations should include the full nonlinearity in the equations. Instead, if the maneuver involves mostly longitudinal motion, measured values for the lateral states are substituted into the equations. Similarly,

for lateral maneuvers, measured values for the longitudinal states are used in the equations. This technique for decoupling the equations was discussed near the end of Chapter 3, in connection with aerodynamic modeling.

Longitudinal states for the kinematic consistency check are  $u, w, \theta, h$ , with measured outputs  $V, \alpha, \theta, h$ . For lateral kinematic consistency check, the states are  $v, \phi, \psi$ , with measured outputs  $\beta_f, \phi, \psi$ . For maneuvers with combined motion, such as windup turns, the full sets of Eqs. (10.14) or (10.17) with (10.15) are used.

## 10.5 Instrumentation Error Estimation Methods

Adjusting the unknown instrumentation error parameters in Eqs. (10.14) and (10.15) so that outputs from the model equations match the measured data in a specified way forms the basis for data compatibility analysis. When combined with the kinematic relationships derived earlier, the measured data from the aircraft instrumentation constitute a redundant set of measurements for the aircraft responses. This redundancy is used to estimate instrumentation error parameters in a postulated model structure for the instrumentation errors. The structure of the instrumentation error model is based largely on experience, but careful residual analysis can be used to help.

Various methods for data compatibility analysis and flight-path reconstruction have been developed and applied to simulated and real flight-test data. These methods can be divided into two groups:

- 1) Methods for estimating states and parameters simultaneously, using filtering and/or smoothing methods;
- 2) Methods for estimating states and parameters separately, using the maximum likelihood method for a stochastic system or the output-error method for a deterministic system.

### 10.5.1 Filters and Smoothers

The state and output equations for data compatibility analysis, Eqs. (10.14) and (10.15), can also be written as

$$\dot{\mathbf{x}}(t) = \mathbf{f}[\mathbf{x}(t), \mathbf{u}(t)] + \mathbf{B}_w[\mathbf{x}(t)]\mathbf{w}(t) \quad (10.18a)$$

$$\mathbf{y}(t) = \mathbf{h}[\mathbf{x}(t)] \quad (10.18b)$$

$$\mathbf{z}(i) = \mathbf{h}[\mathbf{x}(i)] + \mathbf{v}(i) \quad i = 1, 2, \dots, N \quad (10.18c)$$

$$E[\mathbf{x}(0)] = \bar{\mathbf{x}}_0 \quad E\{[\mathbf{x}(0) - \bar{\mathbf{x}}_0][\mathbf{x}(0) - \bar{\mathbf{x}}_0]^T\} = \mathbf{P}_0$$

$$E[\mathbf{w}(t)] = \mathbf{0} \quad E[\mathbf{w}(t_i)\mathbf{w}^T(t_j)] = \mathbf{Q}(t_i)\delta(t_i - t_j)$$

$$E[\mathbf{v}(i)] = \mathbf{0} \quad E[\mathbf{v}(i)\mathbf{v}^T(j)] = \mathbf{R}(i)\delta_{ij} \quad (10.18d)$$

where  $\mathbf{x}(0)$ ,  $\mathbf{w}(t)$ , and  $\mathbf{v}(i)$  are uncorrelated, and

$$\mathbf{x} = [u \ v \ w \ \phi \ \theta \ \psi \ h \ \mathbf{b}^T \ \boldsymbol{\lambda}^T]^T \quad (10.18e)$$

$$\mathbf{u} = [a_x \ a_y \ a_z \ p \ q \ r]^T \quad (10.18f)$$

$$\mathbf{y} = [V \ \beta \ \alpha \ \phi \ \theta \ \psi \ h]^T \quad (10.18g)$$

## DATA COMPATIBILITY

345

and the symbols  $\mathbf{b}$  and  $\boldsymbol{\lambda}$  denote column vectors containing the bias and scale factor error parameters. The physical state vector has been augmented with the instrumentation error parameters. The extended Kalman filter can be used to estimate the augmented state vector. For this nonlinear model in continuous-discrete form, the extended Kalman filter can be developed in the same way as for Eqs. (4.71) in Chapter 4 or can be taken from Ref. 4 as

*Initial conditions:*

$$\begin{aligned}\hat{\mathbf{x}}(0|0) &= \bar{\mathbf{x}}_0 \\ \mathbf{P}(0|0) &= \mathbf{P}_0\end{aligned}\quad (10.19a)$$

*Prediction:*

$$\frac{d}{dt}[\hat{\mathbf{x}}(t)] = \mathbf{f}[\hat{\mathbf{x}}(t), \mathbf{u}(t)] \quad (10.19b)$$

$$\begin{aligned}\frac{d}{dt}[\mathbf{P}(t)] &= \mathbf{A}[\hat{\mathbf{x}}(t)]\mathbf{P}(t) + \mathbf{P}(t)\mathbf{A}^T[\hat{\mathbf{x}}(t)] + \mathbf{B}_w[\hat{\mathbf{x}}(t)]\mathbf{Q}(t)\mathbf{B}_w^T[\hat{\mathbf{x}}(t)] \\ (i-1)\Delta t &\leq t \leq i\Delta t\end{aligned}\quad (10.19c)$$

*Measurement update:*

$$\hat{\mathbf{x}}(i|i) = \hat{\mathbf{x}}(i|i-1) + \mathbf{K}(i)\{\mathbf{z}(i) - \mathbf{h}[\hat{\mathbf{x}}(i|i-1)]\} \quad (10.19d)$$

$$\mathbf{K}(i) = \mathbf{P}(i|i-1)\mathbf{C}^T(i)[\mathbf{C}(i)\mathbf{P}(i|i-1)\mathbf{C}^T(i) + \mathbf{R}(i)]^{-1} \quad (10.19e)$$

$$\mathbf{P}(i|i) = [\mathbf{I} - \mathbf{K}(i)\mathbf{C}(i)]\mathbf{P}(i|i-1) \quad (10.19f)$$

The matrices  $\mathbf{A}[\hat{\mathbf{x}}(t)]$  and  $\mathbf{C}(i)$  are defined as

$$\mathbf{A}[\hat{\mathbf{x}}(t)] = \left. \frac{\partial \mathbf{f}[\mathbf{x}(t), \mathbf{u}(t)]}{\partial \mathbf{x}} \right|_{\mathbf{x}(t) = \hat{\mathbf{x}}(t)} \quad (10.20a)$$

$$\mathbf{C}(i) = \left. \frac{\partial \mathbf{h}[\mathbf{x}(i)]}{\partial \mathbf{x}} \right|_{\mathbf{x}(i) = \hat{\mathbf{x}}(i|i-1)} \quad (10.20b)$$

One of the first algorithms for data compatibility analysis was given by Chen and Eulrich.<sup>5</sup> The estimation technique used was an extended Kalman filter with one-stage optimal smoothing. Application of this smoothing can reduce the bias in the estimates, which is inherent to the extended Kalman filter. To further improve the estimates, the algorithm also included a fixed-point smoother, which gives smoothed initial estimates of states and parameters and their covariance matrix. The same technique was used by Klein and Schiess,<sup>6</sup> augmented by analysis of residuals, which can help in assessing model adequacy for a given set of measured data.

Mulder<sup>7</sup> and Jonkers<sup>8</sup> presented four algorithms for data consistency analysis and reconstruction of output variables not available from measurements.

These techniques include extended and linearized Kalman filters with and without the fixed-point smoother. Leach and Hui<sup>9</sup> describe a novel application of an extended Kalman filter-smoother optimization technique for air data reconstruction. This technique determines the pitot-static position error and calibration curves for the angle of attack and sideslip angle onboard an aircraft.

The algorithm for aircraft state estimation described in Ref. 10 is based on a variational solution of a nonlinear, fixed-interval smoothing problem. It is an iterative scheme, providing improved state estimates until the minimum of a squared error measure is achieved. Linearization is about a nominal trajectory. The solution of the fixed-interval smoothing problem consists of determining initial condition estimate  $\hat{\mathbf{x}}(0)$  and process noise (forcing function)  $\mathbf{w}(i)$  by minimizing the cost function

$$J = \frac{1}{2} [\hat{\mathbf{x}}(0) - \bar{\mathbf{x}}_0]^T \mathbf{P}_0^{-1} [\hat{\mathbf{x}}(0) - \bar{\mathbf{x}}_0] + \frac{1}{2} \sum_{i=1}^N \{ [\mathbf{z}(i) - \mathbf{y}(i)]^T \mathbf{R}^{-1} [\mathbf{z}(i) - \mathbf{y}(i)] + \mathbf{w}(i-1)^T \mathbf{Q}^{-1} \mathbf{w}(i-1) \} \quad (10.21)$$

subject to the constraints given by Eqs. (10.18a)–(10.18c). In the cost function,  $\bar{\mathbf{x}}_0$  is an *a priori* estimate of  $\mathbf{x}(0)$ , and  $\mathbf{P}_0$ ,  $\mathbf{Q}$ , and  $\mathbf{R}$  are weighting matrices. The estimation algorithm is available as a FORTRAN program called SMoothing for AirCraft Kinematics (SMACK). This program has been used for flight data compatibility analysis at NASA Ames and for flight-path reconstruction to assist the National Transportation Safety Board in investigations of aircraft accidents.<sup>10</sup>

### 10.5.2 Output Error

Data compatibility analysis involves the nonlinear kinematic model and output equations (10.14) and (10.15), with constant unknown instrumentation error parameters to be estimated. To simplify the problem, Jonkers<sup>8</sup> and Wingrove<sup>11</sup> recommended that the output-error method be applied. This is the same type of problem solved in Chapter 6 when the unknowns were the aerodynamic model parameters. Model equations (10.14) are simplified by assuming that the translational accelerations and angular rates are measured without random error (no process noise) and that the output measurement noise matrix is time invariant. The resulting equations are given above as Eqs. (10.17) and (10.15), which can be written in concise notation as

$$\dot{\mathbf{x}}(t) = \mathbf{f}[\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\theta}] \quad (10.22a)$$

$$\mathbf{y}(t) = \mathbf{h}[\mathbf{x}(t)] \quad (10.22b)$$

$$\mathbf{z}(i) = \mathbf{h}[\mathbf{x}(i)] + \mathbf{v}(i) \quad i = 1, 2, \dots, N \quad (10.22c)$$

$$E[\mathbf{v}(i)] = \mathbf{0} \quad E[\mathbf{v}(i)\mathbf{v}^T(j)] = \mathbf{R}\delta_{ij} \quad (10.22d)$$

## DATA COMPATIBILITY

347

where

$$\mathbf{x} = [u \ v \ w \ \phi \ \theta \ \psi \ h]^T \quad (10.22e)$$

$$\mathbf{u} = [a_x \ a_y \ a_z \ p \ q \ r]^T \quad (10.22f)$$

$$\mathbf{y} = [V \ \beta \ \alpha \ \phi \ \theta \ \psi \ h]^T \quad (10.22g)$$

$$\boldsymbol{\theta} = [\mathbf{b}^T \ \boldsymbol{\lambda}^T]^T \quad (10.22h)$$

and the symbols  $\mathbf{b}$  and  $\boldsymbol{\lambda}$  denote column vectors containing the bias and scale factor error parameters.

State estimation is reduced to integration of the deterministic dynamic equations (10.22a), which represent Eqs. (10.17a) and (10.17b). The kinematic differential equations are nonlinear, but this does not require any modification of the output-error method, which can compute output sensitivities from central finite differences applied to numerical solutions of the differential equations. The cost function formulation and optimization method for the parameter estimation are identical to what has been described in Chapter 6 for aerodynamic parameter estimation using the output-error method. In Eqs. (10.22), the model is formulated specifically for data compatibility analysis, and the model parameters characterize instrumentation errors instead of aerodynamic dependencies, but otherwise the estimation problem is conceptually the same.

Initial state values can be estimated from smoothed initial measurements (see Chapter 11), or the initial states can be treated as unknown parameters to be estimated, together with the parameters for instrumentation bias errors and scale factor errors.

Once the instrumentation systematic error parameters are estimated, the measured data are corrected by inverting Eq. (10.13), neglecting the random noise,

$$\hat{y}(i) = [z(i) - \hat{b}] / (1 + \hat{\lambda}) \quad i = 1, 2, \dots, N \quad (10.23)$$

For maneuvers at high angles of attack, it is often advantageous to reformulate the air data measurement equations (10.15a)–(10.15c) as follows:

$$V_E(i) = (1 + \lambda_V) \left[ \sqrt{u^2(i) + v^2(i) + w^2(i)} - V_0 \right] + V_0 + b_V + v_V(i) \quad (10.24a)$$

$$\beta_{f_E}(i) = (1 + \lambda_{\beta_f}) \left\{ \tan^{-1} \left[ \frac{v(i)}{u(i)} \right] - \beta_{f_0} \right\} + \beta_{f_0} + b_{\beta_f} + v_{\beta_f}(i) \quad (10.24b)$$

$$\alpha_E(i) = (1 + \lambda_\alpha) \left\{ \tan^{-1} \left[ \frac{w(i)}{u(i)} \right] - \alpha_0 \right\} + \alpha_0 + b_\alpha + v_\alpha(i) \quad (10.24c)$$

In these equations, the values of  $V_0$ ,  $\beta_{f_0}$ , and  $\alpha_0$  are obtained from smoothed initial measurements. This reformulation of the measurement equations reduces correlation between the scale factor and bias error parameters, when initial values are nonzero. For example, for the airspeed measurement, if the measurement equation (10.15a) is used, then  $\lambda_V$  multiplies both the relatively large and

constant part of the airspeed,  $V_0$ , as well as the smaller variation due to the maneuver. Because the bias error  $b_V$  is usually small, and  $V_0$  is relatively large, small changes in  $\lambda_V$  can approximately account for the bias error in the airspeed measurement with little effect on the model fit related to variations in the airspeed from the maneuver. However, the role of accounting for bias error in airspeed belongs to the bias parameter  $b_V$ . The result is a high correlation between  $\lambda_V$  and  $b_V$ , which is detrimental to parameter estimation results. Reformulating the measurement equation as in Eq. (10.24a) separates the roles of the scale factor and bias error parameters, because  $\lambda_V$  only multiplies the variation in airspeed and not the part represented by  $V_0$ .

Note that the role of the scale factor error parameter is different for Eq. (10.15), compared with Eq. (10.24), so the estimated values will be different. The same issue shows up whenever an initial measured value is not close to zero, which happens for many measured quantities at high angles of attack, for example.

More information about the use of the output-error method in data compatibility analysis can be found in Refs. 12–14.

## 10.6 Summary and Concluding Remarks

Successful modeling requires accurate measured data. This chapter is concerned with methods for improving the accuracy of measured data by removing known and estimated systematic errors from the raw measurements.

Every sensor has a static calibration that is done either in the laboratory or by the manufacturer. In addition, for sensors that measure aircraft responses, there are errors due to the position of the sensor on the aircraft. Corrections for both of these types of systematic instrumentation errors are relatively straightforward. Equations specific to position corrections for aircraft response sensors are included in this chapter.

Most of the chapter was concerned with the kinematic relationships among measured aircraft response variables and how these relationships can be used to improve the accuracy of the measured flight data. Typical aircraft response measurements can be checked for compatibility using data from dynamic maneuvers, because there are kinematic relationships among these quantities that would be satisfied if the data were perfectly accurate. The kinematic relationships were developed in this chapter, and it was shown how a subset of the measured aircraft responses can be used with the kinematic equations to reconstruct other aircraft response variables.

If a sensor error model with unknown instrumentation error parameters is introduced, then the kinematic equations can be used in conjunction with a nonlinear optimizer to estimate the most likely values of the instrumentation error parameters. The output-error method of Chapter 6 was shown to apply directly to the solution of this problem. Other methods for data compatibility analysis based on nonlinear filtering and smoothing were also introduced. Once the systematic instrumentation errors are successfully estimated, the measured data can be corrected using the identified instrumentation error model, with the result that the final set of data is more accurate and satisfies the kinematic equations as nearly as possible.

Estimating and removing systematic instrumentation errors is important, because if these errors are not removed, then the accuracy of aerodynamic model parameter estimates can be degraded. A good example is if a scale factor error exists for the angle of attack sensor, but the error is ignored and the data are not corrected, then the scale factor error will pollute the estimates of angle of attack stability derivatives, such as  $C_{m_\alpha}$ .

## References

- <sup>1</sup>Haering, E. A., Jr., "Airdata Measurement and Calibration," NASA TM 104316, 1995.
- <sup>2</sup>Gainer, T. G., and Hoffman, S., "Summary of Transformation Equations and Equations of Motion Used in Free-Flight and Wind-Tunnel Data Reduction Analysis," NASA SP-3070, 1972.
- <sup>3</sup>Klein, V., "Evaluation of the Basic Performance Characteristics of an Instrumentation System," Cranfield Institute of Technology, Cranfield Rep. Aero. No. 22, Cranfield, UK, 1973.
- <sup>4</sup>Gelb, A. (ed.), *Applied Optimal Estimation*, MIT Press, Cambridge, MA, 1974.
- <sup>5</sup>Chen, R. T. N., and Eulrich, B. J., "Parameter and Model Identification of Nonlinear System Using a Suboptimal Fixed-Point Smoothing Technique," *Proceedings of the Joint Automatic Control Conference*, August 1971.
- <sup>6</sup>Klein, V., and Schiess, J. R., "Compatibility Check of Measured Aircraft Responses using Kinematic Equations and Extended Kalman Filter," NASA TN D-8514, 1977.
- <sup>7</sup>Mulder, J. A., "Estimation of Aircraft State in Non-Steady Flight," *Methods for Aircraft State and Parameter Identification*, AGARD-CP-172, Paper 19, 1975.
- <sup>8</sup>Jonkers, H. L., "Application of the Kalman Filter to Flight Path Reconstruction from Flight Test Data Including Estimation of Instrumental Bias Error Corrections," Delft Univ. of Technology, Rep. VTH, Delft, The Netherlands, 1976.
- <sup>9</sup>Leach, B., and Hui, K., "In-Flight Technique For Calibrating Air Data Systems Using Kalman Filtering and Smoothing," AIAA Paper 2001-4260, 2001.
- <sup>10</sup>Bach, R. E., Jr., "State Estimation Applications in Aircraft Flight Data Analysis," NASA RP 1252, 1991.
- <sup>11</sup>Wingrove, R. C., "Quasi-Linearization Technique for Estimating Aircraft States from Flight Data," *Journal of Aircraft*, Vol. 10, No. 5, 1973, pp. 303–307.
- <sup>12</sup>Keskar, D. A., and Klein, V., "Determination of Instrumentation Errors from Measured Data Using Maximum Likelihood Method," AIAA Paper 80-1602, 1980.
- <sup>13</sup>Klein, V., and Morgan, D. R., "Estimation of Bias Errors in Measured Airplane Responses Using Maximum Likelihood Method," NASA TM 89059, 1987.
- <sup>14</sup>Morelli, E. A., "Optimal Input Design for Aircraft Instrumentation Systematic Error Estimation," AIAA Paper 91-2850, 1991.

This page intentionally left blank

# 11

## Data Analysis

This chapter describes some common operations performed on measured data that are either part of the system identification process or preparatory steps. Most of the techniques presented here have very general applicability, but were selected because they have been found useful for tasks related to aircraft system identification. The selected techniques certainly do not represent the full spectrum of techniques available. Ref. 1 is an excellent resource for techniques of the type described here.

Sections 11.1 and 11.2 discuss filtering and smoothing, which are concerned with separating deterministic signal from random noise for measured time series. This is an important operation in system identification, because the general aim is to identify a mathematical model based on the deterministic parts of the measurements. Section 11.3 explains the link between the smoothing methods and interpolation, which provides a means to reconstruct missing or bad data points. Specialized methods for smoothed numerical differentiation are described next. These methods find use in equation-error modeling and sensor position error correction. Practical methods for computing the finite Fourier transform and power spectral estimates, which form the basis for the frequency-domain methods of Chapter 7, are also discussed in detail. Finally, the chapter concludes with a discussion of methods for comparing signal waveforms and visualizing aircraft motion during a flight-test maneuver using animated computer graphics.

### 11.1 Filtering

Filters can be implemented in hardware using electronic components, or by a computer implementing analog or discrete transfer functions. Filters behave like dynamic systems, with associated amplitude and phase changes as a function of input frequency. Consequently, a filter represents additional system dynamics introduced between the physical system and the signal used to identify dynamic models. Since the objective of system identification is often to characterize the amplitude and phase angle changes from input to output for the physical system, filtering can distort the modeling results by modifying the amplitude and phase of the measured signals.

The danger occurs when the break frequency of the filter is near the frequencies of interest, because amplitude and phase angle changes due to the filter are greatest near that frequency. This can be seen from the frequency response plot for a typical low-pass filter shown in Fig. 9.4. The antialiasing filter, with break frequency set according to Eq. (9.6), does not introduce significant amplitude or phase angle modification to the frequencies of interest, because the frequencies of interest are much lower than the break frequency of the antialiasing filter.

Once the data are sampled, digital filtering can be applied using the same principles discussed in Chapter 9 for antialiasing filters. Some analysts apply identical digital filtering to all the measured signals. This approach is fine as long as only linear relationships are being investigated, but is not acceptable for nonlinear modeling. The reason is that inputs to a linear dynamic system produce outputs at the same frequencies, with possibly altered amplitude and phase angle, as described by the transfer function of the system. For a nonlinear system, this is not true, so amplitude changes and phase shifts from the filtering are not applied uniformly when the same filtering is used for all measured signals. Since the signal amplitudes and phase angle relationships are at the core of system identification, it is preferable to separate signal from noise using zero-phase-shift filtering or smoothing, with very low amplitude distortion.

All filtering methods use only current and past data, as discussed in Chapter 4. Although this approach must be taken for real-time applications, postflight data processing can use subsequent data as well. The associated additional information gives significant advantages to smoothing techniques. Furthermore, filtering methods assume that the analyst knows what the frequency range for the filtering should be. In system identification applications, it is not always clear what this frequency band should be, and a good choice for the frequency band is often not the same for different measured time series. This is important, because the modeling will be adversely affected by either admitting noise into the signals or by discarding parts of the deterministic signals. A global smoothing method based on Fourier analysis, described in the next section, can provide information on appropriate cutoff frequencies to be used for system identification.

For postflight analysis, an ordinary digital filter can be run forward and backward in time, which effectively cancels the phase effects of the filter. However, smoothing methods generally give better results.

## 11.2 Smoothing

Smoothing uses data points both before and after the data point to be smoothed. Since subsequent data are used, smoothing can only be applied postflight. However, that situation constitutes the vast majority of cases for system identification applied to aircraft.

### 11.2.1 Frequency-Domain Filter Implemented in the Time Domain

A filter defined in the frequency domain can be implemented in terms of smoothing weights in the time domain, without introducing phase shifts in the

## DATA ANALYSIS

353

smoothed data. Graham<sup>2</sup> describes a procedure for doing this, which is summarized here.

A low-pass filter can be defined in the frequency domain according to

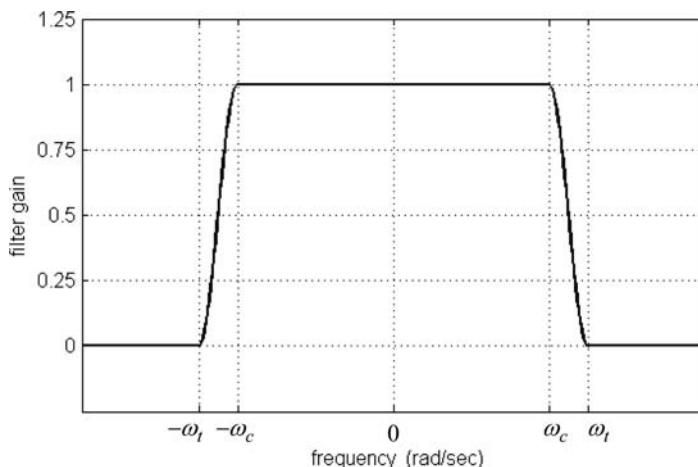
$$\tilde{g}(\omega) = \begin{cases} 1 & |\omega| \leq \omega_c \\ \frac{1}{2} \left\{ \cos \left[ \frac{\pi(\omega + \omega_c)}{(\omega_t - \omega_c)} \right] + 1 \right\} & -\omega_t < \omega < -\omega_c \\ \frac{1}{2} \left\{ \cos \left[ \frac{\pi(\omega - \omega_c)}{(\omega_t - \omega_c)} \right] + 1 \right\} & \omega_c < \omega < \omega_t \\ 0 & |\omega| \geq \omega_t \end{cases} \quad (11.1)$$

where  $\omega_c$  is the selected cutoff frequency for the low-pass filter, and  $\omega_t$  is the end of the frequency roll-off (see Fig. 11.1). Note that this filter definition is symmetric about the zero frequency axis. Using this definition of the low-pass filter, the inverse Fourier transform

$$g(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \tilde{g}(\omega) e^{j\omega t} d\omega \quad (11.2)$$

can be done analytically, resulting in

$$g(t) = \frac{\pi}{2t} \left[ \frac{\sin \omega_t t + \sin \omega_c t}{\pi^2 - (\omega_t - \omega_c)^2 t^2} \right] \quad (11.3)$$



**Fig. 11.1 Low-pass filter gain.**

or, in discrete-time form, for  $t_k = k\Delta t$ ,

$$g_k = \frac{\pi}{2k\Delta t} \left[ \frac{\sin \omega_t k\Delta t + \sin \omega_c k\Delta t}{\pi^2 - (\omega_t - \omega_c)^2(k\Delta t)^2} \right] \quad (11.4)$$

The smoothing is brought about by convolving the measured data with the discrete-time smoothing weights  $g_k$ , which is equivalent to a multiplication by  $\tilde{g}(\omega)$  in the frequency domain (see Appendix A). The values of  $g_k$  are normalized so that

$$\sum_{k=-N_s}^{N_s} g_k = 1 \quad (11.5)$$

where  $N_s$  is the number of smoothing weights used on either side of the measured data point to be smoothed. Figure 11.2 shows an example of the smoothing weights  $g_k$  corresponding to the filter definition in Fig. 11.1. The smoothed value of the  $i$ th data point is computed by applying the smoothing weights shown in Fig. 11.2 to the measured data, with the weighting centered at the  $i$ th point. For  $N$  discrete measured values  $z(i)$ , the smoothed values  $z_s(i)$  are computed from

$$z_s(i) = \sum_{k=-N_s}^{N_s} g_k z(i+k) \quad i = 1, 2, \dots, N \quad (11.6)$$

Near the endpoints, measured values on the interior must be used twice. Since the smoothing weights are symmetric about the smoothed point, this can be done by augmenting the measured data with reflections about the endpoints.

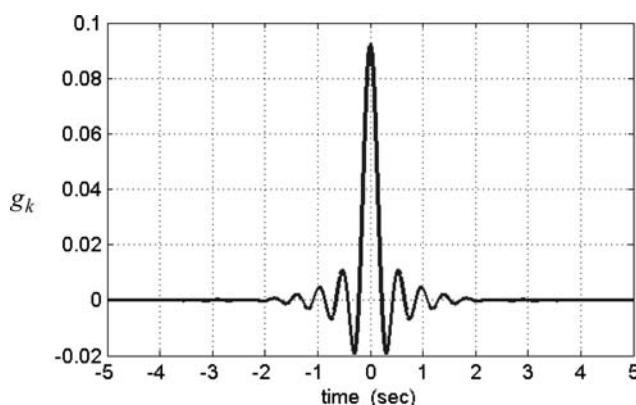


Fig. 11.2 Time-domain fixed smoothing weights.

## DATA ANALYSIS

355

Then the convolution can be done in a straightforward manner for all points in the measurement record, using the augmented data.

To use this method, the frequency cutoff  $\omega_c$  must be chosen. Then there is a tradeoff between values for  $\omega_t$  and  $N_s$ , with larger  $N_s$  giving a steeper roll-off in the frequency domain, and vice versa. The relationship among these quantities that is consistent with achieving less than 0.5% gain error in the pass band is

$$N_s \geq \frac{2\pi^2}{\Delta t(\omega_t - \omega_c)} \quad (11.7)$$

In practice, it is easiest to choose  $N_s$  fairly large for a given  $\omega_c$ , e.g., set  $N_s$  equal to half the original data length,  $N_s = N/2$ , then compute  $\omega_t$  from Eq. (11.7).

The actual frequency characteristic realized by the implemented smoothing weights can be calculated from

$$\hat{\tilde{g}}(\omega) = \sum_{k=-N_s}^{N_s} g_k \cos(\omega k \Delta t) \quad (11.8)$$

since  $g_k$  is an even function that is nonzero only for  $-N_s \leq k \leq N_s$ .

### 11.2.2 Local Smoothing in the Time Domain

Another method for data smoothing is local smoothing in the time domain. This involves fitting a local polynomial in time to measured data points near the point to be smoothed. Using a second-order polynomial model implies that the local data points lie on a parabola, except for random errors. This is equivalent to assuming the second derivative is a constant over the same time period. Defining time relative to the data point to be smoothed, the local model to be identified can be written as

$$y = a_0 + a_1 t + \frac{1}{2} a_2 t^2 \quad (11.9)$$

If  $z(i)$  denotes the  $i$ th measured data point sampled at time  $i\Delta t$ , then the equations used for the local fit to the data are

$$\begin{aligned} a_0 + a_1(-2\Delta t) + a_2(-2\Delta t)^2 &= z(i-2) \\ a_0 + a_1(-\Delta t) + a_2(-\Delta t)^2 &= z(i-1) \\ a_0 &= z(i) \\ a_0 + a_1(\Delta t) + a_2(\Delta t)^2 &= z(i+1) \\ a_0 + a_1(2\Delta t) + a_2(2\Delta t)^2 &= z(i+2) \end{aligned} \quad (11.10)$$

There are five equations for three unknowns, so the least-squares solution of Chapter 5 applies. Once the solution is obtained, the smoothed value for  $z(i)$  is simply equal to the estimate of  $a_0$ , since the time was defined to be zero at

the  $i$ th data point for the local modeling. The equations can be written in matrix form as

$$\begin{bmatrix} 1 & -2\Delta t & 4(\Delta t)^2 \\ 1 & -\Delta t & (\Delta t)^2 \\ 1 & 0 & 0 \\ 1 & \Delta t & (\Delta t)^2 \\ 1 & 2\Delta t & 4(\Delta t)^2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} z(i-2) \\ z(i-1) \\ z(i) \\ z(i+1) \\ z(i+2) \end{bmatrix} \quad (11.11)$$

The resulting normal equations are

$$\begin{aligned} 5a_0 + 10(\Delta t)^2 a_2 &= \sum_{k=i-2}^{i+2} z(k) \\ 10\Delta t a_1 &= \sum_{k=i-2}^{i+2} k z(k) \\ 10a_0 + 34(\Delta t)^2 a_2 &= \sum_{k=i-2}^{i+2} k^2 z(k) \end{aligned} \quad (11.12)$$

so the least-squares estimate of  $a_0$  is

$$\hat{y}_i = \hat{a}_0 = \frac{34}{70} \sum_{k=i-2}^{i+2} z(k) + \frac{1}{7} \sum_{k=i-2}^{i+2} k^2 z(k) \quad (11.13)$$

or

$$\begin{aligned} \hat{y}_i = z_s(i) = \hat{a}_0 &= \frac{1}{70} [-6z(i-2) + 24z(i-1) \\ &\quad + 34z(i) + 24z(i+1) - 6z(i+2)] \end{aligned} \quad (11.14)$$

The preceding expression can be used for local smoothing in the time domain. When  $z(i)$  is at or near an endpoint, more neighboring points on the interior are substituted to supply data for the local polynomial fit, and the calculation of the smoothed value is adjusted accordingly.

The aforementioned procedure can be used to implement local smoothing for different numbers of neighboring points  $k$  and various approximating polynomial orders  $n$ . Since the method involves repeated local smoothing solutions, it is difficult to determine an exact cutoff frequency for smoothing over the complete measured time series. In general, larger values of  $k$  for constant  $n$  give lower cutoff frequencies. Increasing  $n$  improves the capability of the local model to match higher frequency variations, which moves the cutoff frequency higher.

For  $n = 2$ , the cutoff frequency in Hz for this smoother can be approximated by

$$f_c = \frac{1}{2k\Delta t} \quad (11.15)$$

For  $\Delta t = 0.02$  s, using  $k = 10$  and  $n = 2$  gives good local smoothing results with a cutoff frequency near 2.5 Hz.

### 11.2.3 Global Smoothing in the Frequency Domain

Both of the smoothing techniques described earlier compute smoothed values based on local measured data in the time domain. A different approach can be formulated by treating the smoothing problem in a global sense. This approach uses Fourier analysis, and is discussed in Refs. 3 and 4. The latter reference is the source of the material presented here.

Consider a noisy measured scalar  $z(t)$  sampled at constant intervals  $\Delta t$ ,

$$z(i) = z(i\Delta t) \quad i = 0, 1, 2, \dots, N - 1 \quad (11.16)$$

The goal is to separate signal and noise, or equivalently to find  $y(i)$  and  $v(i)$  such that

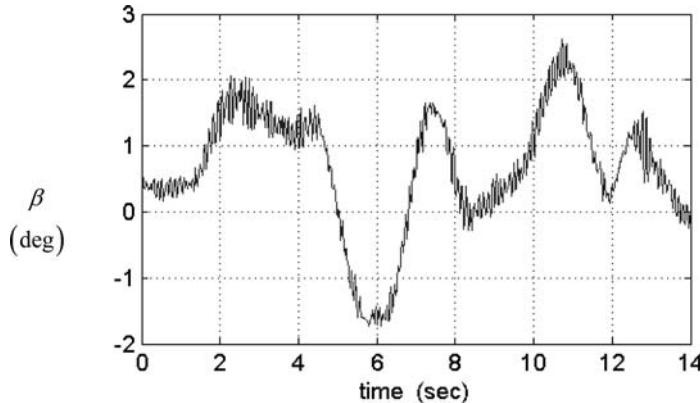
$$z(i) = y(i) + v(i) \quad i = 0, 1, 2, \dots, N - 1 \quad (11.17)$$

The strategy for determining the noise sequence  $v(i)$ ,  $i = 0, 1, 2, \dots, N - 1$ , is to find an accurate description for the signal in the frequency domain, invert the Fourier transform to get  $y(i)$ , then find  $v(i)$  from Eq. (11.17) in the form

$$v(i) = z(i) - y(i) \quad i = 0, 1, 2, \dots, N - 1 \quad (11.18)$$

Fourier series expansion implicitly assumes that the time series under consideration is periodic. For most measured time series from flight tests, making such an assumption implies discontinuities in the amplitude and first time derivative at the endpoints. Figure 11.3 shows a typical measured time series for sideslip angle from a flight-test maneuver. Lanczos<sup>3</sup> shows that Fourier series for functions with these discontinuities have much slower convergence than the Fourier series for a function with no discontinuities in either the amplitude or first derivative. In the former case, the magnitudes of the Fourier coefficients decrease asymptotically with  $k^{-1}$ , where  $k$  is the number of terms in the Fourier series expansion, whereas in the latter case, the asymptotic decrease goes according to  $k^{-3}$ . This seems reasonable, because the sinusoids in the Fourier series have no discontinuities in amplitude or first derivative anywhere, and therefore would be expected to have difficulty representing a time series with those discontinuities.

It is possible to remove the endpoint discontinuities from any arbitrary time series, and thereby achieve the more abrupt decrease in the magnitude of the Fourier coefficients, which corresponds to a faster convergence of the Fourier



**Fig. 11.3 Sideslip angle measured time series.**

series expansion. The importance of the higher rate of convergence will become clear in the subsequent discussion. To remove the discontinuities, subtract a linear trend from the original measured time history to make the amplitudes at the endpoints equal to zero, and then reflect the result about the origin to remove the slope discontinuities at the endpoints. Define the new time history as  $g(i)$ , with  $g(-N + 1) = g(0) = g(N - 1) = 0$ . The values of  $g(i)$  are computed from

$$g(i) = z(i) - z(0) - i \left[ \frac{z(N - 1) - z(0)}{N - 1} \right] \quad i = 0, 1, 2, \dots, N - 1 \quad (11.19a)$$

$$g(-i) = -g(i) \quad i = 1, 2, \dots, N - 1 \quad (11.19b)$$

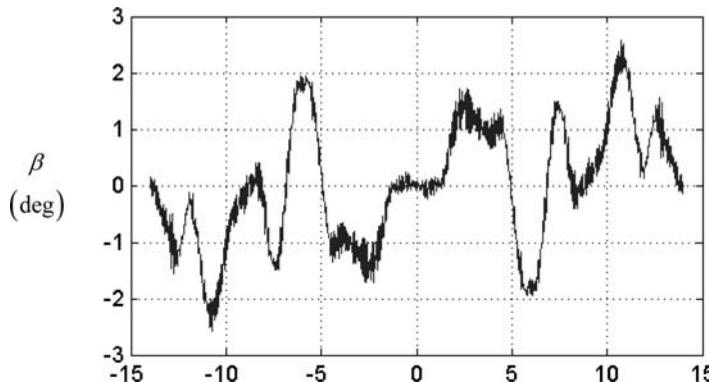
Figure 11.4 shows the result of performing these operations on the time series of Fig. 11.3. The vector

$$\mathbf{g} = [g(-N + 1) \quad g(-N + 2) \quad \dots \quad g(-1) \quad g(0) \quad g(1) \quad \dots \quad g(N - 1)]^T \quad (11.20)$$

is an odd function of time, and therefore can be expanded using a Fourier sine series:

$$\hat{g}(i) = \sum_{k=1}^{N-1} b(k) \sin \left[ k\pi \left( \frac{i}{N-1} \right) \right] \quad i = 0, 1, 2, \dots, N - 1 \quad (11.21)$$

where  $\hat{g}(i)$  denotes the approximation to  $g(i)$  using the Fourier sine series, and the  $b(k)$  are Fourier sine series coefficients. The summation over frequency index  $k$  omits  $k = 0$  (zero frequency), since this is a pure sine series for an odd



**Fig. 11.4 Measured sideslip angle with endpoint discontinuities removed.**

function. Only positive values of  $i$  are included in Eq. (11.21), since those values correspond to the original function.

The abrupt  $k^{-3}$  decrease in the magnitudes of the Fourier sine series coefficients  $b(k)$  can now be expected because the discontinuities in the amplitude and first time derivative at the endpoints have been removed.

The Fourier sine series coefficients for  $\mathbf{g}$  are computed as<sup>3</sup>

$$b(k) = \frac{2}{N-1} \sum_{i=1}^{N-2} g(i) \sin\left[k\pi\left(\frac{i}{N-1}\right)\right] \quad k = 1, 2, \dots, N-1 \quad (11.22)$$

where the index  $i$  runs from 1 to  $(N-2)$ , because  $g(0)$  and  $g(N-1)$  are zero. The upper limit for  $k$  is  $(N-1)$ , which corresponds to the Nyquist frequency. From Eq. (11.21), the period of the  $k$ th Fourier sine series term  $T_k$  is given by

$$T_k = \frac{2(N-1)\Delta t}{k} \quad (11.23)$$

so the  $k$ th frequency  $f_k$  is related to the frequency index  $k$  by

$$f_k = \frac{k}{2(N-1)\Delta t} \quad (11.24)$$

To effectively smooth the data, it is necessary that the Nyquist frequency  $f_N = 1/(2\Delta t)$  be much higher than the highest frequency of the deterministic components in the signal. This consideration is rarely a problem with modern data acquisition systems on aircraft, when proper attention is paid to analog antialiasing filtering before sampling (cf. Chapter 9).

Using Eqs. (11.19) to construct  $\mathbf{g}$  guarantees that the endpoint discontinuities are in the second time derivative of the original time series. Convergence of the Fourier sine series approximation would be further accelerated if endpoint

discontinuities in higher derivatives were also removed. However, since the measured time series are from a dynamic system where application of forces and moments can produce discontinuities in second-order (and higher) time derivatives, any attempt to remove additional endpoint discontinuities runs the risk of inadvertently modifying the signal. Therefore, removing amplitude and first-order time derivative endpoint discontinuities is accepted as the best that can be done without corrupting the desired signal.

The Fourier sine series expansion in Eq. (11.22) contains all spectral components that can be computed from the given finite time series, including both signal and noise. The Fourier series for a coherent signal is fundamentally different than the Fourier series for noise. As mentioned earlier, a coherent signal with discontinuities only in the second-order time derivative or higher at the endpoints has Fourier series coefficient amplitudes that rapidly decrease asymptotically to zero with increasing  $k$ , i.e.,  $|b(k)|$  is proportional to  $k^{-3}$ . On the other hand, since noise is incoherent and theoretically has constant power over the frequency range up to the Nyquist frequency, its Fourier series coefficients do not decrease asymptotically to zero, but instead have a relatively small and constant magnitude for all frequencies. This reflects the fact that the Fourier series expansion for an incoherent time series is divergent, due mainly to the inconsistent phase-amplitude relationships. In the case of a Fourier sine series, the Fourier series coefficients of the noise appear as a relatively constant amplitude oscillation about zero, representing random phase change in the spectral components. The abrupt decrease in the magnitudes of the Fourier sine series coefficients for the coherent signal contrasts sharply with the relatively constant magnitude of the Fourier sine series coefficients for the noise. The use of Eqs. (11.19) to remove endpoint discontinuities before performing the Fourier transform was done specifically to enhance this contrast.

The contrast is so stark, that it is often easy to find the separation point visually. Figure 11.5 shows the situation for the measured time series of Fig. 11.3. For low-pass smoothing typical of aircraft applications, a simple method for implementing the smoothing would be to visually determine the highest frequency in the cubic roll-off of the Fourier sine series coefficients  $b(k)$ , set all the sine series coefficients at higher frequencies equal to zero, and then compute the smoothed time history using the inverse Fourier sine transform,

$$\hat{g}_s(i) = \sum_{k=1}^{k_{\max}} b(k) \sin\left[k\pi\left(\frac{i}{N-1}\right)\right] \quad i = 0, 1, 2, \dots, N-1 \quad (11.25)$$

where  $k_{\max}$  is the maximum frequency index determined visually. The end of the cubic roll-off for the coherent signal is defined as the frequency where the Fourier coefficient magnitudes for the coherent signal reach the noise floor, defined by the relatively constant magnitude of the Fourier coefficients at high frequencies. Figure 11.6 shows the simple frequency domain filter that results, based on the frequency domain data in Fig. 11.5.

This simple form of the filter in the frequency domain can be improved upon, in a way that provides a hedge against an inaccurate visual selection of the

## DATA ANALYSIS

361

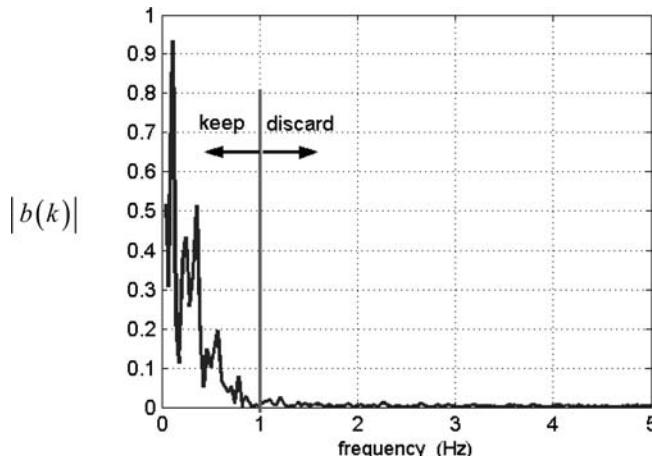


Fig. 11.5 Fourier sine series coefficients for measured sideslip angle with endpoint discontinuities removed.

frequency cutoff. Consider the Fourier transform of the measured time series, which is composed of signal plus noise,

$$\tilde{z}(f) = \tilde{y}(f) + \tilde{v}(f) \quad (11.26)$$

The goal is to design a frequency domain filter that is optimal in the sense of minimizing the squared difference between the true signal  $\tilde{y}(f)$  and the estimated

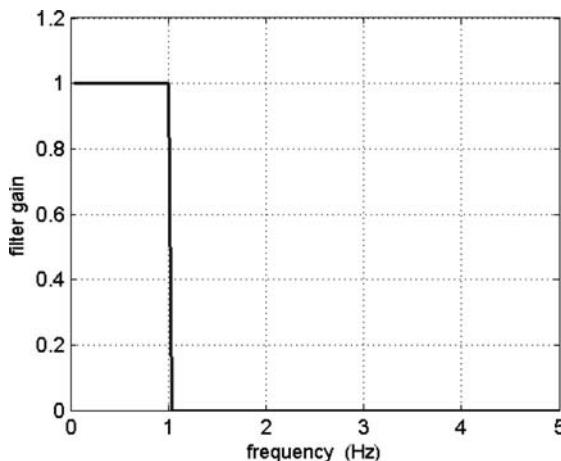


Fig. 11.6 Simple frequency-domain smoothing filter for  $|b(k)|$  in Fig. 11.5.

signal  $\hat{\tilde{y}}(f)$  over the entire frequency range up to the Nyquist frequency, i.e., the integral

$$\int_0^{f_N} [\tilde{y}(f) - \hat{\tilde{y}}(f)]^2 df \quad (11.27)$$

is to be minimized. The estimated signal in the frequency domain will be obtained by multiplying a filter  $\Phi(f)$  with  $\tilde{z}(f)$ ,

$$\hat{\tilde{y}}(f) = \Phi(f)\tilde{z}(f) = \Phi(f)[\tilde{y}(f) + \tilde{v}(f)] \quad (11.28)$$

Substituting for  $\hat{\tilde{y}}(f)$  from Eq. (11.28) and recognizing that  $\tilde{y}(f)\tilde{v}(f)$  integrated over all frequencies will be approximately zero due to the incoherence of the noise, Eq. (11.27) becomes

$$\int_0^{f_N} \{ \tilde{y}^2(f) - 2\Phi(f)\tilde{y}^2(f) + \Phi^2(f)[\tilde{y}^2(f) + \tilde{v}^2(f)] \} df \quad (11.29)$$

Taking the derivative of the integrand in Eq. (11.29) with respect to  $\Phi(f)$ , setting the result equal to zero, and solving for  $\Phi(f)$  gives

$$\Phi(f) = \frac{\tilde{y}^2(f)}{\tilde{y}^2(f) + \tilde{v}^2(f)} \quad 0 \leq f \leq f_N \quad (11.30)$$

or, in terms of the discrete frequency index  $k$ ,

$$\Phi(k) = \frac{\tilde{y}^2(k)}{\tilde{y}^2(k) + \tilde{v}^2(k)} \quad 0 \leq k \leq N - 1 \quad (11.31)$$

Equation (11.31) gives the form of the optimal filter in the frequency domain, also called the Wiener filter.<sup>1</sup> For the problem at hand, the Wiener filter can be constructed by assuming that the Fourier coefficients for the signal are those below the frequency cutoff determined visually, and the constant noise level can be estimated using an average of the Fourier coefficient magnitudes for frequencies higher than the selected cutoff. Figure 11.7 shows the Wiener filter constructed in this manner, using Eq. (11.31) and the same frequency cutoff determined for Fig. 11.6, with frequency domain data from Fig. 11.5. The optimal filter from Eq. (11.31) then multiplies the  $b(k)$  in the inverse Fourier transform, so that the smoothed signal (with endpoint discontinuities still removed) is computed from

$$\hat{g}_s(i) = \sum_{k=1}^{N-1} \Phi(k)b(k) \sin\left[k\pi\left(\frac{i}{N-1}\right)\right] \quad i = 0, 1, 2, \dots, N-1 \quad (11.32)$$

## DATA ANALYSIS

363

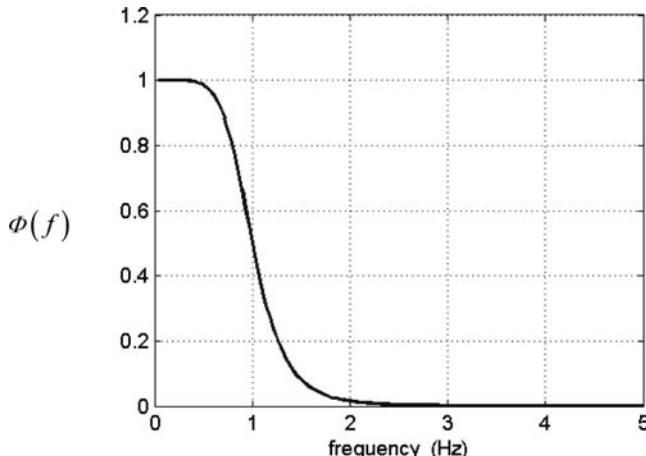


Fig. 11.7 Wiener frequency-domain smoothing filter for  $|b(k)|$  in Fig. 11.5.

The summation over the frequency index  $k$  can be truncated when values of  $\Phi(k)b(k)$  approach zero.

The Wiener filter computed from Eq. (11.31) is near unity at low frequencies, passing the Fourier sine series components for the coherent signal, then transitions smoothly to near zero at high frequencies, removing Fourier sine series components associated with the noise. The optimality of the Wiener filter gives some room for error in the visual selection of the cutoff frequency. Furthermore, the magnitudes of the  $b(k)$  values are small in the region of the cutoff frequency, so including or excluding a few components improperly will have minimal adverse effect on the final smoothed time series.

Naturally, some components of the noise lie in the low-frequency range, but there is no way to distinguish this noise from the signal, which also resides in the low-frequency range. Typically, the large majority of the noise power resides at high frequency relative to the frequencies of the signal, and this noise can be removed very effectively. When the signal-to-noise ratio is high, the power of the noise relative to that of the signal is small in an overall sense, but this situation is improved by the fact that the noise power is spread over a wide frequency range, whereas the signal power resides in a smaller frequency range. Once the high-frequency noise has been removed, the remaining noise components in the frequency range of the signal have very low power, and therefore have little impact on the smoothed time history.

As a final step, the linear trend removed from the original time history using Eqs. (11.19) must be restored to the smoothed values  $\hat{g}_s(i)$ ,  $i = 0, 1, 2, \dots, N - 1$ , from Eq. (11.32). The final smoothed time history is computed from

$$\hat{y}(i) = z_s(i) = \hat{g}_s(i) + z(0) + i \left[ \frac{z(N-1) - z(0)}{N-1} \right] \quad i = 0, 1, 2, \dots, N-1 \quad (11.33)$$

In the preceding development, it is clear that the endpoints of the measured time history,  $z(0)$  and  $z(N - 1)$ , are completely excluded from all smoothing operations. For very noisy data, this can produce significant error in the smoothed time series. To correct this, the endpoints can be smoothed locally using one of the methods described previously. The cutoff frequency for the local endpoint smoothing should be chosen at roughly the same value selected for the global Fourier smoothing, and the endpoints should be smoothed before the global Fourier smoothing is begun. Using this procedure, all the data points are smoothed using a consistent cutoff frequency. Figure 11.8 shows the smoothed coherent signal found by applying the global Fourier smoother to the measured data shown in Fig. 11.3.

The noise sequence  $v(i)$ ,  $i = 0, 1, 2, \dots, N - 1$ , can be obtained from Eq. (11.18) using  $\hat{y}(i)$  computed from Eqs. (11.32) and (11.33). The noise sequence from the global Fourier smoother has only wideband random components; i.e., the noise is not colored by deterministic modeling error. In practice, this noise can be assumed Gaussian and stationary, so that noise characteristics can be estimated from

$$\bar{v} = \frac{1}{N} \sum_{i=0}^{N-1} v(i) \quad (11.34)$$

$$s_v^2 = \frac{1}{N-1} \sum_{i=0}^{N-1} [v(i) - \bar{v}]^2 \quad (11.35)$$

Since the noise sequence is composed of relatively wideband and high-frequency random components, the computed estimate for the mean value is close to zero. Figure 11.9 shows the estimated noise sequence associated with Figs. 11.3 and 11.8.

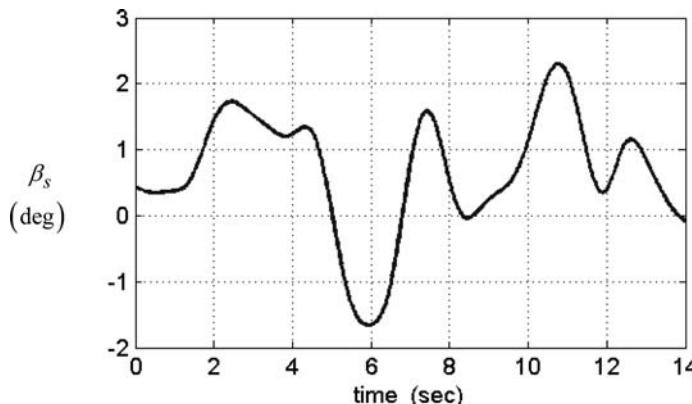


Fig. 11.8 Smoothed sideslip angle time series.

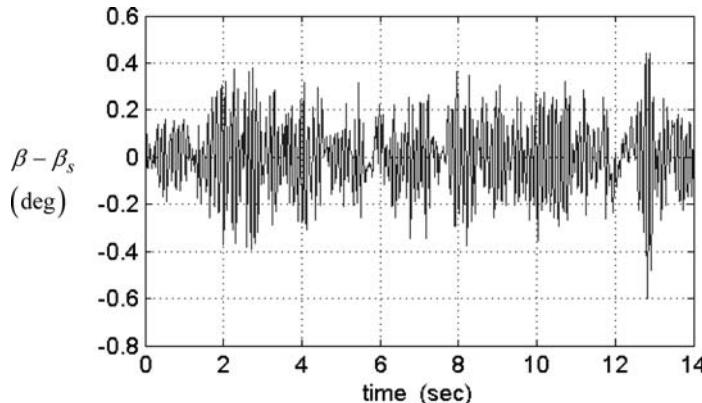


Fig. 11.9 Estimated noise sequence for the sideslip angle time series.

For an experiment with  $n_o$  measured outputs, the measured time series can be arranged in an  $N \times n_o$  matrix,

$$\mathbf{Z} = [z_1 \ z_2 \ \cdots \ z_{n_o}] \quad (11.36)$$

where  $z_j$ ,  $j = 1, 2, \dots, n_o$ , are  $N \times 1$  vectors of measured outputs. Similarly, define

$$\hat{\mathbf{Y}} = [\hat{y}_1 \ \hat{y}_2 \ \cdots \ \hat{y}_{n_o}] \quad (11.37)$$

where  $\hat{y}_j$ ,  $j = 1, 2, \dots, n_o$ , are  $N \times 1$  vectors of the respective smoothed signals. An estimate of the measurement noise covariance matrix  $\mathbf{R}$  can then be obtained from

$$\hat{\mathbf{R}} = \frac{(\mathbf{Z} - \hat{\mathbf{Y}})^T(\mathbf{Z} - \hat{\mathbf{Y}})}{N - 1} \quad (11.38)$$

Figure 11.10 shows the frequency response data of Fig. 11.5 for the entire frequency range, up to the Nyquist frequency at 25 Hz. The data in Fig. 11.10 illustrate the fact that the global smoothing technique can be used to identify other deterministic components in the data, in addition to the rigid-body deterministic components at low frequency. These other deterministic components show up with the same characteristic cubic decrease in the Fourier sine coefficients.

In Fig. 11.10, the Fourier coefficients near 9 and 11 Hz are from structural responses of the wingtip boom on which the sideslip angle sensor is mounted and from wing bending. The deterministic structural response can also be isolated and smoothed by simply discarding all the Fourier sine coefficients at frequencies above and below the relatively large components around 9–11 Hz, and then using the inverse Fourier transform, as before. Figure 11.11 shows the

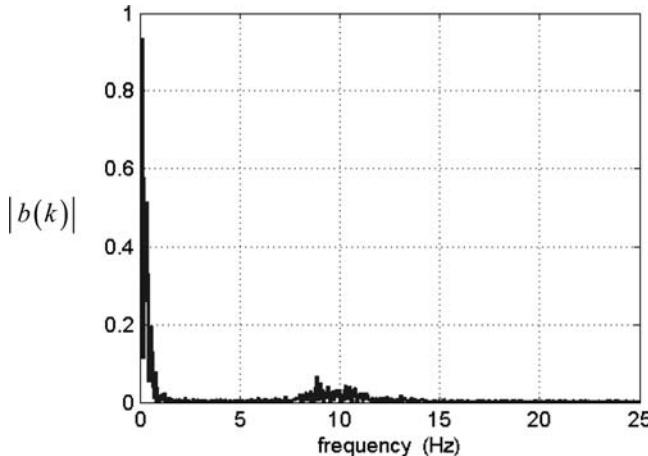


Fig. 11.10 Fourier sine series coefficients for measured sideslip angle with endpoint discontinuities removed for the full frequency range.

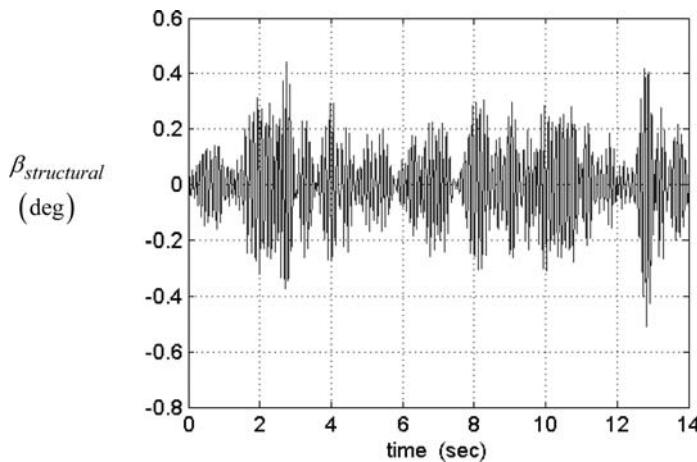


Fig. 11.11 Smoothed structural response from the sideslip angle time series.

deterministic structural response in the frequency range 8–12 Hz, extracted in this way from the sideslip angle measurements shown in Fig. 11.3.

### 11.3 Interpolation

Local interpolation in the time domain is very similar to local time-domain smoothing described in the last section. The difference is that the measurement for the center point in the local smoothing is now missing. Otherwise, the

## DATA ANALYSIS

367

principle of fitting a local polynomial to the nearby data is exactly the same. The interpolation is done by evaluating the local fitted polynomial at the location of the missing data point, which is normally the center location. This gives a locally smoothed value for the interpolated data point. The simplest form of this type of interpolation is the use of a linear function fitted to data points adjacent to the missing point.

The same principle can be used when interpolating data from a lower to a higher sampling rate. When the noise level is low, the simple approach of using linear interpolation between neighboring points can be used; however, a local polynomial model for the deterministic signal gives better results.

A more rigorous approach to generating values at a high sampling rate based on measurements at a lower sampling rate is given by the sampling theorem.<sup>1</sup> This theorem states that, for a sampled time series  $z(i)$ , where the underlying continuous function of time is bandwidth-limited so that all components lie at frequencies below the Nyquist frequency  $f_N = 1/(2\Delta t)$ , the value of the underlying continuous function can be reconstructed at any time using the measured samples, by the following expression:

$$z(t) = \sum_{k=-\infty}^{\infty} z(k) \frac{\sin\left[\pi\left(\frac{t}{\Delta t} - k\right)\right]}{\left[\pi\left(\frac{t}{\Delta t} - k\right)\right]} \quad (11.39)$$

Unfortunately, this expression requires an infinite amount of data. However, a useful approximation can be made using only the available measured data,  $z(i)$ ,  $i = 0, 1, 2, \dots, N - 1$ ,

$$z(t) \approx \sum_{k=0}^{N-1} z(k) \frac{\sin\left[\pi\left(\frac{t}{\Delta t} - k\right)\right]}{\left[\pi\left(\frac{t}{\Delta t} - k\right)\right]} \quad (11.40)$$

The interpolation results from this method are very good overall, but degrade slightly near the endpoints, because the interpolated points near the endpoints have one side with less measured data nearby. This approach is sensitive to noise, because there is no mechanism to distinguish between deterministic signal and noise in the interpolation.

The global Fourier smoothing technique described earlier can also be used for interpolation, by simply using a time vector with a higher sampling rate in the reconstruction of the time history via the inverse Fourier transform in Eq. (11.32). A time vector with a higher sampling rate can be implemented by increasing the value of  $N$  in Eq. (11.32). This approach has the advantage of smoothing the data in addition to the interpolation, resulting in a more accurate result for noisy measured data.

## 11.4 Numerical Differentiation

A common problem in data preparation for system identification purposes is the computation of a numerical time derivative based on noisy measurements.

For example, a numerical time derivative is required when computing values of nondimensional aerodynamic moment coefficients based on measured data, using the dynamic equations for rotational motion. This was discussed in conjunction with equation-error methods using linear regression in Chapter 5. Flight-test instrumentation on an aircraft often does not include sensors for body-axis angular acceleration, so the values of  $\dot{p}$ ,  $\dot{q}$ , and  $\dot{r}$  in the equations must be computed by numerically differentiating body-axis angular rate measurements  $p$ ,  $q$ , and  $r$ . Other uses for numerical differentiation include conducting a rough check on data polarity by comparing  $\dot{\alpha}$  and  $q$  or  $\dot{\beta}$  and  $-r$  [see Eqs. (3.34b) and (3.34c)], incorporating time-derivative terms in a model, such as  $C_{L\dot{\alpha}}(\dot{\alpha}\bar{c}/2V)$ , and correcting sensor measurements to the c.g., discussed in Chapter 10.

The most straightforward method for computing numerical time derivatives is to use central finite differences applied to the measured data. However, measured data are usually noisy, and subtracting nearly equal quantities (i.e., neighboring measured values) to compute the derivative magnifies the noise.

A better approach is to differentiate the local smoothing solution discussed earlier. The local fitted polynomial was identified with sample times defined relative to the current point. If the local polynomial model is differentiated with respect to time, then evaluated at the current time where  $t = 0$ , the value of the local smoothed derivative is equal to the coefficient of the linear term in the polynomial model identified for local smoothing. Using the local model of Eq. (11.9),

$$\dot{y} = a_1 + a_2 t \quad (11.41)$$

The local smoothed derivative is simply equal to the estimate of  $a_1$  at the current point, where  $t = 0$ . This local smoothed derivative calculation is applied at each data point, as was done for local time-domain smoothing.

Smoothed numerical differentiation can also be done using the global Fourier smoothing technique described earlier. In this case, the deterministic part of the measured signal is represented as a weighted sine series with an added linear trend [Eqs. (11.33) and (11.32)],

$$\hat{y}(i) = z_s(i) = \hat{g}_s(i) + z(0) + i \left[ \frac{z(N-1) - z(0)}{N-1} \right] \quad i = 0, 1, 2, \dots, N-1 \quad (11.42a)$$

$$\hat{g}_s(i) = \sum_{k=1}^{N-1} \Phi(k) b(k) \sin \left[ k \pi \left( \frac{i}{N-1} \right) \right] \quad i = 0, 1, 2, \dots, N-1 \quad (11.42b)$$

The smoothed derivative can be found by differentiating the sine series and the linear trend before transforming back to the time domain. The expression for the

smoothed derivative is

$$\frac{d\hat{y}(i)}{dt} = \left[ \frac{z(N-1) - z(0)}{N-1} \right] + \sum_{k=1}^{N-1} \Phi(k) b(k) \left( \frac{k\pi}{N-1} \right) \cos \left[ k\pi \left( \frac{i}{N-1} \right) \right]$$

$$i = 0, 1, 2, \dots, N-1 \quad (11.43)$$

## 11.5 Signal Comparisons

In many cases, system identification techniques are used to find linear relationships between input and output variables. A simple example would be estimating stability and control derivatives for the nondimensional pitching moment coefficient in an equation-error formulation (cf. Chapter 5). In that case, the model equation might be

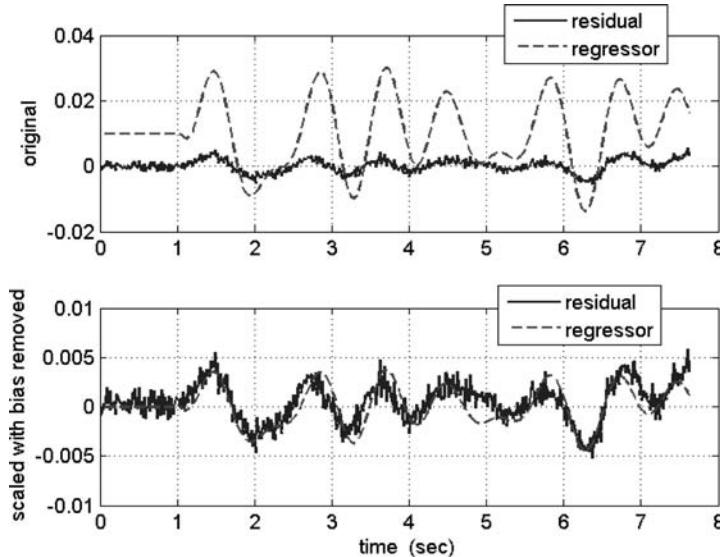
$$C_m = C_{m_0} + C_{m_\alpha} \alpha + C_{m_q} \frac{q\bar{c}}{2V} + C_{m_\delta} \delta + v_m \quad (11.44)$$

In matching the measured  $C_m$  with the expansion on the right, the  $C_{m_0}$  parameter takes care of the bias, and the terms involving stability and control derivatives model the variations in  $C_m$  over the time period for which the data are collected. The stability and control derivative estimates result in the best match with the measured nondimensional pitching moment coefficient in a least-squares sense.

The stability and control derivatives are therefore linear scaling parameters relating regressors formed from independent variables (states and controls) to the dependent variable (nondimensional coefficient). Linear scaling parameters can modify the magnitude of any of the regressors, but cannot modify the time variation in the regressors. The variation of the dependent variable over time must be matched by the terms in the model equation (11.44) or, equivalently, the model structure. When the model structure is inadequate, the error term  $v_m$  will be different from a white noise sequence. In that case, the task is to find a regressor that varies in the same manner as the remaining deterministic part of  $v_m$ , and then add that regressor to the model structure, possibly improving the model. The scaling of this added regressor is irrelevant for model structure determination, because the associated model parameter estimate will provide the necessary scaling.

It is therefore useful to have a method to compare the time variations of potential regressors and the current model residual, with scaling and biases removed. This gives some insight as to which regressors have a time variation that can be used to model the remaining variation in the dependent variable. Such a comparison can be made by removing the bias from each time series, and then scaling the second signal to have the same root-mean-square amplitude as the first signal.

Figure 11.12 shows a model residual time series and a potential model regressor in their original form (upper plot), and after removing biases and scaling



**Fig. 11.12 Comparison of model residuals and postulated regressor.**

(lower plot). The figure shows how this simple data processing can significantly clarify the question of whether or not a postulated model regressor would be useful in the model equation. An alternate graphical method is to cross-plot the time series for the model residual and the candidate regressor. If the time series are perfectly correlated, the result will be a straight line. Figure 5.18 shows examples of this type of plot. Either plot type can be considered a graphical analog of the correlation coefficient defined in Eq. (5.16a).

## 11.6 Finite Fourier Transform

In Chapter 7, it was noted that the finite Fourier transform

$$\tilde{x}(f) \equiv \int_0^T x(t) e^{-j2\pi f t} dt \quad (11.45)$$

can be approximated by

$$\tilde{x}(f) \approx \Delta t \sum_{i=0}^{N-1} x(i) e^{-j2\pi f i \Delta t} \quad (11.46)$$

which is a simple Euler approximation for the finite Fourier transform using  $N$  discrete samples of the continuous time function  $x(t)$ . For a conventional finite Fourier transform, the frequencies are chosen as

$$f_k = \frac{k}{N\Delta t} \quad k = 0, 1, 2, \dots, N-1 \quad (11.47a)$$

## DATA ANALYSIS

371

or

$$\omega_k = 2\pi f_k = 2\pi \frac{k}{N\Delta t} \quad k = 0, 1, 2, \dots, N-1 \quad (11.47b)$$

Using the discrete frequencies defined in Eqs. (11.47), the approximation to the finite Fourier transform in Eq. (11.46) becomes

$$\tilde{x}(k) \approx \Delta t \sum_{i=0}^{N-1} x(i) e^{-j(2\pi k/N)i} \quad k = 0, 1, 2, \dots, N-1 \quad (11.48)$$

The summation in Eq. (11.48) is the discrete Fourier transform  $X(k)$  from Chapter 7, so the finite Fourier transform can be approximated by multiplying the discrete Fourier transform (DFT) by the sampling interval  $\Delta t$ ,

$$\tilde{x}(k) \approx \Delta t X(k) \quad (11.49a)$$

where

$$X(k) \equiv \sum_{i=0}^{N-1} x(i) e^{-j(2\pi k/N)i} \quad k = 0, 1, 2, \dots, N-1 \quad (11.49b)$$

For flight data analysis, there are two main disadvantages to using the preceding approximation:

- 1) The spacing of the discrete frequencies for the DFT in Hz is roughly equal to the reciprocal of the data record length, so that the frequencies for the transformed data are fixed at values defined by  $N$  and  $\Delta t$  in Eqs. (11.47). Efficient fast Fourier transform (FFT) algorithms for computing the DFT apply to this particular selection of frequencies, which means that the frequencies cannot be selected arbitrarily for a given data record length.
- 2) The approximation of the finite Fourier transform using the DFT is a zeroth-order Euler approximation of the integrand  $x(t)e^{-j\omega t} = x(t) \cos(\omega t) - jx(t) \sin(\omega t)$ . Since the integrand oscillates as the integration variable  $t$  changes, the Euler approximation can be inaccurate. The integrand oscillates more rapidly as  $\omega$  increases, which means that the Euler approximation gets worse with increasing frequency. The same problem occurs as the selected  $\Delta t$  for the discrete approximation gets larger.

Using the FFT implies use of the frequencies in Eqs. (11.47). In this case, the frequency resolution becomes more coarse as the data record length decreases, leading to a loss of detail in the frequency domain, with consequent degraded data analysis and modeling results. Adding zeros to the measured time-domain data to artificially increase the data record length, known as zero padding, results in interpolation of the frequency-domain data obtained from the original time series, rather than increased resolution in the frequency domain. Zero

padding is also done to make the number of data points in the time domain equal to an integer power of 2, which is a requirement for most FFT algorithms.

Modal frequencies for rigid-body dynamics of full-scale aircraft are fairly low, usually below 2 Hz. For a small frequency band like this, most of the processing involved in the usual calculation of the discrete Fourier transform is for frequencies outside the range of interest. Finally, typical post-flight data analysis does not involve tight limitations on computation time, so the speed of FFT algorithms is not needed.

To overcome these problems, the finite Fourier transform can be computed using an improved approximation to the defining integral in Eq. (11.45), for frequencies that can be chosen arbitrarily. The idea is to evaluate the finite Fourier transform for selected frequencies, using values of  $x$  that are interpolated based on the sampled values  $x(i)$ , to improve the accuracy of the transform. Since the interpolation approximates the continuous function  $x(t)$ , rather than  $x(t)e^{-j\omega t}$ , the accuracy of the transform is not dependent on the selected frequencies. The work of Filon<sup>5</sup> is the original source for this approach, using a quadratic interpolation scheme for the time-domain data in the integrand of the finite Fourier transform. The approach was extended in Ref. 1 to use cubic interpolation and in Ref. 6 to use the chirp z-transform to achieve arbitrary frequency resolution in the Fourier transform, regardless of the length of the data record. By combining these techniques, the frequency-domain data points can be concentrated in the frequency band of interest and computed very accurately, resulting in high-quality modeling and data analysis results with excellent computational efficiency. Equations for the finite Fourier transform approach described in Refs. 1 and 6 are included here.

The DFT in Eq. (11.49b) can be rewritten as

$$X(k) = \sum_{i=0}^{N-1} x(i) AW^{-ki} \quad (11.50a)$$

where

$$A = 1 \quad W = \exp\left(j\frac{2\pi}{N}\right) \quad (11.50b)$$

The quantity  $AW^k$  for  $k = 0, 1, \dots, N - 1$  represents equally spaced points on the unit circle in the complex plane, separated by the angle  $2\pi/N$ . Interpreting  $AW^k$  as a complex transform variable  $z$ , the selections in Eq. (11.50b) correspond to a z-transform of the time-domain data  $x(i)$ ,  $i = 0, 1, \dots, N$ , for values of  $z$  placed uniformly around the unit circle in the complex plane.

The values of  $A$  and  $W$  in Eq. (11.50a) can be changed to implement the z-transform for a different contour in the complex plane. This is called a chirp z-transform. Specifically, the angular steps implemented by  $W$  can be chosen, while the value of  $A$  can be selected to start the contour at an arbitrary location in the complex plane. For example,  $A$  can be chosen to remain on the unit circle in the complex plane and start the contour at an arbitrary frequency,

## DATA ANALYSIS

373

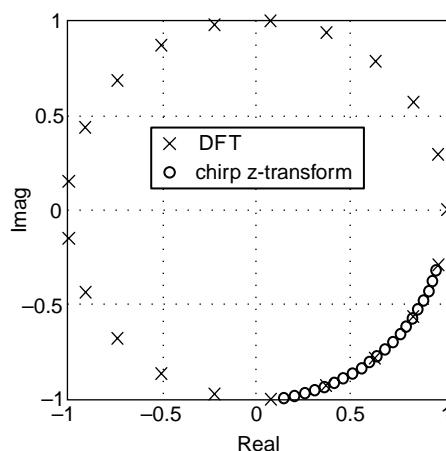
corresponding to the lower limit of a frequency band of interest. The value of  $W$  can be selected to implement the desired frequency resolution for the transform. The chirp  $z$ -transform is therefore a discrete Fourier transform with selectable frequency range and resolution.

Assuming the contour in the complex plane is along the unit circle, the general forms for  $A$  and  $W$  are

$$A = \exp(j\theta_o) \quad W = \exp(j\Delta\theta) \quad (11.50c)$$

where  $\omega_o = \theta_o/\Delta t$  represents the lower limit of the frequency band for the chirp  $z$ -transform in rad/s and  $\Delta\omega = \Delta\theta/\Delta t$  is the frequency resolution. The values of  $\theta_o$  and  $\Delta\theta$  can be chosen arbitrarily, with the limitation that both  $\theta_o/\Delta t$  and  $\Delta\theta/\Delta t$  must be in the range  $[0, \pi/\Delta t]$ , where  $\pi/\Delta t$  is the Nyquist frequency in rad/s. Figure 11.13 shows the values of the complex transform variable  $z$  for the conventional discrete Fourier transform [cf. Eqs. (11.47) and (11.49b)] and a chirp  $z$ -transform defined for a selected frequency range and resolution.

When the selected frequencies are regularly spaced in a frequency band of interest, Rabiner et al.<sup>7</sup> explain how the chirp  $z$ -transform can be computed efficiently using the FFT to implement a high-speed convolution. Uniform frequency spacing is the preferred approach for aircraft system identification, to make sure that all important features in the frequency domain are captured. Interpolation of the time-domain data samples to achieve high accuracy for the finite Fourier transform can be implemented by weightings applied to the values of the chirp  $z$ -transform, as shown in Refs. 1 and 6.



**Fig. 11.13** Values of the transform variable  $z$  for the discrete Fourier transform and the chirp  $z$ -transform.

The chirp  $z$ -transform values are just the discrete Fourier transform evaluated at  $M$  arbitrary frequencies  $f_k$  in Hz, selected in the range  $0 \leq f_k \leq f_N$ ,

$$X(k) = \sum_{i=0}^{N-1} x(i) e^{-j2\pi f_k i} \quad 0 \leq f_k < f_N, \quad k = 0, 1, 2, \dots, M-1 \quad (11.51)$$

The number of frequencies  $M$  can be chosen, but  $M$  should not be greater than approximately  $2N$ , or else numerical problems can occur.

Defining

$$\theta \equiv \omega_k \Delta t = 2\pi f_k \Delta t \quad k = 0, 1, 2, \dots, M-1 \quad (11.52)$$

the expression for high-accuracy calculation of the finite Fourier transform is

$$\begin{aligned} \tilde{x}(\theta) \approx \Delta t \{ W(\theta) X(k) \\ + \gamma_0(\theta)x(0) + \gamma_1(\theta)x(1) + \gamma_2(\theta)x(2) + \gamma_3(\theta)x(3) \\ + e^{j\theta T/\Delta t} [\gamma_0^*(\theta)x(N-1) + \gamma_1^*(\theta)x(N-2) \\ + \gamma_2^*(\theta)x(N-3) + \gamma_3^*(\theta)x(N-4)] \} \end{aligned} \quad (11.53)$$

where the weights  $W(\theta)$ ,  $\gamma_0(\theta)$ ,  $\gamma_1(\theta)$ ,  $\gamma_2(\theta)$ , and  $\gamma_3(\theta)$  are found by analytically evaluating the finite Fourier transform integral in Eq. (11.45), using cubic Lagrange interpolation applied to the sampled time-domain data  $x(i)$ ,  $i = 0, 1, 2, \dots, N-1$ . The resulting expressions for the weights are

$$W(\theta) = \left( \frac{6 + \theta^2}{3\theta^4} \right) (3 - 4 \cos \theta + \cos 2\theta) \approx 1 - \frac{11}{720} \theta^4 + \frac{23}{15120} \theta^6 \quad (11.54a)$$

$$\begin{aligned} \gamma_0(\theta) = & \frac{(-42 + 5\theta^2) + (6 + \theta^2)(8 \cos \theta - \cos 2\theta)}{6\theta^4} \\ & - j \frac{(-12\theta + 6\theta^3) + (6 + \theta^2) \sin 2\theta}{6\theta^4} \\ \approx & -\frac{2}{3} + \frac{1}{45} \theta^2 + \frac{103}{15120} \theta^4 - \frac{169}{226800} \theta^6 \\ & - j\theta \left( \frac{2}{45} + \frac{2}{105} \theta^2 - \frac{8}{2835} \theta^4 + \frac{86}{467775} \theta^6 \right) \end{aligned} \quad (11.54b)$$

$$\gamma_1(\theta) = \frac{14(3 - \theta^2) - 7(6 + \theta^2) \cos \theta}{6\theta^4}$$

$$- j \frac{30\theta - 5(6 + \theta^2) \sin \theta}{6\theta^4}$$

## DATA ANALYSIS

375

$$\approx \frac{7}{24} - \frac{7}{180} \theta^2 + \frac{5}{3456} \theta^4 - \frac{7}{259200} \theta^6 - j\theta \left( \frac{7}{72} - \frac{1}{168} \theta^2 + \frac{11}{72576} \theta^4 - \frac{13}{5987520} \theta^6 \right) \quad (11.54c)$$

$$\begin{aligned} \gamma_2(\theta) &= \frac{-4(3 - \theta^2) + 2(6 + \theta^2) \cos \theta}{3\theta^4} \\ &\quad - j \frac{-12\theta + 2(6 + \theta^2) \sin \theta}{3\theta^4} \\ &\approx -\frac{1}{6} + \frac{1}{45} \theta^2 - \frac{5}{6048} \theta^4 + \frac{1}{64800} \theta^6 \\ &\quad - j\theta \left( -\frac{7}{90} + \frac{1}{210} \theta^2 - \frac{11}{90720} \theta^4 + \frac{13}{7484400} \theta^6 \right) \end{aligned} \quad (11.54d)$$

$$\begin{aligned} \gamma_3(\theta) &= \frac{2(3 - \theta^2) - (6 + \theta^2) \cos \theta}{6\theta^4} \\ &\quad - j \frac{6\theta - (6 + \theta^2) \sin \theta}{6\theta^4} \\ &\approx \frac{1}{24} - \frac{1}{180} \theta^2 + \frac{5}{24192} \theta^4 - \frac{1}{259200} \theta^6 \\ &\quad - j\theta \left( \frac{7}{360} - \frac{1}{840} \theta^2 + \frac{11}{362880} \theta^4 - \frac{13}{29937600} \theta^6 \right) \end{aligned} \quad (11.54e)$$

In the preceding equations,  $\theta$  takes  $M$  different values, corresponding to the  $M$  selected frequencies  $f_k$ , resulting in  $M$  values of the finite Fourier transform. Detailed derivation of the preceding expressions can be found in Ref. 6. The truncated series expansions for  $W(\theta)$ ,  $\gamma_0(\theta)$ ,  $\gamma_1(\theta)$ ,  $\gamma_2(\theta)$ , and  $\gamma_3(\theta)$  given in Eqs. (11.54) are used for small  $\theta$ , where  $\theta$  is defined to be small when it is less than the largest value of  $\theta$  for which identical results are obtained to machine precision using the analytic expression and its truncated series expansion. The series expansions are necessary because of high-order cancellations that make the analytic expressions inaccurate when  $\theta$  is small.

It is advantageous to select the frequencies for the finite Fourier transform to be closely spaced within the frequency band of interest, to ensure that details of the frequency spectrum are accurately captured. The chirp  $z$ -transform can be used with arbitrary frequency resolution, independent of the length of the time record, and without the restriction that  $N$  be a power of 2. The chirp  $z$ -transform therefore decouples the frequency resolution from the length of the time record and can place all calculated frequency points within the frequency band of interest. The price for this flexibility is more computation time, but the magnitude of this extra computation time is relatively small.

For example, a flight test data analyst might select  $M$  discrete frequencies in a frequency band  $[f_0, f_1]$  Hz, so that the selected frequencies would be

$$f_k = f_0 + k\Delta f \quad k = 0, 1, 2, \dots, M - 1 \quad (11.55)$$

where

$$\Delta f = \frac{(f_1 - f_0)}{(M - 1)} \quad (11.56)$$

For rigid-body dynamics of a full-scale aircraft, typical values might be  $f_0 = 0.1$  Hz,  $\Delta f = 0.02$  Hz, and  $f_1 = 2$  Hz, so that the vector of selected frequencies is  $\mathbf{f} = [0.10, 0.12, 0.14, \dots, 2]$  Hz and  $M = 96$ . The result is 96 data points in the frequency domain, evenly spaced in the interval  $[0.1, 2]$  Hz, regardless of the number of data points in the time domain. This is a significant advantage, particularly for iterative methods, since the data analysis and modeling computations will be faster for fewer data points.

Note that the chirp  $z$ -transform will compute the conventional DFT in Eq. (11.49b) for the selections  $M = N$  and  $f_k$  from Eq. (11.47a).

In summary, the procedure for computing the high-accuracy finite Fourier transform is as follows:

- 1) Choose three of the four quantities in Eq. (11.56), and compute the remaining value from Eq. (11.56). This defines the frequencies for the analysis, using Eq. (11.55).
- 2) Compute the  $M$  values of  $\theta$  from Eq. (11.52).
- 3) Use the chirp  $z$ -transform to compute the DFT for the selected frequencies, with Eqs. (11.50).
- 4) Use Eqs. (11.53) and (11.54) to compute the high-accuracy finite Fourier transform for each  $\theta$ .

Although other quadrature methods can be used to accurately compute the finite Fourier transform, the technique presented here has the advantages of convenient and efficient calculation, while at the same time achieving high-accuracy and arbitrary frequency resolution that is independent of the data record length.

## 11.7 Power Spectrum Estimation

In Chapter 7, the expression for computing the spectral density of a time function  $u(t)$  was given as

$$S_{uu}(f) = \tilde{u}(f)\tilde{u}^*(f) \quad (11.57)$$

where  $\tilde{u}(f)$  is the Fourier transform of  $u(t)$ . As discussed in Chapter 7, spectral densities can be used in frequency-domain modeling methods, as well as for the general purpose of examining frequency content of time series. The spectral densities are also called power spectral densities or the power spectrum.

## DATA ANALYSIS

377

Intuitively, the total power of  $u(t)$  should be the same when expressed in either the time domain or the frequency domain. This idea is the basis of Parseval's theorem,

$$\text{Total power} = \int_{-\infty}^{\infty} u^2(t) dt = \int_{-\infty}^{\infty} \tilde{u}(f)\tilde{u}^*(f) df \quad (11.58)$$

Assuming that  $u(t) = 0$  for  $t < 0$ ,

$$\text{Total power} = \int_0^{\infty} u^2(t) dt = \int_{-\infty}^{\infty} \tilde{u}(f)\tilde{u}^*(f) df \quad (11.59)$$

In the literature, many different definitions of power spectral densities and total power are used to enforce this concept.<sup>1</sup> In this book, and in the accompanying software, the power spectral density is normalized so that the sum of the squared spectral density estimates equals the mean squared value of the time series. The discrete form of Parseval's theorem is then

$$\frac{1}{N} \sum_{i=0}^{N-1} u^2(i)\Delta t = \frac{1}{N} \sum_{k=0}^{N-1} [U(k)\Delta f][U^*(k)\Delta f]\Delta f \quad (11.60)$$

where Eq. (11.49a) was used. For the frequencies specified in Eq. (11.47a), the discrete form of Parseval's theorem reduces to

$$\frac{1}{N} \sum_{i=0}^{N-1} u^2(i) = \frac{1}{N^2} \sum_{k=0}^{N-1} U(k)U^*(k) \quad (11.61)$$

or

$$\frac{1}{N} \mathbf{u}^T \mathbf{u} = \frac{1}{N^2} \mathbf{U}^\dagger \mathbf{U} \quad (11.62)$$

where

$$\begin{aligned} \mathbf{u} &= [u(0) \quad u(1) \quad \cdots \quad u(N-1)]^T \\ \mathbf{U} &= [U(0) \quad U(1) \quad \cdots \quad U(N-1)]^T \end{aligned} \quad (11.63)$$

When the frequency spacing is arbitrarily chosen as in Eqs. (11.55) and (11.56), Parseval's theorem only holds when all of the signal power lies within the frequency band selected for the analysis. Assuming that is the case, the discrete form of Parseval's theorem follows from Eq. (11.60) as

$$\frac{1}{N} \sum_{i=0}^{N-1} u^2(i)\Delta t = \frac{2}{N} \sum_{k=0}^{M-1} \tilde{u}(k)\tilde{u}^*(k)\Delta f \quad (11.64)$$

The factor of 2 on the right side of Eq. (11.64) accounts for the fact that the terms associated with negative frequencies in the summation of Eq. (11.60) are not present for the chirp  $z$ -transform. Since the DFT has conjugate symmetry [cf. Eq. (7.14)], the values in the summation of the right side of Eq. (11.64) must be doubled, except for the endpoint frequencies of 0 and  $f_N = 1/(2\Delta t)$  Hz. If the  $f_k$  of the chirp  $z$ -transform include 0 or  $f_N$ , the corresponding term in the summation has a factor of 1 instead of 2. Power spectra that use the factor of 2 and only positive frequencies are called one-sided power spectra. If negative frequencies are included, there is no factor of 2, and the power spectra are called two-sided.

There are three important practical issues associated with the use of the discrete Fourier transform to compute power spectral density. The first is that the number of frequencies used for the Fourier transformation is finite, so that in practice the underlying continuous spectral density function is characterized by discrete values for small frequency bands centered at each frequency  $f_k$ , namely  $[f_k - \Delta f/2, f_k + \Delta f/2]$ . The power spectral density estimates are therefore equivalent values for the continuous frequencies contained in each frequency bin centered on  $f_k$ .

The second issue arises because a finite length of data is used to estimate the power spectrum. A finite length of data on the time interval  $[0, T]$  can be obtained by multiplying an infinite length of data by a function that equals 1 on the time interval  $[0, T]$  and is zero otherwise. This is called the boxcar function, shown in the upper plot of Fig. 11.14. From Chapter 7 and Appendix A, it is known that the

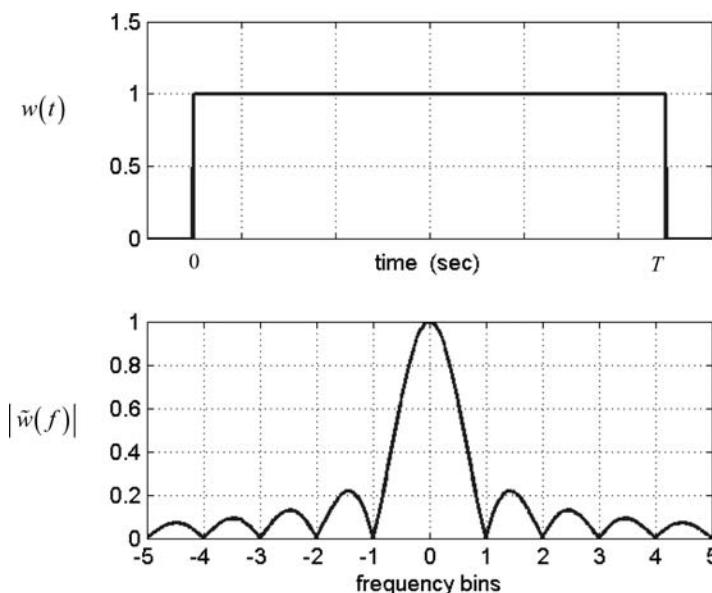


Fig. 11.14 Boxcar function and its Fourier transform.

## DATA ANALYSIS

379

Fourier transform of the product of two time-domain functions is equivalent to convolution of their Fourier transforms in the frequency domain. The Fourier transform of the boxcar function is oscillatory with a poor rate of decrease for the side lobes as a function of frequency, as shown in the lower plot of Fig. 11.14. This behavior is related to the fact that it is difficult for a Fourier series to represent a function with discontinuities, such as the boxcar function. Therefore, the Fourier transform of a finite length of data is equivalent to convolution in the frequency domain of the Fourier transform of the time function of interest (assumed infinite in length) with the Fourier transform of the boxcar function. This causes a smearing or leakage of frequency components from each frequency bin into adjacent ones, resulting in reduced accuracy of the power spectral estimates. Note that there is no leakage if the only frequencies present in the data are those associated with the frequency bins (see the lower plot of Fig. 11.14).

To mitigate leakage, the finite data can be multiplied by a windowing function that changes gradually from zero to one, and then returns to zero. A windowing function without the discontinuity of the boxcar function has a Fourier transform with fewer and smaller side lobes, and therefore reduces leakage. There are many windowing functions that can be used.<sup>1,8</sup> A good practical choice is the Bartlett window, defined by

$$w(i) = 1 - \left| \frac{i - N/2}{N/2} \right| \quad i = 0, 1, \dots, N - 1 \quad (11.65)$$

which is a simple ramp from 0 to 1 and back to 0. When data windowing is implemented, the time function  $u(i)$  in Eq. (11.61), for example, is modified to  $w(i)u(i)$ , where  $w(i)$  is the value of the windowing function at the  $i$ th data point. To maintain the validity of Parseval's theorem, the power spectral density estimates must be divided by the sum of squares of the weighting function,

$$\frac{1}{N} \sum_{i=0}^{N-1} u^2(i) = \frac{1}{N^2 \sum_{i=0}^{N-1} w^2(i)} \sum_{k=0}^{N-1} U_w(k)U_w^*(k) \quad (11.66)$$

where  $U_w(k)$  is computed from the windowed time series  $w(i)u(i)$ ,  $i = 0, 1, \dots, N - 1$ .

Finally, calculating the spectral densities as the product of the Fourier transform with its complex conjugate [cf. Eq. (11.57)] results in spectral density estimates with standard errors equal to 100% of the computed values.<sup>1,8</sup> In practice, this difficulty can be overcome using some form of averaging. One common method is to partition the measured time series into  $n$  segments of data. The standard calculation of the spectral density given in Eq. (11.57) is applied to each data segment. Then the results at each frequency are averaged, resulting in a spectral density estimate with random error variance reduced from 100% by a factor of  $1/n$ , due to the averaging. To achieve more averaging, the data segments are normally overlapped by 50%, meaning that adjacent segments share 50% of

their data points. This increases the number of averages  $n$  but decreases the variance reduction factor to  $9/(11n)$ , because some data are reused in adjacent data segments. However, the increased number of averages from overlapping data segments more than compensates for the decrease in the variance reduction factor.

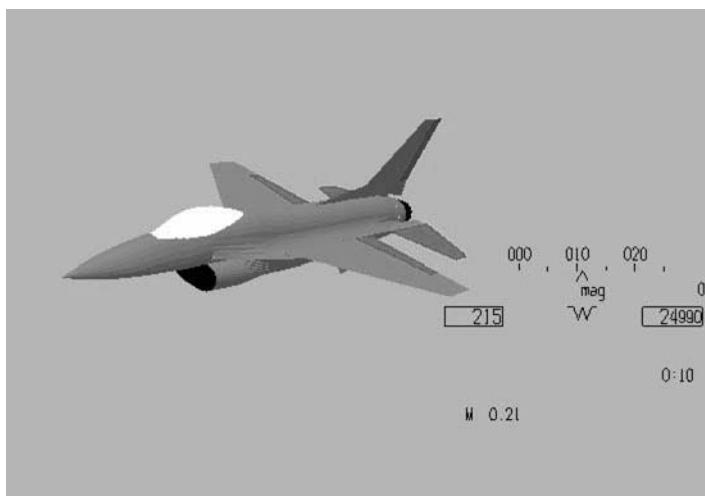
Another method for reducing the variance in the power spectral density estimates, which is nearly identical mathematically to the method just described, is to compute the Fourier transform at a frequency resolution  $n$  times finer than is desired for the end result. Equation (11.57) is used to compute the spectral densities on the fine frequency mesh, and then the results are summed for adjacent  $n$  frequencies to produce results for the desired (coarser) frequency mesh. This results in spectral density estimates with random error variance reduced from 100% by a factor of  $1/n$ . The spectral density values are summed rather than averaged, to maintain the validity of Parseval's theorem, discussed earlier. The summing operation is also consistent with the previous discussion concerning discrete frequency bins.

The second method is particularly easy to implement with the chirp  $z$ -transform described earlier, because the chirp  $z$ -transform has the capability for arbitrary frequency resolution. The approach is simply to determine the number of values  $n$  required to achieve the desired accuracy for the spectral density estimates, and select a frequency mesh fine enough to provide the required number of adjacent values. Then implement Eq. (11.57) and sum the adjacent results to compute an accurate spectral estimate at each desired frequency.

## 11.8 Maneuver Visualization

Measured data from a flight-test maneuver include time series for many different physical quantities. It is extremely difficult to get a good overall mental picture of what the aircraft is doing by looking at many two-dimensional plots, such as angle of attack or elevator deflection versus time. This is particularly true for maneuvers with combined longitudinal and lateral motion or for maneuvers at high angles of attack, among others. Modern three-dimensional graphics rendering software that can run efficiently on desktop personal computers make it possible to create real-time visualizations of the aircraft in flight. In effect, the flight-test maneuver can be replayed for the analyst as often as desired, and using whatever vantage point is useful. This capability gives the analyst insight that simply is not available from studying two-dimensional plots or streams of numbers.

Figure 11.15 shows what the display looks like when replaying a maneuver using measured flight data with a three-dimensional solid model of the F-16. The software used is called Aviator Visual Design Simulator (AVDS), available from RasSimTech, Ltd.; however, there are other software packages available with similar capability. Note that a head-up display (HUD) is included so that the analyst has access to the numerical values of key quantities while watching the recreated aircraft motion. This software can also show the air-relative velocity vector and movement of the individual control surfaces on the aircraft.



**Fig. 11.15 Maneuver visualization.**

## 11.9 Summary and Concluding Remarks

This chapter includes a selection of data analysis techniques that have been found useful in practice for aircraft system identification. The list is by no means exhaustive, but instead represents some proven approaches to common problems. The techniques described here are often among the first to be applied to measured flight data. It is therefore important that the procedures are well understood and properly implemented, so that subsequent modeling results are not compromised.

Most of the data analysis techniques described provide some alternate view of the data. For example, filtering and smoothing can be used to strip away high-frequency noise, revealing an underlying deterministic signal; Fourier transformation and power spectral estimates indicate how power in the time-domain signal is distributed as a function of frequency; and maneuver visualization shows a realistic playback of the flight data using a depiction of the aircraft in flight.

Several of the methods described in this chapter are required for implementing modeling methods described earlier in the book. An example is smoothed numerical differentiation, which is required for equation-error parameter estimation methods discussed in Chapters 5 and 6, as well as sensor position corrections in Chapter 10. The high-accuracy Fourier transform for arbitrary frequencies is important for the frequency-domain methods described in Chapter 7.

This chapter marks the end of the presentation of methods for aircraft system identification. The next and final chapter provides details of the accompanying software package and how to use it effectively.

### References

- <sup>1</sup>Press, W. H., Teukolsky, S. A., Vettering, W. T., and Flannery, B. R., *Numerical Recipes in FORTRAN: The Art of Scientific Computing*, 2nd ed., Cambridge Univ. Press, New York, 1992.
- <sup>2</sup>Graham, R. J., "Determination and Analysis of Numerical Smoothing Weights," NASA TR R-179, 1963.
- <sup>3</sup>Lanczos, C., *Applied Analysis*, Dover, New York, 1988.
- <sup>4</sup>Morelli, E. A., "Estimating Noise Characteristics from Flight Test Data Using Optimal Fourier Smoothing," *Journal of Aircraft*, Vol. 32, No. 4, July–August 1995, pp. 689–695.
- <sup>5</sup>Filon, L. N. G., "On a Quadrature Formula for Trigonometric Integrals," *Proceedings of the Royal Society of Edinburgh*, Vol. 49, 1928, pp. 38–47.
- <sup>6</sup>Morelli, E. A., "High Accuracy Evaluation of the Finite Fourier Transform Using Sampled Data," NASA TM 110340, 1997.
- <sup>7</sup>Rabiner, L. R., Schafer, R. W., and Rader, C. M., "The Chirp  $z$ -Transform Algorithm and Its Application," *Bell System Technical Journal*, Vol. 48, May–June 1969, pp. 1249–1292.
- <sup>8</sup>Bendat, J. S., and Piersol, A. G., *Random Data Analysis and Measurement Procedures*, 2nd ed., Wiley, New York, 1986.

## 12

# MATLAB® Software

This chapter explains the practical use of MATLAB® software that implements the aircraft system identification methods described in the preceding chapters. The software was written in the MATLAB® language, but is not a product of the MathWorks, Inc., which produces and sells MATLAB®. MATLAB® is a high-level programming language with extensive capabilities for numerical computation and graphics. Familiarity with basic operations in MATLAB® is needed to use the software associated with this book. Introductory material in Ref. 1 provides the necessary background information. Excellent help files and tutorials are also readily available within the MATLAB® computing environment.

In the following descriptions, commands typed by the user in the MATLAB® command window at the `>>` prompt will appear in **this font**. All such commands are followed by a carriage return. Text that is output to the command window by MATLAB® will appear in **this font**. Names of m-files will appear in **this font**.

### 12.1 Overview

The software associated with this textbook is a collection of m-files called System IDentification Programs for AirCraft, or SIDPAC. An m-file in MATLAB® is the equivalent of a subroutine in FORTRAN or a function subprogram in C. SIDPAC was developed at NASA Langley Research Center more or less continuously since 1992, and has been applied to flight data, wind-tunnel data, and simulation data from many different projects. Each purchaser of this textbook can download SIDPAC at <http://www.aiaa.org/publications/support> materials.

SIDPAC was developed under several versions of MATLAB® over the years. Proper operation of the SIDPAC software has been validated for MATLAB® version 7.1, release 14. Software validation and the textbook examples were done on an IBM-compatible personal computer (PC). However, MATLAB® is a platform-independent programming language, so the software should work properly on any computer running an appropriate version of MATLAB®.

Each m-file in SIDPAC performs one task or implements one method. There are SIDPAC routines that implement the techniques discussed in this book, as well as some techniques that were not discussed. Comments in each SIDPAC m-file include a list of other SIDPAC m-files called. Any standard MATLAB® routine called by a SIDPAC m-file is not listed as a called routine. SIDPAC does not require any specialized add-on MATLAB® toolboxes, just the standard MATLAB® software.

Some m-files in SIDPAC also have a C mex-file equivalent. The reason is that C mex-files run much faster, so the m-files with C mex equivalents are computationally intensive or called frequently. Each C mex-file code has the same name as its equivalent m-file, but has a .c file extension instead of .m. Compiled C mex-files for the PC have the same name, with a .dll file extension. MATLAB® treats compiled mex-files in the same manner as m-files, in terms of execution; however, MATLAB® will always use a .dll file before using a .m file with the same name. Because of this, SIDPAC will always try to use the faster .dll code first. If the computer being used is not a PC, then the .c mex-files will have to be re-compiled. To do this, first type `mex -setup` at the MATLAB® command prompt, which implements a one-time setup process. Then type `mex filename.c`, where `filename.c` is any of the SIDPAC files with the .c file extension. If the .dll file is deleted or placed in a location that is not in the MATLAB® path, then SIDPAC will use the .m file equivalent, which is slower.

The folder containing SIDPAC m-files can be placed anywhere in the MATLAB® path. The `path` command in MATLAB® can be used to include the SIDPAC folder in the MATLAB® path. The `startup.m` file included with SIDPAC will set the MATLAB® path to automatically include a subdirectory called SIDPAC. Commands in the `startup.m` file are executed automatically when MATLAB® is started. Make sure this `startup.m` file is the first one encountered in the MATLAB® path, which will be true if `startup.m` is in the current working directory.

Calling syntax and descriptive material for the m-files in SIDPAC appear in the header of each m-file. The header is displayed in the MATLAB® command window in response to typing `help filename`, where `filename` is the name of a SIDPAC m-file, without its .m file extension. To execute any m-file, the correct calling syntax can be copied from the header directly to the MATLAB® command window and executed with a carriage return.

The SIDPAC m-files were originally written for research purposes and not for public release, so error handling, user interface, and the like, are Spartan. SIDPAC includes a graphical user interface (GUI), which allows the user to call individual m-files using point-and-click, rather than typing the m-file command line syntax in the MATLAB® command window. To use the SIDPAC GUI, type `sid` at the MATLAB® command prompt. The GUI displays information and options in pull-down menus. SIDPAC m-files can also be used individually from the MATLAB® command line, even after the SIDPAC GUI has been started.

SIDPAC m-files will sometimes make plots or ask the user for information via the MATLAB® command window. Because of this, it is best to arrange the MATLAB® command and figure windows so that they do not overlap and obscure one another.

Examples in the textbook have been implemented in an m-file called `sidpac_text_examples.m`. By typing `sidpac_text_examples` and then entering an example number from the text, the user is taken through all the steps necessary to duplicate the selected textbook example. The MATLAB® code inside `sidpac_text_examples.m` shows how SIDPAC routines can be used to solve aircraft system identification problems.

SIDPAC also includes demonstration programs for some of the more commonly used routines in SIDPAC. All of these programs have names that include `_demo.m`, preceded by the name of the SIDPAC m-file being demonstrated, e.g., `swr_demo.m`. These programs are executed by typing their names (without the `.m` file extension) at the MATLAB® command prompt, followed by a carriage return.

Ref. 2 was used extensively as a source of theoretical and practical information in the development of SIDPAC. An overview of SIDPAC version 1.1 is given in Ref. 3. The software associated with this book is SIDPAC version 2.0. Version 2.0 includes a GUI, along with some upgrades and additions to the functions in version 1.1. Version 1.0 was released in January 2001 as the first version of SIDPAC available to the public.

Much of SIDPAC is based on the concept of a standardized arrangement of the measured data. For typical uses of SIDPAC, the measured data are arranged in a standard data matrix called `fdata`, which is shorthand for “flight data.” Each row of `fdata` corresponds to a data point, and each column represents a particular measurement with specified units. Measured data from a flight-test maneuver or other experiment are arranged by column in a standard format, defined in the SIDPAC file `SIDPAC_Data_Channels.doc` or `SIDPAC_Data_Channels.pdf`. This approach has been taken because each experimental program has its own nomenclature and units for measured signals. By standardizing the form of the measured data, validated SIDPAC routines can be used for many different projects, with little chance of data-handling errors, such as different sign conventions or unit conversion errors. Programming SIDPAC routines is also more standardized and straightforward, reducing the possibility of errors.

The conversion of measured data into the standard SIDPAC data format is done first, before any analysis. Normally, the user creates an m-file for each experimental program to convert measured data to the standard units, and to assign each measured signal to the proper column in the `fdata` matrix. There are spare (unused) columns in `fdata` for special cases, and the size of `fdata` is limited only by computer memory. SIDPAC m-file `f16_ftdatsel.m` is an example of an m-file created to convert data from the F-16 nonlinear simulation described in Appendix D into the standard `fdata` format expected by SIDPAC. Output from the F-16 nonlinear simulation is defined in the header of `f16.m`, which is the function used for batch nonlinear simulation of the F-16. All F-16 nonlinear simulation functions are written in MATLAB®, and are included with SIDPAC.

SIDPAC includes some utilities for loading data into the MATLAB® workspace; however, the native MATLAB® tools for this purpose are very capable. The initial screen of the SIDPAC GUI (which appears in response to typing `sid` at the MATLAB® command prompt) includes a point-and-click utility for

assembling the standard `fdata` matrix from variables loaded into the MATLAB® workspace. The SIDPAC GUI stores the measured data and results from data analysis and modeling done using the GUI in a single data structure named `fds` in the MATLAB® workspace. The `fds` data structure is initialized when the SIDPAC GUI is invoked, unless a variable named `fds` already exists in the MATLAB® workspace. The SIDPAC GUI assumes that the user has already loaded data into the MATLAB® workspace, but the data may or may not have been assembled into the standard `fdata` format.

Brief descriptions of all the functions in SIDPAC can be displayed in the MATLAB® command window by typing `help sidpac` at the MATLAB® command prompt. This list is intended to familiarize the user with SIDPAC capabilities and to help in selecting appropriate tools. A reference list of SIDPAC functions can be found in the SIDPAC file named `Contents.m`, in the SIDPAC directory.

The remainder of this chapter lists and describes individual m-files in SIDPAC that implement the main techniques. Table 12.1 gives a convenient reference for the main SIDPAC routines, including a brief description of what each one does,

**Table 12.1 SIDPAC main functions**

Filename	Description	References
<i>Equation error, Chap. 5</i>		
<code>lesq.m</code>	Linear regression	Chap. 5, Ref. 1; Chap. 7, Ref. 12; Chap. 12, Ref. 2
<code>mof.m</code>	Orthogonal function modeling using Gram-Schmidt orthogonalization	Chap. 5, Refs. 7, 19
<code>offit.m</code>	Orthogonal function modeling using sequential orthogonalization (precursor to <code>mof.m</code> )	Chap. 5, Ref. 7
<code>r_colores.m</code>	Colored residual corrections for Cramér-Rao bounds of linear regression parameter estimates	Chap. 6, Ref. 29
<code>swr.m</code>	Stepwise regression	Chap. 5, Refs. 1, 16; Chap. 12, Ref. 2
<i>Output error, Chap. 6</i>		
<code>colores.m</code>	Colored residual corrections for Cramér-Rao bounds of output-error parameter estimates	Chap. 6, Refs. 26, 29
<code>m_colores.m</code>	Vectorized (faster) version of <code>colores.m</code>	Chap. 6, Refs. 26, 29
<code>oe.m</code>	Output-error parameter estimation in the time domain	Chap. 6, Refs. 15, 18
<i>Frequency domain, Chap. 7</i>		
<code>fdoe.m</code>	Output-error parameter estimation in the frequency domain	Chap. 7, Refs. 12, 13

*(continued)*

## MATLAB® SOFTWARE

387

**Table 12.1 SIDPAC main functions (*continued*)**

Filename	Description	References
fint.m	High-accuracy finite Fourier transform for arbitrary frequencies	Chap. 7, Ref. 4; Chap. 12, Ref. 2
ttest.m	Equation-error parameter estimation for transfer function models	Chap. 7, Refs. 6, 14, 15
msls.m	Constrained sequential least-squares parameter estimation in the time domain	Chap. 8, Ref. 6
rlesq.m	Recursive least-squares parameter estimation	Chap. 8, Refs. 3, 4
rtpid.m	Sequential least-squares parameter estimation in the frequency domain	Chap. 8, Ref. 7; Chap. 9, Ref. 29
<i>Experiment design, Chap. 9</i>		
mkfswp.m	Generates frequency sweep inputs	Chap. 9, Ref. 10
mkmsswp.m	Generates multiple orthogonal multisine sweep inputs	Chap. 9, Refs. 11, 16
mkrdm.m	Generates white or colored noise inputs	Chap. 7, Ref. 2
mksqw.m	Generates arbitrary multistep square wave inputs	Chap. 9, Refs. 18, 19
<i>Data compatibility analysis, Chap. 10</i>		
airchk.m	Kinematic check on data for translational motion	Chap. 10, Ref. 14
dcmp.m	Kinematic equations and instrumentation error model for data compatibility analysis	Chap. 6, Ref. 18; Chap. 10, Refs. 12, 13
rotchk.m	Kinematic check on data for rotational motion	Chap. 10, Ref. 13
<i>Data analysis, Chap. 11</i>		
cmpsigs.m	Graphically compare different signals on a common scale with biases removed	Chap. 11
compfcc.m	Computes nondimensional aerodynamic force coefficients	Chaps. 3 and 5
comppmc.m	Computes nondimensional aerodynamic moment coefficients	Chaps. 3 and 5
deriv.m	Computes locally smoothed derivatives for noisy measured time series data	Chap. 11, Ref. 3
smoo.m	Global Fourier smoothing of noisy measured data	Chap. 11, Refs. 3, 4
xsmep.m	Replace the initial point of a measured time series with a smoothed estimate	Chap. 11

along with references to earlier chapters in this book and other sources for explanations of the algorithms and their implementation. A description of the purpose and operation of each m-file listed in Table 12.1 is given next, along with practical hints. Each description is followed by a listing of the m-file header, which defines inputs, outputs, and command syntax.

## 12.2 Linear Regression

### 12.2.1 *lesq.m*

Function *lesq.m* estimates model parameters for linear least-squares regression problems. The routine can be used with either real or complex data, which means that the same m-file can be used for time-domain data or frequency-domain data. The matrix inversion is done using singular value decomposition, with automatic modification of the singular values when the matrix of regressors is ill conditioned. Model fit error variance and parameter covariance matrix estimates are based on calculations that use the estimated parameter values. Since MATLAB® uses double precision arithmetic by default, it is not necessary to normalize the regressors to account for differences in the scale of the regressors. Accordingly, *lesq.m* does not normalize the regressors.

Function *lesq.m* does not automatically include a constant regressor, so it is necessary to include a column of ones in the regressor matrix *x* to estimate a bias term. Estimated parameter standard errors can be computed from the estimated covariance matrix *crb* by typing *serr=sqrt(diag(crb))*. In general, omit the *svlim* input and let the program compute and use the default value. The input *svlim* can be used to address collinearity problems in some cases. Optional inputs *p0* and *crb0* can be used to include *a priori* information from a previous analysis or another source.

The m-file named *trotter\_demo.m* includes a longitudinal example application of *lesq.m*, using flight-test data from the NASA Twin Otter aircraft. Type *trotter\_demo* at the MATLAB® prompt to run the demonstration.

*Header listing:*

```
%  
% LESQ Least squares linear regression.  
%  
% Usage: [y,p,crb,s2,xm,sv] = lesq(x,z,svlim,p0,crb0);  
%  
% Description:  
%  
% Computes the least squares estimate of the real parameter  
% vector p, where y = x*p and y matches the measured  
% quantity z in a least-squares sense. The model output y,  
% the estimated parameter covariance matrix crb, and  
% the model fit error variance s2, are estimated based on the  
% parameter estimate p. Inputs specifying the minimum  
% singular value ratio svlim, prior estimated parameter vector p0,  
% and prior estimated parameter covariance matrix crb0  
% are optional. This routine works for real or complex data.
```

```
%  
% Input:  
%  
%     x = matrix of column regressors.  
%     z = measured output vector.  
% svlim = minimum singular value ratio  
%         for matrix inversion (optional).  
%     p0 = prior parameter vector (optional).  
%     crb0 = prior parameter covariance matrix (optional).  
%  
% Output:  
%  
%     y = model output vector.  
%     p = vector of parameter estimates.  
%     crb = estimated parameter covariance matrix.  
%     s2 = model fit error variance estimate.  
%     xm = matrix of column vector model terms.  
%     sv = vector of singular values of the information matrix.  
%
```

### 12.2.2 r\_colores.m

Function `r_colores.m` estimates model parameters for a linear least-squares regression problem and computes Cramér-Rao bounds for the covariances of the estimated parameters, both conventionally and accounting for the frequency content of the residuals. This routine is for time-domain data—the correction is not necessary for frequency-domain data. Function `r_colores.m` should be run once at the end of the analysis, when the parameter estimates from least-squares parameter estimation (`lesq.m`, `swr.m`, or `mof.m`) are judged to be acceptable.

Output of `r_colores.m` includes the conventional Cramér-Rao bounds `crbo` and the corrected Cramér-Rao bounds `crb`. The `crbo` output should match the Cramér-Rao bounds from conventional least-squares parameter estimation (`lesq.m`, `swr.m`, or `mof.m`), within round-off error. Input `x` to `r_colores.m` should be the regressor matrix for the final selected model. When using `swr.m` or `mof.m` for least-squares parameter estimation, the regressor matrix `x` will in general be an output from the routine that is not the same as the input regressor matrix. In general, omit the `svlim` input and let the program compute and use the default value.

The m-file `totter_demo.m` includes an example that uses `r_colores.m`, with comments and explanations output to the command window.

*Header listing:*

```
%  
% R_COLORES Parameter covariance for colored residuals from linear  
% regression.  
%  
% Usage: [crb,crbo,y,p,sv] = r_colores(x,z,svlim);  
%  
% Description:
```

```
%  
% Computes the Cramér-Rao bounds for least squares regression  
% parameter estimation in the time domain, both conventionally  
% and accounting for the actual frequency content of the residuals.  
% The regression model is  $y = x^*p$ . The routine also computes  
% the least squares estimate of parameter vector  $p$ ,  
% where  $y = x^*p$  and  $y$  matches the measured quantity  $z$   
% in a least-squares sense. The singular values of the information  
% matrix, which indicate the conditioning of the least-squares  
% solution, are placed in output vector  $sv$ .  
%  
% Input:  
%  
%     x = matrix of column regressors.  
%     z = measured output vector.  
% svlim = minimum singular value ratio for matrix inversion (optional).  
%  
% Output:  
%  
%     crb = corrected Cramer-Rao bounds accounting for colored residuals.  
%     crbo = conventional Cramer-Rao bounds.  
%     y = model output vector.  
%     p = vector of parameter estimates.  
%     sv = vector of singular values of the information matrix.  
%
```

## 12.3 Model Structure Determination

### 12.3.1 swr.m

Function `swr.m` identifies general least-squares models from measured input-output data. All regressors to be considered for inclusion in the model are input as columns of the input matrix  $x$ . Candidate regressors can be swapped in and out of the model manually by the analyst. Least-squares parameter estimation for each candidate model is done using `lesq.m`. Several statistical diagnostics are computed at each step to help in deciding which regressors should be retained in the model. The `swr.m` routine can be used with either real or complex data, which means the same m-file can be used for time-domain data or frequency-domain data.

The `swr.m` program is interactive, and requires direction from the analyst as to which regressor to move in or out of the candidate model. Generally, if adding a particular regressor (column of  $x$ ) to the model decreases the fit error, increases  $R^2$ , decreases predicted squared error (PSE), and has a partial  $F$  ratio greater than the given cutoff value, the regressor should be retained in the model. The regressor to be added to the model at any point is the one with the highest squared partial correlation. Sometimes the addition of a regressor will render a previously selected regressor superfluous. This is indicated by the partial  $F$  value of the previously selected regressor becoming small. In that case, the two regressors

## MATLAB® SOFTWARE

391

(the newly added one and the one whose partial  $F$  became small) are probably highly correlated. This can be checked using `corx.m` or `regcor.m`.

Estimated parameter standard errors can be computed from the estimated covariance matrix `crb` by typing `serr=sqrt(diag(crb))`. In general, omit the `svlim` input and let the program compute and use the default value.

Function `swr.m` automatically includes a constant regressor, so it is not necessary to include a column of ones in the `x` regressor matrix. The regressor matrix `x` must be assembled in MATLAB®, outside of `swr.m`. For example, if spline regressors are desired, each candidate spline regressor must be included as a column in `x` before calling `swr.m`. Splines of various orders can be generated using `splgen.m`. Polynomial terms of arbitrary order can be generated using `reggen.m`. If it is not clear whether or not a particular regressor influences the dependent variable, include that regressor in the input matrix `x`, and `swr.m` will help to figure this out using statistical metrics, as described earlier. The PSE computed for the final model is conservative. Therefore, the squared error for prediction cases will generally be less than the value of PSE computed for the final model.

Function `swr.m` implements stepwise regression, not modified stepwise regression. This simply means that no regressors (linear or otherwise) are automatically included in the model, except for the constant term, which is always included. The estimated parameter vector `p` always has length equal to the number of candidate regressors (i.e., the number of columns of `x`), plus one (for the bias term). Output `pindx` is a vector of indices that specify which columns of `x` were selected for the final model.

The m-file `swr_demo.m` runs an example with comments and explanations output to the command window. Type `swr_demo` at the MATLAB® prompt to run the demonstration.

*Header listing:*

```
%  
% SWR Stepwise Regression.  
%  
% Usage: [y,p,crb,s2,xm,pindx] = swr(x,z,lplot,svlim);  
%  
% Description:  
%  
% Computes interactive stepwise regression estimates  
% of parameter vector p, estimated parameter covariance  
% matrix crb, model output y, model fit error variance  
% estimate s2, and the model regressor matrix xm, using  
% least squares with matrix inversion based on  
% singular value decomposition. The output y is computed  
% from y = xm*p(pindx). A constant term is included  
% automatically in the model as the last column in the  
% model regressor matrix xm. Optional input lplot controls  
% plotting, and optional input svlim specifies minimum singular  
% value ratio. This routine works for real or complex data.  
%
```

```
% Input:  
%  
%     x = matrix of column regressors.  
%     z = measured output vector.  
% lplot = plot flag (optional):  
%         = 0 for no plots (default)  
%         = 1 for plots  
% svlim = minimum singular value ratio  
%         for matrix inversion (optional).  
%  
% Output:  
%  
%     y = model output vector.  
%     p = vector of parameter estimates.  
%     crb = estimated parameter covariance matrix.  
%     s2 = model fit error variance estimate.  
%     xm = matrix of column regressors retained in the model.  
%     pindx = vector of parameter vector indices for  
%             retained regressors, indicating the columns  
%             of [x,ones(npts,1)] retained in the model.  
%
```

### 12.3.2 mof.m

Function **mof.m** identifies general multivariate polynomial models from measured input-output time-domain data. Important independent variables and combinations are determined automatically by generating orthogonal modeling functions directly from the measured data, using Gram-Schmidt orthogonalization, as described in Chapter 5. Function **mof.m** then uses these multivariate orthogonal functions in a least-squares formulation to determine model structure and associated model parameter values. Because of the method used to generate the multivariate orthogonal basis functions, it is possible to decompose each retained basis function into ordinary polynomial functions in the independent variables. Once this is done and the results are combined, the final model form is an ordinary multivariate polynomial in the independent variables. The covariance matrix estimate for the final model parameters is computed based on the error bounds computed for the parameters associated with the retained orthogonal modeling functions, using the fact that the decomposition from orthogonal modeling functions to ordinary polynomials is exact.

Program **mof.m** is a modification of the original program **offit.m** in SIDPAC version 1.1. The main difference in these programs is the method for generating and ordering the orthogonal functions, which is improved in **mof.m**. Generally, **mof.m** produces more accurate models with fewer terms than **offit.m**. Both **offit.m** and **mof.m** functions are included in SIDPAC version 2.0.

Each column of input matrix **x** should contain measurements of an independent variable that the analyst believes might influence the dependent variable **z**. Function **mof.m** automatically includes a constant regressor, so it is not necessary to include a column of ones in the **x** matrix of independent variables.

The input matrix **x** must be assembled in MATLAB® outside of **mof.m**. Input **nord** is a vector of integers with length equal to the number of independent

## MATLAB® SOFTWARE

393

variables (columns of  $x$ ), where each integer indicates the maximum order of the corresponding independent variable in any model term. Input `maxord` is an integer indicating the maximum order for each model term. For example, if model terms of order 4 or less were desired for independent variables vectors  $\alpha$  and  $\beta$ , with maximum order of  $\alpha$  in any model term equal to 2, and maximum order of  $\beta$  in any model term equal to 3, then  $x = [\alpha \beta]$ , `nord = [2 3]` and `maxord = 4`. These choices will allow orthogonalized terms based on  $\alpha^2 \beta^2$ ,  $\alpha^2$ , or  $\alpha\beta^3$ , among others, but not  $\alpha^2 \beta^3$  or  $\beta^4$ .

Input values for `nord` and `maxord` are not critical. A good general choice is to use 4 for all values. Use higher values if a highly nonlinear relationship is expected, and lower values to force a simpler model. If the values chosen are too high, the algorithm will generate more orthogonal functions, which will take more execution time, but the best model in terms of minimum predicted square error (PSE) will be identified just the same.

If independent variable 2 has only 4 different values in the measured data (i.e., column 2 of  $x$  has only 4 unique values), the highest order for that independent variable (corresponding to the second element of input `nord`) cannot be higher than 3. This is a fundamental information limitation, similar to the fact that 4 distinct points are fit exactly with a third-order polynomial.

If it is not clear whether or not a particular independent variable influences the dependent variable, include that independent variable in the input matrix  $x$ , and `mof.m` will figure it out automatically. The price for this is more execution time.

The number of orthogonal functions chosen should generally correspond to the minimum PSE, as shown on the display. Occasionally, with real data, two or three local minima will occur. In such cases, it is generally best to choose the lowest number of orthogonal functions (i.e., the first local minima), which results in a simpler model. Resist the temptation to include more orthogonal function for a tighter function fit. Although the function fit for the given data points may be better with more orthogonal functions, the prediction capability deteriorates as higher numbers of orthogonal functions are chosen. In general, stick with the number of orthogonal functions that gives minimum PSE. Function `mof.m` will use the orthogonal function model corresponding to minimum PSE if the program is run in automatic mode (`auto=1`), or if the user just enters a carriage return when asked how many orthogonal functions to keep in manual mode (`auto=0`). Always choose the number of orthogonal functions to be equal to or less than the orthogonal function number shown on the screen for orthogonality intact. The orthogonal functions are computed sequentially, and when the later orthogonal functions are no longer orthogonal to those already generated, the information in the independent variables has been exhausted. The practical result is that the orthogonal functions generated beyond when the orthogonality remains intact cannot be properly decomposed into ordinary polynomials. This shows up as a mismatch in the model fit error using orthogonal functions compared with using ordinary polynomials. The mismatch can be seen in the last information displayed before the program exits.

The quantities `rmsd` and `drmsd` displayed on the screen when `mof.m` runs are the root-mean-square deviation of the measured dependent variable from the model output, and the change in this quantity as each orthogonal function is added to the model, respectively. The generated orthogonal functions are

added to the model in order of most effective to least effective in reducing `rmsd`. This can be seen from the `rmsd` and `drmsd` data.

The most compact models are generated when the independent variables in the matrix of independent variables `x` are arranged as most important to least important left to right by column. For example, most aerodynamic modeling would have angle of attack in the first column, followed by perhaps a control surface deflection in the second column, etc. Input parameter `ivar` should usually be omitted. This parameter is used when it is desired that every term in the model include at least one power of a particular independent variable. For example, if sideslip angle is independent variable number 1 (i.e., in the first column of input matrix `x`), then setting `ivar=1` will force every term in the resulting model to contain sideslip angle to at least the first power. This is useful if the dependent variable is always zero when the sideslip angle is zero, because the model will then always be exactly correct for zero sideslip angle.

The PSE computed for the final model is conservative for the default `sig2`. Therefore, the squared error for prediction cases will generally be less than the output value `pse` for the final model. Estimated parameter standard errors can be computed from the estimated parameter covariance matrix `crb` by typing `serr=sqrt(diag(crb))`.

*Header listing:*

```
%  
% MOF Model structure determination and parameter estimation using  
% orthogonal functions.  
%  
% Usage: [y,p,ip,crb,pse,xp,a,ia,psi] =  
% mof(x,z,nord,maxord,sig2,auto,iplot,ivar,bvar,maxopt);  
%  
% Description:  
%  
% Identifies polynomial models from measured input-output data  
% using multivariate orthogonal functions generated by  
% Gram-Schmidt orthogonalization, ordered by dynamic programming.  
%  
% Input:  
%  
% x = matrix of independent variable vectors.  
% z = dependent variable vector.  
% nord = vector of maximum independent variable orders.  
% maxord = maximum order of each model term.  
% sig2 = dependent variable noise variance (optional)  
%       = 0 or omit this input for the default value.  
%       = -1 to use a linear model fit to estimate sig2.  
%       = a positive, independently estimated noise variance.  
% auto = flag indicating type of operation (optional):  
%       = 1 for automatic (no user input required).  
%       = 0 for manual (user input required, default).
```

```
% Iplot = flag for model structure metric plots (optional):  
%      =1 for plots.  
%      =0 to skip the plots (default).  
% ivar = x array column number of the independent variable  
%      to be used for the orthogonal function generation seed (optional)  
%      (use zero or omit this input for a constant function seed).  
% bvar = x array column number of the independent variable  
%      to be used for orthogonal blocking (optional)  
%      (use zero or omit this input to skip orthogonal blocking).  
% maxopt = maximum number of ordinary polynomial terms  
%           in the final model (optional).  
%  
% Output:  
%  
%   y = model output vector.  
%   p = parameter vector for the ordinary polynomial function expansion.  
%   ip = vector of integer indices for ordinary polynomial functions.  
%   crb = estimated parameter covariance matrix.  
%   pse = predicted squared error.  
%   xp = matrix of vector polynomial functions for the final model.  
%   a = parameter vector for the orthogonal function expansion.  
%   ia = vector of integer indices for the orthogonal functions.  
%   psi = matrix of vector orthogonal functions.  
%
```

## 12.4 Output-Error Parameter Estimation

### 12.4.1 oe.m

Function `oe.m` estimates dynamic model parameters from measured input-output time-domain data using the output-error method. The nonlinear optimization used by `oe.m` is modified Newton-Raphson, as described in Chapter 6. If this method fails to produce a decrease in the cost function, the optimizer automatically switches to the simplex method for 50 iterations and then goes back to modified Newton-Raphson. This procedure continues until convergence criteria for parameter estimates, cost function, and cost gradients, specified in Chapter 6, are satisfied. After each such convergence, the noise covariance matrix estimate is updated, and the process is repeated until all the convergence criteria given in Chapter 6 are satisfied, including parameter estimates, cost function, cost function gradients, and the estimated noise covariance matrix elements.

The modified Newton-Raphson approach has fast convergence, but sometimes diverges when far from the solution because of inaccurate cost gradient information, whereas the simplex method always moves toward the solution, but slowly. The optimization approach used in `oe.m` combines the advantages of both, and has been found to work well in practice. The optimization can also be controlled manually by the user by setting input `auto=0`. In this case, the user can specify the number of modified Newton-Raphson steps to be taken, and when the noise covariance matrix estimate is updated.

The first input to `oe.m`, is a user-defined dynamic model file that can be of arbitrary complexity, but must have the following call syntax:  $y = \text{dsname}(p, u, t, x_0, c)$ . The user-defined function `dsname.m` contains the dynamic model definition and the MATLAB® code to compute outputs  $y$  based on a candidate parameter vector  $p$ , and the other inputs. The input `dsname` to `oe.m` should be a string containing the name of the user-defined dynamic model function, without its `.m` extension. For example, if the user-defined dynamic model file is `tlonss.m`, with the required calling syntax (i.e., function definition statement)  $[y, x] = \text{tlonss}(p, u, t, x_0, c)$ , then the first input to `oe.m` should be '`tlonss`', including the single quotes. Note that extra variables can be output from the dynamic model file (e.g.,  $x$  for `tlonss.m`), but the first variable must be the computed output of the dynamic model function. The output  $y$  must correspond to the measured outputs in the columns of  $z$ , so that  $y$  and  $z$  have the same dimensions. There are several examples of the user-defined dynamic model files included with SIDPAC, including `tlonss.m` (longitudinal state-space dynamic model with dimensional parameters), `tlontf.m` (longitudinal transfer function model), `nldyn.m` (nonlinear dynamic model with nondimensional parameters), and `tlatss.m` (lateral state-space dynamic model with dimensional parameters), among others, for time-domain analysis.

The parameters in the initial parameter vector  $p_0$ , which is the starting point for the nonlinear optimization, are also treated as *a priori* parameter estimates with covariance matrix  $\text{crb}_0$ , but only if the input  $\text{crb}_0$  is provided. This amounts to using the same input  $p_0$  for two different roles. However, using *a priori* parameter estimates as starting values usually works well. If some of the parameters in  $p_0$  should not be treated as *a priori* estimates, then their associated diagonal elements in  $\text{crb}_0$  can be set to a large value, e.g.,  $10^6$ , which indicates that those elements in  $p_0$  contain no *a priori* information for parameter estimation. If all the elements in  $p_0$  are simply starting values, then input  $\text{crb}_0$  should be omitted.

As with any nonlinear optimization, the starting values of the parameters in  $p_0$  are important. Function `oe.m` will always try to take a modified Newton-Raphson step first, but if that step results in an increased cost, the program automatically switches to a slower, but more robust simplex method. This assures that the program will converge, as long as the problem has been set up properly (e.g., all model parameters can be estimated from the data, parameters have independent roles in modeling the output, etc.). The closer the initial parameter values are to the final answer, the faster that answer will be found, because more of the optimization steps will be modified Newton-Raphson steps. The price for poor starting values is a longer run time for `oe.m`. For most practical aircraft system identification problems, it is not very hard to choose starting values that will lead to convergence to the global minimum of the cost function. However, it is also possible to choose starting values so poor that the optimizer will not converge at all. This is simply the nature of multidimensional nonlinear optimization problems.

In general, let the program compute `d1`, `svlim`, and `auto` by omitting these inputs from the `oe.m` command line. The vector `d1` controls the size of finite central difference perturbations for sensitivity calculations, and the quantity `svlim` can be used to address collinearity problems in some cases. Input parameter `auto` can be used to manually control the optimization process, as mentioned earlier.

## MATLAB® SOFTWARE

397

Measured inputs  $u$  and outputs  $z$  must be assembled in MATLAB® outside of oe.m. The model structure to be used in oe.m is defined by the user in the model file, as described earlier. Normally, this model structure is chosen from analysis using swr.m or mof.m, or by analysis of residuals, experience, or judgment.

The m-file totter\_demo.m includes a demonstration of the use of nldyn.m, which implements a nonlinear dynamic model with nondimensional aerodynamic parameters. The m-files oe\_lon\_demo.m and oe\_lat\_demo.m are examples for longitudinal and lateral cases, respectively, with comments and explanations output to the command window. These examples use flight data from the NASA Twin Otter aircraft and demonstrate the use of linear state-space models with dimensional derivatives.

*Header listing:*

```
%  
% OE Output-error parameter estimation in the time domain.  
%  
% Usage: [y,p,crb,rr] = oe(dsname,p0,u,t,x0,c,z,auto,crb0,del,svlim);  
%  
% Description:  
%  
% Computes the output-error estimate of parameter vector p,  
% the Cramér-Rao bound matrix crb, the discrete noise  
% covariance matrix rr, and the model output y using  
% modified Newton-Raphson optimization.  
% The dynamic system is specified in an m-file or mex-file  
% named dsname. Inputs crb0, auto, del, and svlim are optional.  
%  
% Input:  
%  
% dsname = name of the file that computes the model outputs.  
% p0 = initial vector of parameter values.  
% u = input vector or matrix.  
% t = time vector.  
% x0 = state vector initial condition.  
% c = constants passed to dsname.  
% z = measured output vector or matrix.  
% auto = flag indicating type of operation:  
%       = 1 for automatic (no user input required, default).  
%       = 0 for manual (user input required).  
% crb0 = parameter covariance matrix for p0 (optional).  
% del = vector of parameter perturbations  
%       in fraction of nominal value (optional).  
% svlim = minimum singular value ratio for matrix inversion (optional).  
%  
% Output:  
%  
% y = model output vector or matrix.  
% p = vector of parameter estimates.
```

% crb = estimated parameter covariance matrix.  
% rr = discrete measurement noise covariance matrix estimate.  
%

### 12.4.2 colores.m and m\_colores.m

Functions `colores.m` and `m_colores.m` compute corrected parameter covariance matrices by post-processing results from output-error parameter estimation routine `oe.m`. The correction, discussed in Chapter 6, accounts for the practical fact that the output-error residuals are colored, not white, as assumed in the output-error maximum likelihood formulation. Corrected parameter standard errors from `colores.m` or `m_colores.m` are consistent with the scatter in parameter estimates from repeated flight-test maneuvers, and therefore accurately represent estimated parameter uncertainty.

Function `m_colores.m` is a vectorized version of `colores.m`, so `m_colores.m` runs faster than `colores.m`. The results from both functions are identical within numerical round-off error. The code should be run once at the end of the analysis, when the parameter estimates from the output-error method are judged to be acceptable. Inputs to `colores.m` and `m_colores.m` are the same as for `oe.m`, except that input `p` to `colores.m` and `m_colores.m` is the final parameter estimate `p` from `oe.m`.

In general, let the program compute `del` by omitting this input from the command line. The vector `del` controls the size of finite central difference perturbations for sensitivity calculations. Output of `m_colores.m` or `colores.m` includes the conventional Cramér-Rao bounds `crbo` and the corrected Cramér-Rao bounds `crb`. The `crbo` output should match the Cramér-Rao bounds from conventional output-error parameter estimation (`oe.m`), within round-off error.

The model structure to be used in `m_colores.m` or `colores.m` must be the same as that used for the parameter estimation in `oe.m`, and is defined by the user in a separate m-file or mex-file. The m-files `oe_lion_demo.m` and `trotter_demo.m` include the use of `m_colores.m`, with comments and explanations output to the command window.

*Header listing:*

```
%  
% M_COLORES Vectorized version of colores.m.  
%  
% Usage: [crb,crbo] = m_colores(dsname,p,u,t,x0,c,z,del);  
%  
% Description:  
%  
% Computes the Cramér-Rao bounds for maximum likelihood  
% estimation both conventionally and accounting for  
% the actual frequency content of the residuals.  
% The dynamic system is specified in the file named dsname.  
% Input del is optional. This routine is vectorized  
% for increased execution speed. Results are the same  
% as for the slower routine, colores.m.  
%
```

```
% Input:  
%  
% dsname = name of the file that computes the model outputs.  
% p = vector of parameter values.  
% u = input vector or matrix.  
% t = time vector.  
% x0 = state vector initial condition.  
% c = constants passed to dsname.  
% z = measured output vector or matrix.  
% del = vector of parameter perturbations in  
%       fraction of nominal value (optional).  
%  
% Output:  
%  
% crb = corrected Cramér-Rao bounds accounting for colored residuals.  
% crbo = conventional Cramér-Rao bounds.  
%
```

## 12.5 Frequency Domain

### 12.5.1 fint.m

Function fint.m computes the finite Fourier integral for arbitrary frequencies. The finite Fourier integral is computed based on the chirp  $z$ -transform, with cubic interpolation applied to the integrand for high accuracy.

Program fint.m expects the input data  $h$  to be sampled at regular intervals  $\Delta t$ , corresponding to the time points in  $t$ . The fint.m routine is vectorized, so the measured time-domain data  $h$  can be either a single column vector or a matrix of column vectors. The frequencies selected for the Fourier integral in the input vector  $w$  can have arbitrary resolution on the interval  $[0, \pi/\Delta t)$  rad/sec.

Use of fint.m is demonstrated in the SIDPAC m-files tfest\_demo.m and totter\_demo.m.

*Header listing:*

```
%  
% FINT High-accuracy finite Fourier integral for arbitrary frequencies.  
%  
% Usage: [val,valo,corfac,endcor,f,H] = fint(h,t,w);  
%  
% Description:  
%  
% Evaluates the finite Fourier integral whose integrand is  
%  
% h(t).*exp(-jw*t)  
%  
% over the time defined by vector t, for each (kth)  
% element of w, where w is a vector of frequencies in rad/s.  
% Multiplicative and endpoint corrections are made  
% using third-order Filon coefficients, so that
```

```
% the integral is evaluated with high accuracy.  
% Arbitrary frequency resolution is achieved using a  
% modified discrete Fourier transform called the chirp z-transform.  
%  
% Input:  
%  
% h = matrix of column vector time series in the Fourier integral.  
% t = time vector.  
% w = frequency vector for the chirp z-transform, rad/sec.  
%  
% Output:  
%  
% val = complex value of the Fourier integral.  
% valo = complex value of the Fourier integral without corrections.  
% corfac = multiplicative correction factor for the  
% discrete Fourier transform.  
% endcor = additive correction for the endpoints.  
% f = frequency vector for the chirp-z transform, Hz.  
% H = complex chirp-z transform vector corresponding to h.  
%
```

### 12.5.2 fdoe.m

Function fdoe.m estimates model parameters from measured input-output data using output-error in the frequency domain. This routine is the frequency-domain analog of oe.m, so the inputs, outputs, optimization algorithm, and printed display for fdoe.m are similar to those of oe.m. Measured inputs U and outputs Z are supplied to fdoe.m as frequency-domain data, so the Fourier transformation must be done outside of program fdoe.m. Program fint.m should be used for this purpose. The frequency vector w is an additional input to fdoe.m, and the model file for fdoe.m must have the following call syntax:  $y=dsname(p, U, w, x0, c)$ , where w is the vector of frequencies in rad/s for the Fourier transformation, and U is the transformed input. In general, SIDPAC programs use capital letters for frequency-domain data. The model structure to be used in fdoe.m is defined by the user in a separate m-file or mex-file (dsname), similarly to oe.m. Some example m-files for frequency-domain parameter estimation included in SIDPAC are flonss.m (longitudinal state-space model with dimensional parameters), flontf.m (longitudinal transfer function model), flatss.m (lateral state-space model with dimensional parameters), and flattf.m (lateral transfer function model).

Although fdoe.m was developed for output-error parameter estimation using frequency-domain data, the routine can also be used for equation-error parameter estimation, by modifying the user-defined m-file that specifies the model (dsname). The example model file called flonss.m includes code for both equation-error and output-error parameter estimation in the frequency domain using a state-space model for longitudinal short-period dynamics. The optimizer used in fdoe.m is the same one used in oe.m, but modified for frequency-domain data.

In general, let the program compute `del`, `svlim`, `crb0`, and `auto` by omitting these inputs from the `fdoe.m` command line. Matrix `crb0` is the covariance matrix associated with the initial parameter vector `p0`. Inputs `p0` and `crb0` are treated by `fdoe.m` in the same way that `oe.m` treats these inputs. Input parameter `auto` can be used to manually control the optimization process, in the same manner as for `oe.m`.

The m-file `fdoe_demo.m` runs an example with comments and explanations output to the command window. The example uses simulated data for the closed-loop pitch rate response of a supersonic transport aircraft. Type `fdoe_demo` at the MATLAB® prompt to run the demonstration.

*Header listing:*

```
%  
% FDOE Output-error parameter estimation in the frequency domain.  
%  
% Usage: [Y,p,crb,svv] = fdoe(dsname,p0,U,t,w,c,Z,auto,crb0,del,svlim);  
%  
% Description:  
%  
% Computes the output-error maximum likelihood estimate of  
% parameter vector p, Cramér-Rao bound matrix crb,  
% the power spectral density of the measurement noise svv,  
% and the model output Y in the frequency domain,  
% using modified Newton-Raphson optimization.  
% This routine implements the output-error formulation  
% in the frequency domain. The dynamic system is specified  
% in the file named dsname.  
% Inputs auto, crb0, del, and svlim are optional.  
%  
% Input:  
%  
% dsname = name of the file that computes the model outputs.  
% p0 = initial vector of parameter values.  
% U = input vector or matrix in the frequency domain.  
% t = time vector.  
% w = frequency vector, rad/sec.  
% c = vector or data structure of constants passed to dsname.  
% Z = measured output vector or matrix in the frequency domain.  
% auto = flag indicating type of operation (optional):  
%       =1 for automatic (no user input required, default).  
%       =0 for manual (user input required).  
% crb0 = parameter covariance matrix for p0 (optional).  
% del = vector of parameter perturbations  
%       in fraction of nominal parameter value (optional).  
% svlim = minimum singular value ratio for matrix inversion (optional).  
%  
% Output:
```

```
%  
% Y = model output vector or matrix in the frequency domain.  
% p = vector of parameter estimates.  
% crb = estimated parameter covariance matrix.  
% svv = power spectral density of the measurement noise.  
%
```

### 12.5.3 tfest.m

Function `tfest.m` estimates parameters in a transfer function model using equation error in the frequency domain and `fint.m` to compute high-accuracy finite Fourier integrals for arbitrary frequencies. Numerator order `nord` and denominator order `dord` must be supplied as input. The routine works only for single-input, single-output (SISO) transfer function parameter estimation.

Parameter estimation results from program `tfest.m` can be examined using Bode plots with program `bodecmp.m`. Program `tfest.m` uses linear regression with complex numbers to match the highest output derivative in the frequency domain. Frequency scaling is introduced so that the parameter estimation will not be weighted toward the higher frequencies. This can be modified by changing the regression problem formulation at the end of program `tfregr.m`. Program `tfest.m` expects the input data `u` and `z` to be sampled sequentially at regular intervals  $\Delta t$ , corresponding to the time points in `t`. The selected frequencies in the input frequency vector `w` can have arbitrary resolution on the interval  $[0, \pi/\Delta t]$  rad/sec.

The m-file `tfest_demo.m` shows an example with comments and explanations output to the command window. The example uses roll rate response data from the F-16 fighter aircraft simulation documented in Appendix D. Type `tfest_demo` at the MATLAB® prompt to run the demonstration.

*Header listing:*

```
%  
% TFEST Transfer function parameter estimation using equation-error in the  
% frequency domain.  
%  
% Usage: [y,num,den,p,crb,s2,zr,xr,f,cost] = tfest(u,z,t,nord,dord,w);  
%  
% Description:  
%  
% Estimates the parameters for a constant coefficient  
% single-input, single-output (SISO) transfer function  
% model with numerator order nord and denominator order dord.  
% The parameter estimation uses the equation-error formulation  
% with complex least squares regression in the frequency domain.  
%  
% Input:  
%  
% u = measured input vector time series.  
% z = measured output vector time series.  
% t = time vector.
```

```
% nord = transfer function model numerator order.  
% dord = transfer function model denominator order.  
% w = frequency vector, rad/sec.  
%  
% Output:  
%  
% y = model output vector time series.  
% num = vector of numerator parameter estimates in descending order.  
% den = vector of denominator parameter estimates in descending order.  
% p = vector of parameter estimates.  
% crb = estimated parameter covariance matrix.  
% s2 = equation-error model fit error variance estimate.  
% zr = complex dependent variable vector for the linear regression.  
% xr = complex regressor matrix for the linear regression.  
% f = frequency vector for the complex Fourier transformed data, Hz.  
% cost = value of the cost for the complex least-squares estimation.  
%
```

## 12.6 Real-Time Parameter Estimation

### 12.6.1 rlesq.m

Function `rlesq.m` computes least-squares linear regression model parameter estimates and covariance matrix estimates recursively. Although program `rlesq.m` implements recursive least squares, the inputs to the routine include all the measurements. Consequently, the input regressor matrix  $x$  includes all the data points for each regressor, where each row is one data point. The measured output  $z$  is a vector of all the measured output values.

The recursive identification loop is inside `rlesq.m`. To use the technique in a real-time application, the statements within the loop inside `rlesq.m` would be called at each time step when new measurements are available.

A typical range of possible values for the forgetting factor  $\text{ff}$  is  $0.95 \leq \text{ff} \leq 1.00$ . Input covariance matrix  $\text{crb0}$  quantifies the confidence in the initial parameter estimates  $p_0$ . For arbitrary starting values  $p_0$  or low confidence in the  $p_0$  values, corresponding diagonal elements of  $\text{crb0}$  should be set to a large positive number, e.g.  $10^6$ .

Since `rlesq.m` computes the fit error estimate  $s2$  recursively, and the parameter estimates are usually of low quality early in the maneuver, the final estimated covariance matrix  $\text{crb}$  computed recursively in `rlesq.m` will not agree exactly with the batch estimate from `lesq.m`. However, the computed inverse information matrix  $(X^T X)^{-1}$  is the same, whether computed by the batch method in `lesq.m` or recursively in `rlesq.m`. In addition, the final parameter estimates  $p$  from `rlesq.m` with  $\text{ff}=1$  should agree exactly with the batch parameter estimates from `lesq.m`, based on the same data.

The sequence of parameter estimates and standard error estimates are contained in outputs  $\text{ph}$  and  $\text{seh}$ , respectively. Each history is stored by column. To examine these, type `plot(t, ph(:, k))` or `plot(t, seh(:, k))` for histories associated with the  $k$ th parameter. Program `rlesq.m` uses the final value of the estimated parameter vector  $p$  to compute the model output  $y$ .

*Header listing:*

```
%  
% RLESQ Recursive least squares linear regression.  
%  
% Usage: [y,p,crb,s2,ph,crbh,s2h,seh,kh] = rlesq(x,z,ff,p0,crb0);  
%  
% Description:  
%  
% Computes recursive least squares estimate of parameter vector p,  
% where the assumed model structure is  $y = x^*p$ , ff is the data  
% forgetting factor, and z is the measured output.  
% Inputs specifying the forgetting factor ff, initial estimated  
% parameter vector p0, and initial estimated parameter covariance  
% matrix crb0 are optional.  
%  
% Input:  
%  
% x = matrix of column regressors.  
% z = measured output vector.  
% ff = data forgetting factor, usually  $0.95 \leq ff \leq 1.0$  (default = 1).  
% p0 = initial parameter vector (default = zero vector).  
% crb0 = initial parameter covariance matrix (default =  $10^{-6}$ *identity  
% matrix).  
%  
% Output:  
%  
% y = model output using final estimated parameters.  
% p = final estimated parameter vector.  
% crb = final estimated parameter covariance matrix.  
% s2 = final model fit error variance estimate.  
% ph = estimated parameter vector history.  
% crbh = estimated parameter covariance matrix history.  
% s2h = estimated model fit error variance history.  
% seh = estimated parameter standard error history.  
% kh = recursive least squares innovation weighting vector history.  
%
```

### 12.6.2 rtpid.m

Function rtpid.m computes real-time linear regression model parameter estimates and covariance matrix estimates using sequential least squares in the frequency domain.

Although function rtpid.m implements real-time parameter estimation, the inputs to the routine include all the measurements. The recursive Fourier transform and the sequential least-squares parameter estimation calculations are inside program rtpid.m. To use the technique in a real-time application, the recursive Fourier transform called inside rtpid.m would be called at each time step when new measurements are available. The sequential least squares would

be executed at desired times to produce real-time estimates of the parameters and their covariance matrix. The settings that determine the update rate for the recursive Fourier transform and the sequential least-squares parameter estimation are defined inside `rtpid.m`.

Program `rtpid.m` implements real-time parameter estimation for a single linear regression model, using all the regressors in the input regressor matrix  $x$ . The regressor matrix  $x$  should not include a vector of ones, because the bias terms are omitted in the frequency domain. The model will be a least-squares fit to the measured output  $z$  for  $lder=0$ , or to the first time derivative of  $z$  for  $lder=1$ . If  $lder=1$ , the time derivative of  $z$  is computed in the frequency domain inside `rtpid.m`.

The sequence of parameter estimates, model fit error variance estimates, and standard error estimates are stored in outputs  $ph$ ,  $s2h$ , and  $seh$ , respectively. The vector  $th$  contains the times when each estimate was made. Each history is stored by column. To examine these, type `plot(th,ph(:,k))` or `plot(th,seh(:,k))` for histories associated with the  $k$ th parameter, and `plot(th,s2h)` for the model fit error variance estimate history.

Program `rtpid.m` uses the final value of the estimated parameter vector to compute the model output  $y$ . All biases and slow trends are taken out of the problem when the analysis is done in the frequency domain, because these components correspond to very low frequencies that are omitted from the analysis. Consequently, there may be a bias and/or drift mismatch between measured output  $z$  and model output  $y$  in the time domain.

The m-file `rtpid_demo.m` contains a demonstration of sequential least-squares parameter estimation in the frequency domain using the longitudinal short-period dynamics of the F-16 aircraft, derived from the F-16 nonlinear simulation described in Appendix D. Fore and aft mouse movements in the figure window command the stabilator for the F-16 linear simulation, and the real-time parameter estimation is executed using the resulting noisy data. Plots of the real-time parameter estimation results appear during the simulation run. Type `rtpid_demo` at the MATLAB® prompt to run this demonstration.

*Header listing:*

```
%  
% RTPID Real-time parameter estimation using least squares in the  
% frequency domain.  
%  
% Usage: [y,p,crb,s2,ph,th,seh,s2h,Xh,Zh,f] = rtpid(x,z,t,w,lder,p0,crb0);  
%  
% Description:  
%  
% Real-time parameter estimation using sequential  
% least squares equation-error in the frequency domain.  
% Transformation of input time-domain data into the  
% frequency domain is done using a recursive  
% Fourier transform. Inputs specifying the derivative  
% option lder, initial estimated parameter vector p0,
```

```
% and initial estimated parameter covariance
% matrix crb0, are optional.
%
% Input:
%
% x = matrix of column regressors.
% z = measured output vector.
% t = time vector.
% w = frequency vector, rad/sec.
% lder = derivative flag:
%       = 0 to model output z (default)
%       =1 to model derivative of output z
% p0 = initial parameter vector (default = zero vector).
% crb0 = initial parameter covariance matrix (default = 10^-6* identity
%       matrix).
%
% Output:
%
% y = model output using final estimated parameters.
% p = final estimated parameter vector.
% crb = final estimated parameter covariance matrix.
% s2 = final model fit error variance estimate.
% ph = estimated parameter vector history.
% th = vector of times for parameter estimate history ph.
% seh = estimated parameter standard error history.
% s2_h = estimated model fit error variance history.
% Xh = regressor matrix Fourier transform history.
% Zh = measured output Fourier transform history.
% f = frequency vector, Hz.
```

## 12.7 Input Design

### 12.7.1 mksqw.m

Function `mksqw.m` generates arbitrary square waves of length  $T$  with sampling interval  $\Delta t$ . Input `npulse` is a row vector that represents the multiples of `tpulse` seconds to be used for each pulse of the square wave. For example, a doublet input with the basic pulse width equal to 1 s can be generated with `tpulse=1` and `npulse = [1, 1]`. To generate a 3-2-1-1 input with the basic pulse width equal to 1 s, set `tpulse=1` and `npulse=[3, 2, 1, 1]`. The pulses in the square wave input are adjacent and alternate in sign by default, but this can be changed with the `amp` input. For example, two pulses with alternating sign, amplitude 2, and 1-s duration, separated by 4 s, can be generated with `tpulse=1`, `npulse = [1, 4, 1]`, and `amp = [2, 0, -2]`. The second pulse, with zero amplitude, is automatically assigned a negative sign, because of the automatic alternating sign, so the third element of `amp` is negative to get a pulse with amplitude  $-2$ . If input `amp` is a scalar, the amplitude of all pulses are set equal to `amp`. The input `tdelay` specifies the time delay before the square wave begins.

After generating a desired square wave with mksqw.m, program ratelim.m can be used to implement rate limiting for a practical input.

*Header listing:*

```
%  
% MKSQW Creates multistep square wave inputs.  
%  
% Usage: [u,t] = mksqw(amp,tpulse,npulse,tdelay,dt,T);  
%  
% Description:  
%  
% Creates an alternating square wave input vector of length  
% T, with single pulse time tpulse, pulse amplitudes amp, and  
% individual integer pulse widths given by the elements of npulse.  
%  
% Input:  
%  
% amp = input amplitudes for each pulse.  
% tpulse = time for a single pulse, sec.  
% npulse = vector of integer pulse widths, e.g., npulse = [3 2 1 1].  
% tdelay = time delay before the square wave starts, sec.  
% dt = sampling interval, sec.  
% T = time length, sec.  
%  
% Output:  
%  
% u = alternating square wave vector.  
% t = time vector.  
%
```

### 12.7.2 mkfswp.m

Function mkfswp.m generates linear or logarithmic frequency sweeps of length T with sampling interval dt, covering the frequency range [wmin, wmax]. The logarithmic frequency sweep dwells longer on the lower frequencies than the linear frequency sweep. The result is more power at the lower frequencies for the logarithmic frequency sweep. This is generally advantageous when the dynamic modes of interest are in the lower part of the specified frequency band. Each frequency sweep ends automatically at the zero crossing closest to the final time T, so the input starts and ends at zero. This implements a perturbation input.

After generating a desired frequency sweep with mkfswp.m, program ratelim.m can be used to implement rate limiting for a practical input.

*Header listing:*

```
%  
% MKFSWP Creates linear or log frequency sweep inputs.  
%  
% Usage: [u,t,w] = mkfswp(amp,tdelay,wmin,wmax,dt,T,type);
```

```
%  
% Description:  
%  
% Creates a frequency sweep with amplitude amp  
% covering the frequency range between wmin and wmax inclusive.  
%  
% Input:  
%  
% amp = input amplitude.  
% tdelay = time delay before the frequency sweep starts, sec.  
% wmin = minimum frequency, rad/sec.  
% wmax = maximum frequency, rad/sec.  
% dt = sampling time, sec.  
% T = time length, sec.  
% type = frequency sweep type  
%      = 0 for logarithmic frequency sweep (default).  
%      = 1 for linear frequency sweep.  
%  
% Output:  
%  
% u = frequency sweep vector with amplitude amp covering  
%      the frequency range from wmin to wmax inclusive.  
% t = time vector.  
% w = vector of frequencies for each time step.  
%
```

### 12.7.3 *mkmsswp.m*

Function *mkmsswp.m* generates multiple orthogonal phase-optimized multisine sweep inputs. Each sweep has a minimized peak factor *pf* for a selected power spectrum *pwr* in the selected frequency band. The peak factor *pf* for a vector *y* is computed from the expression  $pf \equiv (\max(y) - \min(y))/(2\sqrt{2}\text{rms}(y))$ .

If the frequencies *fs* of the sinusoidal components are not specified, program *mksswp.m* uses the maximum frequency resolution possible for the sinusoidal components that compose the sweep. The only way to get finer frequency resolution for the sweep is to increase the time length *T*. Finer frequency resolution produces a richer and smoother power spectrum in the selected frequency band.

Program *mksswp.m* can generate multiple inputs that are mutually orthogonal in both the time domain and the frequency domain. This is advantageous for accurate modeling. The program uses flat, uniform power spectra if the power spectra are not specified in *pwr*. Elements in each column of *pwr* indicate a fraction of the total signal power for the corresponding component sinusoids, whose frequencies are specified in *fs*. The *k*th column of *fs* and *pwr* correspond to the *k*th column of the output matrix *y*. Consequently, the sum of each column of *pwr* must equal 1. This provides the capability to create multisine inputs with arbitrary power spectra and minimum peak factors.

Phase angles of the sweeps with minimized peak factor are adjusted so that each input begins and ends at zero. This implements a perturbation input.

After generating a multisine sweep with `mksswp.m`, program `ratelim.m` can be used to implement rate limiting for a practical input.

*Header listing:*

```
%  
% MKMSSWP Creates orthogonal multisine inputs with minimized relative  
% peak factor.  
%  
% Usage: [u,t,pf,f,M,ph] = mkmsswp(amp,fmin,fmax,dt,T,m,fu,pwr);  
%  
% Description:  
%  
% Generates m orthogonal multisine sweeps of time  
% length T with sample rate dt, by combining  
% equally-spaced frequencies between fmin and fmax  
% in Hz, using a power spectrum defined by pwr.  
% Frequencies for each input can be supplied in the  
% corresponding columns of fu. Otherwise, the frequencies  
% are assigned sequentially to each input and the number  
% of harmonic components is maximized. The  
% frequencies then depend on the maneuver time T, the  
% number of inputs m, and the selected  
% frequency band [fmin,fmax]. The signals  
% have minimized relative peak factor for a  
% given spectrum. Outputs are the orthogonal  
% multisine sweeps in columns of u, corresponding  
% time vector t, the frequencies for the harmonic  
% components f, and the relative peak factors:  
%  
% pf = (max(u)-min(u))/(2*sqrt(2)*rms(u))  
%  
% M is a vector containing the number of frequencies  
% in each input, and ph is a matrix of the phase  
% shifts, optimized for minimum peak factors and  
% shifted so that the inputs begin and end at zero.  
% If amp is a scalar, all m signals have that amplitude;  
% otherwise, each element of amp specifies the amplitude  
% of the corresponding column of u.  
% Inputs m, fu, and pwr are optional.  
%  
% Reference:  
%  
% Morelli, E.A., "Multiple Input Design for Real-Time  
% Parameter Estimation in the Frequency Domain,"  
% Paper REG-360, 13th IFAC Symposium on System  
% Identification, Rotterdam, The Netherlands, August 2003.
```

```
%  
% Input:  
%  
% amp = input amplitude(s).  
% fmin = minimum frequency, Hz.  
% fmax = maximum frequency, Hz.  
% dt = sampling interval, sec.  
% T = time length, sec.  
% m = number of signals to generate (default = 1).  
% fu = matrix with columns containing frequencies for each input, in Hz  
% (default = maximum frequency resolution)  
% pwr = matrix with columns defining power spectrum for each input.  
% The sum of the elements in each column of pwr must equal 1  
% (default = flat power spectrum).  
%  
%  
% Output:  
%  
% u = orthogonal multisine sweep(s).  
% t = time vector.  
% pf = relative peak factor(s).  
% f = frequencies of harmonic components, Hz.  
% M = number of harmonic components for each column of u.  
% ph = phase angles of harmonic components, rad.  
%
```

#### 12.7.4 mkrdm.m

Function mkrdm.m generates white or colored random noise sequences. This routine can be used to generate a Gaussian noise vector with broadband frequency content (white noise), band-limited frequency content (band-limited noise), or colored noise, which is a combination of white noise and band-limited noise. The frequency content of the noise is controlled by inputs bw and pwrf. Omitting bw and pwrf gives white noise. Setting pwrf=1 gives band-limited noise with bandwidth bw. Any pwrf between 0 and 1 will give colored noise. The value of pwrf equals the fraction of the total power that is band-limited noise in the frequency interval [0 , bw] Hz.

After generating random noise with mkrdn.m, program ratelim.m can be used to implement rate limiting for a practical input.

*Header listing:*

```
%  
% MKRDM Creates random white or colored noise inputs.  
%  
% Usage: [u,t,ons] = mkrdm(amp,tdelay,tfinal,dt,T,m,bw,pwrf);  
%  
% Description:  
%
```

```
% Creates m Gaussian random noise input vectors of length
% T, with root-mean-square amplitude amp. If optional
% inputs bw and pwr are specified, the noise is colored with
% a band-limited component in the frequency interval [0,bw] Hz,
% and pwr is the fraction of the total noise power that is band-limited:
%
%     total noise power = band-limited power + wide-band power
%
% Inputs m, bw, and pwr are optional. Defaults for m, bw,
% and pwr give a single vector of Gaussian white noise.
%
% Input:
%
%     amp = input amplitude.
%     tdelay = time delay before the random noise input starts, sec.
%     tfinal = quiet time at the end of the random noise input, sec.
%     dt = sampling interval, sec.
%     T = time length, sec.
%     m = number of random noise vectors to be generated (default = 1).
%     bw = bandwidth of the band-limited noise component, Hz (default = 1/
%          (2*dt)).
%     pwr = fraction of total noise power that is band-limited, [0,1] (default = 0).
%
% Output:
%
%     u = matrix of m random noise column vectors.
%     t = time vector.
%     ons = matrix of m original noise sequence column vectors.
%
```

## 12.8 Data Compatibility

As discussed in Chapter 10, data compatibility analysis can be done using output-error parameter estimation in the time domain. It follows that the output-error parameter estimation routine *oe.m*, described earlier, is applicable to this problem. For data compatibility analysis, the user-defined model file integrates the kinematic equations, with instrumentation errors introduced as the unknown parameters. The model equations, inputs, outputs, and parameters are different for data compatibility than for aerodynamic parameter estimation, but these two output-error parameter estimation problems are otherwise conceptually identical. A general model file called *dcmp.m*, described next, has been developed for data compatibility analysis using *oe.m*.

### 12.8.1 *dcmp.m*

Function *dcmp.m* is a model file that computes data compatibility model outputs for output-error parameter estimation in the time domain. Function *dcmp.m* is used as the user-defined model file for *oe.m*, so that the first input

to `oe.m` should be ‘`dcmp`’, including the single quotes. The syntax of the user-defined m-file or mex-file (like `dcmp.m`) must be the same as defined earlier for other model files used with `oe.m`.

Output-error parameter estimation goes much faster if a mex-file based on FORTRAN or C code is used for the user-defined model file. The function `dcmp.m` can use C mex-files for the kinematic equations (`dcmp_eqs.c`) and the numerical integration (`adamb3.c`). On a PC, the command issued at the MATLAB® prompt for compiling `dcmp_eqs.c` into a mex-file is `mex dcmp_eqs.c`, and similarly for `adamb3.c`. Compiling this or any other C mex-file requires a C compiler. Once the user-defined mex-files are compiled, the operation and call syntax of a user-defined mex-file are exactly the same as for a user-defined m-file. Maximum execution speed is achieved when both `dcmp_eqs.c` and `adamb3.c` are first compiled into mex-files.

The model structure for data compatibility analysis (i.e., which instrumentation error parameters should be estimated) can be determined by the analyst, based on deficiencies in the comparison of measured quantities (airspeed, angle of attack, sideslip angle, Euler roll, pitch, and heading angles, and altitude) with reconstructed values obtained by integrating derivative-type measurements (linear accelerations and body-axis angular rates), setting all instrumentation error parameters equal to zero. The demonstration scripts `compat_lon_demo.m` and `compat_lat_demo.m` show plots that can be used for this preliminary step. These demonstration examples use `dcmp.m` with `oe.m` for data compatibility analysis on flight data from the NASA F-18 High Alpha Research Vehicle (HARV).

It is not uncommon for some trial-and-error to be necessary to arrive at an appropriate instrumentation error parameter set. Once found, though, this model structure usually does not change over the course of a flight-test program, unless the instrumentation is changed. The instrumentation error parameters and the manner in which they are used is specified in the user-defined model file.

The outputs to be matched and instrumentation error parameters to be estimated in `dcmp.m` are specified in `dcmp_psel.m`. Setup for the data compatibility problem can be done by setting flags inside `dcmp_psel.m` and then typing `dcmp_psel` at the MATLAB® prompt to implement the settings, followed by running `oe.m` with `dcmp.m` as the user-defined model file. The entire procedure can also be done using the SIDPAC GUI.

The m-files `compat_lon_demo.m` and `compat_lat_demo.m` contain longitudinal and lateral data compatibility examples, with comments and explanations shown in the command window. Type `compat_lon_demo` or `compat_lat_demo` at the MATLAB® prompt to run the demonstrations. For the longitudinal case, measured values are substituted for the lateral states. This is done because the longitudinal maneuver has very little excitation in the lateral states, so any instrumentation error parameters associated with the lateral motion cannot be estimated from this data. Similarly, for the lateral case, measured values are substituted for the longitudinal states. If the maneuver has both longitudinal and lateral excitation (e.g., a windup turn), then it is possible to estimate the values of all the instrumentation error parameters from a single maneuver.

*Header listing:*

```
%  
% DCMP Computes reconstructed outputs for data compatibility analysis.  
%  
% Usage: [y,x] = dcmp(p,u,t,x0,c);  
%  
% Description:  
%  
% Model m-file for data compatibility analysis  
% using output-error parameter estimation.  
%  
% Input:  
%  
% p = vector of parameter values.  
% u = matrix of column vector inputs  
% = [ax,ay,az,p,q,r,u,v,w,phi,the,psi].  
% t = time vector.  
% x0 = initial state vector.  
% c = cell structure (defined in dcmp_psel.m):  
% c{1} = p0c = vector of initial parameter values.  
% c{2} = ipc = index vector to select the parameters to be estimated.  
% c{3} = ims = index vector to select measured states.  
% c{4} = imo = index vector to select model outputs.  
%  
% Output:  
%  
% y = matrix of column vector model outputs  
% = [V,alpha,beta,phi,the,psi].  
% x = matrix of column vector model states  
% = [u,v,w,phi,the,psi].  
%
```

### 12.8.2 rotchk.m

Function `rotchk.m` checks the compatibility of measured data for body-axis angular rates and Euler angles. With some modification, this routine could be used as a user-defined model file (`dsname`) for `oe.m`, to estimate instrumentation error parameters for the body-axis angular rates and Euler angle sensors. Function `rotchk.m` computes Euler angle time histories based on the rotational kinematic differential equations with body-axis angular velocity inputs and initial conditions for the Euler angles. Input data come from the flight-test data matrix `fdata`, with columns defined in `SIDPAC_Data_Channels.pdf`. The relationships included in `rotchk.m` are a subset of the entire set of equations used for data compatibility.

A good way to evaluate the output from `rotchk.m` is to use `cmpplt.m` to compare reconstructed Euler angle time histories computed from `rotchk.m` with the measured Euler angle data. This comparison can be used to get an idea of which instrumentation error parameters should be estimated using `oe.m`.

For example, if `phi` computed from `rotchk.m` drifts off linearly with time, compared with measured `phi`, that would indicate that perhaps one or more of the measured body-axis angular rates has a bias error. A bias integrated over time produces a drift. If `phi` computed from `rotchk.m` differed from measured `phi`, such that the size of the difference correlated with the absolute value of measured `phi`, that would indicate that measured `phi` may have a scale factor error.

*Header listing:*

```
%  
% ROTCHK Computes reconstructed Euler angles for data compatibility  
% analysis.  
%  
% Usage: [phi,the,psi] = rotchk(fdata);  
%  
% Description:  
%  
% Integrates body-axis rotational kinematic equations  
% to obtain reconstructed Euler angle time histories.  
% This routine is used to assess compatibility of the  
% measured data from sensors for rigid-body rotational motion.  
%  
% Input:  
%  
% fdata = flight test data array in standard configuration.  
%  
% Output:  
%  
% phi = reconstructed Euler roll angle, deg.  
% the = reconstructed Euler pitch angle, deg.  
% psi = reconstructed Euler yaw angle, deg.  
%
```

### 12.8.3 `airchk.m`

Function `airchk.m` checks the compatibility of measured air-relative velocity data, translational acceleration, and body-axis angular rates. The code computes reconstructed time histories of airspeed, angle of attack, and sideslip angle, based on the translational kinematic differential equations with translational accelerations and body-axis angular rates as inputs, and initial conditions for the body-axis velocity components `u`, `v`, and `w`. Input data come from the flight-test data matrix `fdata`, with columns defined in `SIDPAC_Data_Channels.pdf`. The relationships included in `airchk.m` are a subset of the entire set of equations used for data compatibility.

A good way to evaluate the output from `airchk.m` is to use `cmpplt.m` to compare reconstructed time histories of airspeed, angle of attack, and sideslip angle computed from `airchk.m` with measured air-relative velocity data. This comparison can indicate which instrumentation error parameters should be estimated using `oe.m`, as described earlier for `rotchk.m`.

*Header listing:*

```
%  
% AIRCHK Computes reconstructed air-relative velocity data for data compatibility analysis.  
%  
% Usage: [vt,beta,alfa,u,v,w] = airchk(fdata);  
%  
% Description:  
%  
% Integrates body-axis translational kinematic equations  
% to obtain reconstructed air-relative velocity data. This routine  
% is used to assess compatibility of the measured data from  
% sensors for rigid-body translational motion.  
%  
% Input:  
%  
% fdata = flight test data array in standard configuration.  
%  
% Output:  
%  
% vt = reconstructed airspeed, ft/sec.  
% beta = reconstructed sideslip angle, deg.  
% alfa = reconstructed angle of attack, deg.  
% u = reconstructed x body-axis velocity component, ft/sec.  
% v = reconstructed y body-axis velocity component, ft/sec.  
% w = reconstructed z body-axis velocity component, ft/sec.  
%
```

## 12.9 Data Analysis

### 12.9.1 smoo.m

Function **smoo.m** implements global Fourier smoothing with an optimal Wiener filter, as described in Chapter 11. The routine makes a plot of Fourier sine series coefficient magnitudes and allows the user to manually choose the frequency cutoff for global smoothing. It is also possible to run **smoo.m** in automatic mode, where the software identifies frequency domain models for signal and noise, using  $k^{-3}$  and a constant, respectively, where  $k$  is the frequency index. These identified models are then used to find the frequency cutoff and construct the Wiener filter.

Function **smoo.m** is vectorized, so the measured  $z$  can be either a single column vector or a matrix of column vectors. Filter cutoff frequencies  $f_{\text{c}}$  can be determined automatically based on the data, or chosen manually by the analyst based on frequency-domain data plots. As currently implemented, the filters are low pass, so the deterministic signal is assumed to be in a frequency range  $[0, f_{\text{c}}]$  Hz.

Function **smoo.m** can be used to find smoothed time series from noisy measured data, and to estimate noise variances. The measurements  $z$  must be made sequentially at regular intervals, because of the use of Fourier series.

Input `fcep` applies only to the endpoints. The value chosen therefore is generally not critical, and the default value suffices in most cases. The value of `fcep` should be set higher for high-frequency content in the deterministic signal near the endpoints. To avoid any problems with endpoints, the best approach is to include extra data points at the beginning and end of `z`, run `smoo.m`, then discard the extra data points at the beginning and end from the result. Setting input `auto=0` allows the analyst to make the decision on where the filter cutoff frequency should be. Because of the formulation of the Fourier smoothing problem, and also because of the use of the Wiener filter, the algorithm is robust to the choice of cutoff frequency for the filter. This means that virtually the same filtering will occur for a range of cutoff frequencies near the best one. Using the frequency-domain data plots, it is only necessary to place the filter cutoff frequency between the large Fourier sine series components and the small components. Since the decay of the Fourier sine series for the deterministic signal goes with the inverse cube of the number of terms in the series, this separation is easy to make visually.

In practice, the data occasionally will show more than one group of relatively large Fourier sine series components. Each of these corresponds to deterministic signal components. For flight-test data, the lowest-frequency group of large magnitude Fourier coefficients corresponds to the deterministic rigid-body response, whereas the higher-frequency groups of large-magnitude Fourier coefficients are often from a structural response.

Manual operation of `smoo.m`, (`auto=0`), allows the analyst to get a better idea of the frequency content of the measured data, and more control over the final results. The automatic method is faster and less trouble to use, but does not provide the same insight.

*Header listing:*

```
%  
% SMOO Optimal global Fourier smoothing.  
%  
% Usage: [zs,fco,rr,b,f,wf,gv,sigab,nseab]=smoo(z,t,fcep,lplot,auto);  
%  
% Description:  
%  
% Computes smoothed time series and noise covariance matrix  
% estimates from measured data, using optimal Fourier smoothing.  
% The analyst can select signal cut-off frequency  
% for the deterministic signal, based on the Lanczos sine series  
% spectrum. Inputs fcep, lplot, and auto are optional.  
%  
% Input:  
%  
%     z = vector or matrix of measured time series.  
%     t = time vector.  
%     fcep = cutoff frequency for low-pass filtering  
%           of the endpoints, Hz (default = 1).  
%     lplot = plot flag:
```

```
%      = 1 for smoothing plots.  
%      = 0 to skip the plots (default).  
% auto = flag indicating type of operation:  
%      = 1 for automatic (no user input required, default).  
%      = 0 for manual (user input required).  
%  
% Output:  
%  
% zs = vector or matrix of smoothed time series.  
% fco = scalar or vector of cutoff frequencies, Hz.  
% rr = scalar or matrix discrete noise covariance estimate.  
% b = vector or matrix of Fourier sine series coefficients  
%      for detrended time series reflected about the origin.  
% f = vector of frequencies for the Fourier  
%      sine series coefficients, Hz.  
% wf = vector or matrix of filter weights in the frequency domain.  
% gv = vector or matrix of measured time series  
%      with endpoint discontinuities removed.  
% sigab = vector or matrix frequency-domain model of  
%      the absolute Fourier sine coefficients  
%      for the deterministic part of the measured time series.  
% nseab = scalar or vector of the constant frequency-domain  
%      model of the absolute Fourier sine coefficients  
%      for the random noise part of the measured time series.  
%
```

### 12.9.2 deriv.m

Function `deriv.m` computes a local smoothed numerical time derivative based on measured time series data. Function `deriv.m` is vectorized, so the input measured `z` can be either a single column vector or a matrix of column vectors. Function `deriv.m` assumes the measurements `z` are made sequentially at regular time intervals of size `dt`.

For very noisy data, the smoothed derivative at the endpoints of measured `z` can be degraded. To avoid any problems with endpoints, the best approach is to include 5–10 extra data points at the beginning and end of `z`, run `deriv.m`, and then discard the extra data points at the beginning and end from the result. If `deriv.m` is being used with `smoo.m` (or any other smoothing or filtering routine), it is best to run `deriv.m` first, then the smoother or filter. The final result is a very smooth and accurate numerical time derivative.

Program `nderiv.m` is a generalization of the algorithm in `deriv.m`, where the order of the local least-squares polynomial fit to the measured data, and the number of neighboring data points used in that fit, are integer parameters that can be changed in the `nderiv.m` code. The default values inside `nderiv.m` have been found to give good results in many flight-test data analysis cases. Using more neighboring points or a lower-order polynomial fit smoothes the local fit more and therefore removes high frequencies, and vice versa.

A related function called `lsmoo.m` does the same local smoothing as `nderiv.m`, but returns the local smoothed value instead of the time derivative. The result is a locally-smoothed time series.

*Header listing:*

```
%  
% DERIV Smoothed numerical differentiation.  
%  
% Usage: zd = deriv(z,dt);  
%  
% Description:  
%  
% Computes smoothed derivatives of measured time series  
% by differentiating a local quadratic least-squares fit  
% to the set of points consisting of each data point  
% and its four nearest neighboring points.  
%  
% Input:  
%  
% z = vector or matrix of measured time series.  
% dt = sampling interval, sec.  
%  
% Output:  
%  
% zd = vector or matrix of smoothed time derivatives.  
%
```

### 12.9.3 `xsmep.m`

Function `xsmep.m` is a local smoother for endpoints of a time series. This function is useful for estimating initial conditions from measured states and controls. Function `xsmep.m` uses a time convolution implementation of a low-pass filter on the measured time series and then extrapolates the smoothed points near the endpoint to obtain smoothed endpoint data, without using the measured data at the endpoint itself. This approach works well, especially when the data are very noisy.

Other routines that perform a similar function are `smep.m`, which is the same as `xsmep.m`, except the endpoint data are included; `csmep.m`, which uses a local cubic polynomial fit over a selected initial time period to compute the smoothed initial point; `zep.m`, which sets the endpoints to zero by removing the bias and a linear trend; and `szep.m`, which is the same as `zep.m`, except that smoothed endpoints from `xsmep.m` are used to remove the bias and linear trend. All of the functions mentioned can be used on matrices of column vector data, or on a single column vector of data.

*Header listing:*

```
%  
% XSMEP Local endpoint smoothing, excluding the endpoint data.  
%
```

```
% Usage: zsmep = xsmep(z,f,dt);  
%  
% Description:  
%  
% Smoothes the endpoints of a measured time  
% series z using a time convolution implementation  
% of a low-pass filter with cutoff frequency f for points  
% adjacent to the endpoints, then extrapolates the smoothed  
% adjacent points to obtain the endpoint estimates. This avoids  
% using the endpoints themselves in the smoothing operation, which  
% produces a better result when the endpoints are very noisy.  
%  
% Input:  
%  
% z = vector or matrix of measured time series.  
% f = low pass filter cutoff frequency, Hz.  
% dt = sampling interval, sec.  
%  
% Output:  
%  
% zsmep = vector or matrix of measured time series  
% with smoothed endpoints.  
%
```

#### 12.9.4 compfc.m

Function `compfcm` computes the nondimensional aerodynamic force coefficients, based on aircraft geometry and measured data arranged in standard format. The columns of input array `fdata` contain measured quantities specified in the document `SIDPAC_Data_Channels.pdf`. Measurement corrections for instrument positions must be done outside of `compfcm`, with results placed in the proper columns of `fdata`. Function `compfcm` assumes that thrust acts along the *x* body axis through the aircraft c.g.

*Header listing:*

```
%  
% COMPFC Computes non-dimensional force coefficients.  
%  
% Usage: [CX,CY,CZ,CD,CYw,CL,CT,phat,qhat,rhat] =  
% compfc (fdata,cbar,bspan,sarea);  
%  
% Description:  
%  
% Computes the non-dimensional force coefficients  
% and non-dimensional angular rates based  
% on measured flight data from input data array fdata.  
% Inputs cbar, bspan, and sarea can be omitted if  
% fdata contains this information.  
%
```

```
% Input:  
%  
% fdata = flight data array in standard configuration.  
% cbar = wing mean aerodynamic chord, ft.  
% bspan = wing span, ft.  
% sarea = wing area, ft2.  
%  
% Output:  
%  
% CX = non-dimensional body-axis X coefficient.  
% CY = non-dimensional body-axis Y coefficient.  
% CZ = non-dimensional body-axis Z coefficient.  
% CD = non-dimensional stability-axis lift coefficient.  
% CYw = non-dimensional wind-axis side force coefficient.  
% CL = non-dimensional stability-axis drag coefficient.  
% CT = non-dimensional thrust coefficient.  
% phat = non-dimensional roll rate.  
% qhat = non-dimensional pitch rate.  
% rhat = non-dimensional yaw rate.  
%
```

### 12.9.5 compmc.m

Function `compmc.m` computes the nondimensional aerodynamic moment coefficients, based on aircraft geometry and measured data arranged in standard format. The columns of input data array `fdata` contain measured quantities specified in the document `SIDPAC_Data_Channels.pdf`. Function `compmc.m` uses angular accelerations from the `fdata` array if those quantities are present (i.e., nonzero); otherwise, `compmc.m` computes the angular accelerations from the angular velocities using a local smoothed numerical differentiation (`deriv.m`). Function `compmc.m` assumes that thrust acts along the *x* body axis through the aircraft c.g.

A related function called `compfmc.m` uses both `compfc.m` and `compmc.m` to compute nondimensional aerodynamic force and moment coefficients, along with nondimensional rotational rates, and then stores the results in the appropriate columns of the standard data matrix `fdata`.

*Header listing:*

```
%  
% COMPNC Computes non-dimensional moment coefficients.  
%  
% Usage: [C1,Cm,Cn,phat,qhat,rhat,aa,iters] =  
%         compmc(fdata,cbar,bspan,sarea);  
%  
% Description:  
%  
% Computes the non-dimensional moment coefficients  
% and non-dimensional angular rates based
```

```
% on measured flight data from input data array fdata.  
% Inputs cbar, bspan, and sarea can be omitted if  
% fdata contains this information.  
%  
% Input:  
%  
% fdata = flight data array in standard configuration.  
% cbar = wing mean aerodynamic chord, ft.  
% bspan = wing span, ft.  
% sarea = wing area, ft2.  
%  
% Output:  
%  
% C1 = non-dimensional rolling moment coefficient.  
% Cm = non-dimensional pitching moment coefficient.  
% Cn = non-dimensional yawing moment coefficient.  
% phat = non-dimensional roll rate.  
% qhat = non-dimensional pitch rate.  
% rhat = non-dimensional yaw rate.  
% aa = angular acceleration matrix = [pdot,qdot,rdot], rad/sec2.  
% iterms = nonlinear inertial terms in the moment equations  
% = [roll, pitch, yaw] equation nonlinear inertial terms.  
%
```

### 12.9.6 *cmpsigs.m*

Many aerodynamic models include terms that consist of a parameter multiplying a regressor or explanatory variable. Since the model also usually includes a bias term, an investigation of the modeling effectiveness of the regressors or explanatory variables should exclude both the scaling (accounted for by the model parameter for each term) and the bias (accounted for by the bias term in the model). Function *cmpsigs.m* strips away the bias and scaling of regressors or explanatory variables and makes a comparison plot. This allows an analyst to quickly determine the level of similarity of the waveforms for various regressors or explanatory variables. In a sense, the *cmpsigs.m* function provides a graphical depiction of the correlations among the regressors or explanatory variables. Pair-wise correlation can be quantified mathematically using the correlation coefficient (see Chapter 5).

All of the vectors to be compared must have the same length as the time vector *t*, and are columns of the input matrix *x*. Scaling is always done relative to the first vector in the matrix *x*. For example, if *x* = [*x*<sub>1</sub>, *x*<sub>2</sub>, *x*<sub>3</sub>], vectors *x*<sub>2</sub> and *x*<sub>3</sub> would be scaled to match the original scale of *x*<sub>1</sub>.

*Header listing:*

```
%  
% CMPSIGS Signal comparision excluding biases and scaling.  
%  
% Usage: xs = cmpsigs(t,x,[plot);
```

```
%  
% Description:  
%  
% Draws a plot comparing the column vectors in x,  
% where each column has bias removed and is scaled so that  
% the waveform information can be directly compared.  
% Scaling is done relative to the signal  
% in the first column of x, and the scaled vectors  
% are output as xs.  
%  
% Input:  
%  
% t = time vector.  
% x = matrix of column vectors to be plotted and compared.  
% lplot = plot flag (optional):  
%       = 0 for no plot  
%       = 1 for plot (default)  
%  
% Output:  
%  
% xs = scaled matrix of column vectors to be plotted and compared.  
%  
% graphics:  
%   comparison plot  
%
```

## References

- <sup>1</sup>Getting Started with MATLAB, Version 7, MathWorks, Natick, MA, 2005.
- <sup>2</sup>Press, W. H., Teukolsky, S. A., Vettering, W. T., and Flannery, B. R., *Numerical Recipes in FORTRAN: The Art of Scientific Computing*, 2nd ed., Cambridge Univ. Press, New York, 1992.
- <sup>3</sup>Morelli, E. A., "System IDentification Programs for AirCraft (SIDPAC)," AIAA Paper 2002-4704, 2002.

# Appendix A

## Mathematical Background

### A.1 Linear Algebra

#### A.1.1 Basics

A matrix is a rectangular array. The matrix

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} = [a_{ij}] \quad \begin{array}{l} i = 1, 2, \dots, m \\ j = 1, 2, \dots, n \end{array} \quad (\text{A.1})$$

is called an  $m \times n$  matrix, meaning there are  $m$  rows and  $n$  columns. The vector

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

is an  $n \times 1$  column vector. The vector

$$\mathbf{y} = [y_1 \quad y_2 \quad \cdots \quad y_m]$$

is a  $1 \times m$  row vector. The dimension of a vector is the number of elements in the vector; e.g., the dimension of the vector  $\mathbf{x}$  above is  $n$ . Matrices can be thought of as consisting of row vectors stacked vertically, or column vectors stacked horizontally.

The transpose of a matrix is the matrix obtained by switching the rows and columns. For the matrix  $A$  in Eq. (A.1), the transpose is an  $n \times m$  matrix

$$A^T = \begin{bmatrix} a_{11} & a_{21} & \cdots & a_{m1} \\ a_{12} & a_{22} & \cdots & a_{m2} \\ \vdots & \vdots & \vdots & \vdots \\ a_{1n} & a_{2n} & \cdots & a_{mn} \end{bmatrix} \quad (\text{A.2})$$

Matrices with the same dimensions can be added by adding their corresponding elements,

$$\mathbf{A} + \mathbf{B} = [a_{ij}] + [b_{ij}] = [a_{ij} + b_{ij}] \quad \begin{array}{l} i = 1, 2, \dots, m \\ j = 1, 2, \dots, n \end{array}$$

Multiplying a matrix by a scalar is the same as multiplying each element in the matrix by that scalar,

$$k\mathbf{A} = \mathbf{A}k = [ka_{ij}] \quad \begin{array}{l} i = 1, 2, \dots, m \\ j = 1, 2, \dots, n \end{array}$$

for any scalar  $k$ . Two matrices  $\mathbf{A}$  and  $\mathbf{B}$  can be multiplied in the order  $\mathbf{AB}$  if the number of columns of  $\mathbf{A}$  equals the number of rows of  $\mathbf{B}$ . When this is true, the matrices  $\mathbf{A}$  and  $\mathbf{B}$  are said to be conformable or to have conformable dimensions. The product of an  $m \times n$  matrix  $\mathbf{A}$  with an  $n \times p$  matrix  $\mathbf{B}$  is an  $m \times p$  matrix  $\mathbf{C}$ , computed as

$$\mathbf{C} = [c_{ij}] = \mathbf{AB} \quad \begin{array}{l} c_{ij} = \sum_{k=1}^n a_{ik}b_{kj} \quad i = 1, 2, \dots, m \\ j = 1, 2, \dots, p \end{array} \quad (\text{A.3})$$

In general, matrix multiplication is not commutative, so  $\mathbf{AB} \neq \mathbf{BA}$ .

The inner product or dot product is a scalar defined for two vectors with the same dimension,

$$\mathbf{u} \cdot \mathbf{v} = \mathbf{u}^T \mathbf{v} = \sum_{i=1}^n u_i v_i$$

where

$$\begin{aligned} \mathbf{u}^T &= [u_1 \quad u_2 \quad \cdots \quad u_n] \\ \mathbf{v}^T &= [v_1 \quad v_2 \quad \cdots \quad v_n] \end{aligned}$$

Each element  $c_{ij}$  of the  $m \times p$  product matrix  $\mathbf{C}$  in Eq. (A.3) is the inner product of the  $i$ th row of  $\mathbf{A}$  with the  $j$ th column of  $\mathbf{B}$ .

The inner product of a vector with itself equals the sum of its squared elements, which is also the squared length of the vector,

$$\|\mathbf{u}\|^2 = \mathbf{u} \cdot \mathbf{u} = \sum_{i=1}^n u_i^2 \quad (\text{A.4})$$

If the inner product of two vectors  $\mathbf{u}$  and  $\mathbf{v}$  is zero, the vectors are said to be orthogonal,

$$\mathbf{u}^T \mathbf{v} = 0 \Leftrightarrow \text{vectors } \mathbf{u} \text{ and } \mathbf{v} \text{ are orthogonal} \quad (\text{A.5})$$

## APPENDIX A: MATHEMATICAL BACKGROUND

425

Vectors that have unit length and are mutually orthogonal are called orthonormal vectors.

The outer product of an  $n \times 1$  vector  $\mathbf{u}$  with an  $m \times 1$  vector  $\mathbf{v}$  is an  $n \times m$  matrix,

$$\mathbf{u}\mathbf{v}^T = [u_i \ v_j] \quad \begin{matrix} i = 1, 2, \dots, n \\ j = 1, 2, \dots, m \end{matrix}$$

A square matrix has the same number of rows and columns,  $m = n$ . A diagonal matrix is a square matrix with all zero values, except for the elements on the diagonal:

$$\mathbf{D} = \begin{bmatrix} d_{11} & 0 & 0 \\ 0 & d_{22} & 0 \\ 0 & 0 & d_{33} \end{bmatrix}$$

A diagonal matrix with ones on the diagonal is called the identity matrix,

$$\mathbf{I} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

A square matrix is called symmetric if

$$\mathbf{A} = \mathbf{A}^T$$

which means that the values on either side of the diagonal are the mirror image of the other side.

A matrix is Hermitian if it equals its complex conjugate transpose (see Sec. A.2),

$$\mathbf{A} = \mathbf{A}^\dagger$$

The trace of a square matrix is the sum of the diagonal elements,

$$\text{Tr}(\mathbf{A}) = \sum_{i=1}^n a_{ii} \quad \mathbf{A} = [a_{ij}] \quad i, j = 1, 2, \dots, n \quad (\text{A.6})$$

For a square matrix  $\mathbf{A}$ , the matrix inverse  $\mathbf{A}^{-1}$  is defined by

$$\mathbf{A}\mathbf{A}^{-1} = \mathbf{I} = \mathbf{A}^{-1}\mathbf{A} \quad (\text{A.7})$$

when the inverse exists. A square matrix  $\mathbf{A}$  is called orthogonal if

$$\mathbf{A}^T\mathbf{A} = \mathbf{I} = \mathbf{A}\mathbf{A}^T$$

so that

$$\mathbf{A}^T = \mathbf{A}^{-1} \quad (\text{A.8})$$

The columns of an orthogonal matrix are orthonormal vectors, and similarly for the rows.

The determinant is a scalar function of a square matrix. For a  $2 \times 2$  matrix

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \quad (\text{A.9})$$

the determinant is defined as

$$\det(\mathbf{A}) \equiv a_{11}a_{22} - a_{12}a_{21} = |\mathbf{A}| \quad (\text{A.10})$$

For a  $3 \times 3$  matrix

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \quad (\text{A.11})$$

the determinant is defined as

$$\begin{aligned} \det(\mathbf{A}) = |\mathbf{A}| = & a_{11}a_{22}a_{33} + a_{12}a_{23}a_{31} + a_{13}a_{21}a_{32} \\ & - a_{11}a_{23}a_{32} - a_{12}a_{21}a_{33} - a_{13}a_{22}a_{31} \end{aligned} \quad (\text{A.12})$$

The cofactor of the element located in the  $i$ th row and  $j$ th column of a square matrix is defined as  $(-1)^{i+j}$  times the determinant of the minor determinant, which is the determinant of the square matrix remaining after the  $i$ th row and  $j$ th column are removed. For example, the cofactor of element  $a_{12}$  in the matrix  $\mathbf{A}$  of Eq. (A.11) is

$$C_{12} = -\det \begin{bmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{bmatrix}$$

For any square matrix, the determinant can be found as the sum of all elements in a single row or column multiplied by its cofactor. For example, the determinant in Eq. (A.12) can be found by a sum of the products of each element in the second row of  $\mathbf{A}$  with its cofactor,

$$\begin{aligned} \det(\mathbf{A}) = & -a_{21} \begin{vmatrix} a_{12} & a_{13} \\ a_{32} & a_{33} \end{vmatrix} + a_{22} \begin{vmatrix} a_{11} & a_{13} \\ a_{31} & a_{33} \end{vmatrix} - a_{23} \begin{vmatrix} a_{11} & a_{12} \\ a_{31} & a_{32} \end{vmatrix} \\ = & -a_{21}a_{12}a_{33} + a_{21}a_{13}a_{32} + a_{22}a_{11}a_{33} \\ & - a_{22}a_{13}a_{31} - a_{23}a_{11}a_{32} + a_{23}a_{12}a_{31} \end{aligned}$$

## APPENDIX A: MATHEMATICAL BACKGROUND

427

The same procedure can be used to find the determinant of a larger square matrix in terms of determinants of smaller square matrices. Using this idea repeatedly, the determinant of any square matrix can be reduced to a sum of  $3 \times 3$  or  $2 \times 2$  determinants, which can be calculated as shown earlier.

If two columns of a square matrix  $A$  are proportional, then the expression for the determinant resulting from expanding along each of these columns is the same except for multiplication by a constant. But the determinant must be the same regardless of which column is used for the expansion, so the determinant of a square matrix with proportional columns must be zero. The same logic applies for proportional rows. This can be easily verified using a  $2 \times 2$  matrix,

$$A = \begin{bmatrix} a_{11} & ka_{11} \\ a_{21} & ka_{21} \end{bmatrix}$$

$$\det(A) \equiv a_{11}ka_{21} - ka_{11}a_{21} = 0$$

When the determinant is nonzero, none of the columns of the matrix are proportional. For a square  $n \times n$  matrix  $A$  composed of  $n$  column vectors,

$$A = [a_1 \ a_2 \ \cdots \ a_n]$$

if  $\det(A) \neq 0$ , then the relation

$$c_1a_1 + c_2a_2 + \cdots + c_na_n = \mathbf{0} \quad (\text{A.13})$$

is only satisfied when  $c_1 = c_2 = \cdots = c_n = 0$ . When this is true, the columns of  $A$  are said to be linearly independent, and  $A$  is called a nonsingular matrix. A matrix  $A$  with  $n$  linearly independent columns is said to have rank  $n$ .

The inverse of a nonsingular square matrix can be computed from

$$A^{-1} = \frac{[\text{Cof}(A)]^T}{|A|} \quad (\text{A.14})$$

where  $\text{Cof}(A)$  is the matrix of cofactors of  $A$

$$\text{Cof}(A) = \begin{bmatrix} C_{11} & C_{12} & \cdots & C_{1n} \\ C_{21} & C_{22} & \cdots & C_{2n} \\ \vdots & \vdots & & \vdots \\ C_{n1} & C_{n2} & \cdots & C_{nn} \end{bmatrix}$$

This expression can be verified by

$$\begin{aligned} A^{-1}A &= \frac{[\text{Cof}(A)]^T}{|A|}A = \frac{1}{|A|} \sum_{k=1}^n C_{ki}a_{kj} = \begin{cases} 1 & \text{for } i=j \\ 0 & \text{for } i \neq j \end{cases} \quad i, j = 1, 2, \dots, n \\ &= I \end{aligned}$$

since the summation is either an expansion for  $|A|$  along a column of  $A$  or an expansion for the determinant of a matrix with two identical columns, which is zero.

From Eq. (A.4), it follows that the inverse of a square matrix exists only when the determinant is nonzero. Equivalently, the matrix inverse exists only for a non-singular matrix, which is a matrix with columns that are linearly independent.

### A.1.2 Useful Matrix Relationships

The following is a list of useful matrix identities. In the expressions,  $A$ ,  $B$ , and  $C$  are matrices with appropriate dimensions,  $k$  is a scalar,  $n$  is a positive integer, and  $I$  is the identity matrix with appropriate dimension:

$$A + B = B + A \quad \text{In general, } AB \neq BA \quad (\text{A.15})$$

$$(AB)^T = B^T A^T \quad (AB)^{-1} = B^{-1} A^{-1} \quad (A + B)^T = A^T + B^T \quad (\text{A.16})$$

$$(A^n)^{-1} = (A^{-1})^n \quad (A^{-1})^T = (A^T)^{-1} \quad IA = AI = A \quad (\text{A.17})$$

$$k(AB) = (kA)B = A(kB) \quad (A^T)^T = A \quad (\text{A.18})$$

$$(A^{-1})^{-1} = A \quad (kA)^{-1} = \frac{1}{k}A^{-1} \quad |kA| = k^n|A| \quad (\text{A.19})$$

$(A = n \times n \text{ matrix})$

$$|AB| = |A||B| \quad |A| = |A^T| \quad |A^{-1}| = \frac{1}{|A|} \quad |I| = 1 \quad (\text{A.20})$$

### A.1.3 Matrix Inversion Lemma

Given square, nonsingular matrices  $A$ ,  $C$ , and  $A + BCD$  with appropriate dimensions,

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(C^{-1} + DA^{-1}B)^{-1}DA^{-1} \quad (\text{A.21})$$

This identity is called the matrix inversion lemma. To prove it, multiply both sides of the last equation by  $A + BCD$ ,

$$I = (A + BCD)[A^{-1} - A^{-1}B(C^{-1} + DA^{-1}B)^{-1}DA^{-1}]$$

By straightforward matrix manipulations,

$$I = I + BCDA^{-1} - B(C^{-1} + DA^{-1}B)^{-1}DA^{-1}$$

$$- BCDA^{-1}B(C^{-1} + DA^{-1}B)^{-1}DA^{-1}$$

$$I = I + BCDA^{-1} - B(I + CDA^{-1}B)(C^{-1} + DA^{-1}B)^{-1}DA^{-1}$$

$$I = I + BCDA^{-1} - BC(C^{-1} + DA^{-1}B)(C^{-1} + DA^{-1}B)^{-1}DA^{-1}$$

$$I = I + BCDA^{-1} - BCDA^{-1}$$

$$I = I$$

The original statement reduces to an identity, which proves the lemma.

## APPENDIX A: MATHEMATICAL BACKGROUND

429

**A.1.4 Expressing Vectors in Different Reference Frames**

An arbitrary three-dimensional vector can be expressed in two different reference frames with the same origin as follows:

$$\begin{aligned} \text{Frame } A: \quad \mathbf{V} &= [v_{A_1} \quad v_{A_2} \quad v_{A_3}]^T \\ \text{Frame } B: \quad \mathbf{V} &= [v_{B_1} \quad v_{B_2} \quad v_{B_3}]^T \end{aligned}$$

Figure A.1 illustrates the situation, showing only one axis from the  $B$  frame, for clarity.

If  $\theta_{ij}$  denotes the angle of the  $i$ th axis of the  $B$  frame relative to the  $j$ th axis of the  $A$  frame, then the components of  $\mathbf{V}$  in the  $B$  frame can be expressed as

$$v_{B_i} = \sum_{j=1}^3 v_{A_j} \cos \theta_{ij}$$

where

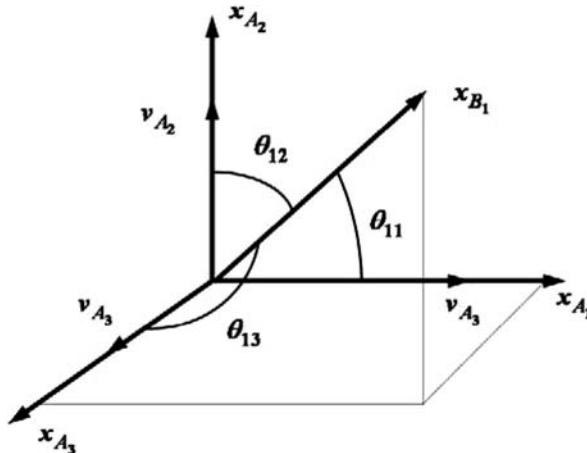
$$l_{ij} \equiv \cos \theta_{ij}$$

is called a direction cosine. If the direction cosines are arranged in a transformation matrix  $L_{BA}$  as

$$L_{BA} \equiv [l_{ij}] \quad i, j = 1, 2, 3$$

then

$$\mathbf{V}_B = \begin{bmatrix} v_{B_1} \\ v_{B_2} \\ v_{B_3} \end{bmatrix} = L_{BA} \begin{bmatrix} v_{A_1} \\ v_{A_2} \\ v_{A_3} \end{bmatrix} = L_{BA} \mathbf{V}_A$$



**Fig. A.1 Vector components.**

The last two equations are the required transformation for components of a vector from one reference frame to another with the same origin.

The vector  $\mathbf{V}$  has the same magnitude, regardless of the reference frame, so

$$|\mathbf{V}| = \mathbf{V}_A^T \mathbf{V}_A = \mathbf{V}_B^T \mathbf{V}_B = \mathbf{V}_A^T \mathbf{L}_{BA}^T \mathbf{L}_{BA} \mathbf{V}_A$$

It follows that

$$\mathbf{L}_{BA}^T \mathbf{L}_{BA} = \mathbf{I}$$

$$\mathbf{L}_{BA}^{-1} = \mathbf{L}_{BA}^T$$

$$|\mathbf{L}_{BA}^T \mathbf{L}_{BA}| = |\mathbf{L}_{BA}^T| |\mathbf{L}_{BA}| = |\mathbf{L}_{BA}|^2 = 1$$

The preceding expressions are orthogonality conditions, which show that the columns of  $\mathbf{L}_{BA}$  are orthonormal, and that  $|\mathbf{L}_{BA}| = 1$ . It follows that the transformation is nonzero and does not alter the vector magnitude. Furthermore,

$$\mathbf{V}_B = \mathbf{L}_{BA} \mathbf{V}_A = \mathbf{L}_{BA} \mathbf{L}_{AB} \mathbf{V}_B$$

Combining the last two sets of expressions,

$$\mathbf{L}_{AB} = \mathbf{L}_{BA}^{-1} = \mathbf{L}_{BA}^T$$

The transformation matrix  $\mathbf{L}_{BA}$  is therefore an orthogonal matrix.

The orthogonality conditions yield a set of six conditions on the columns of the transformation matrix,

$$\begin{aligned} \mathbf{L}_{BA} &= [\mathbf{l}_1 \quad \mathbf{l}_2 \quad \mathbf{l}_3] \\ \mathbf{l}_i^T \mathbf{l}_j &= \delta_{ij} = \begin{cases} 1 & \text{for } i = j \\ 0 & \text{for } i \neq j \end{cases} \quad i, j = 1, 2, 3 \end{aligned}$$

This represents six relations among the nine elements of  $\mathbf{L}_{BA}$ , so only three of the nine elements of  $\mathbf{L}_{BA}$  are independent. These three independent direction cosines specify the orientation of frame  $B$  relative to frame  $A$ .

As noted in Chapter 3, relative orientation of one reference frame to another for aircraft is typically done using single axis rotations. For such rotations, the transformation matrices are as follows:

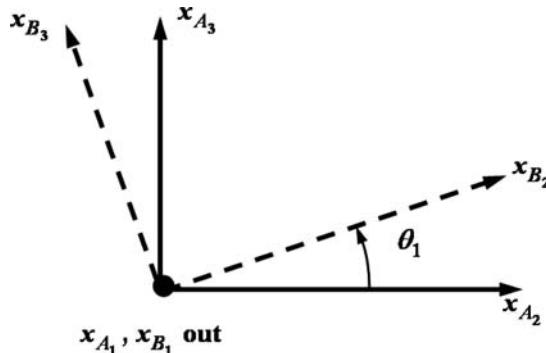
*x-axis rotation* (Fig. A.2):

$$\mathbf{L}_1(\theta_1) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta_1 & \sin \theta_1 \\ 0 & -\sin \theta_1 & \cos \theta_1 \end{bmatrix}$$

$$\mathbf{V}_B = \mathbf{L}_1(\theta_1) \mathbf{V}_A$$

## APPENDIX A: MATHEMATICAL BACKGROUND

431


 Fig. A.2 *x*-axis rotation.

*y*-axis rotation (Fig. A.3):

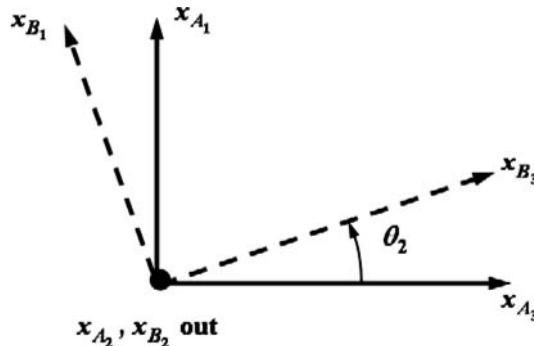
$$L_2(\theta_2) = \begin{bmatrix} \cos \theta_2 & 0 & -\sin \theta_2 \\ 0 & 1 & 0 \\ \sin \theta_2 & 0 & \cos \theta_2 \end{bmatrix}$$

$$V_B = L_2(\theta_2)V_A$$

*z*-axis rotation (Fig. A.4):

$$L_3(\theta_3) = \begin{bmatrix} \cos \theta_3 & \sin \theta_3 & 0 \\ -\sin \theta_3 & \cos \theta_3 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$V_B = L_3(\theta_3)V_A$$


 Fig. A.3 *y*-axis rotation.

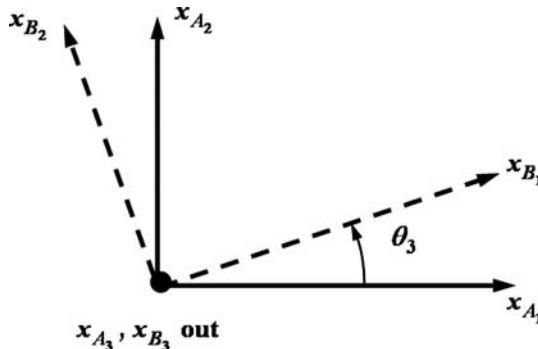


Fig. A.4 z-axis rotation.

Full specification of the relative orientation of one reference frame to another can be made with three single axis rotations. Common transformations for aircraft are from vehicle-carried earth axes to body axes using Euler angles,

$$\mathbf{L}_{BV} = \mathbf{L}_1(\phi)\mathbf{L}_2(\theta)\mathbf{L}_3(\psi) = \mathbf{L}_{VB}^T$$

$$\mathbf{L}_{BV} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & \sin \phi \\ 0 & -\sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix} \begin{bmatrix} \cos \psi & \sin \psi & 0 \\ -\sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{L}_{BV} = \begin{bmatrix} \cos \theta \cos \psi & \cos \theta \sin \psi & -\sin \theta \\ \sin \phi \sin \theta \cos \psi - \cos \phi \sin \psi & \sin \phi \sin \theta \sin \psi + \cos \phi \cos \psi & \sin \phi \cos \theta \\ \cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi & \cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi & \cos \phi \cos \theta \end{bmatrix}$$

and from wind axes to body axes using airflow angles,

$$\mathbf{L}_{BW} = \mathbf{L}_2(\alpha)\mathbf{L}_3(-\beta) = \mathbf{L}_{WB}^T$$

$$\mathbf{L}_{BW} = \begin{bmatrix} \cos \alpha & 0 & -\sin \alpha \\ 0 & 1 & 0 \\ \sin \alpha & 0 & \cos \alpha \end{bmatrix} \begin{bmatrix} \cos \beta & -\sin \beta & 0 \\ \sin \beta & \cos \beta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

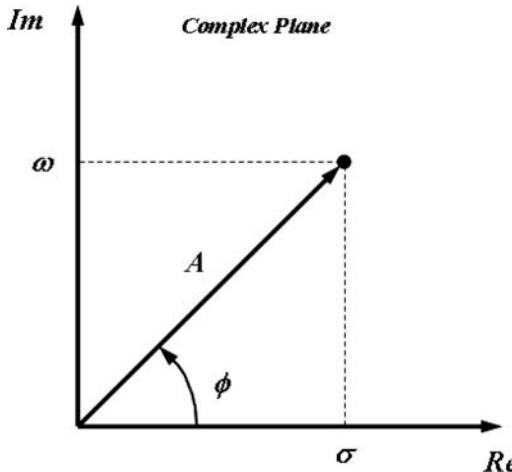
$$\mathbf{L}_{BW} = \begin{bmatrix} \cos \alpha \cos \beta & -\cos \alpha \sin \beta & -\sin \alpha \\ \sin \beta & \cos \beta & 0 \\ \sin \alpha \cos \beta & -\sin \alpha \sin \beta & \cos \alpha \end{bmatrix}$$

## A.2 Complex Numbers

A complex number  $s$  with real part  $\sigma$  and imaginary part  $\omega$  can be written as  $s = \sigma + j\omega$ , and represented by a point in the complex plane (see Fig. A.5).

## APPENDIX A: MATHEMATICAL BACKGROUND

433



**Fig. A.5** Complex number representations.

The same point in the complex plane can also be represented as a vector with magnitude  $A$  and phase angle  $\phi$

$$s = \sigma + j\omega = A(\cos \phi + j \sin \phi) = Ae^{j\phi} \quad (\text{A.22})$$

where  $\sigma$ ,  $\omega$ ,  $A$ , and  $\phi$  are real numbers,  $j$  is the imaginary number,  $j \equiv \sqrt{-1}$ , and

$$A = |s| = \sqrt{\sigma^2 + \omega^2}, \quad \phi = \tan^{-1}\left(\frac{\omega}{\sigma}\right) \quad (\text{A.23})$$

The last equality in Eq. (A.22) uses Euler's identity:

$$e^{j\phi} = \cos \phi + j \sin \phi \quad (\text{A.24})$$

The representation of a complex number in terms of the amplitude  $A$  and the phase angle  $\phi$  shown in Eq. (A.22) is called phasor notation.

For any two complex numbers,

$$s_1 = \sigma_1 + j\omega_1 = A_1 e^{j\phi_1}, \quad s_2 = \sigma_2 + j\omega_2 = A_2 e^{j\phi_2} \quad (\text{A.25})$$

$$s_1 s_2 = A_1 A_2 e^{j(\phi_1 + \phi_2)} \quad (\text{A.26})$$

$$\frac{s_1}{s_2} = \frac{A_1}{A_2} e^{j(\phi_1 - \phi_2)} \quad (\text{A.27})$$

$$s_1 \pm s_2 = (\sigma_1 \pm \sigma_2) + j(\omega_1 \pm \omega_2) \quad (\text{A.28})$$

If  $s$  is complex, then a function  $G(s)$  is also complex, in general. Using Eq. (A.27), a vector with complex elements can be represented using relative magnitudes and phase angles, as in the following example for a complex vector with two elements:

$$\begin{bmatrix} \sigma_1 + j\omega_1 \\ \sigma_2 + j\omega_2 \end{bmatrix} = \begin{bmatrix} A_1 e^{j\phi_1} \\ A_2 e^{j\phi_2} \end{bmatrix} \Rightarrow \begin{bmatrix} 1 \\ \frac{A_2}{A_1} e^{j(\phi_2 - \phi_1)} \end{bmatrix} \quad (\text{A.29})$$

This technique is used to show relative magnitude and phasing of elements in complex vectors.

### A.3 Calculus

#### A.3.1 Matrix Differentiation

For a square, nonsingular matrix  $A$ ,

$$\frac{d}{dA} \ln|A| = (A^{-1})^T \quad (\text{A.30a})$$

$$\frac{\partial}{\partial A} \text{Tr}(A^{-1}B) = -[A^{-1}BA^{-1}]^T \quad (\text{A.30b})$$

If  $A$  is symmetric,

$$\frac{\partial}{\partial A} \text{Tr}(BAB^T) = 2BA \quad (\text{A.30c})$$

When  $A$  is a function of a scalar  $\theta$ ,

$$\frac{d}{d\theta} (A^{-1}) = -A^{-1} \frac{dA}{d\theta} A^{-1} \quad (\text{A.31})$$

For a quadratic form  $Q = \mathbf{x}^T A \mathbf{x}$ , where  $\mathbf{x}$  is an  $n \times 1$  vector and  $A$  is an  $n \times n$  square matrix,

$$\nabla_{\mathbf{x}} Q = \frac{\partial Q}{\partial \mathbf{x}} = 2A\mathbf{x} \quad (\text{A.32})$$

When  $A$  and  $\mathbf{x}$  are a function of a scalar  $\theta$ ,

$$\frac{dQ}{d\theta} = \left( \frac{\partial \mathbf{x}}{\partial \theta} \right)^T A \mathbf{x} + \mathbf{x}^T \frac{\partial A}{\partial \theta} \mathbf{x} + \mathbf{x}^T A \frac{\partial \mathbf{x}}{\partial \theta} \quad (\text{A.33})$$

If  $A$  is symmetric, the last expression reduces to

$$\frac{dQ}{d\theta} = 2 \left( \frac{\partial \mathbf{x}}{\partial \theta} \right)^T A \mathbf{x} + \mathbf{x}^T \frac{\partial A}{\partial \theta} \mathbf{x} \quad (\text{A.34})$$

## APPENDIX A: MATHEMATICAL BACKGROUND

435

**A.3.2 The Laplace Transform**

For a physical signal  $x(t)$  of the type encountered in practical airplane flight, the Laplace transform is defined by

$$\mathcal{L}[x(t)] \equiv \int_0^{\infty} x(t)e^{-st} dt = \tilde{x}(s) \quad (\text{A.35})$$

Since the Laplace transform is an integral transform, it is also a linear operator,

$$\mathcal{L}[ax(t) + by(t)] \equiv \int_0^{\infty} [ax(t) + by(t)]e^{-st} dt = a\tilde{x}(s) + b\tilde{y}(s) \quad (\text{A.36})$$

for any constants  $a$  and  $b$ .

The following formulas can be derived using the definition of the Laplace transform, along with integration by parts in some cases,

$$\mathcal{L}[\dot{x}(t)] = s \int_0^{\infty} x(t)e^{-st} dt - x(0) = s\tilde{x}(s) - x(0) \quad (\text{A.37})$$

$$\mathcal{L}[\ddot{x}(t)] = s \int_0^{\infty} \dot{x}(t)e^{-st} dt - \dot{x}(0) = s[s\tilde{x}(s) - x(0)] - \dot{x}(0) = s^2\tilde{x}(s) - sx(0) - \dot{x}(0)$$

$$\mathcal{L}\left[\frac{d^n x(t)}{dt^n}\right] = s^n\tilde{x}(s) - s^{n-1}x(0) - s^{n-2}\frac{dx}{dt}(0) - \dots - \frac{d^{n-1}x}{dt^{n-1}}(0) \quad (\text{A.38})$$

$$\mathcal{L}[\delta(t)] \equiv \int_0^{\infty} \delta(t)e^{-st} dt = 1 \quad (\text{A.39})$$

$$\mathcal{L}[1(t)] \equiv \int_0^{\infty} 1(t)e^{-st} dt = \frac{1}{s} \quad (\text{A.40})$$

$$\mathcal{L}(e^{-at}) \equiv \int_0^{\infty} e^{-at}e^{-st} dt = \frac{1}{(s+a)} \quad (\text{A.41})$$

$$\mathcal{L}(te^{-at}) \equiv \int_0^{\infty} te^{-at}e^{-st} dt = \frac{1}{(s+a)^2} \quad (\text{A.42})$$

$$\mathcal{L}\left(\frac{t^k e^{-at}}{k!}\right) \equiv \int_0^{\infty} \frac{t^k e^{-at}}{k!} e^{-st} dt = \frac{1}{(s+a)^{k+1}} \quad (\text{A.43})$$

**A.3.3 Convolution Integral**

For a scalar dynamic system, the output can be computed from the convolution integral

$$y(t) = \int_0^t g(t-\tau)u(\tau) d\tau = g(t) * u(t) \quad (\text{A.44})$$

where  $g(t)$  is the impulse response function, also known as the weighting function. Changing variables using  $\xi = t - \tau$ ,

$$\begin{aligned} y(t) &= \int_0^t g(t - \tau)u(\tau) d\tau = \int_t^0 g(\xi)u(t - \xi)(-d\xi) \\ &= \int_0^t u(t - \xi)g(\xi) d\xi = u(t) * g(t) \end{aligned} \quad (\text{A.45})$$

which shows that the convolution operation is commutative. If the dynamic system is causal, meaning that effects always follow their cause, then

$$g(t - \tau) = 0 \quad \text{for } \tau > t$$

The convolution integral can then be written as

$$y(t) = \int_0^\infty g(t - \tau)u(\tau) d\tau$$

Taking the Laplace transform,

$$\mathcal{L}[y(t)] = \mathcal{L}\left[\int_0^\infty g(t - \tau)u(\tau) d\tau\right] = \int_0^\infty e^{-st} \left[\int_0^\infty g(t - \tau)u(\tau) d\tau\right] dt$$

Changing the order of integration and substituting  $\lambda = t - \tau$ ,

$$\begin{aligned} \mathcal{L}[y(t)] &= \int_0^\infty \int_0^\infty g(t - \tau)e^{-st} dt u(\tau) d\tau = \int_0^\infty \int_0^\infty g(\lambda)e^{-s(\lambda+\tau)} d\lambda u(\tau) d\tau \\ \mathcal{L}[y(t)] &= \int_0^\infty \int_0^\infty g(\lambda)e^{-s\lambda} d\lambda u(\tau)e^{-s\tau} d\tau = \int_0^\infty g(\lambda)e^{-s\lambda} d\lambda \int_0^\infty u(\tau)e^{-s\tau} d\tau \\ \tilde{y}(s) &\equiv \mathcal{L}[y(t)] = \mathcal{L}[h(t)]\mathcal{L}[u(t)] = \tilde{g}(s)\tilde{u}(s) \end{aligned} \quad (\text{A.46})$$

The preceding development shows that convolution in the time domain is equivalent to multiplication in the Laplace domain. By changing the roles of the variables, the same development can be used to show that convolution in the Laplace domain is equivalent to multiplication in the time domain.

#### A.3.4 Fourier Transform

Replacing the Laplace transform variable  $s$  with  $j\omega$  gives the Fourier transform,

$$\mathcal{F}[x(t)] = \tilde{x}(j\omega) = \int_0^\infty x(t)e^{-j\omega t} dt = \tilde{x}(s) \Big|_{s=j\omega} \quad (\text{A.47})$$

## APPENDIX A: MATHEMATICAL BACKGROUND

437

Setting  $s = j\omega$  in the transfer function  $[\tilde{y}(s)/\tilde{u}(s)] = G(s)$ ,

$$\frac{\tilde{y}(j\omega)}{\tilde{u}(j\omega)} = G(j\omega) \quad (\text{A.48})$$

results in the frequency response  $G(j\omega)$ , which defines the steady-state response of the dynamic system to sinusoidal inputs.

#### A.4 Polynomial Splines

Spline functions are defined as piecewise polynomial functions of degree  $m$  in one or more independent variables. The term “piecewise” means that the polynomial is different for specific ranges of the independent variables. When continuity constraints are considered, the function values and derivatives agree at the points where the piecewise polynomials join. These points are called knots and are defined as specific values of each independent variable. A polynomial spline  $S_m(x)$  of degree  $m$  with continuous derivatives up to degree  $m - 1$ , for a single independent variable  $x \in [x_0, x_{\max}]$ , can be expressed as

$$S_m(x) = \sum_{r=1}^m C_r x^r + \sum_{i=1}^k D_i (x - x_i)_+^m \quad (\text{A.49})$$

where

$$(x - x_i)_+^m = \begin{cases} (x - x_i)^m & x > x_i \\ 0 & x \leq x_i \end{cases} \quad (\text{A.50})$$

and where  $C_r$  and  $D_i$  are constants. The values  $x_1, x_2, \dots, x_k$  are knots that satisfy the condition

$$x_0 < x_1 < x_2 < \dots < x_k < x_{\max} \quad (\text{A.51})$$

The special case of the polynomial spline for  $m=0$  (a spline of degree zero) represents an approximation by piecewise constants.

A polynomial spline in two independent variables  $x_1$  and  $x_2$  can be introduced to approximate a function of two independent variables over the range  $x_1 \in [x_{10}, x_{1\max}]$  and  $x_2 \in [x_{20}, x_{2\max}]$ . Then, as in the one-dimensional case, the two ranges  $[x_{10}, x_{1\max}]$  and  $[x_{20}, x_{2\max}]$  are subdivided by sets of knots  $x_{1i}$  and  $x_{2i}$  where

$$\begin{aligned} x_{10} < x_{11} < x_{12} < \dots < x_{1k_1} < x_{1\max} \\ x_{20} < x_{21} < x_{22} < \dots < x_{2k_2} < x_{2\max} \end{aligned}$$

The knots partition the rectangle defined by the full independent variable ranges into rectangular panels. A polynomial spline of degree  $m$  for  $x_1$  and degree  $n$  for  $x_2$  with continuous partial derivatives up to degree  $(m - 1)$  and  $(n - 1)$ , respectively, on the rectangle defined by the intervals  $[x_{10}, x_{1\max}]$  and  $[x_{20}, x_{2\max}]$  can be formulated as

$$\begin{aligned} S_m(x_1, x_2) = & \sum_{r=0}^m \sum_{s=0}^n C_{rs} x_1^r x_2^s + \sum_{i=1}^k P_i(x_2)(x_1 - x_{1i})_+^m + \sum_{j=1}^l Q_j(x_1)(x_2 - x_{2j})_+^n \\ & + \sum_{i=1}^k \sum_{j=1}^l D_{ij}(x_1 - x_{1i})_+^m (x_2 - x_{2j})_+^n \end{aligned} \quad (\text{A.52})$$

where  $P_i(x_2)$  and  $Q_j(x_1)$  are polynomials of degree  $n$  and  $m$ , respectively, each with constant coefficients, and where  $C_{rs}$  and  $D_{ij}$  are constants.

## Appendix B

# Probability, Statistics, and Random Variables

### B.1 Random Variables

A random variable  $X$  is a quantity that can take on values randomly according to a probability  $\mathcal{P}(X \leq x)$ , where  $x$  is a selected value. The probability  $\mathcal{P}(X \leq x)$  gives a scalar value on the interval  $[0,1]$  indicating the probability that the random variable  $X$  will take on a value less than or equal to  $x$ . Consequently,  $\mathcal{P}(X \leq x)$  depends on  $x$ . A probability of 0 corresponds to an impossibility, whereas a probability of 1 corresponds to certainty.

Random variables can be discrete or continuous. In the development given here, only continuous random variables will be considered, and each random variable is denoted by a capital letter. These conventions are not adhered to in the chapters.

#### B.1.1 Probability Distribution and Probability Density

The probability distribution function of a random variable  $X$  is defined by

$$P(x) \equiv \mathcal{P}(X \leq x) \quad -\infty < x < \infty \quad (\text{B.1})$$

where  $P(x)$  is a scalar value on the interval  $[0,1]$  indicating the probability that the random variable  $X$  will take a value less than or equal to  $x$ . Some properties of the probability distribution function are

$$\begin{aligned} P(x) &\geq 0 & -\infty < x < \infty \\ P(-\infty) &= 0 & P(\infty) = 1 \\ P(x_1) &\leq P(x_2) & \text{for } x_1 \leq x_2 \end{aligned} \quad (\text{B.2})$$

The probability density function of the random variable  $X$ , also known as the frequency function, is defined as

$$p(x) = \frac{dP(x)}{dx} \quad (\text{B.3})$$

Then

$$\int_{-\infty}^{\infty} p(x) dx = P(\infty) - P(-\infty) = 1 \quad (\text{B.4})$$

and

$$\int_{x_1}^{x_2} p(x) dx = P(x_2) - P(x_1) = \mathcal{P}(x_1 \leq X \leq x_2) \quad (\text{B.5})$$

### B.1.2 Expected Value and Variance

The expected value, or the mean, of a random variable  $X$  is defined by

$$E(X) = \int_{-\infty}^{\infty} x p(x) dx \quad (\text{B.6})$$

where  $p(x)$  is the probability density function for  $X$ . If the same experiment is repeated  $N$  times, with each run producing a sample of the random variable  $X$ , the frequency interpretation of the expected value has the form

$$E(X) \approx \frac{1}{N} \sum_{i=1}^N x_i = \bar{x} \quad (\text{B.7})$$

where the  $x_i$ ,  $i = 1, 2, \dots, N$ , are the sample values of the random variable  $X$ . In this case, the probability density for each sample is the constant  $1/N$ , and summation replaces the integral.

The most likely value of  $X$  is the constant  $x_m$  such that  $p(x_m)$  has the maximum value. If  $p(x)$  is even, then  $p(x) = p(-x)$  and  $E(X) = 0$ . If  $p(x)$  is symmetric about  $x = a$ , then  $p(a - x) = p(a + x)$  and  $E(X) = a$ .

The variance of a random variable  $X$  with expected value  $E(X) = \eta$  is defined by

$$\sigma^2 = E[(X - \eta)^2] \equiv \int_{-\infty}^{\infty} (x - \eta)^2 p(x) dx \quad (\text{B.8})$$

This definition can also be expressed as

$$\sigma^2 = E(X^2 - 2\eta X + \eta^2) = E(X^2) - 2\eta E(X) + \eta^2 = E(X^2) - \eta^2$$

or

$$\sigma^2 = E(X^2) - [E(X)]^2 \quad (\text{B.9})$$

If  $X$  is a random variable with  $E(X) = \eta$  and  $E[(X - \eta)^2] = \sigma^2$ , then  $Y = cX$  for any constant  $c$  is also a random variable, with

$$E(Y) = cE[X] = c\eta \quad \text{Var}(Y) = E[(cX - c\eta)^2] = c^2\sigma^2 \quad (\text{B.10})$$

### B.1.3 Two Random Variables

For two random variables  $X$  and  $Y$ , consider two sets of events defined by  $(X \leq x)$  and  $(Y \leq y)$ , respectively. The probability distribution functions of  $X$  and  $Y$  are  $P(x) = P(X \leq x)$  and  $P(y) = P(Y \leq y)$ , respectively. The joint probability distribution of the random variables  $X$  and  $Y$  is defined by

$$P(x, y) \equiv P(X \leq x, Y \leq y) \quad (\text{B.11})$$

The joint probability distribution has the following properties, which follow from the corresponding properties for a single random variable:

$$\begin{aligned} P(x, y) &\geq 0 & -\infty < x < \infty, & -\infty < y < \infty \\ P(-\infty, -\infty) &= P(-\infty, y) = P(x, -\infty) = 0 \\ P(x, \infty) &= P(x) & P(\infty, y) = P(y) & P(\infty, \infty) = 1 \end{aligned} \quad (\text{B.12})$$

The joint probability density function for the random variables  $X$  and  $Y$  is defined by

$$p(x, y) = \frac{\partial^2 P(x, y)}{\partial x \partial y} \quad (\text{B.13})$$

The joint probability density  $p(x, y)$  is related to the marginal probability densities  $p(x)$  and  $p(y)$  by

$$p(x) = \int_{-\infty}^{\infty} p(x, y) dy \quad (\text{B.14})$$

$$p(y) = \int_{-\infty}^{\infty} p(x, y) dx \quad (\text{B.15})$$

### B.1.4 Uncorrelated, Orthogonal, and Independent Random Variables

Two random variables  $X$  and  $Y$  are called uncorrelated if

$$E(XY) = E(X)E(Y) \quad (\text{B.16})$$

They are orthogonal if

$$E(XY) = 0 \quad (\text{B.17})$$

and independent if

$$p(x, y) = p(x)p(y) \quad (\text{B.18})$$

If two random variables  $X$  and  $Y$  are uncorrelated, then their covariance and correlation coefficient are zero:

$$\begin{aligned}\text{Cov}(X, Y) &= E[(X - \eta_x)(Y - \eta_y)] \\ &= E(XY) - \eta_x E(Y) - E(X)\eta_y + \eta_x \eta_y \\ &= E(X)E(Y) - \eta_x E(Y) - E(X)\eta_y + \eta_x \eta_y = 0\end{aligned}\quad (\text{B.19})$$

$$r = \frac{\text{Cov}(X, Y)}{\sigma_x \sigma_y} = 0 \quad (\text{B.20})$$

where

$$\begin{aligned}\eta_x &= E(X), & \sigma_x^2 &= E[(X - \eta_x)^2] \\ \eta_y &= E(Y), & \sigma_y^2 &= E[(Y - \eta_y)^2]\end{aligned}\quad (\text{B.21})$$

### B.1.5 Functions of Two Random Variables

If the random variables  $X$  and  $Y$  are combined linearly,  $aX + bY$ , where  $a$  and  $b$  are known constants, then the mean value and variance are

$$E(aX + bY) = a\eta_x + b\eta_y \quad (\text{B.22})$$

$$\begin{aligned}\sigma_{aX+bY}^2 &= E\{[a(X - \eta_x) + b(Y - \eta_y)]^2\} \\ &= a^2 \sigma_x^2 + b^2 \sigma_y^2 + 2ab\sigma_x \sigma_y\end{aligned}\quad (\text{B.23})$$

If random variables  $X$  and  $Y$  are combined nonlinearly, e.g.,  $f(X, Y)$ , then

$$E[f(X, Y)] \approx f(\eta_x, \eta_y) \quad (\text{B.24})$$

$$\begin{aligned}\sigma_{f(X, Y)}^2 &= E\{[f(X, Y) - f(\eta_x, \eta_y)]^2\} \\ &\approx \left(\frac{\partial f}{\partial X}\right)^2 \sigma_x^2 + \left(\frac{\partial f}{\partial Y}\right)^2 \sigma_y^2 + 2\left(\frac{\partial f}{\partial X}\right)\left(\frac{\partial f}{\partial Y}\right)r\sigma_x \sigma_y\end{aligned}\quad (\text{B.25})$$

where the two approximate equalities follow from the definitions of the mean and variance, using the approximation

$$f(X, Y) \approx f(\eta_x, \eta_y) + \left(\frac{\partial f}{\partial X}\right)(X - \eta_x) + \left(\frac{\partial f}{\partial Y}\right)(Y - \eta_y) \quad (\text{B.26})$$

with the partial derivatives  $\partial f / \partial X$  and  $\partial f / \partial Y$  evaluated at  $X = \eta_x$  and  $Y = \eta_y$ .

### B.1.6 Conditional Probability and Bayes's Rule

For two random variables  $X$  and  $Y$ , the conditional probability that  $Y \leq y$ , given that  $X \leq x$ , can be expressed as

$$\mathcal{P}(Y \leq y | X \leq x) = \frac{\mathcal{P}(Y \leq y, X \leq x)}{\mathcal{P}(X \leq x)} = \frac{P(x, y)}{P(x)} \quad (\text{B.27})$$

This conditional probability has a simple frequency interpretation as the number of events where both  $Y \leq y$  and  $X \leq x$  occur, divided by the number of events where  $X \leq x$  occurs. The analogous statement using probability density functions is

$$p(y|x) = \frac{p(x, y)}{p(x)} \quad (\text{B.28})$$

Similarly, for the conditional probability density of  $x$ , given that  $Y = y$ ,

$$p(x|y) = \frac{p(x, y)}{p(y)} \quad (\text{B.29})$$

From the last two equations, it follows that

$$p(x, y) = p(x|y)p(y) = p(y|x)p(x) \quad (\text{B.30})$$

or

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)} \quad (\text{B.31})$$

which is known as Bayes's rule. Its alternative form is

$$p(y|x) = \frac{p(x|y)p(y)}{p(x)} \quad (\text{B.32})$$

### B.1.7 Random Vector

A collection of  $N$  random variables  $X_1, X_2, \dots, X_N$  can be arranged as a random vector  $\mathbf{X} = [X_1, X_2, \dots, X_N]^T$ . The joint probability distribution of  $\mathbf{X}$  is defined by

$$P(x_1, x_2, \dots, x_N) \equiv \mathcal{P}(X_1 \leq x_1, X_2 \leq x_2, \dots, X_N \leq x_N) \quad (\text{B.33})$$

The joint probability density  $p(x_1, x_2, \dots, x_N)$  is obtained by differentiating the joint distribution with respect to  $x_1, x_2, \dots, x_N$ .

The mean value and covariance matrix of the random vector  $X$  are defined by

$$E(X) = \boldsymbol{\eta} \quad (\text{B.34})$$

$$\text{Cov}(X) = E[(X - \boldsymbol{\eta})(X - \boldsymbol{\eta})^T] = \boldsymbol{\Sigma} \quad (\text{B.35})$$

If random vectors  $X_1, X_2, \dots, X_N$  are mutually independent, then

$$p(X_1, X_2, \dots, X_N) = p(X_1)p(X_2) \cdots p(X_N) \quad (\text{B.36})$$

where  $p(X_i), i = 1, 2, \dots, N$ , are marginal densities. Random vectors  $X_1$  and  $X_2$  are called uncorrelated if

$$E(X_1 X_2^T) = E(X_1)E(X_2^T) = \boldsymbol{\eta}_1 \boldsymbol{\eta}_2^T \quad (\text{B.37})$$

Random vectors  $X_1$  and  $X_2$  are called orthogonal if their elements are mutually orthogonal, so that

$$E(X_1 X_2^T) = \mathbf{0} \quad (\text{B.38})$$

If a random vector  $X$  is formed as

$$X = \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \quad (\text{B.39})$$

with

$$E(X) = \begin{bmatrix} E(X_1) \\ E(X_2) \end{bmatrix} = \begin{bmatrix} \boldsymbol{\eta}_1 \\ \boldsymbol{\eta}_2 \end{bmatrix} = \boldsymbol{\eta} \quad (\text{B.40})$$

$$\text{Cov}(X) = E[(X - \boldsymbol{\eta})(X - \boldsymbol{\eta})^T] = \boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{12} & \boldsymbol{\Sigma}_{22} \end{bmatrix} \quad (\text{B.41})$$

Combining Eqs. (B.37) and (B.39)–(B.41),  $\boldsymbol{\Sigma}_{12} = \mathbf{0}$  when  $X_1$  and  $X_2$  are uncorrelated.

## B.2 Statistics

### B.2.1 Gaussian (Normal) Distribution

A random variable is said to be normally distributed or Gaussian if its probability density function takes the form

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{(x - \eta)^2}{2\sigma^2}\right] \quad (\text{B.42})$$

where  $\eta$  is the mean value of the random variable, and  $\sigma^2$  is the variance. The normal probability density function is completely determined by the mean  $\eta$

## APPENDIX B: RANDOM VARIABLES

445

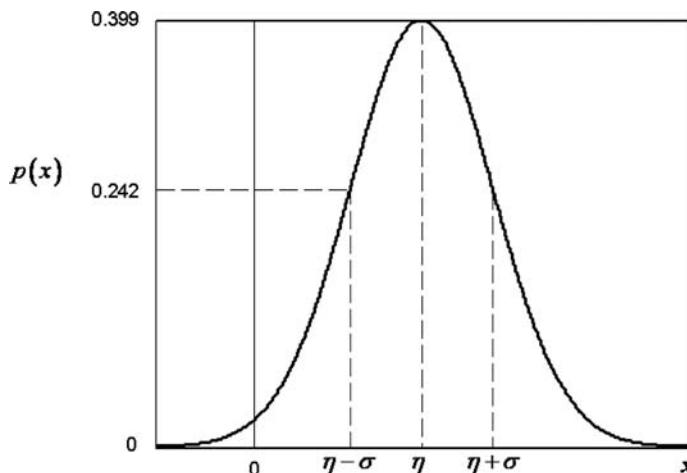


Fig. B.1 Gaussian probability density.

and variance  $\sigma^2$ . The statement that a random variable  $X$  is normally distributed with mean  $\eta$  and variance  $\sigma^2$  can be made as

$$X \text{ is } \mathbb{N}(\eta, \sigma^2) \quad (\text{B.43})$$

The Gaussian probability density and probability distribution functions are sketched in Figs. B.1 and B.2, respectively. If  $\sigma$  is used as the unit on the abscissa, then the area under the  $p(x)$  curve between  $\eta - k\sigma$  and  $\eta + k\sigma$  is given in Table B.1. The area under the  $p(x)$  curve between selected values  $x_1$  and  $x_2$

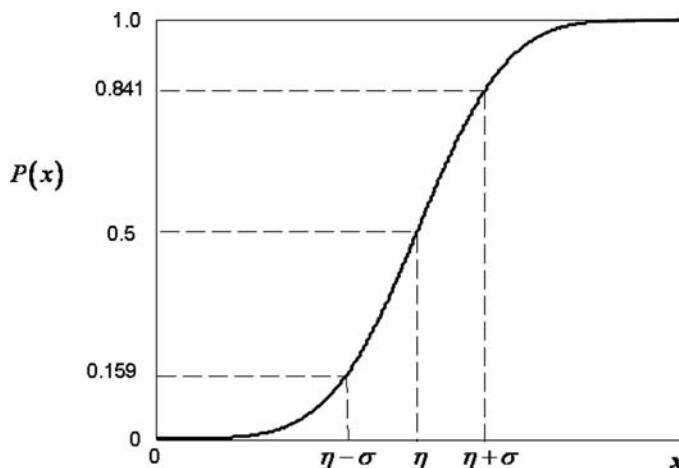


Fig. B.2 Gaussian probability distribution.

**Table B.1 Gaussian distribution probabilities**

$k$	0.6745	1.0	2.0	3.0
Area = $\int_{\eta-k\sigma}^{\eta+k\sigma} p(x) dx$	0.50	0.683	0.955	0.997
$P(\eta - k\sigma \leq X \leq \eta + k\sigma), \%$	50	68.3	95.5	99.7

corresponds to the probability that the random variable takes on a value in the range  $[x_1, x_2]$ .

Introducing the random variable  $U = (X - \eta)/\sigma$ , then

$$p(u) = \frac{1}{\sqrt{2\pi}} \exp\left[-\frac{u^2}{2}\right] \quad (\text{B.44})$$

which means that

$$U \text{ is } \mathbb{N}(0, 1) \quad (\text{B.45})$$

Values for the area under the curve  $p(u)$  from 0 to various selected values of  $u$  are tabulated in numerous textbooks on statistics.

An  $N$ -dimensional random vector  $X$  with probability density function  $p(x)$ ,  $x = [x_1 \ x_2 \ \cdots \ x_N]^T$ , is said to be normally distributed if  $p(x)$  has the form

$$p(x) = \frac{1}{(2\pi)^{N/2} |\Sigma|^{1/2}} \exp\left[-\frac{(X - \boldsymbol{\eta})^T \Sigma^{-1} (X - \boldsymbol{\eta})}{2}\right] \quad (\text{B.46})$$

$$E(X) = \boldsymbol{\eta} \quad (\text{B.47})$$

$$\text{Cov}(X) = E[(X - \boldsymbol{\eta})(X - \boldsymbol{\eta})^T] = \Sigma \quad (\text{B.48})$$

A normally distributed random vector with mean  $\boldsymbol{\eta}$  and covariance  $\Sigma$  is denoted as

$$X \text{ is } \mathbb{N}(\boldsymbol{\eta}, \Sigma) \quad (\text{B.49})$$

### B.2.2 *t* Distribution

The mean  $\bar{x}$  of  $N$  stochastically independent random variables taken from an ensemble of random variables with mean  $\eta$  and variance  $\sigma^2$  is normally distributed with mean  $\eta$  and variance  $\sigma^2/N$ , i.e.,

$$u = \frac{\bar{x} - \eta}{\sigma/\sqrt{N}} \text{ is } \mathbb{N}(0, 1) \quad (\text{B.50})$$

where

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i \quad (\text{B.51a})$$

## APPENDIX B: RANDOM VARIABLES

447

Replacing  $\sigma^2$  by the estimate

$$s^2 = \frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2 \quad (\text{B.51b})$$

the  $u$  variable is changed to a new variable

$$t = \frac{\bar{x} - \eta}{s/\sqrt{N-1}} \quad (\text{B.52})$$

The  $t$  distribution is independent of both parameters  $\eta$  and  $\sigma$ . It depends only on the number of statistical degrees of freedom  $f$  for  $s^2$ , i.e.,  $f = N - 1$ . The number of degrees of freedom indicates how many independent pieces of information are necessary to compute a given statistic. The  $s^2$  statistic computed from Eq. (B.51b) has  $N - 1$  degrees of freedom, because  $\sum_{i=1}^N (x_i - \bar{x}) = 0$  from (B.51a), which means that only  $N - 1$  of the  $x_i$  are independent. As  $N \rightarrow \infty$ , the  $t$  distribution approaches the distribution of  $u$ , which is  $\mathbb{N}(0, 1)$ .

### B.2.3 F Distribution

If the random variables  $u_1, u_2, \dots, u_f$  are each  $\mathbb{N}(0, 1)$ , then

$$\chi^2 = \sum_{i=1}^f u_i^2 \quad (\text{B.53})$$

is called a chi-squared random variable with  $f$  degrees of freedom. The ratio of two chi-squared random variables is a new random variable  $F$ , defined as

$$F = \frac{\chi_1^2/f_1}{\chi_2^2/f_2} \quad (\text{B.54})$$

with  $f_1$  and  $f_2$  degrees of freedom. The random variable  $F$  can take only positive values and has two parameters,  $f_1$  and  $f_2$ . The  $F$  distribution is tabulated in many statistics textbooks as a function of  $f_1$  and  $f_2$ .

### B.2.4 Partitioning Sum of Squares

The relationship among the quantities  $SS_T$ ,  $SS_R$ , and  $SS_E$  given in Chapter 5 is

$$SS_T = SS_R + SS_E \quad (\text{B.55})$$

To derive this relationship, start with the identity

$$z(i) - \bar{z} = [\hat{y}(i) - \bar{z}] + [z(i) - \hat{y}(i)] \quad (\text{B.56})$$

then square both sides and sum over  $N$  data points,

$$\begin{aligned} \sum_{i=1}^N [z(i) - \bar{z}]^2 &= \sum_{i=1}^N [\hat{y}(i) - \bar{z}]^2 + \sum_{i=1}^N [z(i) - \hat{y}(i)]^2 \\ &\quad + 2 \sum_{i=1}^N [\hat{y}(i) - \bar{z}][z(i) - \hat{y}(i)] \end{aligned} \quad (\text{B.57})$$

or

$$\begin{aligned} \sum_{i=1}^N [z(i) - \bar{z}]^2 &= \sum_{i=1}^N [\hat{y}(i) - \bar{z}]^2 + \sum_{i=1}^N [z(i) - \hat{y}(i)]^2 \\ &\quad + 2 \sum_{i=1}^N \hat{y}(i)[z(i) - \hat{y}(i)] - 2\bar{z} \sum_{i=1}^N [z(i) - \hat{y}(i)] \end{aligned} \quad (\text{B.58})$$

The normal equations (5.9c) show that the inner product of each regressor with the residual vector  $z - \hat{y}$  is zero. Since the first regressor is a vector of ones, the sum of the residuals must be zero, and the last summation on the right side of the preceding equation is zero. The estimated output vector  $\hat{y} = \mathbf{X}\hat{\boldsymbol{\theta}}$  is a linear sum of the regressors, so the inner product of  $\hat{y}$  with the residual vector is also zero. Therefore, the third summation on the right is also zero. With these simplifications,

$$\sum_{i=1}^N [z(i) - \bar{z}]^2 = \sum_{i=1}^N [\hat{y}(i) - \bar{z}]^2 + \sum_{i=1}^N [z(i) - \hat{y}(i)]^2 \quad (\text{B.59})$$

or, using the definitions in Eqs. (5.25)–(5.28),

$$SS_T = SS_R + SS_E \quad (\text{B.60})$$

### B.2.5 Alternate Expression for the Fisher Information Matrix

The Fisher information matrix  $\mathbf{M}$  is defined as

$$\mathbf{M} \equiv E \left[ \left( \frac{\partial \ell_n \mathbb{L}}{\partial \boldsymbol{\theta}} \right) \left( \frac{\partial \ell_n \mathbb{L}}{\partial \boldsymbol{\theta}} \right)^T \right] \quad (\text{B.61})$$

where  $\mathbb{L}$  is the likelihood function, equal to the probability density function of  $z$  given  $\boldsymbol{\theta}$ , i.e.,

$$\mathbb{L}(z; \boldsymbol{\theta}) \equiv p(z|\boldsymbol{\theta}) \quad (\text{B.62})$$

## APPENDIX B: RANDOM VARIABLES

449

so that

$$\mathbf{M} \equiv E \left\{ \left[ \frac{\partial \ell_n p(z|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right] \left[ \frac{\partial \ell_n p(z|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right]^T \right\} \quad (\text{B.63})$$

An alternate expression for  $\mathbf{M}$  can be derived by starting with the identity

$$\int_{-\infty}^{\infty} p(z|\boldsymbol{\theta}) dz = 1 \quad (\text{B.64})$$

which follows from the definition of conditional probability density functions. Assuming that  $\ell_n p(z|\boldsymbol{\theta})$  is sufficiently smooth, the gradient with respect to  $\boldsymbol{\theta}$  is

$$\frac{\partial \ell_n p(z|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \frac{1}{p(z|\boldsymbol{\theta})} \left[ \frac{\partial p(z|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right] \quad (\text{B.65})$$

which can be rearranged to

$$\frac{\partial p(z|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = p(z|\boldsymbol{\theta}) \frac{\partial \ell_n p(z|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \quad (\text{B.66})$$

Differentiating Eq. (B.64) twice with respect to  $\boldsymbol{\theta}$ , and using Eq. (B.66),

$$\int_{-\infty}^{\infty} \left[ \frac{\partial \ell_n p(z|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \frac{\partial \ell_n p(z|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}^T} + \frac{\partial^2 \ell_n p(z|\boldsymbol{\theta})}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} \right] p(z|\boldsymbol{\theta}) dz = \mathbf{0} \quad (\text{B.67})$$

Introducing expectation operator notation, it must be that

$$E \left\{ \left[ \frac{\partial \ell_n p(z|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right] \left[ \frac{\partial \ell_n p(z|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right]^T + \frac{\partial^2 \ell_n p(z|\boldsymbol{\theta})}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} \right\} = \mathbf{0}$$

or

$$E \left\{ \left[ \frac{\partial \ell_n p(z|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right] \left[ \frac{\partial \ell_n p(z|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right]^T \right\} = -E \left[ \frac{\partial^2 \ell_n p(z|\boldsymbol{\theta})}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} \right] \quad (\text{B.68})$$

Combining Eqs. (B.61), (B.62), and (B.68),

$$\mathbf{M} \equiv E \left[ \left( \frac{\partial \ell_n \mathbb{L}}{\partial \boldsymbol{\theta}} \right) \left( \frac{\partial \ell_n \mathbb{L}}{\partial \boldsymbol{\theta}} \right)^T \right] = -E \left( \frac{\partial^2 \ell_n \mathbb{L}}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} \right) \quad (\text{B.69})$$

The term on the far right side of Eq. (B.69) is the alternate expression for the Fisher information matrix  $\mathbf{M}$ .

### B.2.6 Derivation of the Cramér-Rao Lower Bound

The theoretical minimum of the estimated parameter covariance matrix for an unbiased estimator, called the Cramér-Rao lower bound, equals the inverse of the Fisher information matrix. This means that the estimated parameter covariance matrix satisfies the inequality

$$\text{Cov}(\hat{\boldsymbol{\theta}}) \geq \mathbf{M}^{-1} \quad (\text{B.70})$$

To derive this expression, start with the following statement that is true by the definition of an unbiased estimator,

$$E(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}) = \int_{-\infty}^{\infty} (\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}) p(z|\boldsymbol{\theta}) dz = \mathbf{0} \quad (\text{B.71})$$

Differentiating with respect to  $\boldsymbol{\theta}$  gives

$$\int_{-\infty}^{\infty} -\mathbf{I} p(z|\boldsymbol{\theta}) dz + \int_{-\infty}^{\infty} (\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}) \left[ \frac{\partial p(z|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right]^T dz = \mathbf{0} \quad (\text{B.72})$$

where  $\mathbf{I}$  is an identity matrix. Since  $\int_{-\infty}^{\infty} p(z|\boldsymbol{\theta}) dz = 1$ ,

$$\int_{-\infty}^{\infty} (\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}) \left[ \frac{\partial p(z|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right]^T dz = \mathbf{I} \quad (\text{B.73})$$

Introducing expectation operator notation, and using Eq. (B.66),

$$E \left\{ (\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}) \left[ \frac{\partial \ln p(z|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right]^T \right\} = \mathbf{I} \quad (\text{B.74})$$

The following lemma is now required. The derivation given here is from Ref. 1.

#### Lemma

Let  $\mathbf{X}$  and  $\mathbf{Y}$  be two random vectors with the same dimension. Then,

$$E(\mathbf{X}\mathbf{X}^T) \geq E(\mathbf{X}\mathbf{Y}^T) [E(\mathbf{Y}\mathbf{Y}^T)]^{-1} E(\mathbf{Y}\mathbf{X}^T) \quad (\text{B.75})$$

*Proof:* Let  $\mathbf{Q}$  be an arbitrary nonrandom matrix with conformable dimensions. Then, since any covariance matrix must be nonnegative,

$$E[(\mathbf{X} - \mathbf{Q}\mathbf{Y})(\mathbf{X} - \mathbf{Q}\mathbf{Y})^T] \geq \mathbf{0} \quad (\text{B.76})$$

## APPENDIX B: RANDOM VARIABLES

451

Expanding the left side of the preceding equation,

$$\begin{aligned} E(\mathbf{X}\mathbf{X}^T) - \mathbf{Q}E(\mathbf{Y}\mathbf{X}^T) - E(\mathbf{X}\mathbf{Y}^T)\mathbf{Q}^T + \mathbf{Q}E(\mathbf{Y}\mathbf{Y}^T)\mathbf{Q}^T &\geq 0 \\ E(\mathbf{X}\mathbf{X}^T) &\geq \mathbf{Q}E(\mathbf{Y}\mathbf{X}^T) + E(\mathbf{X}\mathbf{Y}^T)\mathbf{Q}^T - \mathbf{Q}E(\mathbf{Y}\mathbf{Y}^T)\mathbf{Q}^T \end{aligned} \quad (\text{B.77})$$

Now choose

$$\mathbf{Q} = E(\mathbf{X}\mathbf{Y}^T)[E(\mathbf{Y}\mathbf{Y}^T)]^{-1} \quad (\text{B.78})$$

so that the two terms on the far right side of Eq. (B.77) cancel. Substituting Eq. (B.78) into Eq. (B.77),

$$E(\mathbf{X}\mathbf{X}^T) \geq E(\mathbf{X}\mathbf{Y}^T)[E(\mathbf{Y}\mathbf{Y}^T)]^{-1}E(\mathbf{Y}\mathbf{X}^T) \quad (\text{B.79})$$

This ends the proof of the lemma.

Using Eq. (B.79) with

$$\mathbf{X} \equiv (\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}) \quad \mathbf{Y} \equiv \frac{\partial \ell_n p(\mathbf{z}|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}$$

and invoking Eq. (B.73) results in

$$E[(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})^T] \geq \mathbf{I} \left\{ E \left[ \frac{\partial \ell_n p(\mathbf{z}|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \frac{\partial \ell_n p(\mathbf{z}|\boldsymbol{\theta})}{\partial \boldsymbol{\theta}^T} \right] \right\}^{-1} \mathbf{I} \quad (\text{B.80})$$

Combining this with Eq. (B.63) gives

$$\text{Cov}(\hat{\boldsymbol{\theta}}) \equiv E[(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})^T] \geq \mathbf{M}^{-1} \quad (\text{B.81})$$

which is the Cramér-Rao inequality stated in Eq. (B.70).

## B.3 Random Process Theory

### B.3.1 Random Process

A random process, also called a stochastic process, is a collection of random variables  $X(t)$  indexed by the parameter  $t$  from a set  $T$ . A random process is said to be continuous if  $T$  is a connected set, and discrete if  $T$  is a finite set; i.e.,  $T = (t_1, t_2, \dots, t_N)$ . A realization or sample of a random process is an assignment of specific values to  $X(t)$  for each  $t$  of  $T$ , from among the possible values of  $X(t)$  at each  $t$ . A complete description of a general random process would require specification of all possible joint probability density functions  $p[X(t_1), X(t_2), \dots, X(t_N)]$ .

### B.3.2 Stationary Random Process

A random process  $X(t)$  is said to be a stationary random process if, for each  $(t_1, t_2, \dots, t_N)$  the joint probability distribution of  $[X(t_1 + \tau), X(t_2 + \tau), \dots, X(t_N + \tau)]$  is the same for any  $\tau$ . The properties of a stationary process are as follows:

- 1) The mean value  $\eta = E[X(t)]$  is a constant, independent of  $t$ .
- 2) The covariance matrix  $\text{Cov}[X(t_1), X(t_2)] = E\{[X(t_1) - \eta][X(t_2) - \eta]^T\}$  exists.
- 3) The covariance matrix  $\text{Cov}[X(t_1), X(t_2)]$  depends only on  $t_1 - t_2$  and not on the magnitudes of  $t_1$  or  $t_2$ .

### B.3.3 White Noise Process

A discrete random process  $X(t)$  is said to be a white noise process if for  $E[X(t)] = 0$ , the covariance matrix  $\text{Cov}[X(t_1), X(t_2)]$  can be expressed as

$$\text{Cov}[X(t_1), X(t_2)] = E[X(t_1), X^T(t_2)] = \begin{cases} Q(t_1) & t_1 = t_2 \\ \mathbf{0} & t_1 \neq t_2 \end{cases} \quad (\text{B.82})$$

where  $Q(t_1)$  is a positive semidefinite matrix. The preceding condition implies that  $X(t_1)$  and  $X(t_2)$  are uncorrelated for any  $t_1 \neq t_2$ .

### B.3.4 Correlation Functions

For stationary random processes  $X(t)$  and  $Y(t)$ , the mean values are constant, independent of  $t$ , i.e.,

$$\eta_x = E[X(t)] \quad \text{and} \quad \eta_y = E[Y(t)] \quad (\text{B.83})$$

The covariance functions are also independent of  $t$ , and depend only on the time shift  $\tau$ . For arbitrary  $t$  and  $\tau$  they can be expressed as

$$\begin{aligned} \Sigma_x(\tau) &= E\{[X(t) - \eta_x][X(t + \tau) - \eta_x]\} = E[X(t)X(t + \tau)] - \eta_x^2 \\ \Sigma_y(\tau) &= E\{[Y(t) - \eta_y][Y(t + \tau) - \eta_y]\} = E[Y(t)Y(t + \tau)] - \eta_y^2 \\ \Sigma_{xy}(\tau) &= E\{[X(t) - \eta_x][Y(t + \tau) - \eta_y]\} = E[X(t)Y(t + \tau)] - \eta_x \eta_y \end{aligned} \quad (\text{B.84})$$

where

$$\begin{aligned} E[X(t)X(t + \tau)] &\equiv R_{xx}(\tau) \\ E[Y(t)Y(t + \tau)] &\equiv R_{yy}(\tau) \\ E[X(t)Y(t + \tau)] &\equiv R_{xy}(\tau) \end{aligned} \quad (\text{B.85})$$

and where  $R_{xx}(\tau)$  and  $R_{yy}(\tau)$  are known as the autocorrelation functions of  $X(t)$  and  $Y(t)$ , respectively, and  $R_{xy}(\tau)$  is called the cross-correlation function

## APPENDIX B: RANDOM VARIABLES

453

of  $X(t)$  and  $Y(t)$ . It follows that for a zero-mean stationary random process  $X(t)$ ,

$$\Sigma_x(0) = E[X(t)X(t)] = \text{Var}[X(t)] = \mathcal{R}_{xx}(0) \quad (\text{B.86})$$

The autocorrelation functions are even functions,

$$\begin{aligned} \mathcal{R}_{xx}(-\tau) &= \mathcal{R}_{xx}(\tau) \\ \mathcal{R}_{yy}(-\tau) &= \mathcal{R}_{yy}(\tau) \end{aligned} \quad (\text{B.87})$$

whereas the cross-correlation functions satisfy

$$\mathcal{R}_{xy}(-\tau) = \mathcal{R}_{yx}(\tau) \quad (\text{B.88})$$

The autocorrelation function for white noise is a Dirac delta function at  $\tau = 0$ , indicating that values at each time  $t$  are uncorrelated with values at other times. In the general case, when a random process has an autocorrelation function that is some even function of  $\tau$ , the random process is called colored noise.

### B.3.5 Spectral Density Function

The stationary random processes  $X(t)$  and  $Y(t)$  have autocorrelation functions  $\mathcal{R}_{xx}(\tau)$  and  $\mathcal{R}_{yy}(\tau)$ , respectively, and cross-correlation function  $\mathcal{R}_{xy}(\tau)$ . The Fourier transforms of these functions are the power spectral density functions,<sup>2</sup>

$$\begin{aligned} S_{xx}(\omega) &= \int_{-\infty}^{\infty} \mathcal{R}_{xx}(\tau) e^{-j\omega\tau} d\tau \\ S_{yy}(\omega) &= \int_{-\infty}^{\infty} \mathcal{R}_{yy}(\tau) e^{-j\omega\tau} d\tau \\ S_{xy}(\omega) &= \int_{-\infty}^{\infty} \mathcal{R}_{xy}(\tau) e^{-j\omega\tau} d\tau \end{aligned} \quad (\text{B.89})$$

where  $S_{xx}(\omega)$  and  $S_{yy}(\omega)$  are called two-sided power spectral density functions (or spectral densities) of the random processes  $X(t)$  and  $Y(t)$ , respectively, and  $S_{xy}(\omega)$  is called the two-sided cross-spectral density function (or cross-spectral density) between  $X(t)$  and  $Y(t)$ . The frequency  $\omega$  in rad/s varies over the range  $(-\infty, \infty)$ .

From the symmetry of the autocorrelation function, it follows that

$$\begin{aligned} S_{xx}(-\omega) &= S_{xx}(\omega) \\ S_{yy}(-\omega) &= S_{yy}(\omega) \\ S_{xy}(-\omega) &= S_{xy}^*(\omega) = S_{yx}(\omega) \end{aligned} \quad (\text{B.90})$$

Because the autocorrelation functions are even functions of  $\tau$ , the two-sided power spectral densities are real-valued even functions of  $\omega$ ,

$$\begin{aligned} S_{xx}(\omega) &= \int_{-\infty}^{\infty} \mathcal{R}_{xx}(\tau) \cos \omega\tau d\tau = 2 \int_0^{\infty} \mathcal{R}_{xx}(\tau) \cos \omega\tau d\tau \\ S_{yy}(\omega) &= \int_{-\infty}^{\infty} \mathcal{R}_{yy}(\tau) \cos \omega\tau d\tau = 2 \int_0^{\infty} \mathcal{R}_{yy}(\tau) \cos \omega\tau d\tau \end{aligned} \quad (\text{B.91})$$

## References

<sup>1</sup>Maine, R. E., and Iliff, K. W., "Identification of Dynamic Systems, Theory and Formulation," NASA RP 1138, 1985.

<sup>2</sup>Bendat, J. S., and Piersol, A. G., *Random Data Analysis and Measurement Procedures*, 2nd ed., Wiley, New York, 1986.

## Appendix C

### Reference Information

#### C.1 Properties of Air and the Atmosphere

The forces and moments acting on a body due to fluid flowing over the body depend in part on the properties of the fluid. For an airplane flying in the atmosphere, the properties of air are relevant. These properties generally change with weather conditions and also with altitude.

The air properties of interest are the number of air molecules per unit volume, quantified by the air density, the resistance of air molecules to relative motion with respect to neighboring air molecules, quantified by the viscosity, and the speed with which air molecules transmit pressure disturbances, quantified by the speed of sound. These air properties can be modeled as functions of static air pressure and temperature,

$$\rho = \frac{p}{RT} \quad (C.1)$$

$$\frac{\mu}{\mu_o} = \left( \frac{T}{T_o} \right)^{3/4} \quad (C.2)$$

$$a = (\gamma RT)^{1/2} \quad (C.3)$$

where subscript  $o$  denotes a reference condition, and

$\rho$  = air density, slug/ft<sup>3</sup>

$p$  = static pressure, lbf/ft<sup>2</sup>

$T$  = temperature, °R

$R$  = 1716.50 ft<sup>2</sup>/(s<sup>2</sup> °R)

$\mu$  = viscosity, slug/( ft s)

$a$  = speed of sound, ft/s

$\gamma$  = ratio of specific heats = 1.402 at  $p = 1$  atm and  $T = 520$  °R

To standardize results by removing weather differences among different days, a standard set of atmospheric conditions has been defined, called the standard

atmosphere. Using the standard atmosphere, comparisons can be properly made using the same atmospheric conditions. Table C.1 shows conditions for the 1976 U.S. standard atmosphere.<sup>1</sup> The data in Table C.1 show that the air density, static pressure, and temperature all decrease with increasing altitude. Warmer, denser, higher-pressure air lies at the bottom of the atmosphere.

## C.2 Elementary Aerodynamics

Components of the aerodynamic force and moment acting on an aircraft can be written in terms of nondimensional coefficients as follows:

*Forces:*

$$\begin{array}{ll} \text{Body axes} & \text{Stability axes} \\ X = \bar{q} S C_X & D = \bar{q} S C_D = -X_S \end{array} \quad (\text{C.5})$$

$$Z = \bar{q} S C_Z \quad L = \bar{q} S C_L = -Z_S \quad (\text{C.6})$$

$$Y = \bar{q} S C_Y \quad Y = \bar{q} S C_Y = Y_S \quad (\text{C.7})$$

*Moments:*

$$L = \bar{q} b S C_l \quad (\text{C.8})$$

$$M = \bar{q} c S C_m \quad (\text{C.9})$$

$$N = \bar{q} b S C_n \quad (\text{C.10})$$

**Table C.1 U.S. standard atmosphere, 1976**

Alt., ft	T, °R	p, lb/ft <sup>2</sup>	$\rho/\rho_0$	$\rho, \text{lb}\cdot\text{s}^2/\text{ft}^4$	a, ft/s
0	518.67	2116.2	1.00000	0.0023769	1116.44
1,000	515.10	2040.8	0.97107	0.0023082	1112.60
2,000	511.54	1967.7	0.94278	0.0022409	1108.76
3,000	507.97	1896.7	0.91513	0.0021751	1104.89
4,000	504.41	1827.7	0.88811	0.0021109	1100.98
<b>5,000</b>	500.84	1760.9	0.86170	0.0020482	1097.08
6,000	497.28	1696.0	0.83590	0.0019869	1093.18
7,000	493.71	1633.1	0.81070	0.0019270	1089.27
8,000	490.15	1572.0	0.78609	0.0018684	1085.33
9,000	486.59	1512.9	0.76206	0.0018113	1081.36
<b>10,000</b>	483.02	1455.6	0.73859	0.0017555	1077.40
11,000	479.46	1400.0	0.71568	0.0017011	1073.43
12,000	475.88	1346.2	0.69333	0.0016480	1069.42
13,000	472.34	1294.1	0.67151	0.0015961	1065.42
14,000	468.78	1243.6	0.65022	0.0015455	1061.38

*Continued*

## APPENDIX C: REFERENCE INFORMATION

457

Table C.1 U.S. standard atmosphere, 1976 (Continued)

Alt., ft	T, °R	p, lb/ft <sup>2</sup>	$\rho/\rho_0$	$\rho$ , lb-s <sup>2</sup> /ft <sup>4</sup>	a, ft/s
<b>15,000</b>	465.22	1194.8	0.62946	0.0014962	1057.35
16,000	461.66	1147.5	0.60921	0.0014480	1053.31
17,000	458.09	1101.7	0.58946	0.0014011	1049.25
18,000	454.53	1057.5	0.57021	0.0013553	1045.14
19,000	450.97	1014.7	0.55144	0.0013107	1041.04
<b>20,000</b>	447.42	973.3	0.53317	0.0012673	1036.94
21,000	443.86	933.2	0.51534	0.0012249	1032.81
22,000	440.30	894.6	0.49798	0.0011837	1028.64
23,000	436.74	857.2	0.48108	0.0011435	1024.48
24,000	433.18	821.2	0.46462	0.0011044	1020.31
<b>25,000</b>	429.62	786.3	0.44859	0.0010663	1016.11
26,000	426.07	752.7	0.43300	0.0010292	1011.88
27,000	422.51	720.1	0.41782	0.0009931	1007.64
28,000	418.95	688.9	0.40305	0.0009580	1003.41
29,000	415.40	658.8	0.38869	0.0009239	999.15
<b>30,000</b>	411.84	629.7	0.37473	0.0008907	994.85
31,000	408.28	601.6	0.36115	0.0008584	990.55
32,000	404.73	574.6	0.34795	0.0008270	986.22
33,000	401.17	548.5	0.33513	0.0007966	981.89
34,000	397.62	523.5	0.32267	0.0007670	977.53
<b>35,000</b>	394.06	499.3	0.31058	0.0007382	973.13
36,000	390.51	476.1	0.29883	0.0007103	968.73
37,000	389.97	453.9	0.28525	0.0006780	968.08
38,000	389.97	432.6	0.27191	0.0006463	968.08
39,000	389.97	412.4	0.25920	0.0006161	968.08
<b>40,000</b>	389.97	393.1	0.24708	0.0005873	968.08
42,000	389.97	357.2	0.22452	0.0005336	968.08
44,000	389.97	324.6	0.20402	0.0004849	968.08
46,000	389.97	295.0	0.18540	0.0004407	968.08
48,000	389.97	268.1	0.16848	0.0004005	968.08
<b>50,000</b>	389.97	243.6	0.15311	0.0003639	968.08
52,000	389.97	221.4	0.13914	0.0003307	968.08
54,000	389.97	201.2	0.12645	0.0003006	968.08
56,000	389.97	182.8	0.11492	0.0002731	968.08
58,000	389.97	166.2	0.10444	0.0002482	968.08
<b>60,000</b>	389.97	151.0	0.09492	0.0002256	968.08
65,000	389.97	118.9	0.07475	0.0001777	968.08
70,000	392.25	93.7	0.05857	0.0001392	970.90
75,000	394.97	74.0	0.04591	0.0001091	974.28
80,000	397.69	58.5	0.03606	0.0000857	977.62
85,000	400.42	46.3	0.02837	0.0000674	980.94
90,000	403.14	36.8	0.02236	0.0000531	984.28
95,000	405.85	29.2	0.01765	0.0000419	987.60
<b>100,000</b>	408.57	23.3	0.01396	0.0000332	990.91

where

$$\bar{q} = \text{dynamic pressure} = \frac{1}{2} \rho V^2 \text{lbf/ft}^2$$

$\rho$  = air density,  $\rho(P,T) = 0.002377 \text{ slug/ft}^3$  at sea level

$V$  = airspeed, ft/s

$S$  = wing reference area,  $\text{ft}^2$

$b$  = wing span, ft

$\bar{c}$  = mean aerodynamic chord (MAC), ft

$$\bar{c} \equiv \frac{1}{S} \int_{-b/2}^{b/2} [c(y)]^2 dy \quad (\text{C.11})$$

Thrust can also be written in terms of a nondimensional thrust coefficient,

$$T = \bar{q} S C_T \quad (\text{C.12})$$

When the thrust force comes from a jet engine, the aerodynamic nondimensionalization in Eq. (C.12) can be inadequate. In that case, it is more common to characterize the thrust force directly. For propeller aircraft, the characterization in Eq. (C.12) is appropriate because the propulsion force arises from rotating airfoils that comprise the propeller blades.

Reference geometry definitions for a typical airplane are shown in Fig. C.1. Wing chord  $c$  is the distance from the leading edge of the wing airfoil section to the trailing edge. The straight line connecting the leading edge and the trailing edge of the airfoil is called the chord line. Wing span  $b$  is the distance between wing tips. When the wing is not rectangular, the chord changes along the wing span, as in Fig. C.1. The mean aerodynamic chord  $\bar{c}$  is computed from Eq. (C.11). The mean aerodynamic chord is used as a reference length for nondimensionalization and also as a reference to specify longitudinal location of the c.g. Wing reference area  $S$  normally includes all the shaded area shown in Fig. C.1.

The defining relations in Eqs. (C.5)–(C.12) make all the coefficients nondimensional. Consequently, values of the nondimensional coefficients are tied to the geometric quantities used in the nondimensionalization. For example, if the reference area  $S$  is changed, the nondimensional force and moment coefficients change, but of course the aerodynamic forces and moments do not.

The nondimensional aerodynamic coefficients (and the corresponding aerodynamic forces and moments) are classified as longitudinal and lateral as follows:

*Longitudinal:*

$$C_X, C_Z, C_m \text{ or } C_D, C_L, C_m \quad (\text{C.13})$$

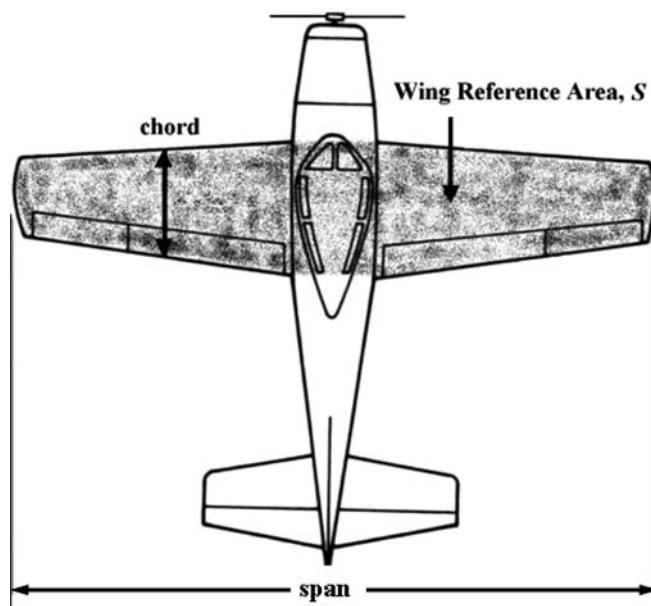


Fig. C.1 Airplane geometry definitions.

*Lateral:*

$$C_Y, C_l, C_n \quad (\text{C.14})$$

The preceding classification arises from the fact that the longitudinal forces and moments affect the airplane motion in the body-axis  $Oxz$  plane defined at the start of a maneuver, and therefore relate to longitudinal motion. The lateral forces and moments affect lateral airplane motion outside the body-axis  $Oxz$  plane defined at the start of the maneuver. Longitudinal motion is also called symmetric flight, and lateral motion is called asymmetric flight.

### C.3 Mass/Inertia Properties

Aircraft mass and longitudinal c.g. position are found by weighing the aircraft using scales at each point where the aircraft contacts the ground. Longitudinal c.g. location is often specified as a location along the mean aerodynamic chord. The mean aerodynamic chord can always be located at some location along the span of the wing. Using this location along the wing span, the c.g. location is then specified as some fraction of the distance along the mean aerodynamic chord, e.g.,  $0.35\bar{c}$ , which would correspond to a location 35% of the distance from the leading edge to the trailing edge of the mean aerodynamic chord. It is also common practice to locate the c.g. by specifying right-handed Cartesian coordinates relative to a datum. Normally, the coordinates

are called fuselage station ( $+x$  aft), butt line ( $+y$  out the right wing), and water line ( $+z$  up).

There are two ways to find the inertial properties for an aircraft, which are quantified by the elements of the inertia tensor. The first method, which might be called the analytical method, involves treating the airplane as a sum of individual parts, such as the fuselage, wings, vertical tail, etc. Inertia tensor elements for each individual part are summed together using the parallel axis theorem,

$$I_p = I + md^2 \quad (\text{C.15})$$

where  $I$  is the moment of inertia about a given axis,  $m$  is the mass, and  $I_p$  is the moment of inertia about a parallel axis displaced a perpendicular distance  $d$  away. Often the shapes of the individual parts are approximated by simple shapes whose inertial properties are known, assuming constant mass density. An example would be approximating the fuselage as a cylinder. This method is also used to modify known inertial properties for added equipment, stores, instrumentation, cargo, or crew/passengers. Modern computer-aided design (CAD) software can be used in this way to compute mass and inertia properties very accurately.

The second method, which might be called the experimental method, involves a ground test where the airplane is set up to oscillate about one of the body axes with negligible damping and a known spring constant. The measured time for the period of the oscillations can be used to calculate the mass moment of inertia about the axis of rotation for the oscillation. The expression for calculating the moment of inertia about the rotation axis is

$$I = K(m_a + m)T^2 + I_a \quad (\text{C.16})$$

where subscript  $a$  denotes properties of the measurement apparatus,  $K$  is the spring constant of the apparatus,  $m$  is the aircraft mass, and  $T$  is the period. For small airplane models, this can be done using a pendulum apparatus, carrying out the experiment for each body axis.

To obtain the cross-product of inertia  $I_{xz}$ , one additional test is done to determine the mass moment of inertia about an intermediate axis in the body-axis  $Oxz$  plane, at an angle  $\xi$  to the body  $x$  axis. Then  $I_{xz}$  is computed from

$$I_{xz} = \frac{I_\xi - I_x \cos^2 \xi - I_z \sin^2 \xi}{\sin 2\xi} \quad (\text{C.17})$$

where  $I_\xi$  denotes the moment of inertia about the intermediate axis.

The cross product of inertia  $I_{xz}$  arises mostly from the vertical tail and therefore has a much lower magnitude than the diagonal elements of the inertia matrix  $I_x$ ,  $I_y$ , and  $I_z$ . This can be seen in the inertia values for the F-16 simulation, given in Appendix D, Table D.2.

## APPENDIX C: REFERENCE INFORMATION

461

### C.4 Greek Alphabet and Conversion Factors

**Table C.2 Greek alphabet**

Letter	Lowercase	Uppercase
Alpha	$\alpha$	Α
Beta	$\beta$	Β
Gamma	$\gamma$	Γ
Delta	$\delta$	Δ
Epsilon	$\varepsilon$	Ε
Zeta	$\zeta$	Ζ
Eta	$\eta$	Η
Theta	$\theta$	Θ
Iota	$\iota$	Ι
Kappa	$\kappa$	Κ
Lambda	$\lambda$	Λ
Mu	$\mu$	Μ
Nu	$\nu$	Ν
Xi	$\xi$	Ξ
Omicron	$\ο$	Ο
Pi	$\pi$	Π
Rho	$\rho$	Ρ
Sigma	$\sigma$	Σ
Tau	$\tau$	Τ
Upsilon	$\upsilon$	Υ
Phi	$\phi$	Φ
Chi	$\chi$	Χ
Psi	$\psi$	Ψ
Omega	$\omega$	Ω

*Conversion factors:*

$$1 \text{ ft} = 0.3048 \text{ m}$$

$$1 \text{ nautical mile} = 1.152 \text{ mile} = 6080.2 \text{ ft}$$

$$1 \text{ kt} = 1.152 \text{ mph} = 1.689 \text{ ft/s} = 0.5148 \text{ m/s}$$

$$1 \text{ slug} = 32.174 \text{ lbm} = 14.59 \text{ kg}$$

$$1 \text{ slug/ft}^3 = 515.2 \text{ kg/m}^3$$

$$1 \text{ lbf} = 4.448 \text{ N}$$

$$1 \text{ atm} = 2116.2 \text{ lbf/ft}^2$$

$$1 \text{ hp} = 550 \text{ ft-lbf/s}$$

$$1 \text{ deg} = 0.01745 \text{ rad}$$

$$g = 32.174 \text{ ft/s}^2 = 9.806 \text{ m/s}^2 \text{ at sea level}$$

### Reference

<sup>1</sup>*U.S. Standard Atmosphere: 1976*, National Oceanic and Atmospheric Administration, NASA, and U.S. Air Force, Washington, DC, 1976.

This page intentionally left blank

## Appendix D

### F-16 Nonlinear Simulation

This appendix provides a detailed description of a nonlinear simulation for the F-16 fighter aircraft. The simulation is written completely in MATLAB<sup>®</sup>, and is included as part of the software package associated with this textbook. It is essentially a MATLAB<sup>®</sup> version of the FORTRAN simulation given in Appendix A of Ref. 1. The simulation is based on information in Ref. 2, which includes a wind-tunnel aerodynamic database for a 16% scale model of the F-16 aircraft, and a ground-test database for engine thrust.

The following subsections provide details for various aspects of the F-16 nonlinear simulation, including the F-16 aircraft, equations of motion, aerodynamic model, engine model, and analysis tools. A list of the files included in the F-16 nonlinear simulation package is given in Table D.1. The F-16 nonlinear simulation documented here is one of several included in Ref. 3.

#### D.1 F-16 Aircraft

The F-16 is a single-seat, multirole fighter with a blended wing/body and a cropped delta wing planform with leading-edge sweep of 40 deg. Figure D.1 shows a three-view drawing of the F-16. The wing is fitted with leading-edge flaps and trailing-edge flaperons (flaps/ailerons). Tail surfaces are swept and cantilevered. The horizontal stabilator is composed of two all-moving tail plane halves, and the vertical tail is fitted with a trailing-edge rudder. Thrust is provided by one General Electric F110-GE-100 or Pratt & Whitney F100-PW-220 afterburning turbofan engine mounted in the rear fuselage.

The aircraft was modeled with controls for throttle  $\delta_{th}$ , stabilator  $\delta_s$ , aileron  $\delta_a$ , and rudder  $\delta_r$ . Speed brake and flaps were assumed fixed at zero deflection. Throttle deflection was limited to the range  $0 \leq \delta_{th} \leq 1$ , stabilator deflection was limited to  $-25 \geq \delta_s \geq 25$  deg, aileron deflection was limited to  $-21.5 \geq \delta_a \geq 21.5$  deg, and rudder deflection was limited to  $-30 \leq \delta_r \leq 30$  deg. These limits represent the physical stops.

#### D.2 Equations of Motion

The aircraft was assumed rigid with constant mass density and symmetry about the  $Oxz$  plane in body axes. Thrust was assumed to act along the  $x$  body

Table D.1 F-16 simulation files

F16_NLS_V1.1 Directory	
ab3.m	f16_demo.diary
atm.m	f16_demo.m
avds_matlab_f16.m	f16_deq.m
avds_matlab_setup.m	f16_engine.m
clo.m	f16_engine_setup.m
cmo.m	f16_fltdatsel.m
cno.m	f16_lores.txt
cnvrg.m	f16_massprop.m
cxo.m	f16_trm.m
czo.m	gen_f16_model.m
dampder.m	grad.m
dlda.m	ic_ftrm.m
dldr.m	Inze.m
dnda.m	mksqw.m
dndr.m	pdot.m
f16.m	rk2.m
f16.sim.ini	rk4.m
f16.txt	rtau.m
f16_aero.m	solve.m
f16_aero_setup.m	tgear.m

axis and through the center of gravity. A stationary atmosphere was assumed, with aircraft flight limited to altitudes less than 50,000 ft and subsonic speeds. The earth's curvature was ignored and the earth was assumed to be fixed in inertial space, so that earth axes were considered inertial axes. The gravity field was assumed to be uniform.

Full nonlinear six-degree-of-freedom rigid-body translational and rotational aircraft motion was modeled, neglecting relative motion of the aircraft internal components, structural distortion, and sloshing of liquid fuel. The gyroscopic effects caused by rotating engine turbomachinery were included. Airplane rotation conventions and control surface deflections follow a standard right-hand rule sign convention, described in Chapter 3.

The six nonlinear rigid-body equations of motion in body axes, derived in Chapter 3, were used to model the nonlinear aircraft dynamics in translational and rotational motion. These equations are

$$\begin{aligned}\dot{u} &= rv - qw - g \sin \theta + \frac{\bar{q}SC_X + T}{m} \\ \dot{v} &= pw - ru + g \cos \theta \sin \phi + \frac{\bar{q}SC_Y}{m} \\ \dot{w} &= qu - pv + g \cos \theta \cos \phi + \frac{\bar{q}SC_Z}{m}\end{aligned}\quad (\text{D.1})$$

## APPENDIX D: F-16 NONLINEAR SIMULATION

465

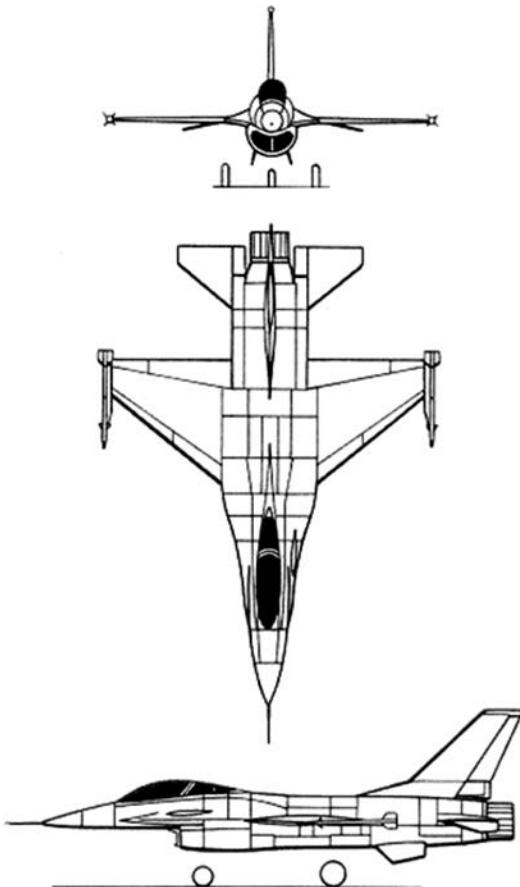


Fig. D.1 F-16 aircraft.

$$\begin{aligned}\dot{p}I_x - \dot{r}I_{xz} &= \bar{q}SbC_l - qr(I_z - I_y) + qpI_{xz} \\ \dot{q}I_y - \dot{s}I_{yz} &= \bar{q}ScC_m - pr(I_x - I_z) - (p^2 - r^2)I_{xz} + rh_{\text{eng}} \\ \dot{r}I_z - \dot{p}I_{xz} &= \bar{q}SbC_n - pq(I_y - I_x) - qrI_{xz} - qh_{\text{eng}}\end{aligned}\quad (\text{D.2})$$

where  $h_{\text{eng}}$  is the magnitude of the angular momentum vector for the rotating mass of the engine, assumed to act along the positive  $x$  body axis,

$$\mathbf{h}_{\text{eng}} = [ h_{\text{eng}} \quad 0 \quad 0 ]^T = [ I_p \Omega_p \quad 0 \quad 0 ]^T \quad (\text{D.3})$$

As discussed in Chapter 3, it is preferable to write the translational equations in terms of  $V$ ,  $\alpha$ , and  $\beta$  instead of  $u$ ,  $v$ , and  $w$ , because  $V$ ,  $\alpha$ , and  $\beta$  can be measured directly on a real aircraft, and have a more direct relationship to piloting and the

aerodynamic forces and moments. The relationships between  $V$ ,  $\alpha$ ,  $\beta$ , and  $u$ ,  $v$ ,  $w$  are

$$\begin{aligned} V &= \sqrt{u^2 + v^2 + w^2} \\ \alpha &= \tan^{-1}\left(\frac{w}{u}\right) \\ \beta &= \sin^{-1}\left(\frac{v}{V}\right) \end{aligned} \quad (\text{D.4})$$

and

$$\begin{aligned} u &= V \cos \alpha \cos \beta \\ v &= V \sin \beta \\ w &= V \sin \alpha \cos \beta \end{aligned} \quad (\text{D.5})$$

Differentiating Eqs. (D.4) with respect to time gives

$$\begin{aligned} \dot{V} &= \frac{u\dot{u} + v\dot{v} + w\dot{w}}{V} \\ \dot{\alpha} &= \frac{u\dot{w} - w\dot{u}}{u^2 + w^2} \\ \dot{\beta} &= \frac{V\dot{v} - v\dot{V}}{V^2 \sqrt{1 - (v/V)^2}} \end{aligned} \quad (\text{D.6})$$

The translational states for the nonlinear aircraft simulation are  $V$ ,  $\alpha$ , and  $\beta$ . The values of  $u$ ,  $v$ , and  $w$  are computed from Eqs. (D.5), using current values of  $V$ ,  $\alpha$ , and  $\beta$ . Time derivatives  $\dot{u}$ ,  $\dot{v}$ , and  $\dot{w}$  are computed from Eqs. (D.1), and then Eqs. (D.6) are used to obtain  $\dot{V}$ ,  $\dot{\alpha}$ , and  $\dot{\beta}$  for use in the numerical integration routines. This approach avoids the lengthy nonlinear calculations associated with the equations of motion written in wind axes, and allows aircraft nondimensional aerodynamic coefficient data to be implemented in the body-axis coordinate system, with no need for converting the data to wind axes components.

For rotational motion, straightforward algebraic manipulation transforms Eqs. (D.2) into a form suitable for numerical integration, with only one time derivative on the left side of each equation,

$$\begin{aligned} \dot{p} &= (c_1 r + c_2 p - c_4 h_{\text{eng}})q + \bar{q}Sb(c_3 C_l + c_4 C_n) \\ \dot{q} &= (c_5 p + c_7 h_{\text{eng}})r - c_6(p^2 - r^2) + \bar{q}S\bar{c}c_7 C_m \\ \dot{r} &= (c_8 p - c_2 r - c_9 h_{\text{eng}})q + \bar{q}Sb(c_4 C_l + c_9 C_n) \end{aligned} \quad (\text{D.7})$$

## APPENDIX D: F-16 NONLINEAR SIMULATION

467

where

$$\begin{aligned}
 c_1 &= \frac{(I_y - I_z)I_z - I_{xz}^2}{I_x I_z - I_{xz}^2} & c_2 &= \frac{(I_x - I_y + I_z)I_{xz}}{I_x I_z - I_{xz}^2} & c_3 &= \frac{I_z}{I_x I_z - I_{xz}^2} \\
 c_4 &= \frac{I_{xz}}{I_x I_z - I_{xz}^2} & c_5 &= \frac{I_z - I_x}{I_y} & c_6 &= \frac{I_{xz}}{I_y} \\
 c_7 &= \frac{1}{I_y} & c_8 &= \frac{(I_x - I_y)I_x - I_{xz}^2}{I_x I_z - I_{xz}^2} & c_9 &= \frac{I_x}{I_x I_z - I_{xz}^2} \quad (\text{D.8})
 \end{aligned}$$

The nonlinear aircraft simulation also includes rotational kinematic equations and navigation equations. The rotational kinematic equations, which relate Euler angular rates to body-axis angular rates, are given by

$$\begin{aligned}
 \dot{\phi} &= p + \tan \theta (q \sin \phi + r \cos \phi) \\
 \dot{\theta} &= q \cos \phi - r \sin \phi \\
 \dot{\psi} &= \frac{q \sin \phi + r \cos \phi}{\cos \theta} \quad (\text{D.9})
 \end{aligned}$$

Equations (D.9) are nonlinear state equations for the Euler angles  $\phi$ ,  $\theta$ , and  $\psi$ , in a form suitable for numerical integration. The rotational kinematic equations describe the time evolution of the aircraft attitude angles, which are required to properly resolve the gravity force along the aircraft body axes in Eqs. (D.1).

A singularity exists in the rotational kinematic state equations at  $\theta = \pm 90$  deg. This is eliminated in the simulation code by limiting  $\theta$  to the ranges 0 to  $\pm 89.99$  deg and  $\pm 90.01$  to  $\pm 180$  deg. The value of  $|(\cos \theta)^{-1}|$  is thereby limited to a maximum value of approximately 5730, which allows sufficiently accurate calculations of the kinematic state derivatives near the singularity. In the nonlinear simulation,  $\theta$  automatically skips from  $\pm 89.99$  to  $\pm 90.01$  deg and vice versa, depending on the direction of aircraft pitch angle motion near  $\theta = \pm 90$  deg.

The navigation equations relate aircraft translational velocity components in body axes to earth-axis components, neglecting wind. These differential equations describe the time evolution of the position of the aircraft c.g. relative to earth axes,

$$\begin{aligned}
 \dot{x}_E &= u \cos \psi \cos \theta + v(\cos \psi \sin \theta \sin \phi - \sin \psi \cos \phi) \\
 &\quad + w(\cos \psi \sin \theta \cos \phi + \sin \psi \sin \phi) \\
 \dot{y}_E &= u \sin \psi \cos \theta + v(\sin \psi \sin \theta \sin \phi + \cos \psi \cos \phi) \\
 &\quad + w(\sin \psi \sin \theta \cos \phi - \cos \psi \sin \phi) \\
 \dot{h} &= u \sin \theta - v \cos \theta \sin \phi - w \cos \theta \cos \phi \quad (\text{D.10})
 \end{aligned}$$

Assuming thrust acts along the  $x$  body axis, the body-axis accelerations  $a_x$ ,  $a_y$ , and  $a_z$  in  $g$  units are calculated from

$$\begin{aligned} a_x &= \frac{\bar{q}SC_X + T}{mg} \\ a_y &= \frac{\bar{q}SC_Y}{mg} \\ a_z &= \frac{\bar{q}SC_Z}{mg} \end{aligned} \quad (\text{D.11})$$

Equations (D.1) through (D.10) are the nonlinear aircraft equations of motion implemented in the F-16 nonlinear simulation.

### D.3 Engine Model

The F-16 is powered by a single afterburning turbofan jet engine, which was modeled taking into account throttle gearing and engine power level lag.

The engine power dynamic response was modeled with an additional state equation, as a simple first-order lag in the actual power level response to commanded power level:

$$\dot{P}_a = \frac{1}{\tau_{\text{eng}}} (P_c - P_a) \quad (\text{D.12})$$

Commanded power level was computed as a function of throttle position,

$$P_c = P_c(\delta_{th}) \quad (\text{D.13})$$

Throttle gearing in Eq. (D.13) is implemented in the file `tgear.m`, which translates the throttle deflection  $\delta_{th}$  in the interval  $[0,1]$  to commanded power level in the interval  $[0,100]$ . Commanded power  $P_c$  as a function of  $\delta_{th}$  is

$$P_c(\delta_{th}) = \begin{cases} 64.94\delta_{th} & \text{if } \delta_{th} \leq 0.77 \\ 217.38\delta_{th} - 117.38 & \text{if } \delta_{th} > 0.77 \end{cases} \quad (\text{D.14})$$

The routine `rtau.m` computes the engine power time constant  $\tau_{\text{eng}}$ , and `pdot.m` implements the engine thrust dynamics in Eq. (D.12).

Engine thrust force is computed by linear interpolation of an engine thrust database, as a function of actual power level  $P_a$ , altitude  $h$ , and Mach number  $M$ ,

$$T = T(P_a, h, M) \quad (\text{D.15})$$

For the F-16, the engine thrust is given in tabular form as a function of altitude and Mach number for idle, military, and maximum power. The

## APPENDIX D: F-16 NONLINEAR SIMULATION

469

data table for each power level includes values for altitude and Mach number in the ranges  $0 \leq h \leq 50,000$  ft and  $0 \leq M \leq 1$ . Engine data tables are defined in the file `f16_engine_setup.m`. Linear interpolation and thrust calculation are performed in the module `f16_engine.m`. Thrust is computed from

$$T = \begin{cases} T_{\text{idle}} + (T_{\text{mil}} - T_{\text{idle}}) \left( \frac{P_a}{50} \right) & \text{if } P_a < 50 \\ T_{\text{mil}} + (T_{\text{max}} - T_{\text{mil}}) \left( \frac{P_a - 50}{50} \right) & \text{if } P_a \geq 50 \end{cases} \quad (\text{D.16})$$

The engine angular momentum  $h_{\text{eng}}$  is assumed to act along the aircraft  $x$  body axis with a fixed value of 160 slug-ft<sup>2</sup>/s.

#### D.4 Aerodynamic Model

Nondimensional aerodynamic force and moment coefficient data were derived from a low-speed static wind-tunnel test and a dynamic forced oscillation wind-tunnel test, both conducted on a 16% scale model of the F-16. The aerodynamic database applies to the F-16 flown out of ground effect, with landing gear retracted, and no external stores.<sup>1,2</sup> Static aerodynamic data are in tabular form as a function of angle of attack and sideslip angle over the ranges  $-10 \leq \alpha \leq 45$  deg and  $-30 \leq \beta \leq 30$  deg, respectively. Dynamic data are provided in tabular form at zero sideslip angle over the angle of attack range  $-10 \leq \alpha \leq 45$  deg. Dependence of the nondimensional coefficients on  $\dot{\alpha}$  is included in the  $q$  dependencies, due to the manner in which the data were collected in the wind tunnel.

Each nondimensional aerodynamic force and moment coefficient is built up from a set of component functions, where the value of each component function is determined by a table look-up in the wind-tunnel database. The aerodynamic database was simplified slightly by dropping second-order dependencies, e.g., dependence of longitudinal aerodynamic forces on sideslip angle. For the table look-ups, angle of attack, sideslip angle, and control surface deflections are in degrees. Aircraft angular velocities are in radians per second. The value for each component function is found by linear interpolation, using current values of the states and controls. For values of states and controls outside the range of the available data, the interpolation routines extrapolate linearly using the nearest data points. Aerodynamic coefficients are referenced to a center of gravity location at  $0.35\bar{c}$ , so  $x_{\text{c,g,ref}} = 0.35$ . Corrections to the flight center of gravity position are made in the coefficient build-up equations.

The nondimensional aerodynamic force and moment coefficients for the F-16 vary with flow angles ( $\alpha, \beta$ ), aircraft angular velocities ( $p, q, r$ ), and control surface deflections ( $\delta_s, \delta_a, \delta_r$ ). Moment coefficients  $C_m$  and  $C_n$  include a correction for the center of gravity position. The coefficients are computed

as follows:

$$\begin{aligned}
 C_X &= C_{X_0}(\alpha, \delta_s) + \left( \frac{q\bar{c}}{2V} \right) C_{X_q}(\alpha) \\
 C_Y &= -0.02\beta + 0.021 \left( \frac{\delta_a}{20} \right) + 0.086 \left( \frac{\delta_r}{30} \right) + \left( \frac{b}{2V} \right) [C_{Y_p}(\alpha)p + C_{Y_r}(\alpha)r] \\
 C_Z &= C_{Z_0}(\alpha) \left[ 1 - \left( \frac{\beta\pi}{180} \right)^2 \right] - 0.19 \left( \frac{\delta_s}{25} \right) + \left( \frac{q\bar{c}}{2V} \right) C_{Z_q}(\alpha) \\
 C_l &= C_{l_0}(\alpha, \beta) + \Delta C_{l_{\delta_a}}(\alpha, \beta) \left( \frac{\delta_a}{20} \right) + \Delta C_{l_{\delta_r}}(\alpha, \beta) \left( \frac{\delta_r}{30} \right) \\
 &\quad + \left( \frac{b}{2V} \right) [C_{lp}(\alpha)p + C_{lr}(\alpha)r] \\
 C_m &= C_{m_0}(\alpha, \delta_s) + \left( \frac{q\bar{c}}{2V} \right) C_{mq}(\alpha) + (x_{cg_{ref}} - x_{cg}) C_Z \\
 C_n &= C_{n_0}(\alpha, \beta) + \Delta C_{n_{\delta_a}}(\alpha, \beta) \left( \frac{\delta_a}{20} \right) + \Delta C_{n_{\delta_r}}(\alpha, \beta) \left( \frac{\delta_r}{30} \right) \\
 &\quad + \left( \frac{b}{2V} \right) [C_{np}(\alpha)p + C_{nr}(\alpha)r] - \left( \frac{\bar{c}}{b} \right) (x_{cg_{ref}} - x_{cg}) C_Y
 \end{aligned}$$

where the symbols with arguments, e.g.,  $C_{l_0}(\alpha, \beta)$ , represent values obtained from table look-ups.

Data tables for the component functions of the nondimensional coefficients are defined in `f16_aero_setup.m`. The nondimensional aerodynamic force and moment coefficients are computed in `f16_aero.m`, using linear interpolation in the functions `cxo.m`, `czm.m`, `cmo.m`, `clo.m`, `cno.m`, `dlda.m`, `dldr.m`, `dnda.m`, `dndr.m`, and `dampder.m`.

## D.5 Atmosphere Model

Air density and the speed of sound are calculated using relations modeling the *U.S. Standard Atmosphere: 1976* (see Appendix C). Quantities that depend on these atmospheric properties, namely Mach number  $M$ , and dynamic pressure  $\bar{q}$ , are also calculated. The relationships are

$$T^* = 1 - 0.703 \times 10^{-5}h$$

$$\rho = 0.002377(T^*)^{4.14}$$

## APPENDIX D: F-16 NONLINEAR SIMULATION

471

**Table D.2 Mass properties of the simulated F-16**

Parameter	Value
Weight, lbf	20,500
$x_{\text{c.g.}}$	0.25
$I_x$ , slug-ft $^2$	9,496
$I_y$ , slug-ft $^2$	55,814
$I_z$ , slug-ft $^2$	63,100
$I_{xz}$ , slug-ft $^2$	982

$$M = \begin{cases} \frac{V}{\sqrt{1.4(1716.3)390}} & h \geq 35,000 \text{ ft} \\ \frac{V}{\sqrt{1.4(1716.3)(519T^*)}} & h < 35,000 \text{ ft} \end{cases}$$

$$\bar{q} = \frac{1}{2} \rho V^2$$

The function `atm.m` is used to compute these quantities, given altitude  $h$  and airspeed  $V$ .

## D.6 Mass/Inertia Properties

The mass properties of the F-16 are given in the file `f16_massprop.m`. These properties include the weight of the aircraft, longitudinal c.g. position  $x_{\text{c.g.}}$ , and moments of inertia  $I_x$ ,  $I_y$ ,  $I_z$ , and  $I_{xz}$ . This module also calculates the constants  $c_1, c_2, \dots, c_9$  for use in the body-axis moment equations. The longitudinal center of gravity location  $x_{\text{c.g.}}$  is given as a fraction of mean aerodynamic chord, and can be adjusted within this module to account for different longitudinal center of gravity locations. Table D.2 lists the mass properties used in the F-16 nonlinear simulation. The default longitudinal c.g. position is set at a forward location, so that the aircraft can be flown open loop, without stability augmentation feedback control.

## D.7 Analysis Tools

### D.7.1 Aircraft States and Controls

For the F-16 nonlinear simulation, the state vector is

$$\mathbf{x} = [V \ \alpha \ \beta \ p \ q \ r \ \phi \ \theta \ \psi \ x_E \ y_E \ h \ P_a]^T \quad (\text{D.17})$$

and the control vector is

$$\mathbf{u} = [\delta_h \ \delta_s \ \delta_a \ \delta_r]^T \quad (\text{D.18})$$

The set of coupled, nonlinear, first-order ordinary differential equations that compose the simulation model can be represented by the vector differential equation

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \quad (\text{D.19})$$

The output equations can be represented by the vector equation

$$\mathbf{y} = \mathbf{h}(\mathbf{x}, \mathbf{u}) \quad (\text{D.20})$$

The nonlinear equations of motion are implemented in `f16_deq.m`. This routine implements Eq. (D.19), which computes the state vector derivatives, given the states and controls. The function `f16.m` is the main routine for the F-16 nonlinear simulation. This routine initializes the data tables, carries out the numerical integration of Eq. (D.19), and implements Eq. (D.20).

### D.7.2 Trim

A steady flight condition with state derivatives that are constant or zero is called a trimmed flight condition. Finding the states and controls associated with these conditions is useful for initializing the nonlinear simulation and defining a reference condition for linearization. The equations defining a trim condition are Eqs. (D.19) with the state derivatives set to a constant vector  $\mathbf{c}_{\text{trim}}$ ,

$$\mathbf{c}_{\text{trim}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \quad (\text{D.21})$$

This constitutes a set of coupled nonlinear algebraic equations that can be solved to find trim values of the states and controls. Note that the state equations related to the position variables  $x_E$  and  $y_E$  are not included in the equations used for trim, because they are not relevant for dynamics and control analysis. The values of the state and control vectors that satisfy the trim equations and any added constraint equations define the trim condition. Constraint equations for common steady flight conditions are described next.

### D.7.3 Trim Constraint for Steady Translational Flight

For a steady translational flight condition,

$$\phi = p = q = r = 0 \quad (\text{D.22})$$

The earth position states ( $x_E$  and  $y_E$ ) are dropped, since their values matter only for navigation. Altitude  $h$  will be specified for a desired flight condition. Heading angle  $\psi$  is arbitrary, so the remaining states are  $V$ ,  $\alpha$ ,  $\beta$ ,  $\theta$ , and  $P_a$ . For a given aircraft wing loading,  $V$  and  $\alpha$  are related, so only one of these can be chosen for the desired flight condition. Sideslip angle  $\beta$  is used to balance lateral asymmetry and side force from lateral control

## APPENDIX D: F-16 NONLINEAR SIMULATION

473

deflections. Controls  $\delta_s$ ,  $\delta_a$ , and  $\delta_r$  balance aerodynamic moments. The engine power  $P_e$  in steady state is a function of throttle  $\delta_{th}$  via the throttle gearing. Engine power combined with altitude and airspeed determine the thrust, which must balance drag.

When airspeed  $V$  and flight-path angle  $\gamma \equiv \theta - \alpha$  are specified for the steady flight condition, then the kinematic relationship between the velocity expressed in wind axes and in earth axes gives

$$\begin{bmatrix} * \\ * \\ -V \sin \gamma \end{bmatrix}_{\text{earth axes}} = L(\psi)^T L(\theta)^T L(\phi)^T L_{WB}(\alpha, \beta)^T \begin{bmatrix} V \\ 0 \\ 0 \end{bmatrix}_{\text{wind axes}}$$

Equating  $z$  components on both sides,

$$\sin \gamma = a \sin \theta - b \cos \theta$$

where

$$a = \cos \alpha \cos \beta$$

$$b = \sin \phi \sin \beta + \cos \phi \sin \alpha \cos \beta$$

Solving for  $\tan \theta$ ,

$$\tan \theta = \frac{ab + \sin \gamma \sqrt{a^2 - \sin^2 \gamma + b^2}}{a^2 - \sin^2 \gamma} \quad \theta \neq \pm \frac{\pi}{2} \quad (\text{D.23})$$

The preceding expression is the rate of climb constraint, which can be appended to the trim equations when computing a trim solution.

#### D.7.4 Trim Constraint for Steady Turning Flight

A steady turn can be specified by turn rate  $\dot{\psi}$ , or centripetal acceleration  $G$  in  $g$  units since

$$G = \frac{1}{g} \frac{V^2}{R} = \frac{\dot{\psi} V}{g} \quad (\text{D.24})$$

When the turn is coordinated, the aerodynamic forces are balanced and there is no acceleration in the body-axis  $y$  direction. The body-axis  $y$  force equation becomes

$$0 = pw - ru + g \cos \theta \sin \phi \quad (\text{D.25})$$

which means that the gravity force is balanced by inertial terms. In a steady turn,  $\dot{\phi} = \dot{\theta} = 0$ , so Eq. (3.25) reduces to

$$\begin{aligned} p &= -\dot{\psi} \sin \theta \\ q &= \dot{\psi} \cos \theta \sin \phi \\ r &= \dot{\psi} \cos \theta \cos \phi \end{aligned} \quad (\text{D.26})$$

Combining Eqs. (D.24–D.26), and substituting for  $u$ ,  $v$ , and  $w$  in terms of  $V$ ,  $\alpha$ , and  $\beta$  from Eqs. (D.5),

$$\sin \phi = G \cos \beta (\sin \alpha \tan \theta + \cos \alpha \cos \phi) \quad (\text{D.27})$$

Equation (D.27) is the coordinated turn constraint, which can be appended to the trim equations when computing a trim solution.

Combining the rate of climb constraint (D.23) and the coordinated turn constraint (D.27) by substituting for  $\tan \theta$  in the coordinated turn constraint gives

$$\tan \phi = G \frac{\cos \beta}{\cos \alpha} \frac{(a - b^2) + b \tan \alpha \sqrt{c(1 - b^2) + G^2 \sin^2 \beta}}{a^2 - b^2(1 + c \tan^2 \alpha)} \quad (\text{D.28})$$

where

$$\begin{aligned} a &= 1 - G \tan \alpha \sin \beta \\ b &= \frac{\sin \gamma}{\cos \beta} \\ c &= 1 + G^2 \cos^2 \beta \end{aligned} \quad (\text{D.29})$$

For  $\gamma = 0$ , Eq. (D.29) becomes

$$\tan \phi = \frac{G \cos \beta}{\cos \alpha - G \sin \alpha \sin \beta} \quad (\text{D.30})$$

The F-16 nonlinear simulation can include the rate of climb constraint Eq. (D.23) and/or the turn constraint Eq. (D.27) in `f16_trm.m` to calculate a trimmed flight condition.

### D.7.5 Linearization

The nonlinear equations of motion can be linearized using finite differences, as discussed in Chapter 3. The function `Inze.m` does this, using the function `f16_deq.m` to compute values of the right side of Eqs. (D.19) for nominal and perturbed states and controls.

## APPENDIX D: F-16 NONLINEAR SIMULATION

475

**D.7.6 Solving Nonlinear Aircraft Equations of Motion**

The nonlinear equations of motion, with experimental data for the aerodynamics and thrust, and arbitrary control surface input time series, require numerical integration to obtain the state and output time series. This is an initial value problem, which is as follows: given the initial state at  $t = 0$  and the control for  $t \geq 0$ , find the state trajectory for  $t > 0$ . A standard algorithm for continuous systems is fourth-order Runge-Kutta, where the state equations are integrated repeatedly over one time step  $\Delta t$ , as follows:

$$\mathbf{x}(t + \Delta t) = \mathbf{x}(t) + \Delta t \left( \frac{\mathbf{k}_1}{6} + \frac{\mathbf{k}_2}{3} + \frac{\mathbf{k}_3}{3} + \frac{\mathbf{k}_4}{6} \right)$$

where

$\Delta t$  = integration time step

$$\mathbf{k}_1 = f(\mathbf{x}, \mathbf{u}, t)$$

$$\mathbf{k}_2 = f(\mathbf{x}^*, \mathbf{u}, t^*)$$

$$\mathbf{k}_3 = f(\mathbf{x}^{**}, \mathbf{u}, t^{**})$$

$$\mathbf{k}_4 = f(\mathbf{x}^{***}, \mathbf{u}, t^{***})$$

and

$$\mathbf{x}^* = \mathbf{x}(t) + \frac{\Delta t}{2} f(\mathbf{x}, \mathbf{u}, t) \quad t^* = t + \frac{\Delta t}{2}$$

$$\mathbf{x}^{**} = \mathbf{x}(t) + \frac{\Delta t}{2} f(\mathbf{x}^*, \mathbf{u}, t^*) \quad t^{**} = t + \frac{\Delta t}{2}$$

$$\mathbf{x}^{***} = \mathbf{x}(t) + \Delta t f(\mathbf{x}^{**}, \mathbf{u}, t^{**}) \quad t^{***} = t + \Delta t$$

Second-order Runge-Kutta is less accurate (depending on the size of  $\Delta t$ , the control inputs, and the aircraft dynamics), but executes twice as fast because half as many state derivative calculations are required:

$$\mathbf{x}(t + \Delta t) = \mathbf{x}(t) + \Delta t \mathbf{k}_2$$

where

$\Delta t$  = integration time step

$$\mathbf{k}_2 = f(\mathbf{x}^*, \mathbf{u}, t^*)$$

and

$$\mathbf{x}^* = \mathbf{x}(t) + \frac{\Delta t}{2} f(\mathbf{x}, \mathbf{u}, t); \quad t^* = t + \frac{\Delta t}{2}$$

A flow chart for nonlinear simulation is shown in Fig. D.2.

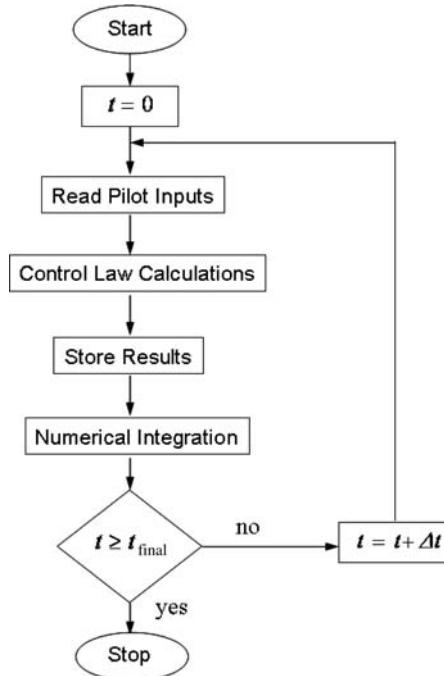


Fig. D.2 Flow chart for nonlinear simulation.

Note that the F-16 nonlinear simulation does not include a feedback control law, so the control law calculations are simply scaling the pilot inputs to commanded control surface deflections.

### D.7.7 Real-Time Piloted Simulation

The F-16 nonlinear simulation can be flown with a joystick or computer mouse in real time, using commercially available software called Aviator Visual Design Simulator (AVDS), available from RasSimTech, Ltd. The interface file `avds_matlab_f16.m` contains comments in the header that include detailed instructions on how to use this capability. AVDS reads pilot inputs and displays a real-time image of the F-16 while the simulation is running in MATLAB®. Full details of this interface are available in Ref. 3.

### References

<sup>1</sup>Stevens, B. L., and Lewis, F. L., (1992) *Aircraft Control and Simulation*, Wiley, New York, 1992.

## APPENDIX D: F-16 NONLINEAR SIMULATION

477

<sup>2</sup>Nguyen, L. T., Ogburn, M. E., Gilbert, W. P., Kibler, K. S., Brown, P. W., and Deal, P. L., "Simulator Study of Stall/Post-Stall Characteristics of a Fighter Airplane with Relaxed Longitudinal Static Stability," NASA TP 1538, 1979.

<sup>3</sup>Garza, F. R., and Morelli, E. A., "A Collection of Nonlinear Aircraft Simulations in MATLAB," NASA TM-2003-212145, 2003.

This page intentionally left blank

# Index

- Aerodynamic model
  - equations
    - mathematical models, 45–47
    - quasi-steady flow, 48
    - unsteady flow, 56–60
  - F-16 nonlinear simulation, 469–470
- Aerodynamics, elementary, 457–459
- Aerodynamics, rigid-body equations of motion, 33–34
- Air, properties of, 455–461
- Aircraft instrumentation errors
  - calibration, 338–339
  - data compatibility, 338–340
  - position errors, 339–340
  - sensor
    - alignment, 339–340
    - dynamics, 340
- Aircraft system identification
  - collinearity diagnostics, 23
  - data compatibility analysis, 21–22
  - experiment design, 20–21
  - model postulation, 20
  - model structure determination, 22
  - model validation, 23–24
  - parameter and state estimation, 22–23
- Aircraft, ordinary least squares
  - application, 120–132
- Air-relative velocity, instrumentation, 297
- Algorithms, simplified, 188–190
- Aliasing, presampling data conditioning, 292–293
- Angular velocity, instrumentation, 298
- Applied forces and moments,
  - rigid-body equations of motion, 36
- Atmosphere model, F-16 nonlinear simulation, 470–471
- Atmosphere, properties of, 455–457
- Backward elimination, stepwise regression, 143
- Bayesian model, parameter estimation, 80–81
- Bayes-like method, nearly singular information matrix, 207–208
- Calculus, 434–437
- Calibration, aircraft instrumentation errors, 338–339
- Closed-loop data, experiment design, 327–328
- Collected equations of motion, mathematical models, 40–42
- Collinearity diagnostics, aircraft system identification, 23
- Complex linear regression, frequency domain methods, 243–250
- Complex numbers, 432–434
- Confidence intervals, ordinary least squares, 103–106
- Continuous-discrete Kalman filter, state estimation, 89–90
- Control surface deflections, instrumentation, 299
- Conversion factors, 461
- Data acquisition system
  - experimental design, 290–297
  - presampling data conditioning, 291–295
  - sampling rate, 290–291
  - sensor range and resolution, 295–297
- Data analysis, 351–381
  - filtering, 351–352
  - finite Fourier transform, 370–376
  - interpolation, 366–367
  - maneuver visualization, 380–381
  - MATLAB software, 415–422
  - numerical differentiation, 367–369

- Data analysis (*continued*)  
power spectrum estimation, 376–380  
signal comparisons, 369–370  
smoothing, 352–366
- Data collinearity  
adverse effects of, 163–164  
assessment of, 159–162  
eigensystem analysis, 160–161  
parameter variance decomposition, 161–162  
regressor correlation matrix, 159–160  
singular value decomposition, 160–161  
detection of, 159–162  
mixed estimator, 164–173  
regression methods, 158–173
- Data compatibility, 333–349  
aircraft instrumentation errors, 338–340  
aircraft system identification, 21–22  
data reconstruction, 336–338  
instrumentation error estimation methods, 344–348  
kinematic equations, 333–336  
MATLAB software, 411–415  
model equations, verification of, 340–344
- Data information content, input design objective, 300–304
- Data partitioning, regression methods, 174–176
- Data reconstruction, data compatibility, 336–338
- Design issues, systems identification, 2–4
- Doubles, single-input design, 312–315
- Dynamic system  
maximum likelihood methods, 182–191  
maximum likelihood parameter estimates, 190–191  
optimization algorithm, 185–187  
process noise, 182–185  
simplified algorithms, 188–190
- Eigensystem analysis, data collinearity assessment of, 160–161
- Elementary aerodynamics, 457–459
- Engine model, F-16 nonlinear simulation, 468–469
- Equation error  
frequency-domain sequential least squares, 275–277  
frequency domain methods, 240–242  
maximum likelihood methods, 216–220
- Equations of motion  
F-16 nonlinear simulation, 463–468  
measured values, substituting of, 69–70  
simplified  
linearization, 60–69  
modeling, 60–71
- Estimation theory, 75–94  
parameter estimation, 79–82  
properties of estimators, 77–79  
state estimation, 83–92
- Estimator properties, estimation theory, 77–79
- Euler attitude angles, instrumentation, 299
- Experimental design, 289–329  
aircraft system identification, 20–21  
data acquisition system, 290–297  
closed-loop data, 327–328  
input design, 299–322  
instrumentation, 297–299  
open-loop unstable aircraft, 327–328  
planning and execution suggestions, 323–327  
data collection, 324–325  
flight-test planning, 323–324  
input design, 324–325  
input optimization, 326–327
- Exponentially weighted least squares, time-varying parameters, 270–271
- Extended Kalman filter, real-time parameter estimation, 282–285
- F-16 nonlinear simulation, 463–476  
aerodynamic model, 469–470  
aircraft background, 463  
analysis tools, 471–476  
atmosphere model, 470–471  
engine model, 468–469  
equations of motion, 463–468  
mass/inertia properties, 471

- Filtering, data analysis, 351–352
- Filters, instrumentation error estimation methods, 344–346
- Finite Fourier transform, data analysis, 370–376
- Fisher model, parameter estimation, 81–82
- Flight test planning, experiment design, planning and execution suggestions, 323–324
- Force equations in wind axes, mathematical models, 38–40
- Forward selection, stepwise regression, 142–143
- Fourier transform, finite, data analysis, 370–376
- Frequency domain
- filter implementation, time domain, 352–355
  - global smoothing, 357–366
  - MATLAB software, 399–403
  - methods, 225–258
    - complex linear regression, 243–250
    - equation-error method, 240–242
    - frequency response, 228–232
    - low-order equivalent system identification (LOES), 250–258
  - maximum likelihood estimator, 232–236
  - measured data transformation, 226–228
  - output-error method, 237–240
- sequential least squares
- equation error, 275–277
  - real-time parameter estimation, 275–282
- recursive Fourier transform, 278
- Frequency response, frequency domain methods, 228–232
- Frequency sweeps, single-input design, 307–312
- Generalized least squares, regression method, 132–137
- Gravity, rigid-body equations of motion, 34
- Greek alphabet, 461
- Impulse, single-input design, 306
- Inertia properties, 459–461
- F-16 nonlinear simulation, 471
- Inertial axes, reference frames, 28
- Input design
- experiment design, 299–322
    - planning and execution suggestions, 324–325
  - maneuver definition, 300
  - MATLAB software, 406–411
  - multiple, 317–322
  - objective, 300–306
    - data information content, 300–304
    - practical constraints, 304–306
  - single-input design, 306–317
- Input optimization, experiment design, planning and execution suggestions, 326–327
- Instrumentation error estimation methods
- data compatibility, 344–348
  - filters and smoothers, 344–346
  - output error, 346–348
- Instrumentation
- air-relative velocity, 297
  - angular velocity, 298
  - control surface deflections, 299
  - Euler attitude angles, 299
  - experiment design, 297–299
  - pilot controls, 299
  - rotational acceleration, 299
  - translational acceleration, 298
- Interpolation, data analysis, 366–367
- Kalman filter
- continuous-discrete, 89–90
  - extended, 282–285
  - state estimation, 86–89
  - time-varying parameters, 271–272
- Kinematic equations, data compatibility, 333–336
- Least-square
- estimator properties, ordinary least squares, 102–103
  - model, parameter estimation, 82
- Levenberg–Marquardt method, nearly singular information matrix, 207
- Linear algebra, 423–432
- Linear model
- mathematical modeling, 10–16
  - quasi-steady flow, 48–52
- Linear regression, MATLAB software, 388–390
- Linear state estimator, state estimation, 83–86

- Linearization, simplified equations of motion, 60–69
- LOES. *See* low-order equivalent system identification
- Low-order equivalent system identification (LOES), frequency domain methods, 250–258
- Maneuver definition, input design, 300
- Maneuver visualization, data analysis, 380–381
- Mass properties, 459–461
- Mass, F-16 nonlinear simulation, 471
- Mathematical background, 423–438
- calculus, 434–437
  - complex numbers, 432–434
  - linear algebra, 423–432
  - polynomial splines, 437–438
- Mathematical models, 27–71
- aerodynamic model equations, 45–47
  - collected equations of motion, 40–42
  - equations of motion simplified, 60–71
  - force equations in wind axes, 38–40
  - linear modeling, 10–16
  - navigation equations, 37–38
  - nonlinear modeling, 16–18
  - output equations, 42–45
  - reference frames, 28
  - rigid-body equations of motion, 31–36
  - rotational kinematic equations, 36–37
  - sign conventions, 29–30
  - system theory elements, 9–17
- MATLAB software, 383–422
- data analysis, 415–422
  - data compatibility, 411–415
  - frequency domain, 399–403
  - input design, 406–411
  - linear regression, 388–390
  - model structure determination, 390–395
  - output-error parameter estimation, 395–399
  - overview, 383–388
  - real-time parameter estimation, 403–406
- Maximum likelihood estimator, frequency domain methods, 232–236
- methods, 181–221
- computational aspects, 195–215
  - computing sensitivities, 197–205
  - nearly singular information matrix, 205–208
- dynamic system, 182–191
- equation-error method, 216–220
- output-error method, 191–195
- parameter estimates, dynamic systems, 190–191
- Measured data, transformation of, frequency domain methods, 226–228
- Mixed estimator, data collinearity, 164–173
- Model equations, data compatibility verification, 340–344
- Model postulation, aircraft system identification, 20
- Model structure determination
- aircraft system identification, 22
  - MATLAB software, 390–395
  - reduced model properties, 139–141
- regression methods, 138–158
- statistical metrics, 145–157
  - stepwise regression, 141–145
  - stopping rules, 145–157
- Model validation, aircraft system identification, 23–24
- Motion equations, rigid body, 31–36
- Multiple input design
- coordination, 318
  - correlation, 318–321
  - effectiveness of, 317–318
  - experiment design, 317–322
  - input comparisons, 322
  - optimal inputs, 321–322
- Multisines, single-input design, 310–312
- Multisteps, single-input design, 312–315
- Multivariate orthogonal functions, ordinary least squares, 115–120
- Navigation equations, mathematical models, 37–38
- Nearly singular information matrix
- Bayes-like method, 207
  - Levenberg–Marquardt method, 207
  - maximum likelihood methods, 205–208
- parameter estimates accuracy, 208–215
- rank deficient method, 206–207
- Nonlinear least squares, regression methods, 137–138

- Nonlinear modeling  
mathematical modeling, 16–18  
quasi-steady flow, 52–56
- Nonlinear state estimator, state estimation, 90–92
- Numerical differentiation, data analysis, 367–369
- Open-loop unstable aircraft, experiment design, 327–328
- Optimization algorithm, dynamic system, 185–187
- Ordinary least squares  
aircraft application, 120–132  
analysis of residuals, 109–113  
confidence intervals, 103–106  
hypothesis testing, 106–109  
least-squares estimator properties, 102–103  
multivariate orthogonal functions, 115–120  
regression methods, 97–132  
standardized regressors, 113–115
- Output equations, mathematical models, 42–45
- Output-error method  
frequency domain methods, 237–240  
instrumentation error estimation, 346–348  
maximum likelihood methods, 191–195
- Output-error parameter estimation, MATLAB software, 395–399
- Parameter and state estimation, aircraft system identification, 22–23
- Parameter estimation  
Bayesian model, 80–81  
estimation theory, 79–82  
Fisher model, 81–82  
least-square model, 82  
nearly singular information matrix, 208–215  
system theory elements, 17–19
- Parameter variance decomposition, data collinearity assessment of, 161–162
- Pilot controls, instrumentation, 299
- Pilot inputs, single-input design, 316–317
- Polynomial splines, 437–438
- Position errors, aircraft instrumentation errors, 339–340
- Power spectrum estimation, data analysis, 376–380
- Presampling data conditioning  
aliasing, 292–293  
data acquisition system, 291–295
- Process noise, dynamic system, 182–185
- Propulsion, rigid-body equations of motion, 35–36
- Quasi-steady flow  
aerodynamic model equations, 48  
linear model, 48–52  
nonlinear model, 52–56
- Random process theory, 451–454
- Random variables, 439–444
- Rank deficient method, nearly singular information matrix, 206–207
- Real-time parameter estimation, 261–286  
extended Kalman filter, 282–285  
frequency-domain sequential least squares, 275–282  
MATLAB software, 403–406  
recursive least squares, 264–270  
time-varying parameters, 270–272
- Recursive Fourier transform, frequency-domain sequential least squares, 278
- Recursive least squares, real-time parameter estimation, 264–270
- Reduced model properties, model structure determination, 139–141
- Reference frames  
inertial axes, 28  
mathematical model, 28–29
- Regression methods, 95–179  
data collinearity, 158–173  
data portioning, 174–176  
generalized least squares, 132–137  
model structure determination, 138–158  
nonlinear least squares, 137–138  
ordinary least squares, 97–132  
summary of, 176–179
- Regressor correlation matrix, data collinearity assessment of, 159–160

- Regularization, real-time parameter estimation, 273–275
- Residuals analysis, ordinary least squares, 109–113
- Rigid-body equations of motion aerodynamics, 33–34 applied forces and moments, 36 gravity, 34 mathematical model, 31–36 propulsion, 35–36
- Rotational acceleration, instrumentation, 299
- Rotational kinematic equations, mathematical models, 36–37
- Sampling rate, data acquisition system, 290–291
- Sensor alignment, aircraft instrumentation errors, 339–340
- Sensor dynamics, aircraft instrumentation errors, 340
- Sensor range and resolution, data acquisition system, 295–297
- Sequential least squares, time-varying parameters, 272–273
- Sign conventions, mathematical model, 29–30
- Signal comparisons, data analysis, 369–370
- Simplified algorithms, dynamic system, 188–190
- Single-input design data information content, 306–317 doublets, 312–315 frequency sweeps, 306–312 impulse, 306 multisines, 310–312 multisteps, 312–315 other designs, 317 pilot input, 316–317
- Singular value decomposition, data collinearity assessment of, 160–161
- Smoothers, instrumentation error estimation methods, 344–346
- Smoothing data analysis, 352–366
- frequency domain, 357–366 local, time domain, 355–357 time domain, frequency-domain filter implementation, 352–355
- Standardized regressors, ordinary least squares, 113–115
- State estimation continuous-discrete Kalman filter, 89–90 estimation theory, 83–92 Kalman filter, 86–89 linear state estimator, 83–86 nonlinear state estimator, 90–92
- Statistical metrics, model structure determination, 145–157
- Statistics, 444–451
- Stepwise regression, 143–145 backward elimination, 143 forward selection, 142–143 model structure determination, 141–145
- Substituting measured values, simplified equations of motion, 69–71
- Systems identification design issues, 2–4 system theory elements, 17–19
- System theory elements, 9–24 aircraft system identification, 9–24 mathematical modeling, 9–17 parameter estimation, 17–19 system identification, 17–19
- Time domain frequency-domain filter implementation, 352–355 local smoothing, 355–357 smoothing, 352–355
- Time-varying parameters exponentially weighted least squares, 270–271 Kalman filter, 271–272 real-time parameter estimation, 270–273 sequential least squares, 272–273
- Translational acceleration, instrumentation, 298
- Unsteady flow, aerodynamic model equations, 56–60