SYSTEM SIMULATION TECHNIQUES WITH MATLAB® AND SIMULINK®

SYSTEM SIMULATION TECHNIQUES WITH MATLAB® AND SIMULINK®

Dingyü Xue
Northeastern University, China

YangQuan Chen
University of California, Merced, USA

WILEY

This edition first published 2014

Registered office
John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, West Sussex, PO19 8SQ, United Kingdom

Library of Congress Cataloging-in-Publication Data

Xue, Dingyü.
System simulation techniques with MATLAB and Simulink / Dingyü Xue, YangQuan Chen.
1 online resource.
Includes bibliographical references and index.
Description based on print version record and CIP data provided by publisher; resource not viewed.
ISBN 978-1-118-69435-0 (Adobe PDF) – ISBN 978-1-118-69437-4 (ePub) – ISBN 978-1-118-64792-9 (cloth)
1. System analysis–Data processing. 2. Computer simulation. 3. MATLAB. 4. SIMULINK. I. Chen, YangQuan, 1966– II. Title.
T57.62
620.00285′53–dc23

2013025348

A catalogue record for this book is available from the British Library.

ISBN: 978-1-118-64792-9

Typeset in 10/12pt Times by Aptara Inc., New Delhi, India

1 2014

Contents

Foreword **xiii**

Preface **xv**

1 Introduction to System Simulation Techniques and Applications **1**
1.1 Overview of System Simulation Techniques 1
1.2 Development of Simulation Software 2
1.2.1 Development of Earlier Mathematics Packages 2
1.2.2 Development of Simulation Software and Languages 4
1.3 Introduction to MATLAB 5
1.3.1 Brief History of the Development of MATLAB 5
1.3.2 Characteristics of MATLAB 6
1.4 Structure of the Book 7
1.4.1 Structure of the Book 7
1.4.2 Code Download and Internet Resources 8
1.4.3 Fonts Used in this Book 8
Exercises 9
References 9

2 Fundamentals of MATLAB Programming **11**
2.1 MATLAB Environment 11
2.1.1 MATLAB Interface 11
2.1.2 MATLAB On-line Help and Documentation 11
2.2 Data Types in MATLAB 13
2.2.1 Constants and Variables 13
2.2.2 Structure of MATLAB Statements 13
2.2.3 Matrix Representation in MATLAB 14
2.2.4 Multi-dimensional Arrays 15
2.3 Matrix Computations in MATLAB 16
2.3.1 Algebraic Computation 16
2.3.2 Logical Operations 19
2.3.3 Comparisons and Relationships 20
2.3.4 Data Type Conversion 20
2.4 Flow Structures 21
2.4.1 Loop Structures 21
2.4.2 Conditional Structures 22
2.4.3 Switches 23
2.4.4 Trial Structure 23

2.5 Programming and Tactics of MATLAB Functions 23
2.5.1 Structures of MATLAB Functions 24
2.5.2 Handling Variable Numbers of Arguments 26
2.5.3 Debugging of MATLAB Functions 26
2.5.4 Pseudo Codes 27
2.6 Two-dimensional Graphics in MATLAB 27
2.6.1 Basic Two-dimensional Graphics 28
2.6.2 Plotting Functions with Other Options 29
2.6.3 Labeling MATLAB Graphics 30
2.6.4 Adding Texts and Other Objects to Plots 30
2.6.5 Other Graphics Functions with Applications 31
2.6.6 Plotting Implicit Functions 32
2.7 Three-dimensional Graphics 33
2.7.1 Three-dimensional Curves 33
2.7.2 Surface Plots 35
2.7.3 Local Processing of Graphics 36
2.8 Graphical User Interface Design in MATLAB 36
2.8.1 Graphical User Interface Tool – Guide 37
2.8.2 Handle Graphics and Properties of Objects 38
2.8.3 Menu System Design 43
2.8.4 Illustrative Examples in GUI Design 43
2.8.5 Toolbar Design 48
2.8.6 Embedding ActiveX Components in GUIs 51
2.9 Accelerating MATLAB Functions 52
2.9.1 Execution Time and Profiles of MATLAB Functions 52
2.9.2 Suggestions for Accelerating MATLAB Functions 53
2.9.3 Mex Interface Design 55
Exercises 60
References 63

3 MATLAB Applications in Scientific Computations 65
3.1 Analytical and Numerical Solutions 66
3.2 Solutions to Linear Algebra Problems 67
3.2.1 Inputting Special Matrices 67
3.2.2 Matrix Analysis and Computation 69
3.2.3 Inverse and Pseudo Inverse of Matrices 72
3.2.4 Similarity Transform and Decomposition of Matrices 74
3.2.5 Eigenvalues and Eigenvectors of Matrices 78
3.2.6 Solution of Matrix Equations 79
3.2.7 Nonlinear Matrix Functions 83
3.3 Solutions of Calculus Problems 85
3.3.1 Analytical Solutions to Calculus Problems 85
3.3.2 Numerical Difference and Differentiation 87
3.3.3 Numerical Integration 89
3.3.4 Numerical Multiple Integration 90
3.4 Solutions of Ordinary Differential Equations 91
3.4.1 Numerical Methods of Ordinary Differential Equations 91
3.4.2 MATLAB Solutions to ODE Problems 92

3.4.3 Conversion of ODE Sets 99
3.4.4 Validation of Numerical ODE Solutions 101
3.4.5 Solutions to Differential Algebraic Equations 102
3.4.6 Solutions to Linear Stochastic Differential Equations 104
3.4.7 Analytical Solutions to ODEs 107
3.4.8 Numerical Laplace Transforms in ODE Solutions 108
3.5 Nonlinear Equation Solutions and Optimization 110
3.5.1 Solutions of Nonlinear Equations 110
3.5.2 Solutions to Nonlinear Equations with Multiple Solutions 113
3.5.3 Unconstrained Optimization 116
3.5.4 Linear Programming 117
3.5.5 Quadratic Programming 118
3.5.6 General Nonlinear Programming 118
3.5.7 Global Search Methods in Optimization Problems 120
3.6 Dynamic Programming and its Applications in Path Planning 120
3.6.1 Matrix Representation of Graphs 120
3.6.2 Optimal Path Planning of Oriented Graphs 121
3.6.3 Optimal Path Planning of Graphs 123
3.7 Data Interpolation and Statistical Analysis 124
3.7.1 Interpolation of One-dimensional Data 124
3.7.2 Interpolation of Two-dimensional Data 126
3.7.3 Least Squares Curve Fitting 129
3.7.4 Data Sorting 129
3.7.5 Fast Fourier Transform 130
3.7.6 Data Analysis and Statistics 131
Exercises 136
References 142

4 Mathematical Modeling and Simulation with Simulink 145
4.1 Brief Description of the Simulink Block Library 146
4.1.1 Signal Sources 147
4.1.2 Continuous Blocks 148
4.1.3 Discrete-time Blocks 150
4.1.4 Lookup Table Blocks 151
4.1.5 User-defined Functions 151
4.1.6 Math Blocks 152
4.1.7 Logic and Bit Operation Blocks 153
4.1.8 Nonlinearity Blocks 153
4.1.9 Output Blocks 154
4.1.10 Signal Related Blocks 155
4.1.11 Ports and Subsystem Blocks 156
4.1.12 Commonly Used Blocks 156
4.1.13 Other Toolboxes and Blocksets 157
4.2 Simulink Modeling 159
4.2.1 Establishing a Model Window 159
4.2.2 Connecting and Simple Manipulation of Blocks 159
4.2.3 Parameter Modification in Blocks 162

4.3 Model Manipulation and Simulation Analysis 164
4.3.1 Model Creation and Fundamental Modeling Skills 164
4.3.2 Model Explorer 165
4.3.3 On-line Help System in Simulink 167
4.3.4 Output and Printing of Simulink Models 168
4.3.5 Simulink Environment Setting 168
4.3.6 Debugging Tools of Simulink Models 171
4.4 Illustrative Examples of Simulink Modeling 172
4.5 Modeling, Simulation and Analysis of Linear Systems 180
4.5.1 Modeling of Linear Systems 180
4.5.2 Analysis Interface for Linear Systems 182
4.6 Simulation of Continuous Nonlinear Stochastic Systems 184
4.6.1 Simulation of Random Signals in Simulink 184
4.6.2 Statistical Analysis of Simulation Results 185
Exercises 188
References 191

5 Commonly Used Blocks and Intermediate-level Modeling Skills 193
5.1 Commonly Used Blocks and Modeling Skills 193
5.1.1 Examples of Vectorized Blocks 193
5.1.2 Signals Labeling in Simulink Models 195
5.1.3 Algebraic Loop and its Elimination in Simulink Models 197
5.1.4 Zero-crossing Detection and Simulation of Simulink Models 201
5.2 Modeling and Simulation of Multivariable Linear Systems 202
5.2.1 Modeling State Space Multivariable Systems 202
5.2.2 Multivariable System Modeling with Control System Toolbox 205
5.3 Nonlinear Components with Lookup Table Blocks 209
5.3.1 Single-valued Nonlinearities 209
5.3.2 Multi-valued Nonlinearities with Memories 211
5.3.3 Multi-dimensional Lookup Table Blocks 215
5.3.4 Code Realization of Static Nonlinearities 216
5.4 Block Diagram Based Solutions of Differential Equations 217
5.4.1 Ordinary Differential Equations 218
5.4.2 Differential Algebraic Equations 219
5.4.3 Delayed Differential Equations 221
5.4.4 Switching Differential Equations 224
5.4.5 Fractional-order Differential Equations 225
5.5 Output Block Library 226
5.5.1 Output Block Group 227
5.5.2 Examples of Output Blocks 229
5.5.3 Model Parameter Display and Model Browser 233
5.5.4 Gauge Display of Signals 234
5.5.5 Digital Signal Processing Outputs 237
5.6 Three-dimensional Animation of Simulation Results 238
5.6.1 Fundamentals of Virtual Reality 238
5.6.2 V-realm Software and World Modeling 239
5.6.3 Browsing Virtual Reality World with MATLAB 242
5.6.4 Virtual Reality World Driven by Simulink Models 243

5.7 Subsystems and Block Masking Techniques 245
5.7.1 Building Subsystems 245
5.7.2 Conditional Subsystems 246
5.7.3 Masking Subsystems 249
5.7.4 Constructing Users' Own Block Library 256
5.7.5 An Illustrative Example: F-14 Aircraft Simulation 257
Exercises 260
References 264

6 Advanced Techniques in Simulink Modeling and Applications 265
6.1 Command-line Modeling in Simulink 265
6.1.1 Simulink Models and File Manipulations 265
6.1.2 Simulink Models and Model Files 266
6.1.3 Drawing Block Diagrams with MATLAB Commands 267
6.2 System Simulation and Linearization 272
6.2.1 Execution of Simulation Process 272
6.2.2 Linearization of Nonlinear Systems 274
6.2.3 Padé Approximation to Pure Time Delays 278
6.3 S-function Programming and Applications 280
6.3.1 Writing S-functions in MATLAB 281
6.3.2 Application Example of S-functions: Simulation of ADRC Systems 284
6.3.3 Level-2 S-function Programming 290
6.3.4 Writing S-functions in C 293
6.3.5 Masking an S-function Block 295
6.4 Examples of Optimization in Simulation: Optimal Controller Design Applications 296
6.4.1 Optimal Criterion Selection for Servo Control Systems 297
6.4.2 Objective Function Creation and Optimal Controller Design 298
6.4.3 Global Optimization Approach 301
Exercises 303
References 306

7 Modeling and Simulation of Engineering Systems 307
7.1 Physical System Modeling with Simscape 308
7.1.1 Limitations of Conventional Modeling Methodology 308
7.1.2 Introduction to Simscape 309
7.1.3 Overview of Simscape Foundation Library 310
7.1.4 Conversions of Two Types of Signals 312
7.1.5 Brief Description of the Simscape Language 315
7.1.6 Modeling and Simulation of Complicated Electrical Network 316
7.2 Description of SimPowerSystems 318
7.3 Modeling and Simulation of Electronic Systems 322
7.3.1 Introduction to the SimElectronics Blockset 323
7.3.2 Modeling of Analogue Electronic Circuits 325
7.3.3 Modeling of Digital Electronic Circuits 328
7.3.4 Modeling of Power Electronics Circuits 332
7.3.5 Embedding Spice Models in Simulink 333
7.4 Simulation of Motors and Electric Drive Systems 336
7.4.1 Simulation of DC Motor Drive Systems 336
7.4.2 Simulation of AC Motor Drive Systems 341

7.5 Modeling and Simulation of Mechanical Systems 346
7.5.1 Simulation of Simple Mechanical Systems 346
7.5.2 Introduction to the SimMechanics Blockset 348
7.5.3 Examples of Mechanical System Simulation 352
7.5.4 Interfacing Simulink with Other CAD Tools 357
Exercises 360
References 362

8 Modeling and Simulation of Non-Engineering Systems 363
8.1 Modeling and Simulation of Pharmacokinetics Systems 363
8.1.1 Introduction to Pharmacokinetics 363
8.1.2 Compartment Modeling of Pharmacokinetics Systems 364
8.1.3 Physiologically based Pharmacokinetic Modeling with Simulink 367
8.1.4 Pharmacodynamic Modeling 374
8.1.5 Nonlinear Generalized Predictive Control of Anesthesia 375
8.2 Video and Image Processing Systems 376
8.2.1 Importing Pictures and Videos into MATLAB 377
8.2.2 Display and Output of Videos and Images 378
8.2.3 Fundamental Blocks for Video and Image Processing 380
8.2.4 Processing of Video and Images through Examples 383
8.2.5 Real-time Processing of Videos and Images 389
8.3 Finite State Machine Simulation and Stateflow Applications 390
8.3.1 Introduction of Finite State Machines 391
8.3.2 Fundamentals of Stateflow 391
8.3.3 Commonly Used Commands in Stateflow 395
8.3.4 Application Examples with Stateflow 396
8.3.5 Describing Flows with Stateflow 408
8.4 Simulation of Discrete Event Systems with SimEvents 408
8.4.1 Concepts of Discrete Event Dynamic Systems 408
8.4.2 Introduction to SimEvents 409
8.4.3 Modeling and Simulation of Queuing Systems 412
Exercises 416
References 417

9 Hardware-in-the-loop Simulation and Real-time Control 419
9.1 Simulink and Real-Time Workshop 419
9.1.1 Introduction to Hardware-in-the-loop Techniques 419
9.1.2 Standalone Code Generation 420
9.1.3 Real-time Simulation and Target Computer Simulation 422
9.1.4 Hardware-in-the-loop Simulation with xPC Target 426
9.2 Introduction to dSPACE and its Blocks 429
9.2.1 Introduction to dSPACE 429
9.2.2 dSPACE Block Library 430
9.3 Introduction to Quanser and its Blocks 430
9.3.1 Introduction to Quanser 430
9.3.2 Quanser Block Library 431
9.3.3 Plants in Quanser Rotary Series 433

9.4 Hardware-in-the-loop Simulation and Real-time Control Examples 433
9.4.1 Mathematical Descriptions of the Plants 433
9.4.2 Quanser Real-time Control Experimentation 436
9.4.3 dSPACE Real-time Control Experimentation 438
9.5 Low Cost Solutions with NIAT 439
9.5.1 Commonly Used Blocks in the NIAT Library 440
9.5.2 Modeling and Simulation of Pendubot Systems 440
9.5.3 Hardware-in-the-loop Simulation Experiment of Pendubot Systems 445
9.6 HIL Solutions with Even Lower Costs 446
9.6.1 Arduino Interface Installation and Settings 446
9.6.2 Applications of Arduino Control 447
9.6.3 The MESABox 449
Exercises 450
References 451

Appendix: Functions and Models 453

Index 459

Foreword

It is a pleasure for me to write a foreword for this book by Dingyü Xue and YangQuan Chen. Dingyü came to the University of Sussex in 1988 to study for his DPhil with me. At the time, computing, relating to control engineering, was starting to move from Fortran to MATLAB, first on terminals connected to a central mainframe computer and then to standalone desktop machines. Digital simulation languages, which had replaced analog computers, were also heading in the same direction. The original version of MATLAB used on the mainframe was written in Fortran, followed by the much faster C version a few years later. One great advantage of MATLAB was that its fundamental data type was the matrix, the concept of which I first came across in the now little known language APL. APL was a very efficient coding language, so much so that a fair comment would be that it required as many lines of commenting as coding for a person to understand a program, and it also required a special keyboard. Other major features of MATLAB were the very good graph plotting facilities and the tools available for providing an excellent graphical user interface for a program. The graphical features provided for programming and for the display of results in Simulink were also a major improvement over the features of existing digital simulation languages.

In the early days of MATLAB, I had several general programs on the mainframe computer which used a question and answer interface and gave the output as a printed plot of points. Dingyü, in doing his research, developed a deep understanding of MATLAB and the capabilities of the GUI, one eventual result of which was the program CtrlLAB which is freely available from the MathWorks library. The genesis of this was a program described in my 1962 doctoral dissertation written in Manchester Autocode, which used paper tape to provide the data input and the values of points as output. Intermediate stages had seen its coding in APL and MATLAB using a question and answer format. Dingyü has therefore used MATLAB and Simulink avidly for the past 25 years, including, I suspect, most of the versions issued over that period. He has spent thousands of hours writing new code and modifying existing routines to be compatible, or to take advantage of new features in the changing versions of MATLAB and Simulink. I have known YangQuan Chen – whom Dingyü first met in Singapore about twenty years ago – for the past ten years. Since they first met, they have cooperated a lot with their complementary research interests being united by their use of MATLAB and Simulink.

This book is therefore written by two people who have had a wealth of first-hand experience of using MATLAB/Simulink in control engineering research and teaching its use to students in China and the USA in both mathematical and control-related courses for over two decades. Also, much of the material has been available in earlier versions of the book in Chinese, where it has been extremely well received, and it is used at many universities. Feedback from these publications has provided suggestions for improvements which have been incorporated here.

The coverage of the book is such that it provides a basic introduction to the use of MATLAB/Simulink before going on to address their usage in many facets of mathematics and engineering. After

covering the general aspects of programming and computation in MATLAB, details of applications in many areas of scientific computation are given, covering areas such as differential equations and optimization. Chapters 3–6 are primarily devoted to Simulink, starting from consideration of the functions of the various blocks and continuing to describe a variety of applications covering topics such as linear and nonlinear system simulations, multivariable systems, vectorized blocks, output blocks, the animation of results, linearization of nonlinear systems, S-functions and optimization in simulations. Chapter 7 discusses the more specific engineering application blocks for electronic systems, electrical drive systems and so on, that are available in Simscape, and in chapter 8 some simulation applications for non-engineering systems, image processing and finite state machines are described which show the wide applicability of modeling and simulation techniques.

I'm sure that this book with its many examples and problems will prove a major asset to you, the reader, in learning the simulation capabilities of MATLAB/Simulink, but as Dingyü and YangQuan would no doubt confirm, the only way to really learn is by the hard work of "doing". So attempt the exercises and also design your own to possibly clarify certain points and gain greater understanding.

Derek P Atherton
Professor Emeritus
University of Sussex, UK
March 2013

Preface

As Confucius has said, "*The mechanic, who wishes to do his work well, must first sharpen his tools*", so MATLAB/Simulink is the right tool to solve problems in the field of systems simulation. It can free the scientist and engineer from tedious, laborious and error-prone work in low-level computer programming, and it is obvious that by the use of MATLAB and Simulink, the efficiencies of researchers can be significantly improved. In communities such as systems simulation and control engineering, MATLAB/Simulink is the de facto international computer language, and the importance of such a tool is being taught in universities worldwide.

Although MATLAB itself was developed and advocated by mathematicians, it was in fact first acknowledged by researchers in the engineering community, and in particular, by the researchers in the field of control engineering. The development of MATLAB and Simulink received a significant amount of innovative contribution from scholars and researchers in the field of control engineering. Already, a significant number of toolboxes and blocksets are oriented to control problems. MATLAB itself has extremely strong capabilities for solving problems in scientific computation and system simulation, with its handy graphical facilities and integrated simulation facilities. It is being used by researchers in more and more engineering and other scientific fields, and it has huge potential and great applications possibilities in related fields.

The authors have been consistently using MATLAB in education and scientific research since 1988, and have had some of their MATLAB packages added to MATLAB Central. A significant amount of first-hand knowledge and experience have been accumulated.

The first author started introducing MATLAB into education more than twenty years ago, and has tried to instruct students in the use such tools. For instance, the book "Computer-aided control systems design — MATLAB languages and applications" published by Tsinghua University Press was regarded as the first of its kind and one of the best in China and has been cited by tens of thousands of journal papers and books. The second author has had more than ten years of experience of scientific research and education in universities in the United States, after his work in industry. He has built up a lot of experience in MATLAB/Simulink based simulation as well as hardware-in-the-loop simulation and real-time design of control systems. Two other books have also been written by the authors and introduced into English world, concentrating on, respectively, the fields of automatic control and scientific computation.

The first edition of this present book was published by Tsinghua University Press in Chinese in 2002, and the second edition was published there in 2011. It has been used as a textbook and reference book by many universities in China. With evolution of MATLAB, Simulink and related products, a lot of new material and innovative work has emerged. It is not possible to cover all the material in one book, so the material here was carefully chosen, and tailored to meet the demands of engineering students and researchers in the relevant disciplines. The current shape of this book was finalized in the course at the Northeastern University, China, and also by offering seminars and

series lectures at Utah State University in the USA, at Baosteel Co. Ltd and at Harbin Institute of Technology in China. Based on the programming and educational experiences of over twenty years, the authors have finally debuted the book to the English-speaking world, and we feel sure that this book will be welcomed by readers worldwide.

The educational work in this book, together with other related educational work, was directed and encouraged by the former supervisors, Professors Xingquan Ren and Xinhe Xu of Northeastern University, China, and Professor Derek P Atherton at Sussex University, UK. It was them who guided the first author into the field of system simulation and, in particular, into the paradise of MATLAB/Simulink programming and education.

A lot of suggestions were received during the preparation of related books, and among them, the authors are in particular grateful for the help given by Professor Hengjun Zhu of Beijing Jiaotong University, the late Professor Jingqing Han of the Institute of System Sciences of Academia Sinica, Professor Xiaohua Zhang of Harbin Institute of Technology, China, and Professor Igor Podlubny of the University of Kosice, Slovakia.

Fruitful discussions with colleagues Drs Feng Pan, Dali Chen, Ying Wei, Jianjiang Cui, Liangyong Wang, Zheng Fang resulted in some new ideas and materials in this book, and Zhuo Li in proofreading an early draft of the book. The Chapter 9.6 was based on contribution of MESA LAB Ph.D. students Brandon Stark, Zhuo Li and Brendan Smith of UC Merced.

We wish to thank editorial staffs of Wiley: Tom Carter, Project Editor; Paul Petralia and **Anne Hunt**. Copyediting by Paul Beverley is particularly appreciated!

This book was supported by the MathWorks Book Program, and MATLAB, Simulink and related products can be acquired from

MathWorks, Inc., 3 Apple Hill Drive, Natick, MA, 01760-2098 USA

Tel:+01-508-647-7000, and FAX: +01-508-647-7101

Email: `info@mathworks.com`, Webpage: `http://www.mathworks.com`

Last but not least, the authors are grateful to their family members for the understanding and support during the years of working. Dingyü Xue would like to thank his wife, Jun Yang, and daughter Yang Xue, and YangQuan Chen would like to thank his wife, Huifang Dou, and his sons Duyun, David and Daniel.

Dingyü Xue, *Northeastern University, China*
YangQuan Chen, *University California, Merced, USA*