

# SYSTEM SIMULATION TECHNIQUES WITH MATLAB<sup>®</sup> AND SIMULINK<sup>®</sup>



UNCORRECTED PROOFS

*UNCORRECTED PROOFS*

# SYSTEM SIMULATION TECHNIQUES WITH MATLAB<sup>®</sup> AND SIMULINK<sup>®</sup>



**Dingyü Xue**

*Northeastern University, Shenyang, China*

**YangQuan Chen**

*University of California, USA*

UNCORRECTED PROOFS

**WILEY**

This edition first published 2013  
© 2013 John Wiley & Sons, Ltd

*Registered office*

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

For details of our global editorial offices, for customer services and for information about how to apply for permission to reuse the copyright material in this book please see our website at [www.wiley.com](http://www.wiley.com).

The right of the author to be identified as the author of this work has been asserted in accordance with the Copyright, Designs and Patents Act 1988.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, except as permitted by the UK Copyright, Designs and Patents Act 1988, without the prior permission of the publisher.

Wiley also publishes its books in a variety of electronic formats. Some content that appears in print may not be available in electronic books.

Designations used by companies to distinguish their products are often claimed as trademarks. All brand names and product names used in this book are trade names, service marks, trademarks or registered trademarks of their respective owners. The publisher is not associated with any product or vendor mentioned in this book.

Limit of Liability/Disclaimer of Warranty: While the publisher and author have used their best efforts in preparing this book, they make no representations or warranties with respect to the accuracy or completeness of the contents of this book and specifically disclaim any implied warranties of merchantability or fitness for a particular purpose. It is sold on the understanding that the publisher is not engaged in rendering professional services and neither the publisher nor the author shall be liable for damages arising herefrom. If professional advice or other expert assistance is required, the services of a competent professional should be sought.

MATLAB<sup>®</sup> is a trademark of The MathWorks, Inc. and is used with permission. The MathWorks does not warrant the accuracy of the text or exercises in this book. This book's use or discussion of MATLAB<sup>®</sup> software or related products does not constitute endorsement or sponsorship by The MathWorks of a particular pedagogical approach or particular use of the MATLAB<sup>®</sup> software.

*Library of Congress Cataloging-in-Publication Data*

[to come, includes ISBN]

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

# Contents

<b>Foreword</b>	<b>xiii</b>
<b>Preface</b>	<b>xv</b>
<b>1 Introduction to System Simulation Techniques and Applications</b>	<b>1</b>
1.1 Overview of System Simulation Techniques	1
1.2 Development of Simulation Software	2
1.2.1 <i>Development of Earlier Mathematics Packages</i>	2
1.2.2 <i>Development of Simulation Software and Languages</i>	4
1.3 Introduction to MATLAB	5
1.3.1 <i>Brief History of the Development of MATLAB</i>	5
1.3.2 <i>Characteristics of MATLAB</i>	6
1.4 Structure of the Book	7
1.4.1 <i>Structure of the Book</i>	7
1.4.2 <i>Code Download and Internet Resources</i>	8
1.4.3 <i>Fonts Used in this Book</i>	8
Exercises	9
References	9
<b>2 Fundamentals of MATLAB Programming</b>	<b>11</b>
2.1 MATLAB Environment	11
2.1.1 <i>MATLAB Interface</i>	11
2.1.2 <i>MATLAB On-line Help and Documentation</i>	11
2.2 Data Types in MATLAB	13
2.2.1 <i>Constants and Variables</i>	13
2.2.2 <i>Structure of MATLAB Statements</i>	13
2.2.3 <i>Matrix Representation in MATLAB</i>	14
2.2.4 <i>Multi-dimensional Arrays</i>	15
2.3 Matrix Computations in MATLAB	16
2.3.1 <i>Algebraic Computation</i>	16
2.3.2 <i>Logical Operations</i>	19
2.3.3 <i>Comparisons and Relationships</i>	20
2.3.4 <i>Data Type Conversion</i>	20
2.4 Flow Structures	21
2.4.1 <i>Loop Structures</i>	21
2.4.2 <i>Conditional Structures</i>	22
2.4.3 <i>Switches</i>	23
2.4.4 <i>Trial Structure</i>	23

2.5	Programming and Tactics of MATLAB Functions	23
2.5.1	<i>Structures of MATLAB Functions</i>	24
2.5.2	<i>Handling Variable Numbers of Arguments</i>	26
2.5.3	<i>Debugging of MATLAB Functions</i>	26
2.5.4	<i>Pseudo Codes</i>	27
2.6	Two-dimensional Graphics in MATLAB	27
2.6.1	<i>Basic Two-dimensional Graphics</i>	28
2.6.2	<i>Plotting Functions with Other Options</i>	29
2.6.3	<i>Labeling MATLAB Graphics</i>	30
2.6.4	<i>Adding Texts and Other Objects to Plots</i>	30
2.6.5	<i>Other Graphics Functions with Applications</i>	31
2.6.6	<i>Plotting Implicit Functions</i>	32
2.7	Three-dimensional Graphics	33
2.7.1	<i>Three-dimensional Curves</i>	33
2.7.2	<i>Surface Plots</i>	34
2.7.3	<i>Local Processing of Graphics</i>	36
2.8	Graphical User Interface Design in MATLAB	36
2.8.1	<i>Graphical User Interface Tool – Guide</i>	37
2.8.2	<i>Handle Graphics and Properties of Objects</i>	38
2.8.3	<i>Menu System Design</i>	43
2.8.4	<i>Illustrative Examples in GUI Design</i>	43
2.8.5	<i>Toolbar Design</i>	48
2.8.6	<i>Embedding ActiveX Components in GUIs</i>	51
2.9	Accelerating MATLAB Functions	52
2.9.1	<i>Execution Time and Profiles of MATLAB Functions</i>	52
2.9.2	<i>Suggestions for Accelerating MATLAB Functions</i>	53
2.9.3	<i>Mex Interface Design</i>	55
	Exercises	60
	References	63
<b>3</b>	<b>MATLAB Applications in Scientific Computations</b>	<b>65</b>
3.1	Analytical and Numerical Solutions	66
3.2	Solutions to Linear Algebra Problems	67
3.2.1	<i>Inputting Special Matrices</i>	67
3.2.2	<i>Matrix Analysis and Computation</i>	69
3.2.3	<i>Inverse and Pseudo Inverse of Matrices</i>	72
3.2.4	<i>Similarity Transform and Decomposition of Matrices</i>	74
3.2.5	<i>Eigenvalues and Eigenvectors of Matrices</i>	78
3.2.6	<i>Solution of Matrix Equations</i>	79
3.2.7	<i>Nonlinear Matrix Functions</i>	83
3.3	Solutions of Calculus Problems	85
3.3.1	<i>Analytical Solutions to Calculus Problems</i>	85
3.3.2	<i>Numerical Difference and Differentiation</i>	87
3.3.3	<i>Numerical Integration</i>	89
3.3.4	<i>Numerical Multiple Integration</i>	90
3.4	Solutions of Ordinary Differential Equations	91
3.4.1	<i>Numerical Methods of Ordinary Differential Equations</i>	91
3.4.2	<i>MATLAB Solutions to ODE Problems</i>	92


3.4.3	<i>Conversion of ODE Sets</i>	99
3.4.4	<i>Validation of Numerical ODE Solutions</i>	101
3.4.5	<i>Solutions to Differential Algebraic Equations</i>	102
3.4.6	<i>Solutions to Linear Stochastic Differential Equations</i>	104
3.4.7	<i>Analytical Solutions to ODEs</i>	107
3.4.8	<i>Numerical Laplace Transforms in ODE Solutions</i>	108
3.5	<b>Nonlinear Equation Solutions and Optimization</b>	110
3.5.1	<i>Solutions of Nonlinear Equations</i>	110
3.5.2	<i>Solutions to Nonlinear Equations with Multiple Solutions</i>	113
3.5.3	<i>Unconstrained Optimization</i>	116
3.5.4	<i>Linear Programming</i>	117
3.5.5	<i>Quadratic Programming</i>	118
3.5.6	<i>General Nonlinear Programming</i>	118
3.5.7	<i>Global Search Methods in Optimization Problems</i>	120
3.6	<b>Dynamic Programming and its Applications in Path Planning</b>	120
3.6.1	<i>Matrix Representation of Graphs</i>	120
3.6.2	<i>Optimal Path Planning of Oriented Graphs</i>	121
3.6.3	<i>Optimal Path Planning of Graphs</i>	123
3.7	<b>Data Interpolation and Statistical Analysis</b>	124
3.7.1	<i>Interpolation of One-dimensional Data</i>	124
3.7.2	<i>Interpolation of Two-dimensional Data</i>	126
3.7.3	<i>Least Squares Curve Fitting</i>	129
3.7.4	<i>Data Sorting</i>	129
3.7.5	<i>Fast Fourier Transform</i>	130
3.7.6	<i>Data Analysis and Statistics</i>	131
	<b>Exercises</b>	136
	<b>References</b>	142
<b>4</b>	<b>Mathematical Modeling and Simulation with Simulink</b>	<b>145</b>
4.1	<b>Brief Description of the Simulink Block Library</b>	146
4.1.1	<i>Signal Sources</i>	147
4.1.2	<i>Continuous Blocks</i>	148
4.1.3	<i>Discrete-time Blocks</i>	150
4.1.4	<i>Lookup Table Blocks</i>	151
4.1.5	<i>User-defined Functions</i>	151
4.1.6	<i>Math Blocks</i>	152
4.1.7	<i>Logic and Bit Operation Blocks</i>	153
4.1.8	<i>Nonlinearity Blocks</i>	153
4.1.9	<i>Output Blocks</i>	154
4.1.10	<i>Signal Related Blocks</i>	155
4.1.11	<i>Ports and Subsystem Blocks</i>	156
4.1.12	<i>Commonly Used Blocks</i>	156
4.1.13	<i>Other Toolboxes and Blocksets</i>	157
4.2	<b>Simulink Modeling</b>	159
4.2.1	<i>Establishing a Model Window</i>	159
4.2.2	<i>Connecting and Simple Manipulation of Blocks</i>	159
4.2.3	<i>Parameter Modification in Blocks</i>	162

4.3	Model Manipulation and Simulation Analysis	164
4.3.1	<i>Model Creation and Fundamental Modeling Skills</i>	164
4.3.2	<i>Model Explorer</i>	165
4.3.3	<i>On-line Help System in Simulink</i>	167
4.3.4	<i>Output and Printing of Simulink Models</i>	168
4.3.5	<i>Simulink Environment Setting</i>	168
4.3.6	<i>Debugging Tools of Simulink Models</i>	171
4.4	Illustrative Examples of Simulink Modeling	172
4.5	Modeling, Simulation and Analysis of Linear Systems	180
4.5.1	<i>Modeling of Linear Systems</i>	180
4.5.2	<i>Analysis Interface for Linear Systems</i>	182
4.6	Simulation of Continuous Nonlinear Stochastic Systems	184
4.6.1	<i>Simulation of Random Signals in Simulink</i>	184
4.6.2	<i>Statistical Analysis of Simulation Results</i>	185
	Exercises	188
	References	191
<b>5</b>	<b>Commonly Used Blocks and Intermediate-level Modeling Skills</b>	<b>193</b>
5.1	Commonly Used Blocks and Modeling Skills	193
5.1.1	<i>Examples of Vectorized Blocks</i>	193
5.1.2	<i>Signals Labeling in Simulink Models</i>	195
5.1.3	<i>Algebraic Loop and its Elimination in Simulink Models</i>	197
5.1.4	<i>Zero-crossing Detection and Simulation of Simulink Models</i>	201
5.2	Modeling and Simulation of Multivariable Linear Systems	202
5.2.1	<i>Modeling State Space Multivariable Systems</i>	202
5.2.2	<i>Multivariable System Modeling with Control System Toolbox</i>	205
5.3	Nonlinear Components with Lookup Table Blocks	209
5.3.1	<i>Single-valued Nonlinearities</i>	209
5.3.2	<i>Multi-valued Nonlinearities with Memories</i>	211
5.3.3	<i>Multi-dimensional Lookup Table Blocks</i>	215
5.3.4	<i>Code Realization of Static Nonlinearities</i>	216
5.4	Block Diagram Based Solutions of Differential Equations	217
5.4.1	<i>Ordinary Differential Equations</i>	218
5.4.2	<i>Differential Algebraic Equations</i>	219
5.4.3	<i>Delayed Differential Equations</i>	221
5.4.4	<i>Switching Differential Equations</i>	224
5.4.5	<i>Fractional-order Differential Equations</i>	225
5.5	Output Block Library	226
5.5.1	<i>Output Block Group</i>	227
5.5.2	<i>Examples of Output Blocks</i>	229
5.5.3	<i>Model Parameter Display and Model Browser</i>	233
5.5.4	<i>Gauge Display of Signals</i>	234
5.5.5	<i>Digital Signal Processing Outputs</i>	237
5.6	Three-dimensional Animation of Simulation Results	238
5.6.1	<i>Fundamentals of Virtual Reality</i>	238
5.6.2	<i>V-realm Software and World Modeling</i>	239
5.6.3	<i>Browsing Virtual Reality World with MATLAB</i>	242
5.6.4	<i>Virtual Reality World Driven by Simulink Models</i>	243



5.7	Subsystems and Block Masking Techniques	245
5.7.1	<i>Building Subsystems</i>	245
5.7.2	<i>Conditional Subsystems</i>	246
5.7.3	<i>Masking Subsystems</i>	249
5.7.4	<i>Constructing Users' Own Block Library</i>	256
5.7.5	<i>An Illustrative Example: F-14 Aircraft Simulation</i>	257
	Exercises	260
	References	264
<b>6</b>	<b>Advanced Techniques in Simulink Modeling and Applications</b>	<b>265</b>
6.1	Command-line Modeling in Simulink	265
6.1.1	<i>Simulink Models and File Manipulations</i>	265
6.1.2	<i>Simulink Models and Model Files</i>	266
6.1.3	<i>Drawing Block Diagrams with MATLAB Commands</i>	267
6.2	System Simulation and Linearization	272
6.2.1	<i>Execution of Simulation Process</i>	272
6.2.2	<i>Linearization of Nonlinear Systems</i>	274
6.2.3	<i>Padé Approximation to Pure Time Delays</i>	278
6.3	S-function Programming and Applications	280
6.3.1	<i>Writing S-functions in MATLAB</i>	281
6.3.2	<i>Application Example of S-functions: Simulation of ADRC Systems</i>	284
6.3.3	<i>Level-2 S-function Programming</i>	290
6.3.4	<i>Writing S-functions in C</i>	293
6.3.5	<i>Masking an S-function Block</i>	295
6.4	Examples of Optimization in Simulation: Optimal Controller Design Applications	296
6.4.1	<i>Optimal Criterion Selection for Servo Control Systems</i>	297
6.4.2	<i>Objective Function Creation and Optimal Controller Design</i>	298
6.4.3	<i>Global Optimization Approach</i>	301
	Exercises	303
	References	306
<b>7</b>	<b>Modeling and Simulation of Engineering Systems</b>	<b>307</b>
7.1	Physical System Modeling with Simscape	308
7.1.1	<i>Limitations of Conventional Modeling Methodology</i>	308
7.1.2	<i>Introduction to Simscape</i>	309
7.1.3	<i>Overview of Simscape Foundation Library</i>	310
7.1.4	<i>Conversions of Two Types of Signals</i>	312
7.1.5	<i>Brief Description of the Simscape Language</i>	315
7.1.6	<i>Modeling and Simulation of Complicated Electrical Network</i>	316
7.2	Description of SimPowerSystems	318
7.3	Modeling and Simulation of Electronic Systems	322
7.3.1	<i>Introduction to the SimElectronics Blockset</i>	323
7.3.2	<i>Modeling of Analogue Electronic Circuits</i>	325
7.3.3	<i>Modeling of Digital Electronic Circuits</i>	328
7.3.4	<i>Modeling of Power Electronics Circuits</i>	332
7.3.5	<i>Embedding Spice Models in Simulink</i>	333
7.4	Simulation of Motors and Electric Drive Systems	336
7.4.1	<i>Simulation of DC Motor Drive Systems</i>	336
7.4.2	<i>Simulation of AC Motor Drive Systems</i>	341

7.5	Modeling and Simulation of Mechanical Systems	346
7.5.1	<i>Simulation of Simple Mechanical Systems</i>	346
7.5.2	<i>Introduction to the SimMechanics Blockset</i>	348
7.5.3	<i>Examples of Mechanical System Simulation</i>	352
7.5.4	<i>Interfacing Simulink with Other CAD Tools</i>	357
	Exercises	360
	References	362
<b>8</b>	<b>Modeling and Simulation of Non-Engineering Systems</b>	<b>363</b>
8.1	Modeling and Simulation of Pharmacokinetics Systems	363
8.1.1	<i>Introduction to Pharmacokinetics</i>	363
8.1.2	<i>Compartment Modeling of Pharmacokinetics Systems</i>	364
8.1.3	<i>Physiologically based Pharmacokinetic Modeling with Simulink</i>	367
8.1.4	<i>Pharmacodynamic Modeling</i>	374
8.1.5	<i>Nonlinear Generalized Predictive Control of Anesthetics Processes</i>	375
8.2	Video and Image Processing Systems	376
8.2.1	<i>Importing Pictures and Videos into MATLAB</i>	377
8.2.2	<i>Display and Output of Videos and Images</i>	378
8.2.3	<i>Fundamental Blocks for Video and Image Processing</i>	380
8.2.4	<i>Processing of Video and Images through Examples</i>	383
8.2.5	<i>Real-time Processing of Videos and Images</i>	389
8.3	Finite State Machine Simulation and Stateflow Applications	390
8.3.1	<i>Introduction of Finite State Machines</i>	391
8.3.2	<i>Fundamentals of Stateflow</i>	391
8.3.3	<i>Commonly Used Commands in Stateflow</i>	395
8.3.4	<i>Application Examples with Stateflow</i>	396
8.3.5	<i>Describing Flows with Stateflow</i>	408
8.4	Simulation of Discrete Event Systems with SimEvents	408
8.4.1	<i>Concepts of Discrete Event Dynamic Systems</i>	408
8.4.2	<i>Introduction to SimEvents</i>	409
8.4.3	<i>Modeling and Simulation of Queuing Systems</i>	412
	Exercises	416
	References	417
<b>9</b>	<b>Hardware-in-the-loop Simulation and Real-time Control</b>	<b>419</b>
9.1	Simulink and Real-Time Workshop	419
9.1.1	<i>Introduction to Hardware-in-the-loop Techniques</i>	419
9.1.2	<i>Standalone Code Generation</i>	420
9.1.3	<i>Real-time Simulation and Target Computer Simulation</i>	422
9.1.4	<i>Hardware-in-the-loop Simulation with xPC Target</i>	426
9.2	Introduction to dSPACE and its Blocks	429
9.2.1	<i>Introduction to dSPACE</i>	429
9.2.2	<i>dSPACE Block Library</i>	430
9.3	Introduction to Quanser and its Blocks	430
9.3.1	<i>Introduction to Quanser</i>	430
9.3.2	<i>Quanser Block Library</i>	431
9.3.3	<i>Plants in Quanser Rotary Series</i>	433

Contents		xi
9.4	Hardware-in-the-loop Simulation and Real-time Control Examples	433
9.4.1	<i>Mathematical Descriptions of the Plants</i>	433
9.4.2	<i>Quanser Real-time Control Experimentation</i>	436
9.4.3	<i>dSPACE Real-time Control Experimentation</i>	438
9.5	Low Cost Solutions with NIAT	439
9.5.1	<i>Commonly Used Blocks in the NIAT Library</i>	440
9.5.2	<i>Modeling and Simulation of Pendubot Systems</i>	440
9.5.3	<i>Hardware-in-the-loop Simulation Experiment of Pendubot Systems</i>	445
9.6	HIL Solutions with Even Lower Costs	446
9.6.1	<i>Arduino Interface Installation and Settings</i>	446
9.6.2	<i>Applications of Arduino Control</i>	447
9.6.3	The <b>MESA Box</b> 	449
	Exercises	450
	References	451
<b>Index</b>		<b>000</b>

UNCORRECTED PROOFS

*UNCORRECTED PROOFS*

# 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 England*  
*March 2013*

UNCORRECTED PROOF

# 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 **revelent** 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 **they** 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 material in this book, and Zhuo Li in proofreading a draft of the book. 

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](mailto: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, Shenyang, China* 

YangQuan Chen, *University California, USA*