Abstracts and Keywords List for each chapter

for the Wiley new book:

[**System Simulation Techniques with MATLAB and Simulink**](http://www.wiley.com/WileyCDA/WileyTitle/productCd-1118647920.html)

[Dingyü Xue](http://www.wiley.com/WileyCDA/Section/id-302475.html?query=Dingy%C3%BC+Xue), [YangQuan Chen](http://www.wiley.com/WileyCDA/Section/id-302475.html?query=YangQuan+Chen)

ISBN: 978-1-118-64792-9

Hardcover

488 pages

**Chapter-01 Introduction to System Simulation Techniques and Applications**

Abstract:

This introductory chapter presents a concise overview of system simulation techniques and developments of simulation software including some historical early simulation softwares and programs. Then, MATLAB history and characteristics are briefly introduced followed by an overview of the book structure and information on code downloading and internet resources, and convention of font uses. Exercises and chapter references cited are included.

Keywords:

System simulation,

simulation software,

MATLAB history,

MATLAB features,

internet resources

**Chapter-02 Fundamentals of MATLAB Programming**

Abstract:

This chapter covers fundamentals of MATLAB programming. With lots of examples, this chapter offers the essence for beginners to get a handle on programming in MALTAB. Starting from introducing various MATLAB windows and on-line help facilities, the fundamentals of MATLAB programming including data types, statements and matrix representation are explained first followed by matrix manipulations, such as algebraic computation, logical and relationship expressions and data conversion. Then, flow charts in MATLAB programming is illustrated, including loop structures, conditional structures, switches and trial structures. MATLAB function programming and pseudo code processing are covered together with two-dimensional and three-dimensional graphics and visualization techniques. MATLAB graphical user interface (GUI) techniques are explained so that the readers will gain new GUI programming skills to design user-friendly interfaces. Finally, programming skills for delivering high speed, high efficiency codes are introduced with special emphasis on commonly used tips, vectorized programming methodology and MEX programming fundamentals for mixed-language programming. Exercises and chapter references cited are included.

Keywords:

Programming,

data types,

program flow control,

functions,

matrix operations,

GUI,

code efficiency,

vectorized programming techniques,

MEX (mixed-language) programming.

**Chapter-03 MATLAB Applications in Scientific Computations**

Abstract:

This chapter covers both numerical computation and analytical problem solutions with MATLAB. Topical parts included in this long chapter are linear algebra, calculus, ordinary differential equation, optimization, and data and signal processing. First, a general discussion about analytical solutions and numerical solutions to mathematical problems is presented with a dedicated argument why using numerical methods. We then start by first explain and illustrate how to solve various linear algebra problems, including the input of special matrices, matrix analysis, similarity transformation, decomposition, eigenvalue problems, algebraic equation solutions and matrix function evaluations. It is demonstrate that the use of MATLAB in the solution of linear algebra problems is very straightforward and reliable. How to solve calculus related problems in MATLAB is presented next which includes numerical solutions to difference, differentiation, integration and multiple integral problems, as well as analytical (symbolic) ways in solving certain classes of calculus problems. Then, it comes to the fundamentals of dynamical system simulation techniques that are mainly based on numerical solutions to ordinary differential equations in MATLAB via numerous examples to illustrate how to solve stiff differential equations, implicit differential equations, stochastic differential equations and differential algebraic equations. Integral transform methods and analytical solutions of differential equations are also dealt with. In particular, the numerical inverse Laplace transform technique is introduced for solving some complicated differential equations. In the numerical solution methods in optimization problems, a universal nonlinear equation solver is presented, for finding with ease the possible multiple solutions to nonlinear equations, together with other approaches. Unconstrained optimization problems are explored, followed by linear programming problems and quadratic programming problems as well as ordinary nonlinear programming problems. Dynamic programming techniques and their use in path planning problem applications are discussed as well in a separate section. Finally, this chapter ends with a section introducing data and signal processing methods, including one- and two dimensional interpolation problems and least squares curve fitting problems, data sorting, pseudo random number generating, fast Fourier series transformation and spectrum analysis. Exercises and chapter references cited are included.

Keywords:

linear algebra,

numerical and symbolic calculus,

ordinary differential equation,

numerical inverse Laplace transform,

optimization,

dynamic programming,

curve fitting,

data interpolation,

random numbers,

signal processing

**Chapter-04 Mathematical Modeling and Simulation with Simulink**

Abstract:

This chapter opens with a historical recall of the evolution of Simulink and a simple explanation of the basic idea of Simulink. This chapter is prepared for those who have little to zero experience with Simulink. First, a brief introduction to various block libraries of Simulink is given, and some of the commonly used blocks are described. Basic manipulations of Simulink blocks such as rotating, connecting and block parameter modification, together with how to build Simulink models, are then introduced and illustrated at the mouse click level. To grow the Simulink modeling skills, some essential tools are introduced, including the use of the model browser, model printing and simulation parameter settings. To put Simulink in actual use, modeling and simulation techniques are demonstrated with some detailed illustrative examples. Equipped with the necessary fundamental knowledge and prepared for more advanced modeling and simulation tasks, the readers are ready to learn linear system modeling and representation methods where LTI Viewer based linear system frequency domain analysis and numerical simulation methods are presented. Finally, simulation methods for continuous systems driven by stochastic inputs are discussed. Statistical analyses of simulation results are given such as probability density function, correlation and power spectral density of the signals in the systems with illustrative examples. Exercises and chapter references cited are included.

Keywords:

Simulink modeling,

Simulink library,

Block-diagram manipulation,

Linear time invariant (LTI),

Frequency domain analysis,

Time domain analysis,

Stochastic signal

Statistical analysis

Correlation analysis

Power spectrum analysis

**Chapter-05 Commonly Used Blocks and Intermediate-level Modeling Skills**

Abstract:

This chapter will take a closer examination of some commonly used Simulink blocks and their uses in Simulink modeling so in the end of this chapter, readers will be at the intermediate-level in Simulink modeling techniques. First covered is a simple example used to further demonstrate the model representation and modeling skills such as including vectorized block modeling and model decoration techniques. Important problems such as the concept of algebraic loops and their elimination, and also the zero-crossing detection method are discussed. Then, Simulink modeling of linear multivariable systems is illustrated, where the LTI block in the Control System Toolbox is recommended for simplifying the modeling process. Commonly used blocks important in Simulink modeling applications such as the lookup table and various switches are explored. General methods in constructing piecewise linear nonlinearities are introduced for both cases: memoryless nonlinearities and nonlinearities with memories. Simulink modeling techniques for various kinds of differential equations are demonstrated in a dedicated section. These include ordinary differential equations, differential algebraic equations, delay differential equations, switching differential equations and even fractional-order (non-integer-order) differential equations. Simulation result visualization is essential in any simulation task so various visualization output blocks in Simulink are presented, such as scope output, workspace variable output and gauges output. More advanced Simulink output visualization methods are presented, including three-dimensional animation methods with virtual reality techniques. Fundamental world modeling with VRML is briefly introduced as well, and the VRML models driven by MATLAB and Simulink

output are discussed. Finally, subsystem modeling is introduced using subsystem masking techniques. An illustrative example of Simulink modeling of a complicated system is presented in detail. Exercises and chapter references cited are included.

Keywords:

Algebraic loop,

LTI blocks,

Nonlinear blocks,

differential algebraic equations,

delay differential equations,

switching differential equations,

fractional-order differential equations,

VRML,

World modeling,

subsystem masking

**Chapter-06 Advanced Techniques in Simulink Modeling and Applications**

Abstract:

Simulink offers powerful direct graphical based programming-free methods to get system simulation tasks completed. In practice, since graphical methods have some limitations, it may be necessary to use command-line based modeling and design methods together with graphical methods. In this chapter, advanced techniques of command-line modeling and application are presented by first introducing how to use MATLAB commands to create Simulink models. By command-line drawing techniques, complicated Simulink models can be created. Then, issues to note during the execution of

Simulink models is introduced are discussed. Linearization techniques of nonlinear systems are also addressed. In particular, the Pad´e approximation to pure time delays is further discussed. It can be seen that not all the models can be constructed with Simulink graphical methods alone. Some of the complicated models can only be created and analyzed using MATLAB commands. Thus, using a dedicated section, advanced techniques are presented for creating complicated models. S-function programming techniques will be presented and illustrated and their use in simulation of automatic disturbance rejection control (ADRC) systems will be demonstrated as a case study. Finally, command-line based optimal controller design technique with Simulink models is introduced, and optimal controller design methods for nonlinear plants are also presented as an advanced Simulink modeling application example. Exercises and chapter references cited are included.

Keywords:

Linearization,

Delay approximation,

S-function programming,

Masking S-function block,

Automatic disturbance rejection control,

Optimal controller design,

Global optimization,

**Chapter-07 Modeling and Simulation of Engineering Systems**

Abstract:

Simulink models can always be constructed since low-level Simulink blocks can be used to model any dynamical system with arbitrary complexity. However, for complex engineering (and non-engineering) system simulation tasks, this chapter explains, promotes and demonstrates the multi-domain physical modeling strategy advocated and implemented in Simulink. Many well-established and specialized blocksets in various disciplines have been developed for use with Simulink. Some of the blocksets have been developed and integrated in the Simscape framework. This chapter dedicated the first section in introducing the concept of multi-domain physical modeling and an introduction to the Simscape blockset. Then, in detail, electrical system modeling with SimPowerSystems and other blocksets is addressed. The rest of the chapter covers the modeling and simulation of electronic systems, motor drive systems and mechanical systems with a lot of examples. Exercises and chapter references cited are included.

Keywords:

multi-domain physical modeling

Simscape,

electrical system modeling,

electronic systems,

Spice circuit model,

motor drive systems,

mechanical systems,

mechanical CAD model

**Chapter- 08 Modeling and Simulation of Non-Engineering Systems**

Abstract:

This chapter serves as a showcase to demonstrate that MATLAB/Simulink can also be used directly in modeling and simulation of many non-engineering systems. There are also a lot of third-party programs and blocksets developed by scholars worldwide. This chapter is only a small showcase chapter. First, modeling and simulation of pharmacokinetics systems are presented. Compartment modeling is briefly introduced, and physiology based pharmacokinetics modeling methods and nonlinear generalized predictive control of anesthetic processes are shown. Then, a dedicated section is included for MATLAB/Simulink based image and video processing. Image Processing Toolbox and Computer Vision System Toolbox blockset are also presented, and real-time video processing systems are explored.

In many non-engineering systems, the finite state machine concept is important. How to use Stateflow to model and simulate complicated supervision problems is presented. Stateflow also generalizes the capabilities of logical or switched systems modeling, and we show that this can be used to describe systems with loops of conditional processes. Finally, this chapter ends with a section on modeling and simulation of discrete event systems and a queuing system is used as an example to demonstrate the use of the SimEvents blockset. Exercises and chapter references cited are included.

Keywords:

pharmacokinetics systems

Compartment modeling

predictive control

image and video processing

finite state machine concept

Stateflow

switched systems

SimEvents blockset

discrete event systems

queuing system

**Chapter-09 Hardware-in-the-loop Simulation and Real-time Control**

Abstract:

Simulation is mostly numerical. However, simulation could be analog. Yet, in this chapter, we will show that the simulation can be partly numerical and partly analog. Including the dynamic plant in the loop of simulation, is referred to as a hardware-in-the-loop simulation. Since this kind of simulation is often performed in real-time, it is sometimes referred to as real-time simulation. This has been made very simple in MATLAB/Simulink due to the Real-TimeWorkshop provided by MathWorks that can translate the Simulink models into C code, and the standalone executable files can also be generated using this tool, so that real-time control can be performed. Third-party software and hardware also provide interfaces to Simulink. This chapter introduces some good examples of these products including dSPACE, with its ControlDesk and Quanser plus WinCon/QuaRT (which can be used to implement hardware-in-the-loop simulation and real-time control experiments). MATLAB and Simulink support many products from well-known hardware manufacturers such as Motorola, Texas Instruments etc., and can directly generate executable code for them from Simulink models. A low-cost NIAT tool can also be used to implement hardware-in-the-loop simulation and experiments. Moreover, a much lower cost platform using Arduino is introduced. In this chapter, these tools are demonstrated with applications in real-time control. Exercises and chapter references cited are included.

Keywords:

Semi-physical simulation

Hardware-in-the-loop simulation

Real-time simulation

Simulink Real-time Workshop (RTW)

Real-time control