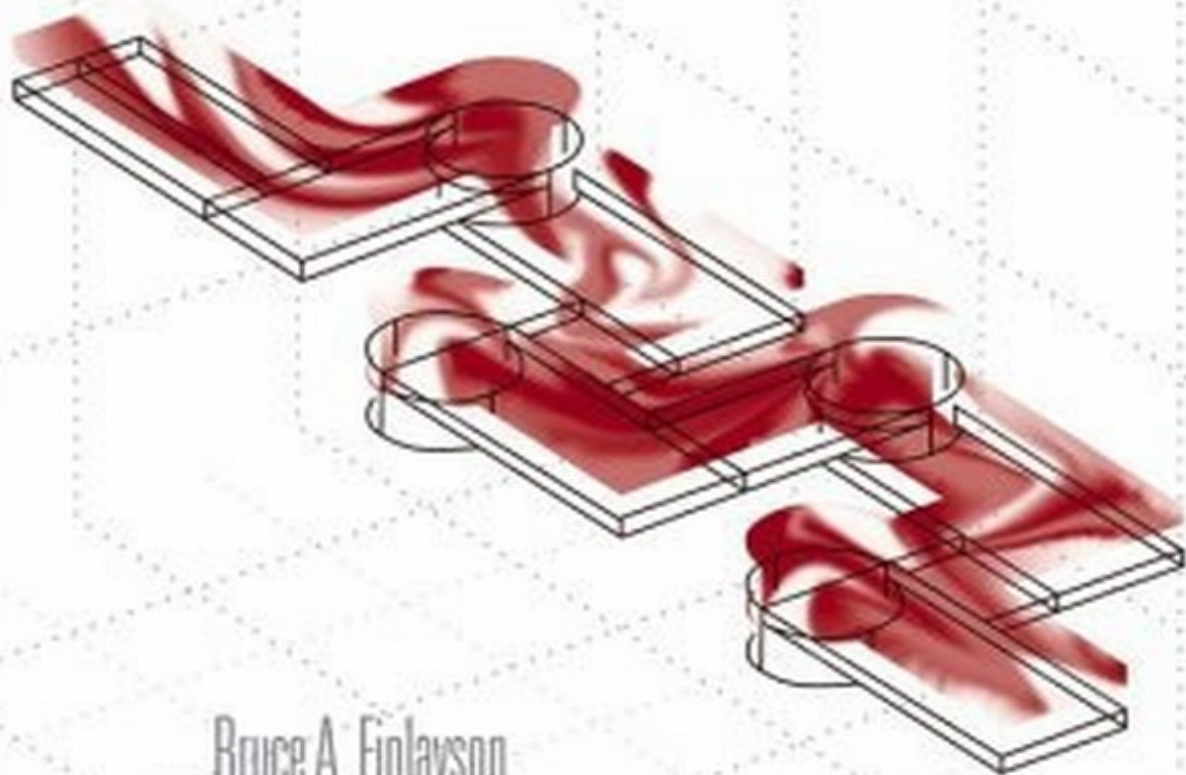


Introduction to Chemical Engineering Computing



Bruce A. Finlayson

INTRODUCTION TO CHEMICAL ENGINEERING COMPUTING

BRUCE A. FINLAYSON, Ph.D.

University of Washington
Seattle, Washington



A JOHN WILEY & SONS, INC., PUBLICATION

Microsoft product screen shots reprinted with permission from Microsoft Corporation. Microsoft Excel is either registered trademark or trademarks of Microsoft Corporation in the United States and/or other countries.

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

FEMLAB[®] is a trademark of COMSOL AB. COMSOL product screen shots reprinted with permission from COMSOL AB.

Copyright © 2006 by John Wiley & Sons, Inc. All rights reserved

Published by John Wiley & Sons, Inc., Hoboken, New Jersey
Published simultaneously in Canada

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, scanning, or otherwise, except as permitted under section 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 750-4470, or on the web at www.copyright.com. Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, (201) 748-6011, fax (201) 748-6008, or online at <http://www.wiley.com/go/permission>.

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. No warranty may be created or extended by sales representatives or written sales materials. The advice and strategies contained herein may not be suitable for your situation. You should consult with a professional where appropriate. Neither the publisher nor author shall be liable for any loss of profit or any other commercial damages, including but not limited to special, incidental, consequential, or other damages.

For general information on our other products and services or for technical support, please contact our Customer Care Department within the United States at (800) 762-2974, outside the United States at (317) 572-3993 or fax (317) 572-4002.

Wiley also publishes its books in a variety of electronic formats. Some content that appears in print may not be available in electronic formats. For more information about Wiley products, visit our web site at www.wiley.com.

Library of Congress Cataloging-in-Publication Data:

Finlayson, Bruce A.

Introduction to chemical engineering computing / Bruce A. Finlayson.

p. cm.

Includes index.

ISBN-13: 978-0-471-74062-9 (cloth)

ISBN-10: 0-471-74062-4 (cloth)

1. Chemical engineering--Data processing. I. Title.

TP184.F56 2005

660'.0285--dc22

2005019351

Printed in the United States of America

10 9 8 7 6 5 4 3 2 1

CONTENTS

Preface	xiii
1 Introduction	1
Organization, 2	
Algebraic Equations, 2	
Process Simulation, 2	
Differential Equations, 3	
Appendices, 4	
2 Equations of State	5
Equations of State – Mathematical Formulation, 6	
Solving Equations of State Using Excel (single equation in one unknown), 8	
Solution Using ‘Goal Seek’, 8	
Solution Using Solver, 9	
Example of a Chemical Engineering Problem Solved Using ‘Goal Seek’, 9	
Solving Equations of State Using MATLAB (single equation in one unknown), 10	
Example of a Chemical Engineering Problem Solved Using MATLAB, 12	
Another Example of a Chemical Engineering Problem Solved Using MATLAB, 13	
Equations of State with Aspen Plus, 16	
Example, 16	
Specific Volume of a Mixture, 18	
Chapter Summary, 22	
Problems, 22	
3 Vapor–Liquid Equilibrium	25
Flash and Phase Separation, 25	

Isothermal Flash – Development of Equations, 26	
Example Using Excel, 28	
Thermodynamic Parameters, 29	
Example Using MATLAB, 30	
Example Using Aspen Plus, 31	
Nonideal Liquids – Test of Thermodynamic Model, 35	
Chapter Summary, 37	
Problems, 37	
4 Chemical Reaction Equilibrium	41
Chemical Equilibrium Expression, 42	
Example of Hydrogen for Fuel Cells, 43	
Solution Using Excel, 44	
Solution Using MATLAB®, 45	
Chemical Equilibria with Two or More Equations, 47	
Multiple Equations, Few Unknowns Using MATLAB, 48	
Method 1 Using the ‘fsolve’ Command, 48	
Method 2 Using the ‘fminsearch’ Function, 49	
Variations in MATLAB, 50	
Chemical Equilibria Using Aspen Plus, 52	
Chapter Summary, 52	
Problems, 52	
5 Mass Balances With Recycle Streams	55
Mathematical Formulation, 56	
Example Without Recycle, 58	
Example With Recycle; Comparison of Sequential and Simultaneous Solution Methods, 60	
Example of Process Simulation Using Excel for Simple Mass Balances, 62	
Example of Process Simulation With Excel Including Chemical Reaction Equilibrium, 63	
Example of Process Simulation With Excel Including Phase Equilibrium, 63	
Did Iterations Converge?, 66	
Extensions, 66	
Chapter Summary, 66	
Class Exercises, 67	
Class Discussion, 67	
Problems, 67	
6 Simulation of Mass Transfer Equipment	73
Thermodynamics, 74	
Example: Multicomponent Distillation With Shortcut Methods, 75	
Mathematical Development, 77	
Multicomponent Distillation With Rigorous Plate-to-Plate Methods, 80	
Example: Packed Bed Absorption, 82	
Example: Gas Plant Production Separation, 85	
Chapter Summary, 87	

Class Exercise, 87
Problems (Using Aspen Plus), 87

7 Process Simulation **89**

Model Library, 90
Example: Ammonia Process, 91
 Utility Costs, 97
Convergence Hints, 100
Optimization, 105
Chapter Summary, 108
Class Exercise, 108
Problems, 108

8 Chemical Reactors **111**

Mathematical Formulation of Reactor Problems, 112
 Example: Plug Flow Reactor and Batch Reactor, 112
 Example: Continuous Stirred Tank Reactor, 114
Using MATLAB to Solve Ordinary Differential Equations, 114
 Simple Example, 114
 Use of the 'Global' Command, 116
 Passing Parameters, 117
 Example: Isothermal Plug Flow Reactor, 118
 Example: Nonisothermal Plug Flow Reactor, 121
Using FEMLAB to Solve Ordinary Differential Equations, 123
 Simple Example, 124
 Example: Isothermal Plug Flow Reactor, 125
 Example: Nonisothermal Plug Flow Reactor, 127
Reactor Problems with Mole Changes and Variable Density, 130
Chemical Reactors with Mass Transfer Limitations, 131
Continuous Stirred Tank Reactors, 134
 Solution Using Excel, 135
 Solution Using MATLAB, 135
 CSTR With Multiple Solutions, 136
 Solutions to Multiple Equations Using MATLAB, 136
Transient Continuous Stirred Tank Reactors, 137
Chapter Summary, 141
Problems, 142

9 Transport Processes in One Dimension **147**

Applications in Chemical Engineering – Mathematical Formulations, 148
 Heat Transfer, 148
 Diffusion and Reaction, 148
 Fluid Flow, 149
 Unsteady Heat Transfer, 151
Example: Heat Transfer in a Slab, 152
Example: Reaction and Diffusion, 154

Parametric Solution, 155
 Example: Flow of a Newtonian Fluid in a Pipe, 156
 Example: Flow of a Non-Newtonian Fluid in a Pipe, 159
 Example: Transient Heat Transfer, 162
 Example: Linear Adsorption, 164
 Example: Chromatography, 167
 Chapter Summary, 169
 Problems, 169

10 Fluid Flow in Two and Three Dimensions 175

Mathematical Foundation of Fluid Flow, 176
 Navier–Stokes Equation, 176
 Non-Newtonian Fluid, 177
 Example: Entry Flow in a Pipe, 179
 Example: Entry Flow of a Non-Newtonian Fluid, 184
 Example: Flow in Microfluidic Devices, 186
 Example: Turbulent Flow in a Pipe, 189
 Example: Start-Up Flow in a Pipe, 191
 Example: Flow Through an Orifice, 193
 Example: Flow in a Serpentine Mixer, 199
 Boundary Conditions, 199
 Nondimensionalization, 201
 Chapter Summary, 203
 Problems, 203

11 Convective Diffusion Equation in Two and Three Dimensions 207

Convective Diffusion Equation, 208
 Nondimensional Equations, 209
 Boundary Conditions, 209
 Example: Heat Transfer in Two Dimensions, 210
 Example: Heat Conduction With a Hole, 213
 Example: Dispersion in Microfluidic Devices, 214
 Effect of Peclet Number, 215
 Example: Concentration-Dependent Viscosity, 217
 Example: Viscous Dissipation, 218
 Example: Chemical Reactor, 221
 Example: Wall Reactions, 221
 Example: Mixing in a Serpentine Mixer, 222
 Chapter Summary, 223
 Problems, 224

Appendix A Hints When Using Excel 229

Cell Organization, 229
 Cell Contents, 229
 Format, 230
 Comments, 230

Pictures, Equations, Web Links, 231
 Select Columns for Charts, Regression, and Printing, 231
 Copy Formulas Across and Down the Spreadsheet, 231
 Insert Rows and Columns, 231
 Split/Freezing, 231
 Iteration on, Tolerance, 232
 Paste, 232
 Plot – *xy* Scatter, Edit, Multiple Curves, Surface Plots, 233
 Arrange Spreadsheet, 235
 Import and Export Text Files, One Column at a Time, 236
 Import and Export Text Files, Multiple Columns, 236
 Export a Text File, 236
 Tools, 237
 Functions, 237
 Matrices, 237
 Excel Help, 238
 Applications of Excel, 238

Appendix B Hints When Using MATLAB[®]

239

General Features, 239
 Start the Program, 239
 Screen Format, 240
 Stop/Closing the Program, 240
 m-Files, 241
 Workspaces and Transfer of Information, 241
 ‘Global’ Command, 241
 Display Tools, 242
 Finding MATLAB Errors, 243
 Debug the Program; That is, Find Your Errors, 243
 Input/Output, 244
 Loops, 245
 Conditional Statements, 245
 Timing Information, 246
 Matrices, 246
 Matrix Multiplication, 246
 Element-by-Element Calculations, 248
 Eigenvalues of a Matrix, 248
 Evaluate an Integral, 248
 Solve Algebraic Equations Using ‘fsolve’, 248
 Solve Algebraic Equations Using ‘fzero’ or ‘fminsearch’ (both in standard MATLAB), 249
 Integrating Ordinary Differential Equations that are Initial Value Problems, 249
 Checklist for Using ‘ode45’ and Other Integration Packages, 251
 Spline Interpolation, 252
 Interpolate Data, Evaluate the Polynomial, and Plot the Result, 252
 Plotting, 253

Plotting Results from Integration of Partial Differential Equations	
Using the Method of Lines, 253	
Simple Plots, 253	
Add Data to an Existing Plot, 254	
Dress Up Your Plot, 254	
Multiple Plots, 255	
Three-Dimensional Plots, 255	
More Complicated Plots, 256	
Use Greek Letters and Symbols in the Text, 257	
MATLAB Help, 257	
Applications of MATLAB, 257	

Appendix C Hints When Using Aspen Plus 259

Quick Tour, 259
Start, 259
Setup, 260
Model Library, 260
Place Units on Flowsheet, 261
Connect the Units with Streams, 262
Data Entry, 262
Specify Components, 262
Specify Properties, 262
Specify the Input Streams, 263
Specify Block Parameters, 264
Run the Problem, 264
Scrutinize the Stream Table, 264
Checking Your Results, 265
Transfer the Flowsheet and Mass and Energy Balance to a
Word Processing Program, 265
Change Conditions, 266
Prepare Your Report, 266
Save Your Results, 267
Getting Help, 268
Applications of Aspen Plus, 268

Appendix D Hints When Using FEMLAB 269

Basic FEMLAB Techniques, 270
Opening Screens, 270
Equations, 270
Draw, 272
Mesh, 272
Subdomain Settings, 273
Boundary Settings, 273
Solve, 275
Postprocessing, 276
Surface Plots, 276
Contour Plots, 278

Cross-Sectional Plots, 278	
Boundary Plots, 278	
Integrals, 279	
Advanced Features, 280	
Model Navigator, 280	
Internal Boundaries, 281	
Expressions, 281	
Coupling Variables and Additional Geometries, 282	
Applications of FEMLAB, 288	

Appendix E Parameter Estimation 293

Mathematical Formulation, 293	
Straight Line, 294	
Straight Line Curve Fit Using Excel, 294	
Plotting the Trendline, 295	
Straight Line Curve Fit Using MATLAB, 295	
Polynomial Regression, 296	
Polynomial Regression Using Excel, 297	
Polynomial Regression Using MATLAB, 298	
Multiple Regression Using Excel, 298	
Nonlinear Regression, 304	
Nonlinear Regression Using Excel, 304	
Nonlinear Regression Using MATLAB, 305	

Appendix F Mathematical Methods 307

Algebraic Equations, 307	
Successive Substitution, 308	
Newton–Raphson, 308	
Ordinary Differential Equations as Initial Value Problems, 310	
Euler’s Method, 310	
Runge–Kutta Methods, 311	
Ordinary Differential Equations as Boundary Value Problems, 312	
Finite Difference Method, 312	
Finite Element Method, 314	
Initial Value Methods, 317	
Finite Difference Method in Excel, 317	
Partial Differential Equations in Time and One Space Dimension, 317	
Partial Differential Equations in Two Space Dimensions, 320	
Finite Difference Method for Elliptic Equations, 321	
Summary, 322	

References 323

Author Index 327

Chemical Index 329

Computer Command Index 331

Subject Index 335

PREFACE

Chemical engineering students and chemical engineers are being asked to solve problems that are increasingly complex, whether the applications are in refineries, fuel cells, micro-reactors, or pharmaceutical plants. Many years ago, students wrote their own programs, first in the FORTRAN programming language, then in languages like MATLAB[®]. With the growth in personal computers, however, software has been written that solves many problems for students, provided they use the programs correctly. Thus, the emphasis has shifted from a small group of people who were interested in writing their own programs to a large group of students who will use the programs, but do not write them. In my 38 years of teaching at the University of Washington, I taught those small groups of students how to use numerical analysis to solve complicated problems. Now, I teach *all* my students how to use the computer wisely. Only a few of the students I teach are interested in the numerical analysis (to my sorrow!), but all the students know they must be able to solve difficult problems, and they need to use the computer to do that.

The goals of this book are to illustrate (a) the problems chemical engineers have to solve, (b) the type of computer programs used to solve them, and (c) how engineers check to be sure they have solved the problems correctly. This is done in the context of how contemporary students learn – minimal reading, just-in-time learning, with lots of computer usage. The programs demonstrated here are Excel[®], MATLAB[®], Aspen Plus[®], and FEMLAB[®].

When writing this book, I assumed that readers are not absolute beginners. Junior and senior chemical engineering students have had experience with spreadsheet programs like Excel, and they can easily learn on the computer when provided a direction and key ideas or phrases. In fact, many students are more computer-savvy than their instructors. However, a beginning chemical engineering student may not know the application very well and may not have gained a solid understanding of the physical phenomena behind an engineering problem. Thus, it is important to give some explanation of why students need to solve certain problems. I have drawn on my experience to give insights into the problems in this book.

My teaching philosophy is that the problems engineers are solving today are usually intractable with analytical methods, but they can be solved with the sophisticated software available today. Thus, every engineer will be solving a problem that no one knows the answer to, and it is the engineer's job to ensure that the problem is posed correctly on paper and in the computer, and it is correctly solved. Engineering students must know how to determine if the computer solved the problem correctly by validating the work done by the computer. If they can do this, they can convince their instructor – or their future boss – that they have a solution that is every bit as reliable as an analytical solution, although without the analytical form and for a problem that cannot be solved analytically.

HOW TO USE THIS BOOK IN TEACHING

This book grew out of a course I developed at the University of Washington, first in the winter quarter, 2003. Student evaluations of the department indicated that students wanted more help when using the computer to solve chemical engineering assignments. Although the students took a programming course in Computer Science, they did not feel it was relevant to their engineering studies. I proposed an elective course for juniors that would introduce them to computer programs they would use in their education. It is called *Chemical Engineering Computer Skills* and is a lecture/laboratory course. Enrollment has grown each year, and in 2005, 70 percent of the junior class enrolled in this course.

As currently taught, I spend one lecture describing a problem and illustrating its solution using the computer programs. Then the class adjourns to a computer classroom where the students work in pairs, with student helpers, solving the same type of problem as just demonstrated in class. Finally, the students work individually on a more difficult problem, using the same techniques, for homework credit. All the homework problems have to be correct; if not, an opportunity is given to redo them. The course is taught credit/no-credit, and credit is given provided 80 percent of the assignments are completed correctly. There are only 10 lectures 50 min long and 10 laboratory sessions in the 10-week quarter. Since the applications cover much of the chemical engineering field, I joke with the students, saying, 'I'm teaching you the entire field in 20 hours.'

This book can also be used in a longer course. Once students have solved the elementary problems, it is easy to complicate the problems with lessons and variations that instructors would like to emphasize. Examples of such problems are provided at the end of each chapter; both introductory and advanced problems are provided. Another way to use the book is to use each chapter within different courses. Once chemical reaction equilibrium has been discussed in the Thermodynamics class, for example, instructors can hold a laboratory session that teaches computer applications, using the chapter on chemical reaction equilibrium. Other chapters would be used in other courses. In this way, the students would use the book during their entire education, in course after course. The hope is, of course, that students would then be able to concentrate more on the chemical engineering principles and use the computer as a tool.

There are four programs that are featured in this book. It is possible that your school does not use all four. While the screen images may be different, the ideas and procedures are the same. Certainly the problems can be solved using other programs. In a working environment, engineers use what their company provides. Thus, engineers may use a less powerful program because it is available. The more powerful program may cost

more, too. Thus, in several chapters, the same problem is solved using different programs, which lets students see first-hand that the more general purpose programs require significantly more programming in order to solve complicated problems. In my experience, when given a suite of programs, students will use the one that allows them to solve their problem fastest.

ACKNOWLEDGMENTS

In writing this book, I owe a great thanks to the students in my classes. The first year there was no written material; students said they wanted it. The second year written material was provided, but it was clear that newer programs like FEMLAB should be emphasized. Many times, a student's question identified something that I did not know about the program either, so all those graduates of the University of Washington (classes of 2004, 2005, and 2006) deserve my thanks. Special thanks go to Barney Santiago, who taught me one of the tricks in the book, and Franklin Lobb, an alumnus working for AspenTech who gave valuable suggestions about Aspen Plus. I also thank Jennifer Foley, a graduate student in bioengineering who learned FEMLAB from me, because she taught me in return how to use coupling variables in FEMLAB. The department provided a challenge grant to write textbooks, funded by a gift in the memory of alumnus Maurice Richford, BS 1926. Without that challenge grant, this book would not have been written. My daughter, Christine Finlayson, improved my writing greatly by serving as a copy-editor, and the clarity is due to her work; any confusion left is my responsibility. The folks at Comsol, the makers of FEMLAB, have been very helpful as FEMLAB has been developed and grown over the past few years; Johan Sundqvist and David Kan were my major contacts. It has been a pleasure working with the folks at Wiley, and they have enhanced the appearance and readability of the book. Most of all, I thank my wife, Pat, for putting up with the long hours of work that such a project requires. She has always supported me and made sacrifices that enabled me to finish.

BRUCE FINLAYSON

Seattle, May 2005

1

INTRODUCTION

Computers have revolutionized the way chemical engineers design and analyze processes, whether designing large units to make polyethylene or small microreactors to detect biological agents. In fact, the engineering problems that many of you will study as undergraduates today are similar in complexity to the problems Ph.D. students solved 30 or 40 years ago. Computer programs can now solve difficult problems in a fraction of the time it used to take. Nowadays, you no longer have to write your own software programs to use computers effectively. Computer programs can do the numerical calculations for you, but you will still need to understand how to apply these programs to specific engineering challenges.

The goal of this book is to help you practice better chemical engineering. Computers are valuable tools that enable progressive, far-reaching chemical engineering. Unfortunately, computers are not as basic as CD players, where you insert a CD, push a button, and get the same result every time. Sometimes computer programs do not work properly for the parameters you have given them. Therefore, you must be careful to use them wisely.

This book will also:

- (1) Illustrate the problems that you as chemical engineers may need to solve;
- (2) Compare the types of computer programs you can use and illustrate which ones are best for certain applications;
- (3) Describe how to check your work to ensure you have solved the problems correctly.

This book demonstrates four computer programs: Excel[®], MATLAB[®], Aspen Plus[®], and FEMLAB[®]. You may have access to other programs created by other companies. While the exact details will not be the same, the steps you take will be similar.

Computer skills are invaluable, but as an engineer, you also need to understand the physical phenomena. Each chemical engineering application chapter starts with a description of the physical problem in general terms. Then those general terms are put into a mathematical context so the computer can represent them. Next, the chapter gives several examples in which such problems are solved, providing step-by-step instructions so you can follow along on your own computer. Sometimes the same problem is solved using different programs so you can see the advantages of each program. Finally, the chapters give more complicated problems your instructor may use as homework.

Examples throughout this book demonstrate how to check your work and how to learn from the answers the computer gives you. When using computers, it is always important to know if the computer has obtained the correct answer. If you follow this strategy you will have no trouble convincing your instructor, or your boss, that you have a solution every bit as reliable as an analytical solution for a problem that cannot be solved analytically:

- (1) Solve the problem
- (2) Validate your work
- (3) Understand how you reached that answer

ORGANIZATION

The book is organized into 11 chapters followed by six appendices, as listed in Table 1.1. Each chapter treats a type of chemical engineering phenomenon, such as process simulation or convective diffusion. The six appendices give additional details about each computer program.

As a modern chemical engineering student, many of you are computer-savvy. This book assumes that you are not a complete beginner, but have some experience with spreadsheet programs such as Excel. The chapters provide examples and step-by-step instructions for using the computer programs to solve chemical engineering problems. If necessary, you can find more detailed information about the individual programs in the Appendices.

Algebraic Equations

Chapters 2–5 deal with chemical engineering problems that are expressed as algebraic equations – usually sets of nonlinear equations, perhaps thousands of them to be solved together. In Chapter 2 you can study equations of state that are more complicated than the perfect gas law. This is especially important because the equation of state provides the thermodynamic basis for not only volume, but also fugacity (phase equilibrium) and enthalpy (departure from ideal gas enthalpy). Chapter 3 covers vapor–liquid equilibrium, and Chapter 4 covers chemical reaction equilibrium. All these topics are combined in simple process simulation in Chapter 5. This means that you must solve many equations together. These four chapters make extensive use of programming languages in Excel and MATLAB.

Process Simulation

Chapter 6 introduces mass transfer problems such as distillation and absorption. Chapter 7 gives a more detailed look at process simulation, where the power of process simulators

TABLE 1.1. Computer Programs Used in Different Chapters

Chapters	Excel	MATLAB	Aspen Plus	FEMLAB
1: Introduction				
2: Equations of State				
3: Vapor–Liquid Equilibrium				
4: Chemical Reaction Equilibrium				
5: Mass Balances with Recycle Streams				
6: Simulation of Mass Transfer Equipment				
7: Process Simulation				
8: Chemical Reactors				
9: Transport Processes in One Dimension				
10: Fluid Flow in Two and Three Dimensions				
11: Convective Diffusion Equation in Two and Three Dimensions				
Appendix A: Hints when Using Excel				
Appendix B: Hints when Using MATLAB®				
Appendix C: Hints when Using Aspen Plus				
Appendix D: Hints when Using FEMLAB				
Appendix E: Parameter Estimation				
Appendix F: Mathematical Methods				

like Aspen Plus really is evident. These chapters make use of commercial codes that are run by inserting data into their custom-designed interface.

Differential Equations

Chapters 8–11 treat problems that are governed by differential equations. Chapter 8 provides methods to model chemical reactors. These are usually initial value problems, which are illustrated in Eq. (1.1).

$$u \frac{dc}{dz} = -kc^2, \quad c(z=0) = c_0 \quad (1.1)$$

Note that the dependent variable, c , is a function of only one independent variable, z , and that the initial value is specified. For reactors, you start at the inlet and integrate down the reactor using either MATLAB or FEMLAB.

Chapter 9 then solves transport problems in one space dimension (1D) using FEMLAB. If you consider heat transfer through a slab, one side of the slab is kept at one temperature, T_0 , and the other side of the slab is maintained at another temperature, T_L . The governing equation is

$$k \frac{d^2 T}{dx^2} = 0 \quad (1.2)$$

with boundary conditions

$$T(0) = T_0, \quad T(L) = T_L \quad (1.3)$$

The differential equation, (1.2), is an ordinary differential equation because there is only one independent variable, x . In this case, equations in one space dimension are boundary value problems, because the conditions are provided at two different locations. While it is also possible to solve this problem using Excel and MATLAB, it is much simpler to use FEMLAB. Transient heat transfer in one space dimension is governed by

$$\rho C_p \frac{\partial T}{\partial t} = k \frac{\partial^2 T}{\partial x^2} \quad (1.4)$$

and this problem is solved using FEMLAB, too.

Chapters 10 and 11 use FEMLAB to solve fluid flow, heat transfer, and mass transfer problems in 2D and 3D. Here again the power of the software program shows through. You get to solve real problems that go beyond the simple 1D cases in your textbook. Those 1D problems are good for learning the subject, but in real-life situations, complications often arise that can only be handled numerically. These problems are partial differential equations, because there are two or more independent variables (say x and y). For example, the Navier–Stokes equations in Cartesian geometry and two dimensions are

$$\begin{aligned} \rho \left(\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} \right) &= -\frac{\partial p}{\partial x} + \mu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \\ \rho \left(\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} \right) &= -\frac{\partial p}{\partial y} + \mu \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) \\ \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} &= 0 \end{aligned} \quad (1.5)$$

Appendices

If you need more background information while solving the problems in the book, consult the appendices. Appendices A–D discuss hints, examples, and step-by-step instructions for the four computer programs demonstrated in this book. For example, Appendix A provides useful options and terminology within Excel, while Appendix B does this for MATLAB. Appendix C provides screen images from Aspen Plus, with explanations, and Appendix D does the same for FEMLAB. Appendix E demonstrates how to use Excel or MATLAB for parameter estimation, and Appendix F illustrates the mathematical methods built into each computer program. While you will not need to program the methods, you may be curious about the mathematical analysis behind the programs.

Whether you tackle one chemical engineering problem or work chapter by chapter through the book, try to enjoy yourself. You and a classmate can sit down and work together – possibly on adjacent computers – to share insights and answer each other’s questions. Remember, too: go back and forth from the application chapters to the computer program appendices; build up your knowledge bit by bit. Your reward is to be a better-trained engineer, able to compete in a fast-paced global environment.

2

EQUATIONS OF STATE

Solving equations of state allows us to find the specific volume of a gaseous mixture of chemicals at a specified temperature and pressure. Without using equations of state, it would be virtually impossible to design a chemical plant. By knowing this specific volume, you can determine the size – and thus cost – of the plant, including the diameter of pipes, the horsepower of compressors and pumps, and the diameter of distillation towers and chemical reactors. Imagine how challenging it would be to design a plant without knowing this important information!

Determining the specific volume is also the first step in calculating the enthalpy and vapor–liquid properties of mixtures. Calculating this enthalpy is especially important when making energy balances to reduce energy use and help the environment.

To solve equations of state, you must solve algebraic equations as described in this chapter. Later chapters cover other topics governed by algebraic equations, such as phase equilibrium, chemical reaction equilibrium, and processes with recycle streams. This chapter introduces the ideal gas equation of state, then describes how computer programs such as Excel[®], MATLAB[®], and Aspen Plus[®] use modified equations of state to easily and accurately solve problems involving gaseous mixtures.

Step-by-step instructions will guide you in using each of these computer programs to determine the specific volume of gaseous mixtures. At the end of the chapter, there are problems to enable you to practice your own calculations. The lessons learned in this chapter carry forward to other applications involving algebraic equations in Chapters 3–6 and 8. After completing this chapter, not only will you be able to solve algebraic equations, but also size equipment in a chemical plant, certainly those pieces of equipment containing gases.

EQUATIONS OF STATE – MATHEMATICAL FORMULATION

The ideal gas equation of state, which relates the pressure, temperature, and specific volume, is a familiar equation:

$$pV = nRT \quad \text{or} \quad p\hat{v} = RT \quad \text{where} \quad \hat{v} = \frac{V}{n} \quad (2.1)$$

The term p is the absolute pressure, V is the volume, n is the number of moles, R is the gas constant, and T is the absolute temperature. The units of R have to be appropriate for the units chosen for the other variables. This equation is quite adequate when the pressure is low (such as one atmosphere). However, many chemical processes take place at very high pressure. For example, ammonia is made at pressures of 220 atmospheres or more. Under these conditions, the ideal gas equation of state may not be a valid representation of reality.

Other equations of states have been developed, usually in conjunction with process simulators, to address chemical processes at high pressure. There are two key features: (1) the equation can represent the real p – V – T behavior; and (2) the parameters must be easily found, including for mixtures. This last criterion is no small requirement. There are more than 25 million chemicals, leading to an infinite number of different mixtures. Obviously, you cannot look up the properties of all those mixtures on the Web.

The first generalization of the ideal gas law was the van der Waals equation of state:

$$p = \frac{RT}{\hat{v} - b} - \frac{a}{\hat{v}^2} \quad (2.2)$$

In this equation, the b accounts for the excluded volume (a second molecule cannot use the same space already used by the first molecule), and the a accounts for the interaction force between two molecules. This extension is just a first step, however, because it will not be a good approximation at extremely high pressures.

The Redlich–Kwong equation of state is a modification of van der Waal’s equation of state:

$$p = \frac{RT}{\hat{v} - b} - \frac{a}{\hat{v}(\hat{v} + b)} \quad (2.3)$$

where

$$a = 0.42748 \left(\frac{R^2 T_c^2}{p_c} \right) \alpha, \quad b = 0.08664 \left(\frac{RT_c}{p_c} \right), \quad \text{and} \quad T_r = \frac{T}{T_c}, \quad \alpha = \frac{1}{T_r^{0.5}} \quad (2.4)$$

In these equations, T_c is the critical temperature (in absolute terms), p_c is the critical pressure, and T_r is the ‘reduced’ temperature (the absolute temperature divided by the critical temperature). The α is particular to the Redlich–Kwong equation of state.

The Redlich–Kwong equation of state was modified further by Soave to give the Redlich–Kwong–Soave equation of state (called RK–Soave in Aspen Plus), which is a common one in process simulators:

$$p = \frac{RT}{\hat{v} - b} - \frac{a}{\hat{v}(\hat{v} + b)} \quad (2.5)$$

Now the parameter α is given by a different formula,

$$\alpha = [1 + m(1 - T_r^{0.5})]^2, \quad m = 0.480 + 1.574\omega + 0.176\omega^2 \quad (2.6)$$

The ω is the ‘acentric’ factor, which is a tabulated quantity for many substances. Thus, the value of α can be computed for each chemical and reduced temperature.

The Peng–Robinson equation is another variation:

$$p = \frac{RT}{\widehat{v} - b} - \frac{a}{\widehat{v}(\widehat{v} + b) + b(\widehat{v} - b)} \quad (2.7)$$

All these equations can be rearranged into a cubic function of specific volume. The form of the Redlich–Kwong and Redlich–Kwong–Soave equation of state is

$$\widehat{v}^3(p) - \widehat{v}^2(RT) + \widehat{v}(a - pb^2 - RTb) - ab = 0 \quad (2.8)$$

When given the temperature and pressure of a gaseous mixture, and the parameters a and b , then to find the specific volume you would have to solve the cubic equation of state for specific volume, $\widehat{v}$. This represents one algebraic equation in one unknown, the specific volume.

For a pure component, the parameters a and b are determined from the critical temperature and critical pressure, and possibly the acentric factor. These are all tabulated quantities, and there are even correlations for them in terms of vapor pressure and normal boiling point, for example. For mixtures it is necessary to combine the values of a and b for each component according to the composition of the gaseous mixture. Common mixing rules are shown in Eqs. (2.9) and (2.10), in which the y_i s are the mole fraction of each chemical in the vapor phase:

$$a_i = 0.42748 \left(\frac{R^2 T_{ci}^2}{p_{ci}} \right) \alpha_i, \quad a = \sum_{i,j=1}^{NCOMP} y_i y_j (a_i a_j)^{0.5}$$

or

$$a = \left(\sum_{i=1}^{NCOMP} y_i a_i^{0.5} \right) \left(\sum_{j=1}^{NCOMP} y_j a_j^{0.5} \right), \quad a = \left(\sum_{i=1}^{NCOMP} y_i a_i^{0.5} \right)^2 \quad (2.9)$$

$$b_i = 0.08664 \left(\frac{RT_{ci}}{p_{ci}} \right), \quad b = \sum_{i=1}^{NCOMP} y_i b_i \quad (2.10)$$

where

$$\alpha_i = \frac{1}{T_{ri}^{0.5}} \quad (\text{for Redlich–Kwong})$$

or

$$\alpha_i = [1 + m_i(1 - T_{ri}^{0.5})]^2 \quad (\text{for Redlich-Kwong-Soave}) \quad (2.11)$$

Thus, the only difference between the problem for a pure component and that for a mixture is in the evaluation of the parameters a and b .

Here is the mathematical problem you must solve: Given a set of chemicals, temperature and pressure, find the specific volume of the mixture. To do this, you must find the critical temperature and pressure of each chemical. Once you have the parameters, you must solve the cubic equation, Eq. (2.8), which is a nonlinear equation in one variable. Because it is a cubic equation, it is possible to find the solution in a series of analytical steps (Perry and Green, 1997, p. 3–114), but this is not usually done because it is quicker to find the solution numerically, albeit iteratively.

Programs such as Excel and MATLAB allow us to easily solve for the specific volumes. However, one advantage of process simulators like Aspen Plus is that the physical properties of many components are saved in a database that users can access. In fact, users do not need to look up the numbers because Aspen Plus will do that when it needs them. The next section illustrates how to use each of these programs to solve equations of state.

SOLVING EQUATIONS OF STATE USING EXCEL* (SINGLE EQUATION IN ONE UNKNOWN)

There are at least two methods to solve algebraic equations using Excel. The first uses ‘Goal Seek’ while the other uses ‘Solver,’ and both are illustrated using a simple example – find the x that makes $f(x)$ zero:

$$f(x) = x^2 - 2x - 8 \quad (2.12)$$

Solution Using ‘Goal Seek’

Step 1 Open a spreadsheet and put the following statement in cell B1:

$$=A1*A1-2*A1-8 \quad (2.13)$$

Cell B1 is the equation that should be zero, and cell A1 contains the variable that is adjusted to make this happen.

Step 2 Under Tools choose Goal Seek. When a small screen appears, fill in the spaces to show the following:

Set cell	\$B\$1	
To value	0.	
By changing cell	A1	(2.14)

*Excel is a registered trademark of Microsoft Corporation, Inc.

Step 3 Click OK. The answer appears in the spreadsheet:

$$-2.000007 \quad 4.1137\text{E}-06 \quad (2.15)$$

Thus the solution found is -2 , with a tiny error – a small fraction of a percent. The test of whether the calculation is correct is shown in cell B1, which is 4.1×10^{-6} . This is not zero, but it is small enough for most purposes.

Step 4 If you want to decrease the tolerance to make the solution more accurate, under Tools and Options choose Calculation. Then, in Maximum Change add a few zeros in the middle (changing it from 0.001 to 0.000001), add a zero to the maximum number of iterations, choose OK, and repeat the Goal Seek. This time the answer is

$$-2 \quad -1.376\text{E}-8 \quad (2.16)$$

Step 5 To get the other root, put the value 3 in cell A1 and choose Goal Seek.

Solution Using Solver

You can solve the same problem using the Solver option in Excel.

Step 1 Under the Tools menu, click on Solver. Note: If the choice Solver does not appear, choose Add-Ins and load Solver from the Analysis ToolPak or the original Excel program disk (or see your system administrator for help).

Step 2 When the window opens, choose the option to make a cell equal to a value (or a maximum or minimum) by changing another cell. If you insert the appropriate cell locations, you will obtain the same answer as with Goal Seek. This time, however, it is much more accurate:

$$-2 \quad 5.3291\text{E}-14 \quad (2.17)$$

Example of a Chemical Engineering Problem Solved Using ‘Goal Seek’

Find the specific volume of *n*-butane at 500 K and 18 atm using the Redlich–Kwong equation of state.

Step 1 You must first find the critical temperature and pressure; *Perry’s Chemical Engineers’ Handbook* gives $T_c = 425.2$ K and $p_c = 37.5$ atm.

Step 2 Calculate values of *a* and *b* using Eq. (2.4). The value of gas constant in these units is 0.08206 l atm/g mol K.

Step 3 Prepare the spreadsheet shown in Figure 2.1. The title, name, and data will be useful when you come back to the problem at a future date.

	A	B	C	D	E	F	G
18			Radlich-Kwong Equation of State				
19			<i>n</i> -Butane				
20			Bruce Finlayson				
21							
22		Parameters - Ammonia			Data		
23	Tc	425.2	K		T	500	K
24	pc	37.5	atm		p	18	atm
25	R	0.08206	liter-atm/g mol K				
26							
27		Formulas			Results		
28	Tr	F23/B23			Tr	1.1759	
29	a	0.42748*(B25^2*B23^2/B24)*(B23/F23)^0.5			a	12.7981	
30	b	0.08664*(B25*B23/B24)			b	0.0806	
31	f(v)	F24*(F31^3-B25*(F23*(F31^2			v	2.0377	liter/g mol
32		+(F29-B25*(F23*(F30-F24*(F30^2)*F31-F29*(F30			f(v)	1.957E-07	Eq. (2.8)
33	ideal gas	volume	B25*(F23/F24)		ideal gas	2.2794	

Figure 2.1. Excel spreadsheet to find the volume of a nonideal gas.

Step 4 You enter the parameters in the parameter box. The cells containing the critical parameters and the temperature and pressure can be named T_c , p_c , T , and p , respectively. That way, the equation for $f(v)$ will be easier to understand.

Step 5 The lower box gives the equations actually used as well as the results. Use the Goal Seek command to make $f(v)$ (cell F32) equal to zero by changing cell v (F31).

Step 6 For reference, the result for an ideal gas is also shown, and indeed *n*-butane is close to behaving as an ideal gas under these conditions.

Step 7 How can you check this result? First, you have to be sure you have put the correct formulas into the spreadsheet, and that the units are consistent. That can only be determined by reference to the original equations and critical properties. It is easy to tell that $f(v) = 0$, but the solution is correct only if the equation for $f(v)$ is correct. In fact, the most challenging part of checking this calculation is the paper and pencil work before you develop the spreadsheet – to test the equations in the spreadsheet.

The techniques used to create this spreadsheet are shown in more detail in Appendix A, including: (1) inserting an equation for calculation; (2) inserting a text version of the equation for display; (3) creating a border around a group of cells; and (4) using Goal Seek.

SOLVING EQUATIONS OF STATE USING MATLAB*

(SINGLE EQUATION IN ONE UNKNOWN)

Nonlinear algebraic equations can be solved using MATLAB, too. First, you have to define the problem to solve by writing a file called an ‘m-file’; then, you check it; finally, you issue a command to solve it. These steps are analogous to the steps used in Excel. You can use MATLAB most effectively if you learn to use the Command Window and learn to create m-files and save them properly. See Appendix B for additional details.

*MATLAB is a registered trademark of The Math Works, Inc.

Step 1 Define the function. It is created using an m-file, called here f.m.,

$$\begin{aligned} \text{function } y=f(x) \\ y=x*x-2*x-8; \end{aligned} \quad (2.18)$$

Change the current directory in MATLAB (at the top of the command window) to a directory in which you wish to save your work and save it as f.m.

Step 2 Check the function. Issue the command

$$>>\text{feval}('f',2) \quad (2.19)$$

to get the result:

$$\text{ans}=-8. \quad (2.20)$$

You can easily calculate Eq. (2.18) to see that, for $x = 2$, the function value is -8 . Now you know the f.m. is correct.

Step 3 To find the value of x that makes $f(x) = 0$ in MATLAB, use the 'fzero' function. In the command window, issue the following command:

$$>>\text{fzero}('f',0) \quad (2.21)$$

$$\text{ans}=-2 \quad (2.22)$$

This command solves the following problem for x :

$$f(x) = 0 \quad (2.23)$$

starting from an initial guess of x_0 . Sometimes the function will have more than one solution, and that can be determined only by using the command with a different x_0 . You can test the result by saying

$$>>\text{feval}('f',\text{ans}) \quad (2.24)$$

To summarize the steps, step 1 defined the problem you wished to solve, step 2 checked your programming, and step 3 instructed MATLAB to solve the problem. It is tempting to skip the second step – checking your programming – but remember: If the programming is wrong, you will solve the wrong problem.

When examining the command 'fzero(f , x_0)' in MATLAB, the f defines which problem to solve, the x_0 is your best guess of the solution, and fzero tells MATLAB to vary x , starting from x_0 until the f is zero. In Excel's Goal Seek, the analogous steps were to make a cell zero by varying the value of another cell. 'Goal Seek' becomes fzero, a cell with an equation becomes f , and another cell becomes x_0 .

In all the commands and m-files above, the f can be replaced by other things, say 'probl'. Just be sure you change it in three places: the filename of the m-file, the first

line of the m-file (not absolutely necessary), and in the command. Additional forms of the command are:

$$\begin{aligned} >> \text{fzero}('function', x0, options) \\ >> z = \text{fzero}('f', x0) \end{aligned} \quad (2.25)$$

In the last example the result is put into the variable z . The options vector allows you to set certain quantities, like the tolerance. See how by typing

$$>> \text{help foptions}. \quad (2.26)$$

For the example used above, you can find the other root by running the program with $x_0 = 3$ in Eq. (2.21). Multiple roots can be found only if you search for them starting with different guesses.

Example of a Chemical Engineering Problem Solved Using MATLAB

Find the specific volume of n -butane at 500 K and 18 atm using the Redlich–Kwong equation of state.

Step 1 First, you need to prepare an m-file that will calculate the $f(x)$, or here $f(v)$, given the temperature, pressure, and thermodynamic properties. The file is shown below.

$$\begin{aligned} &\% \text{ calculate Eq. (2.8), Chapter 2} \\ &\text{function } y = \text{specvol}(v) \\ &\% \text{ in K atm l/gmol} \\ &\% \text{ parameters for n-butane} \\ &Tc = 425.2 \\ &pc = 37.5 \\ &T = 500 \\ &p = 18 \\ &R = 0.08206 \\ &a_{RK} = 0.42748 * (R * Tc)^2 / pc \\ &a_{RK} = a_{RK} * (Tc / T)^{0.5} \\ &b_{RK} = 0.08664 * (R * Tc / pc) \\ &y = p * v^3 - R * T * v^2 + (a_{RK} - p * b_{RK}^2 - R * T * b_{RK}) * v - a_{RK} * b_{RK}; \end{aligned} \quad (2.27)$$

This function, called 'specvol', defines the problem you wish to solve.

Step 2 To test the function 'specvol' you issue either of the following commands:

$$\begin{aligned} &\text{feval}('specvol', 0.2) \\ &\text{ans} = \text{specvol}(0.2) \end{aligned} \quad (2.28)$$

The feval function causes MATLAB to compute the value of y (the output defined in specvol) using the m-file named specvol when $v = 0.2$. The output you get is:

$$\begin{aligned} Tc &= 425.2000 \\ pc &= 37.5000 \\ T &= 500 \\ p &= 18 \\ R &= 0.08206 \\ aRK &= 13.8782 \\ aRK &= 12.7981 \\ bRK &= 0.0806 \\ y &= -0.6542 \end{aligned} \quad (2.29)$$

You should check these results line by line, especially the calculation of aRK , bRK , and y .

Step 3 When you use fzero, the function specvol will be evaluated for a variety of v . Thus, it is inconvenient to have the constants printed out on the screen every iteration. To avoid this, you change the function specvol by adding a semi-colon at the end of each line, ';'. This suppresses the output. Do this and save the m-file, specvol.

Step 4 Next you issue the command:

$$\begin{aligned} v &= \text{fzero}(' \text{specvol}', 0.2) \\ v &= 2.0377 \end{aligned} \quad (2.30)$$

In feval, the 0.2 was the v to be used in the calculation, whereas with fzero, the v is an initial guess of the answer. To check, you might evaluate the function to find how close to zero $f(v)$ is.

$$\begin{aligned} >> \text{ans} = \text{specvol}(v) \\ \text{ans} &= -2.2204\text{e-}15 \end{aligned} \quad (2.31)$$

Of course you expect this to be zero (or very close to zero) because you expect MATLAB to work properly. If MATLAB cannot find a solution, it will tell you. You can also use the command fsolve in the same way. To find out more about fzero and fsolve, enter the command help fzero or help fsolve.

Another Example of a Chemical Engineering Problem Solved Using MATLAB

Next rearrange the MATLAB code to compute the compressibility factor for a number of pressure values. The compressibility factor is defined in Eq. (2.32):

$$Z = \frac{pv}{RT} \quad (2.32)$$

For low pressures, where the gas is ideal, the compressibility factor will be close to 1.0. As the pressure increases, it will change. There are two new features illustrated: the use of global and plotting.

Step 1 The code called 'run_volplot' computes the specific volume for pressures from 1 to 31 atmospheres and then calculates the corresponding compressibility factor. The first statement in 'run_volplot' is a global command. The variables are identified, T , p , T_c , and so on, and then assigned values. In other programs, such as 'specvol,' the same global command is used, and these variables can then be accessed. See Appendix B for more information.

The next part of the program is a loop with 31 steps. The pressure is changed from 1, to 2, 3, 4, ..., 31. For each pressure, the constants a and b are calculated, as a_{RK} and b_{RK} , respectively. Then 'specvol' is called to find the specific volume for those conditions; the answer is stored in the variable vol as an array, or vector. The compressibility factor is calculated and stored in the vector Z . Finally, a plot is made with $pres$ along the x -axis and Z along the y -axis. The result is Figure 2.2.

```
% run_volplot
global T p Tc pc R aRK bRK
% in K atm l/gmol
% parameters for n-butane
Tc=425.2
pc=37.5
T=500
R=0.08206
for i=1:31
    pres(i)=i;
    p=i;
    aRK=0.42748*(R*Tc)^2/pc;
    aRK=aRK*(Tc/T)^0.5;
    bRK=0.08664*(R*Tc/pc);
    vol(i)=fzero('specvol',0.2);
    Z(i)=pres(i)*(vol(i)/R*T;
end
plot(pres,Z)
xlabel('pressure (atm) ')
ylabel('Z')
```

(2.33)

Step 2 The m-file specvol is changed to use the global command, as shown in Eq. (2.34). Now the parameters are available to specvol because they are defined as global variables

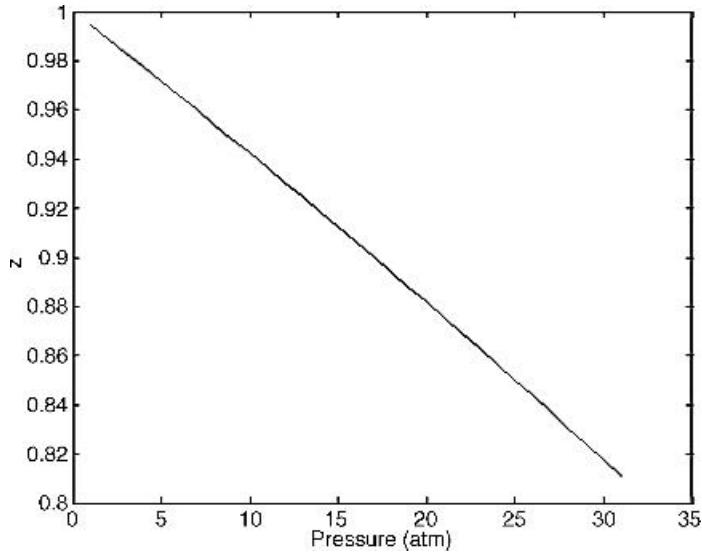


Figure 2.2. Compressibility factor for *n*-butane.

and have been set in the program 'run_volplot.'

```
% calculate Eq. (2.8), Chapter 2
function y=specvol(v)
global T p Tc pc R aRK bRK
y=p*v^3-R*T*v^2+(aRK-p*bRK^2-R*T*bRK)*v-aRK*bRK;
```

(2.34)

Step 3 Because you have already checked the program `specvol`, you do not need to check it again. You will want to put semi-colons (;) at the end of the lines in `specvol` because you do not need intermediate results. You can check the values of *aRK* and *bRK* for one of the pressures, to ensure they are correct in the program `run_volplot`, but here these statements were copied from `specvol` directly, so they do not need to be checked again. The use of *vol(i)*, *Z(i)*, *pres(i)* is easy to check from the graph, Figure 2.2.

Step 4 An alternative way to calculate the compressibility factor would be to use the following command after the loop:

$$Z = \text{pres} \cdot \text{vol} / (R \cdot T); \quad (2.35)$$

This command computes the same thing as

$$Z(i) = \text{pres}(i) \cdot \text{vol}(i) / (R \cdot T); \quad (2.36)$$

inside the loop, but it does it element-by-element outside the loop. Note the symbol '*.**'; the period causes the element-by-element calculation.

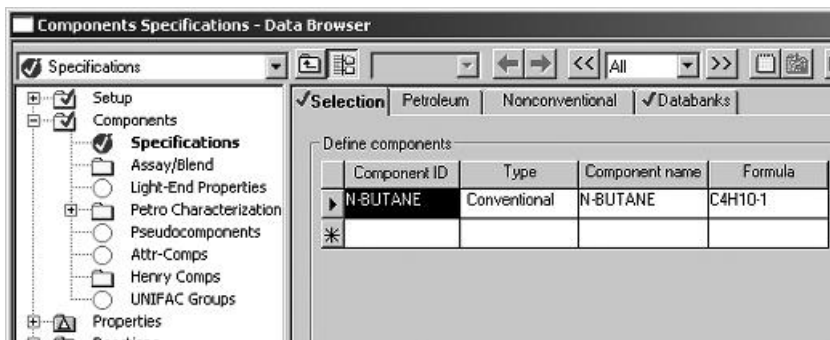


Figure 2.3. Aspen Plus window for component names.

EQUATIONS OF STATE WITH ASPEN PLUS*

You can also find the specific volume using Aspen Plus. One feature of Aspen Plus allows users to find properties of a pure substance. Given here are commands that will enable you to find the specific volume of *n*-butane at the stated conditions. You may need to review Appendix C, too, which has more detail about Aspen Plus.

Example

Find the specific volume of *n*-butane at 500 K and 18 atm using the Redlich–Kwong equation of state option in Aspen Plus.

Step 1

- (1) Start Aspen Plus and choose Template.
- (2) When the window appears, choose General with Metric Units.
- (3) In the Run Type (lower right-hand corner), choose Property Analysis.
- (4) Click OK when the Aspen Plus engine window appears. Note: This last step is specific to your installation.

Step 2 In the list on the left, choose Component/Specifications and enter the names or formulas of the chemicals, as shown in Figure 2.3. If there is no list on the left, click on the eye glasses or choose the menu Data/Components. If Aspen Plus does not recognize your chemical, a window appears that allows you to search again, and it will suggest a number of possibilities. When the components are completely specified, it is important that there is an entry for every chemical in the column labeled ‘Component name.’ The first column is what you are naming the chemicals, but the third column is what Aspen Plus is using when it gets physical properties. If that column is blank, the program will not work.

*Aspen Plus is a registered trademark of Aspen Technology, Inc.

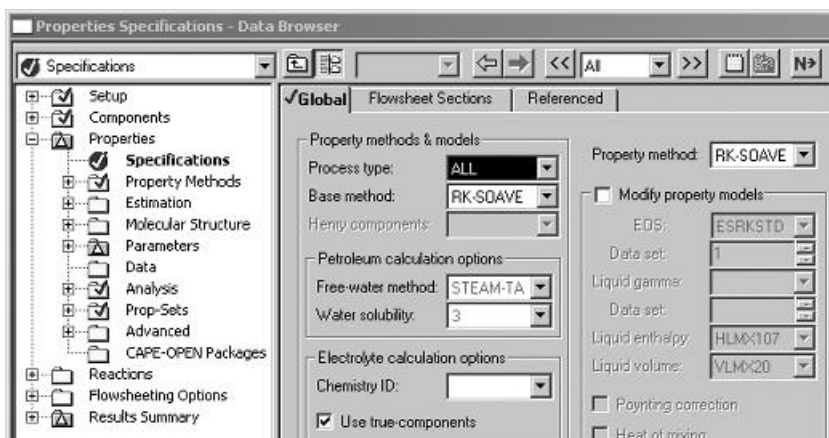


Figure 2.4. Aspen Plus window for property method.

Step 3 In the list on the left, choose Property/Specifications. Set the property method as illustrated in Figure 2.4. For this application, choose RK–Soave by scrolling down. Now all the icons on the left should be blue, meaning the menus are completely filled out. If any menus are still red, go back and complete them.

Step 4 In the menus at the top choose Tools/Analysis/Property/Pure (the pure will be grayed out until you do steps 2 and 3). In the window that appears (see Figure 2.5), choose the following:

Property type: thermodynamic

Property: V (need to move the cursor down to see it)

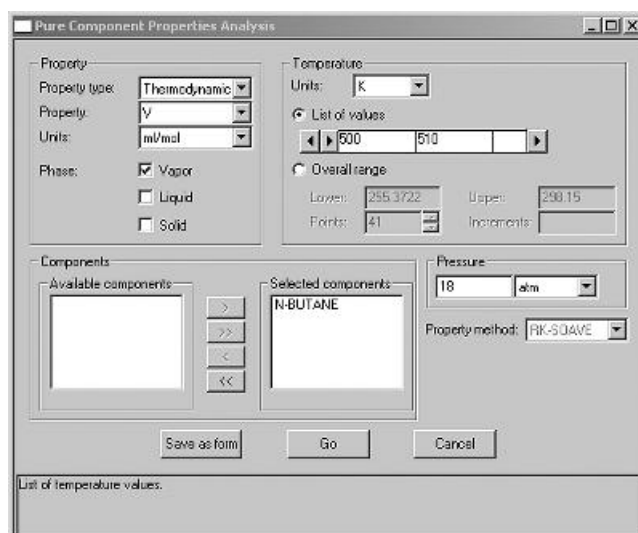


Figure 2.5. Aspen Plus window for pure component property analysis.

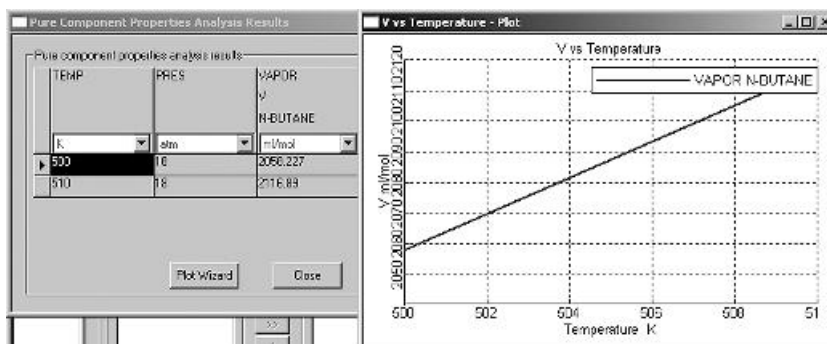


Figure 2.6. Aspen Plus window showing pure component property analysis.

Check vapor, uncheck liquid

Units: choose ml/mol

Components: select *n*-butane

Temperature, choose Units: K and • List, put in 500 and 510

Pressure: choose 18 atm

Property: scroll down to find RK–Soave

Click ‘Go’.

Step 5 A graph appears with the result plotted (see Figure 2.6). You can read results from the graph, or if you move the graph slightly, you will find a table behind it, giving the exact answer. The result is 2058 ml/mol, or $2.058 \text{ cm}^3/\text{g mol}$, which compares favorably with the result of 2.038 when using Excel or MATLAB. The critical properties are slightly different in the three cases.

Specific Volume of a Mixture

If you wish to obtain the specific volume of a mixture, it is necessary to use another approach, since the procedure above works only for pure substances. This time, use one of the units and specify the stream into the unit. The calculation will tell you the specific volume.

Find the specific volume of a mixture consisting of 630 kmol/h of carbon monoxide, 1130 kmol/h of water, 189 kmol/h of carbon dioxide, and 63 kmol/h of hydrogen at 1 atm and 500 K. The specific volume is the solution to the Redlich–Kwong equation of state, Eq. (2.8).

Step 1

- (1) Start Aspen Plus and choose Template; OK.
- (2) When a window appears, choose General with Metric Units.
- (3) In the Run Type (lower right-hand corner), choose Flowsheet.
- (4) Click OK when the Aspen Plus engine window appears. (This last step is specific to your installation.)

Step 2 If the bottom of the screen does not show the units, use the View/Model Library pull-down menu or press the F10 key.

- (1) In the tabs at the bottom, choose Pressure Changes.
- (2) Click on the Compressor.
- (3) Click on the flow sheet, and a compressor appears.

Step 3 To add the input and output streams, click on Material Streams (lower left-hand corner), click on the flowsheet and drag a stream to the red arrow that is input to the compressor unit and click. Next, click the red arrow coming out and drag the stream away and click, giving Figure 2.7. Note: To make changes in the location of the streams or units, you can click on the arrow just above the Material Stream button. You can toggle back and forth between the arrow and Material Stream in the lower left corner as you improve the presentation of your flowsheet. When the flowsheet is showing, click on Material Streams. If any red arrows show in the flowsheet, it means that the unit is not properly connected; an input or output stream is necessary. Fix that before proceeding – you may have placed your cursor incorrectly when you drew the streams.

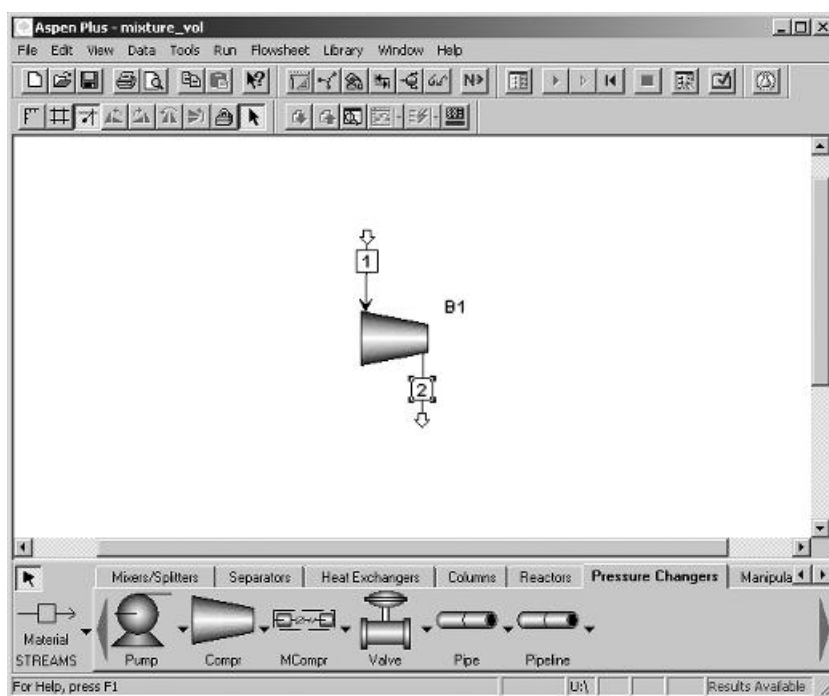


Figure 2.7. Flowsheet for single compressor.

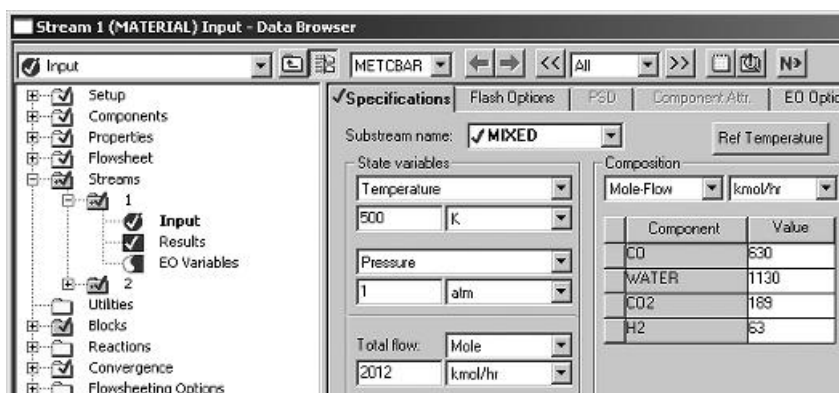
Step 4 Click on the glasses. This will bring up a menu to the left of the screen. The red boxes indicate that you still need to supply information. Start at the top and work down, turning the red boxes into blue boxes by filling in the forms. In the list on the left, choose Component/Specifications (see Figure 2.3). Type in the names or formulas of each chemical. If Aspen Plus does not recognize a chemical, a window appears that allows you to search again, and the program will suggest a number of possibilities. When the components are completely specified, it is important that you have an entry for every chemical in the column labeled ‘Component name’. The first column is what you are naming the chemicals, but the third column is what Aspen Plus uses when it gets physical properties. If that column is blank, the program will not work.

Step 5 In the list on the left, choose Property/Specifications. In property method, scroll down to get RK–Soave (see Figure 2.4). To eliminate the red ‘pair parameters,’ open the window and click on the designated (recommended) data set.

Step 6 In the list on the left, choose Streams by double-clicking on it. Inside that folder are two or more folders, one for each stream. Choose the input stream, click on it, and insert the temperature, pressure, and flow rate in units you choose, as shown in Figure 2.8. You can specify the units for input numbers, thus avoiding having to do unit conversions yourself.

Step 7 In the list on the left, choose Blocks, then B1 (or whatever you have named your compressor). Choose Specifications, and choose Type: isentropic; insert the discharge pressure, as shown in Figure 2.9. (For this problem, you will use the inlet stream; thus you can put in any discharge pressure you want, as long as it is above the inlet pressure.)

Step 8 Choose the  at the top. Notice in Figure 2.9 that there are two of these showing – choose either one. If the input is incomplete, a window will appear to notify you and direct you to the missing data. If the input is complete, a window will appear to notify you of that, too. Click on the button to make it perform the calculation. This will cause the calculation of the process (here one unit) to proceed. Once the calculations finish (read the error messages, if any), click the Results box (lower one) to return to the



Component	Value
CO	530
WATER	1130
CO2	169
H2	53

Figure 2.8. Setting stream information.



Figure 2.9. Setting compressor information.

regular menu. Then look at Results/Streams. The stream data will appear in tabular form as shown in Table 2.1.

Step 9 You can obtain the specific volume by dividing the volumetric flow rate by the molar flow rate, $46449/2012 = 40.98 \text{ m}^3/\text{kmol}$. To obtain Table 2.1 in spreadsheet format, place your cursor in the upper-left cell of the Stream table, copy down to the lower-right cell, copy, and paste into Excel. You can also have the Stream Table copied to the flowsheet, and it can be copied from there as a picture.

Your report of the results should explain what problem you have solved and how you solved it (focus on the chemical engineering information rather than the detailed step-by-step process on each screen), describe how you checked your results, and give the results. In this case you took a stream of specified composition, pressure, and temperature, sent it to a compressor, and obtained the data shown in the Table 2.1. The type of thermodynamic

TABLE 2.1. Stream Information for Compressor Problem

Temperature, °C	226.85	619.601
Pressure, bar	1.01325	7.09275
Vapor fraction	1	1
Mole flow, kmol/h	2012	2012
Mass flow, kg/h	46448.67	46448.67
Volume flow, m ³ /h	82443.57	21067.61
Enthalpy, MMkcal/h	−96.44524	−89.59337
Mole flow, kmol/h		
CO	630	630
Water	1130	1130
CO ₂	189	189
H ₂	63	63

option must be specified (here Redlich–Kwong) and justified (depending upon your level of expertise in chemical engineering). You might assume that Aspen Plus did the calculation correctly, but you can also review the results to see if they are reasonable.

CHAPTER SUMMARY

You have solved a very simple problem to find the specific volume of a pure component or a mixture using three methods: Excel, MATLAB, and Aspen Plus. Excel is readily available, and easy to use. MATLAB is a bit more difficult for beginners because it uses files, which require data transfer. It is extremely powerful, though, and is needed for other classes of problems. With both Excel and MATLAB, you must look up the critical temperature, critical pressure, and perhaps the acentric factor of each chemical. You then must carefully and laboriously check your equations, one by one.

When you use Aspen Plus, the parameters are stored in a database, and the calculations are pre-programmed. Your main concern is to use the graphical user interface (GUI) correctly. Aspen Plus is extremely powerful and is needed for other classes of problems.

For using all three programs, you want to compare at least some of the calculations with experimental data, to verify that the type of thermodynamics you have chosen is appropriate to the physical case you are solving.

PROBLEMS

- 2.1. Find the molar volume of ammonia gas at 56 atm and 450 K using the Redlich–Kwong equation of state, $T_c = 405.5$ K, $p_c = 111.3$ atm, $a = 4.2527$, $b = 0.02590$; units of a and b correspond to v in l/g mol. (1) Use Excel; (2) use MATLAB; (3) use Aspen Plus with the RK–Soave thermodynamic option.
- 2.2. Find the compressibility factor of ammonia gas at conditions from 50 to 250 atm and 400 K using the Redlich–Kwong equation of state in Excel. (Hint: Before beginning your spreadsheet, think about how you can organize it so that you can copy formulas from cell to cell easily.)
- 2.3. Consider the following mixture going into a water–gas shift reactor to make hydrogen for the hydrogen economy. CO, 630 kmol/h; H₂O, 1130 kmol/h; CO₂, 189 kmol/h; H₂, 63 kmol/h. The gas is at 1 atm and 500 K. Use Excel (or MATLAB) to compute the specific volume using: (1) ideal gas law; (2) Redlich–Kwong equation of state; and (3) Redlich–Kwong–Soave equation of state. The acentric factors for the RK–Soave method are:

$$\text{CO}, 0.049; \quad \text{H}_2\text{O}, 0.344; \quad \text{CO}_2, 0.225; \quad \text{H}_2, -0.22.$$

Where did you get the other data you needed? How do the three answers compare? Is the gas ideal or not? Comment. Then redo the calculations for a pressure of 200 atm and comment on the results.

- 2.4. Consider a mixture of 25 percent ammonia, and the rest nitrogen and hydrogen in a 1:3 ratio. The gas is at 270 atm and 550 K. Use Excel (or MATLAB) to compute the specific volume using: (1) ideal gas law; (2) Redlich–Kwong equation of state; and

- (3) Redlich–Kwong–Soave equation of state. Where did you get the data you needed? How do the three answers compare? Is the gas ideal or not? Comment.
- 2.5.** Find the molar volume of methanol gas at 100 atm and 300°C using the Redlich–Kwong equation of state, $T_c = 512.6$ K, $p_c = 79.9$ atm, $a = 3.023$, $b = 0.04561$; units of a and b correspond to v in l/g mol.
- 2.6.** Consider the following mixture that is coming out of a methanol reactor: CO, 100 kmol/h; H₂, 200 kmol/h; methanol, 100 kmol/h. The gas is at 100 atm and 300°C. Compute the specific volume using: (1) ideal gas law; (2) Redlich–Kwong equation of state; and (3) Redlich–Kwong–Soave equation of state. The acentric factors for the RK–Soave method are: CO, 0.049; H₂, -0.22; methanol, 0.559. Where did you get the other data you needed? How do the three answers compare? Is the gas ideal or not? Comment.

3

VAPOR–LIQUID EQUILIBRIUM

Have you ever driven past a refinery and wondered what happens in those tall towers? Some of them are distillation towers that are used to separate a mixture of chemicals into two or more streams, each a relatively pure stream of one of the chemicals. The physical process governing that separation is vapor–liquid equilibrium. It has been estimated that 10 percent of the energy used commercially in the United States is used in distillation. Thus, it is important to make this process as efficient as possible.

Take a mixture of two or more chemicals in a temperature regime where both have a significant vapor pressure. The composition of the mixture in the vapor is different from that in the liquid. By harnessing this difference, you can separate two chemicals, which is the basis of distillation. To calculate this phenomenon, though, you need to predict thermodynamic quantities such as fugacity, and then perform mass and energy balances over the system. This chapter explains how to predict the thermodynamic properties and then how to solve equations for a phase separation. While phase separation is only one part of the distillation process, it is the basis for the entire process. In this chapter you will learn to solve vapor–liquid equilibrium problems, and these principles are employed in calculations for distillation towers in Chapters 6 and 7. Vapor–liquid equilibria problems are expressed as algebraic equations, and the methods used are the same ones as introduced in Chapter 2.

FLASH AND PHASE SEPARATION

Suppose you put some water in an open pan on the stove, initially at room temperature. The partial pressure of water in the air (at equilibrium) will equal the vapor pressure of water at that temperature. Now heat the pan. The vapor pressure increases, since it

increases as the temperature rises, and thus the partial pressure of water also increases. If the partial pressure of water at the pan temperature exceeds the partial pressure of water in the room (usually set by the humidity), the water will evaporate.

Next, imagine doing the same thing with a mixture of two chemicals in a closed vessel. The closed vessel is one with a piston that can move so the pressure inside remains constant. The two chemicals have different boiling points, and different vapor pressures at a given temperature. As you increase the temperature of the vessel, the relative amount of each chemical in the vapor changes, because one is more volatile than the other. At temperatures below the bubble point, T_{bubble} , the mixture is entirely a liquid. At temperatures above the dew point, T_{dew} , the mixture is entirely a vapor. At temperatures in between, both liquid and vapor co-exist. The composition of the liquid and vapor are not the same, however. Thus, as you gradually increase the temperature from a low value, some vapor forms, and this vapor is richer in the more volatile component. As the temperature increases further, more and more vapor forms. Finally, as the last drop of liquid evaporates, all the material is in the vapor phase, which has the same composition as the original liquid. However, between the bubble point and dew point, the composition of the liquid and vapor are changing as the temperature increases, and it is this change that you need to calculate.

There is another scenario that, unfortunately, is purely imaginary. In that scenario, as the temperature increases, chemical one evaporates completely when the temperature reaches its boiling point. As the temperature increases further, the boiling point for the second chemical is reached, and it all evaporates. You wish it did happen this way! Despite the fact that in real life the separation of the two chemicals is never complete in either liquid or vapor phase, it is still a useful phenomenon and forms the basis for distillation.

This chapter looks first at equations governing an isothermal flash, and then shows how you can predict the thermodynamic quantities you need to solve the isothermal flash problem. The problems are all sets of algebraic equations, and you can solve these problems using Excel[®] and MATLAB[®]. The chapter then addresses more complicated vapor-liquid separations, but now using Aspen Plus[®] because of its large database.

ISOTHERMAL FLASH – DEVELOPMENT OF EQUATIONS

Consider the flow sheet shown in Figure 3.1. Suppose you know the temperature, pressure, and overall composition of the inlet stream. The mole fractions of the chemicals in the

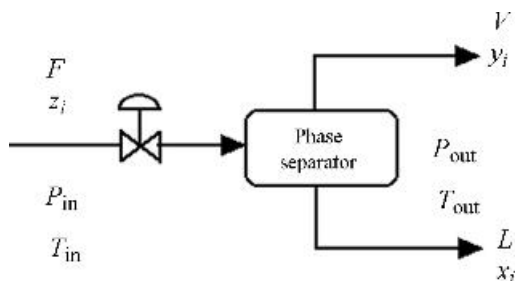


Figure 3.1. Flash phase separator.

inlet are called $\{z_i\}$. In the phase separator, however, the liquid and vapor are separated. The mole fraction of the chemicals in the vapor phase are called $\{y_i\}$ and those in the liquid phase are called $\{x_i\}$. When the vapor and liquid are in equilibrium, you can relate the mole fractions of each chemical in the vapor and liquid by the equation:

$$y_i = K_i x_i \quad (3.1)$$

The members of the set $\{K_i\}$ are called K -values, and they can be predicted from thermodynamics, as shown below. For now, though, assume that you know their values.

To derive the equation governing the phenomenon, you first sum the mole fractions of vapor and liquid over all components.

$$\sum_{i=1}^{NCOMP} y_i = 1, \quad \sum_{i=1}^{NCOMP} x_i = 1 \quad (3.2)$$

Then, subtract those two equations:

$$\sum_{i=1}^{NCOMP} y_i - \sum_{i=1}^{NCOMP} x_i = 0 \quad (3.3)$$

Next, substitute the equilibrium expression and rearrange to obtain:

$$\sum_{i=1}^{NCOMP} K_i x_i - \sum_{i=1}^{NCOMP} x_i = 0 \quad \text{or} \quad \sum_{i=1}^{NCOMP} (K_i - 1)x_i = 0 \quad (3.4)$$

Next, make a mass balance for each component over the phase separator. F is the total molar flow rate, V is the molar flow rate of the vapor, and L is the molar flow rate of the liquid. The mole balance is then

$$z_i F = y_i V + x_i L \quad \text{and} \quad F = V + L \quad (3.5)$$

Divide by F and define v' as the fraction of the feed that is vapor:

$$z_i = y_i v' + x_i(1 - v'), \quad v' = \frac{V}{F} \quad (3.6)$$

By using the equilibrium expression again, you can write this as

$$z_i = K_i x_i v' + x_i(1 - v') = (K_i - 1)x_i v' + x_i \quad (3.7)$$

Solve for the mole fractions in the liquid:

$$x_i = \frac{z_i}{1 + (K_i - 1)v'} \quad (3.8)$$

TABLE 3.1. Mole Fractions and K -values for Vapor–Liquid Example

	Mole Fraction	Phase Equilibrium K -value
Propane	0.1	6.8
<i>n</i> -Butane	0.3	2.2
<i>n</i> -Pentane	0.4	0.8
<i>n</i> -Octane	0.2	0.052
	1	

Put that expression into Eq. (3.4) to obtain the final equation:

$$\sum_{i=1}^{NCOMP} \frac{(K_i - 1)z_i}{1 + (K_i - 1)v'} = 0 \quad (3.9)$$

This is called the Rachford–Rice equation. Notice that, if the K -values and inlet compositions $\{z_i\}$ are known, this is a nonlinear equation to solve for v' . Thus, you can apply here the same methods used with Excel and MATLAB in Chapter 2. Once the value of v' is known, you can calculate the value of the liquid compositions, $\{x_i\}$, and vapor compositions, $\{y_i\}$, using Eqs. (3.1) and (3.3). The mole balance is then complete.

Example Using Excel

Suppose you have a mixture of hydrocarbons in the inlet stream. You want to find the fraction of the stream that is vapor and the mole fraction of each chemical in the vapor and liquid streams. Table 3.1 shows the flow rates and K -values at 180°F and 70 psia, using a basis of one mole per unit of time.

To solve this problem, prepare the spreadsheet as shown in Table 3.2.

Step 1 ‘Term1’ is the numerator in Eq. (3.9) and ‘term2’ is the denominator, and these are calculated in columns D and E.

Step 2 The ratios of term1 over term2 are calculated and put into column F; this column is summed in cell F9, which represents Eq. (3.9).

TABLE 3.2. Spreadsheet for Flash Example

	A	B	C	D	E	F	G	H
1				$v = 0.425837$				
2			Phase					
3			Equilibrium	term1 =	term2 =	Ratio =		
4		z_i	K -value	$(K_i - 1)z_i$	$(K_i - 1)v + 1$	term1/term2	x_i	y_i
5	Propane	0.1	6.8	0.58	3.4699	0.1672	0.0288	0.1960
6	<i>n</i> -Butane	0.3	2.2	0.36	1.5110	0.2383	0.1985	0.4368
7	<i>n</i> -Pentane	0.4	0.8	−0.08	0.9148	−0.0874	0.4372	0.3498
8	<i>n</i> -Octane	0.2	0.052	−0.1896	0.5963	−0.3180	0.3354	0.0174
		1			$f(v) = 8.2719\text{E-}07$		0.9999996	1.0000005

Step 3 You can then use Goal Seek to make the Rachford–Rice equation (cell F9) zero by changing the fraction of the feed that is vapor (cell D1), giving the result shown. Once you find the fraction vapor, the mole fractions in the two phases are easy to calculate using Eqs. (3.8) and (3.1), and these are included in columns G and H.

How might you check these results?

Step 1 Of course, the most important aspect is to have the correct K -values, which can only be determined by comparing them with experimental data. The entire calculation depends upon those K -values.

Step 2 You need the Rachford–Rice equation to sum to zero, which is evident in the spreadsheet.

Step 3 You can check that the sum of vapor mole fractions equals one, and the sum of liquid mole fractions equals one.

Step 4 You can also check one term in the Rachford–Rice equation using detailed calculations, and then copy the equation down. This ensures that the formula is correct for all components.

THERMODYNAMIC PARAMETERS

Where did the vapor–liquid equilibrium K -value come from? It is defined as

$$K_i = \frac{\gamma_i f_i^0}{\phi_i p} \quad (3.10)$$

where γ_i is the activity coefficient, f_i^0 is the fugacity of the pure chemical, ϕ_i is the fugacity coefficient in the vapor phase, and p is the total pressure. Most of these quantities can be calculated using thermodynamics. For example, your thermodynamics textbook derives the following equation for fugacity of a pure substance:

$$\ln\left(\frac{f}{p}\right) = -RT \int_0^p \left(\frac{RT}{p} - \hat{v}\right) dp \quad (3.11)$$

This means that, if you know the p – V – T behavior of a chemical, then you can find the fugacity by calculating the integral. In this text you do not need to do that, because Aspen Plus will do it for you. Instead you can either use K -values from the literature or do the calculations with a process simulator.

The other important parameter in Eq. (3.10) is the activity coefficient. The activity coefficient depends upon the chemicals involved, particularly whether they are polar or not (have a dipole moment). There are many options:

Option 1 – if the solution is ideal, the activity coefficient is 1.0.

Option 2 – the regular solution theory by Scatchard–Hildebrand works best for nonpolar compounds and compounds with similar molecular weights, and it may work for weakly polar compounds.

Option 3 – the Wilson equation is more robust, and it works for polar compounds and can handle azeotropes, but not two liquid phases. It requires pair-interaction parameters, which must be determined from data.

Option 4 – the nonrandom two-liquid (NRTL) equation is more robust, and it works for polar compounds and can handle azeotropes as well as two liquid phases.

Option 5 – the UNIFAC group-contribution method constructs the activity coefficient out of parameters that have been correlated to specific molecular fragments. It works in the same cases as the NRTL equation, but UNIFAC can predict activity coefficients in the absence of specific data provided all the molecular fragments have been correlated from other data.

Chapter 6 illustrates other options available in Aspen Plus. The trick is to know which option to choose for your chemical system. Unfortunately, the only way to know for sure that you have made a good choice is to compare your predictions with experimental results.

For most of these options, the activity coefficient depends upon the liquid mole fractions, and that means the K -values depend upon the liquid mole fractions, too. Thus, there is not just one K -value that you can use in the calculations. Instead, one option is to choose the liquid mole fractions, determine the activity coefficients, then the K -values and see if the liquid mole fractions changed significantly. If so, you might use the new values and repeat the process. If this procedure did not work, you would have to use more sophisticated methods to solve the Rachford–Rice equation using process simulators such as Aspen Plus.

There is one special case that is easy, however. If the solution is ideal, the activity coefficients are 1.0. If the pressure is low enough (say less than 10 atm), the vapor phase is essentially an ideal gas. In that case the fugacity coefficient of the vapor is 1.0, too. The fugacity of the liquid in the standard state is the vapor pressure, too, at low pressures. With those assumptions, the K -value becomes

$$K_i = \frac{vp_i}{p} \quad (3.12)$$

where vp_i is the vapor pressure at a given temperature. In this special case, you can find the K -values knowing the vapor pressure as a function of temperature and the total pressure of the system. Clearly, light components (lower boiling points) have a larger K than heavy components (higher boiling points), and thus their K -value will be larger. If two phases co-exist at equilibrium, at least one of the components needs a K -value greater than 1 and at least one of the other components needs a K -value less than one. Even then two phases are not guaranteed, and the result depends upon the composition as well as the K -values.

Example Using MATLAB

You can solve the same problem using MATLAB. First, you define a function that represents the problem you wish to solve [Eq. (3.9)]. Then, you check the function to make sure it is correct. Finally, you use the ‘fzero’ function to find the solution. The function defining the problem is:

```

{%vapor-liquid equilibrium}
function y=vpequil(v)
z=[0.1 0.3 0.4 0.2]
K=[6.8 2.2 0.8 0.052]
sum1=0.;
for i=1:4
    num=(K(i)-1)*z(i)
    denom=1+(K(i)-1)*v
    sum1=sum1+num/denom
end
y=sum1}

```

(3.13)

Step 1 Program the function and save it in your working directory as vpequil.m.

Step 2 Issue the command 'vpequil(0.2)' in the command window; the output should be:

```

z=0.1000 0.3000 0.4000 0.2000
K=6.8000 2.2000 0.8000 0.0520
num=0.5800
denom=2.1600
sum1=0.2685
....
ans=0.2415

```

(3.14)

Calculation with a calculator gives the same answers, so you have verified that the program is correct.

Step 3 Add semi-colons at the end of each line of the m-file and save it. Solve the problem by issuing the fzero command in the command

```

>>fzero(@vpequil,0.2)
ans=0.4258

```

(3.15)

As expected, the result is the same as that obtained using Excel.

Example Using Aspen Plus

Aspen Plus allows you easily to solve this same problem using the Flash2 unit operation.

Step 1 Start Aspen and choose Template; OK. When the window appears, choose General with English Units. In the Run Type (lower right-hand corner), choose Flowsheet. Click OK when the Aspen engine window appears. (This last step is specific to your installation.)

Step 2 If the bottom of the screen does not show the units, use the View/Model Library pull-down menu or press the F10 key. In the units at the bottom, choose Separations, then click on Flash2.

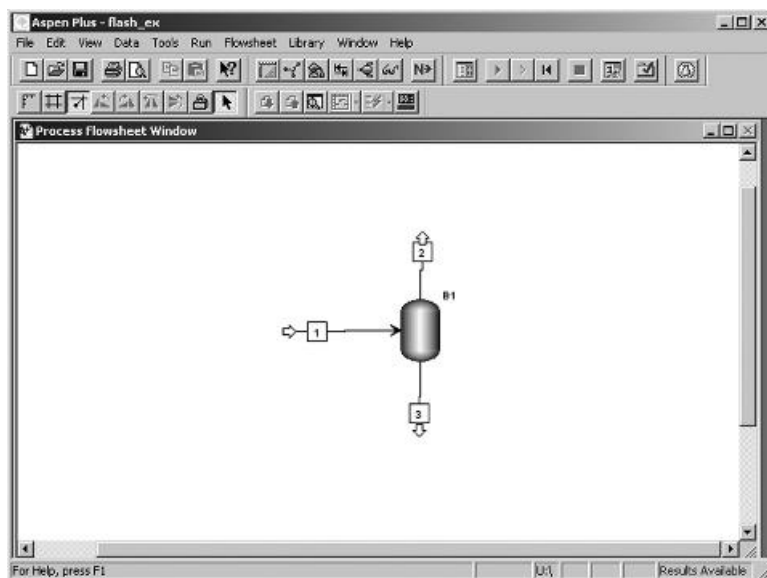


Figure 3.2. Flowsheet for Flash2 unit.

Step 3 Click on the flow sheet, and a flash (phase separation) unit appears. To add the input and output streams, click on Material Streams (lower left-hand corner), click on the flowsheet and drag a stream to the red arrow that is input to the flash unit. Click on the red arrows coming out and drag the stream away. Note: To make changes in the location of the streams or units, you can click on the arrow just above the Material Stream button. You can toggle back and forth between the arrow and Material Stream in the lower left corner as you improve the presentation of your flowsheet. When the flowsheet is showing, click on Material Streams. If any red arrows show in the flowsheet, it means that the unit is not properly connected; an input or output stream is necessary. Fix that before proceeding – you may have placed your cursor incorrectly when you drew the streams. The result should look something like Figure 3.2.

Step 4 Click on the glasses. This will bring a menu to the left of the screen. The boxes that are red indicate that you still need to supply information. Start at the top and work down, turning the red boxes into blue boxes by filling in the forms.

Step 5 In the list at the left, choose Component/Specifications and identify the components as you did in Chapter 2 (see Figure 2.3). Type the name or formula of the chemicals. If Aspen Plus does not recognize your chemical, a window appears that allows you to search again, and it will suggest a number of possibilities. When the components are completely specified, you should have an entry for every chemical in the column labeled 'Component name.' The first column is what you are naming the chemicals, but the third column is what Aspen Plus uses when it obtains physical properties. If that column is blank, the program will not work.

Step 6 In the list at the left, choose Property/Specifications and choose the thermodynamic model as you did in Chapter 2 (see Figure 2.4). In property method, scroll down to get RK-Soave. You also need to tell the computer which database to use for the interaction

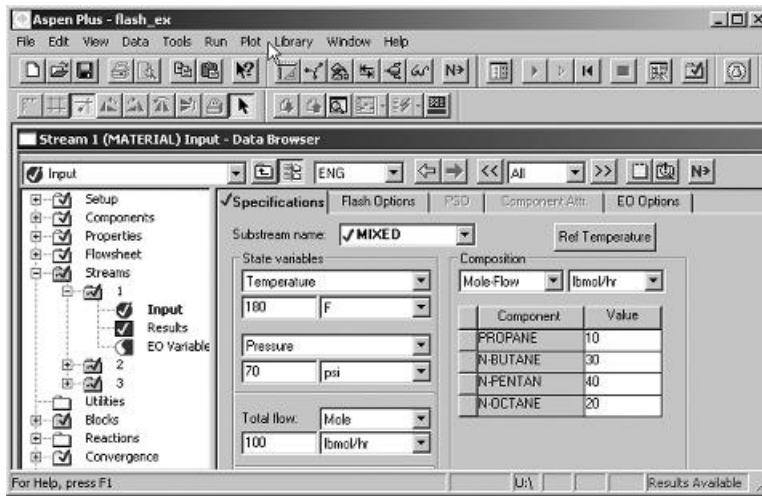


Figure 3.3. Stream information for Flash2 unit.

parameters in RK-Soave. Choose one of the options for interaction parameters. The results shown below use RKSKBV-1 from the SDF table.

Step 7 In the list on the left, choose Streams by double-clicking on it. Inside that folder are two or more folders, one for each stream. Choose the input stream, click on it, and insert the temperature, pressure, and flow rates of every chemical, as shown in Figure 3.3. Note that you can specify the units you would like to use as you put in the numbers. That way you can use the numbers as given to you, without doing unit conversion yourself.

Step 8 In the list at the left, choose Blocks, then B1 (or whatever you have named your Flash2 unit), then Input. Choose Specifications, and insert the temperature and pressure of the unit as shown in Figure 3.4. (For this problem, you can use the same temperature and pressure as the inlet stream, and regard the unit simply as a phase separator.)

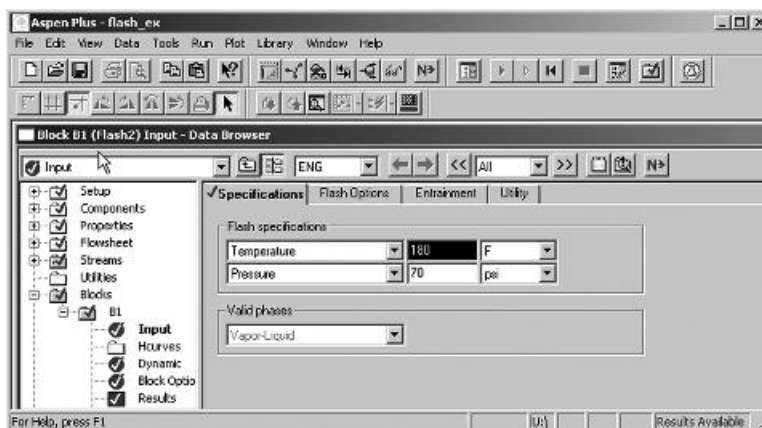


Figure 3.4. Block information for Flash2 unit.

Step 9 Choose the **N2** at the top. Notice in Figure 3.4 that there are two of these showing – choose either one. If the input is incomplete, a window will appear to notify you and direct you to the missing data. If the input is complete, a window will appear to notify you of that, too. Click on the button to make it perform the calculation. This will cause the calculation of the process (here one unit) to proceed. Once the calculations finish (read the error messages, if any), click the Results box (lower one) to return to the regular menu. Then look at Results/Streams. The stream data will appear in tabular form, as shown in Table 3.3. If you click Stream Table, it will also be reproduced on your flowsheet.

Step 10 For your report, select the sketch of the process by putting your cursor on the upper-left corner and dragging it to the lower-right corner. Choose Copy, switch to a word processing program, and choose Paste. Do the same thing with the stream table by clicking on it. Alternatively, place your cursor in the upper-left cell of the Stream table, copy down to the lower-right cell, copy, and paste into Excel. Your written report should include more than the process sketch and stream table. You still need to explain what problem you have solved and how you solved it, focusing on the chemical engineering information, not the detailed steps on the computer screen. Describe how you checked your results, and give the results. In this case, you took a stream of specified composition, pressure and temperature that was in two phases, separated the gas from the liquid, and calculated the composition of each phase. Depending on your level of expertise in chemical engineering, you also need to specify and justify the type of thermodynamic model you chose. You might assume that Aspen Plus did the calculation correctly, but you also should look at the results to see if they are reasonable.

From the Table 3.3, you can see that the vapor fraction is 0.419 (compared with 0.426 in Table 3.2). Since Aspen Plus computed the K -values, it may have used different ones than you did. You can find out what Aspen Plus used by dividing y_i by x_i for each component. The results are: propane, 5.2; n -butane, 2.1; n -pentane, 0.84; n -octane, 0.067. While these numbers differ slightly from the ones used in Table 3.2, the mole fraction of each chemical

TABLE 3.3. Flash2 Results

Temperature, °F	180	180	180
Pressure, psi	70	70	70
Vapor fraction	0.418979	1	0
Mole flow, lb mol/h	100	41.89795	58.10205
Mass flow, lb mol/h	7355.297	2587.239	4768.058
Volume flow, cubic ft/h	3865.909	3735.26	130.6492
Enthalpy, MMBtu/h	-6.767213	-2.262202	-4.505011
Mole flow, lb mol/h			
Propane	10	7.880996	2.119004
n -Butane	30	18.01565	11.98435
n -Pentane	40	15.09625	24.90375
n -Octane	20	0.90505	19.09495
Mole fraction			
Propane	0.1	0.1881	0.03647
n -Butane	0.3	0.429989	0.206264
n -Pentane	0.4	0.36031	0.428621
n -Octane	0.2	0.021601	0.328645

in the vapor and liquid streams is close. The K -values are also listed in the report (see Appendix C).

NONIDEAL LIQUIDS – TEST OF THERMODYNAMIC MODEL

In the examples given above, you were either given the K -values or you were told what thermodynamic model to choose. Usually, that decision is yours to make, and your results are only as good as your choice. Consequently, it is important to test your choice, preferably against experimental data. One easy way to do that is to model the T - xy or p - xy diagram for the vapor–liquid of binary pairs. You can do this easily using the Property Analysis option in Aspen Plus. Given here are commands that will enable you to create such diagrams. You may also need to review Appendix C, which has more detail about Aspen Plus.

Step 1 Start Aspen and choose Template. A window appears; choose General with Metric Units. In the Run Type (lower right-hand corner), choose Property Analysis. Click OK when the Aspen engine window appears. (This last step is specific to your installation.)

Step 2 In the list at the left, choose Component/Specifications and enter the names or formulas of the chemicals. If there is no list at the left, click on the eye glasses or choose the menu Data/Components. If Aspen Plus does not recognize your chemical, a window appears that allows you to search again, and it will suggest a number of possibilities. When the components are completely specified, you should have an entry for every chemical in the column labeled ‘Component name.’ The first column is what you are naming the chemicals, but the third column is what Aspen Plus uses when it gets physical properties. If that column is blank, the program will not work.

Step 3 In the list at the left, choose Property/Specifications. Set the property method as illustrated in Chapter 2 (see Figure 2.4). For this application, scroll down and choose WILS-2. This option uses an ideal gas, Henry’s law, and the Wilson binary parameters.

Step 4 You also need to tell the computer which database to use for the interaction parameter. Choose one of the options for interaction parameters. The results shown below use the Wilson-2 binary parameters. Now all the icons on the left should be blue, meaning the menus are completely filled out. Go back and complete any red menus.

Step 5 In the menus at the top choose Tools/Analysis/Property/Binary. (The binary option will be grayed out until you do steps two, three, and four.) In the window that appears, as illustrated in Figure 3.5, choose the following:

Components: select two components

Pressure: set the pressure for a T - xy diagram

or

Temperature: set the temperature for a p - xy diagram

Click Go.

Step 6 A graph will appear showing the result. You can read from this graph, or if you move the graph slightly, you will find a table behind it, giving tabular values. The

Binary Analysis

Analysis type:

Valid phases:

Components:
 Component 1:
 Component 2:

Compositions:
 Basis:
 Component:
☐ List of values
☐ Overall range
 Lower: Upper:
 Points: Increments:

Pressure:
 Units:
☒ List of values

☐ Overall range
 Lower: Upper:
 Points: Increments:

Property options:
 Property method:
 Henry components:
 Chemistry ID:
 Simulation approach:

Component ID: Vary composition of this component.

Figure 3.5. Data input for binary vapor-liquid diagram.

y - x -diagram for ideal thermodynamics is shown in Figure 3.6. This is not a good fit. You know that ethanol and water form an azeotrope, and the predictions do not give an azeotrope (where $y = x$). You should conclude that an ideal solution is not a good thermodynamic model for this system.

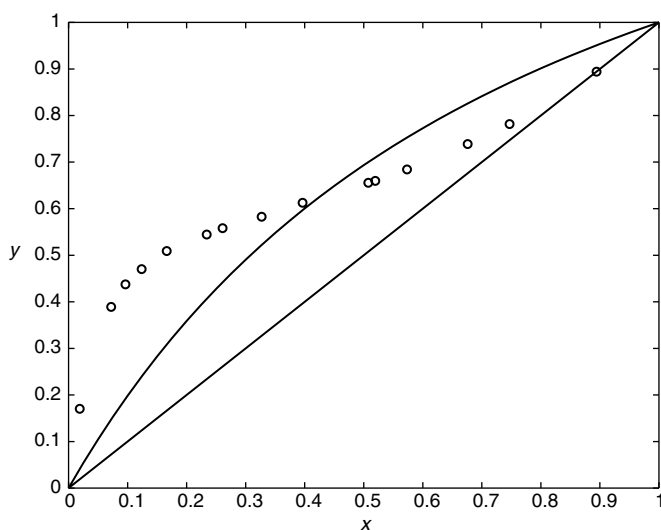


Figure 3.6. Ethanol-water vapor-liquid diagram for ideal thermodynamics.

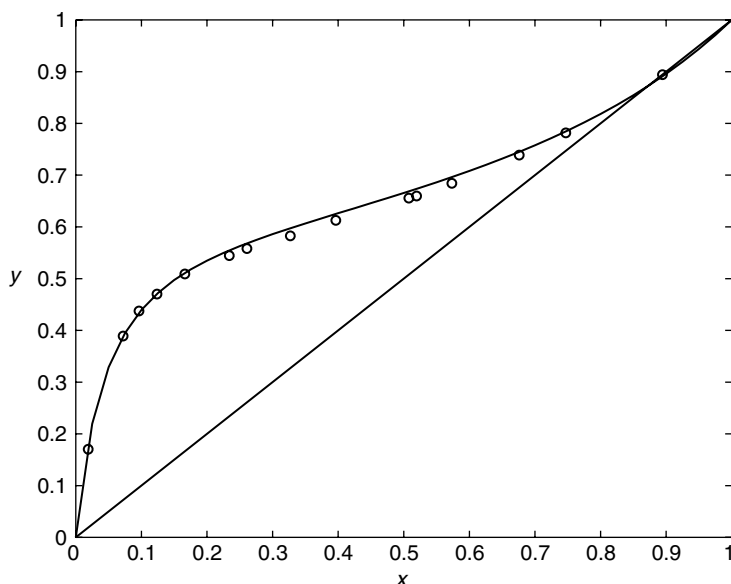


Figure 3.7. Ethanol–water vapor–liquid diagram with Wilson binary interaction parameters.

The results for the Wilson thermodynamics are shown in Figure 3.7, and the calculations agree with the data. The table (behind the graph in Aspen Plus) also lists K -values and activity coefficients. When you have good comparison with data, whether in the form of a T - xy diagram, K -values, or activity coefficients, you have validated a thermodynamics model for this binary pair. In a process with many chemicals, you may have to do this for all binary pairs before deciding on the best thermodynamic model.

CHAPTER SUMMARY

In this chapter, you have derived the equations governing phase equilibrium and seen how the key parameters can be estimated using thermodynamics. You have solved the resulting problems using Excel, MATLAB, and Aspen Plus. You also learned to prepare a T - xy diagram as a way of testing the thermodynamic model chosen to represent the phenomenon.

PROBLEMS

- 3.1.** The following stream is at 100 psia and 178°F. Calculate the fraction that is vapor by solving the Rachford–Rice equation: (1) using Excel; (2) using MATLAB.

Chemical	lb mol/h	K -value
Propane	20	3.7
<i>n</i> -Butane	30	1.4
<i>n</i> -Pentane	50	0.6

- 3.2.** (Aspen Plus) A reservoir in Northern Louisiana contains a volatile oil. The reservoir conditions at discovery were 246°F, 4800 psia. The composition was

Chemical	Mole Fraction
Nitrogen	0.0167
Methane	0.6051
Carbon dioxide	0.0218
Ethane	0.0752
Propane	0.0474
Butanes	0.0412
Pentanes	0.0297
Hexanes	0.0138
Heptanes plus	0.1491

You will have to choose how to model the butanes and pentanes: normal? Iso-? You will also have to choose which chemical to use to model the heptanes plus: heptane? octane? Use the RK-Soave option in Aspen Plus.

- (1) The gas-liquid separator at the surface is at 500 psia, 65°F. Find the composition of the gas and liquid streams and the vapor fraction.
 - (2) The liquid is taken to the stock tank, which is at 14.7 psia, 70°F. Find the vapor fraction at these conditions.
- 3.3.** The following stream comes out of a distillation tower (described in detail in Chapter 6). It is at 138 psia and 197.5°F. If the pressure is reduced (adiabatically) to 51 psia, what will be the vapor fraction and temperature? (Hint: In Aspen Plus, put a valve before the Flash2 unit, and reduce the pressure with the valve block.)

Chemical	lb mol/h
Propane	1.00
Isobutane	297.00
<i>n</i> -Butane	499.79
<i>i</i> -Pentane	400.00
<i>n</i> -Pentane	500.00

- 3.4.** The following stream comes out of a distillation tower (described in detail in Chapter 6). It is at 36 psia and 141.5°F. If the pressure is reduced (adiabatically) to 20 psia, what will be the vapor fraction and temperature? (Hint: In Aspen Plus, put a valve before the Flash2 unit, and reduce the pressure with the valve block.)

Chemical	lb mol/h
<i>n</i> -Butane	4.94
<i>i</i> -Pentane	396.00
<i>n</i> -Pentane	499.65

- 3.5. A stream coming out of an ammonia reactor is at 700°F and 3000 psia. The flow rates are given below.

Chemical	lb mol/h
Nitrogen	143.3
Hydrogen	450.4
Ammonia	221.6
Carbon dioxide	1.8

If the temperature is reduced to 80°F in a flash unit, keeping the pressure the same, determine the flow rate of each species in the vapor and the liquid stream.

- 3.6. The flow rates out of a methanol reactor are: CO, 100 lb mol/h; hydrogen, 200 lb mol/h; methanol, 100 lb mol/h. This stream is at 100 atm and 300°C. Cool it to 30°C and determine the flow rates of each species in the vapor and liquid stream.
- (1) What are the K -values for the three species at these conditions? A flash separation (Flash2 in Aspen) is adequate to model the condenser. Using the mole fractions y_i and x_i obtained by Aspen Plus for the vapor and liquid respectively, calculate the K -values, $K_i = y_i/x_i$.
 - (2) Do the simulation for an ideal solution and also for another choice of thermodynamic model (your choice).
 - (3) Your report should have a flowsheet and give the mass balances. Discuss the difference between the two cases and why you think one might be better. How can you find out which is more realistic? Note: The dipole moments of the three species are: CO, 0.1 debyes; hydrogen, 0 debyes; methanol, 1.7 debyes (Reid et al., 1977).
- 3.7. The flow rates out of an equilibrium reactor to make hydrogen were: CO, 5.25; hydrogen, 1095; carbon dioxide, 699; and water, 364.2 (use these as lb mol/h). Ideally, you could cool the stream and remove all the water. In actuality, you might not be able to condense all the water, and the water may contain trace amounts of the other components. This problem challenges you to find out how much. The outlet from the reactor is at 450 K, 1 atm, and is cooled to 80°F, still at 1 atm.
- (1) What are the K -values for the three species at these conditions? A flash separation (Flash2 in Aspen) is adequate to model the condenser. Using the mole fractions y_i and x_i obtained by Aspen Plus for the vapor and liquid, respectively, calculate the K -values, $K_i = y_i/x_i$.
 - (2) Do the simulation for an ideal solution and also for another choice of thermodynamic model (your choice).
 - (3) Your report should have a flowsheet and give the mass balances. Discuss the difference between the two cases and why you think one might be better. How can you find out which is more realistic?
- 3.8. Using the Rachford–Rice equation, Eq. (3.9), prove that for two phases to co-exist, at least one component needs a K -value greater than one and another component needs a K -value less than one.

- 3.9.** Model the following vapor-liquid equilibria data of ethyl acetate and water at 760 mmHg using the WilsonRK model in Aspen Plus (Ellis and Garbett, 1960):

x_1	y_1	T (°C)
0.0006	0.0405	98.85
0.0011	0.1256	96.20
0.0049	0.5910	86.50
0.0086	0.6680	76.20
0.0459	0.7140	70.55
0.1440	0.7025	70.55
0.2690	0.7060	70.50
0.3540	0.7090	70.45
0.4080	0.7090	70.55
0.5140	0.7100	70.50
0.6080	0.7025	70.50
0.6900	0.7070	70.45
0.7750	0.6990	70.50
0.8737	0.7650	71.45
0.9444	0.8650	73.35

- 3.10.** Model the following vapor-liquid equilibria data of ethyl alcohol and ethyl acetate using the WilsonRK model in Aspen Plus (Murti and Van Winkle, 1959):

x_1	y_1	T (°C)
0.1260	0.2146	73.82
0.1343	0.2146	73.78
0.2271	0.2960	73.04
0.3128	0.3634	72.50
0.3358	0.3643	72.28
0.5052	0.4803	72.18
0.5441	0.5074	72.35
0.6442	0.5618	72.70
0.6828	0.6092	72.90
0.7850	0.6819	74.14
0.8774	0.7908	75.50
0.9482	0.8924	76.70

- 3.11.** Model the following vapor-liquid equilibria data of acetaldehyde and ethyl alcohol using the WilsonRK model in Aspen Plus (De Leeuw, 1911):

x_1	y_1	T (°C)
0.0690	0.2410	30.10
0.2030	0.6530	23.30
0.3160	0.7950	15.90

4

CHEMICAL REACTION EQUILIBRIUM

Hydrogen can be made for fuel cells by using the water–gas shift reaction. In fact, hydrogen is made this way every day in oil refineries. The reaction is very fast, and the effluent from the reactor is close to being in chemical equilibrium. Ammonia is made by reacting hydrogen and nitrogen, and the effluent from an ammonia reactor is usually in chemical equilibrium. The ammonia is then used to make fertilizer, which helps increase food production for the world. Thus, chemical reaction equilibrium is important for both energy and food production.

This chapter shows how to solve problems involving chemical reaction equilibrium. The chemical reaction equilibrium gives the upper limit for the conversion, so knowing the equilibrium conversion is the first step in analyzing a process. The second question, what the rate of reaction is, can then be answered to decide the volume of the reactor. This second question, using kinetics, is treated in Chapter 8. Chemical reaction equilibrium leads to one or more nonlinear algebraic equations which must be solved simultaneously, and such problems are described in this chapter.

When you take some chemicals that can react and mix them in a vessel, the reaction can be represented as



If the reaction is reversible, the reverse reaction can also take place.



You can write this as



If the reaction rate is very fast, then the forward and reverse reactions proceed quickly, and the ultimate mixture contains all four chemicals, in specific amounts. If you change the temperature, these amounts may change. Sometimes, in a gaseous reaction changing the pressure has the same effect. In this chapter, your goal is to predict the composition of the final mixture, determined by chemical reaction equilibrium.

Of course, the reaction rate may not be infinitely high, and you may use a catalyst to speed up the reaction. However, even if the reaction rate is increased, you can never go beyond the composition determined by chemical reaction equilibrium. Here is how to set up that problem mathematically.

CHEMICAL EQUILIBRIUM EXPRESSION

In thermodynamics courses, you have learned that chemical reaction equilibrium is determined by the equilibrium constant, which is defined in terms of the change of Gibbs free energy.

$$\Delta G^o = -RT \ln K \quad (4.4)$$

The Gibbs free energy is tabulated at 298 K for pure components, and it is possible to extend the Gibbs free energy for a reaction to any temperature using the van't Hoff equation:

$$\frac{d \ln K}{dT} = \frac{\Delta H_R^o}{RT^2} \quad (4.5)$$

Here ΔH_R^o is the heat of reaction. When the reaction is



the equilibrium constant is defined in terms of the activities of the species:

$$K = \frac{a_C^c a_D^d}{a_A^a a_B^b} \quad (4.7)$$

In the gas phase, the activity is the fugacity, since the activity is the fugacity divided by the fugacity of the standard state, which is one atmosphere. In turn, you can write the fugacity as the product of the fugacity coefficient (providing a correction from ideal gas behavior) times the total pressure times the mole fraction in the vapor phase:

$$f_i = \phi_i p_i = \phi_i p y_i, \text{ where } \phi_i \equiv \frac{f_i}{p_i} \quad (4.8)$$

The fugacity coefficient can be calculated using the equation of state (Denbigh, 1971, p. 126):

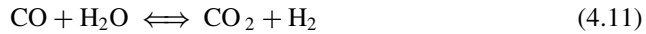
$$\ln \phi_i = \int_0^p \left(\frac{\hat{v}_i}{RT} - \frac{1}{p} \right) dp \quad (4.9)$$

When the pressure is less than 10 atm, the gas is usually ideal. From the formula it is clear that for an ideal gas the integral vanishes. Then $\ln \phi = 0$ or $\phi = 1$. You can use that assumption and combine all the terms in Eq. (4.7) into Eq. (4.10):

$$K = \frac{y_C^c y_D^d}{y_A^a y_B^b} p^{c+d-a-b} \quad (4.10)$$

Example of Hydrogen for Fuel Cells

As a specific example, consider the water–gas shift reaction that can be used in a chemical process to make hydrogen for fuel cell applications:



At equilibrium,

$$K = \frac{y_{\text{CO}_2} y_{\text{H}_2}}{y_{\text{CO}} y_{\text{H}_2\text{O}}} \quad (4.12)$$

Thermodynamic data give the value of $\ln K = 5$ (or $K = 148.4$) at 500 K. If you start with a stoichiometric mixture of carbon monoxide and water, what will the equilibrium composition be? In this case, the number of moles remains the same when the reaction takes place, and you are left with

$$148.4 = \frac{y_{\text{CO}_2} y_{\text{H}_2}}{y_{\text{CO}} y_{\text{H}_2\text{O}}} \quad (4.13)$$

Step 1 To begin, make a mole balance table (see Table 4.1), with a basis of 1 mol each of carbon monoxide and water, which then react to equilibrium. Use x to represent the number of moles reacting, thus giving the values in the table.

Step 2 Put the mole fractions into Eq. (4.13) and simplify:

$$148.4 = \frac{(x/2)(x/2)}{[(1-x)/2][(1-x)/2]} = \frac{x^2}{(1-x)^2} \quad (4.14)$$

You can easily solve this equation by taking the square root, giving $x = 0.924$. In more complicated cases it would not be possible to solve it so easily. Instead, you would

TABLE 4.1. Solution for Equilibrium of Water–Gas Shift Reaction

Species	Start	End	y_i
CO	1	$1 - x$	$(1 - x)/2$
H ₂ O	1	$1 - x$	$(1 - x)/2$
CO ₂		x	$x/2$
H ₂		x	$x/2$
Total	2	2	1

have to solve a nonlinear equation numerically to find the value of x . When there are multiple reactions, which may also be in equilibrium, there will be several equations to solve simultaneously. Thus, being able to solve multiple nonlinear equations is useful when dealing with reactions at equilibrium.

Solution Using Excel®

To solve for the equilibrium in Eq. (4.14) using Excel, you put the equation in one cell, put a guess of x in another cell, and use Goal Seek or Solver to make the first cell zero by changing x . Since the total number of moles does not change, you can change Eq. (4.13) to one involving moles:

$$148.4 = \frac{y_{\text{CO}_2} y_{\text{H}_2}}{y_{\text{CO}} y_{\text{H}_2\text{O}}} = \frac{(n_{\text{CO}_2}/n_t)(n_{\text{H}_2}/n_t)}{(n_{\text{CO}}/n_t)(n_{\text{H}_2\text{O}}/n_t)} = \frac{n_{\text{CO}_2} n_{\text{H}_2}}{n_{\text{CO}} n_{\text{H}_2\text{O}}} \quad (4.15)$$

The calculations are conveniently done in a spreadsheet, and the final results are shown in Table 4.2. Let us see how to construct this spreadsheet.

Step 1 Column B is the initial moles of each species.

Step 2 Column C is computed according to the equation displayed in Column D.

Step 3 Then the equilibrium equation, Eq. (4.15), is calculated in cell C9, according to the formula displayed in cell D9.

Step 4 Finally, you use Goal Seek to make cell C9 zero by varying cell C8. (The formulas listed in column D are displayed by entering them in a cell, preceded by the apostrophe symbol. See Appendix A for more information.)

Once you have prepared the spreadsheet, it is easy to change the conditions, either the equilibrium constant or the starting moles of various species. For another case, with some carbon dioxide and hydrogen initially, you get the results shown in Table 4.3. Notice that you can get more hydrogen by adding water to the initial composition. This is one way to help the hydrogen economy.

TABLE 4.2. Spreadsheet for Equilibrium of Water–Gas Shift Reaction

1	A	B	C	D	E
2	Species	Start	End		Mole Fraction
3	CO	1	0.075861	=B3 – \$C\$8	0.037931
4	H ₂ O	1	0.075861	=B4 – \$C\$8	0.037931
5	CO ₂	0	0.924139	=B5 + \$C\$8	0.462069
6	H ₂	0	0.924139	=B6 + \$C\$9	0.462069
7	Total	2	2.000000	=SUM(C3:C6)	1.000000
8		Reacting	0.924139		
9		Equilibrium equation	1.02887E-11 = 148.4 – C5*C6/(C3*C4)		

TABLE 4.3. Equilibrium of Water–Gas Shift Reaction with Nonstoichiometric Input

1	A	B	C	D	E
2	Species	Start	End		Mole Fraction
3	CO	1	0.011642	=B3 - \$C\$8	0.003638
4	H ₂ O	1.8	0.811642	=B4 - \$C\$8	0.253638
5	CO ₂	0.3	1.288358	=B5 + \$C\$8	0.402612
6	H ₂	0.1	1.088358	=B6 + \$C\$9	0.340112
7	Total	3.2	3.200000	=SUM(C3:C6)	1.000000
8		Reacting	0.988358		
9		Equilibrium equation	2.90498E-10 = 148.4 - C5*C6/(C3*C4)		

Solution Using MATLAB®

To solve for the equilibrium in Eq. (4.14) using MATLAB, you create an m-file that will calculate

$$f(x) = 148.4 - \frac{x^2}{(1-x)^2} \quad (4.16)$$

for any x . Then you use 'fsolve,' 'fzero' or 'fminsearch' to find the x that makes $f(x) = 0$.

Step 1 Construct an m-file that evaluates the function, given x . The name is 'equil_eq.m' and it is listed below. For testing purposes, you leave off the ';' of the commands that involve computation. Save this function in your desired workspace. (Note that the values for the 'in' parameters have been set arbitrarily for the test.)

```
% equil_eq
function y=equil_eq(x)
COin=1.1;
H2Oin=1.2;
CO2in=0.1;
H2in=0.2;
Kequil=148.4;
CO=COin-x
H2O=H2Oin-x
CO2=CO2in+x
H2=H2in+x
y=Kequil-CO2*H2/(CO*H2O)
```

This m-file acts like any function: Give it an x and it comes back with the value of y .

Step 2 To test this function, you evaluate it for a specific value of x . Issue the following command in the command window:

```
equil_eq(0.9)
```

The result is

```
CO=0.2000
H2O=0.3000
CO2=1
H2=1.1000
y=130.0667
ans=130.0667
```

This is the correct answer as determined by manual calculation.

Step 3 Once the check is made, you can add ';' to the end of every line. Note that to test the function you had to do three things correctly: (1) tell MATLAB the name of the function (equil_eq); (2) be sure the function was in the working directory (or some place MATLAB would look); and (3) provide a value of x .

Step 4 To run the problem shown in Table 4.1, you change the function to the following:

```
% equil_eq
function y=equil_eq(x)
COin=1.;
H2Oin=1.;
CO2in=0.;
H2in=0.;
Kequil=148.4;
CO=COin-x;
H2O=H2Oin-x;
CO2=CO2in+x;
H2=H2in+x;
y=Kequil-CO2*H2/(CO*H2O);
```

Step 5 You then issue the next two commands to find the answer:

```
format long
fzero('equil_eq',0.5)
ans=0.92413871189774
```

As expected, this is the same answer as you found using Excel. The initial concentrations can be changed, too. Because the check was for various values (not 0s and 1s), you have checked the code for any set of parameters. In finding the solution, MATLAB will access the function many times. The strategy for finding the solution is built into MATLAB, but you need to provide the function name and an initial guess. MATLAB does the rest (the hard part).

To solve the problem shown in Table 4.3, with different stoichiometry, you can use the same program. Here the program is changed slightly to illustrate the use of the global command.

```
% equil_eq_global
function y=equil_eq_global(x)
global COin H2Oin CO2in H2in Kequil
CO=COin-x;
H2O=H2Oin-x;
CO2=CO2in+x;
H2=H2in+x;
y=Kequil-CO2*H2/(CO*H2O);

% run equil_eq_global
global COin H2Oin CO2in H2in Kequil
COin=1
H2Oin=1
CO2in=0
H2in=0
Kequil=148.4
x=fzero('equil_eq_global',0.5)
```

Step 1 Here you use the global command in run_equil_eq_global, set the variables, and repeat the global command in the function equil_eq_global. Then those variables are accessible to the function (see Appendix B for additional information).

Step 2 Leave off the ‘;’ in the running program so the data is repeated in the output.

Step 3 When using the global command, you must be sure to use the global command in the command window (or a run_program that runs in the command window), set the variables after the global statement, and then repeat the global statement in the function. If you do not follow these rules, MATLAB will inform you that it was asked for an undefined variable or function, and give you the m-file and line number where it appears.

Step 4 When you run this program the results are the same as before, which checks your modification to use the global option. To solve the problem in Table 4.3, you change the input value in run_equil_eq_global to 1, 1.8, 0.3, and 0.1, respectively. The answer is

```
x=0.98835845820682
```

In this case MATLAB could not find a solution when starting from a guess of 0.5. After some experimentation, using a guess of 0.99 worked and gave the answer provided above.

CHEMICAL EQUILIBRIA WITH TWO OR MORE EQUATIONS

In most real cases, multiple reactions can take place, and several of them may be in equilibrium at the same time. Then you will have two equations like Eq. (4.14) that depend upon two variables; you must choose the variables to satisfy both equations. MATLAB is ideally suited to solving such problems.

MULTIPLE EQUATIONS, FEW UNKNOWNNS USING MATLAB

Suppose you want to solve the following two equations:

$$\begin{aligned} 10x + 3y^2 &= 3 \\ x^2 - \exp(y) &= 2 \end{aligned} \quad (4.17)$$

There are two main ways to solve multiple equations in MATLAB. The first is to use the 'fsolve' command. The second way is to use the optimization routines and the 'fminsearch' command. Both are illustrated here.

Method 1 Using the 'fsolve' Command

The steps are: create an m-file, check it, call the appropriate MATLAB function, and check the results.

Step 1 Create an m-file and call it prob2.m.

```
%filename prob2.m
function y2=prob2(p)
% vector components of p are transferred to x and y for
% convenience in remembering the equation
x=p(1)
y=p(2)
% the components of the two equations are calculated
y2(1) =10*x+3*y*y-3
y2(2)=x*x-exp(y)-2
```

Step 2 Test the file prob2.m with $p(1) = 1.5$ and $p(2) = 2.5$. If you tested with values of 1.0 and 1.0 there are several chances for error. For example, if a line is supposed to be $x*x$ but only x is written, then using a value of 1.0 for x gives the same value for x and $x*x$ and you will not detect the error. Using the same value for x and y [or $p(1)$ and $p(2)$] has the same pitfall: If you inadvertently typed x in place of y in the m-file, then you would not detect the error with these values.

```
>>p=[1.5 2.5]
>>feval('prob2',p)
y2=30.75 -11.9325
```

These values agree with the manual calculations, so the functions are computed correctly. Now you put ';' after every line in the code in order to suppress the displayed output.

Step 3 From the command mode, you call the 'fsolve' program with an initial guess, $p0$.

```
>>p0=[0 0]
>>z=fsolve('prob2',p0)
>>z=-1.4456 -2.4122
```

Step 4 Because you do not know how accurate the results are, you should check them by evaluating the function:

```
>>ans=feval('prob2',z)
ans=-1.0120e-04 5.1047e-06
```

Method 2 Using the 'fminsearch' Function

In MATLAB the 'fminsearch' function is used to find the minimum of a function of several variables. You first create an m-file that calculates the function, and then invoke the 'fminsearch' function to minimize it.

Step 1 To solve the same problem, create a slightly different m-file and call it 'prob3.m.' You can do this most easily by opening prob2.m, changing it, and saving it with the new name 'prob3.m.'

```
%filename prob3.m
function y2=prob3(p)
% vector components of p are transferred to x and y for
% convenience in remembering the equation
x=p(1)
y=p(2)
% the components of the two equations are calculated
f1 =10*x+3*y*y-3
f2=x*x-exp(y)-2
y2=sqrt(f1*f1+f2*f2)
```

Step 2 Check the function as you did before. In fact, because you already checked the m-file 'prob2.m' you only have to check that y2 is calculated correctly from f1 and f2. The command is invoked by first setting the initial guess (this also sets the length of the vector of independent variables, i.e. it tells MATLAB how many parameters it has to adjust):

```
>>p0=[1.5 2.5]
>>feval('prob3',p0)
f1=30.75
f2=-11.9325
ans=32.984
```

These values agree with the manual calculations, so the functions are computed correctly.

Step 3 Next, replace the ';' in the m-file and run the problem from an initial guess of [1 1].

```
>>p0=[1 1];
>>xvec=fminsearch('prob3',p0)
```

The results are

```
xvec=-1.4456 -2.4121
```

Step 4 You still do not know how accurate these results are. Check by evaluating the function after you have removed the ';' from the lines calculating f_1 and f_2 so that they will be displayed when the function is called:

```
>>ans=feval('prob3',xvec)
f1=-1.0120e-04
f2=5.1047e-06
```

Variations in MATLAB

Step 5 If these values are not small enough, you should reduce the tolerance using the options variable. The standard tolerance is 10^{-4} . To see the effect, though, you must obtain more significant digits by using the 'format long' command:

```
>>format long
>>options=optimset('TolFun',1e-12)
>>xvec=fminsearch('prob2',p0,options)
xvec=-1.44555236880465 -2.41215834803936
```

Step 6 To see how well the equations are solved, you take out the ';' from the file prob3.m and save it.

```
>>feval('prob3',xvec)
f1=1.4531e-12
f2=-9.0994e-13
ans=1.7145e-12
```

Thus, you have made the functions zero to at least 10 digits. (The numbers you obtain will not be identical to these, because the numbers are so small they are corrupted by round-off error.)

Step 7 To find information about other options, type

```
>>help fminsearch
```

which tells you to use the options Display, TolX, TolFun, MaxFunEvals, and MaxIter. Then

```
>>help optimset
```

tells you how to set those parameters.

TABLE 4.4. Equilibrium of Water–Gas Shift Reaction at 500 K and 1 atm; (a) Feed with Reactants Only, (b) Feed with Reactants and Products

(a)				(b)			
Temperature, °C	226.85	226.85	226.85	Temperature, °C	226.85	226.85	226.85
Pressure, bar	1.01325	1.01325	1.01325	Pressure, bar	1.01325	1.01325	1.01325
Vapor fraction	1	1	1	Vapor fraction	1	1	1
Mole ow, kmol/h	2	2	2	Mole ow, kmol/h	3.2	3.2	3.2
Mass ow, kg/h	46.02568	46.02568	46.02568	Mass ow, kg/h	73.84243	73.84243	73.84243
Volume ow, m ³ /h	82.05606	82.05606	82.05606	Volume ow, m ³ /h	131.2897	131.2897	131.2897
Enthalpy, MMkcal/h	2 0.081087	2 0.089842	2 0.089842	Enthalpy, MMkcal/h	2 0.153428	2 0.153428	2 0.16281
Mole ow, kmol/h				Mole ow, kmol/h			
CO	1	0.078567	0.078567	CO	1	0.012528	0.012528
H ₂ O	1	0.078567	0.078567	H ₂ O	1.8	0.812528	0.812528
CO ₂	0	0.921433	0.921433	CO ₂	0.3	1.287472	1.287472
H ₂	0	0.921433	0.921433	H ₂	0.1	1.087472	1.087472
Mole fraction				Mole fraction			
CO	0.5	0.039284	0.039284	CO	0.3125	0.003915	0.003915
H ₂ O	0.5	0.039284	0.039284	H ₂ O	0.5625	0.253915	0.253915
CO ₂	0	0.460716	0.460716	CO ₂	0.09375	0.402335	0.402335
H ₂	0	0.460716	0.460716	H ₂	0.03125	0.339835	0.339835

CHEMICAL EQUILIBRIA USING ASPEN PLUS®

You can also use the process simulator Aspen Plus to solve chemical reaction equilibrium problems. It has a huge advantage over Excel and MATLAB: Aspen Plus contains the Gibbs free energies of many chemicals, and it can calculate them as a function of temperature. Thus, the data-gathering aspect of the problem is handled for you. Your job is to compare the results and the predicted K -values with experimental information.

The reactor module with Gibbs free energy is RGibbs. You can set up the problem in the same fashion as shown in Chapter 3 (p. 31), except that you choose Reactors, and then choose the RGibbs reactor. In the example shown in Table 4.4, the NRTL model is chosen for property evaluation. By using the mole fractions, you can find the equilibrium constant used, which is 137.5. This is slightly different from the one used in Eq. (4.14), which makes a slight change in the answer. For the problem illustrated in Table 4.1, Aspen Plus obtained a CO molar flow rate of 0.076 vs 0.079 using an equilibrium constant of 148.4. For the problem in Table 4.3 Aspen Plus obtained a CO molar flow rate of 0.013 vs 0.012 and a hydrogen flow rate of 1.087 vs 1.088. The chief difference between the calculations is due to the K -value. Aspen Plus predicts that value from the Gibbs free energy, while the value used in Eq. (4.14) was obtained by reading it off a graph.

CHAPTER SUMMARY

In this chapter, you have derived the equations governing chemical reaction equilibrium and seen how the key parameters can be estimated using thermodynamics. You have solved the resulting problems using Excel, MATLAB, and Aspen Plus. You also learned to solve multiple equations using MATLAB when there are several reactions in equilibrium.

PROBLEMS

- 4.1.** Methanol is formed by reacting carbon monoxide with hydrogen. (This might be an on-board generation of hydrogen for the hydrogen economy.) With the reaction



the condition at equilibrium is

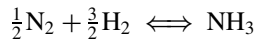
$$K_p = \frac{p_{\text{CH}_3\text{OH}}}{p_{\text{CO}} p_{\text{H}_2}^2} = \frac{y_{\text{CH}_3\text{OH}}}{y_{\text{CO}} y_{\text{H}_2}^2} \frac{1}{p^2} \quad (4.19)$$

When the carbon monoxide and hydrogen enter the reactor in a 1:2 ratio, find the equilibrium conversion when the pressure is 50 atm and $K_p = 0.0016$. First, use Excel; then use MATLAB to solve the problem.

- 4.2. Find the molar flow rates of all species out of an equilibrium reactor when the inlet values of nitrogen, hydrogen, and ammonia are 1.1, 3, and 0.2. The equilibrium constant is 0.05 at 589 K

$$K_p = \frac{P_{\text{NH}_3}}{P_{\text{N}_2}^{1/2} P_{\text{H}_2}^{3/2}} = \frac{y_{\text{NH}_3}}{y_{\text{N}_2}^{1/2} y_{\text{H}_2}^{3/2}} \frac{1}{P} \quad (4.20)$$

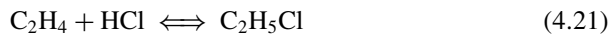
for the reaction written as



Use 220 atm for the pressure. First, use Excel; then use MATLAB.

- 4.3. A process to make hydrogen uses gas from a reformer; the gas is fed to a reactor where the water–gas shift reaction takes place [Eq. (4.11)]. Under usual operating conditions, the stream leaving the reactor is in equilibrium at the outlet temperature (or close to it). The feed rate to this part of the process is (in lb mol/h), at 177°C and 1.36 atm: CO, 1260; H₂, 932; CO₂, 140; steam, 1972. The stream goes to a reactor, and the output of the reactor is in chemical equilibrium at 377°C. The equilibrium constant at 377°C is 15.6. Find the equilibrium composition leaving the reactor using (1) Excel; (2) MATLAB; and (3) Aspen Plus with the RGibbs block.

- 4.4. Ethyl chloride is made by reacting ethylene with hydrogen chloride:



Start with an equimolar amount of ethylene and hydrogen chloride and determine the equilibrium composition at 350°F and 250 psig using Aspen Plus.

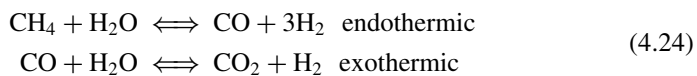
- 4.5. Find the equilibrium conversion for a mixture of sulfur dioxide, oxygen, and sulfur trioxide at 700 K and 2 atm:
- (1) Starting mixture of SO₂ : O₂, 1 : 3, using ideal thermodynamics;
 - (2) Change to NRTL thermodynamics;
 - (3) Increase the temperature to 900 K;
 - (4) Use a starting mixture of SO₂ : O₂, 1 : 1/2, using NRTL at 900 K;
 - (5) Discuss the similarities and differences of (1)–(4).
- 4.6. Ethylene is made by heating ethane at a high temperature. (1) Determine the equilibrium composition of ethane, ethylene, and hydrogen at 1100 K and 1 atm using Aspen Plus:



- (2) Add in a second reaction, increase the temperature to 1367 K, and determine the equilibrium composition:



- 4.7. In a steam reformer, steam is added to methane and two (or more) reactions take place:



The composition out of the reformer is based mainly on chemical equilibrium (Marsh et al., 1994, p. 165). Determine the equilibrium composition under the following conditions using Aspen Plus. (1) 750°C, 14 bar, steam/hydrogen = 3; (2) 650°C, 14 bar, steam/hydrogen = 2.

5

MASS BALANCES WITH RECYCLE STREAMS

Engineers who design and run chemical plants use mass balances to decide which processes are cost-effective. Chemical engineering students are taught in their first course to look at the entire system as well as the individual pieces of equipment. If you know the mass flows of a chemical process, including the input flows and output flows, you can estimate the economic viability of that process by adding up the selling price of the products, and subtracting the cost of the raw materials and waste treatment. If that number is not positive, the process is not cost-effective. You want to find that out before you have spent too much time designing it. Then you can try to make it economically viable, perhaps by using another chemical reaction pathway, or a new separation scheme for purifying the product. Often, the raw material cost is 70 percent of the cost of the product, which makes the mass balance very useful. Also, mass balances allow you to find the energy costs in the process, provided you can calculate the enthalpy of each stream. Since thermodynamics allows you to calculate the enthalpy, you can estimate the energy cost.

Most processes involve a recycle stream. The reason is that all the reactants do not react, and businesses cannot afford to throw the rest away. Furthermore, any leftovers have to be disposed of in an environmentally friendly manner, which costs money. Thus, engineers take the unreacted reactants and put them back in the start of the process and try again. This makes the mass balances a little more complicated, and it leads to iterative methods of solution, which are described in this chapter. The first part of this chapter uses Excel[®] to solve mass balances with recycle streams. Situations in which the energy balances affect the mass balance are treated in Chapters 6 and 7, because these are best done using a process simulator such as Aspen Plus[®].

MATHEMATICAL FORMULATION

The mathematical formulation of mass balances is really quite easy. This section develops the equations for three simple units, which you then combine in myriad ways. Once you can do the mass balance for a simple unit, with the help of Excel you can do the mass balance for a process combining those simple units. Equations are given here for a mixer, separator, and reactor, and then those equations are used in Excel to perform mass balances on processes.

A mixer (called MIXR) adds all the flows in and transfers them to the output stream, as illustrated in Figure 5.1. The mass (or mole) balance is that the mass out equals the sum of the masses in. This is true for each chemical component, and hence for the total mass, too. It is more common to make mole balances, since chemical reactors occur using moles, but the conversion from mass to moles and back again is easy. The mass balance equations for the MIXR unit are:

$$n_i^{\text{out}} = \sum_{j=1}^{NSTRM} n_i^j \quad (5.1)$$

where n_i^j is the flow rate of component i in the j th stream, and n_i^{out} is the flow rate of component i in the outlet stream. If you know the flow rates in, you can easily calculate the flow rates out.

The second unit is a separator, called SEPR. There is one input streams and two (or more) output streams, as illustrated in Figure 5.2. The split fraction sf_i is the fraction of component i going to the overhead stream. The mass balance for each component i is:

$$n_i^2 = n_i^{\text{in}} sf_i, \quad n_i^1 = n_i^{\text{in}} (1 - sf_i) \quad (5.2)$$

If you know the split fractions of each component, and the flow rate in, you can easily calculate the flow rates out of stream 1 and 2 for each component.

The final unit is a reactor, called REAC, illustrated in Figure 5.3. This time there is only one input stream and one output stream, but a reaction occurs. Naturally, you need to

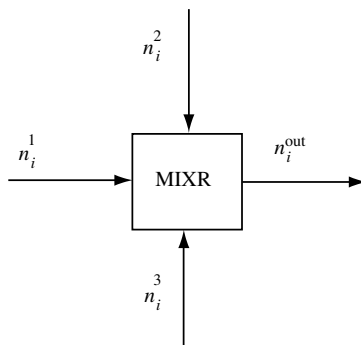


Figure 5.1. MIXR unit.

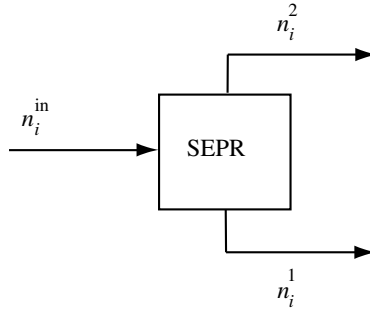


Figure 5.2. SEPR unit.

specify the reaction and the conversion in some way. You can specify the reaction using the stoichiometric equation for the reaction.



If N moles of A react, then $(b/a)N$ moles of B also react and form $(c/a)N$ moles of C and $(d/a)N$ moles of D. You specify the amount reacting by giving a conversion for A (i.e., 50 percent of the A reacts). The conversion is called α . You do have to be careful to specify the conversion based on the limiting reagent, which in the illustration is A. If there were not enough B to react with A, then the conversion of A would be limited. If it were too high, it would give a negative concentration of B. The equation for each of the components is

$$\begin{aligned} n_A^{\text{out}} &= n_A^{\text{in}} - n_A^{\text{in}} \alpha, & n_B^{\text{out}} &= n_B^{\text{in}} - \frac{b}{a} n_A^{\text{in}} \alpha \\ n_C^{\text{out}} &= n_C^{\text{in}} + \frac{c}{a} n_A^{\text{in}} \alpha, & n_D^{\text{out}} &= n_D^{\text{in}} + \frac{d}{a} n_A^{\text{in}} \alpha \end{aligned} \quad (5.4)$$

Note in particular that the equations for components B, C, and D depend upon the amount of A entering, since that is how the conversion is specified.

All these equations can be programmed easily in Excel. It is the combination of them to represent the mass balances in a process with recycle that becomes challenging. Here are some examples in which the individual units are combined in an Excel spreadsheet to make mass balances for an entire (but simple) process.

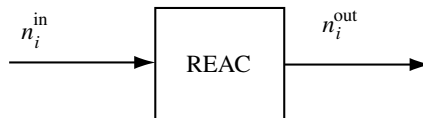


Figure 5.3. REAC unit.

EXAMPLE WITHOUT RECYCLE

A stream in a refinery is at 100 psia and 75°F and contains the following:

	lb mol/h
Propane	100
<i>i</i> -Butane	300
<i>n</i> -Butane	500
<i>i</i> -Pentane	400
<i>n</i> -Pentane	500
Total	1800

Your task is to separate this stream into five streams, each of which is a relatively pure stream of one component.

Step 1 You need to decide what physical mechanism to use (distillation, absorption, membranes, etc.) and the operating parameters of the equipment. For the first example, you can use the SEPR unit above as a simplistic model of any of these units. The chemicals are listed in order of their boiling points with the lowest boiling points first. Thus, choose distillation as the preferred separation method and the distillation train (see Figure 5.4) will distill them off one by one.

Step 2 You can simplify this by taking out all the energy exchange units, as shown in Figure 5.5.

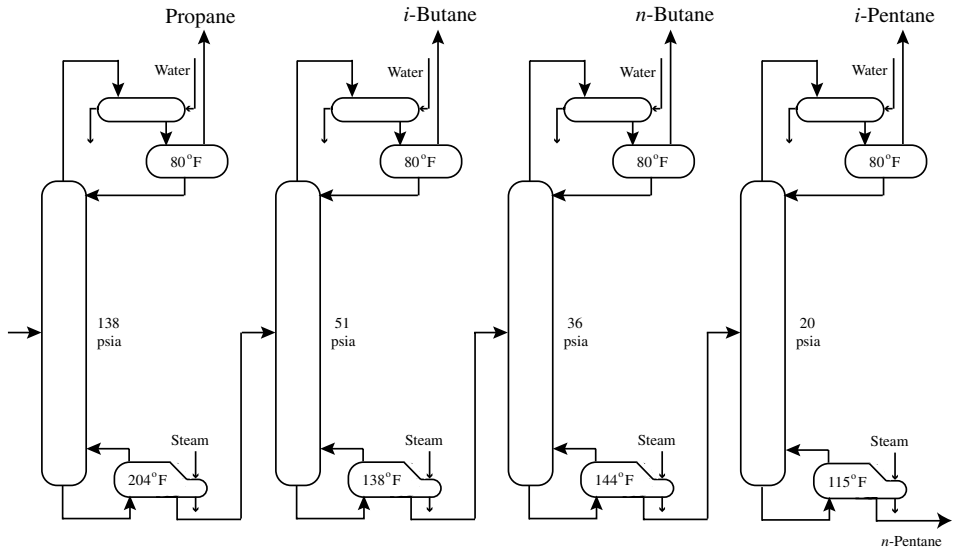


Figure 5.4. Distillation train.

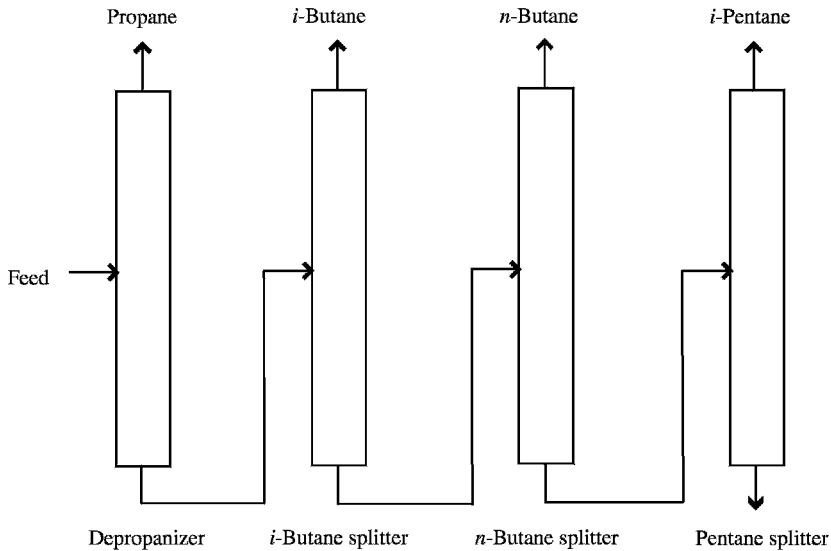


Figure 5.5. Simplistic distillation train.

Step 3 Next, you prepare an even simpler flow sheet, using specified split fractions, as shown in Figure 5.6. In Chapters 6 and 7 you will learn to estimate those split fractions and to model the distillation towers more exactly.

Each unit, SEPR-*x*, involves a simple mass balance in which you specify the fraction of the feed that you want to go out of the overhead and the bottom streams. You do this for each species. The calculation proceeds from left to right, and the results are shown in Table 5.1.

Step 4 No iteration is necessary in this example. The final step is to check the results. You can add up the input and output streams for each component in each unit and determine that the mass of each species is conserved. You can also check to see that you used the appropriate split fractions. Then, and only then, should you accept the results. This was a very simple example, so easy that you could have done it on the back of an envelope. However, if the simple distillation units were replaced by real distillation columns, in which vapor–liquid data were used, the calculations would have been much more difficult and time-consuming.

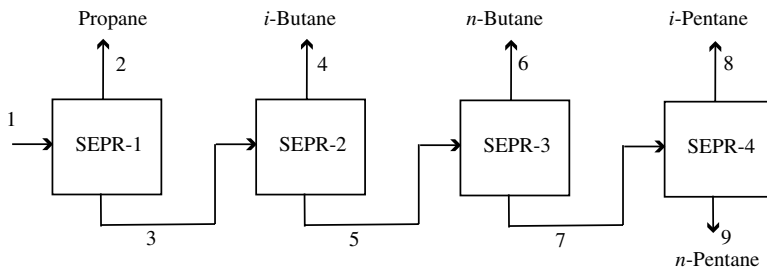


Figure 5.6. Distillation train modeled using simple SEPR units.

TABLE 5.1. Recovery Fractions in Distillation Train

	Stream No.								
	1	2	3	4	5	6	7	8	9
Split fraction		SEPR-1		SEPR-2		SEPR-3		SEPR-4	
C3		0.99		1		1		1	
<i>i</i> -C4		0.005		0.995		1		1	
<i>n</i> -C4		0		0.005		0.995		1	
<i>i</i> -C5		0		0		0.005		0.995	
<i>n</i> -C5		0		0		0		0.005	
lb mol/h									
C3	100	99	1	1	0	0	0	0	0
<i>i</i> -C4	300	1.5	298.5	297	1.5	1.5	0	0	0
<i>n</i> -C4	500	0	500	2.5	497.5	495	2.5	2.5	0
<i>i</i> -C5	400	0	400	0	400.0	2	398	396	2
<i>n</i> -C5	500	0	500	0	500.0	0	500	2.5	497.5

EXAMPLE WITH RECYCLE; COMPARISON OF SEQUENTIAL AND SIMULTANEOUS SOLUTION METHODS

Consider a simple process in which the reactor can only convert 40 percent of the feed to it (perhaps due to equilibrium constraints), but the separation of reactant A and product B is complete and the unreacted A is recycled. The process is illustrated in Figure 5.7.

Step 1 One way to solve this problem is to do a sequential solution. You start with the feed, and solve the mass balance for the mixer, labeled MIXR. Since you do not yet know the amount of A in stream S5, assume it is zero and go on. Then there is one mole of A fed to the reactor, 40 percent of it reacts, and the unreacted part is recycled into stream S5.

Step 2 Now repeat the process. This time, when doing a mass balance on the mixer, you have a value in stream S5, so you can use it. You continue this process, working through the process over and over. After 20 iterations, the value of A in the recycle when you start an iteration (1.4999086) is close to the value when you end the iteration (1.4999452). You can then stop. The solution is shown in Table 5.2; note that some of the rows have been hidden (see Appendix A).

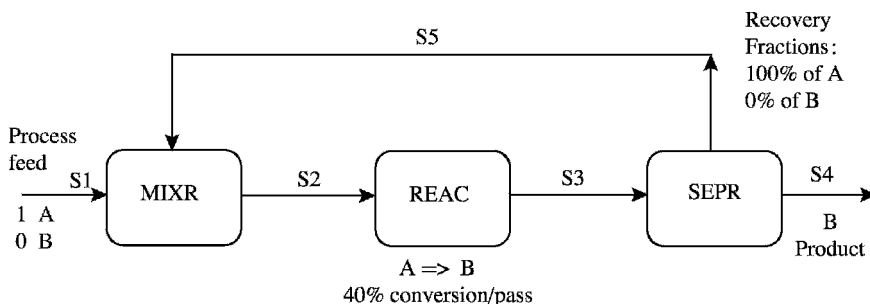
**Figure 5.7.** Process with recycle and 40 percent conversion per pass.

TABLE 5.2. Mass Balance with Recycle, Solved using Sequential Method

Iteration Number	Feed	Recycle (guess)	Into Reactor	Out of Reactor	Recycle (at end of iteration)
1	1	0	1	0.6	0.6
2	1	0.6	1.6	0.96	0.96
3	1	0.96	1.96	1.176	1.176
4	1	1.176	2.176	1.3056	1.3056
5	1	1.3056	2.3056	1.38336	1.38336
10	1	1.4848835	2.4848835	1.4909301	1.4909301
15	1	1.4988245	2.4988245	1.4992947	1.4992947
20	1	1.4999086	2.4999086	1.4999452	1.4999452

Many process simulators use this procedure because it is quite easy to implement. You only need to have a module or subroutine for a mixer, a reactor, and a separator, and the equations for each of these are quite simple. One problem that arises, though, is that the procedure takes many iterations to converge if the conversion is low. And, if there are interlocking recycle streams, convergence may not occur at all. However, it is quite a good scheme, and you can apply it using a spreadsheet.

The other solution technique finds an algebraic solution. Look at Table 5.3. Assign a variable name, R , to the unknown amount of A in stream S5. Work through the same process once; when you finish, the amount of A in the recycle (which you called R) must be equal to the amount of A coming out of the last unit and going into the recycle stream. The equation representing that is

$$R = 0.6^*(1 + R) \quad (5.5)$$

Solving Eq. (5.5) gives $R = 1.5$.

The solution is the same as that found with the sequential method, but you obtained it in one iteration. The only problem is that you had to do it analytically, which is not usually possible. However, you can write the mass balance equations in a general form and create a large set of linear or nonlinear equations to be solved simultaneously on the computer. The difficulty lies in the complexity of the units. The mixer, simple reactor, and simple separator lead to simple equations, but heat exchangers, distillation units, and plug flow reactors do not. In return for your work, though, you obtain enhanced convergence properties, since the simultaneous method will converge when the sequential method does not. In a later chapter you can examine some of the tricks chemical engineers use to make the sequential method work more often. First, let us learn to solve problems with recycle using Excel.

TABLE 5.3. Mass Balance with Recycle, Solved using Simultaneous Method

Feed	Recycle (guess)	Into Reactor	Out of Reactor	Recycle (at end of iteration)
1	R	$1 + R$	$0.6^*(1 + R)$	$0.6^*(1 + R)$

EXAMPLE OF PROCESS SIMULATION USING EXCEL FOR SIMPLE MASS BALANCES

The next example illustrates the use of Excel to solve mass balances for a process consisting of a feed stream, a mixer in which the feed stream is mixed with the recycle stream, and a reactor, followed by a separator where the product is removed and the reactants are recycled. In later examples and problems, there will be inerts, purge streams, and so on, but this problem uses a stoichiometric feed. The reactor is limited by chemical equilibrium considerations, which complicates the solution.

The process takes hydrogen and nitrogen (in a 3:1 ratio) to make ammonia. The reactor is limited by equilibrium; you will prepare the spreadsheet in stages to aid troubleshooting. Thus, first prepare a spreadsheet as shown in Figure 5.8, using 25 percent conversion per pass in the reactor.

Step 1 The mixer takes streams 1 and 6, adds them, and puts them into stream 2.

Step 2 Stream 3 is not really a stream but is the moles reacting so the reaction stoichiometry is clearly displayed.

Step 3 Cell E14 is set to the negative (for reactants) of nitrogen in stream 2 times the conversion (in cell E19).

Step 4 Cell E15 is set to three times cell E14 (hydrogen uses three times the nitrogen reacting), and cell E16 is set to twice the nitrogen in cell E14. In this column, the signs are negative for reactants and positive for products. The spreadsheet clearly shows that the stoichiometry is correct.

Step 5 Stream 4 is the sum of streams 2 and 3.

Step 6 Stream 5 takes 98 percent of the ammonia and 0.5 percent of the nitrogen and hydrogen.

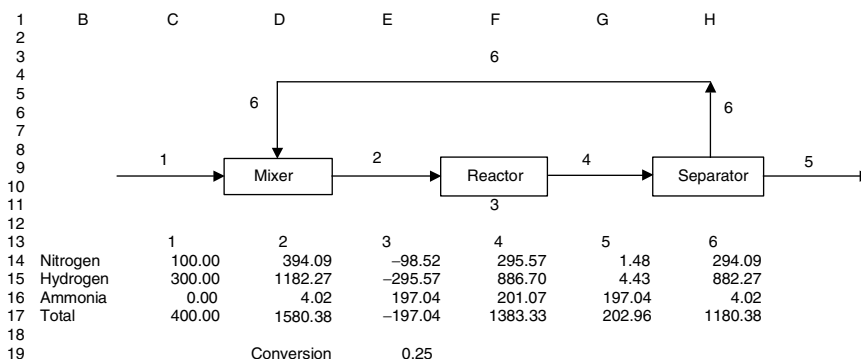


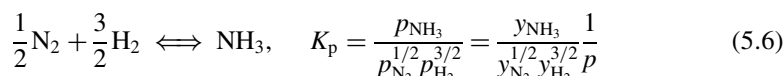
Figure 5.8. Ammonia process with 25 percent conversion per pass in the reactor.

Step 7 The rest goes out in stream 6. You can easily check the mole balance around the separator.

Step 8 Turn on the iteration feature for the circular reference and the problem is solved.

EXAMPLE OF PROCESS SIMULATION WITH EXCEL INCLUDING CHEMICAL REACTION EQUILIBRIUM

Next the reactor conversion is changed to be the equilibrium conversion, which may not be 25 percent. The equilibrium equation is



The spreadsheet for this case is shown in Figure 5.9. The value of K_p and p are put into cells F26 and F27, and the mole fractions of the stream coming out of the reactor are obtained by taking the moles of a species in stream 4 and dividing by the total moles in stream 4. Those mole fractions can be used directly in the chemical equilibrium equation, as shown in cell G28. This is simpler than deriving a chemical equilibrium equation involving the molar flow rates. Notice that, as the conversion changes, the mole balance changes. This in turn changes the mole fractions, which no longer satisfy the chemical equilibria equation. Thus, you should use Solver to set F28 to zero by changing cell E19, where the final result is shown. (Note: Goal Seek did not work for this problem.)* The only remaining check is to take the final mole fractions and compute the chemical equilibria equation yourself. If it is satisfied, then you have the correct solution.

EXAMPLE OF PROCESS SIMULATION WITH EXCEL INCLUDING PHASE EQUILIBRIUM

The next modification you will try allows phase equilibrium in the separator where most of the ammonia is condensed. For simplicity, though, you should set the conversion per pass in the reactor to 25 percent. Change the separator so that it is a phase separation. The K -values are: nitrogen, 4.8; hydrogen, 70; ammonia, 0.051; carbon dioxide, 0.32. (Remember that the symbol K value is used in both chemical reaction equilibria and phase equilibria, but K represents different things in those two cases.) The flow rates of nitrogen and hydrogen into the process are 1 and 3 mol per time unit, respectively, but there is also 0.01 mol per time unit of carbon dioxide. Because of the carbon dioxide,

*When using Excel, sometimes it converges, and sometimes it does not. A simple method to avoid difficulties is to save the spreadsheet as a backup before turning on iteration. If it does not converge, it might leave entries that are indefinite or huge. You can then open the saved spreadsheet. Alternatively, when Solver or Goal Seek are being used, it asks whether you wish to revert to the initial guess; you would choose yes since you cannot begin a spreadsheet calculation with indefinite numbers in cells. If you are using the spreadsheet with iteration, you do not have that choice. If you have to start again with indefinite entries, turn off the iteration feature, go to a cell that has an indefinite entry, cut the equation in the menu bar at the top and paste it right back. The number zero will appear in the cell. You have to do this for all cells with indefinite entries. Then turn on the iteration and try again.

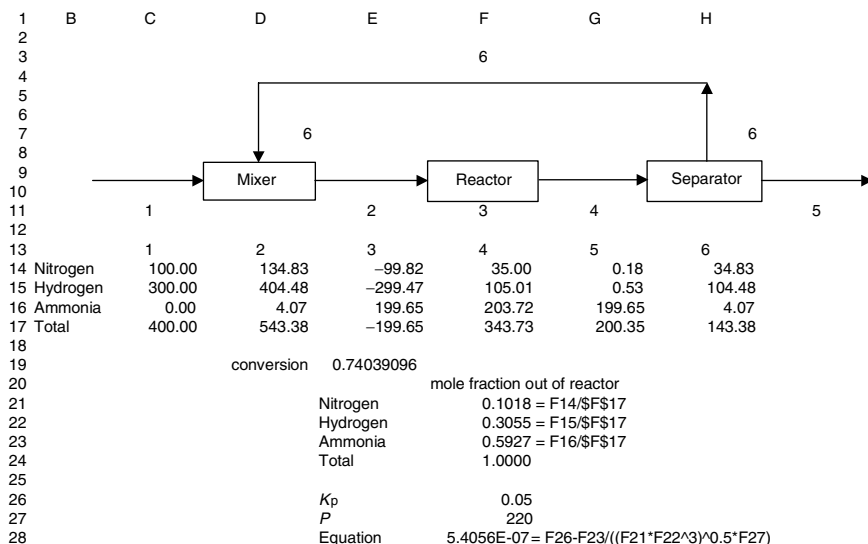


Figure 5.9. Ammonia Process with equilibrium conversion in the reactor.

you will add a purge stream as 1 percent of the recycle stream. Follow the structure of the problem as shown in Figure 5.10.

Now the separator must satisfy the Rachford–Rice equation, Eq. (3.9):

$$\sum_{i=1}^{NCOMP} \frac{(K_i - 1)z_i}{1 + (K_i - 1)v'} = 0 \quad (5.7)$$

Once the value of v' is found that satisfies this equation, the values of mole fraction and molar flow rates are given by Eqs. (3.1)–(3.8):

$$x_i = \frac{z_i}{1 + (K_i - 1)v'}, \quad y_i = K_i x_i \quad (5.8)$$

$$n_i^v = Vy_i = Fv'y_i \quad (5.9)$$

$$n_i^L = Ly_i = F(1 - v')x_i \quad (5.10)$$

In Eqs. (5.8)–(5.10) z_i is the mole fraction of the i th species into the flash unit, K_i is the K -value for the i th species (often the vapor pressure divided by the total pressure), and v' is the fraction of the feed that goes out as vapor (between 0 and 1).

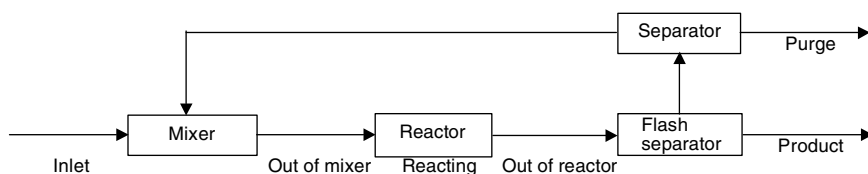


Figure 5.10. Ammonia process with vapor–liquid equilibria and a purge stream.

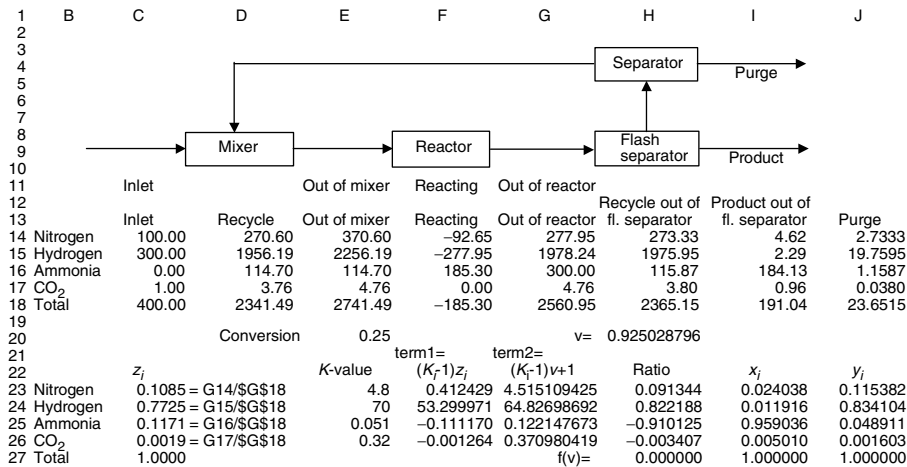


Figure 5.11. Mass balance for ammonia process with vapor-liquid equilibria.

Let us use nitrogen to explain the spreadsheet (Fig. 5.11).

Step 1 The recycle stream (cell D14) is obtained by subtracting the purge from the product out of the separator. Set cell D14 to: = H14 - J14.

Step 2 The stream out of the mixer (cell E14) is the sum of the inlet stream and recycle. Set cell E14 to: = C14 + D14.

Step 3 The amount reacting is the conversion times E14. Set cell F14 to: = E14 * E20.

Step 4 The stream out of the reactor is what came in minus what reacted. Set cell G14 to: = E14 + F14.

Step 5 The stream as recycle out of the separator uses Eq. (5.9). Set cell H14 to: = \$H\$20 * \$G\$18 * J23.

Step 6 The product stream is the difference. Set cell I14 to: = G14 - H14.

Step 7 The purge stream is 1 percent of the stream. Set cell J14 to: = 0.01 * H14. These formulae are copied down, except for column F, in which the amount of hydrogen reacting and ammonia formed are determined as multiples of the amount of nitrogen reacting. The total row (18) is the sum of the flow rates of the components.

Next, consider how the vapor-liquid equilibrium is solved.

Step 8 The mole fractions going into the flash unit are the ones coming out of the reactor. For example, cell C23 uses = G14/\$G\$18.

Step 9 Term1 and term2 are calculated as shown, and the ratio is calculated. The sum, cell H27, should be zero.

Step 10 The liquid mole fractions, cells I23–I26, and vapor mole fractions, cells J23–J26, are calculated using Eq. (5.8). They are summed to check that they add up to 1.0. They will add up to 1.0 only when the proper vapor fraction is found, that is, $f(v') = 0$, in cell H27.

Step 11 You can use Solver to set cell H27 to zero by varying cell H20, the vapor fraction. As Excel changes v' , the values in rows 23–27 change, and these in turn cause changes in rows 14–18 because they depend upon the vapor mole fractions. Unfortunately, Solver cannot find a solution, but you can vary v' yourself to find the solution.

Did Iterations Converge?

In this example, you have two iterations – one because of the circular reference due to the recycle streams and one because of the nonlinear Rachford–Rice equation. Some computer programs cannot handle both of these complications together. Neither Goal Seek nor Solver worked for this example, and you iterated the vapor fraction by hand.

Step 1 In effect you used the spreadsheet to do the molar balances, and you set v' yourself. The spreadsheet shows the value of $f(v')$ for your choice of v' , but it does the molar balance anyway.

Step 2 If $f(v')$ is not zero, you can change v' and do it again.

Repeat You keep this up until $f(v')$ is small enough to satisfy you. Once you are close, Solver should have no problem converging to a tight tolerance. What you are doing is replacing a problem with two iteration loops with a problem in which you supply one of the numbers, and the computer solves the other iteration loop. Then you change your number until the other equation is satisfied.

Extensions

Ideally you would have calculated the equilibrium in the reactor, too. Then you would have three interacting iterations, and it would be the rare problem that Excel could solve. The difficulty is that, during the iterations, the values may be physically unrealistic. Then, the equilibrium relation or the Rachford–Rice equation gives even more unrealistic values. Programs such as Aspen Plus can realize this and take precautionary steps to avoid it. As the flow sheet gets more and more complicated, and involves more and more thermodynamics, the power of Aspen Plus is welcome. See Chapters 6 and 7 for examples.

CHAPTER SUMMARY

In this chapter you learned how to write the mass balance equations for simple units and then combine them into a complicated process topology. You saw that there were two methods to solve these equations, and Excel used essentially the sequential method

because of the circular reference. You learned to handle several complications: recycle streams, chemical reaction equilibrium, phase equilibrium, and purge streams.

CLASS EXERCISES

Divide into teams of two to four students. Work through a spreadsheet (Tables 5.1 and 5.2, and Figs. 5.8, 5.9, and 5.11), with each person of a team taking a unit, showing where the parameters are for that unit and checking the results. Are they correct? Work through the whole process. Alternatively, your instructor can prepare a flowsheet and dataset showing the results, but with some errors for your team to find.

CLASS DISCUSSION

What happened to the mass balances when you introduced a purge stream? (You can run it without carbon dioxide, too.) What happened to the mass balances when vapor–liquid equilibrium was required? Did the ratio of nitrogen to hydrogen in the recycle stream change? Why or why not? What if you had to solve the Rachford–Rice equation in the separator, the chemical equilibrium equation in the reactor, and set the purge fraction to maintain a maximum mole fraction of carbon dioxide in the inlet to the reactor. Could you do that all in Excel? Would it converge? Speculate.

PROBLEMS

- 5.1.** The process for making benzene is described below and illustrated in Figure 5.12, and you are to make mass balances of it using Excel. (*Hydrocarbon Processing*, 1975; 54, p. 115; Otani et al., 1968.)

The process converts toluene to benzene and xylenes by disproportionation (also called transalkylation). The xylenes also disassociate to form toluene and trimethylbenzene. These reactions occur with a catalyst in the presence of hydrogen, with a typical ratio of 7:1 hydrogen to toluene feed to the reactor. The hydrogen also reacts with toluene to form benzene and methane or with xylenes to form benzene and methane. The molar consumption of hydrogen is about 10 percent of that of toluene.

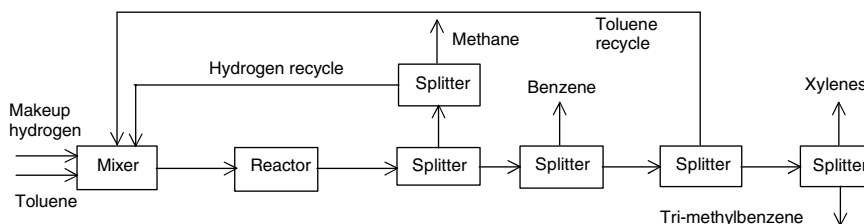
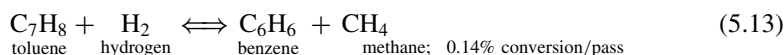
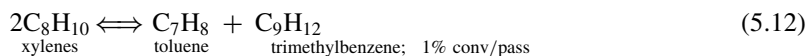
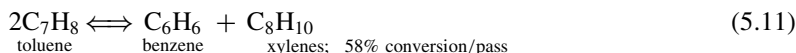


Figure 5.12. Benzene process.

The reactions and their conversion per pass are:



The output from the reactor contains all the components, and the hydrogen and toluene are recycled, the methane is removed from the process, and the benzene and xylenes are removed from the process. The vapor pressures of the components at 20°C are: benzene, 75 mmHg; toluene, 22 mmHg; xylenes, 4.8–6.6 mmHg; trimethylbenzene, 1.1 mmHg. A separation system is used that removes the hydrogen and methane first, then separates the benzene, then the toluene, and then the trimethylbenzene. The hydrogen and methane should be somewhat separated (to recover the hydrogen, which is recycled) and the toluene is recycled. In the separators, use splits of 99.9 percent of the light component out of the top and only 0.1 percent of the next heavier component. Use splits of 100 percent for components lighter than the light key and 0 percent for components heavier than the heavy key. Use Goal Seek or Solver to adjust the hydrogen makeup feed to insure the proper amount of hydrogen enters the reactor (necessary for the catalyst).

- 5.2. Revise the process modeled in Problem 5.1 using split fractions of 99 and 1 percent instead of 99.9 and 0.1 percent in all the separators. How do the total flows change? Would the equipment have to be larger? Would it cost more? Does the separation cost less? Answer these questions qualitatively now; when you finish your chemical engineering studies you will be able to answer them quantitatively.
- 5.3. Ethyl chloride is manufactured in an integrated process; see Figure 5.13 (Stirling, 1984, p. 72). Complete a mass balance for this process using Excel.

Ethane reacts with chlorine to make ethyl chloride and hydrogen chloride, and ethylene reacts with hydrogen chloride to form ethyl chloride.

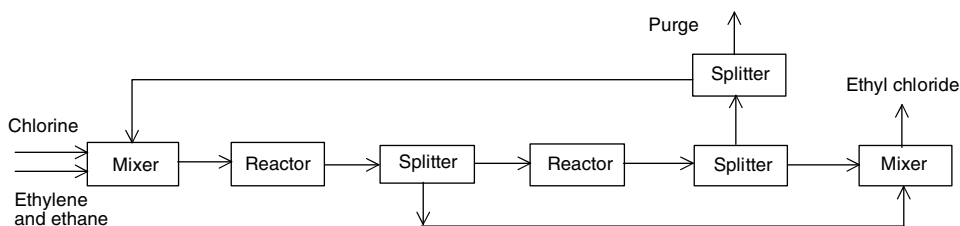
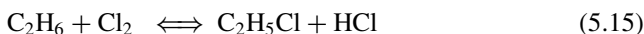


Figure 5.13. Ethyl chloride process.

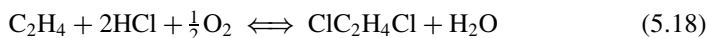
The process is fed with three streams: ethane, ethylene, and chlorine. The ethane and ethylene streams have the same molar flow rate, and the ratio of chlorine to ethane plus ethylene is 1.5. The ethane/ethylene stream also contains 1.5 percent acetylene and carbon dioxide. (For this problem, just use 1.5 percent carbon dioxide.) The feed streams are mixed with an ethylene recycle stream and go to the first reactor (chlorination reactor) where the ethane reacts with chlorine with a 95 percent conversion per pass. The product stream is cooled and ethyl chloride is condensed and separated. Assume that all the ethane and ethyl chloride go out in the condensate stream. The gases go to another reactor (hydrochlorination reactor) where the reaction with ethylene takes place with a 50 percent conversion per pass. The product stream is cooled to condense the ethyl chloride, and the gases (predominately ethylene and chlorine) are recycled. A purge or bleed stream takes off a fraction of the recycle stream (use 1 percent). Complete the mass balance for this process.

- 5.4.** Vinyl chloride monomer is the raw material for making polyvinyl chloride (PVC), which is produced in large quantities. When PVC was invented (as a highly elastic polymer by a University of Washington graduate, Waldo L. Semon, patent no. 1,929,435), the vinyl chloride was made by reacting acetylene with hydrogen chloride. A process which uses cheaper raw materials is now used (Cowfer and Gorenssek, 1997), and a simplification of it is given in Figure 5.14. Prepare a mass balance for this process using Excel.

Chlorine and ethylene are fed to a direct chlorination reactor where the reaction



takes place with essentially 100 percent conversion of the limiting reagent. The selectivity to ethylenedichloride is 99 percent with the main byproduct (and the only one considered here) being 1,1,2-trichloroethane. Ethylene and oxygen are mixed with a hydrogen chloride recycle stream and sent to an oxychlorination reactor, where the reaction



takes place with a conversion of 96 percent for ethylene and ethylenedichloride selectivities of 95 percent. Here you can assume the byproduct is all 1,1,2-trichloroethane. The output from the direct chlorination reactor is sent to a distillation tower where the

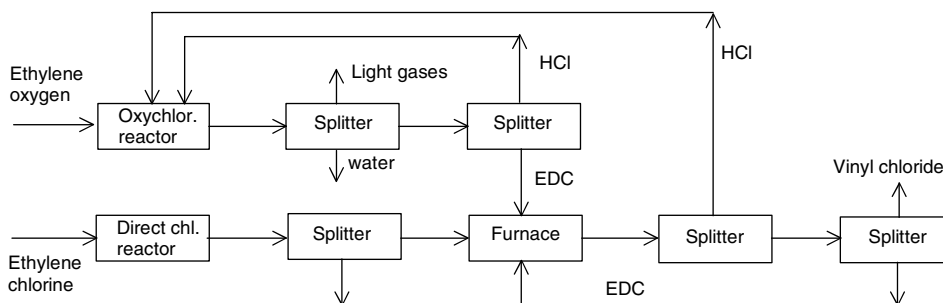
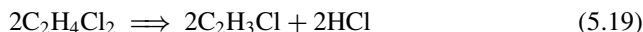


Figure 5.14. Vinyl chloride process.

heavy components are distilled off. The light components are sent to a furnace where a pyrolysis reaction takes place:



The feed to the furnace must be very pure (99.5 percent ethylenedichloride) to achieve good cracking. The output is quenched. The liquid is the vinyl chloride product and the vapor is hydrogen chloride, which is recycled to the oxychlorination reactor. The output from the oxychlorination reactor is cooled to remove the liquid (mostly water). A vent is used to remove light gases. The remainder goes to a distillation column where ethylenedichloride is the heavy component (and is sent to the furnace) and the light components are recycled to the oxychlorination reactor. The feed to the oxychlorination reactor uses a slight excess of oxygen and ethylene to ensure that all the hydrogen chloride reacts. The feed to the direct chlorination reactor is stoichiometric.

- 5.5. Modify Problem 5.4 using air instead of oxygen as the feed to the oxychlorination reactor. Make a mass balance of the revised process.
- 5.6. Synthetic ethanol is made by vapor-phase hydration of ethylene, as shown in Figure 5.15. Make a mass balance of this process using Excel.



Water and ethylene are mixed with a recycle stream and sent to a reactor where the reaction in Eq. (5.20) takes place (5 percent conversion per pass). The ethylene feed is 97 percent ethylene but also contains acetylene (2.9 percent) and inert gases (0.1 percent). The acetylene reacts with water, too, forming acetaldehyde (50 percent conversion per pass).



The reactor effluent is cooled to remove the liquids (ethanol, acetaldehyde, water). The gases (ethylene, acetylene, and inert gases) are recycled, but a purge stream is needed to remove the inert gases.

Assume perfect splits. Adjust the fraction purged to keep the ratio of inert gas to ethylene in the stream fed to the reactor at 0.04, and feed enough water to the process to make the molar ratio of water to ethylene 0.6 in the stream into the reactor. Separate the liquids into relatively pure component streams, first removing

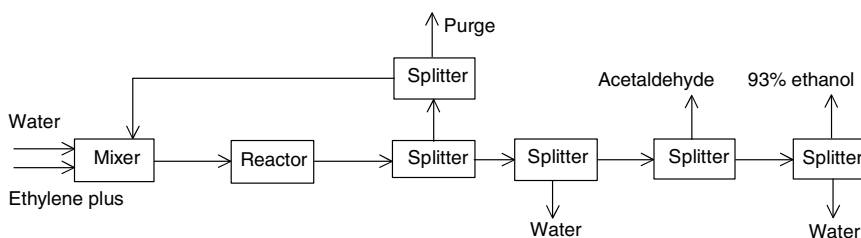


Figure 5.15. Ethanol process.

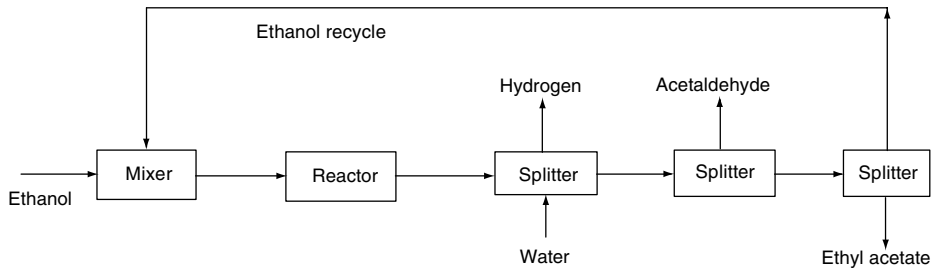


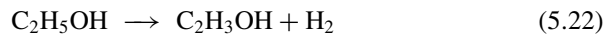
Figure 5.16. Acetaldehyde production by dehydrogenation of ethanol.

60 percent of the water, then the acetaldehyde, then purifying the ethanol to 93 percent by removing water. (Hint: This problem has two constraints. In Excel you can use Goal Seek or Solver to satisfy one of them and adjust the other variable yourself, followed by another use of Goal Seek or Solver. Repeat this process until both constraints are satisfied.)

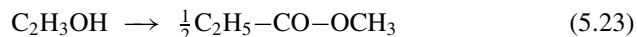
- 5.7.** Consider a process to make acetaldehyde by dehydrogenation of ethanol over a silver catalyst (Aguiló and Penrod, 1976), as illustrated in Figure 5.16.

The following design constraints must be satisfied:

- (1) Ethanol decomposes to form acetaldehyde and hydrogen,



- (2) The equilibrium limit is about 35 percent conversion per pass, so that a recycle stream is necessary;
 (3) Acetaldehyde reacts to form a number of side-products, including ethyl acetate, acetic acid, and butanol; only ethyl acetate is considered here,



- (4) The conversion per pass is about 2 percent for the reaction in Eq. (5.23);
 (5) The reactor effluent must be separated to remove hydrogen;
 (6) The acetaldehyde must be separated;
 (7) The ethyl acetate and water must be removed;
 (8) Ethanol must be recycled.

Make a mass balance using Excel with the following separations:

- (1) 99 percent of the hydrogen and 0.1 percent of the other components go out of the overhead of the first splitter;
 (2) 99 percent of the acetaldehyde, 100 percent of the hydrogen, and 0.5 percent of the other components go out of the overhead of the second splitter;
 (3) 99 percent of the ethanol and 1 percent of the other components go out of the overhead of the third splitter.

6

SIMULATION OF MASS TRANSFER EQUIPMENT

Mass transfer is one subject that is unique to chemical engineering. Typical mass transfer problems include diffusion out of a polymer to provide controlled release of a medicine, diffusion inside a porous catalyst where a desired reaction occurs, or a large absorption column where one chemical is transferred from the liquid phase to the gas phase (or vice versa). The models of these phenomena involve multicomponent mixtures and create some tough numerical problems.

In Chapters 3 and 5 you have already seen separations achieved with flash units, where the pressure is lowered and the vapor and liquid output have different compositions. Such separation works when the chemicals have widely differing boiling points, or are otherwise easy to separate. When that is not the case, engineers must use distillation. Other phenomena used for separation include absorption, liquid–liquid extraction (moving a chemical from one liquid phase to another, with immiscible liquids), and adsorption (used for identification in chromatography and for separation in chemical plants). This chapter provides simple examples of some of those processes. Your courses in mass transfer, separations, thermodynamics, and design study these processes in greater detail. In addition, mass transfer involving diffusion and flow (like in microfluidic devices) is covered in Chapters 10 and 11.

When modeling mass transfer equipment, there are two key points to remember: (1) thermodynamics is important; and (2) convergence is difficult. The corollary is that you have to compare your thermodynamic predictions with experimental data. Also, you may start with ideal thermodynamics and obtain a solution. This solution can then be used as the initial guess when the thermodynamic model is more realistic. Process simulators do not always work, so you need to be flexible about how you approach a problem.

THERMODYNAMICS

The subject of thermodynamics has already been addressed in Chapters 2 and 3. You saw that for high pressures it was necessary to go beyond the perfect gas law, and the activity coefficient of the liquid was a very important parameter. In Aspen Plus[®] there are many options, and it is your job to pick the ones that are relevant to your problem. A few of the choices are described here. The key decisions you must make are (1) what model to use for the vapor phase activity coefficient and (2) what model to use for the liquid phase activity coefficient. The Help/Physical Property Methods/Choosing a Property Method menu gives advice about which thermodynamic model is recommended for different applications. The Guidelines for Choosing a Property Method and Guidelines for Choosing an Activity Coefficient Method are sub-menus that outline decision trees to guide your choice.

The options for the vapor phase are relatively easy. Some of them are used in Chapters 2 and 3, because once you choose an equation of state, the vapor phase activity coefficient can be determined. The options for the activity coefficient of the gaseous phase are: ideal gas, Redlich–Kwong or Redlich–Kwong–Soave, Peng–Robinson, plus a few specialized ones (i.e., HF hexamerization and Hayden–O’Connell).

In the liquid phase, the simplest option is an ideal liquid, with an activity coefficient equal to 1.0. That choice leads to Raoult’s law, which may suffice for similar chemicals. Other models include regular solution theory using solubility parameters (although not in Aspen Plus), NRTL, Electrolyte NRTL, UNIFAC, UNIQUAC, Van Laar, and Wilson. Characteristics of the models are:

- The Electrolyte NRTL is especially suited for acid gas adsorption, which includes the removal of carbon dioxide and hydrogen sulfide from a gas stream. Refineries routinely use this process when making hydrogen. This is also one way of capturing carbon dioxide from a power plant to capture and sequester it.
- The UNIFAC model is a group contribution method that allows the model parameters to be estimated using the molecular structure of each chemical. When experimental data is not available, this is the only method that can be used.
- The UNIQUAC model uses binary parameters, which must be determined from experimental data. Once found, however, the same parameters can be used in multi-component mixtures of three or more chemicals.
- Both UNIFAC and UNIQUAC can be used when two liquid phases or azeotropes are present.
- The Van Laar options are less recommended in Aspen Plus; they are simpler to use than the others, but less successful in general. In Aspen Plus the ease of use is immaterial since someone else has created the program.
- The Wilson equation is an option if there is only one liquid phase, and it does handle azeotropes.

Chapter 6 of the Aspen Plus manual gives a flow chart to help you pick the model to use for your system. The decisions you have to make are

Polar or nonpolar?

Electrolyte or nonelectrolyte?

- Real or pseudocomponents?
- Pressure less than 10 bar or greater than 10 bar?
- Interaction parameters available?
- One or two liquid phases?
- Vapor phase association?
- Degree of polymerization?

This sounds pretty daunting, and it is. Fortunately, the Aspen Plus manual does give some hints on useful choices for particular applications, some of which have been developed especially for the application. A few of them are:

Oil and Gas Production (OIL GAS) – Peng–Robinson with Boston–Mathias α function (PR-BM) or Redlich–Kwong–Soave with Boston–Mathias α function (RKS-BM).

Refinery, medium pressure (Refinery) – Chao–Seader, Grayson, Peng–Robinson, Redlich–Kwong–Soave.

Refinery, hydrogen-rich applications (Refinery) – Grayson, Peng–Robinson, Redlich–Kwong–Soave.

Gas processing, hydrocarbon separations (GASPROC) – Peng–Robinson with Boston–Mathias α function (PR-BM), Redlich–Kwong–Soave with Boston–Mathias α function (RKS-BM), Peng–Robinson, Redlich–Kwong–Soave.

Gas Processing, acid gas absorption (Chemical and Electrol) – electrolyte NRTL.

Petrochemicals, aromatics and ether production (Petchem) – Wilson, NRTL, UNIQUAC.

Chemicals, phenol plants (Chemical) – Wilson, NRTL, UNIQUAC.

Chemicals, ammonia plant (Chemical) – Peng–Robinson, Redlich–Kwong–Soave, SR-Polar (Schwartzentruber–Renon).

Chemicals, inorganic chemicals (Chemical and Electrol) – electrolyte NRTL.

Coal processing, combustion (Coalproc) – Peng–Robinson with Boston–Mathias α function (PR-BM), Redlich–Kwong–Soave with Boston–Mathias α function (RKS-BM), or the combustion databank.

With all these choices, and limited knowledge of your system, you will likely want to use the recommended options and make predictions of vapor–liquid equilibrium using Aspen Plus in order to compare those predictions with experimental data. Chapter 3 presented an example of such a comparison for the ethanol–water system.

EXAMPLE: MULTICOMPONENT DISTILLATION WITH SHORTCUT METHODS

Consider a single distillation column, as shown in Figure 6.1. Short-cut methods have been developed that allow you to see the splits that can be achieved at different pressures, with different reflux ratios, and with a different number of stages. It is often useful to do a short-cut distillation calculation first, before doing the more rigorous plate-to-plate calculations,

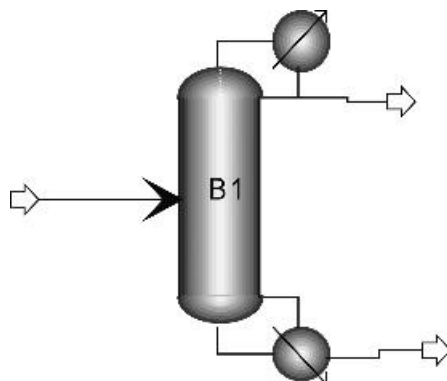


Figure 6.1. Distillation column.

because the preliminary calculations give you an idea of what is reasonable. The methods are based on the relative volatility of the chemicals, and require you to identify two components that will be essentially separated (to the extent you specify). If you line up all the components in the order of their boiling points, and draw a line between two of them, the more volatile component is called the light key and the less volatile component is called the heavy key.

For this example, you choose the first column in the process shown in Figure 5.4. In Aspen Plus you use the module DSTWU for the shortcut method; you also use RK–Soave as the physical property method, because it is a good one for hydrocarbons. The feed is 100 lb mol/h propane, 300 lb mol/h *i*-butane, and the other chemicals as listed in Table 6.1, at 138 psia and 75°F. The column operates at 138 psia with a reflux ratio of 10 (a wild guess initially, confirmed because the column worked). Remember, the minimum number of stages goes together with infinite reflux, so if your column does not work, increase the reflux ratio.

First, you specify the split you would like. To do this, you need the concept of a light key and heavy key. First, line the chemicals up in order of their boiling points, from low to high. (Note: This illustration assumes that the thermodynamics are ideal, so Raoult's law applies.) Because you are using distillation, if most of the propane is to go out in the top stream, a little of the *i*-butane will, too; very small fractions of the other species will go out in the top stream. You want most of the *i*-butane to go out of the bottom. Thus, you draw a line between propane and *i*-butane in Table 6.1. The component above the line is the light key and the component below the line is the heavy key. The distillation column is based on those two components, and the other components are split according to thermodynamics,

TABLE 6.1. Identification of Light and Heavy Key

Chemical	Boiling Point (°C) at 1 atm
Propane	−42.1
<i>i</i> -Butane	−11.9
<i>n</i> -Butane	−0.5
<i>i</i> -Pentane	−27.9
<i>n</i> -Pentane	−36.1

as described below. In this case, propane is the light key, and you want 99 percent of it to go out in the top stream; *i*-butane is the heavy key, and you want 1 percent of it to go out in the top stream.

Mathematical Development

Although you have Aspen Plus to do the calculations, it is instructive to see what equations are being used. First, you need the relative volatility of the light key to the heavy key, which is determined by thermodynamics alone (using the K -values):

$$\alpha_{LK} = \frac{K_{LK}}{K_{HK}}, \quad \alpha_i = \frac{K_i}{K_{HK}} \quad (6.1)$$

Then, use the Fenske equation (Fenske, 1932).

$$(\alpha_{LK})^{S_m} = \left(\frac{r_{LK}}{1 - r_{LK}} \right) \left(\frac{1 - r_{HK}}{r_{HK}} \right) \quad (6.2)$$

where S_m = minimum number of stages, r_i = recovery fraction of the i th component, r_{LK} = recovery fraction of light key component, and r_{HK} = recovery fraction of heavy key component.

Since you specify the fraction of the light and heavy keys, you can use the Fenske equation to obtain the minimum number of stages. Once the minimum number of stages is determined, you can find the splits of the other components from Eq. (6.3):

$$(\alpha_i)^{S_m} = \left(\frac{r_i}{1 - r_i} \right) \left(\frac{1 - r_{HK}}{r_{HK}} \right) \quad (6.3)$$

The next step is to solve the Underwood equation (Underwood, 1932) for the variable ϕ :

$$q(1 - F) = \sum_i \frac{\alpha_i n_{i,F}}{\alpha_i - \phi} \quad (6.4)$$

where $q = 0$ represents a saturated vapor feed, $q = 1$ represents a saturated liquid feed, $n_{i,F}$ = feed flow rate of the i th component, and F = total feed flow rate, $= \sum_i n_{i,F}$.

This is a nonlinear equation with one unknown, which is a class of problems you know how to solve from Chapter 2. Then, you calculate the minimum vapor flow rate using Eq. (6.5):

$$V_{\min} = \sum_i \frac{\alpha_i n_{i,D}}{\alpha_i - \phi}, \quad n_{i,D} \approx n_{i,F} \quad (6.5)$$

$$V_{\min} = R_{\min} + D$$

where $V_{\min}$ = minimum vapor flow rate, $n_{i,D}$ = distillate flow rate of the i th component, $R_{\min}$ = minimum reflux rate, D = distillate flow rate, R = reflux ratio = reflux rate/ D , R_m = minimum reflux ratio = $R_{\min}/D$, and vapor rate = reflux rate + D .

Note in this description the words ‘reflux rate’ and ‘reflux ratio.’ They are different, and you need to be careful to use the right one, since using too small a reflux rate could make your column not work. Finally, you use the Eduljee version (Eduljee, 1975) of the Gilliland correlation (Gilliland, 1940) [Eq. (6.6)] to find the actual number of stages. This equation correlates data on roughly one-hundred distillation columns. You must specify either the reflux ratio (R) or number of stages (S) in order to use this equation.

$$\frac{S - S_m}{S + 1} = 0.75 \left[1 - \left(\frac{R - R_m}{R + 1} \right)^{0.5668} \right] \quad (6.6)$$

In this entire process, only one equation could not be solved analytically, Eq. (6.4). In order to solve the problem, you needed the K -values of the chemicals (or vapor pressures), the flow rates of the feed stream, the thermal condition of the feed stream, the desired split for the light and heavy keys, and either the number of stages or the reflux ratio. The equations then find the splits of all the other components, the minimum number of stages, the minimum reflux rate, and the actual number of stages needed to achieve the desired split. These last three items are very useful when using a more rigorous method of calculation, as shown below.

If the relative volatility varies from the top to the bottom of the column, you can use the geometric mean of the values at the top and bottom (Seader et al., 1997, pp. 13–35):

$$\alpha = (\alpha_{\text{top}} \alpha_{\text{bottom}})^{1/2} \quad (6.7)$$

Winn (1958) developed another way to handle this problem with a correlation for the relative volatility, which is used by Aspen Plus. Additional details about shortcut methods are available in King (1980) and *Perry’s Chemical Engineers’ Handbook* (Perry and Green, 1997).

You can use Aspen Plus to solve this problem using the DSTWU block, which stands for DiSTillation-Winn-Underwood.

Step 1 The flowsheet is Figure 6.1, and you enter the components and feed conditions in the usual manner. (See other examples in Chapters 2–5 and Appendix C.) The thermodynamics model chosen is the Refinery, Chao–Seader option. The parts specific to the distillation column are the block parameters. Shown in Figure 6.2 is the screen where you select the block parameters. You choose the light and heavy keys (propane and *i*-butane), the splits desired (99 and 1 percent), the pressure of the column (138 psia), a total condenser, and the desired number of stages (10).

Step 2 Figure 6.3 shows the solution with stream information. You can check parts of this output. Look first at the mole balances. Indeed, 99 percent of the propane and 1 percent of the *i*-butane went out in the top stream. The other components are heavier, and you would expect them to be split in such a way that only a small fraction goes out through the overhead stream; this is indeed the case. You also get the bubble points of the distillate and bottom product: 82.0°F for the distillate and 200.9°F for the bottom product. These are close to the values shown in Chapter 5, which should give you confidence in the thermodynamics used in the calculations.

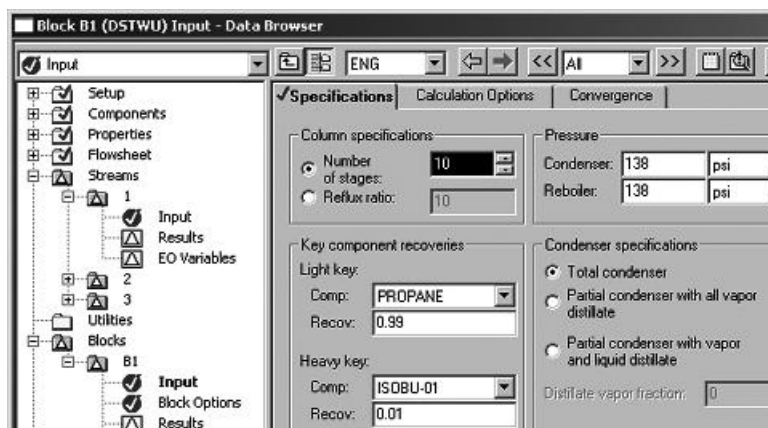


Figure 6.2. Block input information for DSTWU block.

Step 3 You also want to know the parameters of the distillation column, which are obtained by choosing the block for the column and then clicking on the Results tab, giving Figure 6.4. Note that the minimum number of stages is 12.9, the minimum reflux ratio is 2.93, the actual reflux ratio is 3.44, and the actual number of stages is 25.8. In this case, your guess of 10 stages was too small and the program modified it for you. If you had entered 26 stages, the program would have used exactly 26 stages. You could have specified the reflux ratio instead, and Aspen Plus would compute the number of stages. For reflux ratios of 3.44, you had 26 stages. A reflux ratio of 5 requires only 19.1 stages, and a reflux ratio of 10 requires only 15.5 stages. For this split the minimum number of stages is 12.9 and the feed is on the thirteenth stage (from the top).

Note that the ratio of reflux ratio to minimum reflux ratio is $3.44/2.93 = 1.17$, which is a common and economical one. However, during the energy crisis in the 1970s, engineers found that a ratio of 1.1 was more economical, although the distillation tower was harder to control. This is one of the choices (number of stages or reflux ratio) that you will make as a chemical engineer that affect the economics of the process. The cost of the column and the heating and cooling cost must be examined to find the minimum cost. In this case, the

	1	2	3
Substream: MIXED			
Mole Flow lbmol/hr			
PROPANE	100.0000	99.00000	1.000000
ISOBUT-1	300.0000	3.000000	297.0000
N-BUT-1	500.0000	2400226	499.7600
2-MET-1	400.0000	0.0	400.0000
N-PEN-1	500.0000	0.0	500.0000
Total Flow lbmol/hr	1800.000	102.2400	1697.760

Figure 6.3. Multicomponent distillation with DSTWU.

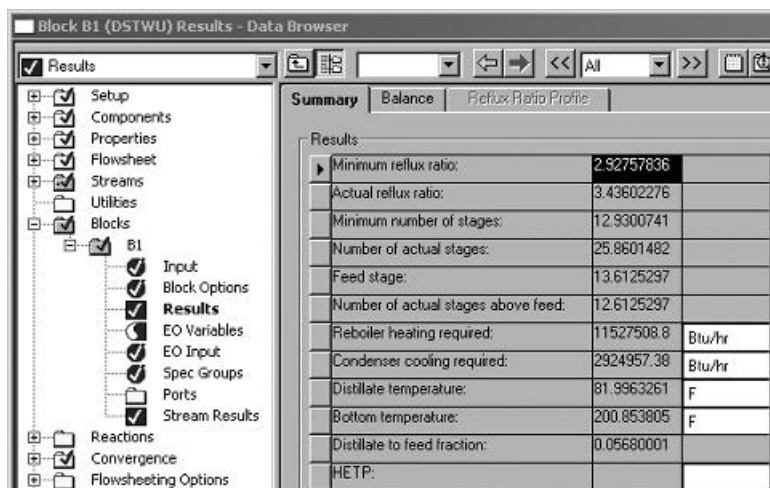


Figure 6.4. Block results for DSTWU block.

reboiler duty is 11.5×10^6 Btu/h and the condenser duty is 2.93×10^6 Btu/h. Since heating is more expensive than cooling, the major operating cost is the cost of the steam to heat the reboiler. If energy is selling for \$6/million Btu, you will have an operating cost for this column of \$69/h, or about \$610,000/year.

Your choice of thermodynamic model does make a difference. Table 6.2 shows results from three simulations, all with exactly 26 stages, using different models. The results differ, which is why it is so important to verify your choice of thermodynamic model by comparison with experimental data.

Difficulties can arise: If you misidentify the light and heavy key (say reversing them), you may get a negative number of stages. If you identify a light and heavy key that has another component that boils between them, that other component will be split in some way, which may not meet your desired specifications. If the thermodynamics is nonideal and the boiling points are not sufficient to guarantee the volatility (i.e., activity coefficients are important), then you may specify the light and heavy key correctly according to their boiling points, but the simulation may act as if they were not specified correctly.

MULTICOMPONENT DISTILLATION WITH RIGOROUS PLATE-TO-PLATE METHODS

The short-cut distillation method gives reasonable answers to straightforward problems, but distillations can have multiple feed streams or multiple product streams. There

TABLE 6.2. Comparison of Different Thermodynamic Models

Thermodynamic Model	RK–Soave	RKS-BM	Refinery/Chao–Seader
Minimum no. stages	10.1	13.4	12.9
Minimum reflux ratio	1.40	3.03	2.93
Actual reflux ratio	1.56	3.62	3.42

TABLE 6.3. Stream Table for Multicomponent Distillation with RadFrac, Reflux Ratio = 3.44

Mole flow lb mol/h			
Propane	100	85.80379	14.19621
<i>i</i> -Butane	300	12.31199	287.688
<i>n</i> -Butane	500	1.883926	498.1161
<i>i</i> -Pentane	400	0.000257	399.9997
<i>n</i> -Pentane	500	4.61×10^{-5}	500
Mole fraction			
Propane	0.055556	0.858038	0.008351
<i>i</i> -Butane	0.166667	0.12312	0.169228
<i>n</i> -Butane	0.277778	0.018839	0.293009
<i>i</i> -Pentane	0.222222	2.57×10^{-6}	0.235294
<i>n</i> -Pentane	0.277778	4.61×10^{-7}	0.294118
Total flow, lb mol/h	1800	100	1700
Total flow, lb/h	115843.6	4608.785	111234.8
Total flow, cubic ft/h	3126.01	147.3943	3489.477
Temperature, °F	75	80	199.5242
Pressure, psi	138	138	138
Vapor fraction	0	0	0
Liquid fraction	1	1	1

might even be azeotropes within the column. To handle complications like that, and to obtain better accuracy, you will want to use a rigorous plate-to-plate method that takes a vapor stream and liquid stream, equilibrates them, and sends off a vapor and liquid stream with different compositions. This is what happens on each plate, or tray, on the column. Such a method is a large computational problem, and it requires the number of plates, trays, or stages to be specified in advance. A good starting point is the number of stages suggested by the short-cut method.

The next simulation is for the same column, but using the RadFrac block in Aspen Plus. The feed is the same, the pressure is 138 psia, and the Refinery/Chao–Seader property method is used. This example uses 26 stages, and you run Aspen Plus to see what the split is. (Notice that you cannot easily set the split and find the number of stages or reflux ratio needed to achieve it.) Set the reflux ratio to 3.44 and enter the feed on the thirteenth stage.

The output is shown in Table 6.3 and Figure 6.5, and only 86 percent of the propane goes out in the top stream; along with 4 percent of the *i*-butane. The condenser is at 80°F and has a heat duty of -2.94×10^6 Btu/h, and the reboiler is at 199.5°F and has a heat duty of 11.5×10^6 Btu/h. Above the feed, the flow rate varies between 283 and 344 lb mol/h, and below it, the flow rate varies between 2998 and 3097 lb mol/h. While DSTWU assumes the same number of moles on each plate or tray, RadFrac calculates what actually occurs. To see the flow rates on each stage, choose the block, then Profiles.

Since the desired separation is not achieved, you must run the problem again with 26 stages and a higher reflux ratio; choose 5. This time 94.2 lb mol/h of propane goes out in the top stream. Increasing the reflux ratio to 7.5 gives 98.2 lb mol/h of propane, and a reflux ratio of 10 gives 99.2 lb mol/h of propane.

The results for a reflux ratio of 10 are given in Table 6.4. The separation is slightly better than required. This, of course, changes the temperatures of the condenser and

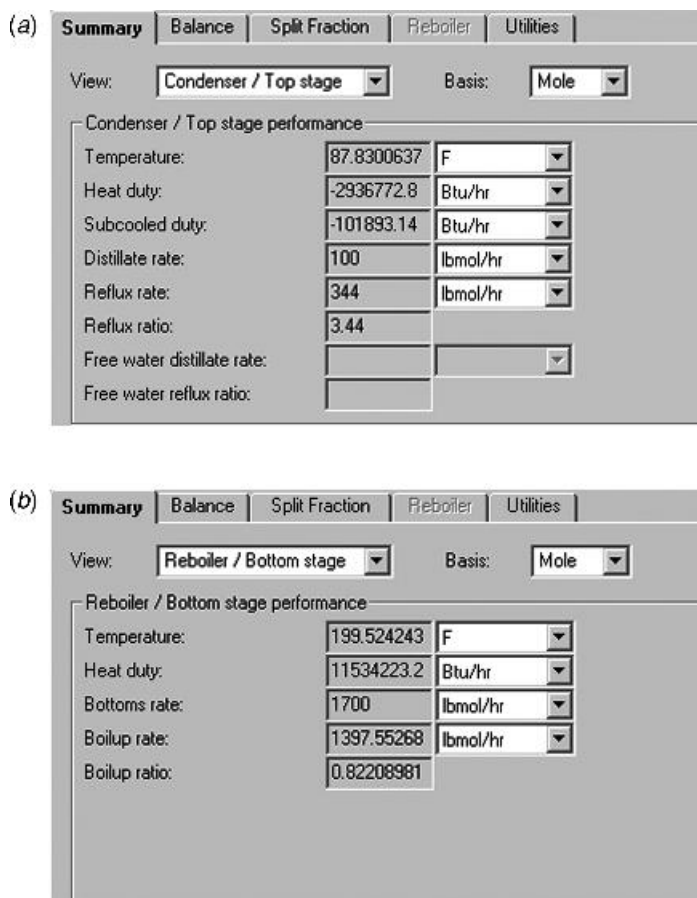


Figure 6.5. Multicomponent distillation with RadFrac, reflux ratio = 3.44. (a) Condenser; (b) reboiler.

reboiler because they are at the bubble points of the mixtures, and these change with composition. The condenser is at 80°F with a heat duty of -7.0×10^6 Btu/h, and the reboiler is at 201°F with a heat duty of 15.6×10^6 Btu/h. Above the feed stream, the flow rate varies between 793 and 1000 lb mol/h, and below it the flow rate varies between 3500 and 3600 lb mol/h. The error was reduced on successive iterations, from 61, 14, 9.2, to 0.83, when the program stopped.

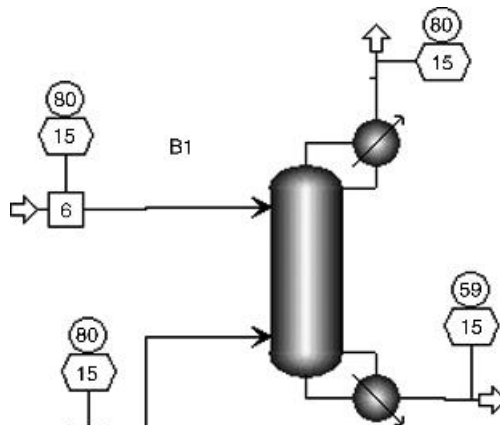
EXAMPLE: PACKED BED ABSORPTION

If a gas contains a contaminant you would like to remove, one way of doing this is to bring the gas into contact with a liquid that can absorb it; this is often done in an absorption column, as illustrated in Figure 6.6. Here you take an air stream containing acetone at a concentration of 2 percent; you want to lower its concentration to 0.4 percent using water in a column. In Aspen Plus, you can use RadFrac. The only difference

TABLE 6.4. Multicomponent Distillation with Radfrac, Reflux Ratio = 10

Mole flow, lb mol/h			
Propane	100	99.20709	0.792907
<i>i</i> -Butane	300	0.743919	299.2561
<i>n</i> -Butane	500	0.048985	499.951
<i>i</i> -Pentane	400	3.01×10^{-6}	400
<i>n</i> -Pentane	500	5.37×10^{-7}	500
Mole fraction			
Propane	0.055556	0.992071	0.000466
<i>i</i> -Butane	0.166667	0.007439	0.176033
<i>n</i> -Butane	0.277778	0.00049	0.294089
<i>i</i> -Pentane	0.222222	3.01×10^{-8}	0.235294
<i>n</i> -Pentane	0.277778	5.37×10^{-9}	0.294118
Total flow, lb mol/h	1800	100	1700
Total flow, lb/h	115843.6	4420.774	111422.9
Total flow, cubic ft/h	3126.01	144.4282	3500.062
Temperature, °F	75	80	200.7896
Pressure, psi	138	138	138
Vapor fraction	0	0	0
Liquid fraction	1	1	1
Solid fraction	0	0	0
Enthalpy, Btu/lb mol	-69471.48	-52010.89	-65438.33
Enthalpy, Btu/lb	-1079.461	-1176.511	-998.4053
Enthalpy, Btu/h	-1.25×10^8	-5201089	-1.11×10^8

from distillation towers is that you will not have a condenser or reboiler, and there are two inputs, one at each end. Before making that model, you have to decide on an appropriate thermodynamic model. Figure 6.7 shows experimental data, along with the predictions of Aspen Plus using the Wilson-2 thermodynamic model, which is Wilson with an ideal gas. The fit is not perfect, but it is acceptable for your purpose.

**Figure 6.6.** Absorption column.

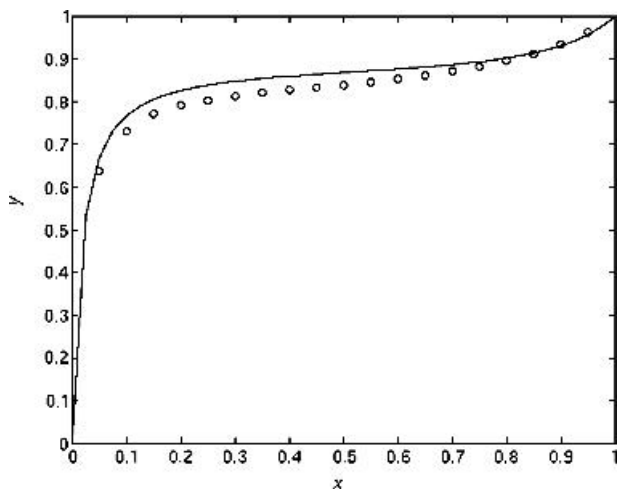


Figure 6.7. Vapor-liquid diagram for acetone-water.

Use a 12-stage column and have the air stream enter on stage 12 and the water stream enters on stage 1, as shown in Figure 6.8. (The stages are numbered from the top.) The feed streams are both taken as 80°F and 15 psia. Table 6.5 shows the feed rates of the air and water stream.

Acetone has been transferred from the air stream to the water stream, and the mole fraction of the air stream leaving is 0.33 percent. Notice also that the air, which was dry on entering, is now wet, with water vapor forming about 3.4 percent of the gas stream. A small amount of air is dissolved in the water, too. In a complete system, of course, the water effluent is taken to another column where the acetone is removed, and the water is recycled. Then the water stream is saturated with air and contains some acetone as well.

✓Configuration ✓Streams ✓Pressure ✓Condenser			
Feed streams			
	Name	Stage	Convention
▶	4	12	On-Stage
	6	1	Above-Stage
Product streams			
	Name	Stage	Phase
	7	1	Vapor
	8	12	Liquid

Figure 6.8. Absorption column parameters in Aspen Plus.

TABLE 6.5. Acetone Removal Stream Table

Mole flow lb mol/h				
Acetone	3.527396	0	0.610603	2.916793
Water	0	176.3698	6.07332	170.2965
Air	172.8424	0	172.5983	0.244157
Total flow, lb mol/h	176.3698	176.3698	179.2822	173.4574
Total flow, lb/h	5208.746	3177.352	5141.682	3244.415
Total flow, cubic ft/h	69485.12	51.3116	70568.4	52.53542
Temperature, °F	80	80	79.51011	58.69392
Pressure, psi	14.7	14.7	14.7	14.7
Vapor fraction	1	0	1	0
Liquid fraction	0	1	0	1

EXAMPLE: GAS PLANT PRODUCT SEPARATION

Consider the gas plant product separation process described in Figure 5.4. The pressures in the units have been chosen so the overhead streams can be cooled with cooling water at 80°F. Thus, the designer looked at the vapor pressure of each component before fixing the pressures. The temperatures of the reboilers are basically at the bubble point of the bottom stream, and this will be an output of the computer program.

Use Aspen Plus and model each distillation column using the DSTWU model as shown in Figure 6.9. First, specify the split of key components: for the light component you want 99 percent out of the top and for the heavy component you want 1 percent out of the top. The other components will be split according to the Fenske equation [Eq. (6.3)].

Choose the thermodynamic property method as Refinery/Chao–Seader. Choose the pressures of the columns to agree with the gas plant in Figure 5.4. Each column has 26 stages, and the temperatures shown in Table 6.6 are the result of thermodynamics predicted with Aspen Plus. They are very close to those listed in Figure 5.4.

Aspen Plus also gives information such as the number of stages, minimum reflux ratio, and heat duties in the reboiler and condenser. With the information shown, engineers can calculate the capital cost of the equipment (sometimes using rules of thumb to account for instrumentation, pumps, valves, etc.). At least some of the operating cost, the raw material cost and product value, and the cooling and heating cost can also be calculated. For distillation towers at normal temperatures, the heating cost per unit of energy is about 10 times the cooling cost when using cooling water. If refrigeration is used to cool the condensers, that energy cost is even more expensive than the cost of

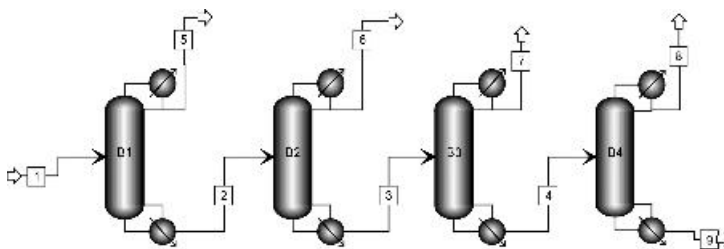
**Figure 6.9.** Gas plant separation in a distillation train.

TABLE 6.6. Results from Gas Plant Separation

	1	2	3	4	5	6	7	8	9
Temperature, °F	75	200.8538	138.655	144.6153	81.99633	79.92966	80.72647	99.93148	115.9679
Pressure, psi	138	138	51	36	138	51	36	20	20
Vapor fraction	0	0	0	0	0	0	0	0	0
Mole flow, lb mol/h	1800	1697.76	1397.732	900.5959	102.24	300.0279	497.1363	401.9845	498.6113
Mass flow, lb/h	115843.6	111289.7	93865.13	64908.83	4553.877	17424.62	28956.3	28933.89	35974.95
Volume	3126.01	3495.803	2655.604	1796.609	148.7339	508.1377	811.6607	768.5361	961.2719
flow, cubic ft/h									
Enthalpy, MMBtu/h	-125.0487	-111.0993	-96.23131	-65.50753	-5.34678	-19.91258	-31.5121	-30.48343	-36.39745
Mole flow, lb mol/h									
Propane	100	1	3.12×10^{-14}	0	99	1	0	0	0
<i>i</i> -Butane	300	297	2.97	0.000665	3	294.03	2.969335	0.000665	6.93×10^{-25}
<i>n</i> -Butane	500	499.76	494.7624	4.947624	0.240023	4.9976	489.8148	4.947624	2.45×10^{-25}
<i>i</i> -Pentane	400	400	399.9997	395.9997	0	0.000259	3.999997	392.0397	3.959997
<i>n</i> -Pentane	500	500	500	499.6478	0	0	0.35218	4.996478	494.6513

TABLE 6.7. Conditions in Distillation Columns in Gas Plant Separation

Column	B1	B2	B3	B4
Minimum reflux ratio	2.93	9.5	1.9	7.9
Actual reflux ratio	3.43	10.8	2.2	8.9
Minimum number of stages	12.9	31.1	11.3	36.8
Number of actual stages	26	62.3	26	73.5
Feed stage	13.7	33.4	13.6	39.2
Reboiler heating required	11.5×10^6	23.8×10^6	13.4×10^6	40.0×10^6
Condenser cooling required	2.9×10^6	28.8×10^6	14.2×10^6	41.3×10^6

steam. Experienced designers know that, and that is why the pressures are chosen so that cooling water can be used in the condensers. The conditions of the columns are given in Table 6.7.

CHAPTER SUMMARY

This chapter introduced you to the many thermodynamic models available in Aspen Plus. The equations for short-cut distillation were summarized, and the Aspen Plus was used to solve a variety of distillation problems, with either short-cut methods (DSTWU) or plate-to-plate methods (RadFrac). You also learned how to solve gas absorption problems using Aspen Plus.

CLASS EXERCISE

Divide into groups of two to four students. Work through each of the examples, with each person in the team taking a unit, showing where the parameters are for that block and checking the results. (1) Tables 6.3 and 6.4 and Figure 6.5; (2) Figure 6.8 and Table 6.5; (3) Tables 6.6 and 6.7. Alternatively, your instructor can prepare a flowsheet and dataset showing the results, but put in some errors for your group to find.

PROBLEMS (USING ASPEN PLUS)

- 6.1.** Model column B2 in Figure 6.8 using a short-cut, DSTWU block. The inlet stream has a temperature of 138.7°F, pressure of 51 psia, and molar flow rates as shown in Table 6.6.
- 6.2.** Model column B2 in Figure 6.8 using a detailed model, RadFrac. (Hint: Use the results from Problem 6.1 to choose key operating parameters.) Prepare a report comparing the output when using DSTWU and RadFrac to model the distillation column. What information was needed for each? How do the results compare?
- 6.3.** (1) Model column B3 in Figure 6.8 using a short-cut, DSTWU block. (2) Then model the same column using RadFrac. How do the results compare?

- 6.4.** (1) Model column B4 in Figure 6.8 using a short-cut, DSTWU block. (2) Then model the same column using RadFrac. How do the results compare?
- 6.5.** Model example 1 on pp. 13–36 in *Perry's Chemical Engineering Handbook* (Seader et al., 1997) using (1) DSTWU; (2) RadFrac.
- 6.6.** Results for the gas plant are given in Tables 6.6 and 6.7. Which separations are the most expensive? If you were the designer, where would you want to spend your time?
- 6.7.** Carbon dioxide from a fermentation process contains 1 mol percent ethyl alcohol. The alcohol needs to be removed by contact with water at 35°C and 1 atm. The gas flow rate is 400 lb mol/h and the water stream is 620 lb mol/h and contains 0.02 mol percent alcohol. Determine the compositions out of the absorption column if you model it with 10 stages.
- 6.8.** Natural gas contains mostly methane but also small amounts of other chemicals that are valuable in themselves. In a refrigerated absorption process, the natural gas is cooled to -40°F at 865 psia and sent to an absorber, which uses dodecane as the absorbing media, with a flow rate of 10,000 lb mol/h. Determine the fraction of each of the chemicals absorbed in a tower with 30 stages.

Component	Flow Rate (lb mol/h)
Methane	700,000
Ethane	27,000
Propane	11,000
<i>i</i> -Butane	3200
<i>n</i> -Butane	2800
<i>i</i> -Pentane	1200
<i>n</i> -Pentane	770
Hexane	820

- 6.9.** Figure 5.14 shows an ethanol process. The last unit is a distillation tower to remove water from a mixture of water and ethanol. Note that the mixture forms an azeotrope. The feed stream is 100 lb mol/h of a 50–50 molar mixture of water and ethanol at 80°F and 1 atm.
- (1) Simulate a distillation tower to create the azeotropic mixture (93 percent ethanol). Use the Wilson-2 option for the thermodynamic model (see Figures 3.6 and 3.7 for the data). Model the column using the DSTWU short-cut model and 10 stages.
 - (2) Use RadFrac with the reflux rate set by the output from the DSTWU model.
 - (3) Repeat (1) and (2) using the ideal option for the thermodynamic model. How do the results differ from (1) and (2)?

7

PROCESS SIMULATION

Process simulation is used to determine the size of equipment in a chemical plant, the amount of energy needed, the overall yield, and the magnitude of the waste streams. Because the results of process simulation depend upon thermodynamics and transport processes, the mathematical models are complicated and would be time-consuming to solve without a computer. This chapter illustrates the use of a process simulator, Aspen Plus, to model a plant to make ammonia.

The problems solved in Chapters 5 and 6 are simple problems with many numerical parameters specified. You may have wondered where those numbers came from. In a real case, of course, you will have to make design choices and discover their impact. In chemical engineering, as in real life, these choices have consequences. Thus, you must make mass and energy balances that take into account the thermodynamics of chemical reaction equilibria and vapor-liquid equilibria as well as heat transfer, mass transfer, and fluid flow. To do this properly requires lots of data, and the process simulators provide excellent databases. Chapters 2–4 discussed some of the ways in which thermodynamic properties are calculated. This chapter uses Aspen Plus exclusively. You will have to make choices of thermodynamic models and operating parameters, but this will help you learn the field of chemical engineering. When you complete this chapter, you may not be a certified expert in using Aspen Plus®, but you will be capable of actually simulating a process that could make money.

First, the chapter lists the possible unit operations in the Aspen Plus Model Library, because the process is a connected set of the units. Then an example process is illustrated that makes ammonia from nitrogen and hydrogen. You will be able to get both the mass balances and the energy balances for the process. With this information you can determine the size of most of the equipment needed, and hence its cost. You can also determine the operating cost for heating, cooling, compression, and other tasks. The process involves a

recycle stream, too, which means you will need to iterate. The chapter ends with a description of factors that may control whether you get a solution at all, such as convergence acceleration techniques and tear streams.

MODEL LIBRARY

When engineers design a process, they choose different modules to represent the different units. You can view these by clicking on the icons at the bottom of the Aspen Plus screen. If the icons are not displayed, choose the View/Model Library menu option. When you click on a unit, the words shown at the bottom of the screen give a brief description of the module. Some of the module choices are described in Table 7.1. Your task is to combine these units with connecting streams into a process, which can then be simulated on the computer.

TABLE 7.1. Selected Major Blocks Available in Aspen Plus

1. Mixers/splitters
a. Mixer – combine material
b. Fsplit – split specifications for output
c. Ssplit – splits using substreams
2. Separators
a. Flash2 – rigorous vapor–liquid split or vapor–liquid–liquid split
b. Flash3 – rigorous vapor–liquid–liquid split
c. Decanter – separate two liquid phases
d. Sep – use split fractions
e. Sep2 – separation based on flows and purities
3. Heat exchangers
a. Heater – heaters, coolers, condensers
b. HeatX – co- and counter-current heat exchangers
c. Hetran – shell and tube heat exchangers
d. Aerotran – air-cooled heat exchangers
4. Columns
a. DSTWU – shortcut distillation using Winn–Underwood–Gilliland equations and correlations
b. Distl – Edmister shortcut distillation
c. RadFrac – rigorous two phase and three phase, absorber, stripper, distillation columns using stages
d. Extract – liquid–liquid extraction
5. Reactors
a. RStoic – stoichiometric fractional conversion or extent of reaction
b. RYield – nonstoichiometric – based on yield distribution
c. REquil – rigorous equilibrium
d. RGibbs – rigorous equilibrium and/or multiphase Gibbs free energy minimization
e. RCSTR – continuous stirred-tank reactor, specify volume
f. RPlug – plug flow reactor, specify length and diameter of tube
g. RBatch – batch reactor, specify cycle times
The RCSTR, Rplug, and RBatch reactors require the choice of a reaction set and the building up of a reaction rate expression
6. Pressure changes
a. Pump
b. Compr – compressor
7. User models

EXAMPLE: AMMONIA PROCESS

Consider the ammonia process, as illustrated in Figure 7.1. The feed to the process is nitrogen, hydrogen, and a small amount of carbon dioxide (left over from the process to make the hydrogen). The process feed is mixed with a recycle stream, heated to the reactor temperature, and sent to the reactor. The feed to the reactor is roughly a 3:1 mixture of hydrogen and nitrogen, with some ammonia, too. The reaction is limited by equilibrium considerations, so the conversion is not known. You must determine it by solving the equilibrium equation (using RGibbs). In this example, the pressure is high because that favors the reaction. The temperature is also high in order to get a fast reaction, even though this limits the conversion in the reactor. The output from the reactor is cooled and sent to a vapor–liquid separator (this example uses Flash2), where the temperature is lowered. Thus, most of the ammonia condenses and is removed as a liquid. The vapor is recycled and is sent through a compressor to get the pressure back to the desired pressure. A small part of the recycle stream is bled off as a purge stream to prevent the buildup of impurities.

In this example, you will take an input stream to the process (at 80°F and 300 psia) of: nitrogen, 100 lb mol/h; hydrogen, 300 lb mol/h; ammonia, 0; and carbon dioxide, 1 lb mol/h.

The inlet stream is compressed to 4000 psi with an isentropic compressor. The stream is mixed with the recycle stream and heated to 900°F, the reactor temperature. In the reactor, there is a pressure drop of 30 psi. The outlet is cooled to 80°F and the liquid and vapor phases are separated. The vapor phase goes to recycle, and 0.01 percent of it is used as purge. A recycle compressor then compresses the rest from 3970 to 4000 psia. In a real process, the heat transfer to preheat the feed to the reactor uses the effluent from the reactor, usually inside the same vessel. In process simulators, though, it is useful to begin as shown in Figure 7.1 to help convergence. NRTL thermodynamics was chosen and is justified *a posteriori*.

The identification of components is shown in Figure 7.2. It is useful to check the connections between the units using the Flowsheet option/Section//GLOBAL, as shown in Figure 7.3. Any streams that were inadvertently not connected to a unit will show up here because those stream numbers will not appear as input stream numbers.

The operating parameters for the compressors are shown in Figure 7.4. Both of them are modeled as isentropic compressors with the appropriate discharge pressure.

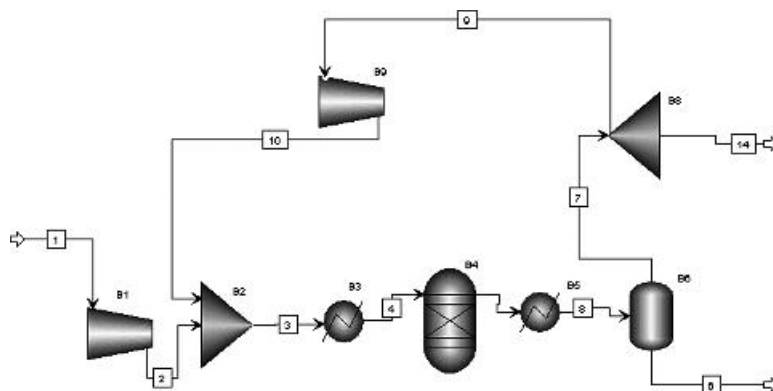


Figure 7.1. Ammonia process.

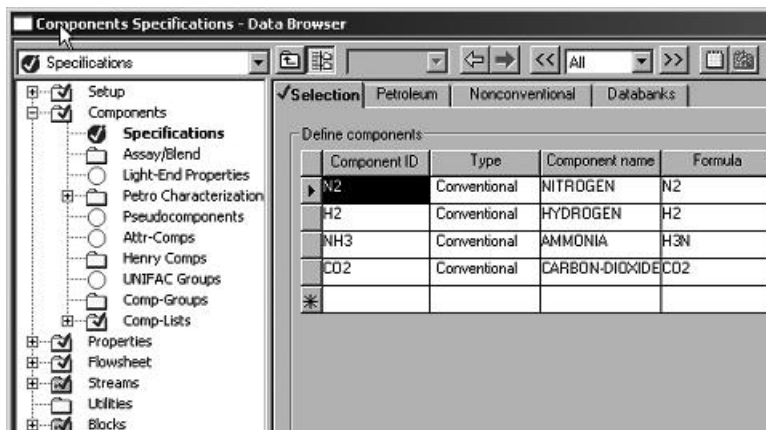


Figure 7.2. Components for ammonia process.

You can see the operating parameters for the two heat exchangers in Figure 7.5. You can see the reactor specifications in Figure 7.6. The specifications for the mixer (B2), vapor–liquid separator (B6) and simple splitter (B8) are shown in Figure 7.7. In Aspen Plus, you can specify different conditions in temperature and pressure than the inlet conditions. This may be useful in a first simulation when you are not as interested in the actual heating and cooling required, but this example keeps the unit conditions the same as the stream entering the unit so no heat transfer or compression/expansion occurs.

Now that the process is fully specified, you can solve it. Because there is a recycle stream, begin the calculations with the first unit, the compressor. The output from that unit is combined with the recycle stream, which initially is zero. Follow the stream through the rest of the process until it computes the recycle stream. This recycle stream is now different from its previous value. Thus, the program does another iteration, and

Block ID	Model	Input streams ID	Port	Output streams ID	Port
B1	Compr	1	F(IN)	2	P(OUT)
B4	RGlobe	4	F(IN)	5	P(OUT)
B6	Flash2	8	F(IN)	6	L(OUT)
				7	V(OUT)
B8	FSpli	7	F(IN)	14	P(OUT)
				5	P(OUT)
B2	Mixer	10	F(IN)	3	P(OUT)
		2	F(IN)		
B3	Heater	3	F(IN)	4	P(OUT)
B5	Heater	5	F(IN)	8	P(OUT)
B9	Compr	9	F(IN)	10	P(OUT)

Figure 7.3. Flowsheet summary.

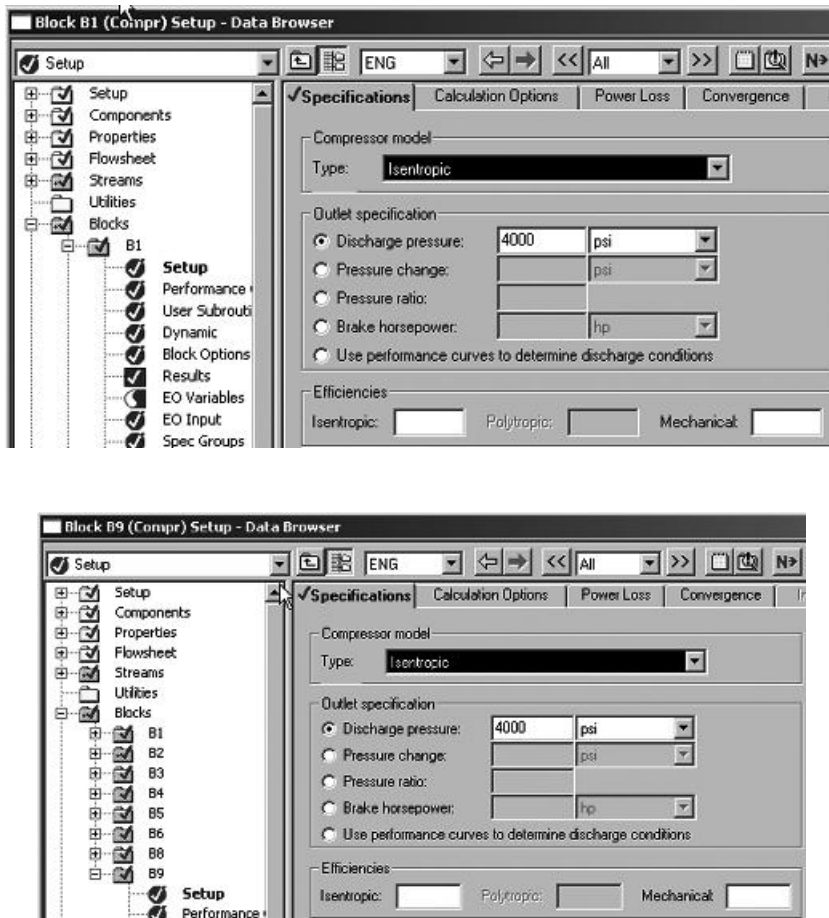


Figure 7.4. Specifications for compressors, units B1 and B9.

it continues until there is no further change. Figure 7.8 shows some of the information Aspen Plus provides about the convergence.

Figure 7.8a indicates that the sequential module is used for iteration. Unit B1 is calculated only once, since the exact input to that unit is known. Then units B2–B9 are calculated. Since B9 is the last unit, Aspen Plus compares the output from unit B9 on this iteration to the output from unit B9 one iteration before. If they are the same, convergence has been achieved. If not, more iterations are necessary. Figure 7.8a indicates that the iterations have not fully converged after 30 iterations, and the mass balance error is in block B9 (the recycle compressor). It gives the total discrepancy as 0.5805 vs 0.5801, for a relative difference of 0.07 percent. This difference is so small that you could accept the results. This simulation used the Wegstein method of accelerating convergence. If you change to the Broyden method (Convergence/Conv Options, Figure 7.8b), it does not converge much better. Finally, a direct method of iteration did converge. The direct method is not fancy; you simply calculate around the loop over and over again. You can see the final conditions by choosing Results/Summary and Convergence (Figure 7.9). The discussion below gives additional details about convergence.

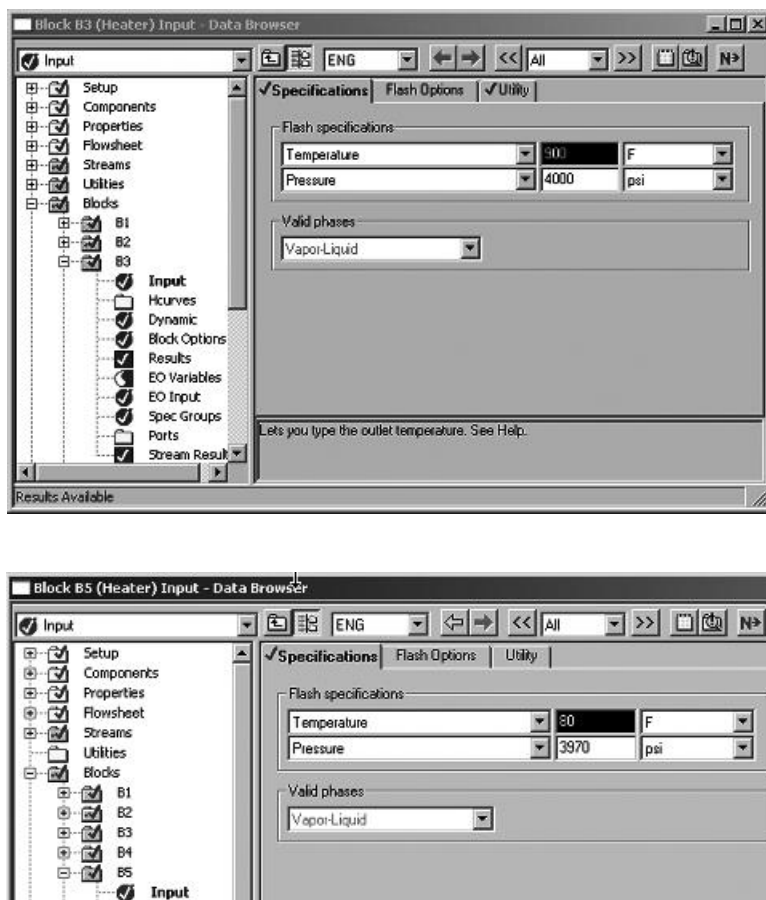


Figure 7.5. Specifications for heat exchangers, units B3 and B5.

The final mass and energy balance are shown in the stream table (Table 7.2). By checking the mass balances around each unit, you can see that they balance except for a small discrepancy between streams 9 and 10. The hydrogen flow rate is off by 0.05 lb mol/h, compared with a total flow of 1490 lb mol/h. The overall conversion (100 mol nitrogen to 197.5 mol ammonia) is reasonable. The rest of the raw materials are lost. Most of the carbon dioxide goes out in the ammonia so perhaps the purge stream was not necessary.

You can learn several things by looking at 'Results' under each block. First, look at Figure 7.10, which is for units B1 and B9. The feed compressor is very large (900 HP). In fact, if the compression ratio was this large, $4000/300 = 13.3$, the job would be split into two compressors, each with a compression ratio of the square root of 13.3, or 3.65. For ideal conditions, you can use thermodynamics to show that the optimal configuration has equal compression ratios with interstage cooling back to the inlet temperature, because this results in less total horsepower. Next, look at Figure 7.11. The heat duties of each exchanger are given, which permits estimation of the capital cost. The heat duties can be combined with the enthalpies of the process streams to verify an energy balance. Notice also that the ratio of hydrogen to nitrogen in the recycle stream is not 3:1. This is because thermodynamics dictates that the

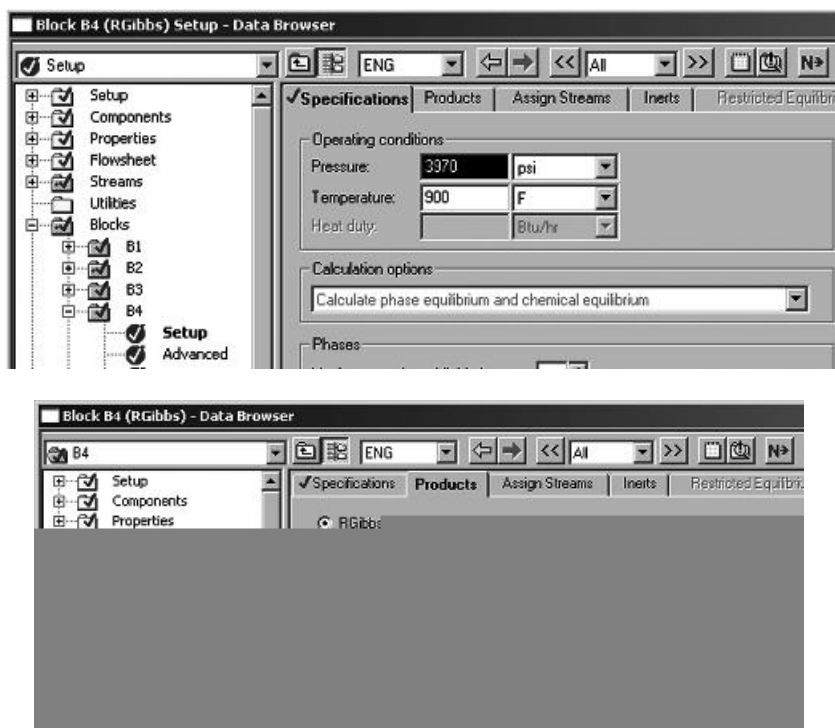


Figure 7.6. Specifications for the reactor, unit B4.

solubility of hydrogen and nitrogen in the ammonia is different, and hydrogen builds up. In the reactor, nitrogen is the limiting reagent, and 73 percent of the nitrogen fed to the reactor reacts. Thus, the conversion per pass is 73 percent, based on nitrogen. If you look at the product, though, 197.5 lb mol ammonia are made from 100 lb mol nitrogen and 300 lb mol hydrogen. Because the raw materials can be converted into a maximum of 200 lb mol, the overall yield of the process is 197.5/200 or 98.8 percent.

Even more information can be obtained using the View/Report menu. This gives details about individual blocks. The View/History menu gives information about the convergence and what calculations have been done. The View/Input Summary lists the parameters you have set.

The thermodynamics can be tested using the results from Table 7.2. The reactor effluent is in equilibrium. Take the mole fractions of stream 5 and compute the K -value:

$$K = \frac{y_{\text{NH}_3}}{(y_{\text{N}_2} y_{\text{H}_2}^3)^{1/2} p} \quad (7.1)$$

With values of $y_{\text{N}_2} = 0.0204$, $y_{\text{H}_2} = 0.8346$, $y_{\text{NH}_3} = 0.1434$, the K -value is 0.00488. Stephenson (1966) gives a measured value of 0.00473 at these conditions. Thus, the chemical reaction equilibrium is satisfactory when using the NRTL thermodynamic model.

Next you can check the vapor–liquid equilibria, as represented by streams 6 and 7. Data reported by Stephen and Stephen (1963, p. 557) give the solubility of nitrogen in ammonia as 22.48 cm³ (measured at 0°C and 760 mmHg) per gram of ammonia

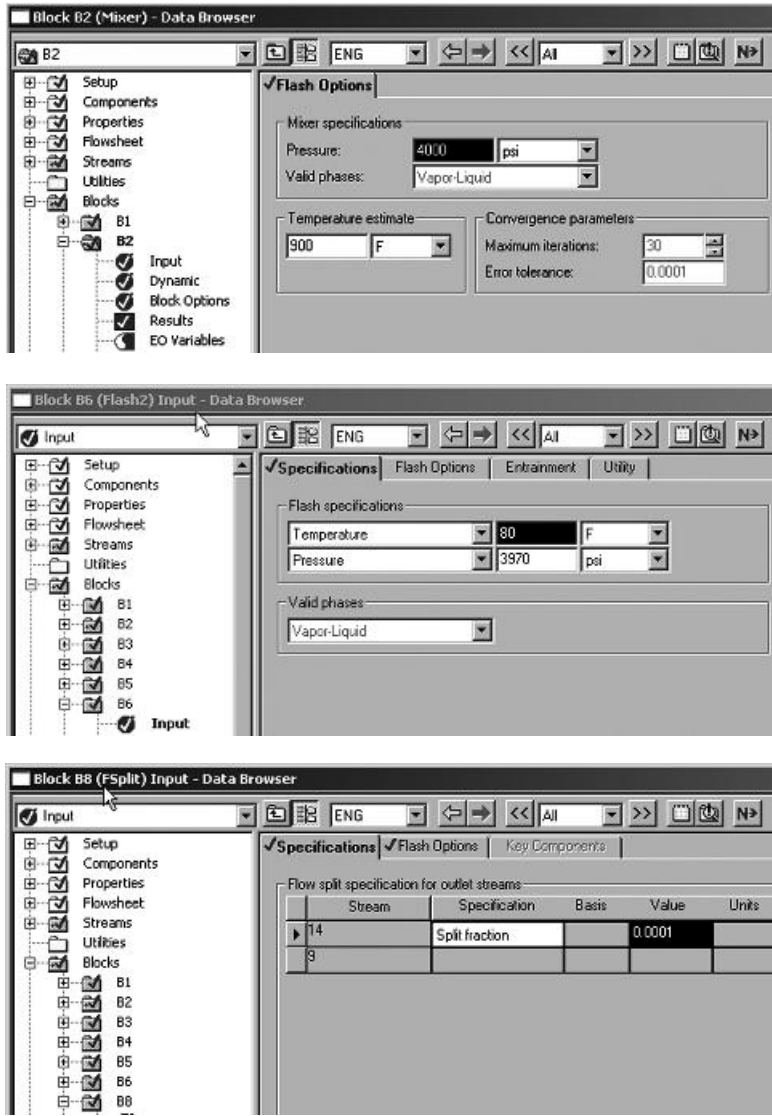


Figure 7.7. Specifications for the mixer and separator, and simple splitter, units B2, B6, and B7.

when the pressure is 200 atm and the temperature is 25°C. The corresponding number is 37.02 cm³/g ammonia when the pressure is 400 atm. A linear interpolation gives 27.6 cm³/g ammonia when the pressure is 270 atm. This is a mole fraction of nitrogen in the liquid of 0.021. Using the values for stream 6 gives a mole fraction of 0.0061, which is much smaller; perhaps this is due to the hydrogen and carbon dioxide also in the stream. Other thermodynamic models give similar results.

Data reported by Stephen and Stephen (1963, p. 537) give the solubility of hydrogen in ammonia as 13.11 cm³ (measured at 0°C and 760 mmHg) per gram of ammonia when the pressure is 200 atm and the temperature is 25°C. The corresponding number is 18.96 cm³/g ammonia when the pressure is 300 atm. A linear interpolation gives 17.2 cm³/g ammonia

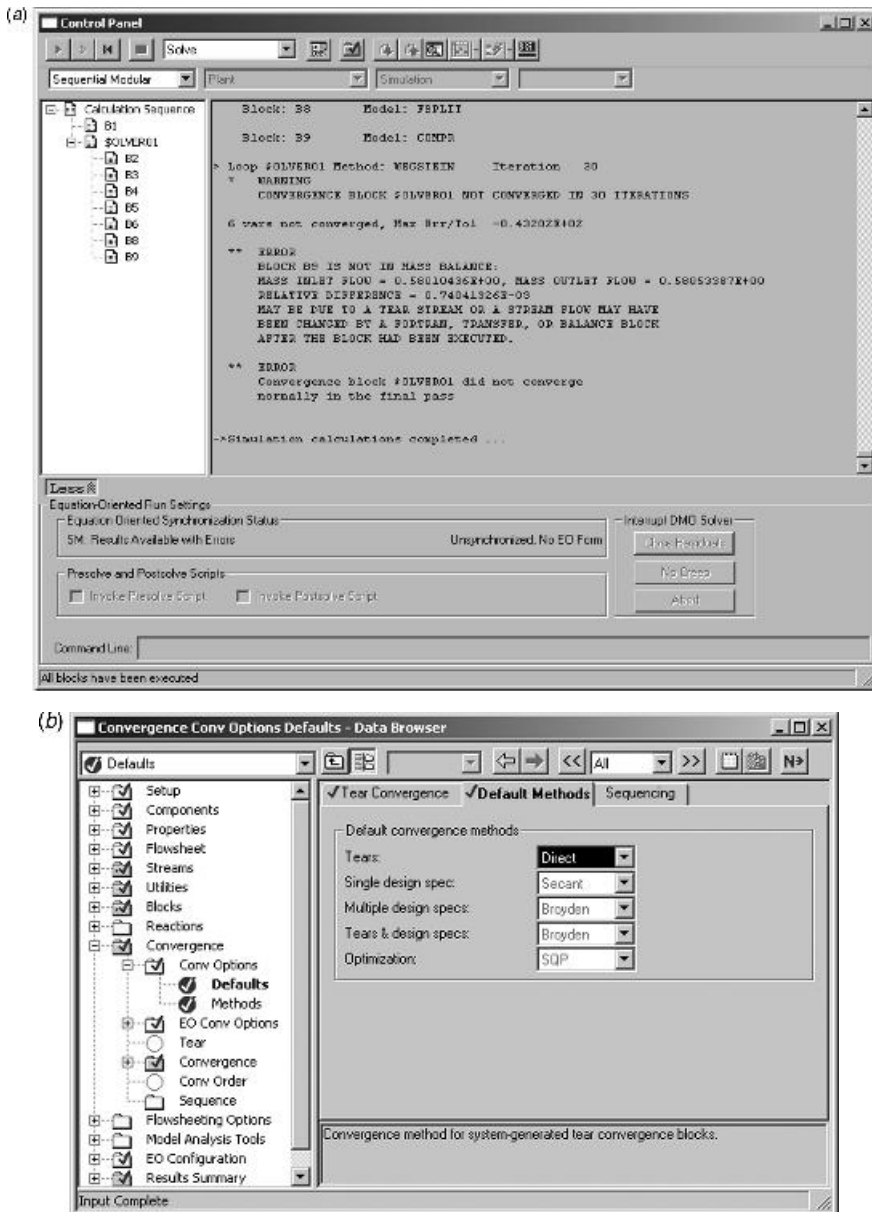


Figure 7.8. Convergence of the mass balance: (a) control panel, (b) convergence methods.

when the pressure is 270 atm. This is a mole fraction of hydrogen in the liquid of 0.013. Using the values for stream 6 gives a mole fraction of 0.018, which is reasonably close.

Utility Costs

The cost of raw materials is usually the major operating cost, but following that the cost of utilities is very important. Aspen Plus provides a convenient way to obtain the cost of

Results Summary Convergence - Data Browser

Convergence

DesignSpec Summary: Tear Summary

Final status for tears:

Tear	Status	Maximum error	Tolerance	Maximum error / Tolerance	Variable with maximum error
10	Converged	1.9321E-08	2.4047E-08	0.80346891	CO2 MOLEFL

Figure 7.9. Convergence of the mass balance.

utilities in the entire process. You begin by opening the Utilities folder in the Data Browser (or use the Data/Utilities menu). There are several possible forms of utilities; choose New, select water and fill out the form shown in Figure 7.12. Then choose gas and fill out the form shown in Figure 7.13. Choose electricity and assign a price of 5 cents/kW h.

The next step is to tell the computer which blocks are going to use utilities. Select block B3 (the heater before the reactor) and click on the input button, then the Utility tab. Select Gas, as shown in Figure 7.14. Do the same thing for block B5 (the cooler after the reactor), and select Water, and then blocks B1 and B9 and select Electricity (at 5 cents/kW h). Now the utilities have been specified to Aspen Plus.

After the mass and energy balances are computed, the program calculates how much cooling water and natural gas are needed, and their cost. That information can be accessed from the Data Browser/Utility folder, under Results, as shown in Figure 7.15. The cost of the natural gas is almost half a million dollars per year. By combining the heater and cooler before and after the reactor, this cost can be reduced. The cost of electricity for the compressors is \$300,000 per year.

This completes the example. The steps are:

Step 1 List process specifications for the ammonia process.

Step 2 Test the thermodynamics and compare the calculations with experimental data.

Step 3 Make design choices for different units: the reactors, compressors, heat exchangers, and distillation towers.

Step 4 Solve the mass and energy balances.

Step 5 Obtain the utility costs.

Step 6 Do the calculations iteratively. Convergence acceleration techniques are sometimes necessary (not shown in the example).

TABLE 7.2. Mass and Energy Balance for Ammonia Process

	1	2	3	4	5	6	7	8	9	10	14
Molar flow, lb mol/h											
N ₂	100.00	100.00	135.25	135.25	36.49	1.24	35.25	36.49	35.25	35.25	0.00
H ₂	300.00	300.00	1789.98	1789.98	1493.70	3.62	1490.08	1493.70	1489.93	1489.98	0.15
NH ₃	0.00	0.00	59.21	59.21	256.73	197.51	59.21	256.73	59.21	59.21	0.01
CO ₂	1.00	1.00	2.91	2.91	2.91	1.00	1.91	2.91	1.91	1.91	0.00
Total flow, lb mol/h	401.00	401.00	1987.34	1987.34	1789.83	203.38	1586.45	1789.83	1586.29	1586.34	0.16
Total flow, lb/h	3450.12	3450.12	8533.52	8533.52	8533.52	3449.73	5083.80	8533.52	5083.29	5083.40	0.51
Total flow, cubic ft/h	7741.20	1452.90	3761.47	7249.42	6578.25	95.66	2314.31	2409.97	2314.08	2303.65	0.23
Temperature, °F	80.00	890.83	245.82	900.00	900.00	80.00	80.00	80.00	80.00	81.62	80.00
Pressure, psi	300	4000	4000	4000	3970	3970	3970	3970	3970	4000	3970
Vapor fraction	1	1	1	1	1	0	1	0.8864	1	1	1
Liquid fraction	0	0	0	0	0	1	0	0.1136	0	0	0
Solid fraction	0	0	0	0	0	0	0	0	0	0	0
Enthalpy, Btu/lb mol	-401.19	5301.56	344.96	5009.06	3038.18	-28222.34	-919.19	-4021.60	-919.19	-907.98	-919.19
Enthalpy, Btu/lb	-46.63	616.19	80.34	1166.54	637.23	-1663.82	-286.84	-843.49	-286.84	-283.35	-286.84
Enthalpy, Btu/h	-160878	2125927	685562	9954720	5437816	-5739722	-1458248	-7197970	-1458102	-1440365	-146
Entropy, Btu/lb mol-R	-4.8010	-3.5080	-9.1477	-4.4755	-6.5801	-42.8532	-11.4165	-14.9886	-11.4165	-11.4105	-11.4165
Entropy, Btu/lb-R	-0.5580	-0.4077	-2.1304	-1.0423	-1.3801	-2.5264	-3.5626	-3.1437	-3.5626	-3.5608	-3.5626
Density, lb mol/cubic ft	0.0518	0.2760	0.5283	0.2741	0.2721	2.1261	0.6855	0.7427	0.6855	0.6886	0.6855
Density, lb/cubic ft	0.4457	2.3746	2.2687	1.1771	1.2972	36.0634	2.1967	3.5409	2.1967	2.2067	2.1967
Average MW	8.6038	8.6038	4.2939	4.2939	4.7678	16.9624	3.2045	4.7678	3.2045	3.2045	3.2045
Liquid volume 60°F, cubic ft/h	344.02	344.02	1704.97	1704.97	1535.52	174.48	1361.04	1535.52	1360.90	1360.95	0.14

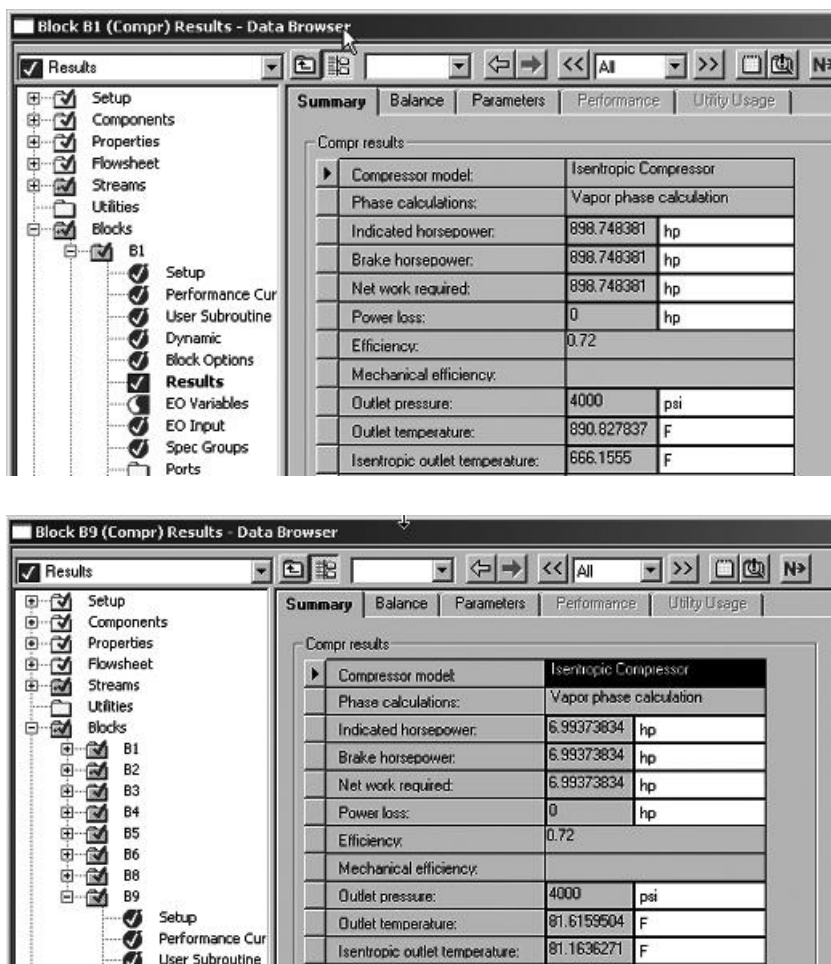


Figure 7.10. Compressor variables, units B1 and B9.

CONVERGENCE HINTS

To understand the convergence process used in most simulators, consider the process shown in Figure 7.16. Chapter 5 discusses the sequential and simultaneous methods. The sequential method is simple, but it might not converge. Thus, convergence acceleration techniques are used to improve the chances of convergence; two of those are described here (Wegstein, 1958; Broyden, 1965, 1970; Perry and Green, 1997). Simultaneous methods are called equation-oriented (EO) methods in Aspen Plus. Take a single component for illustration; in Figure 7.16, you would know the flow rate in the feed stream. The mass balance on the mixer takes the flow rate of streams 1 and 6 and adds them to become stream 2. The reactor changes the amount, and then the separator separates the stream into two streams. One of those is the recycle stream. Usually you do not know stream 6 when you start the calculation; in the first iteration you often

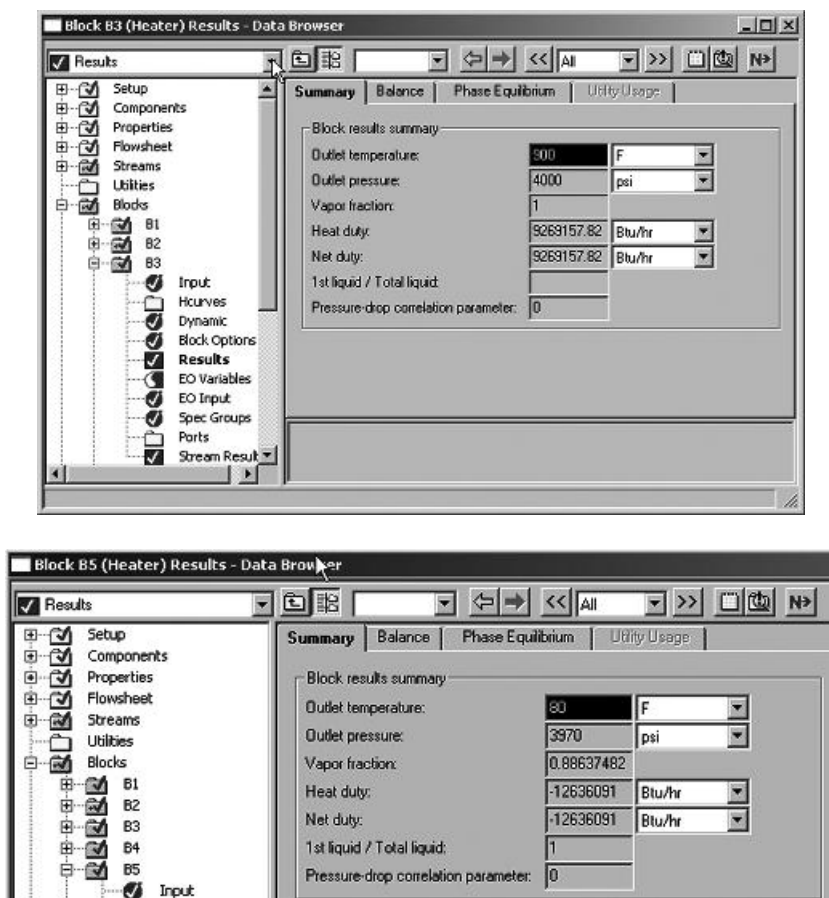


Figure 7.11. Heat exchanger variables, units B3 and B5.

take it as zero, or some other assumed value. After reaching the separator, you have a new possible value for stream 6. Write this process as an equation:

$$x = f(x) \quad (7.2)$$

In this form, x is the flow rate of the chemical in the recycle stream, and $f(x)$ is the value of the flow rate of the chemical in the output of the separator, obtained after doing the calculation once around the loop. You have solved the problem when these two values agree. To make this an iterative process, call x^n the value of x after n iterations.

$$x^{n+1} = f(x^n) \quad (7.3)$$

This iterative process is simple to apply, since the function $f(x)$ is really a series of functions evaluated one after the other, and each step is a mass balance on one simple unit. It may not converge, however. In order to speed up the convergence, the Wegstein method is very effective. The method calculates a provisional value of $\hat{x}^{n+1}$ and then

Utilities WATER (UTILITY) - Data Browser

WATER

ENG

Specifications State Variables Properties Flash Options EO Options

Utility type: Water

Utility cost

Purchase price: \$/lb

Energy price: 0.6 \$/MMBtu

Calculation option

☒ Specify heating/cooling value ☐ Specify inlet/outlet conditions

Energy requirement and temperature specification

Heating/Cooling value: 1 Btu/lb

Inlet temperature: 80 F

Outlet temperature: 120 F

Utility type:

Results Available

Figure 7.12. Specification of cooling water utilities.

uses the last two iterates to calculate a 'modified' value, which is then used as the new value in place of $\hat{x}^{n+1}$.

$$\hat{x}^{n+1} = f(x^n), x^{n+1} = qx^n + (1 - q)\hat{x}^{n+1} \quad (7.4)$$

where

$$q = -\frac{b}{1 - b}, b = \frac{x^n - \hat{x}^{n+1}}{x^{n-1} - x^n}$$

Specifications State Variables Properties Flash Options EO Options

Utility type: Gas

Utility cost

Purchase price: \$/lb

Energy price: 6 \$/MMBtu

Calculation option

☒ Specify heating/cooling value ☐ Specify inlet/outlet conditions

Energy requirement and temperature specification

Heating/Cooling value: 0.0239 MMBtu/lb

Inlet temperature: 1200 F

Outlet temperature: 950 F

Figure 7.13. Specification of gas utilities.

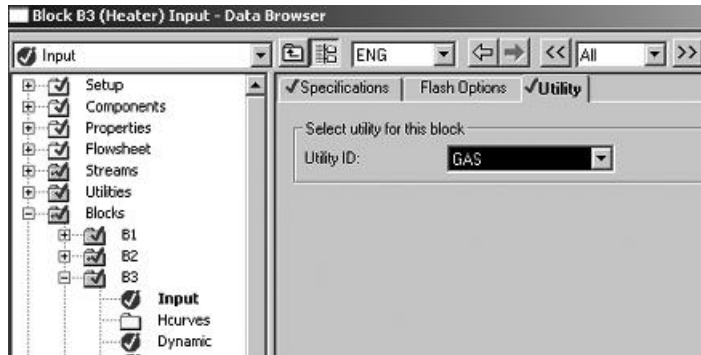


Figure 7.14. Process with recycle stream.

Wegstein (1958) found that the iterations converged differently depending on the value of q :

$q < 0$, convergence is monotone

$0 < q < 0.5$, convergence is oscillatory

$0.5 < q < 1$, direct method diverges in oscillatory manner, Wegstein method converges

$1 < q$, direct method diverges in oscillatory manner, Wegstein method converges

The derivation of the Wegstein method begins with Eq. (7.2), written as a new function, $F(x)$, as shown in Eq. (7.5):

$$F(x) = x - f(x) \quad (7.5)$$

Summary				
Utility summary				
Utility ID:		GAS	WATER	ELECTRIC
Utility type:		GAS	WATER	ELECTRICITY
Costing rate:	\$/hr	55.614945	7.58147851	33.7706637
Mass flow:	lb/hr	387.830879	12635798.2	
Duty:	Btu/hr	9269157.97	12635798.2	2304605.75
Heating/Cooling value:	Btu/lb	23900	1	
Electricity price:	\$/kWhr			0.05
Energy price:	\$/Btu	6E-06	6E-07	
Inlet temperature:	F	1200	80	
Outlet temperature:	F	950	120	

Figure 7.15. Summary of utility cost.

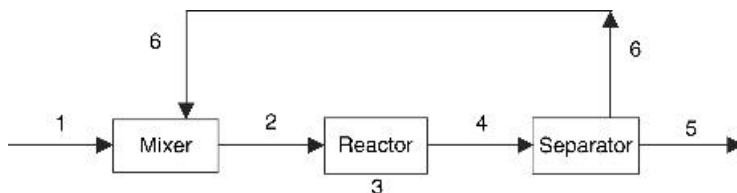


Figure 7.16. Process with recycle stream.

Apply the secant method to this function:*

$$0 = F(x^n) + \frac{F(x^{n-1}) - F(x^n)}{x^{n-1} - x^n} [(x^*)^{n+1} - x^n] \quad (7.6)$$

However, you also know that

$$\begin{aligned} F(x^n) &= x^n - f(x^n) = x^n - x^{n+1} \\ F(x^{n-1}) &= x^{n-1} - f(x^{n-1}) = x^{n-1} - x^n \end{aligned} \quad (7.7)$$

Combining these equations gives

$$(x^*)^{n+1} = \frac{x^{n-1}x^{n+1} - (x^n)^2}{x^{n-1} - 2x^n + x^{n+1}} \quad (7.8)$$

which is the same as the Wegstein method, Eq. (7.4). Sometimes, convergence is enhanced by applying Wegstein only every so many iterations rather than each iteration. The Wegstein iteration method is the default method in Aspen Plus, but there are others, including the direct method and Broyden's method. While Wegstein's method works on one variable at a time, Broyden's method works on all the variables at once using a matrix that is adjusted iteration by iteration. In the direct method, Eq. (7.3) is used as it stands.

Another term used when discussing iteration methods is the tear stream. This is the stream that is guessed (or set to zero) when stepping through the process the first time. In Figure 7.16, if you compute in the order mixer, reactor, separator, then stream 6 is the stream you must assume and check – it is the tear stream. If you compute in the order reactor, separator, mixer, then stream 2 is the tear stream. You apply the Wegstein iteration method to the tear stream. Of course, if the calculations are simple, as would be the case if all physical properties are specified and there are no energy balances, then you can use a spreadsheet. It may be so fast that you do not care whether 10, 100, or 1000 iterations are needed.

Sometimes it is convenient to have the calculations made one unit at a time. This can be done using the Run/Step menu or Ctrl-F5, or by clicking on the open circle: . It may also be necessary to re-initialize, that is, set the variables to their values at the start of the simulation. This is useful if the iterations do not converge and the answer is not reasonable.

*The secant method is the Newton–Raphson method with a numerical version of the derivative based on the last two calculated values (see Appendix F).

Just rerunning the problem will start from the unreasonable answer. To put the variables back to their beginning value, use the re-initialize button: .

OPTIMIZATION

Aspen Plus has the capability of optimizing a function that you define. An example is given here that is very simple, but it illustrates the process. Take a single unit, the reactor in the ammonia process. The reactor RGibbs is used and the production of ammonia is maximized with respect to the reactor pressure. In this example, the reactor is unit B1, with input stream 1 and output stream 2.

Step 1 In the Data Browser, open the Optimization folder. Select New in the Object manager; change the name, if desired, and click OK.

Step 2 Open the Optimization O-1, as shown in Figure 7.17. Initially this window has no entries; select New and define a variable name that will be involved in the optimization (any name is acceptable).

Step 3 In order to define the variable, here called CONV, click on Edit, and complete the window as shown in Figure 7.18. Select in turn the type, Mole-Flow; Stream, 2 (the output stream of this simple example); Component, Ammonia.

Step 4 Click on the Objectives and Constraints, and fill in the window as shown in Figure 7.19. You wish to maximize CONV.

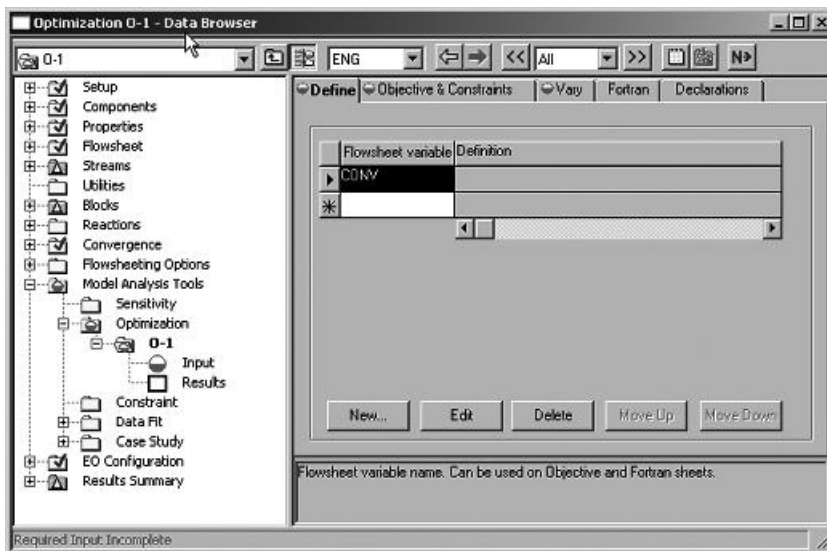


Figure 7.17. Creating an identity for the optimization process.

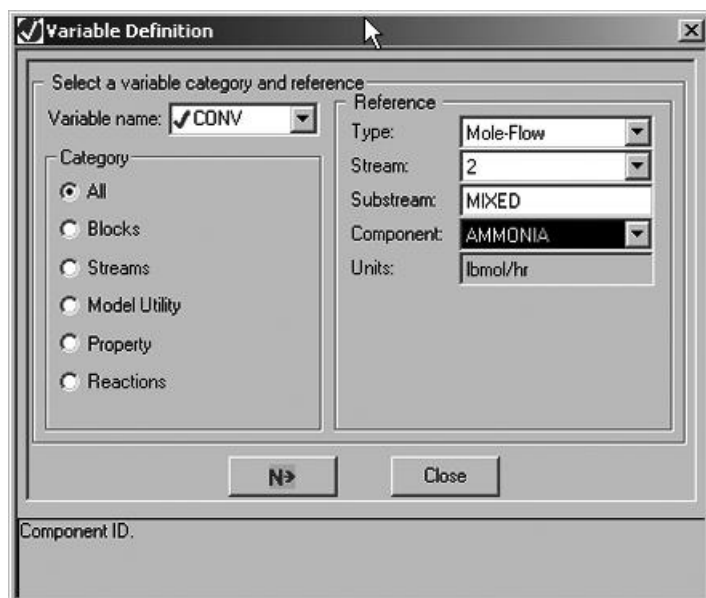


Figure 7.18. Definition of the variable.

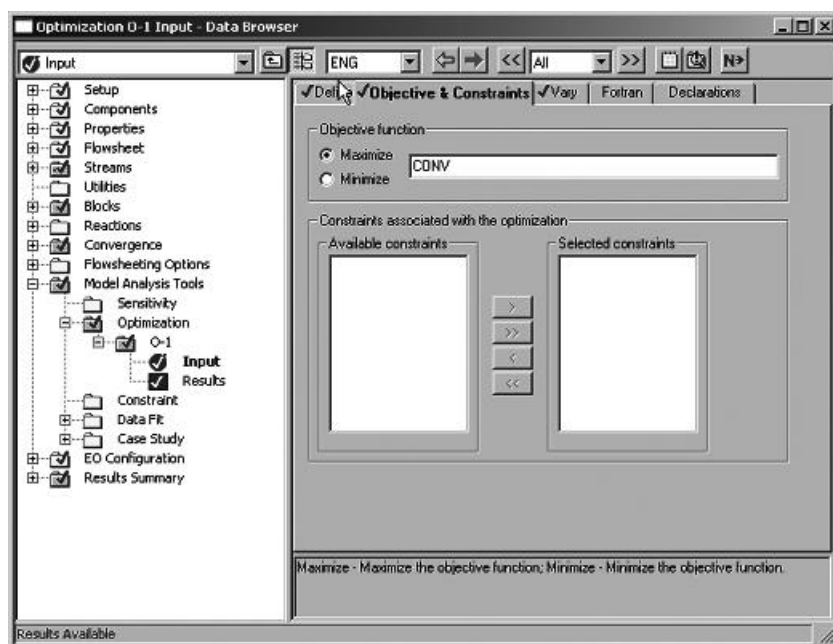


Figure 7.19. Objectives and constraints.

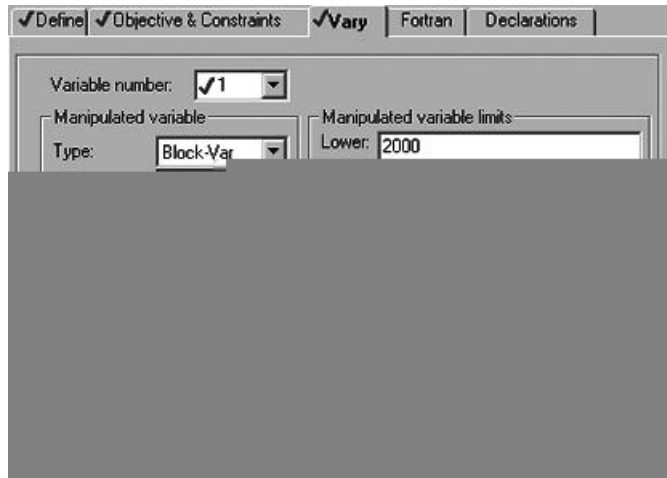


Figure 7.20. Parameters to vary.

Step 5 Next click on the Vary tab in order to define which variable to change. Figure 7.20 shows the choice of Block-Var for block B1 (the reactor in this simple example), and indicates that the pressure is to be varied, between the limits of 2000 and 6000 psia.

Step 6 Solve the problem by running the program. The final results are shown in Figure 7.21.

	1	2	
Temperature F	900.0	900.0	
Pressure psi	2200.00	6000.00	
Vapor Frac	1.000	1.000	
Mole Flow lbmol/hr	400.000	305.601	
Mass Flow lb/hr	3406.112	3406.112	
Volume Flow cuft/hr	2652.940	743.180	
Enthalpy MMBtu/hr	2.313	0.154	
Mole Flow			
HYDROGEN	300.000	158.402	
NITROGEN	100.000	52.801	
AMMONIA		94.399	

Figure 7.21. Optimization results.

Note that you only obtain the final results, but do not know how sensitive the result is to the reactor pressure. In this case, though, the pressure is at the maximum allowed in Figure 7.19, as expected. You can use the Sensitivity option in the Model Analysis Tools to find the conversion as a function of pressure. It operates in a similar way, but provides more information.

CHAPTER SUMMARY

This chapter listed many of the possible units in the Model Library of Aspen Plus. The ammonia process illustrated the procedures (and computer windows) you used to set the process conditions and examine the results. The thermodynamics choices can be verified by comparison with data reported in the literature. Sometimes the calculations do not converge, and then the Wegstein method, or Broyden's method, are useful for accelerating convergence.

CLASS EXERCISE

Divide into teams of two to four students. Work through a spreadsheet (Table 7.2), with each person in a team taking a unit, showing where the parameters are for that unit and checking the results. Are they correct? Does the mass balance? Does the energy balance? Work through the whole process. Alternatively, your instructor can prepare a flowsheet and dataset showing the results, but with some errors for your team to find.

PROBLEMS

- 7.1. Simulate the benzene process (Problem 5.1) using Aspen Plus. Take the feed at room temperature and 1 atm. Compress it to 35 atm. Preheat the feed to the reactor to 550°C and cool the effluent. Model the reactors as RStoic reactors, and keep the hydrogen/methane separations as simple splitters; model the other separations using distillation towers. You will have to decide on the number of stages and reflux ratio, and using DSTWU first might be useful.
- 7.2. Simulate the ethyl chloride process (Problem 5.3) using Aspen Plus. The feed streams are at room temperature and 20 psia and they are compressed to 95 psia. Preheat the feed to the reactor to 800°F and cool the effluent. Model the reactors as RStoic reactors. Take the effluent from the first reactor, cool it, and send the liquid to the product stream. Compress the gases to 265 psia, and heat them to 350°F (if needed). Lower the pressure and cool the effluent from the second reactor and separate the liquid and vapor. Recycle the vapor and send the liquid to the product stream. You will have to decide on the cooling temperature to use in the vapor–liquid separators.
- 7.3. Simulate the vinyl chloride process (Problem 5.4) using Aspen Plus. Take the feed at room temperature and 20 psia. Operate the direct chlorination reactor at 65°C and 560 kPa. A distillation column removes the trichloroethane and the rest of the stream is sent to the furnace. Heat the stream to 1500°F so pyrolysis takes place. Cool the effluent from the furnace, and recycle the vapor (mostly HCl). Send the liquid (vinyl chloride and ethylenedichloride) to a distillation column for separation.

You will have to decide on the number of stages and reflux ratio; using DSTWU first might be useful. The oxychlorination reactor operates at 200°C and 1400 kPa. Wash the effluent from the oxychlorination reactor with water to remove contaminants and then remove the water and light gases in a distillation column. The ethylenedichloride must be dry (<10 ppm) before going to the furnace. The stream then goes to a distillation column where the ethylenedichloride is purified ($>99.5\%$) before it is sent to the furnace. Since the furnace operates at 560 kPa, the pressure is reduced before reaching the furnace.

- 7.4. How does the process in Problem 7.3 change if the furnace is run at the high pressure (1400 kPa)?
- 7.5. Simulate the ethanol process (Problem 5.6) using Aspen Plus. The feed streams are at 1 atm and room temperature, but the reactor operates at 960 psia and 570°F. Thus, you must heat the reactor feed, and after the reaction occurs you must cool the product. The first splitter is a vapor–liquid separator (you choose the temperature that will separate ethanol from ethylene), and the remaining splitters are distillation columns. You will have to decide on the number of stages and reflux ratio; using DSTWU first might be useful.
- 7.6. Modify the process in Problem 7.5 to use a counter-current heat exchanger (and trim heater) to heat the reactor feed and cool the reactor product.

8

CHEMICAL REACTORS

Ethane flowing through a cylindrical tube can be heated to the point that it reacts into hydrogen and ethylene, the precursor to polyethylene. Many reactors, though, require a catalyst to speed up the reaction. The water gas shift reactor makes hydrogen using a cylindrical tube that is filled with catalyst and past which the gas flows. One feature that sets chemical engineers apart from other engineering disciplines is their ability to handle chemical reactions, the process of converting one or more chemicals into other chemicals that are more valuable. Thus, chemical engineers need to understand chemical reactions and be able to model equipment in which the chemical reactions take place, that is, chemical reactors. Sometimes reactors are as simple as a cylindrical tube, with reactants fed in one end and products (and unreacted reactants) coming out of the other end. Reactants may be placed in a vessel and stirred to create new chemicals.

Real-life complexities challenge engineers as they design chemical reactors. In some reactors, the reaction rate is slow. When this happens, a bigger reactor is needed to achieve a specified production rate, and it may be too big to be economical. Then you look for a catalyst, which speeds up the reaction but remains unchanged after the reaction. The tube is packed with catalyst and reactants flow around the catalyst. Often the reaction is exothermic and the reactor becomes hot. The energy must be removed somehow, and heat transfer must be included in the calculations. Sometimes the reaction occurs inside a porous catalyst, perhaps an alumina pellet (Al_2O_3) with catalyst coating the internal surface. Then it may be necessary to include heat and mass transfer effects inside the pellet.

When it is necessary to include these effects – slow reaction rates, catalysts, heat transfer, and mass transfer – it can make an engineering problem extremely difficult to solve. Numerical methods are a must, but even numerical methods may stumble at times. This chapter considers only relatively simple chemical reactors, but to work with these you must learn to solve ordinary differential equations as initial value problems.

The chapter begins with reactor models described in mathematical terms. You will learn how to solve ordinary differential equations using MATLAB[®]. Then, after you have mastered that new skill, you will solve several chemical reactor problems using MATLAB. Sometimes these problems can be solved using FEMLAB[®], too, as the chapter illustrates next. Note that this chapter does not use Excel[®] to solve ordinary differential equations. The reason is that all the other programs have guaranteed error control. While it is possible to solve simple ordinary differential equations with Excel, it becomes cumbersome if the problem is difficult, and you will not know the accuracy of your solution without extra effort. Some reactors, though, are expressed as algebraic equations, and both Excel and MATLAB are used to solve those equations.

MATHEMATICAL FORMULATION OF REACTOR PROBLEMS

A differential equation for a function that depends on only one variable is called an ordinary differential equation. The independent variable is frequently time, t , but for reactors it can also be length down a plug flow reactor. An example of an ordinary differential equation is

$$\frac{dy(t)}{dt} = f(y), \quad y(0) = y_0 \quad (8.1)$$

The general solution to the differential equation includes many possibilities; the engineer needs to provide initial conditions to specify which solution is desired. If all conditions are available at one point [as in Eq. (8.1)], then the problem is an initial value problem and can be integrated from that point on. If some of the conditions are available at one point and others at another point, then the ordinary differential equation becomes a two-point boundary value problem (see Chapter 9). Initial value problems as ordinary differential equations arise in the control of lumped parameter models, transient models of stirred tank reactors, and generally in models where there is no diffusion of the unknowns.

Example: Plug Flow Reactor and Batch Reactor

Consider a plug flow reactor – a cylindrical tube with reactants flowing in one end and reacting as they flow to the outlet. A mole balance is made here for the case when the operation is steady. Let F be the molar flow rate of a chemical, which changes dF in a small part of the tube; take the volume of that small part to be dV . The rate of reaction is expressed as the moles produced per unit of time per unit of volume. If a chemical specie reacts, then the r for that specie is negative. Thus, the overall equation for a small section of the tube is

$$dF = r dV \quad (8.2)$$

The molar flow rate can be related to the velocity in the tube.

$$F = uAc \quad (8.3)$$

In many cases the velocity and cross-sectional area of the tube are constant. Then you can write

$$dF = uA \, dc, \quad dV = A \, dz \quad (8.4)$$

If you put these into Eq. (8.2) and divide by dz you get

$$u \frac{dc}{dz} = r, \text{ plug flow reactor} \quad (8.5)$$

Normally r is a function of c , and this is an ordinary differential equation for $c(z)$. In addition, you need to know the initial value of the concentration, that is,

$$c(0) = c_0 \quad (8.6)$$

If the reactor contains a catalyst, then the reaction rate is expressed as the molar rate of change per mass of catalyst, r' . One must then multiply by the mass of catalyst per unit volume, but otherwise the situation is the same.

$$dF = \rho_B r' \, dV \quad \text{and} \quad u \frac{dc}{dz} = \rho_B r', \text{ catalytic plug flow reactor} \quad (8.7)$$

In a batch reactor, let N be the moles of a chemical; the molar change of a chemical is the reaction rate (molar change per unit time per unit volume) times the volume and the time.

$$dN = rV \, dt \quad (8.8)$$

This can be put into a differential equation, too:

$$\frac{dN}{dt} = rV \quad (8.9)$$

If the volume of the batch reactor is constant, then

$$c = \frac{N}{V} \quad (8.10)$$

and the equation becomes

$$\frac{dc}{dt} = r, \text{ batch reactor} \quad (8.11)$$

This equation is similar to the equation for a plug flow reactor, with the time t taking the place of the variable z/u . You can also introduce catalyst into the batch reactor, in which case the reaction rate r is replaced by $\rho_B r'$. In addition, you need an initial condition to go with Eq. (8.11), $c(t = 0) = c_0$. The structure of the problem is the same as that for a plug

flow reactor, and if you can solve one, you can solve the other. So, there is some economy of effort.

Many chemicals are involved, and you need a reaction rate for each reaction occurring. Thus, use $\{c_j\}$ for all the chemical concentrations and $\{r_i\}$ for all the reactions. If the reaction is



then the rate of reaction of B is b/a times the rate of reaction of A , and the rates of reaction of C and D are $-c/a$ and $-d/a$ times the rate of reaction of A . The important point is that the rate of reaction is an algebraic function of the concentrations, temperature, and pressure. Each reaction rate needs to be known specifically for the reaction in question.

Example: Continuous Stirred Tank Reactor

Another type of reactor is known as a continuous, stirred-tank reactor (CSTR). This reactor is a vessel with an inlet stream and an outlet stream, and the contents are constantly being stirred. Engineers often assume that this reactor is so well mixed that the concentration is the same everywhere. The mole balance on this reactor is

$$F_{\text{out}} - F_{\text{in}} = rV \quad (8.13)$$

where V is the volume of the reactor. The molar flow rates are related to the volumetric flow rates, Q , as follows:

$$F = Qc \quad (8.14)$$

When the volumetric flow rates are constant, Eq. (8.13) becomes

$$Q(c_{\text{out}} - c_{\text{in}}) = rV, \text{ continuous stirred tank reactor} \quad (8.15)$$

In a well-mixed reactor (CSTR), the concentration going out of the reactor is the same as the concentration in the reactor. Since the rate expression is a function of c , it is a function of c_{out} . Then Eq. (8.15) becomes an algebraic equation in one variable (here) for c_{out} . You can solve this using either Excel or MATLAB.

USING MATLAB TO SOLVE ORDINARY DIFFERENTIAL EQUATIONS

Simple Example

In MATLAB, you define the problem by means of a function, called an m-file. You then tell MATLAB to solve the differential equation. Thus, you need to prepare an m-file that defines the equation and then call the subroutine 'ode45' to do the integration. This method may seem mysterious at first because you call a subroutine, which in turn calls your m-file. You do not directly call your m-file when solving the differential equation. This process is

illustrated using a single, simple differential equation:

$$\frac{dy}{dt} = -10y, \quad y(0) = 1 \quad (8.16)$$

Integrate this equation from $t = 0$ to $t = 1$. The exact solution can be found by quadrature and is

$$y(t) = e^{-10t} \quad (8.17)$$

Step 1 To use MATLAB, you first construct an m-file that defines the equation:

```
% filename rhs.m
function ydot=rhs(t,y) %compute the rhs of eqns.for any
                        %input values t, y
ydot=-10*y;           %compute rhs
```

Step 2 Then to test the function, issue the command

```
q=rhs(0.2,3)
ydot=-30
q=-30
```

When $y = 3$, the value of the right-hand side should be -30 , so the m-file is correct. In more complicated cases, you might remove all the ‘;’ from the rhs.m so intermediate results will be printed. This simplifies your task when checking the program.

Step 3 Next create an m-file that provides the script to run the problem. These commands can also be typed in the command window, but when you have several commands that you will use over and over, it is more convenient to create a script that can run all the commands with one message from you.

```
% filename simple.m
y0=1;           %set the initial condition to 1
tspan=[0 1];   %set the initial time to zero and the final
                %time to 1
                %call the routine ode45; use the m-file called
                %rhs,
                %start with y=y0 at tspan(1)
                %and integrate until tspan(2)
[t,y]=ode45(@rhs,tspan,y0)
plot(t,y) %plot the solution
```

The results give a table of values for t and y and Figure 8.1.

```
t=      0      0.0078  0.0268  0.0460  0.0659  0.0865  0.1080  ...
y=1.0000  0.9248  0.7647  0.6314  0.5175  0.4209  0.3395  ...
```

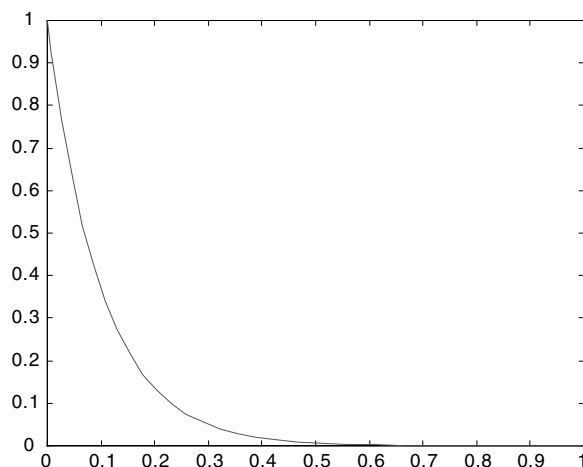


Figure 8.1. Solution to Eq. (8.16) using MATLAB.

Step 4 The validity of your solution depends upon several things. First, you checked the m-file ‘rhs’ and determined that it gave the correct result when you inserted values for t and y . You always have to do this check to show that your work is correct. You depend upon MATLAB to do its job correctly. You can easily check that it used the initial and ending times correctly from the output and can see that the initial condition for y was correct, too. Other than that we accept the MATLAB result on faith. However, you can always solve a problem for which you know the solution, and you try to make that problem as much like your problem as possible. In this case, you did not need to use numerical methods, because you have an exact solution. Thus, you can compare the numerical result to the analytical result (they are the same) and determine that MATLAB did its job correctly. In cases without an analytical solution, you check your m-file ‘rhs’ and then depend upon MATLAB. You can also set a parameter (error tolerance) to improve the result (see Appendix B).

Use of the ‘Global’ Command

In the example above, you solved the problem when the right-hand side was $-10y$, and you simply used the value 10 in the program. There are two other ways to set this value, using the command ‘global’ or by passing k as a parameter. You might want to write this as $-ky$ and solve the problem for several values of k . Because the function only knows what is set inside the function, or what gets transferred to it, you need to learn how to transfer information from the workspace into the function. You will need to make those changes to the code used above. If you use the ‘global’ command, put the variable name in both the workspace and the m-file and assign its value in one place or the other. Then that value is accessible from either the m-file or the workspace.

Step 1 Here, you will set the value of k in the workspace, or calling program, and then it will be available in the m-file.

```
% filename simple2.m %compute the rhs of Eq. (8.16)
global k
k=10; % set k to 10; it is now available to any
      % m-file that has the
      % global k statement in it.
y0=1; % set the initial condition to 1
tspan=[0 1]; % set the initial time to zero and the
             % final time to 1
             % call the routine ode45; use the m-file
             % called rhs2,
             % start with y=y0 at tspan(1) and
             % integrate until tspan(2)
[t,y]=ode45(@rhs2,tspan,y0)
plot(t,y) % plot the solution
```

Step 2 Next, add the global command to the m-file, called rhs2. (Note you added a numerical identifier to both the calling program and m-file to help keep them straight.)

```
% filename rhs2.m
function ydot=rhs(t,y)
      % compute the rhs of Eq. (8.16)
      % for any input values t, y
global k % The m-file can access the numerical
      % value of k, if
      % it has been set to global and given a
      % value elsewhere
disp(k) % display the numerical value of k
      % use this only when testing the function
      % rhs2
ydot=-k*y; % compute rhs
```

Naturally, you expect the same results, which you can verify by running the new code.

Step 3 Check the code as before, with one addition. It is helpful the first time you run the code to print out the variables inside the function to make sure they are accessible. If not, MATLAB will tell you that the variable is not defined. Alternatively, use the debugging techniques described in Appendix B.

Passing Parameters

Still another way to introduce k into the function is to use it as a parameter in the calling argument.

```
% filename simple3.m
k=10; % set k to 10
y0=1; % set the initial condition to 1
tspan=[0 1]; % set the initial time to zero and the
             % final time to 1
```

```

options=[]           % the OPTIONS=[] is a place holder
                    % The symbol [] is made as [] without a
                    % space.
                    % call the routine ode45; use the m-file
                    % called rhs3,
                    % start with y=y0 at tspan(1) and
                    % integrate until tspan(2)
[t,y]=ode45(@rhs3,tspan,y0,options,k)
plot(t,y)           % plot the solution

```

Now the function is:

```

% filename rhs3.m
function ydot=rhs(t,y,k)
                % compute the rhs of eqns. for any input
                % values t, y
disp(k)         % display the numerical value of k; use only
                % for testing of rhs3
ydot=-k*y;      % compute rhs

```

This program gives the same results as given by simple1.m and simple2.m.

Example: Isothermal Plug Flow Reactor

In this section, you will solve the equations for an isothermal plug flow reactor. The first problem is very simple, and is patterned after a problem on the California Professional Engineers License Examination, according to Fogler (2005). Here it is modified. You take a reactor in which components A and C are fed in equimolar amounts, and the following reaction takes place:



You assume that the reaction takes place in the liquid phase and that the volumetric flow rate remains constant even when reaction occurs. The equations are Eq. (8.5) for each species.

$$u \frac{dC_j}{dz} = r_j, \quad j = 1, \dots, 3 \quad (8.19)$$

where C_j is the molar concentration of the j th species, u is the velocity, z is the distance down the tube from the inlet, and r_j is the rate of reaction of the j th species, in mol/vol time. Here the rate of reaction is taken as second order:

$$\text{rate of formation of } B = kC_A^2 \quad (8.20)$$

where the units of k are vol/mol time . The equations for all three species are then

$$u \frac{dC_A}{dz} = -2kC_A^2, \quad u \frac{dC_B}{dz} = +kC_A^2, \quad u \frac{dC_C}{dz} = 0 \quad (8.21)$$

At the inlet you take

$$C_A(0) = 2 \text{ kmol/m}^3, \quad C_B(0) = 0, \quad C_C(0) = 2 \text{ kmol/m}^3 \quad (8.22)$$

and we take $u = 0.5 \text{ m/s}$, $k = 0.3 \text{ m}^3/\text{kmol s}$, and the total reactor length as $z = 2.4 \text{ m}$.

Step 1 The MATLAB program requires you to write a function that defines the right-hand side. The input parameters to the function are the concentrations of all species.

(1) Thus, to solve the problem you use the variables

$$y_1 = C_A, \quad y_2 = C_B, \quad y_3 = C_C \quad (8.23)$$

(2) The function also needs the velocity, u , and the rate constant, k . The distance from the inlet, z , takes the place of time and is the independent variable. (3) The rates of reaction are then evaluated, and the function returns the numerical value of the right-hand side. The code for the function is

```
% rate1.m
% This function gives the right-hand side for a simple
% reactor problem which is isothermal.
function ydot=rate1(VR,y)
% y(1) is CA, y(2) is CB, y(3) is CC
% k=0.3 and u=0.5
CA=y(1);
rate=0.3*CA*CA;
ydot(1)=-2.*rate/0.5;
ydot(2) =+rate/0.5;
ydot(3)=0.;
ydot=ydot';
```

Step 2 You test this m-file by calling it with specific values for y_j , $j = 1, 2, 3$ to ensure that it is correct. Using $y(j)$ for y_j , issue the following commands:

$$y(1) = 0.2; \quad y(2) = 0.3; \quad y(3) = 0.4; \quad \text{rate1}(0.1, y)$$

You get

$$\text{ans} = -0.048, \quad 0.024, \quad 0$$

which agrees with the manual calculations. This is a very important step, because this is where you add value. MATLAB will integrate whatever equations you give it, right or wrong, and only *you* can ensure that the program has solved the right equations.

Step 3 Next, write a code that serves as the driver. This code must (1) set any constants (here they are just put into the function `rate1` for simplicity), (2) set the initial conditions and total reactor length, and (3) call the ode solver.

```
% run_rate1.m
% This is the driver program to solve the simple flow
% reactor example.
% set the initial conditions
y0=[2 0 2]
% set the total volume of the reactor
zspan=[0 2.4]
% call the ode solver
[z y]=ode45(@rate1,zspan,y0)
% plot the result
plot(z,y)
xlabel('length (m)')
ylabel('concentrations (kgmol/m^3)')
legend('A', 'B', 'C')
```

Step 4 The result is shown in Figure 8.2. Now that you have a validated code, you can vary the parameters to see their effect. Figure 8.2 shows the concentrations, but it is difficult to distinguish between the curves when the figure is in black and white. You can plot each concentration with a different symbol by replacing the plot command with the following command.

```
plot(z,y(:,1),'*-',z,y(:,2),'+-',z,y(:,3),'x-')
```

Now each curve is labeled with a different symbol, and the curves are easy to distinguish (see Figure 8.3).

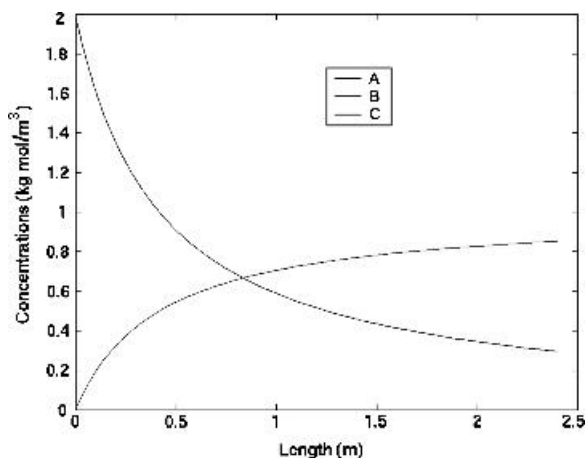


Figure 8.2. Solution to problem posed by Eqs. (8.21)–(8.22).

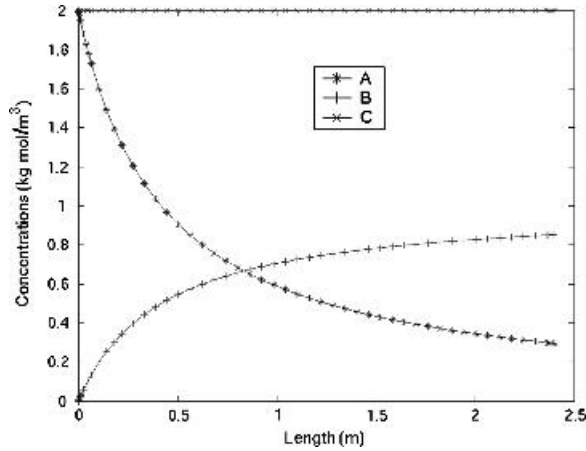


Figure 8.3. Different plot of solution to problem posed by Eqs. (8.21)–(8.22).

Example: Nonisothermal Plug Flow Reactor

Most reactions have heat effects, and this means that reactor models must be able to model the heat transfer as well as the mass balances. The following example models a simple reactor oxidizing SO_2 to form SO_3 (Young and Finlayson, 1973). After some manipulation, the equations are

$$\frac{dX}{dz} = -50R', \quad \frac{dT}{dz} = -4.1(T - T_{\text{surr}}) + 1.02 \cdot 10^4 R' \quad (8.24)$$

where the reaction rate is

$$R' = \frac{X[1 - 0.167(1 - X)]^{1/2} - 2.2(1 - X)/K_{\text{eq}}}{[k_1 + k_2(1 - X)]^2} \quad (8.25)$$

$$\ln k_1 = -14.96 + 11070/T, \quad \ln k_2 = -1.331 + 2331/T, \\ K_{\text{eq}} = -11.02 + 11570/T \quad (8.26)$$

with the parameters: $T_{\text{surr}} = 673.2$, $T(0) = 673.2$, $X(0) = 1$. The variable X is the concentration of SO_2 divided by the inlet concentration, $1 - X$ is the fractional conversion, and T is the temperature in K. The first equation is the mole balance on SO_2 , and the second is the energy balance. The first term on the right-hand side of Eq. (8.24) represents cooling at the wall; the second term there is the heat of reaction.

These equations are appropriate when radial and axial dispersion are not important. Axial and radial dispersion are actually important in this case, as discussed by Young and Finlayson (1973), but the simpler case is a good vehicle for seeing how to include temperature effects in the problem. The reaction rate equation shows that equilibrium can be reached (then $R' = 0$), and the equilibrium constant depends on the temperature. Thus, the conversion depends intimately on the temperature.

Step 1 The MATLAB program requires you to write a function (m-file) that calculates the right-hand side of Eq. (8.24), given the input, z , X , and T , which are the axial location in the reactor and the conversion and temperature at that axial location, respectively. Thus, (1) to solve the problem you use the variables

$$y(1) = X, \quad y(2) = T \quad (8.27)$$

The function will have available to it the axial location, z , and the variables $y(1)$ and $y(2)$. (2) You first move $y(1)$ into X and $y(2)$ into T (for clarity in the program) and then (3) evaluate the rate of reaction. (4) Finally, we calculate the right-hand sides of the differential equations and put them in the vector $ydot$, which is the output from the function.

```
% rateSO2.m for SO2 reaction
function ydot=rateSO2(z,y)
% X is concentration SO2 divided by the inlet concentration
X=y(1);
% T is temperature in degrees K
T=y(2);
k1=exp(-14.96+11070/T);
k2=exp(-1.331+2331/T);
Keq=exp(-11.02+11570/T);
term1=X*sqrt(1-0.167*(1-X));
term2=2.2*(1-X)/Keq;
denom=(k1+k2*(1-X))^2;
rate=(term1-term2)/denom;
ydot(1)=-50*rate;
ydot(2)=-4.1*(T-673.2)+1.02e4*rate;
ydot=ydot';
```

Step 2 You test this m-file by setting the input variables and calling the m-file:

```
y(1)=0.2; y(2)=573.2;
rateSO2(0.1,y)
ans=-0.001148 410.235
```

This agrees with the manual calculations, so the m-file is correct. (To make the comparison easier to do, you can remove the semi-colons from rateSO2. After checking, the semi-colons are added again.) This is a most important step, because this is where you add value. MATLAB will integrate whatever equations you give it, right or wrong, and only *you* can insure that the right equations have been solved.

Step 3 Next you write the code as the driver. This code must set the constants, set the initial conditions for conversion and temperature, decide how far down the reactor to integrate, and then call the ode solver.

```
% run_SO2.m
% set the dimensionless initial conditions
y0=[1 673.2]
```

```
% set the integration range
zspan=[0 1]
% call the solver
[z y]=ode45(@rateSO2,zspan,y0)
% plot the result
plot(z,y(:,1),'r-')
xlabel('dimensionless axial position')
legend('fraction converted')
pause
plot(z,y(:,2),'g-')
xlabel('dimensionless axial position')
legend('temperature (K)')
```

The results are shown in Figure 8.4. The concentration and temperature are plotted in separate graphs because they have such different magnitudes. Now you can change the parameters to see their effect. Change the inlet temperature to 800 and see what happens.

USING FEMLAB TO SOLVE ORDINARY DIFFERENTIAL EQUATIONS

FEMLAB is treated in detail in Chapters 9–11 and Appendix D, but it can also be used to solve reactor problems. The advantage of FEMLAB is that you program with a GUI, so computer errors are less likely. It is still necessary to check your work, though. While the applications in this chapter are all one-dimensional (to compare with MATLAB solutions), it is easy to solve two-dimensional problems, as described in more detail in later chapters. We show here how to solve the same three problems already solved using MATLAB: the simple exponential, Eq. (8.16); the isothermal flow reactor, Eqs. (8.21)–(8.22); and the nonisothermal reactor, Eqs. (8.24)–(8.26).

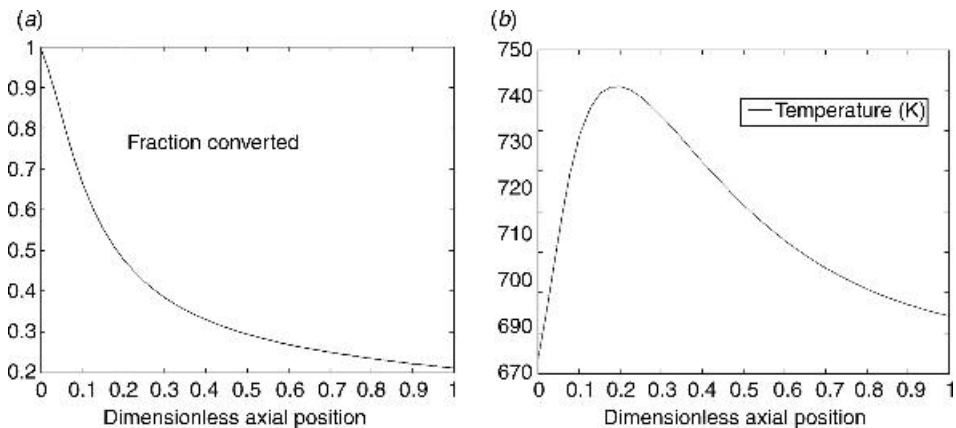


Figure 8.4. Solution to problem posed by Eqs. (8.24)–(8.26).

Simple Example

To solve Eq. (8.16), you first have to draw an analogy between time [in Eq. (8.16)] and space (x) in FEMLAB. The problem can be restated as:

$$\frac{dy}{dt} = -10y, y(0) = 1, \quad u \frac{dy}{dx} = -10y, y(0) = 1 \quad (8.28)$$

where x/u takes the place of t . By setting $u = 1$ you can relate $t \leftrightarrow x$. The equation you have to work with in FEMLAB is the steady convective diffusion equation:

$$u \frac{dy}{dx} = D \frac{d^2y}{dx^2} + f \quad (8.29)$$

Thus, we must choose $D = 0$ and $f = -10^*y$. (For more details about FEMLAB, see Appendix D.)

Step 1 Open FEMLAB and choose 1D (in the top menu), Chemical Engineering Module, then Mass Transfer and convective diffusion (the PDE/Classical/convective diffusion also works). Click OK.

Step 2 Click on the line icon on the left, put the cursor at $x = 0$ and drag to $x = 1$ and click again.

Step 3 Go to Options and then Axes/Grid Settings. Make $x_{\min} = -0.1$ and $x_{\max} = 1.1$. This will make the domain cover most of the screen. Close the window.

Step 4 Click on the triangle to set up the mesh. You will get 15 elements and 16 points. There will be 31 degrees of freedom, because each element has a node at the middle, too; thus there are 15 middle nodes and 16 nodes at the ends of the elements.

Step 5 Choose Physics/Subdomain Settings, and click on subdomain 1. Set the parameters as follows.

$$\delta_{ts} = 0, \quad D = 0, \quad f = -10^*c, \quad u = 1 \quad (8.30)$$

The c is a stand-in for y , and the x is a stand-in for t .

Step 6 Choose Physics/Boundary Settings, and click on 1. Change the boundary condition to 'concentration', and set the value to 1.0; click Apply. This is the value at the left-hand side, that is, $t = 0$. Click on the numeral 2. Change that boundary condition to convective flux, and click OK.

Step 7 Click the '=' to have the computer solve the equations.

Step 8 Choose Post Processing/Domain Plot Parameters. Choose the Line/Extrusion, Expression as c , and click Apply. You will get Figure 8.5.

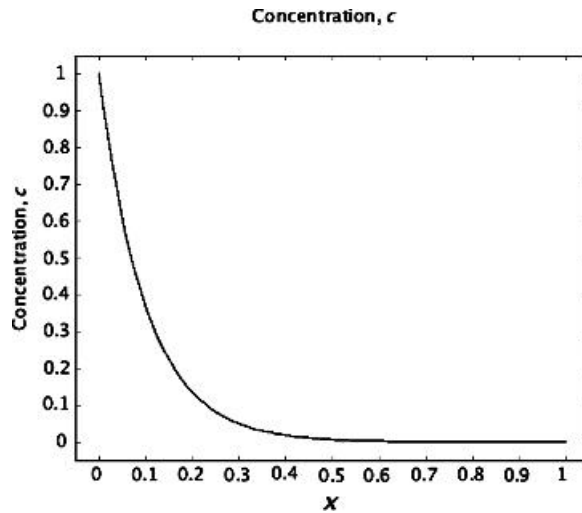


Figure 8.5. Solution to Eq. (8.16) using FEMLAB.

Step 9 Choose the diskette image to export the figure. Usually you would export it as a jpeg figure, which is portable to many different programs. Close the plotting window.

Step 10 Create a new mesh (called remesh) by choosing the icon with a triangle inside a triangle; either click on '=', or the '=' with a circle (to restart from the last solution). If the plots of the solution are substantially the same as the 15-element case, the mesh is good enough.

You see that the solution shown in Figure 8.5 looks the same as that obtained using MATLAB, Figure 8.1. You need to be careful, though, because the solution using FEMLAB is only as good as the mesh, and you have to try at least two meshes to see if it matters. MATLAB, by contrast, adjusts the number of points from one end to the other to maintain the accuracy (which you can reset if desired). Thus, the MATLAB solution comes with a guaranteed accuracy, whereas FEMLAB does not. FEMLAB has other advantages, however, which are demonstrated in other chapters.

Example: Isothermal Plug Flow Reactor

Next solve the reactor problem posed by Eqs. (8.21)–(8.22). Now you must solve three equations together. Repeat steps 1–4 as listed in the simple exponential example, except make the length 2.4.

Step 5 Choose the Multiphysics/Model Navigator option to insert more equations. Remove all equations to clear the problem. Select the Chemical Engineering/Mass balance/Convective diffusion/Steady state and change the dependent variable from c to ca . Click Add. Repeat this operation twice more, using dependent variables cb and cc . Now you have three equations, with dependent variables ca , cb , and cc .

Click OK. The correspondence between the computer variables and the reactor variables is:

Computer Starts With	You Changed to	Reactor
u	ca	C_A
$u2$	cb	C_B
$u3$	cc	C_C
x	x	z

Thus, you are solving the following equations (where you call the velocity vel):

$$vel \frac{dca}{dz} = -2k \cdot ca^2, \quad vel \frac{dcb}{dz} = +k \cdot ca^2, \quad vel \frac{dcc}{dz} = 0 \quad (8.31)$$

Step 6 Choose Options/Constants and insert k and 0.3, and u and 0.5. These values are useable by any expression in the computer program. Click OK.

Step 7 Choose Options/Expressions/Subdomain Expressions. Select domain 1 (the only one) and create a variable named $rate$ and set it equal to $k \cdot ca \cdot ca$. The variable $rate$ can be used elsewhere in the computer program. Click OK.

Step 8 Choose Multiphysics and select the first equation. Next, set the terms in the equation and boundary conditions.

Step 9 Choose Physics/Subdomain Settings. Set the parameters to

$$\delta_{ts} = 0, \quad D = 0, \quad f = -2 \cdot rate, \quad u = vel \quad (8.32)$$

Step 10 Choose Physics/Boundary Settings, and select boundary 1 (the left-hand side). Set the concentration to 2. Select boundary 2 (the right-hand side) and set the boundary condition to 'convective flux'. This has the effect of setting the value of $ca=2$ at the left-hand side, which is the reactor inlet. Click OK.

Step 11 Repeat steps 8–10 for the other two equations. For the equation for cb , use

$$f = rate \quad (8.33)$$

while for the cc equation use $f = 0$.

Step 12 To solve, click on the equal sign, or choose Solve = under the Solve Menu.

Step 13 Under Postprocessing, choose Domain Plot Parameters. In the general tab, click on the 'keep current plot'. This will allow you to plot all the curves on the same graph. Click the 'Line Extrusion' tab, set the variable to be plotted to ca and click Apply. Without destroying or touching the figure, set the variable to be plotted to cb and click Apply. Repeat with cc . Then all three plots are on the same figure. In the General tab,

insert into the boxes Title/Axis: CA, CB, CC; second line: axial position (m); third line: concentration (kmol/m³). Click Apply and export the figure as a jpeg figure with a name you choose. The basic figure is shown in Figure 8.6. If you want, you can put the figure into a drawing program and add other features.

Example: Nonisothermal Plug Flow Reactor

Consider next the problem posed by Eqs. (8.24)–(8.26). Because there are two variables, you need two equations. Repeat steps 1–4 from the simple exponential example. The first equation is the convective diffusion equation, cd , and the variable name is c .

Step 5 Choose Multiphysics/Model Navigator and add Chemical Engineering/Energy/Convection and conduction. The equation is called cc , and the variable is T . After adding the equation, be sure you have selected the cd equation for c .

Step 6 Choose Physics/Subdomain Settings and select 1. Insert the following values and click OK:

$$\delta_{ts} = 0, \quad D = 0, \quad R = -50 \cdot \text{rate}, \quad u = 1 \quad (8.34)$$

Step 7 Choose Physics/Boundary Settings and select boundary 1 (the point on the left-hand side). Set the concentration to 1.0. Select boundary 2 (the right-hand side) and change the boundary condition to ‘convective flux’. Click OK.

Step 8 Choose the Multiphysics menu and select the convection and conduction equation.

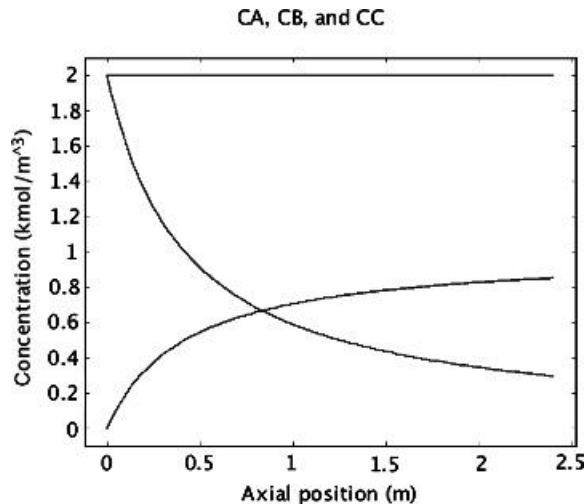


Figure 8.6. Solution to problem posed by Eqs. (8.21)–(8.22) using FEMLAB.

Step 9 Choose Physics/Subdomain Settings and select subdomain 1. Insert the following values.

$$\begin{aligned}\delta_{ts} &= 0, k = 0, \rho = 1, C_p = 1 \\ Q &= 1.02e4 * rate - 4.1 * (T - 673.2) \\ u &= 1, h_i N_{D,i} \text{ inactive}\end{aligned}\tag{8.35}$$

Step 10 This step is very important! While the Physics/Subdomain window is open, select the Init tab. Choose an initial value of $T = 673.2$. The reaction rate constants and equilibrium expressions are not defined for $T = 0$, which is the default starting point. If you skip this step, the program will not work.

Step 11 Choose Physics/Boundary Settings and select boundary 1. Set the boundary condition to 'temperature' and the value to 673.2. Select boundary 2; set the boundary condition to 'convective flux' and click OK.

Step 12 Choose Options/Expressions/Subdomains/. Set the following expressions:

$$\begin{aligned}k_1 &= \exp(-14.96 + 11070/T) \\ k_2 &= \exp(-1.331 + 2331/T) \\ K_{eq} &= \exp(-11.02 + 11570/T) \\ term1 &= c * \text{sqrt}[1 - 0.167 * (1 - c)] \\ term2 &= 2.2 * (1 - c) / K_{eq} \\ denom &= [k_1 + k_2 * (1 - c)]^2 \\ rate &= (term1 - term2) / denom\end{aligned}$$

Step 13 Solve the problem by clicking on the '=' sign.

Step 14 Choose Postprocessing, Domain Plot Parameters, and click on Line/Extrusion. The variable already selected is the concentration c . Click subdomain 1 and Apply. Save

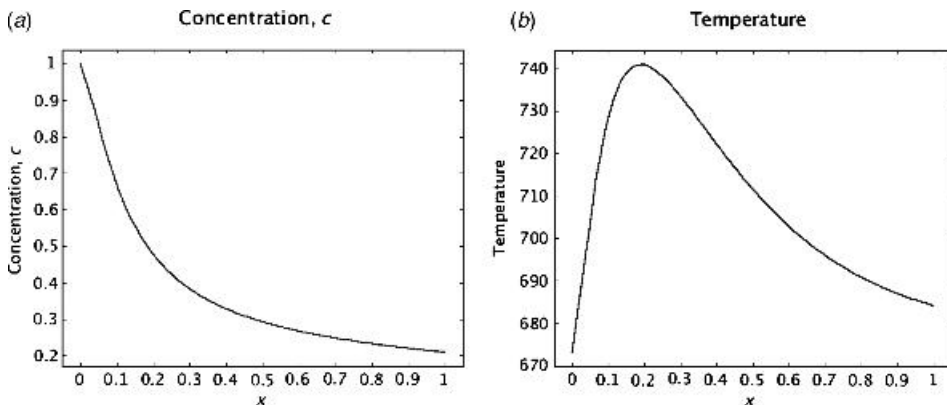


Figure 8.7. Concentration (a) and temperature (b) for SO_2 reactor, Eqs. (8.24)–(8.26)

the figure (see Figure 8.7a). Then go back and plot the temperature in the same way, and save it (see Figure 8.7b).

It is useful to see what part of the reactor is doing the most work. You can plot the rate of reaction by typing in the word *rate* in the same box where c and T appeared. The rate of reaction is shown in Figure 8.8a, and the equilibrium constant is shown in Figure 8.8b. (Remember that K_{eq} depends upon temperature, which changes with axial position.)

When using FEMLAB to run nonisothermal reactor problems, sometimes you run into difficulties. If you click on the '=' sign, the program might indicate error messages such as 'stepsize too small' or 'singular matrix' or 'internal error in numerical routines.' The first thing to do is make the problem easier to solve; if the program solves the easier problem, then your setup is probably correct. In this example, you can set the *rate* to c . You could also remove the heat of reaction and make the temperature constant. This tests the program without the complications of the exponential terms. Since the kinetic constants change rapidly with temperature, if the temperature approximation is bad, the rates could be very large or very small, and essentially unrealistic.

Suppose you do all this, and the program still does not work. You might first suspect a problem with the temperature. Then, you can use the following strategy. Go to Physics/Subdomain Settings for the temperature equation and change the number 1.02×10^4 to β . Go to Solver Manager and choose 'parameter nonlinear.' Call the parameter β and set its values to $[0 \ 1000 \ 3000 \ 1.02 \times 10^4]$. Then the program will solve first for $\beta = 0$. It will also solve for the derivative of the solution with respect to β . Next it solves for the case of $\beta = 1000$, using a linear extrapolation from the case for $\beta = 0$. This continues in sequence until the last solution is solved for 1.02×10^4 . This is always a good strategy when difficulties arise, and the more you know about the problem the more easily will you find the solution. Appendix D gives more details about this procedure.

[Note to experienced users: Sometimes FEMLAB has difficulty with plug flow reactor problems, when MATLAB did not, because FEMLAB is solving them as boundary value problems, whereas MATLAB solves them as initial value problems. They are actually initial value problems; you fool FEMLAB by setting the diffusion coefficient to zero, and the thermal conductivity to zero. The other difficulty is that the kinetic expression makes huge changes when the temperature changes. Anytime you have to iterate to find

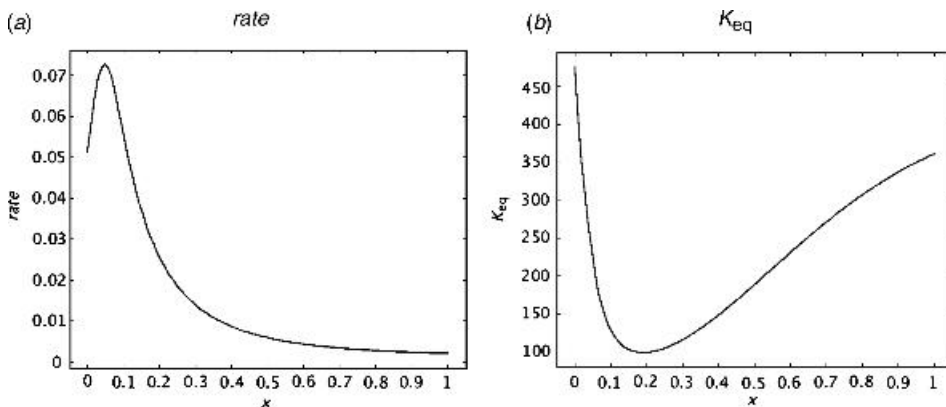


Figure 8.8. Rate of reaction (a) and K_{eq} (b) for SO_2 reactor, Eqs. (8.24)–(8.26).

a solution (which is commonly required for boundary value problems), the values of temperature during the iterations may become too large or too small and the iteration process may never recover. This is why you might change β slowly in order to reach a solution. See Chapter 9 for more information about boundary value problems. The next version of FEMLAB is expected to have equations appropriate for plug flow reactors.]

REACTOR PROBLEMS WITH MOLE CHANGES AND VARIABLE DENSITY

The examples shown above are for liquid streams or dilute gaseous streams where the density does not change drastically. Sometimes the density does change, though, and such problems are not much harder to solve numerically than their constant-density counterparts. The governing equation is Eq. (8.2), here expressed as a differential equation:

$$\frac{dF_j}{dV_R} = r_j, \quad j = 1, \dots, 3 \quad (8.36)$$

where F_j is the molar flow rate of the j th species, V_R is the volume of reactor (from the inlet), and r_j is the rate of reaction of the j th species. Let us take the same reaction, Eq. (8.18), and reaction rate, Eq. (8.20). Putting these together gives the differential equations to be solved:

$$\frac{dF_A}{dV_R} = -2kC_A^2, \quad \frac{dF_B}{dV_R} = +kC_A^2, \quad \frac{dF_C}{dV_R} = 0 \quad (8.37)$$

The equations are differential equations for the molar flow rates, F_A , F_B , and F_C , but the rates of reaction are expressed as concentration, C_A . It is possible to derive the concentration of each chemical from the molar flow rates, and this section illustrates how to do that in MATLAB and FEMLAB. In a gas phase, the total concentration is governed by the perfect gas law, although more complicated equations of state are possible:

$$C_T = \frac{P}{RT} \quad (8.38)$$

Thus, at any point in the reactor, if you know the pressure and temperature, you can find C_T . The pressure is determined by fluid mechanics and the pressure drop, and the temperature is determined by the energy equation. Then the concentration of a chemical is the mole fraction times the total concentration; for example

$$C_j = x_j C_T, \quad \text{and the mole fraction is} \quad (8.39)$$

$$x_j = \frac{F_j}{F_A + F_B + F_C}, \quad j = A, B, \text{ or } C$$

If the temperature and pressure are constant, the total concentration is the total concentration at the inlet, which is taken here to be 5 kmol/m^3 . Thus, the flow rate is

$$F_A + F_B + F_C = \dot{V} C_T \quad (8.40)$$

The problem is summarized here as

$$\begin{aligned}\frac{dF_A}{dV_R} &= -2kC_A^2, & \frac{dF_B}{dV_R} &= +kC_A^2, & \frac{dF_C}{dV_R} &= 0 \\ F_A(0) &= 2 \text{ kmol/s}, & F_B(0) &= 0, & F_C(0) &= 2 \text{ kmol/s} \\ C_A &= \frac{F_A}{F_A + F_B + F_C} C_T\end{aligned}\quad (8.41)$$

The MATLAB program requires you to write a function (m-file) that calculates the right-hand side, given the input, which are the flow rates of all species. Thus, to solve the problem you would use the variables

$$y(1) = F_A, \quad y(2) = F_B, \quad y(3) = F_C \quad (8.42)$$

The function will have available to it the accumulated reactor volume V_R , and the local value of $y(j)$, $j = 1, \dots, 3$ at the same reactor volume. You then calculate the concentrations at that reactor volume using Eq. (8.43):

$$C_A = \frac{y(1)}{y(1) + y(2) + y(3)} C_T \quad (8.43)$$

The rates of reaction are then evaluated, and the function returns the numerical value of the right-hand side. You can complete this example by doing Problem 8.7 at the end of the chapter.

The same considerations apply when using FEMLAB. The variables being solved for are now F_A , F_B , and F_C . In Options/Expressions/Subdomain Expressions you define equations for new variables ca , cb , and cc representing Eq. (8.42), and then use the same reaction rate expression (in terms of ca). The rest of the solution proceeds as before:

$$\begin{aligned}ca &= CT^*F_A/(F_A + F_B + F_C), & cb &= CT^*F_B/(F_A + F_B + F_C) \\ cc &= CT^*F_C/(F_A + F_B + F_C)\end{aligned}\quad (8.44)$$

CHEMICAL REACTORS WITH MASS TRANSFER LIMITATIONS

When a reaction occurs on a catalyst particle, it is necessary for the reactants to get to the catalyst. Sometimes this transfer is hindered by mass transfer, and then the concentration near the catalyst is not the same as it is in the bulk stream. This phenomena is demonstrated with the problem posed in Eqs. (8.21)–(8.22), except that now the rate of reaction is evaluated at a concentration that is near the catalyst. The concentration C_s is the concentration on the surface, expressed as the kmol of a species per volume of the catalyst. In your Chemical Reactor Design courses, you will learn various ways to represent this concentration. The net effect, however, is that Eq. (8.21) is changed to Eq. (8.45):

$$u \frac{dC_A}{dz} = -2k_s C_{A,s}^2, \quad u \frac{dC_B}{dz} = +k_s C_{A,s}^2, \quad u \frac{dC_C}{dz} = 0 \quad (8.45)$$

You will be solving for $C_A(z)$, but you cannot evaluate the rate of reaction in Eq. (8.45) because you do not know $C_{A,s}$. You need a mass balance relating the rate of mass transfer to the catalyst to the rate of reaction. One form of that is Eq. (8.46), where k_m is a mass transfer coefficient in units of m/s, determined from correlations derived in fluid mechanics and mass transfer courses, a is the surface area exposed per volume of the reactor (m^2/m^3), and k_s is a rate of reaction rate constant (here $\text{m}^3/\text{kmol s}$). Other formulations are possible, too:

$$k_m a (C_A - C_{A,s}) = k_s C_{A,s}^2 \quad (8.46)$$

Looking closely at Eqs. (8.45)–(8.46), you can see that, in order to solve the differential equations in Eq. (8.45), you must solve Eq. (8.46) *at every position* z . Thus, this is a problem that combines ordinary differential equations with nonlinear algebraic equations.

MATLAB easily handles these kinds of problems. Basically you call a routine to integrate the ordinary differential equations (e.g., `ode45`). You construct a right-hand side function (m-file) to evaluate the right-hand side. The input variables are z and the three concentrations, and the output variables are the three derivatives. Take C_A and solve Eq. (8.46) using either ‘`fzero`’ or ‘`fsolve`.’ Then you have $C_{A,s}$ at this location, z . You can evaluate the rates of reaction in Eq. (8.45) and put them in the output from the m-file.

Step 1 You first construct an m-file to evaluate Eq. (8.46):

```
function y=mass_rxn(CAs,CA)
global k km
% mass_rxn.m
% this m-file evaluates the mass transfer equations
% km in the program includes the a
% the first entry is the guess of CAs and
% the second entry is the fluid concentration
y=km*(CA-CAs)-k*CAs*CAs;
```

Step 2 Test this function using the `feval` function:

```
global k km
k=0.3; km=0.35
feval('mass_rxn',0.5,0.6)
ans=-0.143
```

This is correct.

Step 3 Next, test the use of the ‘`fzero`’ command:

```
CAGuess=0.5; CA=0.5;
OPTIONS=[]
CAs=fzero(@mass_rxn,CAGuess,OPTIONS,CA);
```

The result is 0.37771, which is correct.

Step 4 Next, construct an m-file, `ratel_mass.m`, to evaluate the right-hand sides of Eq. (8.45). This m-file is almost the same as that shown following Eq. (8.23), except that a global command is used so the parameters can be changed easily. The only significant change to this m-file from that used earlier is the use of `fzero` to find CAs, and then using CAs rather than CA to evaluate the rate of reaction.

```
% ratel_mass.m
% This function gives the right-hand side for a
% reactor problem which is isothermal but has
% mass transfer effects
function ydot=ratel_mass(VR,y)
global k vel km
% y(1) is CA, y(2) is CB, y(3) is CC
CA=y(1);
% solve the mass transfer problem
% the first entry is the guess of CAs and
% the second entry is the fluid concentration
OPTIONS=[]
CAguess=CA;
CAs=fzero(@mass_rxn,CAguess,OPTIONS,CA);
rate=k*CAs*CAs;
ydot(1)=-2.*rate/vel;
ydot(2)=+rate/vel;
ydot(3)=0.;
ydot=ydot';
```

Step 5 The last m-file is the one to run the problem, which you do using the value $k_m a = 0.2$. The only changes to the program `run_ratel` following Eq. (8.23) are (1) the use of 'global' to set the parameters and (2) changing the name of the right-hand side function from `ratel` to `ratel_mass`.

```
% run_ratel_mass.m
% This is the driver program to solve the simple flow reactor
% example.
global k vel km
k=0.3; vel=0.5; km=0.2;
% set the initial conditions
y0=[2 0 2]
% set the total volume of the reactor
zspan=[0 2.4]
% call the ode solver
[z y]=ode45(@ratel_mass,zspan,y0)
% plot the result
plot(z,y(:,1),'*- ',z,y(:,2),'+- ',z,y(:,3),'x-')
% or plot(z,y)
xlabel('length (m)')
ylabel('concentrations (kgmol/m^3)')
legend('A', 'B', 'C')
```

Step 6 As a check, you run this program while changing the `rate1_mass.m` program to use

```
%CAs=fzero(@mass_rxn,CAguess,OPTIONS,CA);
rate=k*CA*CA;
```

instead of

```
CAs=fzero(@mass_rxn,CAguess,OPTIONS,CA);
rate=k*CAs*CAs;
```

This has the effect of removing the mass transfer limitation, and the result should be the same as shown in Figure 8.3, and it is the same. The code is returned to its proper form, and the program `run_rate1_mass` calculates the output (see Figure 8.9). With mass transfer resistance included, the outlet concentration of B is 0.61. When there was no mass transfer limitation, the outlet concentration of B was 0.85. Thus, the reactor is not able to produce as much product, and a bigger reactor is required.

This same type of problem can arise with multiple concentrations or with one concentration and temperature. Then you would need to replace the `fzero` command with the `fsolve` command (from the Optimization Toolbox). You might need to experiment to provide good initial guesses for the concentrations and temperatures, because Eq. (8.46) would be turned into a set of nonlinear algebraic equations, and the solution is very sensitive to the temperature variable.

CONTINUOUS STIRRED TANK REACTORS

Equation (8.15) gave the mass balance for a CSTR. A similar equation can be written as an energy balance. This example considers a CSTR in which a first-order reaction occurs,

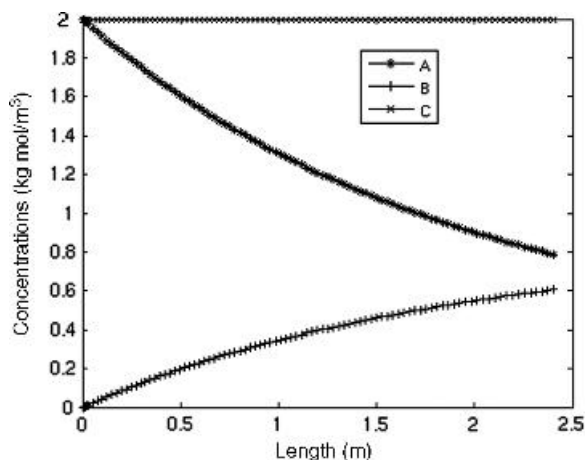


Figure 8.9. Solution to Eqs. (8.45)–(8.46) when mass transfer is important.

but the temperature also changes owing to the heat of reaction. The equations to be solved are:

$$\begin{aligned}\frac{Q}{V_R}(1 - c) &= c \exp[\gamma(1 - 1/T)] \\ \frac{Q}{V_R}(1 - T) &= -\beta c \exp[\gamma(1 - 1/T)]\end{aligned}\quad (8.47)$$

The left-hand sides are the flow rate times a concentration or temperature difference between the input and output, divided by the volume. The equations have been normalized by the inlet concentration and temperature. The right-hand sides are the rate of reaction and the rate of energy generation due to reaction, respectively.

The case described by Eq. (8.47) is for an adiabatic reactor. When the reactor is adiabatic, the equations can be combined by multiplying the first equation by β and adding it to the second equation; then the right-hand side vanishes:

$$\beta(1 - c) + (1 - T) = 0 \quad (8.48)$$

This equation can be solved for T :

$$T = 1 + \beta(1 - c) \quad (8.49)$$

Now the mass balance can be considered a single equation in one unknown (c):

$$\frac{Q}{V_R}(1 - c) = c \exp[\gamma(1 - 1/\{1 + \beta - \beta c\})] \quad (8.50)$$

Solution Using Excel

The first set of parameters used is $Q/V_R = 8.7$, $\beta = 0.15$, $\gamma = 30$. The simple spreadsheet is shown in Figure 8.10; cells are set for T , which is calculated from the c . You can use either Goal Seek or Solver to find the value of c that makes the function zero.

Solution Using MATLAB

MATLAB allows you to extend the method to more complicated situations when Excel does not work.

	A	B	C
1	beta	0.15	
2	gamma	30	
3	flowvel	8.7	
4	c	0.731075498	
5	T	1.040338675 = 1+B1*(1-B4)	
6	rate	2.339642834 = B4*EXP(B2*(1-1/B5))	
7	function	3.34879E-07 = B3*(1-B4)-B6	

Figure 8.10. Spreadsheet solving Eq. (8.50).

Step 1 First, construct an m-file that represents Eq. (8.50):

```
% rate_T
function fn=rate_T(c)
global beta gamma flowvol
T=1+beta*(1-c)
rate=c*exp(gamma*(1-1/T))
fn=flowvol*(1-c)-rate
```

Step 2 Test it using the command

```
>> rate_T(0.5)
T=1.0750
rate=4.0547
fn=0.2953
ans=0.2953
```

The answer is correct.

Step 3 To solve the problem, type the following commands into the command window:

```
>> fzero(@rate_T,0.5)
ans=0.73107565193641
```

This is the same solution as that obtained using Excel, as expected.

CSTR With Multiple Solutions

For a different set of parameters, the CSTR can have more than one solution. For this problem, the solutions all lie between 0 and 1, because the concentration has been normalized by the inlet value, where the normalized concentration is 1.0, and the reaction uses up the material. Which solution you get depends upon the initial guess of c . Use Excel to solve the problem when $Q/V_R = 25$, $\beta = 0.25$, keeping $\gamma = 30$. Successive trials led to the results shown in Table 8.1.

Solutions to Multiple Equations Using MATLAB

When two or more variables must be found, as in Eq. (8.47), a solution can be found to make both the equations zero without rearrangement, like that to produce Eq. (8.50).

TABLE 8.1. Multiple Solutions to Eq. (8.50) when $Q/V_R = 25$, $\beta = 0.25$, $\gamma = 30$

Initial Guess of c	Final Result
0	0.0863
0.5	0.5577
1.0	0.9422

Step 1 You can use the following MATLAB m-file along with the fsolve command (available in the Optimization Toolbox):

```
% rate_T2
function fn=rate_T2(y)
global beta gamma flowvol
c=y(1);
T=y(2);
rate=c*exp(gamma*(1-1/T));
fn(1)=flowvol*(1-c)-rate;
fn(2)=flowvol*(1-T)+beta*rate;
```

Step 2 Test the m-file with the following commands:

```
>> feval(@rate_T2,[0.5 1.1])
ans=-3.29556351331856 0.27683452699778
```

This answer is correct.

Step 3 Then in the command window, issue the following commands:

```
global beta gamma flowvol
beta=0.25; gamma=30; flowvol=25;
fsolve(@rate_T2,[0.0 1.25])
```

With this initial guess, the program returns the first answer in Table 8.1. With an initial guess of [0.5 1.125], it returns the second answer. With an initial guess of [1 1], it returns the third answer. Such problems will not always converge, and convergence is harder for larger β and γ . As the convergence becomes harder, it is more and more crucial to provide a reasonable starting guess.

TRANSIENT CONTINUOUS STIRRED TANK REACTORS

Reactors do not always run at steady state. In fact, many pharmaceuticals are made in a batch mode. Such problems are easily solved using the same techniques presented above because the plug flow reactor equations are identical to the batch reactor equations. Even CSTRs can be run in a transient mode, and it may be necessary to model a time-dependent CSTR to study the stability of steady solutions. When there is more than one solution, one or more of them will be unstable. Thus, this section considers a time-dependent CSTR as described by Eq. (8.51):

$$V \frac{dc'}{dt'} = Q(c' - c'_{in}) - V k_0 c' \exp(-E/RT)$$

$$[\phi(\rho C_p)_f + (1 - \phi)(\rho C_p)_s] \frac{dT'}{dt'} = -(\rho C_p)_f Q(T - T'_{in}) + (-\Delta H_{rxn}) V k_0 c' \exp(-E/RT)$$

(8.51)

The variables are

$$\begin{aligned}
 V &= \text{reactor volume, } Q = \text{volumetric flow rate} \\
 c &= \text{concentration, } t = \text{time} \\
 k_0 &= \text{reaction rate constant, } E = \text{activation energy} \\
 \phi &= \text{void fraction, } \rho = \text{density, } C_p = \text{heat capacity per mass} \\
 &\text{subscript f for fluid, s for solid} \\
 -\Delta H_{\text{rxn}} &= \text{heat of reaction}
 \end{aligned} \tag{8.52}$$

The nondimensional form of these equations is

$$\begin{aligned}
 \frac{dc}{dt} &= (1 - c) - c \cdot Da \cdot \exp[\gamma(1 - 1/T)] \\
 Le \frac{dT}{dt} &= (1 - T) + \beta \cdot c \cdot Da \cdot \exp[\gamma(1 - 1/T)]
 \end{aligned} \tag{8.53}$$

The parameter Le is a Lewis number, and it includes the heat capacity of the system. The Da is a Damköhler number and includes the rate of reaction. The parameters are taken as

$$\beta = 0.15, \gamma = 30, Da = 0.115, Le = 1080, c(0) = 0.7, T(0) = 1 \tag{8.54}$$

Step 1 To integrate these equations using MATLAB, you need an m-file to compute the right-hand side, and then an m-file to call 'ode45'. The right-hand side function is

```
% rate_limit.m
% solve for particle with limit cycle
function ydot=rate_limit(t,y)
global beta gamma Damk Lewis Tin
c=y(1);
T=y(2);
rate=c*Damk*exp(gamma*(1-1/T));
ydot(1)=1-c-rate;
ydot(2)=Tin-T+beta*rate;
ydot(2)=ydot(2)/Lewis;
ydot=ydot';
```

Step 2 After checking this code you can run it using the following m-file.

```
% run_rate_limit
global beta gamma Damk Lewis Tin
beta=0.15
gamma=30
Lewis=1080
Damk=0.115
Tin=1
tic
[t y]=ode45('rate_limit',[0 2],[0.7 1.0])
toc
```

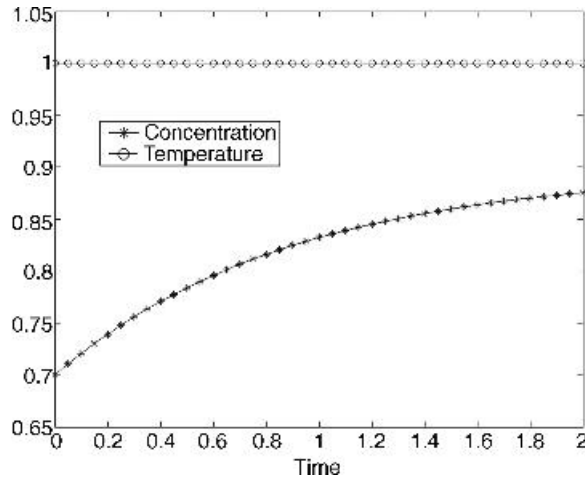


Figure 8.11. Transient CSTR, up to $t = 2$, elapsed time = 0.024 s.

```
plot(t,y(:,1),'*-k',t,y(:,2),'o-k')
xlabel('time')
legend('concentration','temperature')
```

The result is shown in Figure 8.11. It looks like steady state is achieved by the time that $t = 2$. This is not true, however. Integrate to $t = 1000$ and look at the results in Figure 8.12. It still has not reached steady-state. The reason is that the temperature responds much more slowly than does the concentration. Thus, the concentration comes to a steady-state value appropriate to the current temperature, but then the temperature keeps changing and the concentration must change to keep up. Notice the very rapid change of c from the initial value of 0.7 to about 0.9 in Figure 8.12. This is because the value of $c = 0.7$ was not appropriate for a temperature of 1.0. In mathematical terms, the time response of the two variables is very different (the eigenvalues of the equation are widely separated), and the system is called stiff. See Appendix F for more discussion about stiff equations.

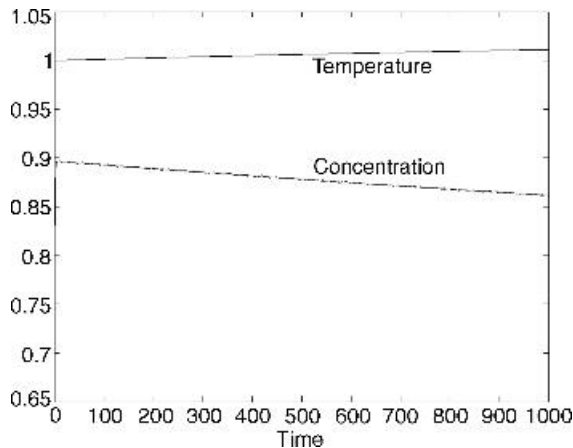


Figure 8.12. Transient CSTR, up to $t = 1000$, elapsed time = 1.22 s.

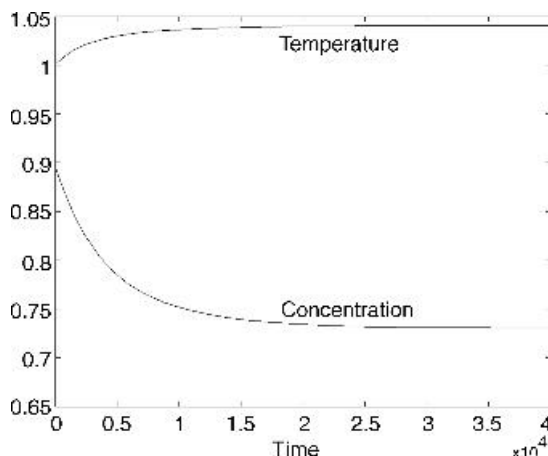


Figure 8.13. Transient CSTR, up to $t = 40,000$, elapsed time = 0.062 s.

MATLAB has some solvers especially suited to stiff problems. You merely change from `ode45` to `ode15s`. When integrating to $t = 1000$, you obtain Figure 8.12, but it only takes 0.040s. Thus, the solution is 30 times faster. Since steady state still has not been reached, integrate to $t = 40,000$. Figure 8.13 shows that steady state has been reached. Now the computation times are 62.2 s for `ode45` and 0.062 s for `ode15s`, giving `ode15s` a speed-up factor of over 1000. The difference is made more dramatic if you remove the `[t y] =` in front of the call to `ode45` and `ode15s`. Then the solution is plotted as it is calculated, and the difference in speed is apparent.

Next you change the parameter Le from 1080 to 0.1 and integrate to $t = 100$ using `ode15s`.

Step 1 The code is shown below:

```
% run_rate_limit
global beta gamma Damk Lewis Tin
beta=0.15
gamma=30
Lewis=0.1
Damk=0.115
Tin=1
tic
[t y]=ode15s('rate_limit',[0 100],[0.7 1.0])
toc
plot(t,y(:,1),'-k',t,y(:,2),'-k')
xlabel('time')
pause
plot(y(:,1),y(:,2),'-k')
xlabel('concentration')
ylabel('temperature')
```

The solution is shown in Figure 8.14. The pattern seems to be repeated over and over.

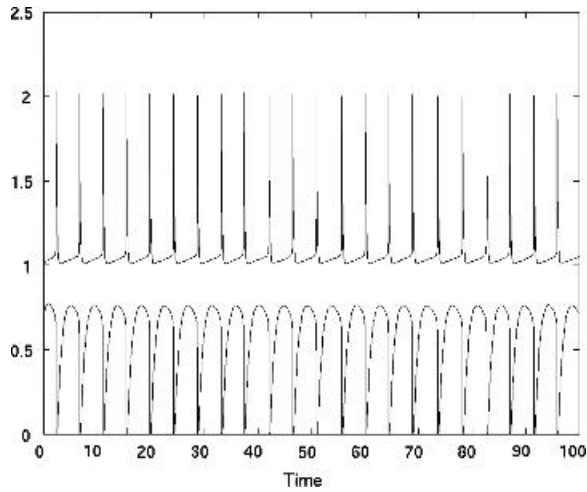


Figure 8.14. Transient CSTR, $Le = 0.1$, integration to $t = 100$, elapsed time = 5.67 s.

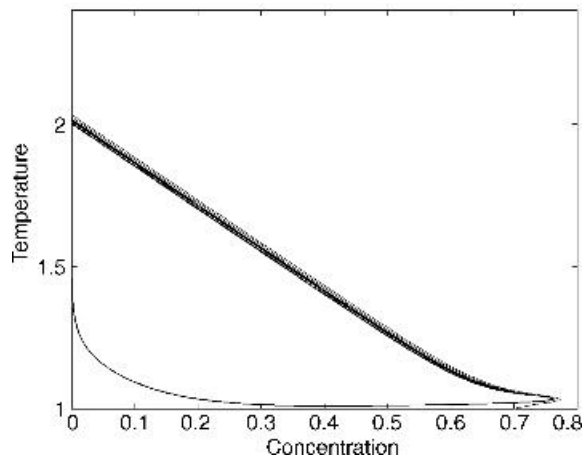


Figure 8.15. Limit cycle display of Figure 8.14.

If the temperature is plotted vs the concentration at the same time, the repeated pattern is apparent (see Figure 8.15). This problem exhibits what is called a limit cycle. This is also a good problem to run without the $[t \ y] =$ in front of the `ode15s`; then you see the solution develop before your eyes.

CHAPTER SUMMARY

In this chapter you developed the equations for different reactors: plug flow, batch, and CSTR. You learned how to integrate ordinary differential equations using MATLAB, and then applied those skills to reactor problems in plug flow reactors (both isothermal and non-isothermal). You also learned to do the same thing using FEMLAB.

Complications arose when the moles or total concentration changed, or when mass transfer effects were important, and you learned how to incorporate those complications. Finally, CSTRs were studied using both MATLAB and Excel in cases where the solution is steady, where multiple steady solutions exist, and you learned to use MATLAB when the solution is time-dependent and the problem is stiff, leading to limit cycles.

PROBLEMS

- 8.1.** Solve for the concentration distribution in a plug flow reactor governed by the following equations. (1) Use MATLAB; (2) use FEMLAB:

$$Q \frac{dc}{dV_R} = -R(c), c(0) = c_{in} \quad (8.55)$$

$$R(c) \equiv \frac{kc}{c+v}, Q = 50, c_{in} = 2, V = 2000 = \text{total volume}, \\ k = 0.198, v = 3.8 \quad (8.56)$$

- 8.2.** Solve for the concentration distribution in a plug flow reactor with the following reaction rate expression and parameters. (1) Use MATLAB; (2) use FEMLAB:

$$R(c) \equiv \frac{kc}{(1+K_1c)^2} \quad \text{and} \quad Q = 2.2, c_{in} = 1, V = 50 = \text{total volume}, \\ k = 2, K_1 = 4 \quad (8.57)$$

- 8.3.** A CSTR is governed by the following equation:

$$Q(c - c_{in}) = -VR(c) \quad (8.58)$$

- (1) Solve for c when

$$R(c) \equiv \frac{kc}{c+v}, Q = 50, c_{in} = 2, V = 2000, k = 0.198, v = 3.8$$

- (2) Solve when

$$R(c) \equiv \frac{kc}{(1+K_1c)^2} \quad \text{and} \quad Q = 2.2, c_{in} = 1, V = 50, k = 2, K_1 = 4$$

- 8.4.** The problem posed by Eqs. (8.21)–(8.22) can be solved for a CSTR, too. In that case the equation is multiplied by the cross-sectional area of the pipe to obtain

$$u \frac{dC_A}{dz} = -2kC_A^2 \rightarrow uA \frac{dC_A}{dAz} = -2kC_A^2 \rightarrow Q \frac{dC_A}{dV_R} = -2kC_A^2 \rightarrow \\ Q(C_A - C_{A,in}) = -2VkC_A^2 \quad (8.59)$$

Take the cross-sectional area as 0.001 m^2 and solve for the same conditions as in Eq. (8.22).

- 8.5.** The growth rate for cells in a bioreactor can follow a variety of growth equations. Monod kinetics is represented by part (1) of Problem 8.3. A substrate-inhibition growth rate is given by (see Shuler, 1988):

$$\text{growth rate} = \frac{\mu_m S}{K_s + S + S^2/K_I} \quad (8.60)$$

where μ_m is the maximum specific growth reaction rate, K_s is the monod constant, K_I is the inhibition constant, and S represents the substrate concentration (the material being reacted). The equation governing cells in a CSTR is

$$Q(S - S_{\text{in}}) = -V \frac{\mu_m S}{K_s + S + S^2/K_I} \quad (8.61)$$

In nondimensional form this is

$$(\sigma - 1) = -Da \frac{\sigma}{\omega + \sigma + \varepsilon \sigma^2}$$

where

$$Da = \frac{V\mu_m}{QS_{\text{in}}}, \quad \omega = \frac{K_s}{S_{\text{in}}}, \quad \varepsilon = \frac{S_{\text{in}}}{K_I} \quad (8.62)$$

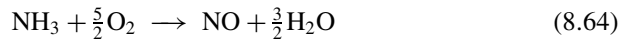
Solve this problem for $Da = 1.19$, $\omega = 3.56 \times 10^{-3}$, $\varepsilon = 2.53$. (Hint: Look for multiple solutions.) See Schmidt (2004) and Fogler (2005) for additional information about biological reactors.

- 8.6.** Solve the same Problem 8.5 with the CSTR replaced by a plug flow reactor. The equations are then

$$\frac{d\sigma}{dz} = Da \frac{\sigma}{\omega + \sigma + \varepsilon \sigma^2}, \quad z = 0 \rightarrow 1, \quad \sigma(0) = 1 \quad (8.63)$$

- 8.7.** Solve Eq. (8.41) for a total volume of 1.2 m^3 , $k = 0.3 \text{ m}^3/\text{kmol s}$, a total concentration of 5 kmol/m^3 . (1) Use MATLAB; (2) use FEMLAB.

- 8.8.** Ammonia undergoes a reaction with oxygen in the presence of a platinum catalyst to form nitric oxide and water. The reaction is carried out in a microreactor (see Chapter 11), which results in a constant temperature. The reaction is



The reaction occurs on the surface of a tube in the microreactor, as governed by the following rate of reaction (in g mol/cm^2 ; Pignet and Schmidt, 1975):

$$\begin{aligned} r_{\text{NH}_3} &= \frac{A}{B \cdot C}, \quad A \equiv 3.4 \times 10^{-8} \exp(21,700/RT) p_{\text{NH}_3} p_{\text{O}_2}^{1/2} \\ B &\equiv 1 + 8 \times 10^{-2} \exp(4400/RT) p_{\text{O}_2}^{1/2} \\ C &\equiv 1 + 1.6 \times 10^{-3} \exp(25,500/RT) p_{\text{NH}_3} \end{aligned} \quad (8.65)$$

The inlet stream contains ammonia and oxygen with partial pressures of 0.046 and 0.068 torr, respectively, and the flow rate is 100 l/h. Solve the problem for a temperature of (1) 473 K and (2) 673 K and compare the results. The microreactor channel is $40 \times 300 \mu\text{m}$, and its length is 1 cm. The governing equations are

$$\frac{dF_i}{dz} = P v_i r_{\text{NH}_3}, F_i(0) = F_{i0} \quad (8.66)$$

where: v_i = stoichiometric coefficient of the i th species; P = perimeter, cm; F_i = flow rate of the i th species, and gmol/s; and F_{i0} = flow rate of the i th species at the inlet, gmol/s. In addition,

$$F_T = \sum_{i=1}^N F_i, \quad y_i = \frac{F_i}{F_T}, \quad p_i = y_i p_T \quad (8.67)$$

- 8.9.** When an enzyme is immobilized on a surface, there may be a mass transfer resistance that limits the concentration of the reacting species on the surface. The following problem (from Shuler, 1988) finds the concentration of substrate on the surface when there is a reaction accompanied by mass transfer resistance. The Michaelis–Menten reaction on a surface is governed by

$$\text{rate of reaction} = \frac{k_0 S}{S + K_m} \quad (8.68)$$

The equation for reaction at a surface with mass transfer limitations approaching the surface is

$$k_L(S_f - S) = \frac{\varepsilon k_0 S}{S + K_m} \quad (8.69)$$

where: k_L = mass transfer coefficient = 10^{-3} cm/s; K_m = kinetic constant = 10^{-4} mol/l; k_0 = kinetic constant = 5×10^{-6} mol/s mg enzyme; ε = enzyme loading = 1.6×10^{-4} mg enzyme/cm²; S_f = external concentration = 10^{-4} – 10^{-2} mol/l; and S = concentration at surface, mol/l. Solve for S when the external concentration is (1) 10^{-4} mol/l and (2) 10^{-2} mol/l.

- 8.10.** The steady state model for a nonisothermal CSTR representing a catalytic converter is:

$$\begin{aligned} Q C_{\text{tot}}(y_{\text{in}} - y) &= \alpha r(y, T) \\ Q C_{\text{tot}} \hat{M} C_{\text{pg}}(T - T_{\text{in}}) &= \alpha V(-\Delta H_{\text{rxn}}) r(y, T) \\ r &= \frac{0.05 k_1 y}{T(1 + K_1 y)^2}, \quad k_1 = 6.70 \times 10^{10} \exp(-12556/T), \quad K_1 = 65.5 \exp(961/T) \end{aligned} \quad (8.70)$$

The reaction rate expression is given for oxidizing carbon monoxide on a catalytic surface. When the inlet mole fraction CO is $y_{\text{in}} = 0.02$, and the inlet temperature is $T_{\text{in}} = 600$ K, find the solution to these equations. The parameters are

$$\begin{aligned} Q &= 0.06555 \text{ m}^3/\text{s}, \quad C_{\text{tot}} = 0.0203 \text{ kg mol/m}^3 \\ \alpha &= 26900 \text{ m}^2/\text{m}^3, \quad V = 6 \cdot 10^{-4} \text{ m}^3 \\ \hat{M} &= 30 \text{ kg/kg mol} \\ C_{\text{pg}} &= 1070 \text{ J/kg K}, \quad -\Delta H_{\text{rxn}} = 2.84 \times 10^8 \text{ J/kg mol} \end{aligned}$$

8.11. Solve the problem in Eq. (8.47) but for a plug flow reactor instead of the CSTR.

$$\begin{aligned} Q \frac{dc}{dV_R} &= c \exp[\gamma(1 - 1/T)], \quad c(0) = 1 \\ Q \frac{dT}{dV_R} &= -\beta c \exp[\gamma(1 - 1/T)], \quad T(0) = 1 \end{aligned} \quad (8.71)$$

Integrate from $V_R/Q = 0$ to 8.7 with $\beta = 0.15$, $\gamma = 30$. Compare the conversion with that achieved with a CSTR.

8.12. Repeat Problem 8.11 but with $V_R/Q = 0-25$ with $\beta = 0.25$, $\gamma = 30$.

8.13. The growth of cells in a transient CSTR is governed by:

$$\frac{d\sigma}{dt} = 1 - \sigma - Da \frac{\sigma}{\omega + \sigma + \varepsilon \sigma^2}, \quad c(0) = c_0 \quad (8.72)$$

For the parameters $\omega = 3.5625 \times 10^{-3}$, $\varepsilon = 2.5216$, $Da = 1.1905$, integrate to $t = 24$ for the following initial conditions for c_0 : (1) 1; (2) 0; (3) 0.146019; (4) 0.142; (5) 0.150. Discuss.

8.14. During the cure of an epoxy in a pre-preg operation to make polymeric composites, the extent of reaction, α , is sometimes modeled with the equation

$$\frac{d\alpha}{dt} = (1 + a\alpha^m)(1 - \alpha)^n, \quad \alpha(0) = 0 \quad (8.73)$$

where $a = 3$, $m = 0.5$, $n = 0.7$. Solve this equation as a function of time.

8.15. Problem 8.10 is considered as a transient CSTR. The equations governing it are

$$\begin{aligned} \varepsilon V C_{\text{tot}} \frac{dy}{dt} &= Q C_{\text{tot}} (y_{\text{in}} - y) - \alpha r(y, T) \\ V[\varepsilon \rho_g C_{\text{pg}} + (1 - \varepsilon) \rho_s C_{\text{ps}}] \frac{dT}{dt} &= Q C_{\text{tot}} \hat{M} C_{\text{pg}} (T - T_{\text{in}}) - \alpha V (-\Delta H_{\text{rxn}}) r(y, T) \end{aligned} \quad (8.74)$$

The additional parameters are

$$C_{pg} = 1070 \text{ J/kgK}, C_{ps} = 1000 \text{ J/kg}, \varepsilon = 0.68$$

$$\rho_g = C_{tot} \hat{M} = 0.609 \text{ kg/m}^3, \rho_s = 2500 \text{ kg/m}^3$$

The inlet mole fraction (of carbon monoxide) is 0.01 and the inlet temperature is 600 K.

Integrate these equations under the following conditions:

$$t < 1 \text{ s}: y_{in} = 0.01, T_{in} = 600 \text{ K}$$

$1 < t < 2 \text{ s}$: inlet conditions increase linearly to

$$y_{in} = 0.03, T_{in} = 700 \text{ K}$$

$$2 < t \text{ s} : y_{in} = 0.03, T_{in} = 700 \text{ K}$$

This model can be used to study the entire Federal Test Cycle covering many hours of operation of the catalytic converter.

9

TRANSPORT PROCESSES IN ONE DIMENSION

Chemical engineering processes involve the transport and transfer of momentum, energy, and mass. Momentum transfer is another word for fluid flow, and most chemical processes involve pumps and compressors, and perhaps centrifuges and cyclone separators. Energy transfer is used to heat reacting streams, cool products, and run distillation columns. Mass transfer involves the separation of a mixture of chemicals into separate streams, possibly nearly pure streams of one component. These subjects were unified in 1960 in the first edition of the classic book, *Transport Phenomena* (Bird et al., 2002). This chapter shows how to solve transport problems that are one-dimensional; that is, the solution is a function of one spatial dimension. Chapters 10 and 11 treat two- and three-dimensional problems. The one-dimensional problems lead to differential equations, which are solved using the computer.

Diffusion problems in one dimension lead to boundary value problems. The boundary conditions are applied at two different spatial locations: At one side the concentration may be fixed and at the other side the flux may be fixed. Because the conditions are specified at two different locations, the problems are not initial value in character. It is not possible to begin at one position and integrate directly because at least one of the conditions is specified somewhere else and there are not enough conditions to begin the calculation. Thus methods have been developed especially for boundary value problems. There is a baseball analogy that illustrates the difference. If you are at bat and there is no one on base, you can hit and run as fast as you can. This is like an initial value problem. If you are at bat and someone is on first base, you can hit, but the way you run is influenced by what the runner ahead of you does. This is like a boundary value problem.

This chapter illustrates heat transfer, diffusion, diffusion with reaction, and flow in pipes, considering steady and unsteady processes. These problems are all solved using FEMLAB[®], which is easy to use for these applications. Details of the finite element

method are not given here, but you will find a brief description in Appendix F, along with references. In essence, you represent the solution at a set of grid points (finite difference) or nodal points (finite element). This is an approximation, so the answer is only as good as the mesh allows it to be, and the accuracy is not guaranteed. Thus, you have to solve the problem on more than one mesh to ensure that the results do not depend appreciably on the mesh. The solution is exact only if you use an infinite number of points, which no one has ever done. However, you can usually make it as accurate as needed, and the FEMLAB software makes it very easy to re-solve the problem on a better mesh so you can test the accuracy.

APPLICATIONS IN CHEMICAL ENGINEERING – MATHEMATICAL FORMULATIONS

Heat Transfer

Consider steady heat transfer across a slab of material that extends infinitely in two directions and has thickness L . In the thin direction, x , the equation for steady heat conduction is

$$\frac{d}{dx} \left[k(T) \frac{dT}{dx} \right] = Q \quad (9.1)$$

$$T(0) = T_0, \quad T(L) = T_L$$

The temperature, T , is a function of position, x . Notice that one condition is at $x = 0$ and the other at $x = L$. This makes it a two-point boundary value problem. The thermal conductivity k can depend upon temperature, and when it does the problem is nonlinear. The rate of energy generation is Q .

Diffusion and Reaction

Diffusion across a flat slab is very similar to Eq. (9.1):

$$\frac{d}{dx} \left(D \frac{dc}{dx} \right) = 0 \quad (9.2)$$

$$c(0) = c_0, \quad c(L) = c_L$$

where the concentration, c , is a function of position, x . The diffusivity, D , can depend upon concentration. This is also a two-point boundary value problem because the concentration is specified at both $x = 0$ and $x = L$.

Many chemical reactions take place inside a catalyst pellet, which is a porous material (such as Al_2O_3 impregnated with the catalyst, usually metals of various kinds). The equation for a spherical catalyst pellet with radius R_p is

$$\frac{1}{r^2} \frac{d}{dr} \left(D_e r^2 \frac{dc}{dr} \right) = R(c) \quad (9.3)$$

$$\frac{dc}{dr}(0) = 0, \quad c(R_p) = c_R$$

where D_e is an effective diffusivity for the diffusion of mass inside the porous catalyst. The concentration, c , is a function of radial position, r . The reaction rate can have a variety of forms, all depending on concentration:

$$R = c, \quad R = c^2, \quad R = \frac{\alpha c}{1 + Kc} \quad (9.4)$$

In most cases, this problem is nonlinear. Again, this problem is a two-point boundary value problem because the boundary conditions are at two different spatial positions. At $r = 0$ there is no flux (center of the pellet), and at $r = R_p$ the concentration takes a specified value equal to the external concentration.

When there is a heat of reaction, the temperature increases due to the reaction. In that case you have to also solve an energy balance on the catalyst pellet. A typical energy balance is

$$\begin{aligned} \frac{1}{r^2} \frac{d}{dr} \left(k_e r^2 \frac{dT}{dr} \right) &= (-\Delta H_{\text{rxn}}) R(c, T) \\ \frac{dT}{dr}(0) &= 0, \quad T(R_p) = T_R \end{aligned} \quad (9.5)$$

The temperature, T , is a function of radial position, r , k_e is an effective thermal conductivity, and $-\Delta H_{\text{rxn}}$ is the heat of reaction. Again, there is no flux at the center, and the temperature at the outer boundary is set at T_R . Now the reaction rate depends upon both concentration and temperature [and Eq. (9.3) has to be written with $R(c, T)$]. Such problems can be very difficult owing to the rapid change in the kinetic constants with temperature, and they provide a severe test of any numerical method.

Fluid Flow

Consider the flow of a Newtonian fluid in a pipe, as illustrated in Figure 9.1. The governing differential equation is (Bird et al., 2002):

$$\begin{aligned} \frac{1}{r} \frac{d}{dr} \left(r \mu \frac{dv}{dr} \right) &= -\frac{\Delta p}{L} \\ \frac{dv}{dr}(0) &= 0, \quad v(R) = 0 \end{aligned} \quad (9.6)$$

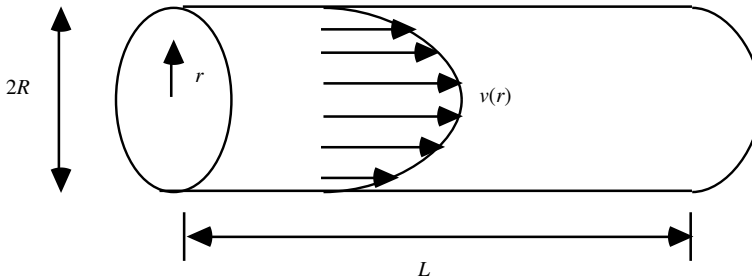


Figure 9.1. Fully developed pipe flow.

where r is the radial position from the center of the pipe of radius R , v is velocity in the axial direction, μ is the viscosity, Δp is the pressure drop along the pipe, and L is the length over which the pressure drop occurs. The boundary conditions are at different locations, making it a two-point boundary value problem. The solution to this problem is Eq. (9.7):

$$v = \frac{1}{\mu} \frac{\Delta p}{L} \frac{R^2}{4} \left(1 - \frac{r^2}{R^2} \right) \quad (9.7)$$

When solving the problem analytically, it is also possible to use a boundary condition at $r = 0$ that says the velocity is finite. Both conditions give the same result, and in fact one can be derived from the other. Note that Δp is a positive number.

In pipe flow the average velocity is defined by Eq. (9.8):

$$\langle v \rangle = \frac{\int_0^{R_p} v(r) r \, dr}{\int_0^{R_p} r \, dr} = \frac{2}{R_p^2} \int_0^{R_p} v(r) r \, dr = 2 \int_0^1 v(r') r' \, dr', \quad \text{where } r' = \frac{r}{R_p} \quad (9.8)$$

Note the $r \, dr$ in the integrand, which is necessary because cylindrical polar coordinates are used. The relationship between the average velocity and the pressure gradient can be obtained by integrating Eq. (9.7) using Eq. (9.8); this is the Hagen–Poiseuille law. The peak velocity is twice the average velocity in pipe flow:

$$\langle v \rangle = \frac{1}{\mu} \frac{\Delta p}{L} \frac{R^2}{8} = \left(\frac{\Delta p}{L} \right) \frac{D^2}{32\mu} \quad (9.9)$$

When the flow is between two wide, flat plates instead of inside a cylinder, Eq. (9.6) becomes

$$\begin{aligned} \frac{d}{dx} \left(\mu \frac{dv}{dx} \right) &= -\frac{\Delta p}{L} \\ \frac{dv}{dx}(0) &= 0, \quad v(H) = 0 \end{aligned} \quad (9.10)$$

The plates are taken at a distance $2H$ apart. The solution is

$$v = \frac{1}{\mu} \frac{\Delta p}{L} \frac{H^2}{2} \left(1 - \frac{x^2}{H^2} \right) \quad (9.11)$$

Now the average velocity is

$$\langle v \rangle = \frac{\int_0^H v(x) \, dx}{\int_0^H dx} = \frac{1}{H} \int_0^H v(x) \, dx = \int_0^1 v(x') \, dx', \quad \text{where } x' = \frac{x}{H} \quad (9.12)$$

and the relationship between average velocity and pressure drop is

$$\langle v \rangle = \frac{1}{\mu} \frac{\Delta p H^2}{L} \frac{1}{3} \quad (9.13)$$

The peak velocity is 1.5 times the average velocity for flow between parallel plates. These formulas are provided here because they provide a good benchmark against which to check any numerical solution by integrating over boundaries.

When the fluid is non-Newtonian, it may not be possible to solve the problem analytically. For example, for the Bird–Carreau fluid (Bird et al., 1987, p. 171) the viscosity is

$$\eta = \frac{\eta_0}{\left[1 + \lambda \left(\frac{dv}{dr} \right)^2 \right]^{(1-n)/2}} \quad (9.14)$$

The viscosity depends upon the shear rate, dv/dr . New parameters are introduced that must be determined from experiments: η_0 , λ , and n . Now the boundary value problem is

$$\begin{aligned} \frac{1}{r} \frac{d}{dr} \left(r \eta \frac{dv}{dr} \right) &= -\frac{\Delta p}{L} \\ \frac{dv}{dr}(0) &= 0, \quad v(R) = 0 \end{aligned} \quad (9.15)$$

It is not possible to solve this equation analytically for v , except for special values of η . For problems like this, numerical methods must be used. If you set $\eta = \mu$ you get the same problem defined above for a Newtonian fluid.

Unsteady Heat Transfer

Consider the following heat transfer problem corresponding to heat transfer in a slab,

$$\frac{\partial T}{\partial t} = \alpha \frac{\partial^2 T}{\partial x^2} \quad (9.16)$$

The temperature depends on both time, t , and position, x . The coefficient of thermal diffusion is

$$\alpha = \frac{k}{\rho C_p} \quad (9.17)$$

The slab is insulated at the left ($x = 0$) and has a temperature value of zero on the other side ($x = L$)

$$\frac{\partial T}{\partial x}(x = 0) = 0, \quad T = 0 \text{ at } x = L \quad (9.18)$$

while the initial temperature distribution is

$$T(x, 0) = f(x) \quad (9.19)$$

You thus start with $T(x, 0)$ and find $T(x, t)$ as a function of x from 0 to L and t from 0 to infinity.

EXAMPLE: HEAT TRANSFER IN A SLAB

Consider the problem of heat transfer in a slab with no internal heat generation but with a thermal conductivity that varies linearly with temperature – Eq. (9.1) with the following parameters:

$$k(T) = 1 + T, \quad Q = 0 \quad (9.20)$$

The boundary conditions are taken as $T = 0$ at $x = 0$ and $T = 1$ at $x = 1$.

Follow the instructions in Table 9.1 (p. 160) for running FEMLAB.

Step 1 Choose 1D, then Chemical Engineering, then Energy Balance, then Conduction.

Step 2 Create the line by clicking on the line icon. Then move the cursor to the beginning of the line and draw to the end.

Step 3 Define the mesh by clicking on the triangle symbol. First 15 elements are created, with 31 nodes. Successive clicks to refine the mesh give 61 and 121 nodes.

Step 4 Specify the parameters for the differential equation. Choose Physics/Subdomain Settings to see the equation:

$$-\nabla \cdot (k \nabla T) = Q + h_{\text{trans}}(T_{\text{ext}} - T) + C_{\text{trans}}(T_{\text{ambtrans}}^4 - T^4) \quad (9.21)$$

You want to solve

$$\begin{aligned} \frac{d}{dx} \left[(1 + T) \frac{dT}{dx} \right] &= 0 \\ T(0) &= 0, \quad T(1) = 1 \end{aligned} \quad (9.22)$$

Thus, in FEMLAB you need to choose the following parameters

$$k = 1 + T, \quad Q = 0, \quad h_{\text{trans}} = C_{\text{trans}} = 0 \quad (9.23)$$

Step 5 Choose Physics/Boundary Settings to set the boundary conditions. Choose 1 (the left-hand side) and set the temperature to 0. Choose 2 (the right-hand side) and set the temperature to 1.

Step 6 Press '=' to solve the problem. The solution is shown in Figure 9.2.

Step 7 It is useful to examine the heat flux. Choose Postprocessing/Domain Plot Parameters, choose the domain (the 1), change the variable to heat flux, and click OK.

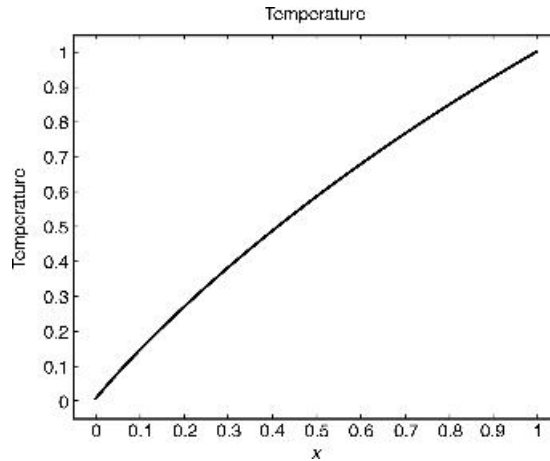


Figure 9.2. Solution to Eq. (9.22).

The result is shown in Figure 9.3. The heat flux is oscillating about 1.5 (which happens to be the exact solution), between 1.497 and 1.501.

Step 8 You can calculate the average temperature. Choose Postprocessing/Subdomain integration, then select the domain (the 1), click Apply or OK. At the bottom of the screen, the value of the integral is given as 0.555556. This is the value of Eq. (9.24):

$$\int_0^1 T(x) \, dx = 0.555556 \quad (9.24)$$

Step 9 Finally, the thermal conductivity is $1 + T$. You can plot this by choosing Postprocessing/Domain Plot Parameter, and typing $1 + T$ in the window. The result is shown in Figure 9.4.

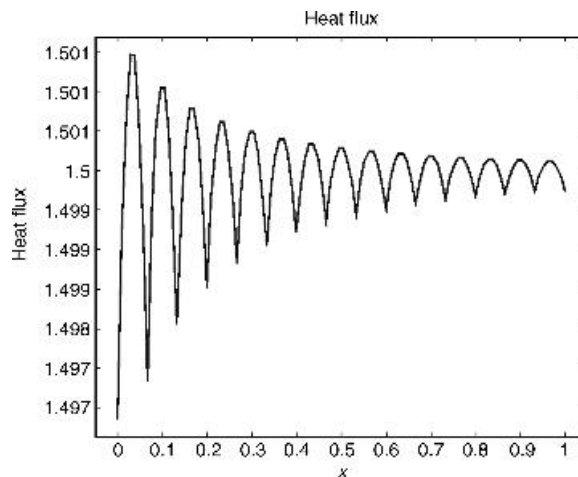


Figure 9.3. Heat flux in problem, Eq. (9.22).

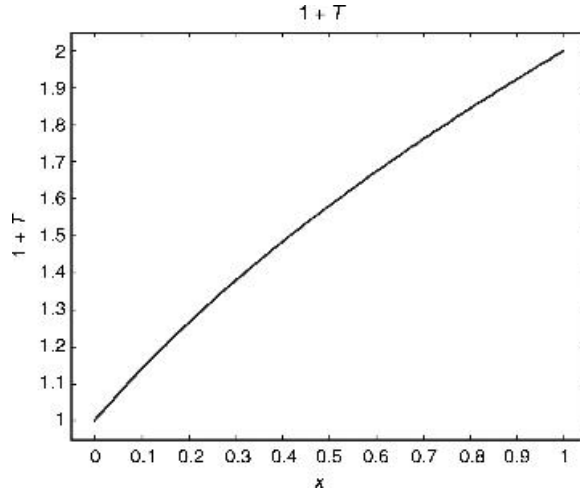


Figure 9.4. Thermal conductivity in problem, Eq. (9.22).

EXAMPLE: REACTION AND DIFFUSION

The next example is a reaction–diffusion problem in a spherical domain. The reaction rate expression is a nonlinear function of concentration, of a type that is appropriate for the Michaelis–Menten reaction in biological systems. The nondimensional form of the problem is in Eq. (9.25), and it is solved for $\alpha = 5$, $K = 2$.

$$\frac{1}{r^2} \frac{d}{dr} \left(r^2 \frac{dc}{dr} \right) = \frac{\alpha c}{1 + Kc}, \quad \frac{dc}{dr}(0) = 0, \quad c(1) = 1 \quad (9.25)$$

Follow these steps to solve this problem in FEMLAB.

Step 1 Open FEMLAB, and choose Chemical Engineering, then Mass Transfer, then Diffusion, then Steady State.

Step 2 Create the line by clicking on the line icon. Then move the cursor to the beginning of the line and draw to the end.

Step 3 Define the mesh by clicking on the triangle symbol. First 15 elements are created, with 31 nodes. Successive clicks give 61 and 121 nodes.

Step 4 To set the equation and parameters in it, choose Physics/Subdomain Selection; Eq. (9.26) is the equation shown in the program.

$$\nabla(-D\nabla c) = R, \text{ which is } \nabla(D\nabla c) = -R$$

in one dimension this is

$$-\frac{d}{dx} \left(D \frac{dc}{dx} \right) = R \quad (9.26)$$

Since you have variable coefficients, expand Eq. (9.25) into

$$\frac{1}{r^2} \frac{d}{dr} \left(r^2 \frac{dc}{dr} \right) = \frac{d^2c}{dr^2} + \frac{2}{r^2} \frac{dc}{dr} = \frac{\alpha c}{1 + Kc}$$

or

$$-\frac{d^2c}{dr^2} = \frac{2}{r^2} \frac{dc}{dr} - \frac{\alpha c}{1 + Kc} \quad (9.27)$$

This equation becomes Eq. (9.26) by choosing

$$r = x, D = 1, R = 2*cx/x - rate, rate = \frac{\alpha c}{1 + Kc} \quad (9.28)$$

In the Subdomain Selection window, set $D=1$, and $R = 2*cx/x - rate$. The variable cx is dc/dx .

Step 5 Set the value of ‘alpha’ and K in Options/Constants, and $rate$ in Options/Expressions/Subdomain expressions.

Step 6 Set the boundary conditions: zero flux at the left (boundary 1) and concentration equal to 1 at the right (boundary 2).

Step 7 Press ‘=’ to solve the problem. The solution is shown in Figure 9.5.

Parametric Solution

It is instructive to solve this problem for multiple parameters, α , and you can do that using the parametric solver. If you had not already used a variable name, alpha, in Physics/Subdomain Selection, you would have to use a variable name now.

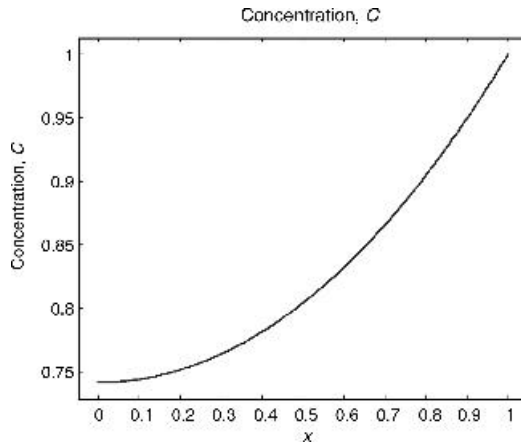


Figure 9.5. Concentration in spherical domain with Michaelis–Menten reaction.

Step 1 Go to Solver Parameters and choose Parametric nonlinear. Fill in the window as shown in Figure 9.6 to have α go from 0 to 20 in steps of (2).

Step 2 Click the '=' to solve the problem. This time the program solves the problem for a variety of α . In fact, it finds the solution and its derivative with respect to α each time. Then the program uses a linear Taylor expansion, Eq. (9.29), to provide a good starting guess for the iterative solution at the new α :

$$c(r, \alpha^{k+1}) = c(r, \alpha^k) + \left. \frac{dc}{d\alpha} \right|_{\alpha^k} (\alpha^{k+1} - \alpha^k) \quad (9.29)$$

Step 3 Next choose Postprocessing/Domain Plot Parameters/General, and fill in the window. Click on the Line Extrusion tab, and choose concentration to plot (it is the default choice). Click on Line Settings, choose Legend, and set the Line Marker to Cycle. Click Apply or OK. Figure. 9.7 is the result.

Figure 9.7 indicates that the concentration decreases from the outer part of the sphere to the inner part of the sphere and that this decrease is more dramatic with larger values of α . This is because the rate of reaction is faster and diffusion cannot keep up, thus decreasing the concentration. Figure 9.8 shows the rate of reaction as a function of position and α . For small values of α , the rate is the same at all positions, but as α increases the rate of reaction varies by a factor of 2 or more from the center to the outer boundary.

EXAMPLE: FLOW OF A NEWTONIAN FLUID IN A PIPE

This example considers flow in a pipe for a Newtonian fluid, Eq. (9.6) You will solve this problem using FEMLAB with $\Delta p = 2.8 \times 10^5$ Pa, $\mu = 0.492$ Pa s, $L = 4.88$,

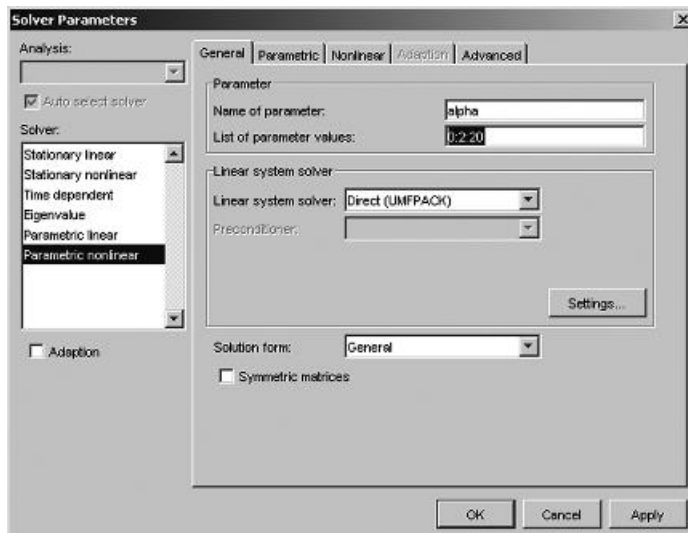


Figure 9.6. Use of parametric solver.

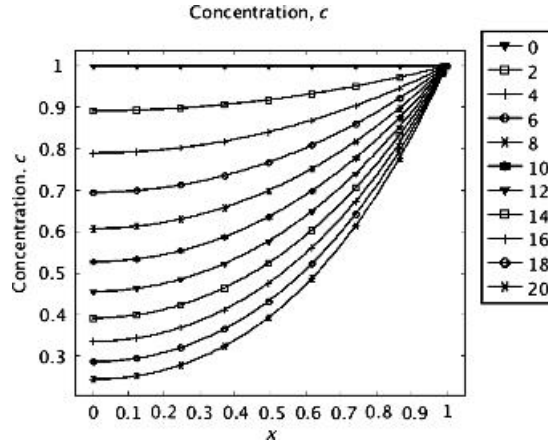


Figure 9.7. Solutions for several values of α .

$R = 0.0025$ m. You must expand the differential equation because the one-dimensional option does not contain cylindrical geometry. (This limitation is removed in two- and three-dimensional problems, but here a one-dimensional solution suffices.) So, rewrite Eq. (9.6) as

$$\frac{d^2v}{dr^2} = -\frac{1}{\mu} \frac{\Delta p}{L} - \frac{1}{r} \frac{dv}{dr}$$

$$\frac{dv}{dr}(0) = 0, \quad v(R) = 0 \quad (9.30)$$

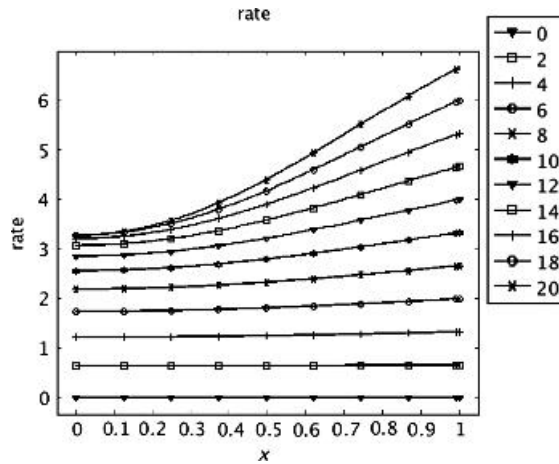


Figure 9.8. Rate of reaction as a function of $x(r)$ and α .

Step 1 Open FEMLAB and choose Chemical Engineering, 1D, Mass Transfer, Diffusion. The variable c is a stand-in for velocity, v . (You can change the variable name if desired.)

Step 2 Use the Axis/Grid Settings to set the screen dimensions from -0.0001 to 0.003 . Choose Draw/Specify Objects/Line and enter $[0 \text{ to } 0.0025]$; click OK.

Step 3 Create a mesh.

Step 4 In the menu Physics/Subdomain Settings set the following values, as shown in Figure 9.9.

$$\delta_{ts} = 0, \quad D = 1, \quad R = 1.166e5 + cx/x \quad (9.31)$$

Step 5 In the Boundary Settings, for boundary 1 (the left-hand side) choose Insulated, which sets the derivative to zero. For boundary 2 (the right-hand side) choose concentration and set (or leave) its value to zero.

Step 6 Click on '=' to solve the problem.

Step 7 After solving, in the Postprocessing menu, choose Domain Plot Parameters. Plot the concentration, which is really the velocity, as shown in Figure 9.10.

Step 8 To compute the average velocity, you need to be careful to get the integrals in Eq. (9.8). Thus, type $c*x$ in the expression and choose Apply. The value of the integral is 2.84668×10^{-7} , which appears in the screen at the bottom. Change the expression to x , and choose Apply. The value of this integral is 3.125×10^{-6} . The ratio of these is the average velocity, which is thus 0.09109 . Note that this is exactly half of the peak velocity, which is expected since that is true of the exact solution. It is also possible to look at the

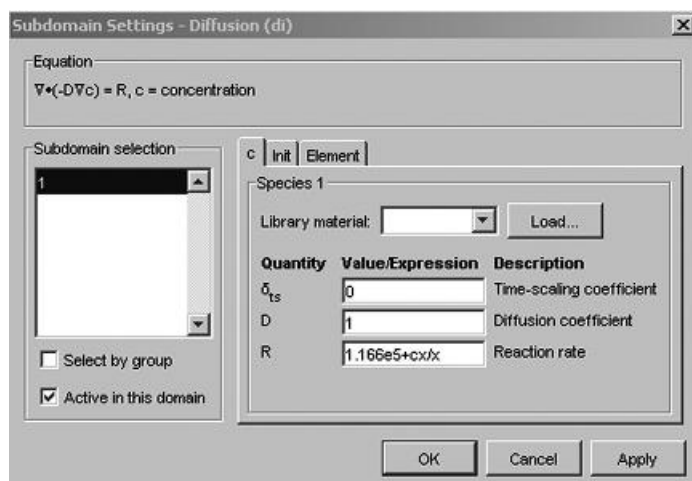


Figure 9.9. Subdomain Settings for cylindrical geometry in one dimension.

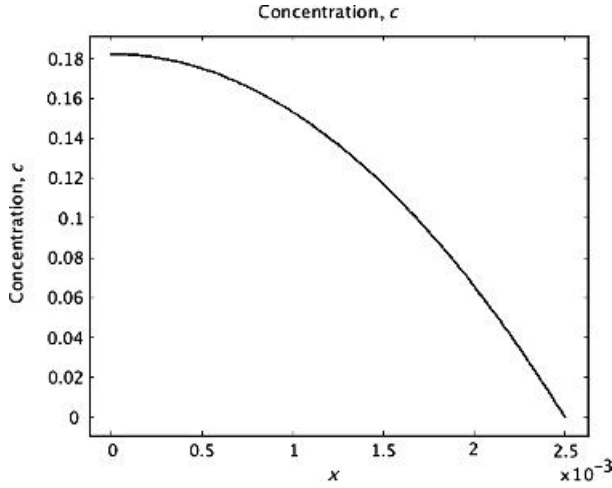


Figure 9.10. Velocity profile for fully developed Newtonian flow in a pipe.

velocity gradient (a straight line from zero at the center to a maximum at the wall). The shear stress is the viscosity multiplied by the velocity gradient, and that is linear with distance, too, for a Newtonian fluid.

EXAMPLE: FLOW OF A NON-NEWTONIAN FLUID IN A PIPE

Next, you will consider flow in a pipe where the fluid is a non-Newtonian fluid, in particular a polymer melt. Take the viscosity as a Carreau function, Eq. (9.14). The parameters used here are $\eta_0 = 0.492$, $\lambda = 0.1$ and $n = 0.8$. The problem is defined by

$$\frac{1}{r} \frac{d}{dr} \left(r \eta \frac{dv}{dr} \right) = -\frac{\Delta p}{L} \quad \text{or} \quad \frac{d}{dr} \left(\eta \frac{dv}{dr} \right) + \frac{\eta}{r} \frac{dv}{dr} = -\frac{\Delta p}{L}$$

$$\frac{dv}{dr}(0) = 0, \quad v(R) = 0 \quad (9.32)$$

Steps 1–3 Follow the instructions in Table 9.1 to choose the equation, using the diffusion equation as a stand-in for velocity. Then define the line, and create a mesh.

Step 4 In the Physics/Subdomain Settings type

$$D = \text{eta}, \quad R = (\text{eta}/x)^*ux + dpL \quad (9.33)$$

The variable x is a stand-in for the radial position, r .

Step 5 Go to the Options/Expressions/Subdomain Expressions and define eta and shearrate . Use

$$\text{shearrate} = \text{abs}(ux) \quad \text{and} \quad \text{eta} = \text{eta}_0 / (1 + (\text{lambda} * cx)^2)^{((1-n)/2)} \quad (9.34)$$

TABLE 9.1. Tutorial for Using FEMLAB to Solve Two-Point Boundary Value Problems**Open**

- Open FEMLAB;
- Scroll from 2D to 1D and choose the equation and click OK; (double click on the folder or click once on the arrow to see the subheadings) or
- Select OK in the Model Navigator and open a previous solution.

Draw

- Click on the line icon, move to the screen and click on the beginning and ending point of the line;
- To set exact dimensions, choose Draw/Specify Options/Line or Apple-L and enter [0 1].

Mesh

- Click once on the triangle icon;
- Click on divided triangle icon to refine the mesh.

Physics/subdomain settings

- Check the differential equation and set the parameters.

Physics/boundary settings

- Click on the number 1 in the window;
- Set the type of boundary condition and set the parameters;
- Click on the number 2 in the window and set the other boundary condition.

Solve/solve problem

= to solve the problem.

Postprocessing/Plot Parameters – set parameters to choose the type of plot;

- For detailed information about the solution, you can use subdomain integration to integrate a variable or expression over the line; you can integrate anything for which you can write an equation.
- You can go back and refine the mesh (click the more refined mesh symbol) and re-solve the problem. This gives you an indication of the accuracy of the solution.

Save your work

- Save in xxx.fl file format for re-start from FEMLAB stand-alone or in xxx.m file format for restart from either FEMLAB running under MATLAB® or FEMLAB stand-alone. (Note: m-Files save all your commands, including mistakes.)

The constants are defined in Options/Constants; here you use the pressure drop per length ($5.74 \cdot 10^4$), whereas for the Newtonian fluid you used the pressure drop per length divided by the viscosity.

Step 6 For the first calculation, you replace η by η_0 in Eq. (9.33) and click ‘=’. The solution should be the same as shown in Figure 9.10. This step solves the problem for a Newtonian fluid, and the solution is used as the first guess for the solution for a non-Newtonian fluid.

Step 7 Finally, you replace η_0 by η in Eq. (9.33) in order to solve the problem for a non-Newtonian fluid. Click ‘=’ with a circular arrow, or choose Solve/Restart. Plot the function $c(x)$, as shown in Figure 9.11. If the pressure drop is changed, the shear rate changes and this causes the viscosity to change. Thus, the velocity changes its shape.

Step 8 The average velocity is given by Eq. (9.8); calculate it the same way as for a Newtonian fluid. The average velocity increases from 0.091 (Newtonian) to 0.17

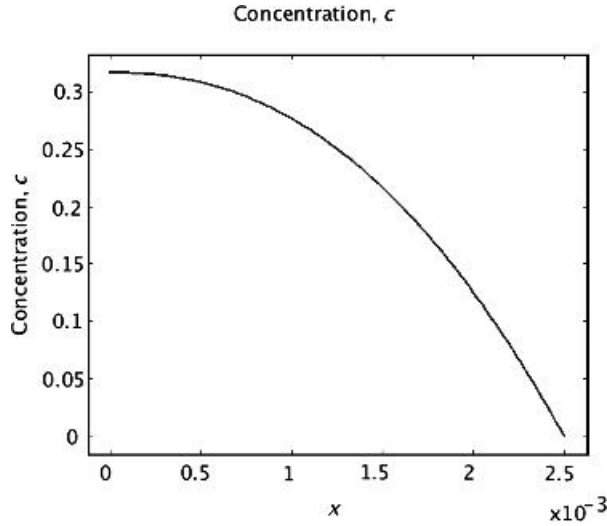


Figure 9.11. Velocity profile for fully developed non-Newtonian flow in a pipe.

(Carreau). The reason for this is shear thinning. The shear rate is zero at the centerline and is a maximum (in absolute value) at the solid boundary. Because the viscosity decreases as the shear rate increases [see (Eq. 9.14)], the viscosity decreases as the radial position increases, as seen in Figure 9.12.

One handy way to get all the solutions for several pressure drops (and avoid convergence problems) is to use the Parametric Nonlinear option in the Solver Parameters. Call the variable 'presdropx' and set its value to 0:1:7. Then in the Physics/Subdomain Settings, where you put the Δp , replace it with $10^{\text{presdropx}}$. The program FEMLAB

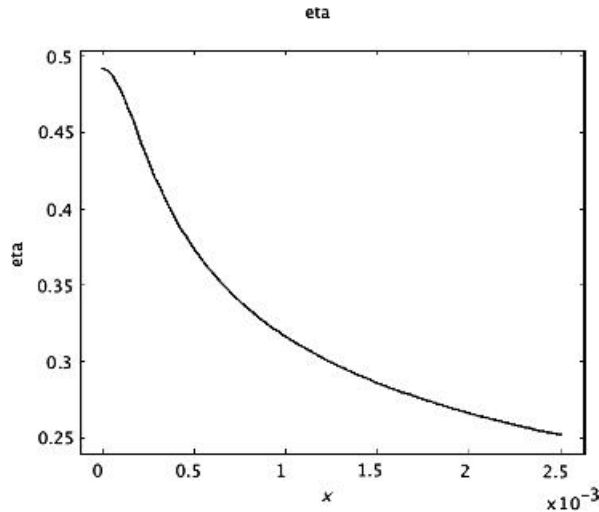


Figure 9.12. Viscosity, Eq. (9.14) for fully developed non-Newtonian flow in a pipe.

will then obtain solutions for pressure drops of 1, 10, 10^2 , 10^3 , $\dots$, 10^7 Pa. A plot of velocity will show the velocity for all the different pressure drops.

The velocities vary over several orders of magnitude for the different pressure drops. You could plot the velocity divided by the pressure drop to improve your graph. An even better option would be to compute the average velocity of each solution, and plot the velocity divided by the average velocity, all on the same graph.

EXAMPLE: TRANSIENT HEAT TRANSFER

The next example is a transient heat transfer problem.

$$\begin{aligned} \frac{\partial T}{\partial t} &= \alpha \frac{\partial^2 T}{\partial x^2}, \quad \alpha = 2 \\ \left. \frac{\partial T}{\partial x} \right|_{x=0} &= 0, \quad T(1, t) = 0, \quad T(x, 0) = 1 \end{aligned} \quad (9.35)$$

Steps 1–3 You make the same FEMLAB choices as for heat conduction.

Step 4 In the Physics/Subdomain Settings mode the equation listed at the top is

$$\delta_{ts} \rho C_p \frac{\partial T}{\partial t} - \nabla \cdot (k \nabla T) = Q + h_{\text{trans}}(T_{\text{ext}} - T) + C_{\text{trans}}(T_{\text{ambtrans}}^4 - T^4) \quad (9.36)$$

Set the following parameters.

$$\delta_{ts} = 1, \quad \rho = 1, \quad C_p = 1, \quad k = 2, \quad Q = 0, \quad h_{\text{trans}} = C_{\text{trans}} = 0 \quad (9.37)$$

Step 5 In this same mode, you choose the Init tab, and set the initial conditions to 1.0.

Step 6 In the Physics/Boundary Settings mode, you set boundary 1 (the left-hand side) to Insulated and boundary 2 (the right-hand side) to Temperature, and set the value to 0.

Step 7 Then choose the Solve/Solver Parameters menu. Change the solution method to Time Dependent. The default times listed are 0:0.1:1. This means a solution will be available for plotting at $t = 0, 0.1, 0.2, 0.3, \dots, 1.0$. Press OK, then click the '=' to solve the problem.

Step 8 Choose Postprocessing, then Domain Plot Parameters; insure that all the items are selected in the General window and click Line/Extrusion. Click on the 1 (choice of domain) and then OK to obtain Figure 9.13.

You can also plot a three-dimensional plot showing $T(x, t)$. Choose the menu Domain Plot Parameters and select all the times listed. Then choose Line/Extrusion, select domain one, and click OK; Figure 9.14 appears.

The spike in the temperature plot at $x = 1$ and $t = 0$ is caused by a discontinuity. It is not possible to have the temperature be both 0 (the boundary condition) and 1 (the initial condition). The net effect of this is to have an artifact in $T(x = 1, t = 0)$. This spike quickly

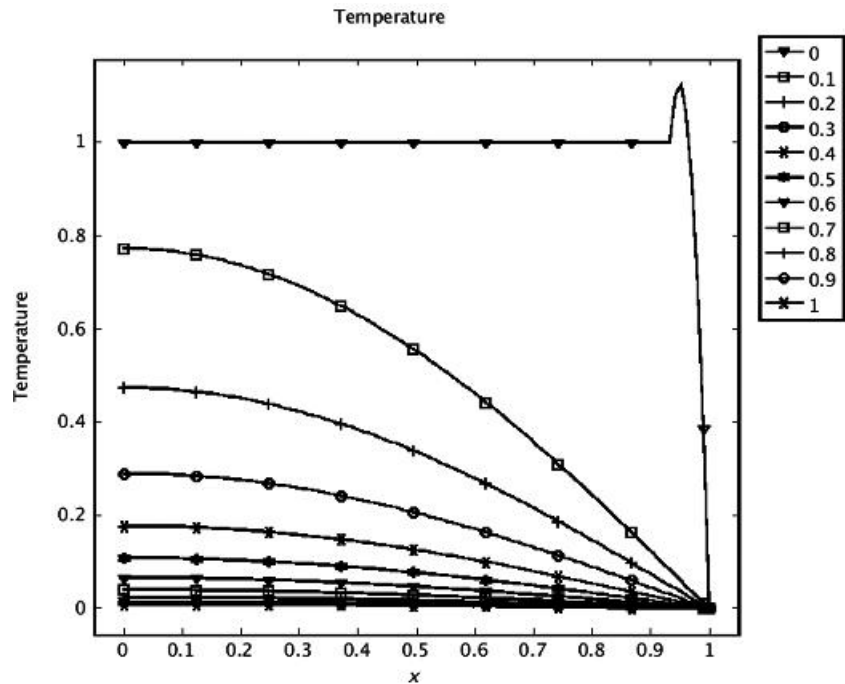


Figure 9.13. Solution of heat transfer problem, Eq. (9.34).

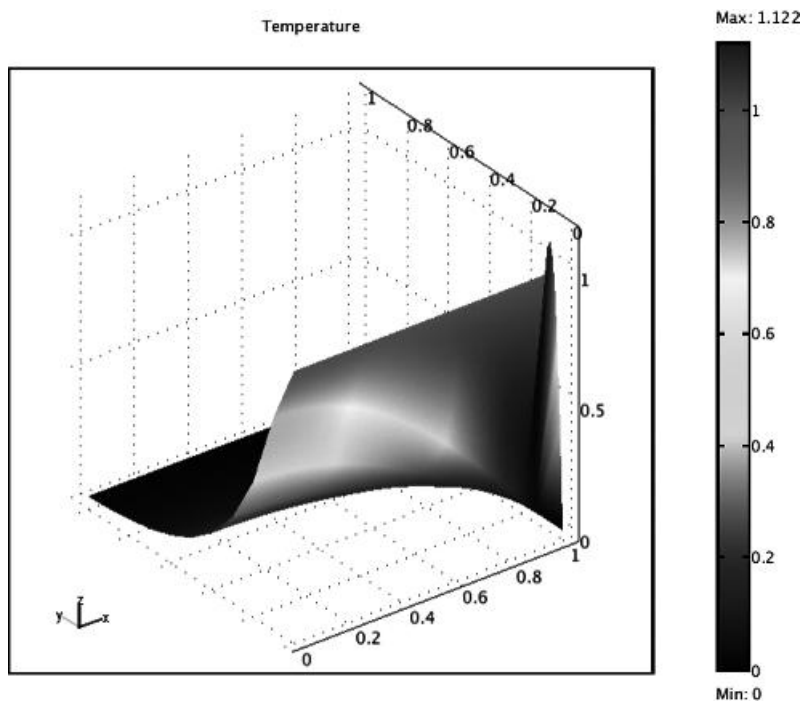


Figure 9.14. Solution of heat transfer problem, Eq. (9.34) plotted in three dimensions.

dampens out in time, though. The spike is really a manifestation of an improperly posed mathematical problem rather than an incorrect method, because such a discontinuity is impossible to create physically.

EXAMPLE: LINEAR ADSORPTION

Adsorption is one way to remove a chemical species from a flowing stream. A cylindrical tube is packed with adsorbent material, often in the form of small spheres. The adsorbent has the property that some materials adsorb while others do not. The flowing stream goes in and out of the interstices between the spheres, and contacts the adsorbent. The chemical that is strongly adsorbed is removed from the flowing stream and appears on the solid adsorbent. The next example illustrates that process.

The differential equations governing linear adsorption are given by two equations, the first a mass balance on the fluid phase and the second a mass balance on the stationary phase.

$$\phi \frac{\partial c}{\partial t'} + \phi V \frac{\partial c}{\partial x'} + (1 - \phi) \frac{\partial n}{\partial t'} = 0, \quad \frac{\partial n}{\partial t'} = k(\gamma c - n) \quad (9.38)$$

where c is the fluid concentration (moles per fluid volume), n is the concentration on the solid adsorbent (moles per solid volume), ϕ is the void fraction in the bed, V is the fluid velocity, t' is time, x' is distance down the bed, k is a mass transfer coefficient, and γ is the slope of the equilibrium line, n vs c . The initial conditions give the concentration in fluid phases and adsorbed phases at time zero, which in this example are no concentration at all.

$$c(x', 0) = 0, \quad n(x', 0) = 0 \quad (9.39)$$

The only boundary condition is the inlet concentration, here taken as 1.0.

$$c(0, t') = 1 \quad (9.40)$$

The equations are made nondimensional using the following variables:

$$x = \frac{x'k}{V}, \quad t = kt' \quad (9.41)$$

Then Eq. (9.38) is

$$\frac{\partial c}{\partial t} + \frac{\partial c}{\partial x} + \frac{1 - \phi}{\phi}(\gamma c - n) = 0, \quad \frac{\partial n}{\partial t} = \gamma c - n \quad (9.42)$$

This problem is linear, and an analytical solution exists, but it is very complicated to evaluate because it depends upon integrals of Bessel functions (Rhee et al., 1986). Numerical solutions are also available for comparison (Finlayson, 1992). When you use FEMLAB it is necessary to add some diffusion to the problem, and you solve an

augmented equation for concentration. The added term involves a Peclet number, defined in terms of an effective diffusion coefficient.

$$\frac{\partial c}{\partial t} + \frac{\partial c}{\partial x} + \frac{1-\phi}{\phi}(\gamma c - n) = \frac{1}{Pe} \frac{\partial^2 c}{\partial x^2}, \quad Pe = \frac{\phi V^2}{Dk} \quad (9.43)$$

This linear adsorption problem is solved in FEMLAB.

Step 1 Open FEMLAB and choose Chemical Engineering Module, Mass Balances, Convection and Diffusion, and Transient Analysis.

Step 2 For the concentration of material on the solid adsorbent, use the Multiphysics menu, and choose the same equation. This time, however, change the name of the variable to ns . Add this equation to the problem.

Step 3 Draw a line from $x = 0$ to $x = 1$ and set the mesh to have 60 elements, or 121 nodes. For the two problems, then, there are 242 degrees of freedom.

Step 4 Be sure the concentration equation is selected (under the Multiphysics menu). Under the Physics/Subdomain Settings, the following equation is displayed:

$$\delta_{ts} \frac{\partial c}{\partial t} + \nabla \cdot (-D \nabla c) = R - \mathbf{u} \cdot \nabla c \quad (9.44)$$

Make this equation correspond to Eq. (9.43) by setting the following parameters:

$$\delta_{ts} = 1, \quad D = 1/1000, \quad R = -(1 - \phi) \cdot \text{rate} / \phi, \quad u = u \quad (9.45)$$

Also choose the Artificial Diffusion option and choose the Petrov–Galerkin method. This adds additional diffusion to the problem. Choose the Init tab and set the initial concentration to zero.

Step 5 Under Options/Expressions/Subdomain Expressions, define the following term for the equilibrium equation.

$$\text{rate} = \text{gamma} * c - ns \quad (9.46)$$

Also, set constants $\phi = 0.4$, $\gamma = 2$ under Options/Constants. These numbers define the case that is solved.

Step 6 Under the Multiphysics menu, select the second equation, for ns . The equation is the same as Eq. (9.44), but this time you will set the following parameters:

$$\delta_{ts} = 1, \quad D = 0, \quad R = \text{rate}, \quad u = 0 \quad (9.47)$$

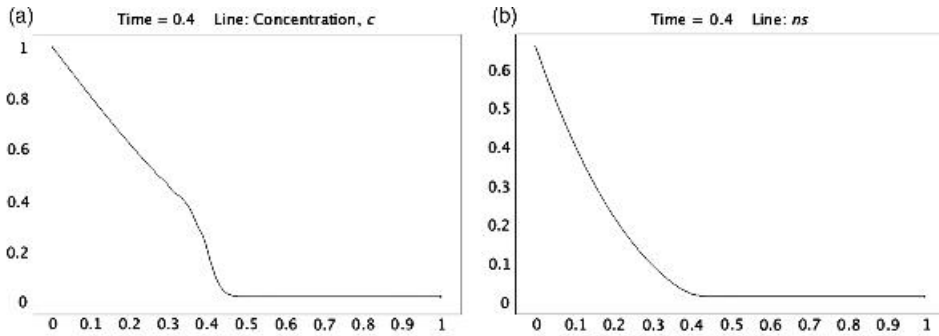


Figure 9.15. Solution of linear adsorption problem, Eqs. (9.43), (9.39), and (9.40).

Step 7 Set the boundary conditions for the first equation to have concentration equal to 1.0 at the left-hand side and convective flux at the right-hand side. Since the material adsorbed cannot flow in or out when it is on the adsorbent, the boundary conditions for the second equation are Insulation/Symmetry (i.e. no flux) at both ends.

Step 8 Press '=' to solve the problem.

Step 9 The concentration in the fluid is shown in Figure 9.15a and the concentration adsorbed is shown in Figure 9.15b. To obtain these figures, choose Plot Parameters. In the General tab, select 'solution at time' as 0.4. In the Line tab, select either c or ns as required.

The solution is a good representation of the solution, but the front, where the concentration drops quickly, is not as steep as it should be (Finlayson, 1992). The front is smoothed somewhat owing to the added diffusion term and the use of the Petrov–Galerkin method. If you solve the problem without either of these artifacts [e.g., $D = 0$ in Eq. (9.45) and using no Artificial Diffusion], the solution oscillates wildly, as seen in Figure 9.16.

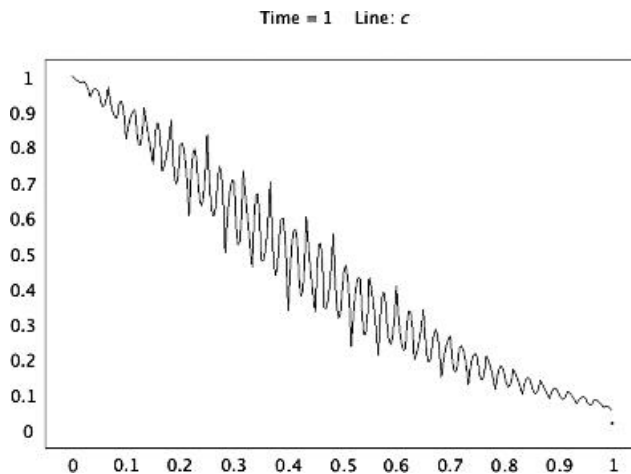


Figure 9.16. Solution of linear adsorption problem without artificial diffusion.

The reason is that the finite element method is not well suited to problems like this, with convection but no diffusion. Your job is to use enough artificial diffusion to eliminate the oscillations in the solution but without obscuring the essential details. A variety of specialized methods are available to do that, as described by Finlayson (1992). The specialized methods include Random Choice, Euler–Lagrange, MacCormack, and Taylor–Galerkin.

This example used a linear isotherm. It is easy to change to a nonlinear isotherm, since the only change is in step 5; choose a different formula that represents another isotherm.

EXAMPLE: CHROMATOGRAPHY

In some columns, mass transfer is so fast that the concentration in the gas phase is in equilibrium with the concentration on the solid adsorbent. In that case the problem is slightly different. This case is a prototype of a chromatography column, which can be used to separate chemicals.

The equation is the first one of Eq. (9.38). Now, however, the solid concentration is an algebraic function of the fluid concentration, since they are in equilibrium. In this example, you will use a Langmuir isotherm with the algebraic expression:

$$n = \frac{\gamma c}{1 + Kc} \quad (9.48)$$

When this equation is substituted into Eq. (9.38) you obtain

$$\phi \frac{\partial c}{\partial t'} + \phi V \frac{\partial c}{\partial x'} + (1 - \phi) \frac{dn}{dc} \frac{\partial c}{\partial t'} = 0 \quad (9.49)$$

For the Langmuir isotherm, the equation is

$$\left[\phi + (1 - \phi) \frac{dn}{dc} \right] \frac{\partial c}{\partial t'} + \phi V \frac{\partial c}{\partial x'} = 0, \quad \frac{dn}{dc} = \frac{\gamma}{(1 + Kc)^2} \quad (9.50)$$

The nondimensional form of the problem is obtained using $t' = t$ and $x = x'/V$:

$$\left[1 + \frac{(1 - \phi)}{\phi} \frac{dn}{dc} \right] \frac{\partial c}{\partial t} + \frac{\partial c}{\partial x} = 0, \quad \frac{dn}{dc} = \frac{\gamma}{(1 + Kc)^2} \quad (9.51)$$

In this example, you will solve the problem with an initial concentration of one between $x = 0.25$ and 0.75 . The inlet concentration is always zero:

$$\begin{aligned} 0, & \quad x < 0.25 \\ c(x, 0) = 1, & \quad 0.25 \leq x \leq 0.75 \\ 0, & \quad 0.75 < x \end{aligned} \quad (9.52)$$

Step 1 Open FEMLAB and choose Chemical Engineering Module, Mass Balances, Convection and Diffusion, and Transient Analysis.

Step 2 Draw a line from $x = 0$ to $x = 2$ and set the mesh to have 60 elements, or 121 nodes.

Step 3 Under the Physics/Subdomain Settings, the following equation is displayed.

$$\delta_{ts} \frac{\partial c}{\partial t} + \nabla \cdot (-D \nabla c) = R - \mathbf{u} \cdot \nabla c \quad (9.53)$$

Make this equation correspond to Eq. (9.51) by setting the following parameters:

$$\delta_{ts} = 1 + (1 - phi)^* dndc/phi, \quad D = 1/1000, \quad R = 0, \quad u = 1 \quad (9.54)$$

Also, choose the Artificial Diffusion option and choose the Petrov–Galerkin method. This adds additional diffusion to the problem.

Step 4 Under Options/Expressions/Subdomain Expressions, define the following term for the equilibrium equation.

$$dndc = gamma/(1 + K^*c)^2 \quad (9.55)$$

Also, set constants $\phi = 0.4$, $\gamma = 2$ under Options/Constants. These numbers define the case that is solved.

Step 5 The initial conditions are obtained by defining the following two functions. The function *fn1* takes the value 1.0 for $x > 0.25$ and the function *fn2* takes the value 1.0 for $x > 0.75$. Thus, the initial condition is satisfied by the expression *fn1* – *fn2*. In FEMLAB, this is achieved using the Boolean operator

$$\begin{aligned} fn1 &= 1^*(x > 0.25) \\ fn2 &= 1^*(x > 0.75) \\ c(x, 0) &= fn1 - fn2 \end{aligned} \quad (9.56)$$

The operator $(x > 0.25)$ takes the value 1 when $x > 0.25$ is true and 0 otherwise.

Step 6 Set the boundary conditions for the first equation to have concentration equal to zero at the left-hand side and convective flux at the right-hand side.

Step 7 Press ‘=’ to solve the problem.

Step 8 The concentration in the fluid is shown in Figure 9.17. To obtain this figure, choose Domain Plot Parameters. Use the stored output time of zero and choose Apply. Click the Save Current Plot, use the stored output time of 1.0 and choose Apply.

The solution obtained in FEMLAB can be compared with Figure 9.9d of Finlayson (1992), which is for a specialized method called MacCormack method with Flux-correction. The FEMLAB solution is a good solution.

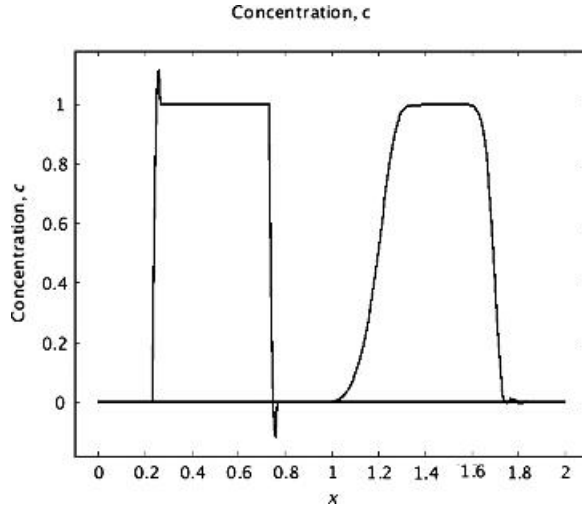


Figure 9.17. Solution of chromatography problem, Eq. (9.51).

CHAPTER SUMMARY

You saw how the equations governing energy transfer, mass transfer, and fluid flow were similar, and examples were given for one-dimensional problems. Examples included heat conduction, both steady and transient, reaction and diffusion in a catalyst pellet, flow in pipes and between flat plates of Newtonian or non-Newtonian fluids. The last two examples illustrated an adsorption column, in one case with a linear isotherm and slow mass transfer and in the other case with a nonlinear isotherm and fast mass transfer. Specific techniques you demonstrated included parametric solutions when the solution was desired for several values of one parameter, and the use of artificial diffusion to smooth time-dependent solutions which had steep fronts and large gradients.

PROBLEMS

Chemical Reaction

9.1. A chemical reactor with axial dispersion is governed by the following equations:

$$\frac{1}{Pe} \frac{d^2 c}{dx^2} - \frac{dc}{dx} - Da \frac{c}{c + v}, \quad -\frac{1}{Pe} \frac{dc}{dx}(0) = 1 - c(0), \quad \frac{dc}{dx}(1) = 0 \quad (9.57)$$

Solve for $Da = 8$, $n = 3$, and $Pe = 15$, 150, and 1500. (Hint: Include streamwise diffusion as needed.)

9.2. The following equation governs diffusion and reaction of carbon monoxide in an isothermal catalyst. Find the average reaction rate and compare the average with the reaction rate if the concentration were everywhere the same as the external

concentration (here $c = 1$). The parameters are $\phi = 32$, $Bi_m = 10$, $\alpha = 209$.

$$\frac{1}{r^2} \frac{d}{dr} \left(r^2 \frac{dc}{dr} \right) = \phi^2 R(c), \quad \frac{dc}{dr}(0) = 0, \quad -\frac{dc}{dr}(1) = Bi_m[c(1) - 1],$$

$$R(c) = \frac{c}{(1 + \alpha c)^2} \quad (9.58)$$

- 9.3.** The transient reaction and diffusion in a packed bed with axial dispersion is governed by the following equations. Begin with an initial concentration in the bed of zero. At time zero start flowing with the inlet concentration = 1.0. Integrate to steady state. Use the parameters $Pe = 100$, $0 \leq x \leq 1$, $Da = 2$, $v = 2$.

$$\frac{\partial c}{\partial t} + \frac{\partial c}{\partial x} = \frac{1}{Pe} \frac{\partial^2 c}{\partial x^2} - Da \frac{c}{c + v}, \quad -\frac{1}{Pe} \frac{\partial c}{\partial x}(0, t) = 1 - c(0, t),$$

$$\frac{\partial c}{\partial x}(1, t) = 0, \quad c(x, 0) = 0 \quad (9.59)$$

Chemical Reaction and Heat Transfer

- 9.4.** A chemical reactor with axial dispersion of both mass and energy is governed by the following equations:

$$\frac{1}{Pe} \frac{d^2 c}{dx^2} - \frac{dc}{dx} - R(c, T), \quad -\frac{1}{Pe} \frac{dc}{dx}(0) = 1 - c(0), \quad \frac{dc}{dx}(1) = 0 \quad (9.60)$$

$$\frac{1}{Pe_H} \frac{d^2 T}{dx^2} - \frac{dT}{dx} - \beta R(c, T), \quad -\frac{1}{Pe_H} \frac{dT}{dx}(0) = 1 - T(0), \quad \frac{dT}{dx}(1) = 0 \quad (9.61)$$

The reaction rate expression is

$$R(c, T) = 3.817c^2 \exp(\gamma - \gamma/T) \quad (9.62)$$

and the parameters are $Pe = Pe_H = 96$, $\gamma = 17.6$, $\beta = -0.056$.

- 9.5.** The following equations govern diffusion and reaction in a catalyst with significant heat effects. Solve for $\phi = 1.1$, $\gamma = 30$ and $\beta = 0, 0.01, 0.03, 0.07, 0.10, 0.12, 0.15$.

$$\frac{1}{r^2} \frac{d}{dr} \left(r^2 \frac{dc}{dr} \right) = \phi^2 R(c, T), \quad \frac{dc}{dr}(0) = 0, \quad c(1) = 1 \quad (9.63)$$

$$\frac{1}{r^2} \frac{d}{dr} \left(r^2 \frac{dT}{dr} \right) = -\beta \phi^2 R(c, T), \quad \frac{dT}{dr}(0) = 0, \quad T(1) = 1,$$

$$R(c, T) = c \exp[\gamma(1 - 1/T)] \quad (9.64)$$

- 9.6.** Solve the following problem for reaction and heat transfer in a porous catalyst pellet for the parameters: $\phi = 1.1$, $\gamma = 30$, $\beta = 0.15$, $Le = 1050$. See Hellinckx et al.

(1972), for a simplified form of the problem.

$$\frac{\partial c}{\partial t} = \frac{1}{r^2} \frac{\partial}{\partial r} \left(r^2 \frac{\partial c}{\partial r} \right) - \phi^2 R(c, T), \quad \frac{\partial c}{\partial r}(0, t) = 0, \quad c(1, t) = 1 \quad (9.65)$$

$$Le \frac{\partial T}{\partial t} = \frac{1}{r^2} \frac{\partial}{\partial r} \left(r^2 \frac{\partial T}{\partial r} \right) + \beta \phi^2 R(c, T), \quad \frac{\partial T}{\partial r}(0, t) = 0, \quad T(1, t) = 1, \\ R(c, T) = c \exp[\gamma(1 - 1/T)] \quad (9.66)$$

9.7. Solve problem 9.6 with $Le = 0.1$.

Mass Transfer

9.8. Solve the following diffusion problem with a variable diffusivity (appropriate for some polymers):

$$\frac{\partial c}{\partial t} = \frac{\partial}{\partial x} \left(D \frac{\partial c}{\partial x} \right), \quad c(0, t) = 1, \quad c(1, t) = 0, \quad c(x, 0) = 0, \quad D = e^{0.5c} \quad (9.67)$$

Heat Transfer

9.9. Solve the following heat transfer problem with radiation. The parameters are $G = 2$, $Nu = 10$.

$$\frac{d^2 T}{dr^2} + \frac{1}{r} \frac{dT}{dr} + G, \quad \frac{dT}{dr}(0) = 0, \quad -\frac{dT}{dr}(1) = Nu[T^4(1) - 1] \quad (9.68)$$

9.10. Solve the following transient problem and integrate to steady state. Do you get the same solution at steady state as found in Problem 9.9?

$$\frac{\partial T}{\partial t} = \frac{\partial^2 T}{\partial r^2} + \frac{1}{r} \frac{\partial T}{\partial r} + G, \quad \frac{\partial T}{\partial r}(0, t) = 0, \quad -\frac{\partial T}{\partial r}(1, t) = Nu[T^4(1, t) - 1], \quad T(r, 0) = 1 \quad (9.69)$$

9.11. Solve the following heat transfer problem. The parameters are $\rho = 491 \text{ lb}_m/\text{ft}^3$,

$$C_p = 0.11 \text{ Btu/lb}_m, \quad R = 0.4 \text{ ft}, \quad h = 30 \text{ Btu/h ft}^2 \text{ }^\circ\text{F}, \quad k = 3 \text{ Btu/h ft }^\circ\text{F},$$

$$T_0 = 120^\circ\text{F}, \quad T_{\text{surr}} = 68^\circ\text{F}$$

$$\rho C_p \frac{\partial T}{\partial t} = \frac{k}{r} \frac{\partial}{\partial r} \left(r \frac{\partial T}{\partial r} \right), \quad \frac{\partial T}{\partial r}(0, t) = 0, \quad -k \frac{\partial T}{\partial r}(R, t) = h[T(R, t) - T_{\text{surr}}],$$

$$T(r, 0) = T_0 \quad (9.70)$$

9.12. Solve the following heat transfer problem. The parameters are $\rho = 491 \text{ lb}_m/\text{ft}^3$, $C_p = 0.11 \text{ Btu/lb}_m$, $R = 0.4 \text{ ft}$, $U = 12.9 \text{ Btu/h ft}^2 \text{ }^\circ\text{F}$, $T_0 = 120^\circ\text{F}$, $T_{\text{surr}} = 68^\circ\text{F}$. Compare the average temperature vs time with the solution found in Problem 9.11.

$$\rho C_p \pi R^2 L \frac{dT}{dt} = -U 2 \pi R L (T - T_{\text{surr}}), \quad T(0) = T_0 \quad (9.71)$$

Electrical Fields

- 9.13.** The following (nondimensional) problem governs the electric potential outside a cylinder that is kept at potential Ψ_0 while the potential at a distance L away from the center of the cylinder is kept at zero. The parameter κ is 0.8, the applied potential is 2, and the radius of the cylinder is 1. Find the potential distribution around a cylinder with radius 1.0:

$$\frac{d^2\Psi}{dr^2} + \frac{1}{r} \frac{d\Psi}{dr} = \kappa^2 e^\Psi, \quad \Psi(R) = \Psi_0, \quad \Psi(L) = 0 \quad (9.72)$$

- 9.14.** The transient charge distribution in a semiconductor is governed by the following partial differential equation. Integrate the equation to steady state with $\beta = 1.5$:

$$\frac{\partial c}{\partial t} = \frac{\partial^2 c}{\partial x^2} - \beta e^c, \quad c(x, 0) = 0.5, \quad c(0, t) = 1, \quad c(1, t) = 1 \quad (9.73)$$

Fluid Flow

- 9.15.** Solve for the flow of a Newtonian fluid in a pipe, following the example. Plot the shear rate as a function of radial position. Calculate the average velocity. Plot the shear stress as a function of radial position.
- 9.16.** Solve the flow of a non-Newtonian fluid in a pipe, following the example, for pressure drops of 10, 10^3 , 10^5 , and 10^7 Pa. The parameters are $\eta_0 = 0.492$ Pa s, $\lambda = 0.1$ s⁻¹, and $n = 0.8$. Plot the shear rate as a function of radial position. Calculate the average velocity. Plot the shear stress as a function of radial position. How do these curves change as the pressure drop is increased?
- 9.17.** A fiber spinning problem is governed by the following equations from Middleman (1977):

$$\frac{d^2 u}{dx^2} = \frac{\rho}{3\mu} u \frac{du}{dx} + \frac{1}{u} \left(\frac{du}{dx} \right)^2, \quad u(0) = u_0, \quad u(L) = u_L \quad (9.74)$$

The ratio of the two velocities is the drawdown ratio,

$$D_R = \frac{u_L}{u_0} \quad (9.75)$$

When $\rho = 0$ is the solution for velocity and radius of the fiber is (Middleman, 1977, p. 237)

$$u = u_0 \exp\left(\frac{x \ln D_R}{L}\right), \quad R(x) = R_0 \exp\left(-\frac{1}{2} \frac{x \ln D_R}{L}\right) \quad (9.76)$$

Solve the problem without inertia ($\rho = 0$) using $u_0 = 16.7$ cm/s, $u_L = 1666.7$ cm/s, and $L = 300$ cm.

- 9.18.** Solve the fiber spinning problem (Problem 9.17) with $\rho = 0.96 \text{ g/cm}^3$ and $\mu = 5000$ poise. Is inertia important? How does the fiber radius compare with the solution to Problem 9.17?
- 9.19.** Solve the fiber spinning problem for a non-Newtonian fluid with viscosity that depends upon shear rate. The problem without inertia is

$$\frac{d}{dx} \left(\eta \frac{du}{dx} \right) = \frac{\eta}{u} \left(\frac{du}{dx} \right)^2, \quad u(0) = u_0, \quad u(L) = u_L, \quad \eta = \frac{\eta_0}{1 + \lambda^2 (du/dx)^2} \quad (9.77)$$

Solve it for $u_0 = 16.7 \text{ cm/s}$, $u_L = 1666.7 \text{ cm/s}$, $\eta_0 = 5000$ poise, $\lambda = 1 \text{ s}$, and $L = 300 \text{ cm}$.

10

FLUID FLOW IN TWO AND THREE DIMENSIONS

Most chemical processes involve fluids flowing in turbulent flow. However, polymer melts (the precursors to plastics) are so viscous that they flow in laminar flow. In modern chemical engineering, there are more and more applications in small devices in which laminar flow prevails. In microfluidics, these devices may be used for detection of biological or chemical weapons (after the payload is dispersed into the air), for medical diagnostics for cheap and rapid measurements, and for analytical tools used in process control. There are even applications where whole chemical processes are done on a microscale. This may be advantageous because the temperature can be carefully controlled (making a certain reaction possible), or the conversion can be increased due to precise mixing (minimizing waste products and recycle streams), or it may be advantageous to have distributed manufacture for safety reasons or in many countries. In microfluidics the dimensions of the channels are 1 mm or less (so small they are no longer called pipes), and the flow is laminar even in complex geometries. Thus, it is worthwhile to learn how to solve laminar flow problems in complex geometries.

Two methods are used in commercial computational fluid dynamics (CFD) codes: the finite volume method and the finite element method. To a beginner, it probably makes little difference which method is used. The author has used finite element methods for fluid flow for over 30 years, and that is the method used in FEMLAB[®], which is the CFD program illustrated here.

What does matter, though, is the difference between laminar and turbulent flow. A common misconception is that if you see vortices (swirling motion), the flow is turbulent. That is not necessarily true. In turbulent flow, there are fine-scale oscillations on time scales of 0.1 ms, and spatial fluctuations on the scale of 30 μm (30×10^{-6} m, or 3% of 1 mm). To model turbulent flow, one option is direct numerical simulation (DNS). In DNS, the Navier–Stokes equations are integrated in time on a very small spatial scale,

with millions of nodes; this is fundamental (no assumptions are made), but it is also very time-consuming. A DNS simulation for even simple problems (such as turbulent flow in a pipe) may take hours on a network of 16 or 32 computers to simulate what happens in seconds in reality. The computer needs to solve for three fluctuating velocities on a very small spatial scale and as a function of time. To overcome this problem, commercial CFD codes use k - ε models, which introduce new variables representing the turbulent kinetic energy (k) and rate of dissipation (ε), and add equations to predict them. The commercial codes then solve for the time-averaged velocities and use k and ε to represent the effect of the fluctuating velocities without solving for the fluctuating velocities. The equations have several empirical parameters that have been determined by comparison with experimental data, but these parameters change depending on the type of flow, and some uncertainty exists in realistic situations. The k - ε model also involves steep boundary layers near solid surfaces, and the CFD codes try to take that into account analytically. To summarize the k - ε model, it contains empiricism but is relatively fast to compute with. There is an intermediate method – large eddy simulation (LES) – which is more reliable than the k - ε models and uses parameters from DNS studies, but it too involves lengthy calculations, somewhere between DNS and k - ε .

This chapter focuses on fluid flow, leaving the combination of fluid flow, heat transfer, and diffusion to Chapter 11. Examples of fluid flow include entry flow into a pipe, flow in a microfluidic T-sensor, turbulent flow in a pipe, time-dependent start-up of pipe flow, flow in an orifice, and flow in a serpentine mixer. The examples demonstrate many of the techniques that are useful in the program FEMLAB.

MATHEMATICAL FOUNDATION OF FLUID FLOW

Navier–Stokes Equation

The flow equations are taken here as the Navier–Stokes equations, with some extensions to purely viscous non-Newtonian fluids. In vector form, the Navier–Stokes equation is

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho \mathbf{u} \cdot \nabla \mathbf{u} = -\nabla p + \mu \nabla^2 \mathbf{u} \quad (10.1)$$

This vector equation applies to the case when the gravitational term is absorbed into the pressure. It also takes the density as constant, leading to an incompressible fluid:

$$\nabla \cdot \mathbf{u} = 0 \quad (10.2)$$

For steady flows, the equation reduces to

$$\rho \mathbf{u} \cdot \nabla \mathbf{u} = -\nabla p + \mu \nabla^2 \mathbf{u} \quad (10.3)$$

In component notation in Cartesian geometry and two dimensions, these equations are

$$\rho \left(\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} \right) = -\frac{\partial p}{\partial x} + \mu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right)$$

$$\rho \left(\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} \right) = -\frac{\partial p}{\partial y} + \mu \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right)$$

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \quad (10.4)$$

Because the program FEMLAB includes all these terms, you merely need to define the problem you want to solve.

When evaluating pressure, one complication in the finite element method is a mathematical condition saying that the approximating functions for pressure have more restricted choices than those for velocity. This is called the LBB condition [named after Ladyshenskaya, Brezzi, and Babuska (Gresho et al., 1998, p. 593)], and the designers of the computer program have taken it into account for you. Another feature of the finite element method is that the computer problem can get very large. Often you will have to make compromises between the accuracy you would like and how long you are willing to wait for a solution. With modern computers, this is not as big a problem as it once was, but in complicated three-dimensional cases it is still a concern. Thus, one aspect of your solution method will be mesh refinement to estimate the accuracy.

The finite element method in two and three dimensions is similar to the finite element method in one dimension, except the bookkeeping is harder. However, again, FEMLAB does that book-keeping for us. Basically, the domain of the problem is covered by small triangles (in 2D) or small tetrahedrons (in 3D), and on each triangle (or tetrahedron) the dependent variable (u or v velocity or w) is approximated by a polynomial, even a straight line along the boundary. In two dimensions, the approximation is a triangular patch looking like a geodesic dome. Engineers and scientists have determined how to make the approximation satisfy the equations as well as possible, but that is beyond the scope of this book. See Appendix F for a brief description of the finite element method. When the solution is represented this way, you need to recognize that you have introduced an approximation, and the solution is no better than the mesh (or set of triangles or tetrahedrons) allows. Thus, you will usually solve the problem on more than one mesh, each one more refined than the last, to ensure that the results do not depend upon the mesh.

Table 10.1 gives a brief tutorial for applying the finite element method using FEMLAB. One decision you must make is whether to solve the equation in dimensional form or nondimensional form. The dimensional form is easier, but you miss characterizing the flow and you may not even calculate a Reynolds number. Furthermore, convergence can be enhanced using a nondimensional form. In the early days of the finite element method computers used only 8 or 16 bits to represent a number, and it was essential to make the equations dimensionless. While that is less important now, this chapter provides a brief description showing how to make the equations dimensionless. A primer on boundary conditions in fluid mechanics and FEMLAB is also given at the end of the chapter.

Non-Newtonian Fluid

For purely viscous non-Newtonian fluids, the viscosity is a function of shear rate, as illustrated in Chapter 9, and the steady-state equation must be written as

$$\rho \mathbf{u} \cdot \nabla \mathbf{u} = -\nabla p + \nabla \cdot [\eta (\nabla \mathbf{u} + \nabla \mathbf{u}^T)]$$

$$\eta = f\left(\frac{1}{2}\Delta : \Delta\right), \quad \Delta = \nabla \mathbf{u} + \nabla \mathbf{u}^T \quad (10.5)$$

$$\frac{1}{2}(\Delta : \Delta) = 2 \left[\left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial y}\right)^2 \right] + \left[\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right]^2 \quad (10.6)$$

The expression for the shear rate in 2D flow, Eq. (10.6) is obtained from Bird et al. (2002, p. 849). The expression for cylindrical geometry is Eq. (10.7):

$$\frac{1}{2}(\Delta : \Delta) = 2 \left[\left(\frac{\partial v_r}{\partial r}\right)^2 + \frac{v_r^2}{r^2} + \left(\frac{\partial v_z}{\partial z}\right)^2 \right] + \left[\frac{\partial v_r}{\partial z} + \frac{\partial v_z}{\partial r} \right]^2 \quad (10.7)$$

TABLE 10.1. Using FEMLAB to Solve Transport Problems (this example is for a 2D flow problem)

Open FEMLAB (Note: FEMLAB makes and reads .m and .fl files. The .m files can be used with MATLAB®.)

- Choose Axial Symmetry 2D;
- Choose Chemical Engineering/Momentum/Incompressible Navier–Stokes and click OK.

Draw

- Click on the square icon and draw a rectangle;
- Double click on the object to set exact dimensions;
- Use Option/Axis/Grid settings to set the plotting range on the screen.

Physics/subdomain settings

- Select domain 1; set $\rho = 0$, $\eta = 1180$ (Newtonian), $F_x = F_y = 0$.

Physics/boundary settings

- Click on a boundary number (1–4); (Note: The corresponding boundary is highlighted in red.)
- Set the boundary condition for each boundary segment. 1, slip/symmetry; 2, inflow/outflow, $v = 0.02$; 3, outflow/pressure; 4, no slip.

Mesh

- Click once on triangle icon or select mesh/initialize mesh;
- Note how many elements are used (you should report this);
- Click on divided triangle icon to refine the mesh if desired;
- To refine locally, click on the ‘refine selection’ icon, select some elements.

Solve

- Click on ‘=’ to solve the problem (click on  to re-start from the last solution).

Postprocessing

- Choose Domain Plot Parameters;
- Select the desired quantities for contour plots and surface plots;
- Or, click on the arrow plot icon, or streamline plot icon (flow plot);
- Choose Cross Section Plot Parameters to make line plots;
- Plot the v velocity along a line by selecting v velocity, put in the (r, z) coordinates of the beginning and ending points.

You can refine the mesh (click the more refined mesh symbol) and re-solve the problem. This gives you an indication of the accuracy of the solution. When you have a figure showing, you can export it in a figure format you choose, such as jpeg. You always want a figure showing the mesh, as well as parts of the solution.

To document your work, show the domain and mesh, give the dimensions, identify the boundary conditions, give values of parameters you used in the subdomain options, list how many elements and degrees of freedom were used, and indicate how solutions with different number of elements compare. Then give the results and indicate why they are reasonable.

Note that the notation is different from that used in FEMLAB. With the transformation

$$v_r \rightarrow u, v_z \rightarrow v, \frac{\partial v_r}{\partial r} \rightarrow ur, \frac{\partial v_r}{\partial z} \rightarrow uz, \frac{\partial v_z}{\partial r} \rightarrow vr, \frac{\partial v_z}{\partial z} \rightarrow vz \quad (10.8)$$

this is

$$\frac{1}{2}(\Delta : \Delta) = 2[ur^2 + (u/r)^2 + (vz)^2] + [uz + vr]^2 \quad (10.9)$$

EXAMPLE: ENTRY FLOW IN A PIPE

Next consider entry flow into a pipe, as illustrated in Figure 10.1. The flow is taken as uniform in radius at the entrance (taking the value 1.0 at each radial position), and the velocity has to develop into its fully developed profile. The pipe radius is taken as 0.5. How far downstream do you have to go to get a fully developed solution? (Another way to pose this question is: How much error do you make if you assume fully developed flow when it is not?)

Step 1 To solve this, enter FEMLAB and choose Chemical Engineering, Axisymmetric 2D/Momentum/Navier–Stokes/Stationary. In axisymmetric geometries, FEMLAB uses the horizontal axis as radius (called r) and the vertical axis as the length axis (called z). The corresponding velocity components are u in the radial direction and v in the axial direction. The sketch shows how the geometry is displayed.

Step 2 Click on the rectangle or use the Draw menu, then ‘rectangle.’ Put the cursor on a point and drag it to make a rectangle. If you double click on the rectangle, you can set the exact dimensions. Make the rectangle 0.5×2 .

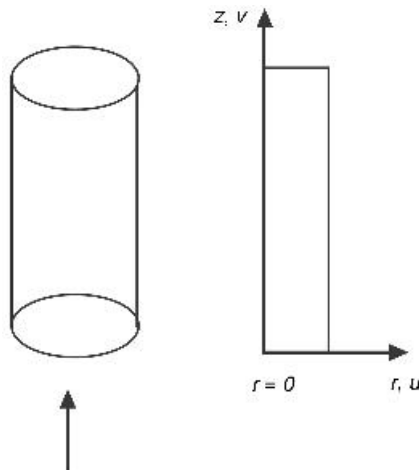


Figure 10.1. Entry flow into a pipe.

Step 3 Choose Options/Axis/Grid Settings to set the screen dimensions so you can see the rectangle easily. In Options/Axis/Grid Setting is a box to force the axes to be equal if you like.

Step 4 Click on the triangle twice to obtain the mesh shown in Figure 10.2a. There are 992 elements.

Step 5 Choose Physics/Subdomain Settings and a window appears showing Eq. (10.10):

$$\rho(\mathbf{u} \cdot \nabla)\mathbf{u} = \nabla \cdot [-p\mathbf{I} + \eta(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)] + \mathbf{F} \quad (10.10)$$

Set $\rho = 10$ and leave $\eta = 1$ (η is the same as viscosity, μ). Close that window.

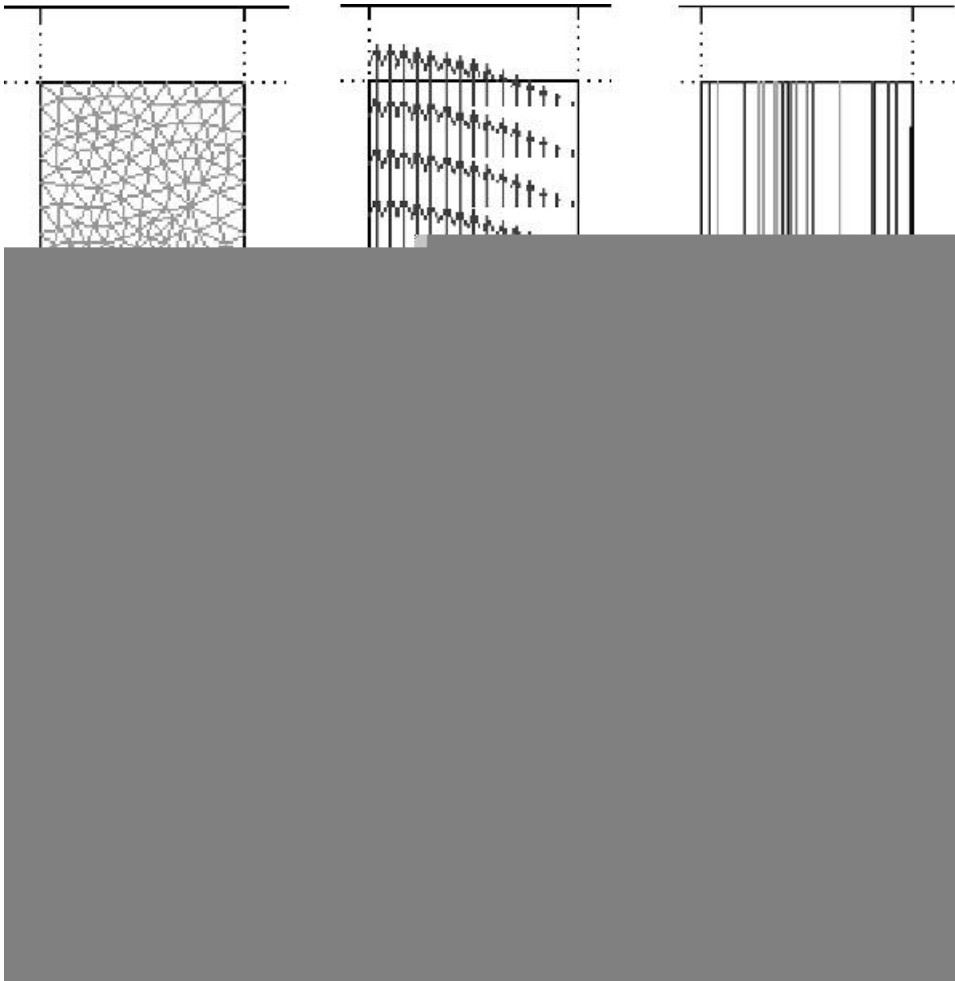


Figure 10.2. Mesh, arrow plot, and streamlines for entry flow into a pipe: (a) mesh; (b) arrow plot; (c) streamlines.

Step 6 Choose Physics/Boundary Settings. Set the boundary conditions as follows:

boundary 1: $n \cdot (u, v) = 0$	Slip/Symmetry	
boundary 2: set $u = 0, v = 1$	(v is the vertical velocity), Inflow/Outflow velocity	(10.11)
boundary 3: $t \cdot (u, v) = 0, p = 0$	Normal Flow/Pressure	
boundary 4: $u, v = 0$	No slip	

Step 7 Solve the problem (press '='). The degrees of freedom are displayed in the bottom window, 4657. To get a general idea of the flow, choose an arrow plot (see Figure 10.2b), which show the velocity vectors. Then choose a flow plot, which shows the streamlines (Fig. 10.2c).

Step 8 It is useful to plot the solution along a boundary surface. In the Postprocessing/Domain Plot Parameters, click on the Line/Extrusion tab and select the number 2 for boundary2, the inlet. Click Apply or OK to obtain Figure 10.3.

Step 9 To plot a variable along a line, choose Postprocessing/Cross-Section Plot Parameters. Click on the Line/Extrusion tab and choose the two points $(r, z) = (0, 0)$ and $(0.5, 0)$. Change the function plotted to be the z -velocity and click OK. The same picture arises, Figure 10.3. This method can be used to integrate over any line, not necessarily a boundary line.

Figure 10.3 shows that the velocity is not exactly 1 for all radii at the inlet, because there is a discontinuity in the boundary conditions. The boundary conditions use $v = 1$ on one boundary and $v = 0$ on an adjoining boundary, and the program has to choose which one to use at the corner. In this case, it chose $v = 0$, which created a sharp change in the v function at $r = 0.5$. This creates the 'bump.' By refining the mesh, you

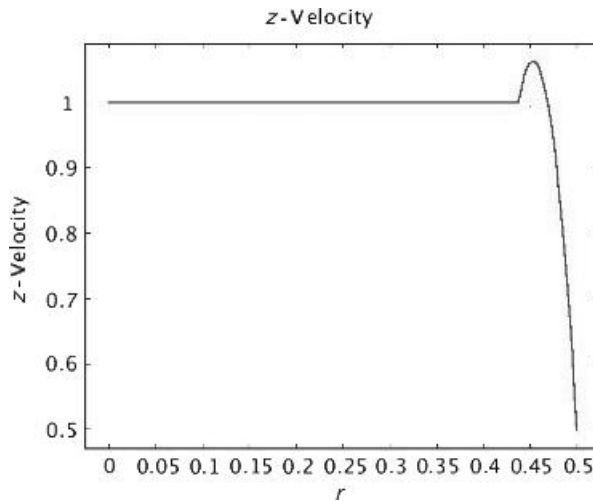


Figure 10.3. Velocity at inlet to pipe.

can make the bump smaller, but it cannot be eliminated by mesh refinement, because it is an error in posing the problem. (It is not possible to achieve those boundary conditions physically.)

Step 10 Return to the Line/Extrusion plot and plot along the points $(r, z) = (0, 0)$ and $(0, 2)$ to obtain Figure 10.4. This figure shows how quickly the center velocity approaches its asymptotic value as z increases. This gives you an indication of the ‘entry length.’

Step 11 Go back and plot along the points $(r, z) = (0, 2)$ and $(0.5, 2)$ or on boundary 3 to obtain Figure 10.5. This figure shows the outlet velocity. The exact solution is a quadratic function of position, which is satisfied in Figure 10.5. However, you also expected a peak value of 2.0. The peak value is not 2.0 because of the slight error in the entry conditions. What *is* true is that the flow rate out equals the flow rate in, which is slightly less than you desired due to the ‘bump’ in the inlet profile. This discrepancy decreases as you refine the mesh.

Step 12 To find the flow rates, use the Postprocessing/Boundary Integration option. Boundary 2 is the inlet. If you ask to integrate the v -velocity, the answer is 0.4948. Put 1 in the box and click Apply; the answer is 0.5. Since the velocity is 1.0 and the radius is 0.5, the v -velocity should integrate to 0.5, too. The error (1%) is due to the phenomena illustrated in Figure 10.3.

Step 13 Of course, the average velocity is given by Eq. (9.10). There are two ways you can compute it. The first way integrates once with v^*r in the expression window (giving 0.1224), and then with r in the expression window (giving 0.125). The average velocity is then $0.1224/0.125 = 0.979$. The second way is to click on the box ‘Compute surface integral (axisymmetric modes)’ (see Figure 10.6). Then the program includes the factor $2\pi r$ in the integral. With v in the expression window the answer is 0.7690, and with

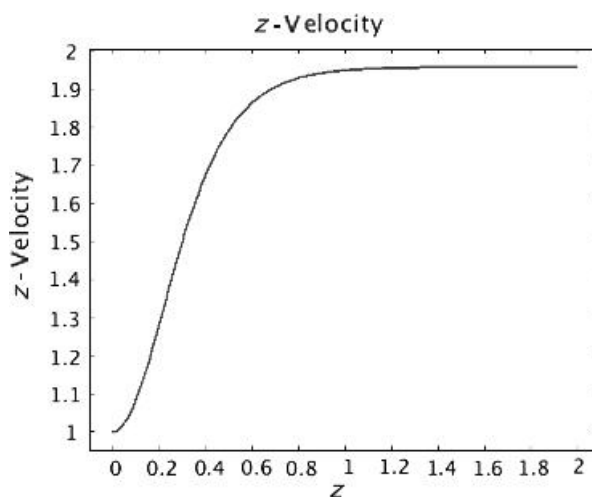


Figure 10.4. Centerline velocity in pipe flow.

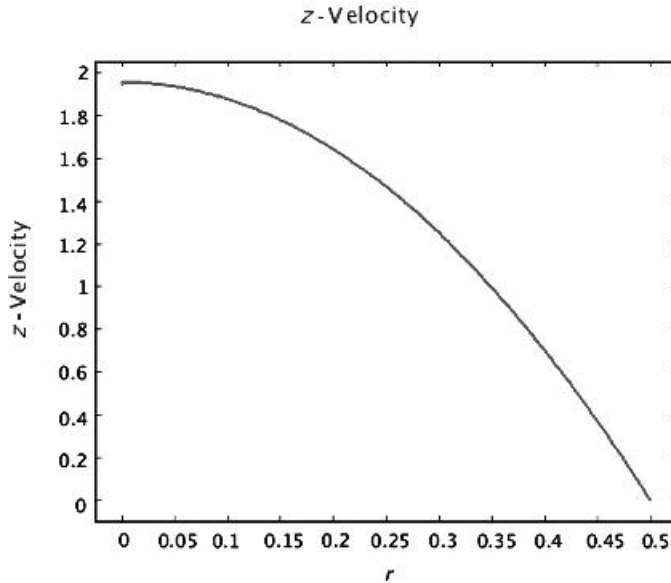


Figure 10.5. Velocity profile at exit.

1 in the expression window the answer is 0.7854. The average velocity is then $0.7690/0.7854 = 0.979$. Both answers are the same, of course. Checking the average velocity at the other end, boundary 3, gives the same value. Thus, continuity is observed even if the velocity has a ‘bump’ in Figure 10.3.

Next, change the problem slightly and take the inlet velocity as the fully developed velocity profile.

$$v(r, 0) = 2 \left[1 - \left(\frac{r^2}{0.5} \right) \right] \quad (10.12)$$

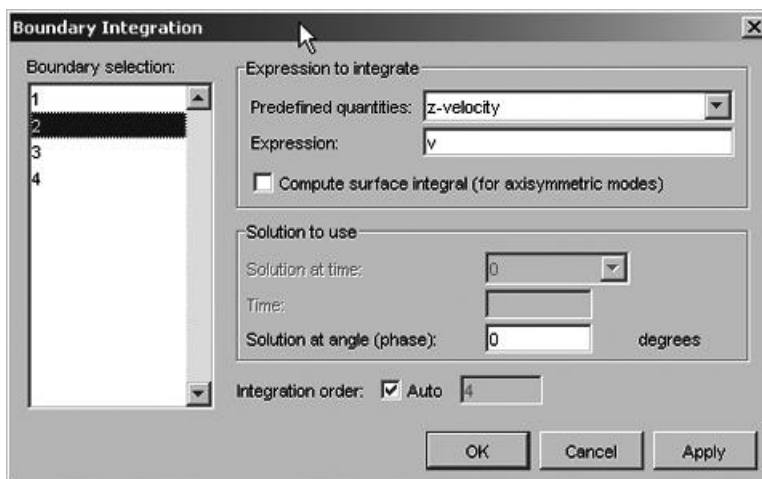


Figure 10.6. Boundary integration in cylindrical geometry.

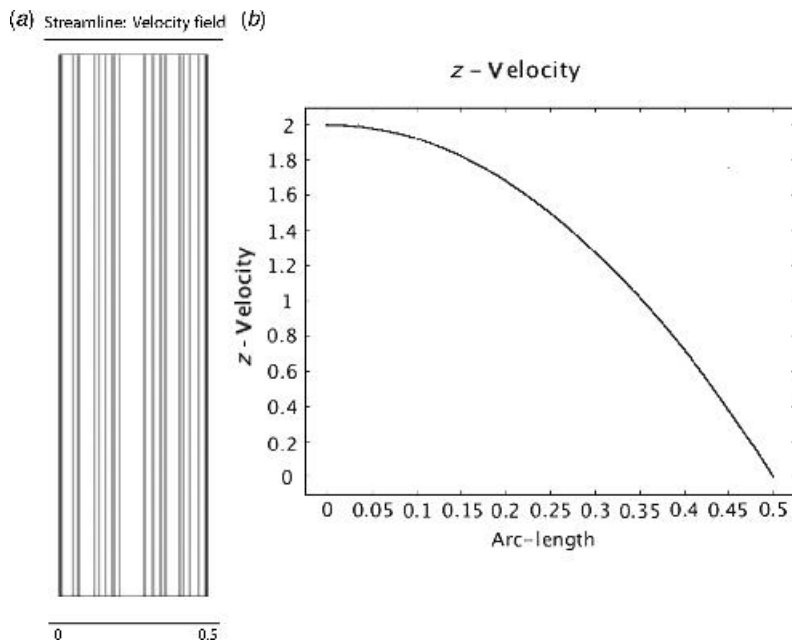


Figure 10.7. Streamlines (a) and inlet velocity profile (b) for flow in pipe with parabolic inlet velocity profile.

Step 14 This is inserted in Physics/Boundary Settings for boundary 2 as the v velocity:

$$2*(1 - (r/0.5)^2) \quad (10.13)$$

The streamlines obtained are shown in Figure 10.7a. Clearly they are straight, indicating that the fully developed flow just continues downstream. Compare Figure 10.7 with Figure 10.2c. The average velocity is now 1.0 exactly, $0.785398/0.785398$. The velocity profile at the inlet is shown in Figure 10.7b. The profile at the outlet is the same.

When running this last problem, the program might have said it did not converge. In fact it did, but since the exact solution was found in one iteration it was comparing the estimated error to zero. This is not an important problem (although it can be confusing the first time it happens to you) because you seldom have a two-dimensional problem in which you provide exact conditions. There is a way to change the iteration scheme if necessary. Use Solve/Solver Parameters and click on the Advanced tab. Change the scaling from Automatic to Manual and insert a nonzero value.

EXAMPLE: ENTRY FLOW OF A NON-NEWTONIAN FLUID

Next consider the entry flow of a non-Newtonian fluid, essentially the same problem as before, except with a different fluid. The only thing you need to change is the expression for viscosity, which is no longer constant. The shear viscosity is modeled as a Carreau

function, which is a slight extension of Eq. (9.14):

$$\frac{\eta - \eta_\infty}{\eta_0 - \eta_\infty} = [1 + (\lambda \dot{\gamma})^2]^{(n-1)/2} \quad (10.14)$$

The η_∞ , η_0 , λ , and n are all parameters that are used to fit data, taken here as $\eta_\infty = 0.05$, $\eta_0 = 0.492$, $\lambda = 0.1$, and $n = 0.4$. A plot of the viscosity vs shear rate is given in Figure 10.8. For small shear rates, the viscosity is essentially constant, as it is for a Newtonian fluid. For extremely large shear rates, the same is true. For moderate shear rates, though, the viscosity changes with shear rate. In pipe flow, or channel flow, the shear rate is zero at the centerline and reaches a maximum at the wall. Thus, the viscosity varies greatly from the centerline to the wall. This complication is easily handled in FEMLAB.

Once you know how to solve the problem for a Newtonian fluid (as in Fig. 10.2), there are two major changes you must make for a non-Newtonian fluid. One of those is to represent the viscosity, Eq. (10.14), in FEMLAB. The other is to be smart about how you obtain a converged solution.

Step 1 Insert the parameters for the viscosity under the Options/Constants and the equation for viscosity, using those parameters, in Options/Expressions/Subdomain Expressions. Then the shear rate is defined in terms of the parameters of the problem. Here the variables are u in the radial direction and v in the axial direction. Making the appropriate conversions to Eq. (10.7) gives Eq. (10.9), which you type as a subdomain expression.

Step 2 If the variable used for viscosity is ‘eta,’ then under the Physics/Subdomain Settings, you insert the word ‘eta’ is inserted in the box for η . That is all that is required to set up the non-Newtonian problem. As the problem is being solved, when the viscosity

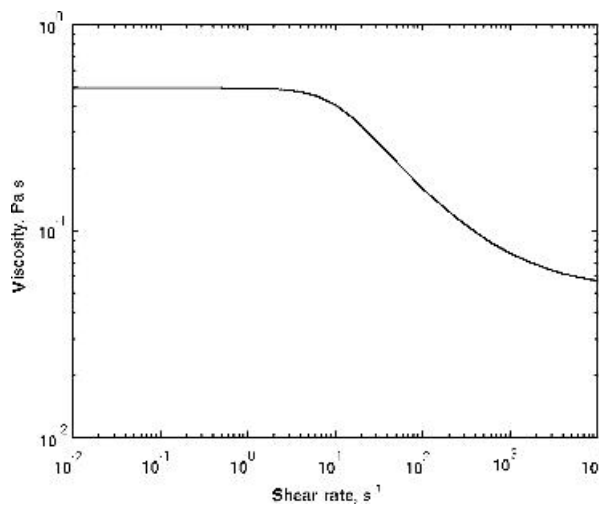


Figure 10.8. Viscosity of a shear-thinning fluid.

is needed, the computer looks at the expression for η – which involves another expression for shear rate and constants – and the computer program then evaluates the viscosity.

Step 3 The second change is more subtle. If you set up this problem, and press ‘=,’ probably the program will return a message saying it cannot converge. The reason is that the power-law form of the viscosity function is a very nonlinear function of the shear rate. The FEMLAB program uses derivatives of this with respect to the parameters, evaluated at the current solution iterate. Since initially you do not know the solution, that current solution is velocity zero everywhere except on boundaries where it has been set. Starting with a zero value may be the cause of the errors. Thus, you should always obtain a solution for a Newtonian fluid first, before invoking the non-Newtonian viscosity: simply replace the word ‘ η ’ with 1.0 or some number appropriate to your case, solve the problem, put back the ‘ η ,’ and use the re-solve option. This time the computer has a solution to work from, and the derivatives in the viscosity function do not give as much difficulty. You can practice this technique in Problems 10.2 and 10.3. FEMLAB also has a non-Newtonian module that you can try.

EXAMPLE: FLOW IN MICROFLUIDIC DEVICES

Next consider flow in what is called a T-sensor. Two flows come together, join, and traverse down one channel, as illustrated in Figure 10.9. This device is used in microfluidic medical devices, which are discussed further in Chapter 11. Here you will consider only the flow (which has no special utility until the convective diffusion equation is added in Chapter 11).

Step 1 Open FEMLAB and choose Chemical Engineering/Momentum Balance/Incompressible Navier–Stokes, Steady-state analysis. Draw a composite object and the mesh, as shown in Figure 10.10.

Step 2 In the first version of this problem, set the density and viscosity to 1.0 (corresponding to a nondimensional problem with Reynolds number = 1).

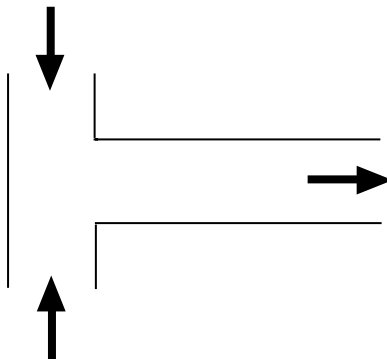


Figure 10.9. Flow in a T-sensor.

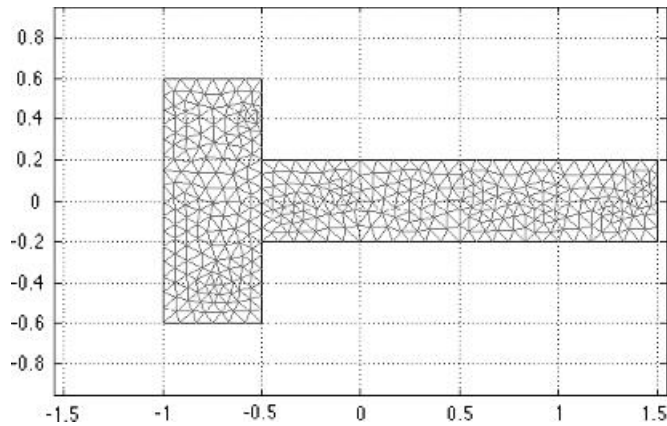


Figure 10.10. Mesh for flow in a T-sensor.

Step 3 Set the vertical velocity equal to 1 on the bottom, left boundary segment; -1 on the top, left boundary segment; use Normal flow/Pressure (0) on the right-hand side, and use no slip (zero velocity) everywhere else. The streamlines are shown in Figure 10.11.

Step 4 Use the Postprocessing option to check the solution. Choose Boundary Integration and select boundary 2, 3, or 8 (the two inflow boundaries and one outflow boundary). The values of various quantities are given in Table 10.2.

Step 5 To integrate 1.0 along the boundary, type 1.0 in the window and choose Apply. The number appears in the small screen at the bottom. Since the domain goes from 0 to 0.5

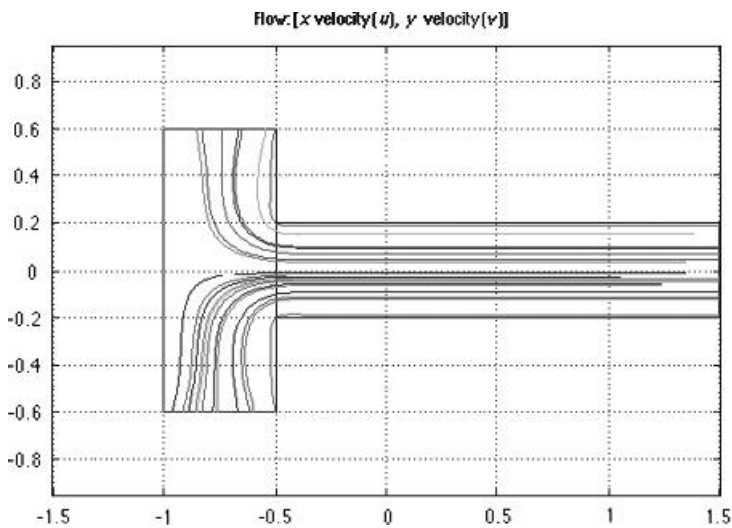


Figure 10.11. Streamlines for flow in a T-sensor.

**TABLE 10.2. Integral of Velocity Over Boundary
(multiply by depth to get flow rate)**

	Segment 2	Segment 3	Segment 8
u or v velocity	0.48611	-0.48611	0.97222
1.0	0.5	0.5	0.5

(half the diameter), the integral is 0.5:

$$\int_0^{0.5} 1.0 \, dx = 0.5 \quad (10.15)$$

Likewise, the velocity averages should be 0.5 on the inlets, because $v(x) = 1$:

$$\int_0^{0.5} v(x) \, dx = 0.5 \quad (10.16)$$

In this example, however, the velocity, is not exactly 0.5. This is because the boundary condition has a ‘bump’ at each corner, due to the discontinuity at the corner. The flow rate is twice as high at the outlet, which is correct because there are two inlets. The sum of the velocity integrals should equal the same integral for the outlet streams. Here

$$\text{inlet } v^*L = 0.48611 + 0.48611 = 0.97222, \quad (10.17)$$

which agrees with the outlet values. Thus, the mass flow rates balances.

Step 6 The streamlines in Figure 10.11 indicate the extent to which fully developed flow between two flat plates occurs as one moves from left to right. There is also some velocity rearrangement at each inlet because the velocity has been assumed to be ± 1 at the inlet, which is inconsistent with the no-slip boundary condition on each wall. This problem can easily be fixed. Instead of using $v = 1$ on the boundary, use Eq. (10.18), which has the



Figure 10.12. Streamlines in a T-sensor when the inlet velocity is parabolic.

same integral of velocity over the inlet. Now the entering streamlines are straight, as shown in Figure 10.12.

$$v = -24(x + 1)(x + \frac{1}{2}) \quad (10.18)$$

EXAMPLE: TURBULENT FLOW IN A PIPE

Next consider flow in a pipe at a flow rate high enough for turbulence to occur. Generally, this occurs whenever the Reynolds number is greater than 2200, where the Reynolds number is defined in terms of the average velocity and the pipe diameter:

$$Re = \frac{\rho \langle v \rangle D}{\mu} = \frac{\langle v \rangle D}{\nu} \quad (10.19)$$

You can solve the problem with an inlet velocity that is flat; thus you find the entry length it takes to achieve fully developed turbulent flow, and the velocity profile downstream is the fully developed one. When you solve for a kinematic viscosity of $10^{-6} \text{ m}^2/\text{s}$ (water), diameter of 0.05 m (about 2 inches) and a velocity of 2 m/s (a common optimal velocity), you will obtain a Reynolds number of 10^5 .

The turbulence model in FEMLAB is in dimensional (SI) units. Turbulence is modeled using the $k-\varepsilon$ model. In this model, the turbulent kinetic energy is represented by k , and the rate of dissipation of turbulent kinetic energy is represented by ε . Furthermore, the viscosity is augmented by a turbulent eddy viscosity, which is a function of k and ε . Special equations have been developed for both variables, and these must be solved along with the momentum equation which has the turbulent eddy viscosity in it as well. All these equations are included in FEMLAB.

The main caution is that these equations are only an approximation of reality, and the formulas and equations have been chosen to model experimental data. This means that they are only as good as the data they were derived from, and the data is usually found in relatively simple flows. The experiments are definitely not easy to carry out, but they are for idealized situations, such as fully developed flow in a pipe or past a flat plate or in a jet. Thus, the equations should be used with caution. Other methods in turbulence are much more time-consuming and may require banks of computers running for hours. That works for research, but is not suitable for day-to-day engineering work.

Step 1 The turbulent problem of flow in a pipe is set up in the same way as for laminar flow, except that the original choice of method is 2D Axial/Chemical Engineering/Momentum balance/ $K-\varepsilon$ Turbulence model. The model solves for the velocity components (averaged over a time interval) and the logarithms of k and ε . The reason is that the equations are better conditioned and the range of values of k and ε is very wide, but their logarithms are better behaved. Explicit, analytic expressions for k and ε are used at solid boundaries. Generally, you need elements near solid walls that are about the size of the wall-layer approximated by those explicit expressions. Those details are beyond the scope of this book, but you will need to study and understand them before using the results or doing many calculations.

Step 2 Set up the entry problem as was done for laminar flow, except put in a kinematic viscosity of 10^{-6} , a radius of 0.025, an inlet velocity of 2 m/s, and a length of 3 m. Note that the momentum equations has been divided by density:

$$\begin{aligned}(\mathbf{u} \cdot \nabla) \mathbf{u} &= -\nabla p / \rho + \nabla \cdot [(v + v_T) \nabla \mathbf{u}] \\(\mathbf{u} \cdot \nabla) k &= \tau_{ij} \partial u_i / \partial x_j - \varepsilon + \nabla \cdot [(v + v_T / \sigma_k) \nabla k] \\(\mathbf{u} \cdot \nabla) \varepsilon &= C_{\varepsilon 1} \frac{\varepsilon}{k} \tau_{ij} \partial u_i / \partial x_j - C_{\varepsilon 2} \frac{\varepsilon^2}{k} + \nabla \cdot [(v + v_T / \sigma_\varepsilon) \nabla \varepsilon]\end{aligned}\quad (10.20)$$

While $C_{\varepsilon 1}$ and $C_{\varepsilon 2}$ are dimensionless, the turbulent viscosity, v_T , is calculated in the SI system. Thus, all dimensions must be in m, m/s, etc.

Step 3 Start with a coarse mesh, and then refine it twice, giving 1424 elements and 13,729 degrees of freedom.

Step 4 Set the boundary conditions as follows. The centerline, inlet velocity, and exit velocity/pressure are set as in the laminar case: slip/symmetry, $v = 2$, Normal flow/Pressure, $p = 0$. The wall boundary condition, though, is set to the Logarithmic wall function. This is an analytic formula for the velocity, turbulent kinetic energy, and rate of dissipation, as determined by experiment (Deen, 1998, pp. 527–528).

Step 5 Solve the problem by pressing ‘=.’ The pressure drop per length is low by about 25%.

Step 6 Refine the mesh once more, with 5696 elements and 51,661 degrees of freedom. The exit velocity profile and centerline velocity are shown in Figure 10.13. Fully developed flow is achieved at a distance of about 2 m from the inlet. The profiles of k and ε are shown in Figure 10.14. The pressure drop per length in the fully developed region is 632 Pa/m, and the wall stress is $\tau_w = 7.9$ Pa. The friction

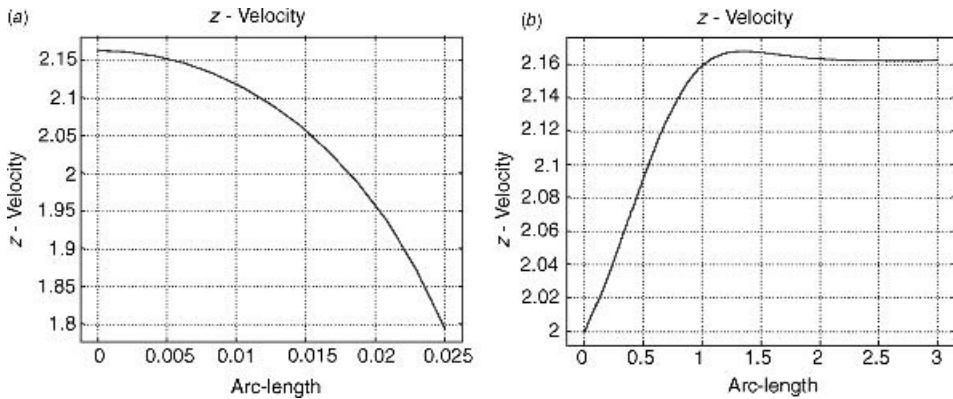


Figure 10.13. (a) Velocity profile at exit and (b) centerline velocity for turbulent entry flow in a pipe.

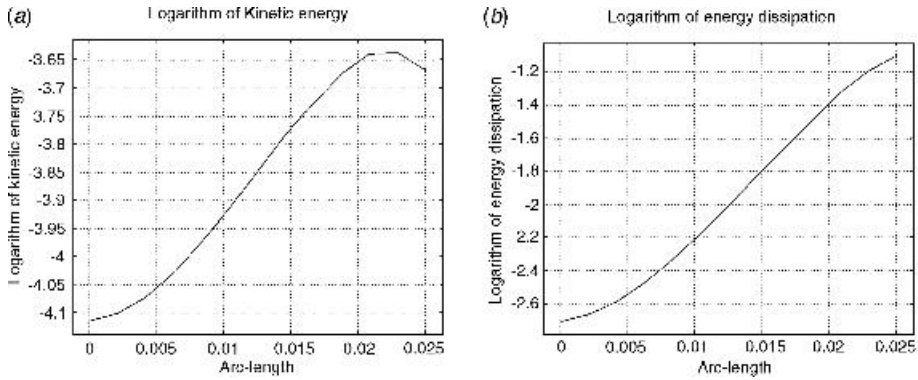


Figure 10.14. Turbulent kinetic energy (a) and rate of dissipation (b) for turbulent flow in a pipe (fully developed at the exit).

factor is then

$$\tau_w = \frac{\Delta p R}{2L}, \quad f = \frac{2\tau_w}{\rho \langle v \rangle^2} = \frac{\Delta p R}{L \rho \langle v \rangle^2} = 4.0 \times 10^{-3} \quad (10.21)$$

This is slightly lower than the value for a smooth pipe at a Reynolds number of 10^5 , where $f = 4.4 \times 10^{-3}$ (Perry's *Chemical Engineers' Handbook*, 1997, pp. 6–10). The pressure vs length down the pipe is shown in Figure 10.15.

EXAMPLE: START-UP FLOW IN A PIPE

A classic problem in fluid mechanics is the transient development of flow in a pipe. In this idealized problem, the fluid is initially at rest with no applied pressure gradient. At time

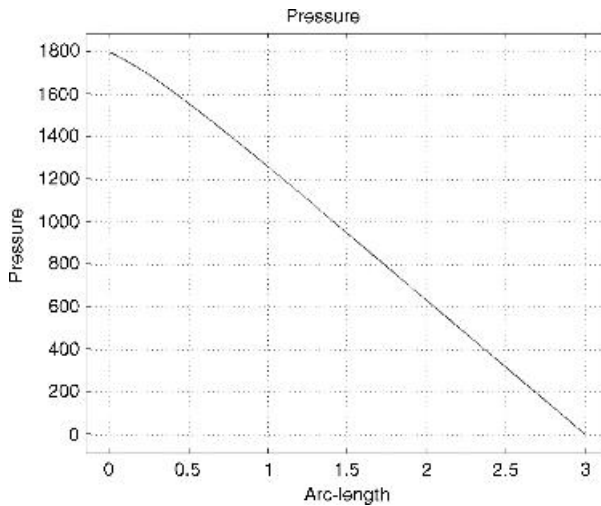


Figure 10.15. Pressure drop in turbulent flow.

zero, a pressure gradient is applied, and flow begins. The velocity is slow enough that the flow is laminar. After a certain amount of time, the flow becomes steady.

Step 1 To solve this problem in FEMLAB, choose Chemical Engineering Module/Momentum balance/Incompressible Navier–Stokes/Transient analysis in the Axial Symmetry (2D) option.

Step 2 Construct a domain with a radius of 0.0025 m and a length of 0.025 m. At this point, you do not know what the length should be, but it turns out that it does not matter.

Step 3 You solve for properties appropriate to water ($\rho = 1000 \text{ kg/m}^3$, $\mu = 0.001 \text{ Pa s}$) in a small pipe with a diameter of 0.005 m. The applied pressure drop is taken as 100 Pa/m. At steady state, the average velocity should then be 0.078 m/s, using Eq. (9.9). Under Subdomain Settings, put in the parameters. Then click the ‘Init’ tab and be sure that the velocity is initially zero.

Step 4 Set the boundary conditions as you did for entry flow: slip/symmetry at the centerline and no-slip on the solid wall. However, at the inlet choose the Normal flow/pressure option and put in the value 2.5 Pa. At the exit choose the Normal flow/pressure option and put in zero.

Step 5 How long do you need to compute? You can set that in Solve/Solver Parameters by clicking the Time-dependent Solver. The time is in seconds, and you think the flow will come to steady state within 10 s. Therefore, put Times as [0:0.1:1 2 3 4 5 6 7].

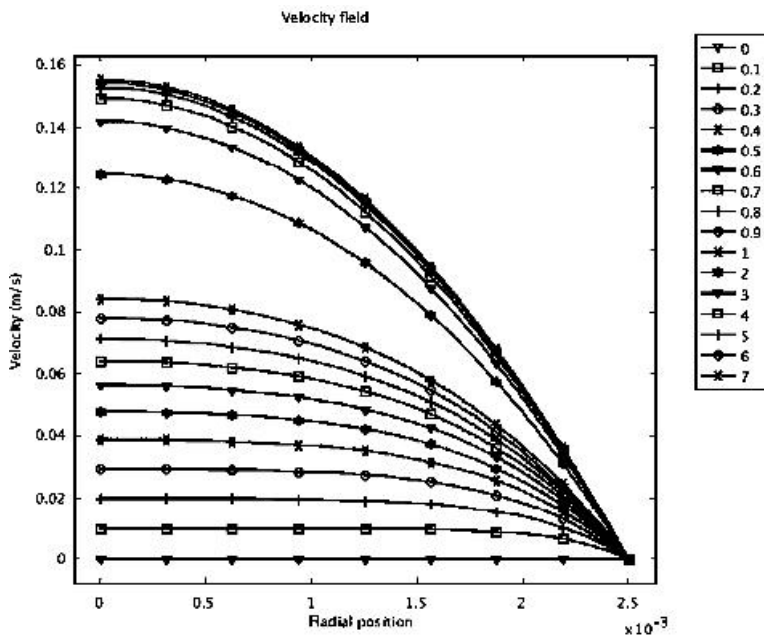


Figure 10.16. Transient laminar flow in a pipe, velocity profile.

Step 6 Click OK, and then ‘=.’

Step 7 After solution, choose Postprocessing/Domain Plot Parameters. Under the General tab, be sure all the time values are selected, choose Line extension, and change Line Settings to make the symbols cycle. If you choose the boundary 2 (the inlet), Figure 10.16 shows the inlet velocity as a function of radial position and time. The same plot is obtained at boundary 3 (the exit).

Step 8 To see the time behavior at a point, choose Postprocessing/Cross Sectional Plot Parameters and then click on the point tab. Plot the v -velocity at $r = 0$ and $z = 0$ as a function of time. (Note: All the time values need to be selected in the General window.) Figure 10.17 shows that the velocity at the corner increases and reaches steady state at about $t = 7$ s.

Step 9 Next, check the Reynolds number, which is 390, ensuring that the flow is laminar. The peak average velocity is 0.078 m/s, and in laminar flow the peak velocity should be twice that, or 0.156 m/s – which it is. If you would like to see a movie of the transient behavior, choose Postprocessing/Plot Parameters/Animate and select the variable you would like plotted. Animate is the last entry in the menu, and you may have to scroll across to see it.

EXAMPLE: FLOW THROUGH AN ORIFICE

The next problem is flow through an orifice. If the pipe or channel is very small, the flow may be laminar. This is usually the case in microfluidic devices. If the channel is small, the

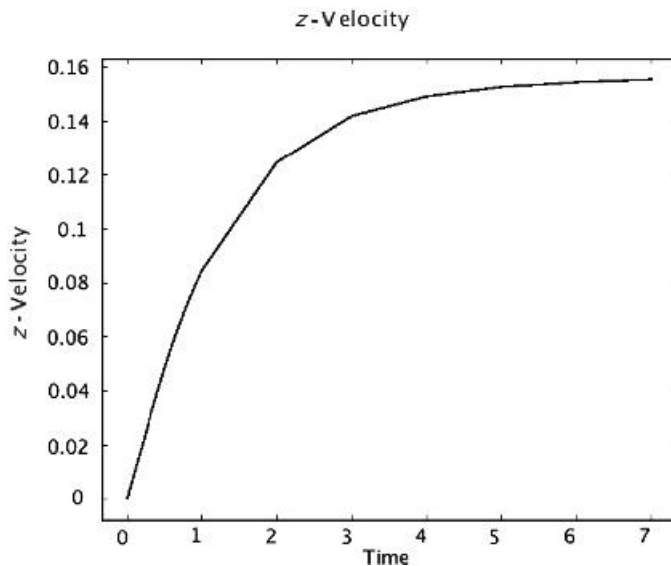


Figure 10.17. Transient laminar flow in a pipe, velocity at $r = 0$, $z = 0$.

orifice will not be infinitesimally small, and the width of the orifice may have an effect. You will solve for the flow here, while in Chapter 11 you will extend the problem to solve for the temperature rise due to viscous dissipation. In this chapter, you assume the viscosity is constant. This problem was solved by a senior chemical engineering student, Febe Kusmanto, while studying the effect of orifices with finite thicknesses (Kusmanto et al., 2004).

Step 1 It is convenient to solve this problem in nondimensional form, because you want solutions for Reynolds numbers from 1 to 1000. For the lower Reynolds numbers, you use the equations in the form

$$Re(\mathbf{u}' \cdot \nabla')\mathbf{u}' = \nabla' \cdot \{-p'\mathbf{I} + [\nabla'\mathbf{u}' + (\nabla'\mathbf{u}')^T]\} + \mathbf{F}' \quad (10.22)$$

Because the equation in FEMLAB is

$$\rho(\mathbf{u} \cdot \nabla)\mathbf{u} = \nabla \cdot \{-p\mathbf{I} + \eta[\nabla\mathbf{u} + (\nabla\mathbf{u})^T]\} + \mathbf{F} \quad (10.23)$$

you will take

$$\rho = Re, \eta = 1, \mathbf{u}' = \mathbf{u}/u_s, p' = px_s/\mu u_s, \mathbf{x}' = \mathbf{x}/x_s, \nabla' = x_s \nabla \quad (10.24)$$

where the variables with a subscript 's' are the standards chosen for velocity and distance. Here you use the average velocity as the velocity standard and the diameter of the orifice as the distance standard. Then, in the computer the average inlet velocity is 1.0.

Step 2 The orifice goes from $r = 0$ to 0.5, in a tube going from $r = 0$ to 3 with a length of 16 diameters. The orifice has a width of 0.15, for $L/D = 0.15$. The orifice goes from $z = 3.925$ to 4.075.

Step 3 For the higher Reynolds numbers, use the equations in the form

$$(\mathbf{u}' \cdot \nabla')\mathbf{u}' = \nabla' \cdot \left\{ -p''\mathbf{I} + \frac{1}{Re} \left[\nabla'\mathbf{u}' + (\nabla'\mathbf{u}')^T \right] \right\} + \mathbf{F}' \quad (10.25)$$

and take

$$\rho = 1, \eta = 1/Re, \mathbf{u}' = \mathbf{u}/u_s, p'' = p/\rho u_s^2, \mathbf{x}' = \mathbf{x}/x_s, \nabla' = x_s \nabla \quad (10.26)$$

The reason you need to do this is that the first method converges well for low Reynolds numbers, but not for high Reynolds numbers, and the reverse holds for the second method.

Step 4 The inlet velocity is taken as a parabola with an average velocity of 1.0:

$$\mathbf{u}' = \mathbf{e}_z v'(r'), \quad v'(r') = 2[1 - (r'/0.5)^2] \quad (10.27)$$

The other boundary conditions are Slip/Symmetry or Axial Symmetry on the centerline, No slip on the solid walls, and Normal flow/Pressure at the outlet velocity.

Step 5 You will construct and refine the mesh once. Since you expect significant flow disturbance near the orifice, the mesh is refined especially well there, as shown in Figure 10.18. This mesh has 5300 elements and gives rise to 24,543 degrees of freedom.

Step 6 In this calculation, you want to solve the problem for several Reynolds numbers. Doing this illustrates one of the important advantages of FEMLAB. In particular, suppose you wish to solve for Reynolds numbers from 1 to 1000. The eventual plot will be a log–log plot, and it is convenient to have equal increments in $\log Re$. Define the Reynolds number as

$$Re = 10^x \quad (10.28)$$

and choose x as $[0:0.1:3]$. Then the Reynolds number will go from 1 to 1000 in equal increments on the log–log plot. To do this in FEMLAB, in Subdomain Settings, replace Re with 10^{xRe} . Then choose Solve/Solver Parameters/Parametric nonlinear. In the ‘Name of parameter’ put xRe . In the ‘List of parameter values’ put 0:0.1:3 (see Figure 10.19). Then the Reynolds number takes values of

$$Re = 1, 1.259, 1.585, \dots, 10, 12.59, 15.85, \dots, 125.9, 158.5, \dots, 794.3, 1000 \quad (10.29)$$

Sometimes it may be necessary to use x from 0 to 1.5 using Eq. (10.22) and then x from 1.5 to 3 using Eq. (10.25).

Step 7 When you click ‘=,’ FEMLAB solves the problem for every Reynolds number. A convenient feature of the finite element method is that, once you have the solution, it is a trivial task to obtain the derivative of the solution with respect

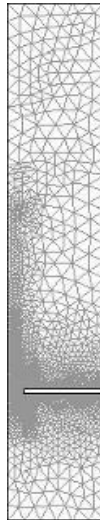


Figure 10.18. Mesh for flow through an orifice.

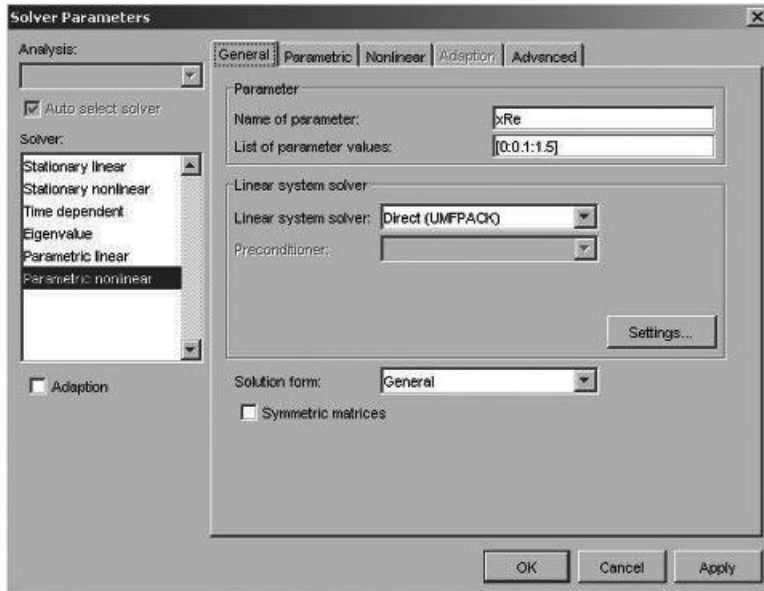


Figure 10.19. Solver parameters screen for $Re = 10^0$ to $10^{1.5}$.

to a parameter, here xRe . Then you can obtain a good guess for the next solution by using linear extrapolation:

$$\mathbf{u}(r, z, xRe_2) = \mathbf{u}(r, z, xRe_1) + \left. \frac{\partial \mathbf{u}(r, z, xRe)}{\partial xRe} \right|_{xRe_1} (xRe_2 - xRe_1) \quad (10.30)$$

Notice during the solution process that only a few iterations are needed for each Reynolds number, because Eq. (10.30) gives a very good initial guess. You can, of course, start with zero velocity everywhere, but the technique described is faster and more likely to converge.

Figure 10.20 shows the streamlines at two Reynolds numbers. This figure illustrates the fact that recirculation occurs, and the region of recirculation increases as the Reynolds number increases, even though the flow is still laminar. Figure 10.21 shows the corresponding pressure profiles. The pressure changes occur very near the orifice, and the pressure drop is almost entirely caused by the orifice with only a small contribution from fully developed flow in the tube. As the Reynolds number increases, the pressure reduces after the orifice. Then, it gradually returns to a higher value in the region known as the *vena contracta*.

Figures 10.22 and 10.23 show the pressure drop as a function of Reynolds number. These results are plotted two ways, corresponding to the two different ways to nondimensionalize the equations. That in Figure 10.22 is most suitable for slow speed laminar flow, because the pressure term plotted is constant with Reynolds number over quite a wide range. That in Figure 10.23 is most suitable for high Reynolds numbers because that form of the pressure term is constant once the Reynolds number gets high enough.

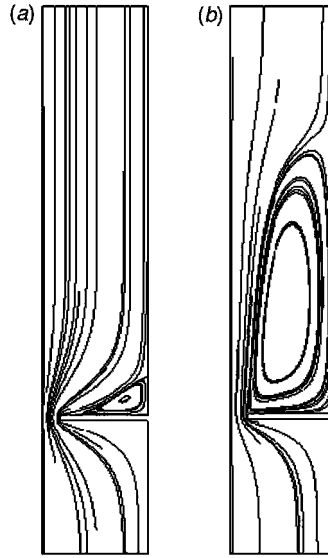


Figure 10.20. Streamlines for $Re = 1$ (a) and $Re = 10^{1.5} = 31.6$ (b).

Figure 10.23 also looks like a typical friction factor plot. Both curves, however, give the same value of pressure drop in pascals, and the plotted coefficients are related by Eq. (10.31):

$$\frac{\Delta p}{\frac{1}{2}\rho\langle v \rangle^2} \frac{Re}{2} = \frac{\Delta p}{\rho\langle v \rangle^2} \frac{\rho\langle v \rangle D}{\mu} = \frac{\Delta p D}{\mu\langle v \rangle} \quad (10.31)$$

The experimental data is from Hasegawa et al. (1997) and the dotted line is the analytical solution to Stokes flow by Dagan et al. (1982).

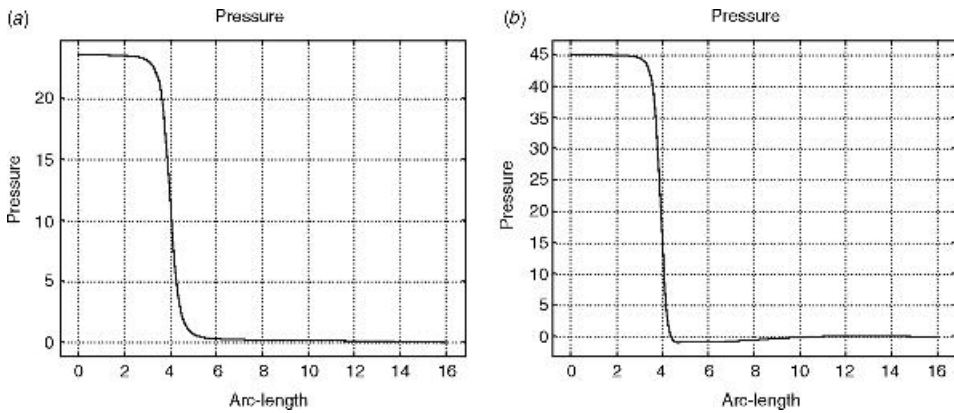


Figure 10.21. Pressure profiles for $Re = 1$ (a) and $Re = 10^{1.5} = 31.6$ (b).

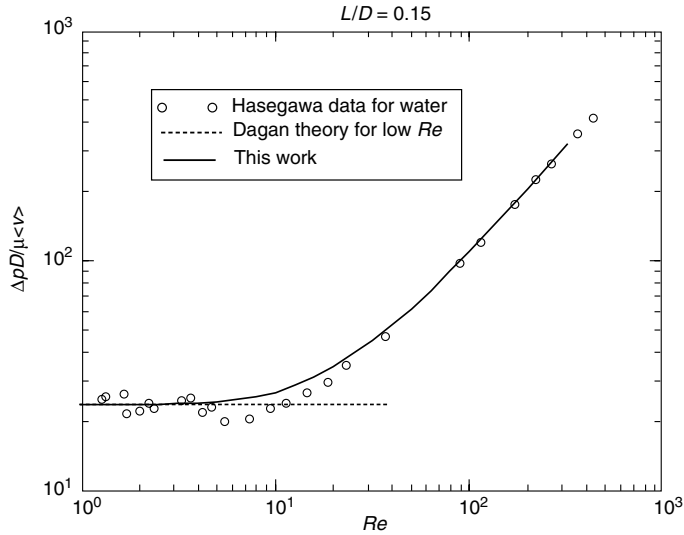


Figure 10.22. Dimensionless pressure drop in orifice for low Reynolds number.

As the Reynolds number increases, you must decide if the length of the domain is sufficiently long that the velocity profiles and pressure profiles have returned to their fully developed values. Another way to determine this is to solve the problem on a longer domain and compare the answers from the problems with different lengths. Because the pressure drop should be the same, as long as the pressure drop from the fully developed flow is negligible or taken into account, this is a valid test.

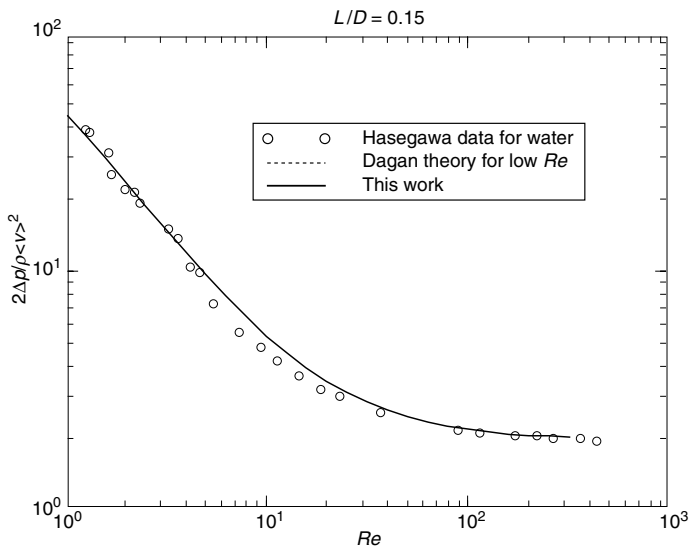


Figure 10.23. Dimensionless pressure drop in orifice for high Reynolds number.

EXAMPLE: FLOW IN A SERPENTINE MIXER

Slow flow in the serpentine mixer pictured in Figure 10.24 is used to mix two chemicals that come in at different sides of the port to the left. This is a complicated geometry and pushes the simulation capabilities, but it is easily solved in FEMLAB. It was solved by Zachery Tyree when he was a senior chemical engineering student at the University of Washington (Neils et al., 2004). A few flow lines are plotted in Figure 10.24, showing how the mixing of the fluid streams occurs as the fluid moves along the channels, turns, and goes up or down a chimney. The mesh is shown in Figure 10.25; it used 36,320 elements and 203,005 degrees of freedom to solve for the three velocity components and pressure.

BOUNDARY CONDITIONS

FEMLAB has many options for boundary conditions. The finite element method is based on integrating the equations by parts and applying the divergence theorem. Thus, the allowable boundary conditions are really determined by the equations. The basic conditions of fluid mechanics are that either the velocity or forces must be specified. In a three-dimensional problem, you would need to specify three velocities, for example, at each boundary point, or some combination of velocities and forces. The finite element method provides the following boundary conditions (Finlayson, 1992):

$$\text{either } \mathbf{n} \cdot \mathbf{u} = u_n \quad \text{1a} \quad \text{or} \quad \mathbf{n} \cdot \boldsymbol{\tau} \cdot \mathbf{n} = -p + \mu \mathbf{n} \cdot [\nabla \mathbf{u} + (\nabla \mathbf{u})^T] \cdot \mathbf{n} = f_n \quad \text{1b} \quad (10.32)$$

$$\text{either } \mathbf{t} \cdot \mathbf{u} = u_t \quad \text{2a} \quad \text{or} \quad \mathbf{n} \cdot \boldsymbol{\tau} \cdot \mathbf{t} = \mu \mathbf{n} \cdot [\nabla \mathbf{u} + (\nabla \mathbf{u})^T] \cdot \mathbf{t} = f_t \quad \text{2b} \quad (10.33)$$

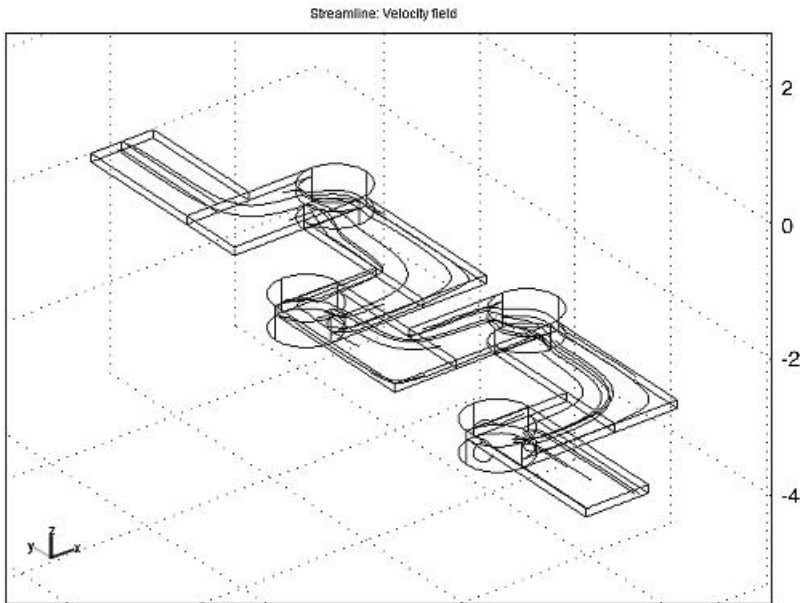


Figure 10.24. Flow lines in serpentine mixer.

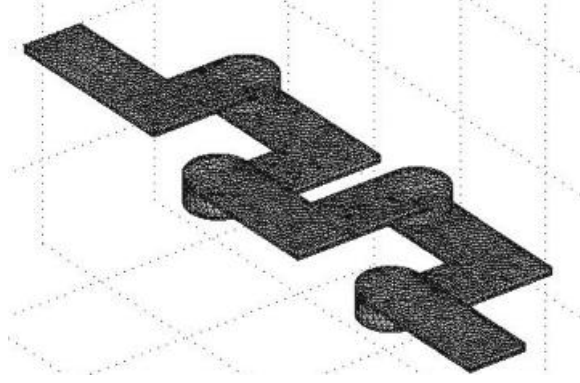


Figure 10.25. Mesh used to model the serpentine mixer.

Equation (10.32) is for the normal velocity or normal force, and Eq. (10.33) is for the tangential velocity or tangential force. The normal and tangent to the surface are determined by the geometry. The quantities u_n , f_n , u_t and f_t must be specified, but can be zero. These quantities represent the interaction between the system and its surroundings. The boundary conditions available in FEMLAB are listed in Table 10.3. At any given boundary, one

TABLE 10.3. Boundary Conditions for the Navier–Stokes Equation

Inflow/outflow velocity, 1a and 2a – set u_n , u_t

$$\mathbf{n} \cdot \mathbf{u} = u_n \quad \text{and} \quad \mathbf{t} \cdot \mathbf{u} = u_t \quad (10.34)$$

Outflow/pressure, 1b and 2b – can set p_0

$$\begin{aligned} -p + \mu \mathbf{n} \cdot [\nabla \mathbf{u} + (\nabla \mathbf{u})^T] \cdot \mathbf{n} &= -p_0 \\ \mu \mathbf{n} \cdot [\nabla \mathbf{u} + (\nabla \mathbf{u})^T] \cdot \mathbf{t} &= 0 \end{aligned} \quad (10.35)$$

Slip/symmetry, 1a and 2b – can set nothing

$$\begin{aligned} \mathbf{n} \cdot \mathbf{u} &= 0 \\ \mu \mathbf{n} \cdot [\nabla \mathbf{u} + (\nabla \mathbf{u})^T] \cdot \mathbf{t} &= 0 \end{aligned} \quad (10.36)$$

No slip, 1a and 2a – can set nothing

$$\mathbf{n} \cdot \mathbf{u} = 0 \quad \text{and} \quad \mathbf{t} \cdot \mathbf{u} = 0 \quad (10.37)$$

Normal flow/pressure, 1b and 2a – can set p_0

$$\begin{aligned} -p + \mu \mathbf{n} \cdot [\nabla \mathbf{u} + (\nabla \mathbf{u})^T] \cdot \mathbf{n} &= -p_0 \\ \mathbf{t} \cdot \mathbf{u} &= 0 \end{aligned} \quad (10.38)$$

Neutral, 1b and 2b – can set nothing

$$\begin{aligned} -p + \mu \mathbf{n} \cdot [\nabla \mathbf{u} + (\nabla \mathbf{u})^T] \cdot \mathbf{n} &= 0 \\ \mu \mathbf{n} \cdot [\nabla \mathbf{u} + (\nabla \mathbf{u})^T] \cdot \mathbf{t} &= 0 \end{aligned} \quad (10.39)$$

Axial symmetry – same as Slip/symmetry

Then the nondimensional Navier–Stokes equation is

$$\frac{\partial \mathbf{u}'}{\partial t'} + Re \mathbf{u}' \cdot \nabla' \mathbf{u}' = -\nabla' p' + \nabla'^2 \mathbf{u}', \quad Re = \frac{\rho u_s x_s}{\mu} \quad (10.46)$$

An alternative is to use a different standard for time,

$$t_s = \frac{x_s}{u_s}, \quad \text{giving} \quad \frac{\rho x_s^2}{\mu t_s} = \frac{\rho u_s x_s}{\mu} = Re \quad (10.47)$$

in which case the nondimensional Navier–Stokes equation is

$$Re \frac{\partial \mathbf{u}'}{\partial t'} + Re \mathbf{u}' \cdot \nabla' \mathbf{u}' = -\nabla' p' + \nabla'^2 \mathbf{u}' \quad (10.48)$$

If you want the pressure in Pascals, take the nondimensional pressure (from the computer) and multiply it by the pressure standard:

$$p = p_s p' = \frac{\mu u_s}{x_s} p' \quad (10.49)$$

Option 2 This option is most applicable for a high Reynolds number. Multiply Eq. (10.43) by $x_s/\rho u_s^2$ to give

$$\frac{x_s}{t_s u_s} \frac{\partial \mathbf{u}'}{\partial t'} + \mathbf{u}' \cdot \nabla' \mathbf{u}' = -\frac{p_s}{\rho u_s^2} \nabla' p'' + \frac{\mu}{\rho u_s x_s} \nabla'^2 \mathbf{u}' \quad (10.50)$$

The symbol p'' is used now for option 2. Choose the pressure standard so that the coefficient of the pressure gradient term is 1.0 and make a similar choice for the time standard:

$$\frac{x_s}{t_s u_s} = 1, \quad t_s = \frac{x_s}{u_s} \quad \text{or} \quad \frac{p_s}{\rho u_s^2} = 1, \quad \text{or} \quad p_s = \rho u_s^2 \quad (10.51)$$

Then the nondimensional Navier–Stokes equation is

$$\frac{\partial \mathbf{u}'}{\partial t'} + \mathbf{u}' \cdot \nabla' \mathbf{u}' = -\nabla' p'' + \frac{1}{Re} \nabla'^2 \mathbf{u}' \quad (10.52)$$

If you want the pressure in Pascals, take the nondimensional pressure (from the computer) and multiply it by the pressure standard:

$$p = p_s p'' = \rho u_s^2 p'' \quad (10.53)$$

By comparing Eq. (10.48) with Eq. (10.52), you see that the two different dimensionless pressures are related by

$$p' = Re p'' \quad (10.54)$$

It is easy to show that the pressure in Pascals is the same in both options. The only reason for having both of them is to improve the rate of convergence, or make convergence

possible in some cases. The dimensional pressure drop in option 1 is given by Eq. (10.49). This is also

$$\Delta p = p_s \Delta p' = \frac{\mu u_s}{x_s} \Delta p' = \frac{\rho u_s^2}{Re} \Delta p', \text{ since } \frac{\rho u_s^2}{Re} = \frac{\mu u_s}{x_s} \quad (10.55)$$

The dimensional pressure drop in option 2 is given by Eq. (10.53). Using Eq. (10.54) you get

$$\Delta p = \rho u_s^2 \Delta p'' = \frac{\rho u_s^2}{Re} \Delta p' \quad (10.56)$$

which is the same as Eq. (10.55).

CHAPTER SUMMARY

This chapter illustrated how to use FEMLAB to solve the Navier–Stokes equations in a variety of situations. Some of these problems are classic, such as entry flow into a pipe and transient start-up of pipe flow. Most examples were for laminar flow in two dimensions, but one model was turbulent flow into a pipe, and another model was for a complicated three-dimensional geometry. To review, the chapter covered the following specific features:

- Axisymmetric geometry;
- Using a velocity profile at a boundary;
- Variable viscosity, depending upon shear rate (non-Newtonian fluid);
- Calculating averages;
- Plotting the solution;
- Checking continuity;
- Time-dependent models;
- Parametric solvers;
- Choosing of boundary conditions; and
- Using nondimensional forms of the equations.

With the skills you learned in this chapter, you will be ready to tackle complicate two- and three-dimensional flow problems.

PROBLEMS

- 10.1.** Solve for the entry flow of a Newtonian fluid into a pipe using the parameters in the example. Reproduce Figure 10.2 through Figure 10.5 plus Figure 10.7.
- 10.2.** Solve for the entry flow of a power-law fluid into a region between two flat plates. The viscosity is given by

$$\eta = 0.2 \dot{\gamma}^{-0.6} \quad (10.57)$$

The plates are a distance of 0.005 m apart, and the applied pressure gradient is 60 Pa in a length of 0.3 m. The density of the fluid is that of water. (Hint: Follow the steps in the example of entry flow in a pipe, except use a power-law formula for viscosity and use two flat plates rather than a pipe.)

- 10.3.** Solve Problem 10.2, but use the Carreau function, Eq. (10.14), with $\eta_\infty = 0.492$, $\eta_0 = 0.05$, $\lambda = 0.1$, and $n = 0.4$. (Hint: Follow the steps in the example of entry flow in a pipe, but use the Carreau function for viscosity and use two flat plates rather than a pipe.)
- 10.4.** Compare the Carreau viscosity as a function of shear rate (Fig. 10.8 and Problem 10.3) with that of the power-law fluid in Problem 10.2.
- 10.5.** Solve for the start-up flow of the power-law fluid and problem described in Problem 10.2. How long does it take to reach steady state?
- 10.6.** Solve for the start-up flow of the Carreau fluid and problem described in Problem 10.3. How long does it take to reach steady state?
- 10.7.** Solve for the flow through an orifice. Reproduce Figures 10.20 and 10.21 for Reynolds numbers from 1 to 31.6 using the parametric solver. Plot the pressure coefficient vs Reynolds number, as in Figure 10.22.
- 10.8.** Solve the example of a microfluidic T-sensor, except have the average velocity coming in the bottom twice as high as in the example. Compare the flow rates in both streams and out to ensure that the total flow in equals the total flow out.
- 10.9.** Consider laminar flow past a flat plate. Create a geometry as a long rectangle. Set the velocity to zero on the plate, and set it to a constant on the left. Use a neutral boundary at the top and convective flux at the right-hand side. Compare your solution with the Blasius solution in your textbook.
- 10.10.** Solve for the drag on a sphere in a flowing stream with a uniform velocity profile upstream. Solve for zero Reynolds number (Stokes flow). Compare the solution with an analytical solution in your textbook. (Hint: Set the density to zero to simulate Stokes flow. The drag is obtained by integrating certain stresses over the boundary of the sphere.)
- 10.11.** Solve for the drag on a sphere in a flowing stream with a uniform velocity profile upstream. Solve for Reynolds number from 1 to 100. Put the sphere in a cylindrical tube, but use neutral boundary conditions on the tube (this will mimic an infinite domain). How does the qualitative behavior of the solution change with Reynolds number?
- 10.12.** Solve for the drag on a sphere in a flowing stream with a uniform velocity profile upstream. Solve for Reynolds number from 1 to 100. Put the sphere in a cylindrical tube with a diameter of 5. For this problem, use zero velocity on the cylindrical tube, which mimics the effect of the wall.
- 10.13.** Use the problem set up in Problem 10.12 and deduce the wall effect when measuring the terminal velocity by dropping a sphere into a fluid. Compare with Perry and Green (1997, p. 6–54).

- 10.14.** Solve for the drag on a cylinder in a flowing stream with a uniform velocity profile upstream. Solve for Reynolds number from 1 to 100. Far from the cylinder use neutral boundary conditions (this will mimic an infinite domain). How does the qualitative behavior of the solution change with Reynolds number?
- 10.15.** Solve for turbulent flow in a pipe using the parameters in the example. Use velocities of (1) 0.2 m/s; (2) 0.6 m/s; and (3) 1.4 m/s. Compare the results with Figures 10.13 and 10.14.

11

CONVECTIVE DIFFUSION EQUATION IN TWO AND THREE DIMENSIONS

Chemical reaction and mass transfer are two unique phenomena that help define chemical engineering. Chapter 8 described problems involving chemical reaction and mass transfer in a porous catalyst, and how to model chemical reactors when the flow was well defined, as in a plug-flow reactor. Those models, however, did not account for the complicated flow situations sometimes seen in practice, where flow equations must be solved along with the transport equation. Microfluidics is the chemical analog to microelectromechanical systems (MEMS), which are small devices with tiny gears, valves, and pumps. The generally accepted definition of microfluidics is flow in channels of size 1 mm or less, and it is essential to include both distributed flow and mass transfer in such devices.

Microfluidic devices often are smaller than a penny, so a new name has been invented for them, 'lab-on-a-chip.' The pharmaceutical industry uses these devices for drug design, delivery, and detection, and the biomedical industry uses them for sensors and drug delivery. Industrial applications include the study of new reaction mechanisms and pathways that lead to improved selectivity with fewer pollutants. Because of the small size of microfluidic devices, a precise control of temperature is possible, which improves kinetic information, especially with catalysts. Some elements of a fuel cell are small enough to be microfluidic devices, too. If large-scale production is required, engineers use 'number up' rather than 'scale up' to achieve high throughput. The techniques of the electronic industry can be used to produce many small devices cheaply, and engineers simply add 'clones' to increase the production rate of chemicals. The following articles give details of these applications: Jensen (1999); Stone and Kim (2001); Chow (2002); Freemantle (2005); Henry (2005), Gokhale et al. (2005), and a useful website is www.Lab-on-a-Chip.com.

This chapter uses FEMLAB[®] to solve problems involving two- or three-dimensional flow (2D or 3D) and diffusion of mass and/or heat. Convection and diffusion of mass

and heat are often governed by the convective diffusion equation. The flow problems were treated in Chapter 10, but sometimes parameters such as viscosity and density depend upon the concentration and temperature. In that case, all the partial differential equations have to be solved together. This chapter illustrates how to use FEMLAB to solve problems in the geometries discussed in Chapter 10: the T-sensor, an orifice, and the serpentine mixer. You will learn how to create a composite geometry, solve the equations either sequentially, one after the other, or simultaneously, make the transport properties such as viscosity functions of temperature or concentration, and learn to assess the relative importance of convection and diffusion, which is essential for interpreting numerical results.

The chapter begins by listing the equations and the boundary conditions available in FEMLAB. The first example is simple heat conduction in a two-dimensional region, followed by applications to microfluidic devices, viscous dissipation in orifice flow, and diffusion in the serpentine mixer.

CONVECTIVE DIFFUSION EQUATION

The equation governing mass transfer with flow is

$$\frac{\partial c}{\partial t} + \mathbf{u} \cdot \nabla c = \nabla \cdot (D \nabla c) \quad (11.1)$$

When the diffusivity is constant it is:

$$\frac{\partial c}{\partial t} + \mathbf{u} \cdot \nabla c = D \nabla^2 c. \quad (11.2)$$

In 2D rectangular coordinates, Eqs. (11.1) and (11.2) become

$$\begin{aligned} \frac{\partial c}{\partial t} + u \frac{\partial c}{\partial x} + v \frac{\partial c}{\partial y} &= \frac{\partial}{\partial x} \left(D \frac{\partial c}{\partial x} \right) + \frac{\partial}{\partial y} \left(D \frac{\partial c}{\partial y} \right) \\ &= D \left(\frac{\partial^2 c}{\partial x^2} + \frac{\partial^2 c}{\partial y^2} \right) \text{ when } D \text{ is constant.} \end{aligned} \quad (11.3)$$

Heat transfer is similar, but with different parameters. The vector representation is

$$\begin{aligned} \rho C_p \left(\frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T \right) &= \nabla \cdot (k \nabla T) \\ &= k \nabla^2 T \text{ when } k \text{ is constant} \end{aligned} \quad (11.4)$$

In 2D rectangular coordinates, the equation is

$$\begin{aligned} \rho C_p \left(\frac{\partial T}{\partial t} + u \frac{\partial T}{\partial x} + v \frac{\partial T}{\partial y} \right) &= \frac{\partial}{\partial x} \left(k \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left(k \frac{\partial T}{\partial y} \right) \\ &= k \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right) \text{ when } k \text{ is constant} \end{aligned} \quad (11.5)$$

The velocity in the equation comes from the solution to the Navier–Stokes equation described in Chapter 10 (or another flow equation).

NONDIMENSIONAL EQUATIONS

Diffusion in liquids can be slow, and this makes the problems hard to solve, as shown below. The best way to learn how important diffusion is for any problem is to calculate the Peclet number. Equation (11.3) is made nondimensional using the techniques illustrated in Eq. (10.42). Let $c'(x', y', t')$ be the nondimensional concentration, and $c = c_s c'(x', y', t')$. Equation (11.3) for a constant diffusivity becomes:

$$\frac{c_s}{t_s} \frac{\partial c'}{\partial t'} + \frac{u_s c_s}{x_s} \left(u' \frac{\partial c'}{\partial x'} + v' \frac{\partial c'}{\partial y'} \right) = \frac{D c_s}{x_s^2} \left(\frac{\partial^2 c'}{\partial x'^2} + \frac{\partial^2 c'}{\partial y'^2} \right) \quad (11.6)$$

Multiplying by x_s^2/Dc_s gives

$$\frac{x_s^2}{t_s D} \frac{\partial c'}{\partial t'} + \frac{u_s x_s}{x_s} \left(u' \frac{\partial c'}{\partial x'} + v' \frac{\partial c'}{\partial y'} \right) = \left(\frac{\partial^2 c'}{\partial x'^2} + \frac{\partial^2 c'}{\partial y'^2} \right) \quad (11.7)$$

You may already have picked the time standard in the flow problem. If you chose $t_s = x_s/u_s$, the coefficients of the first and second terms are the same, and the coefficient is called the Peclet number:

$$Pe = \frac{u_s x_s}{D} \quad (11.8)$$

The equation is then

$$\begin{aligned} Pe \frac{\partial c'}{\partial t'} + Pe \left(u' \frac{\partial c'}{\partial x'} + v' \frac{\partial c'}{\partial y'} \right) &= \left(\frac{\partial^2 c'}{\partial x'^2} + \frac{\partial^2 c'}{\partial y'^2} \right) \quad \text{or} \\ \frac{\partial c'}{\partial t'} + u' \frac{\partial c'}{\partial x'} + v' \frac{\partial c'}{\partial y'} &= \frac{1}{Pe} \left(\frac{\partial^2 c'}{\partial x'^2} + \frac{\partial^2 c'}{\partial y'^2} \right) \end{aligned} \quad (11.9)$$

BOUNDARY CONDITIONS

When there is diffusion or heat conduction but no convection, typical boundary conditions are:

- (1) *Concentration/temperature specified* (called a Dirichelet condition or boundary condition of the first kind):

$$c = c_0, \quad \text{or} \quad T = T_0 \quad (11.10)$$

- (2) *Flux specified* (called a Neumann condition or boundary condition of the second kind), with specified N_0 or q_0 :

$$-D\mathbf{n} \cdot \nabla c = N_0, \quad \text{or} \quad -k\mathbf{n} \cdot \nabla T = q_0 \quad (11.11)$$

- (3) *Flux defined in terms of a mass transfer coefficient, k_m , with an external, known concentration, c_0 , or a heat transfer coefficient, h , with an external, known temperature, T_0* (called a Robin condition or boundary condition of the third kind):

$$-D\mathbf{n} \cdot \nabla c = k_m(c - c_0), \quad \text{or} \quad -k\mathbf{n} \cdot \nabla T = h(T - T_0) \quad (11.12)$$

The boundary conditions permitted in FEMLAB are slightly more general than this; see Table 11.1 for diffusion and conduction and Table 11.2 for diffusion and conduction with convection.

EXAMPLE: HEAT TRANSFER IN TWO DIMENSIONS

The first problem considered is heat conduction in a square, which is a classical problem:

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0 \quad \text{in } 0 \leq x \leq 1, \quad 0 \leq y \leq 1 \quad (11.27)$$

$$T = 1 \quad \text{on } x = 0$$

$$T = 1 \quad \text{on } y = 0$$

$$T = 0 \quad \text{on } y = 1$$

$$\mathbf{n} \cdot \nabla T \equiv \frac{\partial T}{\partial \mathbf{n}} = 0 \quad \text{on } x = 1 \quad (11.28)$$

TABLE 11.1. Boundary Conditions for Diffusion and Conduction Only

Concentration – can set c_0	$c = c_0$	(11.13)
Flux – can set N_0, k_c, c_b	$-\mathbf{n} \cdot \mathbf{N} = N_0 + k_c(c_b - c)$	(11.14)
Insulation/symmetry – can set nothing	$\mathbf{n} \cdot \mathbf{N} = 0$	(11.15)
Temperature – can set T_0	$T = T_0$	(11.16)
Heat flux – can set $q_0, h, T_{\text{inf}}, \text{Const.}, T_{\text{amb}}$	$\mathbf{n} \cdot k\nabla T = q_0 + h(T_{\text{inf}} - T) + \text{Const. } (T_{\text{amb}}^4 - T^4)$	(11.17)
Thermal insulation – can set nothing	$\mathbf{n} \cdot (k\nabla T) = 0$	(11.18)
Zero temperature – same as Eq. (11.16) with $T_0 = 0$		
Axial symmetry – same as Insulation/symmetry or Thermal insulation		

TABLE 11.2. Boundary Conditions for Diffusion/Conduction and Convection

Concentration – can set c_0	$c = c_0$	(11.19)
Flux – can set N_0	$-\mathbf{n} \cdot \mathbf{N} = N_0, \mathbf{N} = -D\nabla c + c\mathbf{u}$	(11.20)
Insulation/symmetry – can set nothing	$-\mathbf{n} \cdot \mathbf{N} = 0, \mathbf{N} = -D\nabla c + c\mathbf{u}$	(11.21)
Convective flux – can set nothing, but use for outflow boundaries	$-\mathbf{n} \cdot \mathbf{N} = 0, \mathbf{N} = -D\nabla c$	(11.22)
Temperature – can set T_0	$T = T_0$	(11.23)
Heat flux – can set q_0	$-\mathbf{n} \cdot \mathbf{q} = q_0, \mathbf{q} = -k\nabla T + \rho C_p T \mathbf{u}$	(11.24)
Thermal insulation – can set nothing	$-\mathbf{n} \cdot \mathbf{q} = 0, \mathbf{q} = -k\nabla T + \rho C_p T \mathbf{u}$	(11.25)
Convective flux – can set nothing, but use for outflow boundaries	$\mathbf{n} \cdot \mathbf{q} = 0, \mathbf{q} = -k\nabla T$	(11.26)
Axial symmetry – same as Insulation/symmetry or Thermal insulation with $\mathbf{u} \cdot \mathbf{n} = 0$		

Step 1 Open FEMLAB for 2D/Chemical Engineering Module/Energy Balance/Conduction/Steady-state analysis.

Step 2 To create the geometry, click on a rectangular icon and draw a rectangle. Double click on it and set the width and height to 1.0, with the corner at $x = 0, y = 0$. Choose Options/Axis/Grid Settings and set the x -values to -0.4 and 1.6 . Close that window.

Step 3 Click on the triangle, to create the mesh. The mesh is shown in Figure 11.1 and has 988 elements and 2037 degrees of freedom. To save this figure, select File/Export/Image and save the file in an appropriate folder using one of the specified image formats.

Step 4 Next set the parameters in the differential equation. Choose Physics/Subdomain Settings, and select domain 1. A window appears with Eq. (11.29) at the top. For this problem, you change to $k = 1$ but leave the other variables set to zero. Click OK.

$$-\nabla \cdot (k\nabla T) = Q + h_{\text{trans}}(T_{\text{ext}} - T) + C_{\text{trans}}(T_{\text{ambtrans}}^4 - T^4) \quad (11.29)$$

Step 5 Next set the boundary conditions. Choose Physics/Boundary Settings. A window opens with the boundary segments numbered (you can also assign names to them). You set a boundary condition either by selecting one of those numbers (the boundary in the figure will turn red, so you can see which one it is), or by double clicking on the boundary

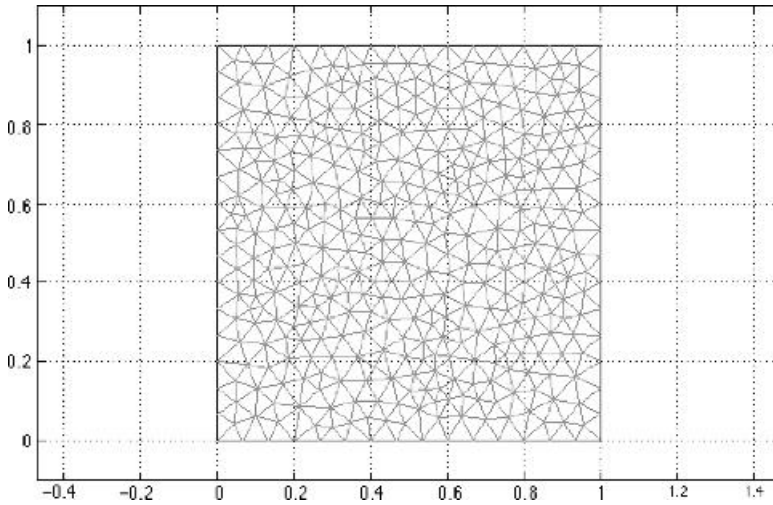


Figure 11.1. Finite element mesh for heat transfer problem.

segment in the figure itself. The boundary conditions here are

$$\begin{aligned}
 \text{boundary1, } x = 0: T &= 1 \\
 \text{boundary2, } y = 0: T &= 1 \\
 \text{boundary3, } y = 1: T &= 0 \\
 \text{boundary4, } x = 1: &\text{Thermal insulation}
 \end{aligned}
 \tag{11.30}$$

Step 6 Click on the ‘=’ sign to solve the problem. A plot appears after it is solved.

Step 7 You can change the plot to other formats. Choose Postprocessing/Plot Parameters. Click on the Surface and Contour boxes. Then click the Surface tab. The new window shows that the surface plot (the colors) will plot the temperature. Click on the Contour tab. The new window shows that the contours will be for temperature. Click OK to obtain Figure 11.2.

Step 8 Notice that the solution goes from -0.056 to 1.056 . There is clearly a numerical error, since the exact solution is bounded between 0 and 1. Try to solve this discrepancy by refining the mesh. Do that by going to mesh mode and choosing Refine Mesh, or click on the four-triangle icon. Solve the problem again using Solve/Restart or the restart icon. Then the algorithm, which is iterative and needs an initial guess, starts from the last solution. This speeds the convergence. The resulting solution still has a minimum of -0.0556 . The reason is that the problem as posed has a discontinuity in it: at $x = 0$ and $y = 1$, the solution is defined as both $T = 0$ and $T = 1$. It cannot be both, and the interpolation does the best it can. Since this is a mathematical anomaly, and is physically unrealistic, you do not need consider it further.

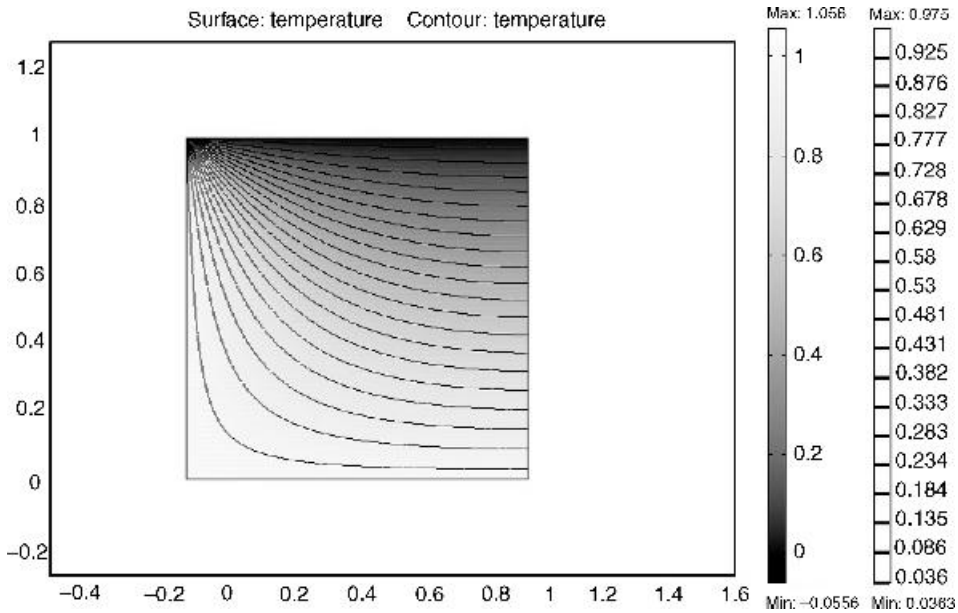


Figure 11.2. Solution to heat transfer problem.

EXAMPLE: HEAT CONDUCTION WITH A HOLE

Next introduce a hole in the middle of the same diagram. The boundary conditions on the sides of the hole are thermal insulation.

Step 1 Select the menu Draw/Specify Objects and select a square. Set the width to 0.4, and the corner to $x = 0.3$ and $y = 0.3$. Then choose Draw/Create Composite Object; set the formula: $R1 - R2$. This causes the domain to be the large one ($R1$) minus the small one ($R2$). You can make very complicated domains by adding and subtracting various objects.

Step 2 Create the mesh, with 824 elements and 1732 degrees of freedom.

Step 3 Set the boundary conditions by selecting the menu Physics/Boundary Settings. The inner boundary is formed from boundary segments 4–7; select them and set Thermal Insulation. The right-hand boundary is now boundary segment 8, and you set it to Thermal Insulation as before.

Step 4 Solve the problem; the plot is shown in Figure 11.3. The settings for the plot are still the same as for Figure 11.2 since you started with that solution. Notice that the contours of constant temperature are perpendicular to the thermally insulated boundaries, which is appropriate because there is no gradient normal to the boundary.

Step 5 If desired, you can choose the Postprocessing/Boundary Integration option and integrate the heat flux over a boundary segment. Select boundary segment 6, type ty for

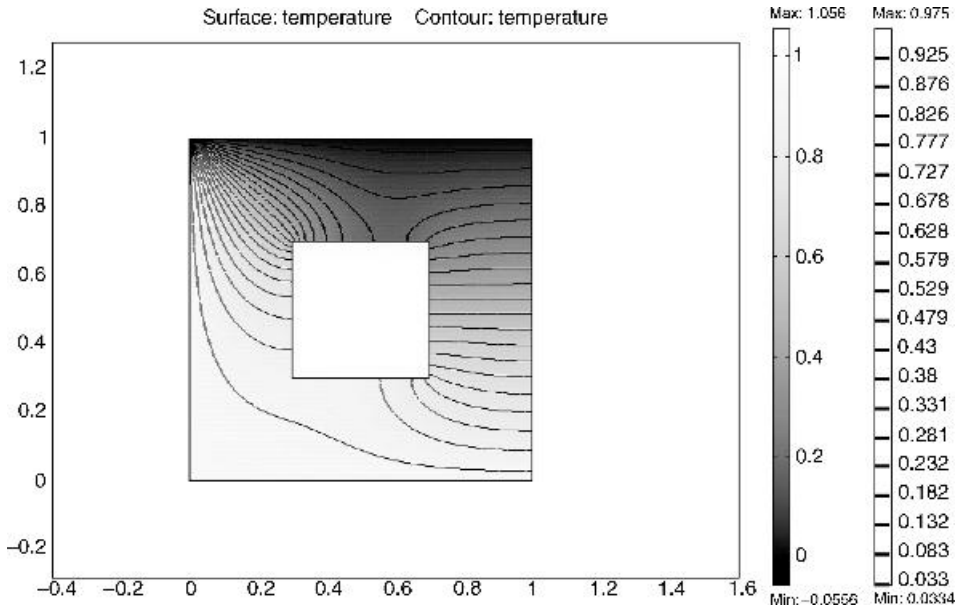


Figure 11.3. Solution to heat transfer problem with a hole in the center.

the expression and click Apply. The integral along the surface $y = 0.7$ from $x = 0.3$ to 0.7 is 0. If you want the area of the domain, choose Postprocessing/Subdomain Integration, insert 1 in the window, select domain 1, and click Apply. The area is 0.84.

EXAMPLE: DISPERSION IN MICROFLUIDIC DEVICES

The T-sensor was described in Chapter 10, but its key use is to transfer a chemical from one flowing stream to the other. Thus, the convective diffusion equation must be solved, too. A fluid (such as water) comes in the top and bottom, but the top stream contains a dissolved chemical that needs to be transferred. The bottom stream may contain a different chemical that will react and fluoresce, thus permitting a visual detection. Your goal is to predict how fast the transfer will take place. See Hatch et al. (2001) and Wiegl et al. (1999).

Step 1 It is easy to add the convective diffusion equation by starting with the fluid flow model of the T-sensor. Choose Multiphysics/Model Navigator. A window appears with a list of possible equations. Scroll down and select: Chemical Engineering Module/Mass Balance/Convection and Diffusion/Steady-state Analysis. Click Add. Now FEMLAB will solve both equations.

Step 2 Note that when you are setting domain parameters or boundary conditions, you do this only one equation at a time, and you have to select the equation you want under the Multiphysics tab. Select the convection and diffusion equation. Then, select Physics/Subdomain Settings, leave $\text{Disotropic} = 1$, and $R = 0$ (this can be a reaction rate depending upon c). In the u and v boxes, put u and v , respectively; otherwise convection will be absent from the problem.

Step 3 Choose Physics/Boundary Settings and set the concentration to 1.0 on the top inlet (boundary segment 3) and 0.0 on the bottom inlet (boundary segment 2). Choose 'Convective flux' for the outlet boundary segment (8). All other boundary conditions are set to Insulation/Symmetry.

Step 4 Solve the problem. The concentration solution is shown in Figure 11.4.

Step 5 You can use the Postprocessing option to check the solution. Choose Boundary Integration and select boundary 2, 3, or 8. To calculate the total flow of chemical in/out of the boundary, integrate v^*c over boundary segment 2, $-v^*c$ for boundary segment 3 (the flow in is in the negative y -direction), and u^*c for boundary segment 8. Then choose Apply and read the value of the integral in the lower, left part of the screen; the results are given in Table 11.3.

Effect of Peclet Number

The previous case had so much diffusion that the concentration was well mixed very quickly. The diffusion coefficient in FEMLAB is actually the Peclet number in Eq. (11.9). Using a value of $D = 1/50$ is really for $Pe = 50$ and gives the results shown in Figure 11.5. To speed the calculation, though, you notice that the flow field should remain the same, since it is unaffected by concentration in this example. To do this, choose Solve/Solver Manager. Choose the Solve For tab and select only the concentration. Then use the re-start icon to solve the problem. This figure shows that the concentration has not mixed as well, because of the lower diffusion coefficient.

Next reduce the diffusion coefficient even more: $D = 1/500$. Now you get the solution shown in Figure 11.6. Notice that there are some 'wiggles' (oscillations) in the

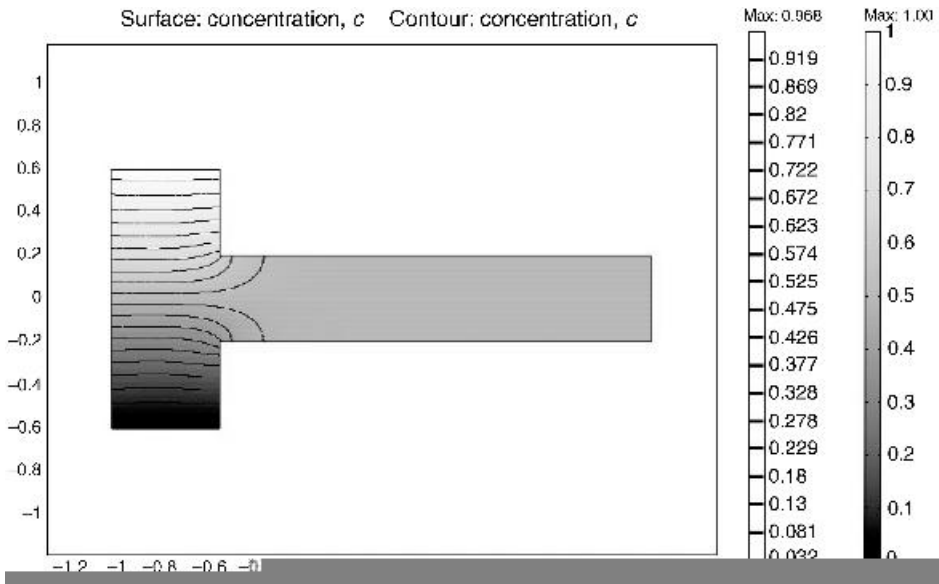


Figure 11.4. Solution to T-sensor diffusion/convection problem with $D = 1$.

TABLE 11.3. Flow Rate of Chemical In/Out of T-Sensor

Segment	2	3	8
Integrate	v^*c	$-v^*c$	u^*c
Value	0	0.5	0.499994

concentration contours; the maximum concentration is 1.067 and minimum concentration is -0.0429 . This indicates that the solution contains errors.

There are two ways to improve the situation. One is to refine the mesh, because the key parameter is the Peclet number times the mesh size. The other is to add Streamwise diffusion in Eq. (11.9). Before doing that, though, examine the solution at the exit. To display the solution at the exit for a Peclet number 50 and 500, choose Postprocessing/Domain Plot Parameters. Then select Line/Extrusion, select boundary 8, plot the concentration and click Apply. Do not close the figure. Solve for another Peclet number, do the same thing, but in the Domain Plot Parameters, General tab, click the ‘Keep current plot’ box. Then the new plot goes on the same figure just prepared. The result is Figure 11.7. Clearly there is better mixing when the Pelect number is 50 than when it is 500, and the unusual oscillations did not appear in the exit concentration. These oscillations are more prominent when there is a steep concentration gradient, which has been smoothed out by the time the flow reaches the exit.

Refine the mesh once and resolve the problem for $Pe = 500$. The mesh now has 3664 elements and 24,478 degrees of freedom. The solution is shown in Figure 11.8, and it is considerably better than Figure 11.6. The peak concentration is 1.002 and the minimum concentration is -0.0024 . These discrepancies are small enough to be acceptable. Thus, you have solved this problem, for Peclet numbers as high as 500.

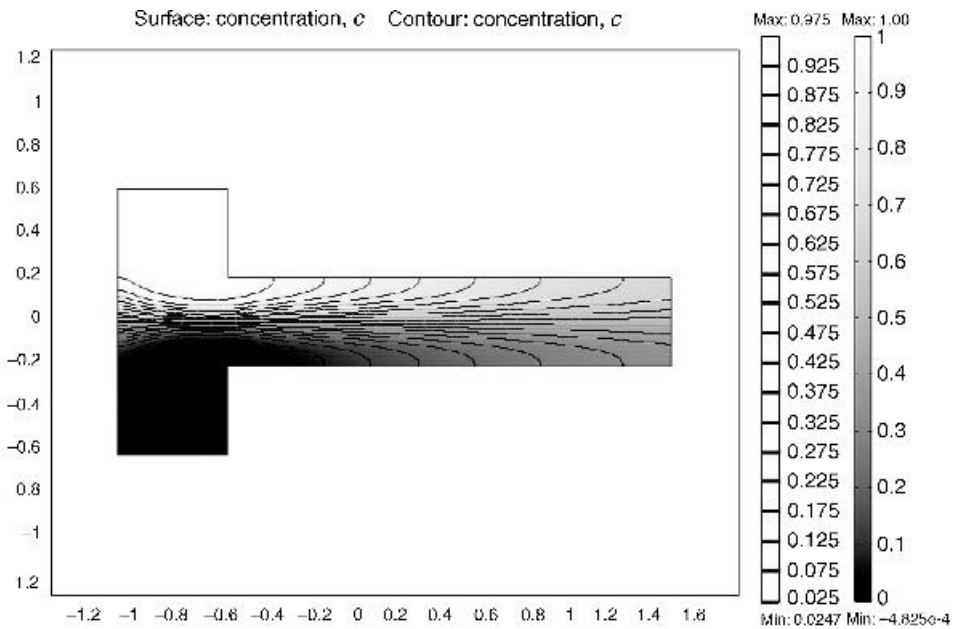


Figure 11.5. Solution to T-sensor diffusion/convection problem with $D = 1/50$.

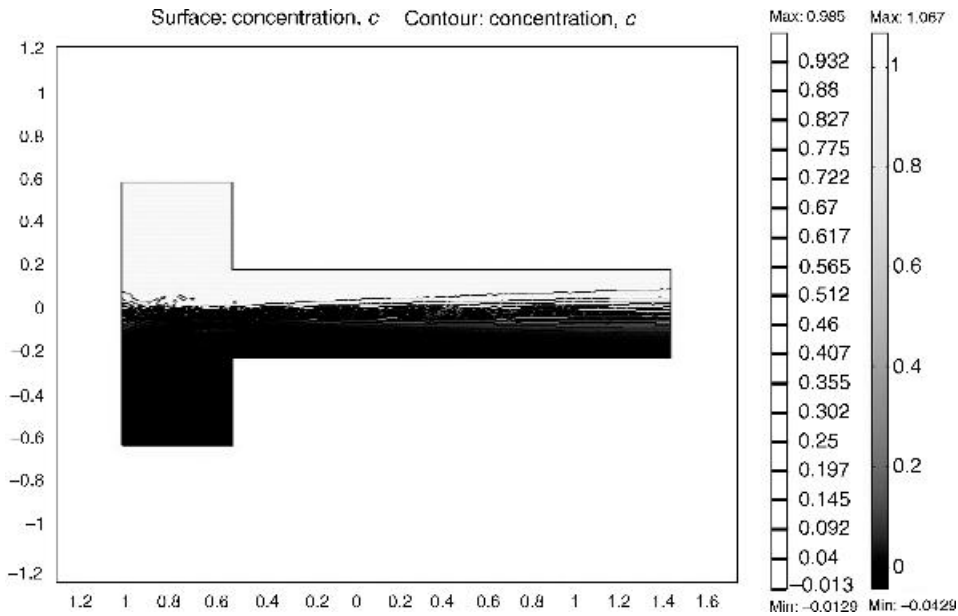


Figure 11.6. Solution to T-sensor diffusion/convection problem with $D = 1/500$.

EXAMPLE: CONCENTRATION-DEPENDENT VISCOSITY

The viscosity of a mixture depends upon its concentration, although often that dependence is small. To resolve the previous problem and allow the viscosity to depend upon concentration you use the formula

$$\mu = 1 + c \quad (11.31)$$

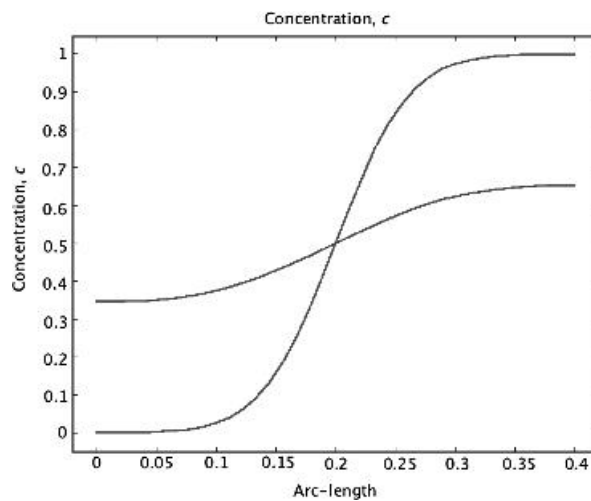


Figure 11.7. Exit concentrations for $Pe = 50$ and 500 ($D = 1/50$ and $1/500$).

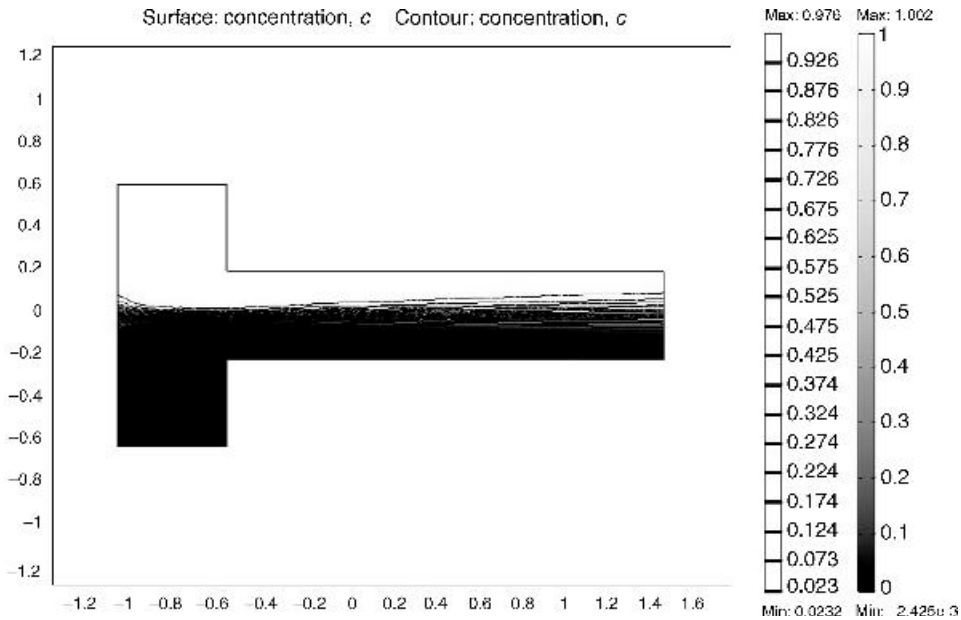


Figure 11.8. Solution to T-sensor diffusion/convection problem with $D = 1/500$ and fine mesh.

Then the viscosity of the material coming into the top of the T-sensor is 2 whereas the viscosity coming into the bottom is 1. This model was solved by Marlina Lukman when she was a senior chemical engineering student at the University of Washington. To modify the model, perform the following steps.

Step 1 Choose the Navier–Stokes equation under the Multiphysics/Model Navigator option. Then choose Physics/Subdomain Settings. Select the domain and type in the formula for viscosity: $1 + c$. If you have several concentration variables, be sure to use the variable names assigned by FEMLAB (or changed by you) in this formula.

Step 2 For this problem, the flow and convection-diffusion problem must be solved together. Choose Solve/Solver manager. Choose the Solve For tab and select both the Navier–Stokes equation and the convective diffusion equation. Use the re-start button to solve. Do this and you will find that the flow field does not change very much, and neither does the concentration profile, as shown in Figure 11.9. While the concentration profile does not change much, now you have quantified the effect.

EXAMPLE: VISCOUS DISSIPATION

So far the examples have assumed that viscous dissipation does not cause an appreciable temperature rise. Viscous dissipation, of course, always increases the temperature, but often the increase is negligible. However, it is important to know the magnitude or provide an upper bound. To see the effect of viscous dissipation, a heat generation

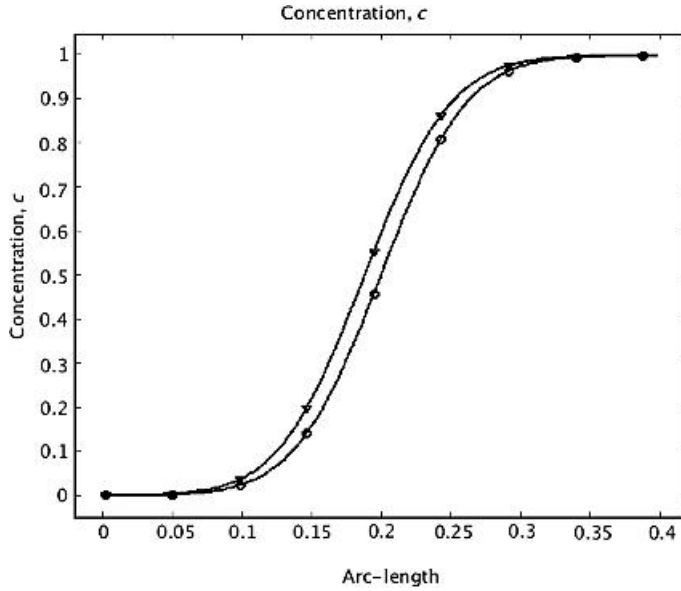


Figure 11.9. T-sensor diffusion/convection problem with $D = 1/500$ and $\mu = 1$ (○) $\mu = 1 + c$ (▽).

term is added to the energy equation. In two-dimensional problems this heat generation term is

$$\Phi_v = \mu \left\{ 2 \left[\left(\frac{\partial u}{\partial x} \right)^2 + \left(\frac{\partial v}{\partial y} \right)^2 \right] + \left[\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \right]^2 \right\}, \text{ Cartesian geometry} \quad (11.32)$$

and

$$\Phi_v = 2\mu \left[\left(\frac{\partial u}{\partial z} \right)^2 + \left(\frac{\partial v}{\partial r} \right)^2 + \left(\frac{v}{r} \right)^2 \right] + \mu \left[\frac{\partial u}{\partial r} + \frac{\partial v}{\partial z} \right]^2, \text{ cylindrical geometry} \quad (11.33)$$

The effect of viscous dissipation is illustrated here for the orifice problem solved in Chapter 10. There are two options for the temperature boundary conditions. In the first option, the system can be adiabatic, where all walls use the boundary condition of no flux. In the second option, the walls can be made from metal, with plenty of coolant outside, and then the boundary condition would use a constant temperature. Here, adiabatic boundary conditions are used. Thus, the temperature out of the device (minus the inlet temperature) gives the temperature rise due to viscous dissipation.

Step 1 Start with the orifice model. To include this term as a generation term in the energy equation, choose Chemical Engineering Module/Energy balance/Convection and Conduction/Steady-state analysis under the Multiphysics/Model Navigator option.

Step 2 Under Physics/Subdomain Settings, insert the following equation for Q , the heat generation term. The notation is changed to use u as the radial velocity and v as the axial velocity.

$$Q = 2^* Br^* (vz^2 + ur^2 + (u/r)^2) + Br^* (vr + uz)^2 \quad (11.34)$$

The Brinkman number is

$$Br = Pr \frac{u_s^2}{C_p T_s}, \quad Pr = \frac{C_p \mu}{k} \quad (11.35)$$

The parameters used in this example are

$$\rho = 1, \quad \eta = 1/Re, \quad Pr = 7, \quad Re = 31.62, \quad C_p = Pr^* Re = 221.3, \quad Br = 0.141 \quad (11.36)$$

Step 3 Under Physics/Boundary Settings, set all boundaries to no flux except for the inlet and outlet boundaries. At the inlet the temperature is specified, and at the exit the option for Convective flux is used.

Step 4 Since the flow field is known, you can solve for temperature by itself. Choose Solve/Solver Manager and choose the Solver For tab; select only the temperature. Solve the problem using the re-start button.

The effect of viscous dissipation is shown in Figure 11.10 when the flow field is that shown in Figures 10.20*b* and 10.21*b*. The average temperature exiting the device is only 0.0074°C higher than the input temperature, and the peak temperature rise is 0.019°C .

It is also possible to make the viscosity depend upon temperature. Before doing that, you can solve the problem with a constant viscosity to determine the maximum

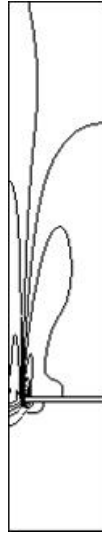


Figure 11.10. Temperature rise (K) in orifice flow, $Re = 10^{1.5} = 31.6$, 20 contour levels from 0 to 0.0182.

temperature rise. If the temperature rise is small, you have verified the assumption that the viscosity is constant. If the temperature rise is large, then the problem can be solved, allowing the viscosity to depend upon temperature. Let the viscosity depend upon temperature, as in Eq. (11.37).

$$\mu = \mu_0 \exp(E/RT) \quad (11.37)$$

Equation (11.38) is inserted for the viscosity for the Physics/Subdomain Settings for the Navier–Stokes equation:

$$\mu = \mu_0 * \exp[E/(R*T)] \quad (11.38)$$

Naturally, you must use the correct units.

EXAMPLE: CHEMICAL REACTOR

It is easy to model the microfluidic device as a chemical reactor. You prepare the flow problem as illustrated in Chapter 10, add the convective diffusion equation, and enter the reaction rate. Suppose the rate of reaction is

$$rate = kc^2 \quad \text{where } rate \text{ is in mol/vol time} \quad (11.39)$$

Step 1 Under Multiphysics, select the Convection and Diffusion option.

Step 2 Under Physics/Subdomain Settings, select the domain and enter the reaction rate in the window for R , the reaction rate. The equation listed at the top is

$$\nabla(-D\nabla c) = R - \mathbf{u} \cdot \nabla c \quad (11.40)$$

so R is the rate of generation. The material is reacting, so the rate of generation is negative. Thus, enter

$$R = -k^*c^*c \quad (11.41)$$

Insert the value for k in Options/Constants.

EXAMPLE: WALL REACTIONS

A wall reaction can be included as well. Suppose rate of reaction on the wall is

$$rate = -k_b c^2 \quad \text{where } rate \text{ is in mol/area time} \quad (11.42)$$

Step 1 Under Multiphysics, select the Convection and Diffusion option.

Step 2 Under Physics/Boundary Settings, select the boundary and choose the Flux boundary condition. The equation is

$$-\mathbf{n} \cdot \mathbf{N} = N_0, \quad \mathbf{N} = -D\nabla c + \mathbf{u}c \quad (11.43)$$

Next you need to augment the flux equation on the wall. The normal velocity on the wall is set to zero for the flow problem. Thus, the diffusive flux equals the rate of reaction. Enter into the window for N_0 the following expression:

$$N_0 = -kb^*c^*c \quad (11.44)$$

Insert the value for kb in Options/Constants.

EXAMPLE: MIXING IN A SERPENTINE MIXER

To illustrate what is possible in 3D, consider the serpentine mixer that was modeled in Chapter 10. The idea is that at the inlet there are two streams entering, and one of them contains a different chemical. The objective is to mix the chemicals in as short a distance as possible. Because mixing by diffusion is slow, chimneys and turns are inserted to enhance the mixing and make it occur in a shorter distance. This was solved by Zachery Tyree when he was a senior chemical engineering student at the University of Washington (Neils et al., 2004).

The concentration on the boundaries is shown in Figure 11.11. This solution is for a Peclet number of 1000, and used 36,320 elements and 267,293 unknowns. This model uses $D = 1$ and $1000*u$, $1000*v$, and $1000*w$ for the three convection terms. The concentration profile at a particular location is shown in Figure 11.12. Notice that there are some small errors, since the maximum concentration is 1.019 and the minimum value is -0.0217 , but these are small enough to be ignored. The region where they occur can be found by plotting only values of concentration greater than 1.0; next plot

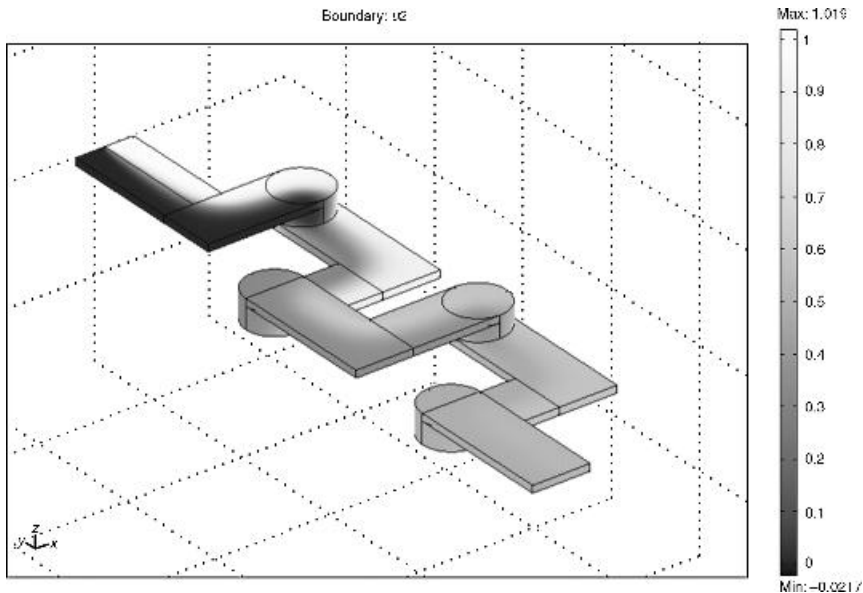


Figure 11.11. Concentration distribution on the wall of a serpentine mixer.

only values of concentration less than 0.0. A plot of the concentration profile inside the domain (Fig. 11.12) is obtained under Postprocessing/Cross-Section Plot Parameters by choosing the Slice tab. Then choose three points that define a plane (here the plot is for $y = -1$, all x and z) and Local coordinate system to plot in.

CHAPTER SUMMARY

This chapter illustrated the use of FEMLAB for problems of heat conduction, heat conduction and convection, and mass diffusion and convection. Problems included heat conduction in a 2D plane, several microfluidic devices (T-sensor and serpentine mixer), and heat effects in orifice flow. Specific methods demonstrated in FEMLAB include:

- Choices of boundary conditions;
- Composite objects;
- Sequential solution of flow, then concentration for efficiency;
- Restart from simpler model to improve convergence and efficiency;
- Plotting in 3D geometries;
- Effect of Peclet number and the necessity for mesh refinement;
- Viscosity depending upon concentration or temperature;
- Viscous dissipation; and
- Chemical reactions in the fluid and on the wall.

These problems are complicated, but they do follow conditions in real-life that engineers must tackle. Knowing how to do this will allow you to work in a variety of modern industries.

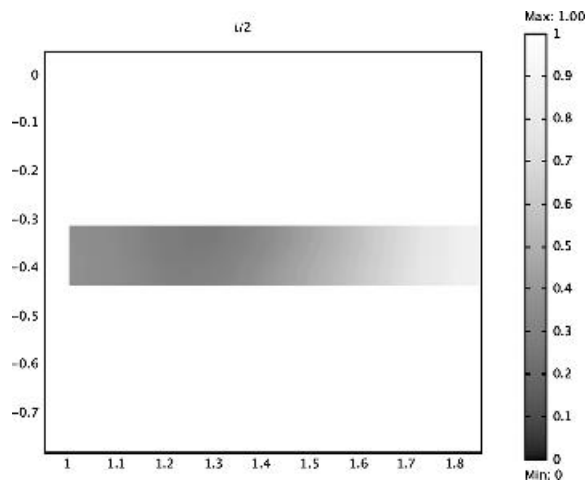


Figure 11.12. Concentration profile at $y = -1$ (after first chimney).

PROBLEMS

Steady, Two-Dimensional Problems

- 11.1.** Heat transfer takes place in the geometry shown in Figure 11.13. Boundaries A are insulated; along boundary B the temperature is 1.0; the boundary condition at C is

$$-k \frac{\partial T}{\partial n} = hT \quad (11.44)$$

- (1) Use $k = 1$, $h = 3$ and solve for the temperature profile. (2) Use $k = 0.01$, $h = 3$ and solve for the temperature profile; compare the results.
- 11.2.** A slotted-electrode electrochemical cell is shown in Figure 11.14 (see Orazem and Newman, 1984). The governing equation for potential is

$$\nabla^2 \phi = 0, \quad \text{or} \quad \frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} = 0 \quad (11.45)$$

Determine the current distribution ($\partial \phi / \partial n$) along the faces AB and EF in Figure 11.15.

- 11.3.** Solve Laplace's equation [Eq. (11.45)] in the region shown in Figure 11.16 with the following boundary conditions:

$$\begin{aligned} u = 0 \quad \text{along} \quad \theta = 0, \quad \frac{\partial u}{\partial n} = \frac{\partial u}{\partial \theta} = 0 \quad \text{along} \quad \theta = \alpha, \\ u = \sin\left(\frac{\pi\theta}{2\alpha}\right) \quad \text{along} \quad r = 1 \end{aligned} \quad (11.46)$$

- (1) $\alpha = \pi/4$; (2) $\alpha = 3\pi/4$; (3) $\alpha = \pi$; (4) $\alpha = 3\pi/2$. Comment on the singularity that occurs at the corner, despite the smooth boundary conditions for u . The exact solution is

$$u = r^{\pi/2\alpha-1} \sin\left(\frac{\pi\theta}{2\alpha}\right) \quad (11.47)$$

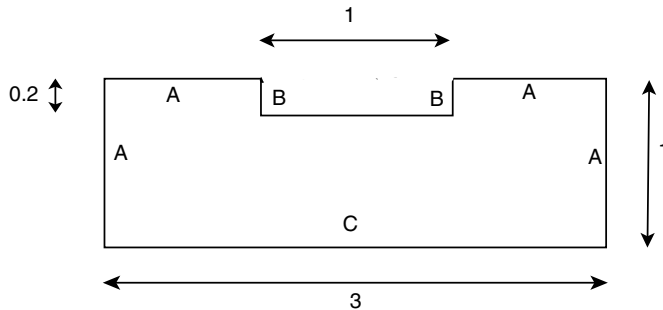


Figure 11.13. Heat transfer in a microchemical reactor.

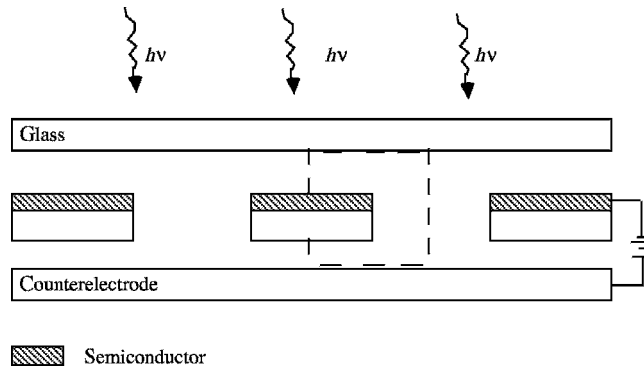


Figure 11.14. Slotted-electrode electrochemical cell.

- 11.4.** Solve Problem 9.2 in a cylindrical pellet whose length equals its diameter. Compare the average reaction rate in the cylindrical pellet with that in a spherical pellet.
- 11.5.** Solve Problem 9.5 in a cylindrical pellet whose length equals its diameter. Compare the average reaction rate in the cylindrical pellet with that in a spherical pellet.

Heat Transfer with Flow

- 11.6.** Solve for the heat transfer from a sphere maintained at temperature 1 when a Newtonian fluid is flowing past it with a temperature of zero. Use constant viscosity and integrate the flux over the surface to find the rate of heat transfer. (1) $Re = 0$ (Stokes solution) and $Pe = 1, 10$, and 100 . (2) Repeat part (1) with $Re = 10$.
- 11.7.** Solve the following Graetz problem using the convection and conduction module in FEMLAB. When there is no axial conduction, you can absorb the Peclet number into the distance, z , and solve the problem once for all Pe . Solve with a radius of 1.0

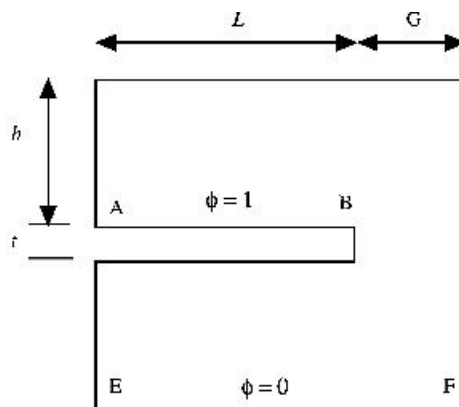


Figure 11.15. Solution domain.

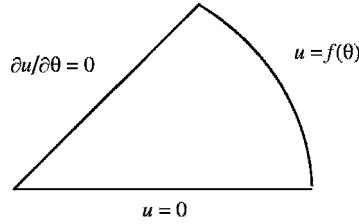


Figure 11.16. Solution domain.

and a length of $z/Pe = 1$; use ‘Convective flux’ as the outlet boundary condition:

$$2(1 - r^2)Pe \frac{\partial T}{\partial z} = 2(1 - r^2) \frac{\partial T}{\partial(z/Pe)} = \frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial T}{\partial r} \right), \quad \frac{\partial T}{\partial r} \Big|_{r=0} = 0, \quad (11.48)$$

$$T(1, z) = 1, \quad T(r, 0) = 0$$

Compute the Nusselt number, which is given by

$$Nu = \frac{-\frac{\partial T}{\partial r} \Big|_{r=1}}{\langle T \rangle - 1}, \quad \langle T \rangle = \text{average temperature}, \quad Pe = \frac{u_{\text{avg}} R}{D} \quad (11.49)$$

- 11.8.** Solve the same Graetz problem as in Problem 11.7, but add in axial conduction. Solve the equations in the form of Eq. (11.50) with a radius of 1.0, and a length of 6:

$$2(1 - r^2) \frac{\partial T}{\partial z} = \frac{1}{Pe} \left[\frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial T}{\partial r} \right) + \frac{\partial^2 T}{\partial z^2} \right], \quad \frac{\partial T}{\partial r} \Big|_{r=0} = 0, \quad (11.50)$$

$$\frac{\partial T}{\partial z} \Big|_{z=L} = 0, \quad T(1, z) = 1, \quad T(r, z=0) = 0$$

- 11.9.** The boundary conditions in Problem 11.8 are not really appropriate since there can be conduction upstream into the unheated region. Solve the problem from $z = -2$ to 6, including the upstream region. Use $Pe = 20$. The equations are

$$2(1 - r^2) \frac{\partial T}{\partial z} = \frac{1}{Pe} \left[\frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial T}{\partial r} \right) + \frac{\partial^2 T}{\partial z^2} \right], \quad \frac{\partial T}{\partial r} \Big|_{r=0} = 0. \quad (11.51)$$

$$\frac{\partial T}{\partial z} \Big|_{z=L} = 0, \quad T(r, z = -2) = 0$$

$$T(1, z) = 0, \quad z \geq 0, \quad T(1, z) = 1, \quad z < 0 \quad (11.52)$$

Compare the answer with the answer in Problem 11.8.

- 11.10.** Solve the Graetz problem (Problem 11.7) but use a boundary condition of $T = 0$ at the outlet. Solve with $Pe = 30$ on a short domain, $L = 6$; solve on a long

domain, $L = 18$. This problem illustrates the importance of having the correct boundary conditions (see Chang and Finlayson, 1980).

Reaction with Flow

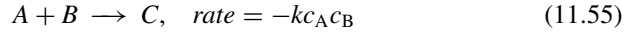
- 11.11.** Convection, diffusion, and reaction in a packed bed reactor are governed by the following equations in a cylindrical domain with length L and radius R :

$$\begin{aligned} \frac{\partial c}{\partial t} + u \frac{\partial c}{\partial z} &= \frac{D}{r} \frac{\partial}{\partial t} \left(r \frac{\partial c}{\partial r} \right) - kc \\ c(r, z, 0) &= c_0, \quad c(r, 0, t) = c_1, \quad \left. \frac{\partial c}{\partial r} \right|_{r=0} = 0, \quad -D \left. \frac{\partial c}{\partial r} \right|_{r=R} = k_m(c|_{r=R} - c_2) \end{aligned} \quad (11.53)$$

Solve with the following parameters:

$$\begin{aligned} c_0 = 0, \quad c_1 = 1, \quad c_2 = 0, \quad u = 10, \quad D = 0.1, \quad L = 10, \\ k = 10, \quad k_m = 1, \quad R = 0.1 \end{aligned} \quad (11.54)$$

- 11.12.** Solve the flow and diffusion problem for the T-sensor, as illustrated in the example. Then consider a similar flow problem with water coming in both inlets. However, one inlet contains one chemical, identified as A, and the other inlet contains another chemical, identified as B. The two react with the reaction rate



To solve this problem, you need to use the Navier–Stokes equation and two convection and diffusion equations. Use the same parameters as in the example, plus $k = 3$, $c_{A\text{in}} = 1$, $c_{B\text{in}} = 1$. Compare solutions with $Pe = 1, 10$, and 100 .

Reaction with No Flow or Known Flow

- 11.13.** Equation (11.56) shows the dimensionless equations for a nonisothermal packed bed reactor in a cylindrical domain:

$$\begin{aligned} \frac{\partial c}{\partial z} &= \alpha \frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial c}{\partial r} \right) + \beta R(c, T), \quad \frac{\partial T}{\partial z} = \alpha' \frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial T}{\partial r} \right) + \beta' R(c, T) \\ \left. \frac{\partial c}{\partial r} \right|_{r=0} &= \left. \frac{\partial T}{\partial r} \right|_{r=0} = 0, \quad \left. \frac{\partial c}{\partial r} \right|_{r=1} = 0, \quad - \left. \frac{\partial T}{\partial r} \right|_{r=1} = Bi_w [T(1, z) - T_w] \\ c(r, 0) &= 1, \quad T(r, 0) = 1 \end{aligned} \quad (11.56)$$

Solve them for the parameters

$$\begin{aligned} R(c, T) &= (1 - c)e^{\gamma - \gamma/T}, \quad \alpha = \alpha' = 1, \quad \beta = 0.3, \\ \beta' &= 0.2, \quad \gamma = 20, \quad T_w = 0.92 \end{aligned} \quad (11.57)$$

and (1) $Bi_w = 1$; (2) $Bi_w = 20$. The dimensionless variables are defined as

$$\begin{aligned}\alpha &= \frac{Ld_p}{R^2Pe_m}, & \alpha' &= \frac{Ld_p}{R^2Pe_h}, & Pe_m &= \frac{Gd_p}{\rho D_e}, & Pe_h &= \frac{C_p}{Gd_p k_e} \\ \beta &= \frac{k_0 L \rho}{G}, & \beta' &= \frac{(-\Delta H_{rxn})k_0 c_0 L}{C_p G T_0}, & Bi_w &= \frac{h_w R}{k_e}\end{aligned}\quad (11.58)$$

where d_p is the particle diameter, G is the mass flux, D_e is the effective diffusivity, k_e is the effective thermal conductivity, and h_w is the heat transfer coefficient at the cylindrical wall. Comment on the radial distribution of temperature and concentration in the two cases. Based on this limited experience, give your recommendations about when the temperature and concentration will vary significantly in the radial direction.

- 11.14.** Equation (11.59) governs the transient reaction and diffusion with a Michaelis–Menten reaction in a porous media. Integrate the equations to steady state using the parameters $\phi^2 = 10$, $v = 2$, $Bi_m = 20$:

$$\begin{aligned}\frac{\partial c}{\partial t} &= \frac{1}{r^2} \frac{\partial}{\partial r} \left(r^2 \frac{\partial c}{\partial r} \right) - \phi^2 R(c), & R(c) &= \frac{c}{v + c} \\ \left. \frac{\partial c}{\partial r} \right|_{r=0} &= 0, & \left. \frac{\partial c}{\partial r} \right|_{r=1} &= Bi_m [c(1, t) - 1], & c(r, 0) &= 0\end{aligned}\quad (11.59)$$

- 11.15.** Equation (11.60) governs the reaction of carbon monoxide in a thin porous material with platinum as the catalyst:

$$\begin{aligned}\frac{\partial c}{\partial t} &= \frac{1}{r^2} \frac{\partial}{\partial r} \left(r^2 \frac{\partial c}{\partial r} \right) - \phi^2 R(c, T), & \frac{\partial T}{\partial t} &= \frac{1}{r^2} \frac{\partial}{\partial r} \left(r^2 \frac{\partial T}{\partial r} \right) + \beta \phi^2 R(c, T) \\ \left. \frac{\partial c}{\partial r} \right|_{r=0} &= \left. \frac{\partial T}{\partial r} \right|_{r=0} = 0, & \left. \frac{\partial c}{\partial r} \right|_{r=1} &= Bi_m [c(1, t) - c_w], & - \left. \frac{\partial T}{\partial r} \right|_{r=1} &= Bi_h [T(1, t) - T_w] \\ c(r, 0) &= 0, & T(r, 0) &= 1, & R(c, T) &= \frac{c}{(1 + Kc)^2} \exp \left[\gamma \left(1 - \frac{1}{T} \right) \right]\end{aligned}\quad (11.60)$$

The parameters are

$$\begin{aligned}\phi^2 &= 625, & \beta &= 0.02, & K &= 20, & \gamma &= 30, & Bi_m &= 200, \\ Bi_h &= 5, & c_w &= 1, & T_w &= 1\end{aligned}\quad (11.61)$$

What is the maximum temperature rise in the catalyst? (Note: If the standard temperature is 600 K, used in the definition of the dimensionless activation energy, then the actual temperature is 600 times the dimensionless temperature.) Where does the maximum reaction rate occur?

Appendix A

HINTS WHEN USING EXCEL[®]

This appendix provides hints when using Microsoft Excel.* Appendix A includes general features that are useful in all the applications solved with Excel in this book. Other features are illustrated in the context of specific examples, and a list of them is provided at the end of Appendix A for handy reference. You may want to skim this appendix first, then start working some of the problems that use Excel before you come back and review this appendix in more detail. That way you will not be burdened with details that do not make sense to you until you see where and how you need them. For a more detailed description of Excel, consult other books devoted to Excel, such as Larsen (2002).

CELL ORGANIZATION

The rectangular array of cells is labeled A, B, C, D... across the top, and 1, 2, 3, 4, ... down the left-hand side (Fig. A.1). Cell B3 is in the second column (B column) and the third row from the top. (As in linear algebra with matrices, columns go down and rows go across.)

CELL CONTENTS

You can enter a number, word, or formula into the cell. To enter a number, click on the cell and type the number. Pressing return will move down one cell, pressing tab will move across one cell, to be ready for the next entry. To enter a word, select the cell and type the word. The word can also be preceded by a single quote, which is useful if the

*Excel is a registered trademark of Microsoft Corporation, Inc.

temperature 25			
	A	B	C
1	temperature	25	degrees
2			
3	formula	1 - 0.01*temperature	
4			
5	density	0.75	

Figure A.1. Cell contents: naming cell, formulas, text.

‘word’ is actually a number. To enter a formula, select the cell, enter ‘=’ followed by the formula. The cell can be referenced by its location, for example B1, but it can also be named. To name it, select the cell and enter the name of the variable in the upper left-hand corner of the spreadsheet and then press return or enter. Then to use the value in B1, you can use the variable name ‘temperature’ rather than B1, as in Figure A.1. The entries in this figure are:

- B1 is named temperature (A.1)
- B3 is: ‘1 – 0.01*temperature’ (A.2)
- B5 is: = 1 – 0.01*temperature (A.3)

FORMAT

Under the Format menu bar are several options that make your spreadsheet more readable. With the Format/Cells . . . you can change the way numbers are displayed, the alignment of them within the cell, and the font properties. You can also set a border around some cells, and change the pattern or color (the Color option is inside the Pattern option). You can even protect the spreadsheet or parts of it so others cannot change it.

With the Format/Row option, you can change the height of one or more rows, and you can hide a row. To make it visible again, choose the rows on both sides of the line, and choose ‘unhide.’ The same options work for columns, and the width of the columns can be changed. You can do this for all rows/columns, for only rows/columns you select, or you can use the scroll bar to change a single row or column by dragging the boundary. The Format/Conditional Formatting option allows you to cause the cell to change color, for example, if a parameter meets certain conditions. This can be used as a red flag (literally!) to highlight mistaken results or mistaken input and provides a form of data validation.

COMMENTS

You can insert comment boxes on your spreadsheet by selecting Insert/Comment. Type the desired comment and click Return. You can also insert a text box of any desired size. Open the drawing menu (🎨), select the textbox symbol (📄) in the drawing tools, and use the mouse to draw a rectangle. You can then type inside the rectangle.

PICTURES, EQUATIONS, WEB LINKS

You can insert pictures, equations, and hyperlinks (to web sites), too. Choose Insert/Picture and choose the picture you want. To enter an equation, Choose Insert/Object and choose Microsoft Equation. You may have to add that feature to your spreadsheet if it was not included during installation. (You need the original CD or ask your System Administrator to install it.) For hyperlinks, simply choose Insert/Hyperlinks and insert the web address. Be sure to try it while connected to the internet to ensure that it works.

SELECT COLUMNS FOR CHARTS, REGRESSION, AND PRINTING

Put the cursor in the upper-left cell that you would like to select and drag down until the shaded box covers all the columns and rows you want. Note that the columns and rows have to be contiguous, or next to each other. You can do this to two or more contiguous columns or rows. Use this same procedure for selecting the print area.

COPY FORMULAS ACROSS AND DOWN THE SPREADSHEET

To copy a formula across the spreadsheet, select the cell and put the cursor on the lower right-hand corner of the cell. Drag the cursor to the right. The column entries of formulae are incremented one by one. Suppose the original formula used B2. If the formula is dragged to the right, B2 is changed to C2. If the formula is dragged down, the B2 is changed to B3. If you do not want that to happen, that is you want the formula in all columns to use B2, then use absolute addresses. In the formula, change B2 to \$B\$2. Options like \$B2 and B\$2 are also possible. If you select a cell tag in the equation and press F4, it turns into an absolute address. On the Apple Macintosh,* selecting the cell tag and pressing command-T does the same thing; if you continue to press it, the tag cycles between B2, \$B\$2, B\$2, \$B2, B2, ...

INSERT ROWS AND COLUMNS

Select a row on the left-hand side and choose Insert/Row. A new row is placed above the one you selected. The cells are renumbered, and all the formulas are changed, too, to reflect the new numbering system. Use this same method to insert a column; the new column will appear to the left of the one you have selected.

SPLIT/FREEZING

You can split the screen into four portions in a 2×2 array. To do this, select the cell that will be in the center, that is, the upper left corner of the lower right rectangle. Then choose Windows/Split. You can later choose Windows/Remove Split if you want. In Figure A.2, the cursor was placed in cell F3 before choosing Windows/Split. Sometimes the

*Registered trademark of Apple Computer, Inc.

	A	B	C	D	E	F	G	H	I	J	K
1	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
2	0.2	0.3	0.5	0.8	1.2	1.7	2.3	3	3.8	4.7	5.7
3	0.4	0.7	1.2	2	3.2	4.9	7.2	10.2	14	18.7	24.4
4	0.6	1.3	2.5	4.5	7.7	12.6	19.8	30	44	62.7	87.1
5	0.8	2.1	4.6	9.1	16.8	29.4	49.2	79.2	123.2	185.9	273
6	1	3.1	7.7	16.8	33.6	63	112.2	191.4	314.6	500.5	773.5

Figure A.2. Split window.

spreadsheet is very large, but the parts you want to see are at the beginning and end. Figure A.3 shows a numerical integration of a differential equation, starting from $t = 0$ (see Appendix F). You are interested in the results at $t = 1$, but want the heading, too. Place the cursor on the 3 designating row3 and choose Window/Freeze Panes. Then rows 1 and 2 remain visible while you scroll down to row 102, as shown in Figure A.4.

ITERATION ON, TOLERANCE

You can use Goal Seek or Solver to find the roots of a function. You may find that the answer is not as accurate as you would like. (The default parameters are set for business applications, where tenths of a cent do not matter.) To get a more accurate answer, you need to make the tolerance smaller. Choose Tools/Options and click Calculation. On the Macintosh, choose Preferences and click on Calculation. Figure A.5 shows the default values. Change the maximum iterations to 1000 and the maximum change to 0.000001 and close, then run Goal Seek. You can make the same choices in Solver by choosing Options. This is also the place to turn off the Iteration function by unchecking the iteration box. This is sometimes useful when solving a problem in which the formulas are intertwined and the formula in one cell depends on the answer in another cell, which depends on the value in the first cell. In these cases it is convenient to turn off iteration while preparing the formulas in the spreadsheet, and then turn it on once the entire spreadsheet is complete.

PASTE

Sometimes you want to reproduce part of your spreadsheet in another spreadsheet or in another area of your existing one. You can select the area of the spreadsheet, choose

	A	B	C	D	E
1	t	y	dy/dt	N	dt
2	0	1	-1	100	0.01
3	0.01	0.99	-0.99		
4	0.02	0.9801	-0.9801		
5	0.03	0.970299	-0.9703		
6	0.04	0.960596	-0.9606		
7	0.05	0.95099	-0.95099		
8	0.06	0.94148	-0.94148		
9	0.07	0.932065	-0.93207		
10	0.08	0.922745	-0.92274		

Figure A.3. Window/Freeze Panes, start.

	A	B	C	D	E
1	t	y	dy/dt	N	dt
2	0	1	-1	100	0.01
92	0.9	0.404732	-0.40473		
93	0.91	0.400685	-0.40068		
94	0.92	0.396678	-0.39668		
95	0.93	0.392711	-0.39271		
96	0.94	0.388784	-0.38878		
97	0.95	0.384896	-0.3849		
98	0.96	0.381047	-0.38105		
99	0.97	0.377237	-0.37724		
100	0.98	0.373464	-0.37346		
101	0.99	0.36973	-0.36973		
102	1	0.366032	-0.36603		

Figure A.4. Window/Freeze Panes, after scrolling down.

Copy, move to the upper, left-most cell of the new area and choose Paste. Be careful because this will copy all formulas, too. If you want only the values from a spreadsheet, copy the part you want and choose Paste Special; then choose the desired option (such as Value). You can also paste into Word. If you simply use Paste, you can manipulate the spreadsheet inside Word (resize it, change the font, etc.). If you use Paste Special, one option is to paste the spreadsheet as a picture. In that case the whole image is frozen before it is put into Word.

PLOT – xy SCATTER, EDIT, MULTIPLE CURVES, SURFACE PLOTS

Plotting in Excel is easy. Choose the cells you wish to plot, such as columns A and B in Figure A.6. If you want, choose the headings, too, and they will appear in the graph

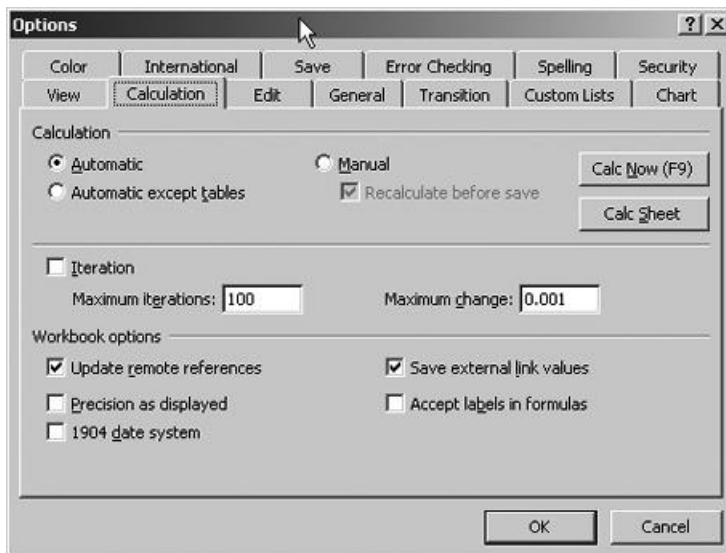


Figure A.5. Iteration and calculation window.

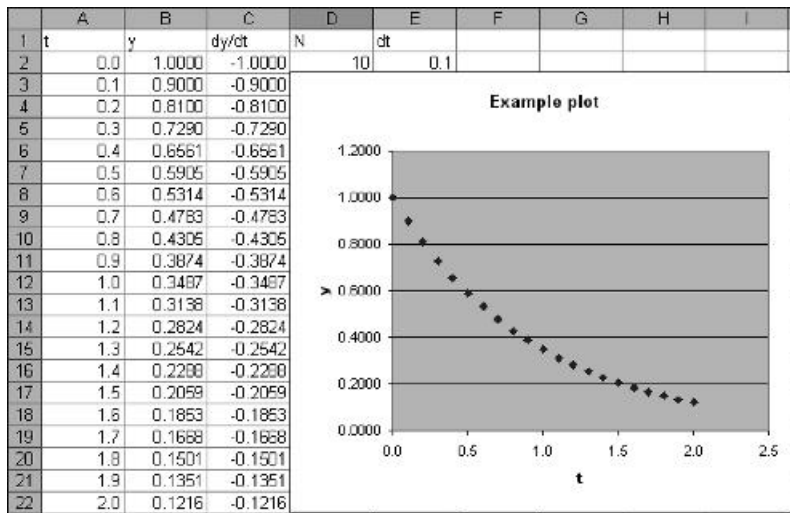


Figure A.6. Plot of a single line.

(although they can later be changed). Then choose Insert/Chart, choose *xy Scatter*, and click on the graph style you want. The program takes you through a series of steps in which you can enter a title, label the *x* and *y* axes, and so on. Finally, tell the program to place the chart on the graph. To remove the meaningless label 'Series 1,' select it and press Delete. To edit any part of the graph put the cursor on the part and right-click (on the Macintosh use Apple-click). A screen will appear that allows you to make other choices for the graph. To put multiple lines on the graph, select more columns. For example, if you select columns A, B, and C, you can produce Figure A.7.

To make a contour plot or 3D view, create the data to be plotted in a rectangular part of the spreadsheet. In Figure A.8, the A column and seventh row are the *y* and *x* values, respectively, and the other values are the ones to be plotted. Select B2–F6, choose

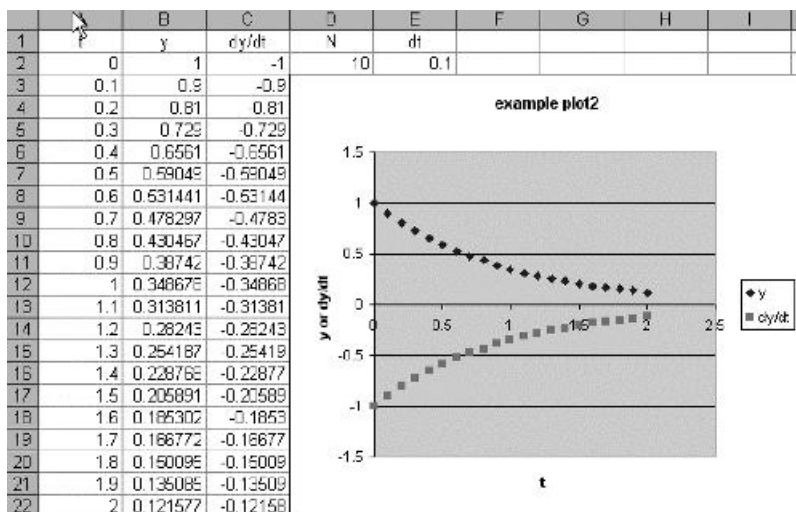


Figure A.7. Plot of two lines.

	A	B	C	D	E	F
1	y					
2	15	0.667	1.333	2.000	2.667	3.333
3	12	0.833	1.667	2.500	3.333	4.167
4	9	1.111	2.222	3.333	4.444	5.556
5	6	1.667	3.333	5.000	6.667	8.333
6	3	3.333	6.667	10.000	13.333	16.667
7	x	10	20	30	40	50

Figure A.8. Data for contour plot.

Insert/Chart, and choose Surface. In this case, choose the contour plot picture (you could also choose the 3D view). Follow the Plot Wizard. In step 2, choose the Series tab, and arrange it as shown in the left-hand side of Figure A.9. This allows you to put x and y values along the axes. The rectangular domain will be plotted as a function of the cell numbers or entry numbers, not any values you assigned to each row or column (see Figure A.9). Also, the plot is flipped (note the location in the spreadsheet where the maximum occurs and the location in the plot where it occurs, too.) If you wish to have more contours, after the plot is made, right-click on the legend and set new contour levels. There is no point in having a different increment between the x points or the y points, because the graph is going to be made using constant increments. This is a serious limitation of Excel for scientific plotting. You can make the final result look like Figure A.9.

ARRANGE SPREADSHEET

To make a professional-looking spreadsheet, it is important to provide documentation for the reader. You can establish different sections, surrounded by boxes (see Figure A.10):

Title, your name and date

Data Input – these are values that the user may change

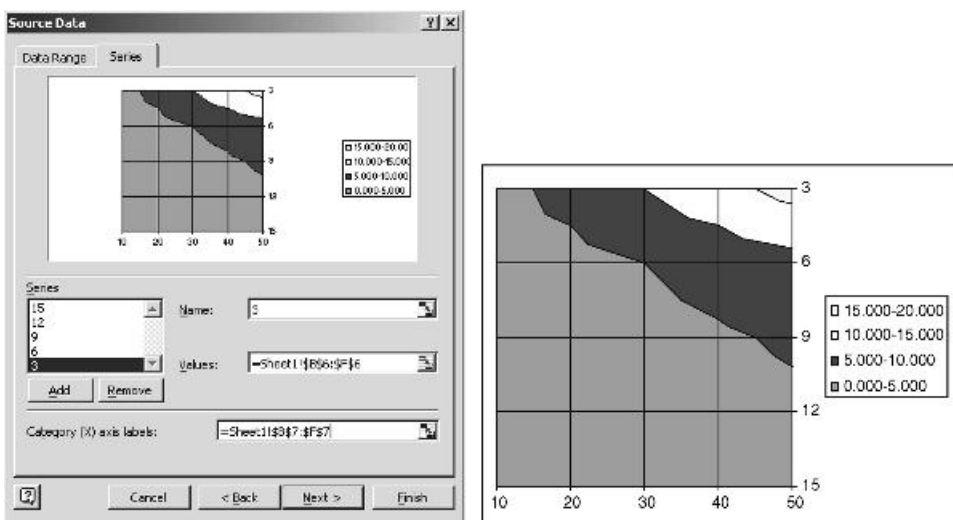


Figure A.9. Contour plot.

	A	B	C	D	E	F
1	Calculate density as a function of temperature					
2		Bruce Finlayson				
3		January, 2005				
4						
5		Data Input				
6	temperature	25 degrees C				
7						
8		Formulas used				
9	formula	1-0.01*temperature				
10						
11		Results				
12	density	0.75				
13						

Figure A.10. Spreadsheet arranged for easy understanding.

Parameter Values – these are values that the user will not change, but are needed (not shown)

Formulas used – this defines the problem

Results

To make the boxes, select the cells to be enclosed, choose Format/Cells/Border, choose Outline, and click OK. In the example above, the equation in cell B12 is $=1 - 0.01*\text{temperature}$, which gives the same result as $=1 - 0.01*B6$.

IMPORT AND EXPORT TEXT FILES, ONE COLUMN AT A TIME

There are several ways to do this. The first method works when you have a single column to transfer. To export from Excel, select the column of cells, choose Copy, open an application that accepts text files, and choose Paste. To get a single column of numbers into Excel, copy the column in another application, in Excel select one cell, and choose Paste.

IMPORT AND EXPORT TEXT FILES, MULTIPLE COLUMNS

To export from Excel, select the columns you wish to export, Copy, change to another application file and Paste. To do the reverse operation, copy the text data, open Excel, put the cursor in one cell, and choose Paste. If the original text file had tabs in it (is tab delimited), each entry will go into a different cell. If you do not prepare the text file appropriately, you may get a mess (the whole file put into one cell or one column. If you create several columns in Word using tabs, you can copy the columns and paste them into the spreadsheet and the entries will go into successive cells.

EXPORT A TEXT FILE

You can export the contents of the spreadsheet to a text file that can be used as the input to another program. This is a nice way to have a user interface for a program that requires

significant computing capability (beyond what Excel offers). You could put the physical parameters for the case you are studying into the Excel spreadsheet, allowing them to be changed easily. You then export the information and have the other computer program read the text file to get the data. To export the information (the contents of the cells, not the figures from the spreadsheet), choose SaveAs, change the format to a text format, such as Text(Windows), and save it. It will have .txt appended to the name. If you open that file, 'name.txt,' in a word processor or text editor, you will see the information displayed in exactly the same format as in the spreadsheet. It is not simple to get these numbers into other programs, which have to read the spaces appropriately, and some knowledge of the C computer language may be necessary, but it does make a nice interface for a program that you will use many times. MATLAB[®] can read such files, for example.

TOOLS

The power of the Excel program lies in its tools. Goal Seek and Solver were discussed Chapters 2–4, and Data Analysis is discussed in Appendix E. Other useful tools include 'Tools/Protection' (to prevent part or all of the spreadsheet from being changed) and 'Tools/Flag for Follow Up' (to notify you on some future date when you may need to update the spreadsheet).

FUNCTIONS

Excel comes equipped with many engineering functions. Choose Insert/Functions. Then choose Math & Trig or Engineering to see the functions. A window appears that explains how to use the functions.

MATRICES

Two of the functions under Math & Trig are MINVERSE and MMULT. These functions have the obvious use. The inverse function, of course, must act on a square matrix, and creates a square matrix. To use it, first create the matrix, as illustrated in cells A1:C3 of Figure A.11.

	A	B	C
1	10	5	2
2	1	11	4
3	3	7	12
4			
5			
6	0.105263	-0.04656	-0.002024291
7	-1.3E-18	0.115385	-0.038461538
8	-0.02632	-0.05567	0.106275304
9			
10			
11	1	0	0
12	-1.4E-17	1	0
13	0	0	1

Figure A.11. Matrix inversion and multiplication.

Then in cell A6 type `=MINVERSE(A1:C3)`. You can also insert the A1:C3 by selecting the cells. Click on cell A6, press the shift key, and select the other corner of the matrix, C8. Press F2, the Ctrl-Shift-Enter. The inverse appears in A6:C8. The matrix multiplication is illustrated in Figure A.11 by multiplying these two matrices together; the result should be the identity matrix. Click on cell A11, type `=MMULT(A1:C3,A6:C8)`. Click on cell A11, press the shift key, and select the other corner of the matrix, C13. Press F2, the Ctrl-Shift-Enter. The matrix multiplication appears in A11:C13. Indeed it is the identity matrix.

EXCEL HELP

When you are truly stuck, try the Help/Search Excel Help. You can type in a question and it will alert you to several possible answers related to your question. This usually leads you to the answer you want.

APPLICATIONS OF EXCEL

Listed here are applications of Excel using specific tools so you can find them easily.

- Equations of state, Chapter 2, p. 8.
- Vapor–liquid equilibrium, Chapter 3, p. 28.
- Chemical reaction equilibrium, Chapter 4, p. 44.
- Process simulation using Excel, simple mass balance, Chapter 5, p. 58, 60, 62.
- Process simulation using Excel, including chemical reaction equilibrium, Chapter 5, p. 63.
- Process simulation using Excel, including phase equilibrium, Chapter 5, p. 63.
- Continuous stirred-tank reactor, Chapter 8, p. 135.
- Straight line curve fit using Excel, Appendix E, p. 294.
- Polynomial regression using Excel, Appendix E, p. 297.
- Multiple regression using Excel, Appendix E, p. 298.
- Nonlinear regression using Excel, Appendix E, p. 304.
- Finite difference method for boundary value problems, Appendix F, p. 317.
- Finite difference method for elliptic problems, Appendix F, p. 321.

Appendix B

HINTS WHEN USING MATLAB[®]

This appendix provides hints and tips when using MATLAB.* It assumes that you are a beginner in using MATLAB, but not an absolute beginner in computer programming. Most likely, you remember concepts from a computer programming class taken earlier. Included in Appendix B are general features that are useful in all the applications solved with MATLAB. Other features are illustrated in the context of specific examples; a list of examples is provided at the end of the appendix for handy reference. You will probably want to skim this appendix first, then start working some of the problems that use MATLAB to gather experience, and finally come back and review this appendix in more detail. That way you will not be burdened with details that do not make sense to you before you see where and how you need them. An excellent book with more details is by Higham and Higham (2005).

GENERAL FEATURES

Start the Program

To start, open MATLAB.

Option 1 Use the set path command to point to an area where you store your programs and data. Add this path to your path directory. When MATLAB looks for a m-file (a program you write), it will look in all these directories, from top to bottom until it finds one with the same name. Then, it will use that m-file.

*MATLAB is a registered trademark of The Math Works, Inc.

Option 2 Another alternative is to set the current directory to the folder where your m-files are stored. Figure B.1 shows the default window.

Screen Format

The Current Directory is identified at the top (see Figure B.1); the right-hand window is the Command Window, and the left-hand windows display the contents of the Current Directory and the Command History. By choosing the Workspace tab in the left-hand window, you will find a list of the variables, descriptions of their format such as whether they are global or not. In the Command History will be commands you have issued before, for ready reference. The Command Window is where you issue commands and where output appears. By choosing various options under Desktop you can hide or display these windows. You can recover commands you have used earlier by using the up-arrow; they appear one by one. This feature makes it easy during the debugging phase to repeat commands without retyping them. The down-arrow works in reverse.

Stop/Closing the Program

When you stop work, save your m-files in a folder, so you can use them next time. You can also save the workspace, which will save all the numerical values that have been calculated. This is convenient if you have to interrupt your work for a short time and do not want to start over. Of course, most of your work is contained in your m-files, which you always want to save.

If a simulation continues for a long time and you want to halt it, press Control+Break or Control+C on the PC or Control+C on the Macintosh. This will stop the program as a forced quit.

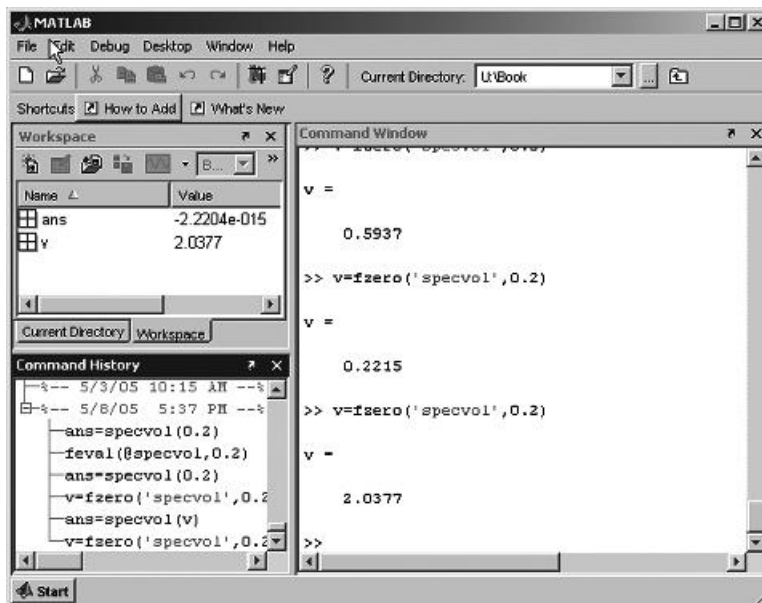


Figure B.1. MATLAB screen.

m-Files

In MATLAB you write computer programs that are called m-files and are saved on your computer. The files can be used at any time by simply typing their name in the command line, and one m-file can use another m-file. You will want to include comments in your m-file that explain what the file does. This is done by inserting a ‘%’; everything after ‘%’ on a line is considered a comment. These comments remind you of details when you come back to the program at a later time.

Workspaces and Transfer of Information

When you use MATLAB, your results will only be as good as the programs you write. Thus, it is important to understand the structure of workspaces and m-files, as well as how to display intermediate results, transfer parameters from one m-file to another, and debug your code. Most of the problems encountered by beginning users involve errors in these areas, and only careful checking will help you find them.

The command workspace is an area of computer memory that is reserved for parameters you define from the command line in MATLAB. Each m-file has its own unique workspace, and this workspace does not communicate with the command workspace unless you make it happen. It is sort of like two of your friends having a disagreement: they choose not to talk until a third party (you) defines a way to do so. There are two ways to set up the communication: using the global command or calling arguments of the m-files.

‘Global’ Command

Figure B.2 shows a command window and two m-files. The global commands in each one determine which variables are available in the m-files. Assume you have typed the command

```
global a b c
```

in the command line. Then give a value to *a*, *b*, and *c*:

```
a=2
b=3
c=4
```

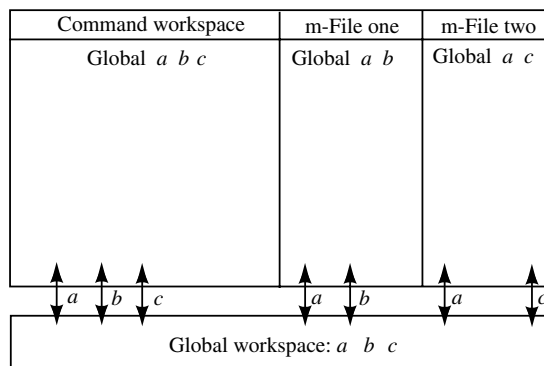


Figure B.2. Global and local variables.

The question is then: How can you get these values of a , b , and c into an m-file? You do this by putting a global command in the m-file; the command lists the variables that you want available. In Figure B.2, the parameters $a = 2$ and $b = 3$ are available in 'm-file one' and $a = 2$ and $c = 4$ are available in 'm-file two'. If a is changed in 'm-file one,' it is changed for all the programs. In other words, the change is made in the central, global location and thus the exchange goes two ways. If the variable c is used in 'm-file one,' it must be defined there since it is a local variable in 'm-file one.' Whatever its value, it does not change the global value of c because c is not listed in the global command of 'm-file one.' Changing c in the second m-file would change it in the command workspace, too.

The second way to pass parameters is to use list the parameters in the function statement of the m-file. For example, the m-file

```
function y=name(x,parameters)
```

expects values of x and *parameters* to be supplied when the m-file is called. The variable *parameters* can be a vector, that is, containing several elements.

Display Tools

When you are checking your programming, you will want to examine the value of various parameters. The ';' represses the display of the result of the calculation. Remove the ';', and you will see the result of each line of your code in the Command Window as it is executed. It is essential that you look at this to verify your code.

You can also cause a variable to be displayed using one of the following three commands:

```
disp(x) gives:           x=2.34
or just say x, which gives: x=2.34
disp('now is the time') gives: now is the time
```

You can issue these commands from the command window to see the value of any variable in the command workspace. If you want to know the value of a local variable in an m-file you must put it in the m-file and execute the m-file. If you need more significant digits, try

```
format long: 1.234567890e+00
format long e: 1.234567890e+00
format short: 123.4568
format short e: 1.2346e+02
```

To eliminate all variables

```
clear all
```

This is helpful when debugging because it eliminates variables that may have been set incorrectly in previous calculations. MATLAB then starts from scratch. MATLAB may be saving a number without you knowing, which makes it hard to find your mistake.

Finding MATLAB Errors

An important lesson you can learn about using MATLAB is how to find errors. One common error message from MATLAB is:

```
???Undefined function or variable 'q'.
```

This means that q is either an undefined function or an undefined variable. If q is a function or m-file, then MATLAB cannot find it. Check the working directory or use set path and make sure that the path is set to the folder where you stored the m-file and that you have added that path to the list. Do not forget to save your pathlist so that it is there the next time you use MATLAB. If q is a variable, it has not been set or was not transferred correctly.

One source of confusion is when you think you have defined a variable, but have not really. Use `disp(x)` to see the current value of x in the command workspace. If the variable is supposed to be in an m-file, run the m-file with a command added to display the variable, and that will give you the value of x in the m-file workspace. That should give you a hint about possible sources of the error. The most common cause of this mistake (and one that is hard to find unless you check the code systematically) is forgetting to pass a parameter to the m-file using either 'global' or as a parameter. The reason it is hard to find these mistakes is that in your mind you have set the value of the parameter in the program somewhere but failed to transfer it to the appropriate m-file.

Another typical error message is:

```
??? Index exceeds matrix dimensions
```

This message means that a matrix like $A(i, j)$ was called with either i or j too big (i.e., beyond the definition of A), or a function uses a matrix that has not been defined yet.

Debug the Program; That is, Find Your Errors

The programs illustrated in this book are fairly simple, and they are easy to debug. All programmers make mistakes; good programmers learn to find them! Finding them is easy if you take the time. First consider a line in a code in the command window. Set all the parameters used in the line and execute the line. Compare the answer with a calculation done with a calculator or one you did by hand. Be sure that all the parameters you use are different, and do not use zero or one, because some mistakes will not be picked up if the variable is zero or one. The parameters you use for testing can be single digit numbers that are easy to calculate in your head, and they can be completely unrelated to the problem you are solving. You can also copy a line from an m-file and paste it into the command window to check it, too.

If the code you wish to check is in an m-file, the first check is to see that all the parameters are passed correctly and have been set in the workspace available to the m-file. The easiest way to do this is to request that the m-file print all the variables you expect to be transferred to the m-file, at the start of the m-file, and check that they are correct. If a variable has not been set, MATLAB will give you an error message without running the code. It tells you the line and column where the undefined variable is. The problem is probably improper passing of parameters or incorrect use of the global command or, sometimes, a typing error.

MATLAB provides several commands to help you debug the program. Hold the cursor over a variable name in a program and MATLAB will show the value of that variable. This works for m-files, too, but only for variables that enter through the global command. That is a good way to check that the variables were entered correctly, though. The command 'whos' will list the variables and their type, as well as whether they are local or global variables. This is also a command that is useful inside an m-file (when checking), since it lists all the variables that the m-file can access, and if the one needed is not there you can then search for the reason.

You can make the program stop at a certain point and return to the keyboard mode. Then you can access values of the parameters. To stop the program, issue the command 'dbstop at linenumber in filename.' 'Filename' is the name of the m-file, and 'linenumber' is the line number in the m-file where you wish to stop. Alternatively, put the command 'keyboard' into the m-file where you wish to stop. The command cursor changes to K >> and you can issue commands. For example, typing the name of a variable will cause its value to be printed. This is a convenient way to access the value of variables in an m-file. If you then want to access variables in the program that called the m-file, say 'dbup.' To continue computing, type 'return' and press the return key. To stop the calculation altogether, type 'dbquit.'

Input/Output

You can ask the user for input with the following command.

```
viscosity=input('What is the viscosity (Pa s)?')
```

You can display in a specified format

```
fprintf('%5.3f %10.5f\n',x,y)
2.345 234.56789
```

These are C-commands, and the notation means:

```
f=floating point
\n is a carriage return
e=exponential format
5.3 means five characters, three after the decimal point.
```

To write a text file, you must open it, write it, and close it. The following command writes a text file that has two columns, x and y , with as many entries as defined by the vectors x and y (which have the same length).

```
fid=fopen('ChEData','w') % w for write
fprintf(fid,'%5.3f %10.5f\n',[x,y])
fclose(fid)
```

To read a text file, you must open it, read it, and close it.

```
fid=fopen('ChEData','r') % r for read
[x y]=fscanf (fid,'%5.3f %10.5f')
fclose(fid)
```

Loops

It is sometimes useful to execute a command over and over, putting each result in one element of a vector. The format is:

```
for i=1:10
    y(i)=...
end
```

If you want to stop before the end

```
for i=1:10
    y(i)=...
    if y(i)>25 break,end
end
```

Conditional Statements

Sometimes the program needs to execute different instructions depending upon a calculated result. Here are some examples.

```
if (condition1) % examples of condition1: i==3,
    i<3 (note double equal sign)
    ...
else
    ...
end
```

Another option:

```
if (condition1)
    ...
elseif (condition2)
    ...
else
    ...
end
```

Still another option:

```
switch lower(num)
    case {1} % execute when num=1
        y=...
    case {2, 3} % execute when num=2 or 3
        y=...
    otherwise % execute when num is neither 1, 2 or 3
        y=...
        display('there is an error') % this is optional if num
            should be 1, 2, or 3
end
```

Timing Information

Turn on clock at one place in the program – tic. Turn off clock at another place in the program – toc. The elapsed time is displayed.

```
elapsed_time=2.34.
```

Matrices

Vectors are row (1×3)

```
x=[2 3 4]
```

or column (3×1)

```
x=[2
   3
   4]
```

The transpose of x is x' .

Set matrix values

```
A=[1 2 3 4 carriage return
   5 6 7 8]
```

Size(A) gives the dimensions of A:

```
size(A)=2 4
```

Operate on all elements at once. If $T(:)$ is a vector of T (K) you might want it in $^{\circ}\text{C}$:

```
t=T-273
```

MATLAB will do this for all elements in $T(:)$.

Operate on one element of a matrix

```
t(i)=T(i)-273
```

Note: MATLAB is case sensitive so that $t(i)$ and $T(i)$ are different. Obtain rows from matrix:

```
a=A(1,:) gives a=[1 2 3 4]
```

You can use a or $A(1,:)$ in calculations or plotting.

Matrix Multiplication

Take the matrix

$$A = \begin{bmatrix} 2 & 4 & 6 \\ 8 & 10 & 12 \\ 14 & 16 & 18 \end{bmatrix}$$

and the vector

$$x' = \begin{bmatrix} 2 \\ 3 \\ 4 \end{bmatrix}$$

The 3×3 matrix A is formed by:

```
A=[2 4 6
8 10 12
14 16 18]
```

Matrix multiplication is achieved with the asterisk:

$$\begin{bmatrix} 2 & 4 & 6 \\ 8 & 10 & 12 \\ 14 & 16 & 18 \end{bmatrix} \begin{bmatrix} 2 \\ 3 \\ 4 \end{bmatrix} = \begin{bmatrix} 40 \\ 94 \\ 148 \end{bmatrix} \quad (\text{B.1})$$

```
A * x'
ans = [40
      94
      148]
```

is $[3 \times 3][3 \times 1] = [3 \times 1]$ matrix. Note the prime on x ; x' is a column vector, which is needed for matrix multiplication. If you had used $A*x$, MATLAB would have told you that the inner matrix elements have to agree. Multiply terms element by element:

$$u(i) = x^2(i), \quad i = 1, 3 \quad (\text{B.2})$$

```
u=x.* x=[4 9 16]
```

Note the period before the asterisk. The multiplication $\sum_{i=1}^3 x^2(i)$ is made by:

```
x * x'=29
```

This is a single number because the matrix multiplication is $[1 \times 3][3 \times 1] = [1 \times 1]$

```
x' * x=[4 6 8
        6 9 12
        8 12 16]
```

is $[3 \times 1][1 \times 3] = [3 \times 3]$

$x * x$ is meaningless.

```
[1 x 3] [1 x 3]=?
```

Element-by-Element Calculations

If you want to calculate something for every element in the vector, you need to precede the $+$, $-$, $*$, $/$, $^$ command with a period. For example, if you have a series of x and y values,

```
x=0:0.1:1; y= 1:0.1:2
```

the command

```
z=x./y
```

computes $z(i) = x(i)/y(i)$ for each i .

EIGENVALUES OF A MATRIX

Eigenvalues are the polynomial roots to Eq. (B.3):

$$|A_{ij} - \lambda \delta_{ij}| = 0 \quad (\text{B.3})$$

They are easy to calculate in MATLAB using the eig command.

```
>> lambda=eig(A)
lambda=32.2337
       -2.2337
       -0.0000
```

EVALUATE AN INTEGRAL

To evaluate the integral, Eq. (B.4), define a set of points in the interval to be integrated, as the vector x (need not be uniformly spaced). Then evaluate the function at each of these points. Calculate the integral using the trapezoid rule.

$$\text{area} = \int_0^2 x^2 dx \quad (\text{B.4})$$

```
x=[0:0.1:2]
y=x.*x
area=trapz(x,y)
```

SOLVE ALGEBRAIC EQUATIONS USING ‘fsolve’

Solve $f_i(\{y_j\}) = 0$, $i = 1, \dots, n$ for the vector $\{y_j\}$. There are three steps.

Step 1 Set the initial guess of the solution.

Step 2 Name and construct the m-file ‘prob2.m,’ which evaluates the function f_i .

Step 3 Call `fsolve`. `fsolve` is in the optimization toolbox in MATLAB, which all versions do not have.

```
y0=[0 0]
z=fsolve('prob2',y0)
z=-1.4456-2.4122i
```

You need to make several checks. The number of unknowns is set by the number of elements in the initial guess, y_0 . The m-file ‘prob2’ calculates f_i , $i = 1, \dots, n$, given $\{y_j\}$, and there have to be as many elements in f as there are in y . The m-file needs to be checked, of course. The only way you can make MATLAB find the solution to your problem is to make sure the m-file gives the correct set of f s when given a set of y s. If this `fsolve` does not work well, try making up an initial value problem and solving it using implicit methods, integrating to a long time.

$$\frac{dy_i}{dt} = f_i(\{y_j\}), \quad y_i(0) = \text{initial guess} \quad (\text{B.5})$$

SOLVE ALGEBRAIC EQUATIONS USING ‘fzero’ OR ‘fminsearch’ (BOTH IN STANDARD MATLAB)

If you have a single function of one variable, you can use the `fzero` function in place of `fsolve`, and it is called in a similar way. If you have several f s that you want to make zero, you can make the norm of the vector f zero. The norm can be the square root of the sum of the squares of the elements, or the maximum of the absolute value of all elements. Then `fminsearch` uses optimization methods to find the vector y which makes the norm a minimum. This may work for simple problems, especially ones with constraints, but it is less successful on large problems.

INTEGRATING ORDINARY DIFFERENTIAL EQUATIONS THAT ARE INITIAL VALUE PROBLEMS

To solve a single ordinary differential equation

$$\frac{dy}{dt} = f(t, y), \quad y(0) = y_0 \quad (\text{B.6})$$

We use the ‘ode’ set of functions in MATLAB. There are four steps.

Step 1 Set the initial condition, y_0 .

Step 2 Set the time span for the integration.

Step 3 Name and construct the m-file `rhs.m`, which evaluates the function $f(t, y)$.

Step 4 Call ode45.

```
y0=3
tspan=[0 10]
[t, y]=ode45(@rhs, tspan, y0) or
[t, y]=ode45(@rhs, [0 10], 3)
```

The m-file is.

```
function ydot=rhs(t,y)
ydot=...
```

Since MATLAB will integrate whatever equation you give it, and the m-file ‘rhs’ is going to be used many times, you must insure that it is correct: Given a t and y it computes the correct $f(t, y)$ and puts the value into the variable $ydot$, which is then used by MATLAB to construct the solution. Once you have checked the program and run it, it is easy to plot the solution.

```
plot(t,y)
```

If you want data output at specific times, use

```
tspan=[0:1:10] for data at t=0, 1, 2, ..., 10.
```

If the integration is taking an interminable amount of time, it is possible that you have defined a stiff system of equations (see Appendix F). In that case, change to an implicit method (for stiff equations, hence the s-designation):

```
[t, y]=ode15s(@rhs, tspan, y0) or ode23s(@rhs, tspan, y0)
```

If you want the solution to be plotted as it is computed, leave off the $[t, y]$:

```
ode45(@rhs, tspan, y0)
```

To solve a set of ordinary differential equations,

$$\frac{dy_i}{dt} = f_i(\{y_j\}), \quad y_i(0) = y_{i0} \quad (\text{B.7})$$

We do the same steps except that the initial condition is now a vector with N elements, the m-file ‘rhs’ must compute N functions using the vector y with N elements, and we have to return the result as a column vector.

```
y0=[0 1 3]
tspan=[0 10]
[t, y]=ode45(@rhs, tspan, y0)
```

The m-file is

```
function ydot=rhs(t,y)
% now there are y(1), y(2) and y(3) and t, which can be used
% to evaluate fi.
ydot(1)=...
ydot(2)=...
ydot(3)=...
ydot=ydot'
```

Now the output from 'ode45' is a vector, t , and a matrix, y . The vector y has as many columns as there are unknowns (here called N), and it has as many rows as there are elements in the vector t . Once the solution is found, all components of it can be plotted. See other commands under the plotting commands for other options.

```
plot(t,y)
```

If you wish to solve the problem more accurately than is done with the default parameters, you can change them. The following commands show how to change the relative tolerance and absolute tolerance from the default values of 10^{-3} to 10^{-9} . You can place different accuracy limits on different variables, too. To see all the options, see help odeset.

```
OPTIONS=odeset('RelTol',1e-9,'AbsTol',1e-9)
[t, y]=ode45(@rhs, tspan, y0, OPTIONS)
```

Checklist for Using 'ode45' and Other Integration Packages

When using 'ode45,' your m-file for the right-hand side must meet these conditions:

- The name in the calling command must be the same as the filename, and the name in the function is immaterial. If 'ode45' identifies @rhs, then the filename of the m-file has to be 'rhs.m'. The m-file can be: function ydot = nothing(t,y).
- The variable *tspan* (or whatever it is called) must have at least two values.
- The number of entries in the vector for the initial conditions must be the same as the number of right-hand sides calculated in the m-file.
- Whether you call the output from the m-file 'ydot' or something else is your choice, but if the output is called 'ydot', the m-file must compute all elements of the vector *ydot*.
- The output of the m-file must be a column vector.
- Variables can be used in the m-file provided they are (a) global variables, (b) passed as parameters in the calling argument, or (c) set in the m-file. Check their value inside the m-file by running the m-file.
- The m-file will be called many times by the 'ode45' function. However, you only have to check the calculation once. Check the m-file by giving it t and all the $y(i)$; compute what you expect the right-hand sides to be and see that the computer gives those values. This is the only way to ensure that MATLAB is solving the problem you want solved.

SPLINE INTERPOLATION

Make a cubic B-spline pass through all the data points: $[x(i), y(i), i = 1, \dots, m]$. A cubic spline is a cubic function of position, defined on small regions between data points. It is constructed so the function and its first and second derivatives are continuous from one region to another. It usually makes a nice smooth curve through the points. The following commands create Figure B.3.

```
x=0:1:10; y=cos(x); % evaluate cos at a set of x values
xx=0:.25:10;        % set up a finer mesh for plotting
yy=spline(x,y,xx);   % evaluate the spline function at
                     % the points xx
plot(x,y, '*',xx,yy) % plot the original points with *
                     % and the spline curve
```

INTERPOLATE DATA, EVALUATE THE POLYNOMIAL, AND PLOT THE RESULT

For data points $x(i), z(i), i = 1, \dots, m$ fit a polynomial of degree n in the least squares sense:

$$y = p_1x^n + p_2x^{n-1} + \dots + p_nx + p_{n+1} \quad (\text{B.8})$$

```
p=polyfit(x,z,n)
```

The command ' $v = \text{polyval}(p, w)$ ' evaluates the polynomial p at the w points (w can be a vector). Plot the data points (y) and the polynomial (v)

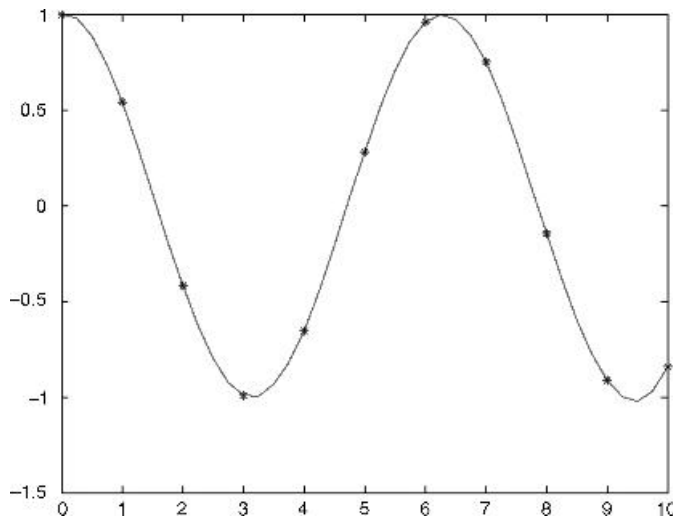


Figure B.3. Spline fit of a set of data points.

```
w=[0:.1:10]
v=polyval(p,w)
plot(x,y,'ro',w,v,'b-')
```

PLOTTING

Plotting Results from Integration of Partial Differential Equations Using the Method of Lines

The result is the value of the solution at several fixed positions and all times:

```
tspan=[0 10], y0=[...]
[t, y]=ode45('rhs',tspan,y0)
plot(t,y)
```

This gives a plot of each variable $y(t, i)$ vs time. Here, $y(t, i)$ is the solution at point $x(i)$ and time t . To plot the solution vs $x(i)$, at various times, specify that you want the solution at a select number of times:

```
tspan=[0:1:10] % 11 different times
y0=[...]
[t, y]=ode45('rhs',tspan,y0) % compute the solution
                                % y(j,i) is a matrix: the column number is i,
                                % and the row number j
                                % identifies the time, t(j)
x=[...] % specify the x-positions of the i-th
                                % variable
                                % there are the same number of entries
                                % for x as for y0

hold on
for i=1:11
    plot(x(:),y(i,:))
end
hold off
```

Simple Plots

For a vector x with n entries, and a vector y with n entries, plot them in one of four ways:

1. `plot(x,y)` 2. `loglog(x,y)` 3. `semilogx(x,y)` 4. `semilogy(x,y)`

To plot more than one variable:

```
plot(x, y1, x, y2) % if the y1 and y2 are known at the
                    same values of x
plot(x1,y1,x2,y2) % if y1 is known at x1 and y2 is
                    known at x2
```

Add titles:

```
title('This is the title')
xlabel('x')
ylabel('y')
```

Add a legend:

```
legend('first curve', 'second curve')
```

Plot only one column of a matrix, here the second column.

```
plot(t,y(:,2))
```

Add Data to an Existing Plot

Issue the command:

```
hold on
```

and continue plotting. Further lines, symbols, and so on will be added to the existing figure. When done, type:

```
hold off
```

Dress Up Your Plot

Plot in different colors using:

```
plot(x, y, 'r', x, y2, 'b')
```

Use different markers:

```
plot(x, y, 'or', x, y2, '*b')
```

Use different line options:

```
plot(x, y, '-r', x, y2, ':b')
```

To get both the symbols and the lines, use both line styles and markers.

```
plot(x, y, '-or', x, y2, ':*b')
```

Limit the plot to $x_{\min} \leq x \leq x_{\max}$, $y_{\min} \leq y \leq y_{\max}$, regardless of the data:

```
axis([xmin xmax ymin ymax])
```

Limit the plot to $x_{\min} \leq x \leq x_{\max}$, regardless of the data:

```
xlim([xmin xmax])
```

Limit the plot to $y_{\min} \leq y \leq y_{\max}$, regardless of the data:

```
ylim([ymin ymax])
```

Get for more information:

```
help axis
```

Multiple Plots

To get multiple plots on the same screen, use subplot. You may wish to have several plots on one screen. Figure B.4 shows six plots on one screen.

```
subplot(2,3,1)
```

makes a 2×3 array of plots on one page and the plotting commands following this apply to the first plot (upper left-hand corner). Follow this with.

```
subplot(2,3,2)
```

and the next plot will be in the middle of the top row of three figures. The final number is the plot sequence, numbered from left to right and then top to bottom.

Sometimes you want to prepare several plots, examine them in turn, and decide whether to save them or print them. One way to do this is to insert the following command after the plot command:

```
pause
```

This causes the program to stop at that point in the code, and you can export the figure or print it. Then press any key to continue.

Three-Dimensional Plots

To plot a function $z(x,y)$, create an x - y grid (rectangular), evaluate the function at each grid point, and plot. Create the x grid:

```
x=0:0.05:1
```

Create the y grid:

```
y= 1:0.2:3
```

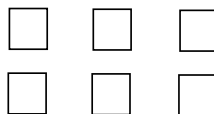


Figure B.4. Multiple plots.

Create the combined mesh:

```
[X,Y]= meshgrid(x,y)
```

Evaluate the function: $Z = fn(x, y)$ where fn is an expression or function; for $z = x^2 + y^2$,

```
Z=X.*X+Y.*Y
```

Plot the 3D plot:

```
mesh(X,Y,Z)
```

or color in the surface:

```
surf(X,Y,Z)
```

or do contours in 2D:

```
contour(X,Y,Z,20)
```

or do contours in 3D:

```
contour3(X,Y,Z,20)
```

More Complicated Plots

Possible symbols are:

.	point	v	triangle (down)
o	circle	^	triangle (up)
x	x-mark	<	triangle (left)
+	plus	>	triangle (right)
*	star	p	pentagram
s	square	h	hexagram
d	diamond		

The colors are in the default order for multiple plots, but white is not used.

b	blue	<i>Line types</i>
g	green	— solid
r	red	: dotted
c	cyan	—· dashdot
m	magenta	— dashed
y	yellow	
k	black	
w	white	

Use Greek Letters and Symbols in the Text

These are T_EX commands

α	<code>\alpha</code>	∞	<code>\infty</code>	<code>\rm</code>	normal.
β	<code>\beta</code>	$\geq$	<code>\geq</code>	<code>\bf</code>	bold
γ	<code>\gamma</code>	$\leq$	<code>\leq</code>	<code>\it</code>	<i>italic</i>
$\cdot$	<code>\cdot</code>	∂	<code>\partial</code>		
ω	<code>\omega</code>	$\pm$	<code>\pm</code>		
Γ	<code>\Gamma</code>		<code>\Re</code>		
Δ	<code>\Delta</code>		<code>\Im</code>		
Θ	<code>\Theta</code>				
$\cdot$	<code>\cdot</code>				
$\cdot$	<code>\cdot</code>				
Ω	<code>\Omega</code>				

MATLAB HELP

When you are really stuck, type the command.

```
help commandname
```

This may lead you to the solution you are looking for. Issuing the command

```
help
```

gives you a table of contents; look at ‘help funfun,’ for example, to see all the methods and options for integrating differential equations.

APPLICATIONS OF MATLAB

Listed here are applications of MATLAB so you can easily find examples in this book:

- Equations of state, Chapter 2, p. 12.
- Equations of state and plotting, Chapter 2, p. 13.
- Vapor–liquid equilibria, Chapter 3, p. 30.
- Chemical reaction equilibria, Chapter 4, p. 45.
- Multiple equations, few unknowns, Chapter 4, p. 48.
- Integrate ordinary differential equation, Chapter 8, p. 114.
- Use of ‘global’ command, Chapter 8, p. 116.
- Isothermal plug flow reactor, Chapter 8, p. 118
- Nonisothermal plug flow reactor, Chapter 8, p. 121.

- Reactor with mole changes and variable density, Chapter 8, p. 130.
- Chemical reactors with mass transfer limitations, Chapter 8, p. 131.
- Continuous stirred-tank reactors, Chapter 8, pp. 135, 136.
- Transient continuous stirred tank reactors Chapter 8, p. 137.
- Fit straight line to data, Appendix E, p. 295.
- Polynomial regression, Appendix E, p. 298.
- Nonlinear regression, Appendix E, p. 305.

Appendix C

HINTS WHEN USING ASPEN PLUS[®]

This appendix gives a few hints and tips on using Aspen Plus.* It assumes you are a beginner, so it shows the main steps you go through to use Aspen Plus, with screen shots to illustrate what you should see on the screen. You should work through this appendix using Aspen Plus to reproduce the examples shown here and review the examples again when you use Aspen Plus in other chapters. Keep in mind that this is a complex program with many options developed over decades by engineers working for Aspen Technology as well as the companies using the program. Some engineers' whole job is doing simulations to model chemical process, and it is unrealistic to expect a beginner to do that. However, you can use Aspen Plus to do many powerful things, and it is important to know the full capabilities available to you.

QUICK TOUR

Start

Begin by opening Aspen Plus. A registration screen (connect to engine) comes up; click OK. Click 'maximize' to fill the screen. This step may differ depending upon your installation. For a new problem, it is useful to choose a template that is similar to your problem; you can also open previous problems. The screen that appears is shown in Figure C.1.

The File menu has the usual items: New, Open, Save, and so on. One menu item is Export, which allows you to export a report, a summary, the input, and any messages provided during the run. The report, in particular, is handy to have when done as a printed summary.

*Aspen Plus is a registered trademark of Aspen Technology, Inc. This book uses version 12.1.

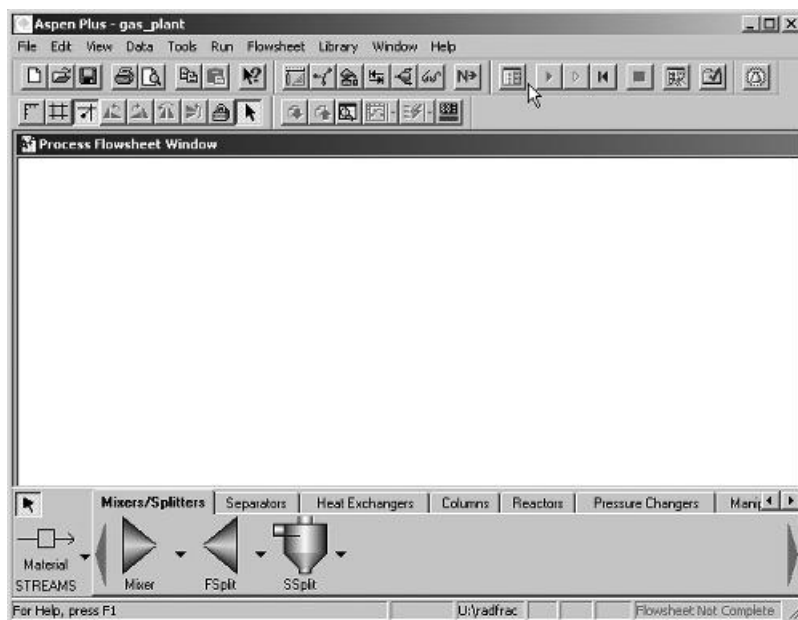


Figure C.1. Opening screen.

Several icons in Figure C.1 are used over and over, so it is worthwhile to identify them. Find the pair of eye glasses in the top row. This icon opens the Data Browser, which provides a list of windows to be completed. The **N>** icon can be used to proceed to the next step. By pressing this icon, filling out the window that appears, and pressing it again, you can proceed through the data entry phase of a problem easily. The **>>** (shown in Fig. C.2) can be used to step from window to window, but it includes all the windows, even if they are complete. Along the lower part of the screen are three items described in detail below: The Model Library (Mixer, Separators, Heat Exchanger, etc.) is where you find the models or units for your process, The Material Streams is used to insert streams into your process, and the left-arrow is used to select units or streams, perhaps to delete them. Locate each of these icons in Figure C.1 because they are especially useful.

Setup

You can specify a title for your problem, choose the units you want (these can be overridden in each stream and block), and create report options. To do this, click on the glasses (Data Browser) and double-click the folder Setup. In the Setup/Report Options window you put checks in the boxes for the properties you wish to include in your report, as illustrated in Figure C.2.

Model Library

The possible models are shown in the menu bar at the bottom (see Figure C.3.) If this does not appear, choose the View/Model Library menu. The models are organized by function,

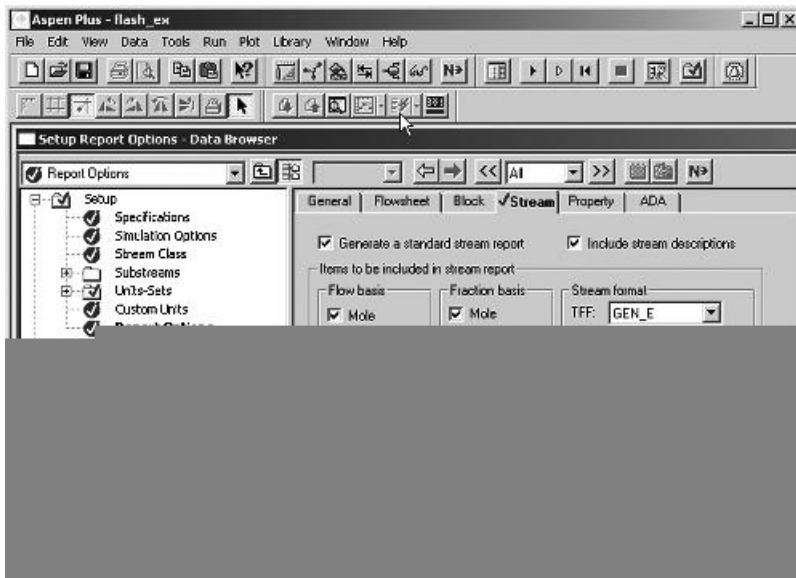


Figure C.2. Setup/Report options for streams.

such as Mixers/Splitters, Separators, Heat Exchangers, Columns, or Reactors, and so on. Click on the tab to see the different possibilities: A new menu lists the different types within that class. Click on Columns, for example, and you see choices such as: DSTWU, Distl, RadFrac, and so on. A brief description about the unit is displayed beneath the model library, but that may not show up unless you have maximized the screen.

Place Units on Flowsheet

To place a model or unit operation on your flowsheet (now called a block), click on the desired model and then click in the screen where you want the block to appear. Figure C.4 shows a DSTWU block. Continue in this fashion until you have the blocks you want. To delete a block, click on it and either choose Edit/Delete or right click and scroll down to delete.

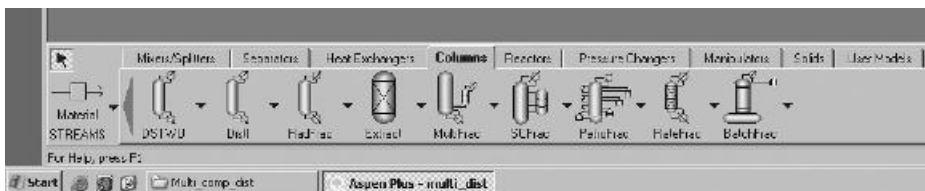


Figure C.3. Model library.

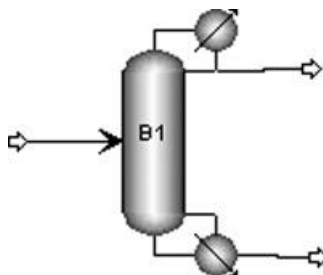


Figure C.4. Flowsheet of a distillation column.

Connect the Units with Streams

Click on the 'Material Streams' icon in the lower, left corner of the screen (see Fig. C.1, where it appears in the lower left-hand corner, or Fig. C.3 where it appears on the left). Then click in the flowsheet where you want the stream to originate and drag the cursor to where you want it to end. Continue in this way until you have placed all the streams in your flowsheet. When you are done, click the arrow to the upper left of the 'Material Streams' box (Figs C.1 and C.3) to disconnect the drawing option. If you added streams you do not want, select them and choose Edit/Delete or right click and scroll down to delete. You can also rename units and streams by using right click and scrolling down to rename.

Data Entry

Aspen Plus has a menu that walks you through the procedure to specify a process. If you click on , a window appears that tells you the next thing you should do. Alternatively, click on the glasses at the top (Data Browser), which places a menu on the left of the screen. The menu items that are in red are not complete, and must be completed. Double click them in turn, starting from the top. Then click on the red sub-menus. A new window will appear, asking you to fill in the needed information. To close up the menu, click on a folder, and the subfolders beneath it will be hidden again. A third alternative is to change the 'All' shown in Figure C.5a to 'Input' and click on the $\gg$. This takes you through each window one by one.

Specify Components

Type the name of the component in the left-hand column of the window. If you type in a name and the third column is not automatically filled in (as in Fig. C.5a for C5H12), double-click on the blank box in the third column. Type the name into the window that appears (Fig. C.5b), choose Find Now, and search for your chemical in the list provided. Close the window after specifying all the chemicals appearing in your process.

Specify Properties

You use Property Method to tell Aspen Plus what thermodynamic model you want to use. Initially all models are available. You can limit the choices by selecting an industry or application, and then only the pertinent choices are available. Chapter 6 discusses this

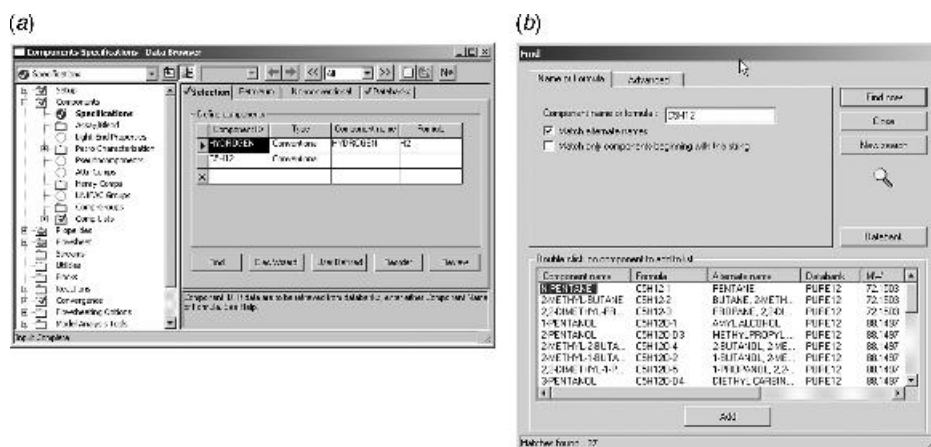


Figure C.5. Component identification.

choice in more detail, but this is one decision that must be validated by comparison with experimental data. In a design situation this is essential.

Specify the Input Streams

When you open up the stream folder (on the left), the streams that are input streams to the process will be shown in red. Click them in turn, filling in the properties of that stream. You must specify the flow rate of each species (even those not in that stream, as zero), the temperature, and the pressure, as shown in Figure C.6. You can specify mole fractions

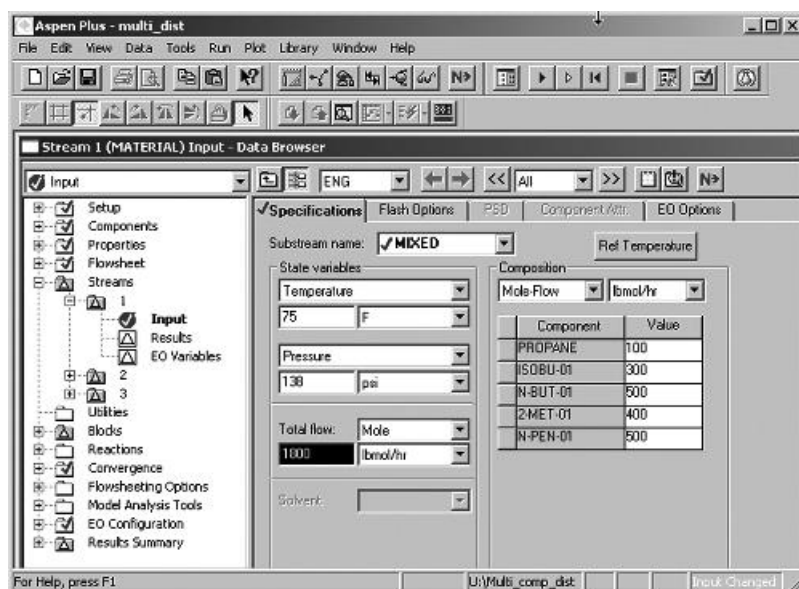


Figure C.6. Stream information.

and the total flow rate if you prefer. You do not need to specify conditions in internal or output streams; the computer will calculate those. Sometimes it is helpful to specify an internal stream to speed convergence, but you need to know which one to use (see Chapter 7).

Specify Block Parameters

Click each block in turn and fill in block parameters (see Fig. C.7). Sometimes the window has multiple options, which you need to complete if they are red.

Run the Problem

To run the problem, choose the blue **N>** in the new window and click OK or click on the closed triangle: **>>**. (If some data is not yet specified, the program will tell you.) By choosing the **N>** button, you get the most information back from the computer, including information about convergence, as shown in Figure C.8.

If you run the problem, make a change in a parameter, and run the problem again, Aspen Plus will use the first solution as the starting guess for the second problem. If you do not want this to happen, choose Run/Reinitialize or the reinitialize button: **⏮**, and the starting values are put back to the default values (flow rates are usually zero). You can also do the iterative calculations one unit at a time by choosing the Run/Step menu or Ctrl + F5 or the open triangle.

Scrutinize the Stream Table

In the left-hand window, click Results/Summary and Stream to get the mass and energy balances, the values of flow rate and enthalpy of each stream, as illustrated in Figure C.9. When the stream information is displayed on the screen, click Stream Table to get the information on the flowsheet.

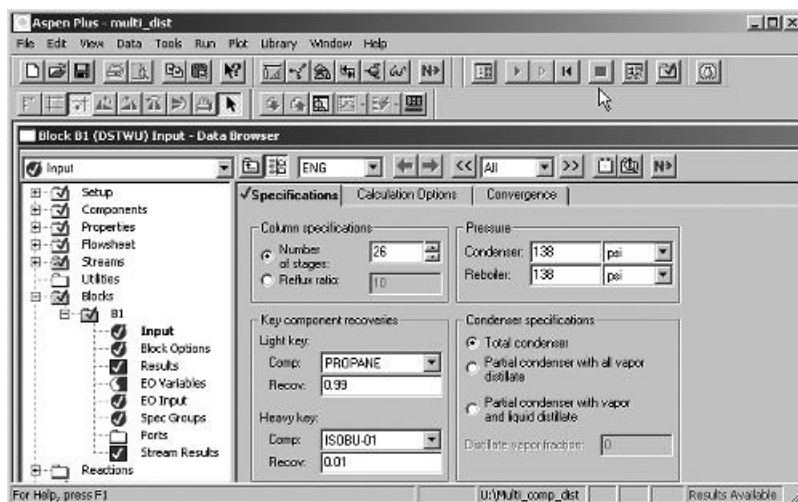


Figure C.7. Block parameters.

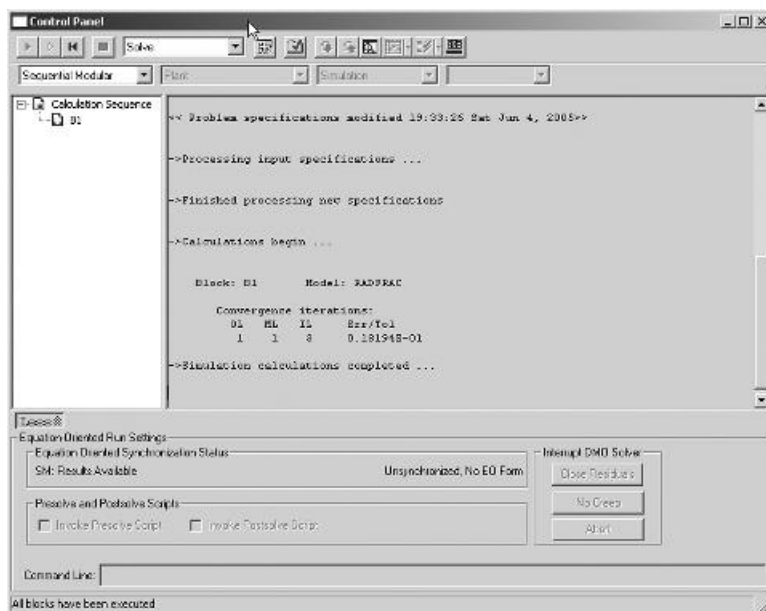


Figure C.8. Convergence information.

Checking Your Results

Take the streams and follow them through the process: Are the calculations correct? Did the computer do what you intended? Are the results reasonable?

You expect Aspen Plus to be correct, but there are two possible problems: lack of convergence and poor choices of thermodynamic correlations. By using the **N>** button to run your problem you will get printed information about the convergence or lack of it. *Read the output!* You get this information from the View/Control Panel menu, too. The proper choice of thermodynamic correlation can only be determined by comparison with experimental data or with experience. (This is one reason why chemical engineers are paid a lot – for their experience.) Naturally, at this point in your career, few of you have that experience. However, you can still look at your mass and energy balances and see if they make sense. *Every number needs to be examined.*

There are three important sources of summary information, all listed under the View menu: Input Summary, History, and Report. You should look at these – you may detect an error by viewing the summaries provided there.

Transfer the Flowsheet and Mass and Energy Balance to a Word Processing Program

Select the flowsheet by dragging the mouse from the upper-left corner to the lower-right corner. Copy the flowsheet and paste it into the application. You can also use the Print Screen key and then go to your word processing program and paste it. Use the Picture/Crop function to select the portion of the diagram that you want displayed. The best way to capture the Stream Table is to select the entire table and paste it into Excel. This is done from the Results/Summary/Stream option.

Results Summary Streams - Data Browser

Streams: 1, 2, 3

Material:

Display: Format:

	1	2	3	
Substream: MIXED				
Mole Flow lbmol/hr				
PROpane	100.0009	99.20908	.7929661	
I-BUTANE	299.9998	.7429663	299.2968	
N-BUTANE	499.9997	.0489218	499.9908	
I-PENTAN	399.9998	3.00065E-6	399.9998	
N-PENTAN	499.9997	5.35846E-7	499.9997	
Mole Frac				
PROpane	.0555560	.9920808	4.66392E-4	
I-BUTANE	.1666666	7.42996E-3	.1760034	
N-BUTANE	.2777776	4.89218E-4	.2940887	
I-PENTAN	.2222221	3.00065E-8	.2352940	
N-PENTAN	.2777776	5.35846E-9	.2941175	
Total Flow lbmol/hr	1800.000	100.0000	1700.000	
Total Flow lb/hr	1.15844E+5	4420.760	1.11423E+5	
Total Flow cuft/hr	3126.010	144.4390	3500.062	
Temperature F	75.00000	60.00000	200.7096	
Pressure psi	138.0000	138.0000	138.0000	
Vapor Frac	0.0	0.0	0.0	
Liquid Frac	1.000000	1.000000	1.000000	

Figure C.9. Stream table.

If you would like the temperature and pressure to appear on the flowsheet, ensure that Global Data is checked in the View menu, choose Tools/Options/Results View and click the boxes for temperature and pressure, as shown in Figure C.10a. Figure C.10b shows the resulting change in the flowsheet.

Change Conditions

To run the problem with different conditions, look for the word 'All' in a white box (see it in Fig. C.2). The options are All, Input, Results; scroll down to Input and make your changes using the menu on the left. Click the button to re-run the problem.

Prepare Your Report

Detailed information can be obtained using the View/Report menu. This gives details about individual blocks. The View/History menu gives information about the convergence and what calculations have been done. The View/Input Summary lists the parameters you have set. Every report should have a flow sheet with blocks and streams labeled, mass and energy balances referenced to the flow sheet, and a text description of the process. You should outline the problem, describe the choices you have made, and list all the ways you examined your results for validity. When you are *really* done, you can choose File and Export. In the 'Save as type' window, scroll down to report Files (*.rep; see Fig. C.11). Save that. It creates a text version of the whole simulation, giving all the detail you supplied and all the results. It is very useful to document your work (perhaps as an Appendix in a report), and the report gives the detail needed to

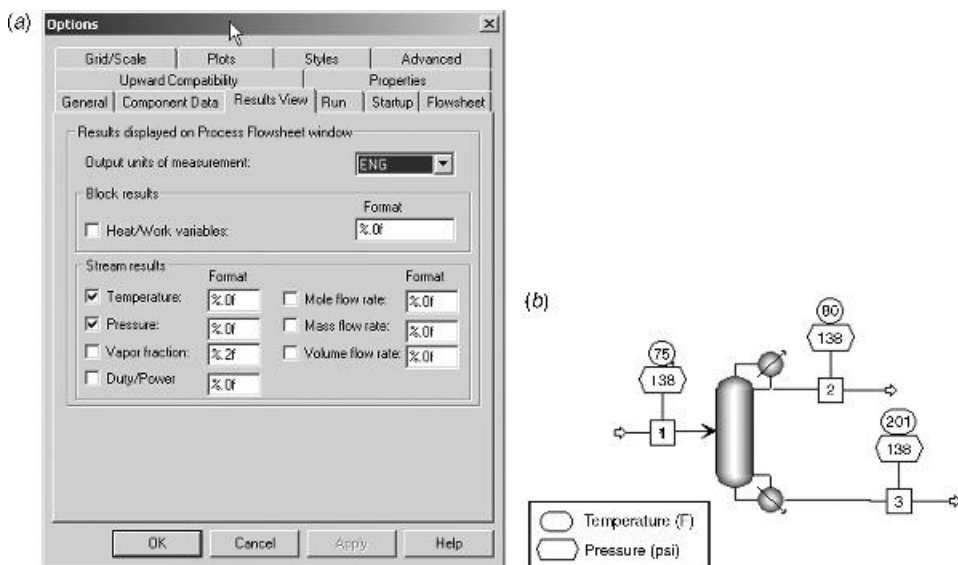


Figure C.10. Adding temperature and pressure to the flowsheet.

write a descriptive report. It does use paper, though, if you print it. For example, the report for the four-column gas plant problem in Chapter 7 was 10 pages long.

Save Your Results

Save your program and results in the computer memory. A good strategy is to create a folder for each new problem and save your program and results in the folder. The reason is that, when you choose File/Save, six files are created, and all six are necessary to restart the problem at a later time.

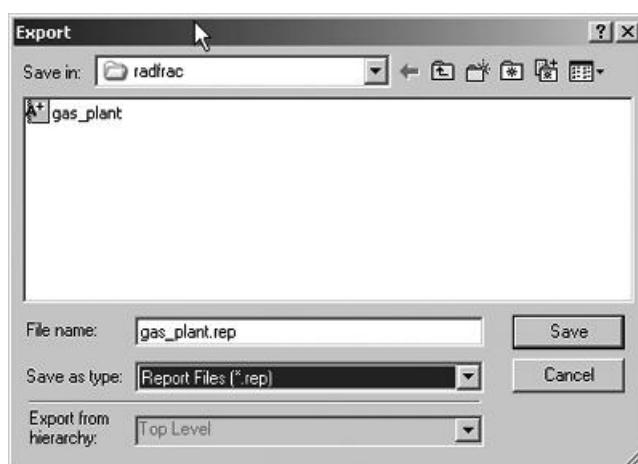


Figure C.11. Report option.

Getting Help

Click  and then the box on the left (in the Data Browser option). The part of the process represented by that box is then described. You can also click on the box and then the F1 key to obtain the same information. The Help menu provides detailed help in all areas. For example, the Help/Physical Property Methods/Choosing a Property Method gives advice about which thermodynamic model is recommended for different applications. The Guidelines for Choosing a Property Method and Guidelines for Choosing an Activity Coefficient Method are sub-menus that outline decision trees to guide your choice.

APPLICATIONS OF ASPEN PLUS

- Equations of state, Chapter 2, pp. 16, 18.
- Vapor–liquid equilibrium, Chapter 3, pp. 31, 35.
- Chemical reaction equilibrium, Chapter 4, p. 52.
- Multicomponent distillation with shortcut methods, Chapter 6, p. 75.
- Multicomponent distillation with rigorous plate-to-plate methods, Chapter 6, p. 80.
- Packed bed adsorption, Chapter 6, p. 82.
- Multicomponent distillation train, Chapter 6, p. 85.
- Process simulation with recycle and phase equilibrium, Chapter 7, p. 91.
- Utility costs, Chapter 7, p. 97.
- Optimization, Chapter 7, p. 105.

Appendix D

HINTS WHEN USING FEMLAB[®]

This appendix gives an introduction to the use of FEMLAB. The program FEMLAB,* uses the finite element method to solve fluid flow, heat transfer, and mass transfer problems, as well as many other equations. In the past decade, computer software has become very powerful, and this allows chemical engineers to solve very complicated problems. This poses a problem though: You may be tempted to pose a problem that is way more complicated than you need. Instead, heed Einstein: 'Make it as simple as possible, and no simpler.' Another problem is that many details of the computer program are hidden from you, or at least not readily accessible. By clicking buttons, or forgetting to make a particular choice, you may inadvertently solve the wrong problem.

On the other hand, engineers used to reduce all problems to ones that could be solved analytically, because then they could solve them. The assumptions used in the simplification might or might not be justified, but the analytical form of the answer aided their understanding of the problem. Modern computer programs can solve those simple problems, and the solution provides a test of the computer program and your use of it: Does the computer solution agree with the analytical solution when it should? However, the programs can also solve realistic problems that are impossible to solve analytically. What you have to do is: (a) Pose a realistic form of the problem; and (b) verify you have solved it accurately enough.

The finite element method replaces a differential equation with a large set of algebraic equations. The details to make this switch are complicated, but fortunately, FEMLAB has done that for you. You still need to know how to use the program, because, after all, it is up to you to decide if you have solved the right problem, determined the accuracy, and derived useful properties from the solution.

*FEMLAB is a registered trademark of Comsol, Inc. The program name has recently been changed to Comsol Multiphysics.

The examples are made with the Chemical Engineering addition to FEMLAB, version 3.1. Appendix F describes the finite element method in one dimension and two dimensions so you have some concept of the approximation: going from a single differential equation to a set of algebraic equations. This appendix presents an overview of many of the choices provided by FEMLAB. Illustrations of how FEMLAB is used to solve problems are given in Chapters 9–11. Thus, you may wish to skim this appendix on a first reading, and then come back to it as you use the program to solve the examples. A more comprehensive account of FEMLAB is available in Zimmerman (2004).

BASIC FEMLAB TECHNIQUES

Opening Screens

Figure D.1 shows the opening screen of FEMLAB. Look at the options across the top. The New tab is what is shown. If you click the Model Library tab, you get a menu of solved examples. These are ideal to investigate the program. The User Models tab provides a place you can put your examples, if you wish. Settings lets you choose a white or black background.

The space dimension can be 2D (shown), 1D, 3D, Axial symmetry (1D), or Axial symmetry (2D). Consider first the 2D option. If you double click on the Chemical Engineering Module, the screen changes to Figure D.2.

Equations

The Chemical Engineering Module has four folders, each of which contains a number of equations. The four folders are Energy balance, Mass balance, Momentum balance, and Pseudo 3D. Choose the Energy Balance option to get the next screen, as shown in Figure D.3.

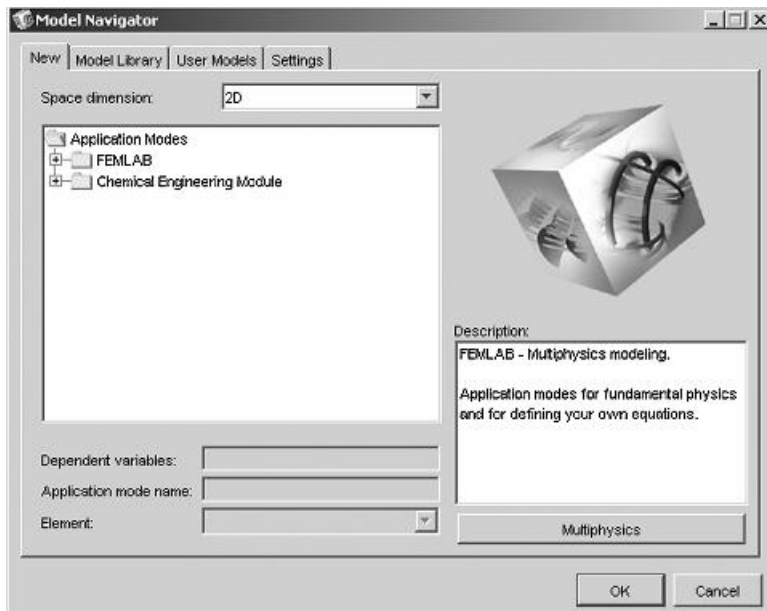


Figure D.1. FEMLAB opening screen.

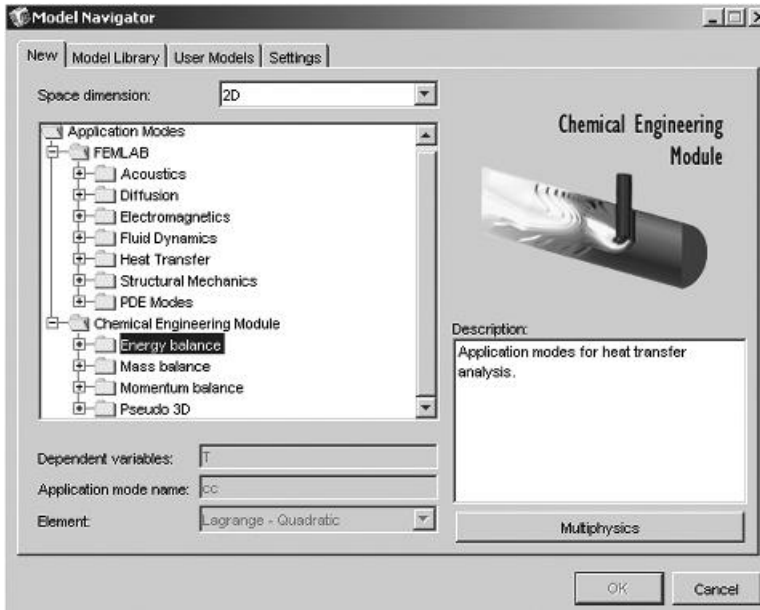


Figure D.2. FEMLAB opening screen for chemical engineering module.

There are two equations in this folder: 'Convection and Conduction' and 'Conduction,' and each of them has a steady-state analysis (highlighted) and a transient analysis. When you click OK, you will enter into the main screen, having picked the conduction equation in two spatial dimensions to solve. The dependent variable will be called 'T' and the

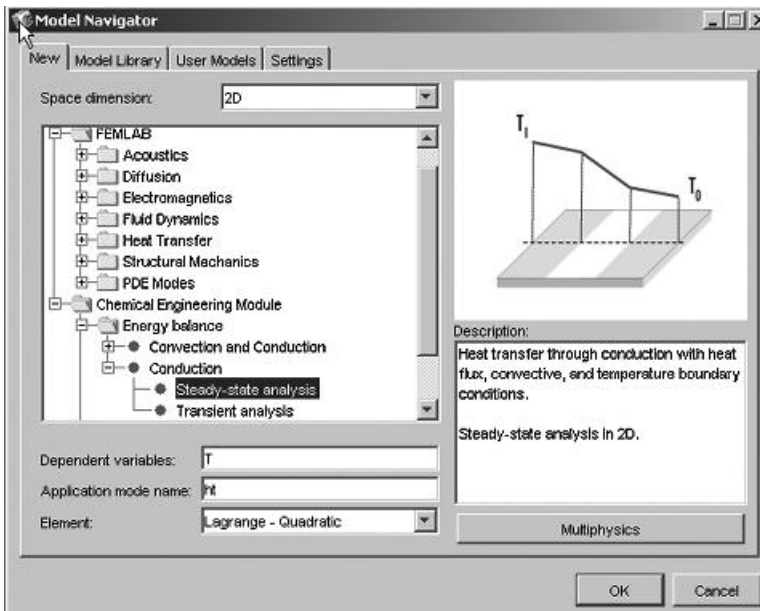


Figure D.3. FEMLAB opening screen for chemical engineering module, energy balance.

application mode name (i.e., the internal designation for the equation) is called ‘ht.’ You can change the dependent variable name here if you wish. The element chosen is Lagrange-Quadratic, which means that a quadratic polynomial is used to interpolate on each element and that the function, but not its normal derivative, will be continuous from one element to the other (Lagrange). The Multiphysics tab would enable you to add other equations, that is, solve combinations of the equations, simultaneously. Indeed, this is one of the valuable features of FEMLAB – to add equations, you simply choose them. The equations for the 2D option are shown in Table D.1. The other folders contain the applications listed in Table D.2.

The total numbers of equations available in each coordinate system are given in Table D.3. Clearly there are many, many options from which to choose! Next click OK to get the conduction equation, energy balance, chemical engineering module. This gives the screen shown in Figure D.4, which is the standard one for all 2D applications.

Draw

Consider first the icons along the top. From left to right they are: New, Open, Save, Print, Cut, Copy, Paste. After the arrow, you get to the finite element options. To explain them, though, you have to define a geometry. Choose the top-most icon on the left-hand side (the rectangle), and draw a rectangle in the blank screen. If you want to make it the same size as in the example below, double click on the rectangle and set the width to 1.2, the height to 0.8, and place the corner at $x = -0.8$, $y = -0.2$.

Mesh

Next click the first triangle to create the mesh. The mesh is a number of small triangles, nonoverlapping, that completely cover the rectangle. If you click the next triangle (the one with a small, inverted triangle inside the other one), the mesh is refined. If you click the next triangle, with a red color filling the inner triangle, you can select a

TABLE D.1. FEMLAB Equations in 2D Chemical Engineering Option

Energy balance	Convection and conduction*
	Conduction
Mass balance	Convection and diffusion*
	Diffusion
	Electrokinetic flow*
	Maxwell–Stefan diffusion and convection*
	Nernst–Planck*
Momentum balance	Brinkman equation
	Compressible Euler
	Darcy’s law
	$k-\varepsilon$ turbulence model
	Incompressible Navier–Stokes
	Nonisothermal flow
Pseudo 3D	Non-Newtonian flow
	Any applications with an asterisk above. In this option, the
	diffusion or conduction in the flow direction is eliminated

TABLE D.2. FEMLAB Equations in 2D Option

Acoustic
Conductive media DC
Electrostatics
Magnetostatics
AC power electromagnetics
Plane stress
Plane strain
Helmholtz equation
Laplace's equation
Poisson's equation
Schrödinger equation
Wave equation
PDE, coefficient form
PDE, general form

portion of the domain and refine only that portion. The first mesh is shown in Figure D.5. If you have refined the mesh but want the original, less dense mesh, simply click the first triangle again. Notice that once you create the mesh, the drawing tools on the left-hand side disappear. You can return to them by choosing the Draw mode in the heading, and if you change the drawing you would have to re-mesh.

Subdomain Settings

Next you would define the parameters in the differential equation. Choose the Physics menu at the top, and go down to Subdomain Settings. The screen appears as shown in Figure D.6. At the top is the equation that will be solved, Once a subdomain is selected (here the 1), you can change the parameters in the boxes. If you have more than one subdomain (perhaps two regions with different physical properties), there would be at least two numbers here and you could set the parameters for each region separately (or together).

Boundary Settings

Next choose the Boundary Settings in the Physics menu. The screen appears as shown in Figure D.7. There are four numbers, which correspond to the four sides of the rectangle (i.e., segments of the boundary). Click on one segment. You can then set the boundary conditions, using four options: Heat Flux, Thermal Insulation, Temperature,

TABLE D.3. Number of Different Equations in Each Coordinate System

1D – 27
2D – 43
3D – 35
Axial symmetry (1D) – 16
Axial symmetry (2D) – 26

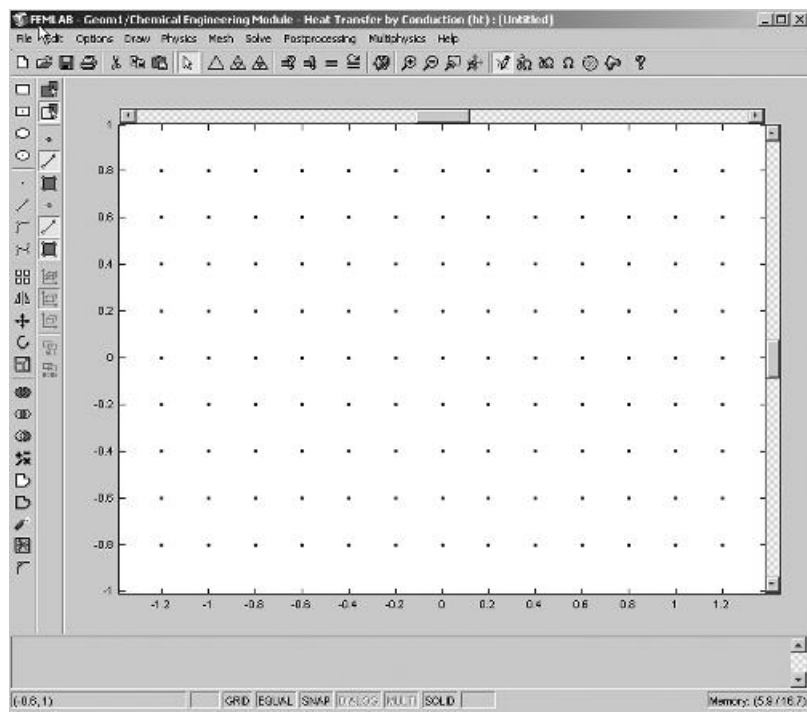


Figure D.4. FEMLAB opening screen for two-dimensional applications.

and Zero Temperature. If you choose Heat Flux, for example, you give a specified value for your problem. That value can be zero, in which case the option is the same as choosing Thermal Insulation. Alternatively, you can choose Temperature, and set the temperature (here set to 3.0). If you set it to zero, it is the same as the Zero Temperature option.

The boundary numbers are identified in Figure D.8. This example sets the temperature to 1.5 along boundary 2 and uses thermal insulation on boundaries 3 and 4 (shown in

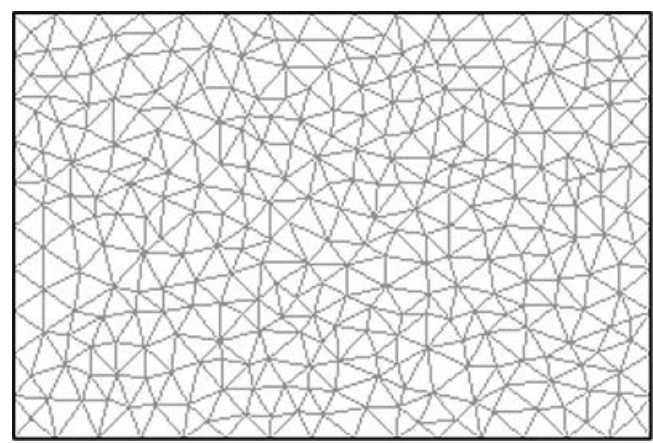


Figure D.5. Finite element mesh.

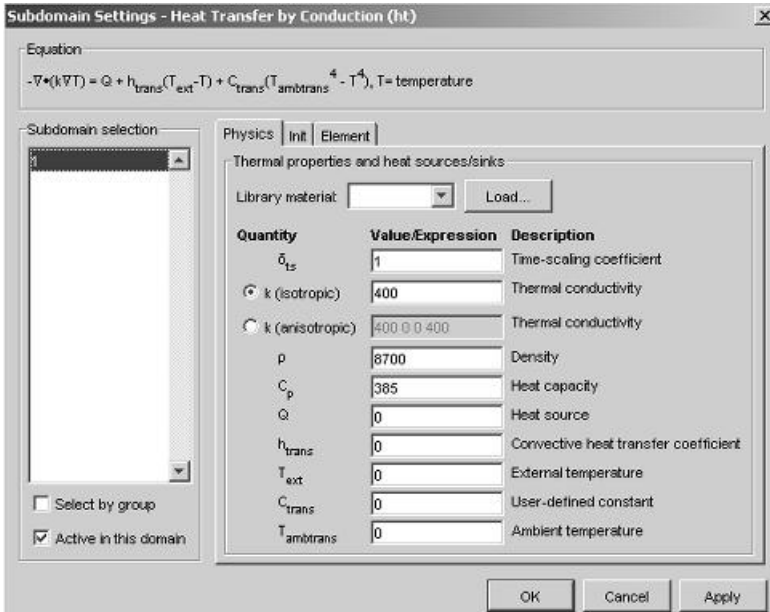


Figure D.6. Physics/Subdomain settings.

Fig. D.9). Notice that, as you select a boundary, the boundary in question turns red in the drawing. The heat flux option has a variety of parameters in it, because radiation is allowed in this equation, as shown in Figure D.10.

Solve

After you have set the boundary conditions, you can solve the problem by clicking on the '=' icon. The solution is found, and plotted for you. A color plot of the solution is

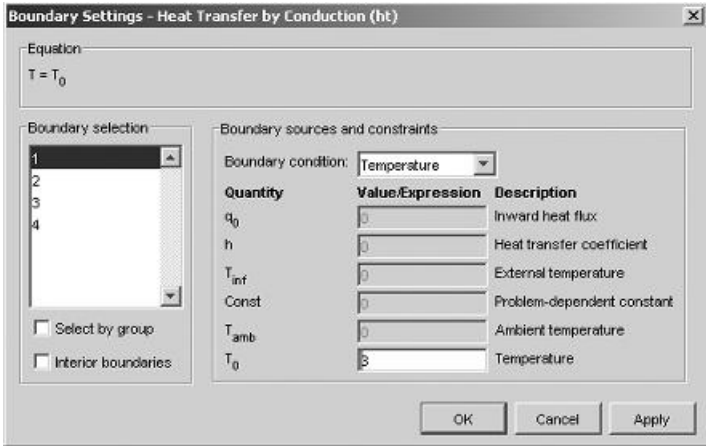


Figure D.7. Physics/Boundary settings for temperature.

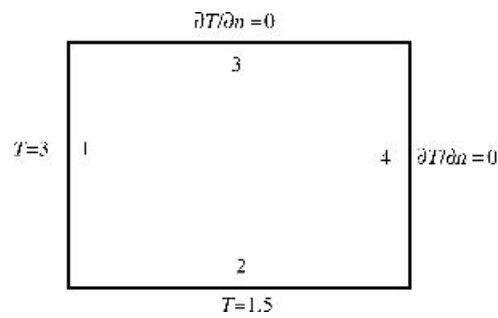


Figure D.8. Physics/Boundary settings for thermal insulation.

generated (see Fig. D.11 for the grayscale version). Notice that the color scale goes from 1.417 to 3.083. Because you know that the solution must be between 1.5 and 3, the values below 1.5 and above 3 indicate errors. Often these can be reduced by refining the mesh and re-solving. (To re-solve, choose the mesh refinement triangle, then choose the equal sign with the circle.) In this case, though, the problem was ill-posed, because the temperature at the corner is either 3 or 1.5; it cannot be both. Thus, the problem contains a singularity that should not be there.

POSTPROCESSING

Surface Plots

You really need more information than the color scale gives you, though, so choose the Postprocessing menu, then Plot Parameters. Make sure that there are checks in both the Surface box and Contour box, as shown in Figure D.12. Click the Surface tab to obtain Figure D.13. Click on the Contour tab to obtain Figure D.14. Contours are added to the plot. Click OK to obtain a new plot, Figure D.15. The scale on the right-hand side of your screen indicates the contour values.

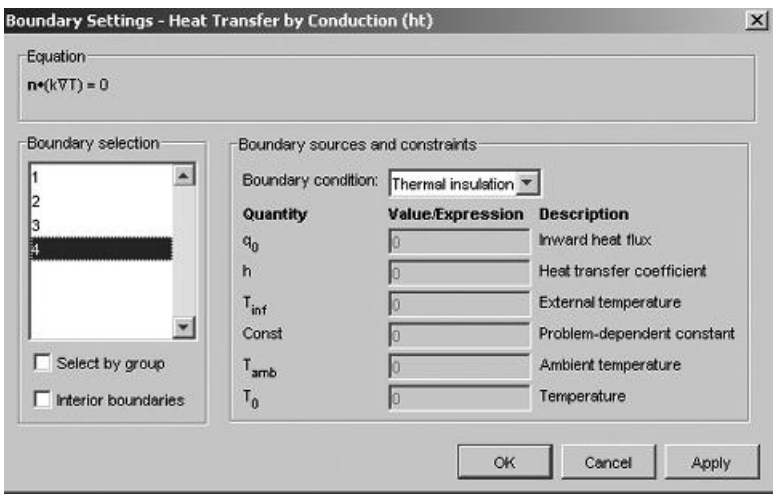


Figure D.9. Physics/Boundary settings for thermal insulation.

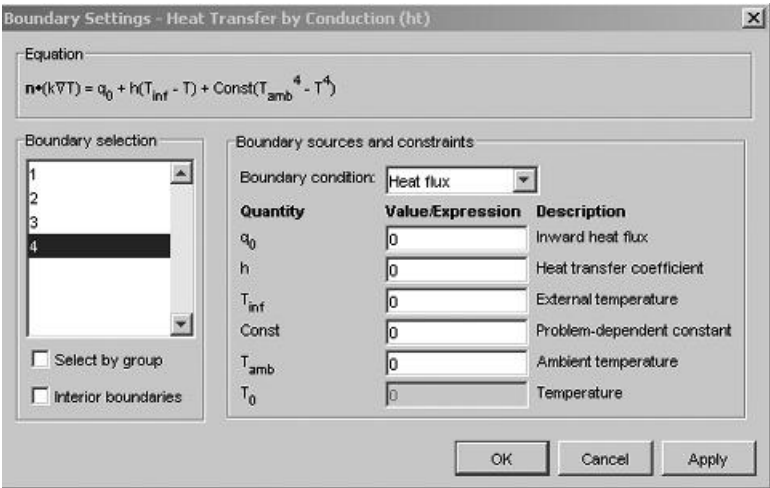


Figure D.10. Physics/Boundary settings for heat flux.

You can change what is plotted, too. Figure D.13 shows that the temperature is to be plotted. To view the properties that can be plotted, select the 'predefined' tab. They include: temperature, heat flux, x -component and y -component, heat flux (the magnitude), temperature gradient (magnitude), density, heat capacity, and heat source. This is where you get to explore your solution by plotting many features of it. It is also possible to plot one variable with the surface plot and another with the contour plot.

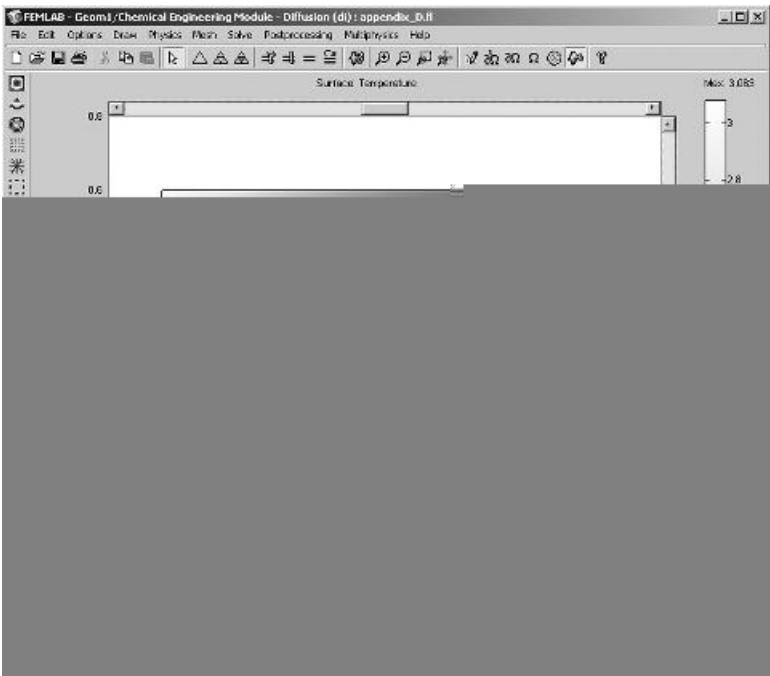


Figure D.11. Surface (color) plot of temperature.

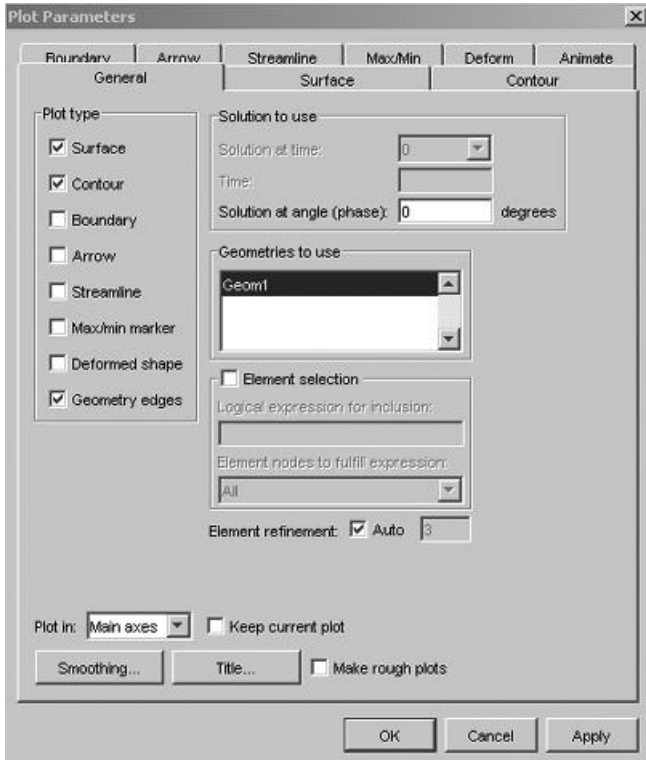


Figure D.12. Plot parameters.

Contour Plots

Similar options are available for the Contour plots, as shown in Figure D.14. You can set the number of contours to be plotted, and you can choose ‘Vector with isolevels’ and type in the numerical values of the contours you want: [1.5 1.6 2.0 3.0]. If you want your plot to have only contours, then unclick the box under the Surface tab, or go to the General tab and unclick the box next to Surface.

Cross-Sectional Plots

If you want a line plot through the domain, you can choose Postprocessing/Cross-sectional Plot Parameters, which gives the window shown in Figure D.16. Click on the Line/Extrusion tab to obtain Figure D.17. To plot the temperature along the line $y = 0$, from $x = -0.8$ to 0.4 , enter the values as shown in Figure D.17. A line plot then appears, Figure D.18. A line also appears on the original diagram (Fig. D.15) showing the location of the line.

Boundary Plots

Next plot the solution along a boundary. Choose Postprocessing/Domain Plot Parameters and click the Line/Extrusion plot, which gives the window shown in Figure D.19. Choose

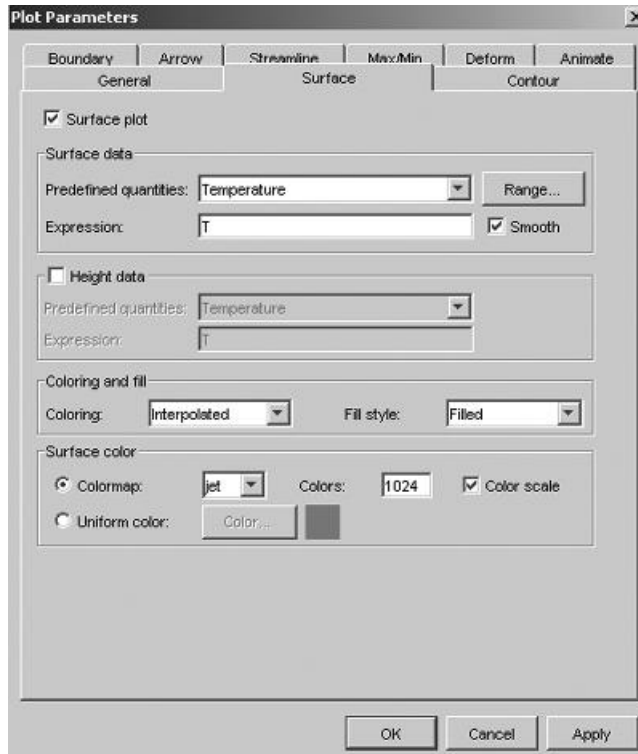


Figure D.13. Plot parameters, surface tab.

boundary segment one and click OK. The temperature is plotted along boundary segment one, as shown in Figure D.20. Now you see why you had a temperature above 3.0. Because of the discontinuity in temperature at the corner, the temperature boundary condition in the first element (which touches the corner) goes above 3.0. This is because the boundary condition requires the temperature to be 2.25 at the left point (the average of 3.0 and 1.5), then then 3.0 at the next two points. A quadratic polynomial is then made to go through these points, and it is guaranteed to go above 3.0 between the points. The actual finite element approximation is a quadratic, but it is plotted as straight lines, so you have to use your imagination.

Integrals

You can also calculate integrals. Choose Postprocessing/Subdomain Integration to obtain Figure D.21. The temperature will be integrated over the subdomains chosen (here there is only one). Click Apply, and look at the window at the bottom of the screen. The value is 1.952224. This is the value of

$$\int_A T \, dx \, dy \quad (\text{D.1})$$

If you want to know the area of the rectangle, type 1.0 in the box and choose Apply. The value is 0.96.

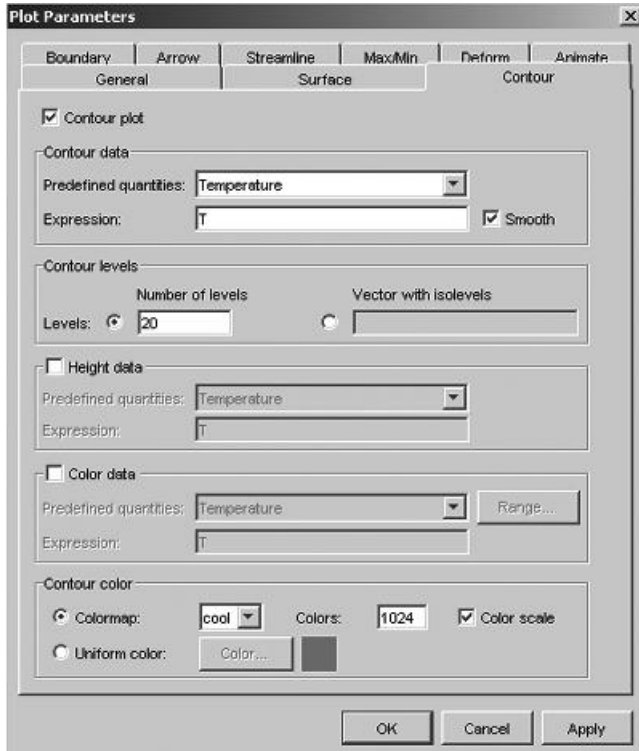


Figure D.14. Plot parameters, contour tab.

You can also choose Postprocessing/Boundary Integration and integrate the temperature along boundary 2. The value is 1.81, as shown in Figure D.22. If you type 1.0 in the Expression box and choose Apply, you find the length of boundary 2 is 1.2:

$$\int_s T \, ds = 1.81, \quad \int_s 1 \, ds = 1.2 \tag{D.2}$$

ADVANCED FEATURES

Model Navigator

Suppose you want to solve the heat conduction equation along with a diffusion equation. Choose the Multiphysics menu, and then Model Navigator. The same window opens that you used to select the equation in the first place, as shown in Figure D.23. Navigate to Mass Transfer/Diffusion/Steady State Analysis and choose Add. The diffusion equation is added to your problem and the dependent variable is called *c*, as shown in Figure D.24.

Click OK. Now you can set the parameters for the diffusion equation and boundary conditions. When you are using the Physics menu, the equation that will be displayed is the one chosen in the Multiphysics menu. You can switch back and forth; the operative equation has a check beside its name.

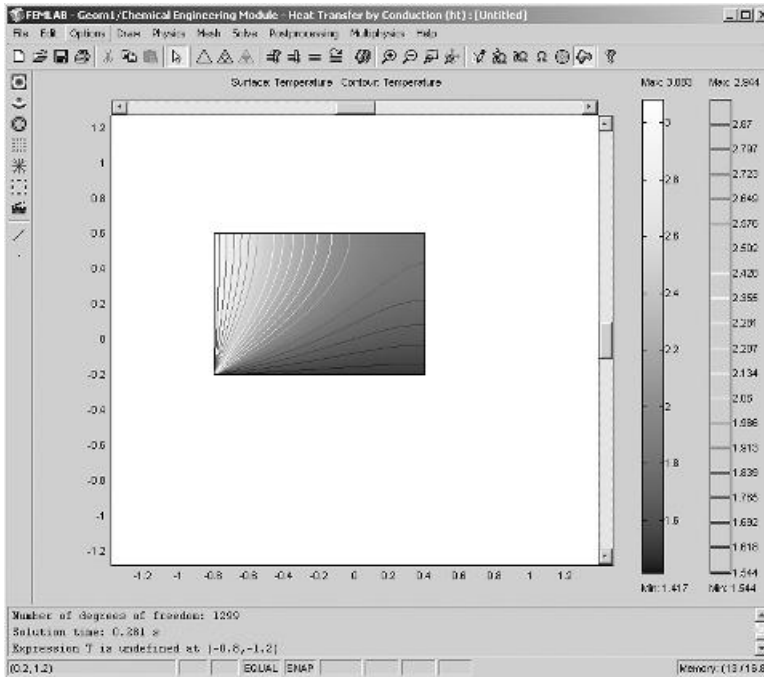


Figure D.15. Surface plot and contour plot of temperature.

Internal Boundaries

You can create many complicated geometries using the drawing tools. Draw a rectangle and circle in FEMLAB, as shown in Figure D.25. Choose Draw/Create Composite Object to obtain Figure D.26. The two objects, the rectangle and circle, are identified by R1 and E1, respectively. You can create the geometry shown in Figure D.27 by typing $R1 - E1$ in the 'Set formula' box, shown in Figure D.26. You can create the geometry shown in Figure D.28 by typing $R1 + E1$ in the 'Set formula' box.

Once you create the composite object, you can set parameters for the domain with one command, which applies to the entire object. To do this, unclick the 'Keep interior boundaries' box. The entire domain is then joined as one. However, if you want to specify different properties in different parts of the domain, click the 'Keep interior boundaries.' This allows you to specify different properties in the three regions shown in Figure D.25: the rectangle without the circle, the place where the circle overlaps the rectangle, and the part of the circle outside the rectangle.

Expressions

Suppose you wish to define a function that takes one value in part of the domain (called *expression1*) and another value in another part of the domain (called *expression2*). The Boolean operator allows you to do this. The function $(x < 0.25)$ takes the value 1.0 if x is less than 0.25 (true) and 0 if x is greater than or equal to 0.25. The following function provides an overall function that takes the appropriate value.

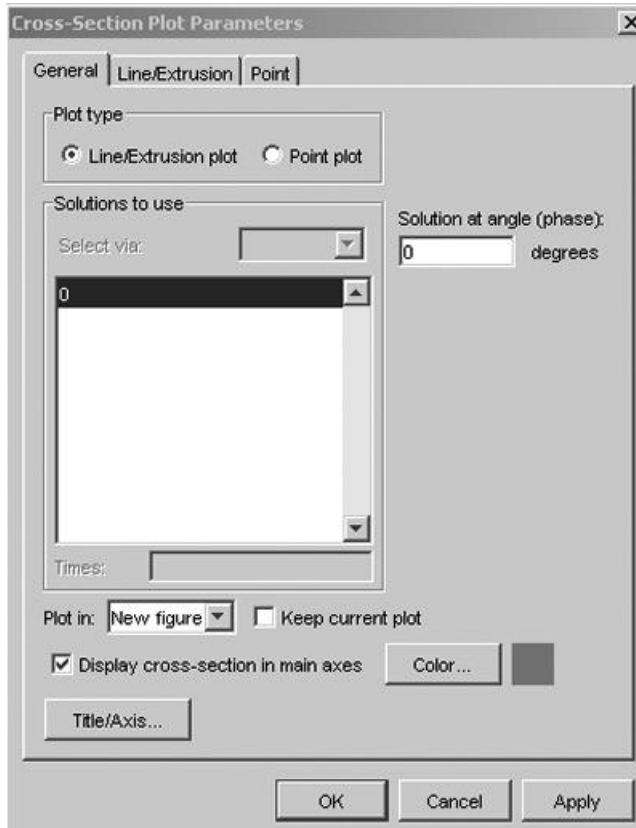


Figure D.16. Cross-section plot parameters.

$$Fn = (x < 0.25) * expression1 + (x \geq 0.25) * expression2 \quad (D.3)$$

Coupling Variables and Additional Geometries

Suppose that you have two problems to solve, and the solution to one of them provides conditions used in the other one. Here you will use coupling variables on the boundary to transfer the information; while the example is trivial, the method is useful in many cases and can be extended to subdomain transfers and point transfers. The two domains are shown in Figure D.29. The goal is to take the solution from the problem in Figure D.29a along the right-hand boundary and use it as the boundary condition along the left-hand boundary of the problem in Figure D.29b.

Step 1 The domain and mesh shown in Figure D.29a is created in the usual fashion, and the dependent variable is T .

Step 2 Click on *Geom1* and set the parameters and boundary conditions under *Physics/Subdomain Settings* and *Physics/Boundary Settings*.

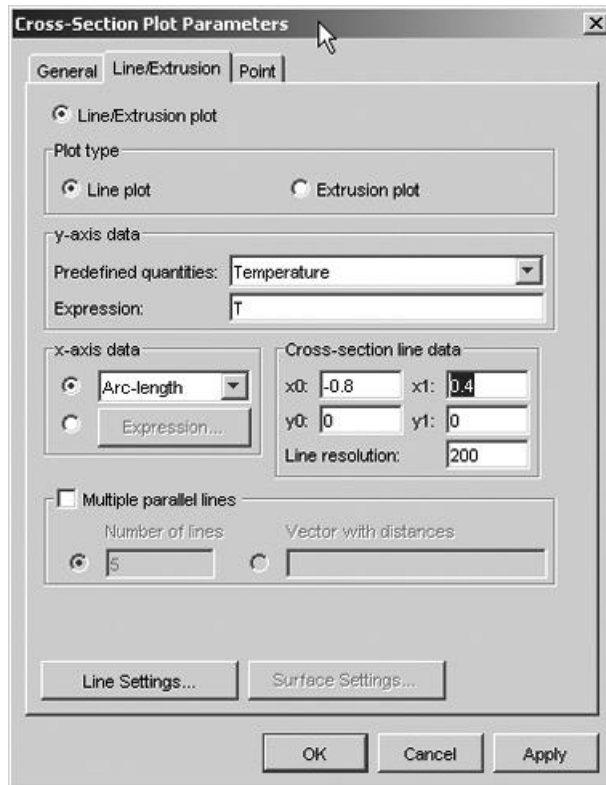
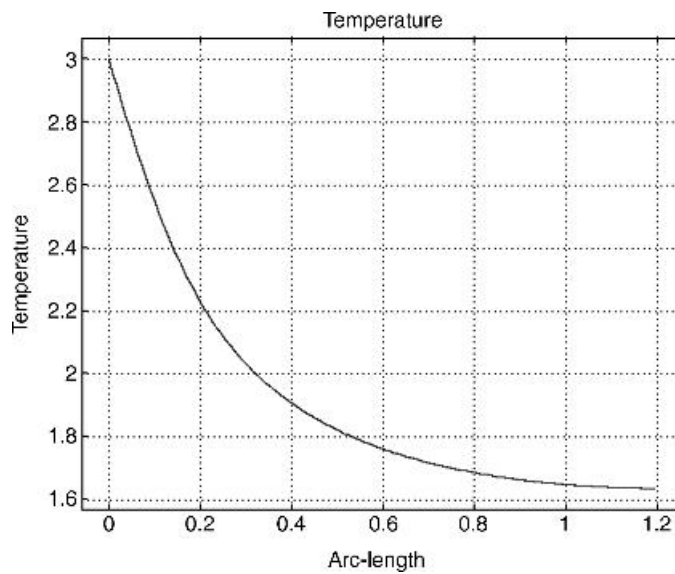


Figure D.17. Line/extrusion parameters.

Figure D.18. Temperature plotted along the line $y = 0$, $x = -0.8-0.4$.

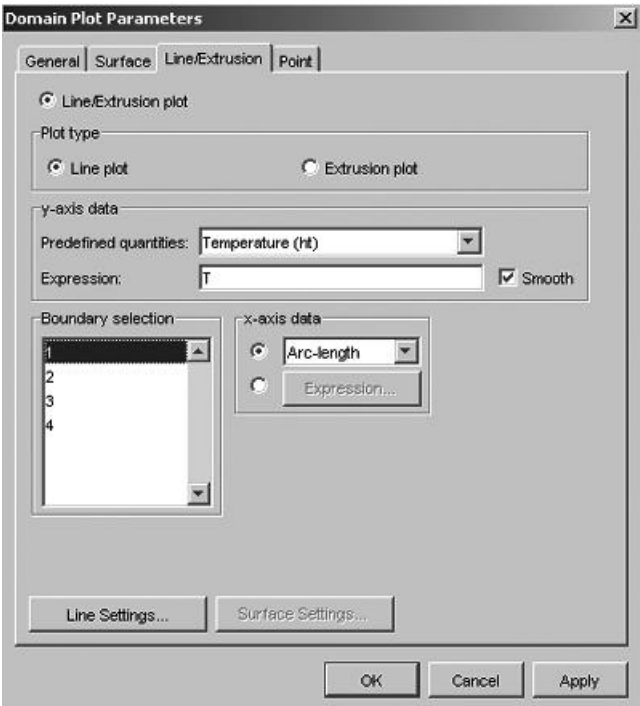


Figure D.19. Domain plot parameters to plot along a boundary.

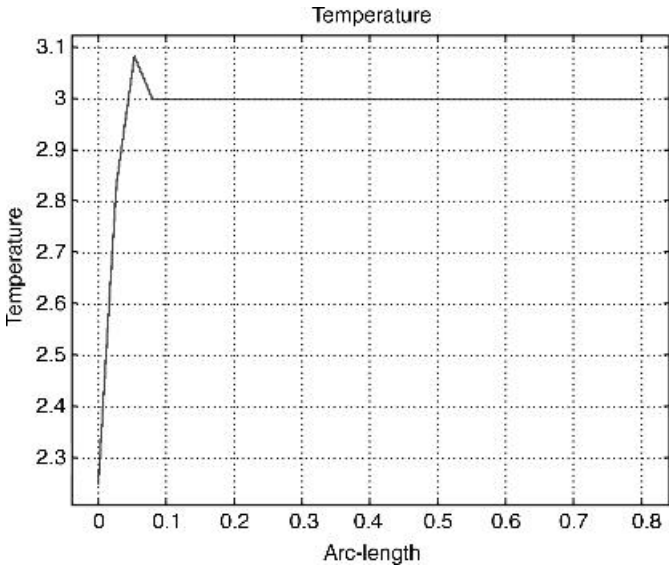


Figure D.20. Temperature plotted along boundary 1.

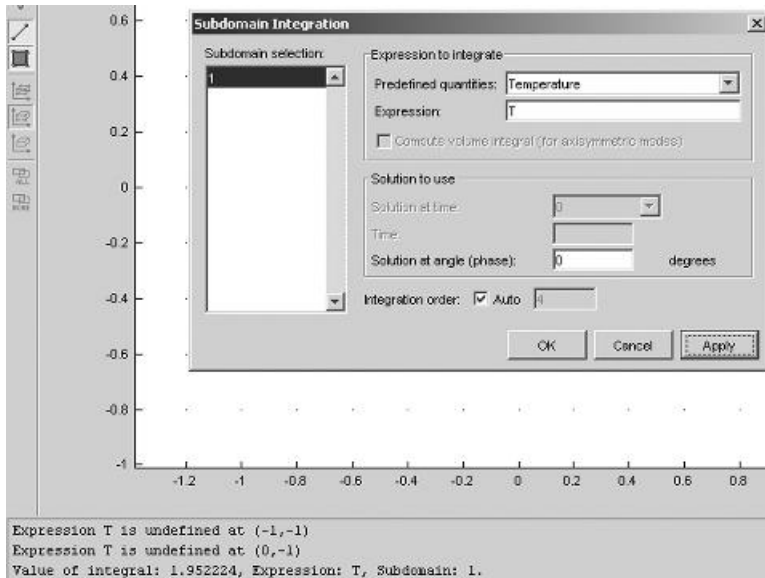


Figure D.21. Subdomain integration of temperature.

Step 3 Then use the Multiphysics/Model navigator menu, add a geometry, and choose an equation for the second geometry, as shown in Figure D.30. The dependent variable is T_2 . Click OK. Now there is a tab at the top of the screen listing Geom1 and Geom2, and Geom2 is selected.

Step 4 Add a domain and mesh in the usual fashion.

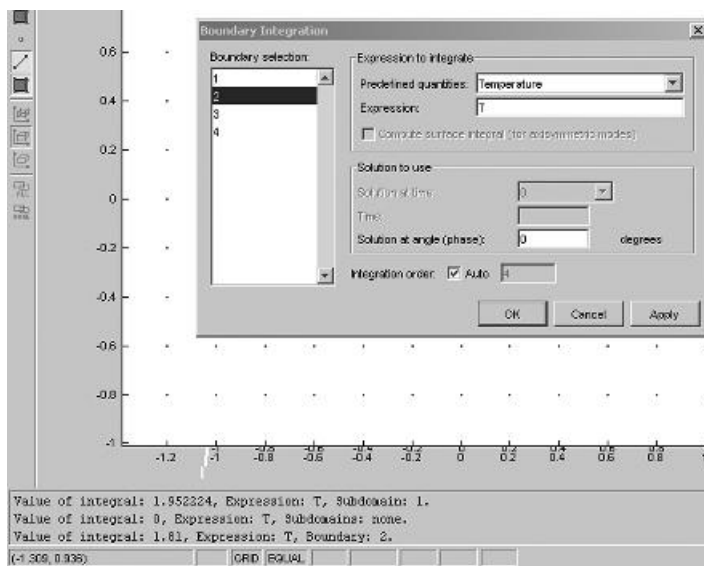


Figure D.22. Boundary integration of temperature.

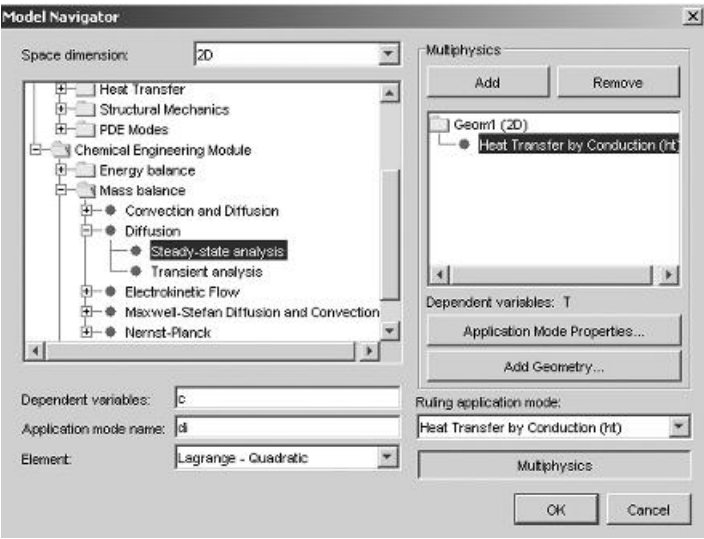


Figure D.23. Model navigator showing choices of equations.

Step 5 Make sure Geom2 is selected and set the parameters and boundary conditions under Physics/Subdomain Settings and Physics/Boundary Settings. For Boundary 1, choose the boundary condition ‘Temperature’ and type in *tout*.

Step 6 Click Geom1. Use the Options/Extrusion Coupling Variables/Boundary Variables menu, and the window shown in Figure D.31 appears. Choose boundary 4, and

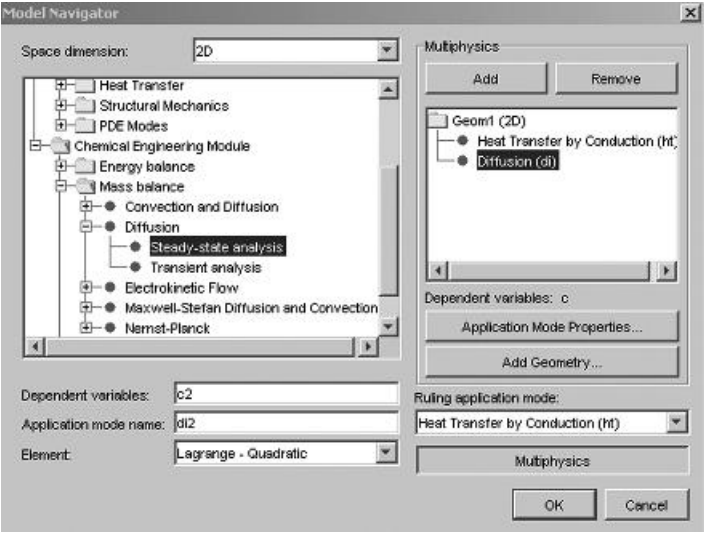


Figure D.24. Model navigator showing the addition of diffusion equation.

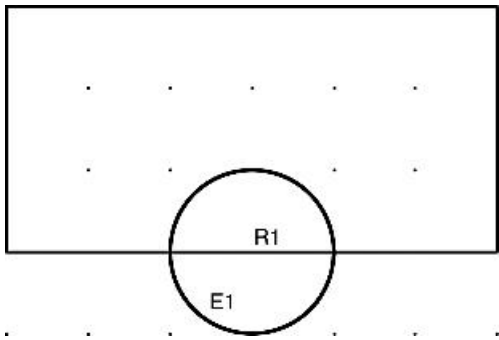


Figure D.25. Rectangle and circle.

name the coupling variable *tout*. Define the expression as *T*, which is the name of the dependent variable in Geom1. Click the General button, and set *x*: *y*. This is because the independent variable along boundary 4 is *y*. Then click on the Destination tab. Make sure Geom2 is selected, and the variable *tout* appears, as shown in Figure D.32. Select Boundary, click on '1', the boundary in Geom2 and click OK.

Step 7 Make sure Geom1 is still selected. In Solve/Solver Manager choose to solve only the equation in Geom1. Click on the '=' to solve this problem. Then click on Geom2, return to Solve/Solver Manager and choose to solve only the equation in Geom2. Choose the Initial tab and make sure the lower box is checked for 'use current solution' for variables not solved for. This insures that *tout* will be used. Click on the '=' to solve this problem.

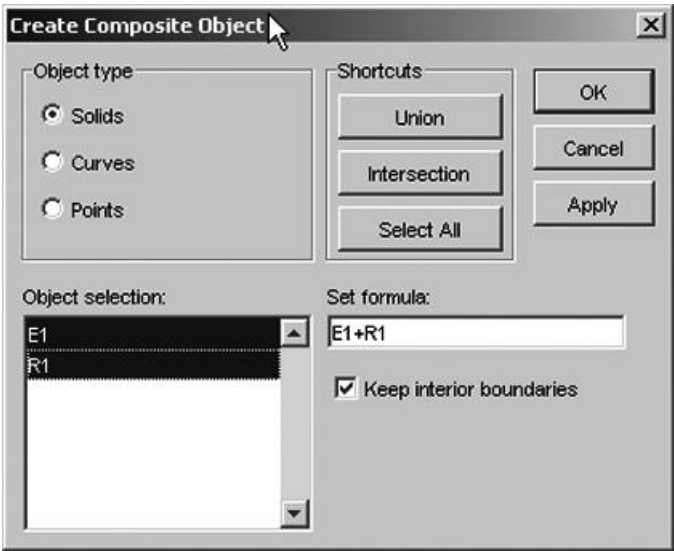


Figure D.26. Create composite object.

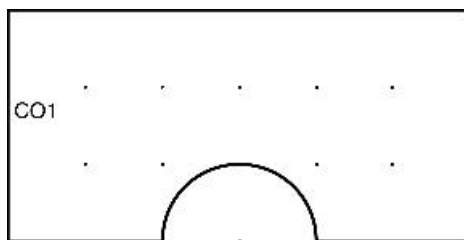


Figure D.27. Composite object, $R1 - E1$.

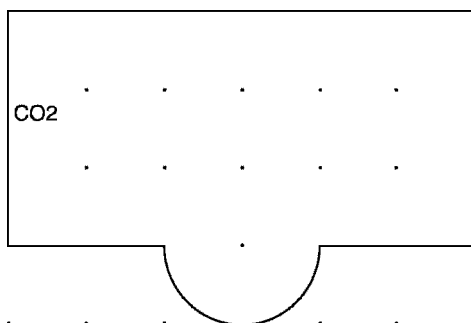


Figure D.28. Composite object, $R1 + E1$.

Step 8 The solutions along boundary 4 in Geom1 and boundary 1 in Geom2 are shown in Figure D.33. (Since they are the same, only one figure is shown.)

APPLICATIONS OF FEMLAB

This FEMLAB options described in this Appendix merely touch the surface of possibilities. Additional examples and applications are provided throughout the book, as listed here:

- Using FEMLAB to solve ordinary differential equations, Chapter 8, p. 123.
- Isothermal plug flow reactor, Chapter 8, p. 125.

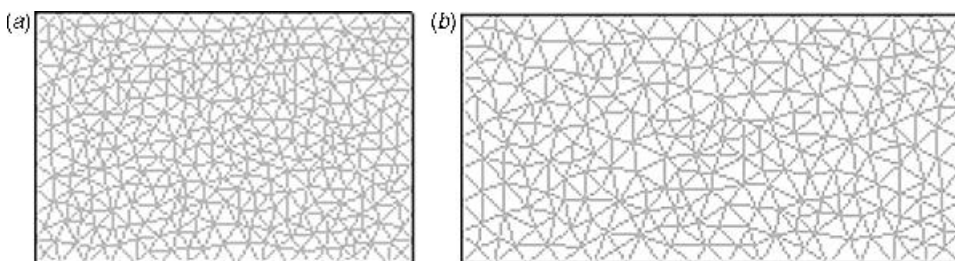


Figure D.29. Geometry 1 (a) and geometry 2 (b).

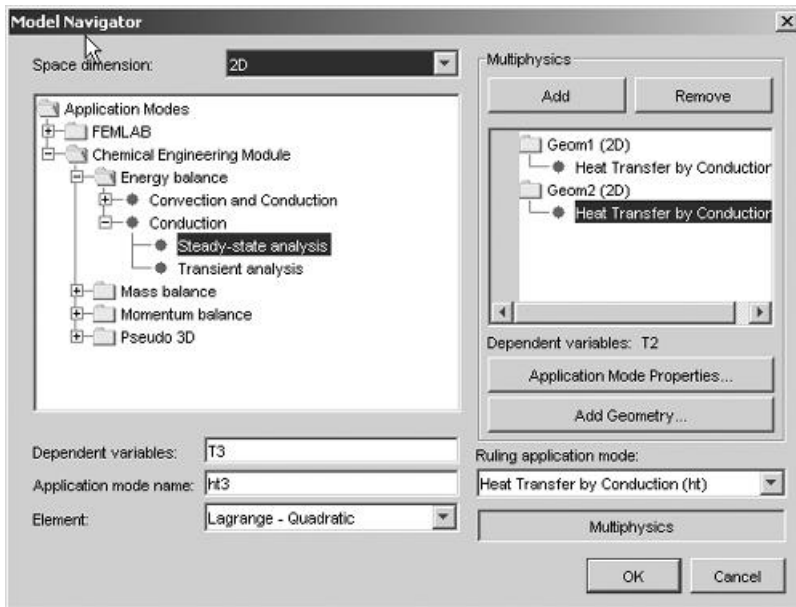


Figure D.30. Selecting second geometry and equation.

- Nonisothermal plug flow reactor, Chapter 8, p. 127.
- Reactors with mole changes and variable density, Chapter 8, p. 130.
- Heat transfer in a slab, Chapter 9, p. 152.
- Reaction and diffusion, Chapter 9, p. 154.
- Parametric solution, Chapter 9, p. 155.
- Flow of a Newtonian fluid in a pipe, Chapter 9, p. 156.
- Flow of a non-Newtonian fluid in a pipe, Chapter 9, p. 159.

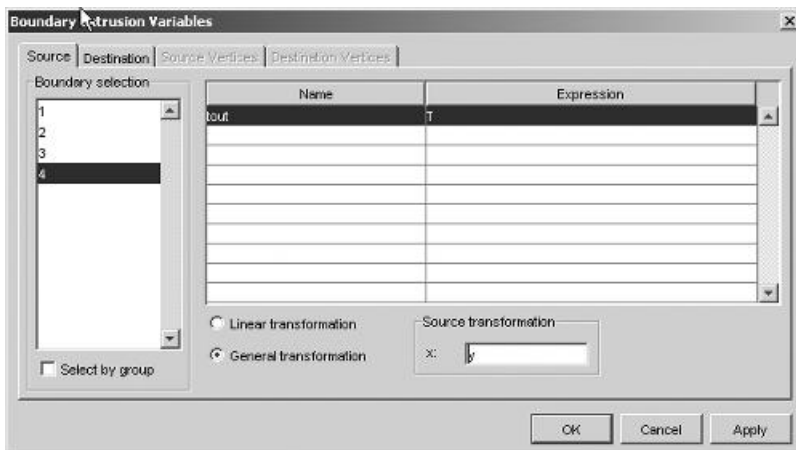


Figure D.31. Source coupling variable.

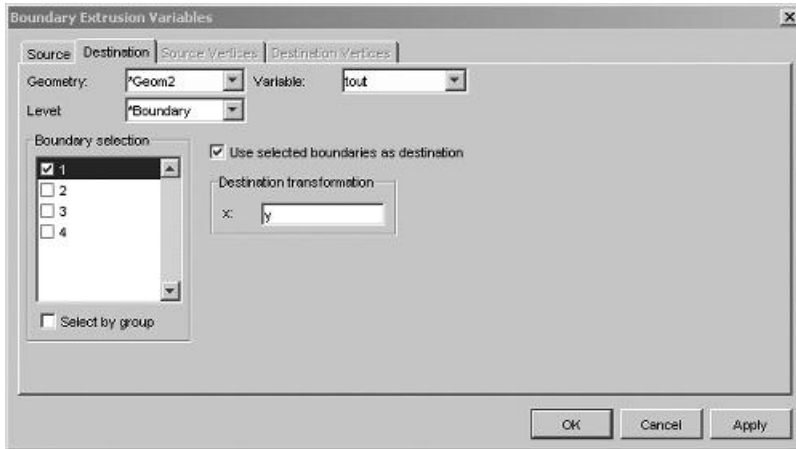


Figure D.32. Destination coupling variable.

- Tutorial for using FEMLAB to solve two-point boundary value problems, Chapter 9, p. 160.
- Transient heat conduction, Chapter 9, p. 162.
- Linear adsorption, Chapter 9, p. 164.
- Chromatography, Chapter 9, p. 167.
- Tutorial for using FEMLAB to solve transport problems, Chapter 10, p. 178.
- Entry flow in a pipe, Chapter 10, p. 179.
- Entry flow of a non-Newtonian fluid, Chapter 10, p. 184.

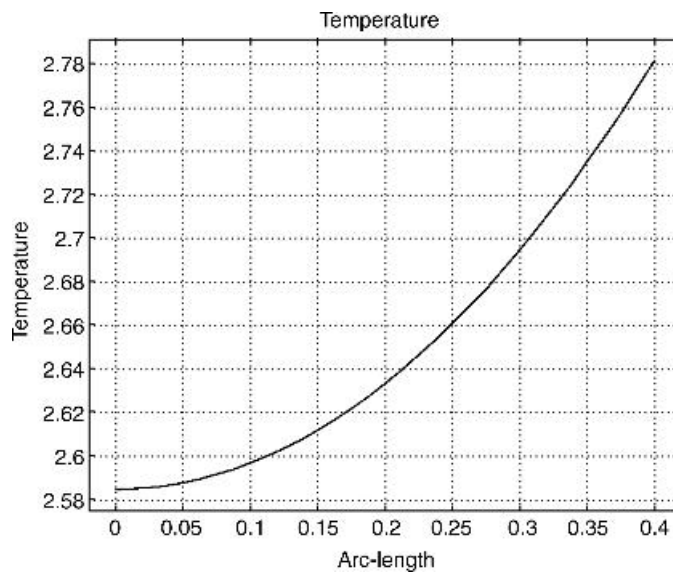


Figure D.33. Solution along boundary 4 in geometry 1 and boundary 1 in geometry 2.

- Flow in a microfluidic device, Chapter 10, p. 186.
- Turbulent flow in a pipe, Chapter 10, p. 189.
- Start-up flow in a pipe, Chapter 10, p. 191.
- Flow through an orifice, Chapter 10, p. 193.
- Flow in a serpentine mixer, Chapter 10, p. 199.
- Heat conduction in two dimensions, Chapter 11, p. 210.
- Heat conduction with a hole, Chapter 11, p. 213.
- Dispersion in microfluidic devices, Chapter 11, p. 214.
- Concentration-dependent viscosity, Chapter 11, p. 217.
- Viscous dissipation, Chapter 11, p. 218.
- Chemical reactions, Chapter 11, p. 221.
- Mixing in a serpentine mixer, Chapter 11, p. 222.

Appendix E

PARAMETER ESTIMATION

Engineers often need to express experimental data in terms of an equation. They must decide on the equation and then determine the parameters that provide the best fit to the data. The problem is simplest if the equation is linear. This appendix describes simple methods using Excel and MATLAB[®] for fitting a straight line to data, then for fitting a polynomial to data (polynomial regression), and finally for fitting any set of functions in which the unknown parameters appear linearly (multiple regression), as well as non-linearly. The data used in the examples is only illustrative, but it has scatter included, as you would find in data taken in the laboratory. After mastering the examples in this chapter, you will be ready to fit an equation of your choice to data you measure.

MATHEMATICAL FORMULATION

Consider a set of data

$$\{y(x_i)\}, \quad \text{for } i = 1-n \quad (\text{E.1})$$

and find an equation that models the data. Write the equation in a general form:

$$y(x; a_1, a_2, \dots, a_M) \quad (\text{E.2})$$

showing that it depends on x , but also on some unknown parameters, $\{a_1, a_2, \dots, a_M\}$. The goal is to find the set of parameters that gives the ‘best fit.’ The best fit is usually defined by minimizing the sum of the square residuals, where the residual is the difference between the predicted value and the data. Because the data may have errors in it, an exact fit will not

be possible in most cases. Thus, you minimize the variance of the residuals (Press et al., 1986, pp. 502–503):

$$\sigma^2 = \sum_{i=1}^N \frac{[y_i - y(x_i)]^2}{N}, \quad y \equiv y(x_i, a_1, a_2, \dots, a_M) \quad (\text{E.3})$$

If the parameters enter the equation linearly, then the minimization problem reduces to a set of linear equations which are solved easily by Excel and MATLAB. The effectiveness of the curve fit is often reported as values of the linear correlation coefficient squared, r^2 . The linear correlation coefficient is defined as (Press et al., 1986, p. 484):

$$r = \frac{\sum_{i=1}^N (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^N (x_i - \bar{x})^2 \sum_{i=1}^N (y_i - \bar{y})^2} \quad (\text{E.4})$$

Values of r near 1 indicate a positive correlation; r near -1 means a negative correlation and r near zero means no correlation.

STRAIGHT LINE

This section describes how to fit a straight line to tabular data using Excel and MATLAB. The data in Table E.1 represent seven measurements, in columns A and B. The goal is to find an equation, $y = a + bx$, that best represents this data.

Straight Line Curve Fit Using Excel

Step 1 First insert the data into Excel as shown.

Step 2 After inserting the data, select the slope cell, E3, and insert the command:

$$=\text{SLOPE}(B2:B8, A2:A8) \quad (\text{E.5})$$

The result is 1.082. Notice that the y -locations are entered first, followed by the x -locations.

TABLE E.1. Simulated Data for Two Measurements of the Same Thing (x and y)

	A	B	C	D	E
1	x	y			
2	110	97			
3	210	206		Slope	1.0820
4	299	310			
5	390	386		Intercept	−22.303
6	480	521			
7	598	551		r^2	0.9687
8	657	742			

Step 3 To get the intercept, you do the same thing for the intercept cell, E5:

$$=INTERCEPT(B2:B8, A2:A8) \quad (E.6)$$

Step 4 Finally, the r^2 value is found using the following command in cell E7:

$$=RSQ(B2:B8, A2:A8) \quad (E.7)$$

The curve fit is then

$$y = -22.303 + 1.082x \quad (E.8)$$

Because the r^2 value is 0.9687, close to 1.0, the curve fit is good. You should plot the data, and this is easily done using the ‘trendline’ option, as described below.

Plotting the Trendline

Step 1 Select the data in columns A2:B8. Then choose ‘Insert/Chart’ and choose the scatter plot with no lines. Follow the instructions, adding titles, and so on, and place the chart on the spreadsheet.

Step 2 Put the cursor on any data point and right click (on the Macintosh, use Ctrl-click). A menu appears; choose ‘Add trendline.’ The trendline is added as shown. Put the cursor on the trendline and right click. Choose ‘Format trendline,’ then ‘Options’ and click ‘Display equation on chart’; also choose ‘Display R-squared on chart.’ The result is Figure E.1; notice that the equation in the figure agrees with Eq. (E.8).

Straight Line Curve Fit Using MATLAB

The same problem can be solved using MATLAB, too.

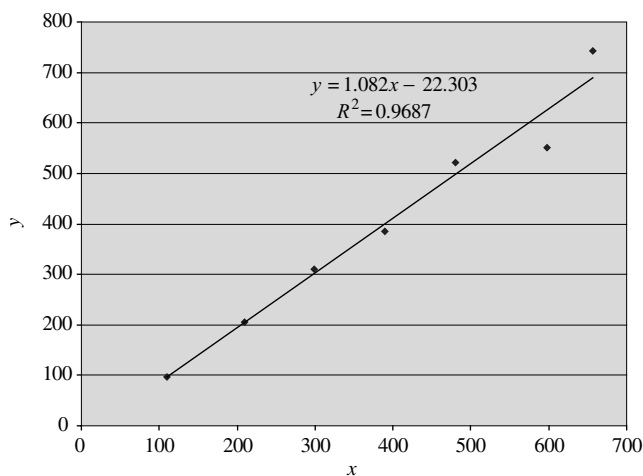


Figure E.1. Linear curve fit to data in Table E.1 using Excel.

Step 1 Put the x values into a vector, x , and the y values into a vector, y . Then issue the command

$$p = \text{polyfit}(x, y, 1) \quad (\text{E.9})$$

Step 2 Next evaluate the polynomial at a set of points from $x = 100$ to 700 for plotting purposes. Because the polynomial is a straight line, only two points are needed:

$$\begin{aligned} w &= [100 \ 700] \\ v &= \text{polyval}(p, w) \end{aligned} \quad (\text{E.10})$$

Finally, plot the data and the straight line; use red symbols for the data and a blue line for the straight line:

$$\text{plot}(x, y, 'ro', w, v, 'b-') \quad (\text{E.11})$$

The result is shown in Figure E.2. (The figure here appears in black, obtained by changing 'ro' to 'ko' and 'b-' to 'k-'.)

POLYNOMIAL REGRESSION

You need not be limited to a straight line when trying to fit data. The first extension is to use a polynomial and determine the coefficients that give the best fit. In that case, the formula is

$$y = a + bx + cx^2 + dx^3 + \dots \quad (\text{E.12})$$

or, more generally,

$$y(x) = \sum_{i=1}^N a_i x^{i-1} \quad (\text{E.13})$$

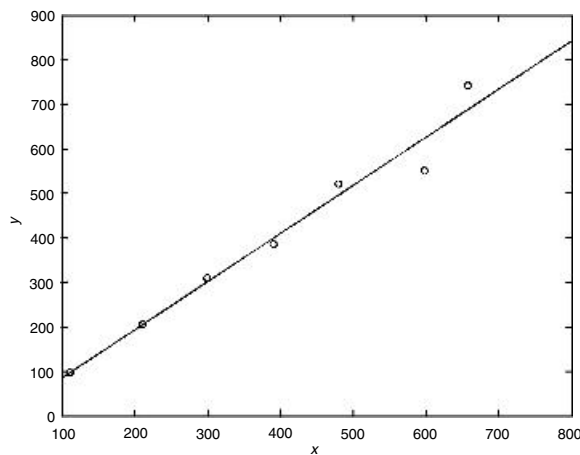


Figure E.2. Linear curve fit to data in Table E.1 using MATLAB.

You can easily carry out this process using either Excel or MATLAB.

As an example, consider the set of data in Table E.2 giving the measured partial pressure of a chemical at different times in a batch chemical reactor. You wish to fit this data to a linear or quadratic equation.

Polynomial Regression Using Excel

Steps 1 and 2 Follow the instructions given above to plot a trendline.

Step 3 Put the cursor on a data point and choose 'Add trendline.' This time, however, choose the option at the top right labeled 'Polynomial,' and pick an order of polynomial. By choosing 'Polynomial,' then 2, you can fit a quadratic function to the data, while choosing 'Polynomial,' then 3, gives a cubic function. The results are shown in Figure E.3a and 3b. The equation is displayed, along with the square of the linear correlation coefficient. The graphs show that the curve fit becomes better as the degree of polynomial is increased, and this is also reflected in the value of r^2 . In mathematical notation (Abramowitz and Stegan, 1964, p. 773), the order of a polynomial is the number of terms (or coefficients), whereas the degree is the highest power. Thus, a second-order polynomial includes two terms – the constant and linear term – and is a first degree polynomial. A third-order polynomial includes three terms, the constant and the coefficients of x and x^2 and is a second-degree polynomial. This nomenclature is not followed in Excel.

TABLE E.2. Partial Pressure versus Time

Time (sec)	Partial Pressure
0.00	7.45
1.02	8.98
1.82	8.80
2.87	10.1
3.83	10.9
4.96	12.1
5.97	12.8
6.60	12.8
8.11	14.4
8.36	13.7
9.10	13.9

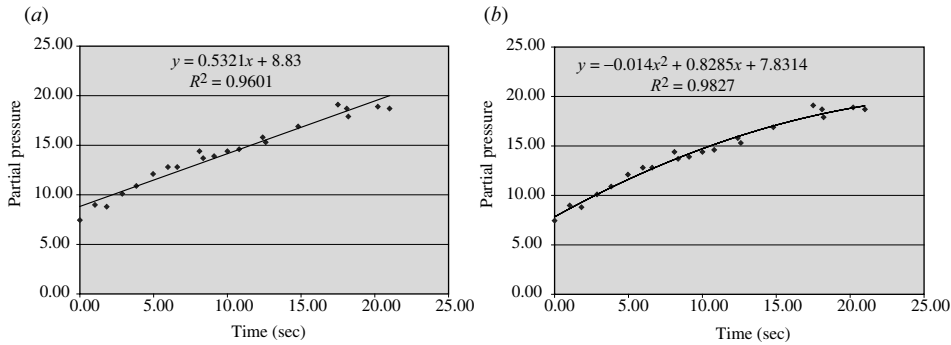


Figure E.3. Linear (a) and quadratic (b) curve fit to the data in Table E.2 using Excel.

Polynomial Regression Using MATLAB

Next consider MATLAB with a polynomial of higher order than a straight line. Put the data into the vectors t and p . The only change is in the 'polyfit' command – just type what order, n , you want:

$$pn = \text{polyfit}(t, p, n) \quad (\text{E.14})$$

This time, though, the curve fit is a curve. Thus, more points are needed to evaluate the curve. You also need to eliminate the previous solution (or at least the size of the vectors and matrices), because they may not be the same in this example. The complete set of commands to generate Figure E.4a is:

```
clear w v           % this eliminates the previous curve fit
p1=polyfit(t,p,1)   % for a 1-st order polynomial
w=[0:1:22]
v=polyval(p1,w)
plot(t,p, 'ro',w,v,'b-')
```

To obtain Figure E.4b, use the same commands except with

$$P2=\text{polyfit}(t,p,2) \% \text{ for a 2nd order polynomial} \quad (\text{E.15})$$

MULTIPLE REGRESSION USING EXCEL

The last example shows how to fit a polynomial to data. The same thing can be done when the functions are not simple powers, but are more complicated functions. However, to keep the problem linear, the unknown coefficients must be coefficients of those functions; that is, the functions are completely specified. Multiple regression simply determines how much of each one is needed. Thus, the form of the equation is

$$y(x) = \sum_{i=1}^M a_i f_i(x) \quad (\text{E.16})$$

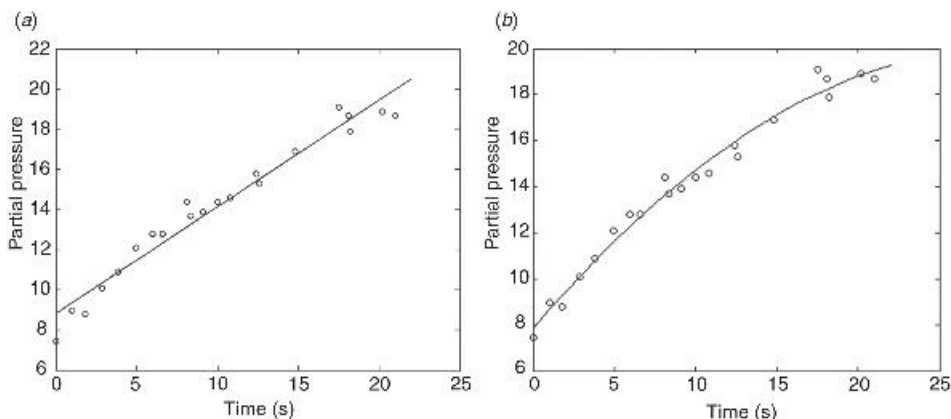


Figure E.4. Linear (a) and quadratic (b) curve fit to the data in Table E.2 using MATLAB.

The goal is to find the best M values of $\{a_i\}$, given the M functions $f_i(x)$ and data $y_i = y(x_i)$, $i = 1, \dots, N$.

In Excel, you put the x values in a column and create additional columns, with each column being a function, evaluated for the x value in that row. The example used here is to find the constants in a reaction rate formula. The expected expression is

$$\text{rate} = k p_A^n p_B^m \quad (\text{E.17})$$

and the goal is to find the values of k , n , and m that give the best fit of the *rate* for various partial pressures of substances A and B. This form is not linear, which is a requirement of multiple regression, but a transformation can make it linear. Take the logarithm of both sides of the equation.

$$\ln(\text{rate}) = \ln k + n \ln p_A + m \ln p_B \quad (\text{E.18})$$

This equation has the following form:

$$y = a + bx_1 + cx_2 \quad (\text{E.19})$$

where the dependence upon two or more variables is clear. The data is entered into the spreadsheet, and the various terms are transformed as shown in Table E.3. Columns A and B are the partial pressures of the two chemicals for which the rate is measured, as indicated in column C.

Step 1 Obtain columns D, E, and F by taking the logarithm of columns A, B, and C, respectively. Do this in the first cell (D2), copy it across the E and F rows, and copy down the three rows (D2, E2, and F2).

Step 2 Proceed with parameter estimation by choosing 'Tools/Data Analysis,' and then choosing 'Regression.' If 'Data Analysis' does not appear in your menu under 'Tools,' you may have to install it from the original Excel CD. Enter F2:F13 for the y values and

TABLE E.3. Reaction Rate Data as a Function of Partial Pressures

1	A pa	B pb	C rate	D ln(pa)	E ln(pb)	F ln(rate)
2	0.1044	0.1036	0.5051	-2.2595	-2.2672	-0.6830
3	0.1049	0.2871	0.6302	-2.2547	-1.2479	-0.4617
4	0.1030	0.5051	0.6342	-2.2730	-0.6830	-0.4554
5	0.2582	0.1507	1.3155	-1.3540	-1.8925	0.2742
6	0.2608	0.3100	1.5663	-1.3440	-1.1712	0.4487
7	0.2407	0.4669	1.5981	-1.4242	-0.7616	0.4688
8	0.3501	0.0922	1.6217	-1.0495	-2.3838	0.4835
9	0.3437	0.1944	1.8976	-1.0680	-1.6378	0.6406
10	0.3494	0.5389	2.1780	-1.0515	-0.6182	0.7784
11	0.4778	0.1017	2.1313	-0.7386	-2.2857	0.7567
12	0.4880	0.2580	2.7227	-0.7174	-1.3548	1.0016
13	0.5014	0.5037	3.1632	-0.6904	-0.6858	1.1516

D2:E13 for the x values. This tells the computer that you want the best line representing $\ln(\text{rate})$ depending linearly upon $\ln pa$ and $\ln pb$.

Step 3 There are several other options; choose residuals, residual plots, and line fit plots. These are all useful for evaluating the results. You can place the results on another sheet in the same Workbook or on the same sheet by specifying a location. Table E.4 shows part of the output. The best fit is for

$$a = 1.9608, \quad b = 0.9804, \quad c = 0.1896, \quad k = e^a = 7.105 \quad (\text{E.20})$$

The curve fit is then

$$\text{rate} = 7.105 p_A^{0.9804} p_B^{0.1896} \quad (\text{E.21})$$

The standard error gives an idea of how accurately the parameter is determined. If this value is a significant fraction of the parameter itself, the data is probably too scattered to be correlated in the way you have chosen. Also note the residuals, which are the errors in predicting the data. ‘Residuals’ should be both positive and negative with no trends. If the first five residuals were negative and the last six residuals were positive, it would indicate some systematic trend that is not accounted for by the formula used, such as Eq. (E.19). Figure E.5 shows one residual plot, indicating that the errors are scattered and both positive and negative with no trends. The r^2 value is 0.9956, which indicates a good correlation.

The rate can be calculated for each data point and compared with the data to obtain a residual. Those residuals are summed to obtain a least squares value for the correlation, as shown in Table E.5.

$$\sigma^2 = \sum_{i=1}^N \frac{[\text{rate}_i - \text{rate}(pa_i, pb_i)]^2}{N} \quad (\text{E.22})$$

TABLE E.4. Multiple Regression of Reaction Rate Data

(a) Summary output			
Regression Statistics			
Multiple R	0.99780809		
r ²	0.99562099		
Adjusted r ²	0.99452624		
Standard error	0.03856026		
Observations	11		
(b) ANOVA			
	d.f.	SS	
Regression	2	2.70450549	
Residual	8	0.01189515	
Total	10	2.71640064	
(c)			
	Coefficients	Standard Error	
Intercept	1.96082013	0.04656974	
- 2.2595256	0.98035818	0.0230596	
- 2.26721795	0.18956607	0.01989072	
(d) Residual output			
Observation	d	- 0.68299884	Residuals
1	- 0.48620447		0.02448642
2	- 0.3970332		- 0.05835772
3	0.27464776		- 0.00043094
4	0.42120076		0.02751539
5	0.42020902		0.04860641
6	0.48001181		0.00346318
7	0.60333283		0.0372571
8	0.81274183		- 0.03433481
9	0.80346734		- 0.04673522
10	1.00064879		0.00097525
11	1.15402923		- 0.00244505
(e)			
	p-Value	Lower 95%	Upper 95%
Intercept	1.1156 × 10 ⁻¹⁰	1.85343006	2.06821021
- 2.2595256	1.0329 × 10 ⁻¹⁰	0.92718261	1.03353376
- 2.26721795	1.2141 × 10 ⁻⁵	0.14369796	0.23543418

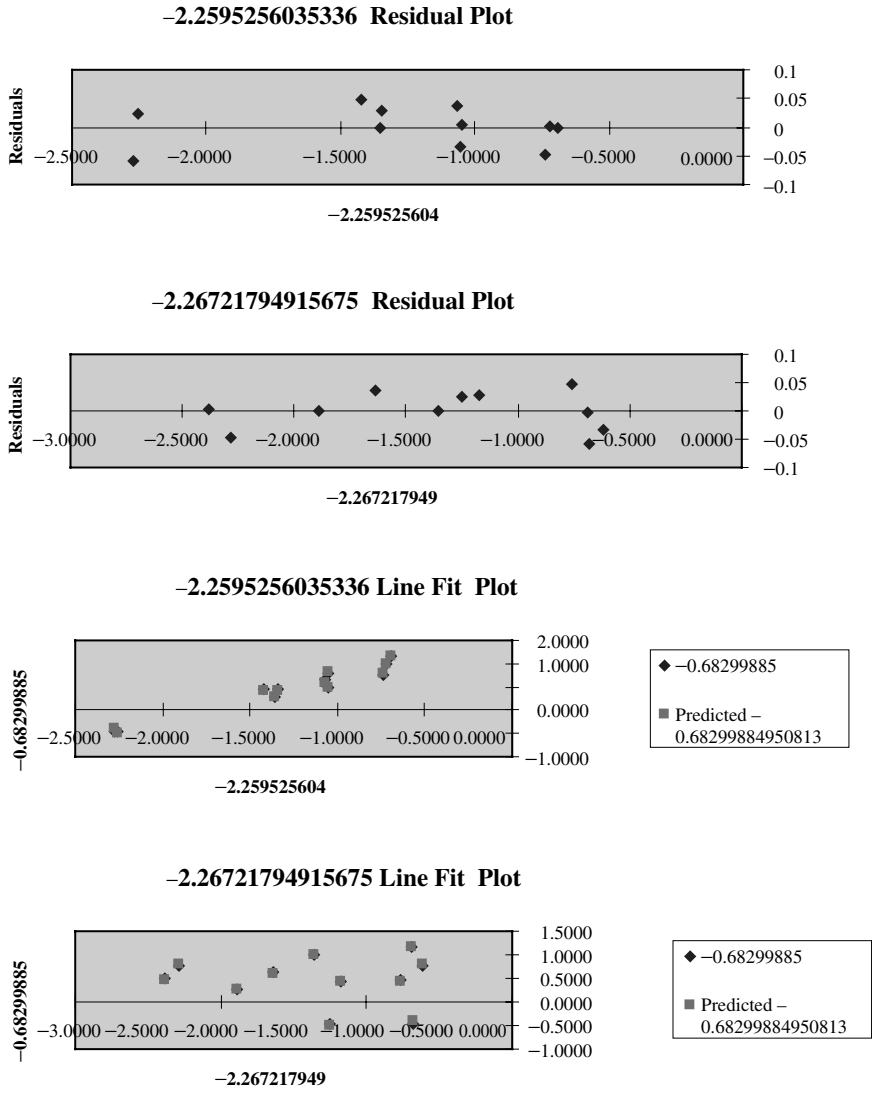


Figure E.5. Residual plot for reaction rate correlation.

You might think at this point the correlation is complete. It is not, though, because the data were transformed to make the parameter estimation problem linear. Thus, the statistics are in terms of the transformed problem. It is always a good idea to calculate the curve fit using the original variables. You can do this most conveniently by duplicating some columns so they are adjacent for plotting purposes, as shown in Table E.6.

Column L is the rate data, duplicated. Column M is the prediction using Eq. (E.21). Columns N and O are duplicated to permit a straight line, as shown in Figure E.6. This figure plots the predicted value, column M, vs the experimental one, column L.

Step 1 You can make the figure by choosing ‘Insert/Chart’ as before, plotting columns L and M using only data points (no line).

TABLE E.5. Least Squares Calculation for Reaction Rate Correlation Using Logarithmic Form

F	G	H	I	J
ln(rate)	Predicted ln(rate)	Predicted Rate using ln Form	Residual Using ln Form	Squares of Residuals
-0.6830	-0.6841	0.5045	-0.0006	3.1571×10^{-7}
-0.4617	-0.4862	0.6150	-0.0152	2.3238×10^{-4}
-0.4554	-0.3970	0.6723	0.0381	1.4525×10^{-3}
0.2742	0.2746	1.3161	0.0006	3.2152×10^{-7}
0.4487	0.4212	1.5238	-0.0425	1.8071×10^{-3}
0.4688	0.4202	1.5223	-0.0758	5.7487×10^{-3}
0.4835	0.4800	1.6161	-0.0056	3.1433×10^{-5}
0.6406	0.6033	1.8282	-0.0694	4.8161×10^{-3}
0.7784	0.8127	2.2541	0.0761	5.7881×10^{-3}
0.7567	0.8035	2.2333	0.1020	1.0398×10^{-2}
1.0016	1.0006	2.7200	-0.0027	7.0438×10^{-6}
1.1516	1.1540	3.1709	0.0077	5.9964×10^{-5}
(Sum of squares)/N				2.5285×10^{-3}

Step 2 If the correlation is perfect, all of the predicted values will lie on a straight line going through the origin with a slope of 1.0. To get that line, right-click (Ctrl-click on the Mac) on the diagram, choose ‘Source Data/Series,’ ‘add a series,’ and add the second curve (columns N and O).

Step 3 Then right-click on one of those data points, choose ‘Format data series,’ and choose ‘line’ and unselect ‘data points.’ The correlation is reasonably good, and the example is finished.

TABLE E.6. Collection of Data for Plotting

L	M	N	O
Rate Data	Predicted Rate	Rate Data	Rate Data
0.5051	0.5045	0.5051	0.5051
0.6302	0.6150	0.6302	0.6302
0.6342	0.6723	0.6342	0.6342
1.3155	1.3161	1.3155	1.3155
1.5663	1.5238	1.5663	1.5663
1.5981	1.5223	1.5981	1.5981
1.6217	1.6161	1.6217	1.6217
1.8976	1.8282	1.8976	1.8976
2.1780	2.2541	2.1780	2.1780
2.1313	2.2333	2.1313	2.1313
2.7227	2.7200	2.7227	2.7227
3.1632	3.1709	3.1632	3.1632

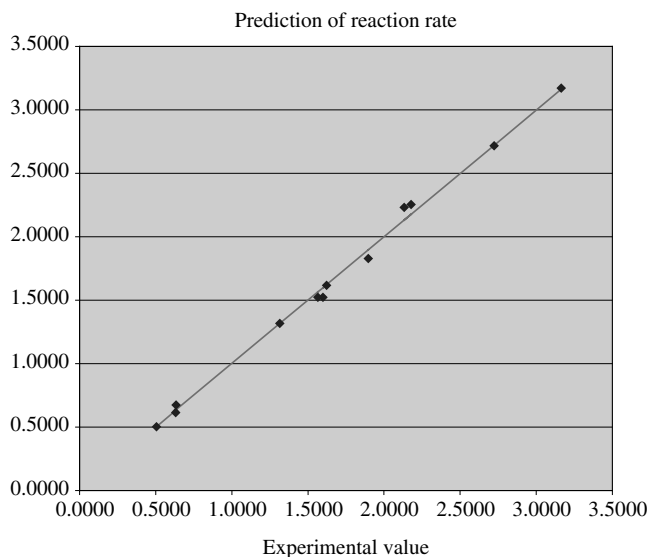


Figure E.6. Comparison of predicted reaction rate vs experimental value.

NONLINEAR REGRESSION

Nonlinear regression is a curve fit in which the unknown parameters enter into the problem in a nonlinear way. Nonlinear regression is much more difficult (for the computer), so it is best to always try to manipulate your model into a form that is linear. Sometimes that is not possible, and then nonlinear regression must be used. You need to be aware, though, that the methods described here do not always work. Nonlinear regression uses techniques borrowed from the field of optimization, and it is difficult to construct a method that works every single time for every problem.

To use nonlinear regression, you minimize Eq. (E.3) with respect to the unknown parameters. Polynomial and multiple regression do this too (behind the scenes), but for nonlinear curve fits it is necessary to use functions such as Solver in Excel and `fminsearch` in MATLAB. This is demonstrated using the same example given above for multiple regression.

Nonlinear Regression Using Excel

Step 1 Place the data on a new sheet in the Workbook, columns A, B, and C from Table E.3, as reproduced in Table E.7.

Step 2 Select some cells for the parameters k , n , and m . Arbitrary values are inserted – use your best guess, because that will possibly mean the difference between success and failure.

Step 3 In column D, calculate the value of rate using the parameters in C16:C18, the data in columns A and B, and the formula, Eq. (E.17).

Step 4 Make column E the difference between columns C and D, and then square the result and put it in column F.

TABLE E.7. Correlation of Rate Expression using ‘Solver’ in Excel; Initial Guess

1	A pa	B pb	C Measured Rate	D Calculated Rate	E Residual	F Residual Squared
2	0.1044	0.1036	0.5051	0.0108	0.4943	2.4432×10^{-1}
3	0.1049	0.2871	0.6302	0.0301	0.6001	3.6010×10^{-1}
4	0.1030	0.5051	0.6342	0.0520	0.5822	3.3893×10^{-1}
5	0.2582	0.1507	1.3155	0.0389	1.2766	1.6297×10^0
6	0.2608	0.3100	1.5663	0.0808	1.4855	2.2066×10^0
7	0.2407	0.4669	1.5981	0.1124	1.4857	2.2074×10^0
8	0.3501	0.0922	1.6217	0.0323	1.5894	2.5263×10^0
9	0.3437	0.1944	1.8976	0.0668	1.8308	3.3518×10^0
10	0.3494	0.5389	2.1780	0.1883	1.9897	3.9589×10^0
11	0.4778	0.1017	2.1313	0.0486	2.0827	4.3377×10^0
12	0.4880	0.2580	2.7227	0.1259	2.5968	6.7433×10^0
13	0.5014	0.5037	3.1632	0.2526	2.9106	8.4719×10^0
14						
15						
16		k	1		(Sum of	
17		n	1		squares/ N)	3.03139933
18		m	1			

Step 5 Sum Column F, divide by the number entries [COUNT(F2:F13)] to obtain the (sum of squares)/ N , Eq. (E.3).

Step 6 The goal is to minimize F17 by choosing values C16:C18. To do that, choose ‘Tools/Solver.’ You might have to add it to the Excel program if that was not done when the program was installed. A screen appears in which you insert F17 as the quantity to be affected, and choose ‘Min’ as the option. Then insert C16:C18 as the cells to be changed, and click ‘Solve.’ The results are shown in Table E.8.

The best correlation is

$$rate = 6.978 p_A^{0.9561} p_B^{0.1957} \quad (E.23)$$

These numbers are slightly different from those obtained using multiple regression. Multiple regression and nonlinear regression obtained the solution by minimizing two different objective functions. Notice that the (sum of squares)/ N is smaller when using nonlinear regression, because nonlinear regression minimizes that exact quantity. If nonlinear regression does not work, though, multiple linear regression is your only option.

Nonlinear Regression Using MATLAB

Step 1 Construct a function which calculates the (sum of the squares)/ N , using these commands.

```
function value=para(parameters)
k=parameters(1);
n=parameters(2);
```

TABLE E.8. Correlation of Rate Expression Using Solver in Excel

	A	B	C	D	E	F
1	pa	pb	Measured Rate	Calculated Rate	Residual	Residual Squared
2	0.1044	0.1036	0.5051	0.5162	-0.0111	1.2379×10^{-4}
3	0.1049	0.2871	0.6302	0.6331	-0.0029	8.2401×10^{-6}
4	0.1030	0.5051	0.6342	0.6948	-0.0606	3.6753×10^{-3}
5	0.2582	0.1507	1.3155	1.3203	-0.0048	2.2770×10^{-5}
6	0.2608	0.3100	1.5663	1.5351	0.0312	9.7617×10^{-4}
7	0.2407	0.4669	1.5981	1.5404	0.0577	3.3307×10^{-3}
8	0.3501	0.0922	1.6217	1.6045	0.0172	2.9704×10^{-4}
9	0.3437	0.1944	1.8976	1.8242	0.0734	5.3903×10^{-3}
10	0.3494	0.5389	2.1780	2.2623	-0.0843	7.1090×10^{-3}
11	0.4778	0.1017	2.1313	2.2018	-0.0705	4.9745×10^{-3}
12	0.4880	0.2580	2.7227	2.6957	0.0270	7.2844×10^{-4}
13	0.5014	0.5037	3.1632	3.1534	0.0098	9.5898×10^{-5}
14						
15						
16		<i>k</i>	6.97755309		(Sum of	
17		<i>n</i>	0.95605386		squares/ <i>N</i>)	0.00222768
18		<i>m</i>	0.195694			

```

m=parameters(3);
pa=[0.1044 0.1049 0.1030 ...];
pb=[0.1036 0.2871 0.5051 ...];
rate1=[0.5051 0.6302 0.6342 ...];
value=0.;
for i=1:12
    rate2(i)=k*(pa(i)^n)*(pb(i)^m);
    value=value+(rate1(i)-rate2(i))^2;
end
value=value/12;

```

Step 2 Test this function by removing the semi-colons. Then issue the following command

```
feval(@para,[2 3 4])
```

and calculate a few terms to check. In this case, Table E.7 can be used to provide values for checking.

Step 3 The minimum of value is found using the fminsearch function:

```

fminsearch(@para,[1 1 1])
ans=6.9776      0.9561      0.1957

```

These numbers are the same as obtained using Solver in Excel, as expected. The best correlation is Eq. (E.23).

Appendix F

MATHEMATICAL METHODS

The software packages Excel, MATLAB[®], Aspen Plus, and FEMLAB use mathematical methods that are pre-programmed. Engineers can solve chemical engineering problems without knowing the details of those mathematical methods, but some of you may be interested in the methods themselves. A brief overview is given here. There are numerous books that describe the methods in greater detail, including Finlayson (1980, 1992), Riggs (1994), Hanna and Sandall (1995), Constantinides and Mostoufi (1999), and Cutlip and Shacham (1999), as well as the *Mathematics* Section of *Perry's Chemical Engineers' Handbook* (Finlayson et al., 1997) and *Ullmann's Encyclopedia of Industrial Chemistry* (Finlayson, 1990). The subjects include:

- Nonlinear algebraic equations;
- Ordinary differential equations as initial value problems;
- Ordinary differential equations as boundary value problems;
- Partial differential equations in time and one space dimension;
- Partial differential equations in two or more spatial dimensions (and possibly time, too).

ALGEBRAIC EQUATIONS

A single nonlinear algebraic equation can be written as

$$f(x) = 0 \tag{F.1}$$

Iterative methods are used to solve such equations, and the successive substitution and Newton–Raphson methods are described here. Let x^k be the value of x at the k th iteration. The goal is to make x^k satisfy Eq. (F.1) as k increases.

Successive Substitution

In this method, the equation is multiplied by a constant, β , and an iterative equation is used.

$$x^{k+1} = x^k + \beta f(x^k) \quad (\text{F.2})$$

Under certain conditions, it can be proved that this method converges (Finlayson, 1980). The conditions require a bounded derivative df/dx and also a particular sign of β . The smaller the value of β the slower the convergence to the solution is, but also then it is more likely that a solution can be found. It is sometimes hard to predict a good value of β .

Newton–Raphson

In this method, a Taylor series is written for Eq. (F.1), evaluated at the x^k value:

$$f(x^{k+1}) = f(x^k) + \left. \frac{df}{dx} \right|_{x^k} (x^{k+1} - x^k) + \left. \frac{d^2f}{dx^2} \right|_{x^k} \frac{(x^{k+1} - x^k)^2}{2!} + \dots \quad (\text{F.3})$$

The terms higher than linear in $x^{k+1} - x^k$ are discarded, and the left-hand side is set to zero. You want the left-hand side to be zero, of course, and when you are close to the solution the higher-order terms are small. The iterations then proceed as follows:

$$x^{k+1} = x^k - \frac{f(x^k)}{df/dx|_{x^k}} \quad (\text{F.4})$$

Convergence proofs are available under certain conditions (Finlayson, 1980), and once the iterate value gets close to the solution, the convergence is very rapid. This method is generally better than the successive substitution method, except for special cases, but sometimes a good initial guess is required.

The Newton–Raphson method can be extended to sets of nonlinear equations. When the equations are

$$f_i(\{x_j\}) = 0, \quad i, j = 1, \dots, n \quad (\text{F.5})$$

the expansion is

$$f_i(\{x^{k+1}\}) = f_i(\{x^k\}) + \sum_{j=1}^n \left. \frac{\partial f_i}{\partial x_j} \right|_{x^k} (x_j^{k+1} - x_j^k) + \dots \quad (\text{F.6})$$

You set the left-hand side to zero, neglect the second-order and higher terms, and rearrange the equations as

$$\sum_{j=1}^n \left. \frac{\partial f_i}{\partial x_j} \right|_{x^k} (x_j^{k+1} - x_j^k) = -f_i(\{x^k\}) \quad (\text{F.7})$$

This is a linear problem, which can be written in the form

$$\sum_{j=1}^n A_{ij}^k (x_j^{k+1} - x_j^k) = -f_i(\{x^k\}), \quad A_{ij}^k = \left. \frac{\partial f_i}{\partial x_j} \right|_{x^k} \quad (\text{F.8})$$

This problem is now a set of linear equations, which is easily solved. The matrix $\mathbf{A}$ is called the Jacobian. Linear algebra is used to solve this set of equations (Finlayson, 1980).

$$\mathbf{A}^k(\mathbf{x}^{k+1} - \mathbf{x}^k) = -\mathbf{f}^k, \quad \mathbf{x}^{k+1} = \mathbf{x}^k - (\mathbf{A}^k)^{-1}\mathbf{f}^k \quad (\text{F.9})$$

After each step, the Jacobian and function f are reevaluated and the procedure is repeated. The Newton–Raphson method for sets of equations also converges quickly once one is close to the solution, but iterations can easily diverge unless special precautions are taken. The programs used in this book take those precautions, which makes them more reliable than programs you might write yourself.

The Newton–Raphson method requires that you differentiate the function with respect to all the variables. The *secant method* avoids that mathematical step and uses a numerical difference to calculate the derivative:

$$\left. \frac{\partial f_i}{\partial x_j} \right|_{\mathbf{x}^k} = \frac{f_i(x_j^k) - f_i(x_j^{k-1})}{x_j^k - x_j^{k-1}} \quad (\text{F.10})$$

The value of the functions is kept after each iteration in order to make this calculation. The Wegstein method is essentially a secant method, with some constraints on the parameters as described in Chapter 7. It is also possible to use a numerical derivative.

$$\left. \frac{\partial f_i}{\partial x_j} \right|_{\mathbf{x}^k} = \frac{f_i(x_j^k + \varepsilon) - f_i(x_j^k)}{\varepsilon} \quad (\text{F.11})$$

One of the advantages of FEMLAB is that it differentiates the equations symbolically, including any terms that you add to the problem, and this enhances convergence. The parametric solver in FEMLAB is possible because the Newton–Raphson method is being used. Consider Eq. (F.1) when it depends upon a parameter:

$$f[x(\alpha), \alpha] = 0 \quad (\text{F.12})$$

Differentiate Eq. (F.12) with respect to α :

$$\left. \frac{\partial f}{\partial x} \right|_{\alpha} \frac{dx}{d\alpha} + \left. \frac{\partial f}{\partial \alpha} \right|_x = 0 \quad (\text{F.13})$$

Once the solution to Eq. (F.12) is known, it is easy to find the derivative of the solution with respect to the parameter. Because the matrix inverse is already known, this calculation is fast. (Actually, a lower-upper (LU) decomposition would be done rather than an inverse, as it is twice as fast; see Finlayson, 1980.) When changing the parameter, a linear extrapolation about the current solution is then possible:

$$x(\alpha + \Delta\alpha) = x(\alpha) + \frac{dx}{d\alpha} \Delta\alpha \quad (\text{F.14})$$

This often gives an excellent guess of the solution for the next value of α . Sometimes even these methods do not solve the very hardest problems. In that case, a homotopy method may be necessary (Finlayson, 1990).

ORDINARY DIFFERENTIAL EQUATIONS AS INITIAL VALUE PROBLEMS

There are excellent methods for solving sets of first-order ordinary differential equations that are initial value problems (ODE-IVP). An example of a ODE-IVP is

$$\frac{dy}{dt} = f(t, y), \quad y(0) = y_0 \quad (\text{F.15})$$

Note that all the conditions are known at one time, $t = 0$. Thus it is possible to calculate the function on the right-hand side at $t = 0$ to obtain the derivative there. This makes the set of equations initial value problems. The equations are ordinary differential equations because there is only one independent variable. Any higher-order ordinary differential equation can be turned into a set of first-order ordinary differential equations; they are initial value problems if all the conditions are known at the same value of the independent variable [Finlayson, 1980, 1997 (p. 3-54), 1990 (Vol. B1, p. 1-55)]. The methods for initial value problems are explained here for a single equation; extension to multiple equations is straightforward. These methods are used when solving plug-flow reactors (Chapter 8) as well as time-dependent transport problems (Chapters 9–11).

Euler's Method

In numerical solutions of ODE-IVP, the solution y at the point t^n is represented by y^n . The simplest method is Euler's method, which is obtained by writing a difference expression for the derivative:

$$\frac{y^{n+1} - y^n}{\Delta t} = f(t^n, y^n) \quad (\text{F.16})$$

The first-order approximation of the derivative is not very accurate, so this method is not very accurate either. It is easy to program, though. Rearrange Eq. (F.16) as

$$y^{n+1} = y^n + \Delta t \, f(t^n, y^n) \quad (\text{F.17})$$

This equation can be applied repeatedly to go from $t = 0$ to any time, t . Indeed, Figure A.3 shows a Euler method applied to the problem:

$$\frac{dy}{dt} = -y, \quad y(0) = 1 \quad (\text{F.18})$$

The time step is taken as 0.01. The cells representing dy/dt are calculated using $-y$ in the same row. Thus, column C is taken as the negative of column D. Then the value of y in the

next row is taken as

$$B3 := B2 + \Delta t * C1. \quad (\text{F.19})$$

which is the equivalent of Eq. (F.17). These equations are copied down to obtain the whole solution shown in Figure A.4.

There are three major problems for the Euler method. First, the accuracy is poor, since the method is based upon Eq. (F.16), in which only a first-order difference expression is used. The errors in the method are proportional to Δt . Second, stability is difficult to achieve for many problems. The only way to have a stable Euler method is to use a small enough time step-size, but you may not know what value is sufficient. Furthermore, a value that is sufficient at the beginning may not be sufficient later on, and it may take an excessively long time to finish the computation. Third, to validate the results it is necessary to solve the problem at least twice, with different time-steps. The method can, however, be programmed in Excel, as Figures A.3 and A.4 demonstrate.

Runge–Kutta Methods

The Euler method can be improved by doing the calculation in two steps:

$$\begin{aligned} \bar{y} &= y^n + \frac{\Delta t}{2} f(t^n, y^n) \\ y^{n+1} &= y^n + \Delta t f\left(t^n + \frac{\Delta t}{2}, \bar{y}\right) \end{aligned} \quad (\text{F.20})$$

This is a second-order Runge–Kutta method (Finlayson, 1980), sometimes called the mid-point rule. The first step is an approximation of the solution halfway between the beginning and ending time, and the second step evaluates the right-hand side at that mid-point. The error goes as $(\Delta t)^2$, which is much smaller than that achieved with the Euler method. The second-order Runge–Kutta methods (there are several) also have a stability limitation.

The method used in MATLAB is a fourth-order Runge–Kutta method with a variable step-size. The Δt is changed to guarantee the accuracy of the calculation. The accuracy is estimated at each step, and the step-size is reduced to meet the specified accuracy. The method still has a stability limitation. The variable step size overcomes two of the problems – knowing what step size to use and using a small enough step-size to guarantee a specified accuracy. This makes the method very robust, which is why it is the workhorse in MATLAB. The method is called ode45 because the fourth-order method is chosen in such a way that it achieves fifth-order accuracy. It is based on an older program called RKF45 (Runge-Kutta-Feldberg) that was written many decades ago. More details can be found in *Perry's Chemical Engineers' Handbook* (Finlayson, 1997, p. 3-55) and *Ullmann's Encyclopedia of Industrial Chemistry* (Finlayson, 1990, Vol. B1, pp. 1-56–57).

In some problems, certain parameters vary quickly and others vary more slowly. For example, in packed-bed chemical reactors, the concentration can change quickly in time, but the temperature will not change rapidly because of the large heat capacity of the solid. Such problems are called ‘stiff’ [Finlayson, 1997 (p. 3-56), 1990 (Vol. B1, p. 1-60–61)]. The mathematical definition is that the eigenvalues of the Jacobian are widely separated, certainly by factors of thousands, perhaps by factors of millions. It

turns out that the stable step size of a Runge–Kutta method is limited by the largest eigenvalue (fastest responding variable), but the time for something to happen is controlled by the smallest eigenvalue (slowest responding variable). Thus, for stiff problems, many time steps would need to be taken with a small time step, and the calculations would be very slow.

Implicit methods have been developed to overcome this problem. The backward Euler method is

$$\frac{y^{n+1} - y^n}{\Delta t} = f(t^{n+1}, y^{n+1}), \quad \text{or} \quad y^{n+1} - \Delta t f(t^{n+1}, y^{n+1}) = y^n \quad (\text{F.21})$$

This method is still first-order, but it is stable for any step size. The only problem is that Eq. (F.21) is a nonlinear algebraic equation for y^{n+1} , and if there are several differential equations, there will be a set of nonlinear algebraic equations to be solved. This is not an easy task. Some higher-order methods are stable, too; the best are Gear's backward difference formulas (Gear, 1971; Finlayson, 1997, p. 3-56). Methods like this are also included in MATLAB, as `ode15s` and `ode23s`. These programs adjust the step size to maintain a specified accuracy. They also use sophisticated methods to solve the set of nonlinear equations. If a solution cannot be found, the step-size is reduced; then a solution is easier to find.

It is also possible to have *differential-algebraic equations*. There will be both differential and algebraic equations, with all the variables possibly occurring in all the equations. This essentially means that the variables determined from the algebraic equations are acting infinitely fast; this is merely a system of stiff equations taken to an extreme. Special methods exist for these problems, too. Because some of the variables are supposed to change immediately, they cannot be included in the error tests used to select the time step.

ORDINARY DIFFERENTIAL EQUATIONS AS BOUNDARY VALUE PROBLEMS

Transport problems involve second-order differential equations. While a single second-order differential equation can be reduced to two first-order differential equations, the problem is not an initial value problem because there are boundary conditions at two different values of the independent variable. This section describes methods for solving such problems, and illustrates them with the finite difference method as well as the finite element method.

The prototype problem is heat conduction in a slab or reaction and diffusion in a flat layer, as described in Chapter 9. Here the methods are demonstrated on a problem with reaction and diffusion in a flat layer with a first-order chemical reaction.

$$D \frac{d^2 c}{dx^2} = kc, \quad \text{with boundary conditions} \quad \frac{dc}{dx}(0) = 0, \quad c(R) = c_0 \quad (\text{F.22})$$

Finite Difference Method

This method is explained first because it is the oldest and simplest to explain, even though it is not the method embedded in FEMLAB. The first step is to divide the spatial region

into smaller regions by placing grid points on the line (see Figure F.1). Here the grid points are evenly spaced; although this is not necessary, it does make the presentation easier to grasp. The unknown function $y(x)$ is expanded in a Taylor series about the (unknown) value at a grid point:

$$y(x) = y(x_i) + \frac{dy}{dx}\bigg|_i (x - x_i) + \frac{d^2y}{dx^2}\bigg|_i \frac{(x - x_i)^2}{2!} + \dots \quad (\text{F.23})$$

This equation is evaluated for points on either side:

$$\begin{aligned} y(x_{i+1}) &= y(x_i) + \frac{dy}{dx}\bigg|_i (x_{i+1} - x_i) + \frac{d^2y}{dx^2}\bigg|_i \frac{(x_{i+1} - x_i)^2}{2!} + \dots \\ y(x_{i-1}) &= y(x_i) + \frac{dy}{dx}\bigg|_i (x_{i-1} - x_i) + \frac{d^2y}{dx^2}\bigg|_i \frac{(x_{i-1} - x_i)^2}{2!} + \dots \end{aligned} \quad (\text{F.24})$$

If the last two equations are subtracted, and rearranged, you can obtain the following approximation for the first derivative when the grid spacing is uniform:

$$\frac{dy}{dx}\bigg|_i = \frac{y(x_{i+1}) - y(x_{i-1}))}{2\Delta x} \quad \text{when } \Delta x = (x_{i+1} - x_i) = (x_i - x_{i-1}) \quad (\text{F.25})$$

This equation is second order, meaning that the errors decrease proportional to Δx^2 once Δx is small enough. The two equations can also be rearranged by neglecting the second-order terms, to obtain two other approximations for the first derivative:

$$\begin{aligned} \frac{dy}{dx}\bigg|_i &= \frac{y(x_{i+1}) - y(x_i)}{\Delta x} \\ \frac{dy}{dx}\bigg|_i &= \frac{y(x_i) - y(x_{i-1}))}{\Delta x} \end{aligned} \quad (\text{F.26})$$

These approximations are only first order, and errors are proportional to Δx when it is small enough. Another rearrangement gives an approximation for the second derivative. Take the sum of Eq. (F.24) and rearrange it to give:

$$\frac{d^2y}{dx^2}\bigg|_i = \frac{y(x_{i+1}) - 2y(x_i) + y(x_{i-1}))}{\Delta x^2} \quad (\text{F.27})$$

This approximation can be shown to be second order in Δx .

The idea in the finite difference method is that the differential equation, valid for all positions x , is replaced by a set of equations representing the equation only at the grid points, using the equations derived above. Thus, for problem Eq. (F.22), the equation at grid point i is:

$$D \frac{c_{i+1} - 2c_i + c_{i-1}}{\Delta x^2} = kc_i, \quad \text{where } c_i = c(x_i) \quad (\text{F.28})$$

Such an equation is written for each interior grid point shown in Figure F.1. At the boundaries, the finite difference method uses the boundary conditions, again with the finite difference representation of derivatives.

$$\left. \frac{dc}{dx} \right|_1 = \frac{c(x_2) - c(x_1)}{\Delta x}, \quad c_{n+1} = 1 \quad (\text{F.29})$$

It is possible to use a more accurate (second-order) derivative, at the cost of extra programming. Equations (F.28) and (F.29) represent a set of $n + 1$ algebraic equations in $n + 1$ unknowns. They are solved using linear algebra techniques. For this simple, one-dimensional problem, a special method is used for a tri-diagonal matrix. See Finlayson (1980) for complete details.

After the linear algebra problem is solved, the result is an approximation for $y(x_i)$ at each i th grid point. This can be plotted, and the points can be joined by straight lines, giving an approximation to the solution of the differential equation. More elaborate methods of interpolation are also possible. The solution to the finite difference equations, though, is an approximation to the solution to the differential equation. To assess its accuracy, you must refine the grid and solve another problem. By doubling the number of points once, and then twice, it is possible to see if the solution changes appreciably. If not, you have found an adequate approximation.

Finite Element Method

The finite element method proceeds in a slightly different way. The unknown solution $y(x)$ is expanded in a series of functions, called trial functions:

$$c(x) = \sum_{i=1}^{n+1} c_i N_i(x) \quad (\text{F.30})$$

The functions in the finite element method are constant, linear, or quadratic functions of position in a small region (called a finite element). The idea is explained for interpolation of a known function, Eq. (F.31):

$$z = \frac{11x^4 + x}{12} \quad (\text{F.31})$$

which is plotted in Figure F.2. The region $0 \leq x \leq 1$ is divided into finite elements using the same grid points shown in Figure F.1. If the approximations are taken as constant in each element, two interpolations are shown in Figure F.3. Even though only eight or 16 elements are used, the general shape of the function is clear.



Figure F.1. Finite difference grid.

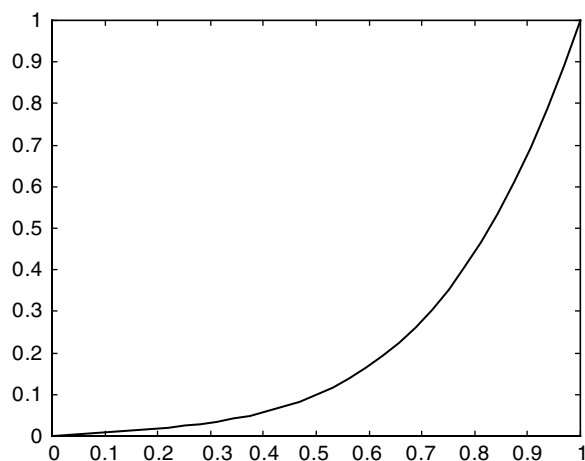


Figure F.2. Function $z = (11x^4 + x)/12$.

Next, the same mesh is used, but with linear trial functions. Interpolation with 4, 8, and 16 elements is shown in Figure F.4. This representation is much better than that in Figure F.3. Indeed, with 16 elements, the curve – while composed of straight-line segments – looks like a smooth curve. Whatever the trial function, the approximation is better when more elements are used, and mesh refinement is the easiest way to ensure that the approximation is accurate enough.

It is beyond the scope of this book to describe the method used to obtain the coefficients in Eq. (F.30), and how the boundary conditions are included, but complete details are available (Finlayson, 1972, 1980). There is a variety of books available about the finite element method. A book focusing on flow and convection/diffusion is by Gresho and Sani (1998). The representation of the second derivative is the same as given by the finite difference method, but the representation of the function is different. The finite

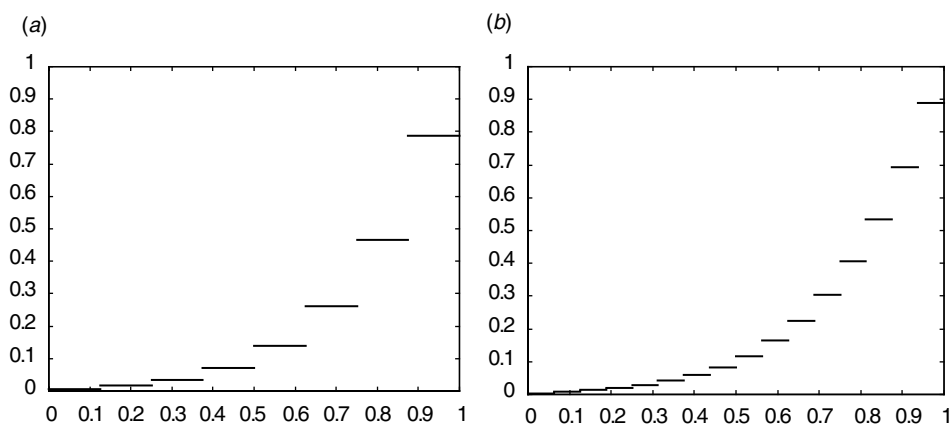


Figure F.3. Interpolation with eight (a) and 16 (b) constant finite elements.

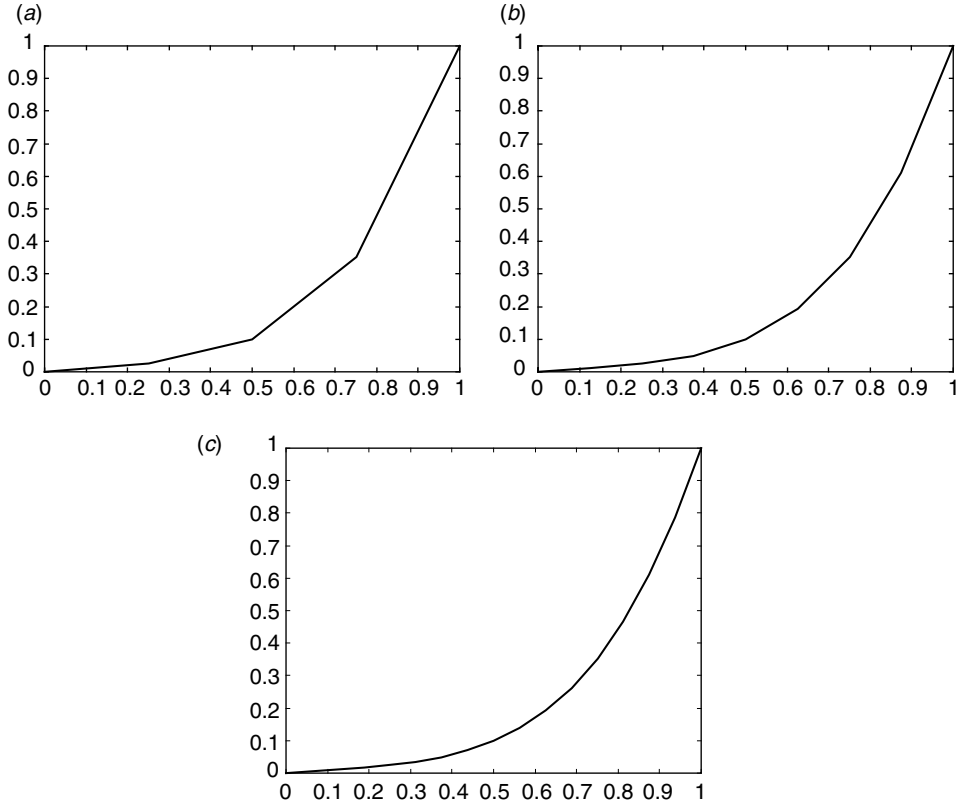


Figure F.4. Interpolation of $z = (11x^4 + x)/12$ with four (a), eight (b), and 16 (c) linear finite elements.

element method applied to Eq. (F.22) with linear trial functions is:

$$D \frac{c_{i+1} - 2c_i + c_{i-1}}{\Delta x^2} = \frac{k}{6}(c_{i+1} + 4c_i + c_{i-1}), \quad \text{where } c_i = c(x_i) \quad (\text{F.32})$$

If quadratic functions are used, there are more differences; in fact, the only time the representations are the same is for first and second derivatives that are second-order and when using linear trial functions in the finite element method.

The boundary conditions are applied in the finite element method in a different way than in the finite difference method, and then the linear algebra problem is solved to give the approximation of the solution. The solution is known at the grid points, which are the points between elements, and a form of the solution is known in between, either linear or quadratic in position as described here. (FEMLAB has available even higher order approximations.) The result is still an approximation to the solution of the differential equation, and the mesh must be refined and the procedure repeated until no further changes are noted in the approximation.

Initial Value Methods

Another option for solving boundary value problems is to treat them like initial value problems. Since a second-order equation can be reduced to two first-order equations, two initial conditions are necessary. One condition will be known at a boundary. Simply assume a value for the other dependent variable at that same boundary, integrate to the other side and check if the required boundary condition is satisfied. If not, change the initial value and repeat the integration. The success of this method depends upon the skill with which you program the iterations from one trial to the next.

Finite Difference Method in Excel

It is possible to program the finite difference method in Excel and use the 'Calculation' feature to handle the circular reference. Turn off the iteration, prepare the spreadsheet, and then turn the calculation back on. Whether this converges depends upon the initial guess.

To see how this is done, rearrange Eq. (F.28) to the following form:

$$c_i = \frac{c_{i+1} + c_{i-1}}{2 + k\Delta x^2/D} \quad (\text{F.33})$$

Identify the cells in the spreadsheet with the value of concentration at a node, i , as shown in Figure F.5. Put the value of $k\Delta x^2/D$ in cell A2. The equation for cell E1 is the analog of Eq. (F.33):

$$= (D1+F1) / (2+\$A\$2) \quad (\text{F.34})$$

Next copy this equation over a series of cells that correspond to the number of grid points. For the first and last cell (grid point) use a different equation, appropriate to the boundary condition. Then turn on the iteration feature to get the solution.

The next step is to re-solve the problem using more grid points, and a smaller Δx . This means you have to program the spreadsheet again, but it does give you an indication of whether the answer changes much as the mesh is refined, which is what insures you that you have solved the problem accurately enough.

The same thing can be done using the finite element method, but the details are more complicated and beyond the scope of this book.

PARTIAL DIFFERENTIAL EQUATIONS IN TIME AND ONE SPACE DIMENSION

Time-dependent transport problems lead to partial differential equations, but these are easily solved by combining the methods for initial value problems and boundary value

D1	E1	F1
c_{i-1}	c_i	c_{i+1}

Figure F.5. Finite difference method in Excel for boundary value problems.

problems. Take the heat conduction problem in Eq. (9.35).

$$\frac{\partial T}{\partial t} = \alpha \frac{\partial^2 T}{\partial x^2} \quad (\text{F.35})$$

Write the temperature as a function of position, x , and time, t , but consider only the positions where there is a grid point, x_i . The function $T(x_i, t)$ is a function of only one independent variable, t , and it can be written as $T_i(t)$. It can be differentiated in time as follows:

$$\frac{\partial}{\partial t}[T(x_i, t)] = \frac{dT_i(t)}{dt} \quad (\text{F.36})$$

Thus, the differential equation (F.35) defined at the x_i grid point using the finite difference method is:

$$\frac{dT_i}{dt} = \alpha \frac{T_{i+1} - 2T_i + T_{i-1}}{\Delta x^2} \quad (\text{F.37})$$

This equation is written for every grid point, and you have a set of initial value problems for the set of $\{T_i(t)\}$.

$$\begin{aligned} \frac{dT_2}{dt} &= \alpha \frac{T_3 - 2T_2 + T_1}{\Delta x^2}, & \frac{dT_3}{dt} &= \alpha \frac{T_4 - 2T_3 + T_2}{\Delta x^2} \\ \frac{dT_i}{dt} &= \alpha \frac{T_{i+1} - 2T_i + T_{i-1}}{\Delta x^2}, & \frac{dT_n}{dt} &= \alpha \frac{T_{n+1} - 2T_n + T_{n-1}}{\Delta x^2} \end{aligned} \quad (\text{F.38})$$

or

$$\frac{dT_i}{dt} = \sum_{j=1}^{n+1} A_{ij} T_j, \quad j = 2, \dots, n$$

The boundary conditions also need to be included, leading to a set of differential-algebraic equations as an initial value problem. Standard methods for ordinary differential equations as initial value problems can be applied if the boundary conditions are incorporated into the differential equations, which is possible when the boundary conditions are linear.

While the standard methods can be applied, the set of equations (F.38) is stiff, and becomes stiffer as the grid resolution becomes smaller. One eigenvalue of the matrix on the right-hand side of Eq. (F.38) approximates the slowest time response of the differential equation, but the largest eigenvalue is some constant divided by the grid spacing squared. Thus, as you refine the mesh, the points become closer together, and the eigenvalues get further apart, making the problem stiffer. Thus, when solving partial differential equations, stiff methods are used. In MATLAB, for example, `ode45` is replaced by `ode15s` or `ode23s`.

Another complication can be important: In a problem with convection, the rate of convection may be much faster than the rate of diffusion. The solution may be like a wave moving along in space. Such problems lead to difficulties when solved on a fixed mesh, and these difficulties apply to both finite element and finite difference methods. For example, the convective diffusion equation representing convection with velocity v

down a packed bed is:

$$\frac{\partial c}{\partial t} + u \frac{\partial c}{\partial x} = D \frac{\partial^2 c}{\partial x^2} \quad (\text{F.39})$$

The boundary and initial conditions are:

$$-D \frac{\partial c}{\partial x} + uc \Big|_{x=0} = uc_0, \quad \frac{\partial c}{\partial x} \Big|_{x=L} = 0, \quad c(x, 0) = c_{\text{in}}(x) \quad (\text{F.40})$$

If the inlet condition changes in time, then a wave moves downstream. This equation can be made nondimensional, and this is instructive here.

$$\frac{\partial c'}{\partial t'} + Pe \frac{\partial c'}{\partial x'} = \frac{\partial^2 c'}{\partial x'^2} \quad (\text{F.41})$$

The Peclet number is the rate of convection divided by the rate of diffusion.

$$Pe = \frac{u}{L} \cdot \frac{L^2}{D} = \frac{uL}{D} \quad (\text{F.42})$$

The finite element method applied to this equation gives the following equation at each grid point:

$$\frac{1}{6} \left[\frac{dc_{i+1}}{dt} + 4 \frac{dc_i}{dt} + \frac{dc_{i-1}}{dt} \right] + Pe \frac{c_{i+1} - c_{i-1}}{2\Delta x} = \frac{c_{i+1} - 2c_i + c_{i-1}}{\Delta x^2} \quad (\text{F.43})$$

It is possible to show that the finite element solution will oscillate wildly above and below the exact solution whenever there is a sharp change in the concentration and the mesh Peclet number is too large (Finlayson, 1992, p. 29). Thus you need

$$Pe\Delta x \leq 2 \quad (\text{F.44})$$

One way to fix this problem is to refine the mesh until Eq. (F.44) is satisfied, but for large problems that approach may require too many points (especially in 2D and 3D). Another way is to add some ‘fake’ diffusion to the problem. In this case, the Petrov–Galerkin method is

$$\frac{1}{6} \left[\frac{dc_{i+1}}{dt} + 4 \frac{dc_i}{dt} + \frac{dc_{i-1}}{dt} \right] + Pe \frac{c_{i+1} - c_{i-1}}{2\Delta x} = \left[1 + \frac{Pe\Delta x}{2} \right] \frac{c_{i+1} - 2c_i + c_{i-1}}{\Delta x^2} \quad (\text{F.45})$$

and this makes the oscillations smaller or eliminates them. The trick is to add enough ‘fake’ diffusion to dampen the oscillations but not so much as to change the solution. There are a number of other techniques for treating this difficulty, as reviewed by Finlayson (1992). In FEMLAB this is called streamline diffusion.

PARTIAL DIFFERENTIAL EQUATIONS IN TWO SPACE DIMENSIONS

You can solve problems in two space dimensions using the same methods as applied in one dimension, except that the details become more complicated. The problems are stiffer, and implicit methods are usually required. The problem of unwanted oscillations is more severe because added diffusion may smooth the solution in the flow direction, as desired, but it may also diffuse it in the direction perpendicular to flow, which is not desired. Careful treatment, though, allows for streamline diffusion; that is, the extra diffusion is only added in the flow direction. These details have been handled by FEMLAB, which frees you to concentrate on the physics and the problem you want to solve.

The trial functions are usually defined on an element. A square element in two dimensions with bilinear trial functions has four trial functions per element. The element covers $0 \leq u \leq 1$, $0 \leq v \leq 1$, and the trial functions are:

$$\begin{aligned} N_1 &= (1-u)(1-v), & N_2 &= u(1-v) \\ N_3 &= uv, & N_4 &= (1-u)v \end{aligned} \quad (\text{F.46})$$

The domain is covered with elements to represent a function on the entire domain. Consider the function

$$z = x^2 \exp(y - 0.5) \quad (\text{F.47})$$

which is plotted in Figure F.6. Plots of the finite element interpolation on $4 \times 4 = 16$ bilinear elements or $8 \times 8 = 64$ bilinear elements are shown in Figure F.7. Sixty-four elements does a very good job of representing the function. Either variational principles or the Galerkin method are used to determine these trial functions to represent the solution to differential equations. The method is beyond the scope of this book and is covered by Finlayson (1972, 1980).

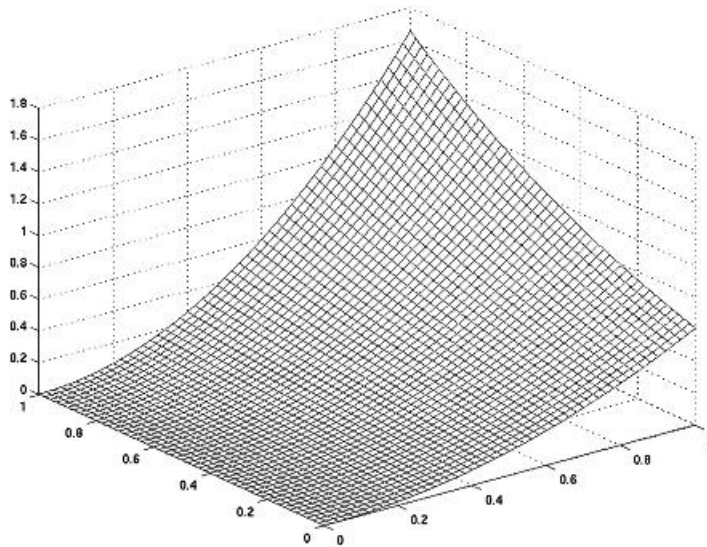


Figure F.6. Function $z = x^2 \exp(y - 0.5)$.

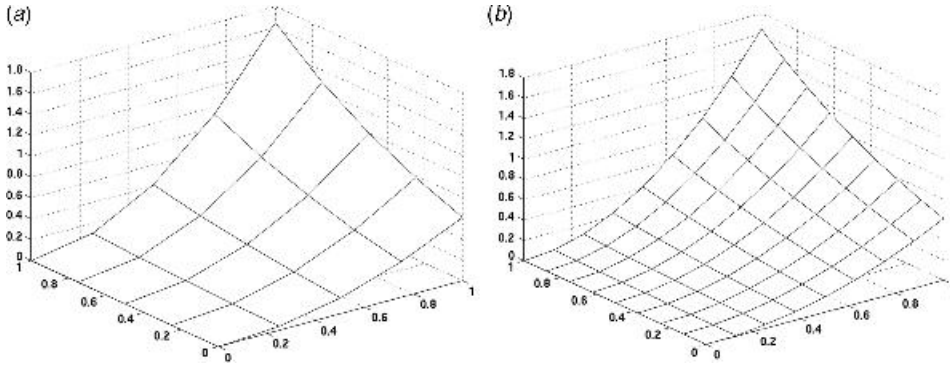


Figure F.7. Interpolation of $z = x^2 \exp(y - 0.5)$ with 16 (a) and 64 (b) bilinear finite elements.

Finite Difference Method for Elliptic Equations

The finite difference method is easy to program in Excel, provided the equation is not too complicated. Solving the problem in Excel does require solving it more than once, though, to assess the accuracy. As an example, take the heat conduction equation with a heat generation term, Eq. (F.48):

$$k \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right) = Q \quad (\text{F.48})$$

The finite difference form of this equation is

$$\frac{T_{i+1,j} - 2T_{i,j} + T_{i-1,j}}{\Delta x^2} + \frac{T_{i,j+1} - 2T_{i,j} + T_{i,j-1}}{\Delta y^2} = \frac{Q}{k} \quad (\text{F.49})$$

where $T_{i,j}$ is the temperature at the i th location in the x -direction and the j th location in the y -direction. Take the case when $\Delta x = \Delta y$ and rearrange as shown in Eq. (F.50):

$$T_{i,j} = \frac{T_{i+1,j} + T_{i-1,j} + T_{i,j+1} + T_{i,j-1}}{4} - \Delta x^2 \frac{Q}{k} \quad (\text{F.50})$$

The spreadsheet is arranged as shown in Figure F.8. When the value of $\Delta x^2 Q/k$ is placed in cell A2 the equation for cell E5 is

$$= (D5 + F5 + E4 + E6) / 4 - \$A\$2 \quad (\text{F.51})$$

	E4-T _{i,j-1}	
D5-T _{i-1,j}	E5-T _{i,j}	F5-T _{i+1,j}
	E6-T _{i,j+1}	

Figure F.8. Finite difference method in Excel for elliptic boundary value problem.

You copy this equation for every internal grid point, set the boundary equations, and turn on the iteration feature to obtaining the solution. Then you have to do it again with a finer mesh to assess the accuracy. If the heat generation term depends upon temperature, it is easy to include that complication just by inserting the formula in place of Q .

SUMMARY

This appendix gives a brief overview of the methods that are preprogrammed in the software packages Excel, MATLAB, Aspen Plus, and FEMLAB. The interested reader may pursue the references for more specific details.

REFERENCES

- Abramowitz, M. A., Stegun, I. A. (eds), *Handbook of Mathematical Functions*. National Bureau of Standards: Gathessburg, MD, 1964.
- Aguiló, A., Penrod, J. D., Acetaldehyde. In *Encyclopedia of Chemical Processing and Design*, Vol. 1, McKetta, J. J. (ed.). Marcel Dekker: New York, 1976.
- Bird, R. B., Armstrong, R. C., Hassager, O., *Dynamics of Polymeric Liquids*. Wiley-Interscience: New York, 1987.
- Bird, R. B., Stewart, W. E., Lightfoot, E. N., *Transport Phenomena*, 2nd edn. Wiley: New York, 2002.
- Broyden, C. G., A class of methods for solving nonlinear simultaneous equations. *Math. Comput.* 1965; 19, 577.
- Broyden, C. G., The convergence of a class of double-rank minimization algorithms. *J. Inst. Math. Applic.* 1970; 6, p. 76–90.
- Chang, M. W., Finlayson, B. A., On the proper boundary condition for the thermal entry problem. *Int. J. Numer. Meth. Engng.* 1980; 15, 935–942.
- Chow, A., Lab-on-a-Chip: opportunities for chemical engineering. *AIChE J.* 2002; 48, 1590–1595.
- Constantinides, A., Mostoufi, N., *Numerical Methods for Chemical Engineers with MATLAB Applications*. Prentice-Hall: Englewood Cliffs, NJ, 1999.
- Cowfer, J. A., Gorensek, M. B., Vinyl chloride. In *Encyclopedia of Chemical Technology*, Vol. 24, Kroschwitz, J. I. (Exec. Ed.), Howe-Grant, M. (ed.), 4th edn. Wiley: New York, 1997.
- Cutlip, M. B., Shacham, M., *Problem Solving in Chemical Engineering with Numerical Methods*. Prentice-Hall: Englewood Cliffs, NJ, 1999.
- Dagan, Z., Weinbaum, S., and Pfeffer, R., An infinite-series solution for the creeping motion through an orifice of finite length. *J. Fluid Mech.* 1982; 115, 505–523.
- Deen, W. M., *Analysis of Transport Phenomena*. Oxford University Press: Oxford, 1998.
- De Leeuw, M. L., *Z. Phys. Chem.* 1911; 77, 2841 [as reported by Hirata et al. (1975).]

- Denbigh, K., *The Principles of Chemical Equilibrium*, 3rd edn. Cambridge University Press: Cambridge, 1971.
- Eduljee, H. E., *Hydro. Proc.* 1975; 54(9), 120.
- Ellis, S. R. M., Garbett, R. O., *Ind. Eng. Chem.* 1960; 52, 385 [as reported by Hirata et al. (1975).]
- Fenske, M. R., Fractionation of straight-run Pennsylvania gasoline. *Ind. Eng. Chem.* 1932; 24, 482–485.
- Finlayson, B. A., *The Method of Weighted Residuals and Variational Principles*. Academic Press: London, 1972.
- Finlayson, B. A., *Numerical Methods for Problems with Moving Fronts*. Ravenna Park, 1992.
- Finlayson, B. A., *Nonlinear Analysis in Chemical Engineering*. McGraw-Hill: New York, 1980. [reprinted, Ravenna Park: Seattle, WA, 2003.]
- Finlayson, B. A., Mathematics in chemical engineering. In *Ullmann's Encyclopedia of Industrial Chemistry*, 5th edn, Vol. B1, Elvers, B., Hawkins, S., Schulz, G. (eds). VCH: New York, 1990.
- Finlayson, B. A., Davis, J. F., Westerberg, A. W., Yamoshita, Y., Mathematics. In *Perry's Chemical Engineers' Handbook*, Perry, R. H., Green, D. W. (eds), 7th edn, Section 3. McGraw-Hill: New York, 1997.
- Fogler, H. S., *Elements of Chemical Reaction Engineering*, 4th edn. Prentice Hall: Englewood Cliffs, NJ, 2005.
- Freemantle, M., Finding the best conditions rapidly. *C&E News*, February 2005; 11.
- Gear, C. W., *Numerical Initial Value Problems in Ordinary Differential Equations*. Prentice-Hall: Englewood Cliffs, NJ, 1971.
- Gilliland, E. R., Multicomponent rectification: estimation of the number of theoretical plates as a function of the reflux ratio. *Ind. Engng. Chem.* 1940; 32, 1220–1223.
- Gokhale, S. V., Tayal, R. K., Jayaraman, V. K., Kulkarni, B. D., Microchannel reactors: applications and use in process development. *Int. J. Chem. React. Engng.* 2005; 3, Review R2.
- Gresho, P. M., Sani, R. L., *Incompressible Flow and the Finite Element Method*. Wiley: New York, 1998.
- Gresho, P. M., Sani, R. L., Engleman, M. S., *Incompressible Flow and the Finite Element Method. Vol. 2: Isothermal Laminar Flow*. Wiley: New York, 1998.
- Hanna, O. T., Sandall, O. C., *Computational Methods in Chemical Engineering*. Prentice-Hall: Englewood Cliffs, NJ, 1995.
- Hasegawa, T., Suganuma, M., Watanabe, H., Anomaly of excess pressure drops of the flow through very small orifices. *Phys. Fluids.* 1987; 9, 1–3.
- Hatch, A., Kamholz, A. E., Hawkins, K. R., Munson, M. S., Schilling, E. A., Weigl, B. H., Yager, P., A rapid diffusion immunoassay in a T-sensor. *Nat. Biotechnol.* 2001; 19, 461–466.
- Hellinckx, L., Grootjans, J., Van den Bosch, B., Stability analysis of catalyst particle through orthogonal collocation. *Chem. Engng. Sci.* 1972; 27, 644–647.
- Henry, C. M., Go with the flow. *C&E News* 28 March 2005; 57–59.
- Higham, D. J., Higham, N. J., *Matlab Guide*, 2nd edn. SIAM: Philadelphia, PA, 2005.
- Hirata, M., Ohe, S., Nagahama, K., *Computer Aided Data Book of Vapor–Liquid Equilibria*. Kodansha and Elsevier: Tokyo and New York, 1975.
- Jensen, K. F., Microchemical systems: status, challenges, and opportunities. *AIChE J.* 1999; 45, 2051–2054.
- King, C. J., *Separation Processes*, 2nd edn. McGraw-Hill: New York, 1980.
- Kusmanto, F., Jacobsen, E. L., Finlayson, B. A., Applicability of continuum mechanics to pressure drop in small orifices. *Phys. Fluids* 2004; 16, 4129–4134.
- Larsen, R. W., *Engineering with Excel*. Prentice Hall: Englewood Cliffs, NJ, 2002.

- Marsh, H. D., Severin, M., Schuster, H. G., Steam reforming. In *Encyclopedia of Chemical Processing and Design*, Vol. 47, McKetta, J. J. (ed.). Marcel Dekker: New York, 1994.
- McKetta, J. J. (ed.), *Encyclopedia of Chemical Processing and Design*. Marcel Dekker: New York, 1976–1994.
- Middleman, S., *Fundamentals of Polymer Processing*. McGraw-Hill: New York, 1977.
- Murti, P. S., Van Winkle, M., *Chem. Engng. Data Ser.* 1959; 3, 72, [as reported by Hirata et al. (1975).]
- Neils, C., Tyree, Z., Finlayson, B., Folch, A., Combinatorial mixing of microfluidic streams. *Lab-on-a-Chip* 2004; 4, 342–350.
- Orazem, M. and Newman, J., Primary current distribution and resistance of a slotted-electrode cell. *J. Electrochem. Soc.* 1984; 131, 2857–2861.
- Otani, S., Matsuoka, S., Sato, M., ‘Tatoray’ process – a new transalkylation process of aromatics developed by Toray. *Jap. Chem. Q.* 1968; 4, 16–18.
- Perry, R. H., Green, D. W. (eds), *Perry’s Chemical Engineers’ Handbook*, 7th edn. McGraw-Hill: New York, 1997.
- Pignet, T., Schmidt, L. D., Kinetics of NH_3 oxidation of Pt, Rh, and Pd. *J. Catal.* 1975; 40, 212–225.
- Press, W. H., Flannery, E. P., Teukolsky, S. A., Vetterling, W. T., *Numerical Recipes*. Cambridge University Press: Cambridge, 1986.
- Reid, R. C., Prausnitz, J. M., Sherwood, T. K. *The Properties of Gases and Liquids*, 3rd edn. McGraw-Hill: New York, 1977.
- Rhee, H. K., Aris, R., Amundson, N. R., *First-Order Partial Differential Equations*, Vol. I, *Theory and Application of Single Equations*. Prentice-Hall: Englewood Cliffs, NJ, 1986.
- Riggs, J. B., *An Introduction to Numerical Methods for Chemical Engineers*, 2nd edn. Texas Tech University Press: Texas, 1994.
- Schmidt, L. D., *The Engineering of Chemical Reactions*. Oxford University Press: Oxford, 2004.
- Seader, J. D., Siirola, J. J., Barnicki, S. D., Distillation. In *Perry’s Chemical Engineers’ Handbook*, Perry, R. H., Green, D. W. (eds), 7th edn, Section 13. McGraw-Hill: New York, 1997.
- Shuler, M. L., Immobilized biocatalysts and reactors. *Chemical Engineering Education in a Changing Environment*, Sandler, S. I., Finlayson, B. A. (eds), pp. 37–56, AIChE: New York, 1988.
- Stephen, H., Stephen, T. (eds), *Solubilities of Inorganic and Organic Compounds*, Vol. 1, *Binary Systems*, Part 1. Pergamon Press: New York, 1963.
- Stephenson, R. M., *Introduction to the Chemical Process Industries*. Reinhold: New York, 1966.
- Stirling, R. G., Ethyl chloride. In *Encyclopedia of Chemical Processing and Design*, Vol. 20, McKetta, J. J. (ed.). Marcel Dekker: New York, 1984.
- Stone, H. A., Kim, S., Microfluidics: basic issues, applications, and challenges. *AIChE J.* 2001; 47, 1250–1254.
- Underwood, A. J. V., The theory and practice of testing stills. *Trans. Inst. Chem. Engng.* (London) 1932; 10, 112–158.
- Wegstein, J. H., Accelerating convergence of iterative processes. *Commun. Assoc. Comput. Mach.* 1958; 1(6), 9–13.
- Weigl, B. H., Yager, P., Microfluidic diffusion-based separation and detection. *Science* 1999; 283, 346–347.
- Winn, F. W., New relative volatility method for distillation calculations. *Petrol. Refin.* 37(5), 216–218.
- Young, L. C., Finlayson, B. A., Axial dispersion in non-isothermal packed bed chemical reactors. *IEC Fund.* 1973; 12, 412–422.
- Zimmerman, W. B. J., *Process Modelling and Simulation with Finite Element Methods*. World Scientific: Singapore, 2004.

AUTHOR INDEX

Abramowitz, M. A., 297, 323

Aguiló, A., 71, 323

Amundson, N. R., 325

Aris, R., 325

Armstrong, R. C., 323

Barnicki, S. D., 325

Bird, R. B., 147, 149, 151, 178, 323

Broyden, C. G., 100, 323

Chang, M. W., 227, 323

Chow, A., 207, 323

Constantinides, A., 307, 323

Cowfer, J. A., 69, 323

Cutlip, M. B., 307, 323

Dagan, Z., 197, 198, 323

Davis, J. F., 324

Deen, W. M., 190, 323

De Leeuw, M. L., 40, 323

Denbigh, K., 42, 324

Eduljee, H. E., 78, 324

Ellis, S. R. M., 40, 324

Elvers, B., 324

Engleman, M. S., 324

Fenske, M. R., 77, 324

Finlayson, B. A., 121, 164, 166, 167, 168, 199,
227, 307, 308, 309, 310, 311, 312, 314,
315, 319, 320, 323, 324, 325

Flannery, E. P., 325

Fogler, H. S., 118, 143, 324

Folch, A., 325

Freemantle, M., 207, 324

Garbett, R. O., 40, 324

Gear, C. W., 312, 324

Gilliland, E. R., 78, 324

Gokhale, S. V., 207, 324

Gorensek, M. B., 69, 323

Green, D. W., 8, 78, 100, 191, 204, 324, 325

Gresho, P. M., 177, 315, 324

Grootjans, J., 324

Hanna, O. T., 307, 324

Hasegawa, T., 197, 198, 324

Hassager, O., 323

Hatch, A., 214, 324

Hawkins, K. R., 324

Hawkins, S., 324

Hellinckx, L., 170, 324

Henry, C. M., 207, 324

Higham, D. J., 239, 324

Higham, N. J., 239, 324

Hirata, M., 323, 324, 325

Howe-Grant, M., 323

Jacobsen, E. L., 324
Jayaraman, V. K., 324
Jensen, K. F., 207, 324

Kamholz, A. E., 324
Kim, S., 207, 325
King, C. J., 78, 324
Kroschwitz, J. I., 323
Kulkarni, B. D., 324
Kusmanto, F., 194, 324

Larsen, R. W., 229, 324
Lightfoot, E. N., 323

McKetta, J. J., 323, 324, 325
Marsh, H. D., 54, 324
Matsuoka, S., 325
Middleman, S., 172, 325
Mostoufi, N., 307, 323
Munson, M. S., 324
Murti, P. S., 40, 325

Nagahama, K., 323, 324, 325
Neils, C., 199, 222, 325
Newman, J., 224, 325

Ohe, S., 323, 324, 325
Orazem, M., 224, 325
Otani, S., 67, 325

Penrod, J. D., 71, 323
Perry, R. H., 8, 78, 100, 191, 204, 324, 325
Pfeffer, R., 197, 198
Pignet, T., 143, 325
Prausnitz, J. M., 325
Press, W. H., 294, 325

Reid, R. C., 39, 325
Rhee, H. K., 164, 325
Riggs, J. B., 307, 325

Sandall, O. C., 307, 324

Sandler, S. I., 325
Sani, R. L., 315, 324
Sato, M., 325
Schilling, E. A., 324
Schmidt, L. D., 143, 325
Schultz, G., 324
Schuster, H. G., 324
Seader, J. D., 88, 325
Severin, M., 324
Shacham, M., 307, 323
Sherwood, T. K., 325
Shuler, M. L., 143, 144, 325
Sirola, J. J., 325
Stegan, I. A., 297, 323
Stephen, H., 95, 96, 325
Stephen, T., 95, 96, 325
Stephenson, R. M., 95, 325
Stewart, W. E., 232
Stirling, R. G., 68, 325
Stone, H. A., 207, 325
Suganuma, M., 324

Talal, R. K., 324
Teukolsky, S. A., 325
Tyree, Z., 199, 222, 325

Underwood, A. J. V., 77, 325

Van den Bosch, B., 324
Van Winkle, M., 40, 325
Vetterling, W. T., 325

Watanabe, H., 324
Wegstein, J. H., 100, 103, 325
Weigl, B. H., 214, 324, 325
Weinbaum, S., 197, 198
Westerberg, A. W., 324
Winn, F. W., 78, 325

Yager, P., 214, 324, 325
Yamoshita, Y., 324
Young, L. C., 121, 325

Zimmerman, W. B. J., 270, 325

CHEMICAL INDEX

Acetaldehyde, 40, 70, 71
Acetic acid, 71
Acetone, 82, 84, 85
Acetylene, 69, 70
Air, 70, 82, 84, 85
Ammonia, 22, 39, 41, 53, 62–65, 91, 95, 96,
99, 105, 143, 144

Benzene, 67, 68, 108
i-Butane, 38, 58, 60, 76–78, 81, 83, 86, 88
n-Butane, 9, 12, 16, 28, 34, 37, 38, 58, 60, 76,
81, 83, 86, 88
Butanol, 71

Carbon dioxide, 18, 22, 38, 39, 43–45, 51, 53,
63, 69, 88, 91, 94, 96, 99
Carbon monoxide, 18, 22, 23, 39, 43–45,
51–54, 145, 146, 169, 228
Chlorine, 68, 69

Ethane, 38, 53, 68, 69, 88, 111
Ethanol, *see* ethyl alcohol
Ethyl acetate, 40, 71
Ethyl alcohol, 36, 37, 40, 70, 71, 88, 109
Ethyl chloride, 53, 68, 69, 108
Ethylene, 53, 68–70, 111
Ethylenedichloride, 69, 70, 108, 109

Hexane, 38, 88
Hydrogen, 18, 22, 23, 39, 41, 43–45, 51–54,
62–65, 67, 68, 71, 91, 94–97, 99, 108,
111
Hydrogen chloride, 53, 68–70, 108

Methane, 38, 54, 67, 68, 88, 108
Methanol, 23, 39, 52

Nitric oxide, 143
Nitrogen, 22, 38, 39, 41, 53, 62–65, 91, 94,
95, 99

n-Octane, 28, 34
Oxygen, 53, 69, 70, 144

i-Pentane, 38, 58, 60, 76, 81, 83, 86, 88
n-Pentane, 28, 34, 37, 38, 58, 60, 76, 81, 83,
86, 88
Polyethylene, 111
Polyvinylchloride, (PVC), 69
Propane, 28, 34, 37, 38, 58, 60, 76–78, 81, 83,
86, 88

Sulfur dioxide, 53, 121
Sulfur trioxide, 53, 121

Toluene, 67, 68

1,1,2-Trichloroethane, 69, 108

Tri-methyl benzene, 67, 68

Vinyl chloride, 69, 70, 108

Water, 18, 22, 36, 37, 39, 40, 43–45, 51, 53,
54, 70, 82, 84, 85, 88, 143, 204, 214

Xylene, 67, 68

COMPUTER COMMAND INDEX

- Aspen Plus blocks
 - compressor, 19, 91, 93, 100
 - DSTWU, 76, 85, 87, 88, 108, 109
 - equation-oriented (EO) method, 100
 - Flash2, 31, 39, 91
 - heat exchanger, 94, 101
 - mixer, 96
 - RadFrac, 81–85, 87, 88
 - as absorption column, 82
 - RGibbs, 52, 91, 95, 105
 - Rstoic, 108
 - separator, 96
 - splitter, 96
 - table of, 90
- Aspen Plus commands
 - block, 20, 33
 - parameters for, 264
 - place and delete, 261
 - change conditions, 266
 - choosing a property method, 268
 - choosing an activity coefficient method, 268
 - component specification, 16, 20, 32, 35, 92, 262
 - connect units, 262
 - edit/delete, 262
 - control panel, 97, 265
 - convergence, 97, 98, 265
 - data browser, 260
 - data entry, 262
 - export report, summary, input, messages, 259, 265, 266
 - eye glasses, 260
 - flowsheet, 264, 266, 267
 - global, 91
 - help, 268
 - material streams, 19, 32, 260
 - maximize screen, 259
 - model library, 90, 260
 - profiles, 81
 - property analysis, 16, 35
 - property specification, 17, 20, 32, 35, 74
 - reinitialize, 264
 - report, 259, 266–267
 - results, 20, 21
 - run, 264
 - save, 267
 - setup, 260
 - streams, 20, 33, 263
 - stream table, 264, 266
 - thermodynamic model, 262
 - see also* choosing . . .
 - transfer to word processing program, 265
 - units, 260
 - see also* optimization
- Excel commands
 - address, absolute, 231
 - alignment, 230
 - border, 230
 - calculation, 9, 232, 233, 322
 - see* circular reference

cell content
 conditional alarm, 230
 equation, 230
 names, 230
 number, 229
 word, 229
 chart, 231, 234
 delete label, 234
 display equation, 295
 display r^2 , 295
 multiple lines, 234
 circular reference, 63, 66, 67
 color, 230
 column, 229, 230
 insert, 231
 comment, 230
 contour plot, 234–235
 copy, 231
 documentation, 235
 equation, 231
 export text, 236
 follow up, 237
 function, 237
 goal seek, 8, 10, 29, 44, 63, 66, 135, 232
 help, 238
 hide, 230
 hyperlinks, 231
 import text, 236
 INTERCEPT, 295
 iteration, *see* calculation
 linear correlation coefficient, 297, 300
 MINVERSE, 237
 MMULT, 237
 parameter estimation
 multiple regression, 299–303
 nonlinear regression, 304–305
 polynomial, 297–298
 straight line, 294–295
 paste, 232
 special, 233
 pattern, 230
 picture, 231
 printing, 231
 protection, 237
 remove split, 231
 residual, 300, 302
 row, 229, 230
 insert, 231
 RSQ, 295
 SLOPE, 294
 solver, 9, 63, 66, 135, 232, 305
 split screen, 231
 spreadsheet, large, 232
 standard error, 300
 text box, 230
 trendline, 295, 297
 unhide, 230

variables, named, 230
 xy-scatter plot, 234

FEMLAB commands

application mode name, 272
 artificial diffusion, 165, 166, 168
 axes, *see* grid settings
 axial symmetry, 194, 200, 210, 270
 axisymmetric geometry, 179
 Boolean operator, 168, 281
 boundary condition, 199–201, 209–211
 boundary plot, 278, 284
 boundary setting, 124, 127, 128, 152, 158, 160, 162, 178, 181, 184, 211, 273, 275–277
 chemical engineering module, 270
 composite object, 213, 281, 287–288
 concentration/temperature specified, 209–211
 constant, 155, 160
 contour plot, 212, 276, 278, 280, 281
 convective flux, 211, 215, 226
 coupling variable, 282, 285–290
 cross-section plot, 181, 278, 282
 degrees of freedom, 213
 dependent variable, 271
 ‘did not converge’, 184, 185, 186
 domain plot, 124, 128, 152, 156, 158, 160, 162, 168, 178, 181
 draw, 160, 178, 179, 213, 272
 see also composite object
 element, 272
 energy balance, 270
 error criteria, 184
 expression, 128, 155, 159, 165, 168, 185, 281
 flux specified, 210–211, 221
 with mass transfer coefficient, 210
 grid setting, 124, 158, 180, 211
 heat flux, 210, 273, 274
 inflow/outflow boundary condition, 181, 200
 init tab, 128, 162, 165, 192
 insulation/symmetry boundary condition, 210–211, 215
 integration
 boundary, 182, 187, 213, 215, 285
 domain, 153, 158, 279, 285
 internal boundary, 281
 ‘internal error in numerical routine’, 129
 Lagrange-quadratic element, 272
 see also finite elements
 legend, 156
 line, 152, 154, 165
 line/extrusion, 124, 128, 156, 162, 181, 193, 278, 283

- line marker, 156
 - line plot, 283
 - line setting, 156
 - logarithmic wall function, turbulent flow, 190
 - mass balance, 270
 - mesh, 124, 152, 154, 158, 160, 178, 211, 272, 274
 - refine, 212, 272, 276
 - model library, 270
 - model navigator, 280
 - momentum balance, 270
 - multiphysics, 127, 165, 214, 272, 280, 285
 - switch between equations, 280
 - neutral boundary condition, 200, 205
 - normal flow/pressure boundary condition, 181, 187, 190, 192, 194, 200
 - no slip boundary condition, 181, 187, 192, 194, 200
 - open, 160
 - options/expressions, 128
 - outflow/pressure, 200
 - parametric solution, 155–157, 161, 195–196, 309
 - plot parameters, 160, 278
 - pseudo 3D, 270
 - restart, 212
 - save plot, 168
 - save work, 160
 - ‘singular matrix’, 129
 - slice, 223
 - slip boundary condition, 181, 190, 192, 194, 200
 - solve, 160, 162, 178, 215, 218, 220, 275
 - time-dependent, 192
 - see also* restart
 - ‘stepsize too small’, 129
 - subdomain setting, 124, 127, 128, 152, 154, 158–160, 162, 165, 168, 178, 180, 185, 211, 220, 273, 275
 - surface plot, 212, 276, 277, 279, 281
 - symmetry condition, 181, 190, 192, 194, 200
 - temperature boundary condition, 210–211, 273, 274
 - thermal insulation, 210–211, 273, 274
 - tutorial, 160, 178
 - user modules, 270
- MATLAB commands**
- break, 240
 - clear, 242
 - command history, 240
 - command window, 240
 - conditional statements, 245
 - see also* switch, if-else
 - dbquit, 244
 - dbstop, 244
 - dbup, 244
 - debug program, 243–244
 - directory, 239, 240
 - display, 50, 242
 - eigenvalue, 248
 - fclose, 244
 - feval, 12, 132, 306
 - fid, 244
 - fminsearch, 45, 48–49, 249, 306
 - format long, 50, 242
 - fprintf, 244
 - fscanf, 244
 - fsolve, 48, 132, 137, 248
 - fzero, 11, 13, 30, 45, 46, 47, 132, 134, 249
 - global, 14, 47, 116, 241, 251
 - help, 257
 - hold on, 254
 - if-else-end, 245
 - if-elseif-end, 245
 - ‘index exceeds matrix dimensions’, 243
 - initial conditions, 249, 251
 - integral, 248
 - legend, 120, 133, 140, 254
 - loop, 245
 - matrix, 246
 - element, 246
 - element by element calculation, 248
 - multiplication, 246
 - transpose, 246
 - MaxFunEvals, 50
 - MaxIter, 50
 - mesh, 256
 - meshgrid, 256
 - m-file, 10, 114, 239, 241, 250, 251
 - name of function, 251
 - odeset, 251
 - ode15s, 140, 250, 312, 318
 - ode23s, 312, 318
 - ode45, 114, 120, 132, 138, 250, 311, 318
 - parameter, passing, 117–118, 242, 251
 - parameter estimation
 - nonlinear regression, 305–306
 - polynomial, 298–299
 - straight line, 295–296
 - pause, 255
 - plot, 14, 120, 133, 250, 253–257, 296
 - axis, 255
 - colors, 254, 256
 - contour plots, 256
 - Greek letters, 257
 - limit plot, 254–255
 - line options, 254
 - log, 253
 - markers, 254

- method of lines, 251
- multiple, 255
- subplots, 255
- symbols, 256
- three-dimensional, 255–256
- polynomial curve fit, `polyfit`, 252, 296–298
- polynomial evaluation, `polyval`, 252, 296–298
- set path, 239
- spline interpolation, 252
- stop, 244
 - see also* `dbstop`
- suppress printing, 242
- switch, 245
- tic-toc, 246
- title, 254
- tolerance, 50
- `TolFun`, 50
- `TolX`, 50
- `Tspan`, 250, 251
- ‘undefined function or variable’, 243
- vector, column, 250, 251
- ‘whos’, 244
- workspace, 241
- `xlabel`, 14, 120, 133, 140, 254
- `ylabel`, 14, 120, 133, 140, 254

SUBJECT INDEX

- Absorption, 73, 82–87
 - column, 88
 - refrigerated column, 88
- Acentric factor, 7, 22
- Activation energy, 228
- Activity, 42
- Activity coefficient, 29, 37, 74, 80
- Adiabatic boundary condition, 219
- Adsorbent, 164
- Adsorption, 73
 - linear, 164–167
- Algebraic equation, 2, 307–310
- Azeotrope, 30, 81, 88

- Backward Euler method, 312
- Bessel function, 164
- Boiling point, normal, 7
- Boston-Mathias function, 75
- Boundary value problem, 4, 112, 147, 312–317
- Brinkman number, 220
- Broyden method, 93, 100, 104
- Bubble point, 26, 78, 82

- Cartesian geometry, 176, 178
- Catalyst, 113
 - Al_2O_3 , 148
 - platinum, 143, 228
 - silver, 71
- Catalytic converter, 144, 146
- Chao-Seader thermodynamics, 75, 78, 81, 85
- Charge distribution, 172
- Chemical reaction equilibrium, 3, 41–54
- Chromatography, 167–169
- Compressibility factor, 13
- Compression ratio, 94
- Compressor, 91, 93, 94, 100
- Computational fluid dynamics (CFD), 175
- Comsol multiphysics, 269
- Condenser duty, 80–82, 85, 87
- Convection, strong, *see* Peclet number, high
- Convective diffusion equation, 3, 208–209
- Convergence acceleration, 100
 - see also* iterative method
- Conversion, 57
 - per pass, 95
- Cost
 - capital, 85, 94
 - cooling cost, 79, 85
 - distillation column, 79
 - electricity, 98
 - gas, 98
 - heating cost, 79, 85
 - operating, 80, 85
 - raw material, 55, 85
 - refrigeration, 85
 - separation, 68
 - steam, 80
 - utility, 97, 103
 - water, 98, 102

- Curve fit
 - multiple regression, 298–304
 - nonlinear regression, 304–306
 - polynomial, 296–299
 - straight line, 294–296
- Cylindrical geometry, 178
- Damköhler number, 138
- Derivative, numerical, 309
- Design choices, 89
- Dew point, 26
- Differential equation, 3
- Differential-algebraic equation, 312, 318
- Diffusion, 148, 312
 - artificial, 165, 166, 168, 319
 - polymer, 73
 - porous catalyst, 73
 - streamline, 319
 - see also* Petrov-Galerkin method, reaction-diffusion
- Diffusivity, 129, 148, 208
 - effective, 149, 165
 - polymer, 171
 - isotropic, 214
- Direct numerical simulation (DNS), 175, 176
- Dispersion
 - axial, 121
 - radial, 121
- Disproportionation, 67
- Distillate flow rate, 77
- Distillation, 58, 70, 73, 79
 - control, 79
 - multicomponent, 75–82
 - plate-to-plate calculation, 75, 80–82
 - short-cut method, 75–80
 - see also* Aspen Plus blocks DSTWU, RadFrac
- Drag
 - on sphere, 204
 - on cylinder, 205
- Drawdown ratio, 172
- Eduljee equation, 78
- Eigenvalues, 139, 311
- Electric potential, 172
- Electrochemical cell, 224
- Electrolyte, 74
- Enzyme, 144
 - loading, 144
- Epoxy, 145
- Equation of state, 3, 5–23
 - cubic equation, 7
 - Hayden-O'Connell, 74
 - HF hexamerization, 74
 - ideal gas, 6, 22, 42, 74, 130
 - Peng-Robinson, 7, 75
 - Redlich-Kwong, 6, 22, 74
 - Redlich-Kwong-Soave, 6, 22, 32, 74, 75
 - van der Waals, 6
- Equilibrium composition, 53
- Equilibrium constant, 42, 52, 129
- Equilibrium conversion, 41, 52, 63
- Euler-Lagrange method, 167
- Euler's method, 310
- Federal test cycle, 146
- Fenske equation, 77, 85
- Fiber spinning, 172, 173
- Finite difference method, 148, 312–314
 - elliptic equation, 321
 - first derivative, 313
 - in Excel, 317
 - partial differential equation, 318
 - second derivative, 313
- Finite element method, 147, 175, 177, 314–316
 - partial differential equation, 319
 - second derivative, 315
 - see also* Petrov-Galerkin method
- Finite elements
 - bilinear, 320
 - constant interpolation, 314–315
 - linear interpolation, 315–316
- Finite volume method, 175
- Flash, isothermal, 26–29
- Flow, 3
 - between flat plates, 203
 - entry
 - Newtonian fluid, 179–184, 203
 - Non-Newtonian fluid, 184–186
 - power-law fluid, 203
 - in a pipe, 149, 150, 156–162, 172, 179–186, 189–191, 191–193, 203, 205
 - past flat plate, 204
 - through orifice, 193–198
 - finite thickness, 194
 - viscous dissipation, 218
- Fluoresce, 214
- Flux, 149
- Flux-corrected method, 168
- Force
 - normal, 200
 - tangential, 200
- Friction factor, 190, 197
- Fuel cell
 - hydrogen for, 41, 43
- Fugacity, 29, 42
- Fugacity coefficient, 29, 42
- Fully developed flow, 183, 188

- Galerkin method, 320
- Gear's backward difference formula, 312
- Gibbs free energy, 42, 52
- Gilliland correlation, 78
- Graetz problem, 225, 226
 - with axial conduction, 225, 226
- Grayson thermodynamics, 75
- Grid point, 148
- Grid refinement, 314–316, 319, 322

- Hagen-Poiseuille law, 150
- Heat exchanger, 91–92, 101
 - duty, 94
 - counter-current, 109
- Heat generation, 219
- Heat of reaction, 42, 135, 149
- Heat transfer, 148, 152–153
 - from sphere, 225
 - in square, 210–213
 - in square with a hole, 213–214
 - lumped parameter model, 171
 - radiation, 171
 - unsteady or transient, 4, 151, 162–164, 317–318
- Heavy key, 76–78, 80
- Homotopy method, 309

- Implicit method, 312
- Incompressible fluid, 176
- Inert gas, 70
- Inhibition constant, 143
- Initial value methods for boundary value problem, 317
- Initial value problem, 3, 112, 249–251, 310–312
- Interpolation, 314
- Iterative method
 - direct, 93
 - equation-oriented (EO) in Aspen Plus, 100
 - see also* Wegstein and Broyden methods, tear stream, sequential solution method, simultaneous solution method

- Jacobian, 309, 311

- K-value
 - phase equilibrium, 27, 29–30, 37, 39, 52, 64, 78
 - see also* equilibrium constant
- k- ϵ model, 176, 189

- Lab-on-a-chip, 207
- Laminar flow, 175
- Langmuir isotherm, 167
- Laplace's equation, 224
- Large eddy simulation (LES), 176
- LBB condition, 177
- Least squares residual, 293, 300, 305
- Lewis number, 138
- Light key, 76–78, 80
- Liquid-liquid extraction, 73
- Limit cycle, 141
- Linear correlation coefficient, 293, 300
- Lower-upper (LU) decomposition, 309
- Lumped parameter model, 112, 171

- MacCormack method, 167, 168
- Mass transfer coefficient, 132, 144, 164
- Mass transfer equipment, 3
- Mass transfer resistance, 144
- Matrix, tridiagonal, 314
- m-file, 10, 114, 239, 241, 250, 251
- Michaelis-Menten reaction, 144, 154, 228
- Microelectromechanical system (MEMS), 207
- Microfluidic device, 73
 - dispersion in, 214
 - see also* flow through orifice, serpentine mixer, T-sensor
- Microfluidics, 175
- Mixer, MIXR, 56, 62, 65, 96
- Monod constant, 143
- Monod kinetics, 143

- Natural gas, 88, 98
- Navier-Stokes equation, 4, 175, 176–177, 201–202, 209
- Newtonian fluid, 149, 150, 156–160, 172, 179–184, 225
- Newton-Raphson iterative method, 308
- Nodal point, 148
- Nondimensionalization
 - convective diffusion equation, 209
 - Navier-Stokes equation, 201–203
- Non-Newtonian fluid, 151, 159–162, 172, 173, 177–179
 - Bird-Carreau fluid, 151, 159, 160, 184–186, 204
 - power-law, 203, 204
- Non-polar compound, 74
- Nonrandom two-liquid (NRTL) equation, 30, 52, 74, 75, 95
 - electrolyte NRTL, 74, 75
- NRTL, *see* Nonrandom two-liquid equation
- Number up, 207
- Nusselt number, 226

- Oil reservoir, 38
- Optimization, 105–107, 137
- Ordinary differential equation, 3, 112, 114, 123, 249–251, 310–317
 - high order, 310
- Packed bed, 170
 - absorption, 82–87
 - reactor, 227
- Pair-interaction parameter, 30, 32
- Partial differential equation, 4, 317–322
- Partial pressure, 25
- Peclet number, 165, 209, 215
 - high, 216, 222, 318
- Perfect gas, *see* Equation of state, ideal gas
- Perimeter, 144
- Petrov-Galerkin method, 165, 166, 168, 319
- Phase equilibria, *see* Vapor-liquid equilibria
- Polar compound, 30, 74
- Polymer melt, 175
- Polymeric composite, 145
- Pressure
 - critical, 67
 - discharge, 91
 - drop, 150, 198
 - partial, 25, 297, 299
 - vapor, 7, 25
- Process simulation, 2, 3, 89–109
- Purge stream, 65, 70, 91, 94
- p-xy diagram, *see* Vapor-liquid equilibria
- Rachford-Rice equation, 28–29, 37, 39, 64, 66
- Random choice method, 167
- Raoult's law, 74, 76
- Reaction rate, 299
 - maximum specific growth, 143
- Reaction-diffusion problem, 154–157, 169, 170, 312
 - cylindrical pellet, 225
- Reactor, 3
 - adiabatic, 135
 - axial dispersion, 169, 170
 - batch, 113
 - bioreactor, 143
 - continuous stirred tank (CSTR), 114, 134, 142, 143
 - multiple solutions, 136
 - non-isothermal, 145
 - transient, 137–141, 145
 - mass transfer, 131–134
 - microfluidic, 221
 - mole change, 130–131
 - plug flow, 112–113
 - isothermal, 118–121, 125–126, 142, 143
 - non-isothermal, 121–123, 127–129, 145
 - radial dispersion, 227
 - REAC, 56, 57, 62
 - variable density, 130
 - wall reaction, 221
- Reboiler duty, 80–82, 85, 87
- Recovery fraction, 77
- Recycle, 3, 55–71, 91, 92, 94
- Reflux rate, 75, 77, 78
 - minimum, 77
- Reflux ratio, 75–81, 87
 - infinite, 76, 85
 - minimum, 77, 79, 80, 85, 87
- Relative volatility, 76, 77
- Residual, variance, 293
- Reynolds number, 177, 186, 189, 193–205
- Runge-Kutta method, 311
- Scale up, 207
- Secant iterative method, 104, 309
- Semiconductor, 172
 - inertia, 173
 - non-Newtonian fluid, 173
- Sensitivity, 108
- Separator, SEPR, 56, 57, 62
- Sequential solution method, 60, 93, 100
- Serpentine mixer
 - flow, 199–200
 - mixing, 222
- Shear rate, 159, 172, 178, 185
- Shear thinning, 161
- Shwartzentruber-Renon solution, 75
- Simultaneous solution method, 61, 93, 100
- Singularity, 224
- Solutions
 - ideal, 29–30, 39
 - regular solution theory, 29
 - two liquid phases, 30
 - see also* Wilson equation, Non-random two liquid equation, UNIFAC group contribution method, UNIQUAC equation, Shwartzentruber-Renon, azeotrope
- Split fraction, 56, 59
- Stability limits, 311, 312
- Stages of distillation column, 75, 78, 79, 85, 87
 - minimum, 77, 79, 80, 87
- Start-up flow
 - Newtonian, 191–193
 - power-law, 204
- Steam reformer, 54
- Step-size adjustment, 311, 312
- Stiff equations, 139, 311, 318
- Stoichiometry, 46
- Stoichiometric coefficient, 57, 144
- Stoichiometric equation, 57
- Stoichiometric feed, 62, 70

- Stokes flow, 204
- Streamlines, 184, 187, 188, 196
- Substrate, 143
- Successive substitution iterative method, 308

- Taylor series, 308
- Taylor-Galerkin method, 167
- Tear stream, 104
- Temperature
 - average, 153
 - critical, 6, 7
 - reduced, 6
- Thermal conductivity, 129, 148, 152, 153
 - effective, 149
- Thermal diffusion, coefficient of, 151
- Thermodynamic choice for
 - coal processing, 75
 - gas processing, 75
 - oil and gas production, 75
 - petrochemicals, 75
 - refinery, 75
- Transalkylation, *see* disproportionation
- Transport process, 3
- Trial functions, 314
 - see also* finite elements
- Trim heater, 109
 - see also* heat exchanger
- T-sensor
 - flow, 186–189, 204
 - dispersion, 214, 227
- Turbulent flow, 175, 176, 189–191, 205
 - eddy viscosity, 189
 - kinetic energy, 189, 191
 - rate of dissipation, 189, 191
 - see also* wall layer
- T-xy diagram, *see* Vapor-liquid equilibria

- Underwood equation, 77
- UNIFAC group contribution method, 30, 74

- UNIQUAC equation, 74, 75
- Unstable solution, 137
- Utilities, 97
 - see also* cost, utility

- Van Laar equation, 74
- van't Hoff equation, 42
- Vapor-liquid equilibria, 3, 35–40, 65
 - T-xy diagram, 35
 - p-xy diagram, 35
- Variational principle, 320
- Velocity
 - average, 150, 158, 160, 183, 184, 188
 - normal, 200
 - gradient, 159
 - see also* shear rate
 - tangential, 200
 - terminal, 209
- Vena contracta, 196
- Viscosity, 150
 - kinematic, 189, 190
 - concentration dependent, 217–218
- Viscous dissipation, 194, 218–221
- Void fraction, 164
- Volume of mixture, 18
 - see also* Equation of state
- Vortices, 175

- Wall effect, 204
- Wall layer, turbulent flow, 189
 - logarithmic wall function, 190
- Wall stress, 190
- Water-gas shift reaction, 41, 43, 53
- Wegstein iterative method, 93, 100–104, 309
- Wilson equation, 30, 35, 37, 74, 75, 83, 88

- Yield, 95