

Computational Mathematics:
Models, Methods and Analysis
with
MATLAB and MPI

R. E. White
Department of Mathematics
North Carolina State University
white@math.ncsu.edu

Updated on August 3, 2003
To Be Published by CRC Press in 2003

Contents

List of Figures	ix
List of Tables	xi
Preface	xiii
Introduction	xv
1 Discrete Time-Space Models	1
1.1 Newton Cooling Models	1
1.2 Heat Diffusion in a Wire	9
1.3 Diffusion in a Wire with Little Insulation	17
1.4 Flow and Decay of a Pollutant in a Stream	25
1.5 Heat and Mass Transfer in Two Directions	32
1.6 Convergence Analysis	42
2 Steady State Discrete Models	51
2.1 Steady State and Triangular Solves	51
2.2 Heat Diffusion and Gauss Elimination	59
2.3 Cooling Fin and Tridiagonal Matrices	68
2.4 Schur Complement	77
2.5 Convergence to Steady State	86
2.6 Convergence to Continuous Model	91
3 Poisson Equation Models	99
3.1 Steady State and Iterative Methods	99
3.2 Heat Transfer in 2D Fin and SOR	107
3.3 Fluid Flow in a 2D Porous Medium	116
3.4 Ideal Fluid Flow	122
3.5 Deformed Membrane and Steepest Descent	130
3.6 Conjugate Gradient Method	138

4	Nonlinear and 3D Models	145
4.1	Nonlinear Problems in One Variable	145
4.2	Nonlinear Heat Transfer in a Wire	152
4.3	Nonlinear Heat Transfer in 2D	159
4.4	Steady State 3D Heat Diffusion	166
4.5	Time Dependent 3D Diffusion	171
4.6	High Performance Computations in 3D	179
5	Epidemics, Images and Money	189
5.1	Epidemics and Dispersion	189
5.2	Epidemic Dispersion in 2D	197
5.3	Image Restoration	204
5.4	Restoration in 2D	213
5.5	Option Contract Models	219
5.6	Black-Scholes Model for Two Assets	228
6	High Performance Computing	237
6.1	Vector Computers and Matrix Products	237
6.2	Vector Computations for Heat Diffusion	244
6.3	Multiprocessors and Mass Transfer	249
6.4	MPI and the IBM/SP	258
6.5	MPI and Matrix Products	263
6.6	MPI and 2D Models	268
7	Message Passing Interface	275
7.1	Basic MPI Subroutines	275
7.2	Reduce and Broadcast	282
7.3	Gather and Scatter	288
7.4	Grouped Data Types	294
7.5	Communicators	301
7.6	Fox Algorithm for AB	307
8	Classical Methods for $Ax = d$	313
8.1	Gauss Elimination	313
8.2	Symmetric Positive Definite Matrices	318
8.3	Domain Decomposition and MPI	324
8.4	SOR and P-regular Splittings	328
8.5	SOR and MPI	333
8.6	Parallel ADI Schemes	339
9	Krylov Methods for $Ax = d$	345
9.1	Conjugate Gradient Method	345
9.2	Preconditioners	350
9.3	PCG and MPI	356
9.4	Least Squares	360
9.5	GMRES	365

<i>CONTENTS</i>	vii
9.6 GMRES(m) and MPI	372
Bibliography	379
Index	381

List of Figures

1.1.1 Temperature versus Time	6
1.1.2 Steady State Temperature	7
1.1.3 Unstable Computation	7
1.2.1 Diffusion in a Wire	11
1.2.2 Time-Space Grid	13
1.2.3 Temperature versus Time-Space	15
1.2.4 Unstable Computation	15
1.2.5 Steady State Temperature	16
1.3.1 Diffusion in a Wire with $c_{sur} = .0000$ and $.0005$	22
1.3.2 Diffusion in a Wire with $n = 5$ and 20	23
1.4.1 Polluted Stream	26
1.4.2 Concentration of Pollutant	30
1.4.3 Unstable Concentration Computation	31
1.5.1 Heat or Mass Entering or Leaving	34
1.5.2 Temperature at Final Time	37
1.5.3 Heat Diffusing Out a Fin	38
1.5.4 Concentration at the Final Time	40
1.5.5 Concentrations at Different Times	40
1.6.1 Euler Approximations	45
2.1.1 Infinite or None or One Solution(s)	52
2.2.1 Gaussian Elimination	64
2.3.1 Thin Cooling Fin	69
2.3.2 Temperature for $c = .1, .01, .001, .0001$	75
2.6.1 Variable $r = .1, .2$ and $.3$	94
2.6.2 Variable $n = 4, 8$ and 16	95
3.1.1 Cooling Fin with $T = .05, .10$ and $.15$	105
3.2.1 Diffusion in Two Directions	108
3.2.2 Temperature and Contours of Fin	113
3.2.3 Cooling Fin Grid	114
3.3.1 Incompressible 2D Fluid	117
3.3.2 Groundwater 2D Porous Flow	118

3.3.3 Pressure for Two Wells	122
3.4.1 Ideal Flow About an Obstacle	123
3.4.2 Irrotational 2D Flow $v_x - u_y = 0$	124
3.4.3 Flow Around an Obstacle	128
3.4.4 Two Paths to (x,y)	129
3.5.1 Steepest Descent $\text{norm}(\mathbf{r})$	137
3.6.1 Convergence for CG and PCG	144
4.2.1 Change in F_1	154
4.2.2 Temperatures for Variable c	158
4.4.1 Heat Diffusion in 3D	167
4.4.2 Temperatures Inside a 3D Fin	170
4.5.1 Passive Solar Storage	171
4.5.2 Slab is Gaining Heat	178
4.5.3 Slab is Cooling	178
4.6.1 Domain Decomposition in 3D	182
4.6.2 Domain Decomposition Matrix	186
5.1.1 Infected and Susceptible versus Space	196
5.2.1 Grid with Artificial Grid Points	199
5.2.2 Infected and Susceptible at Time = 0.3	203
5.3.1 Three Curves with Jumps	206
5.3.2 Restored 1D Image	213
5.4.1 Restored 2D Image	219
5.5.1 Value of American Put Option	222
5.5.2 $P(S,T-t)$ for Variable Times	226
5.5.3 Option Values for Variable Volatilities	226
5.5.4 Optimal Exercise of an American Put	227
5.6.1 American Put with Two Assets	229
5.6.2 $\max(E_1 + E_2 - S_1 - S_2, 0)$	234
5.6.3 $\max(E_1 - S_1, 0) + \max(E_2 - S_2, 0)$	235
6.1.1 von Neumann Computer	238
6.1.2 Shared Memory Multiprocessor	239
6.1.3 Floating Point Add	239
6.1.4 Bit Adder	240
6.1.5 Vector Pipeline for Floating Point Add	241
6.2.1 Temperature in Fin at $t = 60$	248
6.3.1 Ring and Complete Multiprocessors	250
6.3.2 Hypercube Multiprocessor	250
6.3.3 Concentration at $t = 17$	256
6.4.1 Fan-out Communication	262
6.6.1 Space Grid with Four Subblocks	269
6.6.2 Send and Receive for Processors	270
7.2.1 A Fan-in Communication	283

List of Tables

1.6.1 Euler Errors at $t = 10$	45
1.6.2 Errors for Flow	48
1.6.3 Errors for Heat	48
2.6.1 Second Order Convergence	96
3.1.1 Variable SOR Parameter	104
3.2.1 Convergence and SOR Parameter	113
4.1.1 Quadratic Convergence	149
4.1.2 Local Convergence	149
4.2.1 Newton's Rapid Convergence	157
6.1.1 Truth Table for Bit Adder	239
6.1.2 Matrix-vector Computation Times	243
6.2.1 Heat Diffusion Vector Times	246
6.3.1 Speedup and Efficiency	252
6.3.2 HPF for 2D Diffusion	254
6.4.1 MPI Times for trapempi.f	262
6.5.1 Matrix-vector Product mflops	265
6.5.2 Matrix-matrix Product mflops	268
6.6.1 Processor Times for Diffusion	272
6.6.2 Processor Times for Pollutant	273
7.6.1 Fox Times	311
8.3.1 MPI Times for geddmpi.f	328
8.5.1 MPI Times for sorddmpi.f	338
9.3.1 MPI Times for egssormpi.f	360
9.6.1 MPI Times for gmresmmpi.f	376

Preface

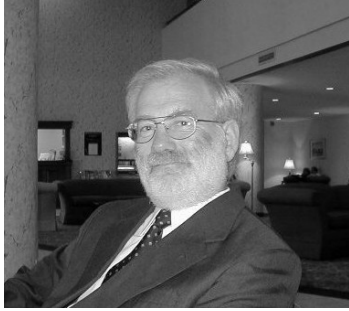
This book evolved from the need to migrate computational science into undergraduate education. It is intended for students who have had basic physics, programming, matrices and multivariable calculus.

The choice of topics in the book has been influenced by the Undergraduate Computational Engineering and Science Project (a United States Department of Energy funded effort), which was a series of meetings during the 1990s. These meetings focused on the nature and content for computational science undergraduate education. They were attended by a diverse group of science and engineering teachers and professionals, and the continuation of some of these activities can be found at the Krell Institute, <http://www.krellinst.org>. Variations of Chapters 1-4 and 6 have been taught at North Carolina State University in fall semesters since 1992. The other four chapters were developed in 2002 and taught in the 2002-03 academic year.

The department of mathematics at North Carolina State University has given me the time to focus on the challenge of introducing computational science materials into the undergraduate curriculum. The North Carolina Supercomputing Center, <http://www.ncsc.org>, has provided the students with valuable tutorials and computer time on supercomputers. Many students have made important suggestions, and Carol Cox Benzi contributed some course materials with the initial use of MATLAB[®]. MATLAB is a registered trademark of The MathWorks, Inc. For product information, please contact:

The MathWorks, Inc.
3 Apple Hill Drive
Natick, MA 01760-2098 USA
Tel: 508-647-7000
Fax: 508-647-7001
E-mail: info@mathworks.com
Web: www.mathworks.com <<http://www.mathworks.com/>>.

I thank my close friends who have listened to me talk about this effort, and especially Liz White who has endured the whole process with me.



Bob White, July 1, 2003

Introduction

Computational science is a blend of applications, computations and mathematics. It is a mode of scientific investigation that supplements the traditional laboratory and theoretical methods of acquiring knowledge. This is done by formulating mathematical models whose solutions are approximated by computer simulations. By making a sequence of adjustments to the model and subsequent computations one can gain some insights into the application area under consideration. This text attempts to illustrate this process as a method for scientific investigation. Each section of the first six chapters is motivated by a particular application, discrete or continuous model, numerical method, computer implementation and an assessment of what has been done.

Applications include heat diffusion to cooling fins and solar energy storage, pollutant transfer in streams and lakes, models of vector and multiprocessing computers, ideal and porous fluid flows, deformed membranes, epidemic models with dispersion, image restoration and value of American put option contracts. The models are initially introduced as discrete in time and space, and this allows for an early introduction to partial differential equations. The discrete models have the form of matrix products or linear and nonlinear systems. Methods include sparse matrix iteration with stability constraints, sparse matrix solutions via variation on Gauss elimination, successive over-relaxation, conjugate gradient, and minimum residual methods. Picard and Newton methods are used to approximate the solution to nonlinear systems.

Most sections in the first five chapters have MATLAB[®] codes; see [14] for the very affordable current student version of MATLAB. They are intended to be studied and not used as a "black box." The MATLAB codes should be used as a first step towards more sophisticated numerical modeling. These codes do provide a learning by doing environment. The exercises at the end of each section have three categories: routine computations, variation of models, and mathematical analysis. The last four chapters focus on multiprocessing algorithms, which are implemented using message passing interface, MPI; see [17] for information about building your own multiprocessor via free "NPACI Rocks" cluster software. These chapters have elementary Fortran 9x codes to illustrate the basic MPI subroutines, and the applications of the previous chapters are revisited from a parallel implementation perspective.

At North Carolina State University Chapters 1-4 are covered in 26 75-minute lectures. Routine homework problems are assigned, and two projects are required, which can be chosen from topics in Chapters 1-5, related courses or work experiences. This forms a semester course on numerical modeling using partial differential equations. Chapter 6 on high performance computing can be studied after Chapter 1 so as to enable the student, early in the semester, to become familiar with a high performance computing environment. Other course possibilities include: a semester course with an emphasis on mathematical analysis using Chapters 1-3, 8 and 9, a semester course with a focus on parallel computation using Chapters 1 and 6-9 or a year course using Chapters 1-9.

This text is not meant to replace traditional texts on numerical analysis, matrix algebra and partial differential equations. It does develop topics in these areas as is needed and also includes modeling and computation, and so there is more breadth and less depth in these topics. One important component of computational science is parameter identification and model validation, and this requires a physical laboratory to take data from experiments. In this text model assessments have been restricted to the variation of model parameters, model evolution and mathematical analysis. More penetrating expertise in various aspects of computational science should be acquired in subsequent courses and work experiences.

Related computational mathematics education material at the first and second year undergraduate level can be found at the Shodor Education Foundation, whose founder is Robert M. Panoff, web site [22] and in Zachary's book on programming [29]. Two general references for modeling are the undergraduate mathematics journal [25] and Beltrami's book on modeling for society and biology [2]. Both of these have a variety of models, but often there are no computer implementations. So they are a good source of potential computing projects. The book by Landau and Paez [13] has number of computational physics models, which are at about the same level as this book. Slightly more advanced numerical analysis references are by Fosdick, Jessup, Schauble and Domik [7] and Heath [10].

The computer codes and updates for this book can be found at the web site:

<http://www4.ncsu.edu/~white>.

The computer codes are mostly in MATLAB for Chapters 1-5, and in Fortran 9x for most of the MPI codes in Chapters 6-9. The choice of Fortran 9x is the author's personal preference as the array operations are similar to those in MATLAB. However, the above web site and the web site associated with Pacheco's book [21] do have C versions of these and related MPI codes. The web site for this book is expected to evolve and also has links to sequences of heat and pollution transfer images, book updates and new reference materials.