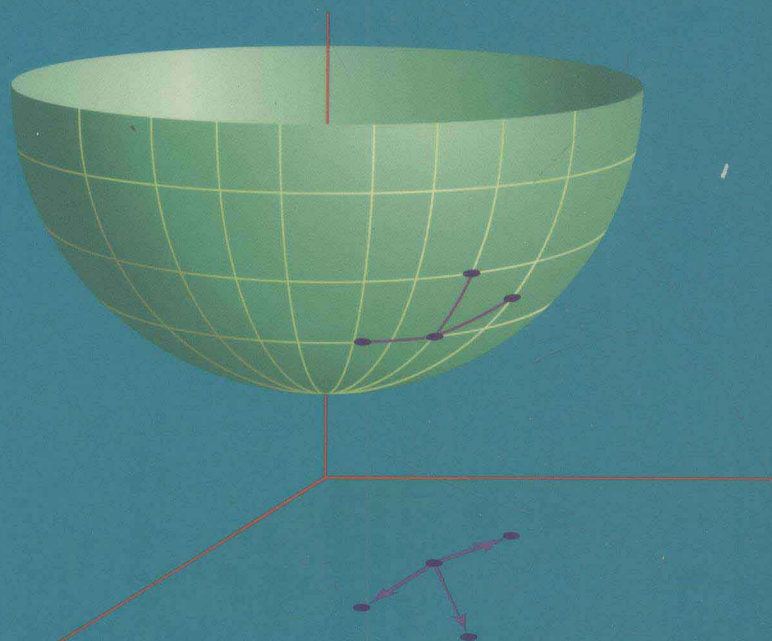


SADRI HASSANI

MATHEMATICAL METHODS

Using *Mathematica*®

FOR STUDENTS OF PHYSICS AND
RELATED FIELDS



CD-ROM
INCLUDED



MATHEMATICAL METHODS USING *MATHEMATICA*[®]

For Students of Physics
and Related Fields

Sadri Hassani

With 93 Illustrations and a CD-ROM



Sadri Hassani
Campus Box 4560
Department of Physics
Illinois State University
Normal, IL 61790-4560
USA
hassani@phy.ilstu.edu

Series Editors

John P. Ertel
Department of Physics
United States Naval Academy
572 Holloway Road
Annapolis, MD 21402-5026
USA
jpe@nadn.navy.mil

David Peak
Department of Physics
Utah State University
Logan, UT 84322
USA

Cindy Schwarz
Department of Physics
Northern Illinois University
De Kalb, IL 60115
USA

Robert C. Hilborn
Department of Physics
Amherst College
Amherst, MA 01002
USA

Thomas D. Rossing
Science Department
New Trier High School
Winnetka, IL 60093
USA

COVER ILLUSTRATION: Gradient or differentiation with respect to distance is shown in two dimensions; the surface represents a function of x and y ; the gradient is a vector in the xy -plane.

Library of Congress Cataloging-in-Publication Data
Hassani, Sadri.

Mathematical methods using *Mathematica*: for students of physics and related fields/
Sadri Hassani.

p. cm. — (Undergraduate texts in contemporary physics)

Includes bibliographical references and index.

ISBN 0-387-95523-2 (softcover: alk. paper)

1. Physics—Mathematical models. 2. Mathematical physics—Data processing.

3. *Mathematica* (Computer file) I. Title. II. Series.

QC20 .H393 2002

530'.0285'53042—dc21

2002070732

ISBN 0-387-95523-2

Printed on acid-free paper.

Mathematica is a registered trademark of Wolfram Research, Inc.

© 2003 Springer-Verlag New York, Inc.

All rights reserved. This work may not be translated or copied in whole or in part without the written permission of the publisher (Springer-Verlag New York, Inc., 175 Fifth Avenue, New York, NY 10010, USA), except for brief excerpts in connection with reviews or scholarly analysis. Use in connection with any form of information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed is forbidden.

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

Printed in the United States of America.

9 8 7 6 5 4 3 2 1

SPIN 10881953

Typesetting: Pages created by the author in LaTeX 2 ϵ using Springer's svsing2e.sty macro.

www.springer-ny.com

Springer-Verlag New York Berlin Heidelberg
A member of BertelsmannSpringer Science+Business Media GmbH

UNDERGRADUATE TEXTS IN CONTEMPORARY PHYSICS

Series Editors

John P. Ertel

Robert C. Hilborn

David Peak

Thomas D. Rossing

Cindy Schwarz

Springer

New York

Berlin

Heidelberg

Hong Kong

London

Milan

Paris

Tokyo

UNDERGRADUATE TEXTS IN CONTEMPORARY PHYSICS

Cassidy, Holton, and Rutherford, Understanding Physics

Enns and McGuire, Computer Algebra Recipes: A Gourmet's Guide to the Mathematical Models of Science

Hassani, Mathematical Methods: For Students of Physics and Related Fields

Hassani, Mathematical Methods Using *Mathematica*®: For Students of Physics and Related Fields

Holbrow, Lloyd, and Amato, Modern Introductory Physics

Möller, Optics: Learning by Computing, with Examples Using Mathcad®

Roe, Probability and Statistics in Experimental Physics, Second Edition

Rossing and Chiaverina, Light Science: Physics and the Visual Arts

To my wife, Sarah,
and to my children,
Dane Arash and Daisy Bitá

Preface

Over two years have passed since the publication of *Mathematical Methods*, my undergraduate textbook to which the present book was to be a companion. The initial motivation for writing this book was to take some examples from *Mathematical Methods* in which to illustrate the use of a symbolic language such as *Mathematica*[®]. However, after writing the first few pages, I realized very quickly that, for the book to be most effective, I had to go beyond the presentation of examples. I had to talk about the theory of numerical integration, discrete differentiation, solution of differential equations, and a number of other topics; thus the delay in the publication of the book.

As a result, the book has become a *self-contained* introduction to the use of computer algebra—specifically, *Mathematica*—for undergraduates in physics and related fields. Although many of the examples discussed here are taken from *Mathematical Methods*, no prior knowledge of the content of that book is essential for learning the *techniques* of computer algebra. Of course, a deeper understanding of the underlying physical ideas requires reading the relevant sections of *Mathematical Methods* or a book like it. For those interested in the underlying theories of the examples being discussed, I have placed the appropriate page (or section) numbers in the margin.

I have to emphasize that the book does not discuss programming in *Mathematica*. Nor does it teach all the principles and techniques of the most elegant utilization of *Mathematica*. The book can best be described as “learning the essentials of *Mathematica* through examples from undergraduate physics.” In other words, *Mathematica* commands and techniques are introduced as the need arises.

I believe that some understanding of the theory behind the numerical calculations is important, especially if it can invoke some *Mathematica* usage. Therefore, I have included an entire chapter on the theory of the numerical solutions of differential equations, and a rather lengthy discussion on the theory behind numerical integration. In both discussions I make use of *Mathematica* to enhance the understanding of the theories.

After introducing the essential *Mathematica* commands in Chapter 1, I introduce vectors—using the calculation of electric fields and potentials of discrete charge distributions—and matrices—using the calculation of normal modes of mass-spring systems—in Chapter 2 as they are used in *Mathematica*. Chapter 3 discusses numerical integration and a variety of its applications in different physical settings such as the evaluation of electric, magnetic, and gravitational fields of various sources. Infinite series and finite sums are the subject of Chapter 4, in which the theory of numerical integration is used as a nice example of the use of summation in *Mathematica*. Chapter 5 is devoted entirely to a theoretical treatment of the numerical solution of differential equations, discussing such techniques as the Euler methods, the Runge–Kutta method, and the use of discrete differentiation in solving eigenvalue problems. In Chapter 6, I have chosen some examples from classical and quantum mechanics to illustrate how *Mathematica* solves ordinary differential equations.

This book can be used in conjunction with any undergraduate mathematical physics book. Many problems are inherently interesting but cannot be solved analytically. Once the student learns the theory and formal mathematics behind a concept and solves a number of simple and ideal examples analytically, he or she ought to be exposed to problems arising from real-world applications. *Mathematica* (or any other computer-algebra software) can be of tremendous help in treating such problems and exhibiting their solutions graphically (or otherwise). However, *Mathematica* has its greatest impact on the process of learning only if the student has completed the preliminary stage of deeply understanding the analytical methods of solution.

This is hardly the place to enter into the controversy surrounding the role of content and memorization in learning. However, as an educator witnessing the alarming rate at which calculators and computer-algebra software are substituting the learning of physics and mathematics, I feel obligated to emphasize the distinction between the *real utility* of technology and its *advertised glamour*. Technology can be a great tool of learning and teaching once students acquire a certain degree of mathematical maturity. And this maturity can be obtained only through a rigorous training in conventional mathematics that emphasizes content at all levels of a student's education.

The neglect of content—such as the multiplication table at the elementary level, and algebraic/trigonometric identities at the high school level—can have a detrimental effect on the mathematical and analytical ability of the pupil's mind. If the educators sequentially postpone the “memoriza-

tion” of the multiplication table, algebraic and trigonometric identities, and differentiation and integration rules, arguing that such “facts” are always available on calculators and computers, then students will develop the skill of “pushing buttons” beautifully but will be incapable of doing the simplest integration. Some educators argue that lack of ability to multiply, integrate, or simplify an algebraic expression is not a drawback as long as there are calculators to do the job. To this I have to respond that heavy reliance on calculating machines does to the mind what heavy reliance on vehicular machines does to the body: it makes the mind lazy and inactive. Our minds need raw data—in the form of numbers and symbols in conjunction with the rules that manipulate them—to develop. A mind without data is like a symphony without notes, an opera without lyrics, a poem without words. I sincerely hope that the readers and users of this book will take this advice to heart.

Sadri Hassani
Campus Box 4560
Department of Physics
Illinois State University
Normal, IL 61790-4560, USA

e-mail: hassani@phy.ilstu.edu

Note to the Reader

I should point out from the very beginning that, as powerful as *Mathematica* is, it is only a tool. And a tool is more useful if its user has thought through the details of the task for which the tool is designed. Just as one needs to master multiplication—both conceptually (where and how it is used) and factually (the multiplication table)—before a calculator can be of any use, so does one need to master algebra, calculus, trigonometry, differential equations, etc., before *Mathematica* can be of any help. In short, *Mathematica* cannot think for you.

Once you have learned the concepts behind the equations and *know how to set up a specific problem*, *Mathematica* can be of great help in solving that problem for you. This book, of course, is not written to help you set up the problems; for that, you have to refer to your physics or engineering books. The purpose of this book is to familiarize you with the simple—but powerful—techniques of calculation used to solve problems that are otherwise insoluble. I have taken many examples from your undergraduate courses and have used a multitude of *Mathematica* techniques to solve those problems.

I encourage you to explore the CD-ROM that comes with the book. Not only does it contain all the codes used in the book, but it also gives many explanations and tips at each step of the solution of a problem. The CD-ROM is compatible with both *Mathematica* 3.0 and *Mathematica* 4.0.

Mathematica, like any other calculational tool, is only as smart as its user can make it!

Contents

Preface	vii
Note to the Reader	xi
1 <i>Mathematica</i> in a Nutshell	1
1.1 Running <i>Mathematica</i>	1
1.2 Numerical Calculations	2
1.3 Algebraic and Trigonometric Calculations	4
1.4 Calculus in <i>Mathematica</i>	8
1.5 Numerical Mathematics	13
1.6 Graphics	18
1.6.1 Simple Plots	18
1.6.2 Parametric Plots	19
1.6.3 Contour and Density Plots	22
1.6.4 Three-Dimensional Plots	23
1.7 Complex Numbers	26
1.7.1 An Example from Optics	28
1.8 Animation	36
1.9 Input and Output Control	41
1.10 Problems	43
2 Vectors and Matrices in <i>Mathematica</i>	49
2.1 Electric Fields	49
2.2 Ionic Crystals	54

2.2.1	One-Dimensional Crystal	54
2.2.2	Two-Dimensional Crystal	56
2.2.3	Three-Dimensional Crystal	58
2.3	Tubing Curves	60
2.4	Matrices	66
2.5	Normal Modes	70
2.5.1	A System of Two Masses	70
2.5.2	A System of Three Masses	73
2.5.3	A System of Five Masses	75
2.6	Normal Modes of a System of n Masses	78
2.7	Problems	82
3	Integration	85
3.1	Integration in <i>Mathematica</i>	86
3.1.1	The Simple Example of a Pendulum	86
3.1.2	Shortcomings of Numerical Integration in <i>Mathematica</i>	87
3.1.3	Other Intricacies of Integration in <i>Mathematica</i>	89
3.2	Integration in Mechanics	91
3.2.1	Position-Dependent Forces	91
3.2.2	Gravitational Fields	95
3.3	Integration in Electrostatics	106
3.3.1	Potential of a Ring	108
3.3.2	Potential of a Spiral	109
3.3.3	Flat Surface Charge Distributions	109
3.4	Integration in Magnetism	110
3.4.1	Circular Loop	111
3.4.2	Current with General Shape	112
3.4.3	Solenoid	113
3.4.4	Rotating Charged Spherical Shell	115
3.4.5	Rotating Charged Hollow Cylinder	118
3.5	Problems	120
4	Infinite Series and Finite Sums	125
4.1	Infinite Sequences	125
4.2	Numerical Integration	128
4.2.1	The Simplest Method	129
4.2.2	Trapezoid Rule	130
4.2.3	Simpson's Rule	132
4.2.4	Gaussian Integration	134
4.3	Working with Series in <i>Mathematica</i>	138
4.4	Equations Involving Series	140
4.5	Fourier Series	146
4.6	Problems	150

5 Numerical Solutions of ODEs: Theory	155
5.1 Various Euler Methods	156
5.1.1 Euler Method	156
5.1.2 Modified Euler Method	157
5.1.3 Improved Euler Method	157
5.1.4 Euler Methods in <i>Mathematica</i>	158
5.1.5 Alternative Derivation of the Improved Euler Method	161
5.2 The Kutta Method	162
5.3 The Runge–Kutta Method	164
5.4 Higher-Order Equations	168
5.5 Eigenvalue Problems	170
5.5.1 Discrete Differentiation	171
5.5.2 Discrete Eigenvalue Problem	173
5.6 Problems	173
6 Numerical Solutions of ODEs: Examples Using <i>Mathematica</i>	177
6.1 Some Analytic Solutions	177
6.2 A One-Dimensional Projectile	180
6.3 A Two-Dimensional Projectile	185
6.4 The Two-Body Problem	191
6.4.1 Precession of the Perihelion of Mercury	196
6.5 The Three-Body Problem	198
6.5.1 Massive Star and Two Planets	198
6.5.2 Light Star and Two Planets	200
6.6 Nonlinear Differential Equations	202
6.7 Time-Independent Schrödinger Equation	210
6.7.1 Infinite Potential Well	211
6.7.2 The General Case	216
6.7.3 Finite Potential Well	218
6.7.4 Harmonic Oscillator	221
6.8 Problems	223
References	227
Index	229

Mathematica in a Nutshell

Mathematica® is a high-level computer language that can perform symbolic, numerical, and graphical manipulation of mathematical expressions. In this chapter we shall learn many of the essential *Mathematica* commands.

1.1 Running *Mathematica*

Installing and running *Mathematica* differ from one computer system to another. However, the heart of *Mathematica*, where the calculations are performed, is the same in all systems. *Mathematica* has two major components, the **kernel** and the **front end**. The front end is the window in which you type in your commands. These windows are generally part of **notebooks**, which are *Mathematica*'s interface with the kernel. The kernel is where the commands are processed. It could reside in the computer where the front end resides, or it could be in a remote computer.

Mathematica is launched by double-clicking on its icon—or any other shortcut your computer system recognizes. Almost all front ends now incorporate notebooks, and I assume that the reader is communicating with *Mathematica* through this medium. The window of a notebook looks like any other window. After typing in your command, hold down the **Shift** key while hitting the **Return** key to execute that command. In Macintosh, the numeric **Enter** key will also do the job.

When you enter a command, *Mathematica* usually precedes it with an input sign such as `In[1] :=`; and when it gives out the result of the calcu-

of **kernels**, front ends,
and notebooks

Shift+Return or
numeric **Enter** tells
Mathematica to start.

lation, the output sign `Out[1]=` appears in front of the answer. The input and output numbers change as the session progresses. This is a convenient way of keeping track of all inputs and outputs for cross referencing. Thus, if you type `2+2` and enter the result, *Mathematica* turns it into `In[1]:= 2+2` and gives the result as `Out[1]= 4`.

In the remaining part of this chapter (and indeed throughout the book), we are going to discuss most of the commands an average user of *Mathematica* will need. Nevertheless, for the important details omitted in this book, the reader is urged to make frequent use of the definitive *Mathematica Book* [Wolf 96] as well as the **Help** menu, which includes the online version of the *Mathematica Book*.

1.2 Numerical Calculations

Mathematica recognizes two types of numerical calculations: integer and floating point. When the input of a mathematical expression is in integer form, *Mathematica*—unless asked specifically—does not approximate the final answer in decimal format. Consider asking *Mathematica* to add $11/3$ to $217/43$, by typing in

```
11/3 + 217/43
```

and pressing the numeric **Enter** key. *Mathematica* will give the answer as $\frac{1124}{129}$. On the other hand, if you type in

```
11./3 + 217/43
```

you will get the answer 8.71318. The difference between the two inputs is the occurrence of the floating (or decimal) point. In the second input *Mathematica* treats 11. and all the other numbers in the expression as real numbers and manipulates them as such.

When *Mathematica* encounters expressions involving integers, it evaluates them and often gives the exact result. For example, for 5^{100} , *Mathematica* gives an exact 70-digit answer:

```
In[2]:= 5^100
```

```
Out[2]= 788860905221011805411728565282786229673206435109\
0230047702789306640625
```

One can always get an approximate decimal answer by ending the input with `// N`

```
In[3]:= 5^100 // N
```

```
Out[3]= 7.88861 × 1069
```

Mathematica calculates
integer expressions
exactly.

use of `// N`

An alternative way of getting approximations is to use `N[expr,n]`, which returns the numerical value of `expr` to `n` significant figures. Here is how to find the numerical value of π to any desired significant figures. Simply replace 40 with some other (positive) integer:

use of `N[,]`

```
In[4]:= N[Pi,40]
```

```
Out[4]= 3.141592653589793238462643383279502884197
```

The example above illustrates how *Mathematica* denotes the constant π . Other mathematical constants also have their own notations:

Pi	$\pi = 3.14159$
E	$e = 2.71828$
I	$i = \sqrt{-1}$
Infinity	∞
Degree	$\pi/180$: as in 30 Degrees

Mathematica understands the usual mathematical functions with two caveats:

- All *Mathematica* functions begin with a capital letter.
- The arguments are enclosed in square brackets.

Functions begin with capital letters; arguments are enclosed in square brackets.

Here is a list of some common functions:

Sqrt[x]	\sqrt{x}	Sin[x], ArcSin[x]	sine, its inverse
Exp[x]	e^x	Cos[x], ArcCos[x]	cosine, its inverse
Abs[x]	absolute value	Tan[x], ArcTan[x]	tangent, its inverse
n!	the factorial	Cot[x], ArcCot[x]	cotangent, its inverse
Log[x]	natural log	Log[b,x]	log to base b

By default, the arguments of the trigonometric functions are treated as radians. You can, however, use degrees:

```
In[5]:= Sin[Pi/3]-Cos[45 Degree]
```

```
Out[5]= -\frac{1}{\sqrt{2}} + \frac{\sqrt{3}}{2}
```

Arguments of the trigonometric functions are treated as radians.

Note that *Mathematica* returns the *exact* result. This is because no floating point appeared in the arguments of the functions. Changing 3 to 3. or 45 to 45. returns 0.158919. Similarly, `Sqrt[2]` will return $\sqrt{2}$, but


```
In[6]:= N[Sqrt[2], 45]
```

```
Out[6]= 1.41421356237309504880168872420969807856967188
```

Reusing the existing
expressions

Mathematica has a very useful shortcut for reusing the existing expressions.

%	the last result generated
%%	the next-to-last result generated
%n	the result on output line Out[n]

Typing in `%^2` squares the last result generated and returns its value. Similarly, `Sqrt[%6]` takes the square root of the result on output line Out[6].

1.3 Algebraic and Trigonometric Calculations

The most powerful aspect of *Mathematica* is its ability to handle symbolic mathematics, including all the manipulations one encounters in algebra. The following is a partial list of algebraic expressions frequently encountered in calculations.

<code>Expand[expr]</code>	multiply products and powers in <i>expr</i>
<code>Factor[expr]</code>	write <i>expr</i> as products of minimal factors
<code>Simplify[expr]</code>	simplify <i>expr</i> (standard)
<code>FullSimplify[expr]</code>	simplify <i>expr</i> (comprehensive)
<code>PowerExpand[expr]</code>	transform $(xy)^p$ to $x^p y^p$; useful for changing $\sqrt{a^2}$ to a

Mathematica allows a convenient method of substituting values for a quantity in an expression:

<code>expr /. x -> value</code>	replace <i>x</i> by <i>value</i> in <i>expr</i>
<code>expr /. {x -> xval, y -> yval}</code>	perform several replacements

Here is an example of the use of some of the above:

```
In[1]:= x^2-2x+1 /. x -> 2 + y
```

```
Out[1]= 1 - 2(2 + y) + (2 + y)^2
```

```
In[2]:= Expand[%]
```