PREFACE

to *The Mathematica GuideBooks*

Mathematica Concepts—Programming Examples—Scientific Applications

---

Bei mathematischen Operationen kann sogar eine gänzliche Entlastung des Kopfes eintreten, indem man einmal ausgeführte Zähloperationen mit Zeichen symbolisiert und, statt die Hirnfunktion auf Wiederholung schon ausgeführter Operationen zu verschwenden, sie für wichtigere Fälle aufspart.

When doing mathematics, instead of burdening the brain with the repetitive job of redoing numerical operations which have already been done before, it's possible to save that brainpower for more important situations by using symbols, instead, to represent those numerical calculations.

Ernst Mach (1883) [45✱]

Computer Mathematics and Mathematica

Computers were initially developed to expedite numerical calculations. A newer, and in the long run, very fruitful field is the manipulation of symbolic expressions. When these symbolic expressions represent mathematical entities, this field is generally called computer algebra [8✱]. Computer algebra begins with relatively elementary operations, such as addition and multiplication of symbolic expressions, and includes such things as factorization of integers and polynomials, exact linear algebra, solution of systems of equations, and logical operations. It also includes analysis operations, such as definite and indefinite integration, the solution of linear and nonlinear ordinary and partial differential equations, series expansions, and residue calculations. Today, with computer algebra systems, it is possible to calculate in minutes or hours the results that would (and did) take years to accomplish by paper and pencil. One classic example is the calculation of the orbit of the moon, which took the French astronomer Delaunay 20 years [12✱], [13✱], [14✱], [15✱], [11✱], [26✱], [27✱], [53✱], [16✱], [17✱], [25✱]. (The *Mathematica GuideBooks* cover the two other historic examples of calculations that, at the end of the 19th century, took researchers many years of hand calculations [1✱], [4✱], [38✱] and literally thousands of pages of paper.)

Along with the ability to do symbolic calculations, four other ingredients of modern general-purpose computer algebra systems prove to be of critical importance for solving scientific problems:
- a powerful high-level programming language to formulate complicated problems
- programmable two- and three-dimensional graphics
- robust, adaptive numerical methods, including arbitrary precision and interval arithmetic
- the ability to numerically evaluate and symbolically deal with the classical orthogonal polynomials and special functions of mathematical physics.

The most widely used, complete, and advanced general-purpose computer algebra system is *Mathematica*. *Mathematica* provides a variety of capabilities such as graphics, numerics, symbolics, standardized interfaces to other programs, a complete electronic document-creation environment (including a full-fledged mathematical typesetting system), and a variety of import and export capabilities. Most of these ingredients are necessary to coherently and exhaustively solve problems and model processes occurring in the natural sciences [41✱], [58✱], [21✱], [39✱] and other fields using constructive mathematics, and as well to properly represent the results. Consequently, *Mathematica*'s main areas of application are presently in the natural sciences, engineering, pure and applied mathematics, economics, finance, computer graphics, and computer science.

Mathematica is an ideal environment for doing general scientific and engineering calculations, for investigating and solving many different mathematically expressable problems, for visualizing them, and for writing notes, reports, and papers about them. Thus, *Mathematica* is an integrated computing environment, meaning it is what is also called a "problem-solving environment" [40✱], [23✱], [6✱], [48✱], [43✱], [50✱], [52✱].

Scope and Goals

The Mathematica GuideBook to Programming is the first in a series of four independent books whose main focus is to show how to solve scientific problems with *Mathematica*. Each book addresses one of the four ingredients to solve nontrivial and real-life mathematically formulated problems: programming, visualization, numerics, and symbolics.

The four *Mathematica GuideBooks* discuss programming, two-dimensional and three-dimensional graphics, numerics, and symbolics (including special functions). While the four books build on each other, each one is self-contained. Each book discusses the definition, use, and unique features of the corresponding *Mathematica* functions, gives small and large application examples with detailed references, and includes an extensive set of relevant exercises and solutions.

The *GuideBooks* have three primary goals:

- to give the reader a solid working knowledge of *Mathematica*
- to give the reader a detailed knowledge of key aspects of *Mathematica* needed to create the "best", fastest, shortest, and most elegant solutions to problems from the natural sciences
- to convince the reader that working with *Mathematica* can be a quite fruitful, enlightening, and joyful way of cooperation between a computer and a human.

Realizing these goals is achieved by understanding the unifying design and philosophy behind the *Mathematica* system through discussing and solving numerous example-type problems. While a variety of mathematics and physics problems are discussed, the *GuideBooks* are not mathematics or physics books (from the point of view of content and rigor; no proofs are typically involved), but rather the author builds on *Mathematica*'s mathematical and scientific knowledge to explore, solve, and visualize a variety of applied problems.

The focus on solving problems implies a focus on the computational engine of *Mathematica,* the kernel—rather than on the user interface of *Mathematica,* the front end. (Nevertheless, for a nicer presentation inside the electronic version, various front end features are used, but are not discussed in depth.)

The *Mathematica GuideBooks* go far beyond the scope of a pure introduction into *Mathematica*. The books also present instructive implementations, explanations, and examples that are, for the most part, original. The books also discuss some "classical" *Mathematica* implementations, explanations, and examples, partially available only in the original literature referenced or from newsgroups threads.

In addition to introducing *Mathematica*, the *GuideBooks* serve as a guide for generating fairly complicated graphics and for solving more advanced problems using graphical, numerical, and symbolical techniques in cooperative ways. The emphasis is on the *Mathematica* part of the solution, but the author employs examples that are not uninteresting from a content point of view. After studying the *GuideBooks*, the reader will be able to solve new and old scientific, engineering, and recreational mathematics problems faster and more completely with the help of *Mathematica*—at least, this is the author's goal. The author also hopes that the reader will enjoy using *Mathematica* for visualization of the results as much as the author does, as well as just studying *Mathematica* as a language on its own.

In the same way that computer algebra systems are not "proof machines" [46✱], [9✱], [37✱], [10✱], [54✱], [55✱], [56✱] such as might be used to establish the four-color theorem ([2✱], [22✱]), the Kepler [28✱], [19✱], [29✱], [30✱], [31✱], [32✱], [33✱], [34✱], [35✱], [36✱] or the Robbins ([44✱], [20✱]) conjectures, proving theorems is not the central theme of the *GuideBooks*. However, powerful and general proof machines [9✱], [42✱], [49✱], [24✱], [3✱], founded on *Mathematica*'s general programming paradigms and its mathematical capabilities, have been built (one such system is *Theorema* [7✱]). And, in the *GuideBooks*, we occasionally prove one theorem or another theorem.

In general, the author's aim is to present a realistic portrait of *Mathematica*: its use, its usefulness, and its strengths, including some current weak points and sometimes unexpected, but often nevertheless quite "thought through", behavior. *Mathematica* is not a universal tool to solve arbitrary problems which can be formulated mathematically—only a fraction of all mathematical problems can even be formulated in such a way to be efficiently expressed today in a way understandable to a computer. Rather, it is often necessary to do a certain amount of programming and occasionally give *Mathematica* some "help" instead of simply calling a single function like `Solve` to solve a system of equations. Because this will almost always be the case for "real-life" problems, we do not restrict ourselves only to "textbook" examples, where all goes smoothly without unexpected problems and obstacles. The reader will see that by employing *Mathematica's* programming, numeric, symbolic, and graphic power, *Mathematica* can offer more effective, complete, straightforward, reusable, and less likely erroneous solution methods for calculations than paper and pencil, or numerical programming languages.

Although the *Guidebooks* are large books, it is nevertheless impossible to discuss all of the 2,000+ built-in *Mathematica* commands. So, some simple as well as some more complicated commands have been omitted. For a full overview about *Mathematica*'s capabilities, it is necessary to study *The Mathematica Book* [60✱] in detail. The commands discussed in the *Guidebooks* are those that an engineer or scientist needs for solving *typical* problems, if such a thing exists [18✱]. These subjects include a quite detailed discussion of the structure of *Mathematica* expressions, *Mathematica* input and output (important for the human–*Mathematica* interaction), graphics, numerical calculations, and calculations from classical analysis. Also, emphasis is given to the powerful algebraic manipulation functions. Interestingly, they frequently allow one to solve analysis problems in an algorithmic way [5✱]. These functions are typically not so well known because they are not taught in classical engineering or physics-mathematics courses, but with the advance of computers doing symbolic mathematics, their importance increases [47✱].

■ A thorough knowledge of the:
- types and syntax of expressions
- formation and dissection of expressions
- order and evaluation of expressions
- arguments, attributes, and options of functions
- key functions for procedural, rule-based, and functional programming
- lists as universal containers and basic linear algebra objects

is a prerequisite for an efficient and successful use of *Mathematica* and its mathematical capabilities to solve problems. *The Mathematica GuideBook to Programming* discusses these subjects.

■ A thorough knowledge of the:
- graphics types
- graphics directives and primitives
- functions for two-dimensional, three-dimensional, contour, and density plots
- options determining the appearance of graphics
- structure of graphics objects

is essential for the creation of visualization and is frequently an efficient and invaluable tool to support mathematical problem-solving activities. *The Mathematica GuideBook to Graphics* discusses these subjects.

■ A thorough knowledge of the:
- machine and high-precision numbers, packed arrays, and intervals
- machine, high-precision, and interval arithmetic

- process of compilation, its advantages and limits
- main operations of numerical analysis, such as equation solving, minimization, summation
- numerical solution of ordinary and partial differential equations

is needed for virtually any nontrivial numeric calculation and frequently also in symbolic computations. *The Mathematica GuideBook to Numerics* discusses these subjects.

■ A thorough knowledge of the:

- structural operations on polynomials, rational functions, and trigonometric functions
- algebraic operations on polynomial equations and inequalities
- process of compilation, its advantages and limits
- main operations of calculus—univariate and multivariate differentiation and integration
- solution of ordinary and partial differential equations

is needed to put the heart of Mathematica—its symbolic capabilities—efficiently and successfully to work in the solution of model and real-life problems. *The Mathematica GuideBooks to Symbolics* discusses these subjects.

Content Overview

■ The Programming Volume

The Mathematica GuideBook to Programming has six chapters. Each chapter is subdivided into sections (which have occasionally subsections), exercises, solutions to the exercises, and references.

Chapter 1 is an overall introduction to *Mathematica*. It gives an outline of *Mathematica*'s syntax, its programming, graphic, numeric, and symbolic capabilities, and shows how these capabilities naturally work together. This chapter contains a sampler of smaller examples that are discussed throughout the four *GuideBooks*.

The five subsequent chapters deal with the structure of *Mathematica* expressions and with *Mathematica* as a programming language. This includes the hierarchical construction of all *Mathematica* objects from symbolic expressions (all of the form *head*[*argument*]), the ultimate building blocks of expressions (which are numbers, symbols, and strings), the definition of functions, rule applications, the recognition of patterns and their efficient application, program flows and program structure, the manipulation of lists (which are the universal containers for *Mathematica* expressions of all kinds), and a number of topics specific to the *Mathematica* programming language. Its powerful functional programming constructs are covered in great detail.

Chapter 2 discusses the basic structure of *Mathematica* expressions, the uniform recursive way to build them, and how to analyze expressions. To have a minimal working set of mathematical functions, we discuss the basic arithmetic operations, as well as the trigonometric and hyperbolic functions and their inverses. Emphasis is given to the branch cut structure of compositions and of inverse functions in *Mathematica*. This is an important area where paper and pencil calculations deviate from computer mathematics-calculations. In addition, the basics of machine and high-precision numericalization of expressions are discussed.

Chapter 3 introduces patterns, immediate and delayed function definitions, attributes of functions (representing such properties as commutativity and associativity), and functions within the λ-calculus.

Chapter 4 deals with *Mathematica* as a programming system and its evaluation semantics. The various scoping constructs are analyzed and compared in detail. The evaluation order of expressions is explained carefully.

Chapter 5 discusses advanced patterns and rule-based programming. We also review Boolean expressions and give a larger set of examples showing how the powerful paradigm of rule-based programming can be used for short and elegant solutions of various problems.

Chapter 6 discusses lists and operations to manipulate them. Because vectors and matrices are represented as lists in *Mathematica*, linear algebra functions are also discussed. The possibility of manipulating lists as whole entities allows for very concise and effective programs. The last section analyzes a number of *Mathematica* programming examples in order to determine the top ten *Mathematica* commands. (Note that the *GuideBooks* use the terms 'command' and 'function' interchangeably.)

The Appendix contains some general references regarding algorithms and applications of computer algebra and *Mathematica*.

The Mathematica GuideBook to Programming deals mostly with *Mathematica*-related issues. General programming issues, not specific to *Mathematica*, such as data structures, program flows, etc., and mathematics and physics applications is only touched on occasionally.

■ The Graphics Volume

The Mathematica GuideBook to Graphics has three chapters. Each chapter is subdivided into sections (which have occasionally subsections), exercises, solutions to the exercises, and references.

This volume deals with two- (2D) and three-dimensional (3D) graphics. The chapters give a detailed treatment of how to create images from graphics primitives, such as points, lines, and polygons. This volume also covers the issue of graphically displaying functions given either in analytical or in discrete form. We also reconstruct a number of images from the *Mathematica* Graphics Gallery. The author hopes that the reader will find *Mathematica* graphics interesting and worth learning. With some imagination, and by putting the universality of *Mathematica*'*s* programming language *Mathematica*'*s* mathematical knowledge/algorithms to work, it is possible to create an unlimited (in number and complexity) variety of meaningful as well as aesthetically pleasing images (including artistic ones) that are virtually impossible to generate in other programming or graphics systems. Also discussed is the generation of scientific visualizations of functions, formulae, and algorithms.

Mathematica's 3D rendering system is geared toward scientific visualization, not photorealistic rendering. Therefore this volume concentrates on the primitives forming a graphics-object rather than on texturing surfaces. Because all graphics primitives are *Mathematica* expressions and even rendered pictures can be converted into *Mathematica* expressions, it is possible to create stunning visualizations from pure mathematics or the natural sciences topics. Further, the use of *Mathematica*'s graphics capabilities provides a very efficient and instructive way to learn how to deal with structures arising in solving complicated problems.

Chapter 1 starts with 2D graphics. After examining graphics primitives and options, the plotting functions are discussed. We devote ten subsections to the construction of various iterative graphics. One larger section deals with the implementation, testing, and application of a more complex graphics problem. This chapter also introduces animations.

Chapter 2 deals with 3D graphics. After discussing the 3D graphics primitives, options, and plotting functions, ten subsections are devoted to the construction of a variety of more complicated graphics. A larger section deals with the construction and visualization of 3D Brillouin zones, a family of complicated polyhedra of relevance to physics and material science.

Chapter 3 covers the subject of contour and density plots. Because of the practical importance of equipotential surfaces, a relatively large section is devoted to 3D contour plots. In this volume, not too much mathematics is used, but the focus is on graphics.

Mathematica graphics are *Mathematica* expressions, allowing one to manipulate them in many ways. This book gives many examples making use of the powerful concept of symbolic expressions. But this volume is not a treatise on computational geometry. In most cases, we focus on a clear, straightforward *Mathematica* implementation instead of on finding an implementation with the lowest algorithmic complexity.

■ The Numerics Volume

The Mathematica GuideBook to Numerics has two chapters. Each chapter is subdivided into sections (which have occasionally subsections), exercises, solutions to the exercises, and references.

This volume deals with *Mathematica*'s numerical mathematics capabilities, the indispensable tools for dealing with virtually any “real life” problem. Fast machine, exact integer, and rational and verified high-precision arithmetic is applied to a large number of examples in the main text and in the solutions to the exercises.

Chapter 1 deals with numerical calculations, which are important for virtually all *Mathematica* users. This volume starts with calculations involving real and complex numbers with an “arbitrary” number of digits. (Well, not really an "arbitrary" number of digits, but on present-day computers, many calculations involving a few million digits are easily feasible.). Then follows a discussion of significance arithmetic, which automatically keeps track of the digits which are correct in calculations with high-precision numbers. Also discussed is the use of interval arithmetic. (Despite being slow, exact and/or inexact, interval arithmetic allows one to carry out validated numerical calculations.) The next important subject is the (pseudo)compilation of *Mathematica* code. Because *Mathematica* is an interpreted language that allows for “unforeseeable” actions and arbitrary side effects at runtime of *Mathematica* code, it generally cannot be compiled. Strictly numerical calculations can, of course, be compiled.

Then, the main numerical functions are discussed: interpolation, Fourier transforms, numerical summation and integration, solution of equations (root finding), minimization of functions, and the solution of ordinary and partial differential equations. To illustrate *Mathematica*'s differential equation solving capabilities, a larger amount of well-known ODEs and PDEs is discussed. Many medium-sized examples are given for the various numerical procedures. In addition, *Mathematica* is used to monitor and visualize various numerical algorithms.

The main part of Chapter 1 culminates with two larger applications, the construction of Riemann surfaces of algebraic functions and the visualization of electric and magnetic field lines of some more complicated two- and three-dimensional charge and current distributions. A large, diverse set of exercises and detailed solutions ends the first chapter.

Chapter 2 deals with exact integer calculations and integer-valued functions while concentrating on topics that are important in classical analysis. Number theory functions and modular polynomial functions, currently little used in most natural science applications, are intentionally given less detailed treatment. While some of the functions of this chapter have analytic continuations to complex arguments and could so be considered as belonging to Chapter 3 of the Symbolics volume, emphasis is given to their combinatorial use in this chapter.

This volume explains and demonstrates the use of the numerical functions of *Mathematica*. It only rarely discusses the underlying numerical algorithms themselves. But occasionally *Mathematica* is used to monitor how the algorithms work and progress.

■ The Symbolics Volume

The Mathematica GuideBook to Symbolics has three chapters. Each chapter is subdivided into sections (which have occasionally subsections), exercises, solutions to the exercises, and references.

This fourth and last volume of the *GuideBooks* deals with *Mathematica*'s symbolic mathematical capabilities—the real heart of *Mathematica* and the ingredient of the *Mathematica* software system that makes it so unique and powerful. In addition, this volume discusses and employs the classical orthogonal polynomials and special functions of mathematical physics. To demonstrate the symbolic mathematics power, a variety of problems from mathematics and physics are discussed.

Chapter 1 starts with a discussion of the algebraic functions needed to carry out analysis problems effectively. Contrary to classical science/engineering mathematics education, using a computer algebra system makes it often a good idea to rephrase a problem—including when it is from analysis—in a polynomial way to allow for powerful algorithmic treatments. Gröbner bases play a central role in accomplishing this task. This volume discusses in detail the main functions to deal with structural operations on polynomials, polynomial equations and inequalities, and expressions containing quantified variables. Rational functions and expressions containing trigonometric functions are dealt with next.

Then the central problems of classical analysis—differentiation, integration, summation, series expansion, and limits—are discussed in detail. The symbolic solution of ordinary and partial differential equations is demonstrated in many examples.

As always, a variety of examples show how to employ the discussed functions in various mathematics or physics problems. The Symbolics volume emphasizes the main uses and discuss the specialities of these operations inside a computer algebra system, as compared to a "manual" calculation. Then, generalized functions and Fourier and Laplace transforms are discussed. The main part of the chapter culminates with three examples of larger symbolic calculations, two of them being classic problems. This chapter has more than 150 exercises and solutions treating a variety of symbolic computation examples from the sciences.

Chapters 2 and 3 discuss classical orthogonal polynomials and the special functions of mathematical physics. Because this volume is not a treatise on special functions, it is restricted to selected function groups and present only their basic properties, associated differential equations, normalizations, series expansions, verification of various special cases, etc. The availability of nearly all of the special functions of mathematical physics for all possible arbitrary complex parameters opens new possibilities for the user, e.g., the use of closed formulas for the Green's functions of commonly occurring partial differential equations or for "experimental mathematics". These chapters focus on the use of the special functions in a number of physics-related applications in the text as well as in the exercises. The larger examples deal with are the quartic oscillator in the harmonic oscillator basis and the implementation of Felix Klein's method to solve quintic polynomials in Gauss hypergeometric functions ${}_2F_1$.

The Symbolics volume employs the built-in symbolic mathematics in a variety of examples. However, the underlying algorithms themselves are not discussed. Many of them are mathematically advanced and outside of the scope of the *GuideBooks*.

Throughout the Symbolics volume, the programming and graphics experience acquired in the first two volumes is used to visualize various mathematics and physics topics.

The Books and the Accompanying DVDs

The *GuideBooks* come with a multiplatform DVDs. Each DVD contains the fourteen main notebooks, the hyperlinked table of contents and index, a navigation palette, and some utility notebooks and files. All notebooks are tailored for *Mathematica* 4 and are compatible with *Mathematica* 5. Each of the main notebooks corresponds to a chapter from the printed books. The notebooks have the look and feel of a printed book, containing structured units, typeset formulas, *Mathematica* code, and complete solutions to all exercises. The DVD contains the fully evaluated notebooks corresponding to the chapters of the corresponding printed book (meaning these notebooks have text, inputs, outputs and graphics). The DVD also includes the unevaluated versions of the notebooks of the other three *GuideBooks* (meaning they contain all text and *Mathematica* code, but no outputs and graphics).

Although the *Mathematica GuideBooks* are printed, *Mathematica* is "a system for doing mathematics by computer" [59✱]. This was the lovely tagline of earlier versions of *Mathematica,* but because of its growing breadth (like data import, export and handling, operating system-independent file system operations, electronic publishing capabilities, web connectivity), nowadays *Mathematica* is called a "system for technical computing". The original tagline (that is more than ever valid today!) emphasized two points: doing mathematics and doing it on a computer. The approach and content of the *GuideBooks* are fully in the spirit of the original tagline: They are centered around *doing* mathematics. The second point of the tagline expresses that an electronic version of the *GuideBooks* is the more natural medium for *Mathematica*-related material. Long outputs returned by *Mathematica*, sequences of animations, thousands of web-retrievable references, a 10,000-entry hyperlinked index (that points more precisely than a printed index does) are space-consuming, and therefore not well suited for the printed book. As an interactive program, *Mathematica* is best learned, used, challenged, and enjoyed while sitting in front of a powerful computer (or by having a remote kernel connection to a powerful computer).

In addition to simply showing the printed book's text, the notebooks allow the reader to:

- experiment with, reuse, adapt, and extend functions and code
- investigate parameter dependencies
- annotate text, code, and formulas
- view graphics in color
- run animations.

The Accompanying Web Site

Why does a printed book need a home page? There are (in addition to being just trendy) two reasons for a printed book to have its fingerprints on the web. The first is for (*Mathematica*) users who have not seen the book so far. Having an outline and content sample on the web is easily accomplished, and shows the look and feel of the notebooks (including some animations). This is something that a printed book actually cannot do. The second reason is for readers of the book: *Mathematica* is a large modern software system. As such, it ages quickly in the sense that in the timescale of $10^{1.\,smallInteger}$ month, a new version will likely be available. The overwhelmingly large majority of *Mathematica* functions and programs will run unchanged in a new version. But occasionally, changes and adaptions might be needed. To accommodate this, the web site of this book—http://www.MathematicaGuideBooks.org—contains a list of changes relevant to the *GuideBooks*. In addition, like any larger software project, unavoidably, the *GuideBooks* will contain suboptimal implementations, mistakes, omissions, imperfections, and errors. As they come to his attention, the author will list them at the book's web site. Updates to references, corrections [51✱], additional exercises and solutions, improved code segments, and other relevant information will be on the web site as well. Also, information about OS-dependent and *Mathematica* version-related changes of the given *Mathematica* code will be available there.

Evolution of the Mathematica GuideBooks

A few words about the history and the original purpose of the *GuideBooks*: They started from lecture notes of an *Introductory Course in Mathematica 2* and an advanced course on the *Efficient Use of the Mathematica Programming System*, given in 1991/1992 at the Technical University of Ilmenau, Germany. Since then, after each release of a new version of *Mathematica*, the material has been updated to incorporate additional functionality. This electronic/printed publication contains text, unique graphics, editable formulas, runable, and modifiable programs, all made possible by the electronic publishing capabilities of *Mathematica*. However, because the structure, functions and examples of the original lecture notes have been kept, an abbreviated form of the *GuideBooks* is still suitable for courses.

Since 1992 the manuscript has grown in size from 1,600 pages to more than three times its original length, finally "weighing in" at nearly 5,000 printed book pages with more than:

- 10 gigabytes of accompanying *Mathematica* notebooks
- 20,000 *Mathematica* inputs with more than 10,000 code comments
- 9,000 references
- 4,000 graphics
- 1,000 fully solved exercises
- 100 animations.

This first edition of this book is the result of more than ten years of writing and daily work with *Mathematica*. In these years, *Mathematica* gained hundreds of functions with increased functionality and power. A modern year-2004 computer equipped with *Mathematica* represents a computational power available only a few years ago to a select number of people [57✱] and allows one to carry out recreational or new computations and visualizations—unlimited in nature, scope, and complexity—quickly and easily. Over the years the author has learned a lot of *Mathematica* and its current and potential applications, and has had a lot of fun, enlightening moments and satisfaction applying *Mathematica* to a variety of research and recreational areas, especially graphics. The author hopes the reader will have a similar experience.

Disclaimer

In addition to the usual disclaimer that neither the author nor the publisher guarantees the correctness of any formula, fitness, or reliability of any of the code pieces given in this book, another remark should be made. No guarantee is given that running the *Mathematica* code shown in the *GuideBooks* will give identical results to the printed ones. On the contrary, taking into account that *Mathematica* is a large and complicated software system which evolves with each released version, running the code with another version of *Mathematica* (or sometimes even on another operating system) will very likely result in different outputs for some inputs. And, as a consequence, if different outputs are generated early in a longer calculation, some functions might hang or return useless results.

The interpretations of *Mathematica* commands, their descriptions, and uses belong solely to the author. They are not claimed, supported, validated, or enforced by Wolfram Research. The reader will find that the author's view on *Mathematica* deviates sometimes considerably from those found in other books. The author's view is more on the formal than on the pragmatic side. The author does not hold the opinion that any *Mathematica* input has to have an immediate semantic meaning. *Mathematica* is an extremely rich system, especially from the language point of view. It is instructive, interesting, and fun to study the behavior of built-in *Mathematica* functions when called with a variety of arguments (like unevaluated, hold, including undercover zeros, etc.). It is the author's strong belief that doing this and being able to explain the observed behavior will be, in the long term, very fruitful for the reader because it develops the ability to recognize the uniformity of the principles underlying *Mathematica* and to make constructive, imaginative, and effective use of this uniformity. Also, some exercises ask the reader to investigate certain "unusual" inputs.

From time to time, the author makes use of undocumented features and/or functions from the `Developer`` and `Experimental`` contexts (in later versions of *Mathematica* these functions could exist in the `System`` context or could have different names). However, some such functions might no longer be supported or even exist in later versions of *Mathematica*.

Acknowledgements

Over the decade the *GuideBooks* were in development, many people have seen parts of them and suggested useful changes, additions, and edits. I would like to thank Horst Finsterbusch, Gottfried Teichmann, Klaus Voss, Udo Krause, Jerry Keiper, David Withoff, and Yu He for their critical examination of early versions of the manuscript and their useful suggestions, and Sabine Trott for the first proofreading of the German manuscript. I also want to thank the participants of the original lectures for many useful discussions. My thanks go to the reviewers of this book: John Novak, Alec Schramm, Paul Abbott, Jim Feagin, Richard Palmer, Ward Hanson, Stan Wagon, and Markus van Almsick, for their suggestions and ideas for improvement. I thank Richard Crandall, Allan Hayes, Andrzej Kozlowski, Hartmut Wolf, Stephan Leibbrandt, George Kambouroglou, Domenico Minunni, Eric Weisstein, Andy Shiekh, Arthur G. Hubbard, Jay Warrendorff, Allan Cortzen, Ed Pegg, and Udo Krause for comments on the prepublication version of the *GuideBooks*.

My thanks are due to Gerhard Gobsch of the Institute for Physics of the Technical University in Ilmenau for the opportunity to develop and give these original lectures at the Institute, and to Stephen Wolfram who encouraged and supported me on this project.

Concerning the process of making the *Mathematica GuideBooks* from a set of lecture notes, I thank Glenn Scholebo for transforming notebooks to T_EX files, and Joe Kaiping for T_EX work related to the printed book. I thank John Novak and Jan Progen for putting all the material into good English style and grammar, John Bonadies for the chapter-opener graphics of the book, and Jean Buck for library work. I especially thank John Novak for the creation of *Mathematica* 3 notebooks from the T_EX files, and Andre Kuzniarek for his work on the stylesheet to give the notebooks a pleasing appearance. My thanks go to

Andy Hunt who created a specialized stylesheet for the actual book printout and printed and formatted the 4×1000+ pages of the *Mathematica GuideBooks*. I thank Andy Hunt for making a first version of the homepage of the *GuideBooks* and Amy Young for creating the current version of the homepage of the *GuideBooks*. I thank Sophie Young for a final check of the English. My largest thanks go to Amy Young, who encouraged me to update the whole book over the years and who had a close look at all of my English writing and often improved it considerably. Despite reviews by many individuals any remaining mistakes or omissions, in the *Mathematica* code, in the mathematics, in the description of the *Mathematica* functions, in the English, or in the references, etc. are, of course, solely mine.

Let me take the opportunity to thank members of the Research and Development team of Wolfram Research whom I have met throughout the years, especially Victor Adamchik, Alexei Bocharov, Arnoud Buzing, Brett Champion, Matthew Cook, Todd Gayley, Unal Goktas, Yifan Hu, Daniel Lichtblau, Jerry Keiper, Robert Knapp, Roman Mäder, Oleg Marichev, John Novak, Mark Sofroniou, Adam Strzebonski, Robby Villegas, Tom Wickham–Jones, David Withoff, and Stephen Wolfram for numerous discussions about design principles, various small details, underlying algorithms, efficient implementation of various procedures, and tricks concerning *Mathematica*. The appearance of the notebooks profited from discussions with John Fultz, Paul Hinton, John Novak, Lou D'Andria, Theodore Gray, Christopher Carlson, and Neil Soiffer about front end, button, and typesetting issues.

I am grateful to Jeremy Hilton from the Corporation for National Research Initiatives for allowing the distribution of the text of Shakespeare's *Hamlet* (used in Chapter 1 of *The Mathematica GuideBook to Numerics*).

It was an interesting and unique experience to work over the last 11 years with five editors: Allan Wylde, Paul Wellin, Maria Taylor, Wayne Yuhasz, and Ann Kostant, with whom the *GuideBooks* were finally published. Many book-related discussions that ultimately improved the *GuideBooks*, have been carried out with Jan Benes from TELOS and associates, Steven Pisano, Jenny Wolkowicki, Henry Krell, Fred Bartlett, Ken Quinn, Jerry Lyons, and Rüdiger Gebauer from Springer–Verlag New York.

The author hopes the *Mathematica GuideBooks* help the reader to discover, investigate, urbanize, and enjoy the computational paradise offered by *Mathematica*.

M. Trott

Champaign–Urbana, Illinois, March 2004

References

✱1 A. Amthor. *Zeitschrift Math. Phys.* 25, 153 (1880).

✱2 K. Appel, W. Haken. *J. Math.* 21, 429 (1977).

✱3 A. Bauer, E. Clarke, X. Zhao. *J. Automat. Reasoning* 21, 295 (1998).

✱4 A. H. Bell. *Am. Math. Monthly* 2, 140 (1895).

✱5 M. Berz. *Adv. in Imaging and Electron Phys.* 108, 1 (2000).

✱6 R. F. Boisvert. *arXiv:cs.MS*/0004004 (2000). *Get Preprint*

✱7 B. Buchberger. *Theorema Project* (1997). ftp://ftp.risc.uni-linz.ac.at/pub/techreports/1997/97-34/ed-media.nb

✱8 B. Buchberger. *SIGSAM Bull.* 36, 3 (2002).

✱9 S.-C. Chou, X.-S. Gao, J.-Z. Zhang. *Machine Proofs in Geometry*, World Scientific, Singapore, 1994.

✱10 A. M. Cohen. *Nieuw Archief Wiskunde* 14, 45 (1996).

✱11 A. Cook. *The Motion of the Moon*, Adam-Hilger, Bristol, 1988.

✱12 C. Delaunay. *Théorie du Mouvement de la Lune*, Gauthier-Villars, Paris, 1860.

✱13 C. Delaunay. *Mem. de l' Acad. des Sc. Paris* 28 (1860).

✱14 C. Delaunay. *Mem. de l' Acad. des Sc. Paris* 29 (1867).

✱15 A. Deprit, J. Henrard, A. Rom. *Astron. J.* 75, 747 (1970).

✱16 A. Deprit. *Science* 168, 1569 (1970).

✱17 A. Deprit, J. Henrard, A. Rom. *Astron. J.* 76, 273 (1971).

✱18 P. J. Dolan, Jr., D. S. Melichian. *Am. J. Phys.* 66, 1 (1998).

✱19 S. P. Ferguson, T. C. Hales. *arXiv:math.MG/* 9811072 (1998). *Get Preprint*

✱20 B. Fitelson. *Mathematica Educ. Res.* 7, n1, 17 (1998).

✱21 A. C. Fowler. *Mathematical Models in the Applied Sciences,* Cambridge University Press, Cambridge, 1997.

✱22 H. Fritsch, G. Fritsch. *The Four-Color Theorem*, Springer-Verlag, New York, 1998.

✱23 E. Gallopoulus, E. Houstis, J. R. Rice (eds.). *Future Research Directions in Problem Solving Environments for Computational Science: Report of a Workshop on Research Directions in Integrating Numerical Analysis, Symbolic Computing, Computational Geometry, and Artificial Intelligence for Computational Science*, 1991. http://www.cs.purdue.edu/research/cse/publications/tr/92/92-032.ps.gz

✱24 V. Gerdt, S. A. Gogilidze in V. G. Ganzha, E. W. Mayr, E. V. Vorozhtsov (eds.). *Computer Algebra in Scientific Computing*, Springer-Verlag, Berlin, 1999.

✱25 M. C. Gutzwiller, D. S. Schmidt. *Astronomical Papers: The Motion of the Moon as Computed by the Method of Hill, Brown, and Eckert*, U.S. Government Printing Office, Washington, 1986.

✱26 M. C. Gutzwiller. *Rev. Mod. Phys.* 70, 589 (1998).

✱27 Y. Hagihara. *Celestial Mechanics* vII/1, MIT Press, Cambridge, 1972.

✱28 T. C. Hales. *arXiv:math.MG/* 9811071 (1998). *Get Preprint*

✱29 T. C. Hales. *arXiv:math.MG/* 9811073 (1998). *Get Preprint*

✱30 T. C. Hales. *arXiv:math.MG/* 9811074 (1998). *Get Preprint*

✱31 T. C. Hales. *arXiv:math.MG/* 9811075 (1998). *Get Preprint*

✱32 T. C. Hales. *arXiv:math.MG/* 9811076 (1998). *Get Preprint*

✱33 T. C. Hales. *arXiv:math.MG/* 9811077 (1998). *Get Preprint*

✱34 T. C. Hales. *arXiv:math.MG/* 9811078 (1998). *Get Preprint*

✱35 T. C. Hales. *arXiv:math.MG/*0205208 (2002). *Get Preprint*

✱36 T. C. Hales in L. Tatsien (ed.). *Proceedings of the International Congress of Mathematicians* v. 3, Higher Education Press, Beijing, 2002.

✱37 J. Harrison. *Theorem Proving with the Real Numbers*, Springer-Verlag, London, 1998.

✱38 J. Hermes. *Nachrichten Königl. Gesell. Wiss. Göttingen*, 170 (1894).

✱39 E. N. Houstis, J. R. Rice, E. Gallopoulos, R. Bramley (eds.). *Enabling Technologies for Computational Science*, Kluwer, Boston, 2000.

✱40 E. N. Houstis, J. R. Rice. *Math. Comput. Simul.* 54, 243 (2000).

✱41 M. S. Klamkin (eds.). *Mathematical Modelling*, SIAM, Philadelphia, 1996.

✱42 H. Koch, A. Schenkel, P. Wittwer. *SIAM Rev.* 38, 565 (1996).

✱43 Y. N. Lakshman, B. Char, J. Johnson in O. Gloor (ed.). *ISSAC 1998*, ACM Press, New York, 1998.

✱44 W. McCune. *Robbins Algebras Are Boolean*, 1997. http://www.mcs.anl.gov/home/mccune/ar/robbins/

✱45 E. Mach (R. Wahsner, H.-H. von Borszeskowski eds.). *Die Mechanik in ihrer Entwicklung*, Akademie-Verlag, Berlin, 1988.

✱46 D. A. MacKenzie. *Mechanizing Proof: Computing, Risk, and Trust*, MIT Press, Cambridge, 2001.

✱47 B. M. McCoy. *arXiv:cond-mat*/0012193 (2000). *Get Preprint*

✱48 K. J. M. Moriarty, G. Murdeshwar, S. Sanielevici. *Comput. Phys. Commun.* 77, 325 (1993).

✱49 I. Nemes, M. Petkovšek, H. S. Wilf, D. Zeilberger. *Am. Math. Monthly* 104, 505 (1997).

✱50 W. H. Press, S. A. Teukolsky. *Comput. Phys.* 11, 417 (1997).

✱51 D. Rawlings. *Am. Math. Monthly*. 108, 713 (2001).

✱52 *Problem Solving Environments Home Page*. http://www.cs.purdue.edu/research/cse/pses

✱53 D. S. Schmidt in H. S. Dumas, K. R. Meyer, D. S. Schmidt (eds.). *Hamiltonian Dynamical Systems*, Springer-Verlag, New York, 1995.

✱54 S. Seiden. *SIGACT News* 32, 111 (2001).

✱55 S. Seiden. *Theor. Comput. Sc.* 282, 381 (2002).

✱56 C. Simpson. *arXiv:math.HO*/0311260 (2003). *Get Preprint*

✱57 A. M. Stoneham. *Phil. Trans. R. Soc. Lond.* A 360, 1107 (2002).

✱58 M. Tegmark. *Ann. Phys.* 270, 1 (1999).

✱59 S. Wolfram. *Mathematica: A System for Doing Mathematics by Computer*, Addison-Wesley, Redwood City, 1992.

✱60 S. Wolfram. *The Mathematica Book*, Cambridge University Press and Wolfram Media, Cambridge, 1999.