Review for

Elementary Numerical Computing

with *Mathematica*

by

Jerry B. Keiper and Robert D. Skeel

General Comments

I wish first to comment on the overall quality of the book.

Overall, I was pleasantly surprised with the content of this book. Despite the fact that this is an elementary numerical analysis text (and I have taught the subject 12 times) and even though the topics are more or less the usual ones, I found the point of view refreshing and unique. I approve wholeheartedly of the emphasis on roundoff error and more generally of the topical organization. I do not see how a "pedagogical" ordering could improve the text. I would not delete any of the topics, either.

As one of the authors specializes in differential equations, and since linear systems of equations appear near the end, I looked for a cursory treatment of linear systems, and for a too-technical treatment of differential equations. However, I was again pleasantly surprised, as both topics were well-motivated and well-explained. The differential equations section contains a historical perspective which I have not seen elsewhere. The linear equations section successfully motivates the underlying processes and imparts intuition while presenting well the concept of condition and the structure of modern software. Other books do one or two of these tasks, but do not do all so well.

I would definitely recommend this book over our present text, *Numerical Analysis, fourth edition*, by R. L. Burden and J. D. Faires. The manuscript gives a coherent explanation of the underlying ideas, so that students see how and why formulas are derived, while Burden and Faires gives more of the impression of a notation-driven catalogue of techniques and simple-minded algorithms, with explanations of pitfalls slighted and no reference to considerations present in state-of-the-art software.

A text being aggressively marketed is K. E. Atkinson, *An Introduction to Numerical Analysis, second edition*, Wiley, 1989. Though I am not intimately familiar with that book, the text under review appears to make better use of figures, diagrams and examples, and also has formulas which are more nicely laid out. Also, the manuscript appears to take a more modern point of view regarding certain items.

Other text books we have used are

- G. Forsythe, M. A. Malcolm and C. B. Moler, *Computer Methods for Mathematical Computations*, Prentice–Hall, 1980,
- J. R. Rice, *Numerical Methods, Software, and Analysis*, McGraw–Hill, 1983,
- R. W. Hamming, *Numerical Methods for Scientists and Engineers, second edition*, McGraw–Hill, 1973, and
- *Numerical Mathematics and Computing*, W. Cheney and D. Kincaid, Brooks / Cole, 1980.

Most of these are somewhat dated at present. Forsythe / Malcolm / Moler explains principles of numerical software well, but does not cover the underlying ideas sufficiently thoroughly. Rice is an excellent reference for some items, but our students rebelled against it; perhaps the explanations were not self-contained at a level they could understand. Hamming has an interesting exposition of the derivation of quadrature formulas (and is thus still valuable as a reference), but is definitely dated. The Cheney and Kincaid book is not unreasonable, but certain topics are not given the modern treatment afforded them in the manuscript under review.

Now let me comment on the *Mathematica* part.

In principle, I approve of using a higher level language to stimulate interest and teach underlying ideas without excessive drudgery. Also, *Mathematica* is very powerful and appropriate. Nonetheless, I have the following reservations about *Mathematica* itself.

Mathematica appears to be becoming more popular and more nearly universal, especially within the mathematics community. However, it is a highly commercial product, of significant cost, and is available from only one source. Contrast this with, say, Fortran, which has been standardized, and is available from a wide variety of vendors. It is clear that a product like *Mathematica* is very useful (and would be even more so if it were universal), but what is unclear is whether, under the just mentioned conditions, it will be sufficiently widespread to be the centerpiece of (i.e. inextricably woven into) a numerical analysis textbook. (I grant you that there is nothing similar to *Mathematica* which has the standardization and acceptance of, say Fortran or C.)

At our university, undergraduate students have just received access to a network of Sun workstations, centrally located in the Computer Center, but *Mathematica* is not yet installed. Otherwise, I know of no other easy possibility on campus for access of undergraduates taking a Mathematics Department numerical analysis course to access *Mathematica*. At institutions even less privileged than ours, such access may not be immediately practical. Students conceivably could provide such access themselves, but *Mathematica* does not run well on a personal computer unless substantial extended memory is available, etc.

Regarding the authors' request for an opinion concerning whether *Mathematica* should be a more integral part of the exposition, I would say "no", due to the intrinsic strength of the text and due partially to my aforementioned reservations. (However, thi is a "close call" for me.) Nonetheless, the computer problems are interesting, and I personally would attempt to use the text in conjunction with *Mathematica* (partially as an "experiment"). It would require some caution, as the *Mathematica* portion of the text is not self-contained, even though the introductory section (§1.4) and the tips sprinkled throughout the book are good. There also appears to be a contrast between the assumed sophistication in the underlying mathematics and the assumed sophistication in using *Mathematica*. For example, the particular students in the course I would teach would already know elementary analytical methods for ordinary differential equations (though a review may do them good), but will not have used *Mathematica* before the course; they may end up spending more time learning *Mathematica* than the underlying numerical analysis.

My opinion may change as I become more familiar with *Mathematica* myself. The text has definitely impressed me with its potential power, and the *Mathematica* sections were complete enough to get me started.

Though providing a rich introduction, the text does not have enough material for a two semester course. Do the authors plan to provide a sequel with e.g. an introduction to multivariate root-finding and optimization, an initial (elementary) treatment of partial differential equations, fast Fourier transforms, etc.?

Finally, I will consider the authors' question concerning how to grade assignments written in *Mathematica*. Such programs could be handled similarly to programming assignments in third generation languages. In particular, students could provide printed copies of their programs along with test output (which may translate into "a cleaned up session of ins and outs"), and a written report. Students could be graded according to correctness of the output, programming style, thoroughness of the testing, and clarity and logical organization of the entire report package. Where appropriate, generality and originality could also count. In some computing environments, programs stored on files can also be examined by the instructor at his / her leisure. Also, the instructor could treat the *Mathematica* session the same as an "oral" exam, having the student demonstrate as the instructor watched. For convenience and because an objective record is then available, I will probably use the "ins and outs" and report.

Detailed Comments

1. *p. 63, above "Supplementary Notes"*: The statement "rounding error is usually not a problem in practice" contrasts with the great care taken subsequently in explaining rounding errors and numerical stability. Will the students wonder why that is being done, if it isn't important? The authors may wish to suitably modify the statement. (Students *do* need to know about rounding errors.)
2. Sections 5.4 and 5.5: Interval arithmetic can be used very effectively in many instances in adaptive quadrature; it can be done basically by using interval evaluations to replace the unknown coefficient ξ in $f^{(n)})(\xi)$ in the error term; see G. F. Corliss and L. B. Rall, "Adaptive, Self-Validating Numerical Quadrature", *SIAM J. Sci. Statist. Comput.* **8**, 5 (1985) and later papers. Since interval arithmetic has previously been extensively introduced, the authors may wish to mention this application, and to perhaps modify the statement on p. 165: "As a consequence, *uncertainty* is unavoidable in the accuracy of the result". I feel that the adaptive algorithm itself is more difficult to understand than employing interval arithmetic in it, and the interval arithmetic *does* provide certainty. (Note that, for adaptive quadrature, you need to assume more than just tabular data, anyway. Also see my comment 4 below.)
3. *p. 164, line -7*: "a partial ordering of subintervals as a *heap* (of the heapsort variety)". Is my computer science education lacking? Do you mean ordered linked list? Can this term be defined more specifically?
4. *p. 198, below the first formula*: There are many who would argue with the statement "The Taylor method has a drawback: it is restricted to situations where symbolic differentiation is possible". In particular, it is possible with *automatic differentiation* to simultaneously obtain numerical values of the function and its Taylor coefficients which are free of truncation error. The process is simple enough conceptually to introduce in an elementary class (and indeed I have done so). It is practical in computer languages with operator overloading, and should be possible to do in *Mathematica*. See, for example, G. F. Corliss, "Applications of Differentiation Arithmetic", in *Reliability in Computing*, ed. R. E. Moore, Academic Press, London. (1988), 127–148. (Note that differentiation arithmetic and interval arithmetic are two different things.)
5. *p. 258, last statement in 8.2.1*: If bounds on the derivatives of f (or indeed interval values of f) are available, then a *rigorous and exhaustive* search to isolate all roots within the interval is possible. In fact, if an interval Newton method is used along with extended interval arithmetic, then *no* subdivisions or preliminary search is necessary if f is univariate.
6. *p. 266, last sentence of first paragraph of 8.3.4*: Again, another possibility is automatic differentiation, which should be mentioned.

Additional Small Corrections

Note: I read the book primarily for content, so I may not have caught *all* of the "typos".

1. *p. 54, line 3 of §3.2*: "name given to" instead of "name to".
2. *p. 78, line 3*: "so that it" instead of "so that is".
3. *p. 116, line 5*: "misleading" instead of "misleadingly".
4. *p. 141, the figure and explanation*: The figure is somewhat confusing, or else the words "light" and "dark" have been reversed in the explanation. (Is the figure a photographic "negative" of what it should be?)
5. *p. 142, line 1 of §5.2.3*: "Often times error estimates" sounds colloquial to me. Would "Often, error estimates" be preferable?
6. I assume that the various figures which were drawn by hand will be completed as professionally as the most professionally done ones in the manuscript.
7. *p. 147, fig. 5.11*: the labelling is garbled. Ditto fig. 5.12 on p. 149.
8. *p. 176, line 3*: "to peal off"??? The phrase is colloquial at best. I can imagine the absolute values ringing loudly, or I can imagine us "peeling" the absolute values from the expression, revealing its juicy segments.
9. *p. 182, second line of first displayed formula*: I would put "average V' on $[t, t + \Delta t]$" in braces or brackets.
10. *p. 187, last line*: "magic adding" does not seem to have been fully explained, as promised in a previous section, or else I missed it because it was not sufficiently highlighted.
11. *p. 190, first and subsequent formulas*: Can something other than "e" be used here? I generally think $e \approx 2.71828 > 1$.
12. *p. 212, second formula and subsequent discussion*: I prefer to call $\mathbf{e}_j$ the *j-th coordinate vector*, since I think of a "unit vector" as any vector whose norm is 1.
13. *p. 223, line -2*: The term "ill conditioning" is used without formally being defined (even though "condition number" has been defined). At least a statement of the form "ill conditioning means ..." is in order here.
14. *p. 238, line -7*: "from Section". From *which* section?
15. *p. 252, third formula set*:

$$\text{subtract} \quad 3 \times \{2\} \quad \text{from} \quad \{3\}$$

    instead of

$$\text{subtract} \quad 3 \times \{1\} \quad \text{from} \quad \{3\}$$

    Also, "subtract $-\frac{1}{2}$" instead of "subtract $\frac{1}{2}$"

16. *p. 254, line 2*: "elements of the product are" instead of "elements of the product is".
17. *p. 260, line 3*: "which is a good approximation for $x \approx x_k$" might be better said "which is a good approximation when $x \approx x_k$, as the former could be construed to mean an approximation for x.
18. *p. 270, first line of Theorem 8.5*: $f''(x)$ instead of $f'(x)'$.