



ELSEVIER

Computer Physics Communications 103 (1997) 100–101

---

Computer Physics  
Communications

---

### Book Review

**Numerical Algorithms with Fortran by G. Engeln-Müllges and F. Uhlig (Springer, Berlin, 1996), ISBN 3-540-60529-0**

**Numerical Algorithms with C by G. Engeln-Müllges and F. Uhlig (Springer, Berlin, 1996), ISBN 3-540-60530-4**

Tim Hopkins

*Reader in Numerical Computation, Computing Laboratory, University of Kent, Canterbury, Kent CT2 7NF, UK*

The authors claim in their preface that these books are not meant to be textbooks on Numerical Analysis and, furthermore, that they do not include a single proof. However, this does not mean that there is any shortage of mathematical content, indeed the pages fairly bristle with theorems and mathematical notation. Numerical algorithms may be the final goal but the reader is provided with all the necessary background to understand exactly how and why the presented algorithm actually works.

There is certainly a good breadth of material with each of the eighteen chapters devoted to an area of numerical computation. All the expected subjects are covered along with several more unusual areas like rational interpolation, Akima and Renner subsplines and cubature. Each chapter contains a short introduction to the problem area, usually only a page or two, and a final section, ranging from a few lines to several pages, which provides advice on how to select an algorithm for a particular problem.

The majority of the chapters stop short of describing (and therefore providing algorithms and code for) state of the art algorithms. As an example, consider the coverage of iterative methods for solving linear systems of equations; the main chapter contains details and algorithms for the Jacobi, Gauss–Seidel and

SOR methods, while the conjugate-gradients method is covered in three pages and is only considered in its simplest form for symmetric, positive definite coefficient matrices. The important subject of preconditioning is only mentioned in passing with the reader being referred to a number of other sources; there are no pointers to any of the more reliable, and widely applicable, related methods.

In other cases the advice section appears to be badly out of date. The section on stiff ordinary differential equations does not mention any modern codes or techniques, indeed the majority of algorithms discussed and the recommended bibliography are almost entirely pre 1980.

The accompanying CD contains Fortran 77 and 'experimental' Fortran 90 versions of many of the algorithms discussed in the text along with test driver programs and sample output. By 'experimental' the authors appear to mean that the original Fortran 77 code is presented using Fortran 90's free format; they certainly have not used any of the new features available in Fortran 90 that would have made the software easier to use. Also included on the disk are the Gnu Fortran 77 compiler for OS2 and a number of other tools including a version of **make** and the zip/unzip programs. An HTML index allows the source codes to be

viewed using a web browser.

It would be very difficult to review these books without making at least a passing comparison with the Numerical Recipe series [1, 2]. In fact the two series are very different, the present volumes concentrate on the underlying mathematics while Numerical Recipes provides far more in the way of problem solving advice. To some extent, therefore, the two sets of books complement one another.

To conclude the present volumes provide a good mathematical background to a wide range of numerical algorithms. Unfortunately neither these, nor the Numerical Recipe series, represents the state of the art in numerical computation; both rely far too heavily on the straightforward coding of straightforward methods. Don't look in these software libraries if you want an efficient code for solving large systems of stiff

ordinary differential equations or a quadrature routine capable of handling singularities in the integrand or a means of solving large, sparse, nonsymmetric linear systems. Despite the claims of the authors, neither have produced numerical software capable of handling difficult problems; such problems are still far safer in the hands of the commercial numerical software libraries.

## References

- [1] W. Press et al., *Numerical Recipes in Fortran: The Art of Scientific Computation*, 2nd ed. (CUP, Cambridge, 1992), ISBN 0-521-43064-X.
- [2] W. Press et al., *Numerical Recipes in C: The Art of Scientific Computation*, 2nd (CUP, Cambridge, 1992), ISBN 0-521-43108-5.