

Book review for  
**Numerical Toolbox for Verified Computing I**  
by  
*R. Hammer, M. Hocks, U. Kulisch and D. Ratz*

Although one aspect of this book is to document a software package for scientific computation in PASCAL-XSC, it can also be viewed as a textbook for numerical analysis from a rigorous perspective. As such, it covers many of the usual topics at a level of more advanced numerical analysis texts. Throughout, the underlying principle distinguishing this book is that the results of numerical computations are to be trusted with the same rigor as mathematical results. A main enabling technique is initial computation of an approximate solution, followed by verification using Brouwer's fixed-point theorem and interval arithmetic. Computational tasks for which this approach, and for which environments like PASCAL-XSC are suited appear in the book (and the Toolbox), whereas tasks in which the approach is problematical or not yet fully understood are absent. Thus, accurate evaluation of polynomials and arithmetic expressions, automatic differentiation, deterministic global optimization, rigorous solution of nonlinear systems, accurate, rigorous solution of linear systems and linear optimization appear, whereas solution of differential equations (ordinary or partial) does not. Unfortunately, an exception is numerical quadrature, for which good, widely applicable algorithms for calculations with rigorous error bounds exist, but which does not appear in this book. On the other hand, use of iterative corrections to obtain accurate values of arithmetic expressions, even in the presence of cancellation errors, is a largely unrecognised technique that does appear in the book. Treatment of quadrature, etc. could also appear in a promised second volume.

The book begins with motivation and explanation of automatic result verification and accurate dot products. The second chapter, "features of PASCAL-XSC," contains a good explanation of the universal operator concept, module concept, and dynamic arrays. These concepts are becoming fundamental in elementary computer science or computer programming curricula. Although introduced as features of PASCAL-XSC, all of these features are also available in analogous form in modern programming languages such as C++ or Fortran 90. Less universal features described in this chapter are control of rounding of decimal input data and accurate dot-product expressions. Predefined arithmetic modules for interval, complex, complex interval, matrix, complex matrix, interval matrix, and complex interval matrix arithmetic are also described here.

Regarding choice of PASCAL-XSC, we see that machine-specific or portable versions of extensions to Fortran, C or Pascal developed at Karlsruhe (the authors' institution) include optimally accurate interval arithmetic and an accurate dot product. PASCAL-XSC should be available on any machine with a "C" compiler, although different versions must be obtained from Karlsruhe for

different compilers. A possible disadvantage is that C++ or even Fortran 90 based extensions appear to be more commonly used now than Pascal.

Explanations of mathematical and floating-point interval arithmetic are given in Chapter 3, "Mathematical Preliminaries." This includes selected, elementary properties of interval arithmetic and interval analysis, with an emphasis on the most important properties for scientific computation. Numerous helpful figures and tables appear. Discussion of floating point operations begins at the level usually found in introductory numerical analysis texts. However, the concept of *semimorphism* (a set of desirable properties for floating-point systems) is introduced, and data conversion is presented in a way suitable for *rigorous* use of the Toolbox routines.

The remainder of the book, divided into one-dimensional and multidimensional problems, treats specific scientific computing tasks, one per chapter. Each chapter contains theoretical background, a compact description of the algorithm in diagram form, the PASCAL-XSC code, examples, explanation of scope and proper use, and references. The organization is clear and reasonably complete.

One topic treated in the book and of interest outside the field of automatic result verification is automatic differentiation. Chapters 5 and 12 contain good elementary explanations of the forward mode of automatic differentiation, along with specifics of the elegant implementation in the "toolbox." Both the presentation and implementation treat derivatives only up to second order, although this is adequate for most tasks in numerical linear algebra. A more disturbing omission is a lack of explanation of the consequences of using operator overloading to first produce an internal representation (code list), then performing the arithmetic according to the code list. However, omission this is forgivable since the book's emphasis is on rigorous computations rather than automatic differentiation.

Treatment of nonlinear systems and global optimization is, overall, reasonable.

The overall value of the accurate dot product and, to a lesser extent, the value of semimorphism properties are somewhat controversial. However, this book succeeds in showing how these can be used effectively.

The book also succeeds in systematically presenting underlying theory, then algorithms, then complete computer programs in a very readable style. However, the question remains whether the computer programs themselves are sophisticated enough as general software tools. The book contains many references and suggestions for additional improvements that the reader will be capable of doing after a thorough reading. Nonetheless, although some of the example problems are touted as "real-world," most of the functions are low-dimensional, involve few operations, and are programmable in several lines. This leaves doubt about the value of the programs for functions arising from complicated simulations, etc. It may also give the wrong impression to non-experts, since astute use of interval techniques *is* fruitful in some such applications.

On balance, the book would be valuable to anyone generally interested in

scientific computing. It would also be a reasonable text in a topics course on verified computing.