Interval Computations: Introduction, Uses, and Resources

R. B. Kearfott
Department of Mathematics
University of Southwestern Louisiana
U.S.L. Box 4-1010, Lafayette, LA 70504-1010 USA
email: rbk@usl.edu

Abstract

Interval analysis is a broad field in which rigorous mathematics is associated with with scientific computing. A number of researchers worldwide have produced a voluminous literature on the subject. This article introduces interval arithmetic and its interaction with established mathematical theory. The article provides pointers to traditional literature collections, as well as electronic resources. Some successful scientific and engineering applications are listed.

1 What is Interval Arithmetic, and Why is it Considered?

Interval arithmetic is an arithmetic defined on sets of intervals, rather than sets of real numbers. A form of interval arithmetic perhaps first appeared in 1924 and 1931 in [8, 104], then later in [98]. Modern development of interval arithmetic began with R. E. Moore's dissertation [64]. Since then, thousands of research articles and numerous books have appeared on the subject. Periodic conferences, as well as special meetings, are held on the subject. There is an increasing amount of software support for interval computations, and more resources concerning interval computations are becoming available through the Internet.

In this paper, boldface will denote intervals, lower case will denote scalar quantities, and upper case will denote vectors and matrices. Brackets "$[\cdot]$" will delimit intervals while parentheses "$(\cdot)$" will delimit vectors and matrices. Underscores will denote lower bounds of intervals and overscores will denote upper bounds of intervals. Corresponding lower case letters will denote components of vectors. The set of real intervals will be denoted by $\mathbb{IR}$. Interval vectors will also be called *boxes*.

If $\boldsymbol{x} = [\underline{x}, \overline{x}]$ and $\boldsymbol{y} = [\underline{y}, \overline{y}]$, then the four elementary operations for *idealized interval arithmetic* obey

$$\boldsymbol{x} \text{ op } \boldsymbol{y} = \{x \text{ op } y \mid x \in \boldsymbol{x} \text{ and } y \in \boldsymbol{y}\} \quad \text{for op } \in \{+, -, \times, \div\} \tag{1}$$

Thus, the image of each of the four basic interval operations is the *exact range* of the corresponding real operation. Although Equation (1) characterizes these operations mathematically, interval arithmetic's usefulness is due to the *operational definitions*. For example,

$$\begin{aligned}
\boldsymbol{x} + \boldsymbol{y} &= [\underline{x} + \underline{y}, \overline{x} + \overline{y}], & (2)\\
\boldsymbol{x} - \boldsymbol{y} &= [\underline{x} - \overline{y}, \overline{x} - \underline{y}], & (3)\\
\boldsymbol{x} \times \boldsymbol{y} &= [\min\{\underline{x}\underline{y}, \underline{x}\overline{y}, \overline{x}\underline{y}, \overline{x}\overline{y}\}, \max\{\underline{x}\underline{y}, \underline{x}\overline{y}, \overline{x}\underline{y}, \overline{x}\overline{y}\}] & (4)\\
\frac{1}{\boldsymbol{x}} &= [1/\overline{x}, 1/\underline{x}] \qquad \text{if } \underline{x} > 0 \text{ or } \overline{x} < 0 & (5)\\
\boldsymbol{x} \div \boldsymbol{y} &= \boldsymbol{x} \times 1/\boldsymbol{y} & (6)
\end{aligned}$$

The ranges of the four elementary interval arithmetic operations are exactly the ranges of the corresponding real operations. If such operations are composed, *bounds on the ranges* of real functions can be obtained. For example, if

$$f(x) = x(x - 1), \tag{7}$$

then

$$\boldsymbol{f}([0,1]) = [0,1]\big([0,1] - 1\big) = [0,1][-1,0] = [-1,0],$$

which contains the exact range $[-1/4, 0]$. (This is *necessarily* so.)

Such bounds on ranges can be used throughout mathematical computations in place of Lipschitz constants. In fact, bounds from interval arithmetic often are sharper, and are simpler to derive than bounds from other techniques. For example, if f is as in Equation (7), then the mean value theorem gives $f(x) \in f(0.5) + f'(\xi)(x - 0.5)$ for some unknown ξ between 0.5 and x. If $x \in [0,1]$, then ξ may be replaced by the interval $[0,1]$ to obtain $f'(\xi) \in 2[0,1] - 1 = [-1,1]$. This leads to a second set of bounds on the range of f:

$$f(x) \in -0.25 + [-1,1][-0.5,0.5] = [-0.75, 0.25]$$

for $x \in [0,1]$.

The power of interval arithmetic lies in its implementation on computers. In particular, *outwardly rounded* interval arithmetic allows *rigorous enclosures* for the ranges of operations and functions. This makes a *qualitative* difference in scientific computations, since the results are now intervals in which the exact result *must* lie. It also enables use of computations for automated theorem proving.

Directed rounding proceeds as follows. Much modern computing equipment (including machines that support IEEE standard arithmetic [96], such as most PC's and workstations) allows the result of an arithmetic operation to be *rounded down* to the nearest machine number less than the mathematically correct result, *rounded up* to the nearest machine number greater than or equal to the mathematically correct result, or rounded to the machine number nearest to the mathematically correct result. For example, take $\boldsymbol{x}+\boldsymbol{y} = [\underline{x}+\underline{y}, \overline{x}+\overline{y}]$. If $\underline{x}+\underline{y}$ is rounded down after computation and $\overline{x}+\overline{y}$ is rounded up after computation, then the resulting interval $\boldsymbol{z} = [\underline{z}, \overline{z}]$ that is represented in the machine *must* contain the exact range of $x+y$ for $x \in \boldsymbol{x}$ and $y \in \boldsymbol{y}$.

Good enclosures for the ranges of transcendental functions such as "exp" and "sin" can be computed. Thus, interval arithmetic can be carried out for virtually any expression that can be evaluated with floating point arithmetic. However, since interval arithmetic is only *subdistributive*, expressions that are equivalent in real arithmetic differ in interval arithmetic. In particular, computations should be arranged so that overestimation of ranges is minimized. The fact that naively arranged computations do not always give adequately narrow bounds on the range has been the source of controversy, but there have been advances in recent years in the astute use of interval arithmetic.

The sets of theory and tools are too extensive to be fully described here. Additional introductory and advanced details and explanations of interval arithmetic can be found in the books [1, 2, 3, 6, 27, 28, 45, 66, 73, 83, 84], and soon in the book [40]. A World-Wide-Web entry point for interval computations is

`http://cs.utep.edu/interval-comp/main.html`

More references are cited below.

2 Interval Computations and Mathematical Proofs

A powerful aspect of interval computations is tied to the Brouwer fixed point theorem:

Theorem 1 (Brouwer fixed point theorem, [7]) *Let* $\mathbf{D}$ *be homeomorphic to the closed unit ball in* $\mathbb{R}^n$*, and suppose* P *is a continuous mapping such that the range* $\boldsymbol{P}^u(\mathbf{D}) \subset \mathbf{D}$*. Then* P *has a fixed point, i.e. there is an* $X \in \mathbf{D}$ *such that* $P(X) = X$*.*

The Brouwer fixed point theorem combined with interval arithmetic enables numerical computations to prove existence of solutions to linear and nonlinear systems. The simplest context in which this can be explained is the one-dimensional interval Newton method.

Suppose $f : \boldsymbol{x} = [\underline{x}, \overline{x}] \to \mathbb{R}$ has a continuous first derivative on $\boldsymbol{x}$, suppose $\check{x} \in \boldsymbol{x}$, and suppose $\boldsymbol{f}'(\boldsymbol{x})$ is a set that contains the range of f' over $\boldsymbol{x}$ (such as when f' is evaluated at $\boldsymbol{x}$ with interval arithmetic). Then the operator

$$\boldsymbol{N}(f; \boldsymbol{x}, \check{x}) = \check{x} - f(\check{x})/\boldsymbol{f}'(\boldsymbol{x}) \tag{8}$$

is termed the *univariate interval Newton method.* (Note: The derivative enclosure $\boldsymbol{f}'(\boldsymbol{x})$ may be replaced by a *slope enclosure*; see [40] or [73] for further information and references.) Applying the Brouwer fixed point theorem in the context of the univariate interval Newton method leads to:

Theorem 2 *If $\boldsymbol{N}(f; \boldsymbol{x}, \check{x}) \subset \boldsymbol{x}$, then there exists a unique solution of $f(x) = 0$ in $\boldsymbol{x}$.*

Existence in Theorem 2 follows from *Miranda's theorem*, a corollary of the Brouwer fixed point theorem. For details and references, see [40] or [73]. Uniqueness is as follows: Suppose there were two solutions $x \in \boldsymbol{x}$ and $\tilde{x} \in \boldsymbol{x}$. Then $f(x) = 0 = f(\tilde{x})$, so there is a $\xi \in \boldsymbol{x}$ with $f(x) = f(\tilde{x}) + f'(\xi)(x - \tilde{x}) = f'(\xi)(x - \tilde{x}) = 0$. However, since $\boldsymbol{N}(f; \boldsymbol{x}, \check{x}) \subset \boldsymbol{x}$, $f'(\xi)$ cannot contain zero, so $0 \notin f'(\boldsymbol{x})(x - \tilde{x})$. But this contradicts $0 = f'(\xi)(x - \tilde{x}) \in f'(\boldsymbol{x})(x - \tilde{x})$.

Existence theory for multivariate interval Newton methods is similar. Uniqueness theory proceeds by proving that the interval derivative matrix or interval slope matrix is regular. There are various ways of doing this computationally. For example, if a preconditioned interval version of Gaussian elimination completes without pivots that contain zero, then the interval matrix cannot contain any singular matrices.

This computational existence-uniqueness theory has wide use, from constructing narrow bounds around approximate solutions to linear systems, within which an actual solution must lie, to proving existence and uniqueness of solutions to operator equations.

For a thorough and careful treatment of this theory, see [73]. For an alternate presentation, with an elementary, intuitive introduction and various examples, see [40]. Both [73] and [40] contain numerous historical and research references.

3 Interval Computations and Scientific Computing

Besides computational existence and uniqueness, interval arithmetic provides several other elementary but powerful tools. The most prominent, already mentioned, is bounding the ranges of functions. For example, bounds on the range of an objective function are extremely useful in global optimization algorithms. The second, with wide use in scientific computing, is bounding the error term in Taylor's Theorem (and other approximations with similar error terms). Finally, in some calculations, interval arithmetic (with directed roundings) can be used to bound the effects of roundoff error. However, direct use of interval arithmetic merely to bound roundoff error must be implemented carefully, and cannot be applied naively or, at present, universally.

Although algorithms differ from point (i.e. non-interval) algorithms, interval computations can be used in most of the areas studied in a first course in numerical analysis.

3.1 Linear Systems

Bounding the solution set of an interval linear system is as fundamental in interval computations as in traditional point computations. An interval linear system is a system of the form

$$\boldsymbol{A}X = \boldsymbol{B}, \tag{9}$$

where $\boldsymbol{A} \in \mathbb{IR}^{n\times n}$ and $\boldsymbol{B} \in \mathbb{IR}^n$. The *united solution set* of the interval linear system (9) is that set $\Sigma(\boldsymbol{A},\boldsymbol{B}) \subseteq \mathbb{R}^n$ such that, if $X \in \Sigma(\boldsymbol{A},\boldsymbol{B})$, there exists an $A \in \boldsymbol{A}$ and a $B \in \boldsymbol{B}$ such that $AX = B$.

Computation of the actual solution set $\Sigma(\boldsymbol{A},\boldsymbol{B})$ is an NP-complete problem, but it is a routine matter to compute interval vectors that bound $\Sigma(\boldsymbol{A},\boldsymbol{B})$, and whose overestimation decreases as the widths of the entries in $\boldsymbol{A}$ and $\boldsymbol{B}$ decrease. For example, interval versions of Gaussian elimination or the Gauss–Seidel method provide such bounds. However, these interval algorithms differ significantly from corresponding point algorithms; for instance, Equation (9) must first be preconditioned with a point matrix for the algorithms to be effective. A good recent introduction to these algorithms appears in [28], while a recent detailed study of the properties, with literature citations, appears in [73]. An up-to-date introduction will also appear in [40].

On the practical side, Korn and Ullrich [51] have shown how approximate solutions to point linear systems $AX = B$ can be computed with existing software libraries such as LINPACK or LAPACK, then be used to obtain tight bounds within which an exact solution is known to exist. The rigorous bounds on the exact solution are not too much wider than those obtained with condition number estimators, and, in some cases, can be obtained far less expensively [50].

Schwandt [90, 91, 92] considers using high-performance computers to bound the solution sets of linear interval systems of equations arising from discretization of linear and nonlinear elliptic partial differential equations.

3.2 Nonlinear Systems/Optimization

Because of interval arithmetic's power to bound ranges of functions, interval arithmetic has arguably been most successful in solution of nonlinear systems and global optimization. In global search algorithms for nonlinear systems of the form

$$F(X) = (f_1(X), \ldots, f_n(X))^T = 0, F : \mathbb{R}^n \to \mathbb{R}^n,$$

if $0 \notin \boldsymbol{f}_i(\boldsymbol{X})$, that is, if an interval evaluation of one of the components reveals that it is non-zero, then the box $\boldsymbol{X}$ need not be considered further. Similarly, in global optimization problems of the form

$$\begin{array}{rrcl} \text{minimize} & \phi(X) & & \\ \text{subject to} & c_i(X) & = & 0, \quad i = 1, \ldots, m, \\ & \underline{x}_{i_j} \leq & x_{i_j}, & j = 1, \ldots q - \mu, \\ & x_{i_j} \leq & \overline{x}_{i_j}, & j = \mu + 1, \ldots q, \end{array}$$

a box $\boldsymbol{X}$ can be removed from consideration if the lower bound of an interval evaluation $\boldsymbol{\phi}(\boldsymbol{X})$ is greater than some previously computed point value $\phi(\check{X})$, or if an interval value $0 \notin \boldsymbol{c}_i$ reveals that a constraint cannot be satisfied within $\boldsymbol{X}$.

In both nonlinear systems and global optimization, interval Newton methods are invaluable. Besides providing computational existence/uniqueness proofs that are sharper than, say, the Kantorovich theorem, interval Newton methods provide a quadratically convergent iteration of the form

$$\boldsymbol{X} \leftarrow \boldsymbol{N}(F; \boldsymbol{X}, \check{X}) \cap \boldsymbol{X}.$$

Such iteration is valuable because any solutions of $F(X) = 0$ within $\boldsymbol{X}$ must also be within $\boldsymbol{N}(F; \boldsymbol{X}, \check{X})$.

When interval methods solve nonlinear systems or global optimization problems, their output differs qualitatively from that of point algorithms in the following sense: The output of interval algorithms consists of

1. a list $\mathcal{R}$ such that each box $\boldsymbol{X} \in \mathcal{R}$ has been verified to contain a unique solution, and

2. a list $\mathcal{U}$, each of whose boxes has relative diameter of size less than an input tolerance, such that all solutions within the original search region and not in boxes in $\mathcal{R}$ are in boxes in $\mathcal{U}$.

Typically, the list $\mathcal{R}$ will contain boxes with narrow coordinate widths, corresponding to solutions or global optima at which the Jacobi matrix or Hessian matrix is well-conditioned, and the list $\mathcal{U}$ will contain one or more boxes corresponding to each of the other solutions.

Despite the qualitative advantages of the solutions given by interval methods, interval methods can be faster than point methods (such as homotopy methods, Monte Carlo methods, genetic algorithms) for finding all solutions or global optima, even when the underlying interval arithmetic is not implemented with optimal efficiency.

For details, see (in chronological order): [84], [73], and [28]. The book [27] describes some PASCAL-XSC software. Soon, the book [40] will both introduce the theory and techniques and describe publicly available Fortran 90 software.

Interval *constraint propagation* [32, 57, 100, 101] is closely tied to global optimization and nonlinear systems of equations, but is also related to core subject areas in computer science. The technique involves explicitly using the interrelationships among intermediate quantities in evaluation of algebraic expressions in nonlinear systems, objective functions, and constraints. The associated iteration processes seem to have been rediscovered several times [4, 9, 14, 38].

3.3 Quadrature

Adaptive quadrature is another area in which interval methods have much to offer. This is because, due to the form of the error term, meaningful interval enclosures for the actual integral can easily be computed. Replacing heuristic error estimates by these rigorous enclosures results in a quadrature algorithm that produces *guaranteed* bounds on the actual integral. In particular, quadrature formulas are of the form

$$L(f) = Q(f) + R(F), R(f) = Ch^{n+1}\frac{d^n f}{dx^n}, \tag{10}$$

where L is the integral of f, Q is the quadrature formula to approximate L, $R(f)$ is the local error term, and h is the distance between sample points, for some n and C independent of f. For example, in Simpson's rule,

$$L(f) = \int_{-h}^{h} f(x)dx, \quad Q(f) = \frac{h}{3}\big\{f(-h) + 4f(0) + f(h)\big\}, \quad R(f) = -\frac{1}{90}h^5 f^{(4)}(\xi),$$

for some $\xi \in [-h, h]$. The interval version of this is

$$\int_{-h}^{h} f(x)dx \in \frac{h}{3}\big\{f(-h) + 4f(0) + f(h)\big\} - \frac{1}{90}h^5 \boldsymbol{f}^{(4)}([-h, h]).$$

A more common interval algorithm, however, is to integrate high-order Taylor polynomial approximations to f [13]. For an illustrative example, to integrate $\cos x$ from $x = -0.1$ to $x = 0.1$, we could take the Taylor polynomial with remainder term:

$$\cos(x) = 1 - \frac{x^2}{2} + \frac{x^4}{24}\cos(\xi) \in 1 - \frac{x^2}{2} + \frac{x^4}{24}[0.9, 1].$$

The integral becomes

$$\begin{aligned} \int_{-.1}^{.1} \cos(x)dx &\in 2\left(0.1 - \frac{0.1^3}{6} + \frac{0.1^5}{120}[0.9, 1.1]\right) \\ &\in [\underline{0.199666}817, \underline{0.199666}834] \end{aligned}$$

In actual verified adaptive quadrature codes, automatic differentiation software enables use of high and variable degree Taylor approximations. See [13, 44, 97].

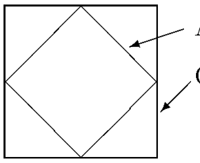

Figure 1: The wrapping effect

3.4 Initial Value Problems

Consider the initial value problem

$$u' = f(t,u), \;\; u(t_0) = u_0, \;\text{ with } u \in \mathbb{R}^n. \tag{11}$$

Interval techniques for (11) provide enclosures for errors in the initial values, mathematical truncation, and roundoff errors, so that, for each time point t_i, intervals are produced that contain the actual solution to (11). However, interval techniques for initial value problems in ordinary differential equations are among the most demanding for algorithm designers. This is due partially to a phenomenon, intrinsic to interval computations, called the *wrapping effect*. The wrapping effect is due to the fact that the image of an interval vector under a map is not an interval vector, and there is thus overestimation in enclosing the image with an interval vector; see Figure 3.4. In Figure 3.4, the interval enclosure has an area that is $\sqrt{2}$ times larger than the area of the actual image set. If the algorithm does not take account of the wrapping effect, the overestimation in the solution increases exponentially from time step to time step.

The wrapping effect can be ameliorated with changes of variables and other techniques, as in Lohner's software AWA [59, 60]. (Also see [74] for a survey and a novel technique.) Nonetheless, interval algorithms for initial value problems tend to be significantly slower than corresponding point algorithms. Furthermore, depending on the stability properties of the system, variable precision arithmetic, along with restarting the iterations, is sometimes used.

Corliss has provided a tutorial [12] on guaranteed error bounds for ordinary differential equations.

3.5 Boundary Value Problems and PDE

Partial differential equations are perhaps the most challenging class of problems for rigorous computation, due to the size of the problems resulting from discretization and due to questions associated with discretization error. As with point methods, systems of PDE's can, in principle, be converted to systems of ordinary differential equations or linear or nonlinear algebraic systems. Considerations such as utilization of structure in the resulting linear or nonlinear interval systems are similar to considerations for point systems. Unique to interval systems is the necessity to rigorously bound the discretization error, so any computational proofs will apply to the original system of differential equations, not just its discretization.

Nonetheless, the discretization error can be taken into account. Kaucher and Miranker [37] develop an arithmetic on functional spaces and corresponding theory for the purpose of computing solutions to function space problems. Kaucher et al. have applied these techniques, e.g. [34].

Lohner has applied AWA, described above, to many boundary value problems in ordinary differential equations. Plum has done extensive work in computational existence and uniqueness proofs, as well as rigorous error bounds for linear and nonlinear elliptic boundary value problems [21, 77, 78, 79, 80, 81]. Nakao [70] has proven existence and uniqueness of solutions to nonlinear elliptic boundary value problems with the aid of finite element discretizations and explicit error bounds.

The book [16] (in Russian) also considers interval techniques for ordinary and partial differential equations.

3.6 Integral Equations

Solution of integral equations combines considerations, as with boundary value problems, of bounding the discretization error for infinite-dimensional problems, with techniques of verified quadrature. In many cases, integral equation formulations are somewhat more tractable than differential equation formulations. See, for example, [15].

4 Some Successful Scientific and Engineering Applications

Widespread application of interval arithmetic has been inhibited in the past by lack of hardware and software. Nonetheless, more real-world applications have appeared in recent years. A relatively early commentary on the use of interval methods in real-world problems is [11]. Since then, use of interval methods has blossomed. More recent, the proceedings [41] contains descriptions of applications in manufacturing quality control, economics, quantum mechanics, and artificial intelligence, as well as fundamental ideas likely to be important in applications. A brief selection of Additional applications are outlined below. The author of this paper wishes to be informed of other successful applications not listed here.

4.1 Chemical Engineering

Technology is currently available to rigorously find all solutions to moderately-sized nonlinear systems of equations [39, 42], an important problem in process design and flowsheeting. Balaji and Seader have used this general technology effectively [5], while Schnepper [87] and Schnepper and Stadtherr [88] have taken advantage of system structure to solve somewhat larger problems.

4.2 Computer Graphics and Computer-Aided Design

Interval computations are well-suited to certain computer graphics applications. Operations such as surface intersection and hidden line removal require robustness in nonlinear equation solvers that can be provided by interval computations. Furthermore, the low-degree polynomial systems and constraints that arise in such operations are easy for interval solvers.

Early, Mudur and Koparkar [69] provide a review of interval analysis techniques that can be useful in computational geometry. Maekawa [61] uses interval-based techniques to robustly solve shape interrogation problems in computational geometry; such solutions are important in automated manufacturing of free-form objects. Others, such as [43, 89, 93], have also shown success in this area.

In a somewhat different application, Enger [17] has shown how to use interval ray tracing to greatly speed up ray tracing algorithms without sacrificing image quality. The basic idea is to take care of regions in the image having nearly constant intensity with a single *interval* ray, rather than with many point rays.

Snyder [94] has provided a relatively recent overview of applications of interval computations to computer graphics.

4.3 Electrical Engineering

Okumura [76] shows that an interval method, besides providing validated results, is hundreds of times faster than a Monte Carlo method for solving AC network equations. Krischuk et al. [52] apply interval computations in quality control in the manufacture of radioelectronic devices.

4.4 Dynamical Systems and Chaos

Grebogi, Hammel, Sauer, and Yorke [10, 22, 26, 86] use techniques related to interval arithmetic to verify, among other things, that computed numerical solutions to chaotic dynamical systems are close to actual solutions with initial conditions that are near the initial condition of the numerical solution.

Neumaier and Rage [75, 82] use interval computations to verify that chaotic behavior occurs in a molecular model. Spreuer and Adams [95] use Lohner's ODE software (see §3.4) to verify existence of homoclinic and heteroclinic orbits of the origin for the Lorenz equations. Mrozek [68] uses interval techniques to determine the qualitative behavior of dynamical systems.

4.5 Control theory

Gross [23], Rohn [85], and others use interval linear algebra to analyze Hurwitz stability, etc. in control theory applications.

4.6 Remote Sensing and Geographic Information Systems

Hager [25] uses interval methods to take account of bounded errors in the data in decisions based on remote sensing. Lodwick [58] uses interval methods in sensitivity analyses in geographic information systems.

4.7 Expert Systems

Kohout et al. [47, 48] develop *interval-valued inference* handle different logical properties of knowledge representations from different medical specialist fields. They apply interval-valued inference to CLINAID, a general medical diagnosis expert system.

4.8 Economics

Jerrell [33] uses classical results in linear interval systems to determine, exactly, the effects of uncertainties in input parameters on the economic output of Coconino County Arizona. Matthews and Broadwater [62] use interval computations to include the effects of forecast uncertainties in break-even analyses for electric utilities.

4.9 Quality Control

Hadjihassan et al. [24] show how to use interval methods for quality control in manufacturing processes in which the factors fluctuate within bounds. They apply the techniques to a realistic model of a temperature controller.

4.10 Correcting Statistical Tables

Wang and Kennedy [102] use interval techniques to discover errors, in some cases in the first or second significant digit of many significant digits printed, in tables of common statistical distributions. They use interval methods to produce tables that are verified to be accurate to many printed digits.

4.11 Computer-Assisted Proofs in Mathematical Physics

Lanford [54, 55, 56] presents a computer-assisted proof of the Feigenbaum conjecture; Koch, Shenkel, and Wittwer [46] review this and give a further analysis.

Fefferman and Seco [18] use interval computations in a computer-assisted proof of an asymptotic formula for the ground-state energy of a nonrelativistic atom.

4.12 Computation of Physical Constants

Holzmann et al. use interval arithmetic to identify critical values whose measurement tolerances must be reduced to determine Newton's gravitational constant G more accurately.

4.13 Minimal Surfaces

Hass [29] uses interval computations in a computer-assisted proof of the solution of a classical variational problem. Namely, he shows that the unique surface of smallest area that encloses two equal volumes is the *double bubble*, made of two pieces of round spheres separated by a disk, meeting along a single circle at an angle of $120°$.

4.14 Fluid Mechanics

Kaucher shows how interval techniques can be used to partially validate some solutions to the incompressible Navier-Stokes equations [34]. More recent, not-yet-published work of Kaucher et al., as well as work of Nakao, Yamamoto and Watanabe [71, 103], contains computational results.

5 Brief Guide to Resources

5.1 Internet Resources

5.1.1 General Internet Pages

A primary entry point to items concerning interval computations is the page:

`http://cs.utep.edu/interval-comp/main.html`

This page contains pointers to much of the information in this article: Namely, it provides an elementary description of interval computations, programming languages for interval computations, home pages of interval computations researchers, information about the journal *Reliable Computing*, bibliographies, etc. This page is maintained by Vladik Kreinovich and Misha Koshelev at the University of Texas at El Paso.

Arnold Neumaier maintains a home page for global optimization (including interval computations) at:

`http://solon.cma.univie.ac.at/~neum/glopt.html`

5.1.2 Bibliographies

- Bohlender has a bibliography, in LaTeX, that contains approximately 1000 entries in BibTeX format, at

  `http://ma70.rz.uni-karlsruhe.de/~ae15/litlist.html`

  Although not comprehensive, this list contains references related to the work at the University of Karlsruhe, as well as other interesting references.

- Nelson Beebe and Jon Rokne maintain a BibTeX bibliography at:

  `ftp://ftp.math.utah.edu/pub/tex/bib/intarith.bib`

  and (the TeX DVI file):

  `ftp://ftp.math.utah.edu/pub/tex/bib/intarith.dvi`

  This bibliography is based upon but extends a two-thousand entry bibliography previously published in the *Freiburger Intervallberichte* [19, 20].

- The Freiburger Intervallberichte bibliography itself is available at:

  `http://solon.cma.univie.ac.at/~neum/intlib`

5.1.3 Software

Software, as follows, is available free of charge over the Internet.

`PROFIL/BIAS` is a C/C++ package, developed by Jansson and Knüppel at Hamburg, that implements an interval data type. It also has substantial support for linear algebra operations, and is notably fast. It is available at:

  `http://www.ti3.tu-harburg.de/indexEnglisch.html`

`INTLIB` is a library of FORTRAN 77 routines for interval arithmetic operations and interval values of standard functions, available at:

  `ftp://interval.usl.edu/pub/interval_math/intlib/`

`INTLIB_90` is a Fortran 90 module that defines an interval data type. It is available at:

  `ftp://interval.usl.edu/pub/interval_math/Fortran_90_software/`

`C-XSC` is one of the well-known "XSC" languages developed under the direction of Prof. Dr. Kulisch at Universität Karlsruhe. The version of the entire package is available free of charge for the Borland C++ compiler version 4.x. For details, see:

  `http://www-iam.mathematik.uni-karlsruhe.de/html/language`

Additional software is available through other distribution channels or for a price. For further information, consult:

`http://cs.utep.edu/interval-comp/main.html`

5.1.4 Mailing List (Discussion Group)

There is an interval mailing list. This amiable discussion group is managed automatically by the "majordomo" software. The list presently consists of roughly 450 subscribers across the world. Items such as conference announcements, problems and solutions, announcements of book publications, and similar information typically are posted here. To send a message to the entire list, send the message to:

`reliable_computing@interval.usl.edu`

To subscribe to the mailing list, send a message to:

`majordomo@interval.usl.edu`

The body of the message should consist of the line:

`subscribe reliable_computing`

Persons may just as easily remove themselves from the list or obtain the email addresses on the list. Details are sent upon subscription to the list.

5.2 Journals

The journal *Interval Computations* started as a joint Soviet-Western enterprise in 1991, and continues as the journal *Reliable Computing.*

Besides that, *Computing* commonly publishes material related to interval computations, as well as the journal *Global Optimization.* Traditional numerical analysis journals, such as *BIT*, the *SIAM Journal on Numerical Analysis*, the *SIAM Journal on Scientific and Statistical Computing*, the *Mathematics of Computation*, and the *ACM Transactions on Mathematical Software* contain articles on interval computations.

5.3 Books

General books on the subject include [1], [2], [3], [6], [16], [27], [28], [37], [45], [65], [66], [73], [83], and [84], soon [40], and others.

Some recent conference proceedings include [30], [35], [36], [41], [53], [63], [67], and [99]. Collections of papers presented at other conferences have appeared in special issues of *Computing* and *Interval Computations/Reliable Computing*.

5.4 Conferences

Numerous conferences on the subject have been held. Perhaps the most well-known of these are the "SCAN" (Society for Computer Arithmetic and Numerics) meetings, generally held biennially in the Fall in Europe, and sponsored by IMACS and GAMM. (The last two have been SCAN'93 in Vienna and SCAN'95 in Wuppertal.) R. E. Moore held a meeting in Columbus, Ohio in 1987. INTERVAL'92 was held in St. Petersburg, Russia, INTERVAL'94 was held in Moscow, and INTERVAL'96 will be held in Würzburg in September, 1996. S. Markov organized conferences on mathematical modeling and scientific computing, heavily featuring interval computations, in Albena, Bulgaria in 1991 and in Sozopol, Bulgaria in 1993. The conference on Numerical Analysis with Automatic Result Verification was held in Lafayette, Louisiana in 1993. A workshop on interval arithmetic will take place in Recife, Brazil in August, 1996. Details of the upcoming conferences can be found from the home page:

`http://cs.utep.edu/interval-comp/main.html`

6 Researchers and Research Centers

Home pages of various researchers can be found from:

`http://cs.utep.edu/interval-comp/main.html`

Researchers not mentioned there should contact Prof. Vladik Kreinovich at:

`vladik@cs.utep.edu`

7 Summary

This paper has briefly introduced the subject of interval computations, and has guided the reader to electronic and printed material for further study and research.

References

[1] O. Aberth. *Precise Numerical Analysis.* Wm. C. Brown, Dubuque, Iowa, 1988.

[2] G. Alefeld and J. Herzberger. *Introduction to Interval Computations.* Wissenschaftsverlag, Mannheim, Wien, Zürich, 1974.

[3] G. Alefeld and J. Herzberger. *Introduction to Interval Computations.* Academic Press, New York, 1983.

[4] A. B. Babichev, O. B. Kadyrova, T. P. Kashevarova, A. S. Leshchenko, and A. L. Semenov. UniCalc, a novel approach to solving systems of algebraic equations. *Interval Computations*, 1993(2):29–47, 1993.

[5] G. V. Balaji and J. D. Seader. Application of interval Newton methods to chemical engineering problems. *Reliable Computing*, 1(3):215–223, 1995.

[6] H. Bauch, K.-U. Jahn, D. Oelschlägel, H. Süsse, and V. Wiebigke. *Intervallmathematik – Theorie und Anwendungen.* Teubner, Leipzig, 1987.

[7] L. E. J. Brouwer. Über Abbildung von Mannigfaltigkeiten. *Math. Ann.*, 71:97–115, 1912.

[8] J. C. Burkill. Functions of intervals. *Proceedings of the London Mathematical Society*, 22:375–446, 1924.

[9] H. M. Chen and M. H. van Emden. Adding interval constraints to the Moore–Skelboe global optimization algorithm. In V. Kreinovich, editor, *Extended Abstracts of APIC'95, International Workshop on Applications of Interval Computations*, pages 54–57. Reliable Computing (Supplement), 1995.

[10] B. A. Cipra. Computer-drawn pictures stalk the wild trajectory. *Science*, 241:1162–1163, September 1988.

[11] G. F. Corliss. Industrial applications of interval techniques. In C. Ullrich, editor, *Contributions to Computer Arithmetic and Self-Validating Numerical Methods*, IMACS Annals on Computing and Applied Math. v. 7, pages 91–113, Basel, 1990. J. C. Baltzer AG.

[12] G. F. Corliss. Guaranteed error bounds for ordinary differential equations. In W. A. Light and M. Marletta, editors, *Theory of Numerics in Ordinary and Partial Differential Equations*, Advances in Numerical Analysis, vol. IV, pages 1–75, London, 1995. Oxford University Press.

[13] G. F. Corliss and L. B. Rall. Adaptive, self-validating numerical quadrature. *SIAM J. Sci. Statist. Comput.*, 8(5):831–47, September 1985.

[14] S. Dallwig, A. Neumaier, and H. Schichl. GLOPT – a program for constrained global optimization, 1996. Preprint, Institut für Mathematik, Universität Wien, Strudlhofgasse 4, A-1090 Wien, Austria.

[15] H.-J. Dobner and E. Kaucher. Inclusion methods for integral equations. In L. Atanassova and J. Herzberger, editors, *Computer Arithmetic and Enclosure Methods. Proceedings of the Third International IMACS-GAMM Symposium (SCAN-91)*, pages 13–24, Amsterdam, Netherlands, 1992. North-Holland.

[16] B. S. Dobronets and V. V. Shaidurov. *Two-Sided Numerical Methods.* Scientific Publishers, Novosibirsk, 1990.

[17] W. Enger. Interval ray tracing – a divide and conquer strategy for realistic computer graphics. *The Visual Computer*, 9:91–104, 1992.

[18] C. L. Fefferman and L. A. Seco. Interval arithmetic in quantum mechanics. In R. B. Kearfott and V. Kreinovich, editors, *Applications of Interval Computations*, Applied Optimization volume 3, pages 145–167, Dordrecht, Netherlands, 1996. Kluwer.

[19] J. Garloff. Interval mathematics: A bibliography. *Freiburger Intervall-Berichte*, 85(6), 1985.

[20] J. Garloff. Bibliography on interval mathematics, continuation. *Freiburger Intervall-Berichte*, 87(2), 1987.

[21] M. Goehlen, M. Plum, and Schröder, J. A programmed algorithm for existence proofs for two-point boundary value problems. *Computing*, 44:91–132, 1990.

[22] C. Grebogi, S.M. Hammel, J.A. Yorke, and T. Sauer. Shadowing of physical trajectories in chaotic dynamics: Containment and refinement. *Physical Review Letters*, 65(13):1527–1530, 1990.

[23] B. Gross. Verification of asymptotic stability for interval matrices and applications in control theory. In E. Adams and U. Kulisch, editors, *Scientific computing with automatic result verification*, pages 357–395, New York, etc., 1993. Academic Press.

[24] S. Hadjihassan, E. Walter, and L. Pronzato. Quality improvement via optimization of tolerance intervals during the design stage. In R. B. Kearfott and V. Kreinovich, editors, *Applications of Interval Computations*, Applied Optimization volume 3, pages 91–131, Dordrecht, Netherlands, 1996. Kluwer.

[25] G. D. Hager. Solving large systems of nonlinear constraints with application to data modeling. *Interval Computations*, 1993(2):169–200, 1993.

[26] S.M. Hammel, J.A. Yorke, and C. Grebogi. Do numerical orbits of chaotic dynamical processes represent true orbits? *Journal of Complexity*, 3(2):136–145, 1987.

[27] R. Hammer, M. Hocks, U. Kulisch, and D. Ratz. *Numerical Toolbox for Verified Computing I.* Springer-Verlag, New York, 1993.

[28] E. R. Hansen. *Global Optimization Using Interval Analysis.* Marcel Dekker, Inc., New York, 1992.

[29] J. Hass and R. Schlafly. Double bubbles minimize, 1995. Preprint, Department of Mathematics, University of California, Davis, Ca 95616.

[30] J. Herzberger, editor. *Topics in Validated Computations*, Studies in Computational Mathematics, Amsterdam, 1994. Elsevier Science Publishers.

[31] O. Holzmann, B. Lang, and H. Schütt. Newton's constant of gravitation and verified numerical quadrature. *Reliable Computing*, 2(3), 1996.

[32] E. Hyvönen. Constraint reasoning based on interval arithmetic. In N. S. Sridharan, editor, *IJCAI-89 Proceedings of the Eleventh International Joint Conference on Artificial Intelligence.*, 1989.

[33] M. E. Jerrell. Applications of interval computations to regional economic input-output models. In R. B. Kearfott and V. Kreinovich, editors, *Applications of Interval Computations*, Applied Optimization, volume 3, pages 133–143, Dordrecht, Netherlands, 1996. Kluwer.

[34] E. Kaucher. Area-preserving mappings, their application for parallel and partially validated computations of Navier-Stokes problems. In E. Kaucher, S. M. Markov, and G. Mayer, editors, *Computer Arithmetic, Scientific Computation and Mathematical Modelling. Proceedings of the Second International Conference*, pages 425–436, Basel, 1990. J. C. Baltzer AG.

[35] E. Kaucher, U. Kulisch, and Ch. Ullrich, editors. *Computer Arithmetic – Scientific Computation and Programming Languages*, Stuttgart, 1987. Teubner.

[36] E. Kaucher, S. M. Markov, and G. Mayer, editors. *Computer Arithmetic, Scientific Computation, and Mathematical Modelling*, IMACS Annals on Computing and Applied Math. 12, Basel, 1991. J. C. Baltzer AG.

[37] E. W. Kaucher and W. L. Miranker. *Self-Validating Numerics for Function Space Problems*. Academic Press, Orlando, 1984.

[38] R. B. Kearfott. Decomposition of arithmetic expressions to improve the behavior of interval iteration for nonlinear systems. *Computing*, 47(2):169–191, 1991.

[39] R. B. Kearfott. Empirical evaluation of innovations in interval branch and bound algorithms for nonlinear algebraic systems, 1994. Accepted for publication in *SIAM J. Sci. Comput.*

[40] R. B. Kearfott. *Rigorous Global Search: Continuous Problems*. Kluwer, Dordrecht, Netherlands, 1996.

[41] R. B. Kearfott and V. Kreinovich, editors. *Applications of Interval Computations*, Applied Optimization, Dordrecht, Netherlands, 1996. Kluwer.

[42] R. B. Kearfott and M. Novoa. Algorithm 681: INTBIS, a portable interval Newton/bisection package. *ACM Trans. Math. Software*, 16(2):152–157, June 1990.

[43] R. B. Kearfott and Z. Xing. Rigorous computation of surface patch intersection curves, 1993. Preprint, Department of Mathematics, Univ. of Southwestern Louisiana, U.S.L. Box 4-1010, Lafayette, La 70504.

[44] R. Kelch. *Ein adaptives Verfahren zur numerischen Quadratur mit automatischer Ergebnisverifikation*. PhD thesis, Universität Karlsruhe, 1989.

[45] R. Klatte, U. Kulisch, A. Wiethoff, C. Lawo, and M. Rauch. *C-XSC; A C++ Class Library for Extended Scientific Computing*. Springer-Verlag, New York, 1993.

[46] H. Koch, A. Schenkel, and P. Wittwer. Computer–assisted proofs in analysis and programming in logic: A case study, 1994. Accepted for publication in *SIAM Rev.*

[47] L. J. Kohout and W. Bandler. Fuzzy interval inference utilizing the checklist paradigm and BK-relational products. In R. B. Kearfott and V. Kreinovich, editors, *Applications of Interval Computations*, Applied Optimization volume 3, pages 291–335, Dordrecht, Netherlands, 1996. Kluwer.

[48] L. J. Kohout and I. Stabile. Interval-valued inference in medical knowledge-based system CLINAID. *Interval Computations*, (3):88–115, 1993.

[49] P. A. Koparkar and S. P. Mudur. Subdivision techniques for processing geometric objects. In R.A. Earnshaw, editor, *Fundamental Algorithms for Computer Graphics. Proceedings of the NATO Advanced Study Institute*, pages 751–801, Berlin, 1986. Springer-Verlag.

[50] C. F. Korn and Ch. Ullrich. Extending linpack by verification routines for linear systems. *Mathematics and Computers in Simulation*, 39(1–2):21–37, 1995.

[51] F. Korn and Ch. Ullrich. Verified solution of linear systems based on common software libraries. *Interval Computations*, (3):116–132, 1993.

[52] V. N. Krishchuk, M. Vasilega, N., and G. L. Kosina. The experience of applying interval calculation for the analysis of the strength characteristics of radio-electronic devices, 1992. Preprint, Zhukovskogo Str., 64, Department of, Radio Designing and Manufacturing, Zaporozhue 330063, Ukraine.

[53] U. W. Kulisch and W. L. Miranker, editors. *A New Approach to Scientific Computation*, New York, 1983. Academic Press.

[54] O. E. Lanford. Computer–assisted proof of the feigenbaum conjecture. *Bull. Amer. Math. Soc.*, 6:427–434, 1982.

[55] O. E. Lanford. Computer-assisted proofs in analysis. *Physica A*, 124:465–470, 1984.

[56] O. E. Lanford. A shorter proof of the existence of the feigenbaum fixed point. *Comm. ACM*, 96:521–538, 1984.

[57] W. A. Lodwick. Constraint propagation, relational arithmetic in AI systems and mathematical programs. *Ann. Oper. Res.*, 21(1-4):143–148, 1989.

[58] W. A. Lodwick, W. Monson, and L. Svoboda. Attribute error and sensitivity analysis of map operations in geographical information systems: Suitability analysis. *Int. J. Geogr. Inf. Syst.*, 4(4):413–428, 1990.

[59] R. Lohner. *Einschließung der Lösung gewöhnlicher Anfangs- und Randwertaufgaben und Anwendungen.* PhD thesis, Universität Karlsruhe, 1988.

[60] R. Lohner. Enclosures for the solutions of ordinary initial and boundary value problems. In J. R. Cash and I. Gladwell, editors, *Computational Ordinary Differential Equations*, pages 425–435, Oxford, 1992. Clarendon Press.

[61] T. Maekawa. *Robust Computational Methods for Shape Interrogation.* PhD thesis, Massachussetts Institute of Technology, Department of Ocean Engineering, 1994.

[62] J. Matthews, R. Broadwater, and L. Long. The application of interval mathematics to utility economic analysis. *IEEE Trans. Power Syst.*, 5(1):177–181, 1990.

[63] K. Meyer and D. Schmidt, editors. *Computer Aided Proofs in Analysis*, IMA vol. in math. and its applications vol. 28, 1989.

[64] R. E. Moore. *Interval Arithmetic and Automatic Error Analysis in Digital Computing.* PhD thesis, Stanford University, October 1962.

[65] R. E. Moore. *Interval Analysis.* Prentice-Hall, Englewood Cliffs, NJ, 1966.

[66] R. E. Moore. *Methods and Applications of Interval Analysis.* SIAM, Philadelphia, 1979.

[67] R. E. Moore, editor. *Reliability in Computing*, New York, 1988. Academic Press.

[68] M. Mrozek. Rigorous numerics of chaotic dynamical systems. In P. Garbaczewski, M. Wolf, and A. Weron, editors, *Chaos - The Interplay Between Stochastic and Deterministic Behaviour*, pages 283–296, New York, 1996. Springer-Verlag.

[69] S. P. Mudur and P. A. Koparkar. Interval methods for processing geometric objects. *IEEE Comput. Graphics and Appl.*, 4(2):7–17, February 1984.

[70] M. T. Nakao. Solving nonlinear elliptic problems with result verification using an H^{-1} type residual iteration. *Computing (Suppl.)*, 9:161–173, 1993.

[71] M. T. Nakao, N. Yamamoto, and Y. Watanabe. Guaranteed error bounds for the finite element solutions of the Stokes problem. In G. Alefeld and A. Frommer, editors, *Scientific Computing and Validated Numerics*, Mathematical Research, Berlin, 1996. Akademie Verlag.

[72] M. Neher. Enclosing solutions of an inverse Sturm–Liouville problem with finite data. *Computing*, 53:379–395, 1994.

[73] A. Neumaier. *Interval Methods for Systems of Equations.* Cambridge University Press, Cambridge, England, 1990.

[74] A. Neumaier. The wrapping effect, ellipsoid arithmetic, stability and confidence regions. *Computing (Suppl.)*, 9:175–190, 1993.

[75] A. Neumaier and T. Rage. Rigorous chaos verification in discrete dynamical systems. *Physica D*, 67:327–346, 1993.

[76] K. Okumura and S. Higashino. A method for solving complex linear equation of AC networks by interval computation. In *Proceedings of IEEE International Symposium on Circuits and Systems - ISCAS '94*, pages 121–124, New York, 1994. IEEE.

[77] M. Plum. Computer-assisted existence proofs for two-point boundary value problems. *Computing*, 46:19–34, 1991.

[78] M. Plum. Existence proofs with error bounds for approximate solutions of weakly nonlinear second-order elliptic boundary value problems. *ZAMM (Z. Angew. Math. Mech.)*, 71, 1991.

[79] M. Plum. Explicit h_2-estimates and pointwise bounds for solutions of second-order elliptic boundary value problems. *J. Math. Anal. Appl.*, 165:36–61, 1992.

[80] M. Plum. Numerical existence proofs and explicit bounds for solutions of nonlinear elliptic boundary value problem. *Computing*, 49:25–44, 1992.

[81] M. Plum. Inclusion methods for elliptic boundary value problems. In J. Herzberger, editor, *Topics in Validated Computations*, Studies in Computational Mathematics, pages 323–380, Amsterdam, Netherlands, 1994. North-Holland.

[82] T. Rage, A. Neumaier, and C. Schlier. Rigorous verification of chaos in a molecular model. *Phys. Rev. E*, 50(4):2682–2688, 1994.

[83] H. Ratschek and J. Rokne. *Computer Methods for the Range of Functions.* Horwood, Chichester, England, 1984.

[84] H. Ratschek and J. Rokne. *New Computer Methods for Global Optimization.* Wiley, New York, 1988.

[85] J. Rohn. An algorithm for checking stability of symmetric interval matrices. *IEEE Trans. Autom. Control*, 41(1):133–136, 1996.

[86] T. Sauer and J. A. Yorke. Rigorous verification of trajectories for the computer simulation of dynamical systems. *Nonlinearity*, 4(3):961–979, 1991.

[87] C. A. Schnepper. *Large Grained Parallelism in Equation-Based Flowsheeting Using Interval Newton / Generalized Bisection Techniques.* PhD thesis, University of Illinois, Urbana, 1992.

[88] C. A. Schnepper and M. A. Stadtherr. Application of a parallel interval Newton / generalized bisection algorithm to equation-based chemical process flowsheeting. *Interval Computations*, 1993(4):40–64, 1993.

[89] P. Schramm. Intersection problems of parametric surfaces in CAGD. *Computing*, 53:355–364, 1994.

[90] H. Schwandt. An interval arithmetic approach for an almost globally convergent method for the solution of the nonlinear Poisson equation. *SIAM J. Sci. Statist. Comput.*, 5(2):427–452, June 1984.

[91] H. Schwandt. The solution of nonlinear elliptic Dirichlet problems on rectangles by almost globally convergent interval methods. *SIAM J. Sci. Statist. Comput.*, 6(3):617–38, July 1985.

[92] H. Schwandt. An interval arithmetic method for the solution of nonlinear systems of equations on a vector computer. *Parallel Comput.*, 4(3):323–337, 1987.

[93] T. W. Sederberg and S. R. Parry. Comparison of three curve intersection algorithms. *Comput. Aided Des.*, 18(1):58–63, 1986.

[94] J. M. Snyder. Interval analysis for computer graphics. *Computer Graphics*, 26(2):121–130, July 1992.

[95] H. Spreuer and E. Adams. On the existence and the verified determination of homoclinic and heteroclinit orbits of the origin for the lorenz equations. *Computing (Suppl.)*, 9:233–246, 1993.

[96] Stevenson, D., chairman, Floating-Point Working Group, Microprocessor Standards Subcommittee. IEEE standard for binary floating point arithmetic (IEEE/ANSI 754-1985). Technical report, IEEE, 1985.

[97] U. Storck. Numerical integration in two dimensions with automatic result verification. In E. Adams and U. Kulisch, editors, *Scientific Computing with Automatic Result Verification*, pages 187–224, New York, etc., 1993. Academic Press.

[98] T. Sunaga. *Theory of an Interval Algebra and its Application to Numerical Analysis*. Gaukutsu Bunken Fukeyu-kai, Tokyo, 1958.

[99] C. Ullrich, editor. *Contributions to Computer Arithmetic and Self-Validating Numerical Methods*, volume 7, Basel, 1990. J. C. Baltzer AG.

[100] P. Van Hentenryck. *Constraint Satisfaction in Logic Programming*. MIT Press, Cambridge, MA, 1989.

[101] P. Van Hentenryck, D. McAllester, and D. Kapur. Solving polynomial systems using a branch and prune approach. Technical Report CS-95-01, Dept. of Comp. Sci., Brown University, 1995.

[102] M. C. Wang and W. J. Kennedy. Self-validating computations of probabilities for selected central and non-central univariate probability functions. *J. Amer. Statist. Assoc.*, 89:878–887, 1994.

[103] Y. Watenabe. *Guaranteed Error Bounds for Finite Element Solutions of the Stokes Equations*. PhD thesis, Kyushu University, Computer Center, 1996.

[104] R. C. Young. The algebra of many-valued quantities. *Math. Ann.*, 104:260–290, 1931.