Verified Solution of Large Systems and Global Optimization Problems

Siegfried M. Rump
Technische Informatik III
TU Hamburg-Harburg
Eißendorfer Straße 38
D-21071 Hamburg

Abstract

Recent results on the solution of large, banded or sparse systems and on global unconstrained optimization problems including verification of the correctness of the result will be discussed. The computing time for the verification for large linear systems is the same as for the decomposition, the total computational effort for verifying the global optimum value is for well-known test examples competitive to the pure floating point algorithms. Computational examples will be demonstrated.

1 Introduction

Numerical algorithms being executed on digital computers in finite precision usually deliver a good approximation to the solution of the given problem, but no verified error bound. Algorithms with result verification are part of numerical analysis. They deliver error bounds for the computed approximate solution with the property that it is verified that a solution exists and possibly is unique within those bounds. This statement is true despite the presence of conversion, rounding and cancellation errors.

The tool for computing those bounds is interval analysis. It is well-known that estimating the error of every single operation (rounding-ε) and putting all those bounds together yields, in principle, a true error bound for the solution. However, it is also well-known that those bounds are frequently very pessimistic, if the algorithm is executable in this way at all. For example, if in Gaussian elimination with pivoting the pivot becomes a number with an error bound so big that it includes 0, the execution must be stopped and no result is delivered.

On the other hand it is a fundamental and very interesting property of interval analysis that bounding the range of a codeable function is possible without any auxiliary knowledge about the function such as, for example, Lipschitz continuity. In the following we will show how this property can be used to design algorithms which reduce the overestimation due to data dependencies to a very low level. Sometimes those algorithms are even faster than

their floating point equivalents. We will restrict our attention to systems of equations with dense and with sparse Jacobian and to global optimization problems. Theory and algorithms for many other standard problems in numerical analysis have been published (see [3], [38], [32] and the literature cited there). Many basic principles can be explained in the above mentioned three areas; therefore we restrict our attention to those.

2 Basic principles

There are different representations for intervals of numbers, vectors and so forth. For example, the classical notation of absolute errors is

$$a \pm \Delta a := \{\, \widetilde{a} \in \mathbb{R} \mid a - \Delta a \leq \widetilde{a} \leq a + \Delta a \,\}.$$

Arithmetical operations like addition and multiplication are defined by

$$(a \pm \Delta a) + (b \pm \Delta b) := (a + b) \pm (\Delta a + \Delta b)$$

$$(a \pm \Delta a) \cdot (b \pm \Delta b) := a \cdot b \pm (|a| \cdot \Delta b + \Delta a \cdot |b| + \Delta a \cdot \Delta b).$$

Notice that these estimations are always *worst case estimations*. For practical applications this representation introduces an unnecessary overestimation, especially for wide intervals. This is because the midpoint of the product or quotient of two intervals does not necessarily coincide with the product or quotient of the midpoint. For example

$$(2 \pm 1) \cdot (4 \pm 1) = 8 \pm (2 + 4 + 1) = 8 \pm 7$$

where taking some $\widetilde{a} \in 2 \pm 1$ and $\widetilde{b} \in 4 \pm 1$ the minimum and maximum products are 3 and 15. Therefore usually a lower bound/upper bound representation of intervals is preferred:

$$A = [a_1, a_2] := \{\, a \in \mathbb{R} \mid a_1 \leq a \leq a_2 \,\}.$$

Then

$$[1, 3] \cdot [3, 5] = [3, 15]$$

with no overestimation. The basic arithmetic operations $+, -, \cdot, /$ for intervals can easily be defined where the lower and upper bound can be computed directly from the bounds of the operands (see [3], [32]). Also, it follows that the diameter of the sum and the difference of two intervals A and B is always equal to the *sum* of the diameters:

$$\text{diam}\ (A + B) = \text{diam}\ (A) + \text{diam}\ (B) \quad \text{and}$$

$$\text{diam}\ (A - B) = \text{diam}\ (A) + \text{diam}\ (B).$$

Therefore the only possibility of diminishing diameters of intervals is the multiplication with a small factor or dividing by a large number. We have to use this frequently in the following.

The most basic and fundamental principle of all interval operations is the *isotonicity*. This means given two intervals A, B we have for dyadic operations $\circ$

$$\forall\, a \in A \quad \forall\, b \in B : a \circ b \in A \circ B \tag{2.1}$$

and for monadic operations σ

$$\forall\, a \in A : \sigma(a) \in \sigma(A). \tag{2.2}$$

Interval operations are not restricted to the four basic operations. Transcendental operations can be executed for intervals as well, always regarding the isotonicity (2.1) and (2.2). For example

$$A = [a_1, a_2] \Rightarrow \exp(A) = [\exp(a_1), \exp(a_2)]$$

which is clear because of the monotonicity of the exponential function. But also non-monotonic functions like sine, sinh, Γ ... can be executed over intervals using a power series expansion and estimating the remainder term or by using some case distinctions. In the practical implementation using floating point numbers on the computer proper rounding has to be used (cf. [6], [24]).

With these observations we can already estimate the range of a function over a domain without any further knowledge about the function. Let, for example, $f(x) = e^x - 2x - 1$ and $X = [-1, 1]$. Then

$$f(X) = \{\, f(x) \mid x \in X \,\} \subseteq e^X - 2X - 1 = [e^{-1}, e^1] - [-2, 2] - 1 = [e^{-1} - 3, e^1 + 1]$$
$$\subseteq [-2.64, 3.72]. \tag{2.3}$$

Using auxiliary knowledge we see that there is a minimum of f within X at $\widetilde{x} = \ln 2$, therefore

$$f(X) = f(-1) \underline{\cup} f(1) \underline{\cup} f(\ln 2) \subseteq [-0.39, 1.37],$$

where $\underline{\cup}$ denotes the convex union. For more complicated functions such an analysis may become involved; the more it is amazing that in (2.3) we obtained a rigorous estimation of the range in a very simple way. However, we also see that the range can be severely overestimated. We will see how this overestimation can be reduced and how the degree of overestimation itself can be estimated.

These observations already lead us to a basic rule for verification algorithms, that is to use

as *much* floating point operations as possible and (2.4)

as *few* interval operations as necessary.

This is very much in the spirit of Wilkinson, who wrote in 1971 [44]

"In general it is the best in algebraic computations to leave the use of interval arithmetic as late as possible so that it effectively becomes an a posteriori weapon." (2.5)

For the following we need the fact that interval vectors and interval matrices can be defined as well as operations over those. An interval vector, for example, can be regarded as the cartesian product of the component intervals. We do not want to go into detail but refer to the literature [3], [32]. Also, we only mention that interval operations satisfying (2.1) and (2.2) are very effectively implementable on digital computers [21], [20], [22], [23]. These packages written in C are available via anonymous ftp from the author's institute.

3 Dense Systems of Equations

Let $f : D \subseteq \mathbb{R}^n \to \mathbb{R}^n$, $f \in C^1$ be given and define

$$g : D \subseteq \mathbb{R}^n \to \mathbb{R}^n \quad \text{with} \quad g(x) := x - R \cdot f(x) \tag{3.1}$$

for some $R \in \mathbb{R}^{n \times n}$. That is we locally linearize f where the application of g represents one step of a simplified Newton iteration. For $\emptyset \neq X \subseteq \mathbb{R}^n$ being compact and convex and

$$g(X) \subseteq X \tag{3.2}$$

Brouwer's Fixed Point Theorem implies the existence of a fixed point $\widehat{x} \in X$ of g, i.e. $g(\widehat{x}) = \widehat{x}$. This yields $R \cdot f(\widehat{x}) = 0$ and if we can verify the regularity of R then $\widehat{x}$ is a zero of f within X. Trying to verify (3.2) by means of

$$g(X) \subseteq X - R \cdot f(X) \overset{!}{\subseteq} X$$

does not work unless the term $R \cdot f(X)$ vanishes completely. Therefore we expand f around some $\widetilde{x} \in D$ using the n-dimensional Mean Value Theorem for all $x \in D$ such that $\widetilde{x} \underline{\cup} x \subseteq D$:

$$f(x) = f(\widetilde{x}) + J \cdot (x - \widetilde{x}) \quad \text{where} \quad J_{i*} = \frac{\partial f_i}{\partial x}(\zeta_i),\ \zeta_i \in x \underline{\cup} \widetilde{x}. \tag{3.3}$$

Such an expansion can be used and implemented on the computer for two reasons:

1) The partial derivatives can be computed very effectively by means of automatic differentiation in a forward or backward mode (see [35], [42], [12]). In backward mode the computational costs for computing the whole Jacobian matrix is at most 5 times the costs for 1 function evaluation. This holds independently of the dimension n.

2) The unknown interior points ζ_i can be surpassed by replacing ζ_i by $x \underline{\cup} \widetilde{x}$ and using interval operations to calculate an interval matrix $J(\widetilde{x} \underline{\cup} x)$ containing J. In this case
$$f(x) \in f(\widetilde{x}) + J(x \underline{\cup} \widetilde{x}) \cdot (x - \widetilde{x}). \tag{3.4}$$

We can use this to derive an inclusion formula. If $\widetilde{x} \underline{\cup} X \subseteq D$ then

$$
\begin{aligned}
g(x) &= x - R \cdot f(x) = x - R \cdot \{f(\widetilde{x}) + J \cdot (x - \widetilde{x})\} \\
&= \widetilde{x} - R \cdot f(\widetilde{x}) + \{I - R \cdot J\} \cdot (x - \widetilde{x}) \\
&\subseteq \widetilde{x} - R \cdot f(\widetilde{x}) + \{I - R \cdot J(\widetilde{x} \underline{\cup} X)\} \cdot (X - \widetilde{x})
\end{aligned} \tag{3.5}
$$

for all $x \in X$. The last term in (3.5) is the Krawczyk operator [26]. It can effectively be used to check $g(X) \subseteq X$ because

- the first part $\widetilde{x} - R \cdot f(\widetilde{x})$ is a real vector, no overestimation occurs
- the potential overestimation in the last part is strongly diminished because
  - for $R \approx J(\widetilde{x})^{-1}$ and small diameter of $\widetilde{x} \underline{\cup} X$ the first factor $I - R \cdot J(\widetilde{x} \underline{\cup} X)$ becomes small and
  - for $\widetilde{x} \approx \widehat{x}$ the second factor $X - \widetilde{x}$ becomes also small.

Thus the *only part* where overestimation may occur is the product of two small terms and therefore very small. $J(\widetilde{x} \underline{\cup} X)$ can be replaced by (cf. [14], [2])

$$
J(\widetilde{x}, X)_{ij} := \frac{\partial f_i}{\partial x_j}(X_1, \ldots, X_{j-1}, (\widetilde{x} \underline{\cup} X)_j, \widetilde{x}_{j+1}, \ldots, \widetilde{x}_n). \tag{3.6}
$$

The sharper our J the better an algorithm works. Furthermore, it is superior not to include the solution itself but the difference to an approximate solution [37]. With $Y := X - \widetilde{x}$ we get from (3.5) and (3.6)

$$
g(x) - \widetilde{x} \subseteq -R \cdot f(\widetilde{x}) + \{I - R \cdot J(\widetilde{x}, X)\} \cdot Y
$$

and therefore

$$
-R \cdot f(\widetilde{x}) + \{I - R \cdot J(\widetilde{x}, X)\} \cdot Y \subseteq Y \Rightarrow g(X) \subseteq X. \tag{3.7}
$$

This the first part. It remains to show the regularity of the matrix R. This can be done using the following lemma [38].

Lemma 3.1. Let $z \in \mathbb{R}^n$, $\boldsymbol{C} \subseteq \mathbb{R}^{n \times n}$ and $X \in \mathbb{IR}^n$. Then

$$
z + \boldsymbol{C} \cdot X \subseteq \operatorname{int}(X) \tag{3.8}
$$

implies $\rho(|C|) < 1$ for all $C \in \boldsymbol{C}$.

Applying this to (3.7) with $z = -R \cdot f(\widetilde{x})$ and $\boldsymbol{C} := I - R \cdot J(\widetilde{x}, X)$ yields the regularity of R and every $M \in J(\widetilde{x}, X)$. This is because for $C := I - A$ with $\rho(C) < 1$ a singular matrix A would imply an eigenvalue 1 of $I - A$. Combining our results already yields an inclusion theorem for systems of nonlinear equations.

Theorem 3.2. Let $f : D \subseteq \mathbb{R}^n \to \mathbb{R}^n$, $f \in C^1$ be given, $\widetilde{x} \in D$, $X \in \mathbb{IR}^n$, such that $\widetilde{x} \underline{\cup} (\widetilde{x} + X) \subseteq D$ and $R \in \mathbb{R}^{n \times n}$. If

$$Y := -R \cdot f(\widetilde{x}) + \{I - R \cdot J(\widetilde{x}, \widetilde{x} + X)\} \cdot X \subseteq \mathrm{int}(X) \tag{3.9}$$

using J defined by (3.6), then R and every matrix $M \in J(\widetilde{x}, \widetilde{x} + X)$ are regular and there is an $\widehat{x} \in \widetilde{x} + Y$ with $f(\widehat{x}) = 0$.

Proof. By (3.7) follows $g(\widetilde{x} + X) \subseteq \widetilde{x} + X$ and therefore the existence of a fixed point $\widehat{x} \in \widetilde{x} + X$ of $g(x) = x - R \cdot f(x)$ with $g(\widehat{x}) = \widehat{x}$. By lemma 3.1 follows the regularity of R and therefore $f(\widehat{x}) = 0$. (3.7) implies $\widehat{x} \in \widetilde{x} + Y$. ■

There are many generalizations and improvements of theorem 3.2 as well as further assertions. For example, $\subseteq \mathrm{int}(X)$ can be replaced by $\subsetneqq$ which means inclusion and componentwise inequality, the inclusion step (3.9) can be replaced by an Einzelschrittverfahren, the matrices J can be replaced by slopes [41], [2], and more. These steps shrink the diameter of the left hand side of (3.9) and make the condition (3.9) more likely to hold.

In order to *find* an interval vector X satisfying (3.9) an iteration can be applied, that is the Y in (3.9) is used as the next X. Applying this to (3.8) it is important to perform a slight inflation in every step, the so-called ε-inflation [37]. In the simplest case it can be defined by

$$X \in \mathbb{IIR}^n: \quad X \circ \varepsilon := X + [-\varepsilon, \varepsilon] \quad \text{for } 0 < \varepsilon \in \mathbb{R}.$$

This yields the following iteration for given $X^0 \in \mathbb{IIR}^n$:

$$Y^k := X^k \circ \varepsilon; \quad X^{k+1} := z + \boldsymbol{C} \cdot Y^k.$$

The remarkable about this iteration is that it produces some $X := Y^k$ satisfying (3.8) iff the absolute value of every $C \in \boldsymbol{C}$ is convergent:

$$\exists\, k \in \mathbb{N}: \; z + \boldsymbol{C} \cdot Y^k \subseteq \mathrm{int}(Y^k) \quad \textit{if and only if} \quad \rho(|C|) < 1 \text{ for all } C \in \boldsymbol{C}.$$

This holds for every starting vector and was proved in [39]. The inflation is called ε-inflation and was introduced in [37]. In practical applications, as a matter of experience there is not too much difference between requiring $\rho(C) < 1$ or $\rho(|C|) < 1$, at least for the matrices occuring in (3.9). The major difference compared to a residual iteration

$$x^{k+1} := x^k + R \cdot (b - Ax^k),$$

which is known to converge for every starting value if and only if $\rho(I - RA) < 1$, is that in floating point computation *convergence* cannot be detected. An inclusion algorithm verifies all of its results.

Another major improvement over theorem 3.2 is the possibility to estimate the overestimation of the computed solution. Let $F : \mathbb{R}^k \times \mathbb{R}^n \to \mathbb{R}^n$ be a parametrized function then a theorem similar to 3.2 can be given for including a zero of $f(c,x)$ for all parameters $c \in C \in \mathbb{IR}^k$. That is for all $\widetilde{c} \in C$ the inclusion interval $\widetilde{x} + Y$ contains one and only one zero of $f_{\widetilde{c}}(x) = f(\widetilde{c}, x)$. The true set of zeros Σ defined by

$$\Sigma := \{\, x \in \widetilde{x} + Y \mid \exists\, c \in C : f(c,x) = 0 \,\}$$

is usually an odd-shaped, nonconvex region in $\mathbb{R}^n$. Nevertheless we can define the elongation of the i-th component of x in Σ by

$$\inf_{x \in \Sigma} x_i \qquad \text{and} \qquad \sup_{x \in \Sigma} x_i$$

and ask for bounds on these quantities. These bounds can be calculated by means of an inclusion formula like (3.9). The precise formulation for systems of nonlinear equations requires a little bit of formalism (see [40]). Therefore we state it for linear systems.

Theorem 3.3. Let $\boldsymbol{A} \in \mathbb{IR}^{n \times n}$, $\boldsymbol{b} \in \mathbb{IR}^n$ be given and define

$$\Sigma(\boldsymbol{A}, \boldsymbol{b}) := \{\, x \in \mathbb{R}^n \mid \exists\, A \in \boldsymbol{A}\ \exists b \in \boldsymbol{b} : Ax = b \,\}.$$

Let $R \in \mathbb{R}^{n \times n}$, $\widetilde{x} \in \mathbb{R}^n$, $X \in \mathbb{IR}^n$ and

$$Z := R \cdot (\boldsymbol{b} - \boldsymbol{A} \cdot \widetilde{x}) \in \mathbb{IR}^n, \quad \Delta := \{I - R \cdot \boldsymbol{A}\} \cdot X.$$

If

$$Z + \Delta \subseteq \operatorname{int}(X)$$

then R and every matrix $A \in \boldsymbol{A}$ are regular and for every $b \in \boldsymbol{b}$ the unique solution of the linear system $Ax = b$ satisfies $A^{-1}b \in \widetilde{x} + Z + \Delta$. Moreover for all i, $1 \le i \le n$

$$\begin{aligned} &\widetilde{x}_i + \inf(Z)_i + \inf(\Delta)_i \le \inf_{x \in \Sigma} x_i \le \widetilde{x}_i + \inf(Z)_i + \sup(\Delta)_i \quad \text{and} \\ &\widetilde{x}_i + \sup(Z)_i + \inf(\Delta)_i \le \sup_{x \in \Sigma} x_i \le \widetilde{x}_i + \sup(Z)_i + \sup(\Delta)_i. \end{aligned} \tag{3.10}$$

The bounds (3.10) are outer *and inner* inclusions on the solution complex Σ. The quality is exactly the diameter of Δ, which is small if the diameters of $\boldsymbol{A}$ and X are small and $R \approx \operatorname{mid}(\boldsymbol{A})^{-1}$. The quality of the inner and outer inclusions is demonstrated by the following example. Let $A \in \mathbb{R}^{n \times n}$ with

$$A_{ij} := \left(\frac{i+j}{p}\right) \quad \text{for} \quad p = n + 1 \text{ prime}.$$

Here $\left(\frac{k}{p}\right)$ denotes the Legendre symbol

$$\left(\frac{k}{p}\right) := \begin{cases} 0 & \text{if } k|p \\ 1 & \text{if } k \equiv c^2 \bmod p \text{ for some } c \\ -1 & \text{otherwise.} \end{cases}$$

The example is taken from the Gregory/Karney collection of test matrices [11]. We choose this example to have a reproducable, dense test system. We computed the right hand side b such that the true solution $x = A^{-1}b$ becomes

$$x_i = \frac{(-1)^{i+1}}{i} \quad 1 \le i \le n.$$

Next we introduce relative perturbations for the matrix and the right hand side

$$\boldsymbol{A} := A \cdot (1 \pm e) \;\text{ and }\; \boldsymbol{b} := b \cdot (1 \pm e) \;\text{ with }\; e := 10^{-5}.$$

The computation is executed in single precision equivalent to approximately 7 decimals. We took $n = 1008$.

Inner and outer inclusions for some solution components								$\frac{\text{diam}(X)}{\text{diam}(Y)}$
[0.999 873,	1.000 127]	$\subseteq$	$\Sigma([A],[b])_1$	$\subseteq$	[0.999 869,	1.000 131]		0.96980
[−0.500 127,	−0.499 873]	$\subseteq$	$\Sigma([A],[b])_2$	$\subseteq$	[−0.500 131,	−0.499 869]		0.96975
[0.333 206,	0.333 460]	$\subseteq$	$\Sigma([A],[b])_3$	$\subseteq$	[0.333 203,	0.333 464]		0.96978
			...					
[−0.001 121,	−0.000 867]	$\subseteq$	$\Sigma([A],[b])_{1006}$	$\subseteq$	[−0.001 125,	−0.000 863]		0.96979
[0.000 866,	0.001 120]	$\subseteq$	$\Sigma([A],[b])_{1007}$	$\subseteq$	[0.000 862,	0.001 124]		0.96981
[−0.001 119,	−0.000 865]	$\subseteq$	$\Sigma([A],[b])_{1008}$	$\subseteq$	[−0.001 123,	−0.000 861]		0.96977

The numbers are to be read as follows. Take, for example, the solution component 1008. Then there are linear system data $A \in \boldsymbol{A}$, $b \in \boldsymbol{b}$ within the tolerances such that the 1008^{th} component of the true solution $\widehat{x} = A^{-1}b$ equals the inner bounds but cannot go beyond the outer bounds:

$$\exists\, A \in \boldsymbol{A}\ \exists\, b \in \boldsymbol{b} : (A^{-1}b)_{1008} = -0.001119$$

$$\exists\, A \in \boldsymbol{A}\ \exists\, b \in \boldsymbol{b} : (A^{-1}b)_{1008} = -0.000865$$

$$\forall\, A \in \boldsymbol{A}\ \forall\, b \in \boldsymbol{b} : -0.001123 \le (A^{-1}b)_{1008} \le -0.000861.$$

In order to estimate the quality of the inner and outer inclusions we gave in the last column of the table the ratio of the diameters of the inner and outer inclusions. For example, the last number means that the diameter of the inner inclusion is 96.977 % of the outer one. The worst ratio of diameters was achieved for the 116^{th} component. Here we have

Inner and outer inclusions for some solution components	$\frac{\mathrm{diam}(X)}{\mathrm{diam}(Y)}$
$[-0.008\,741,\ -0.008\,494] \subseteq \Sigma([A],[b])_{116} \subseteq [-0.008\,751,\ -0.008\,490]$	0.96967

For many practical considerations this means almost equality; we know the elongation of the solution complex Σ up to 3 %. When changing the relative perturbation into absolute perturbation of 10^{-5} the numbers above change only slightly. That means changing the zeros in $\boldsymbol{A}$ does hardly affect the sensitivity of the system.

Finally we give a nonlinear example discussed in [1]. Consider a discretization of the boundary value problem $3\ddot{x}x - \dot{x}^2 = 0$, $x(0) = 0$, $x(1) = 20$:

$$f_1 = 3x_1(x_2 - 2x_1) + x_2^2/4,$$
$$f_i = 3x_i(x_{i+1} - 2x_i + x_{i-1}) + (x_{i+1} - x_{i-1})^2/4, \qquad \text{for } 2 \le i \le n-1,$$
$$f_n = 3x_n(20 - 2x_n + x_{n-1}) + (20 - x_{n-1})^2/4.$$

The exact solution is $x(t) = 20 \cdot t^{3/4}$. For

$$n = 400, \quad \widetilde{x}_i \equiv 10.0 \quad \text{for } 1 \le i \le 400,$$

a fairly poor initial approximation, we performed some steps of a Newton iteration and obtained the following inclusions:

$$\begin{aligned}
X_1 &= [\ 0.206611908273,\ \ 0.206611908274] \\
X_2 &= [\ 0.360737510102,\ \ 0.360737510104] \\
X_3 &= [\ 0.495574119032,\ \ 0.495574119035] \\
&\ \ \vdots \\
X_{398} &= [19.9249135121276,\ 19.9249135121282] \\
X_{399} &= [19.9624596920554,\ 19.9624596920557] \\
X_{400} &= [19.9999823472852,\ 19.9999823472853]
\end{aligned}$$

This computation was performed in double precision equivalent to approximately 17 decimals. All inclusions of the solution components coincide to at least 11 decimals. It should be stressed that we enclosed the solution of the discretized problem, not of the continuous problem. The latter class of problems is considered for example by Lohner [29], Nakao [31] and Plum [34].

4 Sparse systems of equations

In the last example of the previous chapter the function f_i only depends on the variables x_{i-1}, x_i and x_{i+1}. That means, the Jacobian is a tridiagonal matrix whereas its inverse is full. Thus for such sparse systems one should look for another way to obtain an inclusion in order to avoid a full approximate inverse R. We consider (3.9):

$$\forall\ M \in J(\widetilde{x}, \widetilde{x}+X) \quad \forall\ x \in X:\ -R\cdot f(\widetilde{x}) + \{I - R\cdot M\}\cdot x \in \text{ l.h.s. (3.9)}$$

and replace $R\ [\approx \mathrm{mid}(J(\widetilde{x}, \widetilde{x}+X))^{-1}]$ by $U^{-1}L^{-1}$, i.e. rather than computing an approximate inverse of the midpoint of $J(\widetilde{x}, \widetilde{x}+X)$ we consider an LU-decomposition. Then we have

$$\forall\ M \in J(\widetilde{x}, \widetilde{x}+X)\ \forall\ x \in X: U^{-1}L^{-1}\cdot\{-f(\widetilde{x}) + (LU-M)\cdot x\} \in \text{l.h.s. (3.9)} \qquad (4.1)$$

Replacing U^{-1} and L^{-1} by a backward and a forward substitution operator, which we denote by $U\backslash$ and $L\backslash$, respectively we obtain

$$\forall\ M \in J(\widetilde{x}, \widetilde{x}+X)\ \ \forall\ x \in X: U\backslash\{L\backslash[-f(\widetilde{x}) + (LU-M)\cdot x]\} \in \text{l.h.s. (3.9)}.$$

By the basic principle of interval operations, the isotonicity, this proves the following theorem.

Theorem 4.1. Let $f : D \subseteq \mathbb{R}^n \to \mathbb{R}^n$, $f \in C^1$ be given, $\widetilde{x} \in D$, $X \in \mathbb{IIR}^n$ such that $\widetilde{x} \underline{\cup} (\widetilde{x}+X) \subseteq D$ and regular $L, U \in \mathbb{R}^{n\times n}$. If

$$Y := U\ \backslash\{L\ \backslash[-f(\widetilde{x}) + (LU - J(\widetilde{x}, \widetilde{x}+X))\cdot X]\} \subseteq \mathrm{int}(X) \qquad (4.2)$$

with $J(\widetilde{x}, \widetilde{x}+X)$ defined by (3.6) then every matrix $M \in J(\widetilde{x}, \widetilde{x}+X)$ is regular and there is a $\widehat{x} \in \widetilde{x}+X$ with $f(\widehat{x}) = 0$.

Proof. Follows by applying (4.1) and theorem 3.2. ■

If L, U are lower, upper triangular matrices, resp., banded or sparse the problem reduced to check (4.2) which means in particular to calculate an inclusion of $\Sigma(L, \boldsymbol{b}) = \{\, x \in \mathbb{R}^n \mid \exists\ b \in \boldsymbol{b} : Lx = b \,\}$, $\boldsymbol{b} \in \mathbb{IIR}^n$ for lower triangular L.

The obvious approach to do this by interval forward substitution does not work in general because of tremendous overestimations. Consider a most simple example

$$L = \begin{pmatrix} 1 & & & & \\ 1 & 1 & & & \\ 1 & 1 & 1 & & \\ & 1 & 1 & 1 & \\ & & 1 & 1 & 1 \\ & & & & \ddots \end{pmatrix} \quad \text{and} \quad \boldsymbol{b} = \begin{pmatrix} [-1,1] \\ [-1,1] \\ [-1,1] \\ \vdots \end{pmatrix}. \qquad (4.2a)$$

Then the inclusion X for $\Sigma(L, \boldsymbol{b})$ by interval forward substitution computes to

$$\begin{aligned} X_1 &= \boldsymbol{b}_1 = [-1, 1] \\ X_2 &= \boldsymbol{b}_2 - X_1 = [-1, 1] - [-1, 1] = [-2, 2] \\ X_3 &= \boldsymbol{b}_3 - X_1 - X_2 = [-1, 1] - [-1, 1] - [-2, 2] = [-4, 4] \\ X_4 &= \boldsymbol{b}_4 - X_2 - X_3 = [-1, 1] - [-2, 2] - [-4, 4] = [-7, 7] \\ X_5 &= \boldsymbol{b}_5 - X_3 - X_4 = [-1, 1] - [-4, 4] - [-7, 7] = [-12, 12] \\ &\dots \end{aligned} \tag{4.3}$$

Obviously X is always symmetric to the origin and tremendiously growing. It is easy to see that $X = [-x, x]$ were x is the solution of

$$\begin{pmatrix} 1 & & & & \\ -1 & 1 & & & \\ -1 & -1 & 1 & & \\ & -1 & -1 & 1 & \\ & & -1 & -1 & 1 \\ & & & & \dots \end{pmatrix} \cdot x = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ \vdots \end{pmatrix} \tag{4.4}$$

The matrix in (4.4) is Ostrowski's comparison matrix $\langle L \rangle$ (cf. [32]). The true solution complex $\Sigma(L, \boldsymbol{b})$ computes to $L^{-1} \cdot \boldsymbol{b}$, a formula which, however, is not suitable for numerical computations since L^{-1} is again full. For the true solution complex we obtain

$$\Sigma(L, \boldsymbol{b}) = L^{-1}\boldsymbol{b} = \begin{pmatrix} [-1, 1] \\ [-2, 2] \\ [-2, 2] \\ [-3, 3] \\ [-4, 4] \\ [-4, 4] \\ \dots \end{pmatrix}.$$

The overestimation of the X computed by (4.3) compared to the true solution complex $\Sigma(L, \boldsymbol{b}) = L^{-1}\boldsymbol{b}$ is equal to $\|\langle L \rangle^{-1}\|_\infty \;/\; \|L^{-1}\|_\infty$. For small values of n this is

n	10	20	30	40	50	60	70	80	90	100
$\|\langle L \rangle^{-1}\|_\infty \;/\; \|L^{-1}\|_\infty$	$2e1$	$1e3$	$1e5$	$1e7$	$1e9$	$1e11$	$1e13$	$1e15$	$1e17$	$1e19$

(4.5)

demonstrating the exponential behaviour of the overestimation. This way of trying to solve the problem contradicts drastically our basic rules for interval computations (2.4), (2.5).

For important classes of matrices such as M-matrices the approach is suitable. However, the behaviour shown above is *typical* for *general* matrices. Next we discuss a procedure for enclosing $\Sigma(L, \boldsymbol{b})$ for a *general* banded or sparse lower triangular matrix L.

Let $L \in \mathbb{R}^{n \times n}$ lower triangular with nonzero diagonal elements, $\boldsymbol{b} \in \mathbb{IR}^n$ be given.

Then

$$\forall\ b \in \boldsymbol{b} : \|L^{-1}b\|_2 \leq \|L^{-1}\|_2 \cdot \|b\|_2 = \sigma_n(L)^{-1} \cdot \|b\|_2 \leq \sigma_n(L)^{-1} \cdot \| \, |\boldsymbol{b}| \, \|_2 \tag{4.6}$$

where $\sigma_n(L)$ denotes the smallest singular value of L and

$$|\boldsymbol{b}| \in \mathbb{R}^n \text{ with } |\boldsymbol{b}|_i := \max\{ \, |b|_i \mid b \in \boldsymbol{b} \, \}.$$

That means finding a lower bound for the smallest singular value of L solves our problem of bounding $\Sigma(L, \boldsymbol{b}) = \{ \, L^{-1}b \mid b \in \boldsymbol{b} \, \}$. There are a number of very good condition estimators [8], [13] producing good approximation for the smallest singular value and the condition number of a matrix.

By the principle of their construction they deliver *upper bounds* for σ_n where we need lower bounds. Those can be obtained as follows.

$\sigma_n(L)^2$ is the smallest eigenvalue of LL^T. If for some $0 < \widetilde{\lambda} \in \mathbb{R}$ we can prove that $LL^T - \widetilde{\lambda} I$ is positive semidefinite this implies

$$\sigma_n(L) \geq \widetilde{\lambda}^{1/2}.$$

$LL^T - \widetilde{\lambda} I$ is positive semidefinite if its Cholesky decomposition $GG^T = LL^T - \widetilde{\lambda} I$ exists with nonnegative diagonal elements. This means the true, real Cholesky decomposition, not a floating point decomposition. The existence could be verified by performing an interval Cholesky decomposition, that is replacing the real operations by its corresponding interval operations. If all diagonal elements stay nonnegative (i.e. are intervals do not containing negative elements) the basic principle of interval analysis, the isotonicity, implies the existence of Cholesky factors within the computed interval factors.

However, this contradicts our main principles (2.4), (2.5). Most simple examples show tremendous overestimations like in (4.5). This is the typical behaviour for *general* matrices L. Therefore we perform a *floating point* Cholesky decomposition $\widetilde{G}\widetilde{G}^T \approx LL^T - \widetilde{\lambda} I$ and estimate its error by perturbation bounds for eigenvalues of symmetric matrices. We use the following consequence of a result by Wilkinson [43], p. 101–102.

Lemma 4.1. Let B and $B + E$ be $n \times n$ symmetric matrices and denote the smallest, largest eigenvalue of a real symmetric matrix by λ_n, λ_1, respectively. Then for $1 \leq i \leq n$

$$\lambda_i(B) + \lambda_n(E) \le \lambda_i(B+E) \le \lambda_i(B) + \lambda_1(E).$$

Setting $B := LL^T - \widetilde{\lambda} I$ and $E := \widetilde{G}\widetilde{G}^T - B$ implies that the matrix $B + E = \widetilde{G}\widetilde{G}^T$ (which is, of course, not computed) with $\widetilde{G} \in \mathbb{R}^{n\times n}$ is positive semidefinite. Therefore we can conclude

$$0 \le \lambda_n(LL^T - \widetilde{\lambda} I) + \|E\| \quad \Leftrightarrow \quad \lambda_n(LL^T) \ge \widetilde{\lambda} - \|E\|$$

and if $\widetilde{\lambda} \ge \|E\|$ then

$$\sigma_n(L) = \lambda_n(LL^T)^{1/2} \ge (\widetilde{\lambda} - \|E\|)^{1/2}.$$

This holds for any consistent matrix norm, like all p-norms. Summarizing we have the following lemma.

Lemma 4.2. Let $L \in \mathbb{R}^{n\times n}$, $\widetilde{G} \in \mathbb{R}^{n\times n}$, $\widetilde{\lambda} \in \mathbb{R}$

$$E := \widetilde{G}\widetilde{G}^T - (LL^T - \widetilde{\lambda} I).$$

If $\widetilde{\lambda} \ge \|E\|$ for some consistent matrix norm then

$$\sigma_n(L) \ge (\widetilde{\lambda} - \|E\|)^{1/2}.$$

For the application of lemma 4.2 we need a floating point decomposition $\widetilde{G}$ of $LL^T - \widetilde{\lambda} I$ but a verified estimation on E. This can be performed in one step. We define recursively for $1 \le i, j \le n$

$$r_{ij} := \sum_{\nu=1}^{j} L_{i\nu} L_{j\nu} - \sum_{\nu=1}^{j-1} G_{i\nu} G_{j\nu} \quad \text{and} \quad G_{ij} := r_{ij} \;/\; G_{jj} \quad \text{and}$$

$$r_{ii} := \sum_{\nu=1}^{j} L_{i\nu}^2 - \sum_{\nu=1}^{i-1} G_{i\nu}^2 - \widetilde{\lambda}^2 \qquad \text{and} \quad G_{ii} := r_{ii}^{1/2}.$$

These are the exact formulas for computing the Cholesky decomposition G of $LL^T - \widetilde{\lambda} I$. Now we calculate r_{ij} and r_{ii} by interval computations and the $\widetilde{G}_{ij}$ and $\widetilde{G}_{ii}$ by a floating point division and square root using the midpoint of r_{ij} and r_{ii}, respectively. Then

$$E_{ij} \in r_{ij} - \widetilde{G}_{ij}\widetilde{G}_{jj} \quad \text{and} \quad E_{ii} \in r_{ii} - \widetilde{G}_{ii}^2$$

where these operations are again performed using interval arithmetic. If we have a precise scalar product available [27], [28] the r_{ij} and r_{ii} can be calculated *precisely*. In other words the computation of $\widetilde{G}$ and E can be performed *simultaneously* and *without overestimation*. An approximate value $\widetilde{\lambda}$ is easily obtained via inverse power iteration because the resulting linear systems can be solved by backward or forward substitution.

If we apply the same procedure to the solution of a linear system with U and use (4.6) with L replaced by U then we can effectively apply theorem 4.1 to obtain a verified inclusion for systems of nonlinear equations. The main point is that the computing time for banded

$J(\widetilde{x}, \widetilde{x} + X)$ and therefore banded L and U increases *linearly* with n: If $J(\widetilde{x}, \widetilde{x} + X)$ is of lower, upper bandwidth p, q, resp. then

$$L \cdot U \quad \text{costs} \quad n \cdot (p+q) \cdot \min(p,q)$$

$$\text{estimating } \sigma(L) \quad \text{costs} \quad n \cdot p^2$$

$$\text{estimating } \sigma(U) \quad \text{costs} \quad n \cdot q^2.$$

Therefore, and this is the main point, the computing cost grows *linearly* with n. There are two other approaches known in the literature for treating large systems with banded or sparse matrices. The first [25], [9] uses interval forward and backward substitution. It is therefore by the principle of the approach restricted to H-matrices (see the example at the beginning of this chapter). The second approach [4] uses a so called singleton method which effectively computes an inverse of L and U. Therefore the computing time n^2p and n^2q is quadratically growing with n compared to linear growing np^2 and nq^2 of our method.

Let us consider some examples. In all examples the r.h.s. b is computed such that the true solution $\widehat{x}$ satisfies $\widehat{x}_i = (-1)^{i+1}/i$. For $Ax = b$ with $A = 0.1 \cdot LL^T$ and the matrix L from (4.2a) which caused so much trouble we get the following results.

n	cond	$\sigma_{\min}(A)$	$\|\widehat{x} - \widetilde{x}\|_\infty / \|\widetilde{x}\|_\infty$
10 000	1.22E+08	2.72E-04	3.39E-17
20 000	4.87E+08	1.36E-04	1.35E-16
50 000	3.04E+09	5.44E-05	8.47E-16
100 000	1.22E+10	2.72E-05	3.39E-15
500 000	3.04E+11	5.44E-06	8.47E-14
1 000 000	1.22E+12	2.72E-06	3.39E-13

Table 4.1. $A = 0.1 \cdot LL^T$, L defined by (4.2a)

The factor 0.1 is introduced in order to make the factors of the LU-decomposition not exactly representable on the machine. Some sparse systems from the Harwell testcases gave the following results.

Matrix	n	p	q	profile	cond	$\|A - \widetilde{L}\widetilde{U}\|_2$	$\|\widehat{x} - \widetilde{x}\|_\infty / \|\widetilde{x}\|_\infty$
gre_216	216	14	36	876	2.7e+02	3.1e-15	7.9e-27
gre_343	343	18	49	1435	2.5e+02	5.6e-15	2.4e-26
gre_512	512	24	64	2192	3.8e+02	7.4e-15	6.8e-26
west0167	167	158	20	507	2.8e+06	1.6e-16	4.6e-22
west0381	381	363	153	2157	2.0e+06	1.1e-15	8.8e-25
bcsstk08	1074	590	590	7017	6.1e+06	1.6e-16	6.6e-23
bcsstk14	1806	161	161	32630	4.3e+04	1.8e-15	1.8e-25

Table 4.2. Harwell test cases

In both cases we computed an approximate solution $\widetilde{x}$ and bounded the componentwise maximum relative error of $\widetilde{x}$. The computation has been performed in double precision equivalent to approximately 17 decimal places. The fact that we enclose the *difference* of $\widetilde{x}$ and the exact solution $\widehat{x}$ yields in the second example even more accuracy than the precision in use.

5 Global optimization

We will shortly sketch inclusion methods for global optimization and give some examples. Let the problem

$$\mathrm{Min}\{\, f(x) \mid x \in X \,\}, \quad X \in \mathbb{IIR}^n$$

be given. Our only assumption on f is the existence of an inclusion function $F : \mathbb{IIR}^n \to \mathbb{IIR}^n$ with

$$Y \in \mathbb{IIR}^n,\ Y \subseteq X \ \Rightarrow\ f(Y) := \{\, f(y) \mid y \in Y \,\} \subseteq F(Y) := [\underline{F}(Y), \overline{F}(Y)].$$

With these assumptions a branch-and-bound strategy for the computation of verified bounds for the global optimum value $f^* := \mathrm{Min}\{\, f(x) \mid x \in X \,\}$ and the global optimum points $X^* := \{\, x^* \in X \mid f(x^*) = f^* \,\}$ can be applied. Such methods are given in [15], [30], [36] using interval approaches and in [16], [33]. In the following we will describe a new and very interesting approach presented by Jansson [17], [18] and Jansson, Knüppel [19]. The algorithm does not require derivatives.

The computation of verified bounds for X^* may be time consuming. However, if sharp bounds are known on f^* and some $\widetilde{x} \in X$ is known with $f(\widetilde{x}) \approx f^*$ then this is frequently sufficient for practical applications. Therefore Jansson developed a procedure for computing sharp bounds $\underline{F}^*$ and $\overline{F}^*$ for the optimal value f^* with

$$\underline{F}^* \leq f^* \leq \overline{F}^*$$

and delivers an approximation $\widetilde{x} \in X$ with

$$\underline{F}^* \leq f(\widetilde{x}) \leq \overline{F}^*.$$

The method works *without* derivatives. It uses essentially the following two observations:

I) Local descent methods need a reasonably good starting point

II) A box $Y \subseteq X$ with estimated range $F(Y) = [\underline{F}(Y), \overline{F}(Y)]$ and $\underline{F}(Y) > f(\widetilde{x})$ for some already computed $\widetilde{x} \in X$ cannot contain a global optimum point.

The strategy is now to combine advantages of pure floating point local descent methods with the interval estimation of the range of a function. First the initial box X is subdivided into X^1, X^2 with $X^1 \cup X^2 = X$ where the X^i, $i \in \{1, 2\}$ with the smaller lower bound on $f(X^i)$ is further subdivided. Here the heuristic is used that the box with the smaller lower bound on the range of values contains smaller function values. This works very good in practice. The other box is put into a list. After subdividing few times a local descent method is started for the remaining box with midpoint as a starting point. In the examples Brent's algorithms [7] was used as a local descent method.

Now the two observations can benefit *mutually* from each other. According to I) the local descent method needs a good starting point. Therefore the interval subdividing strategy is used to derive a smaller starting box in order to obtain an improved starting value. So the interval method does help the floating point method. On the other hand, if the local descent method finds a good approximation $f(\widetilde{x})$ all boxes Y with greater lower bound on the range of f over Y can be deleted from the list. Thus the floating point method helps by reducing the list.

Combining this with an elaborate strategy for avoiding unnecessary subdivisions calls to the local descent method yields remarkable results. We mention in the following some test results, for more than 50 examples known from the literature see [19].

In [10] some test examples have been given for comparison of global optimization methods. In order to have a fair comparison all times are given in Standard Unit Time (STU) where 1 unit are 1000 evaluations of the Shekel function $S5$ at (4,4,4,4). On a SUN-4 one unit STU is about 0.2 sec.

In the following table different algorithms are compared using the test examples in [10]. The numbers in the upper half are from [5].

method	GP	BR	H3	H6	S5	S7	S10
Törn	4	4	8	16	10	13	15
De Biase	15	14	16	21	23	20	30
Price	3	4	8	46	14	20	20
Branin	-	-	-	-	9	8.5	9.5
Boender et al.	1.5	1	1.7	4.3	3.5	4.5	7
Jansson	0.45	0.45	5.65	6.45	0.70	0.80	0.90
$f^*, \overline{F}^*$	3	0.397887	-3.86278	-3.32237	-10.1532	-10.4049	-10.5364
$\underline{F}^*$	3	0.397887	-4.34853	-4.17324	-10.2008	-10.6772	-10.8517

Table 5.1. Comparison of floating point and verification algorithms

In the lower two lines the global optimum value f^* as well as the computed lower bound $\underline{F}^*$ are given. In all cases the computed approximation $f(\widetilde{x})$ coincides to at least 6 decimal figures with the global optimum value and the verified upper bound $\overline{F}^*$.

The following problem is to find the matrix within a set of matrices $M(x)$, $x \in X$ having the largest distance to the next singular matrix in the 2-norm. That is

$$f(x) := \min_{x \in X} -\sigma_{\min}(M(x)).$$

In our example it is

$$M(x) = \begin{pmatrix} 2\sin \pi x_1 & \sin \pi x_1 & \sin \pi x_2 & \sin \pi x_1 x_2 & \cos \pi x_1 x_2 \\ \sin \pi x_1 & 2\sin 4\pi x_2 & \cos \pi(1-x_1) & \cos \pi(1-x_2) & \cos \pi x_1 \\ \sin \pi x_2 & \cos \pi(1-x_1) & 2\cos 5\pi x_1 x_2 & \cos \pi x_1 & \cos \pi x_2 \\ \sin \pi x_1 x_2 & \cos \pi(1-x_2) & \cos \pi x_1 & 2\sin \pi x_2 & \sin \pi(1-x_1) \\ \cos \pi x_1 x_2 & \cos \pi x_1 & \cos \pi x_2 & \sin \pi(1-x_1) & 2\sin 4\pi x_1 \end{pmatrix}$$

$$0 \le x_i \le 1, \quad \text{for} \quad i = 1, 2.$$

For that example the following result is obtained

STU	$\underline{F}^*$	$f^*, \overline{F}^*$
278	-2.00159	-1.67555
397	-1.71291	-1.67555

The two computing times are for different parameter settings of the algorithm. A graph of $-f(x)$ is given below.

Finally we consider Griewank's function

$$f_G(x) := \sum_{i=1}^{n} \frac{x_i^2}{d} - \prod_{i=1}^{n} \cos \frac{x_i}{\sqrt{i}} + 1$$

with $X = [-600, 600]^n$, $d = 4000$

in n dimensions. For dimension $n = 2$ the function looks like as slightly arched egg carton with several 1000 local minima in the given domain X. The global optimum is $f^* = 0$. The results known to us are

	NRF	STU
Griewank 1981	6600*	-
Snyman, Fatti (1987)	23399	90

* global minimum not found

Table 5.2. Known results for Griewank's function ($n = 10$)

n	NRF	NIF	STU	$\underline{F}^*$	$\overline{F}^*$
10	417	421	4.3	0	$1.31 \cdot 10^{-14}$
50	743	1601	48.1	0	$2.25 \cdot 10^{-14}$

Table 5.3. Results of the verification algorithm for Griewank's function

Acknowledgement. The author wants to thank the referee for many helpful remarks.

References

[1] J.P. Abbott and R.P. Brent. Fast Local Convergence with Single and Multistep Methods for Nonlinear Equations. *Austr. Math. Soc. 19 (Series B)*, pages 173–199, 1975.

[2] G. Alefeld. Intervallanalytische Methoden bei nichtlinearen Gleichungen. In S.D. Chatterji et al., editor, *Jahrbuch Überblicke Mathematik 1979*, pages 63–78. Bibliographisches Institut, Mannheim, 1979.

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

[4] F.L. Alvarado. Practical Interval Matrix Computations. talk at the conference "Numerical Analysis with Automatic Result Verification", Lafayette, Louisiana, February 1993.

[5] C. Boender, A.R. Kan, G. Timmer, and L. Stongie. A Stocastic Method for Global Optimization. *Mathematical Programming 22*, pages 125–140, 1982.

[6] K.D. Braune. *Hochgenaue Standardfunktionen für reelle und komplexe Punkte und Intervalle in beliebigen Gleitpunktrastern*. Dissertation, Universität Karlsruhe, 1987.

[7] R.P. Brent. *Algorithms for Minimization without Derivatives*. Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1973.

[8] A.K. Cline, A.R. Conn, and C. van Loan. Generalizing the LINPACK Condition Estimator. *Numerical Analysis*, No. 909, 1982.

[9] D. Cordes and E. Kaucher. Self-Validating Computation for Sparse Matrix Problems. In *Computer Arithmetic: Scientific Computation and Programming Languages*. B.G. Teubner Verlag, Stuttgart, 1987.

[10] L.C.W. Dixon and G.P. Szegö (eds.). *Towards Global Optimization*. North-Holland, Amsterdam, 1975.

[11] R.T. Gregory and D.L. Karney. *A Collection of Matrices for Testing Computional Algorithms*. John Wiley & Sons, New York, 1969.

[12] A. Griewank. *On Automatic Differentiation*, volume 88 of *Mathematical Programming*. Kluwer Academic Publishers, Boston, 1989.

[13] W. Hager. Condition Estimates. *SIAM J. Sci. and Stat. Comp.*, 5:311–316, 1984.

[14] E.R. Hansen. On Solving Systems of Equations Using Interval Arithmetic. *Math. Comput. 22*, pages 374–384, 1968.

[15] E.R. Hansen. *Global Optimization using Interval Analysis.* Marcel Dekker, New York, 1992.

[16] R. Horst and H. Tuy. *Global Optimization.* Springer-Verlag, Berlin, 1990.

[17] C. Jansson. A Global Minimization Method: The One-Dimensional Case. Technical Report 91.2, Forschungsschwerpunkt Informations- und Kommunikationstechnik, TU Hamburg-Harburg, 1991.

[18] C. Jansson. A Global Optimization Method Using Interval Arithmetic. In L. Atanassova and J. Herzberger, editors, *Computer Arithmetic and Enclosure Methods*, IMACS, pages 259–267. Elsevier Science Publishers B.V., 1992.

[19] C. Jansson and O. Knüppel. A Global Minimization Method: The Multi-dimensional case. Technical Report 92.1, Forschungsschwerpunkt Informations- und Kommunikationstechnik, TU Hamburg-Harburg, 1992.

[20] R.B. Kearfott, M. Dawande, K. Du, and C. Hu. INTLIB: A portable Fortran-77 elementary function library. *Interval Comput.*, 3(5):96–105, 1992.

[21] R. Klatte, U. Kulisch, M. Neaga, D. Ratz, and Ch. Ullrich. *PASCAL-XSC — Sprachbeschreibung mit Beispielen.* Springer, 1991.

[22] O. Knüppel. BIAS — Basic Interval Arithmetic Subroutines. Technical Report 93.3, Forschungsschwerpunkt Informations- und Kommunikationstechnik, Inst. f. Informatik III, TU Hamburg-Harburg, 1993.

[23] O. Knüppel. PROFIL — Programmer's Runtime Optimized Fast Interval Library. Technical Report 93.4, Forschungsschwerpunkt Informations- und Kommunikationstechnik, TUHH, 1993.

[24] W. Krämer. *Inverse Standardfunktionen für reelle und komplexe Intervallargumente mit a priori Fehlerabschätzung für beliebige Datenformate.* Dissertation, Universität Karlsruhe, 1987.

[25] W. Krämer. Verified Solution of Eigenvalue Problems with Sparse Matrices. *Proceedings of 13th World Congress on Computation and Applied Mathematics*, pages 32–33, 1991.

[26] R. Krawczyk. Newton-Algorithmen zur Bestimmung von Nullstellen mit Fehlerschranken. *Computing*, 4:187–201, 1969.

[27] U. Kulisch. *Grundlagen des numerischen Rechnens (Reihe Informatik 19).* Bibliographisches Institut, Mannheim, Wien, Zürich, 1976.

[28] U. Kulisch and W.L. Miranker. *Computer Arithmetic in Theory and Practice*. Academic Press, New York, 1981.

[29] R. Lohner. *Einschließung der Lösung gewöhnlicher Anfangs- und Randwertaufgaben und Anordnungen*. PhD thesis, University of Karlsruhe, 1988.

[30] R.E. Moore. On Computing the Range of Values of a Rational Function of n Variables over a Bounded Region. *Computing 16*, pages 1–15, 1976.

[31] M.R. Nakao. A Numerical Verification Method for the Existence of Weak Solutions for Nonlinear Boundary Value Problems. *Journal of Mathematical Analysis and Applications*, 164:489–507, 1992.

[32] A. Neumaier. *Interval Methods for Systems of Equations*. Encyclopedia of Mathematics and its Applications. Cambridge University Press, 1990.

[33] P.M. Pardalos and J.B. Rosen. Constrained Global Optimization: Algorithms and Applications. *Springer Lecture Notes Comp. Sci. 268, Berlin*, 1987.

[34] M. Plum. Numerical existence proofs and explicit bounds for solutions of nonlinear elliptic boundary value problems. *Computing*, 49(1):25–44, 1992.

[35] L.B. Rall. Automatic Differentiation: Techniques and Applications. In *Lecture Notes in Computer Science 120*. Springer Verlag, Berlin-Heidelberg-New York, 1981.

[36] H. Ratschek and J. Rokne. *New Computer Methods for Global Optimization*. John Wiley & Sons (Ellis Horwood Limited), New York (Chichester), 1988.

[37] S.M. Rump. *Kleine Fehlerschranken bei Matrixproblemen*. PhD thesis, Universität Karlsruhe, 1980.

[38] S.M. Rump. Solving Non-Linear Systems with Least Significant Bit Accuracy. *Computing*, 29:183–200, 1982.

[39] S.M. Rump. New Results on Verified Inclusions. In W.L. Miranker and R. Toupin, editors, *Accurate Scientific Computations*, pages 31–69. Springer Lecture Notes in Computer Science 235, 1986.

[40] S.M. Rump. Rigorous Sensitivity Analysis for Systems of Linear and Nonlinear Equations. *Math. of Comp.*, 54(10):721–736, 1990.

[41] J.W. Schmidt. Die Regula-Falsi für Operatoren in Banachräumen. *Angew. Math. Mech.*, 41:61–63, 1961.

[42] B. Speelpennig. Compiling fast partial derivatives of functions given by algorithms. Urbana, Illinois, 1980.

[43] J.H. Wilkinson. *The Algebraic Eigenvalue Problem.* Oxford Univerity Press, Oxford, 1969.

[44] J.H. Wilkinson. Modern Error Analysis. *SIAM Rev. 13*, pages 548–568, 1971.