SCIENTIFIC COMPUTING
VALIDATED NUMERICS
INTERVAL METHODS

Edited by
W. KRAEMER
University of Wuppertal, Germany

J. WOLFF VON GUDENBERG
University of Wuerzburg, Germany

Kluwer Academic Publishers
Boston/Dordrecht/London

ON THE SOLUTION OF PARAMETRISED LINEAR SYSTEMS

Evgenija D. Popova
Institute of Mathematics & Informatics, Bulg. Acad. of Sci.
Acad. G. Bonchev Str., block 8, BG-1113 Sofia, Bulgaria
e-mail: epopova@bio.bas.bg

Keywords: parameter dependent linear systems, fixed-point iteration, Gauss-Seidel iteration, generalised interval arithmetic, applications

Abstract Considered are parametrised linear systems which parameters are subject to tolerances. Rump's fixed-point iteration method for finding outer and inner approximations of the hull of the solution set is studied and applied to an electrical circuit problem. Interval Gauss-Seidel iteration for parametrised linear systems is introduced and used for improving the enclosures, obtained by the fixed-point method, whenever they are not good enough. Generalised interval arithmetic (on proper and improper intervals) is considered as a computational tool for efficient handling of proper interval problems (to obtain inner interval estimations without inward rounding and to eliminate the dependency problem in parametrised Gauss-Seidel iteration). Numerical results from the application of the above methods to an electrical circuit problem are discussed.

1. Introduction

This work was imposed by the necessity to obtain sharp enclosures for the hull of the solution set of parameter dependent interval linear systems. Latter arise often in solving engineering problems which parameters may be subject to uncertainties.

We consider the application of Rump's fixed-point iteration method [10] to linear systems with several parameters which are subject to varying degree of uncertainties. Whenever the quality of the obtained outer and inner estimations of the interval hull of the parametrised solution set is not good enough, it is desirable that we could improve such inclusions. Interval Gauss-Seidel iteration for parameter dependent interval linear systems is introduced and used for the above purpose. The application of generalised interval arithmetic [3], [11] as a computational tool for efficient handling of conventional interval problems (obtaining inner numerical approximations without inward rounding and eliminating the dependency problem in parametrised Gauss-Seidel iteration) is demonstrated. An electrical circuit example [7] is used in Section 5 to illustrate the computational aspects and the results given by the various methods and also to indicate the importance of the varying degree of uncertainties and the number of parameters.

We use the following notations. $\mathbb{R}^n, \mathbb{R}^{n\times m}$ denote the set of real vectors with n components, the set of real $n \times m$ matrices. By normal (*proper*) interval we mean a real compact interval

$$[a] := [a^-, a^+] := \{a \in \mathbb{R} \mid a^- \leq a \leq a^+\}.$$

The set of proper intervals is denoted by $\mathbb{IR}$. This set is extended in [3] by the set $\overline{\mathbb{IR}} = \{[a^-, a^+] \mid a^-, a^+ \in \mathbb{R}, a^- \geq a^+\}$ of *improper* intervals obtaining thus the set $\mathbb{I}^*\mathbb{IR} = \mathbb{IR} \cup \overline{\mathbb{IR}} = \{[a^-, a^+] \mid a^-, a^+ \in \mathbb{R}\}$ of all ordered couples of real numbers called here generalised intervals. Normal (proper) intervals are a special case of generalised intervals and the conventional interval arithmetic[1] can be obtained as a projection of generalised interval arithmetic on $\mathbb{IR}$. In Section 2 we present only those basic facts from generalised interval arithmetic which are necessary to use it as a computational tool for handling proper interval problems.

By $\mathbb{IR}^n, \mathbb{I}^*\mathbb{IR}^n$ denote interval n-vectors and $\mathbb{IR}^{n\times m}, \mathbb{I}^*\mathbb{IR}^{n\times m}$ denote interval $n \times m$ matrices. The hull of a bounded set $S \in \mathbb{PR}^n$ is the interval vector $\square S := [\inf S,\ \sup S]$. Often it is important to know the quality of an outer approximation of a solution set Σ. The amount of overestimation can be estimated by means of inner approximations [9], [10]. $[x] \in \mathbb{IR}^n$ is called inner approximation for some set $\Sigma \in \mathbb{PR}^n$ if

$$\inf_{\sigma\in\Sigma} \sigma_i \leq x_i^- \quad \text{and} \quad x_i^+ \leq \sup_{\sigma\in\Sigma} \sigma_i, \qquad \text{for every } 1 \leq i \leq n.$$

2. Generalised Interval Arithmetic

The conventional (arithmetic and lattice) operations, order relations and other functions are isomorphically extended onto the whole set of proper and improper intervals [3]. "`Dual`" is an important monadic operator that reverses the end-points of the intervals and expresses an element-to-element symmetry between proper and improper intervals in $\mathbb{I}^*\mathbb{IR}$. For $[a] = [a^-, a^+] \in \mathbb{I}^*\mathbb{IR}$, its dual is defined by $\mathtt{Dual}([a]) = [a^+, a^-]$. `Dual` is applied componentwise to vectors and matrices.

[1]We assume familiarity with interval arithmetic on proper intervals (cf. [1], [5], [6]).

For $[a], [b] \in \mathbb{I}^*\mathbb{IR}$ and $\circ \in \{+, -, \times, /\}$,

$$\texttt{Dual}(\texttt{Dual}([a])) = [a], \qquad \texttt{Dual}([a] \circ [b]) = \texttt{Dual}([a]) \circ \texttt{Dual}([b]). \quad (1.1)$$

The generalized interval arithmetic structure possesses group properties with respect to $+$ and $\times$ operations: for $[a], [b] \in \mathbb{I}^*\mathbb{IR}$, $0 \notin [b]$

$$[a] - \texttt{Dual}([a]) = 0, \qquad [b]/\texttt{Dual}([b]) = 1. \quad (1.2)$$

Lattice operations are closed with respect to the inclusion relation; handling of norm and metric are very similar to norm and metric in linear spaces [3]. For more details on the theory, implementation and applications of generalised interval arithmetic consult [2], [3], [8], [11].

Inner Approximations

Let $\mathbb{F} \subset \mathbb{R}$ denote the set of floating-point numbers on a computer and $\mathbb{I}^*\mathbb{F} = \{[a^-, a^+] \in \mathbb{I}^*\mathbb{IR} \mid a^-, a^+ \in \mathbb{F}\}$ is the set of generalised intervals over $\mathbb{F}$. Denote by $\bigtriangledown$, resp. $\bigtriangleup$ the floating-point directed roundings ($\bigtriangledown, \bigtriangleup : \mathbb{R} \longrightarrow \mathbb{F}$) towards $-\infty$, resp. $+\infty$, defined in the IEEE binary (or radix-independent) floating-point standard or in [4].

Outward ($\Diamond$) and inward ($\bigcirc$) roundings $\Diamond, \bigcirc : \mathbb{I}^*\mathbb{IR} \longrightarrow \mathbb{I}^*\mathbb{F}$ are defined for generalised intervals by a semimorphism the same way as for normal intervals [4]. For $[a] = [a^-, a^+] \in \mathbb{I}^*\mathbb{IR}$,

$$\Diamond[a] := [\bigtriangledown a^-,\ \bigtriangleup a^+] \supseteq [a], \qquad \bigcirc[a] := [\bigtriangleup a^-,\ \bigtriangledown a^+] \subseteq [a].$$

If $\circ \in \{+, -, \times, /\}$ is an arithmetic operation in $\mathbb{I}^*\mathbb{IR}$, the corresponding computer operations $\diamond\!\!\!\circ\,$, $\odot$: $\mathbb{I}^*\mathbb{F} \times \mathbb{I}^*\mathbb{F} \longrightarrow \mathbb{I}^*\mathbb{F}$ are defined by

$$\begin{aligned} [a] \;\diamond\!\!\!\circ\; [b] &:= \Diamond\,([a] \circ [b]) &= [\ \bigtriangledown\,([a] \circ [b])^-,\ \ \bigtriangleup\,([a] \circ [b])^+\ ] \\ [a] \odot [b] &:= \bigcirc\,([a] \circ [b]) &= [\ \bigtriangleup\,([a] \circ [b])^-,\ \ \bigtriangledown\,([a] \circ [b])^+\ ]. \end{aligned}$$

Usually, an inner approximation of the correct interval result is sought together with the outer enclosure in order to estimate the degree of sharpness of the latter. Some safety problems also search for a minimum set of the solutions instead of an enclosure. Obtaining inner approximations on a computer in conventional interval arithmetic is possible only if the four interval arithmetic operations are implemented with inward rounding $\odot$ in addition to the four $\diamond\!\!\!\circ$ operations. The overloading concept of some programming environments does not allow the operators to be distinguished by their result type, which imposes the implementation of inwardly rounded interval operations as functions or subroutines. Unfortunately, most of the wide-spread interval packages do not support inwardly rounded interval arithmetic. The following three properties [2] of the rounded generalised interval arithmetic are of major importance for obtaining inner numerical approximations at no additional cost and show that the latter can be obtained only by means of outward directed rounding and the `Dual` operator in $\mathbb{I}^*\mathbb{F}$.

$$\text{For } [a] \in \mathbb{I}^*\mathbb{IR}, \qquad \bigcirc[a] = \texttt{Dual}(\Diamond\texttt{Dual}([a])). \tag{1.3}$$

$$\text{For } [a],[b] \in \mathbb{I}^*\mathbb{F}, \qquad \circ \in \{+,-,\times,/\},$$
$$[a] \odot [b] = \texttt{Dual}(\texttt{Dual}([a]) \,\diamond\!\!\!\circ\; \texttt{Dual}([b])). \tag{1.4}$$

Let $f[\{\circ_1,\ldots,\circ_m\},\{[a]_1,\ldots,[a]_n\}]$ be a rational function wherein $\circ_i \in \{+,-,\times,/\}$, $i=1,\ldots,m$ and $[a]_j \in \mathbb{I}^*\mathbb{IR}$, $j=1,\ldots,n$, then

$$f[\{\odot_i\}_{i=1}^m, \{\bigcirc[a]_j\}_{i=1}^n] = \texttt{Dual}\left(f[\{\diamond\!\!\!\circ_i\}_{i=1}^m,\ \{\Diamond\texttt{Dual}([a]_j)\}_{j=1}^n]\right). \tag{1.5}$$

We can apply the above three properties to obtain inner estimations of proper interval problems in a computing environment supporting generalised interval arithmetic (cf. [8]). For some input intervals the inwardly rounded conventional interval arithmetic may result in an empty set interval. The corresponding result in generalised interval arithmetic will be an improper interval. That is why, when using generalised interval arithmetic as a computing environment of a proper interval problem, we have to interpret improper interval results as empty sets.

Elimination of the Dependency Problem

Let $f(x_1,...,x_n)$ be a rational function. Denote the range of f over $[x] \in \mathbb{IR}^n$ by $f([x]) = \{f(x) \mid x \in [x]\}$ and by $F([x])$ the interval extension of f over $[x] \in \mathbb{I}^*\mathbb{IR}^n$. The next theorem specifies how to eliminate the dependency problem by using generalised interval arithmetic in range computation over a domain of proper intervals. The corresponding formulation for a domain of generalised intervals can be found in [2], [11].

Theorem 1 ([2]) *Let $f(x,a)$ be a rational function multi-incident on a and there exists a splitting $a' = (a'_1,..,a'_p)$, $a'' = (a''_1,..,a''_q)$ of the incidences of a. Let $g(x,a',a'')$ corresponds to the expression of f with explicit reference to the incidences of a and $g(x,a',a'')$ is continuous on $[x]\times[a]'\times[a]''$. Suppose that $g(x,a',a'')$ is unconditionally $\leq$-isotone for any component of a' and unconditionally $\leq$-antitone for any component of a'' on $[x]\times[a]'\times[a]''$, then*

- *if $f(x,a)$ is unconditionally $\leq$-isotone for a on $[x]\times[a]$,*

$$f([x],[a]) = G\left([x],[a]',\texttt{Dual}([a]'')\right) \subseteq F([x],[a]);$$

- *if $f(x,a)$ is unconditionally $\leq$-antitone for a on $[x]\times[a]$,*

$$f([x],[a]) = G\left([x], \texttt{Dual}([a]'), [a]''\right) \subseteq F([x],[a]).$$

The application of this theorem is illustrated in Section 5.

3. Fixed-Point Iteration Method

Consider linear system

$$A(p)\cdot x \quad = \quad b(p), \tag{1.6}$$

where $A(p)\in\mathbb{R}^{n\times n}$ and $b(p)\in\mathbb{R}^n$ depend on a parameter vector $p\in\mathbb{R}^k$. When p varies within a range $[p]\in\mathbb{IR}^k$, the set of solutions to all $A(p)\cdot x = b(p),\ p\in[p]$, is

$$\begin{aligned}\Sigma^p &:= \Sigma\left(A(p), b(p), [p]\right)\\ &:= \{x\in\mathbb{R}^n \mid A(p)\cdot x = b(p) \text{ for some } p\in[p]\}\,.\end{aligned} \tag{1.7}$$

Each individual component $\{A(p)\}_{ij}$ and $\{b(p)\}_j$ of $A(p)$, resp. $b(p)$ depends linearly on p means that there are vectors

$$\begin{aligned}&w(i,j)\in\mathbb{R}^k \quad \text{for} \quad 0\leq i\leq n,\ 1\leq j\leq n \quad \text{with}\\ &\{A(p)\}_{ij} = w(i,j)^\top\cdot p \quad \text{and} \quad \{b(p)\}_j = w(0,j)^\top\cdot p.\end{aligned} \tag{1.8}$$

Theorem 2 (Rump [10]) *Let $A(p)\cdot x = b(p)$ with $A(p)\in\mathbb{R}^{n\times n}$, $b(p)\in\mathbb{R}^n$, $p\in\mathbb{R}^k$ be a parametrised linear system, where $A(p), b(p)$ are given by (1.8). Let $R\in\mathbb{R}^{n\times n}$, $[Y]\in\mathbb{IR}^n$, $\tilde{x}\in\mathbb{R}^n$ and define $[Z]\in\mathbb{IR}^n$, $[C]\in\mathbb{IR}^{n\times n}$ by*

$$\begin{aligned}[Z]_i &:= \Big(\sum_{j,\nu=1}^{n}\{R_{ij}\cdot(w(0,j)-\tilde{x}_\nu\cdot w(j,\nu))\}^\top\Big)\cdot[p],\\ [C] &:= I - R\cdot A([p]).\end{aligned}$$

Define $[V]\in\mathbb{IR}^n$ by means of the following Einzelschrittverfahren

$$1\leq i\leq n\ :\ V_i := \{\square([Z]+[C]\cdot[U])\}_i, \quad [U] := (V_1,\ldots,V_{i-1},Y_i,\ldots,Y_n)^\top.$$

If $[V]\subsetneqq[Y]$, then R and every matrix $A(p), p\in[p]$ are regular, and for every $p\in[p]$ the unique solution $\widehat{x} = A^{-1}(p)b(p)$ of (1.6) satisfies $\widehat{x}\in\tilde{x}+[V]$. With $[\Delta] := \square\{[C]\cdot[V]\}\in\mathbb{IR}^n$ and the solution set Σ^p, defined by (1.7), the following inner estimation holds true.

$$\begin{aligned}&[\tilde{x}+\inf([Z])+\sup([\Delta]),\ \tilde{x}+\sup([Z])+\inf([\Delta])]\subseteq\\ &\qquad\qquad [\inf(\Sigma^p),\ \sup(\Sigma^p)]\,.\end{aligned} \tag{1.9}$$

It was proven in [9] that for $[Q] = \tilde{x} + [Z]$ with the notations of Theorem 2,

$$[Q] \subseteq \square\, \Sigma^p - \square\{[C]\cdot(\Sigma^p - \tilde{x})\} \subseteq \square\Sigma^p - [\Delta]. \qquad (1.10)$$

The above inclusions, which are in $\mathbb{IR}$, present a special case of the same inclusion in $\mathbb{I}^*\mathbb{IR}$. Applying property (1.2) and due to the isotonicity of the arithmetic operations in $\mathbb{I}^*\mathbb{IR}$, we get

$$[Q] + \mathtt{Dual}([\Delta]) \subseteq \square\, \Sigma^p. \qquad (1.11)$$

Latter inclusion can be also obtained from (1.9), which is a restriction of (1.11) to $\mathbb{IR}$.

The left-hand side of (1.9), written in computer arithmetic, is

$$[\tilde{x} \oplus \inf(\bigcirc[Z]) \oplus \sup(\Diamond[\Delta]),\ \tilde{x} \oplus \sup(\bigcirc[Z]) \oplus \inf(\Diamond[\Delta])], \qquad (1.12)$$

which is equivalent to (1.11) written in computer arithmetic, that is, due to the properties (1.3)–(1.5) and (1.1),

$$\mathtt{Dual}\,(\tilde{x} \mathbin{\Diamond\!\!\!+} (\Diamond\mathtt{Dual}([Z])) \mathbin{\Diamond\!\!\!+} (\Diamond[\Delta])) \subseteq \square\, \Sigma^p.$$

This way, we can easily compute inner estimation of the interval hull either of the parametrised solution set, or of the general solution set corresponding to a non-parametrised system.

For $[a] \in \mathbb{IR}$, $[b] \in \overline{\mathbb{IR}}$, $[a] + [b] \in \mathbb{IR} \iff \omega([a]) \geq \omega([b])$, wherein $\omega([a]) = \{a^+ - a^- \text{ if } [a] \in \mathbb{IR};\ a^- - a^+ \text{ if } [a] \in \overline{\mathbb{IR}}\}$ defines the width of an interval. Since $[\Delta] \in \mathbb{IR}$ and $\mathtt{Dual}([Z]) \in \overline{\mathbb{IR}}$,

$$\mathtt{Dual}\,(\tilde{x} \mathbin{\Diamond\!\!\!+} (\Diamond\mathtt{Dual}([Z])) \mathbin{\Diamond\!\!\!+} (\Diamond[\Delta])) \in \mathbb{IR} \iff \omega(\Diamond\mathtt{Dual}([Z])) \geq \omega(\Diamond[\Delta]),$$

that is, the inner estimation (1.9) is empty iff $\omega([Z]) < \omega([\Delta])$. In practice, empty inner estimations can be obtained either due to round-off errors when the intervals are tight but the matrix is near to singular, or when the tolerances for the parameters are big.

When somehow we have sharpen the outer estimation $\square\, \Sigma^p \subseteq [\hat{V}] \subseteq [V]$, then the improved outer estimation $[\hat{V}]$ can replace $[V]$ in (1.9), resp. (1.12) to get an improved inner estimation of $\square\, \Sigma^p$ (Section 5).

4. Parametrised Gauss-Seidel Iteration

For arbitrary $[x] \in \mathbb{IR}^n$ we are interested in good enclosures for the truncated solution set

$$\Sigma\,(A(p), b(p), [p]) \cap [x].$$

Writing the system $A(p) \cdot x = b(p)$ componentwise

$$\Sigma_{k=1}^{n} A_{ik}(p)x_k = b_i(p), \qquad i = 1, ..., n$$

and assuming that $A_{ii}(p) \neq 0$, we have

$$\begin{aligned} x_i = & \ (b_i(p) - \Sigma_{k\neq i} A_{ik}(p)x_k) / A_{ii}(p) \\ \subseteq & \ \square \{ (b_i(p) - \Sigma_{k\neq i} A_{ik}(p)x_k) / A_{ii}(p) \mid \\ & \qquad\qquad x \in [x], p \in [p] \} := [x]'_i. \end{aligned} \tag{1.13}$$

Suppose that $A_{ii}(p) \neq 0$ for all $p \in [p]$, an interval vector $[x] \in \mathbb{IR}^n$ containing x is known, and if we can find the above hull, then we can apply (1.13) for $i = 1, ..., n$ and obtain another enclosure $[x]'$ for x, such that

$$\Sigma\left(A(p), b(p), [p]\right) \cap [x] \subseteq [x]' \cap [x].$$

To simplify this presentation and due to the lack of space, further on we shall assume that $A_{ii}(p) \neq 0$ for all $p \in [p]$ and $i = 1, ..., n$. For given $[x] \in \mathbb{IR}^n$, $[p] \in \mathbb{IR}^k$ and rational functions $a(p) : \mathbb{R}^k \longrightarrow \mathbb{R}$, $b(p, x) : \mathbb{R}^k \times \mathbb{R}^n \longrightarrow \mathbb{R}$, by analogy with the non-parametric interval Gauss-Seidel iteration (cf. [6]), we define

$$\begin{aligned} \Gamma(a(p), b(p, x), & [x], [p]) \\ & := \ \square \{ x \in [x] \mid a(p)x = b(p, x) \text{ for some } p \in [p] \} \\ & = \ \square \{ b(p, x)/a(p) \mid x \in [x], p \in [p] \} \cap [x] \end{aligned} \tag{1.14}$$

and

$$[y]_i := \Gamma\left(A_{ii}(p), b_i(p) - \Sigma_{k<i} A_{ik}(p)[y]_k - \Sigma_{k>i} A_{ik}(p)[x]_k, [x], [p]\right), \qquad i = 1, ..., n. \tag{1.15}$$

Denote by $\Gamma\left(A(p), b(p), [x], [p]\right)$ the interval vector $[y]$, defined by (1.15), and call $\Gamma\left(A(p), b(p), [x], [p]\right)$ the parametrised Gauss-Seidel operator, applied to $A(p)$, $b(p)$, and $[x]$ for all $p \in [p]$. The main problem in the application of the parametrised Gauss-Seidel operator is the computation of

$$\square \{ (b_i(p) - \Sigma_{k<i} A_{ik}(p)[y]_k - \Sigma_{k>i} A_{ik}(p)[x]_k) / A_{ii}(p) \mid p \in [p] \}. \tag{1.16}$$

We recommend (and will demonstrate this in Section 5) to compute (1.16) by applying Theorem 1 for elimination of the dependency problem always whenever it is possible.

If $[y] = \Gamma\left(A(p), b(p), [x], [p]\right)$ is strictly contained in $[x]$, we may hope to get a further improved enclosure of $\Sigma\left(A(p), b(p), [p]\right)$ by considering the iteration

$$[x]^0 := [x], \quad [x]^{l+1} := \Gamma\left(A(p), b(p), [x]^l, [p]\right) \quad (l = 0, 1, 2, ...)$$

which we call the interval parametrised Gauss-Seidel iteration. The parametrised functional (1.14) and the parametrised Gauss-Seidel operator possess properties similar to the corresponding functional and operator from the non-parametric interval Gauss-Seidel iteration (see e.g [6]). Beside the next theorem, a detailed presentation of the interval parametrised Gauss-Seidel iteration will be done in a separate work.

Theorem 3 *Let* $[x] \in \mathbb{IR}^n$ *be given,* $\Gamma\left(A([p]), b([p]), [x]\right)$ *be the non-parametrised Gauss-Seidel operator, related to the general linear system* $A([p]) \cdot x = b([p])$*, and the interval vector* $[y]$ *is defined by (1.15).*
If there exists i*,* $1 \le i \le n$*, such that the function*

$$\left(b_i(p) - \Sigma_{k<i} A_{ik}(p) y_k - \Sigma_{k>i} A_{ik}(p) x_k\right) / A_{ii}(p)$$

satisfies Theorem 1 with respect to some parameter p_j*,* $1 \le j \le k$*, then for* $\Gamma\left(A(p), b(p), [x], [p]\right)$*, computed by using Theorem 1, it holds*

$$\Gamma\left(A(p), b(p), [x], [p]\right) \subsetneqq \Gamma\left(A([p]), b([p]), [x]\right).$$

5. Applications and Numerical Experiments

As example, we consider a linear resistive network, presented in [7]. The resistive network consists of two current sources J_1 and J_2 and nine resistors. The problem of finding the voltages $v_1, ..., v_5$, when the voltage of each conductance $g_i, i = 1, ..., 9$ varies independently in prescribed bounds $[g]_i, i = 1, ..., 9$, leads to the following parametrised linear system

$$\begin{pmatrix} g_1+g_6 & -g_6 & 0 & 0 & 0 \\ -g_6 & g_2+g_6+g_7 & -g_7 & 0 & 0 \\ 0 & -g_7 & g_3+g_7+g_8 & -g_8 & 0 \\ 0 & 0 & -g_8 & g_4+g_8+g_9 & -g_9 \\ 0 & 0 & 0 & -g_9 & g_5+g_9 \end{pmatrix} v = J, \tag{1.17}$$

where $J = (10, 0, 10, 0, 0)^\top$ and the parameters are subject to tolerances $[g]_i = [1-\delta,\ 1+\delta],\ i = 1, ..., 9$. We solve the system (1.17) for different values of the tolerances δ varying from 0.1% to 10% of the nominal value.

The results from the application of Theorem 2 are presented in Table 1. A refinement iteration using intersection ([10], Theorem 2.2) was used to sharpen the obtained outer, resp. inner approximations. The second column of Table 1 gives the maximum improvement of the outer approximation in percent. Let $[v]^* \subseteq \square\Sigma^p \subseteq [v]$ denote the corresponding improved inner and outer approximations. The quality of the obtained outer estimation is measured by the quotient $\omega([v]^*)/\omega([v])$, presented in the third column of Table 1. For this particular example, the exact bounds of the parametric solution set are given by

$$\square\,\Sigma^p \;:=\; [\min_{u\in U} A(g_u)^{-1}J,\ \max_{u\in U} A(g_u)^{-1}J],$$

wherein g_u is defined componentwise as

$$(g_u)_i = \begin{cases} g_i^+ & \text{if } u_i = 1, \\ g_i^- & \text{if } u_i = -1 \end{cases} \qquad \text{for } [g] = [g^-, g^+] \in \mathbb{IR}^9$$

and each $u \in U = \{u \in \mathbb{R}^9 \mid |u| = (1, ..., 1)^\top\}$. The forth column in Table 1 gives the overestimation of $[v]$ with respect to $\square\,\Sigma^p$ in percent.

Table 1 Results from the fixed-point Rump's method

δ [%]	% *max* *refinement*	$\frac{\omega([v]^*)}{\omega([v])}$ *min* – *max*	$100 \cdot (1 - \frac{\omega(\square\Sigma^p)}{\omega([v])})$ *min* – *max*
0.1	0.06	.99 – .99	0.33 – 0.52
1.0	0.51	.90 – .93	3.27 – 5.22
2.0	0.65	.79 – .86	6.57 – 10.36
3.0	2.16	.69 – .80	9.90 – 15.43
4.0	2.78	.59 – .73	13.26 – 20.43
5.0	3.06	.49 – .66	16.67 – 25.35
6.0	3.30	.39 – .60	20.12 – 30.20
7.0	3.65	.29 – .52	23.62 – 34.98
8.0	3.79	.20 – .45	27.18 – 39.69
9.0	3.67	.10 – .37	30.82 – 44.32
10.0	3.23	.01 – .30	34.52 – 48.88

Table 1 shows that the degree of sharpness of the outer inclusion decreases with increasing the tolerances. The results in the fourth column agree with the results in the third one. As it was pointed by Rump [10], the refinement iteration by intersection does not contribute much to the improvement of the estimations.

Now, we apply the parametrised Gauss-Seidel iteration in order to sharpen the outer inclusion $[v] \supseteq \square\Sigma^p$. To get sharp bounds for the parametrised Gauss-Seidel operator we apply Theorem 1 for elimination of the dependency problem in each of the parameter-dependent functions involved in the operator. Table 2 presents $\Gamma(A(g), J, [v], [g])$ as a result of the application of Theorem 1. Right to the expressions, by arrows are presented: the total monotonicity of the function with respect to the corresponding multi-incident parameter and the monotonicity with respect to each of its incidences.

Table 2 Theorem 1, applied to $\Gamma(A(g), J, [v], [g])$ for (1.17)

exact range	monotonicity
$[y]_1 = \frac{10+\mathtt{Dual}([g]_6)\cdot[v]_2}{[g]_1+[g]_6} \cap [v]_1$	$\downarrow \frac{\uparrow}{\downarrow}$
$[y]_2 = \frac{[g]_6\cdot[y]_1+[g]_7\cdot[v]_3}{[g]_2+\mathtt{Dual}([g]_6)+\mathtt{Dual}([g]_7)} \cap [v]_2$	$\uparrow_6 \frac{\uparrow}{\downarrow}, \quad \uparrow_7 \frac{\uparrow}{\downarrow}$
$[y]_3 = \frac{10+\mathtt{Dual}([g]_7)\cdot[y]_2+\mathtt{Dual}([g]_8)\cdot[v]_4}{[g]_3+[g]_7+[g]_8} \cap [v]_3$	$\downarrow_7 \frac{\uparrow}{\downarrow}, \quad \downarrow_8 \frac{\uparrow}{\downarrow}$
$[y]_4 = \frac{\mathtt{Dual}([g]_9)\cdot[v]_5+[g]_8\cdot[y]_3}{[g]_4+\mathtt{Dual}([g]_8)+[g]_9} \cap [v]_4$	$\uparrow_8 \frac{\uparrow}{\downarrow}, \quad \downarrow_9 \frac{\uparrow}{\downarrow}$
$[y]_5 = \frac{[g]_9\cdot[y]_4}{[g]_5+\mathtt{Dual}([g]_9)} \cap [v]_5$	$\uparrow \frac{\uparrow}{\downarrow}$

For the initial boxes $[v]$, obtained at tolerances less than 10%, all the components of $\Gamma(A(g), J, [v], [g])$ were monotone with respect to the corresponding multi-incident parameters, satisfying thus Theorem 1. For $[v]$, obtained at $\delta = 10\%$, at the first iteration y_2 and y_3 were not monotone with respect to g_7, so that we had an outer inclusion for the exact range of the corresponding component of $\Gamma(A(g), J, [v], [g])$. The monotonicity of y_3 was proven after the first iteration and the monotonicity of y_2 was proven after the second iteration.

Denote by $[\hat{v}]$ the improved outer approximation of $\square\, \Sigma^p$ computed by the parametrised Gauss-Seidel iteration. The corresponding improved inner approximation $[\hat{v}]^*$ is computed by

$$\mathtt{Dual}\left(\tilde{x} \mathbin{\Diamond\kern-0.65em+} (\Diamond\mathtt{Dual}([Z])) \mathbin{\Diamond\kern-0.65em+} (\Diamond([C]([\hat{v}] - \tilde{x})))\right),$$

wherein $[Z], [C], \tilde{x}$ are from Theorem 2.

The results after application of the parametrised Gauss-Seidel iteration and the corresponding inner estimation are summarized in Table 3. The table starts at $\delta = 4\%$ because the Gauss-Seidel iteration was not able to improve any component of $[v]$ for tolerances less than 3.4%. The second column represents how much, in %, the parametrised Gauss-Seidel iteration improves the outer estimation, obtained by the Rump's method. We see a substantial improvement, compared to the second column of Table 1. The improvement increases with the increasing of the tolerances. Computing the inner inclusion, corresponding to the improved outer inclusion, we are able to estimate the degree of sharpness of the results, obtained by the parametrised Gauss-Seidel iteration, which is presented in the third column of Table 3.

Table 3 Results after the parametrised Gauss-Seidel iteration

δ [%]	$100 \cdot (1 - \frac{\omega([\hat{v}])}{\omega([v])})$ *min* – *max*	$\frac{\omega([\hat{v}^*])}{\omega([\hat{v}])}$ *min* – *max*	$100 \cdot (1 - \frac{\omega(\square\Sigma^p)}{\omega([\hat{v}])})$ *min* – *max*
4.0	0 – 1.41	.69 – .79	13.25 – 19.29
5.0	0 – 2.82	.61 – .74	15.73 – 23.93
6.0	1.33 – 5.10	.55 – .70	17.58 – 27.61
7.0	4.83 – 9.67	.49 – .65	19.06 – 29.55
8.0	8.28 – 14.08	.43 – .62	20.11 – 31.70
9.0	13.29 – 19.18	.38 – .59	20.21 – 33.19
10.0	17.93 – 25.65	.36 – .62	14.78 – 31.29

As a whole, the parametrised Gauss-Seidel iteration was able to improve the quality of the outer and inner inclusions for the hull of the solution set to some reasonable bounds. However, it deserves studying other techniques for a further improvement.

6. Conclusion

Our experience and the example considered show that Rump's method is perfect for small to modest tolerances for the parameters. The introduced parametrised Gauss-Seidel iteration could be used as a refinement step for the Rump's method under big tolerances and empty inner estimations. Generalised interval arithmetic is useful for efficient computation of inner estimations and elimination of the dependency problem.

Both, the Rump's method and the parametrised Gauss-Seidel iteration, as well as some other methods for general and parameter-dependent interval linear systems are implemented in the environment of CAS *Mathematica* and a package for generalised interval arithmetic [8]. A discussion on the implementation of these methods and the impact of computer algebra on interval computations will be given separately.

The Rump's method, coupled with parametrised Gauss-Seidel iteration and other methods, should be used extensively in the engineering applications, which as a rule involve dependencies and uncertainties. Unfortunately, it can be seen from the literature that the rare engineering applications of interval methods use general methods for non-parametric or symmetric matrices to problems involving more dependencies than in a symmetric matrix. We hope that the presence of public software, supporting the discussed methods, will increase their application.

Acknowledgments

The author is indebted to Prof. K. Okumura (Kyoto Univ.) for providing the electrical circuit example. Thanks to Prof. R. Muhanna (Georgia Inst. of Technology), who provoked my studies of parametrised interval linear systems. This work was supported by the Bulgarian National Science Fund under grant No. I-903/99.

References

[1] Alefeld, G.; Herzberger, J.: *Introduction to Interval Computations*, Academic Press, New York, 1983.

[2] Gardeñes, E., Trepat, A.: Fundamentals of SIGLA, an Interval Computing System over the Completed Set of Intervals, *Computing* **24** (1980), pp. 161–179.

[3] Kaucher, E.: Interval Analysis in the Extended Interval Space IR, *Computing Suppl.* **2** (1980), pp. 33–49.

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

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

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

[7] Okumura, K.: An Application of Interval Operations to Electric Network Analysis, *Bull. of the Japan Soc. for Industrial & Applied Mathematics*, **2** (1993), pp. 115–127.

[8] Popova, E., Ullrich, C.: *Directed Interval Arithmetic in Mathematica: Implementation and Applications.* TR 96-3, U. Basel, 1996, pp. 1–56. (http://www.math.bas.bg/~epopova/papers/tr96-3.ps)

[9] Rump, S. M.: Rigorous Sensistivity Analysis for Systems of Linear and Nonlinear Equations, *Mathematics of Computation* **54**, 190, (1990), pp. 721–736.

[10] Rump, S. M.: *Verification Methods for Dense and Sparse Systems of Equations*, in Topics in Validated Computations, J. Herzberger, ed., Elsevier Science B. V. (1994), pp. 63–135.

[11] SIGLA/X: Modal Intervals, in Vehi, J., Sainz M. (eds.): *Applications of Interval Analysis to Systems & Control*, Proc. MISC'99, Univ. de Girona, Spain (1999), pp. 139–207.