Solution Sets of Complex Linear Interval Systems of Equations*

Milan Hladík
Charles University, Faculty of Mathematics and Physics, Department of Applied Mathematics, Malostranské nám. 25, 118 00, Prague, Czech Republic
milan.hladik@matfyz.cz

Abstract

We present a solution set description for systems of complex interval equations, where complex intervals have a rectangular form. The solution set is described by a system of nonlinear inequalities, which can be used to obtain a very accurate approximation of the interval hull of the solution set. In our numerical experiments we exploit this approximation to study overestimation for common complex interval equation solvers (Gauss elimination, Hansen-Bliek-Rohn-Ning-Kearfott method).

Keywords: Complex interval systems, complex intervals, solution set
AMS subject classifications: 65F99, 65G40

1 Introduction

Uncertainty which naturally appears in real-life problems can be treated in various ways. In interval analysis models we suppose that the input data varies independently in some (given) compact intervals. Solving interval systems of equations is a basic problem of interval analysis. Sometimes in practical problems (e.g. electrical circuits [9, 10]), complex variables can occur. Complex intervals can be defined as rectangles (see e.g. [1])

$$\begin{aligned}\boldsymbol{a} + \boldsymbol{b}i &\equiv [a_1 + b_1 i, a_2 + b_2 i] \\ &= \{a + bi \in \mathbb{C} \mid a_1 \le a \le a_2,\ b_1 \le b \le b_2\},\end{aligned}$$

or as circles (see e.g. [1])

$$\langle m, r\rangle \equiv \{z \in \mathbb{C} \mid |m - z| \le r\}.$$

Polar form representation of complex intervals and its arithmetic can be found in [3]. In this paper we study complex interval equations for rectangular complex intervals.

*Submitted: September 5, 2007; Revised: May 19, 2009; Accepted: May 25, 2009.

1.1 Notation

The vector $e_n = (1, \dots, 1)^T$ is the n-dimensional vector of ones. An interval matrix is defined as

$$\boldsymbol{A} \equiv [A_1, A_2] = \{A \in \mathbb{R}^{m\times n} \mid A_1 \le A \le A_2\},$$

where $A_1 \le A_2$ are fixed matrices. The set of all $m \times n$ real and complex matrices will be denoted by $\mathbb{IR}^{m\times n}$ and $\mathbb{IC}^{m\times n}$, respectively; n-dimensional interval vectors can be regarded as interval matrices $n \times 1$. By

$$A^c \equiv \frac{1}{2}(A_1 + A_2), \quad A^\Delta \equiv \frac{1}{2}(A_2 - A_1)$$

we denote the midpoint and radius of $\boldsymbol{A}$, respectively. $\square S$ stands for the interval hull of the set S. An enclosure of a set S is any superset of S calculated by a particular method. Inner and outer approximations of S are subsets and supersets, respectively, of S. It is assumed that these approximations are close and determined by possibly exhausting and time consuming computation. Inner and outer approximations serve to show how tight their respective methods are and to measure overestimation of their enclosures.

1.2 Complex interval equations

Let $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{IR}^{n\times n}$, $\boldsymbol{c}, \boldsymbol{d} \in \mathbb{IR}^n$. The complex interval system in question is the following

$$(\boldsymbol{A} + \boldsymbol{B}\,i)z = \boldsymbol{c} + \boldsymbol{d}\,i.$$

This can be rewritten with real variables x, y as

$$(\boldsymbol{A} + \boldsymbol{B}\,i)(x + y\,i) = \boldsymbol{c} + \boldsymbol{d}\,i. \tag{1}$$

The solution set of (1) is defined traditionally as

$$\begin{aligned}\textstyle\sum = \big\{x + y\,i \in \boldsymbol{C} \mid \exists A \in \boldsymbol{A}\ \exists B \in \boldsymbol{B}\ \exists c \in \boldsymbol{c}\ \exists d \in \boldsymbol{d} :\\ (A + B\,i)(x + y\,i) = c + d\,i\big\}.\end{aligned} \tag{2}$$

The system (1) can be transformed into the real form of finding the set of variables $x, y \in \boldsymbol{R}^n$ satisfying

$$\begin{aligned} Ax - By &= c,\\ Bx + Ay &= d \end{aligned} \tag{3}$$

for certain $A \in \boldsymbol{A}$, $B \in \boldsymbol{B}$, $c \in \boldsymbol{c}$, $d \in \boldsymbol{d}$. This is not a standard interval system of equations due to dependences between the matrices $A \in \boldsymbol{A}$ and $B \in \boldsymbol{B}$, but the interval system

$$\begin{aligned} \boldsymbol{A}x - \boldsymbol{B}y &= \boldsymbol{c},\\ \boldsymbol{B}x + \boldsymbol{A}y &= \boldsymbol{d} \end{aligned} \tag{4}$$

can be used to obtain an outer estimate of $\sum$ via various solvers such as Gaussian elimination [1], the Hansen-Bliek-Rohn-Ning-Kearfott method [13] or the Householder method [2] (see Example 1). Other enclosures of $\sum$ can be also obtained directed by the complex interval system solvers: Gaussian elimination [1] and the complex Householder method [4]. We should note that the complex Householder method described in [4] is wrong, because of the missing the complex number ζ (cf. [11]). One can easily see this in their numerical experiments, where

$$[1,5]+[-1,1]\,i=\mathbf{A}^{(1)}_{1,1}\not\subseteq-\mathbf{H}^{(1)}_{1,1}.\alpha=$$
$$=-\,[-0.4306,0.1348].[25.0199,25.5147]=[-3.4394,10.9867].$$

2 Solution set characterization

In this section we derive a description of the solution set Σ. We use the form (3) and exhibit the following theorem from [6] dealing with dependences in interval systems.

Theorem 1. *Let* $\boldsymbol{M}\in\mathbb{IR}^{m\times n}$, $\boldsymbol{P},\boldsymbol{Q}\in\mathbb{IR}^{m\times h}$, $\boldsymbol{b},\boldsymbol{c}\in\mathbb{IR}^{m}$. *Then for certain* $M\in\boldsymbol{M}$, $P\in\boldsymbol{P}$, $Q\in\boldsymbol{Q}$, $p\in\boldsymbol{p}$, $q\in\boldsymbol{q}$ *vectors* $u,v\in\mathbb{R}^n$, $w\in\mathbb{R}^h$ *form a solution of the system*

$$Mu+Pw=p, \tag{5}$$
$$Mv+Qw=q \tag{6}$$

if and only if they satisfy the following system of inequalities

$$M^{\Delta}|u|+P^{\Delta}|w|+p^{\Delta}\geq|r_1|, \tag{7}$$
$$M^{\Delta}|v|+Q^{\Delta}|w|+q^{\Delta}\geq|r_2|, \tag{8}$$
$$\begin{aligned}&P^{\Delta}|w||v|^T+Q^{\Delta}|w||u|^T+p^{\Delta}|v|^T+\\&\qquad+q^{\Delta}|u|^T+M^{\Delta}|uv^T-vu^T|\geq|r_1v^T-r_2u^T|,\end{aligned} \tag{9}$$

where $r_1\equiv-M^cu-P^cw+p^c$, $r_2\equiv-M^cv-Q^cw+q^c$.

Due to this theorem we can simply give a description of the solution set Σ. By setting $\boldsymbol{M}\equiv(\boldsymbol{A}\ \boldsymbol{B})$, $\boldsymbol{P}\equiv\boldsymbol{Q}\equiv0$, $\boldsymbol{p}\equiv\boldsymbol{c}$, $\boldsymbol{q}\equiv\boldsymbol{d}$, $u\equiv(x^T,-y^T)^T$, $v\equiv(y^T,x^T)^T$ we immediately have the following corollary.

Corollary 1. *The solution set* Σ *is described by*

$$A^{\Delta}|x|+B^{\Delta}|y|+c^{\Delta}\geq|r_1|, \tag{10}$$
$$A^{\Delta}|y|+B^{\Delta}|x|+d^{\Delta}\geq|r_2|, \tag{11}$$
$$\begin{aligned}&c^{\Delta}|y|^T+d^{\Delta}|x|^T+A^{\Delta}|xy^T-yx^T|+\\&\qquad+B^{\Delta}|-yy^T-xx^T|\geq|r_1y^T-r_2x^T|,\end{aligned} \tag{12}$$
$$\begin{aligned}&c^{\Delta}|x|^T+d^{\Delta}|y|^T+A^{\Delta}|xx^T+yy^T|+\\&\qquad+B^{\Delta}|-yx^T+xy^T|\geq|r_1x^T+r_2y^T|,\end{aligned} \tag{13}$$

where $r_1\equiv-A^cx+B^cy+c^c$, $r_2\equiv-A^cy-B^cx+d^c$.

The solution set for real interval equations represents a polyhedral set, which is convex in each orthant (see e.g. [5]). This is not generally true for complex interval systems. The solution set Σ is neither a polyhedral set, nor necessarily convex in each orthant, see Figure 2. Nevertheless, this is not surprising, cf. [12].

An approximation of $\square\Sigma$ can be obtained with an interval hull version of the procedure SIVIA from [7], see Table 1. This branch & bound based algorithm computes approximations to the interval hull of the set described by the system of nonlinear inequalities $f(z) \geq 0$; in our case the set Σ is described by (10)–(13). SIVIA returns inner and outer approximations $\underline{\boldsymbol{z}}$, $\overline{\boldsymbol{z}}$, respectively, such that $\underline{\boldsymbol{z}} \subseteq \square\Sigma \subseteq \overline{\boldsymbol{z}}$. The parameter ε determines how tight the approximation will be. The functions $\boldsymbol{l}(\boldsymbol{z})$ and $\boldsymbol{r}(\boldsymbol{z})$ divide the interval $\boldsymbol{x}$ into two parts along the widest components.

Table 1: Version of SIVIA for the set described by $f(z) \geq 0$.

Algorithm SIVIA(in:f,$\boldsymbol{z}$,ε, inout: $\underline{\boldsymbol{z}}$, $\overline{\boldsymbol{z}}$)
if $f(\boldsymbol{z}) \geq 0$ **then**
$\quad \underline{\boldsymbol{z}} = \square(\underline{\boldsymbol{z}} \cup \boldsymbol{z})$; $\overline{\boldsymbol{z}} = \square(\overline{\boldsymbol{z}} \cup \boldsymbol{z})$; **return**;
if $\exists i : f(\boldsymbol{z})_i < 0$ **return**;
if $\boldsymbol{z}^\Delta < \varepsilon$ **then**
$\quad \overline{\boldsymbol{z}} = \square(\overline{\boldsymbol{z}} \cup \boldsymbol{z})$; **return**;
if $\boldsymbol{l}(\boldsymbol{z}) \not\subseteq \underline{\boldsymbol{z}}$ **then** SIVIA(f,$\boldsymbol{l}(\boldsymbol{z})$,$\varepsilon$, $\underline{\boldsymbol{z}}$, $\overline{\boldsymbol{z}}$);
if $\boldsymbol{r}(\boldsymbol{z}) \not\subseteq \underline{\boldsymbol{z}}$ **then** SIVIA(f,$\boldsymbol{r}(\boldsymbol{z})$,$\varepsilon$, $\underline{\boldsymbol{z}}$, $\overline{\boldsymbol{z}}$);

Let $\underline{\boldsymbol{z}} = \underline{\boldsymbol{x}} + \underline{\boldsymbol{y}}\, i$ and $\overline{\boldsymbol{z}} = \overline{\boldsymbol{x}} + \overline{\boldsymbol{y}}\, i$ be (n-dimensional) inner and outer approximations of $\square\Sigma$, respectively, and let $\boldsymbol{z} = \boldsymbol{x} + \boldsymbol{y}\, i$ be an enclosure of $\square\Sigma$ computed by a particular method. The quality of the enclosure $\boldsymbol{z}$ can be measured by various *enclosure ratios*. For the sake of this paper we introduce the following one; it is defined as an interval instead of a real number, since the optimal interval hull of solutions is hardly known exactly.

$$\boldsymbol{\rho}(\boldsymbol{z}) \equiv \left[\frac{e_n^T \boldsymbol{x}^\Delta + e_n^T \boldsymbol{y}^\Delta}{e_n^T \overline{\boldsymbol{x}}^\Delta + e_n^T \overline{\boldsymbol{y}}^\Delta},\ \frac{e_n^T \boldsymbol{x}^\Delta + e_n^T \boldsymbol{y}^\Delta}{e_n^T \underline{\boldsymbol{x}}^\Delta + e_n^T \underline{\boldsymbol{y}}^\Delta} \right].$$

This enclosure ratio measures overestimation of the interval vector $\boldsymbol{z}$ to the optimal one. It satisfies natural requirements on such ratios, that is, if $\boldsymbol{z}$ is optimal then $1 \in \boldsymbol{\rho}(\boldsymbol{z})$ (but not conversely), and the larger $\boldsymbol{z}$ is with respect to the optimum, the larger the values $\boldsymbol{\rho}(\boldsymbol{z})$.

3 Numerical experiments

In this section, we measure overestimation of several complex interval equation solvers on various examples.

Example 1. Let us consider the complex interval system from [1, 4]

$$\begin{pmatrix} [1,5] + [-1,1]\, i & 1 \\ 25 & [-1,1] + [-1,1]\, i \end{pmatrix} (x + y\, i) = \begin{pmatrix} [-1,1] \\ [-1,1] \end{pmatrix}. \tag{14}$$

Hence

$$\boldsymbol{A} = \begin{pmatrix} [1,5] & 1 \\ 25 & [-1,1] \end{pmatrix},\ \boldsymbol{B} = \begin{pmatrix} [-1,1] & 0 \\ 0 & [-1,1] \end{pmatrix},$$
$$\boldsymbol{c} = \begin{pmatrix} [-1,1] \\ [-1,1] \end{pmatrix},\ \boldsymbol{d} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

We will compare diverse solvers and compute their enclosure ratios. The following computations were carried out in MATLAB 7.0.4 (R14.2) with help of the interval toolbox INTLAB v5.3 (see [14]). Solving (14) by the function `verifylss` leads to the solution

$$\boldsymbol{x}^1 = \big([-0.1453, 0.1453],\ [-1.7646, 1.7646]\big)^T,$$
$$\boldsymbol{y}^1 = \big([-0.1453, 0.1453],\ [-1.7646, 1.7646]\big)^T,$$

while the function `intgauss` (interval Gaussian elimination with mignitude pivoting) returns the enclosure

$$\boldsymbol{x}^2 = \big([-0.1373, 0.1373],\ [-1.7185, 1.7185]\big)^T,$$
$$\boldsymbol{y}^2 = \big([-0.1373, 0.1373],\ [-1.7185, 1.7185]\big)^T,$$

The Hansen-Bliek-Rohn-Ning-Kearfott method `hsolve` results in the same solution as `intgauss`

$$\boldsymbol{x}^3 = \big([-0.1373, 0.1373],\ [-1.7185, 1.7185]\big)^T,$$
$$\boldsymbol{y}^3 = \big([-0.1373, 0.1373],\ [-1.7185, 1.7185]\big)^T,$$

All the functions `intgauss`, `verifylss` and `hsolve` applied on real interval system (4) return the same intervals

$$\boldsymbol{x}^4 = \big([-0.1354, 0.1354],\ [-1.7724, 1.7724]\big)^T,$$
$$\boldsymbol{y}^4 = \big([-0.0954, 0.0954],\ [-0.6124, 0.6124]\big)^T,$$

The Householder method from [4] returns

$$\boldsymbol{x}^5 = \big([-0.2086, 0.2086],\ [-2.1663, 2.1663]\big)^T,$$
$$\boldsymbol{y}^5 = \big([-0.11, 0.11],\ [-0.0607, 0.0607]\big)^T,$$

which is wrong (cf. Subsection 1.2), since the width of $\boldsymbol{y}^5$ is too small. For instance $(-0.01, 0.1) \notin \boldsymbol{y}^5$, but

$$\begin{pmatrix} 5-i & 1 \\ 25 & 1-0.5\,i \end{pmatrix} \begin{pmatrix} 0.05-0.01\,i \\ -0.3+0.1\,i \end{pmatrix} = \begin{pmatrix} -0.06 \\ 1 \end{pmatrix}.$$

In our implementation of the interval version of the complex Householder method, we obtained $\boldsymbol{x}^6 = \boldsymbol{R}^2$, $\boldsymbol{y}^6 = \boldsymbol{R}^2$, hence the direct use of this method is very ineffective. Finding an effective implementation that produces narrower intervals is still open problem.

What is the result for *this* example? The best functions are `intgauss` and `hsolve`, but surprisingly the enclosures based of the real system (4) are worse in only one variable and are better in three variables.

In the second part of this example we show how tight these enclosures are. We implemented the algorithm SIVIA (Table 1) in the programming language C++, supplemented by the C++ class library C-XSC 2.1.1 [8]. For $\varepsilon = 0.0001$, we obtained inner and outer approximations

$$\underline{\boldsymbol{x}} = \big([-0.1010, 0.1010],\ [-1.5338, 1.5338]\big)^T,$$
$$\underline{\boldsymbol{y}} = \big([-0.0715, 0.0715],\ [-0.4200, 0.4200]\big)^T,$$
$$\overline{\boldsymbol{x}} = \big([-0.1012, 0.1012],\ [-1.5363, 1.5363]\big)^T,$$
$$\overline{\boldsymbol{y}} = \big([-0.0718, 0.0718],\ [-0.4221, 0.4221]\big)^T,$$

The interval enclosure ratio (as introduced in Section 2) applied on $\boldsymbol{x}^3 + \boldsymbol{y}^3 i$ is

$$\boldsymbol{\rho}^3 = [1.7415, 1.7456],$$

and applied on $\boldsymbol{x}^4 + \boldsymbol{y}^4 i$ is as follows

$$\boldsymbol{\rho}^4 = [1.2272, 1.2301].$$

Hence solving the complex interval system (1) leads to the about 74% overestimation, the real form (4) gives only about 23% overestimation.

One might think that we can apply the algorithm SIVIA directly to the complex interval system (14). But due to the equations, SIVIA generally does not return arbitrarily tight approximations. For our example with $\varepsilon = 0.001$ and the starting enclosure $\boldsymbol{x}^4$, $\boldsymbol{y}^4$, we obtain very wide approximation $\overline{\boldsymbol{x}}^4 = \boldsymbol{x}^4$, $\overline{\boldsymbol{y}}^4 = \boldsymbol{y}^4$, $\underline{\boldsymbol{x}}^4 = \underline{\boldsymbol{y}}^4 = \emptyset$, i.e., no improvement happened (especially due to the noninterval vector $\boldsymbol{d}$).

Example 2. Figure 1 shows real overestimation for four cases and selected methods. Each case consists of seven computations. The horizontal axis represents each of the seven individual cases, while the vertical axis represents the average of the individual components of the solution vector

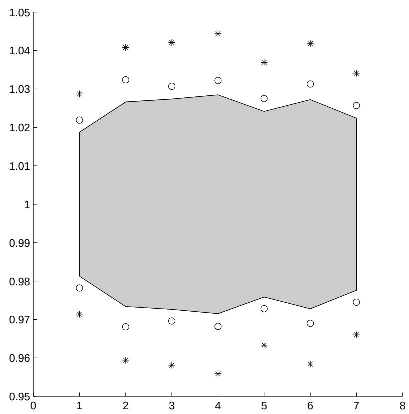
$$\frac{1}{2n}\sum_{i=1}^{n}(\boldsymbol{x}_i + \boldsymbol{y}_i)$$

for that case. The vertical extent of the grey area represents an average solution interval computed by SIVIA (Table 1) with accuracy ε, the dark grey belongs to $\underline{\boldsymbol{z}}$, and the light grey to $\overline{\boldsymbol{z}}$. The difference between $\underline{\boldsymbol{z}}$ and $\overline{\boldsymbol{z}}$ is indistinguishable on the first two figures. The mark $*$ signifies the bounds of the average interval solution obtained by the Hansen-Bliek-Rohn-Ning-Kearfott method applied to the complex interval system (1). The Hansen-Bliek-Rohn-Ning-Kearfott method applied to the real interval system (4) results in an average interval solution the bounds of which are marked by $\circ$. The corresponding average enclosure ratios are denoted by $\boldsymbol{\rho}^*$ and $\boldsymbol{\rho}^\circ$, respectively.

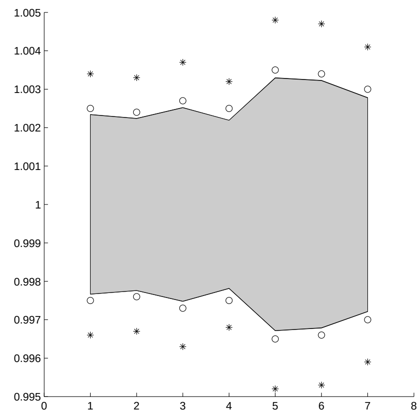
We denote the $n \times n$ matrix of all ones by E_n, the identity matrix by I_n, and Random(n, n) stands for a random $n \times n$ matrix whose elements are uniformly distributed in the interval $(0, 1)$. Then the interval matrices $\boldsymbol{A}$, $\boldsymbol{B}$ and interval vectors $\boldsymbol{c}$, $\boldsymbol{d}$ are given in the following way:

1. $n = 2$, $\delta = 0.01$,
   $A^c = \text{Random}(n, n) + 2E_n,\ B^c = \text{Random}(n, n) + 4E_n - 2I_n,$
   $A^\Delta = B^\Delta = \delta E_n,$
   $c^c = (A^c - B^c)\, e_n,\ d^c = (A^c + B^c)\, e_n,\ c^\Delta = d^\Delta = \delta e_n;$
2. Like situation 1., but $\delta = 0.001$.
3. $n = 4$, $\delta = 0.1$,
   $A^c = \text{Random}(n, n) + E_n + 4I_n,\ B^c = \text{Random}(n, n) + 5I_n,$
   $A^\Delta = B^\Delta = \delta E_n,$
   $c^c = (A^c - B^c)\, e_n,\ d^c = (A^c + B^c)\, e_n,\ c^\Delta = d^\Delta = \delta e_n;$
4. Like situation 3., but $\delta = 0.01$.

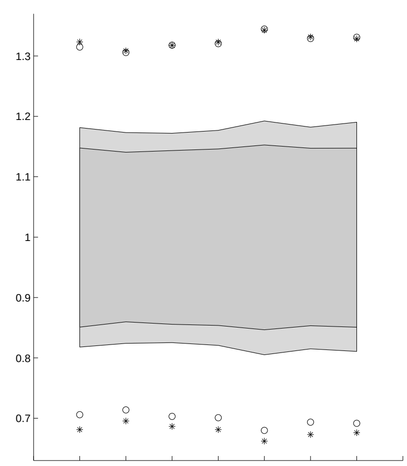
The results were computed on a PC with an AMD Athlon 64 Processor 4400+, 2.2 GHz, 884 MB RAM, GNU/Linux, and the source code was written again in C++ with C-XSC 2.1.1 [8]. In the first and second cases, each example took from one to five



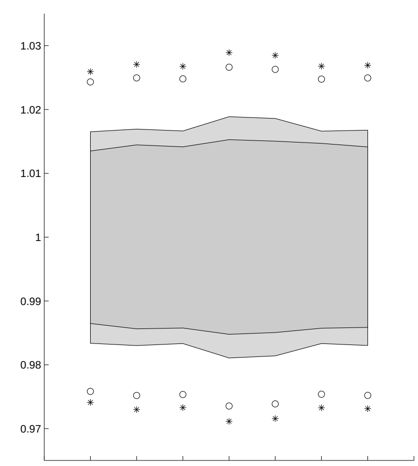
1. $n = 2$, $\boldsymbol{\rho}^* = [1.5318, 1.5321]$,
$\varepsilon = 10^{-6}$, $\boldsymbol{\rho}^\circ = [1.1477, 1.1479]$.



2. $n = 2$, $\boldsymbol{\rho}^* = [1.4633, 1.4658]$,
$\varepsilon = 10^{-6}$, $\boldsymbol{\rho}^\circ = [1.0782, 1.0800]$.

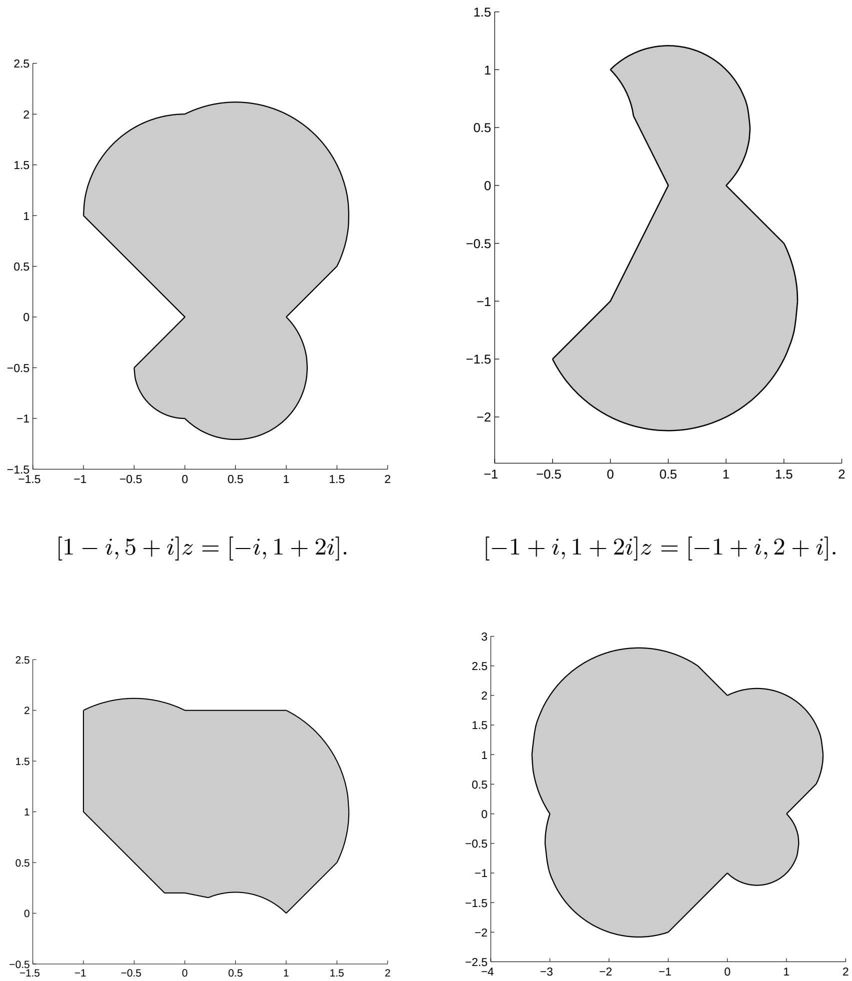


3. $n = 4$, $\boldsymbol{\rho}^* = [1.7538, 2.1997]$,
$\varepsilon = 10^{-2}$, $\boldsymbol{\rho}^\circ = [1.6974, 2.1290]$.



4. $n = 4$, $\boldsymbol{\rho}^* = [1.5751, 1.8886]$,
$\varepsilon = 10^{-3}$, $\boldsymbol{\rho}^\circ = [1.4552, 1.7448]$.

Figure 1: Overestimation of the Hansen-Bliek-Rohn-Ning-Kearfott method for the complex ($*$) and the real ($\circ$) case.



$[1-i, 5+i]z = [-i, 1+2i]$. $[-1+i, 1+2i]z = [-1+i, 2+i]$.

$[1, 5+i]z = [-1+i, 1+2i]$. $[1-i, 5+i]z = [-3-i, 1+2i]$.

Figure 2: Solution set for 1×1 complex interval systems.

hours of CPU time. The last two cases were much more time consuming, and we had to do some improvements: We used the SIVIA method only to compute $\overline{z}$, while the inner approximation $\underline{z}$ was obtained by a Monte Carlo method. Nevertheless, each example took about ten days of CPU time (due to $\overline{z}$).

Example 3. Figure 2 shows solution set images for 1×1 complex interval systems. The boundary of the solution set is always formed by lines and circular arcs. The computations were carried out in MATLAB 7.0.4 (R14.2) with the toolbox INTLAB v5.3 [14].

4 Conclusion

We have described the solution set for complex interval systems of equations (rectangular case) and have shown how it can be used for checking algorithmic efficiency. We have presented several examples which imply that overestimation for the best algorithms is between 5% and 15% for the two dimensional cases, and between 50% and 110% for the four dimensional cases. In higher dimensions the overestimation will be probably much higher, but such cases are intractable at the present time.

The numerical experiments have also shown that the SIVIA method is very time-consuming. Developing more efficient approximation methods is a challenge for the future.

References

[1] Alefeld, G. and Herzberger, J.: *Introduction to interval computations*, Academic Press, London, 1983.

[2] Bentbib, A.H.: Solving the full rank interval least squares problem, *Appl. Numer. Math.* **41** (2002), No. 2, pp. 283–294.

[3] Candau, Y.; Raissi, T.; Ramdani, N. and Ibos, L.: Complex interval arithmetic using polar form, *Reliable Computing* **12** (2006), No. 1, pp. 1–20.

[4] Djanybekov, B. S.: Interval Householder method for complex linear systems, *Reliable Computing* **12** (2006), No. 1, pp. 35–43.

[5] Fiedler, M.; Nedoma, J.; Ramik, J.; Rohn, J. and Zimmermann, K.: *Linear optimization problems with inexact data*, Springer–Verlag, New York, 2006.

[6] Hladík, M.: Solution set characterization of linear interval systems with a specific dependence structure, *Reliable Computing* **13** (2007), No. 4, pp. 361–374.

[7] Jaulin, L.; Kieffer, M.; Didrit, O. and Walter, É.: *Applied interval analysis*, Springer, London, 2001.

[8] Klatte, R.; Kulisch, U.W.; Wiethoff, A.; Lawo, C. and Rauch, M.: *C-XSC. A C++ class library for extended scientific computing*, Springer–Verlag, Berlin, 1993. (`http://www.math.uni-wuppertal.de/~xsc/`)

[9] Kolev, L.V. and Vladov, S.S.: Linear circuit tolerance analysis via systems of linear interval equations. *ISYNT'89 6-th International Symposium on Networks, Systems and Signal Processing*, June 28 – July 1, Zagreb, Yugoslavia, (1989), pp. 57–60.

[10] Kolev, L.V.: *Interval methods for circuit analysis*, Word Scientific, Singapore, 1993.

[11] Morrison, D.D.: Remarks on the unitary triangularization of a nonsymmetric matrix, *J. Assoc. Comput. Mach.* **7** (1960), No. 2, pp. 185–186.

[12] Nickel, K.: Arithmetic of complex sets, *Computing* **24** (1980), pp. 97–105.

[13] Ning, S. and Kearfott, R.B.: A comparison of some methods for solving linear interval equations, *SIAM J. Numer. Anal.* **34** (1997), No. 4, pp. 1289–1305.

[14] Rump, S.M.: *INTLAB – Interval Laboratory, the Matlab toolbox for verified computations, Version 5.3*, 2006. (`http://www.ti3.tu-harburg.de/rump/intlab/`)