A filtering method for the interval eigenvalue problem

Milan Hladik, David Daney, Elias P. Tsigaridas

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

A filtering method for the interval eigenvalue problem

Milan Hladík — David Daney — Elias P. Tsigaridas

N° 7057

October 2009

Thème SYM

ISSN 0249-6399 ISRN INRIA/RR--7057--FR+ENG

INSTITUT NATIONAL
DE RECHERCHE
EN INFORMATIQUE
ET EN AUTOMATIQUE

INRIA

centre de recherche
SOPHIA ANTIPOLIS - MÉDITERRANÉE

A filtering method for the interval eigenvalue problem

Milan Hladík* † , David Daney† , Elias P. Tsigaridas†

Thème SYM — Systèmes symboliques
Équipe-Projet Coprin

Rapport de recherche n° 7057 — October 2009 — 13 pages

Abstract:

We consider the general problem of computing *intervals* that contain the real eigenvalues of *interval* matrices. Given an outer estimation of the real eigenvalue set of an interval matrix, we propose a filtering method that improves the estimation.

Even though our method is based on an sufficient regularity condition, it is very efficient in practice, and our experimental results suggest that, in general, improves significantly the input estimation. The proposed method works for general, as well as for symmetric matrices.

Key-words: Interval matrix, symmetric matrix, interval analysis, eigenvalue, eigenvalue bounds.

* Charles University, Faculty of Mathematics and Physics, Department of Applied Mathematics, Malostranské nám. 25, 118 00, Prague, Czech Republic, e-mail: milan.hladik@matfyz.cz

† INRIA Sophia-Antipolis Méditerranée, 2004 route des Lucioles, BP 93, 06902 Sophia-Antipolis Cedex, France, e-mail: FirstName.LastName@sophia.inria.fr

Centre de recherche INRIA Sophia Antipolis – Méditerranée
2004, route des Lucioles, BP 93, 06902 Sophia Antipolis Cedex
Téléphone : +33 4 92 38 77 77 — Télécopie : +33 4 92 38 77 65

Mthode de filtrage des valeurs propres des matrices intervalles

Résumé :

Mots-clés : Matrices intervalles, matrices symtriques, valeurs propres, bornes sur les valeurs propres, analyse par intervalles.

1 Introduction

To model real-life problems and perform computations we *must* deal with uncertainty and inexactness; they are due to measurement, due to simplification of physical models, due to variations of the parameters of the system, and finaly due to computational errors. Interval analysis is an efficient and reliable tool that allows us to handle the aforementioned problems, even in the worst case where all together are encoutered simultaneously. The input quantities are given with some interval estimation, and the algorithms output verified intervals as results, that, even though they usually have the drawback of overestimation, they cover all the possibilities for the input quantities.

In this paper we are interesting in the interval real eigenvalue problem. That is, given a family of matrices, that is an interval matrix, the task is to estimate the set of all possible eigenvalues. Moreover, there is a need to distinguish general interval matrices from the symmetric ones. Applications arise mostly in the field of mechanics and engineering. We name, for instance, automobile suspension system [21], mass structures [20], vibrating systems [7], robotics [4], and even principal component analysis [8] and independent component analysis [5], which could be considered as a statistics oriented applications. Using the well-known Jordan–Wielandt transformation [9, 13, 19], given a solution to the interval real eigenvalue problem, we can provide and approximation for the singular values and the condition number; both of which have numerous applications.

The first general results for the interval real eigenvalue problem are due to Deif [6], and Deif & Rohn [26]. However, their solution depends on theorems that have very strong assumptions. Later, Rohn [23], introduced a boundary point characterization of the eigenvalue set. Approximation methods were addressed by Qiu et al. [21] and by Hladík et al. [11]. The latter work is based on a branch and prune approach and yields results that depend on a given, arbitrarily high, accuracy.

The symmetric eigenvalue problem is very important in practice. However, it is hard to handle it, in its interval form, since the correlations between the entries of the matrices, make the algorithms that depend on interval analysis to overestimation, usually a lot, the results. The symmetric case was pioneered by Deif [6]. Another theoretical result is due to Hertz [10], see also [25], for determining two extremal points of the eigenvalue set. Diverse approximation algorithms has also been developed. An evolution strategy method by Yuan et al. [29] yields inner approximation of the eigenvalues set. Matrix perturbation theory was used by Qiu et al. [20], who proposed an algorithm for approximating the bounds, and by Leng & He [17] for outer estimation of the eigenvalue set. Outer bounds that are easy and fast to compute were presented by Hladík et al. [12]. Outer estimation for general parametric case was considered by Kolev [16].

In this paper, we propose a filtering method for reducing the overestimation produced by many methods. Generally, filtering is very useful approach used in constrained programming, but few is known for the interval eigenvalue problem. We can, of course, apply any filtering for the interval nonlinear system

of equations arising from eigenvalue definition, but no such approach has been successful yet; cf. [11]. To the best of our knowledge, there is only one paper by Beaumont [2] giving an iterative algorithm based on convex approximation of eigenpairs. We present a new filtering method which is more simple and applicable for both the symmetric and unsymmetric case. Because we do not take into account eigenvectors, the filtering is much more efficient.

The rest of the paper is structure are follows. In the next section we present the basic definitions and our main theoretical result. In Sec. 3 we present our algorithm, while in Sec. 4 we exploit it on numerical examples. Sec. 5 presents our conclusions.

2 Basic definitions and main theorem

Let us introduce some notions from interval analysis. An interval matrix is defined as a family of matrices

$$\boldsymbol{A} := [\underline{A}, \overline{A}] = \{A \in \mathbb{R}^{m\times n};\ \underline{A} \leq A \leq \overline{A}\},$$

where $\underline{A}$, $\overline{A} \in \mathbb{R}^{m\times n}$, $\underline{A} \leq \overline{A}$, are given matrices, and the inequality is considered element-wise. By

$$A_c := \frac{1}{2}(\underline{A} + \overline{A}), \quad A_\Delta := \frac{1}{2}(\overline{A} - \underline{A})$$

we denote the midpoint and radius of $\boldsymbol{A}$, respectively.

Let $\boldsymbol{A} \subseteq \mathbb{R}^{n\times n}$ be a square interval matrix. Its eigenvalue set is defined as

$$\Lambda(\boldsymbol{A}) := \{\lambda \in \mathbb{R};\ Ax = \lambda x,\ x \neq 0,\ A \in \boldsymbol{A}\}.$$

An outer approximation of $\Lambda(\boldsymbol{A})$ is any set having $\Lambda(\boldsymbol{A})$ as a subset.

An important class of matrices is that of symmetric ones. Its generalization to interval matrices is as follows. *A symmetric interval matrix* is defined as

$$\boldsymbol{A}^S := \{A \in \boldsymbol{A} \mid A = A^T\},$$

and its eigenvalue set is denoted similarly to generic case, that is

$$\Lambda(\boldsymbol{A}^S) := \{\lambda \in \mathbb{R};\ Ax = \lambda x,\ x \neq 0,\ A \in \boldsymbol{A}^S\}.$$

A symmetric interval matrix $\boldsymbol{A}^S$ is a proper subset of $\boldsymbol{A}$, and so its eigenvalue set, $\Lambda(\boldsymbol{A}^S)$, is in general a subset of $\Lambda(\boldsymbol{A})$.

Since a real symmetric matrix $A \in \mathbb{R}^{n\times n}$ has always n real eigenvalues, we can sort them in a non-increasing order as follows

$$\lambda_1(A) \geq \lambda_2(A) \geq \cdots \geq \lambda_n(A).$$

We extend this notation for symmetric interval matrices, that is

$$\boldsymbol{\lambda}_i(\boldsymbol{A}^S) := \left\{\lambda_i(A) \mid A \in \boldsymbol{A}^S\right\}.$$

These sets form n compact intervals, which can be disjoint or may overlap, see for example [12]. The union of these interval results $\Lambda(\boldsymbol{A}^S)$. We denote their outer approximations by

$$\boldsymbol{\omega}_i(\boldsymbol{A}^S) \supseteq \boldsymbol{\lambda}_i(\boldsymbol{A}^S), \quad i = 1, \ldots, n.$$

Let $\rho(\cdot)$ be the spectral radius, and $|\cdot|$ the matrix absolute value, understood componentwise. Our main theoretical result is the following. It employs the sufficient regularity conditions by Beeck [3] and Rump [27]; compare Rex & Rohn [22].

Theorem 1. *Let $\lambda^0 \notin \Lambda(\boldsymbol{A})$ and define $\boldsymbol{M} := \boldsymbol{A} - \lambda^0 I$. Then $(\lambda^0 + \lambda) \notin \Lambda(\boldsymbol{A})$ for all real λ satisfying*

$$|\lambda| < \frac{1 - \frac{1}{2}\rho\left(|I - QM_c| + |I - QM_c|^T + |Q|M_\Delta + M_\Delta^T|Q|^T\right)}{\frac{1}{2}\rho\left(|Q| + |Q|^T\right)}, \tag{1}$$

where $Q \in \mathbb{R}^{n\times n}$, $Q \neq 0$, is an arbitrary matrix.

Proof. It suffices to prove that every λ satisfying (1) the interval matrix $\boldsymbol{M} - \lambda I = \boldsymbol{A} - \lambda^0 I - \lambda I$ is regular, i.e., consists of nonsingular matrices only.

It is known [22] that an interval matrix $\boldsymbol{B}$ is regular if for any matrix Q one has

$$\rho(|I - QB_c| + |Q|B_\Delta) < 1.$$

Substituting $\boldsymbol{B} := \boldsymbol{M} - \lambda I$ we obtain a sufficient condition for λ not to be an eigenvalue

$$\rho(|I - Q(M_c - \lambda I)| + |Q|M_\Delta) < 1.$$

By theory of non-negative matrices [13], $\rho(A) \leq \rho(B)$ provided $0 \leq A \leq B$. In our case, we have

$$\rho(|I - Q(M_c - \lambda I)| + |Q|M_\Delta) \leq \rho(|I - QM_c| + |\lambda||Q| + |Q|M_\Delta).$$

It holds [14] that $\rho(B) \leq \frac{1}{2}\rho(B + B^T)$ for any $B \geq 0$. Thus, we obtain

$$\rho(|I-QM_c|+|\lambda||Q|+|Q|M_\Delta) \leq \frac{1}{2}\rho\left(|I - QM_c| + |I - QM_c|^T + |\lambda|(|Q| + |Q|^T) + |Q|M_\Delta + M_\Delta^T|Q|^T\right).$$

The resulting matrix in the right-hand side is symmetric, thus we can exhibit the well-known Weyl's theorem [9, 13, 19, 28] on spectral radius of sum of two symmetric matrices: For A, B symmetric, $\rho(A + B) \leq \rho(A) + \rho(B)$. Thus

$$\begin{aligned}
&\frac{1}{2}\rho(|I - QM_c| + |I - QM_c|^T + |\lambda|(|Q| + |Q|^T) + |Q|M_\Delta + M_\Delta^T|Q|^T) \\
&\leq \frac{1}{2}|\lambda|\,\rho\left(|Q| + |Q|^T\right) + \frac{1}{2}\rho\left(|I - QM_c| + |I - QM_c|^T + |Q|M_\Delta + M_\Delta^T|Q|^T\right).
\end{aligned}$$

Now, the sufficient condition states as follows

$$\leq \frac{1}{2}|\lambda|\,\rho\left(|Q| + |Q|^T\right) + \frac{1}{2}\rho\left(|I - QM_c| + |I - QM_c|^T + |Q|M_\Delta + M_\Delta^T|Q|^T\right) < 1.$$

By eliminating $|\lambda|$ we get the final form (1). Note that the denominator is zero iff $Q = 0$. □

Note that $M_\Delta = A_\Delta$ and $M_c = A_c - \lambda^0 I$, so one can simply rewrite (1) by means of $\boldsymbol{A}$ and λ^0.

By putting $Q := M_c^{-1}$ we have the most convenient and simple form of the formula (1) as stated below. Note however that using floating point arithmetics may affect validity of these conditions. Thus, from the numerical stability point of view, it is recommended to use the original form with Q being an approximate inverse of M_c.

Corollary 1. *Let $\lambda^0 \notin \Lambda(\boldsymbol{A})$ and define $\boldsymbol{M} := \boldsymbol{A} - \lambda^0 I$. Then $(\lambda^0 + \lambda) \notin \Lambda(\boldsymbol{A})$ for all real λ satisfying*

$$|\lambda| < \frac{1 - \frac{1}{2}\,\rho\left(|M_c^{-1}|M_\Delta + M_\Delta^T|M_c^{-1}|^T\right)}{\frac{1}{2}\,\rho\left(|M_c^{-1}| + |M_c^{-1}|^T\right)}. \tag{2}$$

Another simple consequence is obtained for the case of a symmetric interval matrix and its unsymmetric counterpart. Indeed, for $\boldsymbol{A}$ and $\boldsymbol{A}^S$ the results are the same as long as A_Δ and A_c are symmetric.

Corollary 2. *Let $\lambda^0 \notin \Lambda(\boldsymbol{A}^S)$ and define $\boldsymbol{M}^S := \boldsymbol{A}^S - \lambda^0 I$. Then $(\lambda^0 + \lambda) \notin \Lambda(\boldsymbol{A}^S)$ for all real λ satisfying*

$$|\lambda| < \frac{1 - \frac{1}{2}\,\rho\left(|I - QM_c| + |I - M_cQ| + |Q|M_\Delta + M_\Delta|Q|\right)}{\frac{1}{2}\,\rho(|Q|)}, \tag{3}$$

where $Q \in \mathbb{R}^{n\times n}$, $Q \neq 0$, is an arbitrary symmetric matrix.

These results allow us to propose an efficient filtering method for reducing outer estimations of the eigenvalue set. The detailed algorithm is presented in the following section.

3 Algorithm

In this section we propose a filtering algorithm which is based on Theorem 1. Let an interval $\boldsymbol{a} = [\underline{a}, \overline{a}]$ be given. A filtering method is a method which iteratively cuts off some parts (margins) from $\boldsymbol{a}$ that do not include any eigenvalue. Finally, we obtain an interval $\boldsymbol{b} \subseteq \boldsymbol{a}$ such that $(\boldsymbol{a} \setminus \boldsymbol{b}) \cap \Lambda(\boldsymbol{A}) = \emptyset$.

To avoid infinitely many iterations we limit the number by a constant T. To omit the steps that cut off very narrow pieces, we repeat the main loop while the reduction is significant; that is, we prune away at least εa_Δ part of the interval, where $\varepsilon \in (0, 1)$ is given accuracy. The pseudo-code that follows presents our filtering method that "filter" the input intervals from above. Filtering from below is analogous.

The filtering method is quite straightforward. The input interval $\boldsymbol{a}$ could be any initial outer estimation of $\Lambda(\boldsymbol{A})$, or we can split such an outer estimation into several pieces and call the filtering algorithm for all of them. The former approach does not detect gaps which are inside the non-convex set $\Lambda(\boldsymbol{A})$, while

Algorithm 1 (Filtering $\boldsymbol{a}$ from above)

1: $\boldsymbol{b} := \boldsymbol{a}$;
2: $t := 0$;
3: $\lambda := \varepsilon b_\Delta + 1$;
4: **while** $\lambda > \varepsilon b_\Delta$ and $t < T$ **do**
5: $t := t + 1$;
6: $\boldsymbol{M} := \boldsymbol{A} - \overline{b} I$;
7: compute $Q := M_c^{-1}$;
8: $\lambda := \frac{2\rho\left(|I - QM_c| + |I - QM_c|^T + |Q|M_\Delta + M_\Delta^T |Q|^T\right)}{\rho(|Q| + |Q|^T)}$;
9: **if** $\lambda > 0$ **then**
10: $\overline{b} := \overline{b} - \lambda$;
11: **end if**
12: **if** $\overline{b} < \underline{b}$ **then**
13: **return** $\boldsymbol{b} := \emptyset$;
14: **end if**
15: **end while**
16: **return** $\boldsymbol{b}$.

the latter is able to identify them, provided some genericity condition for the splitting.

Algorithm 1 is also applicable for the symmetric eigenvalue problem, but the filtering of $\Lambda(\boldsymbol{A}^S)$ yields the same result, as in the generic case, $\Lambda(\boldsymbol{A})$. The only advantage of the symmetric case is that we can filter directly outer approximations of the eigenvalue sets $\boldsymbol{\lambda}_i(\boldsymbol{A}^S)$, $i = 1, \ldots, n$. These eigenvalue sets have no gaps inside, so it is the most convenient utilization of the filtering. However, the filtering is applicable only for non-overlapping parts; if they overlap, then we will cut off nothing. As we will see in Section 4, the filtering runs very fast, and the reduction is significant. However, it does not converge to the optimal boundaries in general, because it is based on the sufficient condition for interval matrix regularity.

4 Numerical results

Herein we present some examples and numerical results illustrating properties of the proposed filtering method. In all the examples, we call Algorithm 1 with the accuracy coefficient $\varepsilon := 0.01$ and the maximum number of iterations $T := 100$.

The results were carried on an Intel `Pentium(R) 4`, CPU 3.4 GHz, with 2GB RAM, and the program was written in `C++`. We use `GLPK v.4.23` [18] for solving linear programming problems, `CLAPACK v.3.1.1` for its linear algebraic routines, and `PROFIL/BIAS v.2.0.4` [15] for interval arithmetic and basic operations. We have to notice, however, that routines of `GLPK` and `CLAPACK`[1] do not produce verified solutions, and for real-life problems preferably verified software or interval arithmetic should be used.

Example 1. Let us adopt an example by Hladík et al. [12]

$$\boldsymbol{A} = \begin{pmatrix} [-5,-4] & [-9,-8] & [14,15] & [4.6,5] & [-1.2,-1] \\ [17,18] & [17,18] & [1,2] & [4,5] & [10,11] \\ [17,17.2] & [-3.5,-2.7] & [1.9,2.1] & [-13,-12] & [6,6.4] \\ [18,19] & [2,3] & [18,19] & [5,6] & [6,7] \\ [13,14] & [18,19] & [9,10] & [-18,-17] & [10,11] \end{pmatrix}.$$

The Rohn's outer estimation [12, 24] of $\Lambda(\boldsymbol{A})$ is $[-22.1040, 35.4999]$. Calling Algorithm 1 we obtain the following sequences of improvement:

- from above: $35.4999 \to 28.0615 \to 25.6193 \to 24.7389 \to 24.4086$;
- from bellow: $(-22.1040) \to (-18.4018) \to (-17.8239) \to (-17.7346)$.

So we need only seven iterations to achieve the much more tighter outer approximation $[-17.7346, 24.4086]$.

Using Proposition 2 of [12] we have an outer approximation $[-24.4860, 4.5216] \cup [12.1327, 29.3101]$. We will filter both the intervals. In the former case we obtain

- from above: $4.5216 \to 2.4758 \to 0.8342 \to (-0.0951) \to (-0.5335) \to (-0.7149)$;
- from bellow: $(-24.4860) \to (-18.8351) \to (-17.8926) \to (-17.7438)$;

and in the latter one

- from above: $29.3101 \to 26.0645 \to 24.9010 \to 24.4704 \to 24.3053 \to 24.2412$;
- from bellow: $12.1327 \to 13.4809 \to 14.4703 \to 15.1443 \to 15.5761 \to 15.8462 \to 16.0127 \to 16.1143 \to 16.1760$.

Thus, we have in 21 iterations the filtered outer approximation $[-17.7438, -0.7149] \cup [16.1760, 24.2412]$. We can compare this result with the exact solution

$$\Lambda(\boldsymbol{A}) = [-17.5116, -13.7578] \cup [-6.7033, -1.4582] \cup [16.7804, 23.6143].$$

It was obtained by the algorithm of Hladík et al. [11]. We see that the filtered approximation is very tight. There is still one gap remaining which we cannot detect unless we divide the initial approximation into more sub-intervals.

Example 2. Consider the example given by Qiu et al. [20] (see also [12, 29]):

$$\boldsymbol{A}^S = \begin{pmatrix} [2975,3025] & [-2015,-1985] & 0 & 0 \\ [-2015,-1985] & [4965,5035] & [-3020,-2980] & 0 \\ 0 & [-3020,-2980] & [6955,7045] & [-4025,-3975] \\ 0 & 0 & [-4025,-3975] & [8945,9055] \end{pmatrix}^S.$$

To call the filtering method we need some initial outer approximation of the eigenvalue sets. We use the following one by [12]

$$\boldsymbol{\omega}_1(\boldsymbol{A}^S) = [12560.6296,\ 12720.2273],\ \ \boldsymbol{\omega}_2(\boldsymbol{A}^S) = [6990.7616,\ 7138.1800],$$
$$\boldsymbol{\omega}_3(\boldsymbol{A}^S) = [3320.2863,\ 3459.4322],\ \ \boldsymbol{\omega}_4(\boldsymbol{A}^S) = [837.0637,\ 973.1993].$$

Even though this approximation is quite tight, the filtering makes it more tighter. Calling Algorithm 1 we get in only ten iterations

$$\boldsymbol{\omega}_1^f(\boldsymbol{A}^S) = [12560.8129,\ 12720.2273],\ \ \boldsymbol{\omega}_2^f(\boldsymbol{A}^S) = [6999.7862,\ 7129.2716],$$
$$\boldsymbol{\omega}_3^f(\boldsymbol{A}^S) = [3332.7164,\ 3447.4625],\ \ \boldsymbol{\omega}_4^f(\boldsymbol{A}^S) = [841.5328,\ 968.5845].$$

What if we start with another initial outer approximation? We use that produced by of the method of Leng & He [17]

$$\widetilde{\boldsymbol{\omega}}_1(\boldsymbol{A}^S) = [12550.53,\ 12730.53],\ \ \widetilde{\boldsymbol{\omega}}_2(\boldsymbol{A}^S) = [6974.459,\ 7154.459],$$
$$\widetilde{\boldsymbol{\omega}}_3(\boldsymbol{A}^S) = [3299.848,\ 3479.848],\ \ \widetilde{\boldsymbol{\omega}}_4(\boldsymbol{A}^S) = [815.1615,\ 995.1615].$$

Although this estimation is not so tight we obtain tighter result

$$\widetilde{\boldsymbol{\omega}}_1^f(\boldsymbol{A}^S) = [12560.8129,\ 12720.2472],\ \ \widetilde{\boldsymbol{\omega}}_2^f(\boldsymbol{A}^S) = [6999.8026,\ 7129.2716],$$
$$\widetilde{\boldsymbol{\omega}}_3^f(\boldsymbol{A}^S) = [3332.7944,\ 3447.4628],\ \ \widetilde{\boldsymbol{\omega}}_4^f(\boldsymbol{A}^S) = [841.5328,\ 968.5505],$$

and the total number of iteration is 13. We can compare this outer approximation with the exact description [12]

$$\boldsymbol{\lambda}_1(\boldsymbol{A}^S) = [12560.8377,\ 12720.2273],\ \ \boldsymbol{\lambda}_2(\boldsymbol{A}^S) = [7002.2828,\ 7126.8283],$$
$$\boldsymbol{\lambda}_3(\boldsymbol{A}^S) = [3337.0785,\ 3443.3127],\ \ \boldsymbol{\lambda}_4(\boldsymbol{A}^S) = [842.9251,\ 967.1082],$$

Again, we see that the filtering method converges quickly to the tight solution. Moreover, the results are not sensitive to the initial estimation chosen.

Example 3. To be fully convinced about the quality of the filtering method we carried out number of randomly generated examples. Components of the midpoint matrix A_c are taken randomly with uniform distribution in $[-20, 20]$. Components of the radius matrix A_Δ are taken randomly with uniform distribution in $[0, R]$, where R is a given positive real number. We applied our algorithm on the interval matrix $\boldsymbol{M} := \boldsymbol{A}^T\boldsymbol{A}$ since such kinds of symmetric interval matrices often appear in practice. The filtering method was called for all the eigenvalue sets $\boldsymbol{\lambda}(\boldsymbol{M}^S) = \big(\boldsymbol{\lambda}_1(\boldsymbol{M}^S), \ldots, \boldsymbol{\lambda}_n(\boldsymbol{M}^S)\big)^T$.

The results are displayed in Table 1. Each row shows results of a series of 100 tests carried out for a given dimension n and the parameter R. We provide average cut off and its standard deviation, number of iterations (for all parts of an outer approximation together) and average running time. The cut off provides information about the filtering efficiency. It is measured by the ratio

$$1 - \frac{e^T\omega_\Delta^f(\boldsymbol{M}^S)}{e^T\omega_\Delta(\boldsymbol{M}^S)},$$

where $e = (1, \ldots, 1)^T$ denotes the vector of all ones, $\boldsymbol{\omega}^f(\boldsymbol{M}^S)$ an initial outer approximation of $\boldsymbol{\lambda}(\boldsymbol{M}^S)$, and $\boldsymbol{\omega}^f(\boldsymbol{M}^S)$ the result of the filtering procedure. This quotient says how much we cut off from the whole outer approximations of the eigenvalue sets, but the real efficiency (how much we eliminate from the overestimation addition) is much better.

The results show that the proposed filtering method is not only very fast, but also efficient and eliminates quite large parts of given outer approximation of eigenvalue sets. This is particularly true when the input intervals of $\boldsymbol{A}$ are narrow. If they are wide then the filtering method is not so successful, partially because some of the eigenvalue sets overlap.

n	R	cut off quotient		iterations	time
		average	std. deviation		
5	0.001	0.186185	0.034635	15.25	0.0012 s
5	0.01	0.182395	0.042879	16.23	0.0010 s
5	0.1	0.142715	0.041260	16.51	0.0011 s
5	1	0.015686	0.011977	3.99	0.0004 s
10	0.001	0.233480	0.022064	33.41	0.0062 s
10	0.01	0.207711	0.032813	37.63	0.0068 s
10	0.1	0.074682	0.024832	22.13	0.0031 s
10	1	0.003388	0.002995	1.42	0.0005 s
15	0.001	0.242457	0.021210	54.13	0.0239 s
15	0.01	0.181559	0.022123	56.44	0.0250 s
15	0.1	0.025517	0.010055	13.48	0.0082 s
15	1	0.001410	0.001704	0.94	0.0011 s
20	0.001	0.243601	0.017704	76.69	0.0650 s
20	0.01	0.154207	0.021279	67.92	0.0595 s
20	0.1	0.009844	0.006030	7.42	0.0114 s
20	1	0.000493	0.000890	0.60	0.0012 s
25	0.001	0.238694	0.016706	97.15	0.1373 s
25	0.01	0.122852	0.017385	73.67	0.1122 s
25	0.1	0.004785	0.003815	3.84	0.0123 s
25	1	0.000117	0.000369	0.33	0.0033 s
30	0.001	0.232266	0.015133	117.90	0.2679 s
30	0.01	0.093589	0.015072	74.07	0.1812 s
30	0.1	0.002288	0.001849	2.78	0.0121 s
30	1	0.000031	0.000150	0.09	0.0036 s
50	0.001	0.194401	0.011126	184.75	1.5238 s
50	0.01	0.028148	0.006065	48.19	0.5169 s
50	0.1	0.000428	0.000545	1.05	0.0225 s
50	1	0.000000	0.000000	0.00	0.0166 s

Table 1: Filtering procedure for outer estimates of eigenvalue sets of random interval symmetric matrices $\boldsymbol{A}^T\boldsymbol{A}$.

5 Conclusion

We propose a filtering method for improving an outer approximation of the eigenvalue set of an interval matrix. Our method is applicable for both generic and symmetric matrices. Even though the proposed algorithm does not converge always to the optimal bounds, our numerical experiments show that in general they compute very fast, quite accurate results. The algorithm performs well even when the initial (input) outer approximation is not very tight, thus it is not sensitive with respect to the input estimation. A drawback of our approach is that it can not detect (possible) gaps inside the initial outer approximation. Such cases should be handled by splitting into smaller sub-intervals or using another kind of initial approximation. This is problem that further research is needed.

References

[1] CLAPACK – Linear Algebra PACKage written for C. `http://www.netlib.org/clapack/`.

[2] O. Beaumont. An algorithm for symmetric interval eigenvalue problem. Technical Report IRISA-PI-00-1314, Institut de recherche en informatique et systèmes aléatoires, Rennes, France, 2000.

[3] H. Beeck. Zur Problematik der Hüllenbestimmung von Intervallgleichungssystemen. In K. Nickel, editor, *Interval Mathemantics: Proceedings of the International Symposium on Interval Mathemantics*, volume 29, pages 150–159, Berlin, 1975. Springer.

[4] D. Chablat, P. Wenger, F. Majou, and J. Merlet. An interval analysis based study for the design and the comparison of 3-dof parallel kinematic machines. *Int. J. Robot. Res.*, 23(6):615–624, 2004.

[5] P. Comon et al. Independent component analysis, a new concept? *Signal processing*, 36(3):287–314, 1994.

[6] A. S. Deif. The interval eigenvalue problem. *Z. Angew. Math. Mech.*, 71(1):61–64, 1991.

[7] A. D. Dimarogonas. Interval analysis of vibrating systems. *J. Sound Vib.*, 183(4):739–749, 1995.

[8] F. Gioia and C. N. Lauro. Principal component analysis on interval data. *Comput. Stat.*, 21(2):343–363, 2006.

[9] G. H. Golub and C. F. Van Loan. *Matrix computations. 3rd ed.* Johns Hopkins University Press, 1996.

[10] D. Hertz. The extreme eigenvalues and stability of real symmetric interval matrices. *IEEE Trans. Autom. Control*, 37(4):532–535, 1992.

[11] M. Hladík, D. Daney, and E. Tsigaridas. An algorithm for the real interval eigenvalue problem. Research Report RR-6680, INRIA, France, http://hal.inria.fr/inria-00329714/en/, October 2008. sumbitted to J. Comput. Appl. Math.

[12] M. Hladík, D. Daney, and E. Tsigaridas. Bounds on eigenvalues and singular values of interval matrices. Research Report inria-00370603:1, INRIA, France, http://hal.inria.fr/inria-00370603/en/, March 2009. sumbitted to SIAM J. Matrix Anal. Appl.

[13] R. A. Horn and C. R. Johnson. *Matrix analysis.* Cambridge University Press, Cambridge, 1985.

[14] R. A. Horn and C. R. Johnson. *Topics in matrix analysis.* Cambridge University Press, Cambridge, 1994.

[15] O. Knüppel, D. Husung, and C. Keil. PROFIL/BIAS – a C++ class library. http://www.ti3.tu-harburg.de/Software/PROFILEnglisch.html.

[16] L. V. Kolev. Outer interval solution of the eigenvalue problem under general form parametric dependencies. *Reliab. Comput.*, 12(2):121–140, 2006.

[17] H. Leng and Z. He. Computing eigenvalue bounds of structures with uncertain-but-non-random parameters by a method based on perturbation theory. *Commun. Numer. Methods Eng.*, 23(11):973–982, 2007.

[18] A. Makhorin. GLPK – GNU Linear Programming Kit. http://www.gnu.org/software/glpk/.

[19] C. D. Meyer. *Matrix analysis and applied linear algebra.* SIAM, Philadelphia, 2000.

[20] Z. Qiu, S. Chen, and I. Elishakoff. Bounds of eigenvalues for structures with an interval description of uncertain-but-non-random parameters. *Chaos Solitons Fractals*, 7(3):425–434, 1996.

[21] Z. Qiu, P. C. Müller, and A. Frommer. An approximate method for the standard interval eigenvalue problem of real non-symmetric interval matrices. *Commun. Numer. Methods Eng.*, 17(4):239–251, 2001.

[22] G. Rex and J. Rohn. Sufficient conditions for regularity and singularity of interval matrices. *SIAM J. Matrix Anal. Appl.*, 20(2):437–445, 1998.

[23] J. Rohn. Interval matrices: Singularity and real eigenvalues. *SIAM J. Matrix Anal. Appl.*, 14(1):82–91, 1993.

[24] J. Rohn. Bounds on eigenvalues of interval matrices. *ZAMM, Z. Angew. Math. Mech.*, 78(Suppl. 3):S1049–S1050, 1998.

[25] J. Rohn. A handbook of results on interval linear problems. http://www.cs.cas.cz/rohn/handbook, 2005.

[26] J. Rohn and A. Deif. On the range of eigenvalues of an interval matrix. *Comput.*, 47(3-4):373–377, 1992.

[27] S. M. Rump. Solving algebraic problems with high accuracy. In U. Kulisch and W. Miranker, editor, *A new approach to scientific computation, Proc. Symp.*, pages 51–120. Academic Press, New York, 1983.

[28] J. Wilkinson. *The algebraic eigenvalue problem. 1. paperback ed.* Clarendon Press, Oxford, 1988.

[29] Q. Yuan, Z. He, and H. Leng. An evolution strategy method for computing eigenvalue bounds of interval matrices. *Appl. Math. Comput.*, 196(1):257–265, 2008.

Centre de recherche INRIA Sophia Antipolis – Méditerranée
2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Centre de recherche INRIA Bordeaux – Sud Ouest : Domaine Universitaire - 351, cours de la Libération - 33405 Talence Cedex
Centre de recherche INRIA Grenoble – Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier
Centre de recherche INRIA Lille – Nord Europe : Parc Scientifique de la Haute Borne - 40, avenue Halley - 59650 Villeneuve d'Ascq
Centre de recherche INRIA Nancy – Grand Est : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex
Centre de recherche INRIA Paris – Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex
Centre de recherche INRIA Rennes – Bretagne Atlantique : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex
Centre de recherche INRIA Saclay – Île-de-France : Parc Orsay Université - ZAC des Vignes : 4, rue Jacques Monod - 91893 Orsay Cedex

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399