Applied Mathematics and Computation 217 (2011) 5236–5242

Contents lists available at ScienceDirect

Applied Mathematics and Computation

journal homepage: www.elsevier.com/locate/amc

A filtering method for the interval eigenvalue problem

Milan Hladík [a,b,*], David Daney [b], Elias Tsigaridas [c]

[a] Charles University, Faculty of Mathematics and Physics, Department of Applied Mathematics, Malostranské nám. 25, 11800 Prague, Czech Republic
[b] INRIA Sophia-Antipolis Méditerranée, 2004 route des Lucioles, BP 93, 06902 Sophia-Antipolis Cedex, France
[c] Department of Computer Science, Aarhus University, IT-parken, Aabogade 34, DK 8200 Aarhus N, Denmark

ARTICLE INFO

Keywords:
Interval matrix
Symmetric matrix
Interval analysis
Eigenvalue
Eigenvalue bounds

ABSTRACT

We consider the general problem of computing intervals that contain the real eigenvalues of interval matrices. Given an outer approximation (superset) of the real eigenvalue set of an interval matrix, we propose a filtering method that iteratively improves the approximation. Even though our method is based on a sufficient regularity condition, it is very efficient in practice and our experimental results suggest that it improves, in general, significantly the initial outer approximation. The proposed method works for general, as well as for symmetric interval matrices.

© 2010 Elsevier Inc. All rights reserved.

1. Introduction

In order to model real-life problems and perform computations we *must* deal with inaccuracy and inexactness; these are due to measurement, to simplification assumption on physical models, to variations of the parameters of the system, and finally 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 encountered 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) cover all the possibilities for the input quantities.

We are interested in the interval real eigenvalue problem. Given a matrix the elements of which are real intervals, also called interval matrix, the task is to compute real intervals that contain all possible eigenvalues. For formal definitions we refer the reader to the next section.

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 [1], mass structures [2], vibrating systems [3], robotics [4], and even principal component analysis [5] and independent component analysis [6], which could be considered as a statistics oriented applications. Using the well-known Jordan–Wielandt transformation [7,8], if we are given a solution of the interval real eigenvalue problem, we can provide an approximation for the singular values and the condition number; both quantities have numerous applications.

The first general results for the interval real eigenvalue problem were produced by Deif [9], and Deif and Rohn [10]. However their solutions depend on theorems that have very strong assumptions. Later, Rohn [11], introduced a boundary point characterization of the eigenvalue set. Approximation methods were addressed by Qiu et al. [1], Leng et al. [12] and by Hladík et al. [13]. The works [12,13] are based on a branch and prune approach and yield results that depend on a given arbitrarily high accuracy.

* Corresponding author at: Charles University, Faculty of Mathematics and Physics, Department of Applied Mathematics, Malostranské nám. 25, 11800 Prague, Czech Republic.

E-mail addresses: milan.hladik@matfyz.cz, hladik@kam.mff.cuni.cz (M. Hladík), david.daney@sophia.inria.fr (D. Daney), elias.tsigaridas@gmail.com (E. Tsigaridas).

0096-3003/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.amc.2010.09.066

The symmetric eigenvalue problem is very important in practice. However, in its interval form it is hard to handle with since the correlations between the entries of the matrices force the algorithms of interval analysis to overestimate actually a lot the results. The symmetric case was pioneered by Deif [9]. Another theoretical result is attributed to Hertz [14], see also [15], to determine two extremal points of the eigenvalue set. Diverse approximation algorithms have also been developed. An evolution strategy method by Yuan et al. [16] gives an inner approximation (subset) of the eigenvalues set, and under some conditions it converges to the exact bounds. Matrix perturbation theory was used by Qiu et al. [2], who proposed an algorithm for approximating the bounds, and by Leng and He [17] for outer approximation (superset) of the eigenvalue set. Outer bounds that are easy and fast to compute were presented by Hladík et al. [18], and the general parametric case was considered by Kolev [19].

In this paper, we propose a filtering method to reduce the overestimation produced by various methods. Generally, filtering methods start with an initial outer approximation and iteratively make it tighter. Even though filtering is commonly used approach in constrained programming, it is not widely used 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. [13]. To the best of our knowledge, the only related work is by Beaumont [20], where an iterative algorithm is presented, based on convex approximation of eigenpairs. The new filtering method that we propose is more simple and applicable for both the symmetric and unsymmetric cases. Since we do not take into account eigenvectors, it is much more efficient, too.

2. Basic definitions and main theorem

An interval matrix is defined as a family of matrices

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

where $\underline{A},\ \overline{A} \in \mathbb{R}^{m\times n}$, $\underline{A} \leqslant \overline{A}$, are given matrices, and the inequality is considered element-wise. By $A_c := \frac{1}{2}(\underline{A} + \overline{A})$, and $A_\Delta := \frac{1}{2}(\overline{A} - \underline{A})$ we denote the midpoint and the 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

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

That is, matrices in $\boldsymbol{A}$ may have both real and complex eigenvalues, but we focus on the real ones only. 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

$$\boldsymbol{A}^S := \{A \in \boldsymbol{A} | A = A^T\}.$$

Without loss of generality assume that $\boldsymbol{A}^S$ is non-empty, which is easy to check, and that A_Δ and A_c are symmetric. Its eigenvalue set is denoted similarly to generic case, that is

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

Since $\boldsymbol{A}^S$ is a proper subset of $\boldsymbol{A}$, 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

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

We extend this notation for symmetric interval matrices, that is $\boldsymbol{\lambda}_i(\boldsymbol{A}^S) := \left\{\lambda_i(A) | A \in \boldsymbol{A}^S\right\}$. These sets form n convex compact intervals, which can be disjoint or may overlap, see for example [18]. The union of these interval results in $\Lambda(\boldsymbol{A}^S)$. We denote their outer approximations by $\boldsymbol{\omega}_i(\boldsymbol{A}^S) \supseteq \boldsymbol{\lambda}_i(\boldsymbol{A}^S)$, where $i = 1,\ldots,n$.

Let $\rho(\cdot)$ be the spectral radius, and $|\cdot|$ the matrix absolute value, understood componentwise. We now present our main theoretical result, which employs the sufficient regularity conditions by Beeck [21] and Rump [22] (compare Rex and Rohn [23]), and allows us to present the filtering method.

Theorem 1. *Let $\lambda^0 \notin \Lambda(\boldsymbol{A})$ and define $\mathbf{M} := \boldsymbol{A} - \lambda^0 I$. 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 is sufficient to prove that every λ satisfying (1) the interval matrix $\mathbf{M} - \lambda I = \boldsymbol{A} - \lambda^0 I - \lambda I$ is regular, i.e., consists of nonsingular matrices only.

It is known [23] that an interval matrix B is regular if for any matrix Q one has

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

Substituting $B := \mathbf{M} - \lambda I$ 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 [8], $\rho(A) \leqslant \rho(B)$ provided $0 \leqslant A \leqslant B$. In our case, we have

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

It holds [24] that $\rho(B) \leqslant \frac{1}{2}\rho(B + B^T)$ for any $B \geqslant 0$. Thus, we obtain

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

The resulting matrix in the right-hand side is symmetric, thus we can exploit the well-known Weyl's theorem [7,8], on spectral radius of sum of two symmetric matrices: For A, B symmetric, $\rho(A + B) \leqslant \rho(A) + \rho(B)$. Thus

$$\begin{aligned}&\frac{1}{2}\rho(|I - QM_c| + |I - QM_c|^T + |\lambda|(|Q| + |Q|^T) + |Q|M_\Delta + M_\Delta^T|Q|^T)\\ &\quad\leqslant \frac{1}{2}|\lambda|\rho\Big(|Q| + |Q|^T\Big) + \frac{1}{2}\rho\Big(|I - QM_c| + |I - QM_c|^T + |Q|M_\Delta + M_\Delta^T|Q|^T\Big).\end{aligned}$$

Now, the sufficient condition states as follows

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

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

If we set $Q := M_c^{-1}$, then we have the most convenient and simple form of (1), which we state below. However, when we use floating point arithmetic these conditions may be violated. Thus, from the numerical stability point of view, it is recommended to use the original form using Q as the approximate inverse of M_c.

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

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

Another consequence follows for the case of a symmetric interval matrix. Notice that for $\boldsymbol{A}$ and $\boldsymbol{A}^S$ the filtering 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 $\mathbf{M}^S := \boldsymbol{A}^S - \lambda^0 I$. Then $(\lambda^0 + \lambda) \notin \Lambda(\boldsymbol{A}^S)$ for all real λ satisfying*

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

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 approximations of the eigenvalue set. The algorithm is presented in the next 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 $(\boldsymbol{a} \backslash \mathbf{b}) \cap \Lambda(\boldsymbol{A}) = \emptyset$.

To avoid infinitely many iterations we limit the number by a constant T. To skip the steps that cut off very narrow pieces, we repeat the main loop when the reduction is significant; that is, we prune away at least $\varepsilon \cdot a_\Delta$ part of the interval, where

$\varepsilon \in (0,1)$ is a given accuracy. The pseudo-code of Algorithm 1 presents our filtering method that "filters" the input intervals from above. Filtering from below is analogous.

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

```
b := a;
t := 0;
λ := εb_Δ + 1;
while λ > εb_Δ and t < T do
   t := t + 1;
   M := A − b̄I;
   compute Q := M_c^{-1};
   λ := (2 − ρ(|I − QM_c| + |I − QM_c|^T + |Q|M_Δ + M_Δ^T|Q|^T)) / ρ(|Q| + |Q|^T);
   if λ > 0 then
      b̄ := b̄ − λ;
   end if
   if b̄ < b̲ then
      return b := ∅;
   end if
end while
return b.
```

The filtering method is quite straightforward. The input interval **a** could be any initial outer approximation of $\Lambda(\boldsymbol{A})$, or we can split such an outer approximation 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 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 advantage of the symmetric case is that $\Lambda(\boldsymbol{A}^S)$ consists of a union of convex sets $\lambda_i(\boldsymbol{A}^S)$, $i = 1,\dots,n$, and convex sets do not have gaps. Thus we can employ methods [15] for computing outer approximations of the eigenvalue sets $\lambda_i(\boldsymbol{A}^S)$, $i = 1,\dots,n$, and do filtering for these particular sets separately. Note that the filtering is applicable only for non-overlapping parts; if they overlap, then we 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

In this section we present three examples and numerical results illustrating properties of the proposed filtering method. In all the examples, we call Algorithm 1 with accuracy coefficient $\varepsilon := 0.01$. The maximum number of iterations $T := 100$. The results were carried on an Intel `Pentium (R) 4`, CPU 3.4 GHz, with 2 GB RAM, and the program was written in `C++`. We use `GLPK v.4.23` [25] for solving linear programming problems, `CLAPACK v.3.1.1` [26] for its linear algebraic routines, and `PROFIL/BIAS v.2.0.4` [27] for interval arithmetic and basic operations. We have to notice, however, that routines of `GLPK` and `CLAPACK` 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. [18]

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

The Rohn's outer approximation [18,28] 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 [18] 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 the following approximations.

- 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 we obtain

- 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, which is $\Lambda(\boldsymbol{A}) = [-17.5116, -13.7578] \cup [-6.7033, -1.4582] \cup [16.7804, 23.6143]$. It was obtained by the algorithm of Hladík et al. [13].

We notice that the filtered approximation is very tight. There is one gap remaining which we cannot detect unless we divide the initial approximation to sub-intervals.

Example 2. Consider the example given by Qiu et al. [2] (see also [18,16]):

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

To call the filtering method we need some initial outer approximation of the eigenvalue sets. We use the following one by [18]:

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

Even though this approximation is quite tight, the filtering makes it more tighter. Calling Algorithm 1 we get after only 10 iterations:

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

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

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

Even though the initial estimation is not so tight, our filter method improves it substantially after only 13 iterations to

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

We can compare this outer approximation with the exact description from [12,18]:

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

As in the previous example the filtering method converges quickly to the solutions. Moreover, it does not seem to be sensitive to the initial approximation.

Example 3. To be fully convinced about the quality of the filtering method we carried out a 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 $\mathbf{M} := \boldsymbol{A}^T\boldsymbol{A}$ since such kinds of symmetric interval matrices often appear in practice. The filtering method was called for all the eigenvalue sets $\boldsymbol{\lambda}(\mathbf{M}^S) = \left(\boldsymbol{\lambda}_1(\mathbf{M}^S), \ldots, \boldsymbol{\lambda}_n(\mathbf{M}^S)\right)^T$.

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

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

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, average 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 $1-\frac{e^T\omega^f_\Delta(\mathbf{M}^S)}{e^T\omega_\Delta(\mathbf{M}^S)}$, where $e=(1,\dots,1)^T$ denotes the vector of all ones, $\omega^f(\mathbf{M}^S)$ an initial outer approximation of $\lambda(\mathbf{M}^S)$, and $\omega^f(\mathbf{M}^S)$ 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 A are narrow. If they are wide then the filtering method is not so successful, partially because some of the eigenvalue sets overlap.

5. Conclusion

We propose a filtering method to improve 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 they compute very fast and quite accurate results in general. 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 cannot 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 a problem whose further research is needed.

References

[1] Z. Qiu, P.C. Müller, A. Frommer, An approximate method for the standard interval eigenvalue problem of real non-symmetric interval matrices, Commun. Numer. Methods Eng. 17 (2001) 239–251.
[2] Z. Qiu, S. Chen, I. Elishakoff, Bounds of eigenvalues for structures with an interval description of uncertain-but-non-random parameters, Chaos Soliton. Fract. 7 (1996) 425–434.
[3] A.D. Dimarogonas, Interval analysis of vibrating systems, J. Sound Vib. 183 (1995) 739–749.
[4] D. Chablat, P. Wenger, F. Majou, J. Merlet, An interval analysis based study for the design and the comparison of 3-dof parallel kinematic machines, Int. J. Robot. Res. 23 (2004) 615–624.
[5] F. Gioia, C.N. Lauro, Principal component analysis on interval data, Comput. Stat. 21 (2006) 343–363.
[6] P. Comon et al., Independent component analysis, a new concept? Signal process. 36 (1994) 287–314.

[7] G.H. Golub, C.F. Van Loan, Matrix computations, J. Hopkins Univ. Press, 1996.
[8] R.A. Horn, C.R. Johnson, Matrix Analysis, Cambridge University Press, 1985.
[9] A.S. Deif, The interval eigenvalue problem, Z. Angew. Math. Mech. 71 (1991) 61–64.
[10] J. Rohn, A. Deif, On the range of eigenvalues of an interval matrix, Computing 47 (1992) 373–377.
[11] J. Rohn, Interval matrices: singularity and real eigenvalues, SIAM J. Matrix Anal. Appl. 14 (1993) 82–91.
[12] H. Leng, Z. He, Q. Yuan, Computing bounds to real eigenvalues of real-interval matrices, Int. J. Numer. Methods Eng. 74 (2008) 523–530.
[13] M. Hladík, D. Daney, E. Tsigaridas, An algorithm for the real interval eigenvalue problem, J. Comput. Appl. Math., accepted for publication, doi:10.1016/j.cam.2010.11.022. <http://dx.doi.org/10.1016/j.cam.2010.11.022>.
[14] D. Hertz, The extreme eigenvalues and stability of real symmetric interval matrices, IEEE Trans. Autom. Control 37 (1992) 532–535.
[15] J. Rohn, A handbook of results on interval linear problems, 2005. <http://www.cs.cas.cz/rohn/handbook>.
[16] Q. Yuan, Z. He, H. Leng, An evolution strategy method for computing eigenvalue bounds of interval matrices, Appl. Math. Comput. 196 (2008) 257–265.
[17] H. Leng, 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 (2007) 973–982.
[18] M. Hladík, D. Daney, E. Tsigaridas, Bounds on real eigenvalues and singular values of interval matrices, SIAM J. Matrix Anal. Appl. 31 (2010) 2116–2129.
[19] L.V. Kolev, Outer interval solution of the eigenvalue problem under general form parametric dependencies, Reliab. Comput. 12 (2006) 121–140.
[20] O. Beaumont, An algorithm for symmetric interval eigenvalue problem, Technical Report IRISA-PI-00-1314, INRIA, Rennes, France, 2000.
[21] H. Beeck, Zur Problematik der Hüllenbestimmung von Intervallgleichungssystemen, in: K. Nickel (Ed.), Proc. Int. Symp. Interval Math., Vol. 29, Springer, Berlin, 1975, pp. 150–159.
[22] S.M. Rump, Solving algebraic problems with high accuracy, in: U. Kulisch, W. Miranker (Eds.), A New Approach to Scientific Computation, Proc. Symp, Academic Press, New York, 1983, pp. 51–120.
[23] G. Rex, J. Rohn, Sufficient conditions for regularity and singularity of interval matrices, SIAM J. Matrix Anal. Appl. 20 (1998) 437–445.
[24] R.A. Horn, C.R. Johnson, Topics in Matrix Analysis, Cambridge Univ. Press, 1994.
[25] A. Makhorin, GLPK – GNU Linear Programming Kit., 2008. <http://www.gnu.org/software/glpk/>.
[26] CLAPACK – Linear Algebra PACKage written for C., 2007. <http://www.netlib.org/clapack/>.
[27] O. Knüppel, D. Husung, C. Keil, PROFIL/BIAS – a C++ class library., 2006. <http://www.ti3.tu-harburg.de/Software/PROFILEnglisch.html>.
[28] J. Rohn, Bounds on eigenvalues of interval matrices, ZAMM, Z. Angew. Math. Mech. 78 (1998) S1049–S1050.