A Method for Determining the Regularity Radius of Interval Matrices*

L. V. Kolev
Dept. of Theoretical Electrotechnics, Faculty of Automatics, Technical University of Sofia, 1000 Sofia, Bulgaria
lkolev@tu-sofia.bg

Abstract

Determination of the regularity radius r^* of interval matrices is known to be a NP-hard problem. In this paper, a method for determining r^* is suggested whose numerical complexity is not *a priori* exponential. The method is based on an equivalent transformation of the original problem to the problem of determining the real maximum magnitude eigenvalue μ^* of an associated interval generalized eigenvalue problem. The latter problem is solved iteratively, using lower bounds $|\underline{\mu}|$ on $|\mu^*|$ and outer interval or interval hull solutions of corresponding linear interval systems. The method is capable of determining the regularity radius r^* if the interval solutions satisfy certain constant sign conditions; otherwise, it provides a tight upper bound $\overline{r}$ or r^*. If the sign conditions are met for the interval matrix considered, r^* is computed in polynomial time. Numerical examples with interval matrices whose size goes up to $n = 500$ illustrate the potential of the method suggested.

Keywords: interval matrices, regularity, regularity radius
AMS subject classifications: 65L15, 65G40

1 Introduction

Let A, $\underline{A}$ and $\overline{A}$ be real $n \times n$ matrices with $\underline{A} \leq \overline{A}$ (the inequality sign is meant component-wise). The interval matrix

$$\boldsymbol{A} = [\underline{A}, \overline{A}] := \{A : \underline{A} \leq A \leq \overline{A}\} \tag{1.1}$$

is called regular if each $A \in \boldsymbol{A}$ is nonsingular. Checking regularity of interval matrices is a fundamental problem since a number of important properties of interval matrices such as Hurwitz stability, positive definiteness or the P-matrix property (with many applications in control [1, 2, 9, 13] and electrical engineering [6, 8, 17, 18]) can be ascertained via verifying regularity [5]. Regularity, however, is a qualitative characteristic. A better, quantitative measure of regularity is the so-called radius of regularity of an

*Submitted: January 28, 2009; Final revision: January 19, 2011; Accepted: June 6, 2011.

interval matrix [10]. Let the interval matrix from (1.1) be represented equivalently in the form

$$\boldsymbol{A} = [A^c - R, A^c + R] \tag{1.2a}$$

where

$$A^c = (\underline{A} + \overline{A})/2, \quad R = (\overline{A} - \underline{A})/2 \tag{1.2b}$$

are the center and radius of $\boldsymbol{A}$, respectively. The regularity radius of $\boldsymbol{A}$ is defined by [5, 10]

$$r^*(\boldsymbol{A}) = \min\{r \leq 0 : [A^c - rR, A^c + rR] \text{ is singular}\}. \tag{1.3}$$

Evidently, $\boldsymbol{A}$ is regular if and only if $r^*(\boldsymbol{A}) > 1$. Knowing $r^*(\boldsymbol{A})$, however, provides a quantitative measure: the distance from singularity. Thus, the knowledge of $r^*(\boldsymbol{A})$ permits to determine the stability radius or positive definiteness radius [5] of interval matrices. (Alternative quantitative measures are the stability margin [8] or the positive definiteness margin [7] of interval matrices.)

Checking regularity or determining the regularity radius $r^*(\boldsymbol{A})$ of an interval matrix is a NP-hard problem [11] (see also [16]). Indeed, the method from [12] for computing $r^*(\boldsymbol{A})$ requires evaluation of 2^n numbers. Obviously, the method is applicable only for matrices of moderate size n ($n \leq 15$, according to [12]).

An interesting approach to circumventing the exponential complexity in the context of verifying regularity has been suggested in [5]. It results in an algorithm that is not *a priori* exponential although corresponding necessary and sufficient conditions are used. Indeed, the algorithm is constructed in such a way that "it requires an exponential number of operations only in the "worst case" examples and behaves reasonably in the "average" ones" [5]. In some of the examples presented in the paper, the algorithm is used to approximately assess the regularity radius r^* through increasing r by an increment δr until $\boldsymbol{A}(r') = [A^c - r'R, A^c + r'R]$ becomes singular for the current $r' = r + \delta r$. Such an approach to assessing r^*, however, is possible if the algorithm for checking regularity (or singularity) from [5] is applicable (i.e., if its numerical complexity is not exponential). Its main disadvantage is that the determination of r^* within a high accuracy would require a considerable, and sometimes prohibitively large, amount of computations (see Example 1 in [5]) even for the cases of non-exponential complexity.

An alternative approach to overcoming the inherent exponentiality of the problem is suggested in this paper. Its main feature is that it is only based on the use of certain appropriately chosen sufficient conditions. Following this approach, a method for determining the regularity radius r^* is devised. It comprises the following stages: (i) equivalent transformation of the original problem to the problem of determining the real maximum magnitude eigenvalue μ^* of an associated interval generalized eigenvalue problem, (ii) iterative solution of the latter problem using lower bounds $|\underline{\mu}|$ on $|\mu^*|$ and outer interval solutions $\boldsymbol{x}$ and $\boldsymbol{y}$ or interval hull solutions $\boldsymbol{x}^*$ and $\boldsymbol{y}^*$ of corresponding linear interval systems. The method is capable of determining the regularity radius r^* if the following sufficient conditions are satisfied: (i) it is possible to compute the interval solutions $\boldsymbol{x}$ and $\boldsymbol{y}$ or $\boldsymbol{x}^*$ and $\boldsymbol{y}^*$ in polynomial time; (ii) the solutions $\boldsymbol{x}$ and $\boldsymbol{y}$ or $\boldsymbol{x}^*$ and $\boldsymbol{y}^*$ satisfy certain sign constancy conditions. The complexity of the method, under these conditions, is polynomial. Otherwise, it only provides a tight upper bound $\overline{r}$ on r^* (but also for polynomial time).

The paper is structured as follows. The transition from the original problem to the equivalent formulation is presented in Section 2. A method for determining the eigenvalue μ^* is suggested in the next section. In Section 4, a detailed algorithm of the method is developed. Numerical examples illustrating the determination of the

regularity radius r^* are provided in Section 5. The paper ends up with concluding remarks in Section 6.

2 Equivalent transformation

The interval matrix $\boldsymbol{A}$ considered is represented equivalently as

$$\boldsymbol{A} = A^c + [-R, R] = A^c + \boldsymbol{B}, \quad \boldsymbol{B} = [-R, R]. \tag{2.1}$$

In view of (1.3) and (2.1), the regularity radius can be defined equivalently as

$$r^*(\boldsymbol{A}) = \min\{r \geq 0 : \det(A^c + rB) = 0,\ B \in \boldsymbol{B}\}. \tag{2.2}$$

We consider the following interval generalized eigenvalue problem

$$Bx = \mu A^0 x, \quad B \in \boldsymbol{B} \tag{2.3a}$$

where

$$A^0 = -A^c. \tag{2.3b}$$

Let μ^* denote the real maximum magnitude (rmm) eigenvalue of (2.3) defined as

$$\mu^* = \max\{|\mu| : Bx = \mu A^0 x, \quad B \in \boldsymbol{B}\}. \tag{2.4}$$

Lemma 2.1 *Let r^* and μ^* be defined as in (2.2) and (2.4), respectively. Then the regularity radius r^* of the interval matrix $\boldsymbol{A}$ considered is given by*

$$r^* = 1/|\mu^*|. \tag{2.5}$$

Proof: Consider the condition

$$\det(A^c + rB) = 0, \quad B \in \boldsymbol{B}$$

in (2.2). It can be written equivalently as a generalized interval eigenvalue problem

$$A^0 x = rBx, \quad B \in \boldsymbol{B} \tag{2.6}$$

with A^0 defined in (2.3b). The latter problem is transformed equivalently to

$$Bx = \mu A^0 x, \quad B \in \boldsymbol{B} \tag{2.7}$$

where

$$\mu = 1/r. \tag{2.8}$$

We are interested in determining the rmm eigenvalue μ^* of the generalized eigenvalue problem (2.7). Two cases are possible.

Case A: $\mu^* > 0$. Consider the matrix B^*_μ that yields

$$B^*_\mu x^* = \mu^* A^0 x^*. \tag{2.9}$$

On account of (2.8)

$$r^* = 1/\mu^* \tag{2.10}$$

and from (2.9) and (2.10)

$$r^* B^*_\mu x^* = A^0 x^*$$

which is equivalent to

$$\det(A^c + r^* B^*_\mu) = 0. \tag{2.11}$$

Let the matrix B yielding the minimum r^* in (2.2) be denoted B^*_r, i.e.

$$\det(A^c + r^* B^*_r) = 0. \tag{2.12}$$

It is seen from (2.11) and (2.12) that

$$B^*_r := B^*_\mu. \tag{2.13}$$

Case B: $\mu^* < 0$. In this case, on account of (2.8) we can write

$$\breve{\rho} = 1/\mu^* \tag{2.14}$$

which leads to

$$\det(A^c + \breve{\rho} B^*_\mu) = 0$$

or

$$\det(A^c + |\breve{\rho}|(-B^*_\mu)) = 0. \tag{2.15}$$

Comparing (2.15) and (2.12), we see that now

$$B^*_r := -B^*_\mu \tag{2.16}$$

and from (2.14)

$$r^* = |\breve{\rho}| = 1/|\mu^*|. \tag{2.17}$$

This completes the proof. □

Thus, the problem of determining the regularity radius r^* of the interval matrix $\boldsymbol{A}$ is reduced to that of computing the rmm eigenvalue μ^* of the interval generalized eigenvalue problem (2.3).

In the sequel, we shall need the sign vector z^x of a real vector $\boldsymbol{x}$ or z^x of an interval vector $\boldsymbol{x}$ whose components are determined by the formulae

$$z^x_i = (\operatorname{sgn} \boldsymbol{x})_i := \begin{cases} 1, & \text{if} \quad x_i \geq 0 \\ -1, & \text{if} \quad x_i < 0 \end{cases} \tag{2.18a}$$

or

$$z^x_i = \operatorname{sgn} \boldsymbol{x}_i(\underline{\mu}) = \begin{cases} +1, & \text{if} \quad \boldsymbol{x}_i(\underline{\mu}) \geq 0 \\ -1, & \text{if} \quad \boldsymbol{x}_i(\underline{\mu}) < 0 \\ 0, & \text{if} \quad 0 \in \boldsymbol{x}_i(\underline{\mu}), \end{cases} \tag{2.18b}$$

respectively.

3 Determination of the rmm eigenvalue

In this section, a method for determining the rmm eigenvalue μ^* of (2.3) is suggested. As mentioned in the Introduction, it is based on using either outer interval solutions $\boldsymbol{x}$ and $\boldsymbol{y}$ or interval hull solutions $\boldsymbol{x}^*$ and $\boldsymbol{y}^*$ of corresponding linear interval systems. First the case of the use of $\boldsymbol{x}$ and $\boldsymbol{y}$ will be considered.

3.1 Outline of the method

The method is iterative and comprises the following stages.

Stage 1. Find a lower bound $|\underline{\mu}|$ on $|\mu^*|$.

Stage 2. Using $\underline{\mu}$, (2.7) and the corresponding right eigenvector, transform (2.7) into a $(n-1)\times(n-1)$ linear interval system

$$Cu = c, \quad C \in \boldsymbol{C}, \quad c \in \boldsymbol{c}. \tag{3.1a}$$

Find an outer interval solution $\boldsymbol{u}$ of (3.1a). In a similar way, using the left eigenvector of (2.7), form another $(n-1)\times(n-1)$ linear interval system.

$$Dv = d, \quad D \in \boldsymbol{D}, \quad d \in \boldsymbol{d} \tag{3.1b}$$

and find its outer interval solution $\boldsymbol{v}$. If $\boldsymbol{u} = (\boldsymbol{u}_1, ..., \boldsymbol{u}_m)$ and $\boldsymbol{v} = (\boldsymbol{v}_1, ..., \boldsymbol{v}_m)$, $m = n-1$, satisfy the following constant sign conditions

$$0 \notin \boldsymbol{u}_i, \quad \forall i, \quad i = 1, ..., m, \tag{3.2a}$$

$$0 \notin \boldsymbol{v}_i, \quad \forall i, \quad i = 1, ..., m, \tag{3.2b}$$

go to the next stage; otherwise modify the interval matrix $\boldsymbol{B}$in a certain manner to obtain new matrices $\boldsymbol{C}'$ and $\boldsymbol{D}'$ containing less interval entries. Let $\boldsymbol{C} = \boldsymbol{C}'$, $\boldsymbol{D} = \boldsymbol{D}'$ and start a cycle from (3.1a). This cycle (referred to as basic) will terminate for at most n iterates in one of the following two outcomes:

– outcome A: the constant sign condition (3.2) is fulfilled for the current $\boldsymbol{C}$ and $\boldsymbol{D}$, go to next stage;

– outcome B: no further improvement is possible (two consecutive iterates result in the same pattern of components $\boldsymbol{u}_i$ and $\boldsymbol{v}_i$ containing zero);

In the latter case, the method is not capable of determining the exact value of r^*. However, on exit from stage 2, it provides an upper bound $\overline{r}$ on r^* which is given by

$$\overline{r} = 1/|\underline{\mu}|. \tag{3.3}$$

Stage 3. Using the $(n-1)$- dimensional interval vectors $\boldsymbol{u}$ and $\boldsymbol{v}$ obtained at the last iteration of the basic cycle, construct the real $n\times n$ matrix B^*. Compute the real maximum magnitude eigenvalue μ^* of the generalized eigenvalue problem

$$B^*x = \mu A^0 x. \tag{3.4}$$

It will be shown later (Theorems 3.2 and 3.3) that the solution μ^* of (3.4) is, in fact, the solution of the initial generalized interval problem (2.7). On account of (2.17), the regularity radius r^* is then determined by

$$r^* = 1/|\mu^*|. \tag{3.5}$$

3.2 Determination of the lower bound $|\underline{\mu}|$

Consider the real eigenvalue problem

$$Bx = \mu A^0 x \tag{3.6}$$

for a fixed $B \in \boldsymbol{B}$. Let μ' denote the real maximum magnitude (rmm) eigenvalue of (3.6). We assume that μ' is a simple eigenvalue. Let x' denote the corresponding right

eigenvector (that is unique up to normalization). Let x'_s denote a largest magnitude component of x'. We divide x' by x'_s to obtain the normalized eigenvector x with

$$x_s = 1. \tag{3.7}$$

Let y denote the left eigenvector of (3.6) related to μ'. As is well known, this vector could be found by determining the eigenvector associated with the rmm eigenvalue μ'' of the following eigenvalue problem

$$B^T y = \mu (A^0)^T y \tag{3.8}$$

where the symbol T stands for transposition. Since $\mu'' = \mu'$ is known, y can be found in a more efficient way. We require that condition (3.7) be also valid for y, i.e. using the same index s as in (3.7) we let

$$y_s = 1. \tag{3.9}$$

Substituting μ' into (3.8) and taking into account (3.9), we get the system

$$(B - \mu' A^0)^T y = 0, \quad y_s = 1$$

or

$$D' y = 0, \quad y_s = 1, \tag{3.10a}$$

$$D' = (C')^T, \quad C' = B - \mu' A^0. \tag{3.10b}$$

System (3.10) is over-determined since it has n equations with $n-1$ unknowns y_i, $i \neq s$. Next, to make use of (3.9) explicitly we introduce the following notations. Let $D'_{s.}$ denote the sth row of D' while $D'_{.s}$ denotes the sth column of D'. We also need the vector $d' = -D'_{.s}$. Now we introduce the reduced size $(n-1) \times (n-1)$ matrix D obtained from D' by deleting both its sth row and sth column. We also introduce the reduced size vector d by deleting from d' its sth element. On account of these notations system (3.10) is written in the form

$$Dv = d \tag{3.11}$$

where v is a reduced size vector obtained from y by deleting its sth element.

The solution v of (3.11) is now used to determine the left eigenvector sought. Indeed, on account of (3.10)

$$y = (v_1, ..., v_{s-1}, 1, v_s, ..., v_{n-1}). \tag{3.12}$$

At this point, we introduce the scalar

$$\gamma = y^T A^0 x. \tag{3.13a}$$

If $\gamma < 0$, we let $y = -y$ to have

$$\gamma > 0. \tag{3.13b}$$

To express the dependence of μ, x and y on B, we shall use, whenever appropriate, the notation $\mu(B)$, $x(B)$ and $y(B)$. Multiplying (3.6) by y^T and using (3.13), we have

$$y^T(B) B x(B) = \gamma \mu(B) \tag{3.14a}$$

which yields in detailed form

$$\sum_{ij} b_{ij} y_i(B) x_j(B) = \gamma \mu(B), \qquad B \in \boldsymbol{B}. \tag{3.14b}$$

Let B^* denote a real matrix that provides the maximum magnitude in (3.14b); let the corresponding eigenvectors be x^* and y^*, i.e. $x^* = x(B^*)$ and $y^* = y(B^*)$. We have the following result.

Theorem 3.1 *The real maximum magnitude eigenvalue μ^* of (2.7) is obtained if the elements b^*_{ij} of B^* are determined as follows:*

if $\mu^* > 0$,

$$b^*_{ij} = \begin{cases} \bar{b}_{ij} = \ \ r_{ij}, & \text{if} \quad y^*_i x^*_j \geq 0 \\ \underline{b}_{ij} = -r_{ij}, & \text{if} \quad x^*_i y^*_j \leq 0; \end{cases} \tag{3.15a}$$

if $\mu^* < 0$,

$$b^*_{ij} = \begin{cases} \underline{b}_{ij} = -r_{ij}, & \text{if} \quad y^*_i x^*_j \geq 0 \\ \bar{b}_{ij} = \ \ r_{ij}, & \text{if} \quad y^*_i x^*_j \leq 0. \end{cases} \tag{3.15b}$$

Proof: Assume that $\mu^* > 0$. On account of (3.15a) and (3.14b)

$$\sum_{ij} b^*_{ij} y_i(B^*) x_j(B^*) = \sum_{ij} b^*_{ij} y^*_i x^*_j = \gamma^* \mu^*. \tag{3.16}$$

But

$$\sum_{ij} b_{ij} y^*_i x^*_j < \gamma^* \mu^*, \tag{3.17a}$$

if

$$b_{ij} \neq b^*_{ij}. \tag{3.17b}$$

On the other hand, according to [15] each element b^*_{ij} of B^* takes on a boundary value, i.e.

$$\boldsymbol{b}^*_{ij} = \begin{cases} \text{either} \quad \bar{b}_{ij} = \ \ r_{ij} \\ \text{or} \quad \ \ \underline{b}_{ij} = -r_{ij}. \end{cases} \tag{3.18}$$

Thus, on account of (3.16) to (3.18) the maximum in the left-hand side of (17a) is attained if $B = B^*$ with elements defined by (3.15a).

Formula (3.15b) is proved in a similar way. □

Based on Theorem 3.1, we suggest the following procedure for determining a lower bound $|\underline{\mu}|$ on the rmm eigenvalue $|\underline{\mu}^*|$.

Procedure 3.1 *(for finding $|\underline{\mu}|$). The procedure involves the following steps.*

0. *(initialization). Set $n_i = 0$ (n_i is the iteration number) and $\mu^0 = 0$. Introduce the real matrix B formed by the upper endpoints of the interval matrix $\boldsymbol{B}$, i.e. $B = \bar{B} = R$.*
1. *Let $n_i = n_i + 1$. Find the rmm eigenvalue μ' of*

$$Bx = \mu A^0 x$$

and the corresponding right eigenvector x', using some generalized eigenvalue problem solver. Traditionally, an eigenvector is normalized so that

$$|x| = \left(\sum_{i=1}^{v} x_i^2 \right)^{1/2} = 1.$$

For our purposes we use the rule (3.7) to obtain the index s and the normalized eigenvector x. We also introduce the check: if $\mu' < 0$, then we let $B = -B$ and $\mu' = -\mu'$ ensuring that

$$\mu' > 0. \tag{3.19}$$

2. *Using the index s found in step 1, form system (3.11). Using its solution v and (3.12), find the left eigenvector y associated with μ'. Compute γ by (3.13a) and, if necessary, change the sign of y to ensure $\gamma > 0$.*

3. *Find the sign vector z^x of the eigenvector x using (2.18a). In the same way, find the sign vector z^y of the eigenvector y. Form the sign vector $z = (z^x, z^y)$. Introduce the real matrix B^z with components*

$$b^*_{ij} = \begin{cases} \underline{b}_{ij}, & \text{if} \quad z^y_i z^x_j = -1, \\ \bar{b}_{ij}, & \text{if} \quad z^y_i z^x_j = \ \ 1. \end{cases} \tag{3.20}$$

4. *Let $\mu^0 = \mu'$, $B = B^z$ and start a cycle (called an inner cycle) from Step 1 until the sign vector z remains unchanged (the current z is equal to z' of the previous iteration).*

5. *A lower bound $\underline{\mu}$ equal to μ' of the last iteration has been found.*

It can be easily seen that the choice of B^z by (3.19), (3.20) ensures that $\mu' > \mu^0$ before stationarity is reached. Thus, the procedure is guaranteed to find $\underline{\mu}$ in a finite number n_i of iterations (typically for $n_i < n$).

3.3 Forming the linear interval systems

In this subsection, we show how the linear interval systems (3.1) are formed.

First, the system (3.1a) will be considered. It is obtained from (2.7) in the following way. We observe that (2.7) defines μ and x as implicit functions of B, i.e. $\mu = \mu(B)$ and $x = x(B)$. We now fix μ to the value of the lower bound $\underline{\mu}$ (found by Procedure 3.1), i.e. we replace μ by $\underline{\mu}$ in (2.7). Thus, x becomes a function of both $\underline{\mu}$ and B, i.e. we have

$$x = x(\underline{\mu}, B), \quad B \in \boldsymbol{B}. \tag{3.21}$$

Next, we impose the normalization condition (3.7) to be valid for x defined by (3.21), i.e. require that

$$x_s = x(\underline{\mu}, B) = 1, \quad B \in \boldsymbol{B}. \tag{3.22}$$

Taking into account (3.22) and the equality $\mu = \underline{\mu}$, the interval eigenvalue problem (2.7) is transformed into the linear interval system (3.1a) in much the same way as the real eigenvalue (3.8) has been modified into the real linear system (3.11). Indeed, now we have

$$(B - \underline{\mu} A^0)x = 0, \quad B \in \boldsymbol{B}, \quad x_s = 1 \tag{3.23}$$

or equivalently

$$C'x = 0, \quad C' \in \boldsymbol{C}, \quad x_s = 1 \tag{3.24a}$$

$$C' = B - \underline{\mu} A^0, \quad \boldsymbol{C'} = \boldsymbol{B} - \underline{\mu} A^0. \tag{3.24b}$$

System (3.24) can be written symbolically in the form

$$\boldsymbol{C'}x = 0, \quad x_s = 1. \tag{3.25}$$

Now we introduce the following notations: $\boldsymbol{C'}_{s.}$ for the sth row of $\boldsymbol{C'}$ and $\boldsymbol{C'}_{.s}$ for the sth column of $\boldsymbol{C'}$ as well as $\boldsymbol{c'} = -\boldsymbol{C'}_{.s}$; $\boldsymbol{C}$ for the reduced size interval matrix obtained from $\boldsymbol{C'}$ by deleting the corresponding sth row and sth column of $\boldsymbol{C'}$ as well as $\boldsymbol{c}$ for the reduced $\boldsymbol{c'}$ resulting from $\boldsymbol{c'}$ after deleting its sth element. In a similar manner, we introduced the notations $\boldsymbol{B}_{s.}$, $\boldsymbol{C}_{.s}$, $A^0_{s.}$ and $A^0_{.s}$ and form: (i) the reduced-size

$((n-1)\times(n-1)$ matrices $\boldsymbol{B}_r$ and A^0_r and (ii) the reduced-length vectors $\boldsymbol{B}_{rs}$ and A^0_{rs} (obtained from $\boldsymbol{B}_{.s}$ and $A^0_{.s}$, respectively). Thus, (3.25) can be written in the form

$$\boldsymbol{C}u = \boldsymbol{c}, \tag{3.26a}$$

$$\boldsymbol{C} = \boldsymbol{B}_r - \bar{\mu}A^0_r, \quad \boldsymbol{c} = -\boldsymbol{B}_{rs} + \bar{\mu}A^0_{rs}, \tag{3.26b}$$

$$\boldsymbol{C}'_{s.}x = 0. \tag{3.26c}$$

The system (3.26a), (3.26b) obtained is the short (symbolic) form of the interval linear system (3.1a) sought. The interval linear system (3.1b) is formed in an analogous manner as system (3.26a). The only difference is that now the interval matrix $\boldsymbol{D}$ is obtained on the basis of $(\boldsymbol{C}')^T$. Indeed, since we are now interested in the left eigenvector solution of (2.7) we have to consider the problem

$$B^T y = \mu(A^0)^T y, \quad B \in \boldsymbol{B}. \tag{3.27}$$

In this case, similarly to (3.22) we have

$$y = y(\underline{\mu}, B), \quad B \in \boldsymbol{B}$$

and the normalization rule (3.23) becomes

$$y_s(\underline{\mu}, B) = \alpha, \quad B \in \boldsymbol{B} \tag{3.28}$$

where $\alpha_s = 1$, if $\gamma > 0$ or $\alpha_s = -1$, if $\gamma < 0$. On account of (3.27) and (3.28), now we have

$$(B - \underline{\mu}A^0)^T y = 0, \quad B \in \boldsymbol{B}, \quad y_s = 1$$

which, in its turn, leads to

$$\boldsymbol{D}v = \boldsymbol{d}. \tag{3.29}$$

It is easily seen that

$$\boldsymbol{D} = \boldsymbol{C}^T. \tag{3.29a}$$

while $\boldsymbol{d}$ is the reduced form of

$$\boldsymbol{d}' = -(\boldsymbol{C}')^T_{.s} \tag{3.29b}$$

(with $\boldsymbol{C}')^T_{.s}$ being the sth column of $\boldsymbol{C}')^T$).

Let $\boldsymbol{u}$ denote an outer solution of (3.26a). On account of the relation between u and x (by analogy with (3.12))

$$\boldsymbol{x} = (\boldsymbol{u}_1, ..., \boldsymbol{u}_{s-1}, 1, \boldsymbol{u}_s, ..., \boldsymbol{u}_{n-1}). \tag{3.30}$$

In a similar way, if $\boldsymbol{v}$ denotes the outer solution of (3.29), then

$$\boldsymbol{y} = (\boldsymbol{v}_1, ..., \boldsymbol{v}_{s-1}, 1, \boldsymbol{v}_s, ..., \boldsymbol{v}_{n-1}). \tag{3.31}$$

Remark 3.1. To simplify the notation, we have introduced the outer solutions $\boldsymbol{u}$ and $\boldsymbol{x}$. It should, however, be borne in mind that in view of (3.22), (3.24b) and (3.26a) a more precise notation for $\boldsymbol{u}$ would be $\boldsymbol{u}(\underline{\mu})$. From (3.30), we should also use $\boldsymbol{x}(\underline{\mu})$ (rather than the short notation $\boldsymbol{x}$) to underline the dependence of $\boldsymbol{x}$ on $\underline{\mu}$. For similar reasons, $\boldsymbol{v}(\underline{\mu})$ and $\boldsymbol{y}(\underline{\mu})$ are to replace $\boldsymbol{v}$ and $\boldsymbol{y}$ when necessary to stress the dependence of these interval vectors on $\underline{\mu}$. The same remark is valid for the range solutions $\boldsymbol{u}^*(\underline{\mu})$ and $\boldsymbol{v}^*(\underline{\mu})$ of (3.26a) and (3.29), respectively, written in short as $\boldsymbol{u}^*$ and $\boldsymbol{v}^*$, as well as the associated ranges $\boldsymbol{x}^*(\underline{\mu})$ and $\boldsymbol{y}^*(\underline{\mu})$ written in short as $\boldsymbol{x}^*$ and $\boldsymbol{y}^*$.

3.4 Main results

We first consider the problem how to determine the rmm eigenvalue μ^* (formulae (3.15) are inapplicable since the optimal eigenvectors x^* and y^* are unknown). A solution is suggested which is based on the use of the outer interval bounds $\boldsymbol{x}(\underline{\mu})$ and $\boldsymbol{y}(\underline{\mu})$ where $\underline{\mu}$ is the lower bound on μ^* found by Procedure 3.1. We now make the following assumption.

Assumption 3.1. The eigenvalues $\underline{\mu}$ and μ^* are simple. It is also assumed that systems (3.26a) and (3.29) have bounded interval solutions.

We first prove the following auxiliary result.

Lemma 3.1 *If Assumption 3.1 is valid and additionally*

$$\underline{\mu}\mu^* > 0, \tag{3.32}$$

then

$$\boldsymbol{x}(\mu^*) \in \boldsymbol{x}(\underline{\mu}), \quad \boldsymbol{y}(\mu^*) \in \boldsymbol{y}(\underline{\mu}). \tag{3.33}$$

Proof: First, the former inclusion will be proved. Consider the interval systems

$$Bx = \underline{\mu}A^0x, \quad B \in \boldsymbol{B}, \quad Bx = \mu^*A^0x, \quad B \in \boldsymbol{B}$$

related to the outer solutions $\boldsymbol{x}(\underline{\mu})$ and $\boldsymbol{x}(\mu^*)$, respectively. On account of (3.30), it suffices to prove that

$$\boldsymbol{u}(\mu^*) \subseteq \boldsymbol{u}(\underline{\mu}). \tag{3.34}$$

In view of (3.26a) and (3.26b), $\boldsymbol{u}(\underline{\mu})$ is the outer solution of

$$(\boldsymbol{B}_r - \underline{\mu}A^0_r)u = -\boldsymbol{B}_{rs} + \underline{\mu}A^0_{rs} \tag{3.35a}$$

while $\boldsymbol{u}(\mu^*)$ is the outer solution of

$$(\boldsymbol{B}_r - \mu^*A^0_r)u = -\boldsymbol{B}_{rs} + \mu^*A^0_{rs}. \tag{3.35b}$$

On account of (3.32) $\mu^* = \alpha\underline{\mu}$ with $\alpha > 1$ ($|\underline{\mu}| < |\mu^*|$) so we can transform (3.35b) into the equivalent form

$$[(1/\alpha)\boldsymbol{B}_r - \underline{\mu}A^0_r]u = -(1/\alpha)\boldsymbol{B}_{rs} + \underline{\mu}A^0_{rs} \tag{3.35c}$$

Now the inclusion (3.34) follows from the comparison of (3.35a) and (3.35c), on the one hand, and the inclusion monotonicity property of interval operations, on the other hand, since $(1/\alpha)\boldsymbol{B}_r \subseteq \boldsymbol{B}_r$ and $(1/\alpha)\boldsymbol{B}_{rs} \subseteq \boldsymbol{B}_{rs}$.

The latter inclusion in (3.33) is proved in a similar way. □

To simplify the presentation of the method for determining μ^*, we first make (temporarily) the following assumption.

Assumption 3.2. No component $\boldsymbol{x}_i(\underline{\mu})$ of $\boldsymbol{x}(\underline{\mu})$ and $\boldsymbol{y}_i(\underline{\mu})$ of $\boldsymbol{y}(\underline{\mu})$ contain 0, i.e.

$$0 \notin \boldsymbol{x}_i(\underline{\mu}), \quad 0 \notin \boldsymbol{y}_i(\underline{\mu}), \quad i \in I \tag{3.36}$$

where $I = \{1, ...n\}$.

Using Lemma 3.1, we have the following result.

Theorem 3.2 *If Assumptions 3.1 and 3.2 are valid, then μ^* is given by the rmm eigenvalue of the following generalized eigenvalue problem*

$$B^* x = \mu A^0 x \tag{3.37a}$$

where B^* is a real matrix whose elements b_{ij}^* are determined as follows

$$b_{ij}^* = \begin{cases} r_{ij}, & \text{if} \quad \boldsymbol{y}_i(\underline{\mu})\boldsymbol{x}_j(\underline{\mu}) > 0, \\ -r_{ij}, & \text{if} \quad \boldsymbol{y}_i(\underline{\mu})\boldsymbol{x}_j(\underline{\mu}) < 0. \end{cases} \tag{3.37b}$$

Proof: From Theorem 3.1,

$$b_{ij}^* = \begin{cases} \bar{b}_{ij} = r_{ij}, & \text{if} \quad y_i^* x_j^* \geq 0, \\ \underline{b}_{ij} = -r_{ij}, & \text{if} \quad y_i^* x_j^* < 0. \end{cases} \tag{3.38}$$

for $\mu^* > 0$. According to Lemma 3.1 the sign of the product of the unknown values y_i^* and x_j^* is the same as the sign of the product of the known $\boldsymbol{x}_j$ and $\boldsymbol{y}_i$. The validity of (3.37) follows directly from comparison of (3.37) and (3.38). □

Now we show the validity of the present method for determining μ^* (outlined in Section 3.1) when Assumptions 3.1 and 3.2 are fulfilled. After a lower bound $\underline{\mu}$ on μ^* is found in Stage 1 of the method (using Procedure 3.1), we form systems (3.1) and find the outer interval solutions $\boldsymbol{u}$ and $\boldsymbol{v}$. On account of (3.30) to (3.36) the validity of Assumption 3.2 entails the validity of conditions (3.36). So we can construct the real matrix B^* from Stage 3. The elements of this matrix are determined as in (3.37b). Finally, we solve (3.37a) to find the rmm eigenvalue μ^*.

Now it will be shown that the present method for computing μ^* remains applicable if the Assumption 3.2 is replaced with the following less stringent condition.

Assumption 3.3. Let I_1 and J_1 be two subsets of the index set $I = \{1, ..., n\}$ of which at least one is proper and such that

$$0 \notin \boldsymbol{z}_{ij} = \boldsymbol{y}_i \boldsymbol{x}_j, \quad \text{if } i \in I_1,\ j \in J_1, \tag{3.39a}$$

$$0 \notin \boldsymbol{z}_{ij} = \boldsymbol{y}_i \boldsymbol{x}_j, \quad \text{if } i \notin I_1.\ j \notin J_1, \tag{3.39b}$$

It is seen that now the sign constancy condition is required only for part of the intervals $\boldsymbol{z}_{ij}$ since I_1 and J_1 are smaller than I. Hence, the partial sign constancy condition (3.39a) is easier to satisfy than the complete sign constancy condition (3.36). Based on Lemma 3.1 and using the same arguments as in Theorem 3.2, we have the following results.

Theorem 3.3 *If Assumptions 3.1 and 3.3 are valid, those components b_{ij} of B^* for which $i \in I_1$ and $j \in J_1$ are determined by (3.37b); the remaining elements of B^* are unknown.*

The basic cycle of the present method (mentioned in Stage 2 from Section 3.1) makes use of Theorems 3.2 and 3.3.

Remark 3.2. The above results remain valid if throughout the outer interval solutions $\boldsymbol{u}$ and $\boldsymbol{v}$ of the linear interval systems (3.26a) and (3.29) are replaced by the corresponding interval hull solutions $\boldsymbol{u}^*$ and $\boldsymbol{v}^*$.

4 Algorithms

In this section, an algorithm for assessing the regularity radius r^* is suggested which is based on Theorems 3.2 and 3.3. In view of Remark 3.2, it will be presented in two versions:

(A) using the outer solutions or

(B) using the hull solutions.

In either case, if Assumption 3.2 is satisfied, the true r^* can be determined. Using Theorem 3.3, this can be done even in the case where Assumption 3.2 is violated. Indeed, according to that theorem a modified interval matrix $\boldsymbol{B}'$ can be generated with elements

$$\boldsymbol{b}'_{ij} = \begin{cases} \bar{b}_{ij} = \quad r_{ij}, & \text{if} \quad \boldsymbol{z}_{ij} > 0, \\ \underline{b}_{ij} = -r_{ij}, & \text{if} \quad \boldsymbol{z}_{ij} < 0, \\ \boldsymbol{b}_{ij} & \text{otherwise.} \end{cases} \tag{4.1a}$$

It is seen that the new matrix $\boldsymbol{B}'$ has a smaller number of interval entries as compared to the original matrix $\boldsymbol{B}$. Using $\boldsymbol{B}'$, we form a modified matrix

$$\boldsymbol{A}' = A^c + \boldsymbol{B}', \tag{4.1b}$$

let $\boldsymbol{A} = \boldsymbol{A}'$ and update the center and radius of $\boldsymbol{A}$. New matrices $\boldsymbol{C}'$ and $\boldsymbol{D}'$ containing less interval entries are thus set up. We let $\boldsymbol{C} = \boldsymbol{C}'$ and $\boldsymbol{D} = \boldsymbol{D}'$, form the corresponding vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ and check the constant sign condition (3.36). If it is satisfied, the rmm eigenvalue μ^* is computed from (3.37). Otherwise, if the current index sets I_1 and J_1 (associated with $\boldsymbol{B}'$) are not empty, a modified matrix $\boldsymbol{A}'$ is constructed by (4.1b), it is renamed $\boldsymbol{A}$ and a new iteration is initialized. This iterative scheme can terminate in two outcomes:

(i) the current modified matrix $\boldsymbol{B}'$ contains no interval entries: the real maximum eigenvalue sought has been determined;

(ii) no further improvement is possible and the current matrix $\boldsymbol{B}'$ still contains at least one interval entry: in this case the method is not capable of finding μ^* exactly.

Typically, the basic cycle terminates in outcome (i). If, however, it ends up in outcome (ii), most often only a small number of elements remain intervals in the last matrix $\boldsymbol{B}'$. Therefore, a tight lower bound $\underline{\mu}$ can be found applying Procedure 3.1 to the corresponding matrix $\boldsymbol{A}'$.

The first version of the present algorithm referred to as Algorithm 4.1 is based on the above ideas and the use of the outer solutions to (3.26a) and (3.29).

It is easily seen that Algorithm 4.1 has polynomial complexity. Indeed, the transition from the current matrix $\boldsymbol{B}$ to the new matrix $\boldsymbol{B}'$ may result (in the worst scenario) in the reduction of just one interval element $\boldsymbol{b}_{ij}$ to r_{ij} or $-r_{ij}$. Thus, the total number k of iterations of the basic cycle needed to reach outcome (i) is at most n^2; obviously $k < n^2$ if the outcome is (ii). On the other hand, each iteration of the basic cycle requires a polynomial amount of operations for computing $\underline{\mu}$ by Procedure 3.1 and setting up and solving the linear interval systems (3.26a) and (3.29), which leads to polynomial complexity of the algorithm considered.

There exists an opportunity to compute the exact value of μ^* even if the basic cycle terminates in outcome (ii). Let m_1 and m_2 denote the number of indices in the set I_2 and J_2, respectively. Let $m = m_1 + m_2$. If m is not greater than a boundary value $\bar{m}$ (say $\bar{m} = 12$), we can apply the following combinatorial approach to determine μ^*.

Let z_i be the components of the $2n$-dimensional sign vector $z = (z^x, z^y)$. The set of z_i whose indices correspond to the "optimal" components z_i^* (defined by $\boldsymbol{x}^*$, $\boldsymbol{y}^*$ and

(2.18b)) will be denoted t^*; the remaining elements of z will be denoted w. Thus, the vector z can be represented in the form

$$z = (t^*; w). \tag{4.2}$$

On exit from outcome (ii) the elements of w are all zeros. However, being a component of the sign vector, each component w_i of w can take the value of either $+1$ or -1. Let each combination of these $+1$, -1 be denoted $w^{(v)}$. Obviously, the number of all such combinations is $n_m = 2^m$. Each vector $w^{(v)}$ gives rise through (4.2) to a corresponding sign vector

$$z^{(v)} = (t^*; w^{(v)}). \tag{4.3}$$

so the number of all possible vectors $z^{(v)}$ is also n_m. On account of (3.21) let B_v be the matrix associated with the respective $z^{(v)}$. Further, let $\mu(B_v)$ denote the real maximum magnitude eigenvalue of

$$B_v x = \mu A^0 x. \tag{4.4}$$

Obviously

$$\mu^* = \max(\mu(B_v), v = 1, ..., n_m). \tag{4.5}$$

The above procedure for attaining μ^* will be referred to as Procedure 4.1. Since m is bounded by $\bar{m}$, the numerical complexity of Procedure 4.1 and, hence, of the method for computing remains polynomial.

We now present Algorithm 4.1 in detail.

Algorithm 4.1 *(use of outer solutions). The steps of this algorithm are executed successively unless there is a branching.*

0. *(initialization). Set* $\boldsymbol{B} = [-R, R]$ *and choose a value for* $\bar{m}$ *(e.g.,* $\bar{m} = 12$ *). Set* $k = 0$ *(*k *will denote the current number of the basic cycle iteration). Introduce the* $2n$ *- dimensional zero vector* z^0.

1. *Let* $k = k + 1$. *Apply Procedure 3.1 to compute the lower bound* $\underline{\mu}$ *and the corresponding eigenvectors* x *and* y.

2. *Use the eigenvector* x *to fix the index* s *using the normalization rule (3.7). Using* $\underline{\mu}$ *and* s, *set up system (3.26a). Apply the method of [14] (or [3] if the matrix* $\boldsymbol{C}$ *is structured) trying to determine the outer solution* $\boldsymbol{u}$. *If this is possible, form the outer solution* $\boldsymbol{x}$ *using (3.30) and determine by (2.18b) its sign vector* z^x. *Otherwise, go to Step 10.*

3. *Using* $\underline{\mu}$ *and* s, *set up system (3.29). Apply the method from [14] (or [3]) to determine the outer solution* $\boldsymbol{v}$. *If this is possible, form the outer solution* $\boldsymbol{y}$ *using (3.31) and determine by (2.18b) its sign vector* z^y. *Otherwise, go to Step 10.*

4. *Form the* $2n$*-dimensional sign vector* $z = (z^x; z^y)$. *If* z *contains no zero components, go to Step 5; otherwise go to Step 6.*

5. *Using the corresponding outer solutions* $\boldsymbol{x}$ *and* $\boldsymbol{y}$ *found in Steps 2 and 3, form by (3.37b) the real matrix* B^* *and solve the eigenvalue problem (3.37a) to find* μ^*. *Go to Step 9.*

6. *If* $z = z^0$, *go to Step 8.*

7. *Using z, determine the modified matrix $\boldsymbol{B}'$ with elements defined by (4.1a). Form by (4.1b) the modified matrix $\boldsymbol{A}'$, renamed it as $\boldsymbol{A}$ and update its center and radius according to (1.2b). Set $z^0 = z$ and go to Step 1 (to start a new iteration of the basic cycle).*

8. *Let m denote the number of zero components of the current sign vector z. If $m > \bar{m}$, go to Step 10; else apply Procedure 4.1.*

9. *Termination 1. The exact (within rounding errors) value of μ^* has been found. Finally, the regularity radius r^* of the interval matrix $\boldsymbol{A}$ is determined as*

$$r^* = 1/\mu^*.$$

10. *Termination 2. A lower bound $\underline{\mu}$ on μ^* has been found and, hence, an upper bound*

$$\bar{r} = 1/\underline{\mu}$$

on r^* has been determined.

Remark 4.1. If the width of the interval matrix $\boldsymbol{A}$ is relatively small, the algorithm terminates in Step 9 providing the exact regularity radius r^*. For larger widths, however, Algorithm 4.1 fails to compute r^* since $\boldsymbol{C}$ and $\boldsymbol{D}$ are not strongly regular (the matrix M from the method [14] used in Steps 2 and 3 turns out to be not non-negative). Also if $m > \bar{m}$, only an upper bound $\bar{r}$ on r^* is attained.

To ensure polynomial complexity of Algorithm 4.1, the interval matrices $\boldsymbol{C}$ and $\boldsymbol{D}$ in (3.26a) and (3.29), respectively, should be strongly regular. If this is not the case, the problem of determining the regularity radius r^* of the interval matrix $\boldsymbol{A}$ could still be solved in polynomial time if use is made of the interval hull solutions $\boldsymbol{u}^*$ and $\boldsymbol{v}^*$ of systems (3.26a) and (3.29) provided the solution sets X and Y of these systems intersect a "small" number of orthants. More precisely, the numbers p_X and p_Y of nonempty intersections of X and Y with the orthants should be polynomially bounded [4]. In this case, the method of [4] can determine the hull solutions in polynomial time (or establishes the singularity of the interval matrix involved). Thus, the second version of the present algorithm denoted Algorithm 4.2 differs from Algorithm 4.1 only in that the outer solutions $\boldsymbol{u}$ and $\boldsymbol{v}$ in Step 2 and Step 3 are replaced by the interval hull solutions $\boldsymbol{u}^*$ and $\boldsymbol{v}^*$, which are computed by the method of [4].

Remark 4.2. It is easily seen that the use of the interval hull solutions $\boldsymbol{u}^*$ and $\boldsymbol{v}^*$ rather than the outer solutions $\boldsymbol{u}$ and $\boldsymbol{v}$ leads to easier satisfaction of the constant sign conditions (3.36). Thus, it is expected that Algorithm 4.2 will determine the regularity radius r^* in cases where Algorithm 4.1 manages to compute only an upper bound $\bar{r}$. At the same time, determining $\boldsymbol{u}^*$ and $\boldsymbol{v}^*$ is much more time consuming than determining $\boldsymbol{u}$ and $\boldsymbol{v}$. These observations will be confirmed by the numerical evidence reported in the next section.

5 Numerical examples

We illustrate the applicability of the method suggested by way of several examples. The linear interval systems (3.26a) and (3.29) have been solved using the method from [14] or [4] depending on whether Algorithm 4.1 or Algorithm 4.2 is applied to the example considered. Both algorithms have been implemented in a simplified manner without resorting to Procedure 4.1. All computations have been done on a 1.7 GHz PC computer using the MATLAB environment and toolbox INTLAB [19], version 5.5. To

simplify the presentation, the numerical results will be reported only to four decimal places. The first four examples illustrate the application of Algorithm 4.1.

Example 1. In this example, the interval matrix $\boldsymbol{A}$ whose regularity radius we are interested in is constructed in the following manner. First (as in [18]), two 7×7 matrices are generated in a random way (each entry is a random integer from the interval $[-9, 9]$:

$$S = \begin{pmatrix} 4 & 1 & 2 & -4 & -4 & 5 & -9 \\ 5 & 6 & 4 & -9 & -2 & 7 & 6 \\ -2 & 9 & 7 & -8 & 9 & -3 & 0 \\ 5 & -8 & 2 & -1 & -4 & 2 & 3 \\ -4 & -4 & 6 & 6 & 2 & 9 & 8 \\ -5 & 4 & 9 & -5 & 1 & -7 & 9 \\ 3 & -9 & 1 & -8 & -8 & 6 & -4 \end{pmatrix}, \tag{5.1}$$

$$\Delta = \begin{pmatrix} 0 & 2 & 3 & 3 & 9 & 9 & 4 \\ 2 & 2 & 1 & 1 & 6 & 4 & 4 \\ 3 & 1 & 7 & 3 & 6 & 2 & 1 \\ 3 & 1 & 3 & 5 & 0 & 4 & 9 \\ 9 & 6 & 6 & 0 & 2 & 7 & 3 \\ 9 & 4 & 2 & 4 & 7 & 2 & 9 \\ 4 & 4 & 1 & 9 & 3 & 9 & 6 \end{pmatrix}, \tag{5.2}$$

the second matrix being symmetric. Since S is nonsingular, the following matrix is introduced

$$\underline{B} = -S^T S. \tag{5.3}$$

Now a symmetric interval matrix of the form

$$\boldsymbol{B} = [\underline{B}, \underline{B} + \varepsilon.\Delta] \tag{5.4}$$

is constructed where

$$\varepsilon = 0.2. \tag{5.4a}$$

At this point, the matrix in (5.4) is transformed into a non-symmetric matrix by letting

$$\boldsymbol{B}_{ij} = 0, \quad i, j \geq \alpha, \tag{5.5a}$$

$$\alpha = 6, \tag{5.5b}$$

i.e. by nullifying the elements $\boldsymbol{B}_{ij}$ on and above the diagonal $n - \alpha$ of $\boldsymbol{B}$. Finally, $\boldsymbol{A}$ is obtained by subtracting a diagonal matrix from the new $\boldsymbol{B}$ as follows

$$\boldsymbol{A} = \boldsymbol{B} - 100E \tag{5.6}$$

(where E is the identity matrix).

For this example, the problem considered is solved successfully by the method suggested since Algorithm 4.1 terminates in Step 9 providing the exact value of r^*. The basic cycle takes 2 iterations to converge, i.e. $k = 2$. At the first iteration ($k = 1$), the outer solutions $\boldsymbol{x}$ and $\boldsymbol{y}$ are

$$\begin{aligned} \boldsymbol{x} =& ([0.2907, 0.5134], [-0.1823, -0.0550], [-0.4437, -0.2728], \\ & [-0.1639, -0.0627], [-0.2752, -0.1198], [-0.9161, -0.4169], [1.0000, 1.0000]), \end{aligned} \tag{5.7a}$$

$$\boldsymbol{y} = ([0.1352, 0.3718], [-0.0955, 0.0476], [-0.5486, -0.3681], \\ [-0.1790, -0.00870], [-0.1867, -0.0249], [-0.4858, 0.1191], [1.0000, 1.0000]). \tag{5.7b}$$

and it is seen that now

$$0 \in \boldsymbol{y}_2, \quad 0 \in \boldsymbol{y}_6. \tag{5.8}$$

Thus,

$$I_1 = \{1, ..., n\}, \quad I_2 = \emptyset, \quad J_1 = \{1, 3, 4, 5, 7\}, \quad J_2 = \{2, 6\} \tag{5.9}$$

and since $J_2 \neq \emptyset$ we are led to start a second iteration of the basic cycle. On account of (5.9) and (4.1a), the modified interval matrix $\boldsymbol{B}'$ will have proper (non-degenerate) interval entries only in the positions of the second and sixth rows, which are equal to the respective entries of the initial matrix $\boldsymbol{B}$. The remaining elements whose indices $i \in I_1$ and $j \in J_1$ are real numbers and, according to Theorem 3.3, their values are determined by (4.1a). Thus, we have formed the modified matrix $\boldsymbol{B}'$. At this stage, the corresponding matrix $\boldsymbol{A}'$ is set up by (4.1b), $\boldsymbol{A}'$ is renamed $\boldsymbol{A}$ and the second iteration is initialized from Step 1 of Algorithm 4.1. At the end of the iteration, the relevant data are

$$\boldsymbol{x} = ([0.3884, 0.4636], [-0.1588, -0.0870], [-0.3804, -0.3283], \\ [-0.1282, -0.1141], [-0.2418, -0.1987], [-0.8302, -0.5843], [1.0000, 1.0000]), \tag{5.10a}$$

$$\boldsymbol{y} = ([0.2952, 0.3312], [-0.0775, -0.0615], [-0.4340, -0.4106], \\ [-0.1448, -0.1305], [-0.1571, -0.1343], [-0.4135, -0.3463], [1.0000, 1.0000]). \tag{5.10b}$$

It is seen that condition (3.36) is now fulfilled so the sign vector $z = (z^x; z^y)$ formed in Step 4 contains no zero. Thus, the algorithm terminates in Step 9 with

$$\mu^* = \underline{\mu} = 0.3100. \tag{5.11}$$

Finally,

$$r^* = 1/\mu^* = 3.2254. \tag{5.12}$$

The time needed by the algorithm to determine r^* is $T = 0.091$ sec.

The relevant data on the example considered are given in Table 1. The input data

Input data			Output data				
n	α	ε	k	N	$\lvert\mu^*\rvert$	r^*	T[s]
7	6	0.2	2	6	0.3100	3.2254	0.091

Table 1: For the first matrix in Example 1.

are the values of the parameters α and ε defining the matrix studied. The output data are: k (number of the basic cycle iterations), N (total number of inner iterations needed by Procedure 3.1), the values of $|\mu^*|$ and r^* as well as T (execution time in seconds).

A second matrix $\boldsymbol{A}$ has been constructed in the same way as before. The only difference is that now $\alpha = 5$. The relevant results associated with the input data chosen are given in Table 2.

Input data			Output data				
n	α	ε	k	N	$\lvert\mu^*\rvert$	r^*	T[s]
7	5	1	5	17	0.7519	1.3299	0. 227

Table 2: For the second matrix in Example 1.

Example 2. In this example, the interval matrix $\boldsymbol{A}$ is constructed in the following way. Initially, its center matrix A_c is symmetric

$$(A^c)_{ij} = (\gamma_0 + i\gamma_1)\delta_{ij} + (2/(n+1))^{1/2}\sin(ij\pi/(n+1)) \tag{5.13a}$$

(δ_{ij} is the Kronecker symbol) where γ_0, γ_1 are constants, n being the size of the matrix. To make A^c asymmetric, the elements $(A^c)_{ij}$ on and above the diagonal $n-\alpha$ of A^c will be fixed to a constant β. For this example α was chosen to be $\alpha = 2$ so three elements of A^c are modified as follows

$$A^c(1,n) = \beta, \quad A^c(1,n-1) = \beta, \quad A^c(2,n) = \beta. \tag{5.13b}$$

The radius matrix of $\boldsymbol{A}$ is defined by

$$R = \varepsilon|A^c|. \tag{5.13c}$$

Several interval matrices $\boldsymbol{A}$ will be constructed using different values for the constants involved.

We first consider a matrix $\boldsymbol{A}$ for which the defining (input) data are

$$n = 10, \quad \varepsilon = 0.04, \quad \beta = 80, \quad \gamma_0 = 0.4, \quad \gamma_1 = 0.8. \tag{5.13d}$$

The application of Algorithm 4.1 of the present method results in the successful determination of the regularity radius r^* of the matrix $\boldsymbol{A}$ defined by (5.13). The radius r^* is obtained for four iterations of the basic cycle ($k = 4$) and a total of twelve iterations of the inner iterations ($N = 12$) (see Table 3).

Input data					Output data				
n	ε	β	γ_0	γ_1	k	N	$\lvert\mu^*\rvert$	r^*	T[s]
10	0.04	80	0.4	0.8	4	12	0.9294	1.0760	0.180

Table 3: For the first matrix in Example 2.

We next consider a matrix $\boldsymbol{A}$ for which the defining (input) data are

$$n = 20, \quad \varepsilon = 0.08, \quad \beta = 600, \quad \gamma_0 = 4, \quad \gamma_1 = 1. \tag{5.14}$$

The relevant data for this matrix are given in Table 4.

Finally, we construct a matrix $\mathbf{A}$ whose defining data are

$$n = 50, \quad \varepsilon = 0.1, \quad \beta = 5000, \quad \gamma_0 = 4, \quad \gamma_1 = 1.2. \tag{5.15}$$

The relevant data for this matrix are given in Table 5.

Input data					Output data				
n	ε	β	γ_0	γ_1	k	N	$\lvert\mu^*\rvert$	r^*	T[s]
20	0.08	600	4	1	4	15	0.8125	1.2307	0.381

Table 4: For the second matrix in Example 2.

Input data					Output data				
n	ε	β	γ_0	γ_1	k	N	$\lvert\mu^*\rvert$	r^*	T[s]
50	0.1	5000	4	1.2	7	30	0.9294	1.0342	1.232

Table 5: For the third matrix in Example 2.

Example 3. This example has been designed to show that in some cases Algorithm 4.1 of the present method may only terminate with an upper bound $\bar{r}$ on the regularity radius r^*. With this in mind, we construct a family of interval matrices in the following way. As in the previous example, the parameters n, α, ε, γ_0 and γ_1 are fixed at the following values:

$$n = 10, \quad \alpha = 2, \quad \varepsilon = 0.04, \quad \gamma_0 = 0.4, \quad \gamma_1 = 0.8. \tag{5.16}$$

The parameter β, however, is now variable. The objective is to (approximately) determine the boundary value $\underline{\beta}$ of β for which the exact value of r^* can still be determined without resorting to the use of Procedure 4.1. The value of $\underline{\beta}$ was found through decreasing the current value of β by an increment $\Delta\beta$ until failure of the algorithm mode studied. The increment was chosen to be $\Delta\beta = 1$. The relevant results obtained are given in Table 6. It is worth noting that the constant sign pattern at the last iteration

Input data		Output data				
n	β	k	N	$\lvert\mu^*\rvert$	r^*	T[s]
10	8	8	31	0.2323	4.3050	0.550

Table 6: The first experiment for Example 3.

of the basic cycle ($k = 8$) is

$$z^x = (1 \quad -1 \quad -1 \quad -1 \quad -1 \quad -1 \quad -1 \quad -1 \quad -1 \quad -1), \tag{5.17a}$$

$$z^y = (-1 \quad 1 \quad -1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1), \tag{5.17b}$$

i.e. no components $\boldsymbol{y}_i$ and $\boldsymbol{x}_j$ of the outer solutions $\boldsymbol{y}$and $\boldsymbol{x}$ contain zero.

If $\bar{\beta} = 8$ is decreased by $\Delta\beta$ to $\beta = 7$, the method is now incapable of determining the radius r^* (if Procedure 4.1 is not used). In this case, Algorithm 4.1 terminates in Step 10. Indeed, for two consecutive iterations the sign pattern vector $z = (z^x; z^y)$ with components

$$z^x = (1 \quad 0 \quad -1 \quad -1 \quad -1 \quad -1 \quad -1 \quad -1 \quad -1 \quad 0), \tag{5.18a}$$

$$z^y = (-1 \quad 1 \quad 0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 1 \quad 0), \tag{5.18b}$$

remains unchanged. It is seen that the subsets referring to the zero elements of (5.18a) and (5.18b) are $I_2 = \{2, 10\}$ and $J_2 = \{3, 5, 6, 8, 10\}$ so $m_1 = 2$ while $m_2 = 5$. Thus, $m = m_1 + m_2 = 7$. If we assume (just for illustrative purposes) that $\bar{m} = 6$, then $m > \bar{m}$ and, hence, Procedure 4.1 is inapplicable. In Table 7, data about the results obtained are given which illustrate the case where only an upper bound $\bar{r}$ on the regularity radius r^* can be obtained using the lower bound $\underline{\mu}$ found by Procedure 3.1.

Input data		Output data				
n	$\bar{\beta}$	k	N	$\lvert\underline{\mu}\rvert$	$\bar{r}$	T[s]
10	7	7	31	0.2141	4.6713	0. 460

Table 7: The second experiment in Example 3.

Further reduction of β to $\beta = 6$ results in the subsets $I_2 = \{2 \div 10\}$ and $J_2 = \{2 \div 8, 10\}$ with $m_1 = 9$ and $m_2 = 8$, respectively, since for two consecutive iterations

$$z^x = (1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0), \tag{5.19a}$$

$$z^y = (1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad -1 \quad 0). \tag{5.19b}$$

Thus, $m = m_1 + m_2 = 17$, $m > \bar{m}$ (if $\bar{m} = 12$) and, hence, Procedure 4.1 is inapplicable. In Table 8, data about the results concerning the upper bound $\bar{r}$ are given. It is seen

Input data		Output data				
n	$\bar{\beta}$	k	N	$\lvert\underline{\mu}\rvert$	$\bar{r}$	T[s]
10	6	4	23	0.1451	6.8895	0.391

Table 8: The third experiment in Example 3.

that for $\beta = 6$ the failure of the method to determine the regularity radius r^* precisely is due to the fact that too many components $\boldsymbol{y}_i$ and $\boldsymbol{x}_j$ of the outer solutions $\boldsymbol{y}$ and $\boldsymbol{x}$ contain zero.

Finally, we let $\beta = 2$. Once again, Algorithm 4.1 terminates in Step 10. The relevant data for this case are given in the following table. However, Algorithm 4.1

Input data		Output data				
n	$\bar{\beta}$	k	N	$\lvert\underline{\mu}\rvert$	$\bar{r}$	T[s]
10	2	2	6	0. 0832	12.0217	0.080

Table 9: The fourth experiment in Example 3.

terminates in Step 10 because the method of [14] cannot find an outer solution to system (3.26a) in Step 2. Indeed, the matrices $\boldsymbol{C}$ and $\boldsymbol{D}$ related to the interval matrix $\boldsymbol{A}$ considered turn out not to be strongly regular at some iteration k (the matrix M from the method [14] is not non-negative).

Input data		Output data				
n	β	k	N	$\lvert\mu^*\rvert$	r^*	T[s]
10	6	4	18	0.1968	5.0820	42.16

Table 10: Algorithm 4.2 results for the first matrix in Example 4.

Example 4. This example illustrates the application of Algorithm 4.2. Two matrices are considered. The input data for the first matrix are same as those given in (5.16) and Table 8. The output results are given in Table 10. It is seen that Algorithm 4.2 determines r^* exactly while Algorithm 4.1 provides only an upper bound $\bar{r}$. On the other hand, the time needed by the second algorithm is much larger than that of the first algorithm. Nevertheless, Algorithm 4.2 solves the problem considered in polynomial time since p_X and p_Y are polynomial bounded. The values of p_X and p_Y for the corresponding iteration (value of k) are given in Table 11.

k	1	2	3	4
p_X	5	2	1	1
p_Y	24	12	2	1

Table 11: Values for the first matrix in Example 4.

The second matrix has the input data of the example given in (5.16) and Table 9. The corresponding output data for the solution obtained by Algorithm 4.2 are given in Table 12 and Table 13. Once again, Algorithm 4.2 solves the problem exactly

Input data		Output data				
n	β	k	N	$\lvert\mu^*\rvert$	r^*	T[s]
10	2	5	15	0.1388	7.2060	50.83

Table 12: Algorithm 4.2 results for the second matrix in Example 4.

k	1	2	3	4	5
p_X	11	6	3	1	1
p_Y	24	6	2	2	1

Table 13: Values for the second matrix in Example 4.

whereas Algorithm 4.1 fails to compute r^*.

Example 5. This example serves to show that the present method can, when applicable, determine the regularity radius r^* of interval matrices of higher sizes (up to $n = 500$ in the present paper) in polynomial time.

We first consider an interval matrix $\boldsymbol{A}$ of size $n = 100$ that is constructed in the same way as in Example 2, (5.15). The parameter α is again $\alpha = 2$ but $\beta = 50000$.

We try to compute r^* by applying Algorithm 4.1. It turns out that for the matrix $\boldsymbol{A}$ considered the resulting matrices $\boldsymbol{C}$ and $\boldsymbol{D}$ are all strongly regular so Algorithm 4.1 solves the problem of determining r^*. The output data are given in Table 14.

Input data		Output data				
n	$\bar{\beta}$	k	N	$\lvert\mu^*\rvert$	r^*	T[s]
100	50000	4	13	2.6971	0.3708	5.434

Table 14: Algorithm 4.1 output for the first matrix in Example 4.

Next n is increased to $n = 200$. Now Algorithm 4.1 is inapplicable since $\boldsymbol{C}$ and $\boldsymbol{D}$ are not strongly regular. However, if α is changed from $\alpha = 2$ to $\alpha = 6$ Algorithm 4.1 is again applicable yielding the output results given in Table 15.

Input data		Output data				
n	$\bar{\beta}$	k	N	$\lvert\mu^*\rvert$	r^*	T[s]
200	50000	5	21	6.4256	0.1556	30.43

Table 15: Algorithm 4.1 output for the order 200 matrix.

Further increase of n to $n = 300$, $n = 400$ and $n = 500$ showed that Algorithm 4.1 is only capable of finding the upper bound $\bar{r}$ on r^* for either value of $\alpha = 2$ or $\alpha = 6$. Output data for the case of $\alpha = 2$ are given in Tables 16, 17 and 18.

Input data		Output data				
n	$\bar{\beta}$	k	N	$\lvert\underline{\mu}\rvert$	$\bar{r}$	T[s]
300	50000	6	35	0.3657	2.7226	154.8

Table 16: Algorithm 4.1 applied to an order 300 matrix.

Input data		Output data				
n	$\bar{\beta}$	k	N	$\lvert\underline{\mu}\rvert$	$\bar{r}$	T[s]
400	50000	6	42	0.2799	3.5723	407.6

Table 17: Algorithm 4.1 applied to an order 400 matrix.

It is interesting that for $n = 300$ Algorithm 4.1 is incapable of determining r^* since the sign vector $z = (z_x; z_y)$ contains the same pattern of zero elements for two consecutive iterations of the basic cycle. More precisely, the number of zeros in z_x is 235 and there are 266 zeros in z_y.

Finally, the second version Algorithm 4.2 has been applied to the same matrices with $n = 300$, $n = 400$ and $n = 500$ (as defined in Tables 16, 17 and 18) in an attempt to determine r^*. The corresponding output data are reported in Tables 19, 20 and 21.

Input data		Output data				
n	$\bar{\beta}$	k	N	$\lvert\underline{\mu}\rvert$	$\bar{r}$	T[s]
500	50000	5	43	0.2449	4.0834	821.7

Table 18: Algorithm 4.1 applied to an order 500 matrix.

Input data		Output data				
n	$\bar{\beta}$	k	N	$\lvert\mu^*\rvert$	r^*	T[s]
300	50000	1	6	0.3711	2.6945	674.2

Table 19: Algorithm 4.2 applied to an order 300 matrix.

Input data		Output data				
n	$\bar{\beta}$	k	N	$\lvert\mu^*\rvert$	r^*	T[s]
400	50000	1	7	0.2896	3.4528	1620

Table 20: Algorithm 4.2 applied to an order 400 matrix.

Input data		Output data				
n	$\bar{\beta}$	k	N	$\lvert\mu^*\rvert$	r^*	T[s]
500	50000	1	7	0.2482	4.0293	1403

Table 21: Algorithm 4.2 applied to an order 500 matrix.

The smaller time T in Table 21 as compared with Table 20 is explained by the fact that the numbers of intersections p_X and p_Y are larger for the former problem than for the latter one.

It is worth noting the following two points. First, the upper bounds $\bar{r}$ obtained by Algorithm 4.1 (reported in Tables 16, 17 and 18) are fairly good approximations on the corresponding regularity radii r^*. Second, the computer time needed to determine the exact (within rounding) values of r^* by Algorithm 4.2 increases slowly with respect to n. Although this is due to the fact that the number of orthants intersected by the solution sets X and Y for all matrices considered here is small, this experimental observation is encouraging since it seems realistic in various practical problems. Indeed, frequently, due to physical or economical restrictions, many of the problem variables x_i do not change sign, which could lead to small numbers of orthant intersections and, hence, to possible applications of Algorithm 4.2.

Example 6. In this final example, the present method is tested on a set S of randomly generated interval matrices whose coefficients are normally distributed with zero mean. More specifically, as in the last example of [5] the center matrix A^c and radius matrix R of each individual matrix A from $\boldsymbol{A}$ are given by

$$A^c = \texttt{randn}(n),\ \rho = 0.02 \cdot \texttt{randn} \text{ and } R = \rho \cdot \texttt{randn}(n), \tag{5.20}$$

where `randn` is the MATLAB program for creating normal distribution matrices. The

results obtained are given in Tables 22 to 24 for $n = 10$, $n = 20$, and $n = 30$, respectively. The test turns out to be a difficult one for Algorithm 4.1: the regularity

Algorithm 4.1 for $n = 10$				Algorithm 4.2 for $n = 10$			
$\rho \times 10^{-2}$	$\bar{r}$	r^*	T[s]	$\rho \times 10^{-2}$	$\bar{r}$	r^*	T[s]
2.6024	3.8168		0.152	0.6700	4.7335		10.7
0.2108	8.9883		0.055	0.3386	-	3.664	7.29
0.3654		2.1310	0.237	0.6812	10.4380		0.056
2.5280	0.9997		0.025	2.7911	3.6758		48.0
1.8651	22.3132		0.0084	1.5358		5.5647	51.8
2.4903	0.8429		0.027	0.7820	4.8362		0.188
2.0201	1.4668		0.012	1.0290		5.5179	20.3
1.3600	8.3255		0.017	2.1014		1.5314	32.2
3.2781	3.6622		0.089	0.5412	23.6189		0.072
3.8815	1.1156		0.164	2.0072		1.5178	28.8

Table 22: The algorithms applied to randomly generated matrices of order 10.

Algorithm 4.1 for $n = 20$				Algorithm 4.2 for $n = 20$			
$\rho \times 10^{-2}$	$\bar{r}$	r^*	T[s]	$\rho \times 10^{-2}$	$\bar{r}$	r^*	T[s]
1.9907	0.6663		0.137	0.8452		1.6287	150
2.3800	0.3619		0.043	2.4031	0.4579		64.7
2.9500	0.7598		0.030	0.3891		3.0330	208
2.3837	0.3619		0.042	2.3457		0.3162	312
2.1100	2.0778		0.114	1.0564		0.6729	151
0.8600	0.1847		0.669	2.0041		0.1684	36.6
2.1400	0.3740		0.376	0.5610		0.7012	140
1.6327		0.4892	0.694	0.3586		0.9545	37.4
3.0100	0.3948		0.316	1.6027	0.4502		76.7
0.9400	2.6116		0.046	0.6010	7.8567		1.01

Table 23: The algorithms applied to randomly generated matrices of order 20.

radius r^* has been determined only for 3 matrices (one for each n). For the remaining 27 matrices, Algorithm 4.1 is only capable of computing an upper bound $\bar{r}$ on r^*. This is due mainly to the fact that most of the generated matrices (24 out of 27) proved not to be strongly regular and, hence, the outer solutions $\boldsymbol{u}$ and $\boldsymbol{v}$cannot be found by the method of [14]; for 3 matrices, Algorithm 4.1 terminated in an incomplete sign constancy condition.

As expected, Algorithm 4.2 performs much better than Algorithm 4.1: the total number of cases with successful determination of r^* is now 15. For the remaining matrices the algorithm has only yielded an upper bound $\bar{r}$ on r^*.

The data on the times needed by the successful runs of Algorithm 4.2 indicate that the algorithm requires a polynomial amount of computation for the set of matrices considered. Indeed, let T^a_{10}, T^a_{20}, and T^a_{30} denote the average time taken by the

Algorithm 4.1 for $n = 30$				Algorithm 4.2 for $n = 30$			
$\rho \times 10^{-2}$	$\bar{r}$	r^*	T[s]	$\rho \times 10^{-2}$	$\bar{r}$	r^*	T[s]
2.1200	0.6362		0.338	0.8800	0.8551		856
0.3100	2.1946		0.863	0.890	1.0126		1255
0.4600	3.807		0.218		18.198		2.107
3.2400	0.2950		0.471	0.4861		3.6539	2211
1.6100	0.2450		0.864	0.0702	12.570		159
0.9900	0.5481		0.202	0.2030		2.5244	89.6
1.6500	0.8362		0.404	0.7100	2.7909		0.278
0.8237		0.2059	0.579	0.5000	10.871		0.094
1.9400	0.7876		0.125	2.7900	0.4284		178
2.8700	0.8983		0.554	1.7050		0.3527	188

Table 24: The algorithms applied to randomly generated matrices of order 30.

successful runs for $n_1 = 10$, $n_1 = 20$, and $n_1 = 30$, respectively. So

$$T^a_{10} = 28.08, \quad T^a_{20} = 147.9, \quad T^a_{30} = 829.5.$$

It is seen that

$$q_1 = T^a_{20}/T^a_{10} = 5.27 < 10$$
$$q_2 = T^a_{20}/T^a_{10} = 5.61 < 10.$$

If the method had exponential complexity, the corresponding quotients would be

$$q'_1 = 2^{n_2}/2^{n_1} = 2^{10} = 1024$$
$$q'_2 = 2^{n_3}/2^{n_2} = 2^{10} = 1024.$$

The comparison of q_1 with q'_1 and q_2 with q'_2 confirm the conclusion concerning the polynomial complexity of the present method for the set of matrices considered in this example.

Remark 5.1. The choice (5.20) for constructing the random matrices from the present example is not unique. An alternative rule for constructing the random matrices used (as suggested by the first reviewer of the present paper) could be:

$$A^c = \texttt{randn}(n), \quad R = \texttt{ones}(n). \tag{5.21}$$

The choice (5.20), however, seems preferable (at least for the Example 6) for the following three reasons:

(i) The choice (5.21) for R makes the example too unbalanced: numerical evidence shows that the overwhelming majority of matrices thus generated are singular (but as is well known establishing singularity is much easier than checking regularity). The coefficient 0.02 used in the expression for ρ in (5.20) ensures that the set S contains both regular and singular interval matrices.

(ii) Such a choice for R is quite natural for engineering applications (cf., e.g., [6]) since each element r_{ij} is given a "tolerance" which is a certain small percentage of the "nominal" value $(A^c)_{ij}$. A tolerance of 2% is a typical value.

(iii) Finally, the choice (5.20) has already been used in [5] for $n = 10$ and the author of the present paper was curious to obtain comparable numerical results for higher n.

6 Concluding remarks

It has been shown that the problem of determining the regularity radius r^* of an interval matrix can be equated to that of finding the real maximum magnitude eigenvalue μ^* of an associated interval generalized eigenvalue problem (2.3). An iterative method of polynomial complexity for solving the latter problem has been suggested. At each iteration, a lower bound $|\underline{\mu}|$ on $|\mu^*|$ and outer interval bounds $\boldsymbol{x}$ and $\boldsymbol{y}$ or interval hull envelopes $\boldsymbol{x}^*$ and $\boldsymbol{y}^*$ on the right and left eigenvectors of (2.3) are used. The outer bounds $\boldsymbol{x}$ and $\boldsymbol{y}$ or the hull envelopes $\boldsymbol{x}^*$ and $\boldsymbol{y}^*$ are found using the corresponding interval solutions of the linear interval systems (3.26a) and (3.29). The method suggested is capable of determining the exact (within rounding) regularity radius r^* if $\underline{\mu}$ and μ^* are simple eigenvalues and, eventually, the solutions $\boldsymbol{x}$ and $\boldsymbol{y}$ or $\boldsymbol{x}^*$ and $\boldsymbol{y}^*$ satisfy the constant sign conditions (3.36) associated with the original interval matrix $\boldsymbol{A}$ or the current modified matrix (4.1). Otherwise, it only provides a tight upper bound $\bar{r}$ on r^*.

Several numerical examples with interval matrices whose size is $n \leq 500$ illustrate the potential of the new method. It should, however, be borne in mind that its present implementation is based on full-matrix operations, limiting the application of the method to moderate size problems.

Future research will concentrate on enhancing the numerical efficiency and, hence, the applicability of the method. One possibility is to implement the algorithm in Section 4 using sparse matrix techniques. An open problem is to extend the present method to interval matrices where double (or multiple) eigenvalues $\underline{\mu}$ and μ^* are encountered. Another objective is to generalize this paper's approach to the case of parametric matrices, in particular to matrices whose elements depend linearly on a set of interval parameters.

Acknowledgements

The author wishes to express his gratitude to two of the anonymous referees for their valuable comments and suggestions.

References

[1] L. Carotenuto et al. Computational method to analyze the stability of interval matrices. *IEE Proc. Control Theory Appl.*, 151:669–674, 2004.

[2] L. Jaulin et al. *Applied Interval Analysis with Examples in Parameter and State Estimation: Robust control and Robotics*. Springer, London, 2001.

[3] C. Jansson. Interval linear systems with symmetric matrices, skew-symmetric matrices and dependencies in the right hand side. *Computing*, 46:265–274, 1991.

[4] C. Jansson. Calculation of exact bounds for the solution set of linear interval systems. *Linear Algebra and its Applications*, 251:321–340, 1997.

[5] C. Jansson and J. Rohn. An algorithm for checking regularity of interval matrices. *SIAM J. Matrix Anal. Appl.*, 20:756–776, 1999.

[6] L. Kolev. *Interval Methods for Circuit Analysis*. World Scientific, Singapore, New Jersey, London, 1993.

[7] L. Kolev. Determining the positive definiteness margin of interval matrices. *Reliable Computing*, 13:445–466, 2007.

[8] L. Kolev L. and S. Patrakieva. Assessing the stability of linear time-invariant continuous interval dynamic systems. *IEEE Transactions on Automatic Control*, 50:393–397, 2005.

[9] M. Mansour. Robust stability of interval matrices. In *Proceedings of the 28th Conference on Decision and Control, Tampa, FL*, pages 46–51, 1989.

[10] S. Poljak and J. Rohn. Radius of nonsingularity. Technical report, Research Report, KAM Series 88-117, Faculty of Mathematics and Physics, Charles University, Prague, 1988.

[11] S. Poljak and J. Rohn. Checking robust nonsigularity is NP-hard, mathematics of control. *Signals, and Systems*, 6:1–9, 1993.

[12] B. Polyak. Robust linear algebra and robust aperiodicity. In *Directions in Mathematical Systems Theory and Optimization, LNCIS 2003, 286*, pages 249–260. A. Rantzer, C. Byrnes (Eds.), 2003.

[13] J. Rohn. Stability of interval matrices: the real eigenvalue case. *IEEE Transactions on Automatic Control*, 37:1604–1605, 1992.

[14] J. Rohn. Cheap and tight bounds: the recent result by E. Hansen can be made more efficient. *Interval Computations*, pages 13–21, 1993.

[15] J. Rohn. Interval matrices: singularity and real eigenvalues. *SIAM Journal on Matrix Analysis and Applications*, 14:82–91, 1993.

[16] J. Rohn. Checking positive definiteness or stability of symmetric interval matrices is np-hard. *Computat. Math. Univ. Carol.*, 35:795–797, 1994.

[17] J. Rohn. Positive definiteness and stability of interval matrices. *SIAM Journal on Matrix Analysis and Applications*, 15:175–184, 1994.

[18] J. Rohn. An algorithm for checking stability of symmetric interval matrices. *IEEE Trans. on Automatic Control*, 41:133–136, 1996.

[19] S. Rump. *INTLAB INTerval LABoratory, in Tibor Csendes, ed., Developments in Reliable Computing, (1999), pp. 77–105.* Kluwer, Dordrecht, Netherlands, 1999.