Mathematics and Statistics 10(6): 1229-1238, 2022
DOI: 10.13189/ms.2022.100609 http://www.hrpub.org

Iterative Algorithms for Solving the Partial Eigenvalue Problem for Symmetric Interval Matrixes

Alimzhan A. Ibragimov*, **Dilafruz N. Khamroeva**

Department of Informatics, Faculty of Mathematics and Informatics, Navoi State Pedagogical Institute, Uzbekistan

Received July 28, 2022; Revised September 29, 2022; Accepted October 11, 2022

Cite This Paper in the following Citation Styles

***(a)**: [1] Alimzhan A. Ibragimov, Dilafruz N. Khamroeva, "Iterative Algorithms for Solving the Partial Eigenvalue Problem for Symmetric Interval Matrixes," Mathematics and Statistics, Vol.10, No.6, pp. 1229-1238, 2022. DOI: 10.13189/ms.2022.100609*

***(b)**: Alimzhan A. Ibragimov, Dilafruz N. Khamroeva (2022). Iterative Algorithms for Solving the Partial Eigenvalue Problem for Symmetric Interval Matrixes. Mathematics and Statistics, 10(6), 1229-1238. DOI: 10.13189/ms.2022.100609*

Abstract In this paper, we consider iterative methods for solving a partial eigenvalue problem for real symmetric interval matrices. Such matrices have applications in modeling many technical problems where a lot of data suffers from limited variation or uncertainty. In modeling most applied problems, when some parameter values fluctuate with a known amplitude, then it can be considered that it is advisable to use interval methods. The algorithms proposed by us are built on the basis of the power method and its modification, the so-called "Method of scalar products" for solving a partial problem of eigenvalues of an interval symmetric matrix. These methods have not yet been studied in detail and are not justified for interval matrices. In the developed algorithms, boundary matrices are first determined by the Deif theorem, and then a partial eigenvalue problem is solved. We also study the problem of convergence of the power method for boundary matrices of a given interval symmetric matrix. The results of the computational experiment show that the interval eigenvalues obtained by the proposed algorithms are in good agreement with the results obtained by other researchers, and in some cases even better. The obtained numerical results are compared by the number of iterations and the width of the interval solution.

Keywords Partial Eigenvalue Problem, Real Symmetric Interval Matrices, Power Method, Scalar Product Method, Uncertainty, Interval Analysis, Iterative Algorithms, Outer and Inner Approximation

1 Introduction

World scientific practice shows that the solution of the eigenvalue problem is one of the main tasks of linear algebra, and it underlies almost all engineering problems. For example, in electrical and mechanical systems, eigenvalues correspond to natural vibration frequencies, and eigenvectors characterize the corresponding vibration modes. Also, in the theory of dynamical systems, knowledge of eigenvalues allows one to determine the nature of the system's behavior in time and to solve the question of the stability of such a system. However, in these problems, which are real-life problems, exact data is very rare, since the input data is affected by various uncertainties.

The use of mathematical methods in solving a number of applied problems is associated with the need to take into account inaccurate data. Most researchers and practitioners manage in such cases by probabilistic-statistical methods. But, as the problems being solved become more complex, new engineering disciplines are involved in the scientific use of mathematics, the limitations of traditional probabilistic-statistical representations become obvious, their inapplicability for modeling a number of important processes and phenomena. The response to this challenge of practice was the emergence of fuzzy and interval approaches [1-4] to the description of uncertainties. These relatively new approaches are analytically, firstly, extremely efficient in algorithmic and computational terms, and, secondly, allowing to correctly investigate meaningful models. In turn, in situations where inaccurate data have a limited amplitude of fluctuations, then the use of interval analysis methods allows them to be accounted for, as they say, "from beginning to end". This means the following: first, when synthesizing a mathematical model or at the stage of analyzing an already known mathematical model, a "reasonable" intervalization of parameters is performed, further research is carried out within the framework of an appropriate structure, and the

results are obtained using the appropriate software. This process ultimately allows one-time accounting of all error sources and obtaining guaranteed two-sided approximations for the desired solutions.

Based on the above discussions and analyzing numerous research papers devoted to solving the problem of eigenvalues, we can assume that this direction also needs to apply methods that allow taking into account uncertainties in the data.

Interval methods have found wide application in various fields of research, including the development and improvement of interval algorithms for solving algebraic eigenvalue problems in the works of A. Deif, J. Rohn, R. Lohner, G. Mayer, L. Kolev, D. Hertz, M. Hladik, Z. Qiu, X. Leng, D. Hartman, M.-H. Matkovschi, etc.

The first relevant research works in this direction appeared in 1984-1986 of the last century, and some primitive results are given in the works of Barmish [5], Hudak [6], Juang [7] and Qiu [8]. Further results are due to Deif [9] and Rohn [10, 11]. Deif gives a description of the eigenvalue sets for symmetric and nonsymmetric cases, as well as their exact estimates based on the sign invariance of the eigenvectors. J. Rohn in [11] obtained theoretical estimates and practical applications for the boundaries of eigenvalues of asymmetric interval matrices. M. Hladik and his co-authors [12] propose the so-called filtering method for solving the interval eigenvalue problem, and also in [13] give an optimal formula for limiting the real and imaginary parts of the eigenvalues of complex-valued interval matrices. Qiu et al. [14], based on the work of Rohn [11], develop the interval perturbation method for estimating the set of all possible eigenvalues of real nonsymmetric interval matrices. Kolev [15] considers the problem of external interval estimation of eigenvalues with parametric matrices. Mayer's book [16] presents the main results obtained in his research on the interval eigenvalue problem, where he also proposes an enclosure method for eigenvalues of real and complex interval matrices based on the Taylor series expansion.

In many applied problems, we are not interested in all eigenvalues, but only in some of them. For example, high-order matrices are obtained by finite-difference solution of eigenvalue problems for differential equations. In this case, it suffices to calculate only a few smallest eigenvalues corresponding to a small number of zeros of the eigenfunction. Also, in neutron diffusion problems, only one or two eigenvalues have a physical meaning. In such cases, it is unprofitable to solve the full eigenvalue problem. Usually, to solve such problems, iterative processes are used that converge to the maximum (minimum) modulo eigenvalue and the corresponding eigenvector. One of the frequently used such iterative methods is the power method. This method is used in most cases for sparse matrices. For example, the Google search engine uses it to calculate page ranks on the Internet [17], and Twitter uses it to recommend "who to follow" [18].

In this paper, we consider a partial eigenvalue problem for $\boldsymbol{A} \in \mathcal{IR}^{n\times n}$ interval matrices using the power method and some of its modifications.

2 Preliminary facts and notation

In what follows, we will use the standard notation of interval analysis [19].

Definition 1. Let $\underline{a}$ and $\overline{a}$ be real numbers satisfying the conditions $\underline{a} \leq \overline{a}$. Then the set $\boldsymbol{a} = [\underline{a}, \overline{a}] = \{x \in \mathcal{R} \mid \underline{a} \leq x \leq \overline{a}\}$ is called **a real interval**. The set of all real intervals is denoted as $\mathcal{IR}$. The real numbers $\underline{a}$ and $\overline{a}$ are called respectively the **lower** and **upper** bounds of the interval $\boldsymbol{a}$.

Definition 2. Let $\boldsymbol{a} = [\underline{a}, \overline{a}]$ be a real interval. Then

- The **width** of the interval $\boldsymbol{a}$ is defined as $\mathtt{wid}\ \boldsymbol{a} = \overline{a} - \underline{a}$;
- The **radius** of the interval $\boldsymbol{a}$ is calculated as $\mathtt{rad}\ \boldsymbol{a} = \frac{1}{2}(\overline{a} - \underline{a})$;
- The **midpoint** or **center** of the interval $\boldsymbol{a}$ is defined as $\mathtt{mid}\ \boldsymbol{a} = \frac{1}{2}(\underline{a} + \overline{a})$.

Definition 3. The interval square matrix is defined as $\boldsymbol{A} := [\underline{A}, \overline{A}] = \{A \in \mathcal{R}^{n\times n};\ \underline{A} \leq A \leq \overline{A}\}$, matrices $\underline{A}, \overline{A} \in \mathcal{R}^{n\times n}$ are called its boundaries.

Thus, if $\underline{A} = (\underline{a}_{ij})$ and $\overline{A} = (\overline{a}_{ij})$, then $\boldsymbol{A}$ is the set of all matrices $A = (a_{ij})$ satisfying the inequalities $\underline{a}_{ij} \leq a_{ij} \leq \overline{a}_{ij}$ for $i, j = 1, 2, ..., n$. Through

$$A_c = \frac{\underline{A} + \overline{A}}{2} \quad \text{and} \quad A_\Delta = \frac{\overline{A} - \underline{A}}{2},$$

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

It can be seen that the elements of the A_c and A_Δ matrices are real, not interval. Then we can represent the interval matrix as $\boldsymbol{A} = [A_c - A_\Delta,\ A_c + A_\Delta]$.

3 Statement of the problem

We will consider a standard eigenvalue problem with an interval matrix $\boldsymbol{A} \in \mathcal{IR}^{n\times n}$ of the form

$$\boldsymbol{A}x = \lambda x \tag{1}$$

Problem (1) is one of the main problems in the field of interval analysis, which has interesting properties and is widely used in many technical systems.

In general, the set of all eigenvalues for problem (1) is defined as

$$\Lambda(\boldsymbol{A}) := \{\lambda \in \mathcal{C};\quad Ax = \lambda x,\quad x \neq 0,\quad A \in \boldsymbol{A}\}. \tag{2}$$

However, the problem of eigenvalues for complex interval matrices is considered, for example, in [11, 13, 20-21].

The calculation of the set (2) in the general case is a difficult task, and in some cases, even for symmetric interval matrices, it can turn out to be NP-hard [22]. Also, checking whether interval matrices are positive (semi) definite [23] or nonsingular [24, 25] is an NP-hard problem that is closely related to solving the eigenvalue problem.

The authors in [28 (Proposition 2.1.), 30] show that the set of real eigenvalues of an interval matrix is a finite union of

compact real intervals. However, in an analytical sense, we cannot calculate the exact boundaries of these intervals, this is different from calculating the eigenvalues of individual real (ordinary, scalar) matrices, for which there are simple and exact formulas (for example, finding the roots of the characteristic polynomial of a matrix). For the case of interval matrices, we will look for inner and outer estimates (approximations) of these bounds.

Definition 4. In solving the interval problem (1) of eigenvalues we call:

1) *outer approximation*, when the interval $\boldsymbol{\lambda} \in \mathcal{IR}$ is searched, enclosing the set $\Lambda(\boldsymbol{A})$ of solutions of the interval system of equations $\boldsymbol{A}x = \lambda x$, i.e. $\boldsymbol{\lambda} \supseteq \Lambda(\boldsymbol{A})$;

2) *inner approximation*, when the interval $\boldsymbol{\mu} \in \mathcal{IR}$ contained in the set $\Lambda(\boldsymbol{A})$ of solutions of the interval system of equations $\boldsymbol{A}x = \mu x$ is searched, i.e. $\boldsymbol{\mu} \subseteq \Lambda(\boldsymbol{A})$;

In this paper, we will study only the symmetric case $\boldsymbol{A} = \boldsymbol{A}^\top$ and solve the outer interval problem. Here we assume that if we are given a symmetric interval matrix, then all $A \in \boldsymbol{A}$ are symmetric, it is also easy to check that A_c and A_Δ are also symmetric.

A symmetric matrix is usually denoted as [22, 23, 28, 30]:

$$\boldsymbol{A}^s := \{A \in \boldsymbol{A} \mid A = A^\top\}. \tag{3}$$

The symmetric interval matrix $\boldsymbol{A}^s$ is a special case of an arbitrary interval matrix $\boldsymbol{A}$. The set of eigenvalues for a symmetric interval matrix is written similarly to the general case (2), i.e.

$$\Lambda(\boldsymbol{A}^s) := \{\lambda \in \mathcal{R} \mid A^s x = \lambda x,\ x \neq 0,\ A^s \in \boldsymbol{A^s}\}, \tag{4}$$

and we will look for an interval eigenvalue $\boldsymbol{\lambda}_1 \in \mathcal{IR}$ and an interval vector $\boldsymbol{x} \in \mathcal{IR}^n$ such that $\boldsymbol{\lambda}_1$ contains the eigenvalue $\lambda_1^* \in \mathcal{R}$ and $\boldsymbol{x}_1$ contains the corresponding eigenvector $x_1^* \in \mathcal{R}^n \backslash \{0\}$ of the matrix $A^s \in \boldsymbol{A^s}$.

4 Iterative algorithms

If for the eigenvalues of $\lambda_i, \quad i = 1, 2, \ldots, n$, the real matrix $A \in \mathcal{R}^{n \times n}$ the inequality

$$|\lambda_1(A)| > |\lambda_2(A)| \geq |\lambda_3(A)| \geq \ldots \geq |\lambda_n(A)|, \tag{5}$$

then λ_1 is called the *dominant eigenvalue*, and the corresponding eigenvector is called the *dominant eigenvector*.

The power method considered in this paper is designed to solve a partial eigenvalue problem, i.e. finding the dominant eigenvalue and eigenvector of the matrix. The idea underlying it is extremely simple and consists in the fact that if the matrix $A \in R^{n \times n}$ has an eigenvalue λ_1 that exceeds all other eigenvalues in absolute value, then when this matrix acts on an arbitrary vector $x \in \mathcal{R}^n$, the direction ν_1 corresponding to this eigenvalue λ_1 will stretch more the rest (at $\lambda_1 > 1$) or shrink less than the others (at $\lambda_1 \leq 1$).

As an approximate eigenvalue of the matrix A, we can take the "ratio" of two successive vectors generated by our process - $x^{(k+1)} = A^{k+1}x^{(0)}$ and $x^{(k)} = A^k x^{(0)}$, $k = 0, 1, 2, \ldots$.

There are the following ways to solve this issue:

1) consider the ratio of some fixed components of the vectors $x^{(k+1)}$ and $x^{(k)}$, i.e.

$$x_i^{(k+1)} / x_i^{(k)}, \tag{6}$$

for some $i \in \{1, 2, ..., n\}$;

2) consider the ratio of the projections of successive approximations $x^{(k+1)}$ and $x^{(k)}$ onto the direction given by some vector $l^{(k)}$, i.e.

$$\frac{\langle x^{(k+1)}, l^{(k)} \rangle}{\langle x^{(k)}, l^{(k)} \rangle}. \tag{7}$$

In the second case, we marked the design direction as $l^{(k)}$ to emphasize its possible dependence on the step number k. It is also clear that this direction $l^{(k)}$ does not need to be orthogonal to the vector $x^{(k)}$ in order for $\langle x^{(k)}, l^{(k)} \rangle \neq 0$ to be the denominator in (7).

The latter method seems to be more computationally preferable, since it avoids capricious behavior in one single component of the vector $x^{(k)}$, when it can become very small in absolute value or completely become zero, although the vector $x^{(k)}$ will have a significant length as a whole. Finally, it is natural to take $x^{(k)}$ it self as a vector specifying the design direction in the second variant, calculating the ratio at each step

$$\frac{\langle x^{(k+1)}, x^{(k)} \rangle}{\langle x^{(k)}, x^{(k)} \rangle}. \tag{8}$$

where $x^{(k)} = A^k x^{(0)}$. It is easy to see that this expression coincides with the Rayleigh ratio for the approximation of $x^{(k)}$ to the eigenvector.

To organize the computational algorithm of the power method, it is necessary to resolve two more subtle points related to the implementation on a computer.

Firstly, this is a possible unlimited increase (at $|\lambda_1| > 1$) or unlimited decrease (at $|\lambda_1| < 1$) of the norms of the vectors $x^{(k)}$ and $x^{(k+1)}$ participating in our process. The bit grid of modern digital computers, as you know, is finite and allows you to represent numbers from a limited range. To avoid problems caused by going outside this range ("overflow" or "disappearance of order"), it makes sense to normalize $x^{(k)}$. In this case, the most convenient normalization is in the Euclidean norm

$$\|x\|_2 = \left(\sum_{i=1}^{n} |x_i|^2 \right)^{1/2},$$

since then the denominator of relation (8) will become equal to one.

Secondly, when deriving the power method, we implicitly assumed that the initial vector $x^{(0)}$ is chosen so that it has a non-zero projection onto the direction of the dominant eigenvector v_1 of the matrix A. Otherwise, the products of any

powers of the matrix A by $x^{(0)}$ will also have zero projections on v_1, and no differentiation of the length of the components $A^k x^{(0)}$, on which the power method is based, will occur.

This difficulty can be overcome with the help of some a priori information about the dominant eigenvector of the given matrix. In addition, in the practical implementation of the power method on digital computers, the inevitable rounding errors, as a rule, lead to the appearance of nonzero components in the direction, which then is expanded by the required value during iteration. But, strictly speaking, this may not happen in some exceptional cases, and therefore, in responsible calculations, it is recommended to run the power method multiple times with different initial vectors (the so-called "*Multistart*").

4.1 Algorithm of the power method for a symmetric interval matrix

And so, we will extend the notation (5) for symmetric interval matrices, i.e.

$$\boldsymbol{\lambda}_i(\boldsymbol{A^s}) := \{\lambda_i(A^s) \mid A^s \in \boldsymbol{A^s}\}. \tag{9}$$

To describe our algorithm, we will use the Deif theorem [9, theorem 1].

Theorem 1: Let $\boldsymbol{A^s} = [A_c^s - A_\triangle^s, A_c^s + A_\triangle^s]$, and the diagonal matrix $D^i = \mathtt{diag}\left(\mathtt{sgn}(x_1^i), \ldots, \mathtt{sgn}(x_n^i)\right)$, $i = 1, 2, \ldots, n$ obtained from A_c^s and be constant for $\boldsymbol{A^s}$, then the eigenvalue λ_i of matrix $A^s \in \boldsymbol{A^s}$ is contained in the interval

$$\boldsymbol{\lambda}_i := \left[\lambda_i(A_c^s - D^i A_\triangle^s D^i), \lambda_i(A_c^s + D^i A_\triangle^s D^i)\right], \quad i = 1, 2, \ldots, n. \tag{10}$$

In a particular case, if, $D^i = I$, where I is the identity matrix, then the desired interval of eigenvalues is $\boldsymbol{\lambda}_i = \left[\lambda_i(\underline{A}^s), \lambda_i(\overline{A}^s)\right]$, where $\underline{A}^s$ and $\overline{A}^s$ are respectively the lower and upper boundaries of the symmetrical interval matrix $\boldsymbol{A^s}$.

Algorithm 1 that we propose is a modification of the Power Method for finding the dominant eigenvalue, i.e. interval containing the largest eigenvalue of all real symmetric matrices $A^s \in \boldsymbol{A^s}$.

Algorithm 1. Power Method for Symmetric Interval Matrix

Input: $\boldsymbol{A}^s = \left[\underline{A},\ \overline{A}\right] \in \mathcal{IR}^{n\times n}$, given accuracy ε, initial approximation vector $\boldsymbol{y}^{(0)} (\neq 0) \in \mathcal{IR}^n$.
Output: eigenvalue interval $\boldsymbol{\lambda}_1 \in \mathcal{IR}$ and its corresponding normalized interval eigenvector $\boldsymbol{x}_1 \in \mathcal{IR}^n$.

1: Compute: $A_c \leftarrow (\underline{A} + \overline{A})/2$, $A_\triangle \leftarrow (\overline{A} - \underline{A})/2$; eigenvalues $\lambda_i(A_c)$, eigenvectors $x_1^i(A_c)$, $x_2^i(A_c)$, $\ldots$, $x_n^i(A_c)$, diagonal matrix $D^i \leftarrow \mathtt{diag}\left(\mathtt{sign}\left(x_1^i(A_c)\right), \ldots, \mathtt{sign}\left(x_n^i(A_c)\right)\right)$, $\underline{B} \leftarrow A_c - D^i A_\triangle D^i$, $\overline{B} \leftarrow A_c + D^i A_\triangle D^i$; $i = 1, 2, \ldots, n$;
$\underline{x}^{(0)} \leftarrow \underline{y}^{(0)}/\|\underline{y}^{(0)}\|_2$, $\overline{x}^{(0)} \leftarrow \overline{y}^{(0)}/\|\overline{y}^{(0)}\|_2$;
2: $k \leftarrow 1$;
3: $\underline{y}^{(k)} \leftarrow \underline{B}\,\underline{x}^{(k-1)}$;
4: $\underline{x}^{(k)} \leftarrow \underline{y}^{(k)}/\|\underline{y}^{(k)}\|_2$;
5: $\underline{\lambda}_i^{(k)} \leftarrow \underline{y}_i^{(k)}/\underline{x}_i^{(k-1)}$;
6: if $\left|\underline{\lambda}_i^{(k)} - \underline{\lambda}_i^{(k-1)}\right| < \varepsilon$ then
begin
$\underline{\lambda} \leftarrow \left(\sum_{i=1}^n \underline{\lambda}_i^{(k)}\right)/n$; $\underline{x} \leftarrow \underline{x}^{(k)}$;
end;
else
begin
$k \leftarrow k + 1$; goto 3
end;
7: $l \leftarrow 1$;
8: $\overline{y}^{(l)} \leftarrow \overline{B}\,\overline{x}^{(l-1)}$;
9: $\overline{x}^{(l)} \leftarrow \overline{y}^{(l)}/\|\overline{y}^{(l)}\|_2$;
10: $\overline{\lambda}_i^{(l)} \leftarrow \overline{y}_i^{(l)}/\overline{x}_i^{(l-1)}$;
11: if $\left|\overline{\lambda}_i^{(l)} - \overline{\lambda}_i^{(l-1)}\right| < \varepsilon$ then
begin
$\overline{\lambda} \leftarrow \left(\sum_{i=1}^n \overline{\lambda}_i^{(l)}\right)/n$; $\overline{x} \leftarrow \overline{x}^{(l)}$;
end;
else
begin
$l \leftarrow l + 1$; goto 8
end;
12: $\boldsymbol{\lambda}_1 \leftarrow \left[\underline{\lambda}, \overline{\lambda}\right]$; $\boldsymbol{x}_1 \leftarrow [\underline{x}, \overline{x}]$.

The initial approximation vector

$$y^{(0)} = \left([\underline{y}_1^{(0)}, \overline{y}_1^{(0)}], [\underline{y}_2^{(0)}, \overline{y}_2^{(0)}], \ldots, [\underline{y}_n^{(0)}, \overline{y}_n^{(0)})]\right)$$

in *Algorithm 1* is chosen as a degenerate interval, i.e. $\underline{y}_i^{(0)} = \overline{y}_i^{(0)}$ this is because we are only dealing with the boundaries of interval values, and they are real. When the algorithm completes its job successfully, then we illustrate the results as interval eigenvalues: $\boldsymbol{\lambda}_1$ is the approximate interval of eigenvalues for the symmetric interval matrix $\boldsymbol{A}^s$, and $\boldsymbol{x}_i^k$ is the current approximation to the normalized corresponding interval eigenvector.

It should be noted that the direct application of interval arithmetic operations to existing methods for calculating the eigenvalues of interval matrices in many cases leads to an overestimation of the width of the interval solution.

The weak point of *Algorithm 1* is steps 6 and 11, i.e. solving the problem of timely termination of the algorithm. These steps are described from rational considerations and cannot guarantee in all cases (even under the assumptions made) that an eigenpair $\{\boldsymbol{\lambda}_1,\ \boldsymbol{x}_1\}$ is obtained with a given accuracy, since

no error estimate was obtained during the development of the method.

To find eigenvalue pairs of symmetric interval matrices, one can construct another equivalent version of the iterative process. Given a symmetric interval matrix, compute the left and right boundary matrices as in `step 1` of *Algorithm 1*:

$$\underline{B} \leftarrow A_c - D^i A_\Delta D^i \quad \text{and} \quad \overline{B} \leftarrow A_c + D^i A_\Delta D^i. \qquad (11)$$

After that, we perform calculations according to *Algorithm 2*, first for the left border $\underline{B}$, then for the right border $\overline{B}$ in the same way. In order not to distinguish between left and right boundary matrices in the algorithm, we will denote them simply as a real matrix B .

Algorithm 2. Power Method for real Symmetric Matrix

Input: $B \in \mathcal{R}^{n\times n}$, given accuracy ε, initial approximation vector $y^{(0)}(\neq 0) \in \mathcal{R}^n$.
Output: dominant eigenvalue $\lambda_1 \in \mathcal{R}$ and its corresponding normalized interval eigenvector $x_1 \in \mathcal{R}^n$.

1: $x^{(0)} \leftarrow y^{(0)}/\|y^{(0)}\|_2$;

2: $k \leftarrow 1$;

3: $y^{(k)} \leftarrow Bx^{(k-1)}$;

4: $x^{(k)} \leftarrow y^{(k)}/\|y^{(k)}\|_2$;

5: $\lambda_i^{(k)} \leftarrow y_i^{(k)}/x_i^{(k-1)}$ (by vector components $y^{(k)},\ x^{(k-1)}$) such that $\left|x_i^{(k-1)}\right| > \delta$, where $\delta > 0$ is some small number (tolerance);

6: Subject the numbers $\lambda_i^{(k)}$ to convergence.

if $\left|\lambda_i^{(k)} - \lambda_i^{(k-1)}\right| \le \varepsilon$ ($\lambda^{(0)}$ can be set arbitrarily) then
begin
$\lambda \leftarrow \lambda_i^{(k)}$; $x_1 \leftarrow x_i^{(k)}$;
end;
else $k \leftarrow k+1$ goto 3

In `step 6` of *Algorithm 2*, for the largest eigenvalue λ_1, it is preferable to take the value λ_i^k averaged over i, and the vector x_1 as the normalized largest eigenvector $x_i^{(k)}$.

Regarding the nature of the convergence of the power method, it can be argued that under the indicated conditions the iterative process is linear, i.e. converges at the rate of a geometric progression, the denominator of which is determined mainly by the value of the ratio $\left|\frac{\lambda_2}{\lambda_1}\right|$. This means that the convergence will be the better and, as a result, the stopping criterion in `step 6` of *Algorithm 2* is the more reliable, the stronger the eigenvalue λ_1 dominates in the spectrum of the matrix B.

Now consider the question of the convergence of this method for a symmetric interval matrix. To do this, we assume that the symmetric interval matrix $\boldsymbol{A}^s$ is divided by formula (11) into boundary matrices $\underline{B}$ and $\overline{B}$. These boundary matrices are real, and the following theorem ensures the convergence of the method for both of these boundary matrices. Therefore, we formulate the theorem, as in *Algorithm 2* for the real matrix B.

Theorem 2. Let the matrix B of dimensions $n \times n$ be a matrix of simple structure (i.e., diagonalizable) and it has a simple dominant eigenvalue, which corresponds to one linear elementary divisor. If the initial vector $x^{(0)}$ does not lie in the linear span $\mathtt{lin}\{v_2, \ldots, v_n\}$ of the eigenvectors of the matrix B, which are not dominant, then the power method converges.

Proof. Under our assumptions about the matrix B, it can be represented as

$$B = VDV^{-1},$$

where $D = \mathtt{diag}\{\lambda_1, \lambda_2, \ldots, \lambda_n\}$ is a diagonal matrix with eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$ along the diagonal, and V is a matrix that performs similarity transformations, and without loss of generality we can assume that λ_1 is the dominant eigenvalue B. Matrix V is composed of eigenvectors v_i of matrix B as columns:

$$V = (v_1, v_2, ..., v_n) =$$

$$= \begin{pmatrix} (v_1)_1 & (v_2)_1 & \ldots & (v_n)_1 \\ (v_1)_2 & (v_2)_2 & \ldots & (v_n)_2 \\ \vdots & \vdots & \ddots & \vdots \\ (v_1)_n & (v_2)_n & \ldots & (v_n)_n \end{pmatrix}$$

where $v_{(i)_j}$ denotes the j-th component of the i-th eigenvector of the matrix B. In this case, we can assume that $\|v_i\|_2 = 1$. Therefore,

$$B^k x^{(0)} = (VDV^{-1})^k x^{(0)} =$$

$$= \underbrace{(VDV^{-1})(VDV^{-1})\cdots(VDV^{-1})}_{k \text{ times}}\, x^{(0)} =$$

$$= VD(V^{-1}V)D(V^{-1}V)\cdots(V^{-1}V)DV^{-1}x^{(0)} =$$

$$= VD^kV^{-1}x^{(0)} = VD^kz = V\begin{pmatrix} \lambda_1^k z_1 \\ \lambda_2^k z_2 \\ \vdots \\ \lambda_n^k z_n \end{pmatrix} =$$

$$= \left(\lambda_1^k z_1\right) V \begin{pmatrix} 1 \\ \left(\lambda_2/\lambda_1\right)^k (z_2/z_1) \\ \vdots \\ \left(\lambda_n/\lambda_1\right)^k (z_n/z_1) \end{pmatrix},$$

where denoted by $z = V^{-1}x^{(0)}$. The necessary condition for the last transformation of this chain is $z_1 \neq 0$, this is satisfied because, under the conditions of the theorem, the vector $x^{(0)} = Vz$ must have a nonzero first component when expanded in a basis of eigenvectors of the matrix B, i.e. columns of matrix V.

Since λ_1 is the dominant eigenvalue of the matrix B, i.e. condition (5) is true, then all quotients

$\lambda_2/\lambda_1,\ \lambda_3/\lambda_1,\ \ldots,\ \lambda_n/\lambda_1$ are less than unity in absolute value, and therefore, at $k \to \infty$, the vector

$$\begin{pmatrix} 1 \\ (\lambda_2/\lambda_1)^k(z_2/z_1) \\ \vdots \\ (\lambda_n/\lambda_1)^k(z_n/z_1) \end{pmatrix} \tag{12}$$

converges to the vector $(1,0,0,...,0)^\top$. Accordingly, the product

$$V \begin{pmatrix} 1 \\ (\lambda_2/\lambda_1)^k(z_2/z_1) \\ \vdots \\ (\lambda_n/\lambda_1)^k(z_n/z_1) \end{pmatrix}$$

converges to the first column of the matrix V, i.e. to the eigenvector corresponding to λ_1. The vector $x^{(k)}$, which differs from $B^k x^{(0)}$ only in its normalization, converges to the eigenvector v_1, and the value $\lambda^{(k)}$ converges to λ_1.

It follows from the above calculations that the rate of convergence of the power method is determined by the ratios $|\lambda_i/\lambda_1|, \quad i = 2,3,\ldots,n$, - the denominators of the geometric progressions that are the elements of the vector (13). In fact, the largest of these ratios is decisive, i.e. $|\lambda_2/\lambda_1|$, depending on how far the modulus of the dominant eigenvalue is separated from the modulus of the rest of the spectrum. The greater this separation, the faster the convergence of the power method. These convergence results ensure the convergence of the power method for both boundary matrices $\underline{B}$ and $\overline{B}$ of the interval symmetric matrix $\boldsymbol{A}^s$.

4.2 Algorithm of the scalar product method for a symmetric interval matrix.

Now we pay attention to the shortcomings of the power method of finding the largest eigenvalue modulo and ways to eliminate them. Here we consider the class of symmetric positive definite interval matrices. It is known that the matrices $A^s \in \boldsymbol{A^s}$ have a positive real spectrum $\lambda_1, \lambda_2, \ldots, \lambda_n$, an orthonormal basis of eigenvectors $x_1, x_2, \ldots, x_n$, and, naturally, are matrices of a simple structure.

In *Algorithm 1*, the calculation at each iteration step of the ratios of all pairs of the corresponding components of the vectors $\underline{x}, \overline{x}$ and $\underline{y}, \overline{y}$, with certain checks at `step 5`, for large values of n requires significant computational costs, although it provides additional information about the largest eigenvalue λ_1. As stated in [26], the value lies between the smallest and largest of these ratios, i.e. there are two-sided estimates of the eigenvalue λ_1 at each iteration.

To simplify the corresponding `step 5` of *Algorithm 1*, we will carry out the following reasoning.

Let the sequence of iterated vectors $\underline{y}^k$ and $\overline{y}^k$ be constructed according to the following formulas

$$\begin{aligned} \underline{y}^{(k)} &= \underline{B}\,\underline{y}^{(k-1)} = \underline{B}^{(k)}\underline{y}^{(0)} = \\ &= \underline{c}_1\underline{\lambda}_1^k\underline{x}_1 + \underline{c}_2\underline{\lambda}_2^k\underline{x}_2 + ... + \underline{c}_n\underline{\lambda}_n^k\underline{x}_n, \\ \overline{y}^{(k)} &= \overline{B}\overline{y}^{(k-1)} = \overline{B}^{(k)}\overline{y}^{(0)} = \\ &= \overline{c}_1\overline{\lambda}_1^k\overline{x}_1 + \overline{c}_2\overline{\lambda}_2^k\overline{x}_2 + ... + \overline{c}_n\overline{\lambda}_n^k\overline{x}_n. \end{aligned} \tag{13}$$

Consider the scalar products $\left\langle \underline{y}^{(k)}, \underline{y}^{(k)} \right\rangle$, $\left\langle \overline{y}^{(k)}, \overline{y}^{(k)} \right\rangle$ and $\left\langle \underline{y}^{(k)}, \underline{y}^{(k-1)} \right\rangle$, $\left\langle \overline{y}^{(k)}, \overline{y}^{(k-1)} \right\rangle$. Performing multiplications of polynomials and taking into account the orthonormality of eigenvectors, i.e. condition $\left\langle \underline{x}_i, \underline{x}_j \right\rangle = \underline{\delta}_{ij}$ and $\langle \overline{x}_i, \overline{x}_j \rangle = \overline{\delta}_{ij}$ at $i,j \in \{1,2,...,n\}$, we have:

$$\begin{aligned} \left\langle \underline{y}^{(k)}, \underline{y}^{(k)} \right\rangle &= \underline{c}_1^2\underline{\lambda}_1^{2k} + \underline{c}_2^2\underline{\lambda}_2^{2k} + \ldots + \underline{c}_n^2\underline{\lambda}_n^{2k}, \\ \left\langle \overline{y}^{(k)}, \overline{y}^{(k)} \right\rangle &= \overline{c}_1^2\overline{\lambda}_1^{2k} + \overline{c}_2^2\overline{\lambda}_2^{2k} + \ldots + \overline{c}_n^2\overline{\lambda}_n^{2k}; \\ \left\langle \underline{y}^{(k)}, \underline{y}^{(k)} \right\rangle &= \underline{c}_1^2\underline{\lambda}_1^{2k-1} + \underline{c}_2^2\underline{\lambda}_2^{2k-1} + \ldots + \underline{c}_n^2\underline{\lambda}_n^{2k-1}, \\ \left\langle \overline{y}^{(k)}, \overline{y}^{(k)} \right\rangle &= \overline{c}_1^2\overline{\lambda}_1^{2k-1} + \overline{c}_2^2\overline{\lambda}_2^{2k-1} + \ldots + \overline{c}_n^2\overline{\lambda}_n^{2k-1}. \end{aligned} \tag{14}$$

Now consider their ratios for the lower and upper bounds, respectively

$$\begin{aligned} \frac{\left\langle \underline{y}^{(k)}, \underline{y}^{(k)} \right\rangle}{\left\langle \underline{y}^{(k)}, \underline{y}^{(k-1)} \right\rangle} &= \underline{\lambda}_1 \times \\ \times \frac{1 + \left(\frac{\underline{c}_2}{\underline{c}_1}\right)^2 \left(\frac{\underline{\lambda}_2}{\underline{\lambda}_1}\right)^{2k} + \ldots + \left(\frac{\underline{c}_n}{\underline{c}_1}\right)^2 \left(\frac{\underline{\lambda}_n}{\underline{\lambda}_1}\right)^{2k}}{1 + \left(\frac{\underline{c}_2}{\underline{c}_1}\right)^2 \left(\frac{\underline{\lambda}_2}{\underline{\lambda}_1}\right)^{2k-1} + \ldots + \left(\frac{\underline{c}_n}{\underline{c}_1}\right)^2 \left(\frac{\underline{\lambda}_n}{\underline{\lambda}_1}\right)^{2k-1}}, \\ \frac{\left\langle \overline{y}^{(k)}, \overline{y}^{(k)} \right\rangle}{\left\langle \overline{y}^{(k)}, \overline{y}^{(k-1)} \right\rangle} &= \overline{\lambda}_1 \times \\ \times \frac{1 + \left(\frac{\overline{c}_2}{\overline{c}_1}\right)^2 \left(\frac{\overline{\lambda}_2}{\overline{\lambda}_1}\right)^{2k} + \ldots + \left(\frac{\overline{c}_n}{\overline{c}_1}\right)^2 \left(\frac{\overline{\lambda}_n}{\overline{\lambda}_1}\right)^{2k}}{1 + \left(\frac{\overline{c}_2}{\overline{c}_1}\right)^2 \left(\frac{\overline{\lambda}_2}{\overline{\lambda}_1}\right)^{2k-1} + \ldots + \left(\frac{\overline{c}_n}{\overline{c}_1}\right)^2 \left(\frac{\overline{\lambda}_n}{\overline{\lambda}_1}\right)^{2k-1}}. \end{aligned} \tag{15}$$

These relations, under the conditions specified above at $k \to \infty$, have a limit for the largest and smallest values of eigenvalues, and the speed to the limit will be greater than in *Algorithm 1*, since $O\left(\left|\frac{\underline{\lambda}_2}{\underline{\lambda}_1}\right|^{2k}\right)$ instead of $O\left(\left|\frac{\underline{\lambda}_2}{\underline{\lambda}_1}\right|^{k}\right)$, and accordingly $O\left(\left|\frac{\overline{\lambda}_2}{\overline{\lambda}_1}\right|^{2k}\right)$ instead of $O\left(\left|\frac{\overline{\lambda}_2}{\overline{\lambda}_1}\right|^{k}\right)$.

Thus, we present a modification of the power method, the so-called scalar product method (SP-method) for symmetric interval matrices, in the form of Algorithm 3.

Algorithm 3. SP-method for Symmetric Interval Matrix

Input: $\boldsymbol{A}^s = \left[\underline{A}, \overline{A}\right] \in \mathcal{IR}^{n\times n}$, given accuracy ε, $\underline{\lambda}^{(0)}$ and $\overline{\lambda}^{(0)}$ for initial comparison (e.g. 0), initial approximation vector $\boldsymbol{y}^{(0)} = \underline{y}^{(0)} = \overline{y}^{(0)} \neq 0$.

Output: eigenvalue interval $\boldsymbol{\lambda}_1 \in \mathcal{IR}$ and its corresponding normalized interval eigenvector $\boldsymbol{x}_1 \in \mathcal{IR}^n$.

1: Compute: $A_c \leftarrow (\underline{A} + \overline{A})/2, \quad A_\triangle \leftarrow (\overline{A} - \underline{A})/2$; eigenvalues $\lambda_i(A_c)$, eigenvectors $x_1^i(A_c), \quad x_2^i(A_c), \quad \ldots, \quad x_n^i(A_c)$, diagonal matrix $D^i \leftarrow \mathtt{diag}\left(\mathtt{sign}\left(x_1^i(A_c)\right), \ldots, \mathtt{sign}\left(x_n^i(A_c)\right)\right)$;
$\underline{B} \leftarrow A_c - D^i A_\triangle D^i, \quad \overline{B} \leftarrow A_c + D^i A_\triangle D^i$, $i = 1,2,\ldots,n$;
$\underline{s}^{(0)} \leftarrow \left\langle \underline{y}^{(0)}, \underline{y}^{(0)} \right\rangle, \quad \overline{s}^{(0)} \leftarrow \left\langle \overline{y}^{(0)}, \overline{y}^{(0)} \right\rangle$;
$\left\|\underline{y}^{(0)}\right\|_2 \leftarrow \sqrt{\underline{s}^{(0)}}, \quad \left\|\overline{y}^{(0)}\right\|_2 \leftarrow \sqrt{\overline{s}^{(0)}}$;
$\underline{x}^{(0)} \leftarrow \underline{y}^{(0)} / \left\|\underline{y}^{(0)}\right\|_2, \quad \overline{x}^{(0)} \leftarrow \overline{y}^{(0)} / \left\|\overline{y}^{(0)}\right\|_2$;

2: $k \leftarrow 1$;

3: $\underline{y}^{(k)} \leftarrow \underline{B}\,\underline{x}^{(k-1)}$;

4: $\underline{s}^{(k)} \leftarrow \left\langle \underline{y}^{(k)}, \underline{y}^{(k)} \right\rangle, \quad \underline{t}^{(k)} \leftarrow \left\langle \underline{y}^{(k)}, \underline{x}^{(k-1)} \right\rangle;$
$\left\| \underline{y}^{(k)} \right\|_2 \leftarrow \sqrt{\underline{s}^{(k)}}, \quad \underline{x}^{(k)} \leftarrow \underline{y}^{(k)} / \|\underline{y}^{(k)}\|_2;$
$\underline{\lambda}^{(k)} \leftarrow \underline{s}^{(k)} / \underline{t}^{(k)};$

5: if $\left| \underline{\lambda}^{(k)} - \underline{\lambda}^{(k-1)} \right| < \varepsilon$ then
begin
$\underline{\lambda} \leftarrow \underline{\lambda}^{(k)}; \quad \underline{x} \leftarrow \underline{x}^{(k)};$
end;
else
begin
$k \leftarrow k + 1;$ goto 3
end;

6: $l \leftarrow 1$;

7: $\overline{y}^{(l)} \leftarrow \overline{B}\,\overline{x}^{(l-1)}$;

8: $\overline{s}^{(l)} \leftarrow \left\langle \overline{y}^{(l)}, \overline{y}^{(l)} \right\rangle; \quad \overline{t}^{(l)} \leftarrow \left\langle \overline{y}^{(l)}, \overline{x}^{(l-1)} \right\rangle;$
$\left\| \overline{y}^{(l)} \right\|_2 \leftarrow \sqrt{\overline{s}^{(l)}}, \quad \overline{x}^{(l)} \leftarrow \overline{y}^{(l)} / \|\overline{y}^{(l)}\|_2;$
$\overline{\lambda}^{(l)} \leftarrow \overline{s}^{(l)} / \overline{t}^{(l)};$

9: if $\left| \overline{\lambda}^{(l)} - \overline{\lambda}^{(l-1)} \right| < \varepsilon$ then
begin
$\overline{\lambda} \leftarrow \overline{\lambda}^{(l)}; \quad \overline{x} \leftarrow \overline{x}^{(l)};$
end;
else
begin
$l \leftarrow l + 1;$ goto 7
end;

10: $\boldsymbol{\lambda}_1 \leftarrow \left[\underline{\lambda}, \overline{\lambda}\right]; \quad \boldsymbol{x}_1 \leftarrow [\underline{x}, \overline{x}].$

Remark 1. *Algorithm 3* allows faster (i.e., in fewer iterations) than *Algorithm 1* to find with the required accuracy an interval containing the largest eigenvalue of a symmetric matrix, but the accuracy of the approximate $\boldsymbol{x_1} :\approx [\underline{x}^{(k)}, \overline{x}^{(k)}]$ equality for the corresponding eigenvector may be insufficient.

Remark 2. Obviously, in the scalar product method (Algorithm 3), instead of relation (15), which tends to $\boldsymbol{\lambda}_1$ as $k \to \infty$, we can just as well take the relation

$$\frac{\langle \underline{y}^{(k+1)}, \underline{y}^{(k)} \rangle}{\langle \underline{y}^{(k)}, \underline{y}^{(k)} \rangle} = \frac{\langle \underline{B}\underline{y}^{(k)}, \underline{y}^{(k)} \rangle}{\langle \underline{y}^{(k)}, \underline{y}^{(k)} \rangle} = \rho(\underline{y}^{(k)});$$

$$\frac{\langle \overline{y}^{(k+1)}, \overline{y}^{(k)} \rangle}{\langle \overline{y}^{(k)}, \overline{y}^{(k)} \rangle} = \frac{\langle \overline{B}\overline{y}^{(k)}, \overline{y}^{(k)} \rangle}{\langle \overline{y}^{(k)}, \overline{y}^{(k)} \rangle} = \rho(\overline{y}^{(k)}).$$

The latter is nothing but the Rayleigh relation, from here we can move on to a new method, the so-called "Method of partial Rayleigh" for interval matrices.

5 Numerical results

For the purpose of comparing numerical results, in this section we present some examples and numerical results illustrating the properties of the proposed algorithms. The numerical experiment was carried out on a PC Intel(R) CoreTM i3-3217U, CPU 1.83 GHz, 4 GB DDR3 Memory, the source code was written in Embarcadero Dev-C++ v.6.3. We also used the "octave-interval v.3.2.0-3_amd64" software package for interval calculations, which runs on the GNU Octave v.4.2.1 system [27]. This package complies with the IEEE Std 1788-2015 standard for interval arithmetic.

Recall that in this paper we are investigating a partial eigenvalue problem and therefore we are only interested in finding the dominating interval, i.e. containing the largest eigenvalue λ_1 for the matrix $A^s \in \boldsymbol{A}^s$. In our computational experiments, the corresponding eigenvectors are also calculated, but here we will restrict ourselves only to the representation of eigenvalues.

Example 1. Consider the following symmetric interval matrix from M. Hladik et al. [28]

$$\boldsymbol{A}^s = \begin{pmatrix} 1 & 2 & [1,5] \\ 2 & 1 & 1 \\ [1,5] & 1 & 1 \end{pmatrix} \tag{16}$$

In [28, Algorithm 3], intervals of eigenvalues of matrix (16) were found using an outer approximation:

$$\begin{gathered} \boldsymbol{\lambda}_1(\boldsymbol{A}^s) = [3.5230,\ 6.7843], \\ \boldsymbol{\lambda}_2(\boldsymbol{A}^s) = [0.0000,\ 1.0519], \\ \boldsymbol{\lambda}_3(\boldsymbol{A}^s) = [-4.1214,\ -0.2019]. \end{gathered}$$

Here $\boldsymbol{\lambda}_1(\boldsymbol{A}^s) = [3.5230,\ 6.7843]$ (with a width $\mathtt{wid}\left(\left[\underline{\lambda}_1, \overline{\lambda}_1\right]\right) = 3.2613$) is the dominant interval and we can compare our results with this value.

Table 1 shows the results of a numerical experiment for the developed algorithms. Here, *Algorithm 1* and *Algorithm 2* give exactly the same results, as noted above, these algorithms are equivalent. Therefore, we compare their results with the results of *Algorithm 3*. It should be noted that the speed of convergence of the iterative process depends on the choice of the initial vector, since we used all possible values of the initial approximation vectors in the computational experiment.

In problems of outer interval estimation, the desired solution is called *optimal* if the found interval solution has narrower widths. Table 1 shows that the number of iterations (where the parameters k and l mean the number of iterations for calculating the left and right boundaries of the eigenvalues, respectively) in Algorithm 3 is less than in Algorithm 1 (or 2).

In the case of the initial approximation $(1; 0; 1)$, the SP-method (Algorithm 3) gives the optimal solution $[3.770491838, 6.785255908]$ $(\mathtt{wid}\left(\left[\underline{\lambda}_1, \overline{\lambda}_1\right]\right) = 3.014764070)$ than the others. And the power method in $(0; 0; 1)$ initial approximation shows a good result, like [3.732049942, 6.778576374] $(\mathtt{wid}\left(\left[\underline{\lambda}_1, \overline{\lambda}_1\right]\right) = 3.046526432)$.

Example 2. Now consider a tridiagonal 4×4 symmetric

Table 1. Numerical results for the interval matrix (16)

Initial approx. vector	Algorithm 1 or Algorithm 2 (Power method)					Algorithm 3 (SP-Method)				
$y^{(0)}$	k	$\underline{\lambda}_1$	l	$\overline{\lambda}_1$	$\mathtt{wid}\left([\underline{\lambda}_1,\ \overline{\lambda}_1]\right)$	k	$\underline{\lambda}_1$	l	$\overline{\lambda}_1$	$\mathtt{wid}\left([\underline{\lambda}_1,\ \overline{\lambda}_1]\right)$
$(1;1;1)$	3	3.732049942	3	6.783113003	3.051063061	2	3.732026338	2	6.793815135	3.061788797
$(1;1;0)$	3	3.732037783	6	6.779861450	3.047823667	3	3.732048988	2	7.588236331	3.856187343
$(1;0;1)$	4	3.731780052	3	6.783932209	3.052152157	2	3.770491838	3	6.785255908	3.014764070
$(1;0;0)$	5	3.731992245	8	6.782109737	3.050117492	2	3.849999904	2	10.416666984	6.566667080
$(0;1;1)$	4	3.731780052	7	6.780661106	3.048881054	2	3.770491838	2	8.758620262	4.988128424
$(0;1;0)$	5	3.731992245	6	6.781606197	3.049613952	2	3.849999904	3	6.952380657	3.102380753
$(0;0;1)$	4	3.732049942	7	6.778576374	3.046526432	3	3.732026338	2	10.272727012	6.540700674

Table 2. Numerical results for the interval matrix (17)

Initial approx. vector	Algorithm 1 or Algorithm 2 (Power method)					Algorithm 3 (SP-Method)				
$y^{(0)}$	k	$\underline{\lambda}_1$	l	$\overline{\lambda}_1$	$\mathtt{wid}\left([\underline{\lambda}_1,\ \overline{\lambda}_1]\right)$	k	$\underline{\lambda}_1$	l	$\overline{\lambda}_1$	$\mathtt{wid}\left([\underline{\lambda}_1,\ \overline{\lambda}_1]\right)$
$(1;1;1;1)$	14	12560.83398	14	12720.22559	159.39161	14	12560.83496	14	12720.22558	159.39062
$(1;1;1;0)$	11	12560.83594	11	12720.22461	159.38867	11	12560.83496	11	12720.22558	159.39062
$(1;1;0;1)$	10	12560.83398	11	12720.22656	159.39258	10	12560.83398	10	12720.22460	159.39062
$(1;1;0;0)$	14	12560.83398	14	12720.22363	159.38965	14	12560.83496	14	12720.22460	159.38964
$(1;0;1;1)$	22	12560.83594	22	12720.22559	159.38965	37	12560.83789	37	12720.22851	159.39062
$(1;0;1;0)$	13	12560.83594	13	12720.22461	159.38867	13	12560.83594	13	12720.22461	159.38867
$(1;0;0;1)$	14	12560.83496	14	12720.22461	159.38965	14	12560.83496	14	12720.22461	159.38965
$(1;0;0;0)$	16	12560.83301	17	12720.22559	159.39258	16	12560.83594	16	12720.22266	159.38672
$(0;1;1;1)$	11	12560.83594	11	12720.22656	159.39062	11	12560.83496	10	12720.22168	159.38672
$(0;1;1;0)$	14	12560.83691	14	12720.22559	159.38868	13	12560.83398	13	12720.22168	159.38770
$(0;1;0;1)$	10	12560.83398	10	12720.22266	159.38868	10	12560.83398	10	12720.22461	159.39063
$(0;1;0;0)$	15	12560.83594	15	12720.22363	159.38769	15	12560.83496	15	12720.22558	159.39062
$(0;0;1;1)$	18	12560.83594	18	12720.22559	159.38965	17	12560.83398	17	12720.22363	159.38965
$(0;0;1;0)$	12	12560.83594	12	12720.22559	159.38965	11	12560.83301	11	12720.22363	159.39062
$(0;0;0;1)$	13	12560.83496	13	12720.22559	159.39063	13	12560.83594	13	12720.22558	159.38964

interval matrix given by Z. Qui et al. [29] (see also [12, 28]):

$$\boldsymbol{A}^s = \begin{pmatrix} a_{11} & a_{12} & 0 & 0 \\ a_{21} & a_{22} & a_{23} & 0 \\ 0 & a_{32} & a_{33} & a_{34} \\ 0 & 0 & a_{43} & a_{44} \end{pmatrix}, \qquad (17)$$

where $a_{11} = [2975, 3025]$, $a_{12} = [-2015, -1985]$, $a_{21} = [-2015, -1985]$, $a_{22} = [4965, 5035]$, $a_{23} = [-3020, -2980]$, $a_{32} = [-3020, -2980]$, $a_{33} = [6955, 7045]$, $a_{34} = [-4025, -3975]$, $a_{43} = [-4025, -3975]$, $a_{44} = [8945, 9055]$.

Matrix (17) was taken in [29] to solve the problem of an automobile suspension. The computational experiments carried out by the authors show that the method of interval perturbations gives a dominant interval $[12588.29000, 12692.77000]$ with a width $\mathtt{wid}\left([\underline{\lambda}_1, \overline{\lambda}_1]\right) = 104.48000$, and the Deif method $[12560.83772, 12720.22727]$ with a width $\mathtt{wid}\left([\underline{\lambda}_1, \overline{\lambda}_1]\right) = 159.38955$. In works [12, 28, 31, 32], the same matrix is also considered, and there, the method proposed by the authors, the so-called "Filtering Method", an interval solution $[12560.81290, 12720.24720]$ with a width $\mathtt{wid}\left([\underline{\lambda}_1, \overline{\lambda}_1]\right) = 159.43430$ was obtained in 13 iterations.

Table 2 shows the numerical results obtained by our algorithms for matrix (17).

Table 2 shows that the results obtained by the proposed method are in good agreement with the results obtained by other researchers. *Algorithm 1* (or *2*) with initial approximation vector $(1;1;1;0)$ gives the solution $[12560.83594, 12720.22461]$ with width $\mathtt{wid}\left([\underline{\lambda}_1, \overline{\lambda}_1]\right) = 159.38867$ and number of iterations $k = l = 11$, while *Algorithm 3* with initial approximation $(0;1;1;1)$ gives $[12560.83496, 12720.22168]$ with width $\mathtt{wid}\left([\underline{\lambda}_1, \overline{\lambda}_1]\right) = 159.38672$ and with iteration numbers $k = 11$ and $l = 10$. These results are even better than those obtained by the Deif method [29] and the method of M. Hladik et al. [12, 28, 32].

In *example 2*, except for the case with the initial approximation $(1;0;1;1)$, the number of iterations in *Algorithm 3* is less than or equal to that in *Algorithm 1* (or *2*). This shows that in solving the partial eigenvalue problem for real symmetric interval matrices, it is preferable to choose *Algorithm 3*.

6 Conclusions

The main goal of this study was to study iterative methods for solving a partial eigenvalue problem for interval matrices.

In doing so, we chose the power method and the scalar product method, which is often used to solve the eigenvalue problem. A comparative analysis of the obtained numerical results shows that the method of scalar products (*Algorithm 3*) is superior to the power method (*Algorithm 1* or *2*) in terms of convergence speed, i.e. the number of iterations and the width of the interval solution. We are interested in interval solutions with a smaller width, since in this paper we study the interval problem of external estimation.

REFERENCES

[1] O.Kosheleva, S.P.Shary, G.Xiang, R.Zapatin. Beyond Traditional Probabilistic Data Processing Techniques: Interval, Fuzzy etc. Methods and Their Applications, Studies in Computational Intelligence 835, Springer Nature Switzerland AG 2020. https://doi.org/10.1007/978 − 3 − 030 − 31041 − 7_1.

[2] H.T. Nguynen, V.Kreynovich, B.Wu, G.Xiang. Computing Statistics under Interval and Fuzzy Uncertainty: Applications to Computer Science and Enjineering, Studies in Computational Intelligence 393, Springer-Verlag Berlin Heidelberg 2012.

[3] Shary S.P. Finite-dimensional interval analysis. Federal Research Center for Information and Computing Technologies, NSU. - Novosibirsk: Publishing house "XYZ", 645 p., 2020. http://www.nsc.ru/interval/Library/InteBooks/SharyBook.pdf. (In Russ.)

[4] Ibragimov, A.A., Bazarov, M.B., Shokin, Yu.I., Yuldashev, Z.KH. Mathematical modeling by interval methods. -Tashkent: Publishing house "Fan", 160 p., 2013.

[5] B.R.Barmish and C.L.Demacro. A new method for improvement of robustness bounds for linear state equations, Proc. Conf. Information Sciences & Systems, Princeton Univ., 1986.

[6] D.Hudak. Ein Eigenwertproblem fur Interval-Matrizen, J. Appl. Math. Mech. (ZAMM), 64, 503-505, 1984.

[7] Y.T.Juang, T.S.Kuo and C.F.Hsu. Stability robustness for state-space models, Proc. IEEE 25th Conf. Decision and Control, Athens, Greece, 745-750, 1986.

[8] L.Qiu and E.J. Davison. New perturbation bounds for the robust stability of linear state space models, Proc. IEEE 25th Conf. Decision and Control, Athens, Greece, 751-755, 1986.

[9] A.Deif. The interval eigenvalue problem, Z.Angew.Math.Mech., 71, pp.61-64, 1991.

[10] J.Rohn. Real eigenvalue of an interval matrix with rank one radius. Z.Angew.Math.Mech., 70, pp.562-563, 1990.

[11] J.Rohn. Bounds on eigenvalues of interval matrices. Z.Angew. Math.Mech., 78 (Suppl. 3): p.1049-1050, 1998.

[12] M. Hladík. A filtering method for the interval eigenvalue problem. Appl. Math. Comput., 217: 5236-5242, 2011. doi:10.1016/j.amc.2010.09.066.

[13] M. Hladík. Eigenvalues of real and complex interval matrices. Appl. Math. Comput., 219(10): 5584-5591, 2013.

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

[15] Kolev L.V. Outer interval solution of the eigenvalue problem under general form parametric dependencies. Reliable Computing Vol.12, Issue 2: 121-140, 2006.

[16] G. Mayer. Interval Analysis and Automatic Result Verification. De Gruyter Studies in Mathematics 65. De Gruyter, Berlin, 2017.

[17] Ipsen, Ilse and Rebecca M. Wills. 7-th IMACS International Symposium on Iterative Methods in Scientific Computing. 5-8 May, 2005.

[18] Pankaj Gupta, Ashish Goel, Jimmy Lin, Aneesh Sharma, Dong Wang, and Reza Bosagh Zadeh WTF: The who-to-follow system at Twitter, Proceedings of the 22nd international conference on World Wide Web.

[19] R.B.Kearfott, M.T.Nakao, A.Neumaier, S.M.Rump, S.P.Shary, P. Hentenryck. Standardized notation in interval analysis. 2005. http://www.ict.nsc.ru/interval/InteNotation.ps.

[20] G. Mayer. Enclosures for eigenvalues and eigenvectors, in: L. Atanassova, J. Herzberger (Eds.), Computer Arithmetic and Enclosure Methods, Elsevier, Amsterdam, pp. 49-67, 1992.

[21] G. Mayer. Result verification for eigenvectors and eigenvalues, in: J. Herzberger (Ed.), Topics in Validated Computations, Elsevier, Amsterdam, pp. 209-276, 1994.

[22] M. Hladík. Complexity issues for the symmetric interval eigenvalue problem. Open Math., 13, pp. 157-164, 2015.

[23] J. Rohn. Checking positive definiteness or stability of symmetric interval matrices is NP-hard. Comment. Math. Univ. Carolin. 35, pp. 795-797, 1994.

[24] S. Poljak and J. Rohn. Checking robust nonsingularity is NP-hard. Math. Control, Signals Syst. 6, pp. 1-9, 1993.

[25] V.I. Erokhin. On conditions of nonsingularity of interval matrices. Electronic journal "Investigated in Russia". - pp. 2488-2508, - 2003. https://cyberleninka.ru/article/n/ob-usloviyah-nevyrozhdennosti-intervalnyh-matrits.

[26] L.Collatz, J.Albrecht. Problems in applied mathematics [Translated from German by S.N. Kiro, edited by Kh.D. Ikramov]. -Moscow: "MIR", -166 p., 1978.

[27] Oliver Heimlich. Octave-interval real-valued interval arithmetic for Octave. Online available from https://octave.sourceforge.io/interval/index.html.

[28] M. Hladík, D. Daney, E. Tsigaridas. Characterizing and approximating eigenvalue sets of symmetric interval matrices. Comp. and Math. with Appl. 62, pp. 3152-3163, 2011.

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

[30] N. S. Kaleyski. Eigenvalues of symmetric interval matrices. Bachelor Thesis, Department of Applied Mathematics, Charles University, Prague (2014), p. 67.

[31] M. Hladík, D. Daney, E. P. Tsigaridas. A filtering method for the interval eigenvalue problem. [Research Report] RR-7057, INRIA. 2009, pp.14. inria-00422966. https://hal.inria.fr/inria-00422966/document

[32] M. Hladík, D. Daney, E. P. Tsigaridas. Bounds on eigenvalues and singular values of interval matrices. [Research Report] RR-1234, INRIA. 2008, pp.18. inria-00370603. https://hal.inria.fr/inria-00370603/document