

Bounding error of calculating the matrix functions

Marzieh Dehghani Madiseh*

Department of Mathematics, Faculty of Mathematical Sciences and Computer, Shahid Chamran University of Ahvaz, Ahvaz, Iran.

Abstract

Matrix functions play important roles in various branches of science and engineering. In numerical computations and physical measurements there are several sources of error which significantly affect the main results obtained from solving the problems. This effect also influences the matrix computations. In this paper, we propose some approaches to enclose the matrix functions. We then present some analytical arguments to ensure that the obtained enclosures contain the exact result. Numerical experiments are given to illustrate the performance and effectiveness of the proposed approaches.

Keywords. Matrix function, Floating point arithmetic, Interval arithmetic.

2010 Mathematics Subject Classification. 65F60, 65G30.

1. INTRODUCTION

Computing matrix functions is a frequently occurring problem in science and engineering such as differential equations, Markov models, control theory, nonsymmetric eigenvalue problem, exponential integrators and Nuclear Magnetic Resonance, see [3, 4, 23]. Up to now, a large number of papers presented several methods for computing matrix functions [7, 9, 21, 22, 24, 31, 41]. But on the other hand, finite number representation in floating point arithmetic, inevitably causes some errors in computations. This motivated us to propose some approaches for bounding the error of evaluating matrix functions.

Interval arithmetic is an arithmetic defined on the set of intervals, rather than the set of real numbers. As said in [28], a form of interval arithmetic first appeared in 1924 [5] and 1931 [44], then later in [42]. Modern development of interval arithmetic began with R. E. Moore's dissertation [32]. Since then, many research articles and numerous books have appeared on the subject. Also, there is an increasing amount of software package for interval computations. For a preliminary introduction, we refer the interested reader to [2, 13, 14, 20, 27, 34].

1.1. Interval arithmetic as a useful tool in computing a matrix function. Here, we explain the motion of applying interval computations for computing a matrix function by three simple examples.

Sample 1. Let

$$A = \begin{pmatrix} a & 0 \\ 0 & 1 \end{pmatrix},$$

in which a is a desirable parameter. Let $f(x) = x^{75}$. So by diagonal structure of A , we have

$$f(A) = \begin{pmatrix} f(a) & 0 \\ 0 & f(1) \end{pmatrix}.$$

Received: 22 March 2020 ; Accepted: 06 December 2020.

*Corresponding author. Email: m.dehghani@scu.ac.ir.

Now, if we put $a = 1 - 10^{-21}$, performing 10-place arithmetic, we obtain $f(a) = 1$. Using 20-place arithmetic, we obtain the same result, see [33]. So

$$f(A) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

However the exact value of $f(a)$ satisfies $f(a) < 10^{-10}$, i.e., we must have

$$f(A) = \begin{pmatrix} b & 0 \\ 0 & 1 \end{pmatrix},$$

in which $b < 10^{-10}$.

Sample 2. Let

$$A = \begin{pmatrix} \pi & 0 \\ 0 & \pi/2 \end{pmatrix},$$

and $f(x) = \sin x$. Again since A is a diagonal matrix, we obtain

$$f(A) = \begin{pmatrix} f(\pi) & 0 \\ 0 & f(\pi/2) \end{pmatrix}.$$

Performing the following commands in Matlab

```
>> b = sin(pi),
>> c = sin(pi/2),
```

responds, respectively, with

$$b = 1.2246e - 16,$$

$$c = 1.$$

So, by floating point arithmetic, we will have the following result for $f(A)$

$$f(A) = \begin{pmatrix} 1.2246e - 16 & 0 \\ 0 & 0 \end{pmatrix}.$$

while, the exact result is

$$f(A) = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}.$$

Sample 3. Consider the following two variable function

$$f(x, y) = 333.75y^6 + x^2(11x^2y^2 - y^6 - 121y^4 - 2) + 5.5y^8 + x/(2y),$$

and

$$A = \begin{pmatrix} a & 0 \\ 0 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} b & 0 \\ 0 & 1 \end{pmatrix},$$

with $a = 77617.0$ and $b = 33096.0$, for noninterval case of this example see [33]. In this case, we have a two variable matrix function. Matrices A and B are diagonal and B is invertible, so we obtain

$$f(A, B) = \begin{pmatrix} f(a, b) & 0 \\ 0 & f(0, 1) \end{pmatrix}.$$

Now for computing $f(a, b)$, as said in [33], computing powers by successive multiplications on an IBM 370 system using single, double, and extended precision (approximately 7, 16, and 33 decimal digits, respectively), Rump obtained the following results:



single precision	$f(a, b) = 1.17260361\dots$
double precision	$f(a, b) = 1.17260394005317847\dots$
extended precision	$f(a, b) = 1.17260394005317863185\dots$

We might be tempted to conclude that $f(a, b)$ is close to 1.172603. On the other hand $f(0, 1) = 333.75 + 5.5 = 339.25$, so we will have

$$f(A, B) = \begin{pmatrix} 1.172603 & 0 \\ 0 & 339.25 \end{pmatrix}.$$

But it is not true. In fact the exact result for $f(a, b)$ is

$$f(a, b) = -0.827396\dots,$$

and so we should have

$$f(A, B) = \begin{pmatrix} -0.827396\dots & 0 \\ 0 & 339.25 \end{pmatrix}.$$

These three simple examples inspire us to use interval arithmetic as a useful tool to bound the errors arising from floating point arithmetic. If instead of simply computing a numerical approximation using limited-precision arithmetic and then worrying about the accuracy of the results, we proceed in interval arithmetic, i.e., considering intervals which contain the real parameters, we will be sure that the final result contains the exact result.

1.2. Notations. We use boldface letters to denote interval quantities and ordinary letters are used for real quantities. The left and right endpoints of an interval number $\mathbf{x}$ are denoted by $\underline{x}$ and $\bar{x}$, respectively, i.e., $\mathbf{x} = [\underline{x}, \bar{x}]$. The set of real intervals is denoted by $\mathbb{IR}$. By $\mathbb{IR}^{m \times n}$, we denote the set of m -by- n interval matrices. The width of an interval $\mathbf{x} = [\underline{x}, \bar{x}]$ is denoted by $w(\mathbf{x})$ and is defined as $w(\mathbf{x}) = \bar{x} - \underline{x}$, and the width of an interval matrix $\mathbf{A} = (\mathbf{A}_{ij})$ is given by $w(\mathbf{A}) = \max_{i,j} w(\mathbf{A}_{ij})$. For a matrix $A \in \mathbb{R}^{n \times n}$, the eigenvalues of A are shown by $\lambda_1, \dots, \lambda_n$.

If $\mathbf{x} = [\underline{x}, \bar{x}]$ and $\mathbf{y} = [\underline{y}, \bar{y}]$, then the four elementary operations $*$ $\in$ $\{+, -, \times, /\}$ for interval arithmetic obey

$$\mathbf{x} * \mathbf{y} = \{x * y : x \in \mathbf{x} \text{ and } y \in \mathbf{y}\}. \quad (1.1)$$

This paper is organized as follows: In section 2, we present a list of methods for evaluating interval matrix function $f(\mathbf{A})$ and theoretical theorems are given. Also some suggestions and development of the presented methods are briefly discussed in this section. Section 3 contains some numerical experiments. Some applications of this subject in various problems are presented in section 4. Finally, in section 5, we complete the paper with brief concluding remarks.

2. MAIN RESULTS

2.1. Our Approaches. Let A be an n -by- n real matrix and f be an analytic function in a certain domain of the complex plane containing the spectrum of A . We want to enclose the matrix function $f(A)$. For this purpose, we consider an interval matrix $\mathbf{A}$ such that $A \in \mathbf{A}$ and then perform the following methods on $\mathbf{A}$. In the following $\lambda_1, \dots, \lambda_n$ are the eigenvalues of A and I is the n -by- n identity matrix.

1) Interval power series method

If function f has a Taylor series representation on an open disk containing the eigenvalues of A , i.e.,

$$f(x) = \sum_{k=1}^{\infty} \frac{f^{(k)}(x_0)}{k!} (x - x_0)^k,$$

then we define the interval matrix function $f(\mathbf{A})$ by

$$f(\mathbf{A}) = \sum_{k=1}^{\infty} \frac{f^{(k)}(x_0)}{k!} (\mathbf{A} - x_0 I)^k.$$

Theorem 2.1. *The interval power series method requires $\mathcal{O}(n^3 k)$ arithmetic operations per index k of summation.*



Proof. Evaluating $\mathbf{A} - x_0 I$ needs $\mathcal{O}(n)$ operations since subtraction only affects diagonal entries of $\mathbf{A}$. The cost of computing matrix powers $(\mathbf{A} - x_0 I)^k$ is $\mathcal{O}(n^3 k)$. Other computational parts of the summand contain number-number or number-matrix operations including division and multiplication which needs $\mathcal{O}(n^2)$ arithmetic operations. Overall, we conclude that the interval power series method requires $\mathcal{O}(n^3 k)$ arithmetic operations per index k . $\square$

2) Interval Cauchy integral method

Similar to Cauchy's integral definition for a matrix function [16, 37], for a function analytic at the eigenvalues $\lambda_1, \dots, \lambda_n$ of A , we define interval matrix function $f(\mathbf{A})$ by

$$f(\mathbf{A}) = \frac{1}{2\pi i} \int f(z)(zI - \mathbf{A})^{-1} dz,$$

in which $(zI - \mathbf{A})^{-1}$ is hull of the set $\{(zI - B)^{-1} : B \in \mathbf{A}\}$ and integral is taken componentwise over a set of admissible closed paths enclosing each of the distinct characteristic roots of A .

3) Interval Sylvester, Buchheim method

Corresponding to the Sylvester definition of the matrix function $f(A)$, see [43], for determining interval matrix $f(\mathbf{A})$, we propose the following method

$$f(\mathbf{A}) = \sum_{j=1}^n \prod_{i \neq j} \frac{\mathbf{A} - \lambda_i I}{\lambda_j - \lambda_i} f(\lambda_j),$$

where $\{\lambda_1, \dots, \lambda_n\}$ are distinct eigenvalues of A .

If the characteristic roots of A are not necessarily distinct, then analogy with Buchheim's definition for the real case, we define

$$f(\mathbf{A}) = \sum_{i=1}^t \prod_{i \neq j} (\mathbf{A} - \lambda_i I)^{s_i} \sum_{k=0}^{s_j-1} \frac{d^k}{dz^k} \left[\frac{f(z)}{\prod_{h \neq j} (z - z_h)^{s_h}} \right]_{z=\lambda_j} (\mathbf{A} - \lambda_j I)^k,$$

where s_i is the multiplicity of λ_i as a root of the minimum polynomial of A .

Theorem 2.2. *The interval Sylvester method requires $\mathcal{O}(n^4)$ arithmetic operations.*

Proof. Computing the expression under multiplication symbol needs $\mathcal{O}(n^2)$ arithmetic operations since it involves subtraction, division and multiplication between numbers and matrix $\mathbf{A}$. Multiplying $(n-1)$ matrices of size n costs $\mathcal{O}(n^4)$. Finally the operation $\sum_{j=1}^n$ adds n matrices of size n and so needs $\mathcal{O}(n^3)$ operations. So, gathering all computational costs yields that the interval Sylvester method requires $\mathcal{O}(n^4)$ arithmetic operations. $\square$

4) Interval Richter's method

Using the idea of [36] (Richter method), we propose the following formula for evaluating $f(\mathbf{A})$

$$f(\mathbf{A}) = \sum_{j=1}^t R_j \sum_{k=0}^{s_j-1} \frac{1}{k!} f^{(k)}(\lambda_j) (\mathbf{A} - \lambda_j I)^k,$$

where

$$R_j = w_j(\mathbf{A}) \prod_{i \neq j} (\mathbf{A} - \lambda_i I)^{s_i},$$

and the polynomial $w_j(z)$ is a polynomial satisfying

$$w_j(z) \prod_{i \neq j} (z - \lambda_i)^{s_i} \equiv 1, \quad \text{mod}(z - \lambda_j)^{s_j},$$

therein s_j is the multiplicity of λ_j as a root of the minimum polynomial of A .

5) Interval Newton divided difference method and its extension



Another approach for enclosing the matrix function $f(A)$ is

$$f(\mathbf{A}) = \sum_{i=1}^n d_i \prod_{j=1}^{i-1} (\mathbf{A} - \lambda_j I),$$

which is an extension of the divided difference method [8], therein the coefficients $d_1, \dots, d_n$ are defined in [8].

The above definition is applicable only when the eigenvalues of A are distinct. For the case when the eigenvalues are not distinct, we propose the following definition for determining $f(\mathbf{A})$

$$f(\mathbf{A}) = \sum_{i=1}^n k[\lambda_1, \dots, \lambda_i] \prod_{j=1}^{i-1} (\mathbf{A} - \lambda_j I),$$

where $k[\lambda_1, \dots, \lambda_i]$ is defined in [8].

Theorem 2.3. *The interval Newton divided difference method requires $\mathcal{O}(n^4)$ arithmetic operations.*

Proof. The argument is nearly similar to the proof of Theorem 2.2. The highest computational cost is related to the case that the upper limit of multiplication takes amount of $(n-1)$ which occurs when $i = n$. In this case, we have to multiply $(n-1)$ matrices of size n which needs $\mathcal{O}(n^4)$ arithmetic operations. $\square$

2.2. Theoretical consideration. In this subsection, we prove for all the above methods, we have $f(A) \in f(\mathbf{A})$.

Lemma 2.4. [33] *Let $*$ $\in \{+, -, \times, /\}$. If $\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}$, and $\mathbf{d}$ are intervals such that*

$$\mathbf{a} \subseteq \mathbf{c}, \quad \text{and} \quad \mathbf{b} \subseteq \mathbf{d},$$

then

$$\mathbf{a} * \mathbf{b} \subseteq \mathbf{c} * \mathbf{d}.$$

Lemma 2.5. *Let $*$ $\in \{+, -, \times\}$. If $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, and $\mathbf{D}$ are interval matrices such that*

$$\mathbf{A} \subseteq \mathbf{C}, \quad \text{and} \quad \mathbf{B} \subseteq \mathbf{D},$$

then

$$\mathbf{A} * \mathbf{B} \subseteq \mathbf{C} * \mathbf{D}.$$

Proof. For an operator $*$ $\in \{+, -, \times\}$, since the result of the operator between two interval matrices is obtained by combination of the elementary operators between their entries so by Lemma 2.4, the proof is concluded. $\square$

Theorem 2.6. *For the introduced methods in subsection 2.1, we have*

$$f(A) \in f(\mathbf{A}).$$

Proof. Because in each of the methods introduced in subsection 2.1, $f(\mathbf{A})$ is obtained by a combination of the elementary operators between interval matrices and $A \in \mathbf{A}$, so using Lemma 2.5, the proof is completed. $\square$

Corollary 2.7. *If $\mathbf{A} \subseteq \mathbf{B}$, then applying each of the methods proposed in subsection 2.1 for computing $f(\mathbf{A})$ and $f(\mathbf{B})$ yields*

$$f(\mathbf{A}) \subseteq f(\mathbf{B}).$$



2.3. Considerable aspects. Here, we mention some aspects that help us to obtain better results in other cases.

First, we point that for evaluating the interval matrix function $f(\mathbf{A})$, we can present other methods arising from the same ideas in real cases, such as Jordan canonical form definition, Verde-Star's method, Pade approximation, and Fourier expansion method, see [1, 8].

In a more general case, we can consider the problem of determining $f(A)$, where $A \in \mathbb{C}^{n \times n}$, i.e., entries of A are complex. In this case our idea is generalizable also, i.e., we can use complex intervals and consider the complex interval matrix $\mathbf{A}$ such that $A \in \mathbf{A}$. Two most frequently used representations for complex intervals are as follow:

(i) The infimum-supremum representation

$$[x, \bar{x}] = \{x \in \mathbb{K} : \underline{x} \leq x \leq \bar{x}\}, \quad \text{for some } \underline{x}, \bar{x} \in \mathbb{K}, \quad \underline{x} \leq \bar{x},$$

where $\leq$ is the partial ordering $x \leq y \Leftrightarrow \operatorname{Re}(x) \leq \operatorname{Re}(y) \ \& \ \operatorname{Im}(x) \leq \operatorname{Im}(y)$, for $x, y \in \mathbb{C}$, in which $\operatorname{Re}(x)$ and $\operatorname{Im}(x)$ stand for the real part and the imaginary part of x , respectively.

(ii) The midpoint-radius representation

$$\langle x^c, x^\Delta \rangle = \{x \in \mathbb{K} : |x - x^c| \leq x^\Delta\} \quad \text{for some } x^c \in \mathbb{K}, \quad 0 \leq x^\Delta \in \mathbb{R}.$$

for more details about complex intervals, see [38].

By Corollary 2.7, it is obvious that for obtaining tighter interval matrix function $f(\mathbf{A})$, we can use tighter interval matrix $\mathbf{A}$ containing A . We know that the tighter $f(\mathbf{A})$, the more efficient result.

3. TEST PROBLEMS

In this section, we present some numerical experiments to apply the new techniques to obtain some enclosures for the matrix function $f(A)$. Let $\lambda(A)$ be the set of the eigenvalues of A . In below, $f_P(\mathbf{A})$, $f_S(\mathbf{A})$ and $f_N(\mathbf{A})$ denote the obtained interval matrix function $f(\mathbf{A})$ respectively, by the interval power series, Sylvester, and Newton divided difference methods. In each case, the interval power series method has been implemented by its first 40 terms.

Example 3.1. Let A be Hankel matrix

$$A = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 3 & 4 & 5 & 1 \\ 3 & 4 & 5 & 1 & 2 \\ 4 & 5 & 1 & 2 & 3 \\ 5 & 1 & 2 & 3 & 4 \end{pmatrix},$$

and $f(x) = e^x$. The condition number of A is $\operatorname{cond}(A) = 5.7063$. Using Matlab's `funm` function, we obtain

$$e^A = 10^5 \begin{pmatrix} 6.5381 & 6.5379 & 6.5379 & 6.5380 & 6.5382 \\ 6.5379 & 6.5382 & 6.5382 & 6.5380 & 6.5378 \\ 6.5379 & 6.5382 & 6.5383 & 6.5379 & 6.5378 \\ 6.5380 & 6.5380 & 6.5379 & 6.5381 & 6.5381 \\ 6.5382 & 6.5378 & 6.5378 & 6.5381 & 6.5383 \end{pmatrix}.$$

We consider interval matrix $\mathbf{A} = A + [-\epsilon, \epsilon]$, with $\epsilon = 10^{-4}$. Applying the interval power series, Sylvester and Newton divided difference methods, we obtain the following enclosures for $f(A)$

$$f_P(\mathbf{A}) = 10^5 \begin{pmatrix} [6.5348, 6.5414] & [6.5346, 6.5413] & [6.5346, 6.5412] & [6.5347, 6.5414] & [6.5348, 6.5415] \\ [6.5346, 6.5413] & [6.5349, 6.5415] & [6.5349, 6.5415] & [6.5347, 6.5414] & [6.5345, 6.5411] \\ [6.5346, 6.5412] & [6.5349, 6.5415] & [6.5350, 6.5416] & [6.5346, 6.5413] & [6.5345, 6.5411] \\ [6.5347, 6.5414] & [6.5347, 6.5414] & [6.5346, 6.5413] & [6.5348, 6.5414] & [6.5347, 6.5414] \\ [6.5348, 6.5415] & [6.5345, 6.5411] & [6.5345, 6.5411] & [6.5347, 6.5414] & [6.5350, 6.5416] \end{pmatrix},$$



$$f_S(\mathbf{A}) = 10^5 \begin{pmatrix} [6.5367, 6.5395] & [6.5368, 6.5391] & [6.5369, 6.5389] & [6.5368, 6.5393] & [6.5371, 6.5392] \\ [6.5365, 6.5393] & [6.5370, 6.5394] & [6.5372, 6.5392] & [6.5368, 6.5393] & [6.5368, 6.5389] \\ [6.5365, 6.5393] & [6.5370, 6.5393] & [6.5373, 6.5393] & [6.5367, 6.5392] & [6.5368, 6.5389] \\ [6.5367, 6.5394] & [6.5369, 6.5392] & [6.5369, 6.5390] & [6.5368, 6.5394] & [6.5370, 6.5391] \\ [6.5368, 6.5396] & [6.5367, 6.5390] & [6.5368, 6.5388] & [6.5368, 6.5393] & [6.5373, 6.5394] \end{pmatrix},$$

$$f_N(\mathbf{A}) = 10^5 \begin{pmatrix} [6.5367, 6.5395] & [6.5368, 6.5391] & [6.5369, 6.5389] & [6.5368, 6.5393] & [6.5371, 6.5392] \\ [6.5365, 6.5393] & [6.5371, 6.5394] & [6.5372, 6.5392] & [6.5368, 6.5393] & [6.5368, 6.5389] \\ [6.5365, 6.5393] & [6.5370, 6.5393] & [6.5373, 6.5393] & [6.5367, 6.5392] & [6.5368, 6.5389] \\ [6.5367, 6.5394] & [6.5369, 6.5392] & [6.5369, 6.5389] & [6.5368, 6.5394] & [6.5370, 6.5391] \\ [6.5368, 6.5396] & [6.5367, 6.5390] & [6.5368, 6.5388] & [6.5368, 6.5393] & [6.5373, 6.5394] \end{pmatrix}.$$

As one can see, in all results we have $e^A \in f(\mathbf{A})$. The interval Sylvester and Newton divided difference methods are closed together and give tighter results than one obtained by the interval power series method.

Example 3.2. In this example, we utilize the function "lehmer" of Matlab for producing the matrix A , i.e.,

$$A = \text{full}(\text{gallery}('lehmer', 4)),$$

and $f(x) = \cos x$. The condition number of A is $\text{cond}(A) = 12.2063$. Matlab's `funm` function yields

$$\cos(A) = \begin{pmatrix} 0.5012 & -0.3905 & -0.3033 & -0.2398 \\ -0.3905 & 0.4239 & -0.5223 & -0.4373 \\ -0.3033 & -0.5223 & 0.3822 & -0.5573 \\ -0.2398 & -0.4373 & -0.5573 & 0.4282 \end{pmatrix}.$$

If we put interval matrix $\mathbf{A} = A + [-\epsilon, \epsilon]$, with $\epsilon = 10^{-5}$, then by applying the interval power series, Sylvester, and Newton divided difference methods, respectively, we obtain

$$f_P(\mathbf{A}) = \begin{pmatrix} [0.5011, 0.5013] & [-0.3906, -0.3904] & [-0.3034, -0.3032] & [-0.2399, -0.2397] \\ [-0.3906, -0.3904] & [0.4238, 0.4241] & [-0.5224, -0.5221] & [-0.4375, -0.4372] \\ [-0.3034, -0.3032] & [-0.5224, -0.5221] & [0.3820, 0.3823] & [-0.5575, -0.5572] \\ [-0.2399, -0.2397] & [-0.4375, -0.4372] & [-0.5575, -0.5572] & [0.4281, 0.4283] \end{pmatrix},$$

$$f_S(\mathbf{A}) = \begin{pmatrix} [0.5001, 0.5024] & [-0.3920, -0.3891] & [-0.3048, -0.3018] & [-0.2411, -0.2384] \\ [-0.3920, -0.3891] & [0.4222, 0.4257] & [-0.5240, -0.5205] & [-0.4390, -0.4357] \\ [-0.3048, -0.3018] & [-0.5240, -0.5205] & [0.3803, 0.3840] & [-0.5591, -0.5556] \\ [-0.2411, -0.2384] & [-0.4390, -0.4357] & [-0.5591, -0.5556] & [0.4266, 0.4298] \end{pmatrix},$$

$$f_N(\mathbf{A}) = \begin{pmatrix} [0.5012, 0.5013] & [-0.3906, -0.3904] & [-0.3034, -0.3032] & [-0.2398, -0.2397] \\ [-0.3906, -0.3904] & [0.4239, 0.4240] & [-0.5223, -0.5222] & [-0.4374, -0.4372] \\ [-0.3034, -0.3032] & [-0.5223, -0.5222] & [0.3821, 0.3822] & [-0.5574, -0.5573] \\ [-0.2398, -0.2397] & [-0.4374, -0.4372] & [-0.5574, -0.5573] & [0.4281, 0.4283] \end{pmatrix}.$$

As one can see, in this example the interval power series method and the interval Newton divided difference method are closed together and yield tighter enclosures than one obtained by the interval Sylvester method.



TABLE 1. Ratio and execution times (in second) for Example 3.3

n	Ratio			Time		
	Power	Sylvester	Newton	Power	Sylvester	Newton
5	0.0138	1	0.0084	0.0183	0.0225	0.0072
10	1.4196e-05	1	1.0814e-05	0.0123	0.1057	0.0363
15	2.7478e-08	1	1.4641e-08	0.0165	0.4225	0.0658
20	5.8638e-11	1	3.1485e-11	0.0917	0.4779	0.1183
25	1.4248e-13	1	7.6840e-14	0.0205	0.7338	0.2111
30	3.3651e-16	1	1.8685e-16	0.0184	1.0710	0.3108
35	8.3325e-19	1	5.4447e-18	0.2125	1.5941	0.4766
40	2.0980e-21	1	1.4936e-17	0.0257	2.1922	0.6519
45	5.4891e-24	1	1.9975e-17	0.0840	2.9909	0.9044
50	1.4139e-26	1	1.3088e-17	0.0321	4.0329	1.3382

Example 3.3. For the last example of this section, we want to consider the problem in higher dimensions to compare the interval power series, Sylvester and Newton divided difference methods in terms of the execution times and quality of the obtained results. The obtained enclosures for the matrix function $f(A)$ are compared by relative sums of radii with respect to the result obtained by interval Sylvester method. That is for a computed enclosure $\mathbf{Z}$ by a method and enclosure $\mathbf{Y}$ obtained by the interval Sylvester method, we display

$$\text{Ratio} = \frac{\sum_{i,j} \text{rad}(\mathbf{Z}_{ij})}{\sum_{i,j} \text{rad}(\mathbf{Y}_{ij})}.$$

Now, let us consider $A = \text{tridiag}(-0.2625, 1, -0.2375) \in \mathbb{R}^{n \times n}$ and $f(x) = \sin x$. Put $\mathbf{A} = A + [-\epsilon, \epsilon]$ with $\epsilon = 10^{-5}$. The results of executing three methods for different values of dimension n are summarized in Table 1.

The reported numbers in Table 1 show that the quality of the obtained enclosures by the interval power series and Newton divided difference methods are close together up to dimension $n = 35$ and for higher dimensions the interval power series yields the tightest enclosures. Also, the interval power series performs faster than two other methods which confirms the lower computational complexity of this method proved in Theorem 2.1.

Applying the new approach on an ill-conditioned problem. Here we want to apply our approaches for enveloping the inverse of an ill-conditioned matrix, namely Hilbert matrix.

Consider the 3-by-3 Hilbert matrix

$$H = \begin{pmatrix} 1 & \frac{1}{2} & \frac{1}{3} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \end{pmatrix}.$$

As you know, $\frac{1}{3}$ can not be represented exactly by a finite number of decimals, so let us replace $\frac{1}{3}$ by the interval $\mathbf{x} = [0.333333333, 0.333333334]$, and $\frac{1}{2}$, $\frac{1}{4}$, and $\frac{1}{5}$ are replaced by their exact decimal equivalents. We denote the resulting interval matrix by $\mathbf{A}$

$$\mathbf{A} = \begin{pmatrix} 1.0000 & 0.5000 & \mathbf{x} \\ 0.5000 & \mathbf{x} & 0.2500 \\ \mathbf{x} & 0.2500 & 0.2000 \end{pmatrix}.$$

In [19], an enclosure for $\mathbf{A}^{-1}$ has been obtained in the form

$$E(\mathbf{A}^{-1}) = H^{-1} + 10^{-7} \begin{pmatrix} [-8, 8] & [-20, 30] & [-40, 30] \\ [-50, 60] & [-200, 100] & [-200, 200] \\ [-40, 40] & [-100, 200] & [-200, 200] \end{pmatrix},$$



in which H^{-1} is the exact inverse of Hilbert matrix H

$$H^{-1} = \begin{pmatrix} 9 & -36 & 30 \\ -36 & 192 & -180 \\ 30 & -180 & 180 \end{pmatrix}.$$

Now, we obtain other enclosures for $\mathbf{A}^{-1}$ using our techniques. An approach for evaluating the inverse of a real matrix $B \in \mathbb{R}^{n \times n}$ is calculating its matrix function $f(B)$ wherein $f(x) = \frac{1}{x}$. So, for utilizing our approaches it is enough to compute the interval matrix function $f(\mathbf{A})$.

Applying the interval power series method yields

$$f_P(\mathbf{A}) = H^{-1} + 10^{-7} \begin{pmatrix} [-2, 1] & [-8, 11] & [-10, 7] \\ [-7, 10] & [-54, 38] & [-37, 52] \\ [-10, 7] & [-37, 52] & [-50, 35] \end{pmatrix},$$

and using interval Sylvester method, we obtain

$$f_S(\mathbf{A}) = H^{-1} + 10^{-8} \begin{pmatrix} [-5, 10] & [-6, 11] & [-25, 22] \\ [-6, 11] & [-18, 11] & [-6, 11] \\ [-7, 4] & [-58, 11] & [-5, 10] \end{pmatrix}.$$

As one can see, our approaches give better result than one obtained in [19].

4. SOME APPLICATIONS

(i) Double mass-spring system [12]

Second order differential equations have many applications in science and engineering. Here we consider a double mass-spring system which arise from a physical problem.

Consider the double mass-spring system in Figure 1 with the assumption that there are no frictional forces. Here the natural length of each spring, L_1 and L_2 , respectively, takes into account the horizontal dimension of the object that is, if the springs are neither compressed or stretched, then

$$x_1 = L_1 \quad \text{and} \quad x_2 - x_1 = L_2.$$

The motion of the objects in this physical system is described by the solution of the following system

$$\begin{aligned} m_1 \frac{d^2 x_1(t)}{dt^2} &= -(\kappa_1 + \kappa_2)x_1(t) + \kappa_2 x_2(t) + (\kappa_1 L_1 - \kappa_2 L_2), \\ m_2 \frac{d^2 x_2(t)}{dt^2} &= \kappa_2 x_1(t) - \kappa_2 x_2(t) + \kappa_2 L_2, \end{aligned}$$

see [12]. Equivalently,

$$\begin{aligned} \frac{d^2 x_1(t)}{dt^2} &= a_{11}x_1(t) + a_{12}x_2(t) + b_1, \\ \frac{d^2 x_2(t)}{dt^2} &= a_{21}x_1(t) + a_{22}x_2(t) + b_2, \end{aligned} \tag{4.1}$$

where

$$\begin{aligned} a_{11} &= -\frac{\kappa_1 + \kappa_2}{m_1}, & a_{12} &= \frac{\kappa_2}{m_1}, & b_1 &= \frac{\kappa_1 L_1 - \kappa_2 L_2}{m_1}, \\ a_{21} &= \frac{\kappa_2}{m_2}, & a_{22} &= -\frac{\kappa_2}{m_2}, & b_2 &= \frac{\kappa_2 L_2}{m_2}. \end{aligned}$$

If we consider

$$x(t) = \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix}, \quad \ddot{x}(t) = \begin{pmatrix} \frac{d^2 x_1(t)}{dt^2} \\ \frac{d^2 x_2(t)}{dt^2} \end{pmatrix},$$



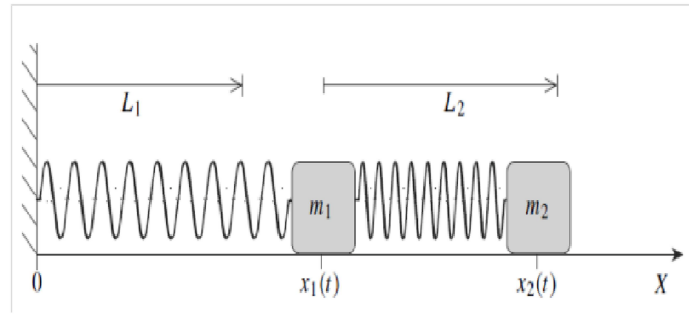


FIGURE 1. A double mass spring system with objects of mass m_1 and m_2 located at positions $x_1(t)$ and $x_2(t)$, respectively. The first spring which has a natural length of L_1 and spring constant κ_1 , connects the object of mass m_1 to the wall. The second spring which has a natural length of L_2 and spring constant κ_2 , connects the two objects together. In this snapshot, the first spring is stretched and the second is compressed.

$$A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}, \quad b = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix},$$

then system of equations (4.1) is equivalent to

$$\ddot{x}(t) = Ax(t) + b.$$

Now suppose $x(0) = x_0$ and $x'(0) = x'_0$ denote the initial states at $t = 0$. Also let $b = 0$, then the second order differential equation corresponding to these conditions is

$$\ddot{x}(t) = Ax(t), \quad x(0) = x_0, \quad x'(0) = x'_0. \quad (4.2)$$

Trigonometric matrix functions arise in the solution of such second order differential equations. The above equation has the following solution

$$x(t) = \cos(\sqrt{-A}t)x_0 + (\sqrt{-A})^{-1} \sin(\sqrt{-A}t)x'_0. \quad (4.3)$$

But $\cos(\sqrt{-A}t)$ and $(\sqrt{-A})^{-1} \sin(\sqrt{-A}t)$ are matrix functions, hence we can compute $x(t)$ by available methods in literature.

Example 4.1. Consider the second order differential equation

$$\begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{pmatrix} = \begin{pmatrix} -6 & 1 \\ 1 & -8 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix},$$

with initial states $x(0) = x_0$ and $x'(0) = x'_0$. We want to determine the state vector $x(t)$ at $t = 1$. According to (4.3)

$$x(1) = \cos(\sqrt{-A})x_0 + (\sqrt{-A})^{-1} \sin(\sqrt{-A})x'_0,$$

where

$$A = \begin{pmatrix} -6 & 1 \\ 1 & -8 \end{pmatrix}.$$



If we consider functions $f(x) = \cos(\sqrt{-x})$ and $g(x) = \frac{\sin(\sqrt{-x})}{\sqrt{-x}}$, then

$$x(1) = f(A)x_0 + g(A)x'_0.$$

We have

$$\lambda(A) = \{-8.4142, -5.5858\}.$$

Using Sylvester method for determining $f(A)$ and $g(A)$, we obtain the following result for $x(1)$

$$x(1) = \begin{pmatrix} -0.7501 & 0.0915 \\ 0.0915 & -0.9332 \end{pmatrix} x_0 + \begin{pmatrix} 0.2656 & 0.0759 \\ 0.0759 & 0.1137 \end{pmatrix} x'_0.$$

Now we consider interval matrix $\mathbf{A} = A + [-\epsilon, \epsilon]$ with $\epsilon = 0.001$. Using interval Sylvester method for enclosing $f(A)$ and $g(A)$, we obtain the following envelope for $x(1)$

$$\begin{aligned} \mathbf{x}(1) = & \begin{pmatrix} [-0.7508, -0.7495] & [0.0909, 0.0922] \\ [0.0909, 0.0922] & [-0.9339, -0.9326] \end{pmatrix} x_0 \\ & + \begin{pmatrix} [0.2654, 0.2657] & [0.0758, 0.0761] \\ [0.0758, 0.0761] & [0.1135, 0.1139] \end{pmatrix} x'_0. \end{aligned}$$

As one can see, interval Sylvester scheme gives a good enclosure for $x(1)$ that guarantees $x(1) \in \mathbf{x}(1)$.

(ii) Phase plane analysis [39]

Phase plane analysis is a graphical method for studying 2D autonomous systems. This method was introduced by mathematicians (among others, Henri Poincare) in 1890.

The basic idea of the method is to generate in the state space motion trajectories corresponding to various initial conditions, and then to examine the qualitative features of the trajectories. As a graphical method, it allows us to visualize what goes on in a nonlinear system starting from various initial conditions, without having to solve the nonlinear equations analytically. Thus, information concerning stability and other motion patterns of the system can be obtained, see [39].

Here we want to consider the first order linear differential equation

$$\dot{x}(t) = Ax(t), \quad x(0) = x_0.$$

This equation has the unique solution $x(t) = e^{tA}x_0$ in which e^{tA} is a matrix function.

Example 4.2. [39] Consider the system $\dot{x}(t) = Ax(t)$ wherein $A = \begin{pmatrix} a & b \\ -b & a \end{pmatrix}$, if the initial condition is $x(0) = x_0$, then $x(t) = e^{tA}x_0$ where

$$e^{tA} = e^{ta} \begin{pmatrix} \cos(tb) & \sin(tb) \\ -\sin(tb) & \cos(tb) \end{pmatrix}.$$

If the initial condition $x(0)$ has polar coordinates (r_0, θ_0) , then the solution is given by

$$x_1(t) = e^{ta}r_0 \cos(\theta_0 - bt) \quad \text{and} \quad x_2(t) = e^{ta}r_0 \sin(\theta_0 - bt), \quad t \geq 0,$$

so the trajectories are spirals if a is nonzero, moving towards the origin if $a < 0$, and outward if $a > 0$. If $a = 0$ the trajectories are circles, see Figure 2. Now for bounding the errors arising from round off, we use the proposed techniques in this paper for determining an enclosure for e^{tA} . Let $g(x) = e^x$. For the matrix A we have $\lambda(A) = \{\lambda_1, \lambda_2\} = \{a - ib, a + ib\}$. Using the interval matrix $\mathbf{A} = A + [-\epsilon, \epsilon]$ with $\epsilon > 0$, and interval Newton divided



difference method we can write

$$\begin{aligned}
 g_N(t\mathbf{A}) &= g(t\lambda_1)I + \frac{g(t\lambda_2) - g(t\lambda_1)}{t\lambda_2 - t\lambda_1}(t\mathbf{A} - t\lambda_1 I) \\
 &= \begin{pmatrix} e^{t(a-ib)} & 0 \\ 0 & e^{t(a-ib)} \end{pmatrix} + \frac{e^{t(a+ib)} - e^{t(a-ib)}}{(ta + itb) - (ta - itb)} \\
 &\quad \left\{ \begin{pmatrix} ta & tb \\ -tb & ta \end{pmatrix} - \begin{pmatrix} t(a-ib) & 0 \\ 0 & t(a-ib) \end{pmatrix} + t[-\epsilon, \epsilon] \right\} \\
 &= e^{ta} \begin{pmatrix} \cos(tb) & \sin(tb) \\ -\sin(tb) & \cos(tb) \end{pmatrix} + \frac{e^{2ta} \sin tb}{b} [-\epsilon, \epsilon] \\
 &= e^{tA} + \frac{e^{2ta} \sin tb}{b} [-\epsilon, \epsilon].
 \end{aligned} \tag{4.4}$$

It is obvious that (4.4) is an enclosure for e^{tA} which its tightness depends on ϵ . Thus for the state vector $x(t)$ we have

$$x(t) \in \{e^{tA} + \frac{e^{2ta} \sin tb}{b} [-\epsilon, \epsilon]\}.$$

(iii) Control theory [8]

Consider the mass-spring-friction system shown in Figure 3(a). The linear motion concerned is in the horizontal direction. The free-body diagram of the system is shown in Figure 3(b). The force equation of the system is

$$f(t) = M \frac{d^2 y(t)}{dt^2} + B \frac{dy(t)}{dt} + Ky(t).$$

The state diagram of the system is constructed as shown in Figure 3(c). By defining the outputs of the integrators on the state diagram as state variables x_1, x_2 the state equations are

$$\frac{dx_2(t)}{dt} = -\frac{K}{M}x_1(t) - \frac{B}{M}x_2(t) + \frac{1}{M}f(t), \quad \frac{dx_1(t)}{dt} = x_2(t),$$

so in matrix form we have

$$\begin{pmatrix} \frac{dx_1(t)}{dt} \\ \frac{dx_2(t)}{dt} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -\frac{K}{M} & -\frac{B}{M} \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{f(t)}{M} \end{pmatrix}.$$

Now if we consider

$$x(t) = \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix}, \quad \dot{x}(t) = \begin{pmatrix} \frac{dx_1(t)}{dt} \\ \frac{dx_2(t)}{dt} \end{pmatrix},$$

$$A = \begin{pmatrix} 0 & 1 \\ -\frac{K}{M} & -\frac{B}{M} \end{pmatrix}, \quad C = \begin{pmatrix} 0 \\ \frac{1}{M} \end{pmatrix},$$

then

$$\dot{x}(t) = Ax(t) + Cf(t).$$

Suppose $f(t) = 0$, so we will have

$$\dot{x}(t) = Ax(t),$$

with initial condition $x(0) = x_0$. As we said previously, this equation has the unique solution $x(t) = e^{tA}x_0$.



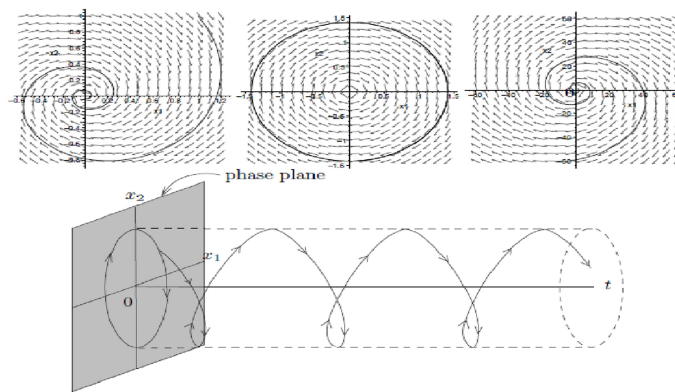


FIGURE 2. The phase plane trajectory of introduced system in Example 4.2. The last figure shows the phase plane trajectory as a projection of the curve $(t, x_1(t), x_2(t))$ in $\mathbb{R}^3$: case when $a = 0$ and $b > 0$.

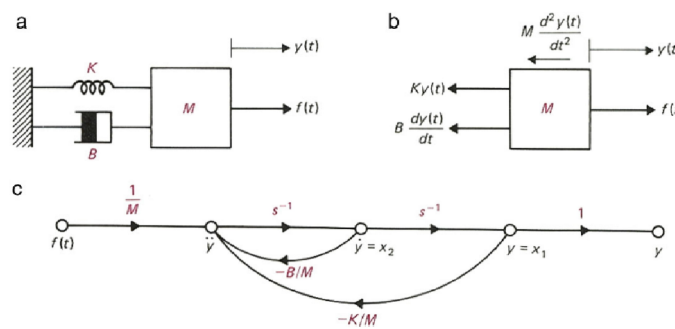


FIGURE 3. (a) Mass spring friction system. (b) Free-body diagram. (c) State diagram.

Example 4.3. [8] Consider the state equation

$$\begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -2 & -3 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} f(t).$$

The problem is determining the state vector $x(t)$ for $t \geq 0$ when input is $f(t) = 0$ for $t \geq 0$. For this example we have

$$sI - A = \begin{pmatrix} s & -1 \\ 2 & s+3 \end{pmatrix}.$$

If we find the inverse of the matrix $(sI - A)$ and take its Laplace transform $\ell^{-1}[(sI - A)^{-1}]$ then we obtain

$$e^{tA} = \ell^{-1}[(sI - A)^{-1}] = \begin{pmatrix} 2e^{-t} - e^{-2t} & e^{-t} - e^{-2t} \\ -2e^{-t} + 2e^{-2t} & -e^{-t} + 2e^{-2t} \end{pmatrix}.$$

Thus for $x(t)$ we have

$$x(t) = \begin{pmatrix} 2e^{-t} - e^{-2t} & e^{-t} - e^{-2t} \\ -2e^{-t} + 2e^{-2t} & -e^{-t} + 2e^{-2t} \end{pmatrix} x(0).$$

Now, for bounding the round off errors, we determine an interval matrix function that encloses e^{tA} . Let $g(x) = e^x$, we compute the interval matrix function $g(t\mathbf{A}) = e^{t\mathbf{A}}$, therein $\mathbf{A} = A + [-\epsilon, \epsilon]$. For the real matrix A , we have $\lambda(A) = \{\lambda_1, \lambda_2\} = \{-2, -1\}$. Using the interval Newton divided difference method we can write

$$\begin{aligned} g_N(t\mathbf{A}) &= g(t\lambda_1)I + \frac{g(t\lambda_2) - g(t\lambda_1)}{t\lambda_2 - t\lambda_1}(t\mathbf{A} - t\lambda_1 I) \\ &= \begin{pmatrix} e^{-2t} & 0 \\ 0 & e^{-2t} \end{pmatrix} + \frac{e^{-t} - e^{-2t}}{-t + 2t} \left\{ \begin{pmatrix} 0 & t \\ -2t & -3t \end{pmatrix} - \begin{pmatrix} -2t & 0 \\ 0 & -2t \end{pmatrix} + t[-\epsilon, \epsilon] \right\} \\ &= \begin{pmatrix} 2e^{-t} - e^{-2t} & e^{-t} - e^{-2t} \\ -2e^{-t} + 2e^{-2t} & -e^{-t} + 2e^{-2t} \end{pmatrix} + (e^{-t} - e^{-2t})[-\epsilon, \epsilon] \\ &= e^{tA} + (e^{-t} - e^{-2t})[-\epsilon, \epsilon]. \end{aligned} \quad (4.5)$$

It is obvious that (4.5) is an enclosure for e^{tA} which depends on ϵ . Thus, for the state vector $x(t)$ for $t \geq 0$ we have

$$x(t) \in \{e^{tA} + (e^{-t} - e^{-2t})[-\epsilon, \epsilon]\}x(0).$$

5. CONCLUDING REMARKS

In this paper, for a given matrix $A \in \mathbb{R}^{n \times n}$ and an analytic function f , we proposed some approaches to obtain an enclosure $f(\mathbf{A})$ for the matrix function $f(A)$. We then proved that for all the proposed approaches $f(A) \in f(\mathbf{A})$. For some numerical examples, we computed interval matrix function $f(\mathbf{A})$ and showed that in all cases $f(A) \in f(\mathbf{A})$ which confirmed the presented analytical arguments. The numerical tests showed the performance of the proposed approaches for bounding the error of determining matrix functions.

ACKNOWLEDGEMENT

The author would like to thank the Shahid Chamran University of Ahvaz for financial support under the grant number SCU.MM99.33518.

REFERENCES

- [1] B. Adcock and D. Huybrechs, *Multivariate modified Fourier expansions*, Proceedings of the International Conference on Spectral and High Order Methods, (2009), 85–92.
- [2] G. Alefeld and J. Herzberger, *Introduction to Interval Computations*, Academic Press, New York, 1983.
- [3] S. Barnett, *Matrices: Methods and Applications*, 2nd ed., Oxford University Press, 1994.
- [4] R. Bellman, *Stability Theory of Differential Equations*, McGraw-Hill, 1953.
- [5] J. C. Burkill, *Functions of intervals*, Proceedings of the London Mathematical Society, 22 (1924), 375–446.
- [6] D. Cheu and L. Longpre, *Towards the possibility of objective interval uncertainty in physics*, Reliab. Comput., 15 (2011) 43–46.
- [7] G. W. Cross and P. Lancaster, *Square roots of complex matrices*, Linear Multilinear Algebra, 1 (1974), 289–293.
- [8] M. Dehghan and M. Hajarian, *Determination of a matrix function using the divided difference method of Newton and the interpolation technique of Hermite*, J. Comput. Appl. Math., 231 (2009), 67–81.
- [9] M. Dehghan and M. Hajarian, *Computing matrix functions using mixed interpolation methods*, Math. Comput. Modelling, 52 (2010), 826–836.
- [10] L. Dymova and M. Pilarek, *Organizing Calculations in Algorithms for Solving Systems of Interval Linear Equations Using the Interval Extended Zero Method*, Parallel Processing and Applied Mathematics, 7204 (2012), 439–446.
- [11] L. Dymova, P. Sevastjanov, and M. Pilarek, *A method for solving systems of linear interval equations applied to the Leontief input output model of economics*, Expert Systems with Applications, 40 (2013), 222–230.
- [12] S. S. Ganji, A. Barari, and D. D. Ganji, *Approximate analysis of two-mass spring systems and buckling of a column*, Comput. Math. Appl., 61 (2011), 1088–1095.
- [13] J. Garloff, *Bibliography on interval mathematics, continuation*, Freiburger Intervall-Berichte, 2 (1987), 1–50.



- [14] J. Garloff and K. P. Schwierz, *A bibliography on interval mathematics*, Comput. Appl. Math., 6 (1980), 67–79.
- [15] M. Ghanbari, T. Allahviranloo, and E. Hagh, *Estimation of algebraic solution by limiting the solution set of an interval linear system*, Soft Comput., 16 (2012), 2135–2142.
- [16] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed., Johns Hopkins University Press, Baltimore, MD, 1996.
- [17] A. Goldsztejn, *Modal intervals revisited, Part 1: A generalized interval natural extension*, Reliab. Comput., 16 (2012), 130–183.
- [18] A. Goldsztejn, *Modal intervals revisited, Part 2: A generalized interval mean value extension*, Reliab. Comput., 16 (2012), 184–209.
- [19] E. R. Hansen, *Interval arithmetic in matrix computations, Part I*, SIAM J. Numer. Anal., 2 (1965), 308–320.
- [20] E. R. Hansen, *Global Optimization Using Interval Analysis*, Dekker, Inc., New York, 1992.
- [21] G. I. Hargreaves and N. J. Higham, *Efficient algorithms for the matrix cosine and sine*, Numer. Algorithms, 40 (2005), 383–400.
- [22] N. J. Higham, *The scaling and squaring method for the matrix exponential revisited*, SIAM J. Matrix Anal. Appl., 26 (2005), 1179–1193.
- [23] N. J. Higham, *Functions of Matrices: Theory and Computation*, SIAM, 2008.
- [24] N. J. Higham and M. I. Smith, *Computing the matrix cosine*, Numer. Algorithms 34 (2003), 13–26.
- [25] M. Hladik, *Solution sets of complex linear interval systems of equations*, Reliab. Comput., 14 (2010), 78–87.
- [26] M. Hladik, *Weak and strong solvability of interval linear systems of equations and inequalities*, Linear Algebra Appl., 438 (2013), 4156–4165.
- [27] R. B. Kearfott and V. Kreinovich, editors. *Applications of Interval Computations*, Applied Optimization, Dordrecht, Netherlands, Kluwer, 1996.
- [28] R. B. Kearfott, *Interval computations: Introduction, uses, and resources*, Euromath Bulletin, (1996), 95–112.
- [29] B. J. Kubica and A. Wozniak, *Interval methods for computing the Pareto-front of a multicriterial problem*, Parallel Processing and Applied Mathematics, 4967 (2008), 1382–1391.
- [30] B. J. Kubica and A. Wozniak, *An Interval Method for Seeking the Nash Equilibria of Non-cooperative Games*, Parallel Processing and Applied Mathematics, 6068 (2010), 446–455.
- [31] Y. Y. Lu, *A Pade approximation method for square roots of symmetric positive definite matrices*, SIAM J. Matrix Anal. Appl. 19 (1998), 833–845.
- [32] R. E. Moore, *Interval Arithmetic and Automatic Error Analysis in Digital Computing*, Thesis (Ph.D.)–Stanford University, 1962
- [33] R. E. Moore, R. B. Kearfott and M. J. Cloud, *Introduction to Interval Analysis*, SIAM, 2009.
- [34] A. Neumaier, *Interval Methods for Systems of Equations*, Cambridge University Press, Cambridge, England, 1990.
- [35] M. Pilarek and R. Wyrzykowski, *Solving Systems of Interval Linear Equations in Parallel Using Multithreaded Model and Interval Extended Zero Method*, Parallel Processing and Applied Mathematics, 7203 (2012), 206–214.
- [36] H. Richter, *Über Matrixfunktion*, Math. Ann., 122 (1950), 16–34.
- [37] R. F. Rinehart, *The equivalence of definitions of a matrix function*, Amer. Math. Monthly, 62 (1955), 395–414.
- [38] S. M. Rump, *Fast and parallel interval arithmetic*, BIT Numer. Math., 39 (1999), 534–554.
- [39] A. Sasane, *Differential Equations*, Department of Mathematics, London School of Economics, Lecture notes.
- [40] S. P. Shary, *On nonnegative interval linear systems and their solution*, Reliab. Comput., 15 (2011), 358–369.
- [41] A. Shehata and M. Abul-Dahab, *On Humbert matrix functions*, J. Egyptian Math. Soc., 20 (2012), 167–171.
- [42] T. Sunaga, *Theory of an Interval Algebra and its Application to Numerical Analysis*, Gaukutsu Bunken Fukeyukai, Tokyo, 1958.
- [43] J. J. Sylvester, *On the equation to the secular inequalities in the planetary theory*, Phill. Mag., 16 (1883), 267–269.
- [44] R. C. Young, *The algebra of many-valued quantities*, Math. Ann., 104 (1931), 260–290.

