International Journal of Applied Mathematics

Volume 37 No. 6 2024, 699–711
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)
doi: http://dx.doi.org/10.12732/ijam.v37i6.7

ANALYSIS ON COMPUTATIONAL ISSUES WHEN APPROXIMATING FRACTIONAL POWERS OF SPARSE SPD MATRICES

Dimitar Slavchev [1,*], **Stanislav Harizanov** [1,2,†] **and Nikola Kosturski** [1,‡]

[1] Institute of Information and Communication Technologies

Bulgarian Academy of Sciences

Acad. G. Bonchev Str. Bl. 25A, Sofia - 1113, BULGARIA

[*] e-mail: dimitargslavchev@parallel.bas.bg

[†] e-mail: sharizanov@parallel.bas.bg

[‡] e-mail: kosturski@parallel.bas.bg

[2] Institute of Mathematics and Informatics

Bulgarian Academy of Sciences

Acad. G. Bonchev Str. Bl. 8, Sofia - 1113, BULGARIA

[†] e-mail: sharizanov@math.bas.bg

Abstract

Fractional diffusion has many applications in science and engineering as it models non-local processes and phenomena. However, numerically solving such problems involves systems of linear algebraic equations with dense matrices. For practical problems such systems can be extremely large and applying the usual *LU* factorization methods becomes an extremely expensive computational task. The Best Uniform Rational Approximation (BURA) and

Received: July 1, 2024

related methods have been developed in order to compute an approximation of the inverse $\mathbb{A}^{-\alpha}$ of a symmetric positive definite matrix $\mathbb{A}$ via an approximation of the scalar function $t^\alpha, \alpha \in (0,1), t \in [0,1]$. Thus, the solution of a system of linear algebraic equations $\mathbb{A}^\alpha \mathbf{u} = \mathbf{f}$ can be computed approximately via computing several auxiliary systems with as sparse matrices as $\mathbb{A}$.

This paper is devoted to the analysis of various numerical issues that arise in the process of the BURA computations when $\alpha \in (1,2)$. Different reformulations of the classical BURA setting are considered in order to improve the stability of the computational process. Furthermore, since the direct BURA method does not preserve the symmetric positive definite property of $\mathbb{A}$, alternatives are proposed. They are based on a superposition of several BURA solvers with smaller $\alpha_i \in (0,1]$. Theoretical and experimental analysis on their behavior is provided.

MSC 2020: 08-08, 15Axx, 41A20, 41A44, 65F45

Key Words and Phrases: fractional linear algebraic equations, best uniform rational approximation, error analysis, scientific computing

1. Introduction

In this paper, we investigate computational issues regarding the derivation of an approximate solution for the fractional linear equation

$$\mathbb{A}^\alpha \mathbf{u} = \mathbf{f}, \quad \alpha \in (1,2). \tag{1}$$

The matrix $\mathbb{A} \in \mathbb{R}^{N\times N}$ is sparse, symmetric, and positive definite (SPD) with eigenvalues $\{\lambda_i\}_{i=1}^N$, such that $0 < \lambda_1 < \lambda_2 < \cdots < \lambda_N < +\infty$, and normalized eigenvectors $\{\Psi_i\}_{i=1}^N$. Denote by $\langle , \rangle$ the Euclidean dot product in $\mathbb{R}^N$. Since

$$\mathbf{f} = \sum_{i=1}^N \langle \mathbf{f}, \Psi_i \rangle \Psi_i =: \sum_{i=1}^N f_i \Psi_i, \tag{2}$$

the eigenvectors of $\mathbb{A}^\alpha$ coincide with the eigenvectors of $\mathbb{A}$, while the eigenvalues of $\mathbb{A}^\alpha$ are the corresponding power of the eigenvalues of $\mathbb{A}$, we obtain that the exact solution of (1) is expanded in the following way:

$$\mathbf{u} = \mathbb{A}^{-\alpha}\mathbf{f} = \sum_{i=1}^N \lambda_i^{-\alpha} \langle \mathbf{f}, \Psi_i \rangle \Psi_i = \sum_{i=1}^N \lambda_i^{-\alpha} f_i \Psi_i. \tag{3}$$

Even though $\mathbb{A}$ is sparse, the matrix $\mathbb{A}^\alpha$ is everywhere dense, thus for large values of N it is practically impossible to explicitly compute it or even store it in the computer memory. Therefore, the action of $\mathbb{A}^{-\alpha}$ needs to be approximated.

When $\alpha \in (0,1)$, the problem (1) has been extensively studied in the last decade within the context of fractional sub-diffusion (see for example

[1, 2, 3, 4, 5, 6, 7] and the surveys [8, 9]). There, the discretization of the problem $\mathcal{A}^\alpha u = f$, where $\mathcal{A}$ is an elliptic operator of second order, is done via finite elements or finite differences approximations, resulting in a symmetric matrix $\mathbb{A}$ that acts on the vector $\mathbf{u}$ of the unknown values of u at the mesh points. Actually, many examples of problems of the type (1) in the literature can be related to approximation of elliptic equations with various boundary conditions.

When $\alpha \in (1, 2)$, the problem (1) is related to fractional super-diffusion. Most of the approaches for the sub-diffusion setting rely heavily on $\alpha \in (0, 1)$, thus are non-applicable here. In this paper we apply the BURA method [10] and our focus is on the variety of numerical issues that arise from that. For the validation of the computations we use theoretical estimates and comparison analysis among different realizations of the method. The additive realization of the BURA solver has been analyzed in [11]. Here, we deal with its multiplicative version.

2. **Best Uniform Rational Approximation**

The abbreviation BURA stands for Best Uniform Rational Approximation. Let us consider the min-max problem: find $r_{\alpha,k} \in \mathcal{R}(k, k)$, where $r_{\alpha,k}(t) = P_{\alpha,k}(t)/Q_{\alpha,k}(t)$, $P_{\alpha,k}$ and $Q_{\alpha,k}$ are polynomials of degree k, such that

$$\max_{t\in[0,1]} |t^\alpha - r_{\alpha,k}(t)| = \min_{r_k(t)\in\mathcal{R}(k,k)} \max_{t\in[0,1]} |t^\alpha - r_k(t)|, \quad \alpha \in (0, 2).$$

Then the error $E_{\alpha,k}$ of the k-BURA element $r_{\alpha,k}$ is the maximal absolute value for the error function $\varepsilon_{\alpha,k}(t)$, namely

$$E_{\alpha,k} := \max_{t\in[0,1]} |\varepsilon_{\alpha,k}(t)| = \max_{t\in[0,1]} |t^\alpha - r_{\alpha,k}(t)|. \tag{4}$$

A sharp estimate of $E_{\alpha,k}$ (compare with Table 1) is derived in [12]:

$$E_{\alpha,k} = 4^{\alpha+1} |\sin(\alpha\pi)| e^{-2\pi\sqrt{\alpha k}}. \tag{5}$$

Following [5] we introduce the approximation of $\mathbb{A}^{-\alpha}$ in the form

$$\mathbb{A}^{-\alpha} \approx \lambda_1^{-\alpha} r_{\alpha,k}(\lambda_1 \mathbb{A}^{-1}) \quad \Longrightarrow \quad \mathbf{u}_{\alpha,k} = \lambda_1^{-\alpha} r_{\alpha,k}(\lambda_1 \mathbb{A}^{-1})\mathbf{f}, \tag{6}$$

where $\mathbf{u}_{\alpha,k}$ is the BURA numerical solution of the linear algebraic system (1). The λ_1-normalization maps the spectrum of $\lambda_1\mathbb{A}^{-1}$ onto the unit interval $(0, 1]$, guaranteeing that 1 is always an eigenvalue. Denoting $t_i := \lambda_1/\lambda_i$ and $\bar{f}_i = f_i/\|\mathbf{f}\|_2$, and using the commutativity of the matrix powers, we obtain the following error estimate:

$$\|\mathbf{u} - \mathbf{u}_{\alpha,k}\|_2^2 = \lambda_1^{-2\alpha} \sum_{i=1}^{N} \left(\varepsilon_{\alpha,k}(t_i)\bar{f}_i\right)^2 \Rightarrow \|\mathbf{u} - \mathbf{u}_{\alpha,k}\|_2 \le \lambda_1^{-\alpha} E_{\alpha,k} \|\mathbf{f}\|_2. \tag{7}$$

TABLE 1. The BURA theoretical error $E_{\alpha,k}$ for different values of α and k.

k	$\alpha = 0.25$	$\alpha = 0.5$	$\alpha = 0.625$	$\alpha = 0.75$	$\alpha = 0.875$	$\alpha = 1.25$	$\alpha = 1.5$	$\alpha = 1.75$
1	9.7494E-02	4.3689E-02	2.8156E-02	1.6457E-02	7.3262E-03	1.1027E-02	1.8931E-02	2.4876E-02
2	3.1116E-02	8.5015E-03	4.3988E-03	2.0799E-03	7.5068E-04	5.8020E-04	5.7554E-04	3.3350E-04
3	1.2348E-02	2.2821E-03	9.9930E-04	4.0408E-04	1.2560E-04	6.3312E-05	4.7278E-05	2.0348E-05
4	5.5662E-03	7.3656E-04	2.8032E-04	9.9540E-05	2.7373E-05	9.8222E-06	5.9201E-06	2.0622E-06
5	2.7348E-03	2.6896E-04	9.0476E-05	2.8676E-05	7.0894E-06	1.9015E-06	9.5789E-07	2.8067E-07
6	1.4312E-03	1.0747E-04	3.2337E-05	9.2522E-06	2.0791E-06	4.3049E-07	1.8526E-07	4.6720E-08
7	7.8650E-04	4.6037E-05	1.2502E-05	3.2566E-06	6.7060E-07	1.0972E-07	4.0960E-08	9.0299E-09
8	4.4950E-04	2.0852E-05	5.1471E-06	1.2288E-06	2.3334E-07	3.0713E-08	1.0062E-08	1.9616E-09
9	2.6536E-04	9.8893E-06	2.2318E-06	4.9096E-07	8.6419E-08	9.2829E-09	2.6930E-09	4.6846E-10
10	1.6100E-04	4.8760E-06	1.0109E-06	2.0584E-07	3.3728E-08	2.9918E-09	7.7431E-10	1.2105E-10

Let us denote the zeros of $P_{\alpha,k}$ and $Q_{\alpha,k}$ by $\{\zeta_i\}_{i=1}^k$ and $\{d_i\}_{i=1}^k$, respectively. For $\alpha \in (0,2)$, all zeros $\{\zeta_i\}$ and poles $\{d_i\}$ of $r_{\alpha,k}$ are real and different. Moreover (see [12]), the following interlacing property holds true:

$$\begin{aligned} 0 > \zeta_1 > d_1 > \zeta_2 > d_2 > \cdots > \zeta_k > d_k > -\infty, \qquad & \alpha \in (0,1), \\ d_1 > 1 > \zeta_1 > 0 > \zeta_2 > d_2 > \cdots > \zeta_k > d_k > -\infty, \qquad & \alpha \in (1,2). \end{aligned} \tag{8}$$

Thus, we can rewrite the BURA element in a computationally more convenient form:

$$r_{\alpha,k}(t) = c_0 \prod_{i=1}^{k} \frac{t-\zeta_i}{t-d_i}, \quad \text{where} \quad c_0 = \pm E_{\alpha,k} \prod_{i=1}^{k} \frac{d_i}{\zeta_i}, \quad \alpha \in (0,2). \tag{9}$$

For the derivation of c_0, we have used that 0 is always an extreme point for $\varepsilon_{\alpha,k}(t)$ (see [12]). Therefore, due to (8), it is straightforward to conclude that $r_{\alpha,k}(0) = E_{\alpha,k}$, when $\alpha \in (0,1)$ and $r_{\alpha,k}(0) = -E_{\alpha,k}$, when $\alpha \in (1,2)$.

In terms of the matrix representation, the BURA solution is in the form

$$\mathbf{u}_{\alpha,k} = \lambda_1^{-\alpha} c_0 \left[\prod_{i=1}^{k} (\lambda_1 \mathbb{I} - \zeta_i \mathbb{A})(\lambda_1 \mathbb{I} - d_i \mathbb{A})^{-1} \right] \mathbf{f}. \tag{10}$$

Because of the exponential decay of $E_{\alpha,k}$ with respect to k (see (5)), it is suggested by (7) that the operator $\lambda_1^{-\alpha} r_{\alpha,k}(\lambda_1 \mathbb{A}^{-1})$ is a very good and adequate ℓ_2-approximation of $\mathbb{A}^{-\alpha}$. This is indeed the case for $\alpha \in (0,1)$, where not only the approximation error is optimal (within the class of the (k,k) rational functions, see [9]), but also key properties of $\mathbb{A}^{-\alpha}$ (such as SPD, double non-negativity, etc.) are inherited by its approximant. However, this is not the case for $\alpha \in (1,2)$ as the spectrum of $\lambda_1^{-\alpha} r_{\alpha,k}(\lambda_1 \mathbb{A}^{-1})$ is

$$\begin{aligned} \left\{ \lambda_1^{-\alpha} r_{\alpha,k}\left(\frac{\lambda^1}{\lambda_N}\right), \lambda_1^{-\alpha} r_{\alpha,k}\left(\frac{\lambda^2}{\lambda_N}\right), \ldots, \lambda_1^{-\alpha} r_{\alpha,k}\left(\frac{\lambda^N}{\lambda_N}\right) \right\} \\ = \left\{ \frac{r_{\alpha,k}(t_1)}{\lambda_1^{\alpha}}, \frac{r_{\alpha,k}(t_2)}{\lambda_1^{\alpha}}, \ldots, \frac{r_{\alpha,k}(t_N)}{\lambda_1^{\alpha}} \right\}, \end{aligned}$$

and $r_{\alpha,k}(t_i) < 0$ whenever $t_i < \zeta_1$. More precisely, if the condition number $\kappa(\mathbb{A})$ of $\mathbb{A}$ is larger than ζ_1^{-1}, the approximant is no longer an SPD matrix. For example, when $\mathbb{A}$ is generated via a finite difference discretization of an elliptic operator $\mathcal{A}$ on a uniform grid with a step h, it is well-known that $\kappa(\mathbb{A}) \sim O(h^{-2})$. According to Table 2, when $h \leq 10^{-4}$ none of its approximants are SPD matrices for all choices of $\alpha \in \{1.25, 1.50, 1.75\}$ and $k \in [1, 10]$.

TABLE 2. Values of ζ_1 for the BURA elements $r_{\alpha,k}$ with $\alpha = \{1.25, 1.5, 1.75\}$ and $k \in [1, 10]$

α \ k	1	2	3	4	5	6	7	8	9	10
1.25	1.49E-02	1.47E-03	2.51E-04	5.67E-05	1.52E-05	4.64E-06	1.55E-06	5.62E-07	2.15E-07	8.72E-08
1.5	3.22E-02	3.52E-03	6.73E-04	1.68E-04	5.02E-05	1.67E-05	6.14E-06	2.40E-06	1.00E-06	4.35E-07
1.75	5.07E-02	5.58E-03	1.15E-03	3.13E-04	1.00E-04	3.60E-05	1.41E-05	5.89E-06	2.60E-06	1.20E-06

An alternative is to apply BURA products. In what follows, we will consider the notation $\bar{\alpha}$ for the set $\{\alpha_1, \alpha_2, \ldots, \alpha_n\}$, $\alpha_i \in (0, 1]$, with the property $\sum \alpha_i = \alpha$, and we will consider the operators

$$\mathbb{A}^{-\alpha} \approx \lambda_1^{-\alpha} r_{\bar{\alpha},k}(\lambda_1 \mathbb{A}^{-1}), \quad \text{where} \quad r_{\bar{\alpha},k} := r_{\alpha_n,k} \circ r_{\alpha_{n-1},k} \circ \cdots \circ r_{\alpha_1,k}. \tag{11}$$

The approximant $\lambda_1^{-\alpha} r_{\bar{\alpha},k}(\lambda_1 \mathbb{A}^{-1})$ is always an SPD matrix. Analogously to (7), we derive

$$\begin{aligned} \|\mathbf{u} - \mathbf{u}_{\bar{\alpha},k}\|_2 &= \lambda_1^{-\alpha} \sqrt{\sum_{i=1}^{k} \left(t_i^{\alpha} - \prod_{j=1}^{k} r_{\alpha_j,k}(t_i) \right)^2 \bar{f}_i^2} \\ &\leq \lambda_1^{-\alpha} \max_i \left| t_i^{\alpha} - \prod_{j=1}^{k} r_{\alpha_j,k}(t_i) \right| \|\mathbf{f}\|_2. \end{aligned} \tag{12}$$

Let us first consider the case $n = 2$, i.e., $\alpha = \alpha_1 + \alpha_2$, $\alpha_{1,2} \in (0, 1]$. At first glance, we directly have the estimate

$$\begin{aligned} &\left| t_i^{\alpha} - \left(t_i^{\alpha_1} + \varepsilon_{\alpha_1,k}(t_i) \right) \left(t_i^{\alpha_2} + \varepsilon_{\alpha_2,k}(t_i) \right) \right| \\ &\qquad = \left| t_i^{\alpha_1} \varepsilon_{\alpha_2,k}(t_i) + t_i^{\alpha_2} \varepsilon_{\alpha_1,k}(t_i) + \varepsilon_{\alpha_1,k}(t_i) \varepsilon_{\alpha_2,k}(t_i) \right| \\ &\qquad \leq E_{\alpha_1,k} + E_{\alpha_2,k} + E_{\alpha_1,k} E_{\alpha_2,k}. \end{aligned}$$

However, the upper bound is not sharp in practice. Indeed, the first two terms of the estimated expression become smaller as t_i decreases, while the order of the last term is a product of orders and is dominated by the former terms. What we observe is that, since $t_1 = 1$ and 1 is also an extreme point for both $\varepsilon_{\alpha_{1,2},k}(t)$ with $r_{\alpha_{1,2},k}(1) = 1 - E_{\alpha_{1,2},k}$, the dominant term in (12) is the one for

t_1, where the computation is explicit and gives rise to

$$\left|t_1^{\alpha} - \left(t_1^{\alpha_1} + \varepsilon_{\alpha_1,k}(t_1)\right)\left(t_1^{\alpha_2} + \varepsilon_{\alpha_2,k}(t_1)\right)\right| = |1 - (1 - E_{\alpha_1,k})(1 - E_{\alpha_2,k})| \\ = E_{\alpha_1,k} + E_{\alpha_2,k} - E_{\alpha_1,k}E_{\alpha_2,k}.$$

Moreover, whenever $t_2 \ll 1$ we have

$$\|\mathbf{u} - \mathbf{u}_{\bar{\alpha},k}\|_2 \approx \lambda_1^{-\alpha}\left(E_{\alpha_1,k} + E_{\alpha_2,k} - E_{\alpha_1,k}E_{\alpha_2,k}\right)|\bar{f}_1| \cdot \|\mathbf{f}\|_2. \tag{13}$$

Finally, according to (5), $E_{\alpha,k}$ is a convex function with respect to α, thus the smallest error for $n = 2$ will be for the choice $\alpha_{1,2} = \alpha/2$. Similarly, in the general case the optimal choice is $\alpha_{1,2,\ldots,n} = \alpha/n$.

3. **Numerical Results**

As an example, from now on we consider $\mathbb{A} = diag(-1, 2, -1)/h^2$ with $h = 2^{-18}$ and $N = 2^{18} - 1$. The eigenvalues and eigenvectors of $\mathbb{A}$ are explicitly known:

$$\lambda_i = \frac{4}{h^2}\sin^2\left(\frac{i\pi h}{2}\right), \quad \Psi_i = \left\{\sin(i\pi j h)\right\}_{j=1}^{N}, \quad i = 1, 2, \ldots, 2^{18} - 1.$$

We investigate the cases $\alpha \in \{1.25, 1.50, 1.75\}$ and $k \in [1, 10]$. For each of them, we compute the direct BURA solution (6) and several product BURA solutions (11). For the latter, we always consider the case $n = 2$ with $\bar{\alpha} = \{1, 1 - \alpha\}$ and $\bar{\alpha} = \{\alpha/2, \alpha/2\}$, while for $n > 2$ we only consider the optimal partition $\alpha_{1,\ldots,n} = \alpha/n$. For the case $\alpha = 1.25$ we consider also $\bar{\alpha} = \{0.50, 0.75\}$. We study the case $\bar{\alpha} = \{1, 1 - \alpha\}$ because of its computational efficiency. Here, there is no need to approximate the action of $\mathbb{A}^{-1}$, thus we solve only $k + 1$ systems instead of $2k$ for all the other partitions.

According to (10), we need to sequentially solve k linear systems with sparse matrices. Moreover, according to (8) we have that

$$\kappa(\lambda_1\mathbb{I} - d_i\mathbb{A}) = \frac{\lambda_1 - d_i\lambda_n}{\lambda_1 - d_i\lambda_1} \quad \Longrightarrow \\ \kappa(\mathbb{A}) < \kappa(\lambda_1\mathbb{I} - d_2\mathbb{A}) < \cdots < \kappa(\lambda_1\mathbb{I} - d_k\mathbb{A}), \ \forall\alpha \in (0, 2).$$

When $\alpha \in (0, 1)$, we have that $\lambda_1\mathbb{I} - d_1\mathbb{I}$ is an SPD matrix with $\kappa(\mathbb{A}) < \kappa(\lambda_1\mathbb{I} - d_1\mathbb{A}) < \kappa(\lambda_1\mathbb{I} - d_2\mathbb{A})$. When $\alpha \in (1, 2)$, we have that $d_1\mathbb{A} - \lambda_1\mathbb{I}$ is an SPD matrix with $\kappa(\mathbb{A}) < \kappa(d_1\mathbb{A} - \lambda_1\mathbb{I})$, and in this case we change the term $(\lambda_1\mathbb{I} - \zeta_1\mathbb{A})(\lambda_1\mathbb{I} - d_1\mathbb{A})^{-1}$ in (10) to $(\zeta_1\mathbb{A} - \lambda_1\mathbb{I})(d_1\mathbb{A} - \lambda_1\mathbb{I})^{-1}$. Thus, all the linear systems we solve numerically are related to sparse and SPD matrices.

We consider two different choices for the right-hand-side $\mathbf{f}$:

$$\mathbf{f_1} = \mathbf{1} \quad \text{and} \quad \mathbf{f_2} = \Psi_1 - \Psi_2 + \Psi_3.$$

For both of them the exact solutions $\mathbf{u_1}$ and $\mathbf{u_2}$ of (1) are computed explicitly via spectral decomposition. We consider two equivalent reformulations of (10),

namely

$$\begin{aligned}\mathbf{u}_{\alpha,k} &= \lambda_1^{-\alpha} c_0 \left[\prod_{i=1}^{k} \left(\mathbb{I} - \frac{\zeta_i}{\lambda_1}\mathbb{A}\right)\left(\mathbb{I} - \frac{d_i}{\lambda_1}\mathbb{A}\right)^{-1}\right]\mathbf{f} \\ &=: \lambda_1^{-\alpha} c_0 \left[\prod_{i=1}^{k} \left(\mathbb{I} - \bar{\zeta}_i\mathbb{A}\right)\left(\mathbb{I} - \bar{d}_i\mathbb{A}\right)^{-1}\right]\mathbf{f}, \quad \bar{\zeta}_i = \frac{\zeta_i}{\lambda_1},\ \bar{d}_i = \frac{d_i}{\lambda_1},\end{aligned} \tag{14}$$

and

$$\begin{aligned}\mathbf{u}_{\alpha,k} &= \lambda_1^{-\alpha} c_0 \left[\prod_{i=1}^{k} \left(\frac{\zeta_i}{d_i}\mathbb{I} + \left(1 - \frac{\zeta_i}{d_i}\right)\left(\mathbb{I} - \frac{d_i}{\lambda_1}\mathbb{A}\right)^{-1}\right)\right]\mathbf{f} \\ &= \lambda_1^{-\alpha} c_0 \left[\prod_{i=1}^{k} \left(\frac{\bar{\zeta}_i}{\bar{d}_i}\mathbb{I} + \left(1 - \frac{\bar{\zeta}_i}{\bar{d}_i}\right)\left(\mathbb{I} - \bar{d}_i\mathbb{A}\right)^{-1}\right)\right]\mathbf{f}.\end{aligned} \tag{15}$$

The numerical computations are performed in MATLAB with the standard explicit solver for linear systems. The zeros, poles, and the theoretical errors are computed using the BRASIL software [13].

In Table 3, Table 4, and Table 5 we compare formulas (10) and (14) for $\mathbf{f} = \mathbf{f_1}$ to study the numerical stability of the MATLAB standard explicit solver, as well as to check the robustness of (13). For each of the considered choices for α and k, we compare the numerically computed corresponding relative error ratios

$$\frac{\lambda_1^{\alpha}\|\mathbf{u} - \mathbf{u}_{\bar{\alpha},k}\|_2}{\left(1 - \prod_{1=1}^{n}(1 - E_{\alpha_i,k})\right)\|\mathbf{f}\|_2} \tag{16}$$

and their values with $|\bar{f}_1|$. In our setting, we have

$$\bar{f}_{1,1} \approx 0.9003, \quad \text{respectively} \quad \bar{f}_{2,1} = \frac{1}{\sqrt{3}} \approx 0.5773.$$

According to (13), the numbers should be in the interval $\left[|\bar{f}_1|, 1\right]$. Whenever they are not, this is an indication that either the order of the linear system solver error significantly affects the computations or the poles and zeros of the BURA solution, respectively the BURA error $E_{\bar{\alpha},k}$ are inaccurately computed. In both cases, the corresponding results are unreliable.

We observe a better numerical stability when (14) is applied. The reason is that after the normalization, the order of the elements of the matrices we compute with decreases. The computational inaccuracies increase with α and k. When product BURA solutions are used, the relative error ratios are practically the same as $|\bar{f}_1|$, validating our observation from the end of Section 2. The numerical computations with (14) behave unstably when for a certain α_i and k the theoretical error $E_{\alpha_i,k}$ from Table 1 reaches order of magnitude -8. This is the case for $\alpha = 0.875$ and $k \geq 9$; $\alpha = 1.25$ and $k \geq 7$; $\alpha = 1.5$ and $k \geq 6$; $\alpha = 1.75$ and $k \geq 6$. The computational issues for (10) start an order

TABLE 3. Relative error ratio for $\mathbf{f} = \mathbf{f}_1$ and $\alpha = 1.25$ with respect to (7) and the practical error estimate (13).

k	$\alpha_1 = 1.25$		$\alpha_1 = 1, \alpha_2 = 0.25$		$\alpha_1 = 0.75, \alpha_2 = 0.5$		$\alpha_{1,2} = 0.675$		$\alpha_{1,2,3,4,5} = 0.25$	
	(10)	(14)	(10)	(14)	(10)	(14)	(10)	(14)	(10)	(14)
1	0.953	0.952	0.901		0.902		0.902		0.902	
2	0.962		0.900		0.901		0.901		0.900	
3	0.984	0.974	0.901		0.901		0.904		0.901	
4	0.932	0.926	0.901		0.903	0.901	0.902	0.901	0.901	
5	0.884	0.954	0.900		0.906	0.900	0.912	0.903	0.900	
6	0.974	0.908	0.900		0.910	0.901	0.924	0.903	0.901	
7	7.522	0.818	0.901		0.917	0.902	0.936	0.901	0.901	
8	65.41	0.515	0.901		0.930	0.902	0.907	0.905	0.901	
9	158.4	1.798	0.901		0.955	0.904	0.900	0.911	0.901	
10	97.35	8.159	0.902	0.900	0.921	0.908	0.999	0.914	0.902	0.900

TABLE 4. Relative error ratio for $\mathbf{f} = \mathbf{f}_1$ and $\alpha = 1.5$ with respect to (7) and the practical error estimate (13).

k	$\alpha_1 = 1.5$		$\alpha_1 = 1, \alpha_2 = 0.5$		$\alpha_{1,2} = 0.75$		$\alpha_{1,2,3} = 0.5$		$\alpha_{1,2,3,4,5,6} = 0.25$	
	(10)	(14)	(10)	(14)	(10)	(14)	(10)	(14)	(10)	(14)
1	0.9429	0.9428	0.9005		0.9019	0.9020	0.9004		0.9009	
2	0.9401	0.9387	0.9007		0.9008	0.9007	0.9007		0.9003	
3	0.9722	0.9584	0.9003	0.9005	0.9025	0.9017	0.9003	0.9005	0.9005	
4	1.0310	0.9784	0.9016	0.9009	0.9087	0.9011	0.9016	0.9009	0.9004	0.9005
5	0.2478	0.9157	0.9042	0.9004	0.9239	0.9025	0.9042	0.9004	0.9002	0.9003
6	2.8965	0.8700	0.9095	0.9009	0.9083	0.9020	0.9094	0.9008	0.9001	0.9004
7	7.1575	0.6067	0.9186	0.9014	0.8814	0.9046	0.9183	0.9012	0.9007	0.9005
8	22.676	1.4584	0.9273	0.9017	0.9914	0.9137	0.9266	0.9012	0.9008	0.9005
9	100.31	6.6371	0.9518	0.9029	1.0308	0.9437	0.9508	0.9015	0.9008	0.9004
10	689.20	27.157	0.9148	0.9057	1.0713	0.9895	0.9128	0.9036	0.9020	0.9004

TABLE 5. Relative error ratio for $\mathbf{f} = \mathbf{f}_1$ and $\alpha = 1.75$ with respect to (7) and the practical error estimate (13).

k	$\alpha_1 = 1.75$		$\alpha_1 = 0.75, \alpha_2 = 1$		$\alpha_1 = 1, \alpha_2 = 0.75$		$\alpha_{1,2} = 0.875$		$\alpha_{1,2,3,4,5,6,7} = 0.25$	
	(10)	(14)	(10)	(14)	(10)	(14)	(10)	(14)	(10)	(14)
1	0.9372	0.9373	0.9009		0.9009		0.9014		0.9006	
2	0.9343	0.9321	0.9006	0.9004	0.9006	0.9004	0.9011	0.9010	0.9003	
3	0.9161	0.9264	0.9016	0.9008	0.9016	0.9008	0.9015	0.9005	0.9004	
4	0.7941	0.9437	0.9084	0.9008	0.9084	0.9008	0.9227	0.9019	0.9003	0.9004
5	1.4346	0.9486	0.9232	0.9018	0.9231	0.9017	0.9085	0.9018	0.9002	0.9003
6	3.3468	0.6627	0.9095	0.9033	0.9093	0.9030	0.2041	0.9060	0.9001	0.9003
7	14.033	1.5968	0.8837	0.9063	0.8835	0.9066	0.6843	0.9228	0.9005	0.9004
8	58.944	10.171	1.0006	0.9220	1.0003	0.9207	3.1998	0.9737	0.9006	0.9004
9	102.28	38.922	1.0515	0.9655	1.0497	0.9645	9.1176	1.1124	0.9008	0.9004
10	15543.5	174.32	1.1337	1.0493	1.1236	1.0509	21.396	1.5432	0.9020	0.9004

of magnitude earlier, e.g., when $E_{\alpha_i,k} \sim O(10^{-7})$. The problem is the double precision arithmetic used by the direct MATLAB solver, which affects the order of the residuals, related to each of the $2kn$ systems of linear equations.

Furthermore, we observe that the ratio (16) is closer to $|\bar{f}_1|$ when smaller α_i's are used in the partition of $\bar{\alpha}$. The reason is the clustering of the extreme points of $\varepsilon_{\alpha_i,k}(t)$ at zero (see [14]) which is stronger, when α_i is smaller. Thus, the ratio (16) tends to $|\bar{f}_1|$ when $\max_i \alpha_i$ tends to zero.

Our numerical experiments show that the most computationally stable reformulation of (10) is (15). Here, apart from the performed λ_1-normalization, all matrix-vector multiplications corresponding to the numerator terms of the BURA element are avoided, due to extracting the integer part from the rational multipliers $(t-\zeta_i)/(t-d_i)$ in (9). The numerically computed relative errors $\|\mathbf{u}-\mathbf{u}_{\alpha,k}\|_2/\|\mathbf{f}\|_2$ for $k \in \{4,5,6,7\}$ are documented in Table 6. As expected, the smallest numbers come from the direct BURA solution, since it is the best approximant among the whole class of (k,k)-rational approximations. However, the half-splitting product BURA solution $\bar{\alpha} = \{\alpha/2, \alpha/2\}$ provides the smallest relative errors among all product BURA solutions and in the same time the matrix $\lambda_1^{-\alpha} r_{\bar{\alpha},k}(\lambda_1 \mathbb{A}^{-1})$ is an SPD. For $\alpha = 1.75$ we have respectively

$$\begin{aligned}
\frac{\|\mathbf{u_1}-\mathbf{u_1}_{,1.75,8}\|_2}{\|\mathbf{f_1}\|_2} &= 2.36\cdot 10^{-10} > 1.48\cdot 10^{-10} = \frac{\|\mathbf{u_1}-\mathbf{u_1}_{,1.75,7}\|_2}{\|\mathbf{f_1}\|_2},\\
\frac{\|\mathbf{u_2}-\mathbf{u_2}_{,1.75,8}\|_2}{\|\mathbf{f_2}\|_2} &= 2.41\cdot 10^{-10} > 2.34\cdot 10^{-10} = \frac{\|\mathbf{u_2}-\mathbf{u_2}_{,1.75,7}\|_2}{\|\mathbf{f_2}\|_2},
\end{aligned}$$

which contradicts with the monotonicity of $E_{\alpha,k}$ with respect to α. This is another indicator for strong computational issues for the direct BURA method when $\alpha = 1.75$ and $k \geq 7$.

TABLE 6. ℓ_2 Relative Error $\|\mathbf{u}-\mathbf{u}_{\alpha,k}\|_2/\|\mathbf{f}\|_2$ for $\mathbf{f} = \mathbf{f_{1,2}}$, where $\mathbf{u}_{\alpha,k}$ is computed with (15)

α	$\alpha = \sum \alpha_i$	$k=4$		$k=5$		$k=6$		$k=7$	
		$\mathbf{f_1}$	$\mathbf{f_2}$	$\mathbf{f_1}$	$\mathbf{f_2}$	$\mathbf{f_1}$	$\mathbf{f_2}$	$\mathbf{f_1}$	$\mathbf{f_2}$
1.25	$\alpha_1 = 1.25$	5.20E-07	3.59E-07	1.04E-07	1.03E-07	2.31E-08	1.63E-08	5.73E-09	5.38E-09
	$\alpha_{1,2} = \{0.25, 1\}$	2.87E-04	1.86E-04	1.41E-04	9.26E-05	7.37E-05	4.88E-05	4.05E-05	2.66E-05
	$\alpha_{1,2} = \{0.5, 0.75\}$	4.31E-05	2.79E-05	1.53E-05	9.96E-06	6.01E-06	4.05E-06	2.54E-06	1.71E-06
	$\alpha_{1,2} = 0.675$	2.89E-05	1.88E-05	9.34E-06	6.58E-06	3.34E-06	2.30E-06	1.29E-06	8.28E-07
	$\alpha_{1,2,3,4,5} = 0.25$	1.42E-03	9.19E-04	7.00E-04	4.60E-04	3.67E-04	2.43E-04	2.02E-04	1.33E-04
1.5	$\alpha_1 = 1.5$	1.87E-07	1.88E-07	2.84E-08	1.90E-08	5.41E-09	5.26E-09	1.00E-09	9.20E-10
	$\alpha_{1,2} = \{0.5, 1\}$	2.14E-05	1.39E-05	7.81E-06	5.03E-06	3.12E-06	2.06E-06	1.34E-06	8.86E-07
	$\alpha_{1,2} = 0.75$	5.79E-06	3.92E-06	1.67E-06	1.13E-06	5.38E-07	3.45E-07	1.90E-07	1.28E-07
	$\alpha_{1,2,3} = 0.5$	6.42E-05	4.16E-05	2.34E-05	1.51E-05	9.37E-06	6.18E-06	4.01E-06	2.66E-06
	$\alpha_{1,2,3,4,5,6} = 0.25$	9.56E-04	6.17E-04	4.73E-04	3.07E-04	2.48E-04	1.62E-04	1.37E-04	8.88E-05
1.75	$\alpha_1 = 1.75$	3.54E-08	3.17E-08	4.86E-09	4.38E-09	6.16E-10	5.49E-10	1.48E-10	2.34E-10
	$\alpha_{1,2} = \{0.75, 1\}$	1.63E-06	1.07E-06	4.70E-07	3.09E-07	1.52E-07	9.75E-08	5.37E-08	3.53E-08
	$\alpha_{1,2} = \{1, 0.75\}$	1.63E-06	1.07E-06	4.70E-07	3.09E-07	1.52E-07	9.75E-08	5.37E-08	3.53E-08
	$\alpha_{1,2} = 0.875$	8.98E-07	6.05E-07	2.33E-07	1.50E-07	6.85E-08	4.53E-08	2.24E-08	1.51E-08
	$\alpha_{1,2,3,4,5,6,7} = 0.25$	6.28E-04	4.04E-04	3.11E-04	2.01E-04	1.63E-04	1.06E-04	9.00E-05	5.80E-05

Finally, applying the BURA realization (15), in Table 7, Table 8, and Table 9 we document the relative error ratios (16) for both considered right-hand-sides $\mathbf{f} = \mathbf{f_{1,2}}$ and $\alpha = 1.25, 1.5, 1.75$, respectively. Again, we observe

TABLE 7. Relative error ratios (16) for $\mathbf{f} = \mathbf{f_{1,2}}$, $\mathbf{u}_{\bar{\alpha},k}$ are computed via (15), and $\alpha = 1.25$.

k	$\alpha_1 = 1.25$		$\alpha_1 = 1, \alpha_2 = 0.25$		$\alpha_1 = 0.75, \alpha_2 = 0.5$		$\alpha_{1,2} = 0.675$		$\alpha_{1,2,3,4,5} = 0.25$	
	$\mathbf{f_1}$	$\mathbf{f_2}$	$\mathbf{f_1}$	$\mathbf{f_2}$	$\mathbf{f_1}$	$\mathbf{f_2}$	$\mathbf{f_1}$	$\mathbf{f_2}$	$\mathbf{f_1}$	$\mathbf{f_2}$
1	0.9525	0.9061	0.9008	0.5943	0.9015	0.5892	0.9024	0.5906	0.9017	0.6184
2	0.9616	0.7597	0.9004	0.5895	0.9010	0.6010	0.9009	0.6225	0.9004	0.5931
3	0.9741	0.9666	0.9009	0.5810	0.9011	0.6064	0.9036	0.6134	0.9009	0.5809
4	0.9260	0.6392	0.9008	0.5844	0.9014	0.5848	0.9008	0.5867	0.9007	0.5844
5	0.9568	0.9481	0.9004	0.5921	0.9004	0.5854	0.9029	0.6357	0.9004	0.5921
6	0.9397	0.6624	0.9005	0.5960	0.9013	0.6076	0.9033	0.6224	0.9005	0.5959
7	0.9142	0.8575	0.9009	0.5923	0.9019	0.6078	0.9012	0.5791	0.9009	0.5924
8	0.7639	0.7334	0.9009	0.5843	0.9015	0.5867	0.9047	0.6081	0.9009	0.5843
9	0.4165	0.7204	0.9006	0.5787	0.9027	0.5808	0.9079	0.6414	0.9006	0.5787
10	1.1664	1.6150	0.9004	0.5801	0.9057	0.5998	0.9102	0.6298	0.9004	0.5800

TABLE 8. Relative error ratios (16) for $\mathbf{f} = \mathbf{f_{1,2}}$, $\mathbf{u}_{\bar{\alpha},k}$ are computed via (15), and $\alpha = 1.50$.

k	$\alpha_1 = 1.5$		$\alpha_1 = 1, \alpha_2 = 0.5$		$\alpha_{1,2} = 0.75$		$\alpha_{1,2,3} = 0.5$		$\alpha_{1,2,3,4,5,6} = 0.25$	
	$\mathbf{f_1}$	$\mathbf{f_2}$	$\mathbf{f_1}$	$\mathbf{f_2}$	$\mathbf{f_1}$	$\mathbf{f_2}$	$\mathbf{f_1}$	$\mathbf{f_2}$	$\mathbf{f_1}$	$\mathbf{f_2}$
1	0.9428	0.8862	0.9005	0.5850	0.9020	0.5868	0.9004	0.5865	0.9009	0.6018
2	0.9387	0.7487	0.9007	0.5865	0.9007	0.6123	0.9007	0.5864	0.9003	0.5858
3	0.9585	0.7426	0.9005	0.5954	0.9017	0.5843	0.9005	0.5953	0.9005	0.5786
4	0.9788	0.9850	0.9009	0.5839	0.9011	0.6101	0.9009	0.5839	0.9005	0.5805
5	0.9195	0.6140	0.9004	0.5804	0.9024	0.6100	0.9004	0.5804	0.9003	0.5847
6	0.9060	0.8802	0.9009	0.5946	0.9015	0.5785	0.9008	0.5945	0.9004	0.5865
7	0.7570	0.6967	0.9013	0.5970	0.9044	0.6087	0.9011	0.5969	0.9005	0.5844
8	0.4707	0.8818	0.9014	0.5843	0.9102	0.6250	0.9009	0.5840	0.9005	0.5803
9	2.2802	2.8424	0.9023	0.5793	0.9294	0.6226	0.9012	0.5785	0.9004	0.5778
10	12.313	12.669	0.9047	0.5896	0.9653	0.6484	0.9024	0.5879	0.9003	0.5787

TABLE 9. Relative error ratios (16) for $\mathbf{f} = \mathbf{f_{1,2}}$, $\mathbf{u}_{\bar{\alpha},k}$ are computed via (15), and $\alpha = 1.75$.

k	$\alpha_1 = 1.75$		$\alpha_1 = 0.75, \alpha_2 = 1$		$\alpha_1 = 1, \alpha_2 = 0.75$		$\alpha_{1,2} = 0.875$		$\alpha_{1,2,3,4,5,6,7} = 0.25$	
	$\mathbf{f_1}$	$\mathbf{f_2}$	$\mathbf{f_1}$	$\mathbf{f_2}$	$\mathbf{f_1}$	$\mathbf{f_2}$	$\mathbf{f_1}$	$\mathbf{f_2}$	$\mathbf{f_1}$	$\mathbf{f_2}$
1	0.9373	0.8463	0.9009	0.5808	0.9009	0.5808	0.9014	0.5873	0.9006	0.5921
2	0.9321	0.8128	0.9004	0.5949	0.9004	0.5949	0.9010	0.5967	0.9003	0.5819
3	0.9264	0.6129	0.9008	0.5797	0.9008	0.5797	0.9005	0.5849	0.9004	0.5778
4	0.9441	0.8461	0.9007	0.5935	0.9008	0.5935	0.9018	0.6070	0.9004	0.5788
5	0.9516	0.8579	0.9017	0.5930	0.9017	0.5929	0.9016	0.5799	0.9003	0.5810
6	0.7241	0.6459	0.9026	0.5794	0.9026	0.5792	0.9052	0.5987	0.9003	0.5819
7	0.8996	1.4212	0.9065	0.5952	0.9066	0.5952	0.9176	0.6195	0.9004	0.5807
8	6.6123	6.7402	0.9185	0.6123	0.9185	0.6111	0.9605	0.6459	0.9004	0.5787
9	23.884	25.097	0.9519	0.6355	0.9511	0.6271	1.0586	0.7297	0.9004	0.5775
10	112.56	126.60	1.0190	0.6917	1.0160	0.6732	1.3461	0.9910	0.9003	0.5780

that for both the cases the relative error ratios for all product BURA solutions are close to the normalized coefficient $\bar{f}_1$ of $\mathbf{f}$, corresponding to the contribution

of Ψ_1 in its spectral decomposition. For $\mathbf{f_1}$ this is not that impressive, as the contribution of Ψ_1 is quite significant ($\bar{f}_{1,1} \approx 0.9003$), while the numbers are uniformly bounded from above by 1. In this case, the behavior of the direct BURA solution does not differ much from the one of the product BURAs. For $\mathbf{f_2}$, however, the difference in the behaviours of the direct BURA solution and the product BURA solutions is much better illustrated. In the direct setting, the contributions of all eigenvectors of $\mathbb{A}$ in the spectral decomposition of the right-hand-side are equally weighted. In the product setting, due to the additional multiplications with powers of t_i in the derivation of the theoretical error estimate, the contribution of Ψ_i decreases towards $\prod_{j=1}^{n} E_{\alpha_j,k}$ when i tends to N. On our particular example, even though the contributions of the first three eigenvectors are equal ($\bar{f}_{2,1} = \bar{f}_{2,2} = \bar{f}_{2,3} = 1/\sqrt{3}$), the effect of $\Psi_{2,3}$ on the approximation ℓ_2 error is almost negligent. On one hand, this can be explained by the sparse representation of $\mathbf{f_2}$ in the basis $\{\Psi_i\}_1^N$ and that the chances for some of $t_{2,3}$ to hit another extreme point of the error function $\varepsilon_{\bar{\alpha},k}$ are slim, but on the other hand those are the two most significant coefficients after $\bar{f}_{2,1}$ in it, thus the example is illustrative enough.

4. **Discussion**

Even though the product BURA method is by far not optimal with respect to minimizing the approximation error $\|\mathbf{u}-\mathbf{u}_{\bar{\alpha},k}\|_2$ and in this aspect it can not compete with the direct BURA method, the SPD property of the matrix $\mathbb{A}^{-1}$ is inherited by the approximant $\lambda_1^{-\alpha} r_{\bar{\alpha},k}(\lambda_1 \mathbb{A}^{-1})$ for all choices of the splitting $\bar{\alpha} = \{\alpha_1, \ldots, \alpha_n\}$ whenever $\alpha_i \in (0, 1]$ for every $i = 1, \ldots, n$. Moreover, due to the decaying contributions into the error estimate of the eigenvectors that correspond to larger eigenvalues, $\|\mathbf{u} - \mathbf{u}_{\bar{\alpha},k}\|_2$ is dominated by

$$\lambda_1^{-\alpha}\left(1 - \prod_{j=1}^{n}\left(1 - E_{\alpha_j,k}\right)\right)|\bar{f}_1| \cdot \|\mathbf{f}\|_{\mathbf{2}}.$$

Thus, the error is robust with respect to the choices of α and k, which is illustrated in Tables 7 – 9. As a byproduct, this allows for a reliable a priori error analysis. One option is to choose a small k (even $k = 1$ seems to work!) and a suitable $\bar{\alpha}$, for which the error term $\|\mathbf{u}-\mathbf{u}_{\bar{\alpha},k}\|_2$ can somehow be estimated, and the derived bounds can be applied to all other choices of k and $\bar{\alpha}$. Another option is to find a lower bound for the smallest eigenvalue λ_1 of $\mathbb{A}$ and then a good approximant for the first eigenvector Ψ_1. Now,

$$\bar{f}_1 = \frac{\langle \mathbf{f}, \Psi_1 \rangle}{\sqrt{\langle \mathbf{f}, \mathbf{f} \rangle}},$$

provides a good guess for (16) and it is not necessary to know anything beforehand for the exact solution $\mathbf{u}$ in order to localize the theoretical value of $\|\mathbf{u} - \mathbf{u}_{\bar{\alpha},k}\|_2$ for all choices of $\alpha \in (1, 2)$ and $k \in \mathbb{N}$.

Applying direct solvers to the linear systems of equations in (10) leads to computational instability for the direct BURA method when α and/or k increases. Efficient iterative solvers need to be taken into account, which will be a subject of future research.

5. Conclusions and future work

In this paper we theoretically and experimentally analyzed a variety of computational issues that arise when numerically solving (1). Since the matrix $\mathbb{A}^{\alpha}$ is dense, it is practically impossible to compute $\mathbb{A}^{-\alpha}$ or even to store in the computer memory the entries of $\mathbb{A}^{\alpha}$ when the size of the matrix is large. Therefore, the exact solution $\mathbf{u}$ can not be explicitly computed and needs to be approximated. The direct BURA method provides optimal error estimates within the class of rational approximations, but fails to inherit the SPD property of $\mathbb{A}$. The proposed product BURA method gives rise to ℓ_2 error of lower order, but on the other hand it inherits the SPD property, the relative error ratio (16) is robust with respect to both α and k, and it can be well-approximated provided we have an a priori information on the smallest eigenvalue of $\mathbb{A}$ and its corresponding eigenvector. Due to the application of a direct MATLAB solver for the systems of linear equations that arise from (10) and the usage of double precision arithmetic, the numerical derivation of the BURA solution $\mathbf{u}_{\alpha,k}$ becomes more and more challenging with the increment of k and α. Among all investigated reformulations of (10), the one in (15) proves to be computationally the most stable one.

For the case $\alpha > 2$ the above analysis is non-applicable. In [12] the author proved that $\alpha - \lfloor \alpha \rfloor$ of the poles and zeros of $r_{\alpha,k}$ are always negative and interlacing. This allowed us to conclude that for $\alpha \in (1, 2)$ the remaining zero ζ_1 and pole d_1 are positive real numbers. When $\alpha > 2$ complex poles and zeros appear, therefore (9) does not hold true, respectively the formula (10) becomes more complicated. We leave this case for future work.

Acknowledgements

The work was partially supported by the Bulgarian National Science Fund under grant No. KP-06-N72/2. The first author was partially supported by the Bulgarian Ministry of Education and Science under the National Research Programme "Young scientists and postdoctoral students -2" approved by DCM 206 / 07.04.2022.

References

[1] A. Bonito, J. Pasciak, Numerical approximation of fractional powers of elliptic operators. *Math. Comput.*, **84** (2015), 2083–2110.

[2] A. Bonito, W. Lei, and J.E. Pasciak, On sinc quadrature approximations of fractional powers of regularly accretive operators. *J. of Numerical Mathematics*, **27**, No 2 (2019), 57–68.

[3] B. Duan, R.D. Lazarov, and J.E. Pasciak, Numerical approximation of fractional powers of elliptic operators. *IMA Journal of Numerical Analysis*, **40**, No 3 (2019), 1746–1771.

[4] P.N. Vabishchevich, An approximate representation of a solution to fractional elliptical BVP via solution of parabolic IVP. *J. Comput. Appl. Math.*, **391**, 13460 (2021); doi:10.1016/j.cam.2021.113460.

[5] S. Harizanov, R. Lazarov, S. Margenov, P. Marinov, J. Pasciak, Analysis of numerical methods for spectral fractional elliptic equations based on the best uniform rational approximation. *J. Comput. Phys.*, **408**, 109285 (2020); doi:10.1016/j.jcp.2020.109285.

[6] L. Aceto and P. Novati, Fast and accurate approximations to fractional powers of operators. *IMA Journal of Numerical Analysis*, **42**, No 2 (2021), 1598–1622.

[7] H. Alzahrani, G. Turkiyyah, O. Knio, and D. Keyes, Space-fractional diffusion with variable order and diffusivity: Discretization and direct solution strategies. *Commun. on Appl. Math. and Computation*, **4**, No 4 (2022), 1416–1440.

[8] S. Harizanov, R. Lazarov, S. Margenov, A survey on numerical methods for spectral space-fractional diffusion problems. *Fract. Calc. Appl. Anal.* **23**, No 6 (2020), 1605–1646; doi:10.1515/fca-2020-0080.

[9] C. Hofreither, A unified view of some numerical methods for fractional diffusion. *Comput. Math. Appl.*, **80** (2020), 332–350.

[10] S. Harizanov, R. Lazarov, S. Margenov, P. Marinov, Y. Vutov, Optimal solvers for linear systems with fractional powers of sparse SPD matrices. *Numer. Linear Algebra Appl.* **25**, e2167 (2018); doi:10.1002/nla.2167.

[11] N. Kosturski and S. Margenov, Analysis of BURA and BURA-based approximations of fractional powers of sparse SPD matrices. *Fract. Calc. Appl. Anal.*, **27**, No 2 (2024), 706–724; doi:10.1007/s13540-024-00256-6.

[12] H. Stahl, Best uniform rational approximation of x^α on $[0,1]$. *Acta Math.*, **190** (2003), 241–306.

[13] Software BRASIL. Available online:
https://baryrat.readthedocs.io/en/latest/#baryrat.brasil
(accessed on 25 June 2021).

[14] E.B. Saff, H. Stahl, Asymptotic distribution of poles and zeros of best rational approximants to x^α on $[0,1]$. In: *Topics in Complex Analysis. Banach Center Publications,* Vol. **31**, Institute of Mathematics, Polish Academy of Sciences: Warsaw, Poland, 1995.