Mathematical Biosciences and Engineering

http://www.aimspress.com/journal/mbe

MBE, 20(3): 4782–4797.
DOI: 10.3934/mbe.2023221
Received: 29 November 2022
Revised: 19 December 2022
Accepted: 21 December 2022
Published: 03 January 2023

Research article

Linear barycentric rational collocation method for solving generalized Poisson equations

Jin Li[1,*], **Yongling Cheng**[1], **Zongcheng Li**[1] **and Zhikang Tian**[2]

[1] School of Science, Shandong Jianzhu University, No. 1000 Fengming Road, Lingang Development Zone, Jinan 250101, Shandong, China

[2] Lab Master at School of Computer Science and Technology, Shandong Jianzhu University, No. 1000 Fengming Road, Lingang Development Zone, Jinan 250101, Shandong, China

* **Correspondence:** Email: lijin@lsec.cc.ac.cn.

Abstract: We consider the Poisson equation by collocation method with linear barycentric rational function. The discrete form of the Poisson equation was changed to matrix form. For the basis of barycentric rational function, we present the convergence rate of the linear barycentric rational collocation method for the Poisson equation. Domain decomposition method of the barycentric rational collocation method (BRCM) is also presented. Several numerical examples are provided to validate the algorithm.

Keywords: linear barycentric rational function; collocation method; error functional; poisson equation; barycentric rational function

1. Introduction

Two-dimensional elliptic [1] boundary value problems

$$\frac{\partial}{\partial t}\left(a(t,s)\frac{\partial u(t,s)}{\partial t}\right)+\frac{\partial}{\partial s}\left(b(t,s)\frac{\partial u(t,s)}{\partial s}\right)=f(t,s),(t,s)\in\Omega \tag{1.1}$$

with

$$u(t,s)=g(t,s),\qquad \frac{\partial u(t,s)}{\partial n}=h(t,s),\qquad (t,s)\in\partial\Omega, \tag{1.2}$$

where $\Omega=[a,b]\times[c,d]$ and $f(t,s),g(t,s),h(t,s)$ on Ω, can be used in many scientific areas, such as electrostatics, mechanical engineering, magnetic fields, thermal fields and brain activity detection [2].

Floater [3–5] proposed a rational interpolation scheme. In [6], a linear rational collocation method was presented for the lower regular function. Wang et al. [7–10] successfully solved initial value problems, plane elasticity problems, incompressible plane problems and non-linear problems by collocation method. The linear barycentric rational collocation method (LBRCM) to solve nonlinear parabolic partial differential equations [11], biharmonic equation [12], fractional differential equations [13], telegraph equation [14], Volterra integro-differential equation [15] and heat conduction equation [16] have been studied.

In the following, we introduce the barycentric formula of a one dimensional function. Let

$$p(t) = \sum_{j=1}^{n} L_j(t) f_j, \tag{1.3}$$

and

$$L_j(t) = \frac{\prod\limits_{k=1,k\neq j}^{n} (t - t_k)}{\prod\limits_{k=1,k\neq j}^{n} \left(t_j - t_k\right)}. \tag{1.4}$$

In order to get the barycentric formula, Eq (1.4) is changed into

$$l(t) = (t - t_1)(t - t_2)\cdots(t - t_n) \tag{1.5}$$

and

$$w_j = \frac{1}{\prod_{j\neq k} \left(t_j - t_k\right)}, \quad j = 1, 2, \cdots, n, \tag{1.6}$$

which means $w_j = 1/l'(t_j)$.

$$L_j(t) = l(t) \frac{w_j}{t - t_j}, \quad j = 1, 2, \cdots, n. \tag{1.7}$$

For Eq (1.2), we get

$$p(t) = l(t) \sum_{j=1}^{n} \frac{w_j}{t - t_j} f_j \tag{1.8}$$

which means

$$p(t) = \frac{\sum\limits_{j=1}^{n} \frac{w_j}{t - t_j} f_j}{\sum\limits_{j=1}^{n} \frac{w_j}{t - t_j}}. \tag{1.9}$$

For the case uniform partition, we get

$$w_j = (-1)^{n-j} C_n^j = (-1)^{n-j} \frac{n!}{j!(n-j)!}. \tag{1.10}$$

For the case the nonuniform partition is chosen, we take the second Chebyshev point [7] a

$$t_j = \cos \frac{(j-1)\pi}{n-1}, \quad j = 1, \cdots, n \tag{1.11}$$

with

$$w_j = (-1)^j \delta_j, \quad \delta_j = \begin{cases} 1/2, j = 1, n \\ 1, otherwise \end{cases} \tag{1.12}$$

For the barycentric rational function, we first set

$$r(t) = \frac{\sum_{i=1}^{n-d} \lambda_i(t) p_i(t)}{\sum_{i=1}^{n-d} \lambda_i(t)} \tag{1.13}$$

where

$$\lambda_i(t) = \frac{(-1)^i}{(t-t_i)\cdots(t-t_{i+d})} \tag{1.14}$$

and

$$p_i(t) = \sum_{k=i}^{i+d} \prod_{j=i, j\neq k}^{i+d} \frac{t-t_j}{t_k-t_j} f_k. \tag{1.15}$$

Combining (1.14) and (1.15), we have

$$\sum_{i=1}^{n-d} \lambda_i(t) p_i(t) = \sum_{i=1}^{n-d} (-1)^i \sum_{k=i}^{i+d} \frac{1}{t-t_k} \prod_{i, j\neq k}^{i+d} \frac{1}{t_k-t_j} f_k = \sum_{k=1}^{n} \frac{w_k}{t-t_k} f_k \tag{1.16}$$

where

$$w_k = \sum_{i\in J_k} (-1)^i \prod_{j=i, j\neq k}^{i+d} \frac{1}{t_k-t_j} \tag{1.17}$$

and $J_k = \{i \in I; k-d \leqslant i \leqslant k\}$. By taking $p(t) = 1$, we have

$$1 = \sum_{k=i}^{i+d} \prod_{j=i, j\neq k}^{i+d} \frac{t-t_j}{t_k-t_j} \tag{1.18}$$

and then have

$$\sum_{i=1}^{n-d} \lambda_i(t) = \sum_{k=1}^{n} \frac{w_k}{t - t_k}. \tag{1.19}$$

Combining (1.16), (1.19) and (1.13), we get

$$r(t) = \frac{\sum_{j=1}^{n} \frac{w_j}{t - t_j} f_j}{\sum_{j=1}^{n} \frac{w_j}{t - t_j}} \tag{1.20}$$

where ω_j is defined as in (1.17).

In this paper, based on linear barycentric rational interpolation of one dimension, we construct a barycentric rational interpolation of a two-dimensional Poisson equation. In order to get the discrete linear equation of a two-dimensional Poisson equation, the equidistant nodes and second kind of Chebyshev points were chosen as collocation point. For the general area, a domain decomposition method of the barycentric rational collocation method is also presented.

2. Differentiation matrices of Poisson equation

Let $a = t_1 < \cdots < t_m = b, h = \frac{b-a}{m}$ and $c = s_1 < \cdots < s_n = d, \tau = \frac{d-c}{n}$ with mesh point $(t_i, s_j), i = 1, 2, \cdots, m; j = 1, 2, \cdots, n$. Then, we have

$$u(t_i, s) = u_i(s), \tag{2.1}$$

on $[a, b]$, and

$$u(t, s) = \sum_{i=1}^{m} \sum_{j=1}^{n} L_i(t) M_j(s) u_{ij} \tag{2.2}$$

where

$$L_i(t) = \frac{\frac{w_i}{t - t_i}}{\sum_{j=1}^{n} \frac{w_j}{t - t_j}} \tag{2.3}$$

and

$$M_j(s) = \frac{\frac{v_j}{s - s_j}}{\sum_{j=1}^{n} \frac{v_j}{s - s_j}}. \tag{2.4}$$

w_i, v_j is the weight function defined as (1.6) or (1.17); see [17].

We have

$$\begin{bmatrix} \sum_{k=1}^{n} M_k''(s)\, u_{1k} \\ \vdots \\ \sum_{k=1}^{n} M_k''(s)\, u_{mk} \end{bmatrix} + \begin{bmatrix} C_{11}^{(2)} & \cdots & C_{1m}^{(2)} \\ \vdots & & \vdots \\ C_{m1}^{(2)} & \cdots & C_{mn}^{(2)} \end{bmatrix} \begin{bmatrix} \sum_{k=1}^{n} M_k(s)\, u_{1k} \\ \vdots \\ \sum_{k=1}^{n} M_k(s)\, u_{mk} \end{bmatrix} = \begin{bmatrix} f_1(s) \\ \vdots \\ f_m(s) \end{bmatrix}.$$

Then, we have

$$\begin{bmatrix} \sum_{k=1}^{n} M_k''\left(s_j\right) u_{1k} \\ \vdots \\ \sum_{k=1}^{n} M_k''\left(s_j\right) u_{mk} \end{bmatrix} + \begin{bmatrix} C_{11}^{(2)} & \cdots & C_{1m}^{(2)} \\ \vdots & & \vdots \\ C_{m1}^{(2)} & \cdots & C_{mn}^{(2)} \end{bmatrix} \begin{bmatrix} \sum_{k=1}^{n} M_k\left(s_j\right) u_{1k} \\ \vdots \\ \sum_{k=1}^{n} M_k\left(s_j\right) u_{mk} \end{bmatrix} = \begin{bmatrix} f_1\left(s_j\right) \\ \vdots \\ f_m\left(s_j\right) \end{bmatrix},$$

where $C_{ij}^{(2)} = L_i''(t_j)$, and

$$C_{ij}^{(2)} = \begin{cases} 2L_i'(t_j)\left(L_i'(t_i) - \dfrac{1}{t_i - t_j}\right), & j \neq i \\ -\displaystyle\sum_{i \neq j} L_i''(t_j), & j = i. \end{cases} \tag{2.5}$$

$u_i = [u_{i1}, u_{i2}, \cdots u_{in}]^{\mathrm{T}}$, $f_i = [f_{i1}, f_{i2}, \cdots, f_{in}]^{\mathrm{T}} = [f_i(s_i), f_i(s_2), \cdots, f_i(s_n)]^{\mathrm{T}}$. With the help of the matrix form, the linear equation systems can be written as

$$\left(I_m \otimes D^{(2)}\right) \begin{bmatrix} u_1 \\ \vdots \\ u_m \end{bmatrix} + \left(C^{(2)} \otimes I_n\right) \begin{bmatrix} u_1 \\ \vdots \\ u_m \end{bmatrix} = \begin{bmatrix} f_1 \\ \vdots \\ f_m \end{bmatrix} \tag{2.6}$$

and $D_{ij}^{(2)} = M_i''(s_j)$,

$$D_{ij}^{(2)} = \begin{cases} 2M_i'(s_j)\left(M_i'(s_i) - \dfrac{1}{s_i - s_j}\right), & j \neq i, \\ -\displaystyle\sum_{i \neq j} M_i''(s_j), & j = i. \end{cases} \tag{2.7}$$

Then, we have

$$\left[\left(C^{(2)} \otimes I_n\right) + \left(I_m \otimes D^{(2)}\right)\right] U = F \tag{2.8}$$

and

$$LU = F \tag{2.9}$$

where

$$L = C^{(2)} \otimes I_n + I_m \otimes D^{(2)} \tag{2.10}$$

and $\otimes$ is the Kronecker product of the matrices. The Kronecker product of $A = (a_{ij})_{m\times n}$ and $B = (b_{ij})_{k\times l}$ is defined as

$$A \times B = (a_{ij}B)_{m\cdot k \times n\cdot l} \tag{2.11}$$

where

$$a_{ij}B = \begin{bmatrix} a_{ij}b_{11} & a_{ij}b_{12} & \cdots & a_{ij}b_{1l} \\ a_{ij}b_{21} & a_{ij}b_{22} & \cdots & a_{ij}b_{2l} \\ \vdots & \vdots & & \vdots \\ a_{ij}b_{k1} & a_{ij}b_{k2} & \cdots & a_{ij}b_{kl} \end{bmatrix} \tag{2.12}$$

and the node of tensor is $(t_i, s_j), i = 1, 2, \cdots, m; j = 1, 2, \cdots, n$. Then, matrix A and B can be can be changed to $(m \times n)$ column vectors as

$$t = [t_1, \cdots, t_1, t_2, \cdots, t_2, \cdots, t_m, \cdots, t_m]$$

$$s = [s_1, s_2, \cdots, s_n, s_1, s_2, \cdots, s_n, \cdots, s_1, s_2, \cdots, s_n],$$

and then we get relationship of the partial differential equation and differential matrix as

$$\frac{\partial^{k+l}u}{\partial^k t \partial^l s} = C^{(k)} \otimes D^{(l)}, \forall k, l \in N. \tag{2.13}$$

3. Domain decomposition method of barycentric rational collocation method for Poisson equation

Consider the generalized elliptic boundary value problems as

$$\nabla[\beta(t, s)\nabla u(t, s)] = f(t, s), (t, s) \in \Omega \tag{3.1}$$

with boundary condition

$$u(t, s) = u_0(t, s), (t, s) \in \Gamma \tag{3.2}$$

where $\beta(t, s)$ is the diffusion coefficient, and $\nabla = \left(\frac{\partial}{\partial t}, \frac{\partial}{\partial s}\right)$ is the gradient operator.

Taking the rectangle domain Ω into two sub-rectangle domains $\Omega_i, i = 1, 2$, the boundary of the domain is Γ, and the boundary of $\Omega_i, i = 1, 2$, is Γ_0. Suppose $\beta(t, s) \in C\Omega$ and the interface conditions of Γ_0 are

$$[u]_\Gamma = 0, \quad [(\beta \nabla u) \bullet \mathbf{n}]_\Gamma = 0.$$

Suppose $\beta(t, s)$ is not continuous on Ω and the interface conditions of Γ_0 are

$$[u]_\Gamma = \delta(t, s), \quad [(\beta \nabla u) \bullet \mathbf{n}]_\Gamma = \gamma(t, s).$$

In the following, we take the two sub-domain $\Omega_i, i = 1, 2$, $(t_{1,i}, s_{1,j})$, the function $u_{1,ij} = u(t_{1,i}, s_{1,j})$; $(t_{2,i}, s_{2,j}), i = 1, 2, \cdots, m_2; j = 1, 2, \cdots, n_2$ and $u_{2,ij} = u(t_{2,i}, s_{2,j})$.

On the sub-domain of Ω_1, the barycentrix function is defined as

$$u(t, s) = \sum_{i=1}^{m_1} \sum_{j=1}^{n_1} R_{1,i}(t) R_{1,j}(s) u_{1,ij} \tag{3.3}$$

where $R_{1,i}(t), R_{1,j}(s)$ are defined as (2.3) and (2.4).

Equation (3.1) can be written as

$$\beta\left(\frac{\partial^2 u}{\partial t^2}+\frac{\partial^2 u}{\partial s^2}\right)+\frac{\partial\beta}{\partial t}\frac{\partial u}{\partial t}+\frac{\partial\beta}{\partial s}\frac{\partial u}{\partial s}=f(t,s),(t,s)\in\Omega \tag{3.4}$$

Taking Eq (3.3) into Eq (3.4), we have

$$\begin{aligned}&\beta(t,s)\left(\sum_{i=1}^{m_1}\sum_{j=1}^{n_1}R''_{1,i}(t)R_{1,j}(s)u_{1,ij}+\sum_{i=1}^{m_1}\sum_{j=1}^{n_1}R_{1,i}(t)R''_{1,j}(s)u_{1,ij}\right)\\&+\frac{\partial\beta(t,s)}{\partial t}\sum_{i=1}^{m_1}\sum_{j=1}^{n_1}R'_{1,i}(t)R_{1,j}(s)u_{1,ij}\\&+\frac{\partial\beta(t,s)}{\partial s}\sum_{i=1}^{m_1}\sum_{j=1}^{n_1}R_{1,i}(t)R'_{1,j}(s)u_{1,ij}=f(t,s),(t,s)\in\Omega\end{aligned} \tag{3.5}$$

Taking $(t_{1,i}, s_{1,j})$ on the sub-domain of Ω_1, we have

$$\begin{aligned}&\beta(t_{1,k},s_{1,l})\left(\sum_{i=1}^{m_1}\sum_{j=1}^{n_1}R''_{1,i}(t_{1,k})R_{1,j}(s_{1,l})u_{1,ij}+\sum_{i=1}^{m_1}\sum_{j=1}^{n_1}R_{1,i}(t_{1,k})R''_{1,j}(s_{1,l})u_{1,ij}\right)\\&+\frac{\partial\beta(t_{1,k},s_{1,l})}{\partial t}\sum_{i=1}^{m_1}\sum_{j=1}^{n_1}R'_{1,i}(t_{1,k})R_{1,j}(s_{1,l})u_{1,ij}\\&+\frac{\partial\beta(t_{1,k},s_{1,l})}{\partial s}\sum_{i=1}^{m_1}\sum_{j=1}^{n_1}R_{1,i}(t_{1,k})R'_{1,j}(s_{1,l})u_{1,ij}\\&=f(t_{1,k},s_{1,l}),(t,s)\in\Omega.\end{aligned} \tag{3.6}$$

As we have used

$$\begin{aligned}R_{1,i}(t_{1,k})&=\delta_{ki},\quad R_{1,j}(s_{1,l})=\delta_{lj}\\R'_{1,i}(t_{1,k})&=C^{1(1)}_{ki},\quad R'_{1,j}(s_{1,l})=D^{1(1)}_{lj}\\R''_{1,i}(t_{1,k})&=C^{1(2)}_{ki},\quad R''_{1,j}(s_{1,l})=D^{1(2)}_{lj}\end{aligned} \tag{3.7}$$

Take the notation

$$\begin{aligned}\mathbf{B}=&\operatorname{diag}(\beta(t_{1,1},s_{1,1}),\beta(t_{1,1},s_{1,2}),\cdots,\beta(t_{1,1},s_{1,n_1}),\cdots,\\&\beta(t_{1,m_1},s_{1,1}),\beta(t_{1,m_1},s_{1,2}),\cdots,\beta(t_{1,m_1},s_{1,n_1})),\end{aligned} \tag{3.8}$$

$$\begin{aligned}\mathbf{B}_{11}=&\operatorname{diag}(\beta_t(t_{1,1},s_{1,1}),\beta_t(t_{1,1},s_{1,2}),\cdots,\beta_t(t_{1,1},s_{1,n_1}),\cdots,\\&\beta_t(t_{1,m_1},s_{1,1}),\beta_t(t_{1,m_1},s_{1,2}),\cdots,\beta_t(t_{1,m_1},s_{1,n_1})),\end{aligned} \tag{3.9}$$

$$\begin{aligned}\mathbf{B}_{12}=&\operatorname{diag}(\beta_s(t_{1,1},s_{1,1}),\beta_s(t_{1,1},s_{1,2}),\cdots,\beta_s(t_{1,1},s_{1,n_1}),\cdots,\\&\beta_s(t_{1,m_1},s_{1,1}),\beta_s(t_{1,m_1},s_{1,2}),\cdots,\beta_s(t_{1,m_1},s_{1,n_1})),\end{aligned} \tag{3.10}$$

$$\begin{aligned}\mathbf{F}_1=&\operatorname{diag}(f(t_{1,1},s_{1,1}),f(t_{1,1},s_{1,2}),\cdots,f(t_{1,1},s_{1,n_1}),\cdots,\\&f(t_{1,m_1},s_{1,1}),f(t_{1,m_1},s_{1,2}),\cdots,f(t_{1,m_1},s_{1,n_1})),\end{aligned} \tag{3.11}$$

$$\mathbf{U}_1 = \text{diag}(u(t_{1,1}, s_{1,1}), u(t_{1,1}, s_{1,2}), \cdots, u(t_{1,1}, s_{1,n_1}), \cdots,$$
$$u(t_{1,m_1}, s_{1,1}), u(t_{1,m_1}, s_{1,2}), \cdots, u(t_{1,m_1}, s_{1,n_1})). \tag{3.12}$$

The matrix equation of (3.5) can be written as

$$[\mathbf{B}_1(\mathbf{C}^{1(2)} \otimes \mathbf{I}_{n_1} + \mathbf{I}_{m_1} \otimes \mathbf{D}^{1(2)}) + \mathbf{B}_{11}(\mathbf{C}^{1(1)} \otimes \mathbf{I}_{n_1}) + \mathbf{B}_{12}(\mathbf{I}_{m_1} \otimes \mathbf{D}^{1(2)})]\mathbf{U}_1 = \mathbf{F}_1 \tag{3.13}$$

where $\mathbf{C}^{1(1)}, \mathbf{C}^{1(2)}, \mathbf{D}^{1(1)}, \mathbf{D}^{1(2)}$ are the one order and two order differential matrices, and $\mathbf{I}_{m_1}, \mathbf{I}_{n_1}$ are the identity matrices. Then, we write

$$\mathbf{L}_1\mathbf{U}_1 = \mathbf{F}_1. \tag{3.14}$$

Similarly in the sub-domain Ω_2, we get the matrix equation

$$[\mathbf{B}_2(\mathbf{C}^{2(2)} \otimes \mathbf{I}_{n_2} + \mathbf{I}_{m_2} \otimes \mathbf{D}^{2(2)}) + \mathbf{B}_{21}(\mathbf{C}^{2(1)} \otimes \mathbf{I}_{n_2}) + \mathbf{B}_{22}(\mathbf{I}_{m_2} \otimes \mathbf{D}^{2(2)})]\mathbf{U}_2 = \mathbf{F}_2, \tag{3.15}$$

and

$$\mathbf{L}_2\mathbf{U}_2 = \mathbf{F}_2. \tag{3.16}$$

Combining Eq (3.14) and Eq (3.16), we get

$$\begin{bmatrix} \mathbf{L}_2 & \mathbf{0} \\ \mathbf{0} & \mathbf{L}_2 \end{bmatrix} \begin{bmatrix} \mathbf{U}_1 \\ \mathbf{U}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{F}_1 \\ \mathbf{F}_2 \end{bmatrix}. \tag{3.17}$$

Then, we have

$$\mathbf{LU} = \mathbf{F}, \tag{3.18}$$

and

$$\mathbf{L} = \begin{bmatrix} \mathbf{L}_2 & \mathbf{0} \\ \mathbf{0} & \mathbf{L}_2 \end{bmatrix}, \quad \mathbf{U} = \begin{bmatrix} \mathbf{U}_1 \\ \mathbf{U}_2 \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} \mathbf{F}_1 \\ \mathbf{F}_2 \end{bmatrix}. \tag{3.19}$$

Points of the boundary are $2(m_1 + m_2 - 2) + n_1 + n_2$. S_b denotes the number of the domain, and boundary points are denoted as $(t_k^b, s_k^b), k \in S_b$. The boundary condition can be discrete, as

$$\mathbf{e}^k_{m_1n_1+m_2n_2}\mathbf{U} = u_0(t_k^b, s_k^b)$$

$$u(t_{1i,l}, s_{1i,l}) = \sum_{i=1}^{m_1}\sum_{j=1}^{n_1} R_{1,i}(t_{i,l})R_{1,j}(s_{i,l})u_{1,ij} \tag{3.20}$$

$$u(t_{2i,l}, s_{2i,l}) = \sum_{i=1}^{m_2}\sum_{j=1}^{n_2} R_{2,i}(t_{2i,l})R_{2,j}(s_{2i,l})u_{2,ij} \tag{3.21}$$

$$u_t(t_{1i,l}, s_{1i,l}) = \sum_{i=1}^{m_1}\sum_{j=1}^{n_1} R'_{1,i}(t_{i,l})R_{1,j}(s_{i,l})u_{1,ij} \tag{3.22}$$

$$u_t(t_{2i,l}, s_{2i,l}) = \sum_{i=1}^{m_2}\sum_{j=1}^{n_2} R'_{2,i}(t_{2i,l})R_{2,j}(s_{2i,l})u_{2,ij} \tag{3.23}$$

$$u_s(t_{1i,l}, s_{1i,l}) = \sum_{i=1}^{m_1}\sum_{j=1}^{n_1} R_{1,i}(t_{i,l})R'_{1,j}(s_{i,l})u_{1,ij} \tag{3.24}$$

$$u_s(t_{2i,l}, s_{2i,l}) = \sum_{i=1}^{m_2}\sum_{j=1}^{n_2} R_{2,i}(t_{2i,l})R'_{2,j}(s_{2i,l})u_{2,ij} \tag{3.25}$$

where $l = 0, 1, \cdots, m_0$, and

$$(t_{1i,l}, s_{1i,l}) = (t_{2i,l}, s_{2i,l}) = (t_{i,l}, s_{i,l}).$$

Define

$$\mathbf{R}_{1,i}(t_{i,l}) = [R_{1,0}(t_{i,l}), R_{1,1}(t_{i,l}), \cdots, R_{1,m_1}(t_{i,l})] \tag{3.26}$$
$$\mathbf{R}_{1,i}(s_{i,l}) = [R_{1,0}(s_{i,l}), R_{1,1}(s_{i,l}), \cdots, R_{1,n_1}(s_{i,l})] \tag{3.27}$$
$$\mathbf{R}_{2,i}(t_{i,l}) = [R_{2,0}(t_{i,l}), R_{2,1}(t_{i,l}), \cdots, R_{2,m_1}(t_{i,l})] \tag{3.28}$$
$$\mathbf{R}_{2,i}(s_{i,l}) = [R_{2,0}(s_{i,l}), R_{2,1}(s_{i,l}), \cdots, R_{2,n_1}(s_{i,l})] \tag{3.29}$$
$$\mathbf{R}_{t1,i}(t_{i,l}) = [R'_{1,0}(t_{i,l}), R'_{1,1}(t_{i,l}), \cdots, R'_{1,m_1}(t_{i,l})] \tag{3.30}$$
$$\mathbf{R}_{s1,i}(s_{i,l}) = [R'_{1,0}(s_{i,l}), R'_{1,1}(s_{i,l}), \cdots, R'_{1,n_1}(s_{i,l})] \tag{3.31}$$
$$\mathbf{R}_{t2,i}(t_{i,l}) = [R'_{2,0}(t_{i,l}), R'_{2,1}(t_{i,l}), \cdots, R'_{2,m_1}(t_{i,l})] \tag{3.32}$$
$$\mathbf{R}_{s2,i}(s_{i,l}) = [R'_{2,0}(s_{i,l}), R'_{2,1}(s_{i,l}), \cdots, R'_{2,n_1}(s_{i,l})]. \tag{3.33}$$

Take as the matrix equation

$$u(t_{1i,l}, s_{1i,l}) = (\mathbf{L}_1 \otimes \mathbf{M}_1)\mathbf{U} \tag{3.34}$$
$$u(t_{2i,l}, s_{2i,l}) = (\mathbf{L}_2 \otimes \mathbf{M}_2)\mathbf{U} \tag{3.35}$$
$$u_t(t_{1i,l}, s_{1i,l}) = (\mathbf{L}_{t1,l} \otimes \mathbf{M}_{1,1})\mathbf{U} \tag{3.36}$$
$$u_t(t_{2i,l}, s_{2i,l}) = (\mathbf{L}_{t2,l} \otimes \mathbf{M}_{2,l})\mathbf{U} \tag{3.37}$$
$$u_s(t_{1i,l}, s_{1i,l}) = (\mathbf{L}_{1,l} \otimes \mathbf{M}_{s1,1})\mathbf{U} \tag{3.38}$$
$$u_s(t_{2i,l}, s_{2i,l}) = (\mathbf{L}_{2,l} \otimes \mathbf{M}_{s2,1})\mathbf{U}. \tag{3.39}$$

The discrete boundary condition condition can be given as

$$[u]_{(t_{1i,l}, s_{1i,l})} = u(t_{2i,l}, s_{2i,l}) - u(t_{1i,l}, s_{1i,l}) = \delta(t_{i,l}, s_{i,l}) \tag{3.40}$$

$$[(\beta\nabla u)\cdot \mathbf{n}]_{(t_{1i,l}, s_{1i,l})} = [u_t(t_{2i,l}, s_{2i,l})n_{tl} - u_s(t_{2i,l}, s_{2i,l})n_{sl}] \tag{3.41}$$
$$-[u_t(t_{1i,l}, s_{1i,l})n_{tl} - u_s(t_{1i,l}, s_{1i,l})n_{sl}] = \gamma(t_{i,l}, s_{i,l}). \tag{3.42}$$

The matrix equations of (3.40) and (3.41) are

$$[\mathbf{L}_{2,l} \otimes \mathbf{M}_{2,l} - \mathbf{L}_{1,l} \otimes \mathbf{M}_{1,l}]\mathbf{U} = \delta(t_{i,l}, s_{i,l}) \tag{3.43}$$
$$\beta_2[(\mathbf{L}_{t2,l} \otimes \mathbf{M}_{2,l})n_{tl} - (\mathbf{L}_{2,l} \otimes \mathbf{M}_{s2,l})n_{sl}]\mathbf{U}$$
$$-\beta_1[(\mathbf{L}_{t1,l} \otimes \mathbf{M}_{1,l})n_{tl} - (\mathbf{L}_{1,l} \otimes \mathbf{M}_{s1,l})n_{sl}]\mathbf{U} = \gamma(t_{i,l}, s_{i,l}) \tag{3.44}$$

4. Numerical examples

Example 1. *Consider*

$$-\nabla^2 u + u = f$$

with $f(t,s) = t^2 - 2$ *; its analytic solutions are*

$$u(t,s) = t^2 + e^s$$

where $\Omega = [-1,1] \times [-1,1]$.

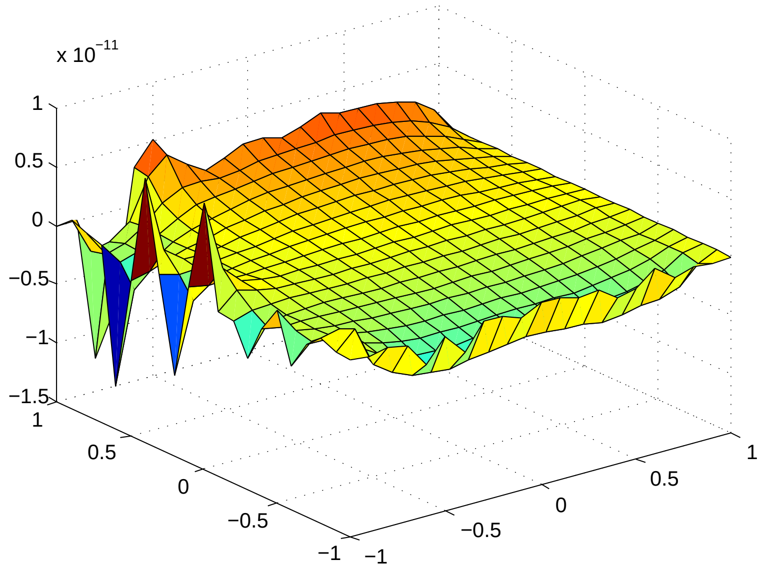

Figure 1. Error estimate of equidistant nodes with $m = 20; n = 20; d_1 = d_2 = 9$.

Table 1. Errors of equidistant nodes with $d_1 = d_2$.

$m \times n$	$d_1 = d_2 = 1$	2	3	4	5
8×8	2.6911e-02	1.6610e-03	7.1154e-04	7.2691e-05	4.0731e-05
16×16	8.0799e-03	2.3171e-04	4.5299e-05	2.9892e-06	6.1418e-07
32×32	2.4157e-03	2.9843e-05	2.8178e-06	1.0109e-07	9.4081e-09
64×64	7.1483e-04	3.7641e-06	1.7520e-07	3.2724e-09	3.8841e-10
h^α	1.7448	2.9285	3.9959	4.8130	5.5594

In Table 1 convergence rate is $O(h^{d_1+1})$ with $d_1 = d_2 = 2, 3, 4, 5$. In Table 2, for the Chebyshev nodes, the convergence rate is $O(\tau^{d_2+2})$ with $d_1 = d_2 = 2, 3, 4, 5$.

Figure 1 shows the error estimate of equidistant nodes, and Figure 2 shows the error estimate of Chebyshev nodes.

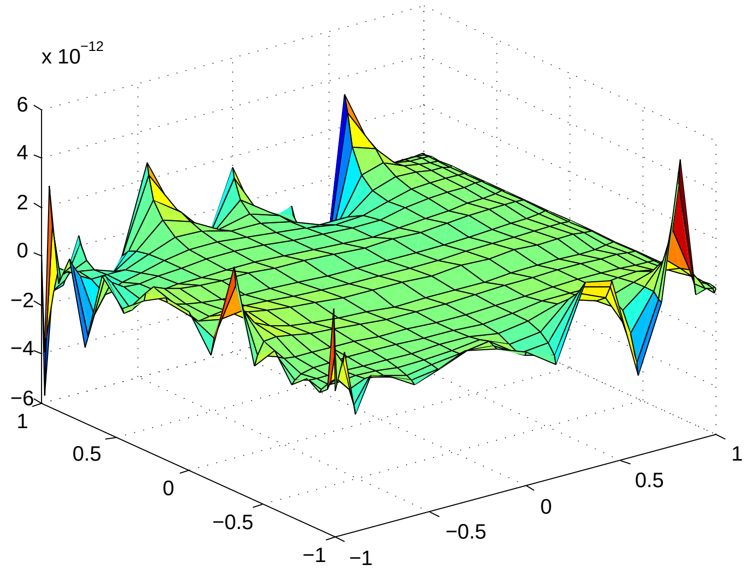

Figure 2. Error estimate of Chebyshev nodes with $m = 20; n = 20; d_1 = d_2 = 9$.

Table 2. Errors of non-equidistant nodes with $d_1 = d_2$.

$m \times n$	$d_1 = d_2 = 1$	2	3	4	5
8×8	6.2451e-02	1.5475e-03	5.5808e-04	1.8735e-05	2.2295e-06
16×16	1.5433e-02	1.1791e-04	9.1048e-06	4.3131e-07	7.8660e-08
32×32	3.5514e-03	6.8688e-06	2.0908e-07	4.9400e-09	3.2320e-10
64×64	8.4301e-04	3.9526e-07	4.5490e-09	4.9775e-09	4.3803e-09
h^α	2.0703	3.9783	5.6349	3.9593	2.9971

Example 2. *Consider*

$$-\nabla^2 u + u = f$$

with $f(t, s) = 3\sin(t + s)$. *Its analytic solutions are*

$$u(t, s) = \sin(t + s)$$

where $\Omega = [-1, 1] \times [-1, 1]$.

Table 3 shows the convergence is $O(h^{d_1+1})$ with $d_1 = d_2 = 2, 3, 4, 5$. In Table 4, for the non-uniform partition with Chebyshev nodes for $d_1 = d_2 = 2, 3, 4, 5$, the convergence rate is $O(\tau^{d_2+2})$.

We choose $m = 20; n = 20; d_1 = 9; d_2 = 9$ to test our algorithm.

Figure 3 shows the error estimate of equidistant nodes, and Figure 4 shows the error estimate of Chebyshev nodes.

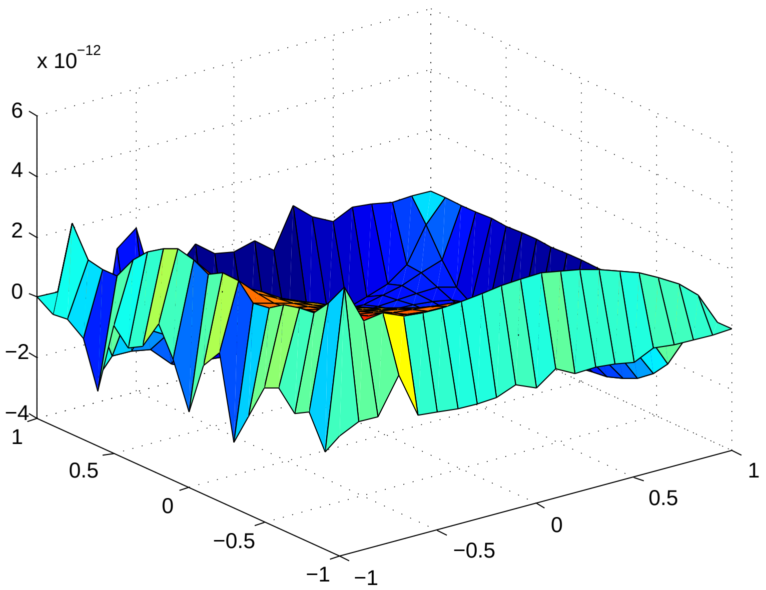

Figure 3. Error estimate of equidistant nodes with $m = 20; n = 20; d_1 = d_2 = 9$.

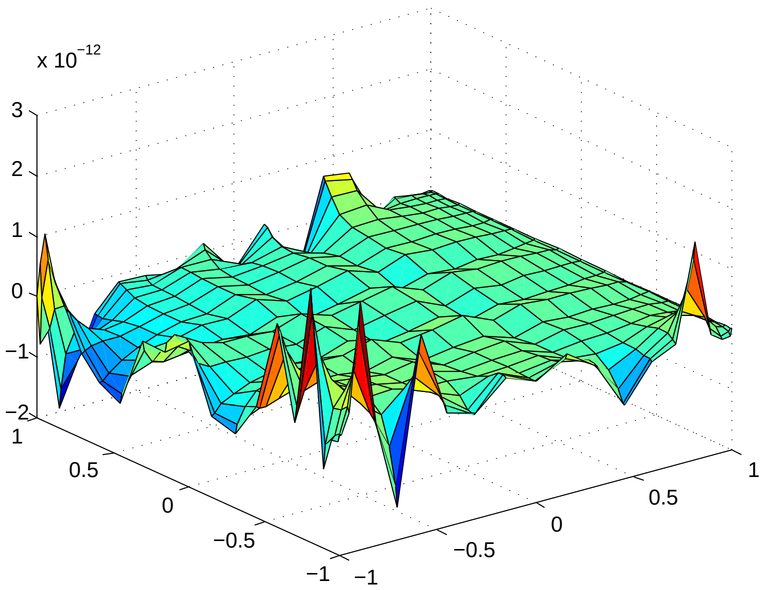

Figure 4. Error estimate of Chebyshev nodes with $m = 20; n = 20; d_1 = d_2 = 9$.

Table 3. Errors of equidistant nodes with $d_1 = d_2$.

$m \times n$	$d_1 = d_2 = 1$	2	3	4	5
8×8	5.0201e-03	1.0360e-03	3.7701e-04	4.8362e-05	2.5361e-05
16×16	1.7960e-03	1.4558e-04	2.1324e-05	1.9874e-06	3.1977e-07
32×32	6.0280e-04	1.8624e-05	1.2263e-06	6.5505e-08	4.2192e-09
64×64	1.9404e-04	2.3331e-06	7.2859e-08	2.0682e-09	2.8110e-10
h^α	1.5644	2.9315	4.1124	4.8377	5.4871

Table 4. Errors of Chebyshev nodes with $d_1 = d_2$.

$m \times n$	$d_1 = d_2 = 1$	2	3	4	5
8×8	1.3288e-02	9.9717e-04	1.1162e-04	7.8012e-06	1.8963e-06
16×16	2.8908e-03	6.3908e-05	4.7613e-06	3.3689e-07	4.0815e-08
32×32	6.2445e-04	4.1969e-06	1.0285e-07	3.7866e-09	1.4989e-10
64×64	1.4493e-04	2.4827e-07	2.4237e-09	1.4696e-09	1.8797e-09
h^α	2.1729	3.9906	5.1637	4.1247	3.3262

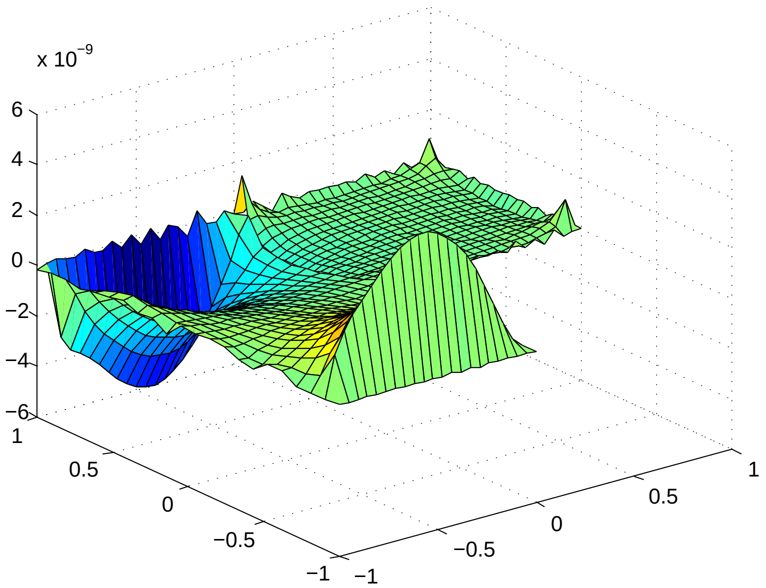

Figure 5. Error estimate of equidistant nodes with $m_1 = m_2 = 20; n_1 = n_2 = 20; d_1 = d_2 = 9$.

Example 3. *Consider the Poisson equation* $\triangle u = -2\sin(\pi t)\cos(\pi s), (t, s) \in \Omega$ *and* $\Omega = \Omega_1 \bigcup \Omega_2 = \{t, s : -1 < t < 1, -1 < s < 1\} \bigcup \{t, s : 0 < t < 1, 0 < s < 1\}$. *Its analytic solutions are*

$$u(t, s) = \sin(\pi t)\cos(\pi s)$$

with the boundary condition

$$[u]_{\Gamma_0} = 0, [u_t]_{\Gamma_0} = 0.$$

We choose $m_1 = m_2 = 20; n_1 = n_2 = 20; d_1 = 9; d_2 = 9$ to test the domain decomposition method of the barycentric rational collocation method. Figure 5 shows the errors under equidistant nodes. From Figure 5 we know that the error can reach 10^{-7} with 21 collocation points.

Example 4. *Consider* $\triangle u(t,s) = 6ts(t^2 - s^2 - 2), (t,s) \in \Omega$ *and* $\Omega = \Omega_1 \bigcup \Omega_2 = \{t, s : -1 < t < 1, -1 < s < 1\} \bigcup \{t, s : -0.5 < t < 0.5, -1 < s < 0\}$. *Its analytic solutions are*

$$u(t,s) = ts(t^2 - 1)(s^2 - 1)$$

with condition

$$[u]_{\Gamma_0} = 0, [u_t]_{\Gamma_0} = 0.$$

We choose $m_1 = m_2 = 20; n_1 = n_2 = 20; d_1 = 9; d_2 = 9$ on each $\Omega_i, i = 1, 2$, to test the domain decomposition method of the barycentric rational collocation method. Figure 6 shows the error estimate of equidistant nodes, and Figure 7 shows the error estimate of Chebyshev nodes. The error of both equidistant nodes and Chebyshev nodes can reach 10^{-11}, which shows the accuracy of our algorithm.

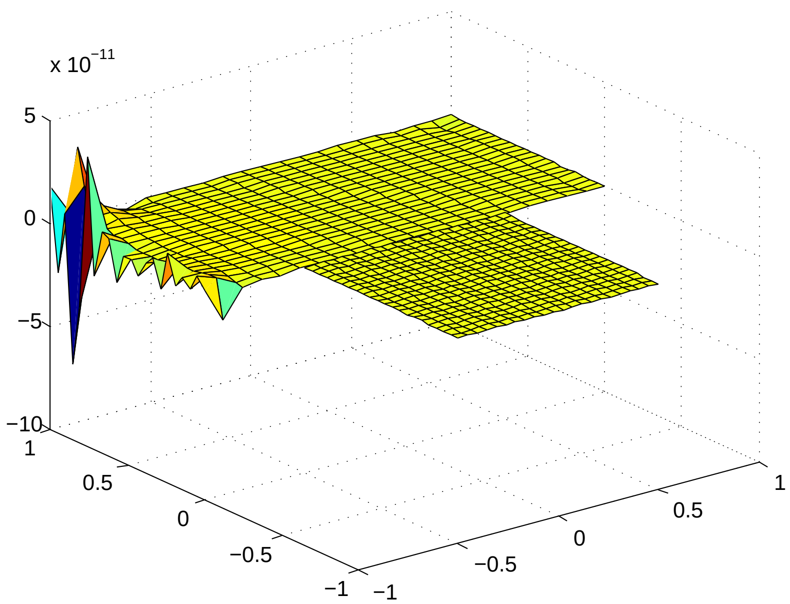

Figure 6. Error estimate of equidistant nodes with $m_1 = m_2 = 20; n_1 = n_2 = 20; d_1 = d_2 = 9$.

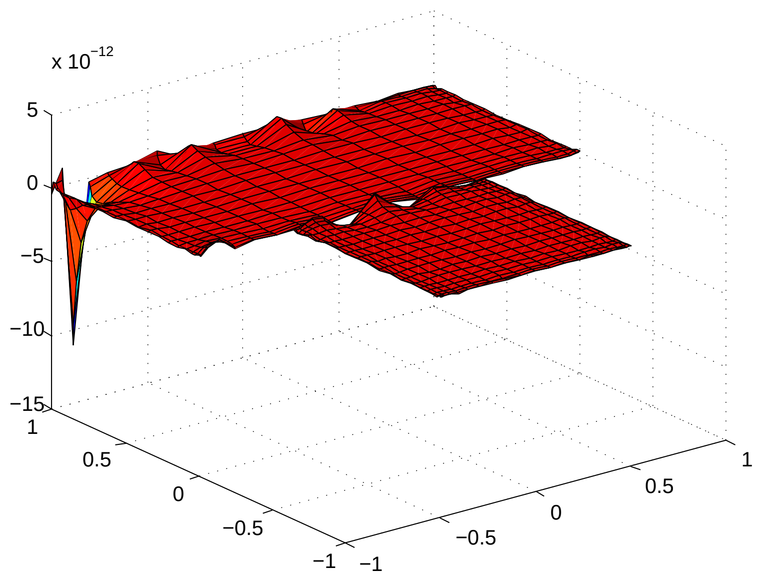

Figure 7. Error estimate of Chebyshev nodes with $m_1 = m_2 = 20; n_1 = n_2 = 20; d_1 = d_2 = 9$.

Acknowledgments

The work of Jin Li was supported by the Natural Science Foundation of Shandong Province (Grant No. ZR2022MA003) and Natural Science Foundation of Hebei Province (Grant No. A2019209533).

The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

Conflict of interest

The authors declare that there are no conflicts of interest.

References

1. Dell' Accio, F. Di Tommaso, O. Nouisser, N. Siar, Solving Poisson equation with Dirichlet conditions through multinode Shepard operators, *Comput. Math. Appl.*, **98** (2021), 254–260. https://doi.org/10.1016/j.camwa.2021.07.021

2. F. DellAccio, F. Di Tommaso, G. Ala, E. Francomano, Electric scalar potential estimations for non-invasive brain activity detection through multinode Shepard method, in *2022 IEEE 21st Mediterranean Electrotechnical Conference (MELECON)*, (2022), 1264–1268. https://doi.org/10.1109/MELECON53508.2022.9842881

3. M. Floater, H. Kai, Barycentric rational interpolation with no poles and high rates of approximation, *Numer. Math.*, **107** (2007), 315–331. https://doi.org/10.1007/s00211-007-0093-y

4. G. Klein, J. Berrut, Linear rational finite differences from derivatives of barycentric rational interpolants, *SIAM J. Numer. Anal.*, **50** (2012), 643–656. https://doi.org/10.1137/110827156

5. G. Klein, J. Berrut, Linear barycentric rational quadrature, *BIT Numer. Math.*, **52** (2012), 407–424. https://doi.org/10.1007/s10543-011-0357-x

6. R. Baltensperger, J. P. Berrut, The linear rational collocation method, *J. Comput. Appl. Math.*, **134** (2001), 243–258. https://doi.org/10.1016/S0377-0427(00)00552-5

7. S. Li, Z. Wang, *High Precision Meshless barycentric Interpolation Collocation Method–Algorithmic Program and Engineering Application*, Science Publishing, Beijing, 2012.

8. Z. Wang, S. Li, *Barycentric Interpolation Collocation Method for Nonlinear Problems*, National Defense Industry Press, Beijing, 2015.

9. Z. Wang, Z. Xu, J. Li, Mixed barycentric interpolation collocation method of displacement-pressure for incompressible plane elastic problems, *Chin. J. Appl. Mech.*, **35** (2018), 195–201. https://doi.org/1000-4939(2018) 03-0631-06

10. Z. Wang, L. Zhang, Z. Xu, J. Li, Barycentric interpolation collocation method based on mixed displacement-stress formulation for solving plane elastic problems, *Chin. J. Appl. Mech.*, **35** (2018), 304–309. https://doi.org/10.11776/cjam.35.02.D002

11. W. H. Luo, T. Z. Huang, X. M. Gu, Y. Liu, Barycentric rational collocation methods for a class of nonlinear parabolic partial differential equations, *Appl. Math. Lett.*, **68** (2017), 13–19. https://doi.org/10.1016/j.aml.2016.12.011

12. J. Li, Linear barycentric rational collocation method for solving biharmonic equation, *Demonstr. Math.*, **55** (2022), 587–603. https://doi.org/10.1515/dema-2022-0151

13. J. Li, X. Su, K. Zhao, Barycentric interpolation collocation algorithm to solve fractional differential equations, *Math. Comput. Simul.*, **205** (2023), 340–367. https://doi.org/10.1016/j.matcom.2022.10.005

14. J. Li, X. Su, J. Qu, Linear barycentric rational collocation method for solving telegraph equation. *Math. Methods Appl. Sci.*, **44** (2021), 11720–11737. https://doi.org/10.1002/mma.7548

15. J. Li, Y. Cheng, Linear barycentric rational collocation method for solving second-order Volterra integro-differential equation, *Comput. Appl. Math.*, **39** (2020). https://doi.org/10.1007/s40314-020-1114-z

16. J. Li, Y. Cheng, Linear barycentric rational collocation method for solving heat conduction equation, *Numer. Methods Partial Differ. Equations*, **37** (2021), 533–545. https://doi.org/10.1002/num.22539

17. K. Jing, N. Kang, A convergent family of bivariate Floater-Hormann rational interpolants, *Comput. Methods Funct. Theory*, **21** (2021), 271–296. https://doi.org/10.1007/s40315-020-00334-9