Research Article

Barycentric Rational Collocation Method for Burgers’ Equation

Qingli Zhao, Ruiwen Wang, Zhaoqing Wang, and Xiaoping Zhang

1 School of Science, Shandong Jianzhu University, Jinan, Shandong 250101, China
2 CMA Earth System Modeling and Prediction Centre, Beijing 100081, China
3 State Key Laboratory of Severe Weather, Beijing 100081, China

Correspondence should be addressed to Ruiwen Wang; wangrw@cma.gov.cn

Received 10 February 2022; Revised 14 April 2022; Accepted 19 April 2022; Published 9 May 2022

In this article, barycentric rational collocation method is introduced to solve Burgers’ equation. The algebraic equations of the barycentric rational collocation method are presented. Numerical analysis and error estimates are established. With the help of the barycentric rational interpolation theory, the convergence rates of the barycentric rational collocation method for Burgers’ equation are proved. Numerical experiments are carried out to validate the convergence rates and show the efficiency.

1. Introduction

Burgers’ equation involves the convection term, diffusion term, and kinetic viscosity coefficient whose characteristic is same as the structure of the Navier–Stokes equation without the stress term. It describes the phenomena such as dispersion in porous media, weak shock propagation, heat conduction, acoustic attenuation in fog, compressible turbulence, gas-dynamics, continuous stochastic processes, and even continuum traffic simulation. Burgers’ equation is as follows [1–6]:

\[ \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} - \nu \frac{\partial^2 u}{\partial x^2} = f(x,t), \quad x \in (\alpha, \beta), \ t > 0, \]

with boundary conditions as

\[ u(\alpha, t) = \phi_1(t), u(\beta, t) = \phi_2(t), \quad t > 0, \]

and initial condition as

\[ u(x, 0) = \psi(x), \quad x \in (\alpha, \beta), \]

where \( \nu > 0 \) be the kinetic viscosity. Boundary conditions sometimes are presented as periodic boundary conditions \( u(\alpha, t) = u(\beta, t) = 0 \).

In view of the universality of Burgers’ equation in describing lots of important physical phenomena, many numerical methods were introduced to solve it such as the finite difference method, finite element method, mixed finite element method, characteristics mixed finite element, spectral method, and meshless method; see [1–6] and the references therein.

With the help of Lagrange interpolation, the barycentric rational interpolation method is obtained [7–9]. A rational interpolation scheme with equidistant and special distributed nodes has been proposed by Floater and Hormann [10]. Compared with Lagrange interpolation, the barycentric rational interpolation has the advantages of stability. Abdi et al. [11, 12] have used the barycentric rational collocation method to solve Volterra and Volterra integro-differential equation. With the further expansion of the application fields, the barycentric rational collocation method has been successfully applied to solve some initial value problems and boundary value problems by Wang et al. [13–15]. The relevant calculation results show the stability advantages and high accuracy of the barycentric rational collocation method. The research of the barycentric rational collocation method for the heat-conduction equation, biharmonic problem, second-order Volterra integro-differential equation, third-order two-point boundary value problem, beam force vibration equation, telegraph equation, and...
incompressible Forchheimer flow in porous media has been presented in recent papers by Li et al. [16–22]. In these papers, error estimation and numerical simulation are given.

The main goal of the present paper is to solve the nonlinear Burgers’ equation with the barycentric rational collocation method. (\(t^d + t^d\)) error estimates are proved. Numerical experiments are carried out to show the convergence rates. Remaining part of the paper is structured as follows. In Section 2, the barycentric rational interpolation formula is given. In Section 3, convergence analysis of the barycentric rational collocation method for the nonlinear Burgers’ equation is presented. Section 4 reports some test examples to show the accuracy, effectiveness, and efficiency.

2. Notations and Barycentric Rational Interpolation

Define the partition of space interval \([\alpha, \beta]\) as

\[
\alpha = x_0 < x_1 < \cdots < x_{m-1} < x_m = \beta, \\
h = \max |x_i - x_{i-1}|.
\]

(4)

Polynomial

\[
P_i(x) = \sum_{k=0}^{i+d} \prod_{j=0, j \neq k}^{i} \frac{x-x_j}{x_k-x_j} y_k,
\]

(5)

denotes the \(d\)-order Lagrange interpolation with \(y_k = y(x_k)\)

\[
P_i(x_k) = y(x_k), \quad k = i, i+1, \ldots, i+d.
\]

(6)

The barycentric interpolation function \(R(x)\) \((d = 0, 1, \ldots, m)\) is presented as

\[
R(x) = \frac{\sum_{i=0}^{m-d} \mu_i(x) P_i(x)}{\left(\sum_{i=0}^{m-d} \mu_i(x)\right)},
\]

(7)

where \(\mu_i(x)\) denotes the blending function as follows:

\[
\mu_i(x) = \frac{(-1)^i}{(x-x_i) \ldots (x-x_{i+d})}
\]

(8)

According to the definition of \(\mu_i(x)\), it can be deduced that

\[
\sum_{i=0}^{m-d} \mu_i(x) P_i(x) = \frac{\sum_{i=0}^{m-d} \sum_{j=0}^{i+d} 1}{x_k-x_j} y_k = \sum_{k=0}^{m} \frac{\omega_k}{x-x_k} y_k,
\]

(9)

where \(\omega_k\) denotes the interpolation weight function as follows:

\[
\omega_k = \sum_{I \in I_k} \prod_{j=0, j \neq k}^{i} \frac{1}{x_k-x_j}, \quad I_k = \{i \in I; k - d \leq i \leq k\}, \quad I = \{0, 1, \ldots, m-d\}.
\]

(10)

Through simple derivation, we know

\[
\sum_{k=i, j \neq k}^{i+d} \frac{x-x_j}{x_k-x_j} = 1, \quad \sum_{i=0}^{m-d} \mu_i(x) = \sum_{k=0}^{m} \frac{\omega_k}{x-x_k}.
\]

(11)

Combining (5)–(11), the barycentric rational interpolation function \(R(x)\) is presented as

\[
R(x) = \frac{\sum_{j=0}^{m} \left(\frac{\omega_j}{(x-x_j)}\right) y_j}{\sum_{j=0}^{m} \left(\frac{\omega_j}{(x-x_j)}\right)} = \sum_{j=0}^{m} R_j(x) y_j,
\]

(12)

\[
y^{(s)}(x_j) := y^{(s)}_i(x_j) = \frac{d^s y(x_j)}{dx^s} = \sum_{j=0}^{m} R_j^{(s)}(x_j) y_j = \sum_{j=0}^{m} D_{ij}^{(s)} y_j, \quad s = 1, 2, \ldots
\]

(13)

Its \(s\)-order differential matrices formulation can be written into

\[
y^{(s)} = D^{(s)} y, \quad s = 1, 2, \ldots
\]

(14)

where

\[
y^{(s)} = \left[y^{(s)}_0, y^{(s)}_1, \ldots, y^{(s)}_n\right], \quad s = 1, 2, \ldots
\]

(15)

\[
D_{ij}^{(s)} = R_{ij}^{(s)}(x_j), \quad s = 1, 2, \ldots
\]

(16)

According to definition of \(R_j(x)\) in (13), we get the first-order derivative of interpolation basis function \(R_j(x)\) as
\[ R'_j(x_i) = \frac{\partial_j}{\partial_i} R_{ij}, \quad j \neq i, \quad (18) \]

\[ R'_i(x_i) = -\sum_{j \neq i} P'_j(x_i). \quad (19) \]

Combining equations (17)–(19) together, the \( s \)-order differential recurrence formula of \( D_{ij}^{(s)} \) \((s = 1, 2, \ldots)\) is

\[
\begin{align*}
D_{ij}^{(s)} &= m \left( \frac{D_{ii}^{(s-1)} D_{ij}^{(1)} - D_{ij}^{(s-1)}}{(x_i - x_j)} \right), \quad i \neq j, \\
D_{ii}^{(s)} &= \sum_{j \neq i} D_{ij}^{(s)}. 
\end{align*}
\]

(20)

For the nonlinear Burgers’ equation with \( \Omega = [\alpha, \beta] \times [0, T] \), we partition the space region \([\alpha, \beta]\) into

\[ \alpha = x_0 < x_1 < \ldots < x_k < \ldots < x_m = \beta, \]

and time interval \([0, T]\) into

\[ 0 = t_0 < t_1 < \ldots < t_k < \ldots < t_n = T, \]

\[ \tau = \max_{1 \leq j \leq n} (t_j - t_{j-1}). \]

(21)

(22)

Function \( u(x, t) \) is approximated by its barycentric rational interpolation as follows:

\[ u(x, t) \approx \sum_{j=0}^{m} R_j(x) u_j(t), \]

(23)

where

\[ u_j(t) = u(x_j, t), \quad j = 0, 1, \ldots, m. \]

(24)

Taking (23) into equation (1), we see

\[ \sum_{j=0}^{m} R_j(x) \hat{u}_j(t) + \left( \sum_{j=0}^{m} R_j(x) u_j(t) \right) \cdot \left( \sum_{j=0}^{m} R'_j(x) u_j(t) \right) \]

\[ -\gamma \left( \sum_{j=0}^{m} \hat{R}_j(x) u_j(t) \right) = f(x, t), \]

(25)

where \( \hat{u}_j(t) \) is the first-order derivative of the function \( u_j(t) \).

Taking \( x = x_i \) in equation (25), we get

\[ \sum_{j=0}^{m} R_j(x_i) \hat{u}_j(t) + \left( \sum_{j=0}^{m} R_j(x_i) u_j(t) \right) \cdot \left( \sum_{j=0}^{m} R'_j(x_i) u_j(t) \right) \]

\[ -\gamma \left( \sum_{j=0}^{m} \hat{R}_j(x_i) u_j(t) \right) = f(x_i, t). \]

(26)

Note that \( R_j(x_i) = \delta_{ij} \); after further simplification of equation (26), we know

\[ \sum_{j=0}^{m} \delta_{ij} \hat{u}_j(t) + \left( \sum_{j=0}^{m} \delta_{ij} u_j(t) \right) \cdot \left( \sum_{j=0}^{m} C_{ij}^{(1)} u_j(t) \right) \]

\[ -\gamma \left( \sum_{j=0}^{m} C_{ij}^{(2)} u_j(t) \right) = f(x_i, t), \]

(27)

where

\[ C_{ij}^{(1)} = R'_j(x_i), \]

(28)

\[ C_{ij}^{(2)} = R_j(x_i). \]

Combining equations (26)–(28), the matrix form is presented as

\[
\begin{bmatrix}
\mathbf{u}_0(t) \\
\vdots \\
\mathbf{u}_m(t)
\end{bmatrix} +
\begin{bmatrix}
\mathbf{C}^{(1)} & \cdots & \mathbf{C}^{(1)}_{mm} \\
\vdots & \ddots & \vdots \\
\mathbf{C}^{(2)}_{m0} & \cdots & \mathbf{C}^{(2)}_{mm}
\end{bmatrix}
\begin{bmatrix}
\mathbf{u}_0(t) \\
\vdots \\
\mathbf{u}_m(t)
\end{bmatrix}
- \gamma
\begin{bmatrix}
\mathbf{u}_0(t) \\
\vdots \\
\mathbf{u}_m(t)
\end{bmatrix}
= \begin{bmatrix}
\mathbf{f}_0(t) \\
\vdots \\
\mathbf{f}_m(t)
\end{bmatrix},
\]

(29)

Further, matrix equation (29) can be rewritten into a simple vector form as follows:

\[ \hat{\mathbf{u}}(t) + \mathbf{diag}(\mathbf{u}(t)) \mathbf{C}^{(1)} \mathbf{u}(t) - \gamma \mathbf{C}^{(2)} \mathbf{u}(t) = \mathbf{f}(t), \]

(30)

where

\[ \mathbf{u}(t) = [\mathbf{u}_0(t), \mathbf{u}_1(t), \ldots, \mathbf{u}_m(t)]^T, \]

\[ \hat{\mathbf{u}}(t) = [\hat{\mathbf{u}}_0(t), \hat{\mathbf{u}}_1(t), \ldots, \hat{\mathbf{u}}_m(t)]^T, \]

\[ \mathbf{f}(t) = [\mathbf{f}_0(t), \mathbf{f}_1(t), \ldots, \mathbf{f}_m(t)]^T. \]

(31)

Through similarly derivation, the discrete scheme of time variable \( t \) is obtained as

\[ u_i(t_j) = u(x_i, t_j) = u_{ij}, \quad i = 0, 1, \ldots, m, \quad j = 0, 1, \ldots, n, \]

(32)

\[ u_i(t) = \sum_{k=0}^{n} R_k(t) u_{ik}, \quad i = 0, 1, \ldots, m. \]

(33)

According to equations (27)–(33), we have

\[ (j = 0, 1, \ldots, n) \]
follows:

matrices are obtained:

\[
\begin{bmatrix}
\sum_{k=0}^{n} \hat{R}_k(t_j)u_{0k} \\
\vdots \\
\sum_{k=0}^{n} \hat{R}_k(t_j)u_{nk}
\end{bmatrix}
+ \begin{bmatrix}
\sum_{k=0}^{n} R_k(t_j)u_{0k} \\
\vdots \\
\sum_{k=0}^{n} R_k(t_j)u_{nk}
\end{bmatrix}
= \begin{bmatrix}
C_{00}^{(1)} & \cdots & C_{0n}^{(1)} \\
\vdots & \ddots & \vdots \\
C_{m0}^{(1)} & \cdots & C_{mn}^{(1)}
\end{bmatrix}
\begin{bmatrix}
\sum_{k=0}^{n} R_k(t_j)u_{0k} \\
\vdots \\
\sum_{k=0}^{n} R_k(t_j)u_{nk}
\end{bmatrix}
\]

\[(34)\]

Equation (34) can be written into vector form as follows:

\[
(I_m \otimes D^{(1)})U + \text{diag}(U)(C^{(1)} \otimes I_n)U - \nu(C^{(2)} \otimes I_n)U = F,
\]

which can be restated as a simple form:

\[LU = F,\]

with

\[
L = I_m \otimes D^{(1)} - \text{diag}(U)(C^{(1)} \otimes I_n) - \nu(C^{(2)} \otimes I_n),
\]

\[
U = [u_{00}, u_{01}, \ldots, u_{0n}, u_{10}, u_{11}, \ldots, u_{1n}, u_{m0}, u_{m1}, \ldots, u_{mn}]^T,
\]

\[
F = [f_0, f_1, \ldots, f_{0v}, f_{1v}, \ldots, f_{0n}, f_{1n}, \ldots, f_{mn}]^T.
\]

Here, operation symbol \( \otimes \) represents the Kronecker product.

Then, we get the \( s \)-order differential at the mesh-point \( x_i \) as

\[\begin{align*}
\bar{u}_s^{(s)}(x_i) &= u_i^{(s)} = \frac{d^su_i(x_i)}{dx^s} = \sum_{j=0}^{n} R_j^{(s)}(x_i)u_j = \sum_{j=0}^{n} C_{ij}^{(s)}u_j, \quad s = 1, 2, \ldots.
\end{align*}\]

Its matrices formulation is

\[\bar{u}_s^{(s)} = C_s^{(s)}u,\]

where

\[
R_j^{(s)}(x_i) = \frac{2}{t_i - t_j} \left( \sum_{k \neq j} \frac{\bar{Q}_j}{x_i - x_k} + \frac{1}{x_i - x_j} \right),
\]

\[j \neq i, R_j^{(s)}(x_i) = -\sum_{j \neq i} R_j^{(s)}(x_i),\]

Then, we get the 1-order time differentiation matrix as follows:

\[D_{ij}^{(1)} = R_j^{(1)}(t_i).\]

Similarly, the 1-order and 2-order space differentiation matrices are obtained:

\[
C_{ij}^{(1)} = R_j^{(1)}(x_i),
\]

\[
C_{ij}^{(2)} = R_j^{(2)}(x_i).
\]

The \( s \)-order differential matrix recurrence formula is presented as follows:
\[ C_{ij}^{(s)} = s \left( \frac{C_{ij}^{(s-1)}}{(t_i - t_j)} - C_{ij}^{(s-1)} \right), \quad i \neq j, \]
\[ C_{ii}^{(s)} = -n \sum_{j=0, j \neq i} C_{ij}^{(s)}. \]

### 3. Convergence Analysis and Error Estimates

Define the error between \( u(x) \) and \( R(x) \) as follows:

\[ E(x) = u(x) - R(x). \]  \hspace{1cm} (45)

According to the error theory of interpolation, it is well known that

\[ E(x) = (x - x_0), \ldots, (x - x_{n+1})u[x_0, x_1, \ldots, x_n, x]. \]  \hspace{1cm} (46)

In the light of the definition of barycentric rational interpolation function \( R(x) \), combining (46) with (45), we have

\[ E(x) = \frac{\left( \sum_{i=0}^{n-d} \mu_i(x)(u(x) - Pi(x)) \right)}{\left( \sum_{i=0}^{n-d} \mu_i(x) \right)} = \frac{\xi(x)}{\eta(x)} \]  \hspace{1cm} (47)

where

\[ \xi(x) = \sum_{i=0}^{n-d} (-1)^i u[x_0, x_1, \ldots, x_i, x], \quad \eta(x) = \sum_{i=0}^{n-d} \mu_i(x). \]  \hspace{1cm} (48)

Define

\[ |E(x)| = \max_{a \leq x \leq b} |E(x)|. \]  \hspace{1cm} (49)

The following lemma has been proved by Berrut et al. in [7].

**Lemma 1.** For the error \( E(x) \) defined in (45), if function \( u(x) \) satisfies certain smoothness conditions on interval \([a, b] \), we have

\[ |E(x)| \leq C h^{d+1}, \quad u(x) \in C^{d+2}[a, b], \]
\[ |E'(x)| \leq C h^{d}, \quad u(x) \in C^{d+3}[a, b], \]
\[ |E''(x)| \leq C t^{d-1}, \quad u(x) \in C^{d+4}[a, b]. \]  \hspace{1cm} (50)

Now, we research the rational interpolation \( R_{mn}(x, t) \) to approximate the function \( u(x, t) \) as follows:

\[ R_{mn}(x, t) = \left( \sum_{i=0}^{m} \sum_{j=0}^{n} \omega_{ij} \frac{u(x) - (x - x_j)(t - t_j)}{\sum_{i=0}^{m} \omega_{ij} (x - x_i)(t - t_i)} \right) \]  \hspace{1cm} (51)

Note that the weight function \( \omega_{ij} \) is defined by

\[ \omega_{ij} = (-1)^{i-d_i + j-d_j} \sum_{k_1 \in I_{d_i}} \sum_{k_2 \in I_{d_j}} \prod_{k_1 \in I_{d_i}} \prod_{k_2 \in I_{d_j}} \frac{1}{1 - \frac{1}{x_i - x_j}} \frac{1}{1 - \frac{1}{t_j - t_i}}. \]  \hspace{1cm} (52)

Here, parameters \( d_i \) and \( d_j \) represent the space interpolation parameter and time interpolation parameter, respectively.

The error function \( E(x, t) \) between \( u(x, t) \) and \( R_{mn}(x, t) \) is defined by

\[ E(x, t) = u(x, t) - R_{mn}(x, t) \]
\[ = (x - x_i)(x - x_{i+1}), \ldots, (x - x_{i+d_i})u[x_0, x_1, \ldots, x_{i+d_i}, x, t] \]
\[ + (t - t_j)(t - t_{j+1}), \ldots, (t - t_{j+d_j})u[t_j, t_{j+1}, \ldots, t_{j+d_j}, x, t]. \]  \hspace{1cm} (53)

Based on Lemma 1, we get the following theorem.

**Theorem 1.** For the error functional \( E(x, t) \), if \( u(x, t) \in C^{d_i+2}[a, b] \times C^{d_j+2}[0, T] \), we have

\[ |E(x, t)| \leq C(h^{d_i+1} + t^{d_j+1}). \]  \hspace{1cm} (54)
\[ E(x, t) = u(x, t) - R_{m,n}(x, t) \]
\[ = (x - x_i)(x - x_{i+1}), \ldots, (x - x_{i+d_i})u[x_i, x_{i+1}, \ldots, x_{i+d_i}, x, t] + (t - t_j)(t - t_{j+1}), \ldots, (t - t_{j+d_j})u[t_j, t_{j+1}, \ldots, t_{j+d_j}, x, t] \]
\[ = \sum_{i=0}^{m-d_i} (-1)^iu[x_i, x_{i+1}, \ldots, x_{i+d_i}, x, t] + \sum_{j=0}^{n-d_j} (-1)^ju[t_j, t_{j+1}, \ldots, t_{j+d_j}, x, t] \]
\[ \sum_{i=0}^{m-d_i} \mu_i(x) + \sum_{j=0}^{n-d_j} \mu_j(t) \]  

Note that
\[ \sum_{i=0}^{m-d_i} \mu_i(x) \geq \frac{1}{d_i!}h_i^{d_i+1} \]  
\[ \sum_{j=0}^{n-d_j} \mu_j(t) \geq \frac{1}{d_j!}r_j^{d_j+1} \]

Combining equations (55)–(57) together, the proof of Theorem 1 is completed. \[ \square \]

**Theorem 2.** For the error functional \( E(x, t) \) defined as (53), if function \( u(x, t) \) satisfies certain smoothness conditions on \( \Omega = [\alpha, \beta] \times [0, T] \), we have

\[ |E_x(x, t)| \leq C(h_i^{d_i+1} + r_j^{d_j+1}), u(x, t) \in C^{d_i+3}[\alpha, \beta] \times C^{d_j+2}[0, T], \]  
\[ |E_t(x, t)| \leq C(h_i^{d_i+1} + r_j^{d_j}), u(x, t) \in C^{d_i+2}[\alpha, \beta] \times C^{d_j+3}[0, T], \]  
\[ |E_{xx}(x, t)| \leq C(h_i^{d_i-1} + r_j^{d_j+1}), u(x, t) \in C^{d_i+4}[\alpha, \beta] \times C^{d_j+2}[0, T]. \]

**Proof.** By equation (53), we know

\[ E_x(x, t) = u_x(x, t) - \frac{dR_{m,n}(x, t)}{dx} \]
\[ = (x - x_i)(x - x_{i+1}), \ldots, (x - x_{i+d_i})u_x[x_i, x_{i+1}, \ldots, x_{i+d_i}, x, t] + (t - t_j)(t - t_{j+1}), \ldots, (t - t_{j+d_j})u_x[t_j, t_{j+1}, \ldots, t_{j+d_j}, x, t] \]
\[ = \sum_{i=0}^{m-d_i} (-1)^iu[x_i, x_{i+1}, \ldots, x_{i+d_i}, x, t] + \sum_{j=0}^{n-d_j} (-1)^ju[t_j, t_{j+1}, \ldots, t_{j+d_j}, x, t] \]
\[ \sum_{i=0}^{m-d_i} \mu_i(x) + \sum_{j=0}^{n-d_j} \mu_j(t) \]

Combining equations (56), (57), and (61), the error estimate (58) is obtained. The proof of (59) and (60) is similar.

Let \( u(x_m, t_n) \) be the numerical solution of function \( u(x, t) \) as follows:

\[ \mathcal{D}u(x_m, t_n) = f(x, t), \]
\[ \lim_{m,n \to \infty} \mathcal{D}u(x_m, t_n) = f(x, t). \]
Table 1: Errors of the barycentric rational collocation methods in the case of Chebyshev nodes with $m \times n = 20 \times 20$ for Example 1.

| $d_1 \times d_2$ | $|u(x, t) - u(x_m, t_n)|$ | $|u(x, t) - u(x_m, t_n)/|u(x, t)|$ | $\|u(x, t) - u(x_m, t_n)\|_2$ | $\|u(x, t) - u(x_m, t_n)/\|u(x, t)\|_2$ |
|-----------------|--------------------------|---------------------------------|-----------------------|--------------------------|
| 1 × 1           | 3.2384e-04              | 3.5797e-04                      | 2.7503e-03           | 3.0402e-03               |
| 2 × 2           | 1.4959e-06              | 1.6536e-06                      | 1.0386e-05           | 1.1481e-05               |
| 3 × 3           | 1.4776e-06              | 1.6334e-06                      | 1.2311e-05           | 1.3609e-05               |
| 4 × 4           | 3.0983e-07              | 3.4249e-07                      | 2.5724e-06           | 2.8438e-06               |
| 5 × 5           | 8.4484e-09              | 9.3388e-09                      | 3.9362e-08           | 4.3510e-08               |
| 6 × 6           | 2.4148e-08              | 2.6693e-08                      | 1.8519e-07           | 2.0471e-07               |
| 7 × 7           | 6.3028e-09              | 6.9671e-09                      | 4.3475e-08           | 4.8057e-08               |
| 8 × 8           | 2.3539e-09              | 2.6020e-09                      | 1.5647e-08           | 1.7296e-08               |
| 9 × 9           | 1.5231e-09              | 1.6837e-09                      | 9.7012e-09           | 1.0724e-08               |
| 10 × 10         | 4.0990e-10              | 4.5310e-10                      | 2.0206e-09           | 2.3473e-09               |
| 11 × 11         | 3.8695e-10              | 4.2774e-10                      | 2.0091e-09           | 2.2208e-09               |
| 12 × 12         | 1.1561e-10              | 1.2779e-10                      | 6.3747e-10           | 7.0466e-10               |
| 13 × 13         | 6.1870e-11              | 6.8391e-11                      | 2.9800e-10           | 3.2941e-10               |
| 14 × 14         | 3.5908e-11              | 3.9693e-11                      | 1.9690e-10           | 2.1765e-10               |
| 15 × 15         | 3.2042e-12              | 3.5149e-12                      | 1.2566e-11           | 1.3890e-11               |
| 16 × 16         | 7.2335e-12              | 7.9960e-12                      | 4.3264e-11           | 4.7824e-11               |
| 17 × 17         | 3.1590e-12              | 3.4919e-12                      | 2.1361e-11           | 2.3613e-11               |
| 18 × 18         | 3.3373e-13              | 3.6891e-13                      | 2.4989e-12           | 2.7623e-12               |
| 19 × 19         | 1.3369e-12              | 1.4778e-12                      | 1.1956e-11           | 1.3216e-11               |

Table 2: Absolute errors and convergence rates in the case of equidistant nodes with time interpolation parameter $d_1 = 9$ for Example 1.

| $m \times n$ | $d_1 = 1$ | $d_1 = 2$ | $d_1 = 3$ | $d_1 = 4$ | $h^n$ |
|--------------|-----------|-----------|-----------|-----------|-------|
| 10 × 10      | 1.0477e-03| 4.0347e-05| 1.6121e-04| 3.7427e-05| —     |
| 20 × 20      | 3.1401e-04| 3.2019e-06| 8.8537e-06| 2.0848e-06| 4.17  |
| 40 × 40      | 1.0667e-04| 6.7471e-07| 5.2818e-07| 7.8919e-08| 4.72  |

Table 3: Absolute errors and convergence rates in the case of equidistant nodes with space interpolation parameter $d_1 = 9$ for Example 1.

| $m \times n$ | $d_1 = 1$ | $d_1 = 2$ | $d_1 = 3$ | $d_1 = 4$ | $r^n$ |
|--------------|-----------|-----------|-----------|-----------|-------|
| 10 × 10      | 4.9734e-05| 4.8510e-05| 4.8182e-05| 4.8190e-05| —     |
| 20 × 20      | 2.9590e-06| 2.9872e-07| 2.4811e-08| 2.4818e-08| 10.92 |
| 40 × 40      | 1.0505e-06| 5.3110e-08| 4.7796e-11| 4.6032e-11| 9.07  |

Table 4: Absolute errors and convergence rates in the case of Chebyshev nodes with time interpolation parameter $d_1 = 9$ for Example 1.

| $m \times n$ | $d_1 = 1$ | $d_1 = 2$ | $d_1 = 3$ | $d_1 = 4$ | $h^n$ |
|--------------|-----------|-----------|-----------|-----------|-------|
| 10 × 10      | 1.2075e-03| 1.5664e-05| 6.9893e-05| 2.6630e-05| —     |
| 20 × 20      | 3.2381e-04| 1.4962e-06| 1.4776e-06| 3.0960e-07| 6.43  |
| 40 × 40      | 9.5077e-05| 2.1888e-07| 5.4753e-08| 4.5608e-09| 6.08  |

Table 5: Absolute errors and convergence rates in the case of Chebyshev nodes with space interpolation parameter $d_1 = 9$ for Example 1.

| $m \times n$ | $d_1 = 1$ | $d_1 = 2$ | $d_1 = 3$ | $d_1 = 4$ | $r^n$ |
|--------------|-----------|-----------|-----------|-----------|-------|
| 10 × 10      | 2.5041e-06| 8.4087e-07| 5.9037e-07| 5.9032e-07| —     |
| 20 × 20      | 5.7915e-07| 3.4124e-08| 1.5666e-09| 1.5257e-09| 8.60  |
| 40 × 40      | 1.3292e-07| 4.0109e-09| 3.5787e-12| 1.5124e-11| 6.66  |
\[ \frac{\partial u(x,t)}{\partial t} - DU(x_m, t_n) = u_t(x, t) + u(x, t)u_s(x, t) - \nu u_{xx}(x, t) - [u_t(x_m, t_n) + u(x_m, t_n)u_s(x_m, t_n) - \nu u_{xx}(x_m, t_n)] \\
= [u_t(x, t) - u_t(x_m, t_n)] + [u(x, t)u_s(x, t) - u(x_m, t_n)u_s(x_m, t_n)] - [\nu u_{xx}(x, t) - \nu u_{xx}(x_m, t_n)] \quad (65) \]

where
As for the first term $A_1$ in equation (65), we know

\[ A_1 = u_t(x,t) - u_t(x_m,t_n), \]

\[ A_2 = u(x,t) u_x(x,t) - u(x_m,t_n) u_x(x_m,t_n), \]

\[ A_3 = -[\nu u_{xx}(x,t) - \nu u_{xx}(x_m,t_n)]. \]

\[ A_1 = u_t(x,t) - u_t(x_m,t_n) \]

\[ = [u_t(x,t) - u_t(x_m,t)] + [u_t(x_m,t) - u_t(x_m,t_n)] \]

\[ = \sum_{s_1=0}^{m-d_1} (-1)^{s_1} u_t \left[ x_{s_1+1}, \ldots, x_{s_1+d_1}, x, t \right] + \sum_{s_2=0}^{n-d_2} (-1)^{s_2} u_t \left[ x_{s_2+1}, \ldots, x_{s_2+d_2}, x_m, t \right] \]

\[ = A_{11} + A_{12}. \]
Table 10: Absolute errors and convergence rates in the case of equidistant nodes with time interpolation parameter $d_2 = 9$ for Example 3.

| $m \times n$ | $d_1 = 1$ | $r\alpha$ | $d_1 = 2$ | $r\alpha$ | $d_1 = 3$ | $r\alpha$ | $d_1 = 4$ | $r\alpha$ |
|--------------|------------|---------|------------|---------|------------|---------|------------|---------|
| 10 $\times$ 10 | 3.2588e-04 | —       | 4.8980e-04 | —       | 1.5235e-04 | —       | 1.3054e-05 | —       |
| 20 $\times$ 20 | 3.8769e-05 | 3.07    | 2.6580e-05 | 4.20    | 8.2946e-06 | 4.20    | 2.1865e-07 | 5.90    |
| 40 $\times$ 40 | 4.7871e-06 | 3.02    | 3.6399e-06 | 2.87    | 1.7144e-06 | 2.27    | 2.1392e-08 | 3.35    |

Table 11: Absolute errors and convergence rates in the case of equidistant nodes with space interpolation parameter $d_1 = 9$ for Example 3.

| $m \times n$ | $d_1 = 1$ | $r\alpha$ | $d_1 = 2$ | $r\alpha$ | $d_1 = 3$ | $r\alpha$ | $d_1 = 4$ | $r\alpha$ |
|--------------|------------|---------|------------|---------|------------|---------|------------|---------|
| 10 $\times$ 10 | 1.0508e-02 | —       | 1.9404e-03 | —       | 1.3672e-04 | —       | 3.2741e-05 | —       |
| 20 $\times$ 20 | 4.4923e-03 | 1.23    | 3.8848e-04 | 2.32    | 2.0154e-05 | 2.76    | 1.9964e-06 | 4.04    |
| 40 $\times$ 40 | 1.7248e-03 | 1.38    | 7.4068e-05 | 2.39    | 2.0977e-06 | 3.26    | 9.4904e-08 | 4.39    |

Table 12: Absolute errors and convergence rates in the case of Chebyshev nodes with time interpolation parameter $d_2 = 9$ for Example 3.

| $m \times n$ | $d_1 = 1$ | $r\alpha$ | $d_1 = 2$ | $r\alpha$ | $d_1 = 3$ | $r\alpha$ | $d_1 = 4$ | $r\alpha$ |
|--------------|------------|---------|------------|---------|------------|---------|------------|---------|
| 10 $\times$ 10 | 2.3715e-02 | —       | 1.5651e-03 | —       | 8.6548e-04 | —       | 6.6468e-05 | —       |
| 20 $\times$ 20 | 5.0071e-03 | 2.11    | 1.3604e-04 | 3.52    | 6.7864e-05 | 3.67    | 1.0970e-06 | 5.92    |
| 40 $\times$ 40 | 6.8410e-04 | 3.01    | 6.6176e-06 | 4.36    | 2.2923e-06 | 4.89    | 1.1683e-08 | 6.55    |

Table 13: Absolute errors and convergence rates in the case of Chebyshev nodes with space interpolation parameter $d_1 = 9$ for Example 3.

| $m \times n$ | $d_1 = 1$ | $r\alpha$ | $d_1 = 2$ | $r\alpha$ | $d_1 = 3$ | $r\alpha$ | $d_1 = 4$ | $r\alpha$ |
|--------------|------------|---------|------------|---------|------------|---------|------------|---------|
| 10 $\times$ 10 | 3.2411e-03 | —       | 3.2588e-04 | —       | 2.9469e-05 | —       | 4.1816e-06 | —       |
| 20 $\times$ 20 | 8.1489e-04 | 1.99    | 3.8769e-05 | 3.07    | 2.0032e-06 | 3.88    | 1.2569e-07 | 5.06    |
| 40 $\times$ 40 | 1.9254e-04 | 2.08    | 4.7871e-06 | 3.02    | 1.2704e-07 | 3.98    | 3.8042e-09 | 5.05    |

Then, we get

![Figure 3: Exact solution, numerical solution, and error of Example 3 ($m = n = 40, d_1 = 4, d_2 = 4$).](attachment:Figure3.png)
\[|A_1| = |A_{11} + A_{12}| \leq |E_i(x,t)| + |E_i(x_m,t)| \leq C(h^{d+1} + \rho^{d_i}). \quad (68)\]

Considering the second term \(A_2\) of equation (65), we have

\[
A_2 = u(x,t)u_t(x,t) - u(x_m,t_n)u_t(x_m,t_n)
\]

\[
= \left[ u(x,t)u_t(x,t) - u(x_m,t)u_t(x,t) \right] + \left[ u(x_m,t)u_t(x,t) - u(x_m,t)u_t(x_m,t) \right]
\]

\[
+ \left[ u(x_m,t)u_t(x_m,t) - u(x_m,t)u_t(x_m,t_n) \right] + \left[ u(x_m,t)u_t(x_m,t_n) - u(x_m,t_n)u_t(x_m,t_n) \right]
\]

\[
= A_{21} + A_{22} + A_{23} + A_{24}
\]

\[
= \sum_{i=0}^{m-d_i} (-1)^i u_t(x,t)u(x_{i+1}, \ldots, x_{i+d_i}, x, t) + \sum_{i=0}^{m-d_i} (-1)^i u_t(x_m,t)u(x_{i+1}, \ldots, x_{i+d_i}, x, t)
\]

\[
+ \sum_{i=0}^{n-d_i} (-1)^i u(x_m,t)u_t(x_{i+1}, \ldots, x_{i+d_i}, x, t) + \sum_{i=0}^{n-d_i} (-1)^i u(x_m,t_n)u_t(x_{i+1}, \ldots, x_{i+d_i}, x, t)
\]

\[
= \sum_{i=0}^{m-d_i} \mu_i(x) + \sum_{i=0}^{n-d_i} \mu_i(x)
\]

Then, we see

\[|A_2| = |A_{21} + A_{22} + A_{23} + A_{24}|
\]

\[\leq C(|E_i(x,t)| + |E_i(x_m,t)| + |E(x_m,t)|)
\]

\[\leq C(h^{d+1} + \rho^{d_i}). \quad (70)\]

\[A_3 = \nu u_{xx}(x,t) - \nu u_{xx}(x_m,t_n)
\]

\[
= \left[ \nu u_{xx}(x,t) - \nu u_{xx}(x_m,t) \right] + \left[ \nu u_{xx}(x_m,t) - \nu u_{xx}(x_m,t_n) \right]
\]

\[
= A_{31} + A_{32}
\]

\[
= \sum_{i=0}^{m-d_i} (-1)^i \nu u_{xx}(x_{i+1}, \ldots, x_{i+d_i}, x, t) + \sum_{i=0}^{n-d_i} (-1)^i \nu u_{xx}(x_{i+1}, \ldots, x_{i+d_i}, x, t)
\]

Then, we have

\[|A_3| = |A_{31} + A_{32}| \leq |A_{31}| + |A_{32}| \leq C(|E_{xx}(x,t)| + |E_{xx}(x_m,t)|)
\]

\[\leq C(h^{d-r+1}). \quad (72)\]

Combining results (68), (70), and (72), the proof is finished. \(\square\)

Remark 1. In the programming of numerical simulation, to deal with the nonlinear characteristic of Burgers’ equation, we adopt the following iteration algorithm:

\[
(I_n \otimes D^{(1)})U_j + \text{diag}(U_{j-1})(C^{(1)} \otimes I_n)U_j - \nu(C^{(2)} \otimes I_n)U_j = F_j, \quad j = 1, 2, 3, \ldots
\]

or the Newton–Rapson iteration algorithm.
4. Numerical Experiments

In this section, some numerical experiments with the barycentric rational collocation method are carried out for Burgers’ equation.

Example 1. Consider the following Burgers’ equation ($\nu = 1$, $\alpha = -4$, $\beta = 4$, $T = 1$):

\[
\begin{cases}
\frac{\partial u(x,t)}{\partial t} + u(x,t) \frac{\partial u(x,t)}{\partial x} - \nu \frac{\partial^2 u(x,t)}{\partial x^2} = 0, & -4 < x < 4, 0 < t < 1, \\
u(-4,t) = \frac{1}{2} - \frac{1}{2} \tanh\left(-1 - \frac{1}{8}t\right), & 0 < t < 1, \\
u(4,t) = \frac{1}{2} - \frac{1}{2} \tanh\left(1 - \frac{1}{8}t\right), & 0 < t < 1, \\
u(x,0) = \frac{1}{2} - \frac{1}{2} \tanh\left(\frac{1}{4}x\right), & -4 < x < 4.
\end{cases}
\] (74)

The analysis solution is chosen to be

\[u(x,t) = \frac{1}{2} - \frac{1}{2} \tanh\left(\frac{1}{4}x - \frac{1}{8}t\right).\] (75)

In Table 1, the errors of the barycentric rational collocation method with Chebyshev nodes with $\alpha = -4$, $\beta = 4$, $m = 20$, $T = 1$, and $n = 20$ are presented. The absolute error

\[E_1 = \|u(x,t) - u(x_m,t_n)\|_2,\]

and relative error

\[E_{r1} = \frac{\|u(x,t) - u(x_m,t_n)\|_2}{\|u(x,t)\|_2},\]

and

\[E_{r2} = \frac{\|u(x,t) - u(x_m,t_n)\|_2}{\|u(x,t)\|_2},\]

are listed.

We can see from Table 1 that the minimum absolute error $E_1$ and $E_2$ can reach $3.6891 \times 10^{-13}$ and $2.7623 \times 10^{-12}$, respectively. The calculation results show that the proposed method has high accuracy feature.

In Table 2, in order to test the convergence rates of space variable in the case of equidistant subdivision, we take the time interpolation parameter with $d_2 = 9$. In Table 3, adopting equidistant subdivision, we take the space interpolation parameter with $d_1 = 9$ to test the convergence rates of time variable. In Table 4, in order to test the convergence rate of space variable in the case of Chebyshev nodes, we take the time interpolation parameter with $d_2 = 9$. In Table 5, adopting Chebyshev nodes, we take the space interpolation parameter with $d_1 = 9$ to test the convergence rates of time variable. Taking $m = 40$, $n = 40$, and $d_1 = d_2 = 4$, Figure 1 shows the exact solution, numerical solution, and error with equidistant nodes for the barycentric rational collocation method.

Example 2. Consider the following Burgers’ equation ($\nu = 0.01$, $\alpha = 0$, $\beta = 1$, $T = 1$):
\[
\begin{aligned}
\frac{\partial u(x, t)}{\partial t} + u(x, t) \frac{\partial u(x, t)}{\partial x} - \nu \frac{\partial^2 u(x, t)}{\partial x^2} &= 0, & 0 < x < 1, 0 < t < 1, \\
u(0, t) &= 0.1 e^{-0.05/\nu (x-0.5)+4.95\nu} + 0.5 e^{-0.25/\nu (x-0.5)+0.75\nu} + e^{-0.5/\nu (x-0.375)}, & 0 < t < 1, \\
u(1, t) &= 0.1 e^{-0.05/\nu (x+0.5)+4.95\nu} + 0.5 e^{-0.25/\nu (x+0.5)+0.75\nu} + e^{-0.5/\nu (1-0.375)}, & 0 < t < 1, \\
u(x, 0) &= 0.1 e^{-0.05/\nu (x-0.5)} + 0.5 e^{-0.25/\nu (x-0.5)} + e^{-0.5/\nu (x-0.375)}, & 0 < x < 1.
\end{aligned}
\]

The analysis solution is set to be
\[
u(x, t) = 0.1 e^{-0.05/\nu (x-0.5)+4.95\nu} + 0.5 e^{-0.25/\nu (x-0.5)+0.75\nu} + e^{-0.5/\nu (x-0.375)}.
\]

In Table 6, in order to test the convergence rates of space variable in the case of equidistant nodes, we take the time interpolation parameter \( d_1 = 9 \). In Table 7, adopting Chebyshev nodes, we take the time interpolation parameter with \( d_1 = 7 \) to test the convergence rates of space variable. In Table 8, in order to test the convergence rates of time variable in the case of equidistant nodes, we take the space interpolation parameter with \( d_1 = 7 \). In Table 9, adopting Chebyshev nodes, we take the space interpolation parameter \( d_1 = 7 \) to test the convergence rates of time variable.

Example 3. Consider the following Burgers’ equation
\[
\begin{aligned}
\frac{\partial u(x, t)}{\partial t} + u(x, t) \frac{\partial u(x, t)}{\partial x} - \nu \frac{\partial^2 u(x, t)}{\partial x^2} &= x \cos (xt) + \frac{t}{2} \sin (2xt) + \frac{t^2}{10} \sin (xt), & x \in (-2, 2), 0 < t < 1, \\
u(-2, t) &= \sin (-2t), & 0 < t < 1, \\
u(2, t) &= \sin (2t), & 0 < t < 1, \\
u(x, 0) &= 0, & x \in (-2, 2).
\end{aligned}
\]

The analysis solution is chosen to be
\[
u(x, t) = \sin (xt).
\]

In Table 10, in order to test the convergence rates of space variable in the case of equidistant subdivision, we take the time interpolation parameter with \( d_1 = 9 \). In Table 11, adopting equidistant subdivision, we take the space interpolation parameter with \( d_1 = 9 \) to test the convergence rates of time variable. In Table 12, in order to test the convergence rate of space variable in the case of Chebyshev nodes, we take the time interpolation parameter with \( d_1 = 9 \). In Table 13, adopting Chebyshev nodes, we take the space interpolation parameter with \( d_1 = 9 \) to test the convergence rates of time variable. Taking \( m = 40, n = 40, d_1 = d_2 = 4 \), Figure 3 shows the exact solution, numerical solution, and error with equidistant nodes for the barycentric rational collocation method.
other papers. In the future, we will research the \((1+2)\) dimensional and \((1+3)\) dimensional Burgers’ equations.

**Data Availability**

No other data were used in this paper.

**Conflicts of Interest**

The authors declare that they have no conflicts of interest.

**Acknowledgments**

The work of the first author was supported by the Foundation of Shandong Jianzhu University (No. H21010Z), the Development Plan of Youth Innovation Team of University in Shandong Province (No. 2021KJ067), the Shandong Province Soft Science Research Project (No. 2020RKB01671), and the Natural Science Foundation of Shandong Province (Nos. ZR2020ZD25 and ZR2021MF009). The work of the second author was supported by the Special Project for Numerical Forecast Development of China Meteorological Administration (No. GRAPES-FZZX-2021).

**References**

[1] C. A. J. Fletcher, "A comparison of finite element and finite difference solutions of the one- and two-dimensional Burgers’ equations," *Journal of Computational Physics*, vol. 51, no. 1, pp. 159–188, 1983.

[2] Z. D. Luo and R. X. Liu, "Mixed finite element analysis and numerical simulation for Burgers’ equation," *Acta Mathematica Sinica*, vol. 21, pp. 257–268, 1999.

[3] A. Dogan, "A Galerkin finite element approach to Burgers’ equation," *Applied Mathematics and Computation*, vol. 157, no. 2, pp. 331–346, 2004.

[4] H. Z. Chen and Z. W. Jiang, "A characteristics-mixed finite element method for Burgers’ equation," *Journal of Applied Mathematics and Computation*, vol. 15, no. 1, pp. 29–51, 2004.

[5] M. Uddin and H. Ali, "The space-time kernel-based numerical method for Burgers’ equations," *Mathematics*, vol. 6, no. 10, p. 212, 2018.

[6] M. Seydaoğlu, "A meshless method for Burgers’ equation using multiquadric radial basis functions with a Lie-Group integrator," *Mathematics*, vol. 7, no. 2, p. 113, 2019.

[7] JP. Berrut, MS. Floater, and G. Klein, "Convergence rates of derivatives of a family of barycentric rational interpolants," *Applied Numerical Mathematics*, vol. 61, no. 9, pp. 989–1000, 2011.

[8] JP. Berrut and G. Klein, "Recent advances in linear barycentric rational interpolation," *Journal of Computational and Applied Mathematics*, vol. 259, pp. 95–107, 2014.

[9] JP. Berrut, SA. Hosseini, and G. Klein, "The linear barycentric rational quadrature method for Volterra integral equations," *SIAM Journal on Scientific Computing*, vol. 36, no. 1, pp. 105–123, 2014.

[10] MS. Floater and K. Hormann, "Barycentric rational interpolation with no poles and high rates of approximation," *Numerische Mathematik*, vol. 107, no. 2, pp. 315–331, 2007.

[11] A. Abdi, JP. Berrut, and S. A. Hosseini, "The linear barycentric rational method for a class of delay Volterra integro-differential equations,” *Journal of Scientific Computing*, vol. 75, no. 3, pp. 1757–1775, 2018.

[12] A. Abdi and S. A. Hosseini, “The barycentric rational difference-quadrature scheme for systems of volterra integro-differential equations,” *SIAM Journal on Scientific Computing*, vol. 40, no. 3, pp. A1936–A1960, 2018.

[13] Z. Wang and S. Li, *Barycentric Interpolation Collocation Method for Nonlinear Problems*, National Defense Industry Press, Beijing, China, 2015.

[14] Z. Wang, Z. Xu, and J. Li, "Mixed barycentric interpolation collocation method of displacement-pressure for incompressible plane elastic problems," *Chinese Journal of Applied Mechanics*, vol. 35, pp. 195–201, 2018.

[15] Z. Wang, L. Zhang, Z. Xu, and J. Li, "Barycentric interpolation collocation method based on mixed displacement-stress formulation for solving plane elastic problems," *Chinese Journal of Applied Mechanics*, vol. 35, pp. 304–309, 2018.

[16] J. Li and Y. Cheng, “Linear barycentric rational collocation method for solving second-order Volterra integro-differential equation,” *Computational and Applied Mathematics*, vol. 39, no. 2, 2020.

[17] J. Li and Y. Cheng, "Barycentric rational method for solving biharmonic equation by depression of order,” *Numerical Methods for Partial Differential Equations*, vol. 37, no. 3, pp. 1993–2007, 2021.

[18] J. Li and Y. Cheng, "Linear barycentric rational collocation method for solving heat conduction equation,” *Numerical Methods for Partial Differential Equations*, vol. 37, no. 1, pp. 533–545, 2021.

[19] Q. Ge and X. P. Zhang, "Numerical solution for third-order two-point boundary value problems with the Barycentric rational interpolation collocation method,” *Journal of Mathematics*, vol. 2021, Article ID 6698615, 6 pages, 2021.

[20] J. Li and Y. Sang, "Linear barycentric rational collocation method for beam force vibration equation,” *Shock and Vibration*, vol. 2021, Article ID 5584274, 11 pages, 2021.

[21] J. Li, X. N. Su, and J. Z. Qu, "Linear barycentric rational collocation method for solving telegraph equation,” *Mathematical Methods in the Applied Sciences*, vol. 44, no. 14, pp. 11720–11737, 2021.

[22] Q. L. Zhao and Y. L. Cheng, "Barycentric rational collocation method for the incompressible forchheimer flow in porous media,” *Journal of Mathematics*, vol. 2021, Article ID 5514916, 8 pages, 2021.