Research Paper
Computational Methods for Differential Equations
http://cmde.tabrizu.ac.ir
Vol. *, No. *, *, pp. 1-11
DOI:10.22034/cmde.2026.71478.3592

Collocation method using auto-correlation functions of compact supported wavelet for solving Burgers' equation

Mahnaz Zafari and Amjad Alipanah*
Department of Applied Mathematics, University of Kurdistan, Sanandaj, Iran.

Abstract

In this paper, we employ a collocation method based on the auto-correlation functions of Daubechies wavelet scaling functions to approximate the solutions of one-dimensional Burgers' equations. By using the properties of these wavelet bases, we approximate the Burgers' equation at dyadic points for x dimensional, effectively reducing it to a system of ordinary differential equations (ODEs). We then apply the fourth-order Runge-Kutta method to solve this system of ODEs. The numerical results obtained demonstrate the efficiency and accuracy of our approach compared to other existing methods.

Keywords. Auto-correlation function, Daubechies wavelet, Collocation method, Burgers' equations, Runge-Kutta.
2010 Mathematics Subject Classification. 65L05, 34K06, 34K28.

Uncorrected Pr

1. Introduction

Studies show that nonlinear equations are essential to many scientific disciplines, such as engineering, physics, and applied mathematics. Because nonlinear processes incorporate several elements that contribute to the fluctuation of each parameter, they have a broad influence. There is still a great need for novel methods to find precise or close solutions to nonlinear partial differential equations in both mathematics and science. It's crucial to remember that nonlinear issues usually don't have precise analytical solutions. Nonetheless, a variety of analytical techniques may be employed by academics to tackle nonlinear problems. In recent years, several authors have focused their study on nonlinear partial differential equation solutions [14, 18, 22]. An English mathematician named Harry Bateman $(1882-1946)$ presented a partial differential equation with the beginning and boundary conditions that go with it in 1915. Burger's equation was later used in 1948 by Dutch scientist Johannes Martinus Burgers $(1895-1981)$ to describe the mathematical modeling of turbulence. Burgers later rose to prominence as a prominent figure in fluid mechanics. This equation is commonly referred to as Burger's equation in recognition of Burgers' contributions. Both Julian David Cole $(1925-1999)$ and Eberhard Hopf $(1902-1983)$ separately developed a transformation that transformed Burger's equation into a linear heat equation that could be solved precisely for any set of beginning circumstances. The Hopf-Cole transformation is the well-known name for this metamorphosis.

The creation of reliable computing algorithms to solve nonlinear partial differential equations (PDEs) in fluid mechanics and heat transport has been the focus of major research over the past several decades. The Burger's equation is a nonlinear PDE that describes a number of physical issues that arise in engineering and are challenging to resolve by nature [6]. Situations where both diffusion and typical nonlinearity occur are modeled using Burger's equation.

There are clear benefits to employing an adaptable grid rather than a static grid when solving a partial differential equation (PDE) numerically. Wavelets are frequently employed for numerical solutions of PDEs on adaptive grids, especially Burger's equation. Daubechies wavelets are used to solve the one-dimensional Burger's equation with periodic boundary conditions on a static grid in [13], and quasi wavelets are used in [25, 26].

Received: 08 February 2026; Accepted: 09 May 2026.
* Corresponding author. Email: a.alipanah@uok.ac.ir.

Numerous numerical techniques, such as finite elements chemes [7, 17], finite difference schemes, spectral methods, and distributed functional approaches, have been devised to solve Burger's problem.

Redundant representations are utilized to make the analysis of coefficients from scale to scale easier when multiresolation methods for signal analysis and image processing are implemented. Prior to the development of compactly supported wavelets in [2, 10], Dubuc [12] and Deslauriers and Dubuc [11] examined the auto-correlation functions of compactly supported scaling functions (also known as the fundamental functions) in the framework of the Lagrange iterative interpolation scheme.

Compactly supported scaling functions have symmetric auto-correlation functions. Named for their inventor, Ingrid Daubechies, Daubechies wavelets are a set of mathematical functions that have found extensive application in image processing, signal processing, and related domains. These wavelets are computationally economical and well-suited for real-world applications because of their compact support, orthogonality, and high degree of smoothness. A multiresolution analysis of a signal is provided by Daubechies wavelets, which enables effective data reduction while maintaining significant characteristics. The Daubechies wavelet family comprises a variety of order P wavelets, where higher degrees of smoothness are associated with rising values of P. As we previously stated, only in dyadic points can we determine the value of the scaling function of Daubechies wavelets. First, we must use a linear system to get the value of $\varphi(k), k \in \mathbb{Z}$. Next, we may use the cascade method to determine the value of $\varphi(x)$ at dyadic points [1, 10, 16]. However, the properties of the auto-correlation functions are superior. In addition to being symmetric, these bases exhibit $P-1$ vanishing moments in wavelet functions and auto-correlation functions of compactly supported scaling [4, 21]. As is well known, the scaling function of the Daubechies wavelets is used to generate the auto-correlation function. The wavelet corresponding to the multiresolution analysis is created by applying the auto-correlation function, which is really the scaling function of a multiresolution analysis. The cascade algorithm is also used to calculate the value of $\theta(k) = \delta_{0,k}$ and for other dyadic points.

In this paper we consider Burger's equation in one-dimension as follows

$$u_t + uu_x = \nu u_{xx}, \tag{1.1}$$

where $\nu > 0$ is the kinematic viscosity parameter related to the Reynolds number $\nu = \frac{1}{R_e}$[13, 14].

The simultaneous presence of nonlinear convective term uu_x and diffusive term νu_{xx} add an additional feature to the Burger's equation. When ν approaches zero, Eq. (1.1) become inviscid Burger's equation, which is a model for nonlinear wave propagation. With Dirichlet conditions

$$u(x,0) = f(x), \quad a \le x \le b, \tag{1.2}$$

$$u(a,t) = g_1(t), \quad u(b,t) = g_2(t), \quad t > 0. \tag{1.3}$$

Many physical problems that occur in science may be mathematically modeled using the unsteady heat equation without internal heat creation. The heat equation is Eq. (1.1) when u gets close to zero. In a solid material, the rate of heat transmission both inside the solid and at its border will change in tandem with changes in temperature over time. Determining the temperature distribution is necessary in many real-world engineering situations in order to compute the rate of local heat transfer, thermal expansion, and thermal stress at a few key locations. Burger's equation, is a basic partial differential equation from fluid mechanics, serves as an excellent illustration for the reasons listed below.

(1) The partial differential equation's precise solution is widely known.
(2) It may be seen as a hyperbolic issue using the heat equation for very small u and artificial diffusion for tiny kinematic viscosity ν.
(3) It may be applied to the computation of the boundary layer for viscous fluid flow.
(4) For the PDE solvers, it creates a typical test issue.
(5) It can be used for analysis in a number of fields, including cosmology, gas dynamics, shock wave theory, growth of molecular interfaces, non-linear wave propagation, sedimentation of polydispersive suspensions and colloids, traffic flow, and longitudinal elastic waves in isotropic solids [6].

In this paper, we approximate Eq. (1.1) and with boundary condition and initial condition by using auto-correlation functions constracted based on Daubechies scaling function of wavelets. Firstly, we approximate domain x of $u(x,t)$

Uncorrected Proof

using auto-correlation function in resolation J and the Eq. (1.1) convert to a system of first order ODE in terms t and them we approximate this system of ODE by Runge-Kutta of order four. Also error analysis, be presented and we implement our method for three examples for different scaling function of Daubechies wavelets in different resolution J. Numerical results are compared with methods in literature.

2. Auto-correlation functions of Daubechies wavelets

A Daubechies scaling function $\varphi(x)$ of order P satisfies the scaling relation [2, 9, 19, 23, 27]

$$\varphi(x) = \sum_{k=0}^{L-1} h_k \varphi(2x-k), \tag{2.1}$$

where h_k are the filter coefficients of the wavelet scale function φ. The function has its support in the interval $[0, L-1]$. The values of scaling function $\varphi(x)$, can be obtained only at dyadic points, $\frac{k}{2^J}, \quad k = 0, 1, 2, \cdots, 2^J(L-1)$. Figure 1, displays the auto-correlation functions of the Daubechies scaling function, $\bar{\theta}(x)$, for orders $P = 2, 3, 4, 5$, computed at a resolution $J = 5$.

Definition 2.1. Let $\varphi(x)$, be the Daubechies scaling function of order P. The auto-correlation function $\bar{\theta}(x)$ of $\varphi(x)$ be defined as

$$\bar{\theta}(x) = (\varphi * \varphi(-.))(x) = \int_{-\infty}^{+\infty} \varphi(y)\,\varphi(y-x)\,dy. \tag{2.2}$$

The function $\bar{\theta}(x)$, defined in (2.2) has well properties as follows

(1) $\bar{\theta}(x) = \sum_{k=-L+1}^{L-1} a_k \bar{\theta}(2x-k)$, where $a_k = a_{-k} = \sum_{i=0}^{L-1-k} h_i h_{i+k}, \quad k \geq 0$,
(2) $\text{supp}(\bar{\theta}) \subseteq [-L+1, L-1]$,
(3) $\bar{\theta}(k) = \delta_{0,k}, \quad k \in \mathbb{Z}$,
(4) $a_{2k} = \delta_{0,k}, \quad k \in \mathbb{Z}$,
(5) $a_k = \bar{\theta}(\frac{k}{2}), \quad k \in \mathbb{Z}$,
(6) $\bar{\theta}(x) = \sum_{k=-L+1}^{L-1} \bar{\theta}(\frac{k}{2})\bar{\theta}(2x-k)$,
(7) If $\bar{\theta}$ is the function defined in (2.2), then we have $\int_{-\infty}^{\infty} \bar{\theta}(x)dx = 1$.

From two scale relation (2.1), it is easy to verify that

$$\bar{\theta}(x) = \bar{\theta}(2x) + \frac{1}{2}\sum_{r=1}^{\frac{L}{2}} a_{2r-1}\Big(\bar{\theta}(2x-2r+1) + \bar{\theta}(2x+2r-1)\Big). \tag{2.3}$$

3. Evaluation of auto-correlation dervatives

By differentiating n times from Eq. (2.3) we have

$$\bar{\theta}^{(n)}(x) = 2^n \left(\bar{\theta}^{(n)}(2x) + \frac{1}{2}\sum_{r=1}^{\frac{L}{2}} a_{2r-1}\left(\bar{\theta}^{(n)}(2x-2r+1) + \bar{\theta}^{(n)}(2x+2r-1)\right)\right). \tag{3.1}$$

Similar the Daubechies scaling function and corresponded auto-correlation function there is not analytical expression for $\bar{\theta}^{(n)}(x), \quad n = 1, 2, \cdots$, for a $x \in [0, L-1)$, but the values at dyadic points can be computed up to the machine precision.

Firstly we obtain $\bar{\theta}^{(n)}(k)$ for all integer $k = -L+1, \cdots, L-1$, which that can be obtained by solving a linear system which is derived from the scaling relation (2.3) [18, 26].

Then the values of $\bar{\theta}^{(n)}(\frac{k}{2^J})$ at dyadic points can be obtained recursively using the cascade algorithm method accurately, in this paper, only derivative values at integer points are required. Actually $\bar{\theta}^{(n)}(k)$, be obtained by solving a linear system which be given as the following proposition. Then by using Eq. (3.1), $\bar{\theta}^{(n)}(x_k), \quad x = (-L+1) + 2^{-J}k, \quad k = 0, 1, \cdots, (L-1)2^J$ be obtained explicitly.

Uncorrected Proof

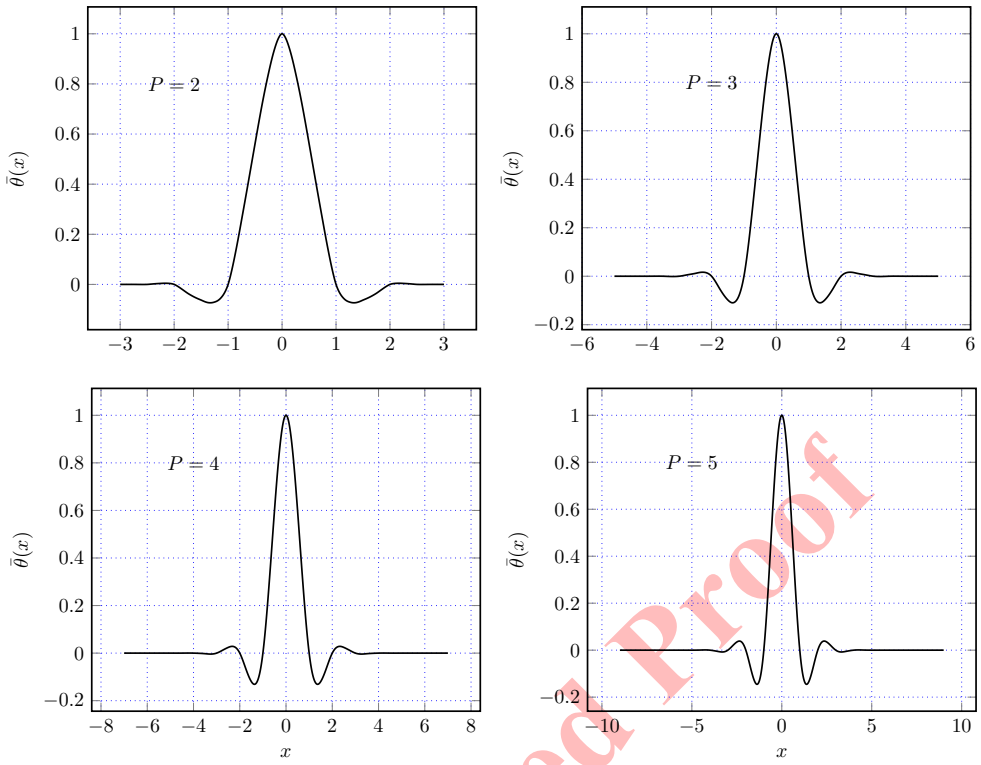

FIGURE 1. Plots of the auto-correlation functions $\bar{\theta}(x)$, for $P = 2, 3, 4, 5$, and $J = 5$.

Uncorrected Proof

Proposition 3.1. *[5] If we show the nth derivative of the aut-ocorrelation function $\bar{\theta}$ by $\tau_x^{(n)} := \bar{\theta}^{(n)}(x)$, the coefficients $\tau_x^{(n)}$, $x = 0, 1, 2, \cdots L-2$ and for $n = 1, 2, \cdots$ satisfy the following system of linear algebraic equations*

$$\tau_x^{(n)} = 2^n \left[\tau_{2x} + \frac{1}{2} \sum_{l=1}^{\frac{L}{2}} a_{2l-1} \left(\tau_{2x-2l+1} + \tau_{2x+2l-1} \right) \right], \tag{3.2}$$

and

$$\sum_{x=-(L-2)}^{L-2} x^n \tau_x^{(n)} = (-1)^n n!, \tag{3.3}$$

in which for special cases $n = 1$, $\tau_x^{(1)} = -\tau_{-x}^{(1)}$ and if $n = 2$, we have $\tau_x^{(2)} = \tau_{-x}^{(2)}$.

Let $f \in L^2[0, 1]$, we approximate f in resolution J as follows

$$I_J(f) = \sum_{k=0}^{2^J} f\left(\frac{k}{2^J}\right) \bar{\theta}(2^J x - k), \tag{3.4}$$

and from bellow theorem it is clearly that $I_J(f)$ uniform convergence to f.

Theorem 3.2. *[3] If $f \in H^{L-1}([0, 1])$ and $0 \le r \le s \le (L-1), s \ge 1$, then*

$$\|f - I_J(f)\|_r \le C 2^{-J(s-r)} \|f\|_s. \tag{3.5}$$

Now we consider the following definition [3]

$$\hat{\Theta}_{Jk}(x)=\begin{cases}\bar{\theta}(2^Jx-k)+\sum_{n=1-L}^{-1}a_{nk}\bar{\theta}(2^Jx-n), & k=0,\cdots,\frac{L}{2},\\ \bar{\theta}(2^Jx-k), & k=\frac{L}{2}+1,\cdots,2^J-\frac{L}{2}-1,\\ \bar{\theta}(2^Jx-k)+\sum_{n=2^J+1}^{2^J+L-1}b_{nk}\bar{\theta}(2^Jx-n), & k=2^J-\frac{L}{2},\cdots,2^J.\end{cases}\tag{3.6}$$

in which the coefficients a_{nk} and b_{nk} are defined by

$$a_{nk}=l^1_{Jk},\qquad b_{nk}=l^2_{Jk}$$

where $l^1_{Jk}(x)$ and $l^2_{Jk}(x)$ represent Lagrange interplation polynomials, defined by

$$l^1_{Jk}(x)=\prod_{i=0,i\neq k}^{\frac{L}{2}}\frac{x-i2^{-J}}{k2^{-J}-i2^{-J}},\qquad l^2_{Jk}(x)=\prod_{i=2^J-\frac{L}{2},i\neq k}^{2^J}\frac{x-i2^{-J}}{k2^{-J}-i2^{-J}}.$$

At the boundary regions, the compact support of the Daubechies autocorrelation functions extends outside the computational domain, causing the loss of essential approximation properties such as polynomial reproduction and the two-scale relation. Eq. (3.6) introduces Lagrange-interpolation–based corrective combinations that compensate for this deficiency and construct boundary-adapted basis functions whose behavior remains consistent with that of the interior wavelets.

4. Solving Burger's equation

For simplicity in Eq. (1.1) we let $a=0,\quad b=L-1$ and we approximate $u(x,t)$ in Eq. (1.1) as follows

$$I_J(u):=u(x,t)\simeq\mathbf{u}_J(x,t):=\sum_{k=0}^{N}U(x_k,t)\hat{\Theta}(2^Jx-k),\tag{4.1}$$

where $N=(L-1)2^J$ and $\hat{\Theta}$ is auto-correlation function in which $\hat{\Theta}_{J,k}\left(\frac{r}{2^J}\right)=\delta_{k,r}$. If we define $U_k(t):=U(x_k,t)$ and $\hat{\Theta}_{J,k}(x)=\hat{\Theta}(2^Jx-k)$ then Eq. (4.1) can be written as

$$\mathbf{u}_J(x,t)=\hat{\Theta}^T_J(x)\mathbf{U}(t),\tag{4.2}$$

where

$$\hat{\Theta}_J(x)=\left[\hat{\theta}_{J,1}(x),\hat{\theta}_{J,2}(x),\cdots,\hat{\theta}_{J,N}(x)\right]^T,$$

and

$$\mathbf{U}(t)=[U_1(t),U_2(t),\cdots,U_N(t)]^T.$$

By substuting Eq.(4.2) in Eq. (1.1) we have

$$\hat{\Theta}^T_J(x)\frac{d\mathbf{U}(t)}{dt}+\left(\hat{\Theta}^T_J(x)\mathbf{U}(t)\right)\left(\frac{d\hat{\Theta}^T_J(x)}{dx}\mathbf{U}(t)\right)=\nu\frac{d^2\hat{\Theta}^T_J(x)}{dx^2}\mathbf{U}(t).$$

In the above equation if we let $x=x_r=\frac{r}{2^J},\ r=1,2,\cdots,(L-1)2^J$, then we have

$$\mathbf{e}^T_r\frac{d\mathbf{U}(t)}{dt}+\left(\mathbf{e}^T_r\mathbf{U}(t)\right)\left(\frac{d\hat{\Theta}^T_J(x_r)}{dx}\mathbf{U}(t)\right)=\nu\frac{d^2\hat{\Theta}^T_J(x_r)}{dx^2}\mathbf{U}(t),\tag{4.3}$$

so

$$\frac{dU_r(t)}{dt}+U_r(t)\left(\frac{d\hat{\Theta}^T_J(x_r)}{dx}\mathbf{U}(t)\right)=\nu\frac{d^2\hat{\Theta}^T_J(x_r)}{dx^2}\mathbf{U}(t),\quad r=1,\cdots,N-1.\tag{4.4}$$

Uncorrected Proof

The system of equations given in Eq. (4.4) can be written as follows

$$\frac{d\mathbf{U}(t)}{dt} + \mathbf{D}(\mathbf{U}(t))A^T\mathbf{U}(t) = \nu B^T\mathbf{U}(t), \tag{4.5}$$

where the matrices A, B and $\mathbf{D}(u)$ are given as

$$B_{rk} = \hat{\theta}''(2^J x_r - k) = \hat{\theta}''(r-k),$$
$$A_{rk} = \hat{\theta}'(2^J x_r - k) = \hat{\theta}'(r-k),$$

and

$$\mathbf{D}(u) = \operatorname{diag}\left[U_1(t), U_2(t), \cdots, U_{N-1}(t)\right],$$

also,

$$\mathbf{e}_k = [0, \cdots, 1, 0, \cdots, 0]^T.$$

Eq. (4.5) is a system of ordinary differential equations, which can be rewrite as

$$\frac{d\mathbf{U}(t)}{dt} = F(t, \mathbf{U}(t)), \quad t \in (0, T), \tag{4.6}$$

where

$$F(t, \mathbf{U}(t)) := -\mathbf{D}(\mathbf{U}(t))A^T\mathbf{U}(t) + \nu B^T\mathbf{U}(t),$$

in which the initial condition for Eq. (4.6) is $\mathbf{U}(0) = [f(a), f(a+\frac{1}{2^J}), \cdots, f(b)]^T$.

We solve Eq. (4.6) by using the $RK4$ method for $T = 0.001$.

5. Error Analysis

In this section, we analyze the error of the proposed numerical scheme. For simplicity, we define the truncation error and establish the convergence order of our method. Let $u(x,t)$ be the exact solution of the Burgers' equation (1.1) with boundary and initial conditions (1.2) and (1.3). Let $\mathbf{u}_j(x,t)$ be the corresponding numerical solution obtained by the aut-corrolation method, specifically by applying a fourth-order Runge-Kutta method to the semi-discrete system (4.6). The spatial discretization resolution parameter is J, and Δt is the time step.

Definition 5.1. (Residual Operator) Let $w(x,t)$, be a sufficiently smooth function, then the nonlinear residual operator $\mathcal{R}(w)$ be defined as

$$\mathcal{R}(w) := w_t + ww_x - \nu w_{xx}. \tag{5.1}$$

From this definition, it follows immediately that if u is the exact solution of (1.1) with boundary and intial condtions given in equations (1.2) and (1.3), then the residual is identically zero, i.e, $\mathcal{R}(u) \equiv 0$.

Theorem 5.2. *Let u be the exact solution of (1.1)-(1.3) and $\mathbf{u}_j(x,t)$ be the numerical solution of our method. Assume the exact solution u is sufficiently smooth, with regularity index s. Then there exist a constant C_1 independent of resolution J and time step Δt such that*

$$\|\mathcal{R}(u) - \mathcal{R}(\mathbf{u}_J)\| \leq O\left((\Delta t)^4\right) + C_1 2^{-J(s-2)}. \tag{5.2}$$

Proof. From Equations (4.1) and (5.1), we have

$$\mathcal{R}(u) - \mathcal{R}(\mathbf{u}_J) = (u_t - \mathbf{u}_t) + (uu_x - \mathbf{u}\mathbf{u}_x) - \nu\left(u_{x,x} - \mathbf{u}_{x,x}\right), \tag{5.3}$$

then, by using the Theorem 3.2, we have

$$\|\mathcal{R}(u) - \mathcal{R}(\mathbf{u}_J)\| = \|\left(u_t - \mathbf{u}_t\right) + \left(uu_x - \mathbf{u}\mathbf{u}_x\right) - \nu\left(u_{x,x} - \mathbf{u}_{x,x}\right)\| \tag{5.4}$$
$$\leq \|u_t - \mathbf{u}_t\| + |\nu|\|u_{x,x} - \mathbf{u}_{x,x}\| + \|uu_x - \mathbf{u}\mathbf{u}_x, \tag{5.5}$$

in the above equation for obtaining a bound for $\|uu_x - \mathbf{u}\mathbf{u}_x\|$ we using the Equation (1.1), so

$$\|uu_x - \mathbf{u}\mathbf{u}_x\| = \|(u_t - \mathbf{u}_t) - \nu(u_{xx} - \mathbf{u}_{xx})\|$$
$$\leq \|u_t - \mathbf{u}_t\| + |\nu|\|u_{xx} - \mathbf{u}_{xx}\|. \tag{5.6}$$

Uncorrected Proof

Now from Equations (5.4)-(5.6) and Theorem 3.2, als for $r = 0, 1$ we conclude that

$$\|\mathcal{R}(u) - \mathcal{R}(\mathbf{u}_J)\| \leq O\left((\Delta t)^4\right) + C_1 2^{-J(s-2)},$$

in which $C1$ is a constant independet of resolution J and Δt. □

6. Numerical results

In this section, we present three numerical results to evaluate the efficiency and accuracy of our method. We compare our approach with other methods available in the literature. Additionally, we demonstrate that as the resolution increases, the accuracy improves, and the computational time decreases compared to the other methods discussed in [15, 17].

$$L_\infty = \left|\left|u(x,T) - \mathbf{u}(x,T)\right|\right|_\infty = \max\left\{\left|u(x_j,T) - U_j(T)\right| : \quad j = 1, 2, \cdots, N-1\right\}. \tag{6.1}$$

Example 6.1. We consider Burger's equation (1.1) with initial and boundary conditions in the following form

$$u(x,0) = \sin(\pi x), \quad 0 \leq x \leq 1, \tag{6.2}$$

$$u(0,t) = u(1,t) = 0, \quad t > 0. \tag{6.3}$$

The exact Fourier series solution of this problem given obtained as follows

$$u(x,t) = 2\pi\nu \frac{\sum_{n=1}^{\infty} a_n exp(-n^2\pi^2\nu t) n\sin(n\pi x)}{a_0 + \sum_{n=1}^{\infty} a_n exp(-n^2\pi^2\nu t)\cos(n\pi x)}, \tag{6.4}$$

where the Fourier coefficients are

$$a_0 = \int_0^1 exp\Big(-\frac{1-\cos(\pi x)}{2\pi\nu}\Big)dx, \tag{6.5}$$

$$a_n = 2\int_0^1 exp\Big(-\frac{1-\cos(\pi x)}{2\pi\nu}\Big)cos(n\pi x)dx, \qquad n = 1, 2, 3, \cdots. \tag{6.6}$$

The numerical results for the Example 6.1 are presented for $\nu = 0.1, \nu = 0.01$ in the Table 1. Table 1, it is concluded that the present scheme gives better results in [15, 17].

Table 1. Comparison between exact and numerical solutions of Example 6.1 for $\nu = 0.1, \quad \nu = 0.01$ at different time T with resolution $J = 4$.

x		$\nu = 0.1$				$\nu = 0.01$			
	T	[17]	[15]	Our method	Exact	[17]	[15]	Our method	Exact
0.25	0.4	0.31429	0.30880	0.3088942	0.3088942	0.34819	0.34191	0.3419149	0.3419149
	0.8	0.19758	0.19565	0.1956756	0.1956756	0.22752	0.22151	0.2214819	0.2214819
	1.0	0.16391	0.16251	0.1625650	0.1625648	0.19375	0.18814	0.1881939	0.1881939
	3.0	0.02743	0.02720	0.0272024	0.0272023	0.07754	0.07537	0.0751141	0.0751141
0.50	0.4	0.57636	0.56953	0.5696323	0.5696324	0.66543	0.66070	0.6607109	0.6607109
	0.8	0.36245	0.35922	0.3592369	0.3592361	0.44526	0.43913	0.4391382	0.4391382
	1.0	0.29437	0.29190	0.2919171	0.2919159	0.38047	0.37434	0.3744200	0.3744200
	3.0	0.04057	0.04020	0.0402051	0.0402049	0.15362	0.15008	0.1501791	0.1501790
0.75	0.4	0.62592	0.62554	0.6254443	0.6254379	0.91201	0.91027	0.9102675	0.9102681
	0.8	0.37713	0.37409	0.3739273	0.3739217	0.65254	0.64739	0.6474014	0.6473952
	1.0	0.29016	0.28746	0.2874779	0.2874744	0.56157	0.55599	0.5560597	0.5560507
	3.0	0.01334	0.02977	0.0297723	0.0297721	0.22874	0.22481	0.2247935	0.2248112

Example 6.2. We now consider Burger's Eq. (1.1) with initial and boundary conditions in the following form

$$u(x,0) = 4x(1-x), \quad 0 \leq x \leq 1. \tag{6.7}$$

$$u(0,t) = u(1,t) = 0, \quad t > 0. \tag{6.8}$$

Uncorrected Proof

The exact Fourier series solution of this problem given by Eq. (6.4), and the Fourier coefficients are

$$a_0 = \int_0^1 exp\Big(-\frac{x^2\,(3-2x)}{3\nu}\Big)dx, \tag{6.9}$$

$$a_n = 2\int_0^1 exp\Big(-\frac{x^2\,(3-2x)}{3\nu}\Big)\cos(n\pi x)dx, \qquad n = 1, 2, 3, \cdots. \tag{6.10}$$

The numerical results for the Example 6.2 are presented for $\nu = 0.1, \nu = 0.01$ in the Table 2. Table 2 makes a comparison of present results with exact and numerical solutions it is found that the present results in [15, 17].

TABLE 2. Comparison between exact and numerical solutions of Example 6.2 for $\nu = 0.1, \quad \nu = 0.01$ at different time T with resolution $J = 4$.

x		$\nu = 0.1$				$\nu = 0.01$			
	T	[17]	[15]	Our method	Exact	[17]	[15]	Our method	Exact
0.25	0.4	0.32091	0.31744	0.3175228	0.3175228	0.36911	0.36213	0.3622606	0.3622594
	0.8	0.20211	0.19952	0.1995554	0.1995553	0.23703	0.23066	0.2304521	0.2304511
	1.0	0.16782	0.16557	0.1655988	0.1655986	0.20069	0.19468	0.1946912	0.1946904
	3.0	0.02828	0.02775	0.0277588	0.0277587	0.07865	0.07613	0.0761342	0.0761341
0.50	0.4	0.58788	0.58443	0.5845371	0.5845372	0.68818	0.68357	0.6836784	0.6836786
	0.8	0.37111	0.36733	0.3673992	0.3673982	0.46011	0.45412	0.4537138	0.4537136
	1.0	0.30183	0.29830	0.2983444	0.2983431	0.39206	0.38563	0.3856763	0.3856758
	3.0	0.04185	0.04106	0.0410652	0.0410650	0.15576	015217	0.1521802	0.1521800
0.75	0.4	0.65054	0.64556	0.6456236	0.6456155	0.92194	0.92064	0.9205026	0.9214626
	0.8	0.39068	0.38526	0.3853420	0.3853355	0.66777	0.66303	0.6627259	0.6627204
	1.0	0.30057	0.29582	0.2958608	0.2958567	0.57491	0.56929	0.5693273	0.5693187
	3.0	0.03106	0.03043	0.0304398	0.0304396	0.23183	0.22774	0.2277275	0.2277430

Example 6.3. We consider Burger's equation (1.1) with boundary conditions

$$u(0,t) = u(1,t) = 0, \quad t > 0, \tag{6.11}$$

and with initial condition

$$u(x,0) = \frac{2\nu\pi\sin(\pi x)}{\sigma + \cos(\pi x)}, \tag{6.12}$$

and with exact solution of this problem given obtained as follows

$$u(x,t) = \frac{2\nu\pi exp(-\pi^2\nu t)\sin(\pi x)}{\sigma + exp(-\pi^2\nu t)\cos(\pi x)}, \quad 0 < x < 1, \tag{6.13}$$

where $\sigma > 1$ is a parameter.

In Tables 1 and 2 compare the results of the present scheme with exact and numerical solutions. As you can see, the present results are better than the results in references [15, 17]. In Table 3 show the error results for different values of orders scaling function P and resolution J. The L_∞ error norm of the auto-correlation functions for the Daubechies scaling functions (order P, resolution J) is presented in Figure 2. The results are in excellent agreement with the theoretical error bound established in Theorem 5.2.

TABLE 3. Error results of Example 6.3 for $P = 3, 4, 5, 6$ and $J = 3, 4, 5, 6, 7$ and $\sigma = 100$.

J	$P = 3$	$P = 4$	$P = 5$	$P = 6$
3	$4.09E-04$	$2.36E-05$	$2.09E-5$	$1.07E-5$
4	$1.31E-08$	$1.29E-10$	$1.41E-11$	$5.58E-12$
5	$8.38E-09$	$2.16E-11$	$1.42E-13$	$6.65E-16$
6	$5.26E-10$	$3.50E-13$	$5.27E-16$	$1.21E-19$
7	$3.29E-11$	$5.50E-15$	$2.07E-18$	$8.62E-23$

Uncorrected Proof

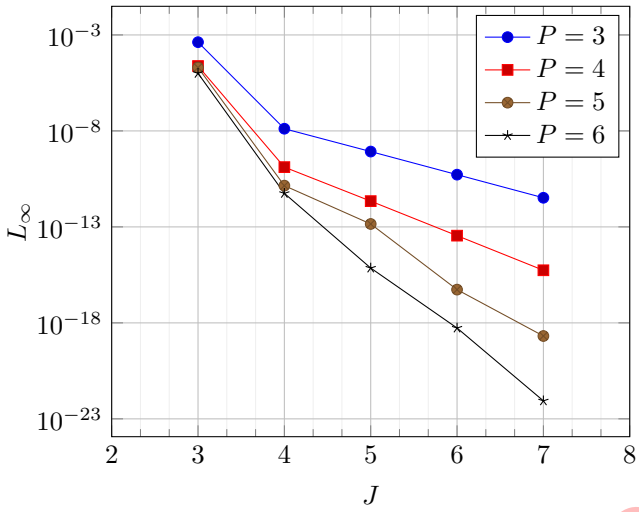

FIGURE 2. The graph of error L_∞ for differnt P and resolution J for Example 6.3.

7. Conclusion

In this paper, we proposed a method based on the auto-correlation functions of Daubechies wavelet scaling functions to solve the Burgers' equation. By precomputing and storing the necessary autocorrelation function data and their derivatives at dyadic points, we significantly reduced the computational time. The resulting system of ODEs was then solved using the fourth-order Runge-Kutta method. We analyzed the convergence of the proposed numerical scheme and compared its performance with existing methods through numerical experiments, as presented in Tables 1 and 2. The results demonstrate that our method offers superior efficiency and accuracy compared to those reported in the literature.

Comparison with previous works, in order to demonstrate the accuracy, efficiency, and stability of the proposed collocation approach based on the auto-correlation functions of Daubechies scaling functions, its results are compared with several recent and well-established numerical schemes that have been applied to Burgers' equation. The comparison indicates that the present approach achieves a much higher accuracy (up to 10^{-23} in several test cases) while retaining low computational cost. Moreover, the theoretical convergence proven in Theorem 3.2 confirms uniform convergence of $I_J(f)$ to f, which is rarely established explicitly in earlier wavelet-based works. Consequently, the auto-correlation Daubechies collocation method provides an optimal balance between accuracy, computational efficiency, and theoretical robustness, outperforming existing wavelet-based schemes for classical Burgers' problems and offering a promising framework for fractional and multi-dimensional extensions in future research.

In the comparison made with reference [24], it was found that the present method, by utilizing the autocorrelation functions of Daubechies wavelets and designing modified boundary basis functions, not only maintains computational efficiency but also provides higher accuracy (up to 10^{-23}) in solving the Burgers equation. Furthermore, the guarantee of uniform convergence in Theorem 3.2 establishes a stronger theoretical foundation for the proposed method.

Uncorrected Proof

Declarations

Ethics, Consent to Participate, and Consent to Publish declarations: Not applicable.

Computing Interests: The authors declare that they have no Computing Interests regarding the publication of this article.

Availability of Data and Materials: Not applicable

Author Contributions: All authors read and approved the final manuscript.

Acknowledgments: The authors are indebted to the referees for their valuable comments and suggestions.

Conflicts of Interest: The authors declare that there are no conflicts of interest and grants regarding the publication of this paper. **Conflict of interest statement**.- The authors have no competing interests

References

[1] A. Alipanah, K. Arzideh, M. Firouzi, and A. Kasnazan, *Daubechies wavelet scaling function approach to solve Volterra's population model*, Int. J. Math. Sci., *2022* (2022), 1–8.

[2] A. Alipanah and M. Zafari, *Collocation method using auto-correlation functions of compact supported wavelets for solving Volterra's population model*, Chaos, Solitons and Fractals., *175* (2023), 113941.

[3] S. Bertoluzza and G. Naldi, *A wavelet collocation method for the numerical solution of partial differential equations*, Appl. Comput. Harmon. Anal., *3* (1996), 1–9.

[4] S. Bertoluzza and V. Perrier, *A new construction of boundary interpolating wavelets for fourth order problems*, Acta Application Mathematicae, *152* (2007), 33–56.

[5] G. Beylkin, *On the representation of operators in bases of compactly supported wavelets*, SIAM, J. Numer. Anal., *6* (1992), 1716–1740.

[6] M. P. Bonkile, A. Awasthi, C. Lakshmi, V. Mukundan, and V. S. Aswin, *A systematic literature review of Burgers' equation with recent advances*, Pramana, *90* (2018), 69–90.

[7] J. Caldwell, P. Wanless, and A.E. Cook, *Solution of Burgers equation for large Reynolds number using finite elements with moving nodes*, Appl. Math. Model., *11* (1987), 211–214.

[8] A. Cohen, I. Daubechies, and J. C. Feauveau, *Biorthogonal bases of compactly supported wavelets*, Commun. Pur. Appl. Math., *45* (1992), 485–560.

[9] I. Daubechies, *Ten Lectures on Wavelets*, SIAM, Philadelphia, PA, 1992.

[10] I. Daubechies, *Orthonormal bases of compactly supported wavelets*, Commun. Pure Appl. Math., *41* (1988), 909–996.

[11] G. Deslauriers and S. Dubuc, *Symmetric iterative interpolation processes*, Constr. Approx., *5* (1989), 49–68.

[12] S. Dubuc, *Interpolation through an iterative scheme*, J. Math. Anal. Appl., *114* (1986), 185–204.

[13] A. Garba, *A wavelet collocation method for numerical solution of Burger's equation*, (Tech.Rep.) International Center for Theorytical Physics, 1996.

[14] K. Goyal and M. Mehra, *A fast adaptive diffusion wavelet method for Burger's equation*, Comput. Math. Appl., *68* (2014), 568–577.

[15] R. Jiwari, R. C Mittal, and K. K. Sharma, *A numerical scheme based on weighted average differential quadrature method for the numerical solution of Burgers' equation*, Appl. Math. Comput., *219* (2013), 6680–6691.

[16] A. Kasnazani and A. AliPanah, *Solving brachistochrone problem via scaling functions of Daubechies wavelets*, Comput. Methods Differ. Equ., *9*(2) (2021), 511–522

[17] S. Kutluay, A. Esen, and I. Dag, *Numerical solution of the Burgers' equation by the least- squares quadratic B-spline finite element method*, J. Comput. Appl. Math., *167* (2004), 21–33.

[18] D. Lal and M. Yadav, *Approximate analytical solution of one dimensional nonlinear Burger's equation using Homotopy Perturbation method*, J. Algebr. Stat., *3* (2022), 5462–5469.

[19] A. Latto, H. L. Resnikoff, and E. Tenenbaum, *The evaluation of connection coefficients of compactly supported wavelets*, Proceedings of the French-USA Workshop on Wavelets and Turbulence, Springer-Verlag, New York, 1991.

[20] K. Pandey, L. Varma, and A. K. Varma, *On a finite difference scheme for Burgers' equation*, Appl. Math. Comput., *215* (2009), 2206–2214.

[21] N. Saito and G. Beylkin, *Multiresolution representations using the auto-correlation functions of compactly supported wavelets,* Schlumberger-Doll Research, Ridgefield, CT, Tech. Rep, Aug. 1991. Expanded abstract in Proc. ICASSP-92, *4* (1992), 381–384.

[22] M. Seydaoglu, U. Erdogan, and T. Ozis, *Numerical solution of Burgers' equation with high order splitting methods*, J. Comput. Appl. Math., *291* (2016), 410–421.

Uncorrected Proof

[23] O. V. Vasilyev and S. Paolucci, *A fast adaptive wavelet collocation algorithm for multidimensional PDEs*, J. Comput. Phys., *138* (1997), 16–56.
[24] O. V. Vasilyev and S. Paolucci, *A multilevel wavelet collocation method for solving partial differential equations in a finite domain*, J. Comput. Phys., *120* (1996), 33–47.
[25] D. C. Wan and G. W. Wei, *The study of quasi wavelets based numerical method applied to Burger's equation*, Appl. Math. Mech., *21* (2000), 1099–1110.
[26] G. W. Wei, *Quasi wavelets and quasi interpolating wavelets*, Chem. Phys. Lett., *296* (1998), 215–222.
[27] H. Zhu, L. Tang, S. Song, Y. Tang, and D. Wang, *Symplectic wavelet collocation method for Hamiltonian wave equations*, J. Comput. Phys., *229* (2010), 2550–2572.

Uncorrected Proof