J Math Chem (2015) 53:1592–1607
DOI 10.1007/s10910-015-0507-5

ORIGINAL PAPER

A Haar wavelet-finite difference hybrid method for the numerical solution of the modified Burgers' equation

Ö. Oruç[1] · F. Bulut[2] · A. Esen[1]

Received: 19 November 2014 / Accepted: 9 April 2015 / Published online: 17 April 2015

Abstract In this paper, we investigate the numerical solutions of one dimensional modified Burgers' equation with the help of Haar wavelet method. In the solution process, the time derivative is discretized by finite difference, the nonlinear term is linearized by a linearization technique and the spatial discretization is made by Haar wavelets. The proposed method has been tested by three test problems. The obtained numerical results are compared with the exact ones and those already exist in the literature. Also, the calculated numerical solutions are drawn graphically. Computer simulations show that the presented method is computationally cheap, fast, reliable and quite good even in the case of small number of grid points.

Keywords Haar wavelet method · Modified Burgers' equation · Linearization · Finite differences · Numerical solution

1 Introduction

Burgers' equation [1,2] attracted much attention in the past few years, a variety of numerical solutions established for the solution of the problem. Different forms of the Burgers' equation can be used in various scientific areas such as, plasma physics, solid state physics, optical fibers, biology, fluid dynamics, chemical kinetics etc. The purpose of this paper is to get numerical solutions with the aid of Haar wavelet method for nonlinear partial differential equations of the following forms

$$u_t + u^n u_x = \nu u_{xx}, \quad a \le x \le b, \quad t \ge 0 \tag{1}$$

✉ F. Bulut
fatih.bulut@inonu.edu.tr

1 Department of Mathematics, İnonu University, Malatya, Turkey

2 Department of Physics, İnonu University, Malatya, Turkey

and

$$u_t + (V + u)\, u_x = \nu u_{xx}, \quad -\infty \le x \le +\infty, \quad t > 0 \tag{2}$$

with the initial condition

$$u(x, 0) = g(x), \quad x \in [a, b]$$

and the boundary conditions

$$u(a, t) = f_1(t), \qquad u(b, t) = f_2(t), \quad t \in [0, T]$$

where $u(x, t)$ represents the velocity for spatial dimension x and time t, ν is a positive constant showing the kinematic viscosity of the fluid, V is a positive constant and n is a positive parameter. For $n = 1$, Eq. (1) corresponds to the well known Burgers' equation which was first introduced by Bateman [1]. Later, Burgers [2] applied this equation to the mathematical model for the turbulence, due to his extensive works on this model, it is called Burgers' equation. Burgers' equation was solved by both Hopf [3] and Cole [4] analytically and independently for arbitrary initial conditions. Since these solutions involve infinite series, they may converge very slowly for small values of viscosity coefficients. Also, in many cases these solutions will fail for $\nu < 0.01$, which corresponds to shock waves. Because of the nonlinear convection term and the occurrence of the viscosity term, Burgers' equation shows similar features with Navier-Stokes equation. Hence, Burgers' equation has been used as a model for the Navier-Stokes equation. Because of the complexity in obtaining the analytical solutions many researchers have used numerical methods [5–8].

For $n \ge 2$, the Eq. (1) is called modified Burgers' equation (MBE). The cases to be studied in which $n = 2$ and $n = 3$ are taken will be denoted hereafter as MBE2 and MBE3, respectively. The Eq. (2) will be denoted hereafter as MBE4 and it was proposed in [9] to judge whether the numerical method has the ability to resolve the large gradient regions. The MBE equation has been solved by several researchers both analytically and numerically. Ramadan and El-Danaf [10] applied collocation method with quintic splines, Ramadan et al. [11] used the collocation method with septic splines, Saka and Dag [12] applied time and space splitting techniques and then employed the quintic B-spline collocation method, Roshan and Bhamra [13] used the Petrov–Galerkin method, Irk [14] employed Crank-Nicolson central differencing scheme for the time and B-spline functions for the space, Brastos [15–17] used various finite difference based methods, Temsah [18] applied El-Gendi method, Grienwank and El-Danaf [19] used a non-polynomial spline based method. Duan et al. [20] used a lattice Boltzmann model. More recently, Zhang et al. [21] have solved the equation by the local discontinuous Galerkin method.

The wavelet methods have become matter of attention lately in solving differential equations numerically, they were first applied for solving differential equations in early 1990s. Chen and Hsiao [22] introduced a method to solve ordinary differential equations based on the Haar wavelets. They recommended to expand the highest derivative of the function appearing in the differential equation into Haar series. Recently, many authors have used Haar wavelet method for solving ordinary and partial differential equations. Lepik applied this method to several problems, solved nonlinear ordinary differential equations and diffusion equation in [23], Burgers' and Sine-Gordon equations in [24] and Poisson equation in [25]. Celik [26] solved Burger–Huxley equation and then solved magnetohydrodynamic flow equation in [27]. Jiwari [28] solved Burgers' equation by a Haar wavelet quasilinearization. Kaur et al. [29] have used Haar wavelet method to solve Lane–Emden equations arising in astrophysics. Zhi Shi et al. [30] applied Haar wavelet method to solve 2D and 3D Poisson equations and biharmonic equations. In this study, we investigate the numerical solution of the MBE using Haar wavelet method.

This paper is organized as follows. In Sect. 2, preliminaries about the Haar wavelets are given. In Sect. 3, we show how to use the Haar wavelet method for MBE. The numerical results obtained by the proposed method for three test problems, tabulated and depicted graphically, in Sect. 4. Finally we conclude the paper in Sect. 5.

2 Haar wavelets

Wavelet analysis is a very useful technique to its solve mathematical problems. Wavelets, especially Haar wavelet family is favored by researchers because of its simplicity. The Haar wavelet family for $x \in [0, 1]$ is defined as follows:

$$h_i(x) = \begin{cases} 1, & \text{for } x \in [\zeta_1, \zeta_2) \\ -1, & \text{for } x \in [\zeta_2, \zeta_3] \\ 0, & \text{elsewhere} \end{cases} \tag{3}$$

where

$$\zeta_1 = \frac{k}{m}, \qquad \zeta_2 = \frac{k+0.5}{m}, \qquad \zeta_3 = \frac{k+1}{m}.$$

Here m and k have integer values as $m = 2^j$, $j = 0, 1, \dots, J$ and J show the resolution of the wavelet and $k = 0, 1, \dots, m-1$ is the translation parameter. The index of h_i in Eq. (3) is calculated by $i = m + k + 1$. For the minimum values of $m = 1, k = 0$ we have $i = 2$; the maximum value of i will be $i = 2M = 2^{J+1}$; where J is the maximum resolution of the wavelet. We also have $i = 1$ corresponding to the scaling function of Haar wavelet family i.e. $h_1(x) = 1$ in [0, 1].

To solve partial differential equations of any order with Haar wavelet Method, we need the following integrals

$$p_{i,1}(x) = \int_0^x h_i(x)dx$$
$$p_{i,n+1}(x) = \int_0^x p_{i,n}(x)dx, \quad n = 1, 2, 3, \dots$$

These integrals can be calculated analytically with the help of Eq. (3); by doing so we get the following equations

$$p_{i,1}(x) = \begin{cases} x - \zeta_1, & \text{for } x \in [\zeta_1, \zeta_2) \\ \zeta_3 - x, & \text{for } x \in [\zeta_2, \zeta_3] \\ 0, & \text{elsewhere} \end{cases} \tag{4}$$

$$p_{i,2}(x) = \begin{cases} \frac{(x-\zeta_1)^2}{2}, & \text{for } x \in [\zeta_1, \zeta_2) \\ \frac{1}{4m^2} - \frac{(\zeta_3-x)^2}{2}, & \text{for } x \in [\zeta_2, \zeta_3) \\ \frac{1}{4m^2}, & \text{for } x \in [\zeta_3, 1] \\ 0, & \text{elsewhere} \end{cases} \tag{5}$$

Because of the orthogonality of Haar wavelets; i.e.

$$\int_0^1 h_i(x)h_l(x)dx = \begin{cases} 2^{-j}, & \text{for } i = l = 2^j + k \\ 0, & \text{for } i \neq l \end{cases}$$

they are very good for constructing transform basis. Any square integrable function $y(x)$ in the interval [0, 1] can be expressed as

$$y(x) = \sum_{i=1}^{\infty} c_i h_i(x), \qquad i = 2^j + k,\ j \geq 0,\ 0 \leq k < 2^j,$$

where the coefficients c_i are determined by [22,31]

$$c_i = 2^j \int_0^1 y(x)h_i(x)dx.,$$

Even though the series expansion of $y(x)$ involves infinite terms, if $y(x)$ is a piecewise constant or it may be approximated as a piecewise constant for each sub-interval, in that case $y(x)$ can be terminated at finite terms. That means $y(x)$ can be expressed as follows

$$y(x) = \sum_{i=1}^{2M} c_i h_i(x) = c_{(2M)}^T h_{(2M)}(x),$$

where the coefficients $c_{(2M)}^T$ and $h_{(2M)}(x)$ are defined as

$$c_{(2M)}^T = [c_1, c_2, \ldots, c_{(2M)}]$$
$$h_{(2M)}(x) = [h_1(x), h_2(x), \ldots, h_{(2M)}(x)]^T$$

here T denotes transpose and $M = 2^j$.

3 Application of the Haar wavelet method to the MBE

In this section, we present the application of Haar wavelet method to get numerical solutions of MBE equations.

3.1 MBE2 Equation

$$u_t + u^2 u_x = \nu u_{xx}$$

Firstly, to discretize this equation, we substitute u_t by forward finite difference and also use time average for $u^2 u_x$ and u_{xx} terms, we get the following relation

$$\frac{u_{j+1} - u_j}{\Delta t} + \frac{\left(u^2 u_x\right)_{j+1} + \left(u^2 u_x\right)_j}{2} = \nu \frac{(u_{xx})_{j+1} + (u_{xx})_j}{2}.$$

If we use the linearization $2u_{j+1}u_j\,(u_x)_j + u_j u_j\,(u_x)_{j+1} - 2u_j u_j\,(u_x)_j$ which is similar to the one used in Rubin-Graves [32] instead of nonlinear term $u^2 u_x$ and simplify the above equation, we obtain

$$\begin{aligned} u_{j+1} &+ \frac{\Delta t}{2}\left(2u_{j+1}u_j\,(u_x)_j + u_j u_j\,(u_x)_{j+1}\right) - \nu\frac{\Delta t}{2}\,(u_{xx})_{j+1} \\ &= u_j - \frac{\Delta t}{2}\left(u^2 u_x\right)_j + \nu\frac{\Delta t}{2}\,(u_{xx})_j + \Delta t\left(u_j u_j\,(u_x)_j\right) \end{aligned} \tag{6}$$

with the initial condition

$$u_0 = g(x) \tag{7}$$

and boundary conditions

$$u_{j+1}(a) = f_1(t_{j+1}),\ u_{j+1}(b) = f_2(t_{j+1}), \quad j = 0, 1, \ldots, N-1 \tag{8}$$

where u_{j+1} is the solution of Eq. (6) at the $j-$th time step.

3.2 MBE3 Equation

Secondly, we will consider following equation

$$u_t + u^3 u_x = \nu u_{xx}.$$

For discretizing this equation, we substitute u_t by forward finite difference and also use time average for u_{xx} term, so we get the following relation

$$\frac{u_{j+1} - u_j}{\Delta t} + \left(u^3 u_x\right)_j = \nu \frac{(u_{xx})_{j+1} + (u_{xx})_j}{2}$$

simplifying the above relation, we obtain

$$2u_{j+1} - \nu\Delta t\,(u_{xx})_{j+1} = 2u_j - 2\Delta t\left(u^3 u_x\right)_j + \nu\Delta t\,(u_{xx})_j \tag{9}$$

with the initial condition

$$u_0 = g(x)$$

and boundary conditions

$$u_{j+1}(a) = f_1(t_{j+1}),\; u_{j+1}(b) = f_2(t_{j+1}), \quad j = 0, 1, \ldots, N-1$$

where u_{j+1} is the solution of Eq. (9) at the $j-$th time step.

3.3 MBE4 Equation

Lastly, we will consider the following equation

$$u_t + (V + u)\,u_x = \nu u_{xx}\,.$$

We firstly rewrite the MBE4 as follows

$$u_t + uu_x = \nu u_{xx} - Vu_x$$

then using forward finite difference for u_t and time average for uu_x and u_{xx} terms, we get the following relation

$$\frac{u_{j+1} - u_j}{\Delta t} + \frac{(uu_x)_{j+1} + (uu_x)_j}{2} = \nu\frac{(u_{xx})_{j+1} + (u_{xx})_j}{2} - V(u_x)_j.$$

By using $u_{j+1}(u_x)_j + u_j(u_x)_{j+1} - u_j(u_x)_j$ linearization technique in [32] for the term uu_x the above equation can be put in the following form

$$\begin{aligned} &2u_{j+1} + \Delta t\left(u_{j+1}(u_x)_j + u_j(u_x)_{j+1}\right) - \nu\Delta t\,(u_{xx})_{j+1} \\ &\quad = 2u_j - 2\Delta t V\,(u_x)_j + \nu\Delta t\,(u_{xx})_j \end{aligned} \tag{10}$$

with the initial condition

$$u_0 = g(x)$$

and boundary conditions

$$u_{j+1}(a) = f_1(t_{j+1}),\; u_{j+1}(b) = f_2(t_{j+1}), \quad j = 0, 1, \ldots, N-1$$

where u_{j+1} is the solution of Eq. (10) at the $j-$th time step.

3.4 Haar wavelet method for spatial-discretization

In a similar manner of the Lepik [23–25], we assume that $(u_{xx})_{j+1}$ can be expanded in terms of Haar wavelets as

$$(u_{xx})_{j+1}(x) = \sum_{i=1}^{2M} c_i h_i(x) = c^T h_{2M}(x) \tag{11}$$

where c^T is a row vector. Integrating Eq. (11) with respect to x from 0 to x, we get the following equation

$$(u_x)_{j+1}(x) = (u_x)_{j+1}(0) + \sum_{i=1}^{2M} c_i p_{i,1}(x). \tag{12}$$

In Eq. (12), $(u_x)_{j+1}(0)$ is unknown so to find it, we need to integrate Eq. (12) from 0 to 1. After that, using boundary conditions (8) we get

$$(u_x)_{j+1}(0) = f_2(t_{j+1}) - f_1(t_{j+1}) - \sum_{i=1}^{2M} c_i p_{i,2}(1). \tag{13}$$

Substituting (13) into Eq. (12) results in the following equation

$$(u_x)_{j+1}(x) = \sum_{i=1}^{2M} c_i p_{i,1}(x) + f_2(t_{j+1}) - f_1(t_{j+1}) - \sum_{i=1}^{2M} c_i p_{i,2}(1). \tag{14}$$

Now, if we integrate again Eq. (14) from 0 to x, we get

$$u_{j+1}(x) = f_1(t_{j+1}) + x\left(f_2(t_{j+1}) - f_1(t_{j+1})\right) + \sum_{i=1}^{2M} c_i p_{i,2}(x) - x\sum_{i=1}^{2M} c_i p_{i,2}(1) \tag{15}$$

where

$$p_{i,2}(1) = \begin{cases} 0.5 & \text{if } i = 1 \\ \frac{1}{4m^2} & \text{if } i > 1 \end{cases}$$

is obtained from Eq.(5).

3.5 Haar wavelet method for time-discretization

Substituting (15), (14) and (11) into the Eq. (6) and discretisizing the results at the collocation points $x_l = \frac{l-0.5}{2M}$, $l = 1, 2, \ldots, 2M$ we get following equation for MBE2:

$$
\begin{aligned}
& f_1(t_{j+1}) + x_l \left(f_2(t_{j+1}) - f_1(t_{j+1})\right) + \sum_{i=1}^{2M} c_i p_{i,2}(x_l) - x_l \sum_{i=1}^{2M} c_i p_{i,2}(1) \\
& \quad + \frac{\Delta t}{2}\Bigg(2\left[f_1(t_{j+1}) + x_l \left(f_2(t_{j+1}) - f_1(t_{j+1})\right)\right. \\
& \qquad \left. + \sum_{i=1}^{2M} c_i p_{i,2}(x_l) - x_l \sum_{i=1}^{2M} c_i p_{i,2}(1)\right] u_j \left(u_x\right)_j\Bigg) \\
& + \frac{\Delta t}{2}\left(u_j u_j \left[\sum_{i=1}^{2M} c_i p_{i,1}(x_l) + f_2(t_{j+1}) - f_1(t_{j+1}) - \sum_{i=1}^{2M} c_i p_{i,2}(1)\right]\right) \\
& - \nu \frac{\Delta t}{2}\left(\sum_{i=1}^{2M} c_i h_i(x_l)\right) = u_j - \frac{\Delta t}{2}\left(u^2 u_x\right)_j + \nu \frac{\Delta t}{2}\left(u_{xx}\right)_j + \Delta t \left(u_j u_j \left(u_x\right)_j\right).
\end{aligned}
$$

This system of algebraic equations is solved by using an appropriate software and the wavelet coefficients c^T are found, and then these are used in (15), (14) and (11) to find the new values of u, u_x and u_{xx} at each time level. We continue the iteration process in this way until the desired time level is reached. For this purpose, the following initial conditions are needed

$$
\begin{aligned}
u_0(x_l) &= g(x_l) \\
\left(u_x\right)_0(x_l) &= g'(x_l) \\
\left(u_{xx}\right)_0(x_l) &= g''(x_l).
\end{aligned}
$$

In a similar way to MBE2, if we substitute (15), (14) and (11) into each of the Eqs. (9), (10) and discretize the results at the collocation points we get system of equations for MBE3 and MBE4 respectively. Then by solving the newly obtained systems, the wavelet coefficients c^T are obtained. Now we can substitute these coefficients into each of the Eqs. (15), (9) and (10) calculate approximate solutions successively.

4 Numerical results

Numerical computations have been done with the free software package GNU Octave and graphical outputs were generated by Matplotlib package [33]. In order to show the performance of the suggested method as compared with the exact solution, we considered the error norms L_2 and L_∞ defined by

$$
\begin{aligned}
L_2 &= \sqrt{\Delta x \sum\nolimits_{i=1}^{2M} \left|u_i^{\text{exact}} - u_i^{\text{num}}\right|^2} \\
L_\infty &= \max_i \left|u_i^{\text{exact}} - u_i^{\text{num}}\right|.
\end{aligned}
$$

Table 1 Comparison of the error norms for obtained results and other researchers' results for $\Delta t = 0.01$ and $\nu = 0.01$

	Δx	$t = 2$		$t = 6$		$t = 10$	
		$L_2 \times 10^3$	$L_\infty \times 10^3$	$L_2 \times 10^3$	$L_\infty \times 10^3$	$L_2 \times 10^3$	$L_\infty \times 10^3$
Haar wavelet	1/16	0.34748	0.75978	0.32246	0.46335	0.54160	1.16480
QBCM [10]	1/200	0.52308	1.21698	0.49023	0.72249	0.64007	1.28124
SBCM [11]	1/50	0.79043	1.70309	0.51672	0.76105	0.80026	1.80239
QBCA1 [12]	1/200	0.37932	0.81680	0.32602	0.52579	0.54701	1.28125
QBCA2 [12]	1/200	0.37951	0.82212	0.32427	0.52579	0.54354	1.28125
SBCM1 [14]	1/200	0.38489	0.82934	–	–	0.54826	1.28127
SBCM2 [14]	1/200	0.39078	0.82734	–	–	0.54612	1.28127

4.1 MBE2 Equation

Consider MBE2 which has the following analytic solution [11]:

$$u(x,t) = \frac{x/t}{1 + \left(\sqrt{t}/c_0\right)\exp(x^2/4\nu t)}, \qquad t \geq 1, \qquad 0 \leq x \leq 1.$$

where $0 < c_0 < 1$. We take the following boundary conditions

$$u(0,t) = u(1,t) = 0$$

and extract initial condition from analytic solution for $t = 1$ as follows

$$u(x,1) = \frac{x}{1 + (1/c_0)\exp(x^2/4\nu)}, \qquad 0 \leq x \leq 1.$$

We have compared our results with the other results existing in the literature we take $c_0 = 0.5$ and tested the proposed method for various values of $\nu = 0.01, 0.005, 0.001$ at $t = 2, 6, 10$ using different time steps $\Delta t = 0.01, 0.001$. In Table 1, we show a comparison of the values of error norms obtained by the present method for $\Delta x = 1/16$, $\Delta t = 0.01$ and $\nu = 0.01$ with the other studies for the values of $\Delta x = 1/200$. It is clearly seen from the table that even though Δx is used as $1/16$, error norms obtained from the results for each time are smaller than those given in Ramadan and Danaf [10], Saka and Dag [12] using Quintic B-Spline collocation method, Ramadan et al. [11] and Irk [14] using sextic B-spline collocation method. In Fig. 1, we depict the numerical results and error graphics for $2M = 64$. It can be easily seen from Fig. 1b that maximum error is getting larger at the right boundary when t becomes approximately greater than 3, which means the chosen interval [0, 1] is not suitable for this problem, this can also be understood from the analytic solution. To overcome this problem, we extended the interval to [0, 1.3] and gave the results on Table 2 and compared with [14].

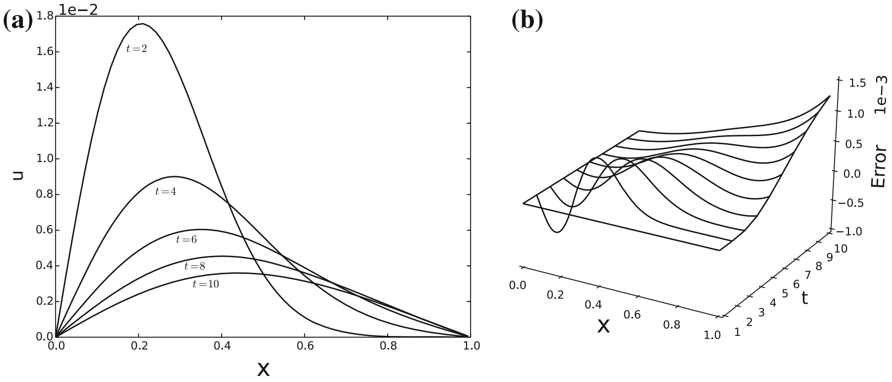

Fig. 1 Numerical solutions and errors for $\Delta x = 1/64$, $\Delta t = 0.01$ and $\nu = 0.01$. **a** Numerical solutions for different times. **b** Errors for different times

Table 2 Comparison of the error norms for obtained results and Ref. [14] for $\Delta t = 0.01$ and $\nu = 0.01$ for $0 \leq x \leq 1.3$

	Δx	$t = 2$		$t = 6$		$t = 10$	
		$L_2 \times 10^3$	$L_\infty \times 10^3$	$L_2 \times 10^3$	$L_\infty \times 10^3$	$L_2 \times 10^3$	$L_\infty \times 10^3$
Haar Wavelet	1/16	0.29252	0.72890	0.24311	0.45606	0.22321	0.32374
SBCM1 [14]	1/260	0.38489	0.82934	–	–	0.25586	0.32723
SBCM2 [14]	1/260	0.39078	0.82734	–	–	0.25259	0.32337

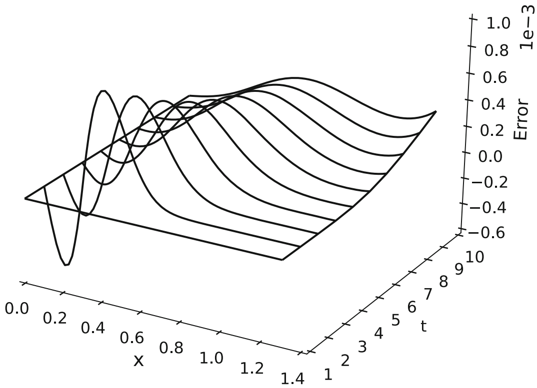

Fig. 2 Errors for $\Delta x = 1/64$, $\Delta t = 0.01$ and $\nu = 0.01$ for $0 \leq x \leq 1.3$

In Fig. 2, we give the errors for the extended interval [0, 1.3], as it can be seen from the figure the errors are smaller than for the interval [0, 1] on the right boundary. In Table 3, we show a comparison of the values of error norms obtained by the present method for $\Delta x = 1/16$ with the other studies for the values of Δx=1/200 for the same values of $\Delta t = 0.01$ and $\nu = 0.005$. It is clearly seen from the table that the

Table 3 Comparison of the error norms for obtained results and other researchers' results for $\Delta t = 0.001$ and $\nu = 0.005$

	Δx	$t = 2$		$t = 6$		$t = 10$	
		$L_2 \times 10^3$	$L_\infty \times 10^3$	$L_2 \times 10^3$	$L_\infty \times 10^3$	$L_2 \times 10^3$	$L_\infty \times 10^3$
Haar wavelet	1/16	0.19508	0.54059	0.16489	0.32340	0.14055	0.22598
QBCM [10]	1/200	0.25786	0.72264	0.22569	0.43082	0.18735	0.30006
SBCM1 [14]	1/200	0.22890	0.58623	–	–	0.14042	0.23019
SBCM2 [14]	1/200	0.23397	0.58424	–	–	0.13747	0.22626

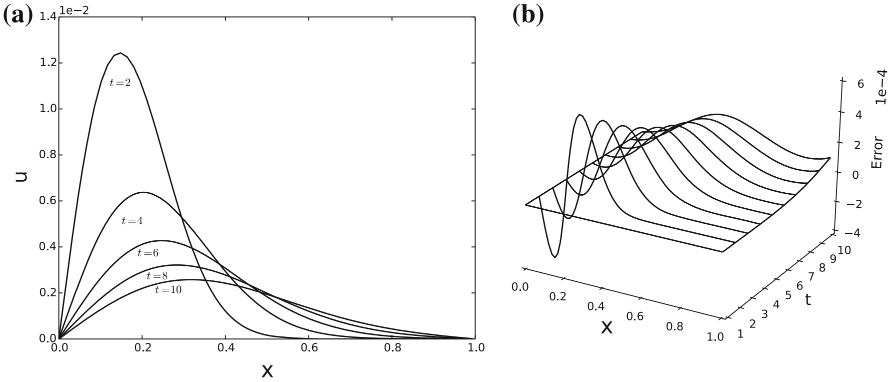

Fig. 3 Numerical solutions and errors for $\Delta x = 1/64$, $\Delta t = 0.001$ and $\nu = 0.005$. **a** Numerical Solutions for different times. **b** Errors for different times

error norms L_2 and L_∞ obtained for each time are smaller than those given in Refs. [10,14]. Again in Fig. 3, we present the the numerical results and error graphics for $2M = 64$. As it is seen from the figure, with the incrementation of time the errors are getting smaller.

In Table 4, we show a comparison of the values of error norms obtained by the present method for $\Delta x = 1/16$, $\Delta t = 0.01$ and $\nu = 0.001$ with the other studies for the values of Δx=1/200. As it is seen from the table, the results obtained from the present method are better than those obtained in Refs. [10,11] while as good as those obtained in Refs. [12,14]. Again in Fig. 4, we present both the numerical results and error graphics for $2M = 64$, respectively.

4.2 MBE3 Equation

We consider MBE3 with the boundary conditions $u(0, t) = u(\pi, t) = 0$ and the initial condition

$$u(x, 0) = A \sin x$$

where $A = 1$.

Table 4 Comparison of the error norms for obtained results and other researchers' results for $\Delta t = 0.01$ and $\nu = 0.001$

	Δx	$t = 2$		$t = 6$		$t = 10$	
		$L_2 \times 10^3$	$L_\infty \times 10^3$	$L_2 \times 10^3$	$L_\infty \times 10^3$	$L_2 \times 10^3$	$L_\infty \times 10^3$
Haar wavelet	1/16	0.11093	0.43656	0.06151	0.16468	0.04816	0.11807
QBCM [10]	1/200	0.06703	0.27967	0.06046	0.17176	0.05010	0.12129
SBCM [11]	1/50	0.18355	0.81862	0.08142	0.21348	0.05512	0.13943
QBCA1 [12]	1/200	0.06811	0.26094	0.04942	0.14810	0.04067	0.10264
QBCA2 [12]	1/200	0.06953	0.27283	0.04917	0.15656	0.04000	0.10835
SBCM1 [14]	1/200	0.06843	0.26233	–	–	0.04080	0.10295
SBCM2 [14]	1/200	0.07220	0.25975	–	–	0.03871	0.09882

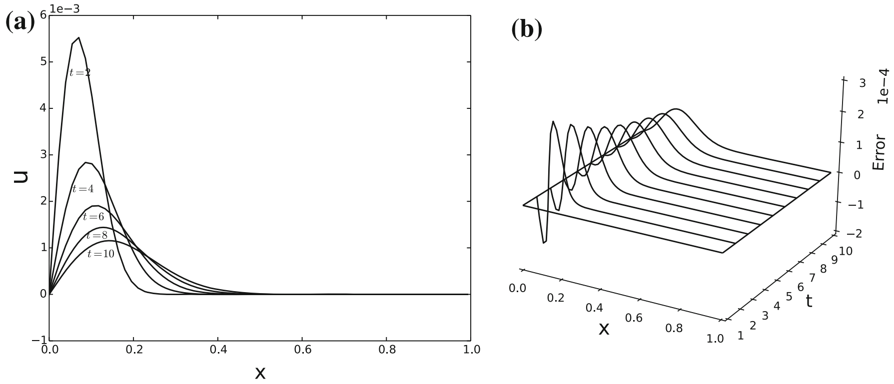

Fig. 4 Numerical solutions and errors for $\Delta x = 1/64$, $\Delta t = 0.01$ and $\nu = 0.001$. **a** Numerical Solution. **b** Errors for different times

To compare our results with the other results existing in the literature, we take $\Delta t = 0.01$, $\nu = 0.005$ and $J = 6$ for different values values of Δx and t. The results are tabulated in Table 5 in which the analytic form of the asymptotic solutions constructed by Sachdev et al. [34] who constructed large-time asymptotic solution of the modified Burgers' equation with sinusoidal initial conditions by using a balancing argument. It is clearly seen from the table that our results are in good agreement with those available in the literature. Among others, first of all, for $t > 100$ the presently proposed method yields more accurate results than the corresponding ones given by Bratsos [16] who proposed a finite-difference scheme based on fourth-order rational approximants to the matrix-exponential term in a two-time level recurrence relation. Secondly, for values of t greater than 200, the error norms L_∞ computed in the present method are smaller than those given by Duan et. al. [20] who developed a special lattice Boltzmann model to solve the modified Burgers' equation. Lastly, the error norms L_2 for values of t greater than 250 are also smaller than those given by Duan et. al. in the same study [20].

Table 5 Comparison of L_2 and L_∞ error norms at different times

	Δx	t	100	150	200	250
Haar wavelet	1/128	$L_2 \times 10^2$	3.3032	0.60539	0.20656	0.076872
		$L_\infty \times 10^2$	3.4523	0.67128	0.18567	0.064929
[16]	1/1000	$L_2 \times 10^2$	3.2761	0.61258	0.22273	0.091238
		$L_\infty \times 10^2$	3.3976	0.68400	0.20416	0.083351
[20]	1/100	$L_2 \times 10^2$	–	0.3227	0.09912	0.05031
		$L_\infty \times 10^2$	–	0.5172	0.1671	0.1400
	Δx	t	300	350	400	450
Haar wavelet	1/128	$L_2 \times 10^2$	0.028506	0.010561	0.0039581	0.0015662
		$L_\infty \times 10^2$	0.024053	0.0090748	0.0035684	0.0015175
[16]	1/1000	$L_2 \times 10^2$	0.041341	0.023070	0.016168	0.012836
		$L_\infty \times 10^2$	0.039559	0.021860	0.014160	0.010361
[20]	1/100	$L_2 \times 10^2$	0.05939	0.06940	0.07567	0.07990
		$L_\infty \times 10^2$	0.1452	0.1488	0.1513	0.1531

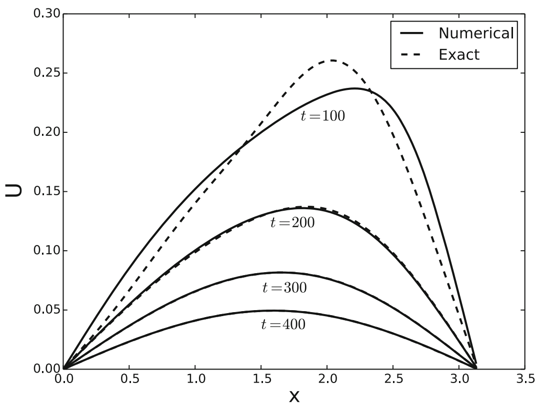

Fig. 5 Numerical solutions of MBE3 with $\Delta t = 0.01$, $\nu = 0.005$ at $t = 100,\ 200,\ 300,\ 400$

It can be easily seen from Fig. 5 that for $t > 100$ the numerical solution and the exact solution are in good agreement. For $t > 200$ the numerical solution and the exact solution are not distinguishable.

4.3 MBE4 Equation

From the studies of Vasilyev and Paolucci [9] and Basdevant et al. [35] the efficiency of a numerical method can be judged from its ability to resolve the large gradient regions that occur in the solution. So to test the present method we solve MBE4 which combined with the initial condition

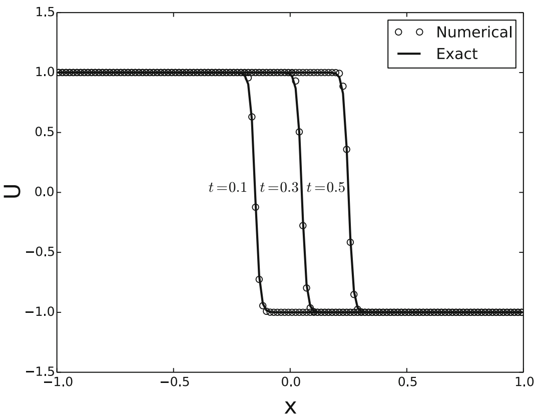

Fig. 6 Numerical solutions of MBE4 with $V = 1$, $\nu = 0.01$, $\Delta t = 0.001$ and $x_0 = -0.25$ at $t = 0.1,\ 0.3,\ 0.5$

$$u(x, 0) = -\tanh\left(\frac{x - x_0}{2\nu}\right)$$

and time dependent boundary conditions

$$u(-1, t) = -\tanh\left(\frac{-1 - x_0 - Vt}{2\nu}\right), \quad u(1, t) = -\tanh\left(\frac{1 - x_0 - Vt}{2\nu}\right)$$

Note that the original problem is given on domain $-\infty < x < \infty$ but for numerical computations we take the domain as $-1 \le x \le 1$. The exact solution of the MBE4 is as follows:

$$u(x, t) = -\tanh\left(\frac{x - x_0 - Vt}{2\nu}\right)$$

For $V = 1$, $\nu = 0.01$, $x_0 = -0.25$ and $J = 6$ the numerical solutions are given in Fig. 6.

It can be seen from Fig. 6 that the numerical solution and the exact solution are in good agreement with each other and also by increasing wavelet resolution J we can get more accurate results.

5 Conclusion

In conclusion, in this paper, Haar wavelet method is used to get numerical solutions of modified Burgers' equation. The obtained solutions are compared with the exact ones and with those available in the literature found by other researchers. The comparison shows that the present method is competitive with other methods. The results of the present method are better in spite of using less collocation points and a simpler scheme when compared to others. The proposed method can safely and quickly be used for the solution of a wide range of similar problems.

Acknowledgments We would like to thank the reviewers for their invaluable suggestions towards the improvement of the paper.

References

1. H. Bateman, Some recent researches on the motion of fluids. Mon. Weather Rev. **43**, 163–170 (1915)
2. J.M. Burgers, A mathematical model illustrating the theory of turbulence. Adv. Appl. Mech. **1**, 171–199 (1948)
3. E. Hopf, The partial differential equation $u_t + uu_x = \mu u_{xx}$. Comm. Pure Appl. Math. **3**, 201–230 (1950)
4. J.D. Cole, On a quasilinear parabolic equation occurring in aerodynamics. Quart. Appl. Math. **9**, 225–236 (1951)
5. E.L. Miller, Predictor–corrector studies of Burgers' model of turbulent flow, M.S. Thesis, (University of Delaware, Newark, DE, 1966)
6. R.C. Mittal, P. Singhal, Numerical solution of Burgers' equation. Commun. Numer. Methods Eng. **9**, 397–406 (1993)
7. S. Kutluay, A. Esen, I. Dag, Numerical solutions of the Burgers' equation by the least-squares quadratic B-spline finite element method. J. Comput. Appl. Math. **167**, 21–33 (2004)
8. S. Kutluay, A. Esen, A lumped Galerkin method for solving the Burgers' equation. Int. J. Comput. Math. **81**(11), 1433–1444 (2004)
9. O.V. Vasilyev, S. Paolucci, A dynamically adaptive multilevel wavelet collocation method for solving partial differential equations in a finite domain. J. Comput. Phys. **125**, 498–512 (1996)
10. M.A. Ramadan, T.S. El-Danaf, Numerical treatment for the modified Burgers' equation. Math. Comput. Simul. **70**, 90–98 (2005)
11. M.A. Ramadan, T.S. El-Danaf, F.E.I. Abd Alaal, A numerical solution of the Burgers' equation using septic B-splines. Chaos Soliton Fract. **26**, 795–804 (2005)
12. B. Saka, I. Dag, A numerical study of the Burgers' equation. J. Frankl. Inst. **345**, 328–348 (2008)
13. T. Roshan, K.S. Bhamra, Numerical solutions of the modified Burgers' equation by Petrov–Galerkin method. Appl. Math. Comput. **218**, 3673–3679 (2011)
14. D. Irk, Sextic B-spline collocation method for the modified Burgers' equation. Kybernetes **38**(9), 1599–1620 (2009)
15. A.G. Bratsos, in *HERCMA 2009: An Implicit Numerical Scheme for the Modified Burgers' Equation*. Hellenic-European Conference on Computer Mathematics and its Applications, vol. 9, 24–26 September 2009, Athens, Greece
16. A.G. Bratsos, A fourth-order numerical scheme for solving the modified Burgers' equation. Comput. Math. Appl. **60**, 1393–1400 (2010)
17. A.G. Bratsos, L.A. Petrakis, An explicit numerical scheme for the modified Burgers' equation. Int. J. Numer. Methods Biomed. Eng. **27**, 232–237 (2011)
18. R.S. Temsah, Numerical solutions for convection–diffusion equation using El- Gendi method. Commun. Nonlinear Sci. Numer. Simul. **14**, 760–769 (2009)
19. A. Griewank, T.S. El-Danaf, Efficient accurate numerical treatment of the modified Burgers' equation. Appl. Anal. **88**(1), 75–87 (2009)
20. Y. Duan, R. Liu, Y. Jiang, Lattice Boltzmann model for the modified Burgers' equation. Appl. Math. Comput. **202**, 489497 (2008)
21. Z. Rong-Pei, Y. Xi-Jun, Z. Guo-Zhong, Modified Burgers' equation by the local discontinuous Galerkin method. Chin. Phys. B **22**(3), 030210 (2013)
22. C. Chen, C.H. Hsiao, Haar wavelet method for solving lumped and distributed parameter systems. IEE Proc. Control Theory Appl. **144**, 87–94 (1997)
23. U. Lepik, Numerical solution of differential equations using Haar wavelets. Math. Comput. Simul. **68**, 127–143 (2005)
24. U. Lepik, Numerical solution of evolution equations by the Haar wavelet method. Appl. Math. Comput. **185**, 695–704 (2007)
25. U. Lepik, Solving PDEs with the aid of two-dimensional Haar wavelets. Comput. Math. Appl. **61**, 1873–1879 (2011)
26. I. Çelik, Haar wavelet method for solving generalized Burgers–Huxley equation. Arab J. Math. Sci. **18**(1), 25–37 (2012)

27. I. Çelik, Haar wavelet approximation for magnetohydrodynamic flow equations. Appl. Math. Model. **37**, 3894–3902 (2013)
28. R. Jiwari, A Haar wavelet quasilinearization approach for numerical simulation of Burgers' equation. Comput. Phys. Commun. **183**, 2413–2423 (2012)
29. H. Kaura, R.C. Mittal, V. Mishra, Haar wavelet approximate solutions for the generalized Lane–Emden equations arising in astrophysics. Comput. Phys. Commun. **184**, 2169–2177 (2013)
30. Z. Shi, Y. Cao, Q.J. Chen, Solving 2D and 3D Poisson equations and biharmonic equations by the Haar wavelet method. Appl. Math. Model. **36**, 5143–5161 (2012)
31. C.F. Chen, C.-H. Hsiao, Wavelet approach to optimising dynamic systems. IEE Proc. Control Theory Appl. **146**(2), 213–219 (1999)
32. S.G. Rubin, R.A. Graves, *Cubic spline approximation for problems in fluid mechanics* (NASA TR R-436, Washington, 1975)
33. J.D. Hunter, Matplotlib: a 2D graphics environment. Comput. Sci. Eng. **9**(3), 90–95 (2007)
34. P.L. Sachdev, Ch. Srinivasa Rao, B.O. Enflo, Large-time asymptotics for periodic solutions of the modified Burgers' equation. Stud. Appl. Math. **114**, 307–323 (2005)
35. C. Basdevant, M. Deville, P. Haldenwang, J.M. Lacroix, Spectral and finite difference solutions of the Burgers' equation. Comput. Fluids **14**, 23–41 (1986)