Computational and Applied Mathematics (2021) 40:309
https://doi.org/10.1007/s40314-021-01692-3

An efficient Strang splitting technique combined with the multiquadric-radial basis function for the Burgers' equation

Muaz Seydaoğlu[1] · **Yusuf Uçar**[2] · **Selçuk Kutluay**[2]

Received: 18 December 2020 / Revised: 12 August 2021 / Accepted: 25 October 2021 /
Published online: 18 November 2021

Abstract
In the present paper, two effective numerical schemes depending on a second-order Strang splitting technique are presented to obtain approximate solutions of the one-dimensional Burgers' equation utilizing the collocation technique and approximating directly the solution by multiquadric-radial basis function (MQ-RBF) method. To show the performance of both schemes, we have considered two examples of Burgers' equation. The obtained numerical results are compared with the available exact values and also those of other published methods. Moreover, the computed L_2 and L_∞ error norms have been given. It is found that the presented schemes produce better results as compared to those obtained almost all the schemes present in the literature.

Keywords Burgers' equation · Strang splitting · Multiquadric-radial basis function

Mathematics Subject Classification 65L05 · 65M20 · 37M15

1 Introduction

In this study, we will deal with the numerical integration of the one-dimensional Burgers' equation

$$\frac{\partial u}{\partial t}+\frac{\partial}{\partial x}(\frac{u^2}{2})=\nu\frac{\partial^2 u}{\partial x^2},\quad (x,t)\in(0,1)\times(0,T], \tag{1.1}$$

with the initial condition

$$u(x,0)=u_0(x),\quad 0\leq x\leq 1, \tag{1.2}$$

Communicated by Eduardo Souza de Cursi.

✉ Muaz Seydaoğlu
m.seydaoglu@alparslan.edu.tr

1 Department of Mathematics, Faculty of Art and Science, Muş Alparslan University, 49100 Muş, Turkey

2 Department of Mathematics, Faculty of Art and Science, İnönü University, 44280 Malatya, Turkey

and the boundary conditions

$$u(0,t) = u_1(t), \quad u(1,t) = u_2(t) \quad 0 \le t \le T, \tag{1.3}$$

where x and t are space and time variables, respectively, T is the total time, $\nu > 0$ is known as kinematic viscosity parameter, and $u_0(x)$, $u_1(t)$, and $u_2(t)$ are the given functions, such that $u_0(x)$ is sufficiently smooth.

Equation (1.1) was first proposed by Bateman (1948) and later treated by Burger (1948) which was solved analytically with a restricted set of initial and boundary conditions in terms of infinite series using Hopf–Cole transformation (Hopf 1950; Cole 1951). Burgers' equation has been studied in different fields of research such as nonlinear and shock wave, molecular dynamics, gas dynamics, traffic flow, and cosmology. On the other hand, the performance of many algorithms can be compared by using the known analytical solutions of the Burger's equation. Therefore, many numerical techniques particularly based on finite difference, finite-element, and spectral methods Fletcher 1983; Kutluay et al. 1999; Öziş et al. 2003; Kutluay et al. 2004; Hassanien et al. 2005; Öziş et al. 2005; Acedo 2006; Liao 2008; Sari and Gurarslan 2009; Xie et al. 2010; Li et al. 2012; Inan and Bahadir 2013; Cordero et al. 2015 have been applied to solve Eqs. (1.1)–(1.3). Recently, Bonkile et al. (2018) have reviewed a brief history from physical and mathematical point of view and recent developments in numerical simulation of the Burgers' equation. Hon and Mao (1998) tailored an efficient numerical scheme for the Burgers' equation with shock wave applying the MQ-RBF as a spatial discretization scheme and a low-order explicit finite difference scheme for the discretization of the time variable. Chen and Wu (2006) tailored a scheme to obtain numerical solution of the Burgers' equation using the derivative of the quasi-interpolation and a low-order forward difference to approximate the spatial and the time derivative of the dependent variables, respectively. Haq et al. (2009) presented classical RBFs collocation method to find approximate solution of the nonlinear dispersive and dissipative KdV–Burgers' equation. Haq et al. (2012) applied a meshless method of lines using MQ, Gaussian and cubic RBFs for the numerical solution of the Burgers-type equations. Islam et al. (2012) presented Local Radial Basis Functions Collocation Method (LRBFCM) for the numerical solution of the transient nonlinear coupled Burgers' equations with large values of Reynolds number. Xie and Li (2013) tailored a new scheme to solve the one-dimensional Burgers' equation numerically by using the MQ-RBF for spatial approximation and a second-order compact finite difference scheme for temporal approximation. Xie et al. (2013) also proposed and implemented the method of particular solutions using inverse MQ and polyharmonic RBFs based on the finite difference scheme in time to solve one-dimensional time-dependent inhomogeneous Burgers' equation. A meshless method for the computation of a numerical solution of the unsteady Burgers-type equation based on thin-plate spline RBF was discussed in (Bouhamidi et al. 2014). Fan et al. (2014) proposed a meshfree approximation scheme derived from a combination of the MQ-RBF collocation method and the fictitious time integration method to approximate the solutions of the two-dimensional Burgers' equations. Sarboland and Aminataei (2014) presented two meshfree methods for solving the one-dimensional nonlinear non-homogeneous Burgers' equation using the MQ quasi-interpolation operator and direct and indirect radial basis function network schemes. Xie et al. (2016) suggested a meshfree scheme by utilizing the finite difference and the MQ-RBF method, for solving the Burgers' equation numerically.

Seydaoğlu et al. (2016) obtained the numerical solutions of Burgers' equation using high-order splitting methods combined with spectral methods, finite difference, and Weighted Essentially Nonoscillatory (WENO) schemes. Sarı et al. (2019) proposed higher order splitting schemes based on cubic B-spline Galerkin finite-element method to approximate solutions of the Burgers' equation model. Özer (2019) proposed the operator splitting techniques based on the quintic B-spline collocation finite-element method to obtain numerical solutions of the Rosenau–KdV–RLW equation. Strang splitting method and high-order splitting methods based on extrapolation technique were performed to approximate the solution of nonlinear coupled system viscous Burgers' equation in (Yağmurlu et al. 2020). Çiçek and Korkut (2021) used the Lie–Trotter splitting method (LSM) to solve the generalized Burgers–Huxley equation (GBHE) numerically.

In this paper, Eq. (1.1) is considered to solve numerically by a second-order Strang splitting technique combined with two different schemes. In both schemes, the MQ-RBF collocation method is performed to discretize Eq. (1.1) in space. The resulting system is then solved by applying the Crank–Nicolson finite difference approximation in the first scheme. In the second scheme, the nonlinear component of the split equation is solved by the classical fourth-order Runge–Kutta method.

The rest of this study is organized as follows: We first give a brief description of the Strang splitting method in Sect. 2 and then present the numerical schemes based on the MQ-RBF in Sect. 3. We perform the proposed combined schemes to the Burgers' equation mentioned before, and discuss the obtained numerical results, in Sect. 4.

2 A brief description of Strang splitting method

Take into consideration the nonlinear differential equation

$$\begin{aligned} \frac{\mathrm{d}u}{\mathrm{d}t} &= C_1(u) + C_2(u) = C(u), \qquad t \in (0, T] \\ u(0) &= u_0, \end{aligned} \tag{2.1}$$

where C, C_1, and C_2 are unbounded operators, and u_0 is a given function. The semi-discretized version of Eq. (1.1) is clearly an example of the form (2.1).

The principal idea of the time-splitting methods is that: first, split Eq.(2.1) into sub-equations as

$$\begin{aligned} \frac{\mathrm{d}u}{\mathrm{d}t} &= C_1(u), \\ \frac{\mathrm{d}u}{\mathrm{d}t} &= C_2(u); \end{aligned} \tag{2.2}$$

second, solve each sub-equation exactly or approximately using a suitable method and finally combine the solution of each sub-equation using the Strang splitting technique to find out the solution of the original problem. Let $\Phi_k^{[1]}$ and $\Phi_k^{[2]}$ be solution operators of each sub-problem in (2.2) for time-step size k, such that $u^{[1]}(t+k) = \Phi_k^{[1]}(u(t))$ and $u^{[2]}(t+k) = \Phi_k^{[2]}(u(t))$ (Blanes and Casas 2016). Next, the second-order Strang splitting method is given in terms of a combination of $\Phi_k^{[1]}$ and $\Phi_k^{[2]}$ as

$$u_{n+1} = \Phi_{k/2}^{[1]}(\Phi_k^{[2]}(\Phi_{k/2}^{[1]}(u_n))) \tag{2.3}$$

with $u(t_n) \approx u_n$ (Blanes and Casas 2016). The Strang splitting method is tailored by composing the Lie–Trotter method and its adjoint with the step size divided by two. Thus, the Strang splitting scheme is symmetric and of order 2 (Hairer et al. 2006).

3 Numerical schemes based on the MQ-RBF

In this section, we will consider two numerical schemes based on the RBF method which will be used to approximate the solutions of each sub-equation (3.5) and (3.4). The idea behind such method is to represent the distribution function $u(x)$ as a linear combination of $N+1$ continuously differentiable radial basis function, that is

$$u(\mathbf{x}) \approx \sum_{j=0}^{N} \lambda_j \phi_j(\mathbf{x}), \qquad \mathbf{x} \in \Omega \subset \mathbb{R}^d, \tag{3.1}$$

where N is the number of distributed points, $\mathbf{x} = (x_0, x_1, \ldots, x_{d-1})$, d is the dimension of the given problem, λ_j are unknown coefficients which are going to be determined using the collocation technique, and ϕ is a RBF (Kansa 1990a, b). In this paper, for the numerical schemes, we will use the MQ-RBF given as

$$\phi_j(\mathbf{x}) = \sqrt{r_j^2 + c^2}, \tag{3.2}$$

where $r_j = \|x - x_j\|$ denotes the Euclidean norm and c is a positive shape parameter which controls the shape of RBF. Therefore, the choice of c plays a crucial role for obtaining accurate solutions. The choice of its value is still an open research topic and most researchers use trial and error.

3.1 Scheme-A (S-A) for sub-equations

Let us take into consideration the viscous Burgers' equation (1.1) of the form

$$u_t + \left(\frac{u^2}{2}\right)_x = \nu u_{xx}, \qquad u(x,0) = u_0(x). \tag{3.3}$$

To employ splitting technique, Eq. (3.3) split into two sub-equations known as the diffusion equation

$$u_t = \nu u_{xx}, \qquad u(x,0) = u_0(x) \tag{3.4}$$

and inviscid Burgers' equation

$$u_t = -\left(\frac{u^2}{2}\right)_x, \qquad u(x,0) = u_0(x), \tag{3.5}$$

to obtain its approximate solution by combining the solutions of Eqs. (3.4) and (3.5). If one denotes the exact (or accurate numerical) solution of Eqs. (3.5) and (3.4) by the maps $\Phi_k^{[1]}$ and $\Phi_k^{[2]}$, respectively, next the approximate solution of (3.3) by the Strang splitting technique for small time-step k is given as

$$u(x,k) = \Phi_{k/2}^{[1]}(\Phi_k^{[2]}(\Phi_{k/2}^{[1]} u_0(x))).$$

Let us discretize (3.4) using the Crank–Nicolson type approximation to obtain

$$\frac{u^{n+1} - u^n}{k} = \nu \frac{u_{xx}^{n+1} + u_{xx}^n}{2}, \quad n \geq 0, \tag{3.6}$$

where $k = t^{n+1} - t^n$ is the time-step and $u^n = u(x, nk)$. Equation (3.6) can be rewritten as

$$u^{n+1} - \frac{k\nu}{2} u_{xx}^{n+1} = u^n + \frac{k\nu}{2} u_{xx}^n, \quad n \geq 0. \tag{3.7}$$

Following (Hon and Mao 1998), one can approximate the u^{n+1} at each time iteration $n+1$ using the modified MQ-RBF by Kansa (1990b) as follows:

$$u^{n+1}(x) \approx \sum_{j=0}^{N} \lambda_j^{n+1} \phi_j(x) + \lambda_{N+1}^{n+1} x + \lambda_{N+2}^{n+1}, \tag{3.8}$$

where $\phi_j(x) = \sqrt{(x-x_j)^2 + c_j^2}$, $x_j = j/N$, $j = 0, 1, \ldots, N$ are $N+1$ distinct and uniformly distributed points over the solution range $[0, 1]$. The c_j are referred to as shape parameters and their values are the cornerstone of the accuracy of the approximation. The values of c_j is computed by the formula

$$c_j = Mj + b, \quad j = 0, 1, \ldots, N, \tag{3.9}$$

in which M and b are parameters to be chosen, such that the error drops to the minimum (Hon and Mao 1998). Additionally, Hon and Mao (1998) developed an adaptive MQ point selecting approach depending on 'chasing the peak' of the shock wave in the presence of small viscosity. The $(N+1)$ points x_j in (3.8) are taken as $x_j = jx^*/L$, $j = 0, 1, \ldots, L$ and $x_j = x^* + (j-L)(1-x^*)/(N-L)$, $j = (L+1), \ldots, N$ where $\mathrm{d}u^{n+1}(x^*)/\mathrm{d}x = 0$. However, the point x^* is computed with Newton's iterative scheme as

$$x^*_{new} = x^*_{old} - \frac{\mathrm{d}u^{n+1}(x^*_{old})/\mathrm{d}x}{d^2u^{n+1}(x^*_{old})/\mathrm{d}x^2}, \tag{3.10}$$

where the initial iteration at each time-step $n+1$ is the x^* calculated by the usage of the last step in time n.

However, the second derivative of the approximate solution is given as

$$\frac{d^2}{\mathrm{d}x^2} u^{n+1}(x) \approx \sum_{j=0}^{N} \lambda_j^{n+1} \frac{d^2}{\mathrm{d}x^2} \phi_j(x). \tag{3.11}$$

To calculate the $N+3$ coefficients λ_j^{n+1} at each time level $n+1$, one can consider the boundary conditions (1.3) which yields the following two equations:

$$u^{n+1}(x_0) = u_1^{n+1}, \quad u^{n+1}(x_N) = u_2^{n+1}, \tag{3.12}$$

and then collocate u^{n+1} at the remaining $(N+1)$ distinct and uniformly distributed points $\hat{x}_i = \frac{i}{N+2}$ in $(0, 1)$ using Eq. (3.7) to obtain

$$u^{n+1}(\hat{x}_i) - \frac{k\nu}{2} u_{xx}^{n+1}(\hat{x}_i) = u^n(\hat{x}_i) + \frac{k\nu}{2} u_{xx}^n(\hat{x}_i), \tag{3.13}$$

where $u^0(\hat{x}_i) = u_0(\hat{x}_i)$ for $i = 1, 2, \ldots, N+1$. Finally, the coefficients λ_j^{n+1} are obtained by solving the resulting system of (3.12) and (3.13).

Now, let us discretize Eq. (3.5) and use again the linearized Crank–Nicolson-type approximation to obtain

$$\frac{u^{n+1} - u^n}{k} = -\frac{u^n u_x^{n+1} + u^{n+1} u_x^n}{2}, \quad n \geq 0, \tag{3.14}$$

or

$$u^{n+1} + \frac{k}{2} u^{n+1} u_x^n + \frac{k}{2} u^n u_x^{n+1} = u^n, \quad n \geq 0. \tag{3.15}$$

Let us approximate u^{n+1} and its first derivative at each time iteration $n+1$ with (3.8) and

$$\frac{d}{\mathrm{d}x}u^{n+1}(x) \approx \sum_{j=0}^{N} \lambda_j^{n+1} \frac{d}{\mathrm{d}x}\phi_j(x) + \lambda_{N+1}^{n+1}, \tag{3.16}$$

respectively. To calculate $(N+3)$ coefficients λ_j^{n+1} at each time level $n+1$, one has two equations (3.12) obtained from the boundary conditions (1.3), and then, the remaining $(N+1)$ equations can be obtained by collocating u^{n+1} at the $(N+1)$ distinct and uniformly distributed points $\hat{x}_i = i/(N+2)$ for $i = 1, 2, \ldots, N+1$ in $(0, 1)$ as

$$u^{n+1}(\hat{x}_i) + \frac{k}{2}u^{n+1}(\hat{x}_i)u_x^n(\hat{x}_i) + \frac{k}{2}u^n(\hat{x}_i)u_x^{n+1}(\hat{x}_i) = u^n(\hat{x}_i), \quad n \geq 0. \tag{3.17}$$

The coefficients λ_j^{n+1} are obtained by solving the resulting system of (3.12) and (3.17). To solve the obtained systems in this subsection, one can use Gaussian elimination with partial pivoting.

3.2 Scheme-B (S-B) for sub-equations

In this scheme, Eq. (1.1) is transformed into a system of ODEs by discretization of the space variable with the MQ-RBF method. Next, the obtained system of ODEs is split into two sub-equations and the approximate solutions are found by the second-order Strang splitting technique (2.3). One can approximate the function $u(x, t)$ as follows:

$$u^N(x,t) = \sum_{j=0}^{N} \lambda_j(t)\phi_j(x) = \boldsymbol{\Phi}^T(x)\boldsymbol{\lambda}, \tag{3.18}$$

where $\phi_j(x)$ is the MQ-RBF given by Eq. (3.2), $x_j = j/N$, $j = 0, 1, \ldots, N$ are $(N+1)$ distinct and uniformly distributed points in the interval $[0, 1]$ and

$$\boldsymbol{\Phi}(x) = (\phi_0(x), \phi_1(x), \ldots, \phi_N(x))^T, \quad \boldsymbol{\lambda} = (\lambda_0(t), \lambda_1(t), \ldots, \lambda_N(t))^T. \tag{3.19}$$

However, the first and second derivatives of the function $u^N(x, t)$ are given as

$$\frac{\partial}{\partial x}u^N(x,t) = \sum_{j=0}^{N} \lambda_j(t)\frac{d}{\mathrm{d}x}\phi_j(x), \tag{3.20}$$

and

$$\frac{\partial^2}{\partial x^2}u^N(x,t) = \sum_{j=0}^{N} \lambda_j(t)\frac{d^2}{\mathrm{d}x^2}\phi_j(x), \tag{3.21}$$

respectively. If one denotes $u^N(x_i, t) = u_i(t) = u(x_i, t)$, for $i = 1, \ldots, N$, then Eq. (3.18) reads

$$\boldsymbol{A}\boldsymbol{\lambda} = \boldsymbol{u}, \tag{3.22}$$

where $\boldsymbol{u} = (u_0(t), u_1(t), \ldots, u_N(t))$ and

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{\Phi}^T(x_0) \\ \boldsymbol{\Phi}^T(x_1) \\ \vdots \\ \boldsymbol{\Phi}^T(x_N) \end{bmatrix} = \begin{bmatrix} \phi_0(x_0) & \phi_1(x_0) & \ldots & \phi_N(x_0) \\ \phi_0(x_1) & \phi_1(x_1) & \ldots & \phi_N(x_1) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(x_N) & \phi_1(x_N) & \ldots & \phi_N(x_N) \end{bmatrix}. \tag{3.23}$$

Using the above notations, one can obtain the following equality from Eqs. (3.18) and (3.22):

$$u^N(x,t) = \boldsymbol{\Phi}^T(x)\boldsymbol{A}^{-1}u = D(x)u, \tag{3.24}$$

where $\boldsymbol{D}(x) = \Phi^T(x)\boldsymbol{A}^{-1} = (D_0(x), D_1(x), \ldots, D_N(x))$. From the definition of the RBF approximation, we have

$$u(x,t) \approx u^N(x,t) = \sum_{j=0}^{N} \lambda_j(t)\phi_j(x) = \boldsymbol{\Phi}^T(x)\boldsymbol{\lambda} = \boldsymbol{\Phi}^T(x)\boldsymbol{A}^{-1}u = D(x)u. \tag{3.25}$$

Inserting (3.25) into (1.1) with the collocation $u^N(x,t)$ at $(N+1)$ distinct and uniformly distributed points $x_i = \frac{i}{N}$ for $i = 0, 1, \ldots, N$ in $[0, 1]$ yields

$$\frac{\mathrm{d}u_i(t)}{\mathrm{d}t} + u_i(t)(\boldsymbol{D}_x(x_i)u(t)) = \nu(\boldsymbol{D}_{xx}(x_i)u(t)), \tag{3.26}$$

where

$$\boldsymbol{D}_x(x_i) = \left(\frac{\partial}{\partial x}D_0(x_i), \frac{\partial}{\partial x}D_1(x_i), \ldots, \frac{\partial}{\partial x}D_N(x_i)\right), \tag{3.27}$$

$$\boldsymbol{D}_{xx}(x_i) = \left(\frac{\partial^2}{\partial x^2}D_0(x_i), \frac{\partial^2}{\partial x^2}D_1(x_i), \ldots, \frac{\partial^2}{\partial x^2}D_N(x_i)\right). \tag{3.28}$$

For simplicity, Eq. (3.26) can be rewritten in the following form:

$$\frac{\mathrm{d}U(t)}{\mathrm{d}t} + U(t) * (\boldsymbol{D}_x U(t)) = \nu \boldsymbol{D}_{xx} U(t), \tag{3.29}$$

where $\boldsymbol{U}=(u_0, u_1, \ldots, u_N)$ and $\boldsymbol{D}_x=\left[\frac{\partial}{\partial x}D_j(x_i)\right]_{(N+1)\times(N+1)}$, $\boldsymbol{D}_{xx}=\left[\frac{\partial^2}{\partial x^2}D_j(x_i)\right]_{(N+1)\times(N+1)}$. Here, the symbol "$*$" represents element-wise multiplication of two vectors. Now, we split Eq. (3.29) into two sub-equations as

$$\frac{\mathrm{d}U(t)}{\mathrm{d}t} = \nu \boldsymbol{D}_{xx} U(t), \tag{3.30}$$

$$\frac{\mathrm{d}U(t)}{\mathrm{d}t} = -U(t) * (\boldsymbol{D}_x U(t)). \tag{3.31}$$

The solution of Eq. (3.30) can be read as follows:

$$U(t) = \exp(\nu \boldsymbol{D}_{xx} t)U(0),$$

where the exponential of matrices can be computed using an efficient method based on scaling-squaring Padé or Taylor algorithms (Al-Mohy and Higham 2009; Sastre et al. 2014; Bader et al. 2015). We have used scaling-squaring Padé algorithm presented in (Al-Mohy and Higham 2009) to compute exponential of matrices. This algorithm implemented in the *expm* MATLAB routine. On the other hand, Eq. (3.31) can also be rewritten in the following form:

$$\frac{\mathrm{d}U(t)}{\mathrm{d}t} = F(U(t)),$$

where $F(U(t)) = -U(t)*(\boldsymbol{D}_x U(t))$ and one can solve this equation by utilizing the classical fourth-order Runge–Kutta method as follows:

$$\begin{aligned} \xi_1 &= F(U_n), \\ \xi_2 &= F(U_n + \frac{k}{2}\xi_1), \\ \xi_3 &= F(U_n + \frac{k}{2}\xi_2), \\ \xi_4 &= F(U_n + k\xi_3), \\ U_{n+1} &= U_n + \frac{1}{6}k(\xi_1 + 2\xi_2 + 2\xi_3 + \xi_4), \end{aligned} \tag{3.32}$$

where n and k represent n^{th} time level and time-step, respectively. Once the solutions of Eqs. (3.30) and (3.31) are computed, then Eq. (3.22) is used to obtain $\boldsymbol{\lambda}$. On the other hand, we deal with the system of ODEs in time. The Rule of Thumb yields accurate predictions of the stability, that is, the eigenvalues of the spatial discretization operator must lie in the stability region of the Runge–Kutta method (Trefethen 2000). Thus, one needs to use small time-steps. Notice that the splitting technique allows us to split Burgers' equation into the linear and nonlinear parts. The linear part corresponds to Eq. (3.30) which is a stiff system of ODEs and it constraints the stability. We solve the linear part exactly and this can offer the possibility to use larger time-steps.

4 Numerical results

In the present section, we are going to consider two test examples, so that each one has an analytical solution. All computations were executed using MATLAB R2013 a software on a Intel(R) Core(TM)i7-4500U CPU 1.80GHz machine with 8 GB of memory. To measure the accuracy of the presented schemes for each example, we will use the error norms L_2 and L_∞ described

$$\begin{aligned} L_2 &= \left(\delta x \sum_{j=1}^{N} (u_j^{num} - u(x_j, t))^2 \right)^{1/2}, \\ L_\infty &= \max_{1 \le j \le N} \left| u_j^{num} - u(x_j, t) \right|. \end{aligned}$$

Example 1 As a first test problem, we consider the Burgers' equation (1.1) over [0, 1] with the following initial condition:

$$u(x, t = 0) = \sin(\pi x), \tag{4.1}$$

and the boundary conditions

$$u(x = 0, t) = u(x = 1, t) = 0, \qquad t > 0. \tag{4.2}$$

The analytical solution for the problem can be expressed by utilizing the Hopf–Cole transformation (Hopf 1950; Cole 1951) as

$$u(x,t) = 2\nu\pi \frac{\sum_{n=1}^{\infty} c_n \exp(-n^2\pi^2\nu t) n \sin(n\pi x)}{c_0 + \sum_{n=1}^{\infty} c_n \exp(-n^2\pi^2\nu t)\cos(n\pi x)}, \tag{4.3}$$

where

$$\begin{aligned} c_0 &= \int_0^1 \exp\left\{-(2\pi\nu)^{-1}\left[1-\cos(\pi x)\right]\right\} \mathrm{d}x, \\ c_n &= 2\int_0^1 \exp\left\{-(2\pi\nu)^{-1}\left[1-\cos(\pi x)\right]\right\} \cos(n\pi x)\mathrm{d}x, \qquad (n=1,2,3\ldots). \end{aligned} \tag{4.4}$$

As the second test problem, we consider the Burgers' equation (1.1) over [0, 1] with

$$u(x, t=0) = 4x(1-x), \tag{4.5}$$

and (4.2). The analytical solution of the present problem is expressed by (4.3) with the following coefficients:

$$\begin{aligned} c_0 &= \int_0^1 \exp\left\{-x^2(3\nu)^{-1}(3-2x)\right\} \mathrm{d}x, \\ c_n &= 2\int_0^1 \exp\left\{-x^2(3\nu)^{-1}(3-2x)\right\} \cos(n\pi x)\mathrm{d}x, \qquad (n=1,2,3\ldots). \end{aligned} \tag{4.6}$$

For S-A, we have used the same values of parameters given in Hon and Mao 1998 for M, b for all time-step k in the numerical results shown in Tables 1-6. In other numerical results, we selected the value of the shape parameters by trial and error. Good agreement between our approximate results and the analytical solutions can be clearly seen from all the presented tables.

Table 1 displays a comparison of the computed numerical solutions of Example 1 using two presented schemes S-A and S-B at time $T = 0.02$ for $\nu = 10$, $N = 10$, $k = 0.001$, 0.0001 with the analytical one and also other results reported in (Hon and Mao 1998; Xie and Li 2013). We have used the values $M = -0.2$, $b = 2.5$ in S-A and shape parameter $c = 0.48$ for $k = 0.001$ and $c = 1.73$ for $k = 0.0001$ in S-B. The numerical results found out by this manuscript are in good agreement with analytical solutions. We have compared our results with the reported ones which are obtained using schemes S-I and S-II in (Xie and Li 2013). S-I is proposed to solve the Burgers' equation using MQ-RBF for space discretization and a second-order compact finite difference scheme for time discretization. Furthermore, S-II is based on the method of lines. As seen in Table 1, our scheme S-B gives more accurate results in comparison with the results given in S-I (Xie and Li 2013) and S-A yields a similarly accurate results to S-I in (Xie and Li 2013) for time-step $k = 0.001$. Because of the stability restriction, the scheme S-II in (Xie and Li 2013) does not give results for $k = 0.001$, but the scheme S-B gives accurate solutions. On the other hand, S-A and S-B provide a similarly accurate solution in comparison with the results given in S-I (Xie and Li 2013) and yield more accurate solutions than those reported in (Hon and Mao 1998) and S-II (Xie and Li 2013) for $k = 0.0001$.

Table 2 illustrates the obtained numerical solutions of Example 1 by both presented methods at time $T = 1$ for $\nu = 0.1$, $N = 10$, $k = 0.1, 0.01$ with analytical solutions and those given in (Hon and Mao 1998; Xie and Li 2013). We have used the values $M = -0.2e$, $b = 2.4$ for all k in S-A and the values of shape parameter as $c = 0.3$, $c = 0.84$ for $k = 0.1$, $k = 0.01$, respectively, in S-B. The approximate values found out by the present manuscript

Table 1 Comparison of analytical and approximate solutions of Example 1 with results from (Hon and Mao 1998; Xie and Li 2013) at $T = 0.02$ for $\nu = 10$

x	Analytical	$k = 0.001$				$k = 0.0001$				
		(Hon and Mao 1998) $N = 10$	S-I Xie and Li 2013 $N = 10$	S-A $N = 10$	S-B $N = 10$	(Hon and Mao 1998) $N = 10$	S-I(Xie and Li 2013) $N = 10$	S-II Xie and Li 2013 $N = 10$	S-A $N = 10$	S-B $N = 10$
0.1	0.042836	0.0469	0.0428	0.042820	0.042834	0.0433	0.0428	0.0424	0.042837	0.042836
0.2	0.081504	0.0893	0.0814	0.081473	0.081503	0.0823	0.0815	0.0807	0.081505	0.081504
0.3	0.112235	0.1230	0.1121	0.112191	0.112232	0.1133	0.1122	0.1111	0.112235	0.112235
0.4	0.132020	0.1447	0.1318	0.131967	0.132016	0.1333	0.1320	0.1307	0.132020	0.132020
0.5	0.138907	0.1522	0.1387	0.138851	0.138902	0.1403	0.1389	0.1375	0.138906	0.138907
0.6	0.132197	0.1449	0.1320	0.132143	0.132191	0.1335	0.1322	0.1309	0.132196	0.132197
0.7	0.112521	0.1234	0.1123	0.112475	0.112515	0.1136	0.1125	0.1114	0.112521	0.112521
0.8	0.081791	0.0897	0.0817	0.081756	0.081785	0.0826	0.0818	0.0810	0.081790	0.081791
0.9	0.043013	0.0472	0.0429	0.042995	0.043007	0.0434	0.0430	0.0426	0.043012	0.043013
L_2				3.9558E-05	4.3699E-06				5.4608E-07	6.9260E-09
L_∞				5.5880E-05	6.1465E-06				8.3266E-07	9.8431E-09

Table 2 Comparison of analytical and approximate solutions of Example 1 with results from (Hon and Mao 1998; Xie and Li 2013) at $T = 1$ for $\nu = 0.1$

x	Analytical	$k = 0.1$				$k = 0.01$				
		(Hon and Mao 1998) $N = 10$	S-I Xie and Li 2013 $N = 10$	S-A $N = 10$	S-B $N = 10$	(Hon and Mao 1998)	S-I (Xie and Li 2013)	S-II Xie and Li 2013 $N = 10$	S-A $N = 10$	S-B $N = 10$
0.1	0.066316	0.0670	0.0665	0.066562	0.067361	0.0664	0.0665	0.0652	0.066514	0.066251
0.2	0.131209	0.1325	0.1313	0.131514	0.132203	0.1314	0.1314	0.1295	0.131430	0.131139
0.3	0.192786	0.1948	0.1929	0.193177	0.193441	0.1930	0.1930	0.1905	0.192991	0.192714
0.4	0.248041	0.2512	0.2481	0.248523	0.248316	0.2483	0.2482	0.2452	0.248200	0.247981
0.5	0.291916	0.2969	0.2919	0.292457	0.291704	0.2923	0.2920	0.2885	0.292001	0.291878
0.6	0.316068	0.3239	0.3159	0.316467	0.315234	0.3167	0.3160	0.3123	0.316047	0.316060
0.7	0.308089	0.3192	0.3077	0.307857	0.306494	0.3090	0.3079	0.3042	0.307940	0.308116
0.8	0.253718	0.2666	0.2531	0.252355	0.251676	0.2548	0.2535	0.2504	0.253451	0.253769
0.9	0.146065	0.1554	0.1455	0.144225	0.144466	0.1468	0.1458	0.1442	0.145775	0.146125
L_2				7.9323E-04	1.3524E-03				1.8478E-04	5.4200E-05
L_∞				1.8405E-03	2.0418E-03				2.9062E-04	7.2033E-05

are in good agreement with analytical ones. From Table 2, it is found that the results given in S-I (Xie and Li 2013) are better than others for $k = 0.1$. However, the solutions obtained with S-B are much better than the solutions reported in (Hon and Mao 1998; Xie and Li 2013) for $k = 0.01$. Furthermore, the results obtained by S-A are similar with those in S-I (Xie and Li 2013) and better than those in S-II (Xie and Li 2013) for $k = 0.01$.

Tables 3–4 present a comparison of the numerical results of Example 1 found out by S-A at final times $T = 1, 0.5$ for values of $\nu = 0.01$, $N = 10$, $k = 0.1$, $k = 0.01$, $k = 0.001$ using $M = -0.2e$, $b = 2.22$ for all k with the analytical ones and also those reported in Hon and Mao 1998; Xie and Li 2013; Zhu and Wang 2009. It is observed that the presented scheme S-A provides much better results than those given in (Hon and Mao 1998; Xie and Li 2013; Zhu and Wang 2009).

In Tables 5 and 6, we have compared the results of Example 1 obtained by S-A proposed in this study combined with and without adaptive MQ given in (Hon and Mao 1998) with the accurate results presented in (Christie and Mitchell 1978) and the best results reported in the (Hassanien et al. 2005; Hon and Mao 1998; Chen and Wu 2006; Iskandar and Mohsen 1992; Christie and Mitchell 1978; Zhu and Wang 2009) on the data set $T = 1$, $\nu = 0.0001$, $N = 10$, $k = 0.1$, $k = 0.01$ and $M = -0.2e$, $b = 2.205$ for all k. One can see from Table 5 that S-A gives better accuracy for all k. Table 6 shows the scheme S-A (Adaptive) which provides higher accuracy for $k = 0.1$, similar accuracy with Hon and Mao 1998 and better accuracy than those given in Hassanien et al. 2005; Chen and Wu 2006; Caldwell et al. 1987; Zhu and Wang 2009 for $k = 0.001$.

Table 7 compares the approximate solutions of Example 1 obtained by algorithms S-A and S-B with the analytical ones and those reported in (Kutluay et al. 1999, 2004; Hassanien et al. 2005; Liao 2008; Wang and Lu 2005; Venkatesh et al. 2017; Xu et al. 2011) on the data set $T = 0.1$, $\nu = 1$, $N = 10$ with $k = 0.01$, $M = -0.2e$, $b = 2.5$ for S-A and $c = 1.89$, $k = 0.001$ for S-B. From the table, it is seen that even for large time-step and less number of collocation points, S-B has better accuracy than the others. Notice that S-A also shows superior performance to the others even with larger time step.

Table 8 illustrates a comparison of the numerical results of Example 1 obtained by S-A and S-B with the analytical ones and also those given in (Kutluay et al. 1999; Öziş et al. 2003, 2005; Jiwari and Alshomrani 2017) at $T = 0.4, 0.6, 0.8, 1$ for $\nu = 0.1$, $N = 10$ with $k = 0.01$ for Example 1. We have used the parameter values $M = -0.2e$, $b = 2.1$ in S-A and $c = 0.29$, $c = 0.3$, $c = 0.89$, $c = 0.82$ at times $T = 0.4, 0.6, 0.8, 1$, respectively, in S-B. It is observed from Table 8 that the accuracy of S-A is similar and sometimes better than the results presented in (Kutluay et al. 1999; Öziş et al. 2005) even for large step size.

In Table 9, we have compared the numerical results of Example 2 obtained by S-A and S-B with the analytical ones and those given in Jiwari 2015 for $\nu = 0.1, 0.01$, $N = 10$ with $k = 0.001$. We have used the parameter values $M = -0.2$, $b = 2.1, 2.3, 2.3, 2.3, 2.5$ for $\nu = 0.1$ and $b = 2.204, 2.199, 2.17, 2.174, 2.063$ for $\nu = 0.01$ at $T = 0.4, 0.6, 0.8, 1, 3$, respectively, in S-A and $c = 0.304$, $c = 0.306$, $c = 0.320$, $c = 0.825$, $c = 0.826$ for $\nu = 0.1$ and $c = 1.854$, $c = 1.959$, $c = 1.958$, $c = 2.071$, $c = 0.352$ for $\nu = 0.01$ at the preselected times $T = 0.4, 0.6, 0.8, 1, 3$, respectively, in S-B. It is observed from Table 9 that the accuracy of S-A is the best for $\nu = 0.1$. On the other hand, S-A is similar and sometimes better than the results presented in (Jiwari 2015) for $\nu = 0.01$.

Table 10 displays the computed numerical results of Example 2 by both algorithms S-A and S-B for $\nu = 0.1$, $N = 10$ and $k = 0.01$. We have taken $M = -0.2e$, $b = 2.095, 2.097, 2.097, 2.098, 2.304$ for $T = 0.4, 0.6, 0.8, 1, 3$, respectively, in S-A and $c = 0.332$, $c = 0.321$, $c = 0.324$, $c = 0.332$, $c = 0.819$ at times $t = 0.4, 0.6, 0.8, 1, 3$, respectively, in S-B. The obtained results have been compared with the analytical ones and

Table 3 Comparison of analytical and approximate solutions of Example 1 with results from (Hon and Mao 1998; Xie and Li 2013; Zhu and Wang 2009) at $T = 1$ for $\nu = 0.01$

x	Analytical	(Hon and Mao 1998)			(Xie and Li 2013)(Non-Unif.)	BSQI (Zhu and Wang 2009)	S-A		
		$N = 10$			$N = 10$	$N = 100$	$N = 10$		
		$k = 0.1$	$k = 0.01$	$k = 0.001$	$k = 0.001$	$k = 0.1$	$k = 0.1$	$k = 0.01$	$k = 0.001$
0.1	7.5382E-02	7.54E-02	7.55E-02	7.55E-02	7.59E-02	7.530E-02	7.5528E-02	7.5481E-02	7.5482E-02
0.2	1.50645E-01	1.506E-01	1.507E-01	1.507E-01	1.506E-01	1.5049E-01	1.50733E-01	1.50672E-01	1.50671E-01
0.3	2.25666E-01	2.253E-01	2.256E-01	2.257E-01	2.255E-01	2.2544E-01	2.25770E-01	2.25671E-01	2.25670E-01
0.4	3.00309E-01	2.995E-01	3.002E-01	3.003E-01	3.003E-01	3.0002E-01	3.00487E-01	3.00312E-01	3.00310E-01
0.5	3.74420E-01	3.726E-01	3.742E-01	3.744E-01	3.745E-01	3.7407E-01	3.74727E-01	3.74423E-01	3.74420E-01
0.6	4.47816E-01	4.444E-01	4.475E-01	4.478E-01	4.480E-01	4.4742E-01	4.48325E-01	4.47821E-01	4.47816E-01
0.7	5.20268E-01	5.143E-01	5.197E-01	5.202E-01	5.205E-01	5.1985E-01	5.21075E-01	5.20281E-01	5.20273E-01
0.8	5.91476E-01	5.813E-01	5.905E-01	5.913E-01	5.911E-01	5.9106E-01	5.92729E-01	5.91445E-01	5.91432E-01
0.9	6.60019E-01	6.438E-01	6.594E-01	6.607E-01	6.651E-01	6.5964E-01	6.63985E-01	6.60920E-01	6.60890E-01
L_2							1.3553E-03	2.8700E-04	2.7758E-04
L_∞							3.9653E-03	9.0105E-04	8.7054E-04

Table 4 Comparison of analytical and approximate solutions with results from (Hon and Mao 1998; Xie and Li 2013) of Example 1 at $T = 0.5$ for $\nu = 0.01$

x	Analytical	(Hon and Mao 1998) $N = 10$			S-A $N = 10$			(Xie and Li 2013)(Non-Unif.) $N = 10$
		$k = 0.1$	$k = 0.01$	$k = 0.001$	$k = 0.1$	$k = 0.01$	$k = 0.001$	$k = 0.001$
0.1	0.121144	0.1211	0.1211	0.1211	0.121210	0.121141	0.121140	0.1211
0.2	0.241514	0.2413	0.2415	0.2415	0.241636	0.241522	0.241520	0.2415
0.3	0.360271	0.3589	0.3601	0.3603	0.360605	0.360271	0.360268	0.3602
0.4	0.476423	0.4724	0.4760	0.4764	0.477176	0.476431	0.476424	0.4765
0.5	0.588696	0.5799	0.5877	0.5886	0.590165	0.588709	0.588695	0.5887
0.6	0.695308	0.6780	0.6935	0.6951	0.697952	0.695334	0.695309	0.6954
0.7	0.793493	0.7615	0.7901	0.7932	0.798052	0.793546	0.793502	0.7936
0.8	0.878303	0.8189	0.8719	0.8776	0.886489	0.878294	0.878215	0.8774
0.9	0.938107	0.8128	0.9264	0.9381	0.960429	0.939606	0.939422	0.9486
L_2					7.7196E-03	4.7435E-04	4.1688E-04	
L_∞					2.2323E-02	1.4987E-03	1.3153E-03	

Table 5 Comparison of accurate (Christie and Mitchell 1978) and approximate solutions of Example 1 with results from Hassanien et al. 2005; Hon and Mao 1998; Iskandar and Mohsen 1992; Caldwell et al. 1987; Zhu and Wang 2009 at $T = 1$ for $\nu = 0.0001$

x	Accurate (Christie and Mitchell 1978)	(Iskandar and Mohsen 1992)	(Caldwell et al. 1987)	(Hassanien et al. 2005)	(Zhu and Wang 2009)	(Hon and Mao 1998)		S-A	
		$N = 18$	$N = 16$	$N = 180$	$N = 100$	$N = 10$		$N = 10$	
		$k = 0.1$	$k = 0.001$	$k = 0.01$	$k = 0.001$	$k = 0.1$	$k = 0.001$	$k = 0.1$	$k = 0.001$
0.05	4.22E-02	4.19E-02	4.22E-02	3.79E-02	4.24E-02	4.22E-02	4.24E-02	4.2274E-02	4.2261E-02
0.11	8.43E-02	8.39E-02	8.44E-02	8.34E-02	8.43E-02	8.43E-02	8.43E-02	8.4290E-02	8.4282E-02
0.16	1.263E-01	1.253E-01	1.266E-01	1.213E-01	1.262E-01	1.263E-01	1.263E-01	1.26338E-01	1.26329E-01
0.22	1.684E-01	1.692E-01	1.687E-01	1.667E-01	1.680E-01	1.682E-01	1.684E-01	1.68374E-01	1.68356E-01
0.27	2.103E-01	2.034E-01	2.108E-01	2.044E-01	2.096E-01	2.100E-01	2.103E-01	2.10344E-01	2.10307E-01
0.33	2.522E-01	2.666E-01	2.527E-01	2.469E-01	2.513E-01	2.517E-01	2.522E-01	2.52227E-01	2.52163E-01
0.38	2.939E-01	2.527E-01	2.946E-01	2.872E-01	2.928E-01	2.931E-01	2.939E-01	2.94008E-01	2.93907E-01
0.44	3.355E-01	3.966E-01	3.362E-01	3.322E-01	3.342E-01	3.343E-01	3.355E-01	3.35662E-01	3.35511E-01
0.50	3.769E-01	2.350E-01	3.778E-01	3.769E-01	3.754E-01	3.751E-01	3.769E-01	3.77165E-01	3.76947E-01
0.55	4.182E-01	5.480E-01	4.191E-01	4.140E-01	4.147E-01	4.156E-01	4.182E-01	4.18494E-01	4.18189E-01
0.61	4.592E-01	2.578E-01	4.601E-01	4.584E-01	4.573E-01	4.557E-01	4.592E-01	4.59620E-01	4.59205E-01
0.66	5.000E-01	6.049E-01	5.009E-01	4.951E-01	4.978E-01	4.952E-01	4.999E-01	5.00514E-01	4.99960E-01
0.72	5.404E-01	6.014E-01	5.414E-01	5.388E-01	5.381E-01	5.341E-01	5.404E-01	5.41155E-01	5.40430E-01
0.77	5.806E-01	4.630E-01	5.816E-01	5.749E-01	5.779E-01	5.722E-01	5.805E-01	5.81483E-01	5.80550E-01
0.83	6.203E-01	7.011E-01	6.213E-01	6.179E-01	6.174E-01	6.093E-01	6.201E-01	6.21419E-01	6.20235E-01
0.88	6.596E-01	6.717E-01	6.605E-01	6.533E-01	6.564E-01	6.455E-01	6.600E-01	6.61319E-01	6.59830E-01
0.94	6.983E-01	7.261E-01	6.992E-01	6.952E-01	6.947E-01	6.783E-01	6.957E-01	6.99000E-01	6.97140E-01

Table 6 Comparison of accurate (Christie and Mitchell 1978) and approximate solutions of Example 1 with results from Hassanien et al. 2005; Hon and Mao 1998; Chen and Wu 2006; Caldwell et al. 1987; Zhu and Wang 2009 at $T = 1$ for $\nu = 0.0001$

x	Accurate (Christie and Mitchell 1978)	(Caldwell et al. 1987)	(Hassanien et al. 2005)	(Zhu and Wang 2009)	(Chen and Wu 2006)	(Hon and Mao 1998)(Adaptive)		S-A (Adaptive)	
		$N = 16$	$N = 180$	$N = 100$	$N = 72$	$N = 10$		$N = 10$	
		$k = 0.001$	$k = 0.01$	$k = 0.001$	$k = 0.001$	$k = 0.1$	$k = 0.001$	$k = 0.1$	$k = 0.001$
0.05	4.22E-02	4.22E-02	3.79E-02	4.24E-02	4.2395E-02	4.21E-02	4.21E-02	4.2102E-02	4.2108E-02
0.11	8.43E-02	8.44E-02	8.34E-02	8.43E-02	8.4337E-02	8.43E-02	8.43E-02	8.4290E-02	8.4272E-02
0.16	1.263E-01	1.266E-01	1.213E-01	1.262E-01	1.2619E-01	1.263E-01	1.264E-01	1.26382E-01	1.26348E-01
0.22	1.684E-01	1.687E-01	1.667E-01	1.680E-01	1.6796E-01	1.682E-01	1.684E-01	1.68400E-01	1.68358E-01
0.27	2.103E-01	2.108E-01	2.044E-01	2.096E-01	2.0957E-01	2.100E-01	2.103E-01	2.10361E-01	2.10306E-01
0.33	2.522E-01	2.527E-01	2.469E-01	2.513E-01	2.5129E-01	2.517E-01	2.522E-01	2.52245E-01	2.52165E-01
0.38	2.939E-01	2.946E-01	2.872E-01	2.928E-01	2.9281E-01	2.931E-01	2.939E-01	2.94027E-01	2.93909E-01
0.44	3.355E-01	3.362E-01	3.322E-01	3.342E-01	3.342E-01	3.343E-01	3.355E-01	3.35681E-01	3.35511E-01
0.50	3.769E-01	3.778E-01	3.769E-01	3.754E-01	3.7544E-01	3.751E-01	3.769E-01	3.77183E-01	3.76946E-01
0.55	4.182E-01	4.191E-01	4.140E-01	4.147E-01	4.1468E-01	4.156E-01	4.182E-01	4.18509E-01	4.18186E-01
0.61	4.592E-01	4.601E-01	4.584E-01	4.573E-01	4.5729E-01	4.557E-01	4.592E-01	4.59635E-01	4.59203E-01
0.66	5.000E-01	5.009E-01	4.951E-01	4.978E-01	4.9783E-01	4.952E-01	4.999E-01	5.00533E-01	4.99964E-01
0.72	5.404E-01	5.414E-01	5.388E-01	5.381E-01	5.3806E-01	5.341E-01	5.404E-01	5.41166E-01	5.40427E-01
0.77	5.806E-01	5.816E-01	5.749E-01	5.779E-01	5.7794E-01	5.722E-01	5.805E-01	5.81496E-01	5.80550E-01
0.83	6.203E-01	6.213E-01	6.179E-01	6.174E-01	6.1739E-01	6.093E-01	6.202E-01	6.21472E-01	6.20275E-01
0.88	6.596E-01	6.605E-01	6.533E-01	6.564E-01	6.5635E-01	6.453E-01	6.595E-01	6.61173E-01	6.59667E-01
0.94	6.983E-01	6.992E-01	6.952E-01	6.947E-01	6.9475E-01	6.785E-01	6.972E-01	6.99207E-01	6.97325E-01

Table 7 Comparison of analytical and approximate solutions of Example 1 with results from (Kutluay et al. 1999, 2004; Hassanien et al. 2005; Wang and Lu 2005; Venkatesh et al. 2017; Xu et al. 2011) at $T = 0.1$ for $\nu = 1$

x	Analytical	(Hassanien et al. 2005) $N = 80$ $k = 0.0001$	(Wang and Lu 2005) $N = 40$ $k = 0.002$	(Venkatesh et al. 2017)	(Kutluay et al. 2004) $N = 160$ $k = 0.00001$	(Kutluay et al. 1999)EFD $N = 80$ $k = 0.00001$	(Kutluay et al. 1999)E-EFD $N = 80$ $k = 0.00001$	(Xu et al. 2011) $N = 40$ $k = 0.00001$	S-A $N = 10$ $k = 0.01$	S-B $N = 10$ $k = 0.001$
0.1	1.09538E-01	1.0954E-01	1.0955E-01	1.0954E-01	1.0965E-01	1.0952E-01	1.0955E-01	1.0953E-01	1.09540E-01	1.09538E-01
0.2	2.09792E-01	2.0979E-01	2.0982E-01	2.0978E-01	2.0998E-01	2.0975E-01	2.0981E-01	2.0978E-01	2.09791E-01	2.09792E-01
0.3	2.91896E-01	2.9190E-01	2.9195E-01	2.9190E-01	2.9213E-01	2.9184E-01	2.9192E-01	2.9189E-01	2.91881E-01	2.91896E-01
0.4	3.47924E-01	3.4792E-01	3.4757E-01	3.4792E-01	3.4818E-01	3.4786E-01	3.4795E-01	3.4791E-01	3.47883E-01	3.47924E-01
0.5	3.71577E-01	3.7158E-01	3.7129E-01	3.7157E-01	3.7185E-01	3.7151E-01	3.7161E-01	3.7156E-01	3.71501E-01	3.71577E-01
0.6	3.59046E-01	3.5905E-01	3.5883E-01	3.5903E-01	3.5932E-01	3.5898E-01	3.5907E-01	3.5903E-01	3.58934E-01	3.59045E-01
0.7	3.09905E-01	3.0991E-01	3.1020E-01	3.0990E-01	3.1017E-01	3.0985E-01	3.0993E-01	3.0989E-01	3.09774E-01	3.09905E-01
0.8	2.27817E-01	2.2782E-01	2.2809E-01	2.2781E-01	2.2805E-01	2.2778E-01	2.2783E-01	2.2781E-01	2.27697E-01	2.27817E-01
0.9	1.20687E-01	1.2069E-01	1.2085E-01	1.2069E-01	1.2083E-01	1.2067E-01	1.2070E-01	1.2068E-01	1.20613E-01	1.20687E-01
L_2									7.5636E-05	1.1233E-07
L_∞									1.3144E-04	1.5171E-07

Table 8 Comparison of analytical and approximate solutions of Example 1 with results from (Kutluay et al. 1999; Öziş et al. 2003, 2005; Jiwari and Alshomrani 2017) at the preselected times for $\nu = 0.1$

x	T	(Kutluay et al. 1999) $N = 40$ $k = 0.001$	(Öziş et al. 2003) $N = 80$ $k = 0.0001$	(Jiwari and Alshomrani 2017) $N = 40$ $k = 0.001$	(Öziş et al. 2005) $N = 80, w = 0$ $k = 0.0001$	S-A $N = 10$ $k = 0.01$	S-B $N = 10$ $k = 0.01$	Analytical
0.25	0.4	3.0891E-01	3.1429E-01	3.0892E-01	3.0889E-01	3.08907E-01	3.08639E-01	3.08894E-01
	0.6	2.4075E-01	2.4373E-01	2.4071E-01	2.4074E-01	2.40761E-01	2.40551E-01	2.40739E-01
	0.8	1.9568E-01	1.9758E-01	1.9566E-01	1.9567E-01	1.95683E-01	1.95558E-01	1.95676E-01
	1.0	1.6257E-01	1.6391E-01	1.6255E-01	1.6256E-01	1.62561E-01	1.62492E-01	1.62565E-01
0.50	0.4	5.6964E-01	5.7636E-01	5.6971E-01	5.6961E-01	5.69646E-01	5.69484E-01	5.69632E-01
	0.6	4.4721E-01	4.5169E-01	4.4730E-01	4.4719E-01	4.47222E-01	4.47056E-01	4.47206E-01
	0.8	3.5924E-01	3.6245E-01	3.5932E-01	3.5923E-01	3.59251E-01	3.59179E-01	3.59236E-01
	1.0	2.9192E-01	2.9437E-01	2.9197E-01	2.9191E-01	2.91922E-01	2.91886E-01	2.91916E-01
0.75	0.4	6.2542E-01	6.2592E-01	6.2540E-01	6.2539E-01	6.25426E-01	6.25069E-01	6.25438E-01
	0.6	4.8721E-01	4.9034E-01	4.8698E-01	4.8719E-01	4.87208E-01	4.86989E-01	4.87215E-01
	0.8	3.7392E-01	3.7713E-01	3.7389E-01	3.7390E-01	3.73901E-01	3.73959E-01	3.73922E-01
	1.0	2.8748E-01	2.9016E-01	2.8743E-01	2.8746E-01	2.87450E-01	2.87541E-01	2.87474E-01

Table 9 Comparison of analytical and approximate solutions of Example 2 with results from (Jiwari 2015) at the preselected times for $\nu = 0.1$, $\nu = 0.01$, $k = 0.001$ and $N = 10$

x	T	$\nu = 0.1$				$\nu = 0.01$			
		(Jiwari 2015)	S-A	S-B	Analytical	(Jiwari 2015)	S-A	S-B	Analytical
0.8	0.4	0.59125	0.59139	0.59145	0.59135	0.94859	0.94864	0.94136	0.94864
	0.6	0.45732	0.45743	0.45759	0.45739	0.82369	0.82372	0.81763	0.82372
	0.8	0.34555	0.34560	0.34588	0.34559	0.70194	0.70200	0.69800	0.70195
	1.0	0.26151	0.26153	0.26173	0.26154	0.60478	0.60481	0.61621	0.60478
	3.0	0.02548	0.02548	0.02548	0.02548	0.24193	0.24169	0.24408	0.24190
0.9	0.4	0.36528	0.36547	0.36600	0.36546	0.96342	0.96314	0.96622	0.96300
	0.6	0.27811	0.27824	0.27863	0.27823	0.89488	0.89496	0.89764	0.89477
	0.8	0.20449	0.20452	0.20486	0.20455	0.77663	0.77673	0.77454	0.77638
	1.0	0.15094	0.15095	0.15117	0.15097	0.67360	0.67344	0.68091	0.67312
	3.0	0.01352	0.01353	0.01353	0.01353	0.24621	0.24559	0.25383	0.24563

Table 10 Comparison of analytical and approximate solutions of Example 2 with results from (Öziş et al. 2003; Kutluay et al. 2004; Jiwari and Alshomrani 2017; Kadalbajoo and Awasthi 2006) at the preselected times for $\nu = 0.1$

x	T	(Kutluay et al. 2004) $N = 80$ $k = 0.0001$	(Öziş et al. 2003) $N = 80$ $k = 0.00001$	(Kadalbajoo and Awasthi 2006) $N = 80$ $k = 0.01$	(Jiwari and Alshomrani 2017) $k = 0.01$	S-A $N = 10$ $k = 0.01$	S-B $N = 10$ $k = 0.01$	Analytical
0.25	0.4	0.32091	0.32679	0.31743	0.31752	0.31756	0.31724	0.31752
	0.6	0.24910	0.25117	0.24609	0.24615	0.24618	0.24594	0.24614
	0.8	0.20211	0.20270	–	0.19957	0.19957	0.19942	0.19956
	1.0	0.16782	0.16780	0.16558	0.16562	0.16560	0.16552	0.16560
	3.0	0.02828	0.02804	0.02776	0.02773	0.02776	0.02775	0.02776
0.50	0.4	0.58788	0.59661	0.58446	0.58457	0.58457	0.58437	0.58454
	0.6	0.46174	0.46581	0.45791	0.45802	0.45801	0.45782	0.45798
	0.8	0.37111	0.37293	–	0.36743	0.36742	0.36733	0.36740
	1.0	0.30183	0.30253	0.29831	0.29836	0.29836	0.29831	0.29834
	3.0	0.04185	0.04155	0.04107	0.04101	0.04106	0.04106	0.04106
0.75	0.4	0.65054	0.64680	0.64558	0.64553	0.64559	0.64523	0.64562
	0.6	0.50825	0.50852	0.50261	0.50257	0.50266	0.50246	0.50268
	0.8	0.39068	0.39117	–	0.38523	0.38531	0.38539	0.38534
	1.0	0.30057	0.30066	0.29582	0.29576	0.29584	0.29594	0.29586
	3.0	0.03106	0.03081	0.03044	0.03039	0.03044	0.03043	0.03044

Table 11 Comparison of analytical and approximate solutions of Example 2 with results from (Kutluay et al. 2004; Jiwari and Alshomrani 2017; Kadalbajoo and Awasthi 2006) at the preselected times for $\nu = 0.01$.

x	T	(Kutluay et al. 2004) $N = 80$ $k = 0.0001$	(Kadalbajoo and Awasthi 2006) $N = 80$ $k = 0.01$	(Jiwari and Alshomrani 2017) $k = 0.01$	S-A $N = 10$ $k = 0.01$	S-B $N = 10$ $k = 0.001$	Analytical
0.25	0.4	0.36911	0.36273	0.36225	0.36222	0.36280	0.36226
	0.6	0.28905	0.28212	0.28203	0.28210	0.28355	0.28204
	0.8	0.23703	-	0.23045	0.23050	0.23155	0.23045
	1.0	0.20069	0.19467	0.19468	0.19474	0.20044	0.19469
	3.0	0.07865	0.07613	0.07611	0.07623	0.07926	0.07613
0.50	0.4	0.68818	0.69186	0.68368	0.68372	0.68323	0.68368
	0.6	0.55425	0.55125	0.54832	0.54833	0.54999	0.54832
	0.8	0.46011	-	0.45371	0.45374	0.45424	0.45371
	1.0	0.39206	0.38627	0.38567	0.38570	0.38510	0.38568
	3.0	0.15576	0.15218	0.15213	0.15228	0.15541	0.15218
0.75	0.4	0.92194	0.94940	0.92050	0.92051	0.91923	0.92050
	0.6	0.78676	0.79399	0.78299	0.78305	0.78095	0.78299
	0.8	0.66777	-	0.66272	0.66275	0.65932	0.66272
	1.0	0.57491	0.57170	0.56932	0.56934	0.56473	0.56932
	3.0	0.23183	0.22778	0.22771	0.22796	0.22450	0.22774

Table 12 Errors in L_2 and L_∞ norm of Example 1 with results from (Mukundan and Awasthi 2015) at the preselected times for $\nu = 1$ and $k = 0.001$

$\nu = 1$	$T = 1$		$T = 2$	
	L_2	L_∞	L_2	L_∞
BDF-1 (Mukundan and Awasthi 2015) ($N = 80$)	1.8457E−06	2.6102E−06	1.9600E−10	2.7713E−10
BDF-2 (Mukundan and Awasthi 2015) ($N = 80$)	4.6043E−08	6.5114E−08	5.0898E−12	7.1885E−12
BDF-3 (Mukundan and Awasthi 2015) ($N = 80$)	4.8173E−09	6.8127E−09	6.9285E−13	9.7552E−13
S-A ($N = 10$)	1.0606E−10	1.5259E−10	4.3378E−15	8.4698e−15
S-B ($N = 10$)	2.0665E−12	2.7515E−12	2.3005E−16	3.2289E−16

Table 13 Comparison of errors in L_2 and L_∞ norm of Example 1 with results from (Mukundan and Awasthi 2015) at the preselected times for $\nu = 0.1$ and $k = 0.01$

$\nu = 0.1$	$T = 3$		$T = 3.5$	
	L_2	L_∞	L_2	L_∞
BDF-1 (Mukundan and Awasthi 2015) ($N = 80$)	4.2130E−04	5.9753E−04	3.0024E−04	4.2509E−04
BDF-2 (Mukundan and Awasthi 2015) ($N = 80$)	7.1682E−06	1.0225E−05	5.6070E−06	7.9505E−06
BDF-3 (Mukundan and Awasthi 2015) ($N = 80$)	1.5263E−06	2.1892E−06	7.0422E−07	1.0085E−06
S-A ($N = 10$)	2.7875E−07	4.0256E−07	1.0436E−07	1.5443E−07
S-B ($N = 10$)	8.6621E−07	1.3358E−06	4.8786E−07	6.9621E−07

Table 14 Comparison of errors in L_2 and L_∞ norm of Example 1 with results from (Mukundan and Awasthi 2015) at the preselected times for $\nu = 0.01$ and $k = 0.01$

$\nu = 0.05$	$T = 5$		$T = 5.2$	
	L_2	L_∞	L_2	L_∞
BDF-1 (Mukundan and Awasthi 2015) ($N = 80$)	1.8000E−03	2.8285E−04	1.7000E−03	2.6549E−04
BDF-2 (Mukundan and Awasthi 2015) $N = 80$)	5.3991E−05	8.9093E−06	5.2835E−05	8.6442E−06
BDF-3 (Mukundan and Awasthi 2015) ($N = 80$)	4.4162E−05	6.9974E−06	3.9302E−05	6.2250E−06
S-A ($N = 10$)	1.0105E−06	1.4327E−06	9.1350E−07	1.2825E−06
S-B ($N = 10$)	8.3684E−06	1.1976E−405	7.4847E−06	1.0750E−05

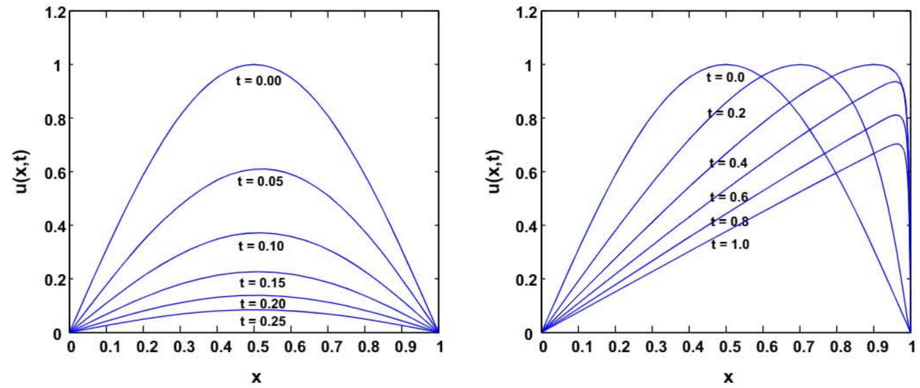

Fig. 1 Physical nature of the approximate solutions $u(x, t)$ of Example 1 for $\nu = 1$ with S-B (left) and $\nu = 1/10000$ with S-A (right) at the preselected final times t with $k = 0.001$ and $N = 500$

also those given in Kutluay et al. 2004; Öziş et al. 2003; Kadalbajoo and Awasthi 2006; Jiwari and Alshomrani 2017. As seen in Table 10, the S-A produces better results among others.

Table 11 presents the computed results of Example 2 using S-A and S-B for $\nu = 0.01$, $N = 10$. We have considered $M = -0.2e$, $b = 2.178, 2.025, 2.022, 2.014, 2.034$ for $T = 0.4, 0.6, 0.8, 1, 3$, respectively, and $k = 0.01$ in S-A and $c = 1.744$, $c = 1.727$, $c = 1.841$, $c = 1.905$, $c = 0.223$ at times $T = 0.4, 0.6, 0.8, 1, 3$, respectively, and $k = 0.001$ in S-B. The obtained results have been compared with the analytical ones and also those given in Kutluay et al. 2004; Kadalbajoo and Awasthi 2006; Jiwari and Alshomrani 2017 in Table 11. It is observed from the table that the results reported in (Jiwari and Alshomrani 2017) are better and S-A produces more accurate results than (Kutluay et al. 2004; Kadalbajoo and Awasthi 2006).

Tables 12-14 display a comparison of the L_2 and L_∞ error norms of Example 1 using S-A and S-B with the results reported in (Mukundan and Awasthi 2015) for $\nu = 1$, $k = 0.001$, $\nu = 0.1, 0.01$, $k = 0.01$ and $N = 10$. We have used the parameters in S-A as $M = -0.2e$, $b = 2.221, 2.068$ at, respectively, $T = 1, 2$ for $\nu = 1$, $M = -0.2e$, $b = 2.311, 2.314$ at, respectively, $T = 3, 3.5$ for $\nu = 0.1$, $M = -0.2e$, $b = 2.084$ at times $T = 5, 5.2$ for $\nu = 0.05$. On the other hand, we have used the parameters in S-B as $c = 1.76$ at times $T = 1, 2$ for $\nu = 1$, $c = 2.21$ at times $T = 3, 3.5$ for $\nu = 0.1$, and $c = 1.01$ at times $T = 5, 5.2$ for $\nu = 0.01$. It can be seen from the tables that S-A produces better accuracy than those reported in (Mukundan and Awasthi 2015).

We have showed the physical behavior of the Example 1 for schemes S-A and S-B in Figs. 2 and 1 for different values of ν at different times. We plot our numerical results for $x_j = j/500$, $j = 0, 1, \ldots, 500$. It can be seen from the figures that the introduced schemes, S-A and S-B, work well and show nonlinear steep characteristics of the problem for small viscosity.

5 Conclusions

Two effective numerical schemes based on the second-order time Strang splitting method together with the finite difference, Runge–Kutta, and collocation techniques using the multiquadric-radial basis function are proposed for the numerical integration of the one-

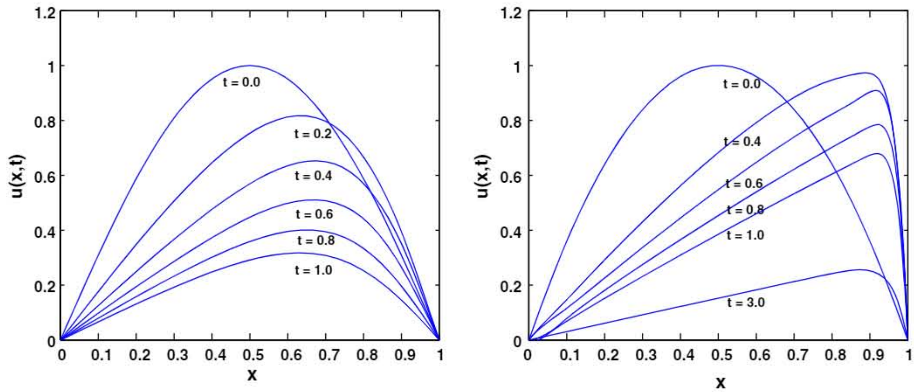

Fig. 2 Physical nature of the approximate solutions $u(x,t)$ of Example 1 for $\nu = 0.1$, $k = 0.01$ with S-A (left) and Example 2 for $\nu = 0.01$, $k = 0.001$ with S-A (right) at the different final times t with $N = 500$

dimensional Burgers' equation. Splitting methods can be used if the vector field of the problem can be split into several pieces and each of them can be solved easier than the original one. Thus, in the case of the inviscid Burgers' equation, it is not possible to apply the Strang splitting method. The presented schemes based on these combinations are utilized on two examples of the Burgers' equation and the computed numerical results are compared to the analytical ones and also existing results. It is observed that the present schemes produce generally high accurate results for different values of viscosity as compared with those in the previous methods. It is also observed that the obtained the error norms given by L_2 and L_∞ are small enough. Moreover, both schemes are able to exhibit steep characteristics of the Burgers' equation for small values of viscosity. As a conclusion, the presented method may be used for the solution of these types of nonlinear partial differential equations.

Acknowledgements The authors would like to thank the editor and the anonymous reviewers for their valuable comments and suggestions.

References

Acedo L (2006) On the gravitational instability of a set of random walkers. Europhys Lett 73:698–704

Al-Mohy AH, Higham NJ (2009) A new scaling and squaring algorithm for the matrix exponential. SIAM J Matrix Anal Appl 31:970–989

Bader B, Blanes S, Seydaoğlu M (2015) The scaling, splitting, and squaring method for the exponential of perturbed matrices. SIAM J Matrix Anal Appl 36:594–614

Bateman H (1948) Some recent researchers on the motion of fluids. Mon Weather Rev 43:171–199

Blanes S, Casas F (2016) A Concise Introduction to Geometric Numerical Integration. CRC Press, Boca Raton

Bonkile MP, Awasthi A, Lakshmi C, Mukundan V, Aswin VS (2018) A systematic literature review of Burgers' equation with recent advances. Pramana 90:69

Bouhamidi A, Hached M, Jbilou K (2014) A meshless RBF method for computing a numerical solution of unsteady Burgers'-type equations. Comput Math Appl 68:238–256

Burger JM (1948) A Mathematical Model Illustrating the Theory of Turbulence. Academic Press, Cambridge

Caldwell J, Wanless P, Cook AE (1987) Solution of Burgers' equation for large Reynolds number using finite elements with moving nodes. Appl Math Model 11:211–214

Chen R, Wu Z (2006) Applying multiquadric quasi-interpolation to solve Burgers' equation. Appl Math Comput 172:472–484

Christie I, Mitchell AR (1978) Upwinding of high order Galerkin methods in conduction-convection problems. Int J Numer Methods Eng 12:1764–1771

Çiçek Y, Korkut S (2021) Numerical Solution of Generalized Burgers-Huxley Equation by Lie-Trotter Splitting Method. Numer Anal Appl 14:90–102

Cole JD (1951) On a quasi linear parabolic equation occurring in aerodynamics. Q Appl Math 9:225–236

Cordero A, Franques A, Torregrosa JR (2015) Numerical Solution of Turbulence Problems by Solving Burgers' Equation. Algorithms 8:224–233

Fan CM, Yang CH, Gu MH (2014) Applications of the local RBF collocation method and the fictitious time integration method for Burgers' equations. Procedia Eng 79:569–574

Fletcher CAJ (1983) A comparison of finite element and finite difference solutions of the one- and two-dimensional Burgers' equations. J Comput Phys 51:159–188

Hairer E, Lubich C, Wanner G (2006) Geometric Numerical Integration. Springer, Structure-Preserving Algorithms for Ordinary Differential Equations

Haq S, Islam SU, Uddin M (2009) A mesh-free method for the numerical solution of the KdV-Burgers equation. Appl Math Model 33:3442–3449

Haq S, Hussain AU, Uddin M (2012) On the numerical solution of nonlinear Burgers'-type equations using meshless method of lines. Appl Math Comput 218:6280–6290

Hassanien IA, Salama AA, Hosham HA (2005) Fourth-order finite difference method for solving Burgers' equation. Appl Math Comput 170:781–800

Hon YC, Mao XZ (1998) An efficient numerical scheme for Burgers' equation. Appl Math Comput 95:37–50

Hopf E (1950) The partial differential equation $U_t + UU_x = \mu U_{xx}$. Commun Pure Appl Math 3:201–230

Inan B, Bahadir AR (2013) Numerical solution of the one-dimensional Burgers' equation: Implicit and fully implicit exponential finite difference methods. Pramana 81:547–556

Iskandar L, Mohsen A (1992) Some numerical experiments on the splitting of Burgers' equation. Numer Methods Partial Differ Equ 8:267–276

Islam SU, Sarler B, Vertnik R, Kosec G (2012) Radial basis function collocation method for the numerical solution of the two-dimensional transient nonlinear coupled Burgers' equations. Appl Math Model 36:1148–1160

Jiwari R (2015) A hybrid numerical scheme for the numerical solution of the Burgers' equation. Comput Phys Commun 188:59–67

Jiwari R, Alshomrani AS (2017) A new algorithm based on modified trigonometric cubic B-splines functions for nonlinear Burgers'-type equations. Int J Numer Method H 27:1638–1661

Kadalbajoo MK, Awasthi A (2006) A numerical method based on Crank-Nicolson scheme for Burgers' equation. Appl Math Comput 182:1430–1442

Kansa EJ (1990) Multiquadrics - a scattered data approximation scheme with applications to computational fluid dynamics - I. Comput Math Appl 19:127–145

Kansa EJ (1990) Multiquadrics - a scattered data approximation scheme with applications to computational fluid dynamics - II. Comput Math Appl 19:147–161

Kutluay S, Bahadir AR, Özdeş A (1999) Numerical solution of one-dimensional Burgers equation: explicit and exact-explicit finite difference methods. J Comput Appl Math 103:251–261

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

Li Q, Zheng Z, Wang S, Liu J (2012) A multilevel finite difference scheme for one-dimensional Burgers' equation derived from the lattice Boltzmann method. J Appl Math 2012:13

Liao W (2008) An implicit fourth-order compact finite difference scheme for one-dimensional Burgers' equation. Appl Math Comput 206:755–764

Mukundan V, Awasthi A (2015) Efficient numerical techniques for Burgers' equation. Appl Math Comput 262:282–297

Özer S (2019) Numerical solution of the Rosenau-KdV-RLW equation by operator splitting techniques based on B-spline collocation method. Numer Methods Partial Differ Equ 35:1928–1943

Öziş T, Aksan EN, Özdeş A (2003) A finite element approach for solution of Burgers' equation. Math Comput 139:417–428

Öziş T, Esen A, Kutluay S (2005) Numerical solution of Burgers' equation by quadratic B-spline finite-elements. Appl Math Comput 165:237–249

Sarboland M, Aminataei A (2014) On the Numerical Solution of One-Dimensional Nonlinear Nonhomogeneous Burgers' Equation. J Appl Math 2014:15

Sari M, Gurarslan G (2009) A sixth-order compact finite difference scheme to the numerical solutions of Burgers' equation. Appl Math Comput 208:475–483

Sarı M, Tunç H, Seydaoğlu M (2019) Higher order splitting approaches in analysis of the Burgers equation. Kuwait J Sci 46

Sastre J, Ibáñez J, Ruiz P, Defez E (2014) Accurate and efficient matrix exponential computation. Int J Comput Math 91:97–112

Seydaoğlu M, Erdoğan U, Öziş T (2016) Numerical solution of Burgers' equation with high order splitting methods. J Comput Appl Math 291:410–421
Trefethen LN (2000) Spectral Methods in MATLAB. Siam, Philadelphia
Venkatesh SG, Ayyaswamy SK, Balachandar SR (2017) An Approximation Method for Solving Burgers' Equation Using Legendre Wavelets. Proc Natl Acad Sci India Sect A 87:257–266
Wang W, Lu T (2005) The alternating segment difference scheme for Burgers' equation. Int J Numer Methods Fluids 49:1347–1358
Xie H, Li D (2013) A messhless method for Burgers' equation using MQ-RBF and high-order temporal approximation. Appl Math Model 37:9215–9222
Xie S, Li G, Heo S (2010) A compact finite difference method for solving Burgers' equation. J Numer Methods Fluids 62:747–764
Xie H, Li D, Li F (2013) A New Numerical Method of Particular Solutions for Inhomogeneous Burgers' Equation. Math Probl Eng 2013:7
Xie H, Zhou J, Jiang Z, Guo X (2016) Approximations for Burgers' equations with C-N scheme and RBF collocation methods. J Nonlinear Sci Appl 9:3727–3734
Xu M, Wang RH, Zhang JH, Fang Q (2011) A novel numerical scheme for solving Burgers' equation. Appl Math Comput 217:4473–4482
Yağmurlu M, Uçar Y, Çelikkaya I (2020) Higher Order Splitting Method for Numerical Solution Of Nonlinear Coupled System Viscous Burger's Equation. Konuralp J Math 8:234–243
Zhu CG, Wang RH (2009) Numerical solution of Burgers' equation by cubic B-spline quasi-interpolation. Appl Math Comput 208:260–272