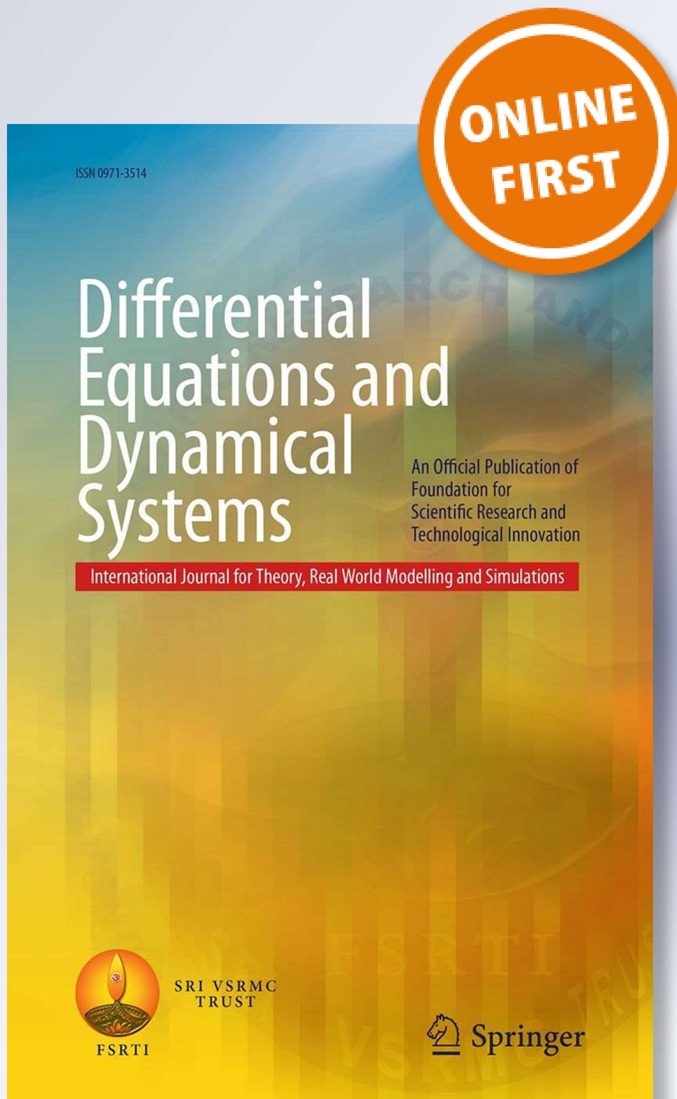
Multistep Methods for the Numerical Simulation of Two-Dimensional Burgers' Equation

Vijitha Mukundan, Ashish Awasthi & V. S. Aswin

Differential Equations and Dynamical Systems
International Journal for Theory, Real World Modelling and Simulations

ISSN 0971-3514

Differ Equ Dyn Syst
DOI 10.1007/s12591-019-00468-w



Springer

Differential Equations and Dynamical Systems
https://doi.org/10.1007/s12591-019-00468-w

ORIGINAL RESEARCH

Multistep Methods for the Numerical Simulation of Two-Dimensional Burgers' Equation

Vijitha Mukundan[1] · Ashish Awasthi[2] · V. S. Aswin[2]

Abstract

In this paper, a numerical technique is proposed to solve a two-dimensional coupled Burgers' equation. The two-dimensional Cole–Hopf transformation is applied to convert the nonlinear coupled Burgers' equation into a two-dimensional linear diffusion equation with Neumann boundary conditions. The diffusion equation with Neumann boundary conditions is semi-discretized using MOL in both x and y directions. This process yielded the system of ordinary differential equations in the time variable. Multistep methods namely backward differentiation formulas of order one, two and three are employed to solve the ode system. Efficiency and accuracy of the proposed methods are verified through numerical experiments. The proposed schemes are simple, accurate, efficient and easy to implement.

Keywords Burgers' equation · Kinematic viscosity · Method of lines · Backward differentiation formulas

Introduction

In 1915, Bateman derived the nonlinear Burgers' equation from the Navier–Stokes equation by dropping the pressure gradient term. This equation has got a time-dependent term, convection term, and diffusion term. The nonlinearity stems from the convection term and when the convective phenomena dominate the equation becomes hyperbolic in nature and when the diffusive phenomena dominate the equation becomes parabolic. Burgers' equation is used in modeling various important nonlinear physical phenomena like 'turbulence', and it has got applications in shock theory [1], viscous flow and turbulence [2], cosmology [3], gas dynamics [4], traffic flow [5], stochastic forcing and many more. Since Burgers' equation

✉ Ashish Awasthi
aawasthi@nitc.ac.in

Vijitha Mukundan
vijithamukundan29@gmail.com

V. S. Aswin
vennengotaswin@gmail.com

1 Department of Mathematics, Sacred Heart College, Chalakudi, Tissur, Kerala 680307, India

2 Department of Mathematics, National Institute of Technology Calicut, Kozhikode, Kerala 673 601, India

Published online: 04 April 2019

has got lots of physical significance in real life problems, we have an unquestionable interest in finding its numerical solution. Two-dimensional Burgers' equation is given by

$$u_t + uu_x + vu_y = \nu(u_{xx} + u_{yy}), \quad (x, y) \in D, t > 0 \tag{1}$$

$$v_t + uv_x + vv_y = \nu(v_{xx} + v_{yy}), \quad (x, y) \in D, t > 0 \tag{2}$$

subject to initial condition

$$u(x, y, 0) = f(x, y), (x, y) \in D \tag{3}$$

$$v(x, y, 0) = g(x, y), (x, y) \in D \tag{4}$$

and boundary conditions

$$u(x, y, t) = f_1(x, y, t), (x, y) \in \partial D, t > 0 \tag{5}$$

$$v(x, y, t) = g_1(x, y, t), (x, y) \in \partial D, t > 0 \tag{6}$$

where, D is the domain, ∂D is it's boundary, $u(x, y, t)$ and $v(x, y, t)$ are the velocity components to be determined, f, g, f_1, and g_1 are known functions, and ν is kinematic viscosity parameter.

In 1978, Jain and Holla [6] used cubic spline functions for solving coupled Burgers' equation in two space variables. Authors have analyzed the algorithms for their stability and convergence. A finite element method based on rectangular elements is developed by Arminjon *et al.* [7]. They checked the accuracy of the numerical methods via grid refinement. In 1983, Fletcher [8] gave exact solutions of some specified two-dimensional Burgers' equation based on two-dimensional Hopf–Cole transformation. In general, unlike the one-dimensional Burgers' equation, two-dimensional Hopf–Cole transformation cannot be used to convert two-dimensional Burgers' equation into a linear heat equation. For using two-dimensional Hopf–Cole transformation, the condition of potential symmetry must be satisfied by the two-dimensional Burgers' equation. In 1995, Esipov [9] derived and studied the physics of coupled Burgers' equation and considered it as a simple model of sedimentation or evolution of scaled volume concentrations of two kinds of particles in fluid suspensions or colloids under the effect of gravity. The study of the fluctuations of concentration during sedimentation may lead to an experimental realization of Burgers' turbulence.

In 2003, Bahadir [10] proposed a fully implicit finite difference scheme, while the non-linear system was solved by Newton's method. In [11], a higher order accurate two point compact alternating direction implicit algorithm is developed to solve the two-dimensional unsteady Burgers' equation. A comparison with fourth order Du Fort Frankel scheme is also presented which is a conditionally stable explicit scheme. Sakai et al. [12] applied the two-dimensional Hopf–Cole transformation to convert Burgers' equation into linear heat equation, and the resulting equation is solved by spectral method.

In 2011, Zhao et al. [13] proposed a numerical solution of two-dimensional Burgers' equation based on local discontinuous Galerkin (LDG) finite element method. Authors have used Hopf–Cole transformations to reduce two-dimensional Burgers' equations into a linear heat equation in two-dimension. Then the linear 2D heat equation is solved by the LDG finite element method. The numerical solution of the heat equation is used to derive the numerical solutions of Burgers' equations directly. In [14], authors have generated three sets of varied initial and boundary conditions from the general analytic solution obtained by using Hopf–Cole transformation and method of separation of variables. The Numerical solution is obtained by using the Crank-Nicolson scheme and explicit scheme. The accuracy in terms of consistency, convergence, and stability is determined by means of L^1 error. A limiter free high-order spectral volume formulation was developed to solve the Burgers' equation in

[15]. Authors have used Hopf–Cole transformation to convert nonlinear Burgers' equation to a linear diffusion equation in two-dimension. Aminikhah [16] combined Laplace transform and new homotopy perturbation methods (LTNHPM) to obtain the closed-form solutions of the coupled Burgers' equation. Author claim that the proposed method can be applied to many complicated linear and nonlinear partial differential equations without doing linearization or discretization. Yang [17] presented a finite volume element method for approximating the solution of two dimensional Burgers' equation. In this paper, the author has used the upwind technique to handle the nonlinear convection term. Also, the semi-discrete scheme and fully-discrete scheme are presented.

Srivastava et al. [18] proposed an implicit logarithmic finite difference method, for the numerical solution of two-dimensional time-dependent coupled viscous Burgers' equation on the uniform grid points. Also in [19], an implicit exponential finite difference scheme has been proposed for solving two dimensional nonlinear coupled viscous Burgers' equations with appropriate initial and boundary conditions.

In [20], a local RBF based method of approximate particular solutions for two dimensional unsteady Burgers' equations is developed. In [21], an implicit logarithmic finite difference method is introduced for the numerical solution of one dimensional coupled nonlinear Burgers' equation. The numerical scheme provides a system of nonlinear difference equations which are linearized using Newton's method. The obtained linear system is solved by Gauss elimination with the partial pivoting algorithm. A Numerical scheme for the coupled Burgers' equation based on collocation of modified Bi cubic B-spline functions was proposed by Mittal and Tripathi [22]. Mittal et al. [23] proposed a numerical method based on the properties of uniform Haar wavelets together with a collocation method and semi discretization along the space direction for solving a coupled viscous Burgers' equation.

In 2015, Mohanty et al. [24] presented a new two level implicit compact operator method for the numerical simulation of coupled viscous Burgers' equation in one spatial dimension. This scheme has an accuracy of order two in time and four in space. Authors have used three spatial grid points, and the obtained nonlinear system was solved by Newton's iterative method. Recently two new modified fourth order exponential time differencing Runge–Kutta (ETDRK) schemes in combination with a global fourth-order compact finite difference scheme (in space) for direct integration of nonlinear coupled viscous Burgers' equations is presented in [25].

Numerical Method

Many researchers have shown a keen interest in solving Burgers' equation in which initial condition is a combination of sine and cosine functions. Recently Gao [26] proposed an analytical solution of 2D Burgers' equation which can also be employed to model shock wave phenomena. We present an efficient hybrid multistep numerical method for solving the following equation.

The Two Dimensional Burgers' Equation

We consider the following two dimensional Burgers' equation

$$u_t + uu_x + vu_y = \nu(u_{xx} + u_{yy}),\ (x, y) \in (0, 1) \times (0, 1),\ t > 0 \tag{7}$$

$$v_t + uv_x + vv_y = \nu(v_{xx} + v_{yy}),\ (x, y) \in (0, 1) \times (0, 1),\ t > 0 \tag{8}$$

subject to the initial condition

$$u(x, y, 0) = \sin(\pi x)\cos(\pi y),\ (x, y) \in (0, 1) \times (0, 1),\ t > 0, \tag{9a}$$

$$v(x, y, 0) = \cos(\pi x)\sin(\pi y),\ (x, y) \in (0, 1) \times (0, 1),\ t > 0, \tag{9b}$$

and boundary conditions

$$u(0, y, t) = u(1, y, t) = 0,\ t > 0 \tag{9c}$$

$$v(x, 0, t) = v(x, 1, t) = 0,\ t > 0 \tag{9d}$$

Here the potential symmetry condition $\frac{\partial u}{\partial y} = \frac{\partial v}{\partial x}$ is satisfied. Hence the two-dimensional Cole–Hopf transformation can be used to reduce Burgers' equation into the two-dimensional diffusion equation.

Consider a function $\phi(x, y, t)$ such that

$$u(x, y, t) = -2\nu \frac{\phi_x}{\phi} \tag{10}$$

and

$$v(x, y, t) = -2\nu \frac{\phi_y}{\phi} \tag{11}$$

These are two dimensional Cole–Hopf transformations. Substituting these transformations in 2D Burgers' equation (7) and after simplification we get,

$$\phi_t = \nu(\phi_{xx} + \phi_{yy}) + \beta_1(y, t)\phi \tag{12}$$

where $\beta_1(y, t)$ is an arbitrary function of y and t alone.

Substituting these transformations in 2D Burgers' equation (8) and after simplification we get,

$$\phi_t = \nu(\phi_{xx} + \phi_{yy}) + \beta_2(x, t)\phi \tag{13}$$

where $\beta_2(x, t)$ is an arbitrary function of x and t alone.

Hence from R.H.S of Eqs. (12) and (13), we conclude that the arbitrary function is a function of time alone. Hence we have,

$$\phi_t = \nu(\phi_{xx} + \phi_{yy}) + \beta(t)\phi \tag{14}$$

where, $\beta(t)$ is an arbitrary function of t alone. Hence $\beta(t)$ can assume any value but the values of $u(x, y, t)$ and $v(x, y, t)$ does not depend on the choice of $\beta(t)$ is implied by the following theorem.

Proposition Let $\phi(x, y, t)$ be the solution of Eq. (14), $u(x, y, t)$ and $v(x, y, t)$ are defined in Eqs. (10) and (11) respectively, then $u(x, y, t)$ and $v(x, y, t)$ are independent of $\beta(t)$.

Since $u(x, y, t)$ and $v(x, y, t)$ are independent of $\beta(t)$ without loss of generality we can take $\beta(t) = 0$. Hence the Eq. (14) reduces to

$$\phi_t = \nu(\phi_{xx} + \phi_{yy}) \tag{15}$$

with the initial condition

$$\phi(x, y, 0) = \exp\left(\frac{\cos(\pi x)\cos(\pi y) - 1}{2\nu\pi}\right), \tag{16}$$

and Neumann boundary conditions

$$\phi_x(0, y, t) = \phi_x(1, y, t) = \phi_y(x, 0, t) = \phi_y(x, 1, t) = 0. \tag{17}$$

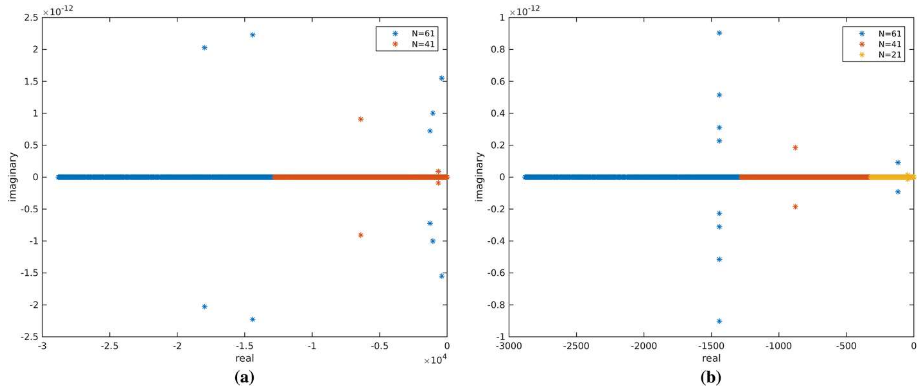


Fig. 1 Eigen values of the matrix A for different number of nodal points, **a** $\nu = 1$, **b** $\nu = 0.1$

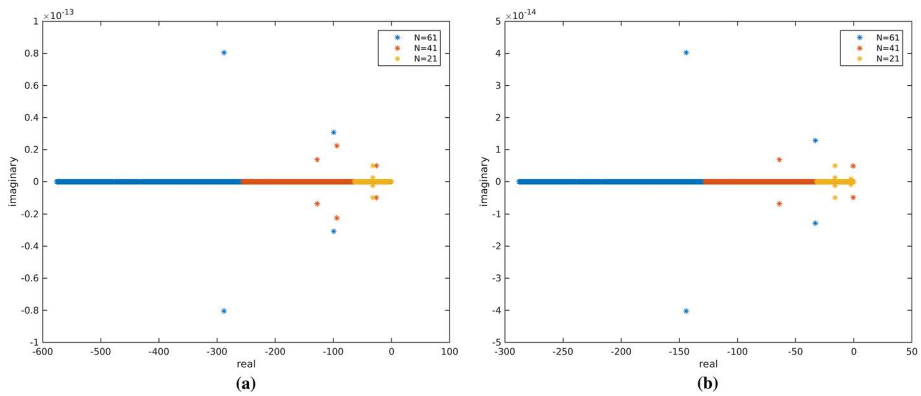


Fig. 2 Eigen values of the matrix A for different number of nodal points, **a** $\nu = 0.02$ **b** $\nu = 0.01$

Next, we propose a hybrid technique to solve this two-dimensional diffusion equation. Our scheme comprises of method of lines technique and multistep method.

Spatial Discretization

The technique of the method of lines is applied to solve the two-dimensional diffusion equation. We consider the domain D as a square domain $[0, 1] \times [0, 1]$. This domain is divided into $(N+1) \times (N+1)$ uniform grids described by the set of nodes (x_i, y_j) where $x_i = ih_x$ and $y_j = jh_y$ for $i, j = 0, 1, 2, \ldots N$. Here, h_x and h_y denote grid size along x-direction and y-direction respectively, with $h_x = h_y = 1/(N)$. Second order derivatives appearing in diffusion equation are approximated using second order central finite difference operators, as given below.

$$\left(\frac{\partial^2 \phi}{\partial x^2}\right)_{i,j}(t) = \frac{\phi_{i+1,j}(t) - 2\phi_{i,j}(t) + \phi_{i-1,j}(t)}{h_x^2},\ i, j = 0, 1, 2, \ldots N.$$
$$\left(\frac{\partial^2 \phi}{\partial y^2}\right)_{i,j}(t) = \frac{\phi_{i,j+1}(t) - 2\phi_{i,j}(t) + \phi_{i,j-1}(t)}{h_y^2},\ i, j = 0, 1, 2, \ldots N.$$

Table 1 The numerical results of u, using $\Delta x = 0.025$, $\Delta y = 0.025$, $\nu = 1$

(x,y)	(0.25,0.25)	(0.5,0.25)	(0.75,0.25)	(0.25,0.5)	(0.75,0.5)	(0.25,0.75)	(0.5,0.75)	(0.75,0.75)
T = 0.25								
BDF1	3.776e−03	5.339e−03	3.775e−03	2.523e−06	−2.523e−06	−3.775e−03	−5.339e−03	−3.776e−03
BDF2	3.597e−03	5.087e−03	3.597e−03	2.080e−06	−2.080e−06	−3.597e−03	−5.087e−03	−3.597e−03
BDF3	3.599e−03	5.090e−03	3.599e−03	2.090e−06	−2.090e−06	−3.599e−03	−5.090e−03	−3.599e−03
Exact	3.593e−03	5.081e−03	3.593e−03	2.055e−06	−2.055e−06	−3.593e−03	−5.081e−03	−3.593e−03
T = 0.50								
BDF1	2.856e−05	4.039e−05	2.856e−05	1.608e−10	−1.609e−10	−2.856e−05	−4.039e−05	−2.856e−05
BDF2	2.592e−05	3.666e−05	2.592e−05	1.093e−10	−1.092e−10	−2.592e−05	−3.666e−05	−2.592e−05
BDF3	2.595e−05	3.670e−05	2.595e−05	1.103e−10	−1.104e−10	−2.595e−05	−3.670e−05	−2.595e−05
Exact	2.584e−05	3.654e−05	2.584e−05	1.063e−10	−1.063e−10	−2.584e−05	−3.654e−05	−2.584e−05
T = 0.75								
BDF1	2.161e−07	3.056e−07	2.161e−07	−1.038e−14	−4.672e−14	−2.161e−07	−3.056e−07	−2.161e−07
BDF2	1.868e−07	2.641e−07	1.868e−07	2.595e−14	4.152e−14	−1.868e−07	−2.641e−07	−1.868e−07
BDF3	1.871e−07	2.646e−07	1.871e−07	−4.152e−14	−5.191e−14	−1.871e−07	−2.646e−07	−1.871e−07
Exact	1.858e−07	2.628e−07	1.858e−07	5.497e−15	−5.497e−15	−1.858e−07	−2.628e−07	−1.858e−07
T = 1.00								
BDF1	1.634e−09	2.311e−09	1.635e−09	−6.748e−14	−5.191e−15	−1.634e−09	−2.311e−09	−1.634e−09
BDF2	1.346e−09	1.903e−09	1.346e−09	2.076e−14	0.000e+00	−1.346e−09	−1.903e−09	−1.346e−09
BDF3	1.349e−09	1.908e−09	1.349e−09	1.038e−14	−7.786e−14	−1.349e−09	−1.908e−09	−1.349e−09
Exact	1.337e−09	1.890e−09	1.337e−09	2.843e−19	−2.843e−19	−1.337e−09	−1.890e−09	−1.337e−09

Table 2 The numerical results of v, using $\Delta x = 0.025$, $\Delta y = 0.025$, $\nu = 1$

(x,y)	(0.25,0.25)	(0.5,0.25)	(0.75,0.25)	(0.25,0.5)	(0.75,0.5)	(0.25,0.75)	(0.5,0.75)	(0.75,0.75)
T = 0.25								
BDF1	3.776e−03	2.523e−06	−3.775e−03	5.339e−03	−5.339e−03	3.775e−03	−2.523e−06	−3.776e−03
BDF2	3.597e−03	2.080e−06	−3.597e−03	5.087e−03	−5.087e−03	3.597e−03	−2.080e−06	−3.597e−03
BDF3	3.599e−03	2.090e−06	−3.599e−03	5.090e−03	−5.090e−03	3.599e−03	−2.090e−06	−3.599e−03
Exact	3.593e−03	2.055e−06	−3.593e−03	5.081e−03	−5.081e−03	3.593e−03	−2.055e−06	−3.593e−03
T = 0.50								
BDF1	2.856e−05	1.609e−10	−2.856e−05	4.039e−05	−4.039e−05	2.856e−05	−1.608e−10	−2.856e−05
BDF2	2.592e−05	1.092e−10	−2.592e−05	3.666e−05	−3.666e−05	2.592e−05	−1.093e−10	−2.592e−05
BDF3	2.595e−05	1.101e−10	−2.595e−05	3.670e−05	−3.670e−05	2.595e−05	−1.103e−10	−2.595e−05
Exact	2.584e−05	1.063e−10	−2.584e−05	3.654e−05	−3.654e−05	2.584e−05	−1.063e−10	−2.584e−05
T = 0.75								
BDF1	2.161e−07	3.633e−14	−2.161e−07	3.056e−07	−3.056e−07	2.161e−07	5.191e−15	−2.161e−07
BDF2	1.868e−07	−5.710e−14	−1.868e−07	2.641e−07	−2.641e−07	1.868e−07	−1.661e−13	−1.868e−07
BDF3	1.871e−07	−2.284e−13	−1.871e−07	2.646e−07	−2.646e−07	1.871e−07	9.343e−14	−1.871e−07
Exact	1.858e−07	5.497e−15	−1.858e−07	2.628e−07	−2.628e−07	1.858e−07	−5.497e−15	−1.858e−07
T = 1.00								
BDF1	1.635e−09	2.076e−14	−1.635e−09	2.312e−09	−2.311e−09	1.635e−09	3.633e−14	−1.634e−09
BDF2	1.346e−09	−3.633e−14	−1.346e−09	1.903e−09	−1.903e−09	1.346e−09	−1.142e−13	−1.346e−09
BDF3	1.349e−09	−1.557e−13	−1.349e−09	1.908e−09	−1.908e−09	1.349e−09	1.038e−13	−1.349e−09
Exact	1.337e−09	2.843e−19	−1.337e−09	1.890e−09	−1.890e−09	1.337e−09	−2.843e−19	−1.337e−09

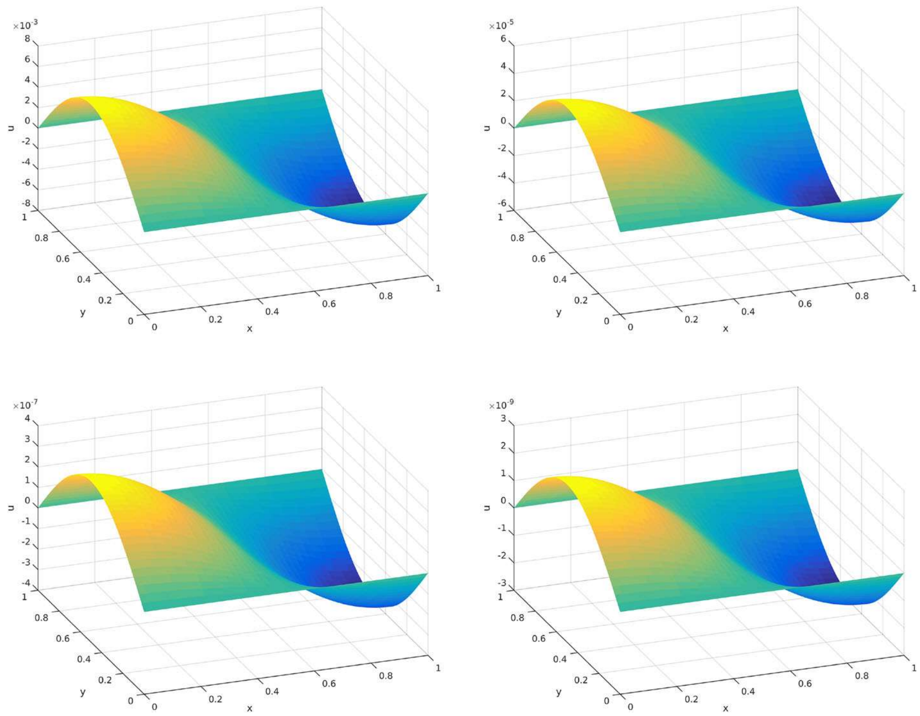


Fig. 3 Numerical solutions for u at $\Delta x = 0.025$, $\Delta y = 0.025$, $\Delta t = 0.001$, $\nu = 1$, u(x,y,t=0.25)(top left), u(x,y,t=0.5) (top right), u(x,y,t=0.75)(bottom left), u(x,y,t=1) (bottom right)

substituting in Eq. (15), we get

$$\frac{d\phi_{i,j}(t)}{dt} = \nu \left(\frac{\phi_{i+1,j}(t) + \phi_{i-1,j}(t)}{h_x^2} + \frac{\phi_{i,j+1}(t) + \phi_{i,j-1}(t)}{h_y^2} - 2\phi_{i,j} \left(\frac{1}{h_x^2} + \frac{1}{h_y^2} \right) \right)$$

with the initial condition

$$\phi_{i,j}(0) = \phi_0(x_i, y_j),\ i,\ j = 0, 1, 2, \ldots N$$

where, $\phi_{i,j}(t)$ is the discrete approximation of $\phi(x, y, t)$ at the uniform mesh (ih_x, jh_y).

The above system of ordinary differential equations in time variable can be written in matrix form

$$\begin{aligned} \frac{d\mathbf{\Phi}}{dt} &= \mathbf{F}(\mathbf{\Phi}, \mathbf{t}), \\ \mathbf{\Phi}(0) &= \mathbf{\Phi_0} \end{aligned} \tag{18}$$

where, $\mathbf{\Phi}(t) = (\phi_{00}, \phi_{01}, \ldots, \phi_{0N}, \phi_{10}, \phi_{11}, \ldots, \phi_{1N}, \ldots, \phi_{N0}, \phi_{N1}, \ldots, \phi_{NN})$.

Here, $\mathbf{F}$ is a function of $\mathbf{\Phi}$ with elements $f_{i,j}$ which are defined as:

$$\begin{aligned} f_{i,j} &= p_1(\phi_{i+1,j}(t) + \phi_{i-1,j}(t)) + p_2(\phi_{i,j+1}(t) \\ &\quad + \phi_{i,j-1}(t)) - 2p\phi_{i,j},\ i,\ j = 0, 1, 2, \ldots N \\ p_1 &= \frac{\nu}{(h_x)^2},\ p_2 = \frac{\nu}{(h_y)^2},\ p = p_1 + p_2 \end{aligned} \tag{19}$$

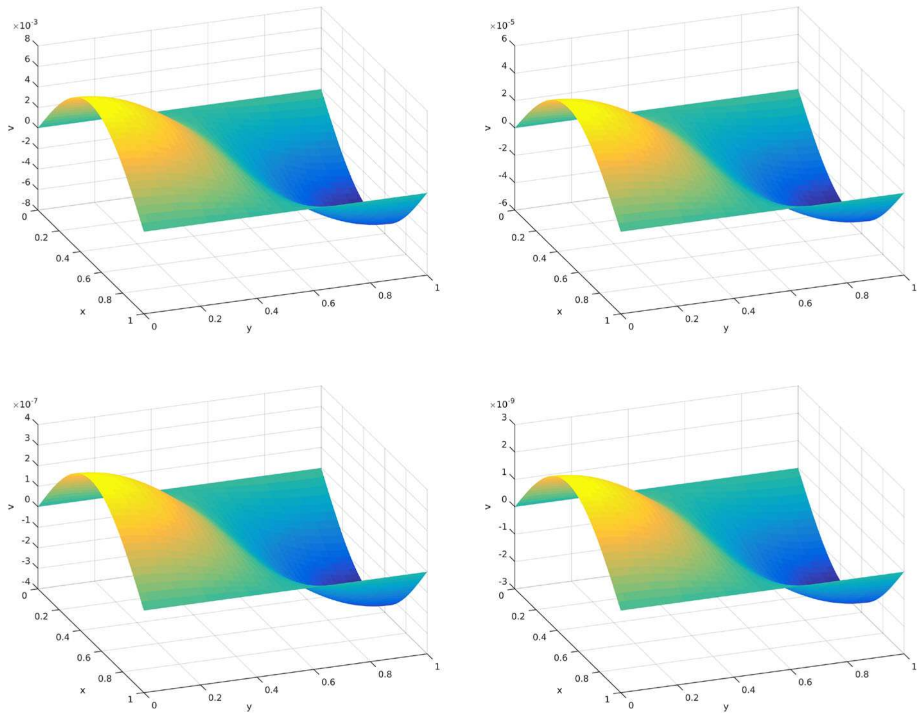


Fig. 4 Numerical solutions for v at $\Delta x = 0.025$, $\Delta y = 0.025$, $\Delta t = 0.001$, $\nu = 1$, v(x,y,t=0.25)(top left), v(x,y,t=0.5) (top right), v(x,y,t=0.75)(bottom left), v(x,y,t=1) (bottom right)

Since **F** is a linear function of $\mathbf{\Phi}$, the system (18) can be written as

$$\frac{d\mathbf{\Phi}}{dt} = A\mathbf{\Phi}, \tag{20}$$

where, $\mathbf{\Phi} = (\phi_{00}, \phi_{01}, \ldots, \phi_{0N}, \phi_{10}, \phi_{11}, \ldots, \phi_{1N}, \ldots, \phi_{N0}, \phi_{N1}, \ldots, \phi_{NN})^T$.

The matrix A is given by

$$A = \begin{bmatrix} A_1 & B_1 & 0 \\ B_1/2 & A_1 & B_1/2 \\ 0 & B_1 & A_1 \end{bmatrix},$$

where,

$$A_1 = \begin{bmatrix} -2p & 2p_2 & & & \\ p_2 & -2p & p_2 & & \\ & \ddots & \ddots & \ddots & \\ & & p_2 & -2p & p_2 \\ & & & 2p_2 & -2p \end{bmatrix},$$

Table 3 The numerical results of u, using $\Delta x = 0.025$, $\Delta y = 0.025$, $\nu = 0.1$

(x,y)	(0.25,0.25)	(0.5,0.25)	(0.75,0.25)	(0.25,0.5)	(0.75,0.5)	(0.25,0.75)	(0.5,0.75)	(0.75,0.75)
T = 0.25								
BDF1	2.886e−01	4.288e−01	2.888e−01	8.634e−02	−8.634e−02	−2.888e−01	−4.288e−01	−2.886e−01
BDF2	2.884e−01	4.287e−01	2.888e−01	8.647e−02	−8.647e−02	−2.888e−01	−4.287e−01	−2.884e−01
BDF3	2.884e−01	4.287e−01	2.888e−01	8.647e−02	−8.647e−02	−2.888e−01	−4.287e−01	−2.884e−01
Exact	2.887e−01	4.290e−01	2.887e−01	8.674e−02	−8.674e−02	−2.887e−01	−4.290e−01	−2.887e−01
T = 0.50								
BDF1	1.736e−01	2.531e−01	1.734e−01	4.236e−02	−4.236e−02	−1.734e−01	−2.531e−01	−1.736e−01
BDF2	1.734e−01	2.528e−01	1.734e−01	4.227e−02	−4.227e−02	−1.734e−01	−2.528e−01	−1.734e−01
BDF3	1.734e−01	2.528e−01	1.734e−01	4.227e−02	−4.227e−02	−1.734e−01	−2.528e−01	−1.734e−01
Exact	1.735e−01	2.529e−01	1.735e−01	4.229e−02	−4.229e−02	−1.735e−01	−2.529e−01	−1.735e−01
T = 0.75								
BDF1	1.059e−01	1.516e−01	1.057e−01	1.714e−02	−1.714e−02	−1.057e−01	−1.516e−01	−1.059e−01
BDF2	1.057e−01	1.514e−01	1.056e−01	1.706e−02	−1.706e−02	−1.056e−01	−1.514e−01	−1.057e−01
BDF3	1.057e−01	1.514e−01	1.056e−01	1.706e−02	−1.706e−02	−1.056e−01	−1.514e−01	−1.057e−01
Exact	1.057e−01	1.514e−01	1.057e−01	1.703e−02	−1.703e−02	−1.057e−01	−1.514e−01	−1.057e−01
T = 1.00								
BDF1	6.465e−02	9.185e−02	6.456e−02	6.591e−03	−6.591e−03	−6.456e−02	−9.185e−02	−6.465e−02
BDF2	6.450e−02	9.167e−02	6.446e−02	6.543e−03	−6.543e−03	−6.446e−02	−9.167e−02	−6.450e−02
BDF3	6.450e−02	9.167e−02	6.446e−02	6.543e−03	−6.543e−03	−6.446e−02	−9.167e−02	−6.450e−02
Exact	6.448e−02	9.166e−02	6.448e−02	6.518e−03	−6.518e−03	−6.448e−02	−9.166e−02	−6.448e−02

Table 4 The numerical results of v, using $\Delta x = 0.025$, $\Delta y = 0.025$, $\nu = 0.1$

(x,y)	(0.25,0.25)	(0.5,0.25)	(0.75,0.25)	(0.25,0.5)	(0.75,0.5)	(0.25,0.75)	(0.5,0.75)	(0.75,0.75)
T = 0.25								
BDF1	2.886e−01	8.634e−02	−2.888e−01	4.288e−01	−4.288e−01	2.888e−01	−8.634e−02	−2.886e−01
BDF2	2.884e−01	8.647e−02	−2.888e−01	4.287e−01	−4.287e−01	2.888e−01	−8.647e−02	−2.884e−01
BDF3	2.884e−01	8.647e−02	−2.888e−01	4.287e−01	−4.287e−01	2.888e−01	−8.647e−02	−2.884e−01
Exact	2.887e−01	8.674e−02	−2.887e−01	4.290e−01	−4.290e−01	2.887e−01	−8.674e−02	−2.887e−01
T = 0.50								
BDF1	1.736e−01	4.236e−02	−1.734e−01	2.531e−01	−2.531e−01	1.734e−01	−4.236e−02	−1.736e−01
BDF2	1.734e−01	4.227e−02	−1.734e−01	2.528e−01	−2.528e−01	1.734e−01	−4.227e−02	−1.734e−01
BDF3	1.734e−01	4.227e−02	−1.734e−01	2.528e−01	−2.528e−01	1.734e−01	−4.227e−02	−1.734e−01
Exact	1.735e−01	4.229e−02	−1.735e−01	2.529e−01	−2.529e−01	1.735e−01	−4.229e−02	−1.735e−01
T = 0.75								
BDF1	1.059e−01	1.714e−02	−1.057e−01	1.516e−01	−1.516e−01	1.057e−01	−1.714e−02	−1.059e−01
BDF2	1.057e−01	1.706e−02	−1.056e−01	1.514e−01	−1.514e−01	1.056e−01	−1.706e−02	−1.057e−01
BDF3	1.057e−01	1.706e−02	−1.056e−01	1.514e−01	−1.514e−01	1.056e−01	−1.706e−02	−1.057e−01
Exact	1.057e−01	1.703e−02	−1.057e−01	1.514e−01	−1.514e−01	1.057e−01	−1.703e−02	−1.057e−01
T = 1.00								
BDF1	6.465e−02	6.591e−03	−6.456e−02	9.185e−02	−9.185e−02	6.456e−02	−6.591e−03	−6.465e−02
BDF2	6.450e−02	6.543e−03	−6.446e−02	9.167e−02	−9.167e−02	6.446e−02	−6.543e−03	−6.450e−02
BDF3	6.450e−02	6.543e−03	−6.446e−02	9.167e−02	−9.167e−02	6.446e−02	−6.543e−03	−6.450e−02
Exact	6.448e−02	6.518e−03	−6.448e−02	9.166e−02	−9.166e−02	6.448e−02	−6.518e−03	−6.448e−02

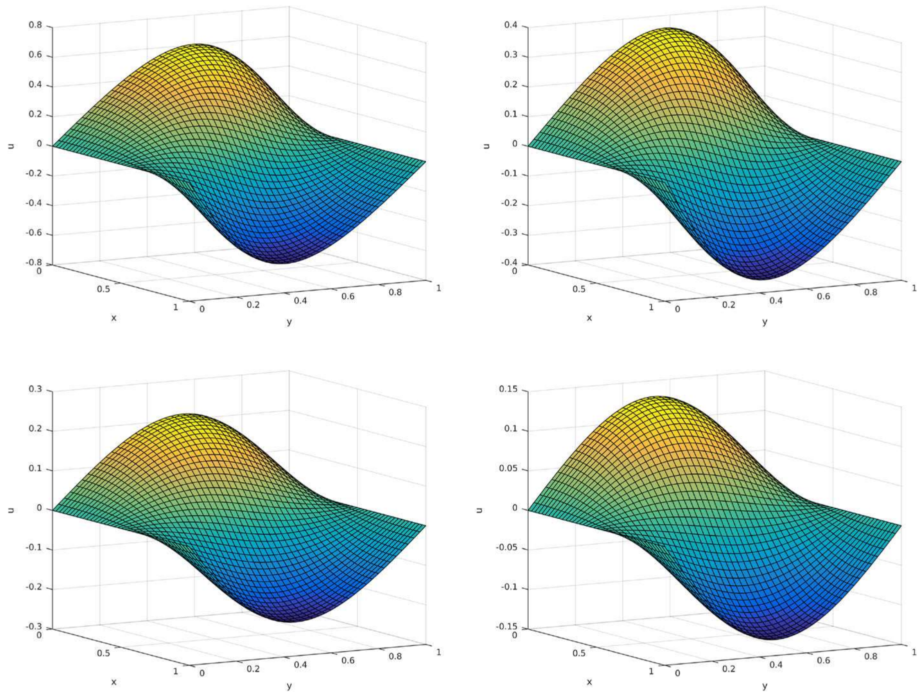


Fig. 5 Numerical solutions for u at $\Delta x = 0.025$, $\Delta y = 0.025$, $\Delta t = 0.001$, $\nu = 0.1$, u(x,y,t=0.25)(top left), u(x,y,t=0.5) (top right), u(x,y,t=0.75)(bottom left), u(x,y,t=1) (bottom right)

and

$$B_1 = \begin{bmatrix} 2p_1 & & & & \\ & 2p_1 & & & \\ & & \ddots & & \\ & & & 2p_1 & \\ & & & & 2p_1 \end{bmatrix}.$$

Here, 0 is the null matrix, A_1 and B_1 are matrices of order $(N+1) \times (N+1)$. Next, we will integrate the system (20) with respect to the time variable using backward differentiation formulas of order one, two and three.

Temporal Discretization

For temporal discretization, we use backward differentiation formulas of order one two and three. We divide the time interval $[0, T]$ into M equal subintervals $0 = t_0 \leq t_1 \leq \ldots \leq t_M = T$ with $\Delta t = T/M$, i.e $t_n = n\Delta t$, $n = 0, 1, 2, \ldots, M$.

Backward Differentiation Formula of Order (BDF-1)

$$\mathbf{\Phi^{n+1}} = \mathbf{\Phi^{n}} + A\mathbf{\Phi^{n+1}},\ n = 0, 1, \ldots, M-1. \tag{21}$$

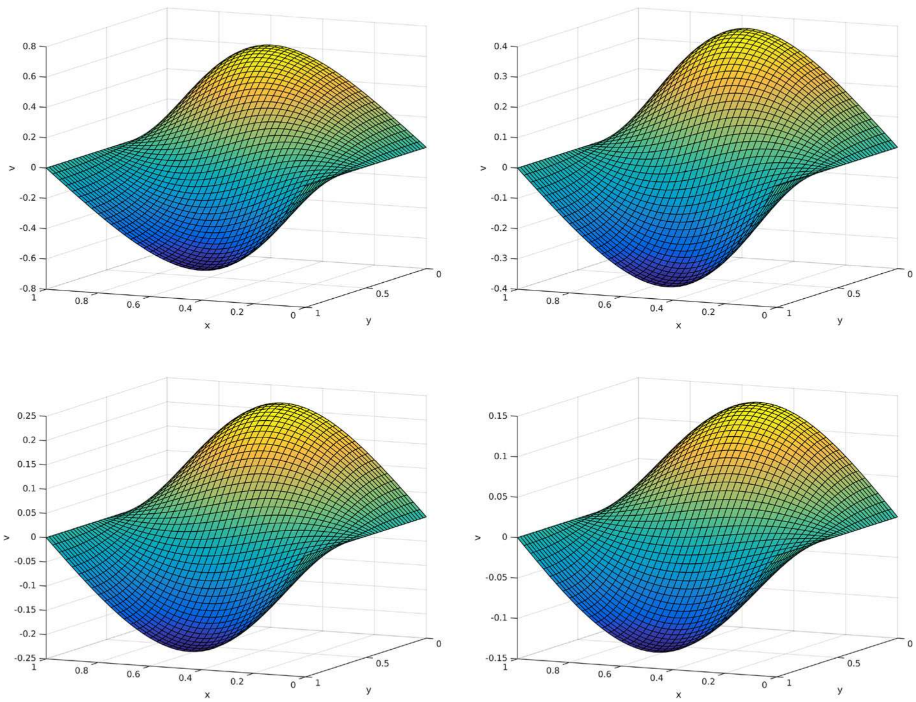


Fig. 6 Numerical solutions for v at $\Delta x = 0.025$, $\Delta y = 0.025$, $\Delta t = 0.001$, $\nu = 0.1$, v(x,y,t=0.25)(top left), v(x,y,t=0.5) (top right), v(x,y,t=0.75)(bottom left), v(x,y,t=1) (bottom right)

$\mathbf{\Phi^0}$ is the initial condition. It is also known as backward Euler Formula

Backward Differentiation Formula of Order Two (BDF-2)

$$\mathbf{\Phi^{n+1}} = \frac{4}{3}\mathbf{\Phi^{n}} - \frac{1}{3}\mathbf{\Phi^{n-1}} + \frac{2}{3}A\mathbf{\Phi^{n+1}},\ n = 1, \ldots, M-1. \tag{22}$$

Here, the solution at first time level i.e. $\mathbf{\Phi^1}$ is obtained from BDF-1.

Backward Differentiation Formula of Order Three (BDF-3)

$$\mathbf{\Phi^{n+1}} = \frac{18}{11}\mathbf{\Phi^{n}} - \frac{9}{11}\mathbf{\Phi^{n-1}} + \frac{2}{11}\mathbf{\Phi^{n-2}} + \frac{6}{11}A\mathbf{\Phi^{n+1}},\ n = 2, \ldots, M-1. \tag{23}$$

The solution at first time level $\mathbf{\Phi^1}$ is obtained from BDF-1 and second time level $\mathbf{\Phi^2}$ is obtained from BDF-2.

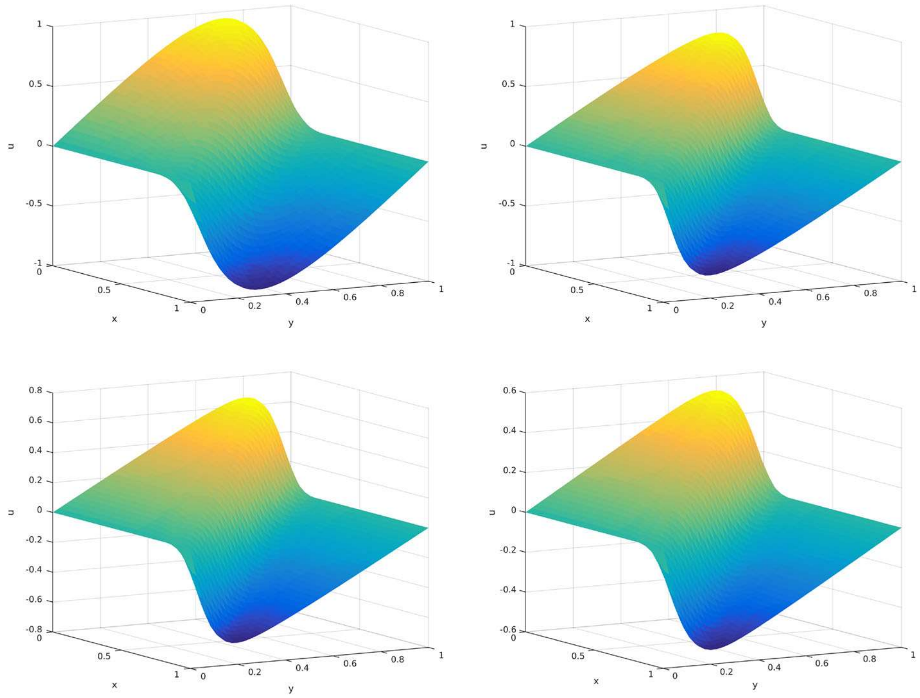


Fig. 7 Numerical solutions for u at $\Delta x = 0.025$, $\Delta y = 0.025$, $\Delta t = 0.001$, $\nu = 0.02$, u(x,y,t=0.25)(top left), u(x,y,t=0.5) (top right), u(x,y,t=0.75)(bottom left), u(x,y,t=1) (bottom right)

Numerical Solution of Two-Dimensional Burgers' Equation

The solution of Burgers' equation can be derived from the solution of two-dimensional diffusion equation using the inverse Cole–Hopf transformation. The numerical solution of the two-dimensional Burgers' Equation in terms of the solution of fully discretized diffusion equation and Cole–Hopf transformation equations (10, 11) is given by

$$\begin{aligned} u_{i,j}^{n} &= -(2\nu)\left\{\frac{\phi_{i+1,j}^{n}-\phi_{i-1,j}^{n}}{2h\phi_{i,j}^{n}}\right\} \\ &= -\left(\frac{\nu}{h}\right)\left\{\frac{\phi_{i+1,j}^{n}-\phi_{i-1,j}^{n}}{\phi_{i,j}^{n}}\right\},\ i,\ j=0,1,\ldots,N,\ n=1,\ldots,M \end{aligned} \tag{24}$$

$$\begin{aligned} v_{i,j}^{n} &= -(2\nu)\left\{\frac{\phi_{i,j+1}^{n}-\phi_{i,j-1}^{n}}{2h\phi_{i,j}^{n}}\right\} \\ &= -\left(\frac{\nu}{h}\right)\left\{\frac{\phi_{i,j+1}^{n}-\phi_{i,j-1}^{n}}{\phi_{i,j}^{n}}\right\},\ i,\ j=0,1,\ldots,N,\ n=1,\ldots,M \end{aligned} \tag{25}$$

In the next section we will analyze the stability of the proposed numerical method.

Table 5 The numerical results of u, using $\Delta x = 0.025$, $\Delta y = 0.025$, $\nu = 0.02$

(x,y)	(0.25,0.25)	(0.5,0.25)	(0.75,0.25)	(0.25,0.5)	(0.75,0.5)	(0.25,0.75)	(0.5,0.75)	(0.75,0.75)
T = 0.25								
BDF1	3.826e−01	6.638e-01	3.906e−01	2.392e−01	−2.392e−01	−3.906e−01	−6.638e−01	−3.826e−01
BDF2	3.828e−01	6.646e−01	3.914e−01	2.392e−01	−2.392e−01	−3.914e−01	−6.646e−01	−3.828e−01
BDF3	3.828e−01	6.646e−01	3.914e−01	2.392e−01	−2.392e−01	−3.914e−01	−6.646e−01	−3.828e−01
Exact	3.835e−01	6.543e−01	3.835e−01	2.378e−01	−2.378e−01	−3.835e−01	−6.543e−01	−3.835e−01
T = 0.50								
BDF1	2.836e−01	5.374e−01	2.878e−01	2.419e−01	−2.419e−01	−2.878e−01	−5.374e−01	−2.836e−01
BDF2	2.836e−01	5.381e−01	2.886e−01	2.423e−01	−2.423e−01	−2.886e−01	−5.381e−01	−2.836e−01
BDF3	2.836e−01	5.381e−01	2.886e−01	2.423e−01	−2.423e−01	−2.886e−01	−5.381e−01	−2.836e−01
Exact	2.840e−01	5.354e−01	2.840e−01	2.425e−01	−2.425e−01	−2.840e−01	−5.354e−01	−2.840e−01
T = 0.75								
BDF1	2.223e−01	4.299e−01	2.241e−01	2.030e−01	−2.030e−01	−2.241e−01	−4.299e−01	−2.223e−01
BDF2	2.222e−01	4.303e−01	2.246e−01	2.033e−01	−2.033e−01	−2.246e−01	−4.303e−01	−2.222e−01
BDF3	2.222e−01	4.303e−01	2.246e−01	2.033e−01	−2.033e−01	−2.246e−01	−4.303e−01	−2.222e−01
Exact	2.225e−01	4.296e−01	2.225e−01	2.036e−01	−2.036e−01	−2.225e−01	−4.296e−01	−2.225e−01
T = 1.00								
BDF1	1.821e−01	3.512e−01	1.829e−01	1.669e−01	−1.669e−01	−1.829e−01	−3.512e−01	−1.821e−01
BDF2	1.821e−01	3.514e−01	1.832e−01	1.670e−01	−1.670e−01	−1.832e−01	−3.514e−01	−1.821e−01
BDF3	1.821e−01	3.514e−01	1.832e−01	1.670e−01	−1.670e−01	−1.832e−01	−3.514e−01	−1.821e−01
Exact	1.822e−01	3.513e−01	1.822e−01	1.673e−01	−1.673e−01	−1.822e−01	−3.513e−01	−1.822e−01

Table 6 The numerical results of v, using $\Delta x = 0.025$, $\Delta y = 0.025$, $\nu = 0.02$

(x,y)	(0.25,0.25)	(0.5,0.25)	(0.75,0.25)	(0.25,0.5)	(0.75,0.5)	(0.25,0.75)	(0.5,0.75)	(0.75,0.75)
T = 0.25								
BDF1	3.826e−01	2.392e−01	−3.906e−01	6.638e−01	−6.638e−01	3.906e−01	−2.392e−01	−3.826e−01
BDF2	3.828e−01	2.392e−01	−3.914e−01	6.646e−01	−6.646e−01	3.914e−01	−2.392e−01	−3.828e−01
BDF3	3.828e−01	2.392e−01	−3.914e−01	6.646e−01	−6.646e−01	3.914e−01	−2.392e−01	−3.828e−01
Exact	3.835e−01	2.378e−01	−3.835e−01	6.543e−01	−6.543e−01	3.835e−01	−2.378e−01	−3.835e−01
T = 0.50								
BDF1	2.836e−01	2.419e−01	−2.878e−01	5.374e−01	−5.374e−01	2.878e−01	−2.419e−01	−2.836e−01
BDF2	2.836e−01	2.423e−01	−2.886e−01	5.381e−01	−5.381e−01	2.886e−01	−2.423e−01	−2.836e−01
BDF3	2.836e−01	2.423e−01	−2.886e−01	5.381e−01	−5.381e−01	2.886e−01	−2.423e−01	−2.836e−01
Exact	2.840e−01	2.425e−01	−2.840e−01	5.354e−01	−5.354e−01	2.840e−01	−2.425e−01	−2.840e−01
T = 0.75								
BDF1	2.223e−01	2.030e−01	−2.241e−01	4.299e−01	−4.299e−01	2.241e−01	−2.030e−01	−2.223e−01
BDF2	2.222e−01	2.033e−01	−2.246e−01	4.303e−01	−4.303e−01	2.246e−01	−2.033e−01	−2.222e−01
BDF3	2.222e−01	2.033e−01	−2.246e−01	4.303e−01	−4.303e−01	2.246e−01	−2.033e−01	−2.222e−01
Exact	2.225e−01	2.036e−01	−2.225e−01	4.296e−01	−4.296e−01	2.225e−01	−2.036e−01	−2.225e−01
T = 1.00								
BDF1	1.821e−01	1.669e−01	−1.829e−01	3.512e−01	−3.512e−01	1.829e−01	−1.669e−01	−1.821e−01
BDF2	1.821e−01	1.670e−01	−1.832e−01	3.514e−01	−3.514e−01	1.832e−01	−1.670e−01	−1.821e−01
BDF3	1.821e−01	1.670e−01	−1.832e−01	3.514e−01	−3.514e−01	1.832e−01	−1.670e−01	−1.821e−01
Exact	1.822e−01	1.673e−01	−1.822e−01	3.513e−01	−3.513e−01	1.822e−01	−1.673e−01	−1.822e−01

Table 7 The numerical results of u, using $\Delta x = 0.025$, $\Delta y = 0.025$, $\nu = 0.01$

(x,y)	(0.25,0.25)	(0.5,0.25)	(0.75,0.25)	(0.25,0.5)	(0.75,0.5)	(0.25,0.75)	(0.5,0.75)	(0.75,0.75)
T = 0.25								
BDF1	3.939e−01	6.988e−01	3.988e−01	2.660e−01	−2.660e−01	−3.988e−01	−6.988e−01	−3.939e−01
BDF2	3.946e−01	7.004e−01	4.002e−01	2.655e−01	−2.655e−01	−4.002e−01	−7.004e−01	−3.946e−01
BDF3	3.946e−01	7.005e−01	4.002e−01	2.655e−01	−2.655e−01	−4.002e−01	−7.005e−01	−3.946e−01
Exact	3.935e−01	6.862e−01	3.935e−01	2.620e−01	−2.620e−01	−3.935e−01	−6.862e−01	−3.935e−01
T = 0.50								
BDF1	2.910e−01	5.627e−01	2.944e−01	2.619e−01	−2.619e−01	−2.944e−01	−5.627e−01	−2.910e−01
BDF2	2.913e−01	5.649e−01	2.965e−01	2.627e−01	−2.627e−01	−2.965e−01	−5.649e−01	−2.913e−01
BDF3	2.913e−01	5.649e−01	2.965e−01	2.627e−01	−2.627e−01	−2.965e−01	−5.649e−01	−2.913e−01
Exact	2.912e−01	5.605e−01	2.912e−01	2.620e−01	−2.620e−01	−2.912e−01	−5.605e−01	−2.912e−01
T = 0.75								
BDF1	2.272e−01	4.481e−01	2.284e−01	2.176e−01	−2.176e−01	−2.284e−01	−4.481e−01	−2.272e−01
BDF2	2.274e−01	4.494e−01	2.299e−01	2.181e−01	−2.181e−01	−2.299e−01	−4.494e−01	−2.274e−01
BDF3	2.274e−01	4.494e−01	2.299e−01	2.182e−01	−2.182e−01	−2.299e−01	−4.494e−01	−2.274e−01
Exact	2.274e−01	4.480e−01	2.274e−01	2.180e−01	−2.180e−01	−2.274e−01	−4.480e−01	−2.274e−01
T = 1.00								
BDF1	1.857e−01	3.686e−01	1.861e−01	1.815e−01	−1.815e−01	−1.861e−01	−3.686e−01	−1.857e−01
BDF2	1.858e−01	3.694e−01	1.870e−01	1.819e−01	−1.819e−01	−1.870e−01	−3.694e−01	−1.858e−01
BDF3	1.858e−01	3.694e−01	1.870e−01	1.819e−01	−1.819e−01	−1.870e−01	−3.694e−01	−1.858e−01
Exact	1.858e−01	3.688e−01	1.858e−01	1.819e−01	−1.819e−01	−1.858e−01	−3.688e−01	−1.858e−01

Table 8 The numerical results of v, using $\Delta x = 0.0250$, $\Delta y = 0.0250$, $\nu = 0.01$

(x,y)	(0.25,0.25)	(0.5,0.25)	(0.75,0.25)	(0.25,0.5)	(0.75,0.5)	(0.25,0.75)	(0.5,0.75)	(0.75,0.75)
T = 0.25								
BDF1	3.939e−01	2.660e−01	−3.988e−01	6.988e−01	−6.988e−01	3.988e−01	−2.660e−01	−3.939e−01
BDF2	3.946e−01	2.655e−01	−4.002e−01	7.004e−01	−7.004e−01	4.002e−01	−2.655e−01	−3.946e−01
BDF3	3.946e−01	2.655e−01	−4.002e−01	7.005e−01	−7.005e−01	4.002e−01	−2.655e−01	−3.946e−01
Exact	3.935e−01	2.620e−01	−3.935e−01	6.862e−01	−6.862e−01	3.935e−01	−2.620e−01	−3.935e−01
T = 0.50								
BDF1	2.910e−01	2.619e−01	−2.944e−01	5.627e−01	−5.627e−01	2.944e−01	−2.619e−01	−2.910e−01
BDF2	2.913e−01	2.627e−01	−2.965e−01	5.649e−01	−5.649e−01	2.965e−01	−2.627e−01	−2.913e−01
BDF3	2.913e−01	2.627e−01	−2.965e−01	5.649e−01	−5.649e−01	2.965e−01	−2.627e−01	−2.913e−01
Exact	2.912e−01	2.620e−01	−2.912e−01	5.605e−01	−5.605e−01	2.912e−01	−2.620e−01	−2.912e−01
T = 0.75								
BDF1	2.272e−01	2.176e−01	−2.284e−01	4.481e−01	−4.481e−01	2.284e−01	−2.176e−01	−2.272e−01
BDF2	2.274e−01	2.181e−01	−2.299e−01	4.494e−01	−4.494e−01	2.299e−01	−2.181e−01	−2.274e−01
BDF3	2.274e−01	2.182e−01	−2.299e−01	4.494e−01	−4.494e−01	2.299e−01	−2.182e−01	−2.274e−01
Exact	2.274e−01	2.180e−01	−2.274e−01	4.480e−01	−4.480e−01	2.274e−01	−2.180e−01	−2.274e−01
T = 1.00								
BDF1	1.857e−01	1.815e−01	−1.861e−01	3.686e−01	−3.686e−01	1.861e−01	−1.815e−01	−1.857e−01
BDF2	1.858e−01	1.819e−01	−1.870e−01	3.694e−01	−3.694e−01	1.870e−01	−1.819e−01	−1.858e−01
BDF3	1.858e−01	1.819e−01	−1.870e−01	3.694e−01	−3.694e−01	1.870e−01	−1.819e−01	−1.858e−01
Exact	1.858e−01	1.819e−01	−1.858e−01	3.688e−01	−3.688e−01	1.858e−01	−1.819e−01	−1.858e−01

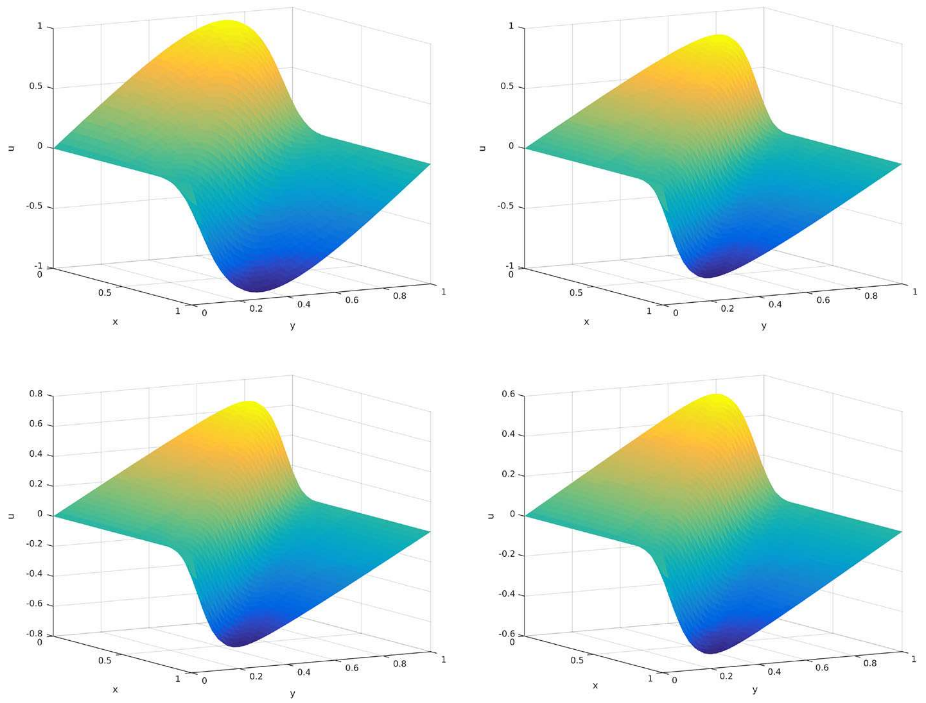


Fig. 8 Numerical solutions for v at $\Delta x = 0.025$, $\Delta y = 0.025$, $\Delta t = 0.001$, $\nu = 0.02$, v(x,y,t=0.25)(top left), v(x,y,t=0.5) (top right), v(x,y,t=0.75)(bottom left), v(x,y,t=1) (bottom right)

Stability Analysis

Stability of the present scheme depends on the stability of the system of ordinary differential equations (20).

$$\frac{d\mathbf{\Phi}}{dt} = A\mathbf{\Phi} \tag{26}$$

where, $\mathbf{\Phi} = (\phi_{00}, \phi_{01}, \ldots, \phi_{0N}, \phi_{10}, \phi_{11}, \ldots, \phi_{1N}, \ldots, \phi_{N0}, \phi_{N1}, \ldots, \phi_{NN})^T$ and the matrix A is given by

$$A = \begin{bmatrix} A_1 & B_1 & 0 \\ B_1/2 & A_1 & B_1/2 \\ 0 & B_1 & A_1 \end{bmatrix}$$

If the domain is divided into 3×3 uniform grids then the corresponding matrix A is given by

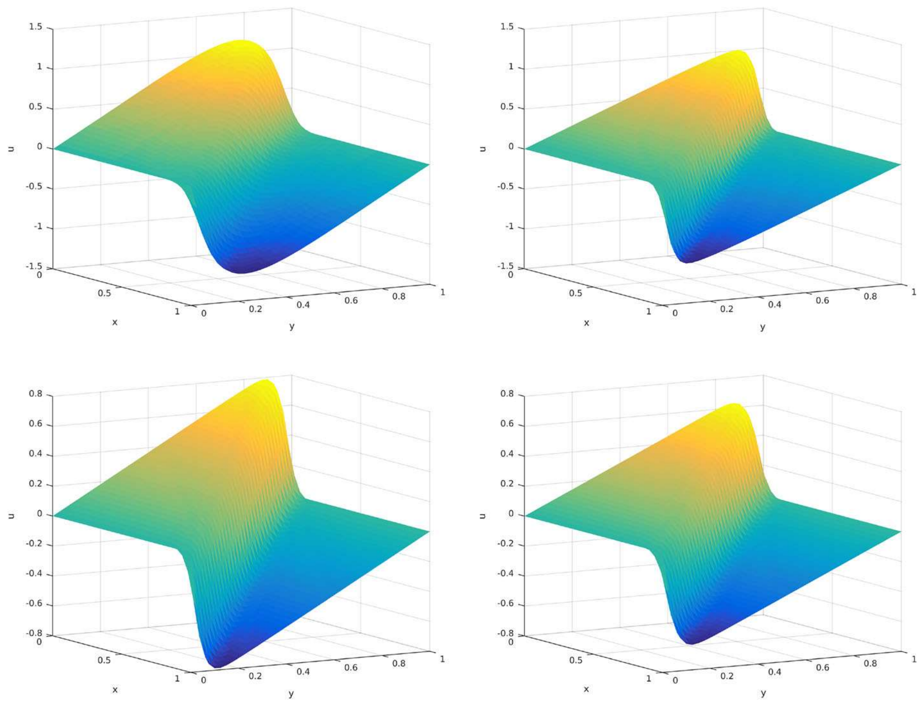


Fig. 9 Numerical solutions for u at $\Delta x = 0.025$, $\Delta y = 0.025$, $\Delta t = 0.001$, $\nu = 0.01$, u(x,y,t=0.25)(top left), u(x,y,t=0.5) (top right), u(x,y,t=0.75)(bottom left), u(x,y,t=1) (bottom right)

$$A = \begin{bmatrix} \begin{bmatrix} -2p & 2p_2 & 0 \\ p_2 & -2p & p_2 \\ 0 & 2p_2 & -2p \end{bmatrix} & \begin{bmatrix} 2p_1 & 0 & 0 \\ 0 & 2p_1 & 0 \\ 0 & 0 & 2p_1 \end{bmatrix} & \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \\ \begin{bmatrix} p_1 & 0 & 0 \\ 0 & p_1 & 0 \\ 0 & 0 & p_1 \end{bmatrix} & \begin{bmatrix} -2p & 2p_2 & 0 \\ p_2 & -2p & p_2 \\ 0 & 2p_2 & -2p \end{bmatrix} & \begin{bmatrix} p_1 & 0 & 0 \\ 0 & p_1 & 0 \\ 0 & 0 & p_1 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 2p & 0 & 0 \\ 0 & 2p & 0 \\ 0 & 0 & 2p \end{bmatrix} & \begin{bmatrix} -2p & 2p_2 & 0 \\ p_2 & -2p & p_2 \\ 0 & 2p_2 & -2p \end{bmatrix} \end{bmatrix}$$

The stability of the ODE system is analyzed through the value of eigenvalues of the matrix A, which in turn depends upon eigenvalues of the matrices A_1 and B_1. If the eigenvalues of the matrix A have either negative real part or zero value, then the system (26) is stable. We have explored the stability of the system by finding the eigenvalues of the matrix A for different grid sizes and different values of kinematic viscosity. The Fig. 1 shows eigenvalues of the matrix A by taking viscosity $\nu = 1, 0.1$ at different grid sizes. Eigenvalues of the matrix A for viscosity $\nu = 0.02, 0.01$ at different grid sizes are presented in Fig. 2. It is clear from figures that all eigenvalues of the matrix A lie in the negative half plane.

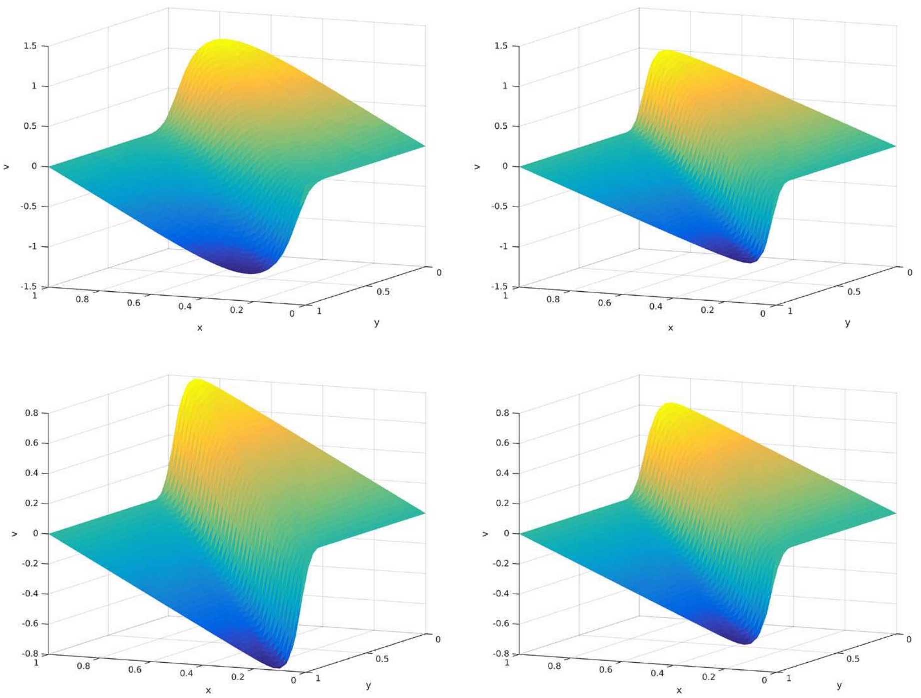


Fig. 10 Numerical solutions for v at $\Delta x = 0.025$, $\Delta y = 0.025$, $\Delta t = 0.001$, $\nu = 0.01$, v(x,y,t=0.25)(top left), v(x,y,t=0.5) (top right), v(x,y,t=0.75)(bottom left), v(x,y,t=1) (bottom right)

Numerical Experiments

In this section, we present numerical simulation outcomes obtained by the implementation of the proposed scheme on two-dimensional coupled Burgers' equation. Several numerical experiments are conducted for different values of kinematic viscosity ν, taking various grid sizes at different time levels. The numerical results are validated through comparison with the exact solution proposed by Gao [26]. Tables 1 and 2 show the comparison between exact solution and numerical solution generated through BDF1, BDF2 and BDF3 taking kinematic viscosity $\nu = 1$ and at final time $T = 0.25, 0.5, 0.75, 1$ for u and v respectively. Profiles of velocity u and v are presented in Figs. 3 and 4 respectively.

A comparison between numerical and exact solution for kinematic viscosity $\nu = 0.1$ is presented in Tables 3 and 4 for u and v respectively. It is evident from the tables that numerical solution exhibits excellent agreement with the exact solution. Figs. 5 and 6 show the profiles of velocity u and v.

We repeated our experiment with small values of kinematic viscosity ν. The obtained numerical results are tabulated and reported in tables. Tables 5 and 6 show the comparison between exact solution and numerical solution generated through BDF1, BDF2 and BDF3 taking kinematic viscosity $\nu = 0.02$ for u and v respectively. On the other hand, numerical and exact values presented in Tables 7 and 8 are obtained by taking $\nu = 0.01$ for u and v respectively. The above experiment demonstrates that numerical solution matches with exact solution even for small value of kinematic viscosity. Numerical profiles for velocity u and v are depicted in Figs. 7 and 8 for $\nu = 0.02$ and Figs. 9 and 10 for $\nu = 0.01$.

Table 9 Numerical rate of convergence and L_2, L_∞ error norms at $T = 0.5$, with $\Delta x = \Delta y$ and $\Delta t = 0.0001$ for $\nu = 0.01$

N	BDF1			BDF2			BDF3		
	L_2	L_∞	ROC	L_2	L_∞	ROC	L_2	L_∞	ROC
41	0.030035	0.010026	...	0.033934	0.010966	...	0.033976	0.010978	...
81	0.008281	0.002195	2.231006	0.012429	0.002852	1.978235	0.012485	0.002862	1.974336
121	0.004304	0.000755	2.658456	0.006591	0.001280	1.996386	0.006658	0.001290	1.985483
161	0.005307	0.000642	0.567073	0.004364	0.000710	2.061004	0.004441	0.000721	2.038961

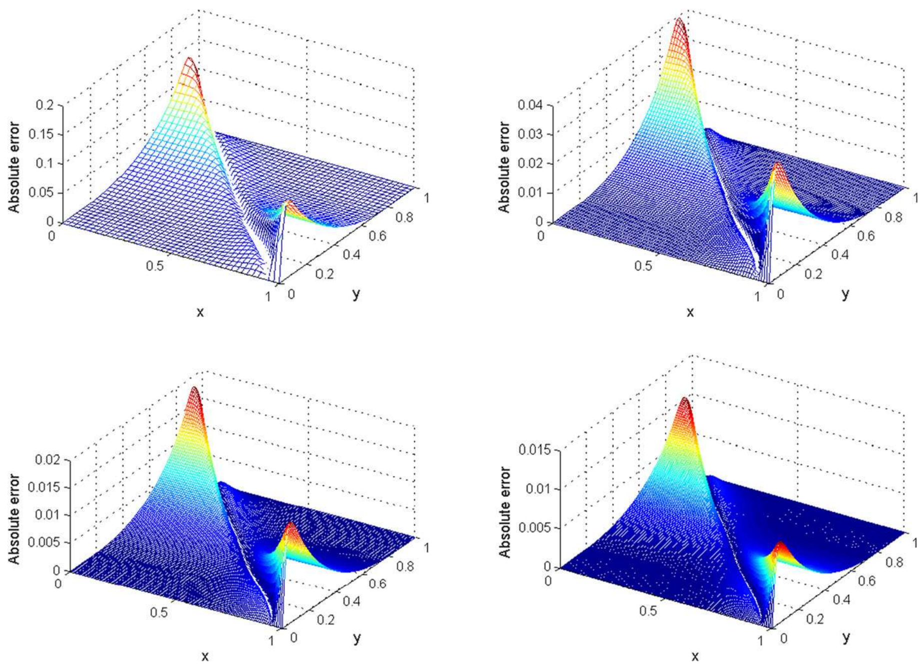


Fig. 11 Absolute error plots at $T = 0.5$ and $x = 0.5$ for various grid sizes computed with $\Delta t = 0.001$ and $\nu = 0.01$, $N = M = 41$ (top left), $N = M = 81$ (top right), $N = M = 121$(bottom left), $N = M = 161$ (bottom right)

To verify the accuracy of the proposed schemes, we have computed L_2 and L_∞ error norm. The error aroused in the numerical computations are estimated using the formula

$$L_2 = \left\| u^a - u \right\|_2 = \left(h \sum_{i=0}^{N} \left| u_i^a - u_i \right|^2 \right)^{\frac{1}{2}},$$

$$L_\infty = \left\| u^a - u \right\|_\infty = \max_{0 \le i \le N} \left| u_i^a - u_i \right|. \tag{27}$$

Here $u(x,t)$ denote exact solution and u^a denote numerical solution generated by the proposed method and $u_i = u(x_i, t)$. The numerical rate of convergence (ROC) of the proposed scheme is estimated using the formula

$$\text{ROC} \approx \frac{\log(E(N_2)/E(N_1))}{\log(N_1/N_2)}, \tag{28}$$

where $E(N_j)$ indicates the L_∞ error value obtained by using N_j number of grid points. In Table 9, L_2 and L_∞ error norm are estimated for kinematic viscosity $\nu = 0.01$ and at final time $T = 0.5$. It is observed from the tables that as the grid sizes decreased, L_2 and L_∞ error also decreased. Rate of convergence (ROC) of the proposed schemes for different grid sizes are also presented in the table.

Figure 11 shows absolute error obtained by implementing the present scheme using BDF2, for $\nu = 0.01$, $\Delta t = 0.001$ using different grid sizes at final time $T = 0.5$. As the grid sizes decreased, absolute error also decreased and hence accuracy of the schemes increased.

Conclusion

In this paper, we have developed a robust numerical method for solving the two-dimensional coupled Burgers' equation with the sinusoidal initial condition. The nonlinear coupled Burgers' equation is first transformed into the linear diffusion equation by using the nonlinear transformation. The MOL technique is employed to reduce the partial differential equation to the system of ordinary differential equations. Backward differentiation formulas of order one, two and three are employed to solve the ODE system. Accuracy and efficiency of the present scheme are demonstrated through numerical experiments.

Acknowledgements The authors are very grateful to the reviewers for their valuable comments and suggestions.

References

1. Reyna, L.G., Ward, M.J.: On the exponentially slow motion of a viscous shock. Commun. Pure Appl. Math. **XLVIII**, 79–120 (1995)
2. Bec, J., Khanin, K.: Burgers turbulence. Phys. Rep. **447**, 1–66 (2007)
3. Vergassola, M., Dubrulle, B., Frisch, U., Nullez, A.: Burgers' equation, devil's staircases and the mass distribution for large-scale structures. Astronom. Astrophys. **289**, 325–356 (1994)
4. Lighthill, M.: Viscosity effects in sound waves of finite amplitude. In: Surveys in Mechanics. Cambridge University Press, Cambridge, pp. 250–351 (1956)
5. Greenshields, B.: A study of traffic capacity. Proc. Highway Res. Board **14**, 448–447 (1935)
6. Jain, P., Holla, D.: Numerical solutions of coupled burgers' equation. Int. J. Non-Linear Mech. **13**(4), 213–222 (1978)
7. Arminjon, P., Beauchamp, C.: A finite element method for burgers' equation in hydrodynamics. Int. J. Numer. Methods Eng. **12**(3), 415–428 (1978)
8. Fletcher, C.A.J.: Generating exact solution of the two-dimwnsional Burgers' equation. Int. J. Numer. Methods Fluids **3**, 213–216 (1983)
9. Esipov, S.E.: Coupled burgers equations: a model of polydispersive sedimentation. Phys. Rev. E **52**, 3711–3718 (1995)
10. Bahadir, A.R.: A fully implicit finite difference scheme for two dimensional Burgers equations. Appl. Math. Comput. **137**, 131–137 (2003)
11. Radwan, S.F.: Comparison of higher-order accurate schemes for solving the two-dimensional unsteady Burgers' equation. J. Comput. Appl. Math. **174**, 383–397 (2005)
12. Sakai, K., Kimura, I.: A numerical sheme based on a solution of nonlinear advection-diffusion equations. J. Comput. Appl. Math. **173**, 39–55 (2005)
13. Zhao, G., Yu, X., Zhang, R.: The new numerical method for solving the system of two-dimensional Burgers equations. Comput. Math. Appl. **62**, 3279–3291 (2011)

14. Kweyu, M.C., Manyonge, W.A., Koross, A., Ssemaganda, V.: Numerical solutions of the Burgers' system in two dimensions under varied initial and boundary conditions. Appl. Math. Sci. **6**(113), 5603–5615 (2012)
15. Kannan, R., Wang, Z.J.: A high order spectral volume solution to the Burgers' equation using the Hopf-Cole transformation. Int. J. Numer. Methods Fluids **69**, 781–801 (2012)
16. Aminikhah, H.: An analytical approximation for coupled viscous Burgers equation. Appl. Math. Model. **37**, 5979–5983 (2013)
17. Yang, Q.: The upwind finite volume element method for two-dimensional Burgers equation. Abstr. Appl. Anal. **2013**, 1–11 (2013)
18. Srivastava, V.K., Awasthi, M.K., Singh, S.: An implicit logarithmic finite-difference technique for two dimensional coupled viscous Burgers equation. AIP Adv. **3**(12), 122–125 (2013)
19. Srivastava, V.K., Singh, S., Awasthi, M.K.: Numerical solutions of coupled Burgers' equations by an implicit finite-difference scheme. AIP Adv. **3**(8), 0–7 (2013)
20. Zhang, X., Tian, H., Chen, W.: Local method of approximate particular solutions for two-dimensional unsteady Burgers equations. Comput. Math. Appl. **66**, 2425–2432 (2014)
21. Srivastava, V.K., Tamsir, M., Awasthi, M.K., Singh, S.: One-dimensional coupled Burgers' equation and its numerical solution by an implicit logarithmic finite-difference method. AIP Advances **4**(3), 0–10 (2014)
22. Mittal, R.C., Tripathi, A.: Numerical solutions of two-dimensional Burgers equations using modified Bi-cubic B-spline finite elements. Eng. Comput. **32**, 1275–1306 (2015)
23. Mittal, R., Kaur, H., Mishra, V.: Haar wavelet-based numerical investigation of coupled viscous Burgers' equation. Int. J. Comput. Math. **92**(8), 1643–1659 (2015)
24. Mohanty, R.K., Dai, W., Han, F.: Compact operator method of accuracy two in time and four in space for the numerical solution of coupled viscous Burgers' equations. Appl. Math. Comput. **256**, 381–393 (2015)
25. Bhatt, H.P.: Fourth-order compact schemes for the numerical simulation of coupled Burgers' equation. Comput. Phys. Commun. **200**, 117–138 (2016)
26. Gao, Q., Zou, M.: An analytical solution for two and three dimensional nonlinear Burgers' equation. Appl. Math. Model. **45**, 255–270 (2017)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.