TÜBİTAK

Turkish Journal of Mathematics

http://journals.tubitak.gov.tr/math/

Research Article

Turk J Math
(2021) 45: 529 – 539
© TÜBİTAK
doi:10.3906/mat-2001-7

Numerical investigation of viscous effects on the nonlinear Burgers equation

Muhammad Imran KHAN, Abdul RAUF, Abdullah SHAH*
Department of Mathematics, COMSATS University Islamabad, Islamabad, Pakistan

Received: 03.01.2020 • **Accepted/Published Online:** 19.12.2020 • **Final Version:** 21.01.2021

Abstract: This research article presents the numerical solution of the viscous Burgers equation. The diagonally implicit fractional step θ (DIFST) scheme is used for the time discretization and the space derivative is discretized by the conforming finite element method with quadrilateral mesh. The viscosity effect on the shock wave is calculated with an estimation of the L_2 error. For comparison of different time discretization schemes, three test problems are computed. The stability and accuracy of the schemes are given by estimating the L_2 error norm. Numerical simulation for one- and two-dimensional problems are given and illustrated graphically. The effect of the viscosity parameter on the nonlinearity of the Burger equation is computed. The stability of the schemes for different time steps with CPU time is also given.

Key words: Burgers equation, diagonally implicit fractional step $\theta-$ scheme, conforming finite element method, DUNE-PDELab

1. Introduction

Many natural and physical phenomena in science and engineering are governed by partial differential equations (PDEs). However, finding the exact solution of some nonlinear and stiff partial differential equations is difficult. So, the numerical methods [1, 2] can be used to compute the approximate solution. The Burgers equation and advection equation [3] are PDEs with several applications in different areas such as fluid mechanics [4], nonlinear acoustics [5], gas dynamics and traffic flow [6], fluid dynamics [7], electromagnetic [8], heat transfer [9] and turbulence phenomena in a viscous fluid [10, 11]. In this research work, some numerical techniques are used to study the diffusion phenomena at the sharp corner (jump discontinuities) in solving the viscous Burgers equation.

In the literature, different numerical methods have been developed to find the numerical solution of the Burgers equation i.e. Mittal et al. [12] and Kutluay et al. [13] has developed an explicit finite difference method (FDM) and Bahadir et al. [14] proposed the mixed boundary element method for the numerical solutions of the one-dimensional Burgers equation. Kadalbajoo et al. [15] proposed the implicit finite difference scheme on uniform grids. The finite volume method (FVM) is proposed by Shih and Qin [16] and Kutluay et al. [17] developed the finite element method (FEM). The main advantage of the FEM over FDM is that the resulting procedures are stable and have flexibility in choosing the basis function. Also, the FEM is more suitable for controlling the dispersion and dissipation errors in the propagation of discontinuities [18].

This research work deals with the numerical solution of the viscous Burgers equation by using the FEM

*Correspondence: abdullah_shah@comsats.edu.pk
2010 *AMS Mathematics Subject Classification:* 65M60 65M22

for the discretization of the space derivative [19] and the DIFST scheme is used for temporal discretization. The L_2 error estimates are calculated and comparison of different time discretization schemes is provided. The distributed and unified numerics environment (DUNE) is a modular toolbox for solving partial differential equations (PDEs). The numerical simulations are carried out using the DUNE-PDELab [20, 21]. DUNE is a software platform that substantially reduces the time to implement discretization and a solver for solving partial differential equations (PDEs) [22].

The paper is organized as follow: Section 2 provides a description of the model problem. In Section 3, numerical discretization is discussed in detail. Section 4 consists of numerical results obtained by considering different test problems and grid-independent study. Section 5 concludes this paper.

2. Problem formulation

Let us consider the following form of the Burgers equation [23]:

$$\partial_t v - \nu \Delta v + v \nabla v = 0 \qquad \text{in} \quad \Omega \times \Sigma, \tag{2.1}$$

$$v(.,t) = g(.,t) \qquad \text{on} \quad \Gamma_D \subseteq \partial\Omega, \tag{2.2}$$

$$-\nabla v \cdot n(.,t) = j \qquad \text{on} \quad \Gamma_N = \partial\Omega \setminus \Gamma_D, \tag{2.3}$$

$$v(.,t) = v_0(.,t) \qquad \text{at} \quad t = t_0. \tag{2.4}$$

Here, $\Omega \subset R^d (d = 1,2,3)$ is a bounded domain, t in the subscript is the time derivative, v is the conserved variable, ν is the viscosity and Δ is the Laplacian operator. Equations 2.2 and 2.3 represent the Dirichlet and the Neumann boundary conditions respectively, where n is the outward normal to the surface and v_0 is the given initial condition [23].

3. Numerical discretization

In this section, we give details of the discrtization schemes in both space and time.

3.1. Discretization in space

Consider the steady-state Burgers equation of the form:

$$-\nu \Delta v + v \nabla v = 0, \qquad \text{in } \Omega. \tag{3.1}$$

To obtain the weak form of Equation 3.1, we multiply it with a suitable test function as follows:

$$v \in V \text{ such that } \quad r(v,w) = 0 \quad \forall w \in W, \tag{3.2}$$

$$V = \{w \in H^1(\Omega) : w = g \text{ on } \Gamma_D\} \text{ and } W = \{w \in H^1(\Omega) : w = 0 \text{ on } \Gamma_D\},$$

are the functional spaces and the first Sobolev space $H^1(\Omega)$ is given by,

$$H^1(\Omega) = \left\{ w(x) ; x \epsilon \Omega, \int_\Omega w^2 dx < \infty \text{ and } \int_\Omega (w')^2 dx < \infty \right\}.$$

The $r^{NLP}(v,w)$ is the residual form of nonlinear problem defined as:

$$r^{NLP}(v,w) = \int_\Omega \left(-\nu \nabla v \cdot \nabla w + \frac{1}{2}\nabla v^2 \cdot w)\right) dx. \tag{3.3}$$

The conforming finite element method [24] depends on the weak formulation of the problem. In the weak formulations, the functional spaces V and W are replaced by $V_h \subset V$ and $W_h \subset W$ which are the discrete finite dimensional subspaces, such that:

$$V_h = \text{span}\{\varphi_1, \varphi_2 \ldots, \varphi_n\}, \quad W_h = \text{span}\{\Psi_1, \Psi_2 \ldots, \Psi_m\}.$$

Here, the mesh size is denoted by the parameter h. Let us consider the approximate solution "v_h" as $v_h = \sum_{j=1}^{n}(y)_j\varphi_j$ by considering the coefficient vector $y \in \mathbb{R}^n$, then Equation 3.2 becomes:

$$\begin{aligned} & v_h \in V_h \quad \text{such that } r(v_h, w) = 0, \ \forall \quad w \in W_h, \\ \Leftrightarrow\ & r\left(\textstyle\sum_{j=1}^{n}(y)_j\varphi_j, \Psi_i\right) = 0, \ \forall \quad i = 1, \ldots, m, \\ \Leftrightarrow\ & R(y) = 0\,. \end{aligned}$$

The approximate solution of the resulting nonlinear algebraic equation $R(y) = 0$ is computed by using the damped Newton's method of the form:

$$y^{(k+1)} = y^{(k)} - \mu^k X(y^{(k)})R(y^{(k)}). \tag{3.4}$$

With μ^k as a damping parameter, $(J(y^{(k)}))_{i,j} = \frac{\partial R_i}{\partial y_j}(y^{(k)})$ is a Jacobian matrix and $X(y^{(k)})$ is a preconditioning matrix. By considering $n = m$ and $J(y^{(k)})$ an invertible Jacobian matrix in each time step, the resulting linear system $J(y^{(k)})w = R(y^{(k)})$ is obtained which is solved by an efficient linear solver such as biconjugate gradient (BiCG) method.

3.2. Discretization in time

For the unsteady part of Equation 2.1, we find $v \in L_2(t_0, t_0 + T; v_g + W(t))$ such that

$$\frac{d}{dt}\int_\Omega vw\,dx + \left(-\int_\Omega \nu\nabla v \cdot \nabla w + \frac{1}{2}\nabla v^2 w\right) dx = 0, \qquad \forall w \in W(t), \text{ and } t \in \Sigma, \tag{3.5}$$

where $W(t) = \{w \in H^1(\Omega) \,:\, w = 0 \text{ on } \Gamma_D(t)\}$ and $H^1(\Omega) \ni v_g(t)|_{\Gamma_D} = g$.

Now, Equations 3.3 and 3.5 can be rewritten as:

$$\frac{d}{dt}m(v,w;t) + r(v,w;t) = 0 \quad \forall w \in W(t), t \in \Sigma, \tag{3.6}$$

where the residual form for space $r(v,w;t)$ now depend on time as well with $m(v,w;t) = \int_\Omega vw\,dx$ as time residual. The conforming finite element space $W_h^{k,d}(\mathcal{T}_h, t)$ and the function $v_{h,g}(t)$ are chosen in such a way that $v_h(t) \in V_h(t) = v_{h,g}(t) + W_h(t)$. Therefore, both the residuals for time and space are approximated as

$m(v,w;t) \approx m_h(v_h,w_h;t)$ and $r(v,w;t) \approx r_h(v_h,w_h;t)$ respectively. Next, by dividing the time interval into subintervals and taking integration of the ODE system results as:

$$\overline{\Sigma} = \{t^0\} \cup (t^0,t^1] \cup \ldots \cup (t^{N-1},t^N],$$

with $t^0 = t_0$, $t^N = t_0 + T$. $t^{k-1} < t^k$ for $1 \leq k \leq N$, and we set the time step as $\Delta t^k = t^{k+1} - t^k$. We used the one-step $\theta-$scheme Wang and Li [25] to solve the system of ODE, as follows:

find $v_h^{k+1} \in V_h(t^{k+1})$ such that

$$\frac{1}{\Delta t^k}(m_h(v_h^{k+1},w;t^{k+1}) - m_h(v_h^k,w;t^k)) + \theta r_h(v_h^{k+1},w;t^{k+1}) + (1-\theta)r_h(v_h^k,w;t^k) = 0, \tag{3.7}$$

$$\forall w \in W_h(t^{k+1}). \tag{3.8}$$

By rearranging the terms for $(k+1)$ and k, we get a solution to a nonlinear Equation 3.7, i.e.

find $v_h^{k+1} \in V_h(t^{k+1})$ such that

$$m_h(v_h^{k+1},w;t^{k+1}) + \Delta t^k \theta r_h(v_h^{k+1},w;t^{k+1}) - m_h(v_h^k,w;t^k) + \Delta t^k(1-\theta)r_h(v_h^k,w;t^k) = 0, \tag{3.9}$$

For diagonally implicit fractional θ-scheme, put $\theta = 1 - \frac{1}{2}\sqrt{2}$ and Equation 3.9 becomes

$$(m_h + \Delta t^k(1-\frac{2}{\sqrt{2}})r_h)(v_h^{k+1},w;t^{k+1}) - (m_h + \Delta t^k(\frac{2}{\sqrt{2}})r_h)(v_h^k,w;t^k) = 0,$$

$$\forall w \in W_h(t^{k+1}).$$

At each time step, the damped Newton's method is used to solve the resulting nonlinear system of equations, while the SSOR-BiCG method [26] is used to solve the linear system at each iteration of the Newton's method.

3.3. Shu–Osher form

All explicit and implicit Runge-Kutta (RK) methods are represented by Shu–Osher [27] form which is introduced by mathematical relation that simplifies the resulting numerical methods. The Shu–Osher form [28, 29] for the time residual m_h and space residual r_h can be written as:

1. $v_h^{(0)} = v_h^k$,
2. $i = 1,\ldots,p \in N$, find $v_h^{(i)} \in v_{h,g}(t^k + e_i\Delta t^k) + v_h(t^{k+1})$:

   $$\sum_{j=0}^{s}\left[b_{ij}m_h\left(v_h^{(j)},w;t^k + e_j\Delta t^k\right) + c_{ij}\Delta t^k r_h\left(v_h^{(j)},w;t^k + e_j\Delta t^k\right)\right] = 0,\ \forall\ v \in V_h(t^{k+1}),$$

3. $v_h^{k+1} = v_h^{(s)}$.

Here, we assume that in the interval $(t^k,t^{k+1}]$, the boundary conditions remain the same.

4. Results and discussion

In this section, the Burgers equation (2.1) is solved using the FEM for space discretization and the DIFST scheme is used for the temporal discretization. By numerical experiment of different test problems, we studied the viscosity parameter μ effect on the solution. For numerical simulations, Intel Core i7, 3.20×4 GHz CPU and 16 GB of RAM with the Ubuntu 16.04 operating system are used.

In problems 1 and 2, we consider the 1D viscous Burgers equation and in problem 3, 2D viscous Burgers equation is solved. General form of viscous Burgers equation is:

$$\partial_t v - \nu \Delta v + v \nabla v = 0, \qquad \text{in } \Omega \times \Sigma, \tag{4.1}$$

with the Dirichlet boundary condition on the given domain Ω. We choose different exact solutions of the Burgers equation and the L_2 error estimate is calculated with different degrees of freedom (DOF).

4.1. Test problem 1

In this case, we take the exact solution [23]:

$$v(x) = \sin(\pi x) \tag{4.2}$$

with the Dirichlet boundary condition on the given domain $\Omega = (0, 1)$. The L_2 error norm and CPU time for different values of nodes and degree of freedom (DOF) are presented in Tables 1 and 2 by using Q_1 and Q_2 conforming FEM, respectively. It is noted that in Tables 1 and 2, when there is an increase in the number of nodes and degrees of freedom, then L_2 error is decreased. It is also observed that the Q_2 conforming FEM is more accurate than Q_1 at the cost of more computational time.

Table 1. L_2-error estimate using the Q_1 conforming FEM.

N	L_2 error ($\Delta t = 0.1\Delta x$)	DOF	CPU time	N	L_2 error ($\Delta t = 0.5\Delta x$)	DOF	CPU time
15	2.5821E–03	256	8.84010E–02	15	2.5821E–03	256	2.1706E–02
30	6.4027E–04	961	3.3693E–01	30	6.4027E–04	961	8.0237E–02
60	1.6150E–04	3721	1.3813E+00	60	1.6150E–04	3721	3.3305E–01
120	4.0386E–05	14641	5.0733E+01	120	4.0386E–05	14641	1.3497E+00

Table 2. L_2-error estimate using the Q_2 conforming FEM.

N	L_2 error ($\Delta t = 0.1\Delta x$)	DOF	CPU time	N	L_2 error ($\Delta t = 0.5\Delta x$)	DOF	CPU time
15	3.1234E–05	961	3.9869E–01	15	3.1234E–05	961	1.02120E–01
30	3.9062E–06	3721	1.5299E+00	30	3.9062E–06	3721	4.1586E–01
60	4.8833E–07	14641	6.9242E+00	60	4.8833E–07	14641	2.1009E+00
120	6.1044E–08	58081	2.8276E+01	120	6.1044E–08	58081	8.0652E+00

For $\nu = 0.1$, it has shown that the shock becomes prominent with time as shown in Figure 1.

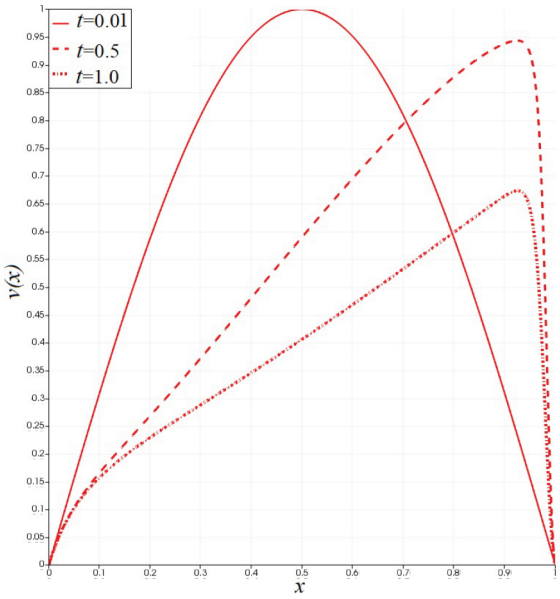

Figure 1. Solution at different times levels.

4.2. Test problem 2

We take the exact solution as:

$$v(x,t) = \frac{v_R + v_L}{2} - \frac{v_R + v_L}{2} \tanh\left(\frac{(x - 0.5 \times t)(v_L - v_R)}{4\nu}\right) \tag{4.3}$$

with, $v_R = 0$ and $v_L = 1$

In Tables 3 and 4, the L_2 error estimates and CPU times are computed at different nodes for Q_1 and Q_2. It is observed that the L_2 error estimates are decreased when there is an increase in the number of nodes and DOF. Although more accurate, the Q_2 is computationally expensive (in terms of CPU times) than that of Q_1.

Table 3. The L_2-error estimate using the Q_1 conforming FEM.

N	L_2 error ($\Delta t = 0.1\Delta x$)	DOF	CPU time	N	L_2 error ($\Delta t = 0.5\Delta x$)	DOF	CPU time
15	5.0846E–02	256	8.3789E–02	15	5.0846E–02	256	1.6461E–02
30	1.6665E–02	961	3.0538E–01	30	1.6665E–02	961	7.5153E–02
60	4.2455E–03	3721	1.2671E+00	60	4.2455E–03	3721	2.4192E–01
120	1.0672E–03	14641	4.8239E+00	120	1.0672E–03	14641	1.0219E+00

In Table 5, as ν increased from 0.01 to 0.3, it is observed that the L_2 error is decreased.

It is also observed that by increasing the value of ν, the solution becomes more diffusive (smooth).

The diffusive nature of the solution for different values of ν is illustrated in Figures 2a–2d. It is observed that by increasing ν, the sharpness of the solution decreases which is given in Figure 3.

Table 4. The L_2-error estimate using the Q_2 conforming FEM.

N	L_2 error ($\Delta t = 0.1\Delta x$)	DOF	CPU time	N	L_2 error ($\Delta t = 0.5\Delta x$)	DOF	CPU time
15	1.0185E–02	961	3.8950E–01	15	1.0185E–02	961	8.0398E–02
30	1.0478E–03	3721	1.5598E+00	30	1.0478E–03	3721	3.1547E–01
60	1.3780E–04	14641	6.0810E+00	60	1.3780E–04	14641	1.2102E+00
120	1.73487E–05	58081	2.2447E+01	120	1.73487E–05	58081	1.7811E+01

Table 5. The L_2-Error estimate with different ν.

N	ν	DOF	L_2 error
120	0.01	14641	3.5479E–02
120	0.02	14641	1.1627E–02
120	0.03	14641	6.3793E–03
120	0.1	14641	1.0672E–03
120	0.2	14641	3.7783E–04
120	0.3	14641	2.0571E–04

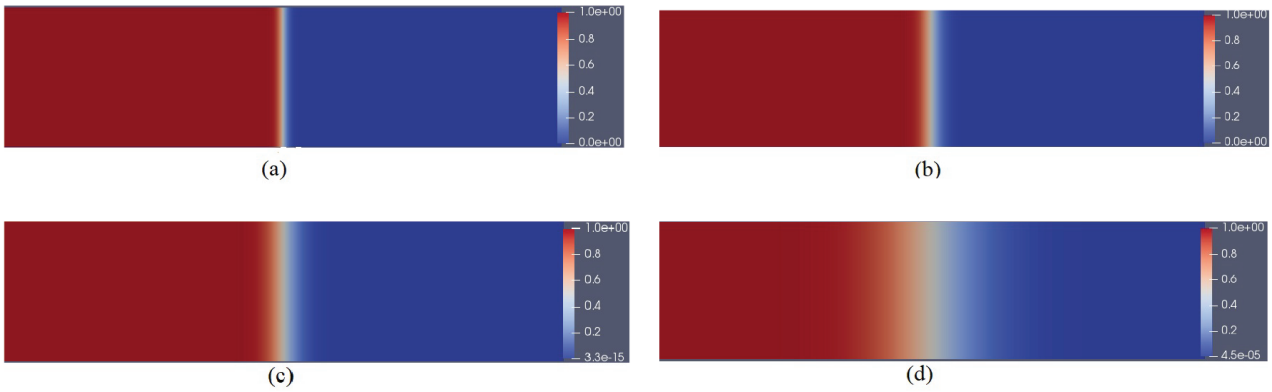

Figure 2. Solution behavior at (a) $\nu = 0.01$, (b) $\nu = 0.02$, (c) $\nu = 0.03$, and (d) $\nu = 0.1$.

4.3. Test problem 3

We choose the exact solution [30] as:

$$v(x, y, t) = \frac{3}{4} - \frac{1}{4 + 4e^{R(-t-4x+4y)/32}} \tag{4.4}$$

with $R = 5.0$. The computational domain is shown in Figure 4a, and approximate solution at the initial time is shown in Figure 4b.

In Tables 6 and 7, the L_2 error estimates and CPU times are given using the DIFST scheme at different nodes for Q_1 and Q_2. It is observed that the L_2 error estimates are decreased when there is an increase in the number of nodes and DOF. The Q_2 is computationally expensive (in terms of CPU times) than that of Q_1.

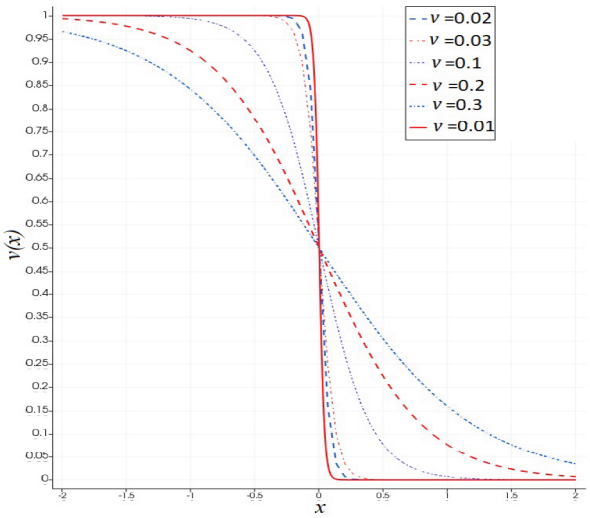

Figure 3. Solution profile at different values of ν.

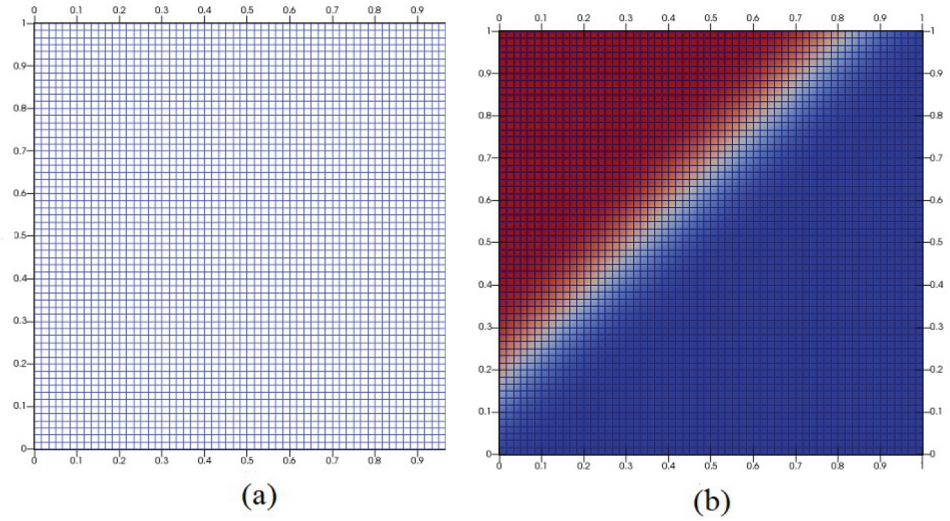

Figure 4. (a) Computational domain, (b) initial profile at $t = 0$.

Table 6. The L_2-error estimate using the Q_1 conforming FEM.

$N_x = N_y$	L_2 error ($\Delta t = 0.1\Delta x$)	DOF	CPU time	$N_x = N_y$	L_2 error ($\Delta t = 0.5\Delta x$)	DOF	CPU time
60	1.7078E–04	3721	1.0280E+01	60	1.7078E–04	3721	2.2178E+00
120	4.2882E–05	14641	1.0541E+02	120	4.2882E–05	14641	1.7483E+01
180	1.9074E–05	32761	3.7313E+02	180	1.9074E–05	32761	6.6094E+01
240	1.0732E–05	58081	7.4698E+02	240	1.0732E–05	58081	1.6466E+02

By the numerical solution of the viscous Burgers equation, it is observed that L_2 error estimate decrease with an increase of the number of grids and Q_2 finite element method is more accurate than Q_1 finite element

Table 7. The L_2-error estimate using the Q_2 conforming FEM.

$N_x = N_y$	L_2 error ($\Delta t = 0.1\Delta x$)	DOF	CPU time	$N_x = N_y$	L_2 error ($\Delta t = 0.5\Delta x$)	DOF	CPU time
60	1.9778E–06	14641	4.4042E+01	60	1.9778E–06	14641	1.7441E+01
120	2.4701E–07	58081	4.0352E+02	120	2.4701E–07	58081	8.6631E+01
180	7.3175E–08	130321	1.7309E+03	180	7.3175E–08	130321	3.3313E+02
240	3.0869E–08	231361	3.0073E+03	240	3.0869E–08	231361	8.5384E+02

method but it is computationally expensive.

4.4. Stability of the scheme

Now, we discuss the stability of the different time discretization scheme for test problem 1. All the results are taken by keeping viscosity parameter fixed ($\nu = 0.1$) and number nodes N = 60. In Table 8, it is observed that the one-step θ method, Alexander (order2), Alexander (order3) and DIFST method are stable for lagre time steps but explicit Euler's method, Heun's, Shu-third order and RK4 methods are unstable for lagre time steps.

Table 8. Computational costs and stability of the time discretization schemes.

Scheme	$\Delta t = 10^{-4}$	$\Delta t = 10^{-3}$	$\Delta t = 10^{-2}$	$\Delta t = 10^{-1}$
One-step θ	2.1446E+02	2.1652E+01	3.9256E+00	4.4606E-01
Alexander (order 2)	4.3964E+02	4.3089E+01	5.7173E+00	9.7595E-01
Alexander (order 3)	6.4294E+02	6.5925E+01	1.0005E+00	1.3727E+00
Fractional Step θ	5.7120E+02	6.5185E+01	7.8561E+00	1.3086E+00
Explicit Euler's	5.9160e+01	Fails	Fails	Fails
Heun's	1.1176e+02	Fails	Fails	Fails
Shu-third order	1.6737e+02	Fails	Fails	Fails
RK4	1.6737e+02	Fails	Fails	Fails

4.5. Grid-independent solution

In this section, we provided the grids independence study. By taking the different numbers of grids, the L_2 error estimates are computed. It is observed that L_2 error decreases with the increase in the number of grid points. It is also noted that the solutions become almost identical at the two grids 1680 and 3120 as shown in Table 9.

Table 9. The L_2-error estimate using the Q_1 conforming FEM.

No. of grids ($N_x = N_y$)	L_2-error
480	3.8584E–09
960	4.8229E–10
1680	8.9990E–11
3120	1.4049E–11

5. Conclusion

In this paper, a numerical method for solving the Burgers equation is proposed. The DIFST scheme is used for the time discretization and compared with some other time discretization schemes. The conforming finite element method (FEM) is used for the space discretization. The Newton's method and the BiCG methods are used for solving the resulting nonlinear and linear system of equations respectively. On the uniform grids, the unsteady solution is computed. Some examples with a known solution are numerically solved to calculate the L_2 error norm and CPU time. The computation for viscosity parameter effects is also given and it is observed that with the increase in the value of viscosity parameter ν, the diffusion occurs in the solution. The L_2 error estimate is also calculated and it is noted that L_2 error decreases with the increase in the viscosity parameter. A comparison for different time integration schemes is provided to show that the DIFST scheme is more computationally efficient and stable.

References

[1] Atashyar A, Khalsaraei MM, Shokri A. A new two-step hybrid singularly P-stable method for the numerical solution of second-order IVPs with oscillating solutions. Iranian Journal of Mathematical Chemistry 2020; 11 (2): 113-132.

[2] Shokri A, Tahmourasi M. A new two-step Obrechkoff method with vanished phase-lag and some of its derivatives for the numerical solution of radial Schrödinger equation and related IVPs with oscillating solutions. Iranian Journal of Mathematical Chemistry 2017; 8 (2): 137-159.

[3] Arabshahi MM, Golbabi A, Nikan O. Numerical approximation of time fractional advection-dispersion model arising from arising from solute transport in rivers. Turkic World Mathematical Society Journal of Pure and Applied Mathematics 2019; 10 (1): 117-131.

[4] Xi G, Zhang W, Zhang CH. An explicit Chebyshev pseudospectral multigrid method for incompressible Navier–Stokes equations. Computers and Fluids 2010; 39 (1): 178-188.

[5] Crighton DG. Model equations of nonlinear acoustics. Annual Review of Fluid Mechanics 1979; 11 (1): 11-33.

[6] Bateman H. Some recent researches on the motion of fluids. Monthly Weather Review 1915; 43 (4): 163-170.

[7] Toro EF. Reimann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction. Berlin, Germany: Springer-Verlag, 1997.

[8] Buck JA, Hayt WH. Engineering Electromagnetics, Vol. 6. New York, NY, USA: McGraw-Hill, 1981, pp. 445-446.

[9] Ozisik MN. Heat Transfer: A Basic Approach, Vol. 1. New York, NY, USA: McGraw-Hill, 1985.

[10] Metzner AB, Seyer FA. Turbulence phenomena in drag reducing systems. American Institute of Chemical Engineers 1969; 15 (3): 426-434.

[11] Bahadır AR, Sağlam M. A mixed finite difference and boundary element approach to one-dimensional Burgers' equation. Applied Mathematics and Computation 2005; 160 (3): 663-673.

[12] Jiwari R, Mittal RC. Differential quadrature method for two-dimensional Burgers' equations. International Journal for Computational Methods in Engineering Science and Mechanics 2009; 10 (6): 450-459.

[13] Kutluay S, Bahadir AR, Özdeş A. Numerical solution of one-dimensional Burgers equation: explicit and exact-explicit finite difference methods. Journal of Computational and Applied Mathematics 1999; 103 (2): 251-261.

[14] Bahadır AR, Saglam M. A mixed finite difference and boundary element approach to one-dimensional Burgers' equation. Applied Mathematics and Computation 2005; 160 (3): 663-673.

[15] Bhattacharya MC. Finite-difference solutions of partial differential equations. Communications in Applied Numerical Methods 1990; 6 (3): 173-184.

[16] Shih T, Qin Y. A method for estimating grid-induced errors in finite-difference and finite-volume methods. In: 41st Aerospace Sciences Meeting and Exhibit; Reno, NV, USA; 2003. p. 845.

[17] Kutluay S, Esen A, Dag I. Numerical solutions of the Burgers' equation by the least-squares quadratic B-spline finite element method. Journal of Computational and Applied Mathematics 2004; 167 (1): 21-33.

[18] Alford RM, Boore DM, Kelly KR. Accuracy of finite-difference modeling of the acoustic wave equation. Geophysics 1974; 39 (6): 834-842.

[19] Allen M, Cahn JW. A microscopic theory for antiphase boundary motion and its application to its application to antiphase domain coarsening. Acta Materialia 1979; 27: 1085-1095.

[20] Bastain P, Shah A, Sabir M. An efficient time-stepping scheme for numerical simulation of dendritic crystal growth. European Journal of Computational Mechanics 2017; 25 (6): 475-488.

[21] Bastian P, Birken K, Johannsen K, Lang S, Neuß N et al. UG – a flexible software toolbox for solving partial differential equations. Computing and Visualization in Science 1997; 1 (1): 27-40.

[22] Bastian P, Heimann F, Marnach S. Generic implementation of finite element methods in the distributed and unified numerics environment (DUNE). Kybernetika 2010; 46(2): 294-315.

[23] Chen Y, Zhang T. A weak Galerkin finite element method for Burgers' equation. Journal of Computational and Applied Mathematics 2019; 348 (1): 103-119.

[24] Aliev S, Gadjiev T, Galandarova S. A priori estimates for solutions to Dirichlet boundary value problems for polyharmonic equations in generalized Morrey spaces. Turkic World Mathematical Society Journal of Pure and Applied Mathematics 2018; 9 (2): 231-242.

[25] Wang Y, Li J. Phase field modeling of defects and deformation. Acta Materialia 2010; 58: 1212-1235.

[26] Smith JT. Conservative modeling of 3-D electromagnetic fields, Part II: biconjugate gradient solution and an accelerator. Geophysics 1996; 61 (5): 1319-1324.

[27] Ferracina L, Spijker M. An extension and analysis of the Shu-Osher representation of Runge-Kutta methods. Mathematics of Computation 2005; 74 (249): 201-219.

[28] Gottlieb S, Ketcheson DI, Shu CW. Strong stability preserving Runge-Kutta and multistep time discretizations. Singapore: World Scientific, 2011.

[29] Shu CW, Osher S. Efficient implementation of essentially non-oscillatory shock-capturing schemes. Journal of Computational Physics 1988; 77 (2): 439-471.

[30] Ding M, Shu H, Zhu H. Numerical solutions of two-dimensional Burgers' equations by discrete Adomian decomposition method. Computers and Mathematics with Applications 2010; 60 (3): 840-848.