Differ Equ Dyn Syst (April 2017) 25(2):169–186
DOI 10.1007/s12591-016-0318-6

ORIGINAL RESEARCH

A Higher Order Numerical Implicit Method for Non-Linear Burgers' Equation

Vijitha Mukundan[1] · **Ashish Awasthi**[1]

Published online: 20 August 2016

Abstract This paper proposes a higher order implicit numerical scheme to approximate the solution of the nonlinear partial differential equation (PDE). This equation is a simplified form of Navier–Stoke's equation also known as Burgers' equation. It is an important nonlinear PDE which arises frequently in mathematical modeling of turbulence in fluid dynamics. In order to handle nonlinearity a nonlinear transformation is used which converts the nonlinear PDE into a linear PDE. The linear PDE is semi-discretized in space by method of lines to yield a system of ordinary differential equations in time. The resulting system of differential equations is investigated and found to be a stiff system. A system of stiff differential equations is further discretized by a low-dispersion and low-dissipation implicit Runge–Kutta method and solved by using MATLAB 8.0. The proposed scheme is unconditionally stable. Moreover it is simple, easy to implement and requires less computational time. Finally, the adaptability of the scheme is demonstrated by means of numerical computations by taking three test problems. The present implicit scheme have been compared with existing schemes in literature which shows that the proposed scheme offers more accuracy with less computational time than the numerical schemes given in Jiwari (Comput Phys Comm 183:2413–2423, 2012), Kutluay et al. (J Comput Appl Math 103:251–261, 1998), Kutluay et al. (J Comput Appl Math 167:21–33, 2004).

Keywords Burgers' equation · Cole–Hopf transformation · Finite differences · Runge–Kutta method · Method of lines · Kinematic viscosity

Mathematics Subject Classification 65N06 · 65N40 · 65L05

✉ Ashish Awasthi
aawasthi@nitc.ac.in

Vijitha Mukundan
vijithamukundan29@gmail.com

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

Introduction

Burgers' equation is an important nonlinear partial differential equation from fluid mechanics which not only describes various phenomena such as mathematical model of turbulence, but also has got various applications in many fields of science and engineering. We consider the following nonlinear partial differential equation

$$u_t + uu_x = \nu u_{xx}, \quad x \in [0, 1] \quad \text{and} \quad t \in [0, T] \tag{1a}$$

with initial condition

$$u(x, 0) = u_0(x), \tag{1b}$$

and boundary conditions

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

where $\nu > 0$ is the kinematic viscosity parameter and $u_0(x)$ is given sufficiently smooth function. This equation appears in various areas of applied mathematics such as modeling of fluid dynamics, boundary layer behavior, turbulence, gas dynamics and shock wave formation. Burgers' equation can be considered as an analogue to Navier–Stoke's equation as this single equation has convection term, diffusion term and time dependent term. Solution of this equation exhibit a delicate balance between the nonlinear advection and the diffusion terms. Moreover it is considered as a test problem for validating several numerical algorithms. In 1915, Bateman [3] introduced Burgers' equation and gave its steady state solutions. It was later treated by Johannes Martinus Burgers who considered it as a mathematical model of turbulence [6] and studied its various aspects [6,7]. Benton and Platzman [4] surveyed the exact solutions of the one-dimensional Burgers' equation. Since various problems can be modeled through the Burgers' equation, many researchers have shown enormous interest in its solution.

So far, various numerical algorithms such as Galerkin finite element method [32], automatic differentiation [2], meshless method of lines [16], Crank–Nicolson scheme [19], finite element method [31], group explicit method [11] Haar wavelet quasilinearization approach [18], explicit and exact-explicit finite difference methods [21], lattice Boltzmann method [14], homotopy analysis method [27] have been developed. Kadalbajoo, Sharma and Awasthi [20] used a parameter-uniform implicit difference scheme for solving time dependent Burgers' equation. Bhatti and Bhatta [5] used Galerkin formulation of Burgers' equation and matrix formulation in which system of equations in time variable is solved using Runge–Kutta method of order four. Analytical methods for solving Burgers' equation is given in [28], while an explicit analytical solution of generalised Burger and Burger–Fisher equations using Homotopy perturbation method is discussed in [26]. Arora and Singh [1] have developed numerical solution of Burgers' equation with modified cubic B-spline differential quadrature method while Ganaie and Kukreja [13] used cubic Hermite collocation method to solve Burgers' equation. Mukundan and Awasthi [24] presented a comparative study of three level explicit and implicit numerical scheme for Burgers' equation. In [23], authors used variational iteration method to solve fractional coupled Burgers' equation. A detailed survey of various techniques to solve Burgers' equation both numerically and analytically is discussed in [10]. In 2015, Mukundan and Awasthi [25] introduced some efficient numerical techniques to solve one dimensional Burgers' equation. In 2016, Guo et al. [15] proposed higher order compact scheme based on finite volume method to solve Burgers' equation. Three-dimensional coupled viscous Burgers' equation is solved by modified cubic B-spline differential quadrature method [33].

In this paper, Burgers' equation is solved numerically by combinations of a nonlinear transformation, method of lines and implicit Runge–Kutta method. The nonlinear transformation reduces the Burgers' equation into a linear diffusion equation. The method of lines is a semi-discretization technique, introduced by Rothe [29] in 1930, in which discretization is performed along the spatial direction alone. The MOL technique transforms the linear diffusion equation into a system of first order ordinary differential equations. This system of equations is investigated and found to be highly stiff so that explicit methods cannot be employed to find its numerical solution. Hence the stiff system is solved by an implicit low-dispersion and low-dissipation Runge–Kutta method of order four. The proposed numerical scheme is explained in Sect. 2. Its stability is discussed in Sect. 3. Three test problems are presented in Sect. 4 to demonstrate the applicability and accuracy of the proposed numerical scheme. Conclusions are drawn in Sect. 5.

Numerical Scheme

In this paper, we propose a hybrid scheme which comprises of nonlinear transformation, method of lines and low-dispersion and low-dissipation implicit Runge–Kutta method of order four.

Nonlinear Transformation

The given nonlinear parabolic one-dimensional Burgers' equation Eq. (1) with Dirichlet boundary conditions is linearized into diffusion equation

$$\phi_t = \nu\phi_{xx} \tag{2a}$$

with initial condition

$$\phi(x,0) = \exp\left(-\frac{1}{2\nu}\int_0^x u_0(\xi)d\xi\right), \quad 0 \le x \le 1, \tag{2b}$$

and boundary conditions

$$\phi_x(0,t) = 0 = \phi_x(1,t), \quad t \ge 0. \tag{2c}$$

by using the nonlinear transformation, also known as Cole–Hopf transformation [9,17]

$$u = -2\nu\frac{\phi_x}{\phi}. \tag{3}$$

The linear diffusion equation Eq. (2) is semi-discretized in the spatial direction by method of lines. This procedure leads to a stiff system of ordinary differential equations in the time variable 't' which can be integrated in time.

Method of Lines

The linear diffusion equation Eq. (2) is semi-discretized in the spatial variable by method of lines (MOL). The spatial domain [0, 1] is divided into N equal subintervals, $0 = x_0 \le x_1 \le \cdots \le x_N = 1$ with constant spacing $h = (1)/N$ and $x_i = ih$ for $i = 0, 1, 2, \ldots N$. In MOL, the second order spatial derivative ϕ_{xx} approximated as follows

$$\phi_{xx}(x_i,t) = \frac{1}{h^2}(\phi_{i+1}(t) - 2\phi_i(t) + \phi_{i-1}(t)), \quad i = 0, 1, 2, \ldots N.$$

where $\phi_i(t) = \phi(x_i, t)$, $i = 0, 1, 2, \ldots, N$.

Substituting in Eq. (2), we obtain a system of ordinary differential equations with initial condition

$$\frac{d\phi_i}{dt}(t) = \frac{\nu}{h^2}(\phi_{i+1}(t) - 2\phi_i(t) + \phi_{i-1}(t))$$
$$\phi_i(0) = \exp\left(-\frac{1}{2\nu}\int_0^{x_i} u_0(\xi)d\xi\right), \quad i = 0, 1, 2, \ldots N$$

where, $\phi_i(t) = \phi(x_i, t)$, taking into account the discretization of boundary condition $\phi_{-1}(t) = \phi_1(t)$ and $\phi_{N+1}(t) = \phi_{N-1}(t)$, we obtain a system of ordinary differential equations which can be written in vector form

$$\frac{d\Phi}{dt} = \frac{\nu}{h^2} S\Phi(t), \quad \Phi(0) = \Phi_{\mathbf{0}} \tag{4}$$

where, $\Phi(t) = [\phi_0(t), \ldots, \phi_N(t)]^T$, $\Phi_{\mathbf{0}} = [\phi_0(0), \ldots, \phi_N(0)]^T$ is the initial condition and S is $(N+1) \times (N+1)$ tridiagonal matrix given by

$$S = \begin{pmatrix} -2 & 2 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ & & & 2 & -2 \end{pmatrix}$$

Stiff Differential Equations

The system of ordinary differential equations Eq. (4) is a stiff system. A system of equations with constant coefficients

$$\frac{d\Phi}{dt} = P\Phi$$

where P is a $(N+1) \times (N+1)$ constant matrix and $\Phi(t) = [\phi_0(t), \ldots, \phi_N(t)]^T$, is called a stiff system if the range of the magnitudes of the eigenvalues is large ($|\lambda|_{max}/|\lambda|_{min} \gg 1$) and the solution is desired over a large span of the independent variable t. Stiff systems are associated with numerical difficulties which can be overcome by using implicit methods. With implicit methods there is no restriction on the time step due to numerical stability. Stiffness of the system Eq. (4) can be proved as follows.

The system of ordinary differential equations Eq. (4) is proved a stiff system by finding all eigenvalues of the matrix S . The eigenvalues λ_i of S in Eq. (4) are given by

$$Det(S - \lambda I) = 0, \tag{5}$$

after simplification we get,

$$\left((\lambda + 2)^2 - 4)\right) Q_{N-1}(\lambda) = 0 \tag{6}$$

where

$$Q_N(\lambda) = \begin{vmatrix} -(2+\lambda) & 1 & & & & \\ 1 & -(2+\lambda) & 1 & & & \\ . & . & . & & & \\ & . & . & . & & \\ & & . & . & 1 & \\ & & 1 & -(2+\lambda) & 1 & \\ & & & 1 & -(2+\lambda) & \end{vmatrix}$$

$$= (-1)^N \prod_{i=1}^{N} \left(4\cos^2 \frac{\pi i}{2(N+1)} + \lambda\right) \quad (7)$$

using Eq. (7) in Eq. (6), we have

$$(-1)^{N-1}\left((\lambda+2)^2-4\right) \prod_{i}^{N-1} \left(4\cos^2 \frac{\pi i}{2N} + \lambda\right) = 0$$

$$\lambda(\lambda+2) \prod_{j=1}^{N-1} \left(4\cos^2 \frac{\pi i}{2N} + \lambda\right) = 0,$$

which gives

$$\lambda = 0, \quad \text{or} \quad \lambda = -2 \quad \text{or}$$

$$\lambda_i = -4\cos^2 \frac{\pi i}{2N}, \quad i = 0, 1, 2, \ldots, N.$$

$$|\lambda|_{min} = 0, \quad \text{and} \quad |\lambda|_{max} = 4$$

Hence, we have

$$|\lambda|_{max}/|\lambda|_{min} \gg 1$$

Therefore, the system of ordinary differential equations Eq. (4) is a stiff system. The variation of stiffness ratio with nodal points is given in Fig. 1. Since the stiffness ratio is very high , the ode is a stiff system where explicit methods fails to provide a numerical solution.

Discretization of Time Variable

The system of stiff differential equations is discretized in time variable by a low-dispersion and low-dissipation implicit Runge–Kutta method of order four. The time interval $[0, T]$ is

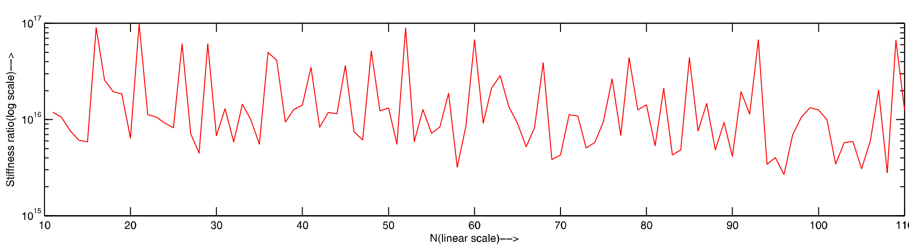

Fig. 1 Variation of stiffness ratio with nodal points(N)

divided into M equal subintervals $0 = t_0 \leq t_1 \leq \cdots \leq t_M = T$ with $\Delta t = T/M$, i.e $t_n = n\Delta t, n = 0, 1, 2, \ldots, M$.

General form of p stage Runge–Kutta method to solve a system of ordinary differential equation of the form

$$\frac{d\Phi}{dt} = \mathbf{F}(\Phi, t), \quad \Phi(0) = \Phi_{\mathbf{0}}$$

is given by

$$\Phi^{n+1} = \Phi^n + \sum_{j=1}^{p} b_j K_{ij}, \quad i = 0, 1, \ldots, N, \quad n = 0, 1, \ldots, M-1 \tag{8}$$

where,

$$K_{ij} = \Delta t \mathbf{F}(\Phi^n + \sum_{\beta=1}^{p} a_{j\beta} K_{i\beta}, t_n + c_j \Delta t), \quad i = 0, 1, \ldots, N, \quad j = 1, \ldots, p. \tag{9}$$

where, $\Phi^n = [\phi_0^n, \phi_1^n, \ldots, \phi_N^n]^T$, $\mathbf{F}(\Phi, t) = \frac{\nu}{h^2} S\Phi(t)$, Δt is the time step, b_j, c_j and $a_{j\beta}$ are constants for, $j = 1, \ldots, p$.

Butcher [8] gave another notation to represent a p stage Runge–Kutta scheme in a coefficient table,

$$\begin{array}{c|c} \mathbf{c} & \mathbf{A} \\ \hline & \mathbf{b}^T \end{array} \tag{10}$$

where $\mathbf{c} = (c_1, c_1, \ldots, c_p)$ represents position of stage values within the time step, $\mathbf{b} = (b_1, b_1, \ldots, b_p)$ represents weight coefficients, in Eq. (8) and matrix $\mathbf{A}$ is the matrix of $a_{j\beta}$ as given in Eq. (9)

We choose a three stage fourth order low-dispersion and low-dissipation implicit Runge–Kutta scheme presented in [30]. This scheme is more accurate than the standard fourth-order explicit RK scheme and three-stage fourth-order singly diagonal implicit Runge–Kutta scheme.

In the present scheme matrix $\mathbf{A}$ is given by

$$\mathbf{A} = \begin{pmatrix} a_{11} & 0 & 0 \\ a_{21} & a_{22} & 0 \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$$

values of $\mathbf{A}$, $\mathbf{b}$, $\mathbf{c}$ of Eq. (10) are constants as given in Table 1.

Finally, numerical solution of the Burgers' Equation Eq. (1) in terms of numerical solution of the linear diffusion equation Eq. (2) and nonlinear transformation Eq. (3) is given by

$$\begin{aligned} u_i^n &= -(2\nu) \left\{ \frac{\phi_{i+1}^n - \phi_{i-1}^n}{2h\phi_i^n} \right\} \\ &= -\left(\frac{\nu}{h}\right) \left\{ \frac{\phi_{i+1}^n - \phi_{i-1}^n}{\phi_i^n} \right\}, \quad i = 0, 1, \ldots, N, \quad n = 1, \ldots, M-1. \end{aligned} \tag{11}$$

where ϕ_i^n is the approximate value of $\phi(x, t)$ at $x = x_i$ and $t = t_n$.

Table 1 Coefficients for fourth order, implicit Runge–Kutta scheme

Parameter	Value
a_{11}	0.377847764031163
a_{21}	0.385232756462588
a_{22}	0.461548399939329
a_{31}	0.675724855841358
a_{32}	−0.061710969841169
a_{33}	0.241480233100410
b_1	0.750869573741408
b_2	−0.362218781852651
b_3	0.611349208111243
c_1	0.257820901066211
c_2	0.434296446908075
c_3	0.758519768667167

Stability

Implementing the three stage fourth order low-dispersion and low-dissipation implicit Runge–Kutta scheme to the system of stiff differential equations Eq. (4) we get,

$$\Phi^{n+1} = \Phi^n + b_1 K_{i1} + b_2 K_{i2} + b_3 K_{i3}, \quad i = 0(1)N, \quad n = 0(1)M \tag{12}$$

where, $K_{i1} = \frac{\Delta t \nu}{h^2} S(\Phi^n + a_{11} K_{i1})$, let $\lambda = \frac{\Delta t \nu}{h^2}$ so we obtain

$$K_{i1} = A_{11}^{-1}(\lambda S \Phi^n) \tag{13}$$

where $A_{11} = I - \lambda S a_{11}$

$$\begin{aligned} K_{i2} &= \lambda S(\Phi^n + a_{21} K_{i1} + a_{22} K_{i2}) \\ \implies K_{i2} &= A_{22}^{-1}(\lambda S \Phi^n) A_2 \end{aligned} \tag{14}$$

where $A_{22} = I - \lambda S a_{22}$ and $A_2 = I + \lambda a_{21} A_{11}^{-1} S$
and

$$\begin{aligned} K_{i3} &= \lambda S(\Phi^n + a_{31} K_{i1} + a_{32} K_{i2} + a_{33} K_{i3}) \\ \implies K_{i3} &= A_{33}^{-1}(\lambda S \Phi^n) A_3 \end{aligned} \tag{15}$$

where $A_{33} = I - \lambda S a_{33}$ and $A_3 = I + \lambda a_{31} A_{11}^{-1} S + \lambda a_{32} A_{22}^{-1} A_2 S$
Substituting Eqs. (13), (14) and (15) in Eq. (12) we get

$$\Phi^{n+1} = \Phi^n C, \quad n = 0(1)M \tag{16}$$

where C is a matrix of order $(N+1)X(N+1)$ given by

$$C = I + \lambda S(b_1 A_{11}^{-1} + b_2 A_{22}^{-1} A_2 + b_3 A_{33}^{-1} A_3) \tag{17}$$

Now we will investigate the eigenvalues of the matrix C.

In Sect. 2.3 we have shown that eigenvalues of the matrix S say (λ_i) is given by

$$\lambda_i = -4\cos^2 \frac{\pi i}{2N}, \quad i = 0, 1, 2, \ldots, N. \tag{18}$$

where S is real $(N+1)X(N+1)$ tridiagonal matrix.

Table 2 Comparison of the numerical solution with the exact solution at different space points of Example 1 at $T = 1$ for $\nu = 1$ and $\Delta t = 0.0001$

x	Computed solution			Exact solution
	$N = 60$	$N = 80$	$N = 100$	
0.1	0.000016	0.000016	0.000016	0.000016
0.2	0.000030	0.000030	0.000030	0.000030
0.3	0.000042	0.000042	0.000042	0.000042
0.4	0.000049	0.000049	0.000049	0.000049
0.5	0.000052	0.000052	0.000052	0.000052
0.6	0.000049	0.000049	0.000049	0.000049
0.7	0.000042	0.000042	0.000042	0.000042
0.8	0.000030	0.000030	0.000030	0.000030
0.9	0.000016	0.000016	0.000016	0.000016

Table 3 Comparison of the numerical solution with the exact solution at different space points of Example 1 at $T = 1$ for $\nu = 0.1$ and $\Delta t = 0.0001$

x	Computed solution			Exact solution
	$N = 60$	$N = 80$	$N = 100$	
0.1	0.06631	0.06631	0.06632	0.06632
0.2	0.13120	0.13121	0.13121	0.13121
0.3	0.19277	0.19278	0.19279	0.19279
0.4	0.24802	0.24804	0.24805	0.24804
0.5	0.29189	0.29191	0.29192	0.29192
0.6	0.31604	0.31606	0.31608	0.31607
0.7	0.30806	0.30809	0.30810	0.30809
0.8	0.25369	0.25372	0.25373	0.25372
0.9	0.14605	0.14607	0.14607	0.14607

Table 4 Comparison of exact solution and the computed solution at different times for Example 1 at $\nu = 0.05$, $\Delta x = 0.0125$ and $\Delta t = 0.0001$

x	T	Exact solution	Computed solution
0.25	1.0	0.18215	0.18214
	1.5	0.13261	0.13260
	2.0	0.10322	0.10322
0.50	1.0	0.35893	0.35893
	1.5	0.25900	0.25899
	2.0	0.19625	0.19624
0.75	1.0	0.48484	0.48481
	1.5	0.32167	0.32164
	2.0	0.21989	0.21987

Eigenvalues of the matrix A_{11} is given by (see [12], p. 18)

$$1 + 4\lambda a_{11} \cos^2 \frac{\pi i}{2N}, \quad i = 0, 1, 2, \ldots, N.$$

Similarly, eigenvalues of the matrix A_{22} is given by

$$1 + 4\lambda a_{22} \cos^2 \frac{\pi i}{2N}, \quad i = 0, 1, 2, \ldots, N.$$

and eigenvalues of the matrix A_{33} is given by

$$1 + 4\lambda a_{33} \cos^2 \frac{\pi i}{2N}, \quad i = 0, 1, 2, \ldots, N.$$

Table 5 Comparison between exact and existing numerical solutions of Example 1 for $\nu = 0.01$ at different times T and x

x	T	[22] $\Delta t = 0.0001$	[21] $\Delta t = 0.001$	Present method $\Delta t = 0.001$	Exact
0.25	0.4	0.34819	0.34244	0.34198	0.34191
	0.6	0.27536	0.26905	0.26897	0.26896
	0.8	0.22752	0.22145	0.22148	0.22148
	1.0	0.19375	0.18813	0.18819	0.18819
	3.0	0.07754	0.07509	0.07511	0.07511
0.50	0.4	0.66543	0.67152	0.66228	0.66071
	0.6	0.53525	0.53406	0.52999	0.52942
	0.8	0.44526	0.44143	0.43938	0.43914
	1.0	0.38047	0.37568	0.37454	0.37442
	3.0	0.15362	0.15020	0.15018	0.15018
0.75	0.4	0.91201	0.94675	0.91636	0.91026
	0.6	0.77132	0.78474	0.76964	0.76724
	0.8	0.65254	0.65659	0.64846	0.64740
	1.0	0.56157	0.56135	0.55658	0.55605
	3.0	0.22874	0.22502	0.22482	0.22481

Table 6 Comparison of the numerical solution with the exact solution at different space points of Example 2 at $T = 1$ for $\nu = 1$ and $\Delta t = 0.0001$

x	Computed solution			Exact solution
	$N = 40$	$N = 80$	$N = 100$	
0.1	0.000017	0.000016	0.000016	0.000016
0.2	0.000031	0.000031	0.000031	0.000031
0.3	0.000043	0.000043	0.000043	0.000043
0.4	0.000051	0.000051	0.000051	0.000051
0.5	0.000053	0.000053	0.000053	0.000053
0.6	0.000051	0.000051	0.000051	0.000051
0.7	0.000043	0.000043	0.000043	0.000043
0.8	0.000031	0.000031	0.000031	0.000031
0.9	0.000017	0.000016	0.000016	0.000016

Eigenvalues of the matrix A_2 is given by (see [12], Theorem 1.4, p. 18)

$$\frac{1-4\lambda a_{21}\cos^2\frac{\pi i}{2N}}{1+4\lambda a_{11}\cos^2\frac{\pi i}{2N}},\quad i=0,1,2,\ldots,N.$$

Similarly, eigenvalues of the matrix A_3 is given by

$$1-\frac{4\lambda a_{31}\cos^2\frac{\pi i}{2N}}{1+4\lambda a_{11}\cos^2\frac{\pi i}{2N}}-\frac{4\lambda a_{32}\cos^2\frac{\pi i}{2N}}{1+4\lambda a_{22}\cos^2\frac{\pi i}{2N}}\frac{1-4\lambda a_{21}\cos^2\frac{\pi i}{2N}}{1+4\lambda a_{11}\cos^2\frac{\pi i}{2N}},\quad i=0,1,2,\ldots,N.$$

The matrix C is given by

$$C=I+\lambda S(b_1A_{11}^{-1}+b_2A_{22}^{-1}A_2+b_3A_{33}^{-1}A_3) \tag{19}$$

Table 7 Comparison of the numerical solution with the exact solution at different space points of Example 2 at $T=4$ for $\nu=0.0125$ and $\Delta t=0.001$

x	Computed solution			Exact solution
	$N=20$	$N=30$	$N=40$	
0.1	0.02331	0.02332	0.02332	0.02332
0.2	0.04661	0.04663	0.04664	0.04663
0.3	0.06992	0.06994	0.06995	0.06994
0.4	0.09323	0.09325	0.09326	0.09325
0.5	0.11650	0.11652	0.11653	0.11651
0.6	0.13956	0.13958	0.13959	0.13957
0.7	0.16123	0.16134	0.16137	0.16138
0.8	0.17465	0.17510	0.17526	0.17542
0.9	0.14499	0.14615	0.14657	0.14705

Table 8 Comparison of exact solution and the computed solution at different times for Example 2 at $\nu=0.1$, $\Delta x=0.0125$ and $\Delta t=0.001$

x	T	Exact solution	Computed solution
0.25	2.0	0.06951	0.06957
	2.5	0.04425	0.04429
	3.0	0.02776	0.02779
0.50	2.0	0.11020	0.11031
	2.5	0.06727	0.06734
	3.0	0.04106	0.04111
0.75	2.0	0.08868	0.08878
	2.5	0.05142	0.05148
	3.0	0.03044	0.03048

Table 9 Comparison between exact and existing numerical solutions of Example 2 for $\nu = 0.1$ at different times T and x

x	T	[22] $\Delta t = 0.0001$	[18] $\Delta t = 0.001$	Present method $\Delta t = 0.001$	Exact
0.25	0.4	0.32091	0.30887	0.31751	0.31752
	0.6	0.24910	0.24609	0.24613	0.24614
	0.8	0.20211	0.19952	0.19955	0.19956
	1.0	0.16782	0.16557	0.16560	0.16560
	3.0	0.02828	0.02775	0.02776	0.02775
0.50	0.4	0.58788	0.56979	0.58452	0.58454
	0.6	0.46174	0.45790	0.45796	0.45798
	0.8	0.37111	0.36734	0.36739	0.36740
	1.0	0.30183	0.29829	0.29834	0.29834
	3.0	0.04185	0.04105	0.04107	0.04106
0.75	0.4	0.65054	0.62567	0.64560	0.64562
	0.6	0.50825	0.48715	0.50266	0.50268
	0.8	0.39068	0.38525	0.38532	0.38534
	1.0	0.30057	0.29578	0.29585	0.29586
	3.0	0.03106	0.03043	0.03044	0.03044

Table 10 Comparison of computed solution with the exact solution at different times and spatial points for Example 3 at $\nu = 0.0125, 0.01$, $\Delta x = 0.0125$ and $\Delta t = 0.001$

T	x	$\nu = 0.0125$		$\nu = 0.01$	
		Computed solution	Exact solution	Computed solution	Exact solution
1	0.25	0.01870	0.01870	0.01524	0.01524
	0.50	0.03470	0.03471	0.02846	0.02846
	0.75	0.03570	0.03570	0.02961	0.02961
2	0.25	0.01700	0.01700	0.01413	0.01413
	0.50	0.03068	0.03068	0.02578	0.02579
	0.75	0.02997	0.02998	0.02569	0.02569
3	0.25	0.01541	0.01541	0.01308	0.01308
	0.50	0.02712	0.02712	0.02336	0.02336
	0.75	0.02537	0.02537	0.02241	0.02242

Table 11 Errors in terms of L_2 norm and L_∞ norm at $\nu = 1, 0.1, 0.02, 0.005$, $T = 1, 2, 3$ and $\Delta t = 0.001$, corresponding to example 3

ν	T = 1		T = 2		T = 3	
	L_2	L_∞	L_2	L_∞	L_2	L_∞
1	1.25725E-06	1.77802E-06	7.26719E-11	1.02783E-10	1.38427E-14	3.50952E-14
0.1	7.49648E-05	1.09803E-04	3.05870E-05	4.34716E-05	1.27895E-05	1.80961E-05
0.02	2.14782E-06	3.54995E-06	1.54623E-06	2.87520E-06	1.33643E-06	2.55146E-06
0.005	2.21117E-06	3.19673E-06	2.01722E-06	2.90996E-06	1.84120E-06	2.65184E-06

Eigenvalues of the matrix C say (λ_{C_i}) is given by (see [12], p. 18)

$$\lambda_{C_i} = 1 - 4\lambda\cos^2\frac{\pi i}{2N}\left(\frac{b_1}{1+4\lambda a_{11}\cos^2\frac{\pi i}{2N}} + \frac{b_2}{1+4\lambda a_{22}\cos^2\frac{\pi i}{2N}}\,\frac{1-4\lambda a_{21}\cos^2\frac{\pi i}{2N}}{1+4\lambda a_{11}\cos^2\frac{\pi i}{2N}} + \frac{b_1}{1+4\lambda a_{33}\cos^2\frac{\pi i}{2N}} \times\left(1 - \frac{4\lambda a_{31}\cos^2\frac{\pi i}{2N}}{1+4\lambda a_{11}\cos^2\frac{\pi i}{2N}} - \frac{4\lambda a_{32}\cos^2\frac{\pi i}{2N}}{1+4\lambda a_{22}\cos^2\frac{\pi i}{2N}}\,\frac{1-4\lambda a_{21}\cos^2\frac{\pi i}{2N}}{1+4\lambda a_{11}\cos^2\frac{\pi i}{2N}}\right)\right), \quad i = 0(1)N.$$

All the eigenvalues are clearly less than one for all values of $\lambda > 0$ as $b_2 < 0$ and $a_{32} < 0$. Hence the fully discretized scheme is unconditionally stable for all values of $\lambda > 0$.

Numerical Computations

In order to demonstrate the efficiency of proposed numerical method several numerical experiments are carried out. Numerical method adopted in this paper is a combination of Cole–Hopf transformation, method of lines and implicit Runge–Kutta method. Numerical results gener-

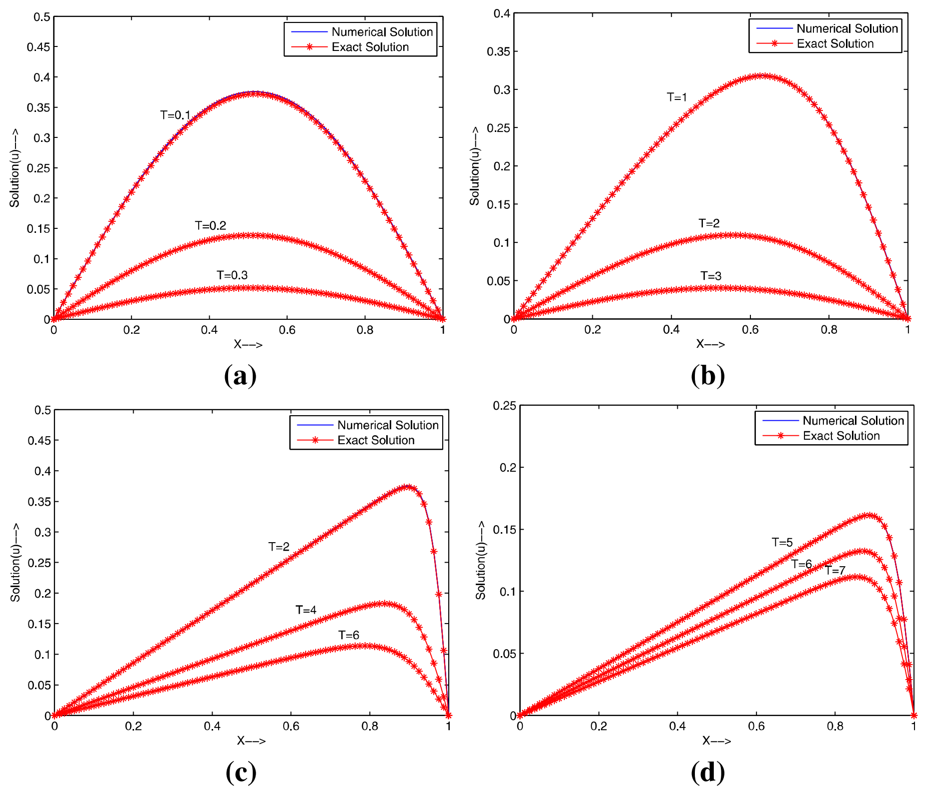

Fig. 2 Numerical solutions of Example 1 at different times for $\Delta x = 0.0125$ and different values of ν and Δt, **a** $\nu = 1$, $\Delta t = 0.0001$, **b** $\nu = 0.1$, $\Delta t = 0.01$, **c** $\nu = 0.01$, $\Delta t = 0.01$ and **d** $\nu = 0.005$, $\Delta t = 0.01$

ated by proposed method are compared with exact. Implementation of the numerical method has been carried out using MATLAB-8.0.

Example 1 Consider the Burgers' Eq. (1) with the initial condition

$$u(x, 0) = \sin(\pi x), \quad 0 \leq x \leq 1, \tag{20}$$

and the homogeneous boundary conditions

$$u(0, t) = u(1, t) = 0, \quad 0 \leq t \leq T. \tag{21}$$

Burgers' equation is converted to linear diffusion equation by means of Cole–Hopf transformation, which is then solved by the method of separation of variables. The exact solution of the problem is

$$u(x, t) = 2\pi\nu \frac{\sum_{n=1}^{\infty} C_n \exp\left(-n^2\pi^2\nu t\right) n \sin(n\pi x)}{C_0 + \sum_{n=1}^{\infty} C_n \exp\left(-n^2\pi^2\nu t\right) \cos(n\pi x)} \tag{22}$$

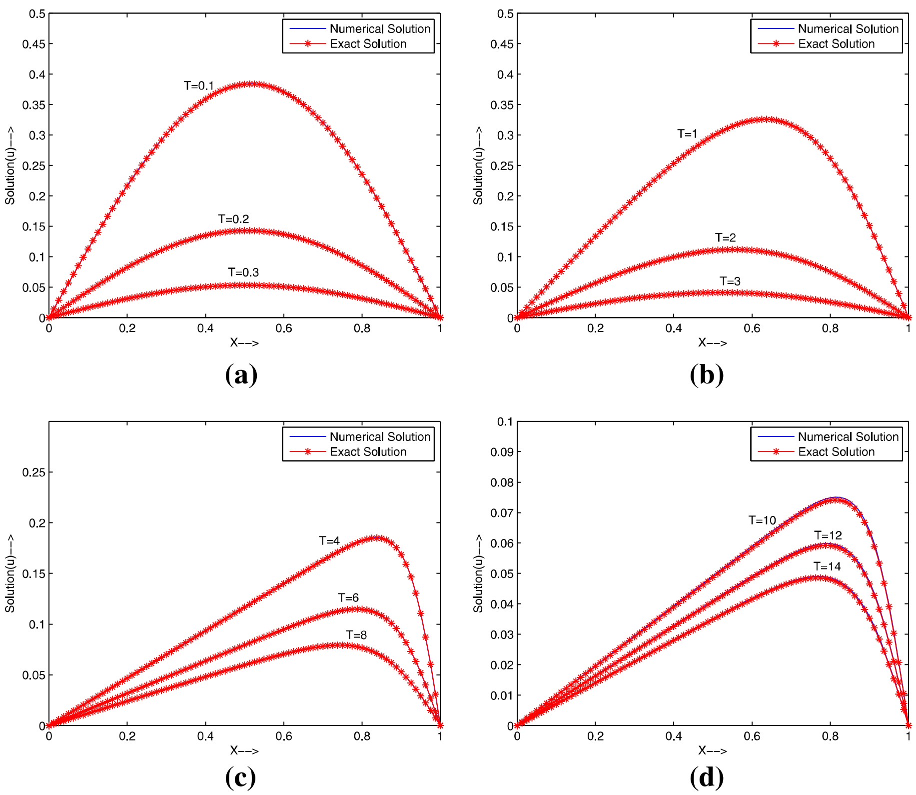

Fig. 3 Numerical solutions of Example 2 at different times for $\Delta x = 0.0125$ and different values of ν and Δt, **a** $\nu = 1$, $\Delta t = 0.0001$, **b** $\nu = 0.1$, $\Delta t = 0.001$, **c** $\nu = 0.01$, $\Delta t = 0.01$ and **d** $\nu = 0.005$, $\Delta t = 0.1$

where

$$C_0 = \int_0^1 \exp\left\{-\frac{1}{2\pi\nu}[1-\cos(\pi x)]\right\} dx, \tag{23}$$

$$C_n = 2\int_0^1 \exp\left\{-\frac{1}{2\pi\nu}[1-\cos(\pi x)]\right\}\cos(n\pi x)\, dx, \tag{24}$$

Example 2 Next, we consider Burgers' equation with the following initial condition

$$u(x,0) = 4x(1-x), \quad 0 \le x \le 1, \tag{25}$$

and boundary condition

$$u(0,t) = 0 = u(1,t), \quad 0 \le t \le T. \tag{26}$$

The exact solution of the problem is

$$u(x,t) = 2\pi\nu \frac{\sum_{n=1}^{\infty} D_n \exp\left(-n^2\pi^2\nu t\right) n \sin(n\pi x)}{D_0 + \sum_{n=1}^{\infty} D_n \exp\left(-n^2\pi^2\nu t\right)\cos(n\pi x)} \tag{27}$$

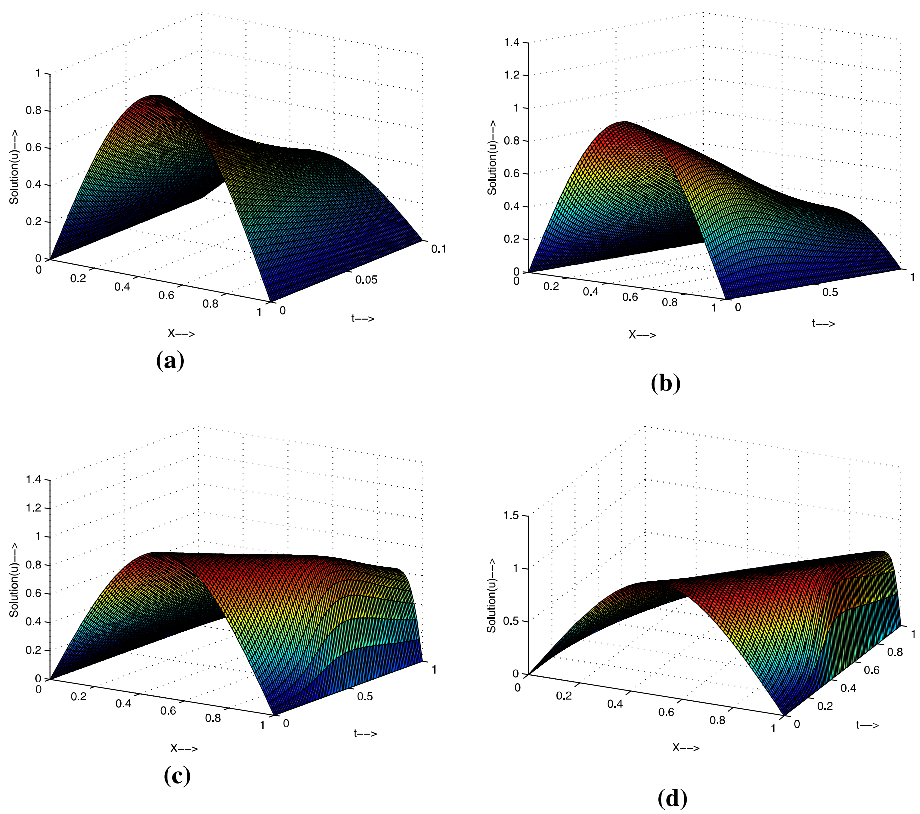

Fig. 4 Numerical solution profile for Example 1 **a** $\nu = 1$, $\Delta t = 0.001$; **b** $\nu = 0.1$, $\Delta t = 0.01$; **c** $\nu = 0.02$, $\Delta t = 0.01$ and **d** $\nu = 0.01$, $\Delta t = 0.01$

where we note that the Fourier coefficients D_0 and D_n are the following

$$D_0 = \int_0^1 \exp\left\{-\frac{1}{3\nu}[x^2(3-2x)]\right\} dx, \tag{28}$$

$$D_n = \int_0^1 \exp\left\{-\frac{1}{3\nu}[x^2(3-2x)]\right\} \cos(n\pi x)\, dx. \tag{29}$$

Example 3 We also consider Burgers' equation

$$u_t + uu_x = \nu u_{xx},\ x \in [0, 1] \quad \text{and} \quad t \in [0, T] \tag{30a}$$

on $0 \le x \le 1$ with the boundary conditions

$$u(0, t) = 0 = u(1, t), \qquad 0 \le t \le T. \tag{30b}$$

and initial condition

$$u(x, 0) = \frac{2\nu\pi \sin(\pi x)}{2 + \cos(\pi x)}, \qquad 0 \le x \le 1, \tag{30c}$$

The exact solution for this problem is

$$u(x, t) = \frac{2\nu\pi \exp(-\pi^2 \nu t) \sin(\pi x)}{2 + \exp(-\pi^2 \nu t) \cos(\pi x)} \tag{31}$$

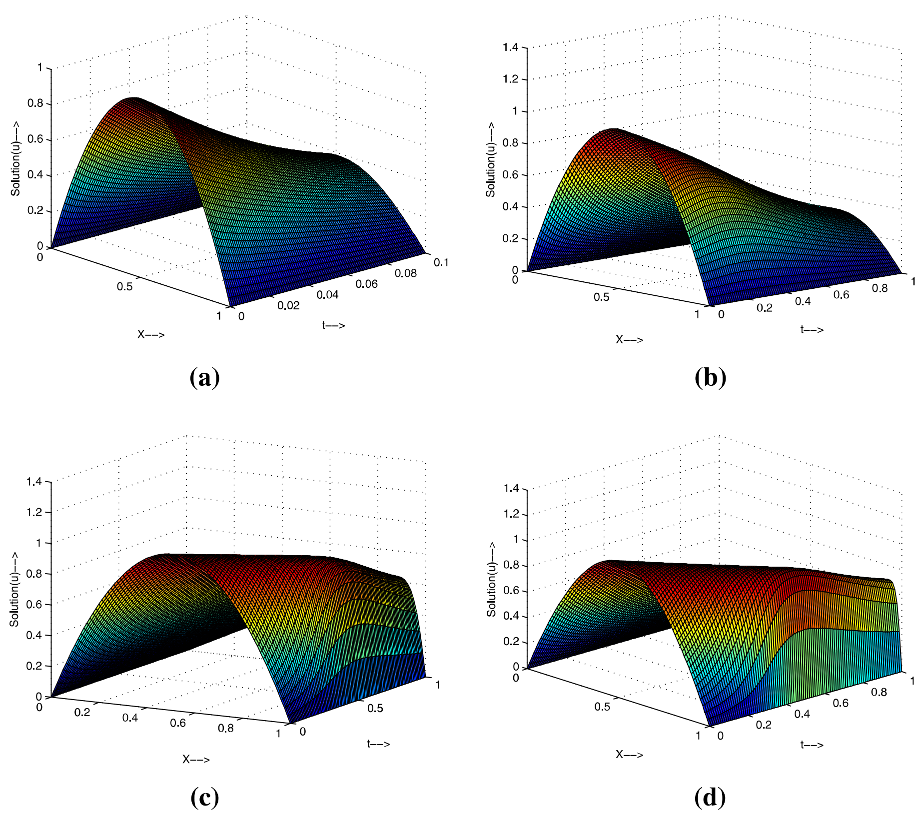

Fig. 5 Numerical solution profile for Example 2 **a** $\nu = 1$, $\Delta t = 0.001$; **b** $\nu = 0.1$, $\Delta t = 0.01$; **c** $\nu = 0.02$, $\Delta t = 0.01$ and **d** $\nu = 0.01$, $\Delta t = 0.01$

A comparison of numerical results obtained by the proposed numerical method with the exact solution for Example 1 has been presented in Tables 2–4. In Table 2, results are tabulated at different space points for $\nu = 1$, $\Delta t = 0.0001$ and $T = 1$, while in Table 3, $\nu = 0.1$, $\Delta t = 0.0001$ and $T = 1$. Numerical solutions are computed at different spatial points and time levels in Table 4. The proposed numerical scheme is compared with existing numerical schemes which include explicit finite difference method, least-squares quadratic B-spline finite element method and Haar wavelet method. The numerical results computed using the proposed scheme for $N = 180$ are tabulated in Table 5 and compared with existing schemes and it indicates that present numerical method is better than the schemes given in [21,22].

Comparison of numerical and exact solution of Example 2 at different space points for $\nu = 1$, $\Delta t = 0.0001$, $T = 1$ and $\nu = 0.0125$, $\Delta t = 0.001$, $T = 4$ are tabulated in Tables 6–7. Table 8 shows the comparison between numerical and exact solutions at different times levels and at different spatial points. Tabulated results indicates that numerical results presented in this paper lies very close to the exact solution and the accuracy increases as the number of nodal points increases. Table 9 shows comparison between proposed numerical method with $N = 180$ and existing numerical schemes in [18,22]. The numerical schemes mentioned in [22] give solution for very small Δt ($\Delta t = 0.0001$) while the present scheme provide better solution even for $\Delta t = 0.001$. Hence, very small value of Δt is not required for computing the numerical solution which in turn reduces the computational time. Most of the numerical schemes fails to capture the physical behavior of solution for very small value of kinematic viscosity. In Example 3 we repeated our experiment with small value of kinematic viscosity. In Table 10, a comparison between computed solution and exact solution is presented for $\nu = 0.0125, 0.01$ at time $T = 1, 2, 3$, $\Delta x = 0.0125$ and $\Delta t = 0.001$. From the Table it is evident that even for small value of kinematic viscosity there is excellent agreement between the exact and computed solutions. We computed L_2 and L_∞ errors at $T = 1, 2, 3$ and kinematic viscosity $\nu = 1, 0.1, 0.02, 0.005$. Results are reported in Table 11. It shows that the present numerical scheme is accurate even for small value of kinematic viscosity.

Figures, 2 and 3, indicate that the numerical results obtained by the proposed method are comparatively in good agreement with analytical solution for modest values of kinematic viscosity, ν. In order to show the physical behavior of the given problem, we give surface plots of the computed solutions for different values of kinematic viscosity, ν in Figs. 4, 5.

Conclusions

The nonlinear Burgers' equation is reduced to linear diffusion equation by using a nonlinear transformation. The linear diffusion equation is then semi-discretized in variable 'x' by method of lines which results in a stiff system of first order ordinary differential equations. This stiff system of ordinary differential equations is solved by low-dispersion and low-dissipation implicit Runge–Kutta method of order four. Numerical results are compared with exact solutions at different times, for modest values of kinematic viscosity. Tables and Figures indicate that computed results are reasonably in good agreement with the exact solution. Comparison with the existing numerical schemes shows that the numerical scheme introduced in this paper is efficient than the schemes given in [18,21,22]. This method is fourth order accurate in time and second order accurate in space.

References

1. Arora, G., Singh, B.K.: Numerical solution of Burgers' equation with modified cubic B-spline differential quadrature method. Appl. Math. Comput. **224**, 166–177 (2013)
2. Asaithambi, A.: Numerical solution of the Burgers equation by automatic differentiation. Appl. Math. Comput. **216**, 2700–2708 (2010)
3. Bateman, H.: Some recent researches in motion of fluids. Mon. Weather Rev. **43**, 163–170 (1915)
4. Benton, E.R., Platzman, G.W.: A table of solutions of the one-dimensional Burgers' Equation. Q. Appl. Math. **30**, 195–212 (1972)
5. Bhatti, M.I., Bhatta, D.D.: Numerical solution of Burgers' equation in a B-polynomial basis. Phys. Scr. **73**, 539–544 (2006)
6. Burgers, J.M.: A mathematical model illustrating the theory of turbulence. Adv. Appl. Mech. **1**, 171–199 (1948)
7. Burgers, J.M.: Mathematical examples illustrating relation occurring in the theory of turbulent fluid motion. Trans. R. Neth. Acad. Sci. Amst. **17**, 1–53 (1939)
8. Butcher, J.: Numerical Methods for Ordinary Differential Equations, 2nd edn. Wiley, New York (2008)
9. Cole, J.D.: On a quasi-linear parabolic equation occurring in aerodynamics. Q. Appl. Math. **30**, 225–236 (1972)
10. Dhawan, S., Kapoor, S., Kumar, S., Rawat, S.: Contemporary review of techniques for the solution of nonlinear Burgers' equation. J. Comput. Sci. **3**, 405–419 (2012)
11. Evans, D.J., Abdullah, A.R.: The group explicit method for the solution of Burgers' equation. Computing **32**, 239–253 (1984)
12. Evans, G., Blackledge, J., Yardley, P.: Numerical Methods for Partial Differential Equations. Springer, Berlin (2000)
13. Ganaie, I.A., Kukreja, V.K.: Numerical solution of Burgers' equation by cubic Hermite collocation method. Appl. Math. Comput. **237**, 571–581 (2014)
14. Gao, Y., Le, L.-H., Shi, B.-C.: Numerical solution of Burgers' equation by lattice Boltzmann method. Appl. Math. Comput. **219**, 7685–7692 (2013)
15. Guo, Y., Shi, Y., Li, Y.: A fifth-order finite volume weighted compact scheme for solving one-dimensional Burgers' equation. Appl. Math. Comput. **181**, 172–185 (2016)
16. Haq, S., Hussain, A., Uddin, M.: On the numerical solution of nonlinear Burgers' type equations using meshless method of lines. Appl. Math. Comput. **218**, 6280–6290 (2012)
17. Hopf, E.: The partial differential equation $u_t + uu_x = \nu u_{xx}$. Commun. Pure Appl. Math. **3**, 201–230 (1950)
18. Jiwari, R.: A haar wavelet quasilinearization approach for numerical simulation of Burgers' equation. Comput. Phys. Comm. **183**, 2413–2423 (2012)
19. Kadalbajoo, M.K., Awasthi, A.: A numerical method based on Crank–Nicolson scheme for Burgers' equation. Appl. Math. Comput. **182**, 1430–1442 (2006)
20. Kadalbajoo, M.K., Sharma, K.K., Awasthi, A.: A parameter-uniform implicit difference scheme for solving time-dependent Burgers' equation. Appl. Math. Comput. **170**, 1365–1393 (2005)
21. Kutluay, S., Bahadir, A.R., Ozdes, A.: Numerical solution of one-dimensional Burgers' equation: explicit and exact explicit methods. J. Comput. Appl. Math. **103**, 251–261 (1998)
22. Kutluay, S., Esen, A., Dag, I.: Numerical solutions of the Burgers' equation by the least-squares quadratic B-spline finite element method. J. Comput. Appl. Math. **167**, 21–33 (2004)
23. Prakash, A., Kumar, M., Sharma, K.K.: Numerical method for solving fractional coupled Burgers equations. Appl. Math. Comput. **262**, 314–320 (2015)
24. Mukundan, V., Awasthi, A.: A comparative study of three level explicit and implicit numerical scheme for convection diffusion equation. In: Proceedings of International conference on Mathematical and computational sciences, pp. 58–64. Narosa Publishing house (2015)
25. Mukundan, V., Awasthi, A.: Efficient numerical techniques for Burgers' equation. Appl. Math. Comput. **262**, 282–297 (2015)
26. Rashidi, M.M., Ganji, D.D., Dinarvand, S.: Explicit analytical solutions of the generalized Burger and Burger–Fisher Equations by homotopy perturbation method. Numer. Methods Partial. Differ. Equ. **25**(2), 409–417 (2009)
27. Rashidi, M.M., Domairry, G., Dinarvand, S.: Approximate solutions for the Burger and regularized long wave equations by means of the homotopy analysis method. Commun. Nonlinear Sci. Numer. Simul. **14**(3), 708–717 (2009)
28. Rashidi, M.M., Erfani, E.: New analytical method for solving Burgers' and nonlinear heat transfer equations and comparison with HAM. Comput. Phys. Commun. **180**, 1539–1544 (2009)

29. Rothe, E.: Zweidimesionale parabolische Randwertaufgaben als Grenzfall eindimensionaler Randwertaufgaben. Math. Ann. **102**, 650–670 (1930)
30. Najafi-Yazdi, A., Mongeau, L.: A low-dispersion and low-dissipation implicit Runge–Kutta scheme. J. Comput. Phys. **233**, 315–323 (2013)
31. Ozin, T., Aksan, E.N., Ozdes, A.: A finite element approach for solution of Burgers' equation. Appl. Math. Comput. **139**, 417–428 (2003)
32. Shao, L., Feng, X., He, Y.: The local discontinuous Galerkin finite element method for Burgers' equation. Math. Comput. Model. **54**, 2943–2954 (2011)
33. Shukla, H.S., Tamsir, M., Srivastava, V.K., Rashidi, M.M.: Modified cubic B-spline differential quadrature method for numerical solution of three-dimensional coupled viscous Burger equation. Mod. Phys. Lett. B (in press)

Online first articles

Volumes and issues

Sign up for alerts

About this journal

Electronic ISSN 0974-6870

Print ISSN 0971-3514

Abstracted and indexed in

ACM Digital Library

Baidu

CLOCKSS

CNKI

CNPIEC

Dimensions

EBSCO Discovery Service

Emerging Sources Citation Index

Google Scholar

Japanese Science and Technology Agency (JST)

Mathematical Reviews

Naver

Norwegian Register for Scientific Journals and Series

OCLC WorldCat Discovery Service

Portico

ProQuest-ExLibris Primo

ProQuest-ExLibris Summon

SCImago

SCOPUS

TD Net Discovery Service

UGC-CARE List (India)