Fully Implicit Scheme for Solving Burgers' Equation based on Finite Difference Method

N. A. Mohamed*

Department of Engineering Mathematics and Physics, Faculty of Engineering, Zagazig University, P.O. 44519, Zagazig, Egypt.

Article history:
Received 18 July 2018
Received in revised form 12 August 2018
Accepted on 28 August 2018
Available online 15 November 2018

Keywords:
Burgers’ equation
nonlinear one-dimensional
finite difference method
kinematic viscosity

1. Introduction

In this paper, a new scheme based on finite difference method is presented for solving nonlinear one-dimensional Burgers’ equation. The effectiveness of the scheme is illustrated by solving two test problems with known exact solutions. High accuracy of the present scheme is achieved at different values of kinematic viscosity. The obtained numerical solutions are in excellent agreement with the exact solutions and the results are compared with the most popular known explicit methods for solving this equation.

In this work, we consider the one-dimensional time dependent Burgers’ equation in the form:

\[
\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = \nu \frac{\partial^2 u}{\partial x^2}, \quad (x, t) \in \Omega \times [0, T],
\]

with the initial condition

\[
u(x,0) = f(x), \quad 0 \leq x \leq 1 \tag{2}
\]

and Dirichlet boundary conditions

\[
u(0,t) = u(1,t) = 0; \quad 0 \leq t \leq T, \tag{3}
\]

where \(\Omega = (0,1)\) and \(\nu > 0\) is kinematic viscosity coefficient. The structure of Burgers' equation is similar to the Navier–Stokes equation due to the combination of convection, diffusion and a time dependent terms. The Burgers equation (1) is one of the very few nonlinear partial differential equations which can be solved exactly for restricted set of initial functions \(f(x)\) only. So, various numerical methods have been introduced by researchers for studying the properties of it due to its wide range of applicability in respective fields of science and engineering.

Burgers' equation was first introduced by Bateman [1] who studied its steady state solution. Later, describing a mathematical model of turbulence, it was proposed by Burgers and due to the extensive work of Burgers, therefore it is referred as “Burgers’ equation”. In a series of papers [2, 3], Burgers investigated various aspects of turbulence and also studied the statistical and spectral aspects of the equation and related systems of equations. Burgers' equations have been solved analytically in Hopf [4] and Cole [5] for a restricted set of arbitrary initial conditions. Burgers’ equation is studied by many researchers for many reasons:
First, its analytical solution was obtained by Cole [5] so it is easy to have the numerical comparison.

Second, it contains the simplest form of nonlinear convection term and diffusion term for simulating the physical phenomena of wave motion.

Third, its shock wave behavior when \( \nu \) is very small.

Several numerical treatments of this equation have been presented by many researchers. For instance, treatments based on the Finite difference method in [6, 7, 8], Finite elements method in [9, 10], spectral least-squares method in [11, 12, 13], B-splines collocation method in [14, 15, 16], explicit and exact explicit finite difference methods in [17], variational iteration method in [18, 19], Adomian-Pade technique in [20], homotopy analysis method in [21, 22], spectral collocation method in [23, 24], polynomial based differential quadrature method in [25], B-spline differential quadrature method in [26, 27, 28], cubic Hermite collocation method in [29], hybrid numerical scheme based on Haar wavelets in [30], high order splitting method in [31], higher-order accurate finite difference method in [32], etc.

In this paper, the nonlinear term in Burger’s equation is treated with a numerical formula that is proposed in KAY [33]. The technique based on finite difference method and is used to solve the one-dimensional burger’s equation and compared with the most popular known explicit methods as Euler forward discretization (EF) and Mac Cormack discretization (MCOR).

2. The Solution procedure

The Burgers equation in (1) can be rewritten as:

\[
\frac{\partial u}{\partial t} = -u \frac{\partial u}{\partial x} + \nu \frac{\partial^2 u}{\partial x^2}, \quad P_1 \leq x \leq P_2, \quad t \in [0, T]. \tag{4}
\]

with initial and boundary conditions as in (2), (3) respectively. First, Semi-discretization is obtained by using forward finite differenced discretization along \( t \) direction while \( x \) direction remains undiscretized. Then, the full-discretization is obtained by using central finitely differenced discretization along \( x \) direction.

\[
\frac{1}{\kappa_{n+1}}u^{n+1} - \nu u_{xx}^{n+1} + u_{x}^{n+1}u_{x}^{n+1} = \frac{1}{\kappa_{n+1}}u^{n} \tag{5}
\]

From equation (5) the nonlinear term \( u^{n+1}u_{x}^{n+1} \) is needed to compute at every time step. In this work the linearization of this term is done as in [33], such that \( u^{n+1}u_{x}^{n+1} \approx w^{n+1}u_{x}^{n+1} \), where \( w^{n+1} \) is computed by linear extrapolation using \( u^{n} \) and \( u^{n-1} \) as:

\[
u^{n+1} \approx w^{n+1} = \left( 1 + \left( \frac{\kappa_{n+1}}{\kappa_{n}} \right) \right) u^{n} - \left( \frac{\kappa_{n+1}}{\kappa_{n}} \right) u^{n-1} \tag{6}
\]

Substitution of (6) in (5) yields,

\[
\frac{1}{\kappa_{n+1}}u^{n+1} - \nu u_{xx}^{n+1} + \left( 1 + \left( \frac{\kappa_{n+1}}{\kappa_{n}} \right) \right) u^{n} - \left( \frac{\kappa_{n+1}}{\kappa_{n}} \right) u_{x}^{n-1}u_{x}^{n+1} = \frac{1}{\kappa_{n+1}}u^{n} \tag{7}
\]

On discretizing along \( x \) direction, the Central Finite Difference discretization for the terms \( u_{xx}^{n+1} \) and \( u_{x}^{n+1} \) in equation (7) is used.

2.2 Full-discretization

Divide the interval \([0, T]\) into \( N \) steps \( 0 = t_0 \leq t_1 \leq \cdots \leq t_N = T, \Delta t = T/N \) and defining the current time step as \( \kappa_{n+1} = t_{n+1} - t_n \) and apply the forward finite difference formula along \( t \) direction in equation (4), it becomes:

\[
\frac{1}{\kappa_{n+1}}u^{n+1} - \nu u_{xx}^{n+1} + u_{x}^{n+1}u_{x}^{n+1} = \frac{1}{\kappa_{n+1}}u^{n} \tag{5}
\]
In this work, equal time step is used such that $K_n = K_{n+1} = \Delta t$, so equation (8) will be written as:

$$
\frac{1}{2h} \left( 1 + (\frac{K_{n+1}}{K_n}) \right) u_i^0 - \left( \frac{K_{n+1}}{K_n} \right) u_i^{-1} \right] [u_i^{n+1} - u_i^{-1}] = \frac{1}{K_{n+1}} u_i^n (8)
$$

Multiplication of equation (14) by $\Delta t h^2$ yields:

$$
[h^2 + 2\nu \Delta t] u_i^{n+1} + \left[ -\nu \Delta t + \frac{1}{2} \Delta t h (2u_i^n - u_i^{n-1}) \right] u_i^{n+1} + \left[ -\nu \Delta t - \frac{1}{2} \Delta t h (2u_i^n - u_i^{n-1}) \right] u_i^{n+1} = h^2 u_i^n (10)
$$

Equation (10) can be written in a simple form as:

$$
\alpha_i u_i^{n+1} + \beta_i u_i^{n+1} + \gamma_i u_{i-1}^{n+1} = f_i (11)
$$

where,

$$
\alpha_i = h^2 + 2\nu \Delta t \\
\beta_i = -\nu \Delta t + \frac{1}{2} \Delta t h (2u_i^n - u_i^{n-1}) \\
\gamma_i = -\nu \Delta t - \frac{1}{2} \Delta t h (2u_i^n - u_i^{n-1}) \\
f_i = h^2 u_i^n 
$$

Apply equation (11) at every point $i$, then solve the system of equations $Bu = F$ at every time step to find $u$ of all point at every time step.

3. The Numerical Results

Burgers' equation in (1-3) is solved by the proposed method and the numerical results are compared with the exact solution and with other known explicit methods i.e., Euler forward discretization (EF) and Mac Cormack discretization (MCOR) at different nodal points and at different values of kinematic viscosity $\nu$.

Since the exact solution is given, the $L_2$-discretization error norm is computed after each time step by using the following definitions:

$$
L_2 := \left\| u_{\text{exact}} - u_{\text{computed}} \right\|_2 = \sqrt{\frac{\sum_{j=1}^N \left| u_j^{\text{exact}} - u_j^{\text{computed}} \right|^2}{N}}
$$

The discretization error norms at the last time step are listed and compared with Euler forward (EF) and Mac Cormack discretization (MCOR).

Example 1:

Solve Burgers' equation in (1-3) with Initial condition:

$$
u(x, 0) = f(x) = \frac{2\nu x \sin (\frac{\pi x}{a})}{a + \cos (\frac{\pi x}{a})}, \quad a > 1
$$

whose exact solution is given by Wood [34]:

$$
u(x, t) = f(x) = \frac{2\nu e^{-\frac{\pi^2}{a^2} \sin (\frac{\pi x}{a})}}{a + e^{-\frac{\pi^2}{a^2} \cos (\frac{\pi x}{a})}}
$$

Numerical results obtained by the proposed method for $a = 1.1, h = 0.0125, T = 1, \nu = 0.1$ and time step $\Delta t = 0.01$ are listed at different nodal points in Table (1). Table (2) reports the comparison of the proposed method with other known explicit methods, (EF) and (MCOR), to get $L_2$ norm for $a = 1.1, h = 0.0125, T = 1, 10$ for $\nu = 0.01, 0.1, 1$. To investigate the stability of proposedmethod, the $L_2$- error norms at the last time step are listed in Table (3) for different values of initial condition parameter: $a$ and viscosity $\nu$. These results show that the proposed method is stable for a wide range of $\nu$. As mention above, other known numerical methods converges only within limited range of $\nu$.

Fig.1 showed the numerical solution obtained by the proposed method and exact solution for $a = 1.1, h = 0.0125, T = 1, \Delta t = 0.001$ and $\nu = 0.01$. It is cleared that the numerical solution is almost identical to the exact solution. Fig. 2 showed the effect of increasing the coefficient $a$ on the decreasing of $L_2$-error norm for the values of $\nu = 1, 10$. In Fig.3, $L_2$-error norm is shown for $a=1.1, h = 0.0125, \Delta t = 0.001$ and $T=1$ for different value of $\nu$. It is cleared that $L_2$- error norm decreased as $\nu$ increased. It is cleared from Fig.2 and Fig.3 that decreasing the value of $\nu$, needs more of the time iterations to reduce the error norm.
Table 1. Comparison of the proposed method with exact solution at different space points and the $L_2$- error norm for $a = 1.1$, $h = 0.0125$, $T = 1$, $10$ for $\nu = 0.1$.

<table>
<thead>
<tr>
<th>$x$</th>
<th>Proposed Method</th>
<th>Exact Solution</th>
<th>Proposed Method</th>
<th>Exact Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\Delta t = 0.001$</td>
<td>$\Delta t = 0.01$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.1</td>
<td>0.049787</td>
<td>0.049754</td>
<td>9.6339 e-06</td>
<td>9.12930 e-06</td>
</tr>
<tr>
<td>0.2</td>
<td>0.098285</td>
<td>0.098212</td>
<td>1.8325 e-05</td>
<td>1.73650 e-05</td>
</tr>
<tr>
<td>0.3</td>
<td>0.143751</td>
<td>0.143627</td>
<td>2.5222 e-05</td>
<td>2.39011 e-05</td>
</tr>
<tr>
<td>0.4</td>
<td>0.183471</td>
<td>0.183281</td>
<td>2.9651 e-05</td>
<td>2.80978 e-05</td>
</tr>
<tr>
<td>0.5</td>
<td>0.213164</td>
<td>0.212890</td>
<td>3.1178 e-05</td>
<td>2.95442 e-05</td>
</tr>
<tr>
<td>0.6</td>
<td>0.226510</td>
<td>0.226149</td>
<td>2.9652 e-05</td>
<td>2.80986 e-05</td>
</tr>
<tr>
<td>0.7</td>
<td>0.215483</td>
<td>0.215063</td>
<td>2.5224 e-05</td>
<td>2.39024 e-05</td>
</tr>
<tr>
<td>0.8</td>
<td>0.172786</td>
<td>0.172388</td>
<td>1.8326 e-05</td>
<td>1.73663 e-05</td>
</tr>
<tr>
<td>0.9</td>
<td>0.097316</td>
<td>0.097065</td>
<td>9.6349 e-06</td>
<td>9.13007 e-06</td>
</tr>
</tbody>
</table>

$L_2$ 2.5719 e-04 1.1549 e-06

Fig.1. Numerical solution by proposed method and Exact solution for $a = 1.1$, $h = 0.0125$, $T = 1$, $\Delta t = 0.001$ and $\nu = 0.01$

Fig.2. $L_2$- error norm for different values of Burger’s coefficient $a$ by using Proposed Method for $h = 0.0125$, $\Delta t = 0.001$, $T = 1$ for (I) $\nu = 1$ and (II) $\nu = 10$. 
Example 2:

Solve Burgers' equation in (1-3) with
Initial condition:

\[ u(x, 0) = f(x) = \sin(\pi x) \]

whose exact solution is given by [5]:

\[ u(x, t) = f(x) = 2\nu \sum_{n=1}^{\infty} c_n \exp(-n^2\pi^2\nu t) n\sin(n\pi x) \]

where the Fourier coefficients are:

\[ c_0 = \int_0^1 \exp\left\{-\frac{1}{2\nu}\left[1 - \cos(\pi x)\right]\right\} dx \]

\[ c_n = 2\int_0^1 \exp\left\{-\frac{1}{2\nu}\left[1 - \cos(\pi x)\right]\right\} \cos(n\pi x) dx \]

Numerical results obtained by the proposed method for \( h = 0.0125 \) and \( 0.01, T = 0.5, \nu = 1 \) and time step \( \Delta t = 0.001 \) are listed at different nodal points in Table (4). In Table (5) the comparison of the proposed method with exact solution at different time for \( h = 0.01, \nu = 0.1 \) and time steps \( \Delta t = 0.01, 0.001 \) are tabulated. To investigate the stability of the Proposed Method, the \( L_2 \)- error norms at the last time step are listed in Table (6) for different values of initial condition parameter viscosity \( \nu \). These results show that the proposed method is stable for a wide range of \( \nu \).
Table 4. Comparison of the proposed method with exact solution at different space points and the $L_2$-error norm for $h = 0.0125$ and 0.01, $T = 0.5$, $\nu = 1$ and time step $\Delta t = 0.001, 0.0001$.

<table>
<thead>
<tr>
<th>$X$</th>
<th>Proposed Method</th>
<th>Exact Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\Delta t = 0.001$</td>
<td>$\Delta t = 0.0001$</td>
</tr>
<tr>
<td>M=80</td>
<td>0.00227</td>
<td>0.00227</td>
</tr>
<tr>
<td>M=100</td>
<td>0.00432</td>
<td>0.00432</td>
</tr>
<tr>
<td>M=80</td>
<td>0.00594</td>
<td>0.00594</td>
</tr>
<tr>
<td>M=100</td>
<td>0.00699</td>
<td>0.00699</td>
</tr>
<tr>
<td>M=80</td>
<td>0.00735</td>
<td>0.00735</td>
</tr>
<tr>
<td>M=100</td>
<td>0.00999</td>
<td>0.00999</td>
</tr>
<tr>
<td>M=80</td>
<td>0.00595</td>
<td>0.00595</td>
</tr>
<tr>
<td>M=100</td>
<td>0.00432</td>
<td>0.00432</td>
</tr>
<tr>
<td>M=80</td>
<td>0.00227</td>
<td>0.00227</td>
</tr>
</tbody>
</table>

Table 5. Comparison of the proposed method with exact solution at different time for $h = 0.01, \nu = 0.1$ and time steps $\Delta t = 0.01, 0.001$.

<table>
<thead>
<tr>
<th>$X$</th>
<th>$T$</th>
<th>Proposed Method $\Delta t = 0.01$</th>
<th>Exact Solution $\Delta t = 0.001$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.5</td>
<td>2.5</td>
<td>0.044038</td>
<td>0.043459</td>
</tr>
<tr>
<td>2.5</td>
<td>3.0</td>
<td>0.027693</td>
<td>0.027261</td>
</tr>
<tr>
<td>2.5</td>
<td>3.5</td>
<td>0.017243</td>
<td>0.016932</td>
</tr>
<tr>
<td>2.5</td>
<td>0.25</td>
<td>0.066948</td>
<td>0.065988</td>
</tr>
<tr>
<td>2.5</td>
<td>0.5</td>
<td>0.040973</td>
<td>0.040296</td>
</tr>
<tr>
<td>2.5</td>
<td>0.75</td>
<td>0.025076</td>
<td>0.024608</td>
</tr>
<tr>
<td>2.5</td>
<td>3.0</td>
<td>0.051179</td>
<td>0.050374</td>
</tr>
<tr>
<td>2.5</td>
<td>3.5</td>
<td>0.030375</td>
<td>0.029844</td>
</tr>
<tr>
<td>2.5</td>
<td>0.04</td>
<td>0.018248</td>
<td>0.017896</td>
</tr>
</tbody>
</table>

Table 6. $L_2$-error norm at $h = 0.01$, $\Delta t = 0.001$ for different values of $\nu$ and $T$

<table>
<thead>
<tr>
<th>$\nu$</th>
<th>$T = 1$</th>
<th>$T = 2.5$</th>
<th>$T = 3.0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.8365 e-06</td>
<td>1.7740 e-12</td>
<td>1.5505 e-14</td>
</tr>
<tr>
<td>0.1</td>
<td>2.6350 e-04</td>
<td>9.4185 e-05</td>
<td>6.5146 e-05</td>
</tr>
<tr>
<td>0.04</td>
<td>4.8017 e-04</td>
<td>1.9112 e-04</td>
<td>1.4670 e-04</td>
</tr>
</tbody>
</table>

In Fig. 4, $L_2$-error norm for different values of $\nu$ by using Proposed Method for $h = 0.01$, $\Delta t = 0.001$, $T = 2.5$ are shown. As shown in Fig. 4, $\nu=1$ is needed 2500 time iterations to achieve the obtained error norm but decreasing $\nu$ doesn't achieve the same rate of reducing the error norm. This phenomena takes place in case of small viscosity due to appear of a thin layer close to the wall that is known as the boundary layer which is needed to increase grid points into the boundary layer region than the rest of the domain to obtain oscillation-free solutions. Also, small time increment is chosen to ensure high accuracy.

4. Conclusions

In this work the new scheme based on finite difference method was proposed to obtain the numerical solution of the 1D-Burgers’ equation. The nonlinear term in Burgers’ equation linearized without any transformation formula. The proposed scheme provides accurate solutions of the two test problems, be stable and converge even for low viscosity and long time. Moreover, the proposed scheme is simple and can be to implement easily.

Acknowledgment

The author is very thankful to the reviewers for their valuable comments and suggestions to improve the quality of the research work.

References


