A fully implicit Finite-difference solution to one dimensional Coupled Nonlinear Burgers’ equations

Vineet K. Srivastava, Mukesh K. Awasthi, and Mohammad Tamsir

Abstract—A fully implicit finite-difference method has been proposed for the numerical solutions of one dimensional coupled nonlinear Burgers’ equations on the uniform mesh points. The method forms a system of nonlinear difference equations which is to be solved at each iteration. Newton’s iterative method has been implemented to solve this nonlinear assembled system of equations. The linear system has been solved by Gauss elimination method with partial pivoting algorithm at each iteration of Newton’s method. Three test examples have been carried out to illustrate the accuracy of the method. Computed solutions obtained by proposed scheme have been compared with analytical solutions and those already available in the literature by finding $L_2$ and $L_{\infty}$ errors.

Keywords—Burgers’ equation, Implicit Finite-difference method, Newton’s method, Gauss elimination with partial pivoting.

I. INTRODUCTION

The one dimensional coupled viscous Burgers equation was derived by Esipov [1] to study the model of polydisperse sedimentation. This system of coupled equation is 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. Burgers [2] and Cole [3] found that this system of equations describe various kinds of phenomena such as a mathematical model of turbulence and the approximate theory of flow through a shock wave traveling in viscous fluid.

Cubed coupled equation is of interest from the numerical point of view, because in general, analytical solutions are not available. Exact solution of one dimensional coupled Burgers’ equations have been obtained by Kaya [5] using adomian decomposition method, whereas Soliman [6] used a modified extended tanh function method. The numerical solution of one dimensional coupled Burgers’ equations has been solved by several researchers and scientists. Esipov [1] presented numerical simulations and compared the results with experimental data. Abdou and Soliman [7] used the variational iteration method to solve one-dimensional Burgers and coupled Burgers equations. Wei and Gu [8] applied conjugate gradient approach, Khater et al. [9] used the Chebyshev spectral collocation method, Dehghan et al. [10] obtained numerical results of coupled viscous Burgers equations by using the Adomian-Pade technique. Rashid and Ismail [11] applied Fourier pseudo-spectral method. Recently Mittal and Arora [12] has used cubic B-spline collocation scheme based on Crank-Nicolson formulation for time integration and cubic B-spline functions for space integration by linearizing the nonlinear terms to solved coupled viscous Burger’s equation whereas Mokhtari et al. [13] has applied a generalized differential quadrature method. To the best of our knowledge, the explicit or implicit finite-difference schemes has not been applied to solve one dimensional coupled Burgers equation while there are several finite-difference schemes available for single one dimensional Burgers’ equation, two and three-dimensional Burgers’ equations, see references [14-23].

The purpose of this work is to solve one dimensional unsteady nonlinear coupled Burgers’ equations using a fully implicit finite-difference method. The advantage of using fully implicit scheme is that there is no need to linearize the nonlinear terms before discretization and also no additional constraints are required unlike the method given by Mittal and Arora [12]. Comparison of the scheme with the analytical solutions and those already available in the literature has been made in terms of accuracy and computational efficiency by finding the $L_2$ and $L_{\infty}$ errors.

II. GOVERNING EQUATIONS AND TEST PROBLEMS

Consider the generalized form of one dimensional coupled nonlinear Burgers’ equations:

\[ u_t + \delta u_{xx} + \eta uu_x + \alpha(uv)_x = 0 \]
\[ v_t + \mu v_{xx} + \xi vv_x + \beta(uv)_x = 0 \]

with the initial conditions

\[ u(x, 0) = a_1(x), \quad \forall x \in \Omega \]
\[ v(x, 0) = a_2(x) \]

and the Dirichlet boundary conditions

\[ u(x, t) = b_1(x, t), \quad v(x, t) = b_2(x, t), \quad \forall x \in \Omega, \quad t > 0 \]

where $\Omega = \{ x : c \leq x \leq d \}$ is the computational domain; $\delta, \mu, \eta$ and $\xi$ are real constants, $\alpha$ and $\beta$ are arbitrary constants depending on the system parameters such as Peclet number, stokes velocity of particles due to gravity and the Brownian diffusivity [4]. $u(x, t)$ and $v(x, t)$ are the velocity components to be determined; $a_1, a_2, b_1$ and $b_2$ are known functions; $u_t$, $v_t$ is unsteady term; $uu_x$ is the nonlinear convection term; $u_{xx}$ is the diffusion term and

\[ u_t = \frac{\partial u}{\partial t}, \quad v_t = \frac{\partial v}{\partial t}, \quad u_x = \frac{\partial u}{\partial x} \]

\[ v_x = \frac{\partial v}{\partial x}, \quad u_{xx} = \frac{\partial^2 u}{\partial x^2}, \quad v_{xx} = \frac{\partial^2 v}{\partial x^2} \]
Numerical solutions of the equations (1) and (2) has been obtained for the following three test example.

**Example 1:** In this example we set parameters $\delta = -1, \mu = -1, \eta = -2, \xi = -2, \alpha = 1, \beta = 1$ which yields Eqns. (1) and (2) as

\[
\begin{align*}
    u_t - u_{xx} &= 2u_{xx} + (uv)_x = 0 \\
    v_t - v_{xx} &= 2v_{xx} + (uv)_x = 0
\end{align*}
\]

The initial and boundary conditions are taken from the exact solution.

The exact solution of the Eqns. (5) and (6) is given by [4] as

\[
\begin{align*}
    u(x,t) &= \exp(-t) \sin(x) \\
    v(x,t) &= \exp(-t) \sin(x)
\end{align*}
\]

**Example 2:** In the second example we set parameters $\delta = -1, \mu = -1, \eta = 2, \xi = 2$, so that Eqns. (1) and (2) takes the following form:

\[
\begin{align*}
    u_t - u_{xx} &= 2u_{xx} + \alpha (uv)_x = 0 \\
    v_t - v_{xx} &= 2v_{xx} + \beta (uv)_x = 0
\end{align*}
\]

The exact solutions of Eqns. (8) and (9) are taken from [5] as

\[
\begin{align*}
    u(x,t) &= a_0 \left(1 - 2A \left(\frac{2\alpha - 1}{4\beta - 1}\right) \tanh(A(x - 2At))\right), \\
    v(x,t) &= a_0 \left(\left(\frac{2\beta - 1}{4\alpha - 1}\right) - 2A \left(\frac{2\alpha - 1}{4\beta - 1}\right) \tanh(A(x - 2At))\right),
\end{align*}
\]

where $A = a_0 \left(\frac{4\alpha - 2}{4\beta - 2}\right)$ and $a_0, \alpha, \beta$ are arbitrary constants. The initial and boundary conditions are taken from the exact solution.

**Example 3:** Here, parameters are taken as $\delta = -1, \mu = -1$, so that Eqns. (1) and (2) takes the following form:

\[
\begin{align*}
    u_t - u_{xx} &= \eta u_{xx} + \alpha (uv)_x = 0 \\
    v_t - v_{xx} &= \xi v_{xx} + \beta (uv)_x = 0
\end{align*}
\]

where $\eta, \xi, a_0, \alpha, \beta$ are arbitrary constants. Subject to the initial conditions [12]

\[
\begin{align*}
    u(x,0) &= \sin(2\pi x), x \in [0, 0.5] \\
    &\quad 0, x \in (0.5, 1) \\
    v(x,0) &= \begin{cases} 0, x \in [0, 0.5] \\
    &\quad -\sin(2\pi x), x \in (0.5, 1) 
\end{cases}
\end{align*}
\]

and zero boundary conditions.

**III. Solution Procedure**

The computational domain $\Omega$ is discretised on the uniform grid. Denote the discrete approximation of $u(x,t)$ and $v(x,t)$ at the grid point $(i\Delta x, n\Delta t)$ by $u^n_i$ and $v^n_i$, respectively ($i = 0, 1, 2, ..., n_x; n = 0, 1, 2, ....$) where $\Delta x = 1/n_x$ is the grid size in x-direction and $\Delta t$ represents time increment.

A fully implicit finite-difference approximation to (1) is given by

\[
\begin{align*}
    \frac{u^{n+1}_i - u^n_i}{\Delta t} + \delta \left(\frac{u^{n+1}_{i+1} - 2u^{n+1}_i + u^{n+1}_{i-1}}{(\Delta x)^2}\right) + \alpha \left(\frac{v^{n+1}_i}{2\Delta x}\right) = 0
\end{align*}
\]

Similarly, a fully implicit finite-difference approximation to equation (2) is given by

\[
\begin{align*}
    \frac{u^{n+1}_i}{\Delta t} + \mu \left(\frac{u^{n+1}_{i+1} - 2u^{n+1}_i + u^{n+1}_{i-1}}{(\Delta x)^2}\right) + \alpha \left(\frac{v^{n+1}_i}{2\Delta x}\right) = 0
\end{align*}
\]

Newton’s method is used to linearize the nonlinear Eqns. (15) and (16) and computed solution is obtained by iteration. The resulting linearized equations form a block tridiagonal matrix system of order $n$, as in the following form:

\[
\begin{align*}
    a_i \delta_{i-1} + b_i \delta_i + c_i \delta_{i+1} &= \bar{r}_i, \quad i = 1, 2, ..., n
\end{align*}
\]

where $a_i, b_i$ and $c_i$ are block matrices of order two, $\delta = [\delta u_1 \delta v_1]^T$ is the change in the solution vector, and $\bar{r}$ is the right hand-side vector, each of order two. At each iteration, Gauss elimination method with partial pivoting algorithm is implemented to obtain the solution of the system (17). In the iterative method, solution at the previous time step is taken as the initial guess for the convergence point of view.

The accuracy and consistency of the scheme is measured in terms of error norms $L_2$ and $L_\infty$ defined as:

\[
L_2 := \|u_{exact} - u_{computed}\|_2 = \frac{\sum_{j=0}^{N} |u_{exact}^j - u_{computed}^j|^2}{\sum_{j=0}^{N} |u_{exact}^j|^2}
\]

\[
L_\infty := \|u_{exact} - u_{computed}\|_\infty = \max_j |u_{exact}^j - u_{computed}^j|
\]

Where $u_{exact}$ and $u_{computed}$ represent exact and computed solutions respectively.

**IV. Numerical Results And Discussions**

The numerical computations have been performed on the uniform mesh. For example 1, results have been calculated by considering domain $x \in [-\pi, \pi]$ and $\Delta t = 0.001$ which have been shown in Table 1 and Table 2 at different time levels where $t \in [0, 1]$ and with different number of partitions. From these tables, we have seen that the scheme is consistent because as the number of partitions refines, error reduces. The numerical and exact solutions of $u$ and $v$ have been compared in figure 1 with number of partitions 20. It can be observed that the computed results show excellent agreement with the exact solution.

For example 2, the computations have been carried out in the domain $x \in [-10, 10]$, $\Delta t = 0.01$ and number of partitions 100. The errors $L_2$ and $L_\infty$ have been calculated and compared in Table 3 and Table 4 with the those already available in the literature. The errors computed from the present method is very less as compared to the errors obtained by Mittal and Arora [12]. This shows that the proposed method is highly accurate in comparison to the other existing methods. The computed and exact solutions of $u(x,t)$ and $v(x,t)$ have been compared in figure 2 with number of partitions 10, $\Delta t = 0.1$, $t = 1, \alpha = 1, \beta = 2$ and the excellent agreement has been found between the numerical
In figure 3, the variation of \( u(x, t) \) have been calculated in the domain \( x \in [0, 1]; \Delta t = 0.01, \alpha = \beta = 10 \) for different values of \( \eta \) and \( \xi \). As the time interval increases, the peak values of \( u(x, t) \) decreases. It can also be seen that \( u(x, t) \) first increases and then decreases, on increasing the values of \( \eta \) and \( \xi \). The variation of \( v(x, t) \) has been shown in figure 4 for the same parametric values as taken in figure 3. The peak values of \( v(x, t) \) decreases on increasing \( t \). It is also observed that the numerical solution decays to zero with increasing time levels and with the increasing values of \( \eta \) and \( \xi \).
Fig. 3. Solution profile of $u(x,t)$ at different time levels when $3(a)\eta = \xi = 1, 3(b)\eta = \xi = 10$ and $3(c)\eta = \xi = 100$, for $\alpha = \beta = 10$, in Example 3.

Fig. 4. Solution profile of $v(x,t)$ at different time levels when $4(a)\eta = \xi = 1, 4(b)\eta = \xi = 10$ and $4(c)\eta = \xi = 100$, for $\alpha = \beta = 10$, in Example 3.

Figure 5 shows the variation the $u(x,t)$ for similar values as taken in figure 3 except $\alpha = \beta = 100$. It has been observed that the behavior of $u(x,t)$ is same as obtained in figure 3 but the peak values are smaller. We observe a sharp decay in the computed solution for the higher values of $\alpha$ and $\beta$. From figure 6, it can easily seen that the variation of $v(x,t)$ is same as observed from figure 4 but values decay as $\alpha$ and $\beta$ increases.

V. CONCLUSION

In this work, a numerical approximation has been proposed for solving one dimensional coupled nonlinear Burgers’ equations using a fully implicit finite-difference scheme. The

<table>
<thead>
<tr>
<th>$t$</th>
<th>Present method (No. of Partition=200)</th>
<th>Mittal (No. of Partition=200)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>$8.86e^{-05}$ 5.20e^{-04}</td>
<td>$2.07e^{-05}$ 4.46e^{-06}</td>
</tr>
<tr>
<td>0.5</td>
<td>$2.94e^{-04}$ 1.79e^{-04}</td>
<td>$1.49e^{-05}$ 4.10e^{-05}</td>
</tr>
<tr>
<td>1.0</td>
<td>$5.91e^{-04}$ 2.17e^{-04}</td>
<td>$3.00e^{-05}$ 8.21e^{-05}</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$t$</th>
<th>Present method (No. of Partition=400)</th>
<th>Mittal (No. of Partition=400)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>$5.30e^{-05}$ 4.80e^{-05}</td>
<td>$2.03e^{-06}$ 1.86e^{-06}</td>
</tr>
<tr>
<td>0.5</td>
<td>$2.67e^{-04}$ 1.62e^{-04}</td>
<td>$1.02e^{-05}$ 6.22e^{-06}</td>
</tr>
<tr>
<td>1.0</td>
<td>$5.38e^{-04}$ 1.98e^{-04}</td>
<td>$2.04e^{-05}$ 7.56e^{-06}</td>
</tr>
</tbody>
</table>
efficiency and accuracy of the scheme has been demonstrated taking three test examples. The numerical results show that fully implicit finite-difference scheme performs well in the case of 1D coupled Burgers’ equation. The proposed method is highly accurate as compared to the other numerical method and shows excellent agreement with the exact solution.

REFERENCES


<table>
<thead>
<tr>
<th>$t$</th>
<th>$\alpha$</th>
<th>$\beta$</th>
<th>Present method $L_2$</th>
<th>Mittal $L_\infty$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>0.1</td>
<td>0.3</td>
<td>$6.601e^{-04}$</td>
<td>$6.903e^{-04}$</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.03</td>
<td>$4.122e^{-04}$</td>
<td>$4.313e^{-05}$</td>
</tr>
<tr>
<td>1.0</td>
<td>0.1</td>
<td>0.3</td>
<td>$1.304e^{-03}$</td>
<td>$8.151e^{-03}$</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.03</td>
<td>$1.388e^{-03}$</td>
<td>$8.541e^{-05}$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$t$</th>
<th>$\alpha$</th>
<th>$\beta$</th>
<th>Present method $L_2$</th>
<th>Mittal $L_\infty$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>0.1</td>
<td>0.3</td>
<td>$4.890e^{-04}$</td>
<td>$7.056e^{-04}$</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.03</td>
<td>$2.133e^{-05}$</td>
<td>$4.916e^{-05}$</td>
</tr>
<tr>
<td>1.0</td>
<td>0.1</td>
<td>0.3</td>
<td>$9.534e^{-04}$</td>
<td>$1.591e^{-04}$</td>
</tr>
<tr>
<td></td>
<td>0.3</td>
<td>0.03</td>
<td>$9.779e^{-05}$</td>
<td>$2.250e^{-03}$</td>
</tr>
</tbody>
</table>

Fig. 5. Solution profile of $u(x,t)$ at different time levels when $5(\alpha)\eta = \xi = 1$, $5(b)\eta = \xi = 10$ and $5(c)\eta = \xi = 100$, for $\alpha = \beta = 100$, in Example 3.

Fig. 6. Solution profile of $v(x,t)$ at different time levels when $6(\alpha)\eta = \xi = 1$, $6(b)\eta = \xi = 10$ and $6(c)\eta = \xi = 100$, for $\alpha = \beta = 100$, in Example 3.

TABLE III
COMPARISONS OF ERRORS AT DIFFERENT TIME FOR $u(x,t)$ FOR EXAMPLE 3.

TABLE IV
COMPARISONS OF ERRORS AT DIFFERENT TIME FOR $v(x,t)$ FOR EXAMPLE 3.


