Available online at http://ijim.srbiau.ac.ir/
Int. J. Industrial Mathematics (ISSN 2008-5621)
Vol. 7, No. 2, 2015 Article ID IJIM-00554, 11 pages
Research Article

Numerical solution of the one dimensional non-linear Burgers equation using the Adomian decomposition method and the comparison between the modified Local Crank-Nicolson method and the VIM exact solution

AR. Haghighi *†, M. Shojaeifard ‡

Abstract

The Burgers equation is a simplified form of the Navier-Stokes equations that very well represents their non-linear features. In this paper, numerical methods of the Adomian decomposition and the Modified Crank Nicholson, used for solving the one-dimensional Burgers equation, have been compared. These numerical methods have also been compared with the analytical method. In contrast to the conventional Crank-Nicolson method, the MLCN method is an explicit and unconditionally stable method. The Adomian decomposition method includes the unknown function U (x), in which each equation is defined and solved by an infinite series of unbounded functions. Velocity parameters u in the direction of the X axis, are examined at different times with different Reynolds numbers over a fixed time step. Also the accuracy of the Adomian and the Crank-Nicolson methods at different Reynolds numbers have been studied using two examples with different initial conditions, and the Adomian decomposition method is closer to the analytical method.

Keywords : Non-linear Burgers equation; Adomian method; the modified Local Crank-Nicolson method.

1 Introduction

The Burgers equation is a special form of the Navier-Stokes and the continuity equations, which was introduced in 1915 by Bateman [1]. In this equation continuity and pressure components of the Navier-Stokes is omitted. Burgers equation is a fundamental partial differential equation of the fluid mechanics [2]. This equation is widely used in many physical phenomena, such as models of gas dynamics, plasma dynamics, simulation of traffic flows, shock wave, simplified model of the behavior of the boundary layer, sound attenuation in fog, etc [3, 4, 5, 6, 7, 8, 9, 10, 11]. Because of their widespread use, these equations have been studied by many researchers and are appropriate context for research activities and several studies about different, accurate and explicit numerical solutions are done to generalize these equations to higher dimensions [12]. Different numerical methods such as the finite difference, the finite element, and the spectral methods, are used for solving the Burgers equations [13, 14, 15, 16]. In recent years, the Adomian decomposition method (ADM) has been considered by many researchers for solving the Burgers equation [17, 18]. Our best strategy, which hasnt

*Corresponding author. ah.haghighi@gmail.com
†Department of Mathematics, Urmia University of Technology, Urmia, Iran.
‡Department of Mathematics, Urmia University of Technology, Urmia, Iran.

yet been used for solving the Burgers equation is the ADM discretization method. The ADM discretization method was first utilized for obtaining numerical solutions, to discretize the nonlinear Schrdinger equation [19]. The Adomian decomposition method was presented in early 1980 by George Adomian [20]. Abduwali has introduced the Crank-Nicholson (CN) and the modified local Crank-Nicholson (MLCN) methods in order to solve the motion, heat, and Burgers equations respectively [21, 22]. Various methods have been introduced for the numerical solution of the Burgers equations in higher dimensions. In these methods a variety of linear conversions such as the Hopf-cole and Auto and Backlund or an Ancillary Function have been used for accurate solutions of these equations [23].

It is notable that, no conversion has been used in the MLCN method. In this method, partial differential equations are converted into ordinary differential equations. The MLCN method converts the Coefficients matrix to a simple block matrix, which is an explicit and unconditional method [24].

The organization of this paper is as follows. the Burgers one-dimensional equation is defined in setion 2. The solution of this equation using the Adomian decomposition method is given in Section 3. Section 4 modifies the local Crank-Nicholson and the analytical solution is used to solve the one-dimensional Burgers equations. In Section 5, numerical examples for both methods and their comparison to the analytical solution is given. And at the end a conclusion of this paper is presented.

2 The one-dimensional Burgers equation

The general form of the one-dimensional Burgers equation is as follows: [25]

$$u_t + uu_x = \frac{1}{R}(u_{xx}). \tag{2.1}$$

With initial conditions:

$$u(x,0) = f(x), \quad x \in D \tag{2.2}$$

And boundary conditions:

$$u(x,t) = f_1(x,t), \quad x \in \partial D \tag{2.3}$$

where $D = \{x \,|\, a \leqslant x \leqslant b\}$ and ∂D is its boundary, $u(x,y)$ determine the velocity components f, f_1 are known functions, and R is the Reynolds number.

To solve system (1) with initial conditions, Bahadir proposed a fully implicit finite-difference scheme as follows [26]:

$$\frac{1}{\tau}(u_i^{n+1} - u_i^n) + \frac{1}{2h_x}(u_{i+1}^{n+1} - u_{i-1}^{n+1})u_i^{n+1} = \frac{1}{Rh_x^2}(u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1}). \tag{2.4}$$

In the above definition, the space domain $[0\,,\,N_x]$ is divided into a N_x mesh with the spatial step size $h_x = \frac{1}{N_x}$ in x direction, the time step size τ represent the increment in time. A discrete approximation of $u(x,y)$ at the uniform mesh $(ih_x, n\tau)$ is denoted as u_i^n.

3 The discrete Adomian decomposition method

In this section, we describe the discrete ADM method as it is applied to the 1D Burgers equations system, in which the fully implicit finite difference part has been used. For the system of the Burgers equations the following operator form can be used:

$$D_\tau^+ u_i^n + (D_{h_x} u_i^{n+1})u_i^{n+1} = \frac{1}{R}(D_{h_x}^2 u_i^{n+1}) \tag{3.5}$$

With the initial conditions:

$$u_i^0 = f_i. \tag{3.6}$$

where $i \in Z\,, n \in N_0$ and The standard forward difference is:

$$D_\tau^+ u_i^n = \frac{(u_i^{n+1} - u_i^n)}{\tau}. \tag{3.7}$$

And $D_{h_x} u_i^{n+1}$ denotes the central difference given by:

$$D_{h_x} u_i^{n+1} = \frac{(u_{i+1}^{n+1} - u_{i-1}^{n+1})}{2h_x}. \tag{3.8}$$

The standard second order difference $D_{h_x}^2 u_i^{n+1}$ are given by:

$$D_{h_x}^2 u_i^{n+1} = \frac{(u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1})}{h_x^2}. \tag{3.9}$$

In this method, the linear operator is determined as follows:

$$D_\tau^+ w^n = \frac{(w^{n+1} - w^n)}{\tau}. \qquad (3.10)$$

And the inverse operator $(D_\tau^+)^{-1}$ of this system is defined as:

$$(D_\tau^+)^{-1} w^n = \tau \sum_{m=0}^{n-1} w^m, \quad n \in N_0. \qquad (3.11)$$

Thus

$$(D_\tau^+)^{-1} D_\tau^+ u_i^n = u_i^n - u_i^0. \qquad (3.12)$$

Applying the inverse operator $(D_\tau^+)^{-1}$ to Eq. (5):

$$u_i^n = u_i^0 - (D_\tau^+)^{-1}(D_{h_x} u_i^{n+1}) u_i^{n+1} + \frac{1}{R}(D_{h_x}^2 u_i^{n+1}). \qquad (3.13)$$

The nonlinear operator of (13) can be defined as:

$$M_1(u_i^{n+1}) = (D_{h_x} u_i^{n+1}) u_i^{n+1}. \qquad (3.14)$$

Submitted Eq. (14) into Eq. (13):

$$u_i^n = f_i - -(D_\tau^+)^{-1} M_1(u_i^{n+1}) + \frac{1}{R}(D_{h_x}^2 u_i^{n+1}). \qquad (3.15)$$

The proposed discrete ADM suggests the expression of u_i^n in decomposition form as follows:

$$u_i^n = \sum_{l=0}^{\infty} u_{i,l}^n. \qquad (3.16)$$

Similar to the continuous ADM, the nonlinear operators $M_1(u_i^{n+1})$ can be defined by the infinite series of the Adomian polynomial as:

$$M_1(u_i^{n+1}) = \sum_{l=0}^{\infty} A_l. \qquad (3.17)$$

Where A_l are called as Adomian polynomials. The zero components $u_{i,0}^n$ and the remaining component $(u_{i,l}^n, \, l \geq 0)$ can be determined using the following recursive relation:

$$u_{i,0}^n = f_i. \qquad (3.18)$$

$$u_{i,l+1}^n = - - (D_\tau^+)^{-1} A_l + \frac{1}{R}(D_{h_x}^2 u_i^{n+1}). \qquad (3.19)$$

Where the Adomian polynomial A_l are evaluated with the following formula

$$A_l = \frac{1}{l!}\left[\frac{d^l}{d\lambda^l} M_1\left(\sum_{k=0}^{\infty}(\lambda^k u_{i,k}^{n+1})\right)\right]_{\lambda=0}. \qquad (3.20)$$

The first few terms of the Adomian polynomial A_l can be obtained from the above equation as follow:

$$\begin{aligned}
A_0 &= (D_{h_x} u_{i,0}^{n+1}) u_{i,0}^{n+1}, \\
A_1 &= (D_{h_x} u_{i,0}^{n+1}) u_{i,1}^{n+1} + (D_{h_x} u_{i,1}^{n+1}) u_{i,0}^{n+1}, \\
A_2 &= (D_{h_x} u_{i,0}^{n+1}) u_{i,2}^{n+1} + (D_{h_x} u_{i,1}^{n+1}) u_{i,1}^{n+1} \\
&\quad + (D_{h_x} u_{i,0}^{n+1}) u_{i,2}^{n+1}.
\end{aligned}$$

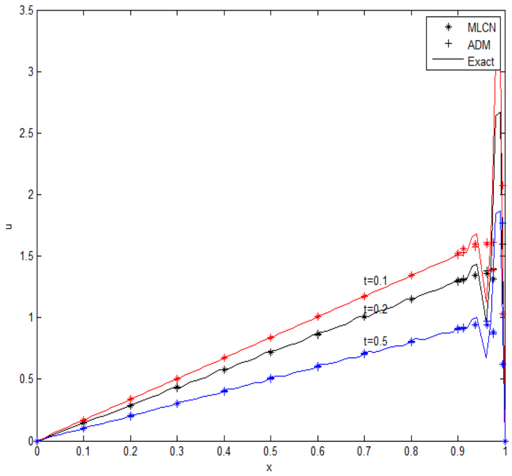

Figure 1: Comparison of the analytical method and the ADM and MLCN in the three steps of time $t = 0.1, 0.2, 0.5$ for $R = 100$.

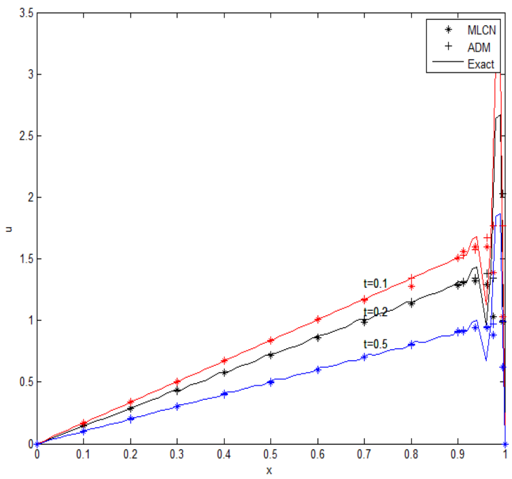

Figure 2: Comparison of the analytical method and the ADM and MLCN in the three steps of time $t = 0.1, 0.2, 0.5$ for $R = 10$.

Table 1: The ADM and MLCN compared to the analytical solution of the R = 100 and $\tau = 10^{-4}$ and $t = 0.1$.

point	*ADM*	*MLCN*	*Exact*
0.1	0.1678	0.1677	0.1678
0.2	0.3355	0.3354	0.3356
0.3	0.5033	0.5031	0.5034
0.4	0.6711	0.6707	0.6711
0.5	0.8388	0.8384	-0.8389
0.6	1.0066	1.0061	1.0067
0.7	1.1743	1.1738	1.1745
0.8	1.3421	1.3415	1.3423
0.9	1.5099	1.5091	1.5101

Table 2: The ADM and MLCN compared to the analytical solution of the R = 100 and $\tau = 10^{-4}$ and $t = 0.2$.

point	*ADM*	*MLCN*	*Exact*
0.1	0.1437	0.1442	0.1445
0.2	0.2873	0.2885	0.2892
0.3	0.4310	0.4327	0.4335
0.4	0.5747	0.5769	0.5780
0.5	0.7183	0.7212	0.7225
0.6	0.8620	0.8654	0.8681
0.7	1.0056	1.0096	1.0116
0.8	1.1493	1.1539	1.1516
0.9	1.2930	1.2961	1.3006

Table 3: The ADM and MLCN compared to the analytical solution of the R = 100 and $\tau = 10^{-4}$ and $t = 0.5$.

point	*ADM*	*MLCN*	*Exact*
0.1	0.1004	0.1012	0.1020
0.2	0.2008	0.2027	0.2041
0.3	0.3012	0.3040	0.3061
0.4	0.4016	0.4053	0.4082
0.5	0.5020	0.5066	0.5102
0.6	0.6024	0.6080	0.6122
0.7	0.7028	0.7093	0.7143
0.8	0.8032	0.8106	0.8163
0.9	0.9036	0.9117	0.9184

Table 4: The ADM and MLCN compared to the analytical solution of the R = 10 and $\tau = 10^{-4}$ and $t = 0.1$.

point	*ADM*	*MLCN*	*Exact*
0.1	0.1678	0.1677	0.1678
0.2	0.3355	0.3354	0.3356
0.3	0.5033	0.5031	0.5034
0.4	0.6711	0.6707	0.6711
0.5	0.8388	0.8384	0.8389
0.6	1.0066	1.0061	1.0067
0.7	1.1743	1.1738	1.1745
0.8	1.3421	1.3415	1.3423
0.9	1.5099	1.5091	1.5101

Table 5: The ADM and MLCN compared to the analytical solution of the R = 10 and $\tau = 10^{-4}$ and $t = 0.2$.

point	*ADM*	*MLCN*	*Exact*
0.1	0.1437	0.1442	0.1445
0.2	0.2873	0.2885	0.2892
0.3	0.4310	0.4327	0.4335
0.4	0.5747	0.5769	0.5780
0.5	0.7183	0.7212	0.7225
0.6	0.8620	0.8654	0.8681
0.7	1.0056	1.0096	1.0116
0.8	1.1493	1.1539	1.1516
0.9	1.2930	1.2961	1.3006

Table 6: The ADM and MLCN compared to the analytical solution of the R = 10 and $\tau = 10^{-4}$ and $t = 0.5$.

point	*ADM*	*MLCN*	*Exact*
0.1	0.1004	0.1012	0.1020
0.2	0.2008	0.2027	0.2041
0.3	0.3012	0.3040	0.3061
0.4	0.4016	0.4053	0.4082
0.5	0.5020	0.5066	0.5102
0.6	0.6024	0.6080	0.6122
0.7	0.7028	0.7093	0.7143
0.8	0.8032	0.8106	0.8163
0.9	0.9036	0.9117	0.9184

Table 7: The ADM and MLCN compared to the analytical solution of the R = 80 and $\tau = 10^{-4}$ and $t = 0.1$.

point	*ADM*	*MLCN*	*Exact*
0.1	0.1678	0.1677	0.1678
0.2	0.3355	0.3354	0.3356
0.3	0.5033	0.5031	0.5034
0.4	0.6711	0.6707	0.6711
0.5	0.8388	0.8384	0.8389
0.6	1.0066	1.0061	1.0067
0.7	1.1743	1.1738	1.1745
0.8	1.3421	1.3415	1.3423
0.9	1.5099	1.5091	1.5101

Table 8: The ADM and MLCN compared to the analytical solution of the R = 80 and $\tau = 10^{-4}$ and $t = 0.2$.

point	*ADM*	*MLCN*	*Exact*
0.1	0.1437	0.1442	0.1445
0.2	0.2873	0.2885	0.2892
0.3	0.4310	0.4327	0.4335
0.4	0.5747	0.5769	0.5780
0.5	0.7183	0.7212	0.7225
0.6	0.8620	0.8654	0.8681
0.7	1.0056	1.0096	1.0116
0.8	1.1493	1.1539	1.1516
0.9	1.2930	1.2961	1.3006

Table 9: The ADM and MLCN compared to the analytical solution of the R = 80 and $\tau = 10^{-4}$ and $t = 0.5$.

point	*ADM*	*MLCN*	*Exact*
0.1	0.1004	0.1012	0.1020
0.2	0.2008	0.2027	0.2041
0.3	0.3012	0.3040	0.3061
0.4	0.4016	0.4053	0.4082
0.5	0.5020	0.5066	0.5102
0.6	0.6024	0.6080	0.6122
0.7	0.7028	0.7093	0.7143
0.8	0.8032	0.8106	0.8163
0.9	0.9036	0.9117	0.9184

Table 10: The ADM and MLCN compared to the analytical solution of the R = 100 and $\tau = 10^{-4}$ and $t = 0.1$.

point	*ADM*	*MLCN*	*Exact*
0.1	0.0906	0.0916	0.0240
0.2	0.1802	0.1814	0.3076
0.3	0.2680	0.2703	0.2322
0.4	0.3531	0.3572	0.1931
0.5	0.4345	0.4313	1.1852
0.6	0.5114	0.5218	0.1884
0.7	0.5831	0.5983	0.8068
0.8	0.6488	0.6700	0.4066
0.9	0.7079	0.7361	0.4217

Table 11: The ADM and MLCN compared to the analytical solution of the R = 100 and $\tau = 10^{-4}$ and $t = 0.2$.

point	*ADM*	*MLCN*	*Exact*
0.1	0.0819	0.0834	0.0488
0.2	0.1629	0.1663	0.1747
0.3	0.2423	0.2482	0.4250
0.4	0.3192	0.3286	0.1156
0.5	0.3928	0.4070	0.5927
0.6	0.4624	0.4829	0.4232
0.7	0.5272	0.5557	0.4056
0.8	0.5867	0.6250	0.8655
0.9	0.6402	0.6898	0.4710

Table 12: The ADM and MLCN compared to the analytical solution of the R = 10 and $\tau = 10^{-4}$ and $t = 0.1$.

point	*ADM*	*MLCN*	*Exact*
0.1	0.0907	0.0903	0.0967
0.2	0.1804	0.1801	0.1870
0.3	0.2683	0.2686	0.2433
0.4	0.3535	0.3548	0.6304
0.5	0.4350	0.4379	0.4652
0.6	0.5120	0.5160	0.4859
0.7	0.5638	0.5834	0.8068
0.8	0.6495	0.6049	0.6172
0.9	0.4378	0.4422	0.4217

Table 13: The ADM and MLCN compared to the analytical solution of the R = 10 and $\tau = 10^{-4}$ and $t = 0.2$.

point	*ADM*	*MLCN*	*Exact*
0.1	0.0820	0.0823	0.0828
0.2	0.1631	0.1642	0.1635
0.3	0.2426	0.2451	0.2431
0.4	0.3191	0.3241	0.3234
0.5	0.3932	0.3996	0.4001
0.6	0.4629	0.4671	0.4649
0.7	0.5278	0.5095	0.5034
0.8	0.5873	0.4815	0.4755
0.9	0.4409	0.3140	0.3115

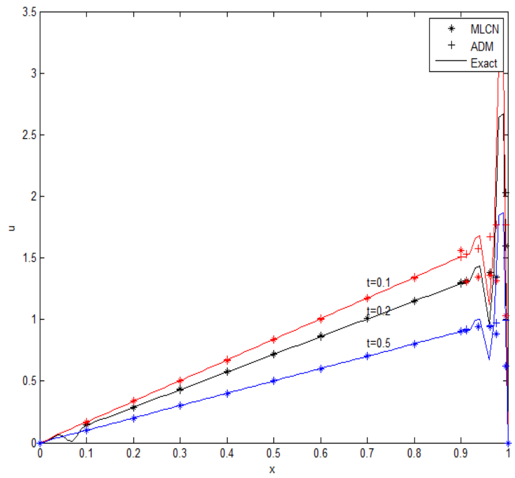

Figure 3: Comparison of the analytical method and the ADM and MLCN in the three steps of time $t = 0.1, 0.2$ and 0.5 for $R = 80$.

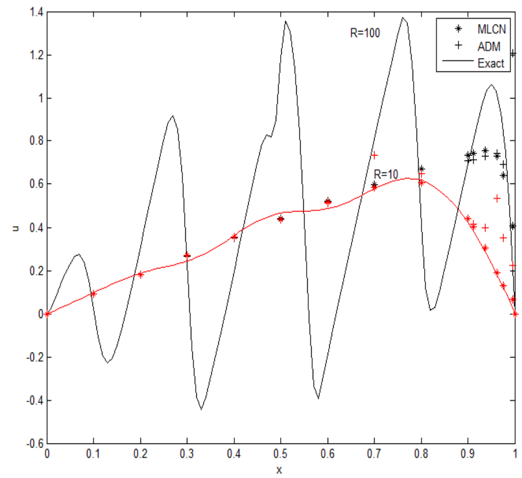

Figure 4: Comparison of analytical method and the MLCN and ADM with Reynolds numbers 10 and 100 in time step $t = 0.1$.

4 The modified local Crank-Nicolson method and Analytical solution for one dimentional Bergur's solution

4.1 The modified local Crank-Nicolson method

The Crank-Nicolson method (CN) is a central finite difference method. It should be noted that the Crank-Nicholson method is an implicit method, to obtain the values of u in the next steps, a set of algebraic equations must be solved, because the partial differential values are non-linear. Therefore, the discretization of these values must be non-linear. For many Burgers equations and many other equations it can be shown that the Crank-Nicholson method is unconditional stable. The modified local Crank-Nicholson method (MLCN) converts the partial differential equations into ordinary differential equations. The MLCN converts Coefficients matrixes into simple block matrixes. Using the Hopf - Cole conversion or using other conversion, the Burgers nonlinear equation becomes a linear equation [24], To solve equation (1) with the given boundary conditions using the central difference the following discretization equation is achieved [24].

$$\frac{dV(t)}{dt} = \frac{1}{2h^2} AV(t). \tag{4.21}$$

Where the vector $V(t)$ represents approximate u values in the Burgers equation (1), h is the spatial step, Δt is the time step and A is a tree diagonal matrix with dimensions $(M - 1) \times (M - 1)$. Then the by integrating equation (21) and defininge the vector $V(t_n) = [v(x_1, t_n), v(x_2, t_n), ..., v(x_{M-1}, t_n)]^T$,

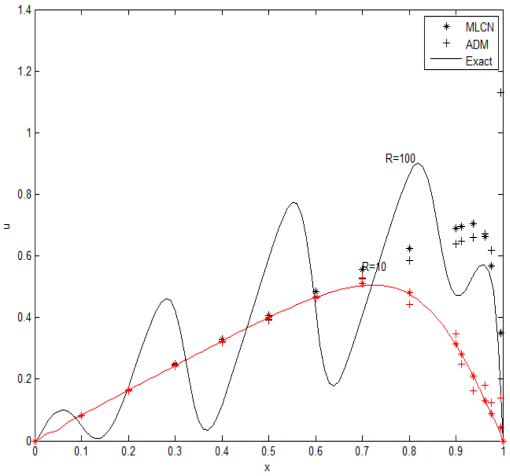

Figure 5: Comparison of the analytical method and the MLCN and ADM with Reynolds numbers 10 and 100 in time step $t = 0.2$.

we will have:

$$V(t_{n+1}) = \exp\left(\frac{\Delta t}{2h^2}A\right)V(t_n). \tag{4.22}$$

However, the Crank-Nicholson method for the Burgers equation (1) gives the following result:

$$V(t_{n+1}) = \left((1-\lambda A)^{-1})\right)\left((1+\lambda A)\right)V(t_n). \tag{4.23}$$

Where $\lambda = \frac{\tau}{4h^2}$ is called the networking ratio. The following equation is obtained by comparing equations (22) and (23):

$$\exp\left(\frac{\Delta t}{2h^2}A\right) \approx \left((1-\lambda A)^{-1})\right)\left((1+\lambda A)\right).$$

4.2 Analytical solution

The non-linear equation is given as [27]:

$$Lu(t) + Nu(t) = g(t). \tag{4.24}$$

Where L is a linear operator, N a nonlinear operator, and $g(t)$ is a known analytical function. According to the variational iteration method (VIM) we can have the following recursive relation:

$$u_{n+1}(x,t) = u_n(x,t) + \int_0^t \lambda(\xi)(Lu_n(\xi) + N\widehat{u}_n(\xi) - g(\xi))d\xi \tag{4.25}$$

Where λ is the general Lagrange multiplier which can be identified by the variational theory. Is an initial $u_0(t)$ approximation that may be unknown. And $\widehat{u}_n$ is considered as boundary change and $\delta\widehat{u}_n = 0$. Thus, in the first iteration of the Lagrange multiplier λ is characterized which is obtained using fractional integration. Successive approximations $u_{n+1}(t)$, are for the solution of $u(t)$ which is easily obtained using the Lagrange multiplier and using any selective functions u_0 . Consequently, the exact solution can be achieved using $u = \lim_{n\mapsto\infty} u_n$.

Equation (1) with initial and boundary conditions is considered, for solving Eq. (1) with initial condition (2), via VIM, And with replacement in (26), it is written in the form of the original equation [27] .

$$\begin{aligned} u_{n+1}(x,t) = u_n(x,t) &+ \int_0^t \lambda(\xi)(\frac{\partial u_n}{\partial \xi}(x,\xi) \\ &+ \widehat{u}_n\frac{\partial \widehat{u}_n}{\partial x}(x,\xi) \\ &- \nu\frac{\partial^2 \widehat{u}_n}{\partial x^2}(x,\xi))d\xi \end{aligned} \tag{4.26}$$

To make this correction functional stationary, $\delta u_n(x,0) = 0$ having we derive:

$$\delta u_{n+1}(x,t) = \delta u_n(x,t) + \int_0^t \lambda(\xi)(\delta u_n(x,\xi))' d\xi. \tag{4.27}$$

Its stationary conditions can be determined as follows:

$$\lambda'(\xi) = 0 \quad 1 + \lambda(\xi)|_{\xi=t} = 0$$

From which the Lagrange multiplier can be identified $\lambda = -1$, and the following iteration formula is obtained:

$$\begin{aligned} u_{n+1}(x,t) = u_n(x,t) &- \int_0^t (\frac{\partial u_n}{\partial \xi}(x,\xi) + u_n\frac{\partial u_n}{\partial x}(x,\xi) \\ &- \nu\frac{\partial^2 u_n}{\partial x^2}(x,\xi))d\xi \end{aligned} \tag{4.28}$$

Beginning with $u_0 = u(x,0) = f(x)$ the approximate solution of (1) can be determined by iterative formula (28). Similarly, to solve Eq. (1) with boundary conditions (2) by VIM we have:

$$\begin{aligned} u_{n+1}(x,t) &= u_n(x,t) \\ &+ \int_0^t \lambda(\eta)(\frac{\partial u_n}{\partial \xi}(\eta,t) \\ &+ \widehat{u}_n \frac{\partial \widehat{u}_n}{\partial x}(\eta,t) \\ &- \nu \frac{\partial^2 \widehat{u}_n}{\partial x^2}(\eta,t))d\eta. \qquad (4.29) \end{aligned}$$

To find the optimal value of λ, having $\delta u_n(0,t) = 0$ leads to:

$$\begin{aligned} \delta u_{n+1}(x,t) &= \delta u_n(x,t) - \nu \delta u_n^{'}(\eta,t)|_0^x \\ &- \lambda^{'}(\eta)(\delta u_n(\eta,t))|_0^x \\ &+ \int_0^t \lambda^{"}(\eta)(\delta u_n(\eta,t))d\eta. \qquad (4.30) \end{aligned}$$

Therefore, the stationary conditions are obtained as:

$$\begin{aligned} \lambda^{"}(\eta) &= 0 \\ 1 + \nu\lambda^{'}(\eta)|_{\eta=x} &= 0 \\ \lambda(\eta)|_{\eta=x} &= 0 \end{aligned}$$

This results in $\lambda(\eta) = \frac{1}{\nu}(x-\eta)$ and a desired iterative relation can be constructed as:

$$\begin{aligned} u_{n+1}(x,t) &= u_n(x,t) \\ &+ \frac{1}{\nu}\int_0^t (x-\eta)(\frac{\partial u_n}{\partial \xi}(\eta,t) \\ &+ u_n \frac{\partial u_n}{\partial x}(\eta,t) \\ &- \nu \frac{\partial^2 u_n}{\partial x^2}(\eta,t))d\eta. \qquad (4.31) \end{aligned}$$

Beginning with the $u_0 = f_1(t) + xf_2(t)$ an approximate solution of (1) can be determined via the iterative formula (31).

5 Numerical examples

Problem 1. In this example, the numerical solution of the Burgers equation using the ADM and the MLCN and the comparison between the result and the analytical solution is given [27]

$$u(x,t) = \frac{2x}{1+2t}$$

The above equation for t = 0 has its own initial conditions. With initial and boundary conditions for t, and changing t, we solve the Eq. In this example, the length of time step $\tau = 0.004$ and the node $h_x = 0.0125$ but t changes:

$$u(x,0) = 2x.$$

$$u_{i,0}^n = 0.025 \times i.$$

$$u_{i,1}^n = -\frac{0.05 \times i}{(1+2(n+1)0.004)^2}$$

$$u_{i,2}^n = \left(\frac{-0.1 \times i \times (2+2(n+1)0.004)^2}{(1+2(n+2)0.004)^2}\right) \times \left(\frac{1}{(1+2(n+1)0.004)^2}\right)$$

And finally u_i^n comes in the form below:

$$u_i^n \approx u_{i,0}^n + u_{i,1}^n + u_{i,2}^n.$$

In this example, the greater the Reynolds number, the closer the Adomian answer to the analytical solution answers (See Figure 1 and Table 1, 2, 3), and the smaller the Reynolds number, the closer the MLCN answer to the analytical solution answer (See Figure 2 and Table 4, 5, 6), and if t is smaller the Adomian answer is more precise, so in any Reynolds number if t is considered very small, the Adomian method can be more accurate than the MLCN (See Figure 3 and Table 7, 8, 9) .

Problem 2. In this example, the numerical solution of the Burgers equation using the ADM and the MLCN and the comparison of the result to the analytical solution results are given:

$$u(x,t) = e^{-t}\sin(x).$$

The above equation for t = 0 has its own initial conditions. With the initial and boundary conditions for t, and changing t, we solve Eq. In this example, the length of time step $\tau = 0.004$ and the node $h_x = 0.0125$ but t changes:

$$u(x,0) = \sin(x).$$

In functions which Trigonometric relationships are used that the smaller the Reynolds number the closer the ADM and MLCN answers to the analytical solution results (See Figure 4 and Table 10, 11), And when t is larger can be seen that in greater Reynolds numbers Again, the answer to both ADM and MLCN method is closer to the analytical solution (See Figure 5 and Table 12, 13), but in such functions the MLCN works better than the ADM method in larger t amounts and with smaller Reynolds numbers the answer is more precise .

6 Discussion and conclusion

The Burgers equation is a mix of convection and diffusion sentences, and is a simplified form of the Navier-Stockes equation. The Adomian numerical decomposition method has been presented for solving the one-dimensional Burgers equations and has been compared to the modified Local Crank-Nicolson numerical method, and the results of the two methods have been compared to the results of the analytical method. MLCN method is an explicit unconditional stability. Two examples with different initial conditions have been solved and the accuracy of the solution methods ADM and MLCN in different Reynolds numbers has been studied. It has been observed that in Trigonometric functions with smaller Reynolds numbers and larger t amounts the MLCN method behaves better than the ADM method.

References

[1] H. Bateman, *Some recent researches on the motion of fluids*, Mon. Weather Rev, 43 (1915) 163-170.

[2] M. Beck, *Burgers Equation. Department of Mathematics*, (2012) 1-4.

[3] H. Aminikhah, *An analytical approximation for coupled viscous Burgers equation*, Appl. Math. Modelling, 37 (2013) 5979-5983.

[4] A.H. Salas, *Symbolic computation of solutions for a forced Burgers equation*, Appl. Math. Comput, 216 (2010) 18-26.

[5] A.M. Wazwaz, *Multiple soliton solutions and rational solutions for the (2+1)-dimensional dispersive long waterwave system*, Ocean Engineering, 60 (2013) 95-98.

[6] C.Q. Dai, and F.B. Yu, *Special solitonic localized structures for the one-dimensional Burgers equation in water waves*, Wave Motion, 51 (2014) 52-59.

[7] J. Zhou, *Novel soliton-like and multi-solitary wave solutions of (3+1)-dimensional Burgers equation*, Appl. Math. Comput, 204 (2008) 461-467.

[8] A.M. Wazwaz, *A study on the one-dimensional and the two-dimensional higher-order Burgers equations*, Appl. Math. Letters, 25 (2012) 1495-1499.

[9] F. Kong, and S. Chen, *New exact soliton-like solutions and special soliton-like structures of the (2+1) dimensional Burgers equation*, Chaos, Solitons and Fractals, 27 (2006) 495-500.

[10] E. Demetriou, M.A. Christou, and C. Sophocleous, *On the classification of similarity solutions of a two-dimensional diffusionadvection equation* , Appl. Math. Comput, 187 (2007) 1333-1350.

[11] G. Zhao, X. Yu, and R. Zhang, *he new numerical method for solving the system of two-dimensional Burgers equations* , Appl. Math. Comput, 62 (2011) 3279-3291.

[12] M.A. Christou, and C. Sophocleous, *Numerical similarity reductions of the (1+3)-dimensional Burgers equation* ,Appl. Math. Comput, 217 (2011) 7455-7461.

[13] W. Liao, *An implicit fourth-order compact finite difference scheme for one-dimensional Burgers equation*, Appl. Math. Comput, 206 (2008) 755-764.

[14] A.H.A.E. Tabatabaei, E. Shakour, and M. Dehghan, *Some implicit methods for the numerical solution of Burgers equation* , Appl. Math. Comput, 191 (2007) 560-570.

[15] H. Chen, and Z. Jiang, *A characteristics-mixed finite element method for Burgers equation. Korean* , Appl. Math. Comput. J. 15 (2004) 29-51.

[16] J. D. Cole, *On a Quasi-Linear Parabolic Equation Occurring in Aerodynamics*,Quart. Appl. Math. 9 (1951) 225-236.

[17] S. Abbasbandy, M. T. Darvishi, *A numerical solution of Burgers equation by time discretization of Adomians decomposition method*, Appl. Math. Comput. 170 (2005) 95-102.

[18] S. Abbasbandy, M. T. Darvishi, *A numerical solution of Burgers equation by modified Adomian method* , Appl. Math. Comput. 163 (2005) 1265-1272.

[19] H. Zhu, H. Shu, M. Ding, *Numerical solutions of two-dimensional Burgers equations by discrete Adomian decomposition method* , Appl. Math. Comput. 60 (2010) 840-848.

[20] G. Adomian, *Application of the decomposition method to the Navier-Stokes equations* , Appl. Math. Analytic. J. 119 (1986) 340-360.

[21] A. M. Abduwali, *A corrector local C-N method for the two-dimensional heat equation)*, Math. Numer. Shin. 19 (1997) 267-276.

[22] A. M. Abduwali, M. Sakakihara, and H. Niki, *A local Crank-Nicolson method for solving the heat equation* , Hiroshima Math. J. 24 (1994) 1-13

[23] A. M Wazwaz, *A (3+1)-dimensional nonlinear evolution equation with multiple soliton solutions and multiple singular soliton solutions* , Appl. Math. Comput. 215 (2009) 1548-1552.

[24] P. Huang, A. Abduwali, *The Modified Local Crank-Nicolson method for one- and two-dimensional Burgers equations* , Appl. Math. Comput. 59 (2010) 2452-2463.

[25] E. R. Benton, G. W. Platzman, *A table of solutions of the one-dimensional Burgers equation* , Quarterly of Appl. Math 30 (1972) 195-216.

[26] A. R. Bahadr, *A fully implicit finite-difference scheme for two-dimensional Burgers equations* , Appl. Math. Comput. 137 (2003) 131-137.

[27] J. H. A. Biazar, *Exact and numerical solutions for non-linear Burgers equation by VIM*, Math. Comput. Modelling 49 (2009) 1394-1400.

Ahmad Reza Haghighi is an Assistant Professor in the department of mathematics at Urmia University of technology, Urmia, Iran. He completed his Ph. D degree in applied mathematics from Pune University, India. His research interest includes Bio-Mathematics, Computational Fluid Dynamics, Partial Differential Equations and Control

Masoumeh Shojaeifard was born in 1989 in Kangavar, Iran. He received his B.Sc. degree in applied mathematics from Kermanshah University, Iran in 2012. Also, she holds master degree in applied mathematics from Urmia University of Technology, Urmia, Iran in 2014. His research interests include Fluid mechanics, Dynamical systems numerical methods, numerical integration.