Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.12785/amis/090130

An Efficient Numerical Scheme for Coupled Nonlinear Burgers' Equations

Maryam Sarboland and Azim Aminataei∗

Faculty of Mathematics, Department of Applied Mathematics, K. N. Toosi University of Technology, 16315-1618, Tehran, Iran

Received: 30 Mar. 2014, Revised: 30 Jun. 2014, Accepted: 1 Jul. 2014
Published online: 1 Jan. 2015

Abstract: In this paper, coupled nonlinear Burgers' equations are solved through a variety of meshless methods known as multiquadric quasi-interpolation scheme. In this scheme, the extension of univariate quasi-interpolation method is used to approximate the unknown functions and their spatial derivatives and the Taylor series expansion is used to discretize the temporal derivatives. The multiquadric quasi-interpolation scheme is a one-dimensional method that can be extend to the two-dimensional by converting to a compact form and using of a tensor product scheme. The method is tested on three experiments to show the efficiency and accuracy of it. Also, we demonstrate the validity and applicability of our method by error analysis technique based on residual function.

Keywords: Coupled nonlinear Burgers' equations, Quasi-interpolation scheme, Radial basis functions, Taylor series expansion, Residual error technique.

1 Introduction

The nonlinear Burgers' equation is a fundamental partial differential equation (PDE) from fluid mechanics. It appears in various areas of applied mathematics and physics such as the phenomena of turbulence and supersonic flow, flow of a shock wave traveling in a viscous fluid, sedimentation of two kinds of particles in fluid suspensions under the effect of gravity, acoustic transmission, heat conduction, traffic and aerofoil flow theory [2,5,24,29,30,31]. Due to its wide range of applicability some researchers have been interested in studying its solution using various numerical methods such as, finite difference, finite element, discrete Adomian decomposition, spectral and Eulerian-Lagrangian methods. For a survey of these methods, one refers to [1, 10,16,28,29,32] and references cited therein.

Recently, considerable attention has been given to radial basis function (RBF) meshless method for solving the various types of PDEs. Contrary to the mesh based methods like the finite difference and finite element methods, meshless methods use a set of uniform or random points which are not necessarily interconnected in the form of a mesh. Due to this advantageous feature, meshless methods have got increased acceptance since mesh generation in multi-dimensional problems is a non trivial task. Kansa [19] was the first researcher who derived a meshless method based on multiquadric (MQ) RBFs for the numerical solution of PDEs. This idea was extended later on by Golberg et al. [13]. The existence, uniqueness, and convergence of the RBFs approximations was studied by Micchelli [23] , Madych [22], Frank and Schaback [12]. In 1986, Micchelli has shown that the system of equations obtained from the RBFs approximations is always solvable for distinct interpolation points. The authors have recently used the radial basis collocation method to obtain meshless numerical solution of the nonlinear coupled PDEs.

In recent years, other meshless method is proposed based on MQ namely as MQ quasi-interpolation scheme that do not require to solve any linear system of equation and one do not meet the question of the ill-condition of the matrix. Therefore one can save the computational time and decrease the numerical error. Hon and Wu [14], Wu [26] and others have provided some successful examples using MQ quasi-interpolation scheme for solving differential equations. Beatson and Powell [4] proposed three univariate MQ quasi-interpolations, namely, $\mathscr{L}_A$, $\mathscr{L}_B$ and $\mathscr{L}_C$. Wu and Schaback [27] presented the univariate MQ quasi-interpolation $\mathscr{L}_D$ and proved that the scheme is shape preserving and convergent. In [7,8], Chen and Wu used MQ

∗ Corresponding author e-mail: ataei@kntu.ac.ir

quasi-interpolation to solve Burgers' equation and hyperbolic conservation laws. Recently, Jiang et al. [17] have introduced a new multi-level univariate MQ quasi-interpolation approach with high approximation order compared with initial MQ quasi-interpolation scheme, namely $\mathscr{L}_{\mathscr{W}}$ and $\mathscr{L}_{\mathscr{W}2}$. This approach is based on inverse multiquadric (IMQ) RBF interpolation, and Wu and Schaback's MQ quasi-interpolation operator $\mathscr{L}_D$ that have the advantages of high approximation order.

The Jiang et al. MQ quasi-interpolation operator $\mathscr{L}_{\mathscr{W}2}$ is one-dimensional operator that uses for interpolation of univariate functions. Also, it is summation of two series that the second series coefficients are combined of first series coefficients. By giving relation between two series coefficients based on function values, we can convert it to a compact form which is based on one series and extend it to operator that can be used for higher dimensions. Also, we do not require to solve any linear system of equation for getting of the first series coefficients at each time step, see [18]. In numerical solution of time dependent PDEs, such as Burgers' and Sine-Gordon equations, by using MQ quasi-interpolation scheme, if we do not want to solve a system of equations at each time step it is necessary to use a low order finite difference approximation for discretization of time derivatives (see, [7,18]) else higher-order approximations can be applied to discretize the time derivatives (see [15]).

In this paper, the extension of the MQ quasi-interpolation scheme and the differential of it are used to approximate the solution functions and their spatial derivatives, respectively. Also, a two order approximation based on Taylor series expansion is used for discretization of the temporal derivatives. Hence, the system of equations must be solved at each time step.

The organization of this paper is as follows. In Section 2, we describe the MQ quasi-interpolation scheme. In Section 3, the method is applied on the two-dimensional nonlinear Burgers' equations. The error analysis technique based on the residual function is developed for the present method in Section 4. In Section 5, the results of three experiments are reported and compared with the analytical solutions and the results in [1,16,32]. Finally, a brief discussion and conclusion is presented in Section 6.

2 The MQ quasi-interpolation scheme

In this section, at first, we review some elementary knowledge about three univariate MQ quasi-interpolation operators $\mathscr{L}_{\mathscr{D}}, \mathscr{L}_{\mathscr{W}}$ and $\mathscr{L}_{\mathscr{W}_2}$. Then, we describe our approach which converts operator $\mathscr{L}_{\mathscr{W}_2}$ to the compact form. For more information about MQ quasi-interpolation operators see [4,17,27].

For a given interval $\Omega = [a,b]$ and a finite set of distinct points

$$a = x_0 < x_1 < \ldots < x_{\mathscr{N}} = b, \quad h = \max_{1\leqslant i\leqslant \mathscr{N}} (x_i - x_{i-1}),$$

quasi-interpolation of a univariate function $f : [a,b] \to \mathbb{R}$ is given by

$$\mathscr{L}(f) = \sum_{i=0}^{\mathscr{N}} f(x_i)\phi_i(x),$$

where function $\phi_i(x)$ is a linear combination of the MQs

$$\psi_i(x) = \sqrt{c^2 + (x - x_i)^2},$$

and $c \in \mathbb{R}^+$ is a shape parameter. In [27], Wu and Scheback presented the univariate MQ quasi-interpolation operator $\mathscr{L}_{\mathscr{D}}$ that is defined as

$$\mathscr{L}_{\mathscr{D}} f(x) = \sum_{i=0}^{\mathscr{N}} f(x_i)\widetilde{\psi}_i(x), \tag{1}$$

where

$$\widetilde{\psi}_0(x) = \frac{1}{2} + \frac{\psi_1(x) - (x - x_0)}{2(x_1 - x_0)},$$

$$\widetilde{\psi}_1(x) = \frac{\psi_2(x) - \psi_1(x)}{2(x_2 - x_1)} - \frac{\psi_1(x) - (x - x_0)}{2(x_1 - x_0)},$$

$$\widetilde{\psi}_i(x) = \frac{\psi_{i+1}(x) - \psi_i(x)}{2(x_{i+1} - x_i)} - \frac{\psi_i(x) - \psi_{i-1}(x)}{2(x_i - x_{i-1})}, \quad 2 \leqslant i \leqslant \mathscr{N} - 2, \tag{2}$$

$$\widetilde{\psi}_{\mathscr{N}-1}(x) = \frac{(x_{\mathscr{N}} - x) - \psi_{\mathscr{N}-1}(x)}{2(x_{\mathscr{N}} - x_{\mathscr{N}-1})} - \frac{\psi_{\mathscr{N}-1}(x) - \psi_{\mathscr{N}-2}(x)}{2(x_{\mathscr{N}-1} - x_{\mathscr{N}-2})},$$

and

$$\widetilde{\psi}_{\mathscr{N}}(x) = \frac{1}{2} + \frac{\psi_{\mathscr{N}-1}(x) - (x_{\mathscr{N}} - x)}{2(x_{\mathscr{N}} - x_{\mathscr{N}-1})}.$$

In RBFs interpolation, high approximation order can be gotten by increasing the number of interpolation centers but we have to solve unstable linear system of equations. By using MQ quasi-interpolation scheme, we can avoid this problem, whereas, the approximation order is not good. Therefore, Jiang et al. [17] defined two MQ quasi-interpolation operators denoted as $\mathscr{L}_{\mathscr{W}}$ and $\mathscr{L}_{\mathscr{W}_2}$, which pose the advantages of RBFs interpolation and MQ quasi-interpolation scheme. The process of MQ quasi-interpolation of $\mathscr{L}_{\mathscr{W}}$ and $\mathscr{L}_{\mathscr{W}_2}$ are as follows that is described in [17].

Suppose that $\{x_{k_j}\}_{j=1}^{\bar{\mathscr{N}}}$ is a smaller set from the given points $\{x_i\}_{i=0}^{\mathscr{N}}$ where $\bar{\mathscr{N}}$ is a positive integer satisfying $\bar{\mathscr{N}} < \mathscr{N}$ and $0 = k_0 < k_1 < \ldots < k_{\bar{\mathscr{N}}+1} = \mathscr{N}$. Using the IMQ-RBF, the second derivative of $f(x)$ can be approximated by RBF interpolant $S_{f''}$ as

$$S_{f''}(x) = \sum_{j=1}^{\bar{\mathscr{N}}} \alpha_j \bar{\varphi}(|x - x_{k_j}|), \tag{3}$$

where

$$\bar{\varphi}(r) = \frac{s^2}{(s^2 + r^2)^{3/2}},$$

and $s \in \mathbb{R}^+$ is a shape parameter. The coefficients $\{\alpha_j\}_{j=1}^{\bar{\mathcal{N}}}$ are uniquely determined by the interpolation condition

$$S_{f''}(x_{k_i}) = \sum_{j=1}^{\bar{\mathcal{N}}} \alpha_j \bar{\varphi}(|x_{k_i} - x_{k_j}|) = f''(x_{k_i}), \quad 1 \leqslant i \leqslant \bar{\mathcal{N}}. \quad (4)$$

Since, the Eq. (4) is solvable [22], so

$$\alpha = A_X^{-1}.f_X'', \quad (5)$$

where

$$X = \{x_{k_1}, \ldots, x_{k_{\bar{\mathcal{N}}}}\}, \quad \alpha = [\alpha_1, \ldots, \alpha_{\bar{\mathcal{N}}}]^T,$$

and

$$A_X = [\bar{\varphi}(|x_{k_i} - x_{k_j}|)], \quad f_X'' = [f''(x_{k_1}), \ldots, f''(x_{k_{\bar{\mathcal{N}}}})]^T.$$

By using f and the coefficient α defined in Eq. (5), a function $e(x)$ is constructed in the form

$$e(x) = f(x) - \sum_{i=1}^{\bar{\mathcal{N}}} \alpha_i \sqrt{s^2 + (x - x_{k_i})^2}. \quad (6)$$

Then the MQ quasi-interpolation operator $\mathcal{L}_{\mathcal{W}}$ by using $\mathcal{L}_{\mathcal{D}}$ defined by Eqs. (1) and (2) on the data $(x_i, e(x_i))_{0 \leqslant i \leqslant \mathcal{N}}$ with the shape parameter c is defined as follows:

$$\mathcal{L}_{\mathcal{W}} f(x) = \sum_{i=1}^{\bar{\mathcal{N}}} \alpha_i \sqrt{s^2 + (x - x_{k_i})^2} + \mathcal{L}_{\mathcal{D}} e(x). \quad (7)$$

The shape parameters c and s should not be the same constant in Eq. (7). In Eq. (4), the value of $f''_{x_{k_j}}$ can be replaced by

$$f''_{x_{k_j}} = \frac{2[(x_{k_j} - x_{k_{j-1}}) f(x_{k_{j+1}}) - (x_{k_{j+1}} - x_{k_{j-1}}) f(x_{k_j}) + (x_{k_{j+1}} - x_{k_j}) f(x_{k_{j-1}})}{(x_{k_j} - x_{k_{j-1}})(x_{k_{j+1}} - x_{k_j})(x_{k_{j+1}} - x_{k_{j-1}})}$$

when the data's $\{(x_{k_i}, f(x_{k_i}))\}_{i=1}^{\bar{\mathcal{N}}}$ are given. So, if f_X'' in Eq. (5) is replaced by

$$F_X'' = [f''_{x_{k_1}}, \ldots, f''_{x_{k_{\bar{\mathcal{N}}}}}]^T, \quad (8)$$

then the quasi-interpolation operator defined by Eqs. (6) and (7) is denoted by $\mathcal{L}_{\mathcal{W}_2}$. The linear reproducing property and the high convergence rate of $\mathcal{L}_{\mathcal{W}2}$ were also studied in [17].

The operator $\mathcal{L}_{\mathcal{W}_2}$ can be written in the compact form

$$\mathcal{L}_{\mathcal{W}_2} f(x) = \sum_{i=0}^{\mathcal{N}} f(x_i) \widehat{\psi}_i(x), \quad (9)$$

where the basis functions $\widehat{\psi}_i(x)$ are obtained by substituting Eqs. (5), (6) and (8) into Eq. (7). Such as, let $X = \{x_0, x_1, x_2, x_3, x_4\}$ and $X' = \{x_2\}$. So $\mathcal{N} = 4$, $\bar{\mathcal{N}} = 1$, $k_1 = 2$ and

$$\alpha = \frac{2f(x_{k_2})}{s(x_{k_2} - x_{k_1})(x_{k_2} - x_{k_0})} - \frac{2f(x_{k_1})}{s(x_{k_1} - x_{k_0})(x_{k_2} - x_{k_1})} + \frac{2f(x_{k_0})}{s(x_{k_1} - x_{k_0})(x_{k_2} - x_{k_0})}, \quad (10)$$

$$e(x) = f(x) - \alpha \sqrt{s^2 + (x - x_{k_1})^2}. \quad (11)$$

Substituting Eq. (10) into Eq. (11), yields

$$e(x) = f(x) - [\frac{2f(x_{k_2})}{s(x_{k_2} - x_{k_1})(x_{k_2} - x_{k_0})} - \frac{2f(x_{k_1})}{s(x_{k_1} - x_{k_0})(x_{k_2} - x_{k_1})} + \frac{2f(x_{k_0})}{s(x_{k_1} - x_{k_0})(x_{k_2} - x_{k_0})}] \sqrt{s^2 + (x - x_{k_1})^2}. \quad (12)$$

whereas $f(x_{k_0}) = f(x_0)$, $f(x_{k_1}) = f(x_2)$ and $f(x_{k_2}) = f(x_4)$. Hence, the substitution of Eqs. (10) and (12) into Eq. (7), leads to

$$\mathcal{L}_{\mathcal{W}2} f(x) = [\frac{2f(x_4)}{s(x_4 - x_2)(x_4 - x_0)} - \frac{2f(x_2)}{s(x_2 - x_0)(x_4 - x_2)} + \frac{2f(x_0)}{s(x_2 - x_0)(x_4 - x_0)}] \sqrt{s^2 + (x - x_2)^2} + \sum_{i=0}^{4} f(x_i) \widetilde{\psi}_i(x) - \sum_{i=0}^{4} [\frac{2f(x_4)}{s(x_4 - x_2)(x_4 - x_0)} - \frac{2f(x_2)}{s(x_2 - x_0)(x_4 - x_2)} + \frac{2f(x_0)}{s(x_2 - x_0)(x_4 - x_0)}] \sqrt{s^2 + (x_i - x_2)^2} \widetilde{\psi}_i(x).$$

Hence, the basic functions $\widehat{\psi}_i(x)$ are arrived as follows:

$$\widehat{\psi}_0(x) = \frac{2[\sqrt{s^2 + (x - x_2)^2} - \chi(x)]}{s(x_2 - x_0)(x_4 - x_0)} + \widetilde{\psi}_0(x),$$

$$\widehat{\psi}_2(x) = \frac{-2[\sqrt{s^2 + (x - x_2)^2} - \chi(x)]}{s(x_2 - x_0)(x_4 - x_2)} + \widetilde{\psi}_2(x),$$

$$\widehat{\psi}_4(x) = \frac{2[\sqrt{s^2 + (x - x_2)^2} - \chi(x)]}{s(x_4 - x_2)(x_4 - x_0)} + \widetilde{\psi}_4(x),$$

and

$$\widehat{\psi}_i(x) = \widetilde{\psi}_i(x), \quad i = 1, 3,$$

where $\chi(x) = \sum_{i=0}^{4} \sqrt{s^2 + (x_i - x_2)^2} \widetilde{\psi}_i(x)$. In this paper, $\mathcal{N} = 2\bar{\mathcal{N}}$ is considered.

By converting operator (7) to the compact form of (9), we can easily extend MQ quasi-interpolation scheme to two-dimension space. Also, we will not require to solve any linear system of equations for getting of the coefficients α_i at each time step when MQ quasi-interpolation is used for solving of PDEs.

3 The numerical method

We consider the two-dimensional coupled nonlinear Burgers' equations:

$$u_t + uu_x + vu_y = \frac{1}{R}(u_{xx} + u_{yy}), \quad (13)$$

$$v_t + uv_x + vv_y = \frac{1}{R}(v_{xx} + v_{yy}), \quad (14)$$

with the initial conditions:

$$u(x, y, 0) = f_1(x, y), \quad (x, y) \in \Omega, \quad (15)$$

Table 1: Comparison of numerical solutions with the solutions in [1,32] of u at $t=0.01$ and $t=0.5$ with $R=100$ of experiment 1.

Points	$t=0.01$				$t=0.5$			
	MQQI $N=121$	FDM [1] $N=441$	ADM [32] $N=441$	Exact	MQQI $N=121$	FDM [1] $N=441$	ADM [32] $N=441$	Exact
(0.1,0.1)	1.16E-04	5.30E-05	5.91E-05	0.62305	3.28E-03	9.72E-04	2.78E-04	0.54332
(0.5,0.1)	7.45E-06	1.21E-05	4.84E-06	0.50162	7.11E-05	7.13E-04	4.52E-04	0.50035
(0.9,0.1)	9.51E-07	1.10E-05	3.41E-08	0.50001	1.39E-04	6.92E-04	3.37E-06	0.50000
(0.3,0.3)	2.58E-05	6.30E-05	5.91E-05	0.62305	2.15E-04	1.25E-03	2.78E-04	0.54332
(0.7,0.3)	7.33E-07	2.07E-06	4.84E-06	0.50162	1.89E-05	7.43E-04	4.52E-04	0.50035
(0.1,0.5)	8.80E-06	4.04E-06	1.64E-06	0.74827	6.51E-04	9.14E-04	2.86E-04	0.74221
(0.5,0.5)	2.58E-05	6.30E-05	5.91E-05	0.62305	5.01E-04	1.10E-03	2.78E-04	0.54332
(0.9,0.5)	4.43E-06	2.07E-06		0.50162	2.57E-04	3.83E-04		0.50035
(0.3,0.7)	1.55E-06	4.04E-06		0.74827	7.61E-07	7.64E-04		0.74221
(0.7,0.7)	2.59E-05	6.30E-05		0.62305	1.15E-04	8.92E-04		0.54332
(0.1,0.9)	4.83E-07	8.29E-06		0.74999	1.76E-06	8.16E-04		0.74995
(0.5,0.9)	2.65E-06	4.04E-06		0.74827	1.24E-04	2.04E-04		0.74221
(0.9,0.9)	1.10E-04	6.30E-05		0.62305	3.74E-04	1.00E-03		0.54332

Table 2: Comparison of numerical solutions with the solutions in [1,32] of v at $t=0.01$ and $t=0.5$ with $R=100$ of experiment 1.

Points	$t=0.01$				$t=0.5$			
	MQQI $N=121$	FDM [1] $N=441$	ADM [32] $N=441$	Exact	MQQI $N=121$	FDM [1] $N=441$	ADM [32] $N=441$	Exact
(0.1,0.1)	1.16E-04	7.30E-05	5.91E-05	0.87695	3.28E-03	9.08E-04	2.78E-04	0.95668
(0.5,0.1)	7.45E-06	7.93E-06	4.84E-06	0.99838	7.11E-05	1.38E-03	4.52E-04	0.99965
(0.9,0.1)	9.51E-07	9.00E-06	3.41E-08	0.99999	1.39E-04	1.39E-03	3.37E-06	1.00000
(0.3,0.3)	2.58E-05	6.30E-05	5.91E-05	0.87695	2.15E-04	7.18E-04	2.78E-04	0.95668
(0.7,0.3)	7.33E-07	2.07E-06	4.84E-06	0.99838	1.89E-05	1.38E-03	4.52E-04	0.99965
(0.1,0.5)	8.80E-06	5.96E-06	1.64E-06	0.75173	6.51E-04	7.96E-04	2.86E-04	0.75779
(0.5,0.5)	2.58E-05	6.30E-06	5.91E-05	0.87695	5.01E-04	1.72E-04	2.78E-04	0.95668
(0.9,0.5)	4.43E-06	2.07E-06		0.99838	2.57E-04	6.17E-04		0.99965
(0.3,0.7)	1.55E-06	4.04E-06		0.75173	7.61E-07	5.56E-04		0.75779
(0.7,0.7)	2.59E-05	6.30E-05		0.87695	1.15E-04	7.82E-04		0.95668
(0.1,0.9)	4.83E-07	1.71E-06		0.75001	1.76E-06	8.14E-04		0.75005
(0.5,0.9)	2.65E-06	4.04E-06		0.75173	1.24E-04	2.40E-05		0.75779
(0.9,0.9)	1.10E-04	6.30E-05		0.87695	3.74E-04	1.09E-03		0.95668

$$v(x,y,0)=f_2(x,y),\qquad (x,y)\in\Omega, \tag{16}$$

and the boundary conditions:

$$u(x,y,t)=g_1(x,y,t),\qquad (x,y)\in\Gamma, \tag{17}$$

$$v(x,y,t)=g_2(x,y,t),\qquad (x,y)\in\Gamma, \tag{18}$$

where $\Omega=\{(x,y)|a\leqslant x\leqslant b, c\leqslant y\leqslant d\}$ and Γ is its boundary. $u(x,y,t)$ and $v(x,y,t)$ are the two unknown variables which can be regarded as the velocities in fluid-related problems. $f_1(x,y)$, $f_2(x,y)$, $g_1(x,y,t)$ and $g_2(x,y,t)$ are all known functions and R is the Reynolds number.

First, we discretize Eq. (13) in time with time step Δt by using Taylor series expansion. The main idea behind the discretization is to use more time derivatives in Taylor series expansion. This approach was demonstrated by Lax and Wendroff in finite difference [20] and used by Dağ for the one-dimensional Burgers' equation in [9]. In this approach, the term $u_t^n=u_t(x,t_n)$ is arranged with the help of Taylor series expansion as

$$u_t^n=\frac{u^{n+1}-u^n}{\Delta t}-\frac{\Delta t}{2}u_{tt}^n+O(\Delta t^2). \tag{19}$$

Differentiating Eq. (13) with respect to time, u_{tt}^n may be written as

$$u_{tt}^n=(\mu u_{xx}^n+\mu u_{yy}^n-u^nu_x^n-v^nu_y^n)_t=\mu(u_t^n)_{xx}$$
$$+\mu(u_t^n)_{yy}-u^n(u_t^n)_x-u_t^nu_x^n-v^n(u_t^n)_y-v_t^nu_y^n. \tag{20}$$

where $\mu=\frac{1}{R}$.

For the time derivative u_t^n in Eq. (20), using forward difference formula, u_{tt}^n can be rewritten as:

$$\Delta t u_{tt}^n=\mu(u_{xx}^{n+1}-u_{xx}^n)+\mu(u_{yy}^{n+1}-u_{yy}^n)-u^n(u_y^{n+1}-u_x^n)$$
$$-(u^{n+1}-u^n)u_x^n-v^n(u_y^{n+1}-u_y^n)-u_y^n(v^{n+1}-v^n). \tag{21}$$

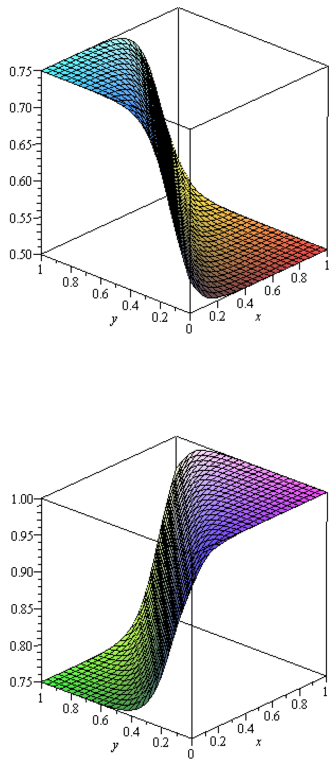

Fig. 1: The estimated solutions $u(x,y,t)$ (up) and $v(x,y,t)$ (down) at $t=0.5$ with $R=100$, $\Delta t=0.001$ and $N=121$ of experiment 1.

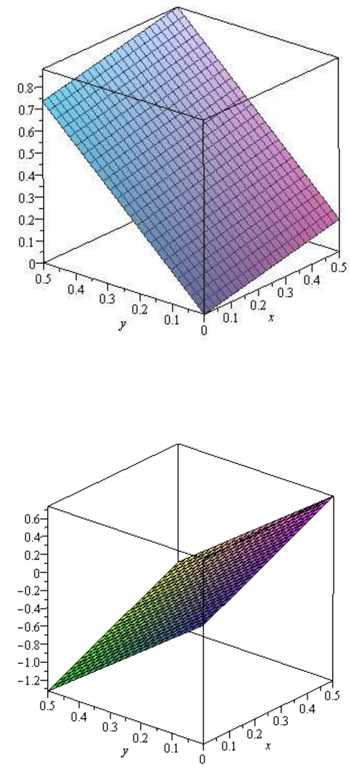

Fig. 2: The estimated solutions $u(x,y,t)$ (up) and $v(x,y,t)$ (down) at $t=0.4$ with $\Delta t=0.001$ and $N=25$ of experiment 2.

Substituting Eq. (21) into Eq. (19) and using the expression achieved in Eq. (13), yields the following time discretized form of Burgers' equation (13):

$$2u^{n+1}+\Delta t(u^n u_x^{n+1}+u_x^n u^{n+1}+v^n u_y^{n+1}+v^{n+1}u_y^n)$$

$$-\mu\Delta t(u_{xx}^{n+1}+u_{yy}^{n+1})=2u^n+\mu\Delta t(u_{xx}^n+u_{yy}^n). \quad (22)$$

Also, the time discretized form of Eq. (14) is given as follows:

$$2v^{n+1}+\Delta t(u^n v_x^{n+1}+v_x^n u^{n+1}+v^n v_y^{n+1}+v^{n+1}v_y^n)$$

$$-\mu\Delta t(v_{xx}^{n+1}+v_{yy}^{n+1})=2v^n+\mu\Delta t(v_{xx}^n+v_{yy}^n). \quad (23)$$

Now, we use the multivariate quasi-interpolation scheme for approximation of u and v similar to work that Ling did in [21]. In this scheme, u^n and v^n are approximated as follows:

$$u^n(x,y)=\sum_{i=0}^{n_x}\sum_{j=0}^{n_y}u_{ij}^n\widehat{\psi}_i(x)\widehat{\psi}_j(y), \quad (24)$$

and

$$v^n(x,y)=\sum_{i=0}^{n_x}\sum_{j=0}^{n_y}v_{ij}^n\widehat{\psi}_i(x)\widehat{\psi}_j(y), \quad (25)$$

where $\widehat{\psi}_i(x)$ and $\widehat{\psi}_j(y)$ are the known basis functions derived from one-dimensional basis functions, is defined in Eq. (9), associated with the x and y directions, respectively; and u_{ij} and v_{ij} are the values of u and v at the intersection of the ith horizontal grid line and the jth vertical grid line, respectively. We can rewrite Eqs. (24) and (25) as follows:

$$u^n(x,y)=\sum_{i=1}^{N}u_i^n\varphi_i(x,y), \qquad v^n(x,y)=\sum_{i=1}^{N}v_i^n\varphi_i(x,y), \quad (26)$$

where basis functions $\varphi(x,y)$ are given by tensor product $\widehat{\psi}(x)$ in $\widehat{\psi}(y)$ and $N=(n_x+1)(n_y+1)$. Now, By substituting the above approximations into Eqs. (22) and (23) and using collocation method, we obtain the following matrix form:

$$[A]^n[X]^{n+1}=[B]^{n+1}, \quad (27)$$

where $[X]^{n+1} = [u_1^{n+1}, u_2^{n+1}, \ldots, u_N^{n+1}, v_1^{n+1}, v_2^{n+1}, \ldots, v_N^{n+1}]$, $[B]_i^{n+1} = u_i^{n+1}$, $[B]_{i+N}^{n+1} = v_i^{n+1}$ for $(x_i, y_i) \in \partial\Omega$, $[B]_i^{n+1} = 2u_i^n + \mu\Delta t(u_{xx} + u_{yy})_i^n$, $[B]_{i+N}^{n+1} = 2v_i^n + \mu\Delta t(v_{xx} + v_{yy})_i^n$ elsewhere, and the matrix $[A]^n$ can be split into $[A]^n = A_b + [A_d]^n$ where $A_{b_{ij}} = A_{b_{(i+N)(j+N)}} = \varphi_j(x_i, y_i) = \varphi_{ij}$, for $i, j = 1, \ldots, N, (x_i, y_i) \in \Gamma$ and $A_{b_{ij}} = 0$ elsewhere, also

$$A_{d_{ij}} = 2\varphi_{ij} + \Delta t(u_i^n \hat{\varphi}_{ij} + (u_x)_i^n \varphi_{ij} + v_i^n \check{\varphi}_{ij}) - \mu\Delta t(\acute{\varphi}_{ij} + \grave{\varphi}_{ij}),$$

$$A_{d_{(i+N)(j+N)}} = 2\varphi_{ij} + \Delta t(u_i^n \hat{\varphi}_{ij} + (v_y)_i^n \varphi_{ij} + v_i^n \check{\varphi}_{ij})$$
$$-\mu\Delta t(\acute{\varphi}_{ij} + \grave{\varphi}_{ij}),$$

$A_{d_{i(j+N)}} = \Delta t(u_y)_i^n \varphi_{ij}$ and $A_{d_{(i+N)j}} = \Delta t(v_x)_i^n \varphi_{ij}$ for $i, j = 1, \ldots, N$ where $\hat{\varphi}_i = \frac{\partial\varphi}{\partial x}, \check{\varphi}_i = \frac{\partial\varphi}{\partial y}, \acute{\varphi}_i = \frac{\partial^2\varphi}{\partial x^2}$ and $\grave{\varphi}_i = \frac{\partial^2\varphi}{\partial y^2}$. Using the initial conditions and the boundary conditions, represented by equations (15)-(18), equation (27) can give $[X]^{n+1}$.

4 The error estimate

In this section, an error estimation for the approximate solutions of the two-dimensional nonlinear Burgers' equations are obtained. This error estimation has been presented in [25] for integro-differential equations. We modify the error estimation studied in [25] for the two-dimensional nonlinear Burgers' equations.

Let us call $e_{1,N}^n = u^n(x,y) - u_N^n(x,y)$ and $e_{2,N}^n = v^n(x,y) - v_N^n(x,y)$ as the error functions of the approximations $u_N^n(x,y)$ and $v_N^n(x,y)$ to $u^n(x,y)$ and $v^n(x,y)$ at $n-th$ time level, respectively, where $u^n(x,y)$ and $v^n(x,y)$ are the exact solutions of Eqs. (22) and (23). Thus, $u_N^n(x,y)$ and $v_N^n(x,y)$ satisfy the following problems:

$$2u_N^{n+1} + \Delta t(u_N^n u_{N,x}^{n+1} + u_{N,x}^n u_N^{n+1} + v_N^n u_{N,y}^{n+1} + v_N^{n+1} u_{N,y}^n) - \mu\Delta t$$
$$(u_{N,xx}^{n+1} + u_{N,yy}^{n+1}) = 2u_N^n + \mu\Delta t(u_{N,xx}^n + u_{N,yy}^n) + R_{1,N}^{n+1}, \quad (28)$$

$$2v_N^{n+1} + \Delta t(u_N^n v_{N,x}^{n+1} + v_{N,x}^n u_N^{n+1} + v_N^n v_{N,y}^{n+1} + v_N^{n+1} v_{N,y}^n) - \mu\Delta t$$
$$(v_{N,xx}^{n+1} + v_{N,yy}^{n+1}) = 2v_N^n + \mu\Delta t(v_{N,xx}^n + v_{N,yy}^n) + R_{2,N}^{n+1}, \quad (29)$$

where $R_{1,N}^{n+1}(x,y)$ and $R_{2,N}^{n+1}(x,y)$ are the residual function associated with $u_N^{n+1}(x,y)$ and $v_N^{n+1}(x,y)$.
By subtracting (28) and (29) from (22) and (23), respectively, the error functions $e_{1,N}(x,y)$ and $e_{2,N}(x,y)$ satisfy the equations:

$$2e_{1,N}^{n+1} + \Delta t[(e_{1,N}^n + u_N^n)e_{1,N,x}^{n+1} + (e_{1,N,x}^n + u_{N,x}^n)e_{1,N}^{n+1} + (e_{2,N}^n$$
$$+v_N^n)e_{1,N,y}^{n+1} + (e_{1,N,y}^n + u_{N,y}^n)e_{2,N}^{n+1} + u_{N,x}^{n+1} e_{1,N}^n + u_N^{n+1} e_{1,N,x}^n$$
$$+u_{N,y}^{n+1} e_{2,N}^n + v_N^{n+1} e_{1,N,y}^n] - \mu\Delta t(e_{1,N,xx}^{n+1} + e_{1,N,yy}^{n+1})$$
$$= 2e_{1,N}^n + \mu\Delta t(e_{1,N,xx}^n + e_{1,N,yy}^n) - R_{1,N}^{n+1}, \quad (30)$$

and

$$2e_{2,N}^{n+1} + \Delta t[(e_{1,N}^n + u_N^n)e_{2,N,x}^{n+1} + (e_{2,N,x}^n + v_{N,x}^n)e_{1,N}^{n+1} + (e_{2,N}^n$$
$$+v_N^n)e_{2,N,y}^{n+1} + (e_{2,N,y}^n + v_{N,y}^n)e_{2,N}^{n+1} + v_{N,x}^{n+1} e_{1,N}^n + u_N^{n+1} e_{2,N,x}^n$$
$$+v_{N,y}^{n+1} e_{2,N}^n + v_N^{n+1} e_{2,N,y}^n] - \mu\Delta t(e_{2,N,xx}^{n+1} + e_{2,N,yy}^{n+1})$$
$$= 2e_{2,N}^n + \mu\Delta t(e_{2,N,xx}^n + e_{2,N,yy}^n) - R_{2,N}^{n+1}, \quad (31)$$

with the homogeneous initial conditions:

$$e_{1,N}^0(x,y) = e_{2,N}^0(x,y) = 0, \qquad (x,y) \in \Omega, \quad (32)$$

and the homogeneous boundary conditions:

$$e_{1,N}^{n+1}(x,y) = e_{2,N}^{n+1}(x,y) = 0, \qquad (x,y) \in \Gamma. \quad (33)$$

By solving the error problems (30) and (31) by the method described in Section 3, the approximations $e_{1,N,M}^{n+1}$ and $e_{2,N,M}^{n+1}$ to $e_{1,N}^{n+1}$ and $e_{2,N}^{n+1}$ are found, respectively. We note that if the exact solution of the problem is not known, then we can estimate the error functions by $e_{1,N,M}^{n+1}$ and $e_{2,N,M}^{n+1}$.

5 The numerical experiments

Three experiments is studied to investigate the robustness and accuracy of the proposed method. We compare the numerical results of the two-dimensional nonlinear Burgers' equations by using the presented scheme with the analytical solutions and solutions in [1, 16, 32]. We denote our scheme by MQQI. In all of the experiments, the shape parameter $c = 0.815h$ and the shape parameter $s = 2c$ are denoted.

The computations associated with our experiments are performed in Maple 16 on a PC with a CPU of 2.4 GHZ.

Experiment 1. In this experiment, we consider the two-dimensional nonlinear Burgers' equations (13) and (14) with exact solutions

$$u(x,y,t) = \frac{3}{4} - \frac{1}{4[1 + \exp(-4x + 4y - t)/(32\mu)]},$$
$$v(x,y,t) = \frac{3}{4} + \frac{1}{4[1 + \exp(-4x + 4y - t)/(32\mu)]}. \quad (34)$$

Above solutions obtained using a Hopf-Cole transformation in [11]. The initial conditions are obtained from (34) at $t = 0$, and the boundary conditions can be

Table 3: Comparison of the actual and estimated absolute errors of u for various values of N and M at $t = 0.1$ of experiment 1.

Points	$N = 25$			$N = 49$		
	Actual	Estimated		Actual	Estimated	
		$M = 81$	$M = 121$		$M = 81$	$M = 121$
(0.1,0.1)	8.36599E-3	4.11070E-3	4.47301E-3	4.56647E-3	2.01370E-3	2.43801E-3
(0.5,0.1)	9.20573E-3	8.57831E-3	8.40817E-3	7.72798E-4	1.01854E-3	8.66505E-4
(0.9,0.1)	3.68497E-3	4.14192E-3	4.00703E-3	5.60015E-4	5.30799E-4	5.63710E-4
(0.3,0.3)	6.26737E-3	4.98879E-3	5.90883E-3	2.28397E-3	1.99799E-3	1.99956E-3
(0.7,0.3)	3.80637E-3	3.98595E-3	3.82165E-3	1.33129E-3	1.47575E-3	1.33432E-3
(0.1,0.5)	6.50740E-3	5.88473E-3	5.64751E-3	5.22721E-4	4.08371E-4	4.35187E-4
(0.5,0.5)	4.95619E-3	4.21532E-3	4.67833E-3	1.59856E-3	1.29532E-3	1.42484E-3
(0.9,0.5)	8.72283E-3	8.13613E-3	8.96242E-3	1.52989E-3	1.47864E-3	1.48977E-3
(0.3,0.7)	2.99622E-3	3.00702E-3	2.99346E-3	3.38276E-4	3.41523E-4	3.43224E-4
(0.7,0.7)	5.94465E-3	4.84669E-3	5.81430E-3	2.14067E-3	1.83520E-3	1.87950E-3
(0.1,0.9)	2.22972E-3	2.64162E-3	2.47664E-3	1.30503E-4	1.12074E-4	1.32391E-4
(0.5,0.9)	6.17633E-3	5.78486E-3	6.26674E-3	7.01168E-5	6.44743E-5	2.30941E-5
(0.9,0.9)	8.14830E-3	5.47280E-3	6.70321E-3	4.22399E-3	3.50706E-3	3.96765E-3

Table 4: Comparison of the actual and estimated absolute errors of v for various values of N and M at $t = 0.1$ of experiment 1.

Points	$N = 25$			$N = 49$		
	Actual	Estimated		Actual	Estimated	
		$M = 81$	$M = 121$		$M = 81$	$M = 121$
(0.1,0.1)	8.36599E-3	4.02052E-3	4.40868E-3	4.56647E-3	2.00242E-3	2.42845E-3
(0.5,0.1)	9.20573E-3	8.53947E-3	8.37281E-3	7.72798E-4	1.01774E-3	8.67754E-3
(0.9,0.1)	3.68497E-3	4.16592E-3	4.01610E-3	5.60015E-4	5.30799E-4	5.63785E-3
(0.3,0.3)	6.26737E-3	4.95325E-3	5.84778E-3	2.28397E-3	1.99367E-3	1.99810E-3
(0.7,0.3)	3.80637E-3	3.99094E-3	3.82330E-3	1.33129E-3	1.47349E-3	1.33341E-3
(0.1,0.5)	6.50740E-3	5.90057E-3	5.65963E-3	5.22721E-4	4.07764E-4	4.33195E-3
(0.5,0.5)	4.95619E-3	4.20151E-3	4.64820E-3	1.59856E-3	1.29630E-3	1.42400E-3
(0.9,0.5)	8.72283E-3	8.16894E-3	8.96959E-3	1.52989E-3	1.47586E-3	1.48457E-3
(0.3,0.7)	2.99622E-3	2.99749E-3	2.98899E-3	3.38276E-4	3.41985E-4	3.44046E-3
(0.7,0.7)	5.94465E-3	4.85485E-3	5.75885E-3	2.14067E-3	1.83304E-3	1.87742E-3
(0.1,0.9)	2.22972E-3	2.64536E-3	2.47788E-3	1.30503E-4	1.12086E-4	1.32401E-3
(0.5,0.9)	6.17633E-3	5.78838E-3	6.26142E-3	7.01168E-5	6.38928E-5	2.28619E-3
(0.9,0.9)	8.14830E-3	5.42632E-3	6.62525E-3	4.22399E-3	3.50227E-3	3.95704E-3

Table 5: Comparison of numerical solutions with the solutions in [32] for u at $t = 0.1$ and $t = 0.4$ of experiment 2.

Points	$t = 0.1$				$t = 0.4$			
	MQQI	Error	ADM[32]	Error	MQQI	Error	ADM[32]	Error
	$N = 25$		$N = 441$		$N = 25$		$N = 441$	
(0.1,0.1)	0.18367	1.46E-29	0.18368	3.31E-06	0.17647	7.28E-29	0.17657	1.02E-04
(0.3,0.1)	0.34694	7.83E-30	0.34694	5.56E-06	0.23529	3.57E-29	0.23585	5.59E-04
(0.2,0.2)	0.36735	2.68E-29	0.36735	6.62E-06	0.35294	5.18E-29	0.35314	2.04E-04
(0.4,0.2)	0.53061	2.41E-29	0.53062	8.87E-06	0.41176	4.00E-29	0.41242	6.61E-04
(0.1,0.3)	0.38776	1.72E-29	0.38776	7.67E-06	0.47059	8.77E-30	0.47044	1.51E-04
(0.3,0.3)	0.55102	4.03E-29	0.55103	9.92E-06	0.52941	6.63E-29	0.52972	3.06E-04
(0.2,0.4)	0.57143	2.67E-29	0.57144	1.10E-05	0.64706	2.51E-29	0.64701	4.90E-05
(0.3,0.4)	0.65306	2.46E-29	0.65307	1.21E-05	0.67647	4.34E-29	0.67665	1.79E-04
(0.5,0.5)	0.91837	8.75E-31	0.91838	1.65E-05	0.88235	7.68E-30	0.88286	5.10E-04

Table 6: Comparison of numerical solutions with the solutions in [32] for v at $t = 0.1$ and $t = 0.4$ of experiment 2.

Points	$t = 0.1$				$t = 0.4$			
	MQQI	Error	ADM[32]	Error	MQQI	Error	ADM[32]	Error
	$N = 25$		$N = 441$		$N = 25$		$N = 441$	
(0.1,0.1)	-0.02041	4.48E-31	-0.02041	1.05E-06	-0.11765	8.34E-30	-0.11729	3.55E-04
(0.3,0.1)	0.18367	1.07E-29	0.18368	3.31E-06	0.17647	4.57E-30	0.17657	1.02E-04
(0.2,0.2)	-0.04082	1.64E-30	-0.04082	2.11E-06	-0.23529	3.22E-29	-0.23458	7.10E-04
(0.4,0.2)	0.16327	1.06E-29	0.16327	2.54E-06	0.05882	1.05E-29	0.05928	4.57E-04
(0.1,0.3)	-0.26531	1.43E-29	-0.26531	7.52E-06	-0.64706	3.53E-29	-0.64574	1.32E-03
(0.3,0.3)	-0.06122	5.65E-30	-0.06123	3.16E-06	-0.35294	4.79E-29	-0.35188	1.06E-03
(0.2,0.4)	-0.28571	1.97E-29	-0.28572	8.58E-06	-0.76471	4.47E-29	-0.76303	1.67E-03
(0.3,0.4)	-0.18367	9.65E-30	-0.18368	6.40E-06	-0.61765	3.92E-29	-0.61610	1.55E-03
(0.5,0.5)	-0.10204	8.00E-31	0.10205	5.27E-06	-0.58824	5.34E-30	-0.58646	1.77E-03

Table 7: Comparison of the actual and estimated absolute errors of u and v for $N = 25$ and various values of M at $t = 0.1$ of experiment 2.

Points	u			v		
	Actual	Estimated		Actual	Estimated	
		$M = 121$	$M = 169$		$M = 121$	$M = 169$
(0.1,0.1)	1.46202E-29	1.32664E-29	1.29426E-29	4.48113E-31	7.54813E-31	667521.E-31
(0.3,0.1)	7.83593E-30	4.98913E-30	5.38983E-30	1.07899E-29	7.80647E-30	9.01124E-30
(0.2,0.2)	2.68547E-29	2.37818E-29	2.58809E-29	1.64432E-30	1.69025E-30	1.18328E-30
(0.4,0.2)	2.41042E-29	1.83444E-29	2.42130E-29	1.06676E-29	9.04872E-30	8.03798E-30
(0.1,0.3)	1.72958E-29	1.26705E-29	1.56837E-29	1.43162E-29	9.51576E-30	1.34034E-29
(0.3,0.3)	4.03099E-29	3.78846E-29	3.77727E-29	5.65158E-30	5.31464E-30	5.31109E-30
(0.2,0.4)	2.67397E-29	2.41620E-29	2.25413E-29	1.97598E-29	1.63000E-29	1.67743E-29
(0.3,0.4)	2.46994E-29	1.49021E-29	2.02653E-29	9.65210E-30	8.91184E-30	7.44918E-30
(0.5,0.5)	8.75109E-31	6.58501E-59	4.00627E-61	8.00562E-31	8.12006E-61	2.01538E-61

Table 8: Comparison of numerical solutions with the solutions in [16] for u and v at $t = 0.625$ with $R = 500$ of experiment 3.

Points	u				v			
	MQQI		Jain and Holla [16]		MQQI		Jain and Holla [16]	
	$N = 121$	$N = 289$	$N = 121$	$N = 441$	$N = 121$	$N = 289$	$N = 121$	$N = 441$
(0.15,0.10)	0.72051	0.91232	0.75954	0.95691	-0.03638	0.06210	-0.12880	0.10177
(0.30,0.10)	1.04093	0.99454	1.03780	0.95616	0.10857	0.08993	-0.25386	0.13287
(0.10,0.20)	0.76793	0.81584	0.79536	0.84257	0.13783	0.16339	0.22765	0.18503
(0.20,0.20)	0.86793	0.85065	0.83338	0.86399	0.15585	0.15203	0.27094	0.18169
(0.10,0.30)	0.62630	0.66129	0.63127	0.67667	0.23121	0.25134	0.31462	0.26560
(0.30,0.30)	0.86238	0.79483	0.78637	0.76876	0.28376	0.23263	0.40238	0.25142
(0.15,0.40)	0.46529	0.53335	0.44135	0.54408	0.25621	0.29381	0.18416	0.32084
(0.20,0.40)	0.61387	0.57685	0.58494	0.58778	0.34908	0.28156	0.41766	0.30927

obtained from the exact solutions. The computational domain for this problem is $\Omega = \{(x,y)|0 \leqslant x \leqslant 1, 0 \leqslant y \leqslant 1\}$.The absolute errors of u and v using our scheme and the exact solutions for $R = 100$ are listed in Tables 1 and 2, respectively. The experiment is also solved by finite difference method (FDM) [1] and discrete Adomian

decomposition method (ADM) [32]. The numerical computations are performed using $N = 121$ points and $\Delta t = 0.001$, whereas the results of FDM and ADM were obtained with a uniform mesh $N = 441$ and $\Delta t = 0.0001$. Moreover, the graphs of the estimated solutions are plotted in Fig. 1.

Tables 1 and 2 indicate that the proposed method requires less nodes to attain the accuracy of the FDM [1] and ADM [32]. In addition, the actual absolute errors for the various of N and M, are compared with the estimated absolute errors of $u(x,y,t)$ and $v(x,y,t)$ at $t = 0.2$ in Tables 3 and 4.

From these comparisons, we see that the estimated absolute errors are almost the same as with the actual absolute errors. We observe from Tables 3 and 4 that the estimated errors are closer to the actual errors while value M increases.

Experiment 2. In this experiment, we take the two-dimensional nonlinear Burgers' equations with the initial conditions at $t = 0$ that are given by

$$f_1(x,y) = x+y, \quad f_2(x,y) = x-y.$$

The exact solutions are given by [3]

$$u(x,y,t) = \frac{x+y-2xt}{1-2t^2},$$

$$v(x,y,t) = \frac{x-y-2yt}{1-2t^2},$$

and the boundary functions $g_1(x,y,t)$ and $g_2(x,y,t)$ can be obtained from the exact solutions. In this experiment, we consider $\Delta t = 0.001$ and $\Omega = \{(x,y)|0 \leqslant x \leqslant 0.5, 0 \leqslant y \leqslant 0.5\}$.

The numerical computations were performed using $N = 25$ points that distributed uniformly and $\Delta t = 0.001$. The numerical solutions are compared with the solutions in [32] at internal points at $t = 0.1$ and $t = 0.4$ for arbitrary Reynolds number R in Tables 5 and 6. In [32], the numerical results were calculated by ADM with $N = 441$ and $\Delta t = 0.0001$.

The graphs of estimated functions u and v at $t = 0.4$ are given in Fig. 2. Therewith, the estimated absolute errors are compared with the actual absolute errors in Table 7 and it is seen that they are consistent.

Experiment 3. In the third experiment, the computational domain is taken as $\Omega = \{(x,y)|0 \leqslant x \leqslant 0.5, 0 \leqslant y \leqslant 0.5\}$ and Burgers' equations (13) and (14) are taken with the initial conditions:

$$u(x,y,0) = \sin(\pi x) + \cos(\pi y),$$

$$v(x,y,0) = x+y,$$

and the boundary conditions:

$$u(0,y,t) = \cos(\pi y), \ \ u(0.5,y,t) = 1+\cos(\pi y),$$

$$v(0,y,t) = y, \qquad v(0.5,y,t) = 0.5+y,$$

for $0 \leqslant y \leqslant 0.5, t \geqslant 0$ and

$$u(x,0,t) = 1+\sin(\pi x), \ \ u(x,0.5,t) = \sin(\pi x),$$

$$v(x,0,t) = x, \qquad v(x,0.5,t) = x+0.5,$$

for $0 \leqslant x \leqslant 0.5, \ t \geqslant 0$.

The numerical computations were performed using $N = 121$ and $N = 289$ points for $R = 500$ at $t = 0.625$ with $\Delta t = 0.001$ and $\Delta t = 0.01$, respectively.

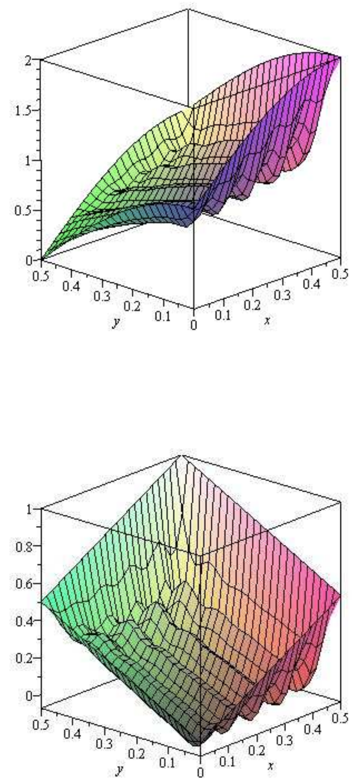

Fig. 3: The graphs of the estimated solutions of $u(x,y,t)$ (up) and $v(x,y,t)$ (down) at $t = 0.625$ with $R = 500$, $\Delta t = 0.001$ and $N = 121$ of experiment 3.

Since this experiment has not an exact solution, the numerical values of the approximate solutions are compared with the values of Jain and Holla [16] method in Table 8 at some points. Also, since the exact solution is unknown, the estimated absolute error functions are used for measurement of the reliability. The estimated absolute errors are tabulated for $N = 169$ and $M = 289$ in Table 9.

We plot the graph of estimated functions u and v at $t = 0.625$ in Fig. 3.

6 Conclusion

In this paper, we have presented a numerical scheme based on high accuracy MQ quasi-interpolation scheme for solving the two-dimensional nonlinear Burgers' equation. The numerical results which were given in the previous section demonstrate the efficiency and accuracy of the presented scheme. Tables show that the proposed scheme requires less points to attain accuracy than FDM and ADM. Also, we could getting results similar to the

Table 9: Numerical solutions and estimated absolute errors of u and v for $N = 169$, $M = 289$ and $\Delta t = 0.005$ at $t = 0.625$ of experiment 3.

Points	u		v	
	MQQI	Estimated Error	MQQI	Estimated Error
(0.3,0.1)	0.8486	1.63E-1	0.0101	8.92E-2
(0.1,0.2)	0.7620	6.40E-2	0.1334	3.57E-2
(0.2,0.2)	0.7416	1.33E-1	0.0870	8.01E-2
(0.1,0.3)	0.6207	4.87E-2	0.2238	3.21E-2
(0.3,0.3)	0.7111	8.85E-2	0.1658	6.95E-2
(0.2,0.4)	0.5468	3.45E-2	0.2739	2.30E-3

results of FDM and ADM by using bigger step time. Moreover, we have estimated the errors by Eqs. (30)-(33) and seen that the estimated absolute errors are almost the same as with the actual absolute errors. By using the error estimation given in Section 4, we can estimate the absolute error for the cases that the exact solution is unknown.

Although, we used equidistant data in our numerical experiments but our scheme can be used for the scattered data.

References

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

[2] M. Basto, V. Semiao and F. Calheiros, Dynamics and synchronization of numerical solutions of the Burgers' equation, Comput. Appl. Math. **231**, 793-806 (2009).

[3] J. Biazar and H. Aminikhah, Exact and numerical solutions for non-linear Burger's equation by VIM, Math. Comput. Modelling **49**, 1394-1400 (2009).

[4] R.K. Beatson and M.J.D. Powell, Univariate multiquadric approximation: quasi-interpolation to scattered data, Constr. Approx. **8**, 275-288 (1992).

[5] J.M. Burger, A mathematical model illustrating the theory of turbulence, Adv. Appl. Mech. **1**, 171-199 (1948).

[6] R.H. Chen and Z.M. Wu, Solving partial differential equation by using multiquadric quasi-interpolation, Appl. Math. Comput. **186**, 1502-1510 (2007).

[7] R.H. Chen and Z.M. Wu, Applying multiquadric quasi-interpolation to solve Burgers' equation, Appl. Math. Comput. **172**, 472-484 (2006).

[8] R.H. Chen and Z.M. Wu, Solving hyperbolic conservation laws using multiquadric quasi-interpolation, Numer. Methods Partial Differential Equations, 776-796 **22** (2006).

[9] Ï. Dağ, A. Canivar and A. Şahin, Taylor-Galerkin and Taylor-collocation methods for the numerical solutions of Burgers' equation using B-splines, Commun. Nonlinear Sci. Numer. Simul. **16**, 2696-2708 (2011).

[10] C.A.J. Fletcher, A comparison of finite element and finite difference solution of the one- and two-dimensional Burgers' equations, Int. J. Comput. Phys. **51**, 159-188 (1983).

[11] C.A.J. Fletcher, Generating exact solutions of the two-dimensional Burgers' equation, Int. J. Numer. Methods Fluids **3**, 213-216 (1983).

[12] C. Franke and R. Schaback, Convergence order estimates of meshless collocation methods using radial basis functions, Adv. Comput. Math. **8**, 381-399 (1998).

[13] M.A. Golberg, C.S. Chen and S.R. Karur, Improved multiquadric approximation for partial differential equations, Eng. Anal. Bound. Elem. **18**, 9-17 (1996).

[14] Y.C. Hon and Z.M. Wu, A quasi-interpolation method for solving stiff ordinary differential equations, Int. J. Numer. Methods Eng. **48**, 1187-1197 (2000).

[15] Y.C. Hon, A Quasi-Radial Basis Functions Method for American Options Pricing, Comput. Math. Applic. **43**, 513-524 (2002).

[16] P.C. Jain and D.N. Hola, Numerical solution of coupled Burgers' equations, Int. J. Numer. Meth. Eng. **12**, 213-222 (1978).

[17] Z.W. Jiang, R.H. Wang, C.G. Zhu and M. Xu, High accuracy multiquadric quasi-interpolation, Appl. Math. Modelling **35**, 2185-2195 (2011).

[18] Z.W. Jiang and R.H. Wang, Numerical solution of one-dimensional SineGordon equation using high accuracy multiquadric quasi-interpolation, Appl. Math. Comput. **218**, 7711-7716 (2012).

[19] E.J. Kansa, Multiquadric-a scattered data approximation scheme with applications to computational fluid dynamics I, Comput. Math. Appl. **19**, 127-145 (1990).

[20] P.D. Lax and B. Wendroff, Difference schemes for hyperbolic equations with high order of accuracy, Commun. Pure Appl. Math. **17**, 381-98 (1964).

[21] L. Ling, Multivariate quasi-interpolation schemes for dimension-splitting multiquadric, Appl. Math. Comput. **161**, 195-209 (2005).

[22] W.R. Madych and S.A. Nelson, Multivariate interpolation and conditionally positive definite functions, Math. Comp. **54**, 211-230 (1990).

[23] C.A. Micchelli, Interpolation of scattered data: distance matrix and conditionally positive definite functions, Constr. Approx. **2**, 11-22 (1986).

[24] J. Nee and J. Duan, Limit set of trajectories of the coupled viscous Burgers equations, Appl. Math. Lett. **11**, 57-61 (1998).

[25] S. Shahmorad, Numerical solution of the general form linear FredholmVolterra integro-differential equations by the Tau method with an error estimation, Appl. Math. Comput. **167**, 1418-1429 (2005).

[26] Z.M. Wu, Dynamically knot and shape parameter setting for simulating shock wave by using multiquadric quasi-interpolation, Eng. Anal. Bound. Elem. **29**, 354-358 (2005).

[27] Z.M. Wu and R. Schaback, Shape preserving properties and convergence of univariate multiquadric quasi- interpolation, Acta. Math. Appl. Sinica (English Ser.) **10**, 441-446 (1994).

[28] F.W. Wubs and E.D. de Goede, An explicit-implicit method for a class of time-dependent partial differential equations, Appl. Numer. Math. **9**, 157-181 (1992).

[29] D.L. Young, C.M. Fan, S.P. Hu and S.N. Atluri, The Eulerian-Lagrangian method of fundamental solutions for two-dimensional unsteady Burgers' equations, Eng. Anal. Bound. Elem. **32**, 395-412 (2008).

[30] W. Zhang, C. Zhang and G. Xi, An explicit Chebyshev pseudospectral multigrid method for incompressible NavierStokes equations, Comput. Fluids **39**, 178-188 (2010).
[31] X.H. Zhang, J. Ouyang and L.Zhang, Element-free characteristic Galerkin method for Burgers equation, Eng. Anal. Bound. Elem. **33**, 356-362 (2009).
[32] H. Zhu, H. Shu and M. Ding, Numerical solution of two-dimensional Burgers' equation by discrete Adomian decomposition method, Comput. Math. Appl. **60**, 840-848 (2010).

Maryam Sarboland is a PhD student in the Faculty of Mathematical Science, K. N. Toosi University of Technology, Tehran, Iran. Her research interests are in the areas of applied mathematics including the mathematical methods and numerical analysis of evolution partial differential equations.

Azim Aminataei is an associate Professor in the Faculty of Mathematical Science, K. N. Toosi University of Technology, Tehran, Iran. He has many publications is well-known journals. His fields of interests are scientific computing and numerical analysis of ordinary and partial differential equations, and integral equations in applied mathematics.