Research Article

On the Numerical Solution of One-Dimensional Nonlinear Nonhomogeneous Burgers' Equation

Maryam Sarboland and Azim Aminataei

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

Correspondence should be addressed to Azim Aminataei; ataei@kntu.ac.ir

The nonlinear Burgers' equation is a simple form of Navier-Stocks equation. The nonlinear nature of Burgers' equation has been exploited as a useful prototype differential equation for modeling many phenomena. This paper proposes two meshfree methods for solving the one-dimensional nonlinear nonhomogeneous Burgers' equation. These methods are based on the multiquadric (MQ) quasi-interpolation operator $\mathscr{L}_{\mathscr{W}_2}$ and direct and indirect radial basis function networks (RBFNs) schemes. In the present schemes, the Taylors series expansion is used to discretize the temporal derivative and the quasi-interpolation is used to approximate the solution function and its spatial derivatives. In order to show the efficiency of the present methods, several experiments are considered. Our numerical solutions are compared with the analytical solutions as well as the results of other numerical schemes. Furthermore, the stability analysis of the methods is surveyed. It can be easily seen that the proposed methods are efficient, robust, and reliable for solving Burgers' equation.

1. Introduction

In this paper, we consider the one-dimensional nonlinear nonhomogeneous Burgers' equation:

$$u_t + uu_x = vu_{xx} + F(x,t), \quad x \in \Omega = [a,b], \; t \geqslant t_0, \quad (1)$$

with the initial condition,

$$u(x,t_0) = g(x), \quad (2)$$

and the boundary conditions,

$$u(a,t) = g_1(t), \qquad u(b,t) = g_2(t), \quad (3)$$

where $g(x)$, $g_1(t)$, and $g_2(t)$ are known functions, v is the positive parameter that related to the Reynolds number $R = 1/v$, and $F(x,t)$ is a known nonhomogeneous term.

This equation was first derived from the hydrodynamics equations and used in surveying the laser generation of sound [1]. Later on, it was applied to other physical phenomena such as wind forcing the buildup of water waves, electrohydrodynamic field in plasma physics, and design of feedback control [2–4].

When $F(x,t) = 0$, (1) is the well-known Burgers' equation:

$$u_t + uu_x = vu_{xx}. \quad (4)$$

Burgers' equation in homogeneous form was first introduced by Bateman [5] who found its steady solutions, descriptive of certain viscous flows. It was later presented by Burgers as one of class of equations describing mathematical models of turbulence [6]. In the context of gas dynamics, it was surveyed by Hopf [7] and Cole [8]. The homogeneous Burgers' equation appears in various areas of applied mathematics and physics such as the phenomena of turbulence and supersonic flow, heat conduction, elasticity, and fusion [6–10].

From an analytical point of view, the nonhomogeneous form is poorly studied, the complete analytical solution being closely dependent on the form of the nonhomogeneous term. For example, Karabutov et al. [1] obtained the analytical solution of the nonhomogeneous Burgers' equation with $F(x,t) = A\sin(x)$, $A > 0$, Ding et al. [11] studied the solution of (1) for the time-independent nonhomogeneous term $F(x,t) = -kx$, and Rao and Yadav [12] represented

the solutions of the nonhomogeneous Burgers' equation for the nonhomogeneous term $F(x,t) = kx/(2\beta t + 1)^2$ that $k > 0$ and $\beta > 0$ are constant. Recently, Moreau and vallée have obtained the analytical solution of the nonhomogeneous Burgers' equation with an elastic forcing term $F(x,t) = -kx + f(t)$, $k \in \mathbb{R}$ [3].

Up to now, various numerical methods are presented for the homogeneous Burgers' equation such as finite difference, finite element, boundary element, and collocation methods. For a survey of these methods refer to [13–16] and references cited therein. Among the methods that are mentioned above, the spatial domain where the partial differential governing equations are defined is often discretized into meshes. In these methods, the creation of suitable meshes is very essential for getting accurate results. However, the procedure of mesh generation consumes a lot of time and labor for some problems, especially for discontinuous and high gradient problems, for which these methods will become complicate. The root of these difficulties is the use of mesh in the formulation step. To avoid the mesh generation, recently, a kind of so-called meshfree or meshless method has extended quickly. In these methods, the scattered nodes are only used instead of meshing the domain of the problem.

For the last 20 years, the radial basis functions (RBFs) method was known as a powerful tool for the scattered data interpolation problem. The use of RBFs as a meshless process for the numerical solution of partial differential equations (PDEs) is based on the collocation scheme. The meshless methods based on RBFs were studied for approximating the solution of PDEs since initial development of Kansa's work (1990) [17]. Kansa's method was extended to solve various ordinary and partial differential equations [18–20]. In these works, the solution function is decomposed into RBFs and its derivatives are then arrived through differentiation that caused the reduction in convergence rate. In order to avoid this problem, Mai-Duy and Tran-Cong proposed an integrated MQ-RBFNs scheme for the approximation of function and its derivatives [21]. Numerical experiments and theoretical analysis demonstrate that for solving PDEs integrated RBF (IRBF) procedure is more accurate in comparison with direct RBF (DRBF) procedure. Also, IRBF scheme is more stable than DRBF for a range of PDEs [21, 22].

In both DRBF and IRBF schemes, one must resolve a linear system of equations at each time step. In the past decade, the other meshless method was introduced by using a MQ quasi-interpolation without solving a linear system of equations. MQ quasi-interpolation is constructed directly from linear combination of MQ-RBF and the approximated function. In 1992, Beaston and Powell [23] presented three univariate MQ quasi-interpolations named as $\mathcal{L}_{\mathcal{A}}$, $\mathcal{L}_{\mathcal{B}}$, and $\mathcal{L}_{\mathcal{C}}$. Wu and Schaback [24] proposed the MQ quasi-interpolation $\mathcal{L}_{\mathcal{D}}$ and indicated that the scheme is shape preserving and convergent. Recently, Jiang et al. [25] have introduced a new multilevel univariate MQ quasi-interpolation approach with high approximation order compared with initial MQ quasi-interpolation scheme named as $\mathcal{L}_{\mathcal{W}}$ and $\mathcal{L}_{\mathcal{W}_2}$. This approach is based on inverse multiquadric (IMQ) RBF interpolation and Wu and Schaback's MQ quasi-interpolation

operator $\mathcal{L}_{\mathcal{D}}$. Chen and Wu applied initial MQ quasi-interpolation scheme for solving one-dimensional nonlinear homogeneous Burgers' equation [26].

In numerical solution of time dependent PDEs, such as Burgers' and Sine-Gordon equations, by using MQ quasi-interpolation scheme, there is a limitation for discretization of the temporal derivative. One has to use low order finite difference approximation for discretization of time derivatives because one does not solve any system of equations at each time step; otherwise one must solve a system of equations [27]. Also, large number of nodes must be used for getting appropriate accuracy; see [26, 28, 29].

In this paper, we present two numerical methods by using MQ quasi-interpolation for the numerical solution of the nonhomogeneous Burgers' equation. In both of them, we use a two-order approximation for discretization of the time derivative. 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 [30] and used by Dağ et al. for the homogeneous Burgers' equation [31]. By using this discretization, we have to solve a linear system of equations at each time step that the size of the system is equivalent to the number of the centers in the spatial domain. Also, because the IRBFN scheme requires fewer centers in comparison with the DRBFN scheme, we apply MQ quasi-interpolation scheme in the indirect form in order not to encounter with large scale matrix.

The Jiang et al. MQ quasi-interpolation operator $\mathcal{L}_{\mathcal{W}_2}$ is summation of two series that the second series coefficients are combined with first series coefficients. By giving relation between two series coefficients based on function values, we can convert it to a compact form based on one series and use it in direct and indirect forms for the numerical solution of PDEs.

The rest of present paper is organized as follows. A brief explanation of the MQ quasi-interpolation scheme is given in Section 2. Our numerical methods are applied on the nonlinear Burgers' equation in Section 3. In Section 4, the stability analysis of the methods is discussed. The results of several numerical experiments are reported in Section 5. Finally, some conclusions based on obtained results are drawn in Section 6.

2. The MQ Quasi-Interpolation Scheme

In this section, three univariate MQ quasi-interpolation schemes named as $\mathcal{L}_{\mathcal{D}}$, $\mathcal{L}_{\mathcal{W}}$, and $\mathcal{L}_{\mathcal{W}_2}$ are described. Then, we describe our approach which converts operator $\mathcal{L}_{\mathcal{W}_2}$ to the compact form. More details can be seen in [23–25].

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

$$a = x_0 < x_1 < \cdots < x_N = b, \qquad h = \max_{1 \leq i \leq N} (x_i - x_{i-1}), \quad (5)$$

if we are supplied with a function $f : [a, b] \rightarrow \mathbb{R}$, quasi-interpolation of f takes the form:

$$\mathcal{L}(f) = \sum_{i=0}^{N} f(x_i) \phi_i(x), \qquad (6)$$

where each function $\phi_i(x)$ is a linear combination of the Hardy MQs basis function [32],

$$\psi_i(x) = \sqrt{c^2 + (x - x_i)^2}, \tag{7}$$

and low order polynomials and $c \in \mathbb{R}^+$ is a shape parameter. This formula requires the derivative values of f at the end points that is not convenient for practical purposes [23]. Wu and Schaback [24] presented the univariate MQ quasi-interpolation operator $\mathscr{L}_{\mathscr{D}}$ that is defined as

$$\mathscr{L}_{\mathscr{D}} f(x) = \sum_{i=0}^{N} f(x_i) \widetilde{\psi}_i(x), \tag{8}$$

where

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

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

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

$$2 \leqslant i \leqslant N - 2,$$

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

$$\widetilde{\psi}_N(x) = \frac{1}{2} + \frac{\psi_{N-1}(x) - (x_N - x)}{2(x_N - x_{N-1})}. \tag{9}$$

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

$$S_{f''} = \sum_{j=1}^{\overline{N}} \alpha_j \overline{\varphi}\left(\left|x - x_{k_j}\right|\right), \tag{10}$$

where

$$\overline{\varphi}(r) = \frac{s^2}{(s^2 + r^2)^{3/2}}, \tag{11}$$

and $s \in \mathbb{R}^+$ is a shape parameter.

The coefficients $\{\alpha_j\}_{j=1}^{\overline{N}}$ are uniquely determined by the interpolation condition

$$S_{f''}\left(x_{k_i}\right) = \sum_{j=1}^{\overline{N}} \alpha_j \overline{\varphi}\left(\left|x_{k_i} - x_{k_j}\right|\right) = f''\left(x_{k_i}\right), \quad 1 \leqslant i \leqslant \overline{N}. \tag{12}$$

Since (12) is solvable [33], so

$$\alpha = A_X^{-1} \cdot f_X'', \tag{13}$$

where

$$X = \left\{x_{k_1}, \ldots, x_{k_{\overline{N}}}\right\}, \qquad \alpha = \left[\alpha_1, \ldots, \alpha_{\overline{N}}\right]^T,$$

$$A_X = \left[\overline{\varphi}\left(\left|x_{k_i} - x_{k_j}\right|\right)\right], \tag{14}$$

$$f_X'' = \left[f''\left(x_{k_1}\right), \ldots, f''\left(x_{k_{\overline{N}}}\right)\right]^T.$$

By using the f and the coefficient α defined in (13), a function $e(x)$ is constructed in the form

$$e(x) = f(x) - \sum_{j=1}^{\overline{N}} \alpha_j \sqrt{s^2 + \left(x - x_{k_j}\right)^2}. \tag{15}$$

Now, the MQ quasi-interpolation operator $\mathscr{L}_{\mathscr{W}}$ by using $\mathscr{L}_{\mathscr{D}}$ defined by (8) and (9) on the data $\{(x_i, e(x_i))\}_{i=1}^{N}$ with the shape parameter c is given by

$$\mathscr{L}_{\mathscr{W}} f(x) = \sum_{j=1}^{\overline{N}} \alpha_j \sqrt{s^2 + \left(x - x_{k_j}\right)^2} + \mathscr{L}_{\mathscr{D}} e(x). \tag{16}$$

The shape parameters c and s should not be the same constant in (16).

In (12), the value of $f_{x_{k_j}}''$ can be replaced by

$$f_{x_{k_j}}'' = \left(2\left[\left(x_{k_j} - x_{k_{j-1}}\right) f\left(x_{k_{j+1}}\right)\right.\right.$$
$$\left. - \left(x_{k_{j+1}} - x_{k_{j-1}}\right) f\left(x_{k_j}\right)\right.$$
$$\left.\left. + \left(x_{k_{j+1}} - x_{k_j}\right) f\left(x_{k_{j-1}}\right)\right]\right)$$
$$\times \left(\left(x_{k_j} - x_{k_{j-1}}\right)\left(x_{k_{j+1}} - x_{k_j}\right)\left(x_{k_{j+1}} - x_{k_{j-1}}\right)\right)^{-1}, \tag{17}$$

when the data's $\{(x_{k_i}, f(x_{k_i}))\}_{i=1}^{\overline{N}}$ are given. So, if f_X'' in (13) is replaced by

$$F_X'' = \left[f_{x_{k_1}}'', \ldots, f_{x_{k_{\overline{N}}}}''\right]^T, \tag{18}$$

then the quasi-interpolation operator defined by (15) and (16) is denoted by $\mathscr{L}_{\mathscr{W}_2}$. The linear reproducing property and the high convergence rate of $\mathscr{L}_{\mathscr{W}_2}$ were also studied in [25].

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

$$\mathscr{L}_{\mathscr{W}_2} f(x) = \sum_{i=0}^{N} f(x_i) \widehat{\psi}_i(x), \tag{19}$$

where the basis functions $\widehat{\psi}_i(x)$ are obtained by substituting (13), (15), and (18) into (16). As such, let

$X = \{x_0, x_1, x_2, x_3, x_4\}$ and $X' = \{x_2\}$. So $N = 4$, $\overline{N} = 1$, $k_1 = 2$, and

$$\alpha = \left(2 \left[\left(x_{k_1} - x_{k_0} \right) f \left(x_{k_2} \right) \right. \right.$$
$$\left. - \left(x_{k_2} - x_{k_0} \right) f \left(x_{k_1} \right) + \left(x_{k_2} - x_{k_1} \right) f \left(x_{k_0} \right) \right] \right)$$
$$\times \left(s \left(x_{k_1} - x_{k_0} \right) \left(x_{k_2} - x_{k_1} \right) \left(x_{k_2} - x_{k_0} \right) \right)^{-1},$$

$$(20)$$

$$e(x) = f(x) - \alpha \sqrt{s^2 + \left(x - x_{k_1} \right)^2}. \tag{21}$$

Substituting (20) into (21) yields

$$e(x) = f(x) - \left(\left(2 \left[\left(x_{k_1} - x_{k_0} \right) f \left(x_{k_2} \right) \right. \right. \right.$$
$$\left. - \left(x_{k_2} - x_{k_0} \right) f \left(x_{k_1} \right) \right.$$
$$\left. + \left(x_{k_2} - x_{k_1} \right) f \left(x_{k_0} \right) \right] \right)$$
$$\times \left(s \left(x_{k_1} - x_{k_0} \right) \left(x_{k_2} - x_{k_1} \right) \left(x_{k_2} - x_{k_0} \right) \right)^{-1} \right)$$
$$\times \sqrt{s^2 + \left(x - x_{k_1} \right)^2}.$$

$$(22)$$

Hence, the substitution of (20) and (22) into (16) leads to

$$\mathcal{L}_{\mathcal{W}_2} f(x) = \left(\left(2 \left[\left(x_{k_1} - x_{k_0} \right) f \left(x_{k_2} \right) \right. \right. \right.$$
$$\left. - \left(x_{k_2} - x_{k_0} \right) f \left(x_{k_1} \right) \right.$$
$$\left. + \left(x_{k_2} - x_{k_1} \right) f \left(x_{k_0} \right) \right] \right)$$
$$\times \left(s \left(x_{k_1} - x_{k_0} \right) \left(x_{k_2} - x_{k_1} \right) \left(x_{k_2} - x_{k_0} \right) \right)^{-1} \right)$$
$$\times \sqrt{s^2 + \left(x - x_{k_1} \right)^2}$$
$$+ \sum_{i=0}^{4} \left(f(x_i) - \left(\left(2 \left[\left(x_{k_1} - x_{k_0} \right) f \left(x_{k_2} \right) \right. \right. \right. \right.$$
$$\left. - \left(x_{k_2} - x_{k_0} \right) f \left(x_{k_1} \right) \right.$$
$$\left. + \left(x_{k_2} - x_{k_1} \right) f \left(x_{k_0} \right) \right] \right)$$
$$\times \left(s \left(x_{k_1} - x_{k_0} \right) \left(x_{k_2} - x_{k_1} \right) \right.$$
$$\left. \times \left(x_{k_2} - x_{k_0} \right) \right)^{-1} \right)$$
$$\times \sqrt{s^2 + \left(x_i - x_{k_1} \right)^2} \right) \tilde{\psi}_i(x),$$

$$(23)$$

whereas $f(x_{k_0}) = f(x_0)$, $f(x_{k_1}) = f(x_2)$, and $f(x_{k_2}) = f(x_4)$. Therewith, (23) can be rewritten as

$$\mathcal{L}_{\mathcal{W}_2} f(x) = \left(\left(2 \left[\left(x_2 - x_0 \right) f(x_4) \right. \right. \right.$$
$$\left. - \left(x_4 - x_0 \right) f(x_2) + \left(x_4 - x_2 \right) f(x_0) \right] \right)$$
$$\times \left(s \left(x_2 - x_0 \right) \left(x_4 - x_2 \right) \left(x_4 - x_0 \right) \right)^{-1} \right)$$
$$\times \sqrt{s^2 + \left(x - x_2 \right)^2} + \sum_{i=0}^{4} f(x_i) \tilde{\psi}_i(x)$$
$$- \sum_{i=0}^{4} \left(\left(2 \left[\left(x_2 - x_0 \right) f(x_4) - \left(x_4 - x_0 \right) f(x_2) \right. \right. \right.$$
$$\left. + \left(x_4 - x_2 \right) f(x_0) \right] \right)$$
$$\times \left(s \left(x_2 - x_0 \right) \left(x_4 - x_2 \right) \left(x_4 - x_0 \right) \right)^{-1} \right)$$
$$\times \sqrt{s^2 + \left(x_i - x_2 \right)^2} \tilde{\psi}_i(x).$$

$$(24)$$

Hence, the basis function $\hat{\psi}_i(x)$ are arrived as follows:

$$\hat{\psi}_0(x) = \frac{2}{s \left(x_2 - x_0 \right) \left(x_4 - x_0 \right)}$$
$$\times \left[\sqrt{s^2 + \left(x - x_2 \right)^2} \right.$$
$$\left. - \sum_{i=0}^{4} \sqrt{s^2 + \left(x_i - x_2 \right)^2} \tilde{\psi}_i(x) \right] + \tilde{\psi}_0(x),$$

$$\hat{\psi}_2(x) = \frac{-2}{s \left(x_2 - x_0 \right) \left(x_4 - x_2 \right)}$$
$$\times \left[\sqrt{s^2 + \left(x - x_2 \right)^2} \right.$$
$$\left. - \sum_{i=0}^{4} \sqrt{s^2 + \left(x_i - x_2 \right)^2} \tilde{\psi}_i(x) \right] + \tilde{\psi}_2(x),$$

$$\hat{\psi}_4(x) = \frac{2}{s \left(x_4 - x_2 \right) \left(x_4 - x_0 \right)}$$
$$\times \left[\sqrt{s^2 + \left(x - x_2 \right)^2} \right.$$
$$\left. - \sum_{i=0}^{4} \sqrt{s^2 + \left(x_i - x_2 \right)^2} \tilde{\psi}_i(x) \right] + \tilde{\psi}_4(x),$$

$$\hat{\psi}_i(x) = \tilde{\psi}_i(x), \quad i = 1, 3.$$

$$(25)$$

By writing operator $\mathcal{L}_{\mathcal{W}_2}$ in the compact form (23), we can use it in two indirect and direct forms for the numerical solution of PDEs.

3. The Numerical Methods

In this section, the numerical schemes are presented for solving the nonlinear Burgers' equation (1) by using the MQ quasi-interpolation $\mathscr{L}_{\mathscr{W}_2}$. In our approach, the MQ quasi-interpolation approximates the solution function and the spatial derivatives of the differential equation and Taylor's series expansion is employed to approximate the temporal derivative similar to the work that Dağ et al. did in [31]. The MQ quasi-interpolation method is applied in direct and indirect forms.

We discretize the problem using the following Taylor's series expansion with step size Δt:

$$u_t^n = \frac{u^{n+1} - u^n}{\Delta t} - \frac{\Delta t}{2} u_{tt}^n + O\left(\Delta t^2\right), \tag{26}$$

where $u_t^n = u_t(x, t_n)$ and $t_n = t_0 + n\Delta t$.

Differentiating (1) with respect to time, u_{tt}^n can be written as follows:

$$\begin{aligned} u_{tt}^n &= \left(-u^n u_x^n + v u_{xx}^n + F^n\right)_t \\ &= -u^n (u_t^n)_x - u_x^n u_t^n + v(u_t^n)_{xx} + F_t^n, \end{aligned} \tag{27}$$

where $F_t^n = F_t(x, t_n)$. Using forward difference formula for the time derivative u_t^n in (27), u_{tt}^n can be rewritten as

$$\begin{aligned} \Delta t u_{tt}^n &= -u^n \left(u_x^{n+1} - u_x^n\right) - u_x^n \left(u^{n+1} - u^n\right) \\ &\quad + v \left(u_{xx}^{n+1} - u_{xx}^n\right) + \Delta t F_t^n. \end{aligned} \tag{28}$$

Substituting (28) into (26) and using the expression achieved in (1), the following time discretized form of nonlinear Burgers' equation is yielded:

$$\begin{aligned} u^{n+1} &+ \frac{\Delta t}{2} u^n u_x^{n+1} + \frac{\Delta t}{2} u_x^n u^{n+1} - v\frac{\Delta t}{2} u_{xx}^{n+1} \\ &= u^n + v\frac{\Delta t}{2} u_{xx}^n + \Delta t \left(F^n + \frac{\Delta t}{2} F_t^n\right). \end{aligned} \tag{29}$$

3.1. The Direct MQ Quasi-Interpolation Scheme. In this scheme, the unknown function u^n is approximated by using MQ quasi-interpolation scheme, and its spatial derivatives u_x^n and u_{xx}^n are calculated by differentiating such closed form of quasi approximation as follows:

$$u^n(x) = \sum_{i=0}^{N} u_i^n \widehat{\psi}_i(x), \tag{30}$$

$$u_x^n(x) = \sum_{i=0}^{N} u_i^n \frac{\partial \widehat{\psi}_i}{\partial x}(x) = \sum_{i=0}^{N} u_i^n \check{\psi}_i(x), \tag{31}$$

$$u_{xx}^n(x) = \sum_{i=0}^{N} u_i^n \frac{\partial^2 \widehat{\psi}_i}{\partial x^2}(x) = \sum_{i=0}^{N} u_i^n \overline{\psi}_i(x), \tag{32}$$

where $\partial \widehat{\psi}_i / \partial x = \check{\psi}_i$ and $\partial^2 \widehat{\psi}_i / \partial x^2 = \overline{\psi}_i$.

Now, replacing (30)–(32) into (29) and applying collocation method yield

$$\begin{aligned} \sum_{k=0}^{N} u_k^{n+1} &\left[\left(1 + \frac{\Delta t}{2} \sum_{j=0}^{N} u_j^n \check{\psi}_{ij}\right) \widehat{\psi}_{ik}\right. \\ &\left. + \frac{\Delta t}{2}\left(\sum_{j=0}^{N} u_j^n \widehat{\psi}_{ij}\right) \check{\psi}_{ik} - v\frac{\Delta t}{2} \overline{\psi}_{ik}\right] \\ = \sum_{j=0}^{N} u_j^n &\left(\widehat{\psi}_{ij} + v\frac{\Delta t}{2}\overline{\psi}_{ij}\right) \\ &+ \Delta t\left(F_i^n + \frac{\Delta t}{2} F_t^n(x_i)\right), \quad 1 \leqslant i \leqslant N-1, \end{aligned} \tag{33}$$

where $F_i^n = F(x_i, t_n)$, $F_t^n(x_i) = F_t(x_i, t_n)$, $\widehat{\psi}_{ij} = \widehat{\psi}_j(x_i)$, $\check{\psi}_{ij} = \check{\psi}_j(x_i)$, and $\overline{\psi}_{ij} = \overline{\psi}_j(x_i)$, whereas, according to (3), we have

$$u_0^n = u\left(x_0, t_n\right) = u\left(a, t_n\right) = g_1\left(t_n\right), \tag{34}$$

$$u_N^n = u\left(x_N, t_n\right) = u\left(b, t_n\right) = g_2\left(t_n\right). \tag{35}$$

Substituting (34) and (35) into (33), wherein (33) generates a system of $N-1$ linear equations in $N-1$ unknown parameters u_i^{n+1}.

Equation (33) can be written in the matrix form

$$\begin{aligned} &\left[\mathbf{A}_1 + \frac{\Delta t}{2}\mathbf{u}_x^n * \mathbf{A}_1 + \frac{\Delta t}{2}\mathbf{u}^n * \mathbf{D}_1 - v\frac{\Delta t}{2}\ddot{\mathbf{D}}_1\right]\mathbf{u}^{n+1} \\ &= \left[\mathbf{A}_1 + v\frac{\Delta t}{2}\ddot{\mathbf{D}}_1 - \frac{\Delta t}{2}\sum_{j=1}^{2} g_j^{n+1}\left(\widehat{\Psi}_j * \mathbf{D}_1 + \check{\Psi}_j * \mathbf{A}_1\right)\right]\mathbf{u}^n \\ &\quad - \sum_{k=0}^{1}\sum_{j=1}^{2} g_j^{n+k}\left[\widehat{\Psi}_j - v\frac{\Delta t}{2}\overline{\Psi}_j\right] \\ &\quad - \Delta t\sum_{j=1}^{2} g_j^{n+1} g_j^n \widehat{\Psi}_j * \check{\Psi}_j \\ &\quad - \frac{\Delta t}{2}\left[g_1^{n+1} g_2^n + g_2^{n+1} g_1^n\right]\left[\check{\Psi}_2 * \widehat{\Psi}_1 + \check{\Psi}_1 * \widehat{\Psi}_2\right] \\ &\quad + \Delta t\mathbf{F}_1^n + \frac{\Delta t^2}{2}\acute{\mathbf{F}}_1^n, \end{aligned} \tag{36}$$

where symbol $*$ stands for component by component multiplication,

$$\mathbf{A}_{1_{ij}} = \left[\widehat{\psi}_{ij}\right]_{i,j=1}^{N-1}, \qquad \mathbf{D}_{1_{ij}} = \left[\check{\psi}_{ij}\right]_{i,j=1}^{N-1}, \qquad \ddot{\mathbf{D}}_{1_{ij}} = \left[\overline{\psi}_{ij}\right]_{i,j=1}^{N-1},$$

$$\widehat{\Psi}_1 = \left[\widehat{\psi}_0(x_1), \widehat{\psi}_0(x_2), \ldots, \widehat{\psi}_0(x_{N-1})\right]^T,$$

$$\widehat{\Psi}_2 = \left[\widehat{\psi}_N(x_1), \widehat{\psi}_N(x_2), \ldots, \widehat{\psi}_N(x_{N-1})\right]^T,$$

$$\check{\Psi}_1 = \left[\check{\psi}_0(x_1), \check{\psi}_0(x_2), \ldots, \check{\psi}_0(x_{N-1})\right]^T,$$

$$\check{\Psi}_1 = \left[\check{\psi}_N(x_1), \check{\psi}_N(x_2), \ldots, \check{\psi}_N(x_{N-1})\right]^T,$$

$$\overline{\Psi}_1 = \left[\overline{\psi}_0(x_1), \overline{\psi}_0(x_2), \ldots, \overline{\psi}_0(x_{N-1})\right]^T,$$

$$\overline{\Psi}_2 = \left[\overline{\psi}_N(x_1), \overline{\psi}_N(x_2), \ldots, \overline{\psi}_N(x_{N-1})\right]^T,$$

$$\mathbf{F}_1^n = \left[F^n(x_1), F^n(x_2), \ldots, F^n(x_{N-1})\right]^T,$$

$$\acute{\mathbf{F}}_1^n = \left[F_t^n(x_1), F_t^n(x_2), \ldots, F_t^n(x_{N-1})\right]^T. \tag{37}$$

Subsequently, (36) can be written as

$$\mathbf{u}^{n+1} = \mathbf{M}_1^{-1}\mathbf{N}_1\mathbf{u}^n + \mathbf{M}_1^{-1}\Psi, \tag{38}$$

where

$$\mathbf{M}_1 = \mathbf{A}_1 + \frac{\Delta t}{2}\mathbf{u}_x^n * \mathbf{A}_1 + \frac{\Delta t}{2}\mathbf{u}^n * \mathbf{D}_1 - v\frac{\Delta t}{2}\ddot{\mathbf{D}}_1,$$

$$\mathbf{N}_1 = \mathbf{A}_1 + v\frac{\Delta t}{2}\ddot{\mathbf{D}}_1 - \frac{\Delta t}{2}\sum_{j=1}^{2}g_j^{n+1}\left(\widehat{\Psi}_j * \mathbf{D}_1 + \check{\Psi}_j * \mathbf{A}_1\right),$$

$$\Psi = -\sum_{k=0}^{1}\sum_{j=1}^{2}g_j^{n+k}\left[\widehat{\Psi}_j - v\frac{\Delta t}{2}\overline{\Psi}_j\right]$$

$$+ \Delta t\left(-\sum_{j=1}^{2}g_j^{n+1}g_j^n\widehat{\Psi}_j * \check{\Psi}_j\right.$$

$$\left. - \frac{1}{2}\left[g_1^{n+1}g_2^n + g_2^{n+1}g_1^n\right]\left[\check{\Psi}_2 * \widehat{\Psi}_1 + \check{\Psi}_1 * \widehat{\Psi}_2\right]\right.$$

$$\left. + \mathbf{F}_1^n + \frac{\Delta t}{2}\acute{\mathbf{F}}_1^n\right). \tag{39}$$

In order to make reduction in error, the obtained u_i from (38) is substituted in the right hand side of (30) that can be written as follows:

$$\mathbf{u}^n = \mathbf{A}_1\mathbf{u}^n + g_1^n\widehat{\Psi}_1 + g_2^n\widehat{\Psi}_2, \tag{40}$$

and the obtained value is considered as u_i. Therefore, from (38) and (40), it yields that

$$\mathbf{u}^{n+1} = \mathbf{A}_1\mathbf{M}_1^{-1}\mathbf{N}_1\mathbf{A}_1^{-1}\mathbf{u}^n + \mathbf{A}_1\mathbf{M}_1^{-1}\Psi$$

$$- \mathbf{A}_1\mathbf{M}_1^{-1}\mathbf{N}_1\mathbf{A}_1^{-1}\left(g_1^n\widehat{\Psi}_1 + g_2^n\widehat{\Psi}_2\right) + g_1^{n+1}\widehat{\Psi}_1 + g_2^{n+1}\widehat{\Psi}_2. \tag{41}$$

Hence, the unknown parameters u_i are specified from (41) instead of (38).

3.2. The Indirect MQ Quasi-Interpolation Scheme.

In indirect scheme, the highest order derivatives (second order in this paper) of the solution function are first approximated by (19), and their lower order derivatives and the solution function are then obtained by symbolic integration. Therefore, u_{xx}^n can be approximated by MQ quasi-interpolation $\mathscr{L}_{\mathscr{W}_2}$ on data $\{x_j\}_{j=1}^{N-1}$ as follows:

$$u_{xx}^n(x) = \sum_{j=1}^{N-1}u_{xx}^n(x_j)\widehat{\psi}_j(x). \tag{42}$$

Now, integrating (42) yields

$$u_x^n(x) = \sum_{j=1}^{N-1}u_{xx}^n(x_j)\int\widehat{\psi}_j(x)\,dx + C_1, \tag{43}$$

$$u^n(x) = \sum_{j=1}^{N-1}u_{xx}^n(x_j)\int\int\widehat{\psi}_j(x)\,dx\,dx + C_1x + C_2. \tag{44}$$

Equations (42)–(44) can be rewritten in the compact form as follows:

$$u^n(x) = \sum_{j=0}^{N}w_j^n\varphi_j(x),$$

$$u_x^n(x) = \sum_{j=0}^{N}w_j^n\dot{\varphi}_j(x), \tag{45}$$

$$u_{xx}^n(x) = \sum_{j=0}^{N}w_j^n\ddot{\varphi}_j(x),$$

where

$$\varphi_j(x) = \int\int\widehat{\psi}_j(x)\,dx\,dx, \quad 1 \leqslant j \leqslant N-1,$$

$$\varphi_0(x) = x, \qquad \varphi_N(x) = 1,$$

$$\dot{\varphi}_j(x) = \int\widehat{\psi}_j(x)\,dx, \quad 1 \leqslant j \leqslant N-1,$$

$$\dot{\varphi}_0(x) = 1, \qquad \dot{\varphi}_N(x) = 0, \tag{46}$$

$$\ddot{\varphi}_j(x) = \widehat{\psi}_j(x), \quad 1 \leqslant j \leqslant N-1,$$

$$\ddot{\varphi}_0(x) = 0, \qquad \ddot{\varphi}_N(x) = 0,$$

$$w_j^n = u_{xx}^n(x_j), \quad 1 \leqslant j \leqslant N-1,$$

$$w_0^n = C_1, \qquad w_N^n = C_2.$$

Similar to direct scheme, replacing (45) into (29) and (3) and applying collocation method lead to

$$\sum_{k=0}^{N} w_k^{n+1} \left[\left(1 + \frac{\Delta t}{2} \sum_{j=0}^{N} w_j^n \dot{\varphi}_{ij} \right) \varphi_{ik} \right.$$

$$\left. + \frac{\Delta t}{2} \left(\sum_{j=0}^{N} w_j^n \varphi_{ij} \right) \dot{\varphi}_{ik} - v \frac{\Delta t}{2} \ddot{\varphi}_{ik} \right]$$

$$= \sum_{j=0}^{N} w_j^n \left(\varphi_{ij} + v \frac{\Delta t}{2} t \ddot{\varphi}_{ij} \right) \tag{47}$$

$$+ \Delta t \left(F_i^n + \frac{\Delta t}{2} F_t^n (x_i) \right), \quad 1 \leqslant i \leqslant N - 1,$$

$$\sum_{k=0}^{N} w_k^{n+1} \varphi_k (x_0) = g_1^{n+1},$$

$$\sum_{k=0}^{N} w_k^{n+1} \varphi_k (x_N) = g_2^{n+1},$$

where $\varphi_{ij} = \varphi_j(x_i)$, $\dot{\varphi}_{ij} = \dot{\varphi}_j(x_i)$, $\ddot{\varphi}_{ij} = \ddot{\varphi}_j(x_i)$, and $g_i^{n+1} = g_i(t_{n+1})$, $i = 1, 2$.

Equations (47) generate a system of $N+1$ linear equations in $N + 1$ unknown parameters w_i^{n+1}.

Similar to the direct quasi-interpolation scheme, (47) can be written in matrix form

$$\left[\mathbf{A}_d + \mathbf{A}_b + \frac{\Delta t}{2} \mathbf{u}_x^n * \mathbf{A}_d + \frac{\Delta t}{2} \mathbf{u}^n * \mathbf{D}_2 - v \frac{\Delta t}{2} \ddot{\mathbf{D}}_2 \right] \mathbf{w}^{n+1}$$

$$= \left[\mathbf{A}_d + v \frac{\Delta t}{2} \ddot{\mathbf{D}}_2 \right] \mathbf{w}^n + \mathbf{G}^{n+1} + \Delta t \mathbf{F}_2^n + \frac{\Delta t^2}{2} \acute{\mathbf{F}}_2^n, \tag{48}$$

where, in this case,

$$\mathbf{A}_{d_{(i+1)(j+1)}} = \varphi_{ij}, \qquad \mathbf{A}_{b_{(i+1)(j+1)}} = 0,$$

$$\mathbf{D}_{2_{(i+1)(j+1)}} = \dot{\varphi}_{ij}, \qquad \ddot{\mathbf{D}}_{2_{(i+1)(j+1)}} = \ddot{\varphi}_{ij}, \tag{49}$$

for $i = 1, \dots, N - 1$; $j = 0, 1, \dots, N$ and

$$\mathbf{A}_{d_{(i+1)(j+1)}} = 0, \qquad \mathbf{A}_{b_{(i+1)(j+1)}} = \varphi_{ij},$$

$$\mathbf{D}_{2_{(i+1)(j+1)}} = 0, \qquad \ddot{\mathbf{D}}_{2_{(i+1)(j+1)}} = 0, \tag{50}$$

for $i = 0, N$; $j = 0, 1, \dots, N$ and

$$\mathbf{G}^{n+1} = \left[g_1^{n+1}, 0, \dots, 0, g_2^{n+1} \right]^T,$$

$$\mathbf{F}_2^n = \left[0, F^n (x_1), \dots, F^n (x_{N-1}), 0 \right]^T, \tag{51}$$

$$\acute{\mathbf{F}}_2^n = \left[0, F_t^n (x_1), \dots, F_t^n (x_{N-1}), 0 \right]^T.$$

Subsequently, (48) can be written as

$$\mathbf{w}^{n+1} = \mathbf{M}_2^{-1} \mathbf{N}_2 \mathbf{w}^n + \mathbf{M}_2^{-1} \mathbf{G}^{n+1} + \mathbf{M}_2^{-1} \ddot{\mathbf{F}}^n, \tag{52}$$

where

$$\mathbf{M}_2 = \mathbf{A}_2 + \frac{\Delta t}{2} \mathbf{u}_x^n * \mathbf{A}_d + \frac{\Delta t}{2} \mathbf{u}^n * \mathbf{D}_2 - v \frac{\Delta t}{2} \ddot{\mathbf{D}}_2,$$

$$\mathbf{N}_2 = \mathbf{A}_d + v \frac{\Delta t}{2} \ddot{\mathbf{D}}_2, \tag{53}$$

$$\ddot{\mathbf{F}}^n = \Delta t \mathbf{F}_2^n + \frac{\Delta t^2}{2} \acute{\mathbf{F}}_2^n,$$

and $\mathbf{A}_2 = \mathbf{A}_d + \mathbf{A}_b$. From (42), it yields that

$$\mathbf{u}^n = \mathbf{A}_2 \mathbf{w}^n. \tag{54}$$

Hence, the combination of (52) and (54) is given as

$$\mathbf{u}^{n+1} = \mathbf{A}_2 \mathbf{M}_2^{-1} \mathbf{N}_2 \mathbf{A}_2^{-1} \mathbf{u}^n + \mathbf{A}_2 \mathbf{M}_2^{-1} \mathbf{G}^{n+1} + \mathbf{A}_2 \mathbf{M}_2^{-1} \ddot{\mathbf{F}}^n. \tag{55}$$

4. The Stability Analysis

In this section, the stability analysis from direct and indirect quasi-interpolation schemes is presented by using spectral radius of the amplification matrix similar to the work that Siraj-ul-Islam et al. did in [34]. Let $\mathbf{u}$ be the exact and $\mathbf{u}^*$ the numerical solution of (1); then the error vector $\varepsilon^{n+1} = \mathbf{u}^{n+1} - \mathbf{u}^{*n+1}$ in the direct and indirect quasi-interpolation schemes can be written as

$$\varepsilon^{n+1} = \mathbf{u}^{n+1} - \mathbf{u}^{*n+1} = \mathbf{A}_1 \mathbf{M}_1^{-1} \mathbf{N}_1 \mathbf{A}_1^{-1} \varepsilon^n = \mathbf{E}_1 \varepsilon^n,$$

$$\varepsilon^{n+1} = \mathbf{u}^{n+1} - \mathbf{u}^{*n+1} = \mathbf{A}_2 \mathbf{M}_2^{-1} \mathbf{N}_2 \mathbf{A}_2^{-1} \varepsilon^n = \mathbf{E}_2 \varepsilon^n, \tag{56}$$

where $\mathbf{E}_1 = \mathbf{A}_1 \mathbf{M}_1^{-1} \mathbf{N}_1 \mathbf{A}_1^{-1}$ and $\mathbf{E}_2 = \mathbf{A}_2 \mathbf{M}_2^{-1} \mathbf{N}_2 \mathbf{A}_2^{-1}$. For the stability of the numerical schemes, we must have $\varepsilon^n \to 0$ as $n \to \infty$; that is, $\rho(\mathbf{E}_1) \leqslant 1$, $\rho(\mathbf{E}_2) \leqslant 1$, which is the necessary and sufficient condition for the numerical schemes to be stable, where $\rho(\mathbf{E}_1)$ and $\rho(\mathbf{E}_2)$ denote the spectral radius of the amplification matrices $\mathbf{E}_1$ and $\mathbf{E}_2$, respectively. Equations (56) can be written as

$$\mathbf{M}_1 \mathbf{A}_1^{-1} \varepsilon^{n+1} = \mathbf{N}_1 \mathbf{A}_1^{-1} \varepsilon^n,$$

$$\mathbf{M}_2 \mathbf{A}_2^{-1} \varepsilon^{n+1} = \mathbf{N}_2 \mathbf{A}_2^{-1} \varepsilon^n. \tag{57}$$

Equations (57) can be written into the following forms by using the values of $\mathbf{M}_1$, $\mathbf{N}_1$, $\mathbf{M}_2$, and $\mathbf{N}_2$ defined in (39) and (53):

$$\left[\mathbf{I} + \left(\frac{\Delta t}{2} \right) \mathbf{R}_1 \right] \varepsilon^{n+1} = \left[\mathbf{I} + \left(\frac{\Delta t}{2} \right) \mathbf{R}_2 \right] \varepsilon^n,$$

$$\left[\mathbf{I} + \left(\frac{\Delta t}{2} \right) \mathbf{S}_1 \right] \varepsilon^{n+1} = \left[\mathbf{K} + \left(\frac{\Delta t}{2} \right) \mathbf{S}_2 \right] \varepsilon^n, \tag{58}$$

where

$$\mathbf{R}_1 = \left[\mathbf{u}_x^n * \mathbf{A}_1 + \mathbf{u}^n * \mathbf{D}_1 - v \ddot{D}_1 \right] \mathbf{A}_1^{-1},$$

$$\mathbf{R}_2 = \left[v \ddot{D}_1 - \sum_{j=1}^{2} g_j^{n+1} \left(\widehat{\Psi}_j * \mathbf{D}_1 + \check{\Psi}_j * \mathbf{A}_1 \right) \right] \mathbf{A}_1^{-1}, \tag{59}$$

$$\mathbf{S}_1 = \mathbf{A}_2^{-1} \left[\mathbf{u}_x^n * \mathbf{A}_d + \mathbf{u}^n * \mathbf{D}_2 - v \ddot{D}_2 \right],$$

$$\mathbf{K} = \mathbf{A}_2^{-1} \mathbf{A}_d, \qquad \mathbf{S}_2 = \mathbf{A}_2^{-1} \left[v \ddot{D}_2 \right].$$

The condition of stability will be satisfied if maximum eigenvalue of the matrix $\mathbf{E}_1 = [\mathbf{I} + (\Delta t/2)\mathbf{R}_1]^{-1}[\mathbf{I} + (\Delta t/2)\mathbf{R}_2]$ and maximum eigenvalue of the matrix $\mathbf{E}_2 = [\mathbf{I} + (\Delta t/2)\mathbf{S}_1]^{-1}[\mathbf{K} + (\Delta t/2)\mathbf{S}_2]$ are less than unity (in direct and indirect MQ quasi-interpolation schemes, resp.); that is,

$$\left| \frac{1 + (\Delta t/2)\,\eta_{R_2}}{1 + (\Delta t/2)\,\eta_{R_1}} \right| \leqslant 1,$$

$$\left| \frac{\eta_K + (\Delta t/2)\,\eta_{S_2}}{1 + (\Delta t/2)\,\eta_{S_1}} \right| \leqslant 1, \tag{60}$$

where η_{R_1}, η_{R_2}, η_{S_1}, η_{S_2}, and η_K denote the eigenvalues of the matrices $\mathbf{R}_1$, $\mathbf{R}_2$, $\mathbf{S}_1$, $\mathbf{S}_2$, and $\mathbf{K}$, respectively. It is clear from (60) that the stability of the methods depends on the time step Δt and eigenvalues of the matrices η_{R_1}, η_{R_2}, η_{S_1}, η_{S_2}, and η_K. The condition numbers and magnitude of the eigenvalues of the matrices $\mathbf{R}_1$, $\mathbf{R}_2$, $\mathbf{S}_1$, $\mathbf{S}_2$, and $\mathbf{K}$ depend on the shape parameter and the number of collocation points. Hence, the condition number and the spectral radius of the matrices $\mathbf{E}_1$ and $\mathbf{E}_2$ are dependent on the shape parameter and the number of collocation points. Since it is not possible to find explicit relationship among the spectral radius of the matrices and the shape parameter, this dependency is approximated numerically by keeping the number of collocation points fixed.

5. The Numerical Experiments

Five test experiments are studied to investigate the robustness and the accuracy of the proposed methods. The solution function of Burgers' equation is approximated by direct MQ quasi-interpolation (DMQQI) and indirect MQ quasi-interpolation (IMQQI) schemes and the results are compared with analytical solutions and the results in [18, 26, 31, 35]. The L_∞ and L_2 error norms which are defined by

$$L_\infty = \left\| u^{*n} - u^n \right\|_\infty = \max_{0 \leqslant j \leqslant N} \left| u^{*n}\left(x_j\right) - u^n\left(x_j\right) \right|,$$

$$L_2 = \left\| u^{*n} - u^n \right\|_2 = \sqrt{h \sum_{j=0}^{N} \left(u^{*n}\left(x_j\right) - u^n\left(x_j\right) \right)^2} \tag{61}$$

are used to measure the accuracy. Also, the stability analysis of the methods is considered for first experiment. In all experiments, the shape parameter s is considered twice the shape parameter c and N is chosen twice $\overline{N}$. Also, the centers and the collocation points have been chosen as the same and equidistant.

The computations associated with the experiments discussed above were performed in Maple 16 on a PC with a CPU of 2.4 GHZ.

Experiment 1. In this experiment, we consider nonlinear Burgers' equation (1) with $F(x, t) = 0$ and the initial and the boundary conditions:

$$u(x, 0) = \sin(\pi x), \quad 0 \leqslant x \leqslant 1,$$

$$u(0, t) = u(1, t) = 0, \quad t \geqslant 0. \tag{62}$$

The exact series solution of this experiment was given by Cole [8]:

$$u(x, t) = \frac{2\pi v \sum_{k=1}^{\infty} k a_k \sin(k\pi x) \exp\left(-k^2\pi^2 v t\right)}{a_0 + \sum_{k=1}^{\infty} a_k \cos(k\pi x) \exp\left(-k^2\pi^2 v t\right)}, \tag{63}$$

where

$$a_0 = \int_0^1 \exp\left(-\frac{1 - \cos(\pi x)}{2\pi v}\right) dx,$$

$$a_k = 2 \int_0^1 \cos(k\pi x) \exp\left(-\frac{1 - \cos(\pi x)}{2\pi v}\right) dx, \tag{64}$$

$$(k = 1, 2, 3, \ldots).$$

Numerical results are presented for $v = 0.1$ and $v = 0.01$ with $\Delta t = 0.001$ and compared with the exact solutions and the results of the MQ quasi-interpolation scheme (MQQI; see [26]) and adaptive MQ scheme (AMQ; see [18]) for the cases of $v = 0.1$ and $v = 0.01$ in Tables 1 and 2, respectively. Also, the numerical solutions are compared with the results obtained by MQQI scheme, AMQ scheme, and Galerkin scheme [35] for $v = 0.0001$ in Table 3. In the cases $v = 0.1$ and $v = 0.01$, the shape parameter c is $0.815h$. In the case $v = 0.0001$, the parameter c is 2.78×10^{-1} and 1.389×10^{-4} for $N = 36$ and $N = 72$, respectively. The space-time graph of the estimated solution for $v = 0.1$ and $v = 0.01$ is presented in Figures 1 and 2.

Numerical comparison in these cases shows that the obtained results, particularly in IMQQI scheme, are in good agreement with the exact solutions and the results of the other schemes.

Relation between the spectral radius of the matrices $\mathbf{E}_1$ and $\mathbf{E}_2$ and the different values of the shape parameter c is shown in Table 4 by keeping the number of collocation points fixed. It is clear from Table 4 that if the values of shape parameter c are greater than the critical value $c = 0.1$ ($c = 0.01$), then the solution obtained from the IMQQI (DMQQI) method breaks down and hence the IMQQI and DMQQI methods become unstable. Therefore, the interval stability of IMQQI and DMQQI schemes is $(0, 0.1)$ and $(0.004, 0.01)$, respectively.

It can be seen from Table 4 that the schemes are very sensitive to the values of the shape parameter c and the interval stability of methods is a small interval.

Experiment 2. In this experiment, we consider the shock propagation solution of the homogeneous Burgers' equation [31] as a numerical experiment. This solution is given by

$$u(x, t) = \frac{x}{t\left(1 + \sqrt{t/t_*}\,\exp\left(x^2/4vt\right)\right)},$$

$$t \geqslant 1, \quad t_* = \exp\left(\frac{1}{8v}\right), \quad 0 \leqslant x \leqslant 1.2. \tag{65}$$

The initial condition of the problem is obtained from (65) at time $t = 1$ and the boundary conditions in (3) can be obtained from the exact solution. Propagation of the shock is studied

TABLE 1: Comparison of results with the exact solution and the results in [26] of $v = 0.1$ with $\Delta t = 0.001$ at $t = 1$ for different values of x of Experiment 1.

x	Exact	MQQI [26]	IMQQI		DMQQI		Error	Error
		$N = 100$	$N = 10$	$N = 20$	$N = 10$	$N = 20$	IMQQI	DMQQI
0.1	0.0663	0.0712	0.0663	0.0663	0.0662	0.0662	$8.59E - 07$	$4.41E - 05$
0.2	0.1312	0.1343	0.1313	0.1312	0.1310	0.1311	$2.18E - 06$	$7.77E - 05$
0.3	0.1928	0.1934	0.1929	0.1928	0.1925	0.1926	$4.49E - 06$	$1.21E - 04$
0.4	0.2480	0.2454	0.2483	0.2480	0.2476	0.2478	$8.59E - 06$	$1.88E - 04$
0.5	0.2919	0.2852	0.2924	0.2919	0.2913	0.2916	$1.55E - 05$	$2.85E - 04$
0.6	0.3161	0.3047	0.3169	0.3161	0.3150	0.3155	$2.26E - 05$	$4.15E - 04$
0.7	0.3081	0.2929	0.3095	0.3081	0.3064	0.3072	$3.39E - 05$	$5.53E - 04$
0.8	0.2537	0.2378	0.2558	0.2538	0.2515	0.2526	$4.96E - 05$	$5.98E - 04$
0.9	0.1461	0.1354	0.1481	0.1461	0.1441	0.1455	$4.47E - 05$	$4.16E - 04$

TABLE 2: Comparison of results with the exact solution and the results in [26] of $v = 0.01$ with $\Delta t = 0.001$ at $t = 1$ for different values of x of Experiment 1.

x	Exact	MQQI [26]	IMQQI		DMQQI		Error	Error
		$N = 100$	$N = 20$	$N = 30$	$N = 30$	$N = 60$	IMQQI	DMQQI
0.1	0.0754	0.0787	0.0754	0.0754	0.0727	0.0751	$4.17E - 07$	$2.84E - 04$
0.2	0.1506	0.1520	0.1506	0.1506	0.1496	0.1505	$1.43E - 07$	$1.12E - 04$
0.3	0.2257	0.2255	0.2256	0.2257	0.2253	0.2256	$3.20E - 08$	$3.16E - 05$
0.4	0.3003	0.2990	0.3003	0.3003	0.3002	0.3003	$4.40E - 10$	$8.83E - 06$
0.5	0.3744	0.3723	0.3744	0.3744	0.3744	0.3744	$6.09E - 08$	$4.67E - 06$
0.6	0.4448	0.4478	0.4478	0.4478	0.4478	0.4478	$6.97E - 08$	$4.55E - 06$
0.7	0.5203	0.5164	0.5205	0.5203	0.5203	0.5203	$5.19E - 07$	$4.99E - 06$
0.8	0.5915	0.5862	0.5910	0.5915	0.5915	0.5915	$5.05E - 06$	$4.54E - 06$
0.9	0.6600	0.6296	0.6539	0.6604	0.6598	0.6600	$7.78E - 05$	$1.10E - 06$

TABLE 3: Comparison of results with the results of [18, 26, 35] for $v = 0.0001$ and $\Delta t = 0.001$ at $t = 1$ for different values of x of Experiment 1.

x	Galerkin method [35]	AMQ [18]	MQQI [26]	IMQQI		DMQQI
		$N = 10$	$N = 72$	$N = 36$	$N = 72$	$N = 72$
0.056	0.0422	0.0424	0.0424	0.0562	0.0421	0.0421
0.167	0.1263	0.1263	0.1262	0.1260	0.1263	0.1263
0.278	0.2103	0.2103	0.2096	0.2103	0.2103	0.2103
0.389	0.2939	0.2939	0.2928	0.2939	0.2939	0.2938
0.500	0.3769	0.3769	0.3754	0.3769	0.3769	0.3766
0.611	0.4592	0.4592	0.4572	0.4592	0.4592	0.4565
0.722	0.5404	0.5404	0.5381	0.5404	0.5404	0.5231
0.833	0.6203	0.6201	0.6174	0.6203	0.6203	0.5296
0.944	0.6983	0.6957	0.6948	0.6967	0.6982	0.3815

TABLE 4: The spectral radius and L_∞ and L_2 error norms versus shape parameter c when $\Delta t = 0.001$, $N = 100$, and $v = 0.01$ at $t = 1$ of Experiment 1.

	IMQQI			DMQQI			
c	$\rho(\mathbf{E}_1)$	L_∞	L_2	c	$\rho(\mathbf{E}_2)$	L_∞	L_2
$1.00E - 20$	0.99925	$2.4628E - 04$	$5.1531E - 04$	$4.00E - 03$	0.99925	$5.0580E - 02$	$8.8134E - 03$
$1.00E - 10$	0.99925	$2.4628E - 03$	$5.1531E - 04$	$6.00E - 03$	0.99925	$9.7461E - 03$	$1.6022E - 03$
$1.00E - 05$	0.99925	$2.4642E - 03$	$5.1413E - 04$	$8.00E - 03$	0.99925	$4.7521E - 03$	$7.5521E - 04$
$1.00E - 02$	0.99925	$2.1783E - 03$	$2.8715E - 04$	$1.00E - 02$	0.99925	$5.1918E - 03$	$6.7218E - 04$
$1.00E - 01$	0.99924	$2.8855E - 04$	$3.3028E - 05$	$3.00E - 02$	1.00049	$8.3973E - 01$	$1.2667E - 01$
$1.20E - 01$	5.10176	$1.3242E + 04$	$1.7896E + 04$	$5.00E - 02$	1.05591	$9.3609E - 01$	$1.7319E - 01$

(a)   (b)
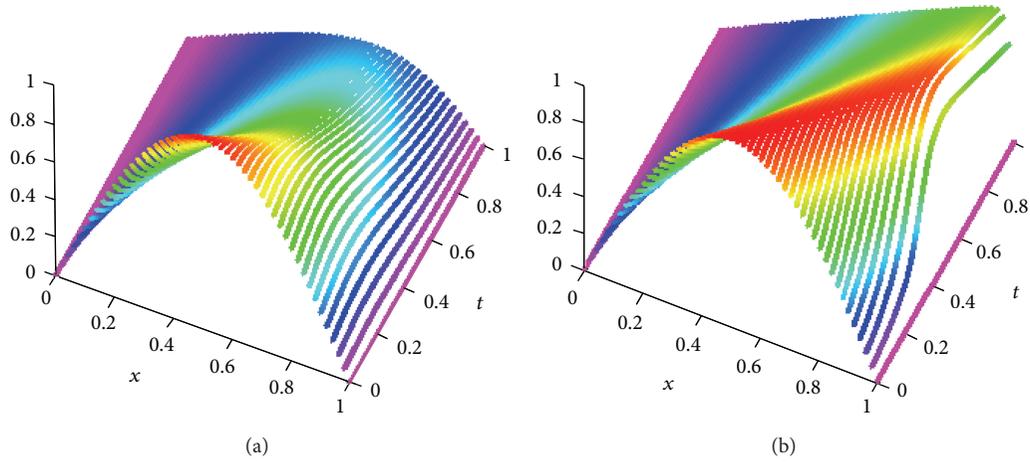
FIGURE 1: The space-time graph of the estimated solution of Burgers' equation by using IMQQI for $x \in [0,1]$, $t \in [0,1]$, $v = 0.1$ (a), and $v = 0.01$ (b) of Experiment 1.
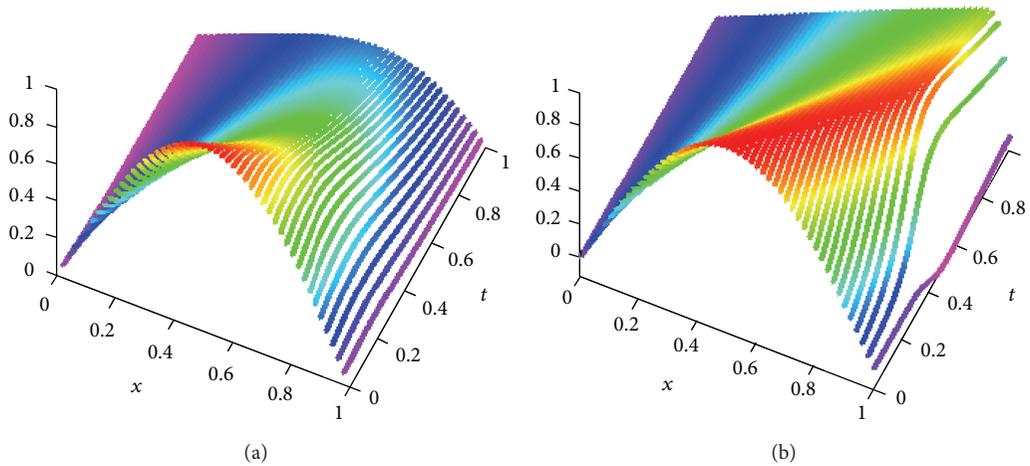

(a)   (b)

FIGURE 2: The space-time graph of the estimated solution of Burgers' equation by using DMQQI for $x \in [0,1]$, $t \in [0,1]$, $v = 0.1$ (a), and $v = 0.01$ (b) of Experiment 1.

with $\Delta t = 0.01$ for $v = 0.005$ and $v = 0.001$. The shape parameter c is denoted by 6.0×10^{-2}, 2.4×10^{-2}, and 1.2×10^{-2} for $N = 20$, $N = 50$, and $N = 100$, respectively. The L_2 and L_∞ error norms are calculated in 240 and 2400 points for $v = 0.005$ and $v = 0.001$, respectively, and compared with the results of [31] in Table 5 at different times. The space-time graph of the estimated solution for $v = 0.005$ is showed in Figure 3.

Experiment 3. In this experiment, we study the fusion phenomenon of the two solitary waves of the homogeneous Burgers' equation. The fusion phenomenon happens when two or more solitons will fusion to one soliton at a specific time. In [9], Wang et al. studied the following Burgers' equation:

$$u_t + 2uu_x - u_{xx} = 0. \tag{66}$$

They obtained the two-solitary-wave solution

$$u(x,t) = -\frac{k_1 e^{k_1(x+k_1 t)} + k_2 e^{k_2(x+k_2 t)}}{1 + e^{k_1(x+k_1 t)} + e^{k_2(x+k_2 t)}}, \tag{67}$$

where k_1 and k_2 are constant. Let

$$t \longrightarrow t, \quad x \longrightarrow \frac{x}{2}. \tag{68}$$

Hence, (66) converts to Burgers' equation form (1) wherein $F(x,t) = 0$ and $v = 0.25$. In this case, two-solitary-wave fusion happens at a specific time $t = 0$. Because we can show the fusion phenomenon, we consider an interval $[-5,5]$ for t and x. For this purpose, we introduce a new time variable $\tau = t + 5$ and approximate the solution $u(x,\tau)$ of (1) by using our schemes for $\tau \in [0,10]$. Then, we obtain $u(x,t)$ for $t \in [-5,5]$. The initial condition can be obtained from the exact solution at $\tau = 0$. The boundary conditions can be also taken from the exact solution.

TABLE 5: The comparison of L_2 and L_∞ errors between the numerical results by using our schemes and the results of [31] with $\Delta t = 0.01$, $v = 0.005$, and $v = 0.001$ of Experiment 2.

$v = 0.005$	L_∞			L_2		
	$t = 1.7$	$t = 2.4$	$t = 3.1$	$t = 1.7$	$t = 2.4$	$t = 3.1$
IMQQI; $N = 20$	$9.88E - 03$	$3.74E - 03$	$9.70E - 04$	$2.18E - 03$	$7.85E - 04$	$2.48E - 04$
IMQQI; $N = 50$	$7.63E - 05$	$2.88E - 05$	$1.45E - 05$	$1.79E - 05$	$8.35E - 06$	$4.86E - 06$
DMQQI; $N = 50$	$1.82E - 04$	$1.58E - 04$	$9.88E - 05$	$4.12E - 05$	$3.24E - 05$	$2.09E - 05$
TCM [31]; $N = 240$	$6.48E - 05$	$4.32E - 05$	$3.13E - 05$	$1.69E - 05$	$1.21E - 05$	$9.20E - 06$
TGM [31]; $N = 240$	$1.78E - 03$	$1.28E - 03$	$1.00E - 03$	$3.23E - 04$	$2.99E - 04$	$2.75E - 04$
$v = 0.001$	$t = 1.7$	$t = 3.0$	$t = 3.5$	$t = 1.7$	$t = 3.0$	$t = 3.5$
IMQQI; $N = 100$	$7.57E - 03$	$2.85E - 03$	$1.70E - 03$	$1.07E - 03$	$3.20E - 04$	$2.21E - 04$
DMQQI; $N = 100$	$1.71E - 02$	$4.03E - 03$	$5.29E - 03$	$2.26E - 03$	$5.37E - 04$	$6.43E - 04$
TCM [31]; $N = 2400$	$1.54E - 03$	$3.58E - 04$	$2.41E - 04$	$1.89E - 04$	$5.12E - 05$	$3.59E - 05$
TGM [31]; $N = 2400$	$1.56E - 03$	$3.67E - 04$	$2.49E - 04$	$1.93E - 04$	$5.54E - 05$	$4.07E - 05$

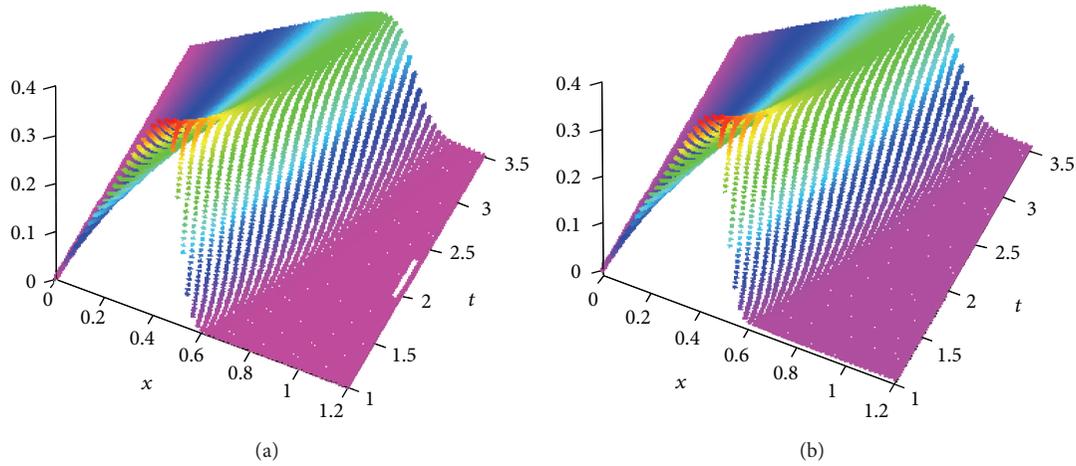(a)   (b)

FIGURE 3: The space-time graph of the estimated solution of Burgers' equation by using IMQQI (a) and DMQQI (b) for $x \in [0, 1.2]$ and $t \in [1, 3.5]$ of Experiment 2.

The L_2 and L_∞ errors of IMQQI scheme are calculated in 1000 points for $k_1 = 1$, $k_2 = -1$, $\Delta t = 0.001$, and $c = h$ and listed in Table 6. From our numerical experiment whose results are not given here, we see that the accuracy of the DMQQI scheme is very bad in this experiment but we can clearly see from Table 6 that the results of the IMQQI scheme are in good agreement with the exact solutions.

The space-time graph of the estimated solution by using IMQQI is presented in Figure 4.

Experiment 4. In this experiment, we consider Burgers' equation (1) with the nonhomogeneous term $F(x, t) = kx/(2\beta t + 1)^2$ that $k > 0$ and $\beta \geqslant 0$ are constant. So, Burgers' equation (1) has the following form:

$$u_t + uu_x = vu_{xx} + \frac{kx}{(2\beta t + 1)^2}. \tag{69}$$

In 2010, Rao and Yadav [12] obtained the solution of Burgers' equation (69) with the initial condition $u(x, 0) = u_0(x) \in L^2(\mathbb{R}, e^{x^2/2})$. They showed that the solution of (69) is given by

$$u(x, t) = \frac{A_0 x}{2\beta t + 1}, \quad A_0 = \beta + \sqrt{\beta^2 + k}, \tag{70}$$

for the initial condition $u(x, 0) = kx$, $k > \beta$.

In this paper, we simulate this solution for $k = 5$ and $\beta = 2$ in $x \in [-1, 1]$. The boundary conditions in (3) can be obtained from the exact solution (70). The L_2 and L_∞ error norms are calculated in 1000 points for arbitrary v, $\Delta t = 0.01$, and $c = h$ and listed in Table 7. Also, the space-time graph of the estimated solution by using IMQQI and DMQQI schemes is presented in Figure 5.

Table 7 shows that the accuracy of the DMQQI scheme is low even if the number of collocation points increases,
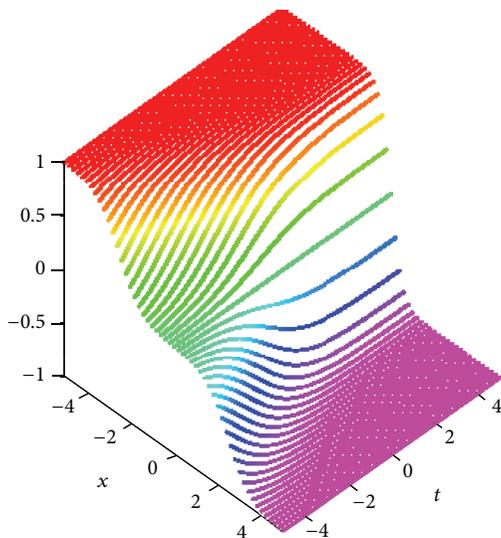
FIGURE 4: The space-time graph of the estimated solution of Burgers' equation by using IMQQI for $x \in [-5, 5]$ and $t \in [-5, 5]$ of Experiment 3.
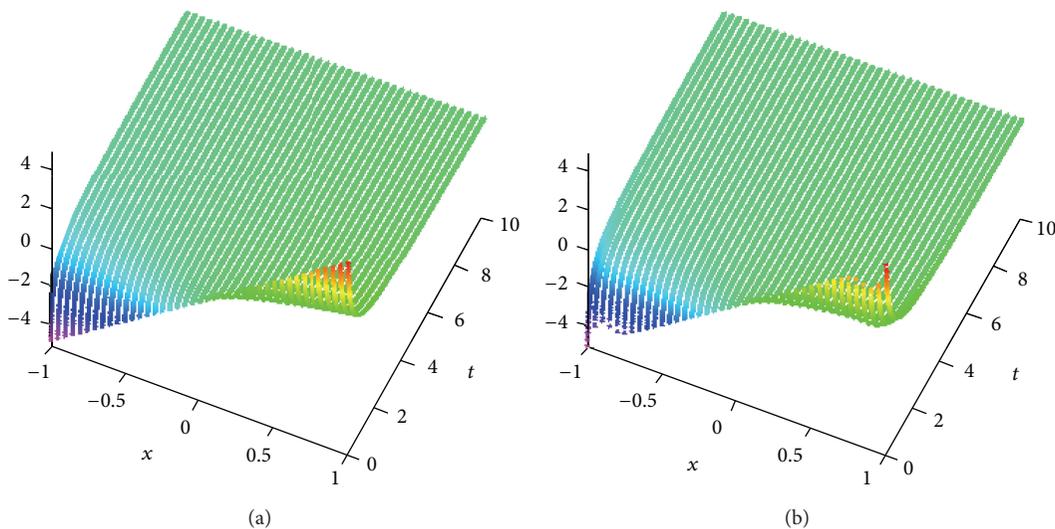

(a)

(b)

FIGURE 5: The space-time graph of the estimated solution of Burgers' equation by using IMQQI (a) and DMQQI (b) for $x \in [-1, 1]$ and $t \in [0, 10]$ of Experiment 4.

TABLE 6: The L_2 and L_∞ errors of the IMQQI scheme with $\Delta t = 0.001$ at different times of Experiment 3.

	L_∞			L_2		
	$t = -4$	$t = 0$	$t = 5$	$t = -4$	$t = 0$	$t = 5$
IMQQI; $N = 20$	$3.907E - 04$	$3.242E - 03$	$1.592E - 02$	$5.660E - 04$	$3.181E - 03$	$1.728E - 02$
IMQQI; $N = 40$	$7.678E - 06$	$2.661E - 05$	$3.379E - 04$	$1.174E - 05$	$1.670E - 05$	$2.119E - 04$

TABLE 7: The comparison of L_2 and L_∞ errors between the numerical results of our schemes with $\Delta t = 0.01$ of Experiment 4.

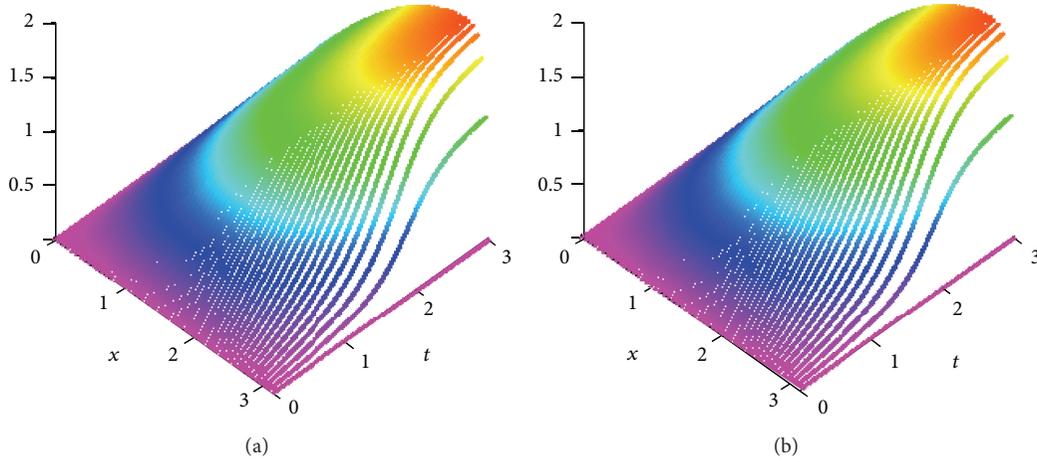|	L_∞			L_2		
	$t = 1$	$t = 5$	$t = 10$	$t = 1$	$t = 5$	$t = 10$
IMQQI; $N = 10$	$1.171E - 06$	$2.816E - 09$	$1.876E - 10$	$8.394E - 08$	$2.020E - 09$	$1.345E - 10$
DMQQI; $N = 20$	$1.533E - 01$	$4.146E - 02$	$2.159E - 02$	$7.441E - 02$	$2.102E - 02$	$1.101E - 02$
DMQQI; $N = 40$	$1.234E - 01$	$3.446E - 02$	$2.034E - 02$	$6.612E - 02$	$2.102E - 02$	$1.037E - 02$

Figure 6: The space-time graph of the estimated solution of Burgers' equation by using IMQQI (a) and DMQQI (b) for $v = 0.1$, $x \in [0, \pi]$ and $t \in [0, 3]$ of Experiment 5.

Table 8: Comparison of numerical results with the exact solutions for $v = 1$ with $q = 20$ and $\Delta t = 0.001$ at $t = 3$ of Experiment 5.

x	Exact	IMQQI		DMQQI		Error IMQQI	Error DMQQI
		$N = 10$	$N = 20$	$N = 20$	$N = 30$		
0.5	2.1481	2.1500	2.1481	2.1343	2.1474	$1.025E - 05$	$6.383E - 04$
1.0	4.1562	4.1563	4.1562	4.1512	4.1558	$6.026E - 06$	$4.374E - 04$
1.5	5.8928	5.9011	5.8928	5.8842	5.8924	$1.619E - 05$	$3.516E - 04$
2.0	7.2404	7.2593	7.2404	7.2264	7.2400	$3.306E - 05$	$3.083E - 04$
2.5	8.0302	8.0156	8.0358	8.0056	8.0298	$2.109E - 04$	$3.599E - 04$
3.0	4.5140	4.8006	4.5143	4.4773	4.4965	$2.169E - 04$	$1.750E - 02$

whereas the IMQQI scheme provides the good results with a small number of points.

Experiment 5. We finally closed our analysis by considering the following Burgers' equation with the nonhomogeneous term:

$$u_t + u u_x = v u_{xx} + A \sin(x), \quad A > 0, \tag{71}$$

with the initial condition

$$u(x, 0) = 0, \quad x \in [0, \pi], \tag{72}$$

which was discussed in [1], as a nonlinear model for describing hypersound generation in prescribed light field. The solution of the differential equation (71) can be written as a series of Mathieu's functions [1]:

$$u(x, t)$$
$$= 2v \frac{\partial}{\partial x} \ln \left[\sum_{k=0}^{\infty} a_{2k} \exp\left(-\frac{v \lambda_{2k}(q)}{4} t \right) ce_{2k}\left(\frac{\pi - x}{2}, q \right) \right], \tag{73}$$

where

$$a_{2k} = \frac{\left[\int_0^{2\pi} ce_0(x/2, q) \, dx \right]}{\left[\int_0^{2\pi} ce_{2k}^2(x/2, q) \, dx \right]} \tag{74}$$

and $q = A/v^2$. The notations used here correspond to those from the book by Strutt [36]. The boundary conditions are

$$u(0, t) = u(\pi, t) = 0, \quad t \geq 0. \tag{75}$$

The numerical results compared with the exact solutions for $v = 1$ with $q = 20$ and $v = 0.1$ with $q = 100$ in Tables 8 and 9, respectively. The numerical calculations are performed with $\Delta t = 0.001$ and $c = h$. The space-time graph of the estimated solutions is also presented for $v = 0.1$ in Figure 6.

6. Conclusion

In this paper, two numerical schemes based on high accuracy MQ quasi-interpolation scheme and RBFs approximation schemes (IRBF and DRBF approximation schemes) have been presented for solving the nonlinear nonhomogeneous Burgers' equation. The accuracy of the methods can be increased by selecting the appropriate shape parameter. The choice of the shape parameter is still a pendent question.

The numerical results which are given in the previous section indicate that the performance of the methods specially IMQQI is in excellent agreement with the exact solutions. Tables 1–9 show that the IMQQI scheme is more accurate than DMQQI scheme as expected, and the interval stability of the IMQQI method is greater than the DMQQI method. Also, the IMQQI scheme required less nodes in comparison with the DMQQI scheme. We can even get good results with

TABLE 9: Comparison of numerical results with the exact solutions for $v = 0.1$ with $q = 100$ and $\Delta t = 0.001$ at $t = 3$ of Experiment 5.

x	Exact	IMQQI		DMQQI		Error IMQQI	Error DMQQI
		$N = 20$	$N = 30$	$N = 30$	$N = 60$		
0.5	0.4824	0.4853	0.4851	0.4822	0.4851	$2.679E - 03$	$2.696E - 03$
1.0	0.9331	0.9392	0.9392	0.9383	0.9391	$6.056E - 03$	$6.006E - 03$
1.5	1.3221	1.3318	1.3300	1.3317	1.3320	$7.894E - 03$	$9.878E - 03$
2.0	1.6222	1.6382	1.6271	1.6370	1.6371	$4.911E - 03$	$1.490E - 02$
2.5	1.8102	1.8281	1.8122	1.8322	1.8322	$1.833E - 03$	$2.195E - 02$
3.0	1.6155	1.6657	1.6207	1.5897	1.6476	$5.433E - 03$	$3.238E - 02$

less number of points specially in IMQQI method but the results are bad at the ends of interval that we can improve it by using the knot method [37]. Hence, we will not encounter with large scale matrix. Besides, we use equidistant points in our numerical experiments but our schemes can be used for the scattered points.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

References

[1] A. A. Karabutov, E. A. Lapshin, and O. V. Rudenko, "Interaction between light waves and sound under acoustic nonlinearity conditions," *Journal of Experimental and Theoretical Physics*, vol. 44, pp. 58–63, 1976.

[2] J. A. Atwell and B. B. King, "Stabilized finite element methods and feedback control for Burgers' equation," in *Proceedings of the American Control Conference*, pp. 2745–2749, June 2000.

[3] E. Moreau and O. Vallée, "The Burgers' equation aselectro-hydrodynamic model in plasma physics," http://arxiv.org/abs/physics/0501019.

[4] E. Moreau and O. Vallée, "Connection between the Burgers equation with an elastic forcing term and a stochastic process," *Physical Review E*, vol. 73, no. 1, Article ID 016112, 4 pages, 2006.

[5] H. Bateman, "Some recent researches on the motion of fluids," *Monthly Weather Review*, vol. 43, pp. 163–170, 1915.

[6] J. M. Burgers, "A mathematical model illustrating the theory of turbulence," in *Advances in Applied Mechanics*, pp. 171–199, Academic Press Inc., New York, N. Y., 1948, edited by Richard von Mises and Theodore von Kármán,.

[7] E. Hopf, "The partial differential equation $u_t + uu_x = \mu u_{xx}$," *Communications on Pure and Applied Mathematics*, vol. 3, pp. 201–230, 1950.

[8] J. D. Cole, "On a quasi-linear parabolic equation occurring in aerodynamics," *Quarterly of Applied Mathematics*, vol. 9, pp. 225–236, 1951.

[9] S. Wang, X. Tang, and S.-Y. Lou, "Soliton fission and fusion: Burgers equation and Sharma-Tasso-Olver equation," *Chaos, Solitons & Fractals*, vol. 21, no. 1, pp. 231–239, 2004.

[10] A. Chen, F.-F. Wang, and W. Zhang, "Soliton solutions of Burgers' equation and the modified Kadomtsev-Petviashvili equation," *Journal of Physics. A. Mathematical and Theoretical*, vol. 43, no. 36, Article ID 365202, 11 pages, 2010.

[11] X. Ding, Q. Jiu, and C. He, "On a nonhomogeneous Burgers' equation," *Science in China. Series A. Mathematics*, vol. 44, no. 8, pp. 984–993, 2001.

[12] Ch. S. Rao and M. K. Yadav, "On the solution of a nonhomogeneous Burgers equation," *International Journal of Nonlinear Science*, vol. 10, no. 2, pp. 141–145, 2010.

[13] A. Hashemian and H. M. Shodja, "A meshless approach for solution of Burgers' equation," *Journal of Computational and Applied Mathematics*, vol. 220, no. 1-2, pp. 226–239, 2008.

[14] S. Kutluay, A. R. Bahadır, and A. Özdeş, "Numerical solution of one-dimensional Burgers equation: explicit and exact-explicit finite difference methods," *Journal of Computational and Applied Mathematics*, vol. 103, no. 2, pp. 251–261, 1999.

[15] X. H. Zhang, J. Ouyang, and L. Zhang, "Element-free characteristic Galerkin method for Burgers' equation," *Engineering Analysis with Boundary Elements*, vol. 33, no. 3, pp. 356–362, 2009.

[16] A. R. Bahadır and M. Sağlam, "A mixed finite difference and boundary element approach to one-dimensional Burgers' equation," *Applied Mathematics and Computation*, vol. 160, no. 3, pp. 663–673, 2005.

[17] E. J. Kansa, "Multiquadrics—a scattered data approximation scheme with applications to computational fluid-dynamics. I. Surface approximations and partial derivative estimates," *Computers & Mathematics with Applications*, vol. 19, no. 8-9, pp. 127–145, 1990.

[18] Y. C. Hon and X. Z. Mao, "An efficient numerical scheme for Burgers' equation," *Applied Mathematics and Computation*, vol. 95, no. 1, pp. 37–50, 1998.

[19] Y.-C. Hon, K. F. Cheung, X.-Z. Mao, and E. J. Kansa, "Multiquadric solution for shallow water equations," *Journal of Hydraulic Engineering*, vol. 125, no. 5, pp. 524–533, 1999.

[20] M. Zerroukat, H. Power, and C. S. Chen, "A numerical method for heat transfer problem using collocation and radial basis functions," *International Journal of Numerical Methods in Engineering*, vol. 42, pp. 1263–1278, 1992.

[21] N. Mai-Duy and T. Tran-Cong, "Numerical solution of differential equations using multiquadric radial basis function networks," *Neural Networks*, vol. 14, no. 2, pp. 185–199, 2001.

[22] C. Shu and L. Wu, "Integrated radial basis functions-based differential quadrature method and its performance," *International Journal for Numerical Methods in Fluids*, vol. 53, pp. 969–984, 2007.

[23] R. K. Beatson and M. J. D. Powell, "Univariate multiquadric approximation: quasi-interpolation to scattered data," *Constructive Approximation*, vol. 8, no. 3, pp. 275–288, 1992.

[24] Z. M. Wu and R. Schaback, "Shape preserving properties and convergence of univariate multiquadric quasi-interpolation," *Acta Mathematicae Applicatae Sinica. English Series*, vol. 10, no. 4, pp. 441–446, 1994.

[25] Z.-W. Jiang, R.-H. Wang, C.-G. Zhu, and M. Xu, "High accuracy multiquadric quasi-interpolation," *Applied Mathematical Modelling*, vol. 35, no. 5, pp. 2185–2195, 2011.

[26] R. Chen and Z. Wu, "Applying multiquadratic quasi-interpolation to solve Burgers' equation," *Applied Mathematics and Computation*, vol. 172, no. 1, pp. 472–484, 2006.

[27] Y. C. Hon, "A quasi-radial basis functions method for American options pricing," *Computers & Mathematics with Applications*, vol. 43, no. 3–5, pp. 513–524, 2002.

[28] Z.-W. Jiang and R.-H. Wang, "Numerical solution of one-dimensional Sine-Gordon equation using high accuracy multiquadric quasi-interpolation," *Applied Mathematics and Computation*, vol. 218, no. 15, pp. 7711–7716, 2012.

[29] M. L. Xiao, R. H. Wang, and C. G. Zhu, "Applying multiquadric quasi-interpolation to solve KdV equation," *Journal of Mathematical Research and Exposition*, vol. 31, no. 2, pp. 191–201, 2011.

[30] P. D. Lax and B. Wendroff, "Difference schemes for hyperbolic equations with high order of accuracy," *Communications on Pure and Applied Mathematics*, vol. 17, pp. 381–398, 1964.

[31] İ. Dağ, A. Canıvar, and A. Şahin, "Taylor-Galerkin and Taylor-collocation methods for the numerical solutions of Burgers' equation using B-splines," *Communications in Nonlinear Science and Numerical Simulation*, vol. 16, no. 7, pp. 2696–2708, 2011.

[32] R. L. Hardy, "Multiquadric equations of topography and other irregular surfaces," *Journal of Geophysical Research*, vol. 76, no. 8, pp. 1905–1915, 1971.

[33] W. R. Madych and S. A. Nelson, "Multivariate interpolation and conditionally positive definite functions. II," *Mathematics of Computation*, vol. 54, no. 189, pp. 211–230, 1990.

[34] Siraj-ul-Islam, S. Haq, and M. Uddin, "A meshfree interpolation method for the numerical solution of the coupled nonlinear partial differential equations," *Engineering Analysis with Boundary Elements*, vol. 33, no. 3, pp. 399–409, 2009.

[35] I. Christie and A. R. Mitchell, "Upwinding of high order Galerkin methods in conduction-convection problems," *International Journal for Numerical Methods in Engineering*, vol. 12, no. 11, pp. 1764–1771, 1978.

[36] M. J. O. Strutt, *Lame, Mathieu and Related Functions in Physics and Technology*, ONTI, Moscow, Russia, 1935, reprinted by Edwards Brothers, 1944.

[37] B. Fornberg, T. A. Driscoll, G. Wright, and R. Charles, "Observations on the behavior of radial basis function approximations near boundaries," *Computers & Mathematics with Applications*, vol. 43, no. 3–5, pp. 473–490, 2002.