Hindawi
Journal of Mathematics
Volume 2021, Article ID 7240300, 13 pages
https://doi.org/10.1155/2021/7240300

Research Article

A Composite Algorithm for Numerical Solutions of Two-Dimensional Coupled Burgers' Equations

Vikas Kumar,[1] **Sukhveer Singh**,[2] **and Mehmet Emir Koksal**[3]

[1]*Department of Mathematics, D. A. V. College Pundri, Kaithal 136026, Haryana, India*
[2]*Department of Mathematics, Indian Institute of Technology, Roorkee 247667, India*
[3]*Department of Mathematics, Ondokuz Mayıs University, Atakum, Samsun 55139, Turkey*

Correspondence should be addressed to Mehmet Emir Koksal; mekoksal@omu.edu.tr

Received 10 May 2021; Accepted 19 June 2021; Published 1 July 2021

Academic Editor: M. M. Bhatti

In this study, a new composite algorithm with the help of the finite difference and the modified cubic trigonometric B-spline differential quadrature method is developed. The developed method was applied to two-dimensional coupled Burgers' equation with initial and Dirichlet boundary conditions for computational modeling. The established algorithm is better than the traditional differential quadrature algorithm proposed in literature due to more smoothness of cubic trigonometric B-spline functions. In the development of the algorithm, the first step is semidiscretization in time with the forward finite difference method. Furthermore, the obtained system is fully discretized by the modified cubic trigonometric B-spline differential quadrature method. Finally, we obtain coupled Lyapunov systems of linear equations, which are analyzed by the MATLAB solver for the system. Moreover, comparative study of these solutions with the numerical and exact solutions which are appeared in the literature is also discussed. Finally, it is found that there is good suitability between exact solutions and numerical solutions obtained by the developed composite algorithm. The technique can be extended for various multidimensional Burgers' equations after some modifications.

1. Introduction

In this paper, the authors considered the following dimensionless form of two-dimensional (2D) coupled Burgers' equation:

$$\frac{1}{\mathrm{Re}}\nabla^2 u = \frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y}, \tag{1a}$$

$$\frac{1}{\mathrm{Re}}\nabla^2 v = \frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y}, \tag{1b}$$

with initial conditions (ICs),

$$u(x, y, 0) = \psi_1(x, y), \quad (x, y) \in [\alpha, \beta] \times [\gamma, \delta], \tag{2a}$$

$$v(x, y, 0) = \psi_2(x, y), \quad (x, y) \in [\alpha, \beta] \times [\gamma, \delta], \tag{2b}$$

and Dirichlet boundary conditions (BCs),

$$\begin{aligned} u(\alpha, y, t) &= h_1(y, t), \\ u(\beta, y, t) &= h_2(y, t), \\ u(x, \gamma, t) &= h_3(x, t), \\ u(x, \delta, t) &= h_4(x, t), \end{aligned} \tag{3}$$

where $\nabla^2 = (\partial^2/\partial x^2) + (\partial^2/\partial y^2)$ is Laplace operator, $u(x, y, t)$ and $v(x, y, t)$ are velocity components to be determined. Also, $h_i, i = 1, 2, \ldots, 4$ are known smooth functions and $\mathrm{Re} = (\rho|\overrightarrow{u}|L/\mu)$ is the Reynolds number with density ρ, viscosity μ, characteristic length L, and $\overrightarrow{u} = [u, v]^T$.

The nonlinear convection-diffusion model is simply represented by Burgers' equation [1]. This famous equation describes the flow theory through a shockwave moving in viscous liquid [2], phenomena of turbulence [3], and various other kinds of phenomena in aerodynamics.

Due to its extensive scope of applicability, various numerical schemes have been constructed to study its numerical solutions. Moreover, due to its application in various fields of science and technology, researchers and scientists are still interested in developing algorithms to find their numerical and exact solutions. A great number of works has been studied for finding approximate solutions of Burgers' equation, for example, cubic spline method [4], finite element and difference methods [5–8], multilevel alternating direction implicit schemes [9], and various explicit and implicit methods [10, 11]. Furthermore, the decomposition method [12], spectral method [13], Chebyshev collocation method [14], and local discontinuous radial basis function collocation method [15, 16] are investigated in literature. Also, Haar wavelet quasilinearization approach [17] and differential quadrature methods (DQMs) [18–23] have been developed. In recent years, new meshless methods [24–26] for various types of Burgers' equations have been developed.

In Lagrange interpolation-based DQMs [18–22], Lagrange's fundamentals are used to compute the weighting coefficients. In these cases, as the number of grid points increases the weights become unstable. Herein, to reduce this instability, the modified cubic trigonometric B-spline functions are used to the weighting coefficients of DQMs.

In this article, a new numerical algorithm is developed based on the finite difference and the modified cubic trigonometric B-spline (CTBS) DQMs for approximate solutions of coupled two-dimensional Burgers' equations' weighting coefficients (WCs) of DQM are calculated by using the modified CTBS functions as test functions which are different from the conventional technique of Lagrange interpolation [27]. Some well-known test problems are worked out to inspect the correctness and competence of the planned approach. The techniques lead to correct results with insignificant L_∞, RMS and L_2 errors.

2. Differential Quadrature Method

Recently, DQMs have become popular for solving nonlinear partial differential equations (PDEs) arising in nonlinear phenomena. DQMs discretize the first and second derivatives over 1D domain $\Omega = [\alpha, \beta]$ as follows:

Table 1: Coefficients of CTBSs and their derivatives at knots x_j.

x	x_{j-2}	x_{j-1}	x_j	x_{j+1}	x_{j+2}
$B_j(x)$	0	α_1	α_2	α_1	0
$B_j'(x)$	0	α_3	0	α_4	0
$B_j''(x)$	0	α_5	α_6	α_5	0

$$u_x(x_i, t) = \sum_{j=1}^{N} \alpha_{ij}^{(1)} u(x_j, t), \quad i = 1, 2, \ldots, N, \tag{4}$$

$$u_{xx}(x_i, t) = \sum_{j=1}^{N} \alpha_{ij}^{(2)} u(x_j, t), \quad j = 1, 2, \ldots, N, \tag{5}$$

where $\alpha_{ij}^{(1)}$ and $\alpha_{ij}^{(2)}$ are unknown coefficients weighting the first and second derivatives, respectively, and $x_i, i = 1, 2, \ldots, N$, are uniform grids as well as nonuniform grids that exist in the domain. Bellman et al. [28] introduced two approaches to calculate WCs. Furthermore, to modify Bellman's approaches for finding WCs, many efforts have been carried out such as Lagrange interpolated cosine functions, spline functions, Legendre polynomials, Lagrange interpolation polynomials, and radial basis functions (see [19, 29–35] and the references therein) to determine these coefficients. In this study, we determine WCs with the use of CTBS functions after some modifications.

2.1. Cubic Trigonometric B-Spline Functions. In this section, we mesh the solution domain $\alpha \le x \le \beta$ into N subintervals $[x_i, x_{i+1}], i = 0, 1, \ldots, N-1$ with the help of knots x_i such that $\alpha = x_0 < x_1, \ldots, < x_N = \beta$ is a uniform partition with step length $a = x_{i+1} - x_i = (\beta - \alpha)/N, i = 0, 1, \ldots, N-1$.

Now, the piecewise CTBS basis functions $\tau B_j(x)$ over the uniform mesh are defined as follows [36, 37]:

$$B_j(x) = \frac{1}{\omega} \begin{cases} r^3(x_i), & x \in [x_{i-2}, x_{i-1}), \\ r(x_i)[r(x_i)s(x_{i+2}) + s(x_{i+3})r(x_{i+1})] + s(x_{i+4})r^2(x_{i+1}), & x \in [x_{i-1}, x_i), \\ s(x_{i+4})[r(x_{i+1})s(x_{i+3}) + s(x_{i+4})r(x_{i+2})] + r(x_i)s^2(x_{i+3}), & x \in [x_i, x_{i+1}), \\ s^3(x_{i+4}), & x \in [x_{i+1}, x_{i+2}), \\ 0, & \text{otherwise}, \end{cases} \tag{6}$$

where

$$r(x_i) = \sin\left(\frac{x - x_i}{2}\right),$$
$$s(x_i) = \sin\left(\frac{x_i - x}{2}\right), \tag{7}$$
$$\omega = \sin\left(\frac{a}{2}\right)\sin(a)\sin\left(\frac{3a}{2}\right).$$

The basis over the region $\alpha \le x \le \beta$ is formed by the set

$$\{B_{-1}(x), B_0(x), \ldots, B_N(x), B_{N+1}(x)\}. \tag{8}$$

Every CTBS covers four elements. Now, with the help of Table 1, we have tabulated the values of $B_j(x)$ and its derivatives as follows:

$$\alpha_1 = \frac{\sin^2(a/2)}{\sin(a)\sin(3a/2)},$$
$$\alpha_2 = \frac{2}{1 + 2\cos(a)},$$
$$\alpha_3 = \frac{-3}{4\sin(3a/2)},$$
$$\alpha_4 = \frac{3}{4\sin(3a/2)}, \tag{9}$$
$$\alpha_5 = \frac{3[1 + 3\cos(a)]}{16\sin^2(a/2)[2\cos(a/2) + \cos(3a/2)]},$$
$$\alpha_6 = \frac{3\cos^2(a/2)}{2\sin^2(a/2)[1 + 2\cos(a)]}.$$

2.2. Modified Cubic Trigonometric B-Spline Functions. In this work, we compute WCs of DQM with the help of modified CTBS function defined in (6) as follows:

$$\begin{cases} \widetilde{B}_0(x) = B_0(x) + 2B_{-1}(x), & j = 0, \\ \widetilde{B}_1(x) = B_1(x) - B_{-1}(x), & j = 1, \\ \widetilde{B}_j(x) = B_j(x), & j = 2, 3, \ldots, N-2, \\ \widetilde{B}_{N-1}(x) = B_{N-1}(x) - B_{N+1}(x), & j = N-1, \\ \widetilde{B}_N(x) = B_N(x) + 2B_{N+1}(x), & j = N. \end{cases} \tag{10}$$

It is worth mentioning that the modified functions $\{\tau\widetilde{B}_j(x)\}$, $j = 0, 1, \ldots, N$, are linearly independent. On the solution domain $[\alpha, \beta]$, these functions create a family of basis functions.

2.3. Weighting Coefficients for Modified Cubic Trigonometric B-Spline Differential Quadrature Method. Now, substitute the modified functions $\{\tau\widetilde{B}_j(x)\}$, for $j = 0, 1, \ldots, N$, into equation (4). The matrix form of the equation is as follows:

$$\begin{cases} Aw_0^{(1)} = B_0 \\ Aw_1^{(1)} = B_1 \\ \vdots \\ Aw_{N-1}^{(1)} = B_{N-1} \\ Aw_N^{(1)} = B_N, \end{cases} \tag{11}$$

where A is $(N+1) \times (N+1)$ coefficient matrix:

$$A = \begin{bmatrix} \alpha_2 + 2\alpha_1 & \alpha_1 & & & & & \\ 0 & \alpha_2 & \alpha_1 & & & & \\ & \alpha_1 & \alpha_2 & \alpha_1 & & & \\ & & \cdot & \cdot & \cdot & & \\ & & & \cdot & \cdot & \cdot & \\ & & & & \cdot & \cdot & \cdot \\ & & & & \alpha_1 & \alpha_2 & 0 \\ & & & & & \alpha_1 & \alpha_2 + 2\alpha_1 \end{bmatrix}. \tag{12}$$

The matrix $w_k^{(1)} = [\alpha_{k0}^{(1)}, \alpha_{k1}^{(1)}, \ldots, \alpha_{kN}^{(1)}]^T$, for $k = 0, 1, \ldots, N$, and B_k at x_k, for $k = 0, 1, \ldots, N$, are as follows:

$$B_0 = \begin{bmatrix} -2\alpha_4 \\ 2\alpha_4 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}, B_1 = \begin{bmatrix} -\alpha_4 \\ 0 \\ \alpha_4 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}, \ldots, B_{N-1} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ -\alpha_4 \\ 0 \\ \alpha_4 \end{bmatrix}, B_N = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ -2\alpha_4 \\ 2\alpha_4 \end{bmatrix}. \tag{13}$$

Furthermore, with the help of Thomas algorithm WCs, $\alpha_{ij}^{(1)}$ are achieved as solutions of tridiagonal systems of equation (11). Similarly, with the help of the above method, it is easy to calculate second-order WCs $\beta_{ij}^{(2)}$.

2.4. Two-Dimensional Modified Cubic Trigonometric B-Spline Differential Quadrature Method. In order to apply this method to 2D nonlinear problems, first of all, decompose the domain $\Omega = \{(x, y)\colon \alpha_1 \le x \le \beta_1; \alpha_2 \le y \le \beta_2\}(\pi/6)$ as $\Omega^1 = \{(x_i, y_j), i = 1, 2, \ldots, N; j = 1, 2, \ldots, M\}$ by adopting step length $\Delta x = x_i - x_{i-1}$ and $\Delta y = y_j - y_{j-1}$ in x and y direction, respectively. This modified technique helps to estimate the 1$^{\text{st}}$ order partial derivatives of $u(x, y, t)$ at a point as follows:

$$u_x(x_i, y_j, t) = \sum_{k=1}^{N} \alpha_{ik}^{(1)} u(x_k, y_j, t), \quad i = 1, 2, \ldots, N, \tag{14}$$

$$u_y(x_i, y_j, t) = \sum_{k=1}^{M} \beta_{jk}^{(1)} u(x_i, y_k, t), \quad j = 1, 2, \ldots, M, \tag{15}$$

where $\alpha_{ij}^{(1)}$ is WCs for the 1st order derivatives w.r.t. x. Similarly, $\beta_{jk}^{(1)}$ are coefficients w.r.t. y.

In order to compute the 2D WCs, we can define the functions $\{\tau\widetilde{B}_j(y)\}$, $j=0,1,\ldots,N$, as in equation (10). Furthermore, take the test functions as $T_{ij}(x,\ y)=\tau\widetilde{B}_i(x)\tau\widetilde{B}_j(y)$. Now, with the help of the axioms of vector space and substituting the value of $T_{ij}(x,y)$ into equations (14) and (15), we have

$$\begin{aligned}\widetilde{B}_j'(x_i) &= \sum_{k=1}^{N}\alpha_{ik}^{(1)}\widetilde{B}_j(x_k), \quad j,i=1,2,\ldots,N,\\ \widetilde{B}_j'(y_i) &= \sum_{k=1}^{M}\beta_{ik}^{(1)}\widetilde{B}_j(y_k), \quad j,i=1,2,\ldots,M.\end{aligned} \tag{16}$$

Furthermore, applying the well-known algorithm "Thomas algorithm" and proceeding with the same methods as in the case of equation (11), the solutions of the systems give the value of $\alpha_{ij}^{(1)}$ and $\beta_{jk}^{(1)}$. In 2D case, the WCs in higher-order derivatives can be considered as follows:

$$\begin{aligned}\alpha_{ij}^{(r)} &= r\left[\alpha_{ij}^{(1)}\alpha_{ii}^{(r-1)} - \frac{\alpha_{ij}^{(r-1)}}{x_i-x_j}\right], \quad \text{for } i,j=1,2,\ldots,N; i\neq j; r=2,3,\ldots,N-1,\\ \alpha_{ii}^{(r)} &= -\sum_{j=1,j\neq i}^{N}\alpha_{ij}^{(r)}, \quad \text{for } i=j,\\ \beta_{ij}^{(r)} &= r\left[\beta_{ij}^{(1)}\beta_{ii}^{(r-1)} - \frac{\beta_{ij}^{(r-1)}}{y_i-y_j}\right], \quad \text{for } i,j=1,2,\ldots,N; i\neq j; r=2,3,\ldots,N-1,\\ \beta_{ii}^{(r)} &= -\sum_{j=1,j\neq i}^{M}\beta_{ij}^{(r)}, \quad \text{for } i=j,\end{aligned} \tag{17}$$

where $\alpha_{ij}^{(r)}$ and $\beta_{ij}^{(r)}$ are WCs for r^{th} order partial derivatives w.r.t. x and y, respectively.

3. Numerical Algorithm for Two-Dimensional Coupled Burgers' Equation

In this section, the numerical algorithm is developed in the following sections.

3.1. Semidiscretization in Time. Applying forward difference on time derivatives and weighted average on spatial derivatives, we have

$$\frac{u(x,y,t^{n+1})-u(x,y,t^n)}{\Delta t} = \frac{1}{\mathrm{Re}}\left[\theta\nabla^2 u(x,y,t^{n+1}) + (1+\theta)\nabla^2 u(x,y,t^n)\right] - (uu_x)^{n+1} - (uu_y)^{n+1}, \quad n=0,1,\ldots,K, \tag{18a}$$

$$\frac{v(x,y,t^{n+1})-v(x,y,t^n)}{\Delta t} = \frac{1}{\mathrm{Re}}\left[\theta\nabla^2 v(x,y,t^{n+1}) + (1+\theta)\nabla^2 v(x,y,t^n)\right] - (vv_x)^{n+1} - (vv_y)^{n+1}, \quad n=0,1,\ldots,K, \tag{18b}$$

where $u(x, y, t^{n+1}) = u(x, y, t + n\Delta t)$, $v(x, y, t^{n+1}) = v(x, y, t + n\Delta t)$, Δt step length in time direction, and $0 \leq \theta \leq 1$. The nonlinear term is linearized in the following manner:

$$\begin{aligned} (uu_x)^{n+1} &= u^n u_x^n, \\ (vu_y)^{n+1} &= v^n u_y^n, \end{aligned} \tag{19a}$$

$$\begin{aligned} (uv_x)^{n+1} &= u^n v_x^n, \\ (vv_y)^{n+1} &= v^n v_y^n, \end{aligned} \tag{19b}$$

with ICs

$$u(x, y, t^0) = \psi_1(x, y), \quad (x, y) \in [\alpha, \beta] \times [\gamma, \delta], \tag{20a}$$

$$v(x, y, t^0) = \psi_2(x, y), \quad (x, y) \in [\alpha, \beta] \times [\gamma, \delta], \tag{20b}$$

and prescribed BCs (3).

After simplification, equations (18a) and (18b) can be written as follows:

$$u^{n+1} - \frac{\Delta t}{\mathrm{Re}} \theta \nabla^2 u^{n+1} = (1 - \Delta t u_x^n) u^n + \Delta t (1 - \theta) \Delta^2 u^n - \Delta t v^n u_y^n, \tag{21a}$$

$$v^{n+1} - \frac{\Delta t}{\mathrm{Re}} \theta \nabla^2 v^{n+1} = (1 - \Delta t v_y^n) v^n + \Delta t (1 - \theta) \Delta^2 v^n - \Delta t u^n v_x^n, \tag{21b}$$

which is a system of second-order differential equations, where $u^{n+1}(x, y) = u(x, y, t^{n+1})$ and equations (21a) and (21b) are a system of second-order differential equations.

3.2. Fully Discretization in Space. In this section, spatial derivatives that occur in equations (21a) and (21b) are discretized by modified CTBS DQM over the given domain. After spatial discretization, equations (21a) and (21b) convert into a system of linear equations for each n in the following form:

$$u_{ij}^{n+1} - \frac{\Delta t\theta}{\mathrm{Re}} \left(\sum_{k=1}^{N} \alpha_{ik}^{(2)} u_{kj}^{n+1} + \sum_{k=1}^{M} \beta_{jk}^{(2)} u_{ik}^{n+1} \right) = \left(1 - \Delta t \sum_{k=1}^{N} \beta_{ik}^{(1)} u_{kj}^{n} \right) u_{ij}^{n} + \Delta t (1 - \theta) \left(\sum_{k=1}^{N} \alpha_{ik}^{(2)} u_{kj}^{n} + \sum_{k=1}^{M} \beta_{jk}^{(2)} u_{ik}^{n} \right) - \Delta t u_{ij}^{n} \sum_{k=1}^{M} \beta_{jk}^{(1)} u_{ik}^{n}, \tag{22a}$$

$$v_{ij}^{n+1} - \frac{\Delta t\theta}{\mathrm{Re}} \left(\sum_{k=1}^{N} \alpha_{ik}^{(2)} v_{kj}^{n+1} + \sum_{k=1}^{M} \beta_{jk}^{(2)} v_{ik}^{n+1} \right) = \left(1 - \Delta t \sum_{k=1}^{N} \alpha_{ik}^{(1)} v_{kj}^{n} \right) v_{ij}^{n} + \Delta t (1 - \theta) \left(\sum_{k=1}^{N} \alpha_{ik}^{(2)} v_{kj}^{n} + \sum_{k=1}^{M} \beta_{jk}^{(2)} v_{ik}^{n} \right) - \Delta t v_{ij}^{n} \sum_{k=1}^{M} \alpha_{jk}^{(1)} v_{ik}^{n}, \tag{22b}$$

where $u_{ij}^n = u^n(x_i, y_i)$ and $\alpha_{ik}^{(2)}$ and $\beta_{ik}^{(2)}$ are WCs of 2nd order partial derivatives w.r.t. x and y.

3.3. Implementation of Dirichlet Boundary Conditions. The Dirichlet BCs given in equation (3) as $(\alpha, y, t) = h_1(y, t), u(\beta, y, t) = h_2(y, t)$, $u(x, \gamma, t) = h_3(x, t)$, and $u(x, \delta, t) = h_4(x, t)$ can be implemented directly as follows:

$$u_{1,j} = h_1(y_j, t), u_{N,j} = h_2(y_j, t), \quad y_j \in [\gamma, \delta], j = 1, 2, \ldots, M, \tag{23a}$$

$$u_{i,1} = h_3(x_i, t), u_{i,M} = h_4(x_i, t), \quad x_i \in [\alpha, \beta], i = 1, 2, \ldots, N. \tag{23b}$$

As a result of applying the BCs on systems (23a) and (23b), the system can be written as follows:

$$u_{ij}^{n+1} - \frac{\Delta t\theta}{\mathrm{Re}} \left(\sum_{k=1}^{N} \alpha_{jk}^{(2)} u_{kj}^{n+1} + \sum_{k=1}^{M} \beta_{jk}^{(2)} u_{ik}^{n+1} \right) = S_{ij}, \tag{24a}$$

$$\begin{aligned} v_{ij}^{n+1} - \frac{\Delta t\theta}{\mathrm{Re}} \left(\sum_{k=1}^{N} \alpha_{ik}^{(2)} u_{kj}^{n+1} + \sum_{k=1}^{M} \beta_{jk}^{(2)} v_{ik}^{n+1} \right) &= T_{ij}, \\ S_{ij} = \left(1 - \Delta t \sum_{k=1}^{N} \alpha_{jk}^{(1)} u_{kj}^{n} \right) u_{ij}^{n} + \Delta t (1 - \theta) \left(\sum_{k=1}^{N} \alpha_{ik}^{(2)} u_{kj}^{n} + \sum_{k=1}^{M} \beta_{jk}^{(2)} u_{ik}^{n} \right) &- \Delta t u_{ij}^{n} \sum_{k=1}^{M} \beta_{jk}^{(1)} u_{ik}^{n}, \\ T_{ij} = \left(1 - \Delta t \sum_{k=1}^{M} \beta_{ik}^{(1)} v_{kj}^{n} \right) v_{ij}^{n} + \Delta t (1 - \theta) \left(\sum_{k=1}^{N} \alpha_{ik}^{(2)} v_{kj}^{n} + \sum_{k=1}^{M} \beta_{jk}^{(2)} v_{ik}^{n} \right) &- \Delta t u_{ij}^{n} \sum_{k=1}^{N} \alpha_{ik}^{(1)} v_{kj}^{n}. \end{aligned} \tag{24b}$$

The system of equations (18a) and (18b) is a Lyapunov system of equations of the form

$$[A_1][U]+[U][B_1]+[C_1]=0, \tag{25a}$$

$$[A_2][U]+[U][B_2]+[C_2]=0, \tag{25b}$$

where

$$[A_1]=\begin{bmatrix}1 & & & & \\ & 1 & & & \\ & & \ddots & & \\ & & & 1 & \\ & & & & 1\end{bmatrix}_{(N-2)\times(N-2)} - \frac{\Delta t\theta}{\mathrm{Re}}\begin{bmatrix}\alpha_{22}^{(2)} & \alpha_{23}^{(2)} & \cdots & \alpha_{2(N-2)}^{(2)} & \alpha_{2(N-1)}^{(2)} \\ \alpha_{32}^{(2)} & \alpha_{33}^{(2)} & \cdots & \alpha_{3(N-2)}^{(2)} & \alpha_{3(N-1)}^{(2)} \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ \alpha_{(N-2)2}^{(2)} & \alpha_{(N-2)3}^{(2)} & \cdots & \alpha_{(N-2)(N-2)}^{(2)} & \alpha_{(N-2)(N-1)}^{(2)} \\ \alpha_{(N-1)2}^{(2)} & \alpha_{(N-2)3}^{(2)} & \cdots & \alpha_{(N-1)(N-2)}^{(2)} & \alpha_{(N-1)(N-1)}^{(2)}\end{bmatrix}_{(N-2)\times(N-2)},$$
$$[B_1]=-\frac{\Delta t\theta}{R}\begin{bmatrix}\beta_{22}^{(2)} & \beta_{23}^{(2)} & \cdots & \beta_{2(M-2)}^{(2)} & \beta_{2(M-1)}^{(2)} \\ \beta_{32}^{(2)} & \beta_{33}^{(2)} & \cdots & \beta_{3(M-2)}^{(2)} & \beta_{3(M-1)}^{(2)} \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ \beta_{(M-2)2}^{(2)} & \beta_{(M-2)3}^{(2)} & \cdots & \beta_{(M-2)(M-2)}^{(2)} & \beta_{(M-2)(M-1)}^{(2)} \\ \beta_{(M-1)2}^{(2)} & \beta_{(M-2)3}^{(2)} & \cdots & \beta_{(M-1)(M-2)}^{(2)} & \beta_{(M-1)(M-1)}^{(2)}\end{bmatrix}_{(M-2)\times(M-2)},$$
$$[C_1]=\begin{bmatrix}S_{22} & S_{23} & \cdots & S_{2(M-2)} & S_{2(M-1)} \\ S_{32} & S_{33} & \cdots & S_{3(M-2)} & S_{3(M-1)} \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ S_{(N-2)2} & S_{(N-2)3} & \cdots & S_{(N-2)(M-2)} & S_{(N-2)(M-1)} \\ S_{(N-1)2} & S_{(N-1)3} & \cdots & S_{(N-1)(M-2)} & S_{(N-1)(M-1)}\end{bmatrix}_{(N-2)\times(M-2)}. \tag{26}$$

Also,

$$[A_2] = \begin{bmatrix} 1 & & & & \\ & 1 & & & \\ & & \ddots & & \\ & & & 1 & \\ & & & & 1 \end{bmatrix}_{(N-2)\times(N-2)} - \frac{\Delta t\theta}{\mathrm{Re}} \begin{bmatrix} \alpha_{22}^{(2)} & \alpha_{23}^{(2)} & \cdots & \alpha_{2(N-2)}^{(2)} & \alpha_{2(N-1)}^{(2)} \\ \alpha_{32}^{(2)} & \alpha_{33}^{(2)} & \cdots & \alpha_{3(N-2)}^{(2)} & \alpha_{3(N-1)}^{(2)} \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ \alpha_{(N-2)2}^{(2)} & \alpha_{(N-2)3}^{(2)} & \cdots & \alpha_{(N-2)(N-2)}^{(2)} & \alpha_{(N-2)(N-1)}^{(2)} \\ \alpha_{(N-1)2}^{(2)} & \alpha_{(N-2)3}^{(2)} & \cdots & \alpha_{(N-1)(N-2)}^{(2)} & \alpha_{(N-1)(N-1)}^{(2)} \end{bmatrix}_{(N-2)\times(N-2)},$$

$$[V] = \begin{bmatrix} v_{22}^{n+1} & v_{23}^{n+1} & \cdots & v_{2(M-2)}^{n+1} & v_{2(M-1)}^{n+1} \\ v_{32}^{n+1} & v_{33}^{n+1} & \cdots & v_{3(M-2)}^{n+1} & v_{3(M-1)}^{n+1} \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ v_{(N-2)2}^{n+1} & v_{(N-2)3}^{n+1} & \cdots & b_{(N-2)(M-2)}^{n+1} & b_{(N-2)(M-1)}^{n+1} \\ v_{(N-1)2}^{n+1} & b_{(N-2)3}^{n+1} & \cdots & b_{(N-1)(M-2)}^{n+1} & v_{(N-1)(M-1)}^{n+1} \end{bmatrix}_{(N-2)\times(M-2)}, \tag{27}$$

$$[B_2] = -\frac{\Delta t\theta}{R} \begin{bmatrix} \beta_{22}^{(2)} & \beta_{23}^{(2)} & \cdots & \beta_{2(M-2)}^{(2)} & \beta_{2(M-1)}^{(2)} \\ \beta_{32}^{(2)} & \beta_{33}^{(2)} & \cdots & \beta_{3(M-2)}^{(2)} & \beta_{3(M-1)}^{(2)} \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ \beta_{(M-2)2}^{(2)} & \beta_{(M-2)3}^{(2)} & \cdots & \beta_{(M-2)(M-2)}^{(2)} & \beta_{(M-2)(M-1)}^{(2)} \\ \beta_{(M-1)2}^{(2)} & \beta_{(M-2)3}^{(2)} & \cdots & \beta_{(M-1)(M-2)}^{(2)} & \beta_{(M-1)(M-1)}^{(2)} \end{bmatrix}_{(M-2)\times(M-2)},$$

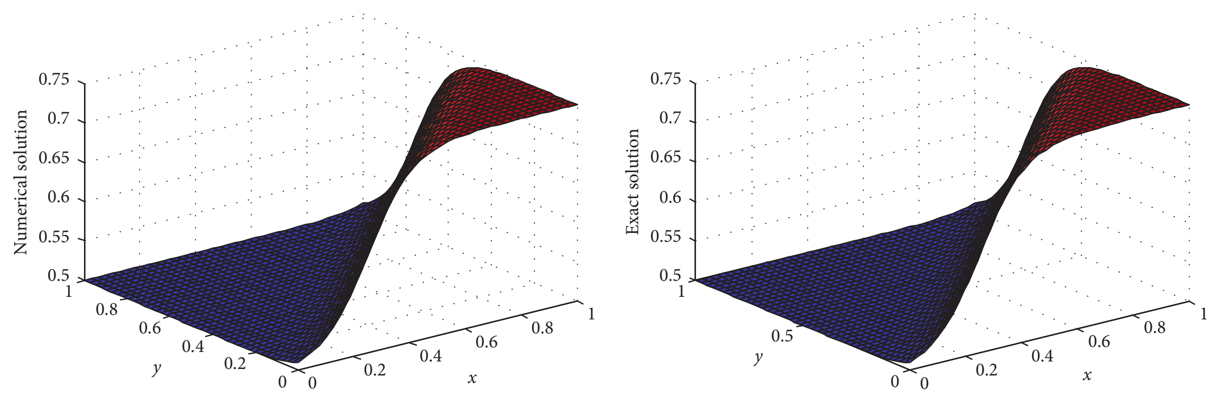
$$[C_2] = -\begin{bmatrix} T_{22} & T_{23} & \cdots & T_{2(M-2)} & T_{2(M-1)} \\ T_{32} & T_{33} & \cdots & T_{3(M-2)} & T_{3(M-1)} \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ T_{(N-2)2} & T_{(N-2)3} & \cdots & T_{(N-2)(M-2)} & T_{(N-2)(M-1)} \\ T_{(N-1)2} & T_{(N-1)3} & \cdots & T_{(N-1)(M-2)} & T_{(N-1)(M-1)} \end{bmatrix}_{(N-2)\times(M-2)}.$$

TABLE 2: L_∞, RMS, and L_2 errors of Problem 1 at different times and Reynolds number for $u(x, y, t)$ with $N = M = 40$.

T	Re = 50			Re = 100		
	L_∞	RMS	L_2	L_∞	RMS	L_2
0.1	$7.064E-04$	$1.020E-04$	$4.081E-03$	$2.287E-04$	$4.676E-05$	$1.871E-03$
0.5	$6.864E-04$	$1.102E-04$	$4.409E-03$	$2.393E-04$	$6.091E-05$	$2.437E-03$
1.0	$6.386E-04$	$1.068E-04$	$4.273E-03$	$2.424E-04$	$6.153E-05$	$2.461E-03$
1.5	$6.150E-04$	$1.031E-04$	$4.124E-03$	$2.403E-04$	$5.857E-05$	$2.343E-03$
2.0	$6.028E-04$	$9.924E-05$	$3.969E-03$	$2.348E-04$	$5.521E-05$	$2.208E-03$

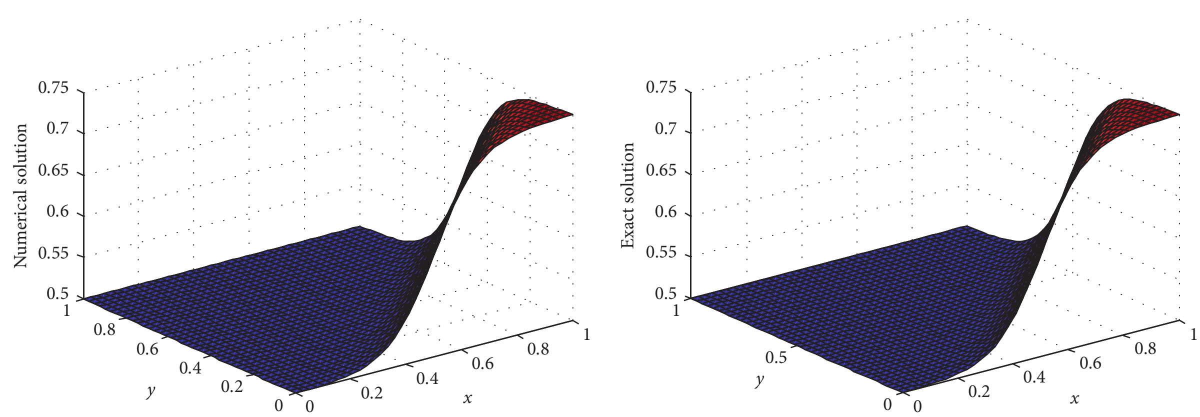
TABLE 3: L_∞, RMS, and L_2 errors of Problem 1 at different times and Reynolds number for $u(x, y, t)$ with $N = M = 40$.

T	Re = 200			Re = 400		
	L_∞	RMS	L_2	L_∞	RMS	L_2
0.1	$9.269E-04$	$1.265E-05$	$5.059E-03$	$3.269E-03$	$2.390E-04$	$3.563E-03$
0.5	$9.948E-04$	$8.723E-05$	$3.489E-03$	$3.742E-03$	$3.349E-04$	$3.768E-03$
1.0	$9.167E-04$	$1.345E-05$	$5.282E-03$	$3.910E-03$	$3.214E-04$	$3.867E-03$
1.5	$9.020E-04$	$1.367E-05$	$5.458E-03$	$4.481E-03$	$3.514E-04$	$3.989E-03$
2.0	$9.037E-04$	$1.332E-05$	$5.321E-03$	$4.891E-03$	$3.890E-04$	$4.891E-03$



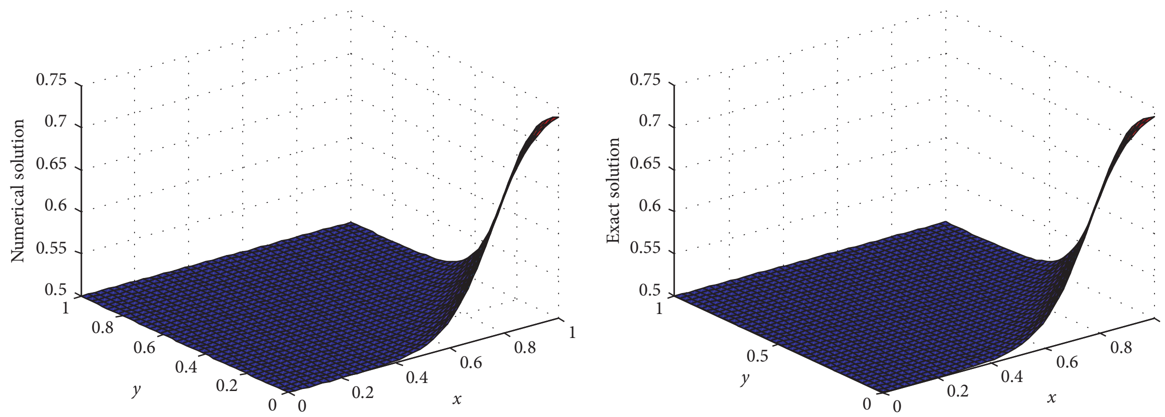
(a) (b)

FIGURE 1: NSs and ESs of $u(x, y, t)$ in 3D form for $T = 1.0$ of Problem 1.



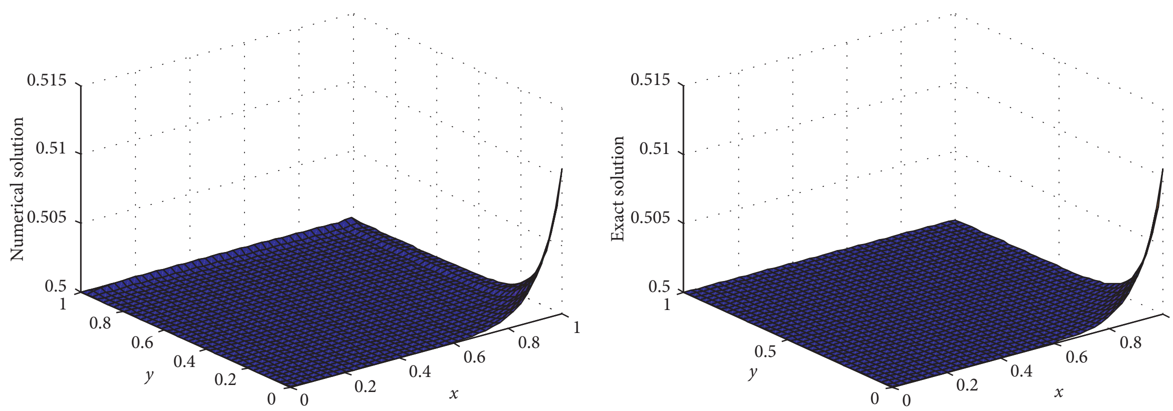
(a) (b)

FIGURE 2: NSs and ESs of $u(x, y, t)$ in 3D form for $T = 2.0$ of Problem 1.



(a) (b)

Figure 3: NSs and ESs of $u(x, y, t)$ in 3D form for $T = 3.0$ of Problem 1.



(a) (b)

Figure 4: NSs and ESs of $u(x, y, t)$ in 3D form for $T = 5.0$ of Problem 1.

Table 4: Maximum abosulte error L_∞ of the problems at different nodes for $u(x, y, t)$.

T	Problem 2 Re = 100			Problem 1 Re = 100		
	$N = M = 10$	$N = M = 20$	$N = M = 40$	$N = M = 10$	$N = M = 20$	$N = M = 40$
0.1	$2.923E-04$	$5.569E-05$	$2.950E05$	$0.747E-03$	$0.326E-03$	$2.287E-04$
0.5	$4.958E-04$	$7.518E-05$	$2.421E-05$	$0.381E-03$	$0.103E-03$	$2.393E-04$
1.0	$3.932E-04$	$8.167E-05$	$1.387E-05$	$0.444E-03$	$9.972E-04$	$2.424E-04$

Equations (19a) and (19b) are the coupled Lyapunov system, first solved for $n = 0$ and then solved simultaneously for $n = 1, 2, \ldots, K$ by developing code in MATLAB 7.

4. Numerical Experiments and Discussion

Under this heading, to check the correctness and competence of the algorithm modified CTBS DQM, two test problems have been considered, which are available in the literature. All the computation work is conducted by using MATLAB 7.0. The following formulas are used for computing maximum absolute error L_∞, root mean square (RMS) error, and L_2 error, respectively:

$$L_\infty = \max_{\substack{1\le i\le N \\ 1\le j\le M}} \left|e_{ij}\right|,$$

$$\mathrm{RMS} = \left[\frac{1}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M}\left|\left(e_{ij}\right)\right|^2\right]^{1/2}, \tag{28}$$

$$L_2 = \left[\sum_{i=1}^{N}\sum_{j=1}^{M}\left|\left(e_{ij}\right)\right|^2\right]^{1/2},$$

Table 5: L_∞, RMS, and L_2 errors of Problem 2 at different times and Reynolds number for $u(x, y, t)$ with $N = M = 40$.

T	Re = 50			Re = 100		
	L_∞	RMS	L_2	L_∞	RMS	L_2
0.1	$1.123E-04$	$2.437E-05$	$4.874E-04$	$2.950E05$	$8.381E-06$	$1.676E-05$
0.5	$7.569E-05$	$3.865E-05$	$7.730E-04$	$2.421E-05$	$1.759E-05$	$3.519E-04$
1.0	$4.621E-05$	$3.983E-05$	$7.966E-04$	$1.892E-05$	$2.120E-05$	$4.240E-04$
1.5	$2.818E-05$	$3.509E-05$	$7.018E-04$	$1.478E-05$	$2.227E-05$	$4.454E-04$
2.0	$1.271E-05$	$2.949E-05$	$5.898E-04$	$1.154E-05$	$2.190E-05$	$4.380E-04$

Table 6: L_∞, RMS, and L_2 errors of Problem 2 at different times and Reynolds number for $u(x, y, t)$ with $N = M = 40$.

T	Re = 200			Re = 400		
	L_∞	RMS	L_2	L_∞	RMS	L_2
0.1	$7.557E-06$	$2.561E-06$	$5.122E-05$	$1.915E-06$	$3.575E-07$	$1.422E-05$
0.5	$6.847E-06$	$6.965E-06$	$1.393E-04$	$4.103E-06$	$8.296E-07$	$3.318E-05$
1.0	$6.637E-06$	$9.197E-06$	$1.839E-04$	$4.844E-06$	$1.089E-06$	$4.359E-05$
1.5	$6.944E-06$	$1.040E-05$	$2.079E-04$	$5.187E-06$	$1.262E-06$	$4.985E-05$
2.0	$6.938E-06$	$1.112E-05$	$2.224E-04$	$5.378E-06$	$1.353E-06$	$5.413E-05$

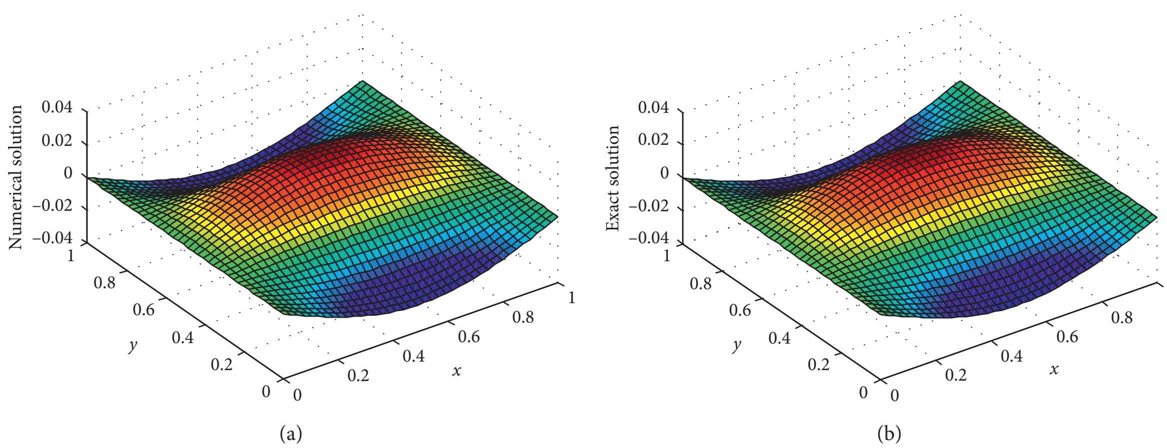


Figure 5: NSs and ESs of $u(x, y, t)$ in 3D form for $T = 2.0$ of Problem 2.

where u_{ij} and $\overline{u_{ij}}$ are approximate and exact solutions, respectively, and $e_{ij} = u_{ij} - \overline{u_{ij}}$.

Problem 1. As the first problem, consider 2D Burgers' equations (1a) and (1b). The exact solutions over the domain $D = \{(x, y): 0 \le x \le 1, 0 \le y \le 1\}$ is generated by the Hopf-Cole transformation [12, 15, 17] and obtained as

$$u(x, y, t) = \frac{3}{4} - \frac{1}{4\left[1 + e^{\mathrm{Re}(4y-4x-t)/32}\right]}, \quad (29)$$

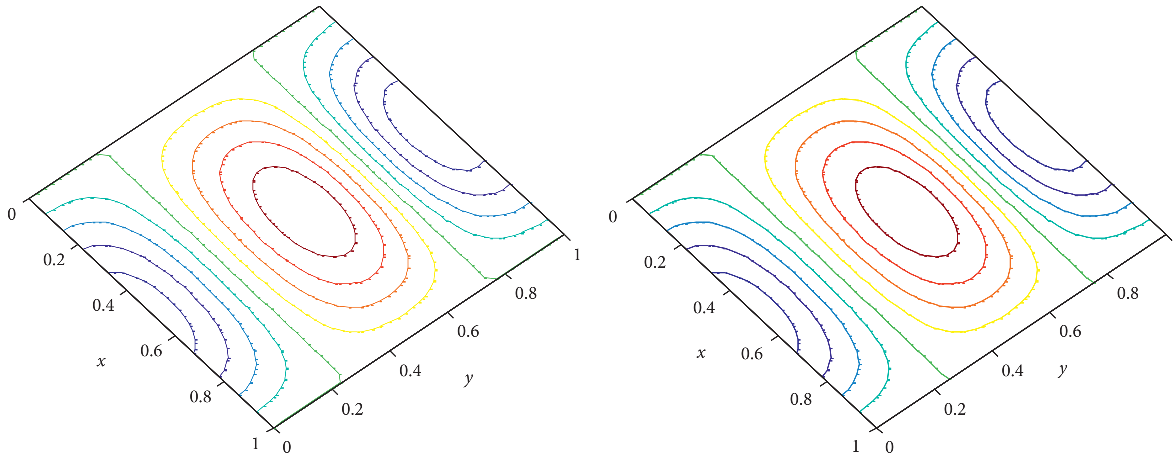
$$v(x, y, t) = \frac{3}{4} + \frac{1}{4\left[1 + e^{\mathrm{Re}(4y-4x-t)/32}\right]}. \quad (30)$$

ICs and BCs are taken from exact solutions (29) and (30). The numerical results are shown with the help of Tables 2 and 3 and Figures 1–4 in form of errors, three-dimension, and contour plots. Convection prevails the flow which causes the errors become larger and larger as we increase the value of Re. L_∞ is smaller than [15] for $T = 2.0, \mathrm{Re} = 100$ with less grid points $N = M = 40$. The figures show that exact solutions and numerical solutions are well consistent in three-dimensional and contour form. Table 4 shows that, as we increase the values of M and N, the absolute errors decrease which shows the convergence of the method.

Problem 2. Consider 2D Burgers' equations (1a) and (1b) over the computational domain $D = \{(x, y): 0 \le x \le 1, 0 \le y \le 1\}$ with the ICs [12],

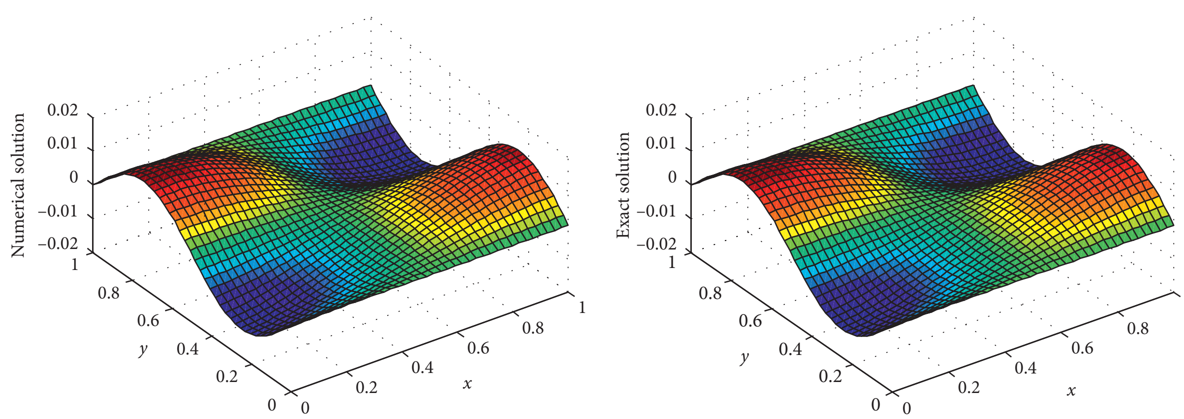
$$\begin{aligned} u(x, y, 0) &= \frac{-4\pi \cos(2\pi x)\sin(\pi y)}{\mathrm{Re}[2 + \sin(2\pi x)\sin(\pi y)]}, \quad (x, y) \in D, \\ v(x, y, 0) &= \frac{-2\pi \sin(2\pi x)\cos(\pi y)}{\mathrm{Re}[2 + \sin(2\pi x)\sin(\pi y)]}, \quad (x, y) \in D, \end{aligned} \quad (31)$$

and BCs,



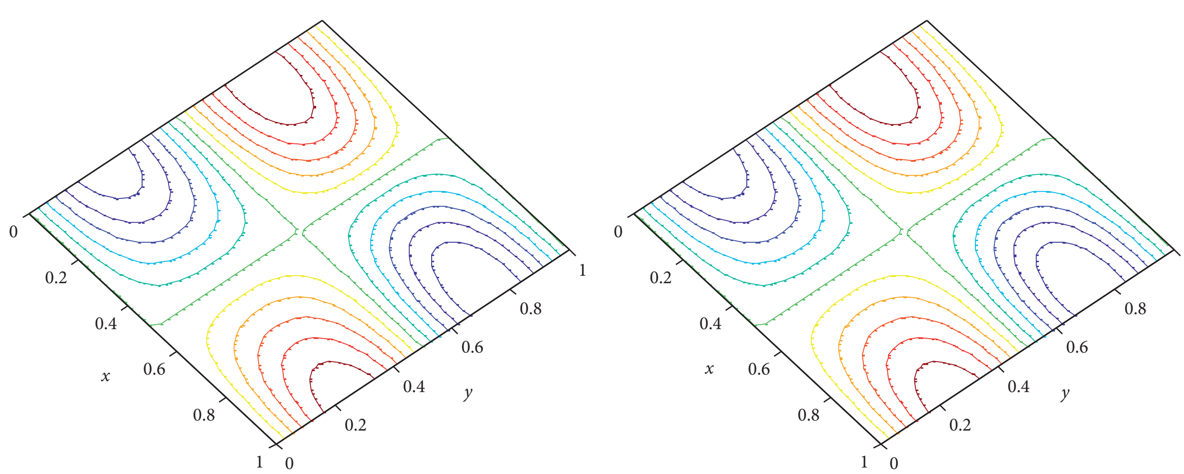
(a) (b)

Figure 6: NSs and ESs of $u(x, y, t)$ in contour form for $T = 2.0$ of Problem 2.



(a) (b)

Figure 7: NSs and ESs of $v(x, y, t)$ in 3D form for $T = 2.0$ of Problem 2.



(a) (b)

Figure 8: NSs and ESs of $v(x, y, t)$ in contour form for $T = 2.0$ of Problem 2.

$$u(0,y,t) = -2\pi\ \exp\left(\frac{-5\pi^2 t}{\mathrm{Re}}\right)\sin\frac{(\pi y)}{\mathrm{Re}},$$

$$u(1,y,t) = -2\pi\ \exp\left(\frac{-5\pi^2 t}{\mathrm{Re}}\right)\sin\frac{(\pi y)}{\mathrm{Re}},$$

$$v(0,y,t) = 0,\ v(1,y,t) = 0,\quad 0 \le y \le 1, t \ge 0,$$

$$u(x,0,t) = 0,\ u(x,1,t) = 0,$$

$$v(x,0,t) = -\pi\ \exp\left(\frac{-5\pi^2 t}{\mathrm{Re}}\right)\sin\frac{(2\pi x)}{\mathrm{Re}},$$

$$v(x,0.5,t) = \pi\ \exp\left(\frac{-5\pi^2 t}{\mathrm{Re}}\right)\sin\frac{(2\pi x)}{\mathrm{Re}},\quad 0 \le x \le 1, t \ge 0. \tag{32}$$

The exact solutions of the problem are given by

$$u(x,y,t) = \frac{-4\pi\ \exp\left(-5\pi^2 t/\mathrm{Re}\right)\cos(2\pi x)\sin(\pi y)}{\mathrm{Re}\left[2 + \exp\left(-5\pi^2 t/\mathrm{Re}\right)\sin(2\pi x)\sin(\pi y)\right]},$$

$$v(x,y,t) = \frac{-2\pi\ \exp\left(-5\pi^2 t/\mathrm{Re}\right)\sin(2\pi x)\cos(\pi y)}{\mathrm{Re}\left[2 + \exp\left(-5\pi^2 t/\mathrm{Re}\right)\sin(2\pi x)\sin(\pi y)\right]}. \tag{33}$$

Tables 5 and 6 show L_∞, RMS, and L_2 errors for different values of Re and time, while Figures 5–8 show a comparison of numerical and exact solution in three-dimensional form. Convection prevails the flow which causes the errors become larger and larger as we increase the value of Re. The figures show that exact and numerical solutions are well consistent in three-dimensional and contour form. Table 4 shows that, as we increase the values of M and N, the absolute errors decrease which shows the convergence of the method.

5. Conclusion

In this study, a modified CTBS DQM and a new algorithm to reveal the computational modeling of 2D coupled Burgers' equations are developed. The proposed algorithm is tested on two benchmark problems appearing in the literature. The main results of this study are summarized as follows:

(i) A different technique using modified CTBS functions is presented to determine the WCs of 2D DQM than Lagrange interpolation traditional technique [22].

(ii) CTBS DQ algorithm proposed in [33] has extended for 2D problems in different forms, and it has concluded the algorithm worked nicely for the same problems.

(iii) The developed algorithm is better than the DQ algorithms proposed in [31, 32, 34] due to more smoothness of CTBS functions.

(iv) The presented method leads to quite similar results to those treated in [12, 15, 17, 18] and good accuracy in the case of a small number of grid points.

(v) After some modifications, the presented method can be extended to solve 2D or higher-dimensional equations. In this way, it can be used to analyze many other biological, mechanical or physical events, such as reaction, linear diffusion, dispersion, and nonlinear convection.

Comparison of numerical solutions (NSs) and exact solutions (ESs) are given on left and right sides, respectively, in Figures [1–8] for $R = 100$ and $\Delta t = 0.001$.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] C. J. Fletcher, *Computational Techniques for Fluid Dynamics*, Springer, Berlin, Germany, 1991.

[2] J. D. Cole, "On a quasi-linear parabolic equation occurring in aerodynamics," *Quarterly of Applied Mathematics*, vol. 9, no. 3, pp. 225–236, 1951.

[3] J. M. Burgers, "A mathematical model illustrating the theory of turbulence," *Advances in Applied Mechanics*, vol. 1, pp. 171–199, 1948.

[4] P. C. Jain and D. N. Holla, "Numerical solutions of coupled Burgers' equation," *International Journal of Non-linear Mechanics*, vol. 13, no. 4, pp. 213–222, 1978.

[5] C. A. J. Fletcher, "A comparison of finite element and finite difference solutions of the one- and two-dimensional Burgers' equations," *Journal of Computational Physics*, vol. 51, no. 1, pp. 159–188, 1983.

[6] W. Y. Liao, "A fourth order finite-difference method for solving the system of two-dimensional Burgers' equations," *International Journal for Numerical Methods in Fluids*, vol. 64, pp. 565–590, 2010.

[7] F. Liu and S. Weiping, "Numerical solutions of two-dimensional Burgers' equations by lattice Boltzmann method," *Communications in Nonlinear Science and Numerical Simulation*, vol. 16, no. 1, pp. 150–157, 2011.

[8] G. Zhao, X. Yu, and R. Zhang, "The new numerical method for solving the system of two-dimensional Burgers' equations," *Computers & Mathematics with Applications*, vol. 62, no. 8, pp. 3279–3291, 2011.

[9] O. Goyon, "Multilevel schemes for solving unsteady equations," *International Journal for Numerical Methods in Fluids*, vol. 22, no. 10, pp. 937–959, 1996.

[10] F. W. Wubs and E. D. de Goede, "An explicit-implicit method for a class of time-dependent partial differential equations," *Applied Numerical Mathematics*, vol. 9, no. 2, pp. 157–181, 1992.

[11] A. Bahadir, "A fully implicit finite-difference scheme for two-dimensional Burgers' equations," *Applied Mathematics and Computation*, vol. 137, pp. 131–137, 2003.

[12] M. S. EI-Sayed and D. Kaya, "On the numerical solution of the system of two-dimensional Burgers' equations by

decomposition method," *Applied Mathematics and Computation*, vol. 158, pp. 101–109, 2004.

[13] A. N. Boules and I. J. Eick, "A spectral approximation of the two dimensional Burgers' Equation," *Indian Journal of Pure and Applied Mathematics*, vol. 34, pp. 299–309, 2003.

[14] A. H. Khater, R. S. Temsah, and M. M. Hassan, "A Chebyshev spectral collocation method for solving Burgers'-type equations," *Journal of Computational and Applied Mathematics*, vol. 222, pp. 245–251, 2008.

[15] A. Ali, S. Siraj-ul-Islam, and S. Haq, "A computational meshfree technique for the numerical solution of the two-dimensional coupled Burgers' equations," *International Journal for Computational Methods in Engineering Science and Mechanics*, vol. 10, no. 5, pp. 406–422, 2009.

[16] S. Siraj-ul-Islam, B. Šarler, R. Vertnik, and G. Kosec, "Radial basis function collocation method for the numerical solution of the two-dimensional transient nonlinear coupled Burgers' equations," *Applied Mathematical Modelling*, vol. 36, no. 3, pp. 1148–1160, 2012.

[17] R. Jiwari, "A Haar wavelet quasilinearization approach for numerical simulation of Burgers' equation," *Computer Physics Communications*, vol. 183, no. 11, pp. 2413–2423, 2012.

[18] R. C. Mittal and R. Jiwari, "A differential quadrature method for solving Burgers'-type equation," *International Journal of Numerical Methods for Heat*, vol. 22, no. 1012, pp. 880–895, 2016.

[19] R. C. Mittal and R. Jiwari, "Differential quadrature method for two-dimensional Burgers' equations," *International Journal for Computational Methods in Engineering Science and Mechanics*, vol. 10, no. 6, pp. 450–459, 2009.

[20] R. C. Mittal and R. Jiwari, "Differential quadrature method for numerical solution of coupled viscous Burgers' equations," *International Journal of Computational Methods in Engineering Science Mechanics*, vol. 13, pp. 1–5, 2012.

[21] R. Jiwari, R. C. Mittal, and K. K. Sharma, "A numerical scheme based on weighted average differential quadrature method for the numerical solution of Burgers' equation," *Applied Mathematics and Computation*, vol. 219, no. 12, pp. 6680–6691, 2013.

[22] R. C. Mittal, R. Jiwari, and K. K. Sharma, "A numerical scheme based on differential quadrature method to solve time dependent Burgers' equation," *Engineering Computations*, vol. 30, pp. 117–131, 2013.

[23] H. S. Shukla, M. Tamsir, V. K. Srivastava, and J. Kumar, "Numerical solution of two dimensional coupled viscous Burger equation using modified cubic B-spline differential quadrature method," *AIP Advances*, vol. 4, no. 11, Article ID 117134, 2014.

[24] J. Lin, W. J. Feng, S. Reutskiy, H. F. Ku, and Y. J. He, "A semi-analytical method for solving a class of time fractional partial differential equations with variable coefficients," *Applied Mathematics Letters*, vol. 112, Article ID 106112, 2021.

[25] J. Lin, Y. H. Zhang, S. Reutskiy, and W. J. Feng, "A novel meshless space-time backward substitution method and its application to nonhomogeneous advection-diffusion problems," *Applied Mathematics and Computation*, vol. 398, no. 1, Article ID 125964, 2021.

[26] Y. Zhang, J. Lin, S. Reutskiy, H. G. Sun, and W. J. Fen, "The improved backward substitution method for the simulation of time-dependent nonlinear coupled Burgers' equations," *Results in Physics*, vol. 18, Article ID 103231, 2020.

[27] C. Shu, *Differential Quadrature and its Application in Engineering*, Springer-Verlag, Britain, UK, 2000.

[28] R. Bellman, B. G. Kashef, and J. Casti, "Differential quadrature: a technique for the rapid solution of nonlinear partial differential equations," *Journal of Computational Physics*, vol. 10, no. 1, pp. 40–52, 1972.

[29] A. Korkmaz and İ. Dağ, "Shock wave simulations using sinc differential quadrature method," *Engineering Computations*, vol. 28, no. 6, pp. 654–674, 2011.

[30] A. Korkmaz and İ. Dağ, "A differential quadrature algorithm for nonlinear Schrödinger equation," *Nonlinear Dynamics*, vol. 56, no. 1-2, pp. 69–83, 2009.

[31] R. Jiwari, R. K. Gupta, and V. Kumar, "Polynomial differential quadrature method for numerical solutions of the generalized Fitzhugh-Nagumo equation with time-dependent coefficients," *Ain Shams Engineering Journal*, vol. 5, no. 4, pp. 1343–1350, 2014.

[32] R. Jiwari, S. Pandit, and R. C. Mittal, "Numerical simulation of two-dimensional sine-Gordon solitons by differential quadrature method," *Computer Physics Communications*, vol. 183, no. 3, pp. 600–616, 2012.

[33] R. Jiwari, S. Pandit, and R. C. Mittal, "A differential quadrature algorithm to solve the two dimensional linear hyperbolic telegraph equation with Dirichlet and Neumann boundary conditions," *Applied Mathematics and Computation*, vol. 218, no. 13, pp. 7279–7294, 2012.

[34] V. Kumar, R. Jiwari, and R. Kumar Gupta, "Numerical simulation of two dimensional quasilinear hyperbolic equations by polynomial differential quadrature method," *Engineering Computations*, vol. 30, no. 7, pp. 892–909, 2013.

[35] J. Lin, Y. Zhao, D. Watson, and C. S. Chen, "The radial basis function differential quadrature method with ghost points," *Mathematics and Computers in Simulation*, vol. 173, pp. 105–114, 2020.

[36] M. Abbas, A. A. Majid, A. I. M. Ismail, and A. Rashid, "The application of cubic trigonometric B-spline to the numerical solution of the hyperbolic problems," *Applied Mathematics and Computation*, vol. 239, pp. 74–88, 2014.

[37] R. Jiwari, S. Singh, and A. Kumar, "Numerical simulation to capture the pattern formation of coupled reaction-diffusion models," *Chaos, Solitons & Fractals*, vol. 103, pp. 422–439, 2017.