Scientia Iranica A (2023) 30(6), 1942–1954

Sharif University of Technology
Scientia Iranica
Transactions A: Civil Engineering
http://scientiairanica.sharif.edu

Global thin plate spline differential quadrature as a meshless numerical solution for two-dimensional viscous Burgers' equation

A.M. Behroozi and M. Vaghefi*

Department of Civil Engineering, Persian Gulf University, Shahid Mahini St., Bushehr, P.O. Box 75169, Iran.

Received 10 April 2022; received in revised form 31 August 2022; accepted 7 November 2022

KEYWORDS
Thin Plate Spline Differential Quadrature (TPS-DQM);
Burgers' equation;
Radial basis function;
Mesh-less method;
Numerical methods.

Abstract. This paper aims to present the Global Thin Plate Spline Differential Quadrature (GTPS-DQM) method to achieve a numerical solution to viscous Burgers' equation. This meshless and high-order model is introduced with the motive of diminishing computational effort and dealing with irregular geometries. A Thin Plate Spline Radial Basis Function (TPS-RBF) is used as a test function to determine coefficients of derivatives in differential quadrature. The present algorithm is applied to discretize and solve the two-dimensional Burgers' equation in both rectangular and irregular non-rectangular computational domains with randomly distributed computation nodes. To evaluate the capability of the present model, several problems with different boundary and initial conditions and Reynolds numbers are solved and the obtained results are compared with the analytical solutions and other previous numerical models. The obtained results show the higher accuracy of the present model for solving Berger's equation with fewer computational nodes than the previous models even in irregular domains.

1. Introduction

Bateman (1915) first presented Burgers' equation [1] that was later picked up by Burgers (1948) as a simple mathematical expression of turbulence phenomenon in hydrodynamics [2]. This system of nonlinear second-order equations has many applications and is applicable to the mathematical modeling of physical phenomena such as boundary layer problems, turbulence, studying shock wave formation, mass transfer, traffic problems, acoustics, and propagation in porous media [3]. This equation is defined in a dimensionless form as follows:

$$\frac{\partial u(x,t)}{\partial t}+u(x,t)\frac{\partial u(x,t)}{\partial x}+v(x,t)\frac{\partial u(x,t)}{\partial y}=\frac{1}{Re}\nabla^2 u(x,t),\quad x=(x,y)\in\Omega\subset\mathbb{R}^2. \tag{1}$$

$$\frac{\partial v(x,t)}{\partial t}+u(x,t)\frac{\partial v(x,t)}{\partial x}+v(x,t)\frac{\partial v(x,t)}{\partial y}=\frac{1}{Re}\nabla^2 v(x,t),\quad x=(x,y)\in\Omega\subset\mathbb{R}^2. \tag{2}$$

The coupled set of Eqs. (1) and (2) is solved with

*. *Corresponding author.*
E-mail addresses: am.behroozi@mehr.pgu.ac.ir (A.M. Behroozi); Vaghefi@pgu.ac.ir (M. Vaghefi)

doi: 10.24200/sci.2022.60247.6685

the following initial condition:

$$u(x,0)=u_0(x), \qquad v(x,0)=v_0(x), \qquad x\in\Omega, \quad (3)$$

and boundary conditions:

$$u(x,t)=L_1(x,t), \qquad x\in\Gamma, \quad t>0,$$

$$v(x,t)=L_2(x,t), \qquad x\in\Gamma, \quad t>0, \quad (4)$$

where $u(x,t)$ and $v(x,t)$ are the components of the velocity; u_0, v_0, L_1, and L_2 all are known functions. *Re* is the dimensionless Reynolds number that shows the ratio of inertial forces to viscous forces. Indeed, the Burgers' equation is the same as the Navier-Stokes equation in which there is no pressure gradient term and it can demonstrate the properties of this equation. Therefore, Burgers' equation is often used to evaluate the stability and accuracy of numerical methods in computational fluid dynamics [4].

Due to its non-linearity, Burgers' equation only has analytical solutions in special boundary and initial conditions and it generally needs to be solved numerically.

A mathematical transformation called HOPF-COLE was used in [5,6] to transfer Eqs. (1) and (2) from nonlinear to linear form and to present an analytical solution to this equation based on infinite series. Later, this method was extended in [7] to deal with two- and three-dimensional forms of Burgers' equation. Several numerical models have also been presented for the solution of this equation in previous research.

The isoparametric space-time finite element approach was introduced in [8]. A finite element-based algorithm with element size dependent on the solving results in the previous step was also presented in [9]. The finite difference solution was followed in [10,11]. The Generalized Boundary Elements Method (GBEM) was adopted in [12], the direct variation method in [13], the spectral element method in [14], approximating functional in [15], and the Distributed Approximating Functional (DAF) approach in [4]. The coupled viscous Burgers' equation was solved by the Differential Quadrature Method (DQM) in [16]. The differential quadrature element method as a high-order solution to Burgers' equation was employed in [17]. The cubic B-spline finite elements method was utilized in [18]. A scheme based on the polynomial differential quadrature technique for one-dimensional Burgers' equation was introduced in [19]. The order-splitting extrapolation method was adopted in [20] as a high-order solution to Burgers' equation. An iterative reproducing kernel method with a variable coefficient was presented for the numerical solution of one-dimensional fractional Burgers' equation in [21]. A mesh-free spectral method to solve the time-fractional coupled viscous Burgers' equation was utilized in [22]. The high convergence of the results was reported as the main challengeof the numerical solution of Burgers' equation at high Reynolds numbers [8,23,24]. This instability results from the formation of an inviscid boundary layer due to the domination of the convection term over the diffusion term and consequently, occurrence of a shock wave in the domain. A large number of mesh points are needed to describe the field behavior in this thin boundary layer region. This increase in the number of mesh points brings about high computational effort and reduces accuracy [15]. In addition, a majority of previous studies have been limited to the numerical solution of the Burgers' equation in a regular rectangular computational domain.

The motivation behind this research was to overcome the aforementioned shortcomings and by relying on the well capability of Thin Plate Spline Radial Basis Function (TPS-RBF) for interpolating scattered data in complex geometry, non-requirement for additional parameters like shape factor, etc., and the accuracy of traditional DQM, an efficient and truly meshless method called Global Thin Plate Spline Differential Quadrature (GTPS-DQM) is developed for two-dimensional Burgers' equation.

The paper is outlined as follows: Section 2 presents a brief description of the theory and numerical formulation of TPS-DQM. Section 2.2 outlines the discretization process of governing equation and implementation of the algorithm. Section 3 resolves the issues of various case studies via TPS-DQM in order to evaluate the model efficiency. Finally, Section 4 is dedicated to the conclusion.

2. Methodology

2.1. Numerical formulation of TPS-DQM

The standard DQM was first proposed by Bellman and Casti [25] based on the idea of quadrature integration as a highly accurate and potent approach to approximate partial differential equations, and the method has been recently used in many engineering problems owing to its efficiency [26–29]. According to the main concept of the DQM, the nth-order partial derivatives of the function $f(x,y)$ at any mesh point (x_i,y_j) relative to x and y are approximated by the linear weighted summation of the function values at all mesh points on the same axis x as follows:

$$\frac{\partial f^n(x_i,y_j)}{\partial x}=\sum_{k=1}^{Nx}W_x^n(i,k)..f(x_k,y_j) \quad (i=1,2,\cdots,Nx), \quad (5)$$

$$\frac{\partial f^n(x_i,y_j)}{\partial x}=\sum_{k=1}^{Ny}W_y^n(j,k)\cdot f(x_i,y_k) \quad (j=1,2,...,Ny), \quad (6)$$

where W_x^n and W_y^n are coefficients of the nth order derivatives with respect to x, y and Nx, Ny are the number of mesh points in x, y directions of a Cartesian computation domain. In traditional DQM, the coefficients of the derivatives are determined by using polynomials and harmonic functions as test functions and cannot be used in complex and irregular geometries without additional techniques such as domain decomposition and domain transformation techniques [30].

To overcome this issue, RBF-DQM as a new class of mesh-free methods was developed in [23] utilizing Multiquadric Radial Basis Functions (MQ-RBF) as the test function. In this approach, the computation nodes that are distributed arbitrarily discretize the computation domain and the nth-order partial derivatives of the function $f(x,y)$ at the ith computation node $x_i = (x_i, y_i)$ are approximated by the linear weighted summation of function values in the domain as follows:

$$\frac{\partial f^n(x_i,y_i)}{\partial x} = \sum_{k=1}^{M} W_x^n(i,k)\cdot f(x_k,y_k), \quad (i=1,2,\cdots,N), \tag{7}$$

$$\frac{\partial f^n(x_i,y_i)}{\partial x} = \sum_{k=1}^{M} W_x^n(i,k)\cdot f(x_k,y_k), \quad (i=1,2,\cdots,N). \tag{8}$$

Despite the flexibility and good performance of this method for solving partial differential equations, it suffers from choosing the optimal value for additional parameters such as shape factor [31]. In the present study, Thin Plate Spline (TPS) was used due to its good capability for the interpolation of 2D scattered data and non-requirement for additional parameters such as shape factor as follows [32]:

$$\varphi_j(x,y) = \varphi_j(r) = r^2\ln(r), \tag{9}$$

where r denotes the Euclidean norm as follows:

$$r = \|X - X_j\| = \sqrt{(x-x_j)^2 + (y-y_j)^2}. \tag{10}$$

Moreover, partial derivatives of TPS φ can be obtained through simple differentiation rules. For computing coefficients of derivatives for function $\varphi(x,y)$ by substituting Eq. (9) into Eqs. (7) and (8) at all computation nodes, a linear system of equations is formed as follows:

$$\begin{bmatrix} \varphi_1(x_1,y_1) & \varphi_1(x_2,y_2) & \cdots & \cdots & \varphi_1(x_N,y_N) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \varphi_N(x_1,y_1) & \varphi_N(x_2,y_2) & \cdots & \cdots & \varphi_N(x_N,y_N) \end{bmatrix} \begin{Bmatrix} Wx_{i,1}^{(n)} \\ \cdots \\ \cdots \\ \cdots \\ Wx_{i,N}^{(n)} \end{Bmatrix} = \begin{Bmatrix} \frac{\partial^n \varphi_1(x_i,x_i)}{\partial x^n} \\ \cdots \\ \cdots \\ \cdots \\ \frac{\partial^n \varphi_N(x_i,x_i)}{\partial x^n} \end{Bmatrix}. \tag{11}$$

Since the value of function $\varphi(x,y)$ and its nth order partial derivatives are all known, the linear system of Eqs. (11) with N unknowns can be solved for weighting coefficients $W_x^n(i,k)$. The algorithm can be similarly used for determining the weighting coefficients $W_y^n(i,k)$ of the y-derivatives and higher-order derivatives. It is pertinent to note that for problems within irregular domains, the weighting coefficients are contingent upon specific cases. These coefficients are ascertained a priori through the resolution of an algebraic system, as delineated in Eq. (11) [33]. If N is considered as the total number of mesh points ($N = M$), the model called GTPS-DQM is considered; otherwise, if it is selected as the limited number of mesh points in the neighborhood of the reference node ($N < M$), the model is called Local Thin Plate Spline Differential Quadrature (LTPS-DQM). The GTPS-DQM method, in which all points participate in the estimation of derivative terms, has relatively higher computational accuracy than LTPS-DQM. Although in problems with a large number of computation points, GTPS-DQM may increase the computation time and cause an ill-condition in the coefficients matrix [26,27,33–39]. GTPS-DQM was utilized in the present work.

Figure 1 represents a sketch for the arrangement of computation nodes in an irregular domain Ω.

2.2. Implementing TPS-DQM for Burgers' equation

The general description of the TPS-DQM method was presented in the previous section. The discretization process of the spatial derivatives of the Burgers' equation using the TPS-DQM method is presented in this section. By approximating temporal derivatives of Burgers' equation using first-order forward finite difference, Eqs. (1) and (2) are discretized in the implicit form:

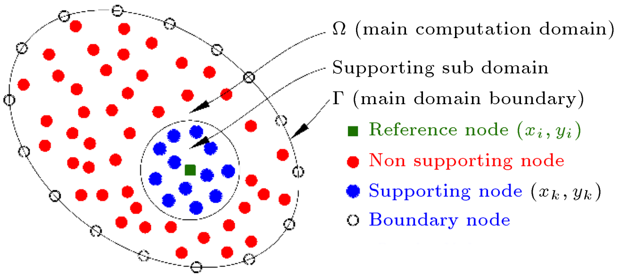

Ω (main computation domain)
Supporting sub domain
Γ (main domain boundary)
■ Reference node (x_i, y_i)
● Non supporting node
● Supporting node (x_k, y_k)
○ Boundary node

Figure 1. Reference and support nodes definition in TPS-DQM.

$$\left\{\frac{\partial u(x,t)}{\partial t}\right\}^{t+1} \cong \frac{u^{t+1}(x,t)-u^{t}(x,t)}{\Delta t}$$
$$= \left\{\frac{1}{Re}\nabla^2 u(x,t) - u(x,t)\frac{\partial u(x,t)}{\partial x} - v(x,t)\frac{\partial u(x,t)}{\partial y}\right\}^{t+1}, \tag{12}$$

$$\left\{\frac{\partial v(x,t)}{\partial t}\right\}^{t+1} \cong \frac{v^{t+1}(x,t)-v^{t}(x,t)}{\Delta t}$$
$$= \left\{\frac{1}{Re}\nabla^2 v(x,t) - u(x,t)\frac{\partial v(x,t)}{\partial x} - v(x,t)\frac{\partial v(x,t)}{\partial y}\right\}^{t+1}, \tag{13}$$

where superscripts t and $t+1$ represent the values in two consecutive time steps with interval Δt. By implementing TPS-DQM rolls as Eqs. (7) and (8) to approximate spatial derivatives and some mathematical simplifications, we have:

$$u_i^{n+1} = u_i^n + \frac{1}{2}\Delta t\left\{\frac{1}{Re}\left(\sum_{k=1}^{N} w_x^2(i,k)u_i^{n+1} + \sum_{k=1}^{N} w_y^2(i,k)u_i^{n+1}\right) - u_i^{n+1}\sum_{k=1}^{N} w_x^1(i,k)u_i^{n+1} - v_i^{n+1}\sum_{k=1}^{N} w_y^1(i,k)u_i^{n+1}\right\}, \tag{14}$$

$$v_i^{n+1} = v_i^n + \frac{1}{2}\Delta t\left\{\frac{1}{Re}\left(\sum_{k=1}^{N} w_x^2(i,k)v_i^{n+1} + \sum_{k=1}^{N} w_y^2(i,k)v_i^{n+1}\right) - u_i^{n+1}\sum_{k=1}^{N} w_x^1(i,k)v_i^{n+1} - v_i^{n+1}\sum_{k=1}^{N} w_y^1(i,k)u_i^{n+1}\right\}, \tag{15}$$

where $i = 1, 2, \cdots, N$. The system of nonlinear Eqs. (14) and (15) can be solved by applying the iterative Newton Raphson method to achieve the field values of velocity components u and v. Several MATLAB® functions were developed to implement the present algorithm, and open-source software GMSH was used for the unstructured meshing of the two-dimensional domains.

3. Numerical results and discussion

To evaluate the performance of the TPS-DQM,three two-dimensional examples were solved. The first and third examples had exact solutions. In the second example, a case with a well-known numerical solution in the literature was resolved using the present method. The third example was considered to demonstrate the capability of TPS-DQM to deal with irregular computation domains and geometry.

In the above examples, the accuracy of the present model was appraised through error analysis for the obtained results by computing root mean squared error norms L_2 and maximum absolute error L_∞ for error indicators as follows [31]:

$$L_2 = \sqrt{\frac{1}{N}\sum_{i=1}^{N}|u_i - \hat{u}_i|^2}, \qquad i = 1, \cdots, N, \tag{16}$$

$$L_\infty = \max\left(|u_i - \hat{u}_i|\right), \qquad i = 1, \cdots, N, \tag{17}$$

where u_i and $\hat{u}_i$ are the results obtained from the proposed model and the exact solution of the equation, respectively, and N is the number of mesh points.

3.1. Example 1

Two-dimensional Burgers' equation with the following analytical solution is considered as the first example case. In [7], an exact solution to this example was derived utilizing the HOPF-COLE transformation, a mathematical technique that converts nonlinear partial differential equations, particularly those of the Burgers' type, into linear ones, thereby facilitating easier analytical solutions.

$$u(x,y,t) = \frac{3}{4} - \frac{1}{4\left[1 + \frac{Re}{32}exp(-4x+4y-t)\right]}, \tag{18}$$

$$0 \le x, \quad y \le 1.0,$$

$$v(x,y,t) = \frac{3}{4} + \frac{1}{4\left[1 + \frac{Re}{32}exp(-4x+4y-t)\right]}, \tag{19}$$

$$0 \le x, \quad y \le 1.0.$$

Dirichlet-type boundary conditions and the initial condition for this example can be evaluated by the exact solution. This example is solved in a rectangular domain using the TPS-DQM method at Reynolds numbers 100 and 1000. The solutions were implemented with different numbers of nodes distributed randomly in computation, as illustrated in Figure 2 and time step $\Delta t = 5\text{e}^{-4}$.

Figure 3 represents the velocity components for this example at $t = 2.0$ with $Re = 1000$ obtained from TPS-DQM. As expected and mentioned in [40] at high Reynolds numbers due to the domination of convection term and a sharp gradient consequently, a wave with constant velocity, like an intense shock wave, parades in the computational domain.

To demonstrate the effect of mesh resolution on the accuracy of the obtained result, the graphs of absolute error for the example at $Re = 1000$ with $N = 272$, 441, 1507, and 2132 are portrayed in Figure 4. As is clear, the present method has a high degree of accuracy in solving the example, although its performance is highly dependent on the number of points.

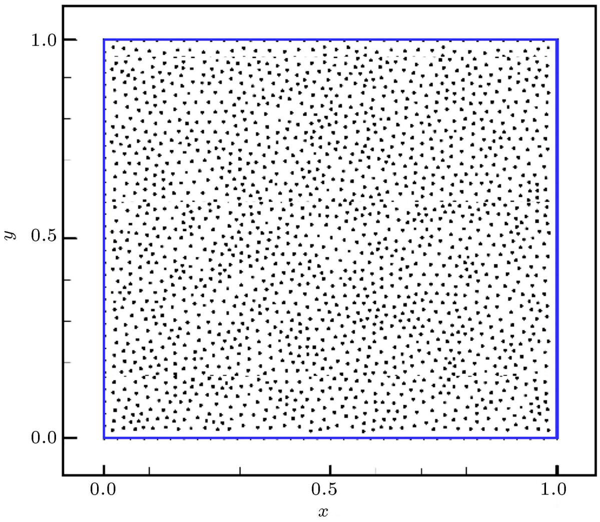
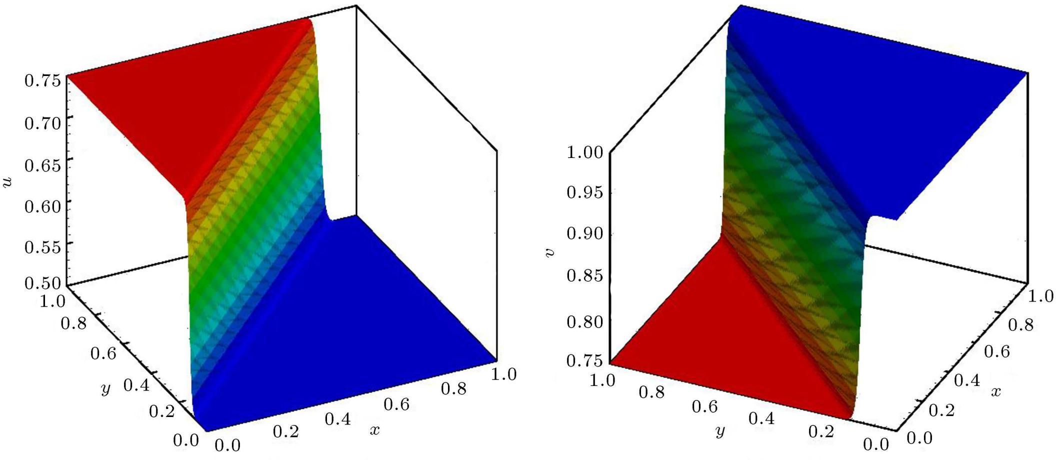


Figure 2. Unstructured distribution of computation nodes in a rectangular domain.

The performance of the present method in solving the example with $Re = 100$ and different numbers of computation points is compared with those of Galerkin-reproducing kernel method [40], Global RBFs collocation method accompanied by first-order forward difference approximation in time [41], fully implicit finite-difference method [42], local RBFs collocation method accompanied by first-order forward difference approximation in time [43], and method of fundamental solution [44] as various well-known numerical solutions.

The results obtained at the selected points, which are summarized in Tables 1 and 2, imply the higher efficiency and accuracy of the present method than other methods, despite having equal computational nodes.

The graph of velocity components for the example



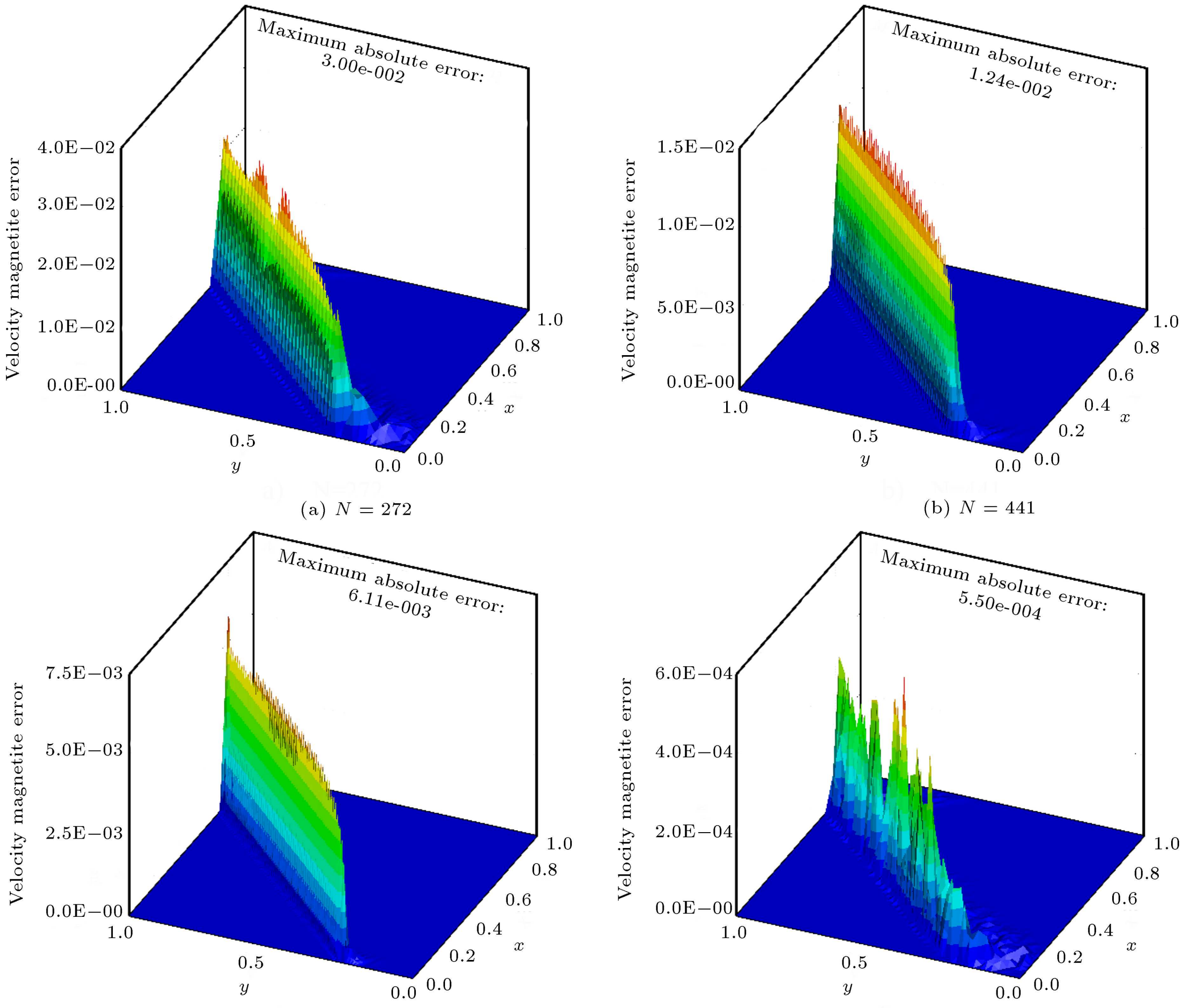
(a) u component

(b) v component

Figure 3. Numerical solution of velocity components for Example 1 at $t = 2.0$ with $Re = 1000$ and $N = 1507$.

Table 1. Numerical solution for u component with $Re = 100$ at $t = 2.0$.

	***u* component**								
		(x, y)						**L_∞**	**L_2**
	N	**(0.1,0.1)**	**(0.3,0.3)**	**(0.5,0.5)**	**(0.3,0.7)**	**(0.1,0.9)**	**(0.5,0.9)**		
Exact	—	0.500482	0.500482	0.500482	0.555675	0.744256	0.555675	—	—
TPS-DQM	141	0.5004843	0.5004861	0.5004872	0.5570958	0.7440880	0.5561600	1.42e-03	6.17e-04
	205	0.5004824	0.5004829	0.5004832	0.5561744	0.7441940	0.5558355	4.99e-04	2.16e-04
	325	0.5004817	0.5004818	0.5004818	0.5557064	0.7442516	0.5556860	3.14e-05	1.37e-05
	441	0.5004817	0.5004817	0.5004817	0.5557026	0.7442519	0.5556808	2.76e-05	1.16e-05
Ref. [40]	441	0.500479	0.500479	0.500479	0.555729	0.744234	0.555714	5.40e-05	2.87e-05
Ref. [41]	441	0.500470	0.500441	0.500414	0.554805	0.744197	0.554489	1.19e-03	6.02e-04
Ref. [42]	441	0.50035	0.50042	0.50046	0.55609	0.74409	0.55604	4.15e-04	2.43e-04
Ref. [43]	441	0.49983	0.49977	0.49973	0.55429	0.74340	0.55413	1.55e-03	1.04e-03
Ref. [44]	441	0.50012	0.50042	0.50041	0.55413	0.74416	0.55637	1.55e-03	7.09e-04



(a) $N = 272$

(b) $N = 441$

(c) $N = 1507$

(d) $N = 2132$

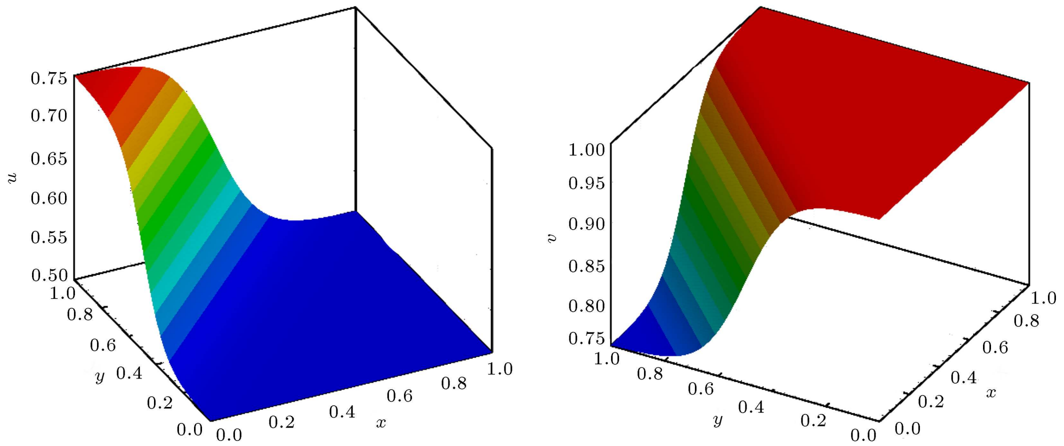
Figure 4. Absolute error of velocity magnetite for Example 1 at $t = 2.0$ with $Re = 1000$.

with $Re = 100$ at $t = 2.0$ obtained by TPS-DQM based on the results with $N = 441$ is illustrated in Figure 5. Reduction in the intensity of the velocity gradient at the wavefront is quite clear, compared to $Re = 1000$.

Graphs of the distribution of absolute error with $Re = 100$ at $t = 2.0$ for different mesh resolutions, as portrayed in Figure 6, show the reduction of errors as the number of mesh points increases.

Table 2. Numerical solution for v component with $Re = 100$ at $t = 2.0$.

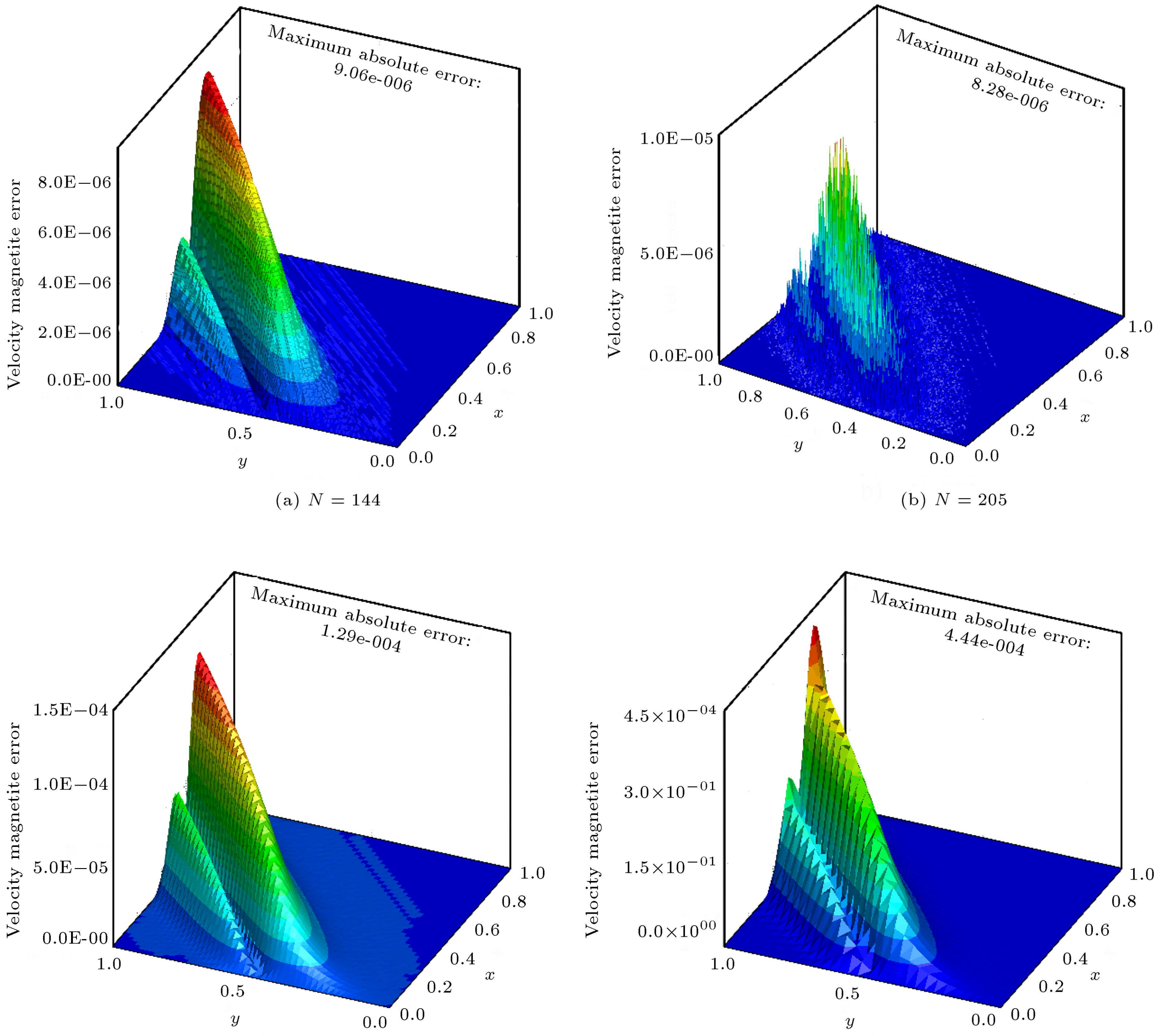
	***v* component**								
		(x, y)						**L_∞**	**L_2**
	N	**(0.1,0.1)**	**(0.3,0.3)**	**(0.5,0.5)**	**(0.3,0.7)**	**(0.1,0.9)**	**(0.5,0.9)**		
Exact	—	0.999518	0.999518	0.999518	0.944325	0.755744	0.944325	—	—
TPS-DQM	141	0.9995157	0.9995139	0.9995128	0.9429042	0.7559120	0.9438400	1.42e-03	6.17e-04
	205	0.9995176	0.9995171	0.9995168	0.9438256	0.7558060	0.9441645	4.99e-04	2.16e-04
	325	0.9995183	0.9995182	0.9995182	0.9442936	0.7557484	0.9443140	3.14e-05	1.37e-05
	441	0.9995183	0.9995183	0.9995183	0.9442974	0.7557481	0.9443192	2.76e-05	1.16e-05
Ref. [40]	441	0.999520	0.999521	0.999521	0.944270	0.755765	0.944285	1.19e-03	6.02e-04
Ref. [41]	441	0.999530	0.999559	0.999586	0.945195	0.755803	0.945511	4.55e-04	2.68e-04
Ref. [42]	441	0.99936	0.99951	0.99958	0.94387	0.75592	0.94392	1.31e-03	8.88e-04
Ref. [43]	441	0.99826	0.99861	0.99821	0.94409	0.75500	0.94441	8.75e-04	4.15e-04
Ref. [44]	441	0.99946	0.99938	0.99941	0.94387	0.75558	0.94345	1.28e-04	5.59e-05



(a) u component

(b) v component

Figure 5. Numerical solution of velocity components for Example 1 at $t = 2.0$ with $Re = 100$ and $N = 441$.



(a) $N = 144$

(b) $N = 205$

(c) $N = 325$

(d) $N = 441$

Figure 6. Absolute error of velocity magnetite for Example 1 at $t = 2.0$ with $Re = 100$.

3.2. Example 2

Consider the two-dimensional Burgers' equation with the following boundary and initial conditions [45], we have:

$$u(x,y,t_0=0)=\sin\pi x\cdot\sin\pi y,$$

$$0\le x,\quad y\le 1.0,\quad t\ge 0, \tag{20}$$

$$v(x,y,t_0=0)=(\sin\pi x+\sin 2\pi x)(\sin\pi y+\sin 2\pi y),$$

$$0\le x,\quad y\le 1.0,\quad t\ge 0, \tag{21}$$

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

$$0\le x,\quad y\le 1.0,\quad t\ge 0, \tag{22}$$

$$v(0,y,t)=v(1,y,t)=v(x,0,t)=v(x,1,t)=0,$$

$$0\le x,\quad y\le 1.0,\quad t\ge 0. \tag{23}$$

The example was solved at $Re = 100$ in a rectangular domain, the same as that in Figure 2, using TPS-DQM with 3750 mesh points and $\Delta t = 0.001$. Since there is not any analytical solution to this problem, the obtained results are compared against another well-known numerical solution based on DAF [4]. The comparison of the results at the selected point at $t = 0.5$, $t = 1.0$, as summarized in Table 3, demonstrates that the proposed model achieves results comparable to those in [4], despite far fewer mesh points in consequence.

Figure 7 represents the contours of velocity components for Example 2 at $t = 0.5, 1.0$. The compression of contours of velocity components near the boundaries and corners of the computational domain results from the high value of the velocity gradient, which the proposed model can capture accurately.

3.3. Example 3

To demonstrate the capability of the present method to solve the Burgers' equation in an irregular and non-rectangular computational domain, Example 1 with the exact solution as Eqs. (18) and (19) was again resolved in an elliptical computational domain. The ellipse with a large diameter of 1.0 and a small diameter of 0.5 was irregularly meshed, as shown in Figure 8.

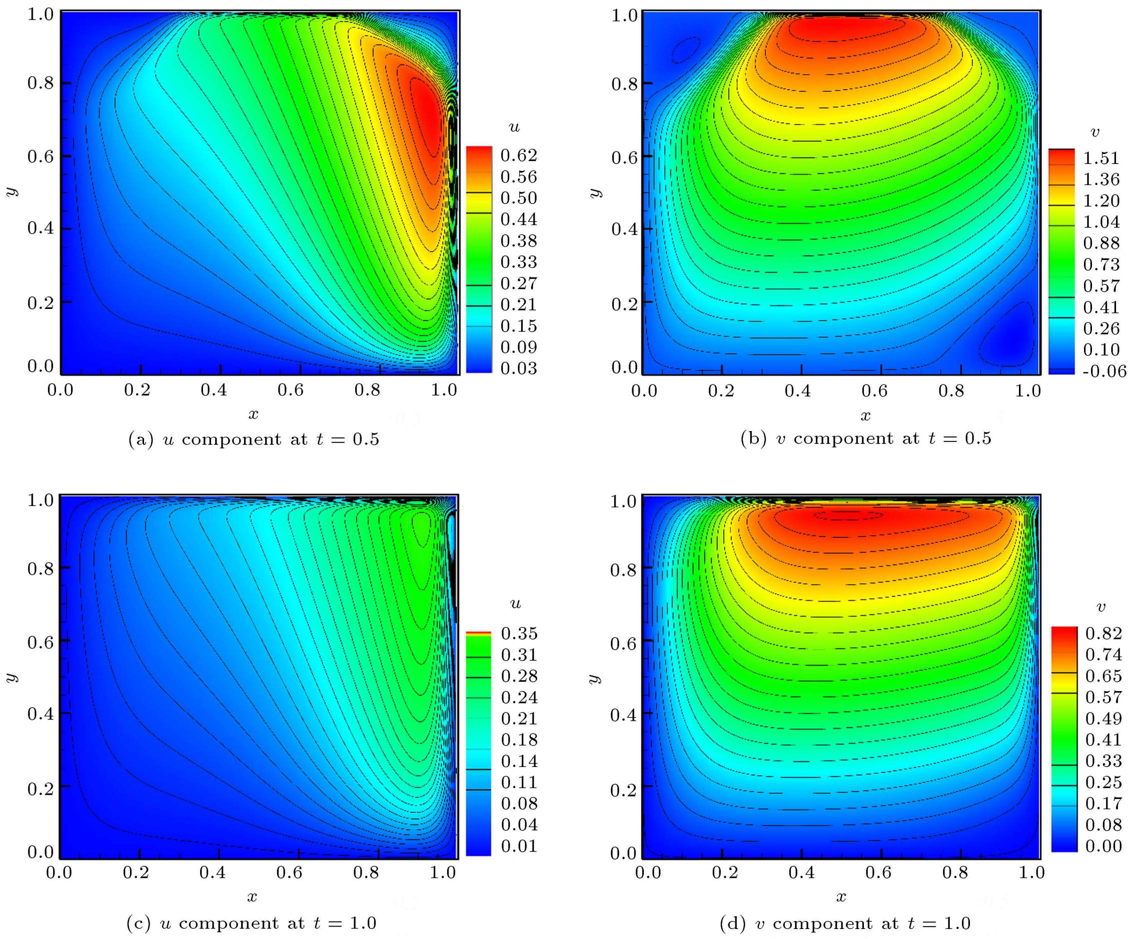
The solution was performed using TPS-DQM at $Re = 100$, 1000 with different mesh resolutions and time increments. The graphs of velocity components obtained from results at $t = 1.0$ are shown in Figures 9 and 10.

Likewise, in this example, increase in the gradient of the velocity components followed by the increase of Reynolds number in the wavefront is visible, which can be captured easily by the present method. The error analysis is carried out for velocity components in this example with $Re = 100$, 100 and various mesh resolutions at $t = 2.0$ in Table 4. In this table, the CPU times are also presented based on the run time of MATLAB® code on a notebook featuring an Intel Core i5-4200M @ 2.50 GHz, 6.0 GB of RAM.

The good agreement between the obtained results with exact solution and the stability of method is observed. In addition, the accuracy of the solution rises upon increasing the mesh resolution, although increasing the number of computational points leads to high computational effort.

Table 3. Numerical solution of Eexample 2 for $Re = 1.0\text{e}2$.

			DAF		**TPS-DQM**	
			($N = 10000, \Delta t = 0.001$)		**($N = 3750, \Delta t = 0.001$)**	
Location			**$t = 0.5$**	**$t = 1.0$**	**$t = 0.5$**	**$t = 1.0$**
x	0.1	u	0.0150939	0.0072642	0.0150855	0.0072614
y	0.1	v	0.1216201	0.0554181	0.1215602	0.0554008
x	0.2	u	0.1583939	0.0807567	0.1579790	0.0807072
y	0.8	v	0.9865377	0.5817142	0.9828976	0.5814791
x	0.4	u	0.1282258	0.0704512	0.1282049	0.0704444
y	0.4	v	0.700210	0.3690011	0.7000421	0.3689213
x	0.7	u	0.1335301	0.0681616	0.1336236	0.0682031
y	0.1	v	0.0999871	0.0744617	0.0998923	0.0744419
x	0.8	u	0.5637791	0.2957080	0.5638457	0.2957010
y	0.8	v	1.1851231	0.6967743	1.1854839	0.6967343
x	0.9	u	0.2812818	0.3664864	0.2796991	0.3664107
y	0.9	v	0.221959	0.7523752	0.2204293	0.7522460



(a) u component at $t = 0.5$

(b) v component at $t = 0.5$

(c) u component at $t = 1.0$

(d) v component at $t = 1.0$

Figure 7. Contours of velocity component for Example 2 with $Re = 100$ and $N = 3750$.

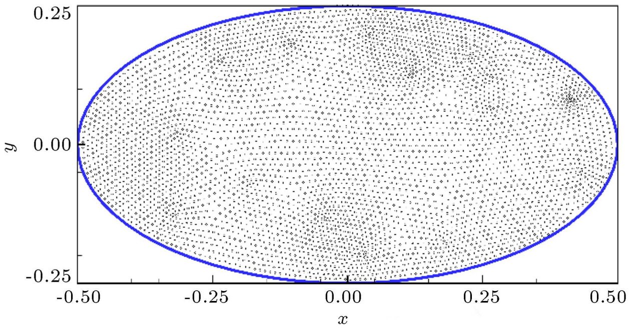


Figure 8. Unstructured distribution of computation nodes in an irregular domain.

Figures 11 and 12 illustrate graphs of the absolute error distribution for velocity magnitude for $Re = 100$, 1000 with various discretization settings. As expected, the highest value of absolute errors occurred near the wavefront region. As mentioned earlier, it is due to the large value of gradients near the wavefront. Moreover, intensive domination of convection term and, consequently, stronger pseudo shock wave at higher Reynolds numbers led to the higher concentration of errors near the wavefront region. Therefore, at $Re = 100$, the errors spread over the whole computational domain, while at $Re = 1000$, the errors are almost concentrated in the wavefront region.

4. Conclusion

In this study, being a high-order and meshless method, Thin Plate Spline Differential Quadrature (TPS-DQM) was employed to solve the Burgers' equation. The spatial derivatives were discretized implicitly by this method in two-dimensional domains with unstructured mesh distribution, and the equations were solved in rectangular and non-rectangular irregular computation domains. Results were compared with an exact and well-known numerical solution. The results demonstrated that despite the need for fewer mesh points and, consequently, less computational effort, the TPS-DQM model managed to solve the Burgers' equation more accurately than previous models. Moreover, due to its flexibility and truly meshless nature, the present model was able to solve Burgers' equation in irregular computation domains. This process showed that the TPS-DQM could accurately capture the physical behavior of Burgers' equation in different conditions and geometry.

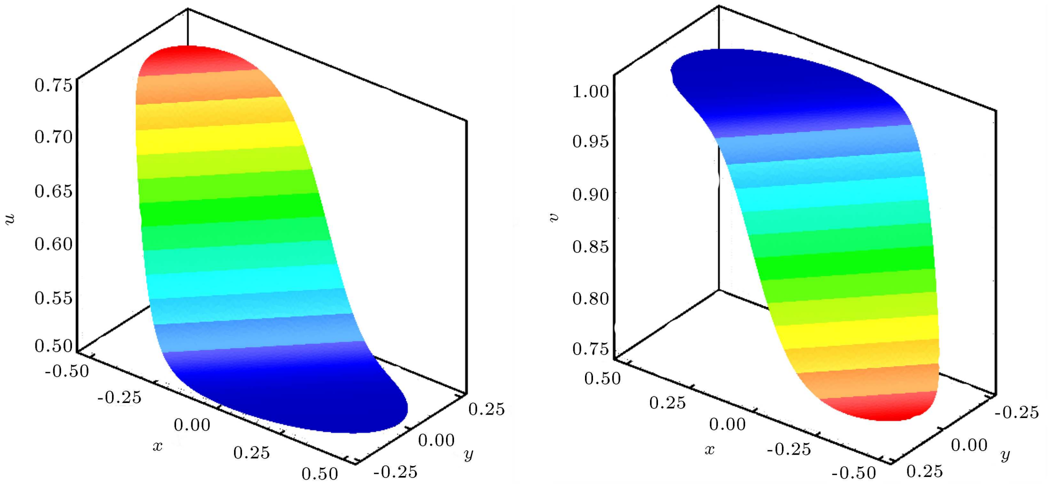
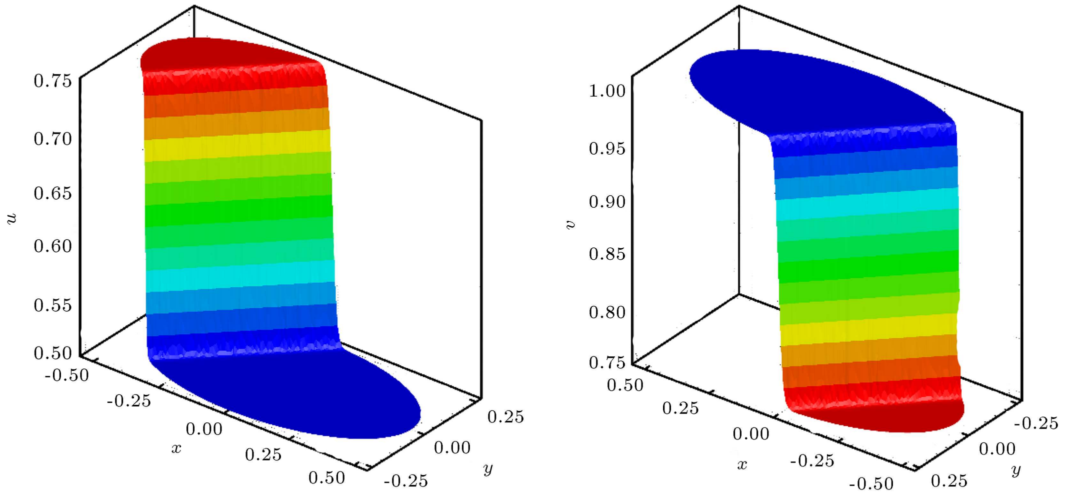


Figure 9. Numerical solution of velocity components for Example 3 at $t = 1.0$ with $Re = 100$ and $N = 678$.



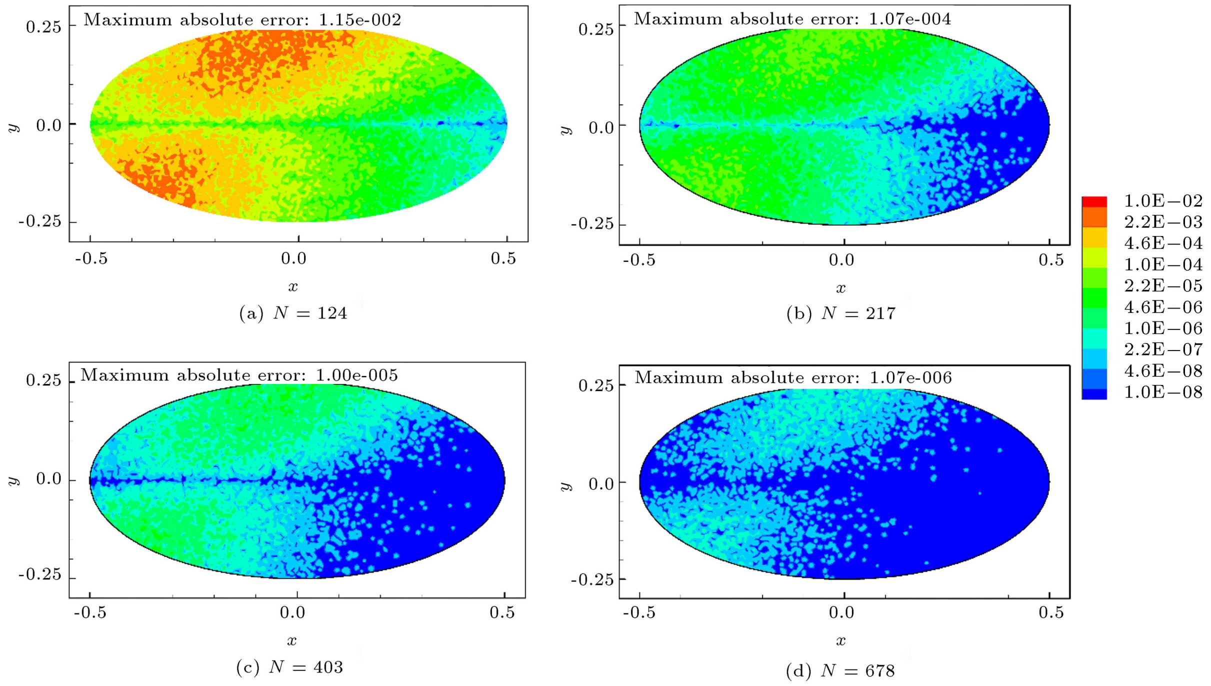
(a) u component

(b) v component

Figure 10. Numerical solution of velocity components for Example 3 at $t = 1.0$ with $Re = 1000$ and $N = 2135$.

Table 4. Error norms for velocity components in Example 3 at $t = 2.0$.

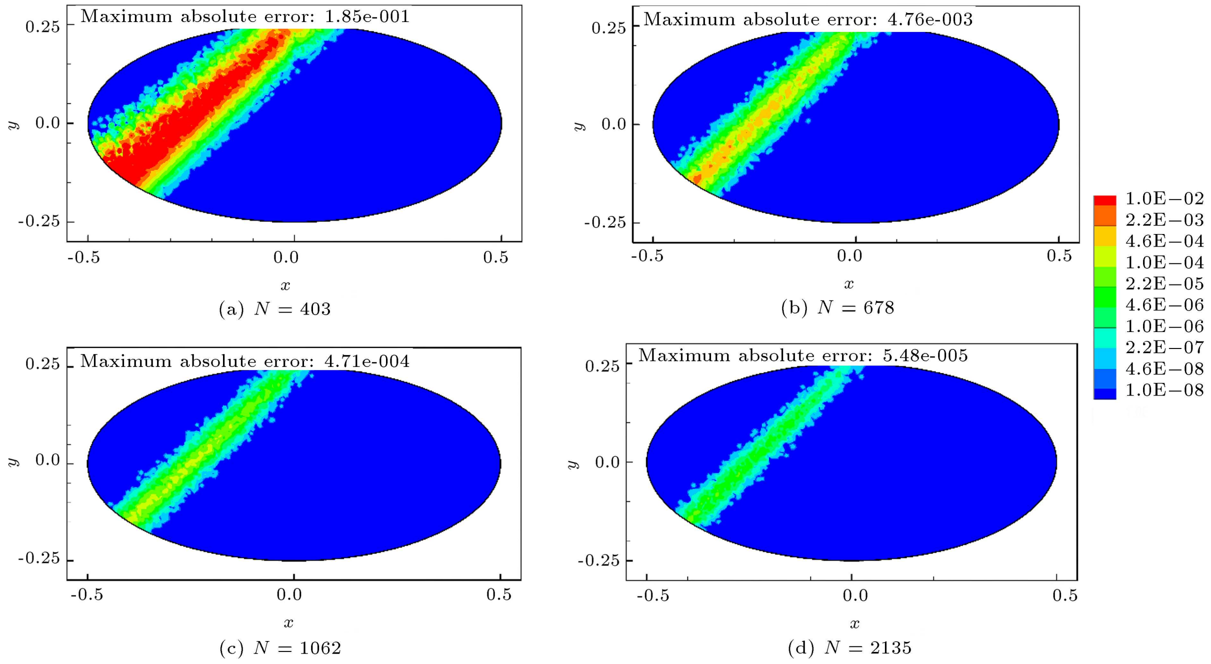
Re	N	Δt	CPU time (s)	u		v	
				L_∞	L_2	L_∞	L_2
100	124	5e-3	23.26	1.37e-2	2.99e-03	1.14e-2	2.32e-03
	217	1e-3	71.11	2.17e-4	1.25e-05	2.34e-4	1.53e-05
	403	5e-4	125.36	1.74e-5	2.12e-06	2.3e-5	2.54e-06
	678	1e-4	317.12	1.62e-6	1.26e-07	2.12e-6	1.13e-07
1000	403	5e-4	209.18	2.12e-2	2.11e-03	1.92e-2	1.35e-03
	678	1e-4	392.75	3.34e-4	1.39e-05	2.86e-4	2.08e-05
	1062	1e-4	521.17	2.11e-5	3.10e-06	1.92e-5	2.91e-06
	2135	5e-5	612.74	3.17e-6	2.16e-07	2.97e-6	1.72e-07



(a) $N = 124$ (b) $N = 217$

(c) $N = 403$ (d) $N = 678$

Figure 11. Absolute error of velocity magnetite for Example 3 at $t = 2.0$ with $Re = 100$.



(a) $N = 403$ (b) $N = 678$

(c) $N = 1062$ (d) $N = 2135$

Figure 12. Absolute error of velocity magnetite for Example 3 at $t = 2.0$ with $Re = 1000$.

It is believed that this method can be employed as a powerful, simple, and accurate tool for solving other hydrodynamic problems.

References

1. Bateman, H. "Some recent researches on the motion of fluids", *Mon. Weather Rev.*, **43**(4), pp. 163–170 (1915).
2. Burgers, J.M. "A mathematical model illustrating the theory of turbulence", In *Advances in Applied Mechanics, Elsevier*, **1**(C), pp. 171–199 (1948).
3. Wazwaz, A.-M., *Partial Differential Equations: Methods and Applications*, AA Balkema (2003).
4. Wei, G.W., Zhang, D.S., Kouri, D.J., et al. "Dis-

tributed approximating functional approach to Burgers' equation in one and two space dimensions", *Comput. Phys. Commun.*, **111**(1–3), pp. 93–109 (1998).

5. Hopf, E. "The partial differential equation $ut + uux = \mu xx$", *Commun. Pure Appl. Math.*, **3**(3), pp. 201–230 (1950).

6. Cole, J.D. "On a quasi-linear parabolic equation occurring in aerodynamics", *Q. Appl. Math.*, **9**(3), pp. 225–236 (1951).

7. Fletcher, C.A.J. "Generating exact solutions of the two-dimensional Burgers' equations", *Int. J. Numer. Methods Fluids*, **3**(3), pp. 213–216 (1983).

8. Varoğlu, E. and Liam Finn, W.D. "Space-time finite elements incorporating characteristics for the burgers' equation", *Int. J. Numer. Methods Eng.*, **16**(1), pp. 171–184 (1980).

9. Caldwell, J., Wanless, P., Cook, A.E., et al. "A finite element approach Burgers' equation to", *Appl. Math. Model.*, **5**(June), pp. 189–193 (1981).

10. Kutluay, S., Bahadir, A.R., and Özdeş, A. "Numerical solution of one-dimensional Burgers equation: explicit and exact-explicit finite difference methods", *J. Comput. Appl. Math.*, **103**(2), pp. 251–261 (1999).

11. Evans, D.J. and Abdullah, A.R. "The group explicit method for the solution of Burger's equationDie gruppenexplizite methode für die Lösung der Burgerschen Gleichung", *Computing*, **32**(3), pp. 239–253 (1984).

12. Kakuda, K. and Tosaka, N. "The generalized boundary element approach to Burgers' equation", *Int. J. Numer. Methods Eng.*, **29**(2), pp. 245–261 (1990).

13. Ozis, T., Ozdes, A., Of, J., et al. "Guidelines for the safe use of Doppler Ultrasound for clinical applications", *Eur. J. Ultrasound*, **2**(2), pp. 167–168 (1995).

14. Bar-Yoseph, P., Moses, E., Zrahia, U., et al. "Space-time spectral element methods for one-dimensional nonlinear advection-diffusion problems", *J. Comput. Phys.*, **119**(1), pp. 62–74 (1995).

15. Zhang, D.S., Wei, G.W., Kouri, D.J., et al. "Burgers' equation with high Reynolds number", *Phys. Fluids*, **9**(6), pp. 1853–1855 (1997).

16. Mittal, R.C. and Jiwari, R. "Differential quadrature method for two-dimensional Burgers' equations", *Int. J. Comput. Methods Eng. Sci. Mech.*, **10**(6), pp. 450–459 (2009).

17. Vaghefi, M., Rahideh, H., Haghighi, M.R.G., et al. "Distributed approximating functional approach to Burgers' equation using element differential quadrature method", *J. Appl. Sci. Environ. Manag.*, **16**(1), pp. 143–149 (2012).

18. Esen, A. and Tasbozan, O. "Numerical solution of time fractional burgers equation by cubic B-spline finite elements", *Mediterr. J. Math.*, **13**(3), pp. 1325–1337 (2016).

19. Aswin, V.S., Awasthi, A., and Rashidi, M.M. "A differential quadrature based numerical method for highly accurate solutions of Burgers' equation", *Numer. Methods Partial Differ. Equ.*, **33**(6), pp. 2023–2042 (2017).

20. Erdo, U., Ozis, T., Ozis, T., et al. "Numerical solution of Burgers' equation with high order splitting methods", *J. Comput. Appl. Math.*, **291**(January 2016), pp. 410–421 (2016).

21. Sakar, M.G., Saldır, O., and Erdogan, F. "Numerical solution of time-fractional Burgers' equation in reproducing kernel space", arXiv Prepr. arXiv1805.06953 (2018).

22. Hussain, M., Haq, S., Ghafoor, A., et al. "Numerical solutions of time-fractional coupled viscous Burgers' equations using meshfree spectral method", *Comput. Appl. Math.*, **39**(1), p. 6 (2019).

23. Wu, Y.L. and Shu, C. "Development of RBF-DQ method for derivative approximation and its application to simulate natural convection in concentric annuli", *Comput. Mech.*, **29**(6), pp. 477–485 (2002).

24. Sugihara, M. and Fujino, S. "Numerical solutions of Burgers' equation with a large Reynolds number", *Reliab. Comput.*, **2**(2), pp. 173–179 (1996).

25. Bellman, R. and Casti, J. "Differential quadrature and long-term integration", *J. Math. Anal. Appl.*, **34**(2), pp. 235–238 (1971).

26. Motaman, F., Rakhshandehroo, G.R., Hashemi, M.R., et al. "Application of RBF-DQ method to time-dependent analysis of unsaturated seepage", *Transp. Porous Media*, **125**(3), pp. 543–564 (2018).

27. Parand, K. and Hashemi, S. "RBF-DQ method for solving non-linear differential equations of Lane-Emden type", *Ain Shams Eng. J.*, **9**(4), pp. 615–629 (2018).

28. Sun, D., Ai, Y., Zhang, W., et al. "Direct solution of Navier-Stokes equations by using an upwind local RBF-DQ method", *J. Vibroengineering*, **16**(1), pp. 78–89 (2014).

29. Homayoon, L., Abedini, M.J., and Hashemi, S.M.R. "RBF-DQ solution for shallow water equations", *J. Waterw. Port, Coastal, Ocean Eng.*, **139**(1), pp. 45–60 (2012).

30. Behroozi, A.M. and Vaghefi, M. "Multi-block DQM/RBF-DQ as a meshless model for numerical investigation of laminar flow and forced convection in a channel with two circular fins", *Eng. Anal. Bound. Elem.*, **125**, pp. 33–45 (2021).

31. Ding, H., Shu, C., and Tang, D.B. "Error estimates of local multiquadric-based differential quadrature (LMQDQ) method through numerical experiments", *Int. J. Numer. Methods Eng.*, **63**(11), pp. 1513–1529 (2005).

32. Bookstein, F.L. "Principal warps: thin-plate splines and the decomposition of deformations", *IEEE Trans. Pattern Anal. Mach. Intell.*, **11**(6), pp. 567–585 (1989).

33. Behroozi, A.M. and Vaghefi, M. "Radial basis function differential quadrature for hydrodynamic pressure on

dams with arbitrary reservoir and face shapes affected by earthquake", *J. Appl. Fluid Mech.*, **13**(06), pp. 1759–1768 (2020).

34. Behroozi, A.M. and Vaghefi, M. "Numerical simulation of water hammer using implicit Crank-Nicolson local multiquadric based differential quadrature", *Int. J. Press. Vessel. Pip.*, **181**, p. 104078 (2020).

35. Abbaszadeh, M. and Dehghan, M. "An upwind local radial basis functions-differential quadrature (RBFs-DQ) technique to simulate some models arising in water sciences", *Ocean Eng.*, **197**, p. 106844 (2020).

36. Watson, D.W., Karageorghis, A., and Chen, C.S. "The radial basis function-differential quadrature method for elliptic problems in annular domains", *J. Comput. Appl. Math.*, **363**, pp. 53–76 (2020).

37. Liu, J., Li, X., and Hu, X. "A RBF-based differential quadrature method for solving two-dimensional variable-order time fractional advection-diffusion equation", *J. Comput. Phys.*, **384**, pp. 222–238 (2019).

38. Ghalandari, M., Shamshirband, S., Mosavi, A., et al. "Flutter speed estimation using presented differential quadrature method formulation", *Eng. Appl. Comput. Fluid Mech.*, **13**(1), pp. 804–810 (2019).

39. Jafarabadi, A. and Shivanian, E. "Numerical simulation of nonlinear coupled Burgers' equation through meshless radial point interpolation method", *Eng. Anal. Bound. Elem.*, **95**, pp. 187–199 (2018).

40. Mohammadi, M., Mokhtari, R., and Panahipour, H. "A Galerkin-reproducing kernel method: Application to the 2D nonlinear coupled Burgers' equations", *Eng. Anal. Bound. Elem.*, **37**(12), pp. 1642–1652 (2013).

41. Ali, A., Siraj-ul-Islam, I., and Haq, S. "A computational meshfree technique for the numerical solution of the two-dimensional coupled Burgers' equations", *Int. J. Comput. Methods Eng. Sci. Mech.*, **10**(5), pp. 406–422 (2009).

42. Bahadır, A.R. "A fully implicit finite-difference scheme for two-dimensional Burgers' equations", *Appl. Math. Comput.*, **137**(1), pp. 131–137 (2003).

43. Siraj-ul-Islam, I., Šarler, B., Vertnik, R., et al. "Radial basis function collocation method for the numerical solution of the two-dimensional transient nonlinear coupled Burgers' equations", *Appl. Math. Model.*, **36**(3), pp. 1148–1160 (2012).

44. Young, D.L., Fan, C.M., Hu, S.P., et al. "The Eulerian–Lagrangian method of fundamental solutions for two-dimensional unsteady Burgers' equations", *Eng. Anal. Bound. Elem.*, **32**(5), pp. 395–412 (2008).

45. Arminjon, P. and Beauchamp, C. "Numerical solution of Burgers' equations in two space dimensions", *Comput. Methods Appl. Mech. Eng.*, **19**(3), pp. 351–365 (1979).

Biographies

Abdol Mahdi Behroozi was born in Bushehr, Iran in 1978. He earned BSc and MSc degrees in Civil Engineering from Persian Gulf University, Bushehr, Iran in 1999 and 2022, respectively. His research interests include the areas of computational mechanics, CFD, river engineering, hydraulic structures, hydrodynamics, and sediment transport. He has published 8 journal papers.

Mohammad Vaghefi was born in Shiraz, Iran in 1973. He earned BSc and MSc degrees in Civil Engineering from Shiraz University, Iran in 1997 and 1999, respectively, and a PhD degree from Tarbiat Modares University, Tehran, Iran in 2009. He is currently an Associate Professor of Civil Engineering at Persian Gulf University, Bushehr, Iran. His areas of interest include hydraulic engineering, fluid mechanics, open channel hydraulics, river engineering, water flow engineering, hydrodynamics, and hydraulic structures. He has presented 230 papers in national and international conferences and has published 148 journal papers.