Research Paper
Computational Methods for Differential Equations
http://cmde.tabrizu.ac.ir
Vol. 10, No. 3, 2022, pp. 686-700
DOI:10.22034/cmde.2021.44511.1885

Numerical investigation of the generalized Burgers-Huxley equation using combination of multiquadric quasi-interpolation and method of lines

Maysam Askari[1,*] **and Hojatollah Adibi**[2]

[1]Department of Mathematics, Professor Hesabi Branch, Islamic Azad University, Tafresh, Iran.

[2]Department of Mathematics, Amirkabir University of Technology, Tehran, Iran.

Abstract

In this article, an efficient method for approximating the solution of the generalized Burgers-Huxley (gB-H) equation using a multiquadric quasi-interpolation approach is considered. This method consists of two phases. First, the spatial derivatives are evaluated by MQ quasi-interpolation, So the gB-H equation is reduced to a nonlinear system of ordinary differential equations. In phase two, the obtained system is solved by using ODE solvers. Numerical examples demonstrate the validity and applicability of the method.

Keywords. Generalized Burgers-Huxley equation, Multiquadric quasi-interpolation, Method of lines.

2010 Mathematics Subject Classification. 65L05, 34K06, 34K28.

1. Introduction

The multiquadric radial basis function was first introduced by Hardy in 1968 [11]. He showed that multiquadric can be applied in various branches of science such as hydrology, geology, mining, and digital modeling [12].

The first multiquadric quasi-interpolation was introduced by Powell [23] in 1990 and he showed that this MQ quasi-interpolation has linear reproducing properties. Later, Beatson and Powell [4] introduced three multiquadric quasi-interpolation operators L_A, L_B, and L_C. Coefficients in the operators L_A and L_B only depend on the function values while in L_C operator values of derivatives at the beginning and the end is needed too, so L_C is not convenient for practical cases. In [33] Wu and Schaback presented a multiquadric quasi-interpolation operator L_D which is monotonicity preserving and has a convergence of order $O(h^2 logh)$ if $c = O(h)$. Later, Ling [19] presented a univariate multiquadric quasi-interpolation based on the Wu and Schaback method which was applicable in practice because did not need the values of function derivatives and converged with $O(h^{2.5} logh)$ when $c = O(h)$. Chen et. al [7] introduced a multiquadric quasi-interpolation with preserving monotonicity and linear reproducing properties. They proved that their MQ quasi-interpolation has the accuracy of $O(h^2)$ if $c = O(h^2)$. In [34] the error estimate for the accuracy of Wu-Schaback MQ quasi-interpolation for special classes of functions that have low smoothness was studied. Authors of [17] presented two MQ quasi-interpolation operators with better accuracy than the Wu-Schaback operator . In these schemes, they used the IMQ radial basis function and unlike the RBF method, the ill-conditioning was avoided. Authors in [28] by using Hermite interpolation introduced a MQ quasi-interpolation operator in one-dimensional case. Wu et al. [31] applied multidimensional divided differences to modify Ling [20] approach to provide a family of multivariate MQ quasi-interpolation operators. MQ quasi-interpolation methods are convenient tools for the solution of various engineering issues and have been the focus of researchers for many years [8, 10, 16, 30].

Consider the nonlinear generalized Burgers-Huxley (gB-H) equation

$$u_t + \alpha u^\delta u_x - u_{xx} = \beta u(1 - u^\delta)(u^\delta - \gamma) \tag{1.1}$$

Received: 09 May 2021 ; Accepted: 22 February 2021.
∗ Corresponding author. Email: maysam.askari@gmail.com.

$x \in [a, b],\ t \geq 0$

with the following initial and boundary conditions

$$u(x, 0) = f(x)$$

$$u(a, t) = g_a(t),\ u(b, t) = g_b(t).$$

In [29], exact solution of the gB-H equation by using nonlinear transformations is presented as

$$u(x,t) = \{\frac{\gamma}{2} + \frac{\gamma}{2} tanh\ (\sigma\gamma x - \frac{\sigma\gamma^2\alpha t}{1+\delta} + \frac{\sigma\gamma(1+\delta-\gamma)(\rho-\alpha)t}{2(1+\delta)})\}^{\frac{1}{\delta}} \tag{1.2}$$

where $\alpha,\ \beta,\ \delta \geq 0,\ \gamma \in (0,1),\ \sigma = \frac{\delta(\rho-\alpha)}{4(1+\delta)}$ and $\rho = \sqrt{\alpha^2 + 4\beta(1+\delta)}$.

The gB-H equation, when the parameters are small, has critical roles in nonlinear physics. Note that when $\alpha = 0$ the gB-H equation reduces to Fitzhugh-Nagumo equation and for $\beta = 0,\ \delta = 1$ this equation becomes Burgers equation.

For the solution of Burgers and gB-H equations, various numerical methods were proposed. In [9], FDM and Galerkin method are used to find the solution of Burgers equation. Authors of [15] applied the Newton-Raphson method in connection with FEM for the inviscid Burgers equation. In [2], mixed FDM and Hermite cubic spline multiwavelets are investigated to find the solution of the Burgers equation. Authors of [1] implemented the hyperbolic-trigonometric tension B-spline method to solve the B-H equation and studied the convergence analysis of the method. Bratsos [5] used an implicit fourth-order finite-difference scheme in a two-time level recurrence relation for the numerical solution of the gB-H equation. Authors of [13, 14] used the Adomian decomposition method to solve the gB-H equation. In [27] time and spatial derivatives are approximated by the Euler scheme and cubic B-spline quasi-interpolation respectively to solve the gB-H equation. In [25], Crank-Nicolson FDM and 3-scale Haar wavelets was used to obtain the solution of the gB-H equation. Authors of [18] implemented Chebyshev and Legendre cardinal functions to solve the gB-H equation by the nodal Galerkin method. In [26], modified cubic B-spline differential quadrature method is investigated for the numerical solution of gB-H equation.

In this paper, an efficient method that used multiquadric quasi-interpolation and Method of Lines (MOL) is presented to solve the gB-H equation numerically. For this aim, we applied the multiquadric quasi-interpolation method to approximate the spatial derivatives in the gB-H equation. So, the gB-H equation is reduced to a system of nonlinear ordinary differential equations. By solving this system using ODE solvers such as the Runge-Kutta method we have the approximate solution of the gB-H equation.

The outline of this article is as follows: In section 2, we introduce the multiquadric quasi-interpolation proposed by Chen et al. [7] and examine the accuracy of its derivatives. In section 3, the combination of MQ quasi-interpolation and method of lines are applied for the solution of the gB-H equation. In section 4, numerical results are presented for some examples and we compare these results with exact solutions. Also, we have given a comparison between the presented method and some numerical techniques which were previously proposed. At the end of section 5, we have given the conclusion.

2. Multiquadric quasi-interpolation

Suppose a real function $f(x)$ is defined on the interval $[a, b]$. The univariate multiquadric quasi-interpolation operator $Lf(x)$ on scattered data points $\{x_j\}_{j=0}^m$ where $a = x_0 < x_1 < \cdots < x_m = b$, has the form

$$Lf(x) = \sum_{j=0}^{m} f(x_j)\alpha_j(x) \tag{2.1}$$

where $\alpha_j(x)$ are linear combinations of the MQ radial basis functions. Given a data points $\{x_i, f(x_i)\}_{i=0}^m$, Wu and Schaback [33] presented a multiquadric quasi-interpolation as

$$L_D f(x) = \sum_{j=0}^{m} f(x_j)\alpha_j(x) \tag{2.2}$$

where

$$\alpha_0(x) = \frac{1}{2} + \frac{\phi_1(x) - (x - x_0)}{2(x_1 - x_0)},$$
$$\alpha_1(x) = \frac{\phi_2(x) - \phi_1(x)}{2(x_2 - x_1)} - \frac{\phi_1(x) - (x - x_0)}{2(x_1 - x_0)},$$
$$\alpha_j(x) = \frac{\phi_{j+1}(x) - \phi_j(x)}{2(x_{j+1} - x_j)} - \frac{\phi_j(x) - \phi_{j-1}(x)}{2(x_j - x_{j-1})}, \ j = 2, \cdots, m-2,$$
$$\alpha_{m-1}(x) = \frac{(x_m - x) - \phi_{m-1}(x)}{2(x_m - x_{m-1})} - \frac{\phi_{m-1}(x) - \phi_{m-2}(x)}{2(x_{m-1} - x_{m-2})},$$
$$\alpha_m(x) = \frac{1}{2} + \frac{\phi_{m-1}(x) - (x_m - x)}{2(x_m - x_{m-1})},$$
$$\phi_j(x) = \sqrt{(x - x_j)^2 + c^2}, \ j = 1, \cdots, m-1.$$

Chen et al. [7] introduced the MQ quasi-interpolation as

$$Lf(x) = \sum_{j=0}^{m} f(x_j)\alpha_j(x) \tag{2.3}$$

where

$$\alpha_j(x) = \frac{\phi_{j+1}(x) - \phi_j(x)}{2(x_{j+1} - x_j)} - \frac{\phi_j(x) - \phi_{j-1}(x)}{2(x_j - x_{j-1})}, \ j = 0, \cdots, m,$$
$$\phi_{-1}(x) = \phi_0(x) + x_0 - x_{-1},$$
$$\phi_m(x) = \phi_0(x) - 2x + x_m + x_0,$$
$$\phi_{m+1}(x) = \phi_m(x) + x_{m+1} - x_m,$$

and

$$\phi_j(x) = \sqrt{(x - x_j)^2 + c^2}, \ j = 0, \cdots, m-1.$$

By using simple manipulation this MQ quasi-interpolation can be rewritten as

$$\begin{aligned} Lf(x) = &\frac{1}{2}\sum_{j=1}^{m-1}\left(\frac{\phi_{j+1}(x) - \phi_j(x)}{x_{j+1} - x_j} - \frac{\phi_j(x) - \phi_{j-1}(x)}{x_j - x_{j-1}}\right)f(x_j) \\ &+ \frac{1}{2}\left(1 + \frac{\phi_1(x) - \phi_0(x)}{x_1 - x_0}\right)f(x_0) + \frac{1}{2}\left(1 - \frac{\phi_m(x) - \phi_{m-1}(x)}{x_m - x_{m-1}}\right)f(x_m). \end{aligned} \tag{2.4}$$

The above MQ quasi-interpolation has two other forms as

$$\begin{aligned} Lf(x) = &\frac{1}{2}\sum_{j=1}^{m-1}\left(\frac{f(x_{j+1}) - f(x_j)}{x_{j+1} - x_j} - \frac{f(x_j) - f(x_{j-1})}{x_j - x_{j-1}}\right)\phi_j(x) + \frac{f(x_0) + f(x_m)}{2} \\ &+ \frac{f(x_1) - f(x_0)}{2(x_1 - x_0)}\phi_0(x) - \frac{f(x_m) - f(x_{m-1})}{2(x_m - x_{m-1})}\phi_m(x), \end{aligned} \tag{2.5}$$

and

$$Lf(x) = \frac{f(x_0) + f(x_m)}{2} + \frac{1}{2}\sum_{j=0}^{m-1}\frac{\phi_j(x) - \phi_{j+1}(x)}{x_{j+1} - x_j}(f(x_{j+1}) - f(x_j)). \tag{2.6}$$

In addition, on $x \in [x_0, x_m]$, the derivatives of $Lf(x)$ can be calculated by

$$(Lf(x))^{(k)} = \frac{1}{2}\sum_{j=0}^{m-1} \frac{\phi_j^{(k)}(x) - \phi_{j+1}^{(k)}(x)}{x_{j+1} - x_j}(f(x_{j+1}) - f(x_j)). \tag{2.7}$$

In the following theorems, we mention some attributes of $Lf(x)$:

Theorem 2.1. *The MQ quasi-interpolation $Lf(x)$ has preserving monotonicity and linear reproducing properties on $[x_0, x_m]$ (see [7]).*

Theorem 2.2. *Let $f \in C^2(x_0, x_m)$ and $h = max\{x_j - x_{j-1}\},\ 1 \le j \le m$, then for any real number $c > 0$, we have*

$$\|Lf(x) - f(x)\|_\infty \le k_0 C_h + k_1 h^2 + k_2 ch + k_3 c^2 logh \tag{2.8}$$

where $C_h = min\{c, \frac{c^2}{h}\}$, k_1, k_2 and k_3 are constants independent of h and c (see [7]).

Providing estimation for accuracy of MQ quasi-interpolation derivations has been in focus of attention in recent years [4, 6, 21, 22, 33]. In the following, we investigate about the accuracy of $(Lf)^{(k)}(x)$ for approximating $f^{(k)}(x)$. At first, this issue is studied for $k = 1$.

Theorem 2.3. *Let f be differentiable and $f'(x)$ is Lipschitz continuous, then*

$$\|(Lf)'(x) - f'(x)\|_\infty \le O(\frac{h^2}{c}) + O(\sqrt{h^2 + c^2}) + O(h). \tag{2.9}$$

Proof. Suppose $x_{-1} \le x_0$, $x_0 - x_{-1} \le h$, $x_m \le x_{m+1}$ and $x_{m+1} - x_m \le h$. If we set $f(x_{-1}) = f(x_0)$ and $f(x_{m+1}) = f(x_m)$ then $(Lf)^{(k)}$ can be presented by

$$(Lf)^{(k)}(x) = \frac{1}{2}\sum_{j=0}^{m}(\frac{f(x_{j+1}) - f(x_j)}{x_{j+1} - x_j} - \frac{f(x_j) - f(x_{j-1})}{x_j - x_{j-1}})\phi_j^{(k)}(x). \tag{2.10}$$

For any $x \in [a, b]$, if we define $p(y) = f(x) + f'(x).(y - x)$, then $p(x) = f(x)$. So

$$\begin{aligned} |(Lf)'(x) - f'(x)| &= \frac{1}{2}|\sum_{j=0}^{m}(f[x_j, x_{j+1}] - f[x_{j-1}, x_j])\phi_j'(x) - \sum_{j=0}^{m}(p[x_j, x_{j+1}] - p[x_{j-1}, x_j])\phi_j'(x)| \\ &= \frac{1}{2}|\sum_{j=0}^{m}(f[x_{j-1}, x_j, x_{j+1}] - p[x_{j-1}, x_j, x_{j+1}])(x_{j+1} - x_{j-1})\phi_j'(x)|. \end{aligned} \tag{2.11}$$

Because $p(y)$ is a linear function then $p[x_{j-1}, x_j, x_{j+1}] = 0$. Therefore, we have

$$\begin{aligned} |(Lf)'(x) - f'(x)| &= \frac{1}{2}|\sum_{j=0}^{m} f[x_{j-1}, x_j, x_{j+1}](x_{j+1} - x_{j-1})\phi_j'(x)| \\ &= \frac{1}{2}|\sum_{j=0}^{m}(f[x_j, x_{j+1}] - f[x_{j-1} - x_j])\phi_j'(x)| \\ &\le \sum_{j=0}^{m} |f'(\varepsilon_j) - f'(\eta_j)|\ |\phi_j'(x)| \end{aligned} \tag{2.12}$$

where, $x_j < \varepsilon_j < x_{j+1}$ and $x_{j-1} < \eta_j < x_j$. Now, since $f'(x)$ is Lipschitz continuous then there is a constant M such that

$$|(Lf)'(x) - f'(x)| \le \frac{M}{2}\sum_{j=0}^{m} |\phi_j'(x)|\ |x_{j+1} - x_{j-1}|. \tag{2.13}$$

Hence,

$$\begin{aligned}|(Lf)'(x)-f'(x)| &\le \frac{M}{2}\Big(\sum_{\substack{j=0\\|x-x_j|\le h}}^{m}|\phi_j'(x)|\,|x_{j+1}-x_{j-1}|+\sum_{\substack{j=0\\|x-x_j|>h}}^{m}|\phi_j'(x)|\,|x_{j+1}-x_{j-1}|\Big)\\ &\le \frac{M.h}{2c}\sum_{j=0}^{m}|x_{j+1}-x_{j-1}|+M\int_{|x-t|>h}\frac{|x-t|}{\sqrt{(x-t)^2+c^2}}d_t+O(h)\\ &\le O(\frac{h^2}{c})+O(\sqrt{h^2+c^2})+O(h).\end{aligned} \tag{2.14}$$

□

To state the error bound for $|(Lf)^k(x)-f^k(x)|$, $(k\ge 2)$ we need some preliminaries:

Lemma 2.4. *If $\varphi(x)=\sqrt{x^2+c^2}$, then the k-th-order $(k\ge 2)$ derivatives of $\varphi(x)$ can be bounded as*

$$|\varphi^{(k)}(x)|\le \frac{c_k.c^2}{|x|^{k+1}}, \tag{2.15}$$

and

$$|\varphi^{(k)}(x)|\le \frac{c_k}{c^{k-1}}, \tag{2.16}$$

where c_k is a constant, which depends on k.

Theorem 2.5. *Let $\hat{\psi}(w)$ is the Fourier transform of $\psi(x)$ and*

$$|\hat{\psi}(w)-1|\le O(w^k),\ \ w\to 0. \tag{2.17}$$

If function f(x) has inverse Fourier transform as $f(x)=\int e^{iwx}\hat{f}(w)\mathrm{d}w$, $\int e^{iwx}w^k\mathrm{d}x$ exists, then

$$\|(\psi_\varepsilon * f)(x)-f(x)\|_\infty\le O(\varepsilon^k), \tag{2.18}$$

where $\psi_\varepsilon(x)=\frac{1}{\varepsilon}\psi(\frac{x}{\varepsilon})$ (see [32]).

In [24], authors showed that condition (2.17) can be satisfied by modifying a given ψ. In particular, by a finite linear combination of shifts of the function ψ if $\hat{\psi}\in C^k$ and $\hat{\psi}(0)\ne 0$, one can satisfy the condition (2.17).

Set $\psi(x)=\frac{1}{4(1+x^2)^{\frac{3}{2}}}$ then $\psi_c(x)=\frac{1}{c}\psi(\frac{x}{c})$, so $\frac{1}{c}\psi_c(x)=\frac{\varphi''(x)}{4}$. The fourier transform of $\psi(x)$ satisfies the condition

$$|\hat{\psi}(w)-1|\le O(w^2),\ \ w\to 0.$$

So, by using theorem 2.5, we can get

$$\|\int\frac{f(t).\varphi''(x-t)}{4}\,\mathrm{d}t-f(x)\|_\infty\le O(c^2). \tag{2.19}$$

At this point, we use the approach which is used in [21] to prove the following result:

Theorem 2.6. *Let $k\ge 2$ and assume $f\in C^{(k+2)}([a,b])$ then*

$$|(Lf)^{(k)}(x)-f^{(k)}(x)|\le O(\frac{h}{c^k})+O(\frac{h}{c^{k-1}})+O(c^2) \tag{2.20}$$

Proof. By using Taylor series formula

$$f(x_{j+1})=f(x_j)+(x_{j+1}-x_j)f'(x_j)+\frac{1}{2!}(x_{j+1}-x_j)^2f''(x_j)+\frac{1}{3!}(x_{j+1}-x_j)^3f'''(\xi_{1_j}),$$

$$f(x_{j-1})=f(x_j)+(x_{j-1}-x_j)f'(x_j)+\frac{1}{2!}(x_{j-1}-x_j)^2f''(x_j)+\frac{1}{3!}(x_{j-1}-x_j)^3f'''(\xi_{2_j})$$

where $x_j \leq \xi_{1_j} \leq x_{j+1}$ and $x_{j-1} \leq \xi_{2_j} \leq x_j$, we have

$$\frac{f(x_{j+1})-f(x_j)}{x_{j+1}-x_j}-\frac{f(x_j)-f(x_{j-1})}{x_j-x_{j-1}}=\frac{1}{2}(x_{j+1}-x_{j-1})f''(x_j)+\frac{1}{6}\{(x_{j+1}-x_j)^2f'''(\xi_{1_j})-(x_j-x_{j-1})^2f'''(\xi_{2_j})\}. \quad (2.21)$$

According to the relation (2.10), we get

$$(Lf)^{(k)}(x)=\frac{1}{4}\sum_{j=0}^{m}(x_{j+1}-x_{j-1})f''(x_j).\phi_j^{(k)}(x)+\frac{1}{12}\sum_{j=0}^{m}\{(x_{j+1}-x_j)^2f'''(\xi_{1_j})-(x_j-x_{j-1})^2f'''(\xi_{2_j})\}\phi_j^{(k)}(x). \quad (2.22)$$

Let

$$I=|\int_a^b\frac{f''(t)\varphi^{(k)}(x-t)}{4}\,\mathrm{d}t-\sum_{j=0}^{m}\frac{(x_{j+1}-x_{j-1})f''(x_j)\phi_j^{(k)}(x)}{4}|.$$

By using the integral mean value theorem, we have

$$\begin{aligned}I=&|\sum_{j=1}^{m-1}\int_{x_{j-1}}^{x_{j+1}}\frac{f''(t)\varphi^{(k)}(x-t)}{4}\,\mathrm{d}t-\sum_{j=0}^{m}\frac{(x_{j+1}-x_{j-1})f''(x_j)\phi_j^{(k)}(x)}{4}|\\ &\leq|\sum_{j=1}^{m-1}\frac{(x_{j+1}-x_{j-1})}{4}\{f''(\gamma_j)\varphi^{(k)}(x-\gamma_j)-f''(x_j)\phi_j^{(k)}(x)\}|\\ &\quad+\frac{|(x_1-x_{-1})f''(x_0)\phi_0^{(k)}(x)}{4}|+|\frac{(x_{m+1}-x_{m-1})f''(x_m)\phi_m^{(k)}(x)}{4}|\end{aligned}$$

where $x_{j-1}\leq\gamma_j\leq x_{j+1}$. By noting that $\phi_j^{(k)}(x)=\varphi^{(k)}(x-x_j)$ and by utilizing Taylor series expansion, we get

$$\begin{aligned}I\leq&|\sum_{j=1}^{m-1}\frac{(x_{j+1}-x_{j-1})}{4}(\gamma_j-x_j)\{f'''(x_j)\varphi^{(k)}(x-x_j)-f''(x_j)\varphi^{(k+1)}(x-\delta_j)\}|\\ &+\frac{|(x_1-x_{-1})f''(x_0)\phi_0^{(k)}(x)}{4}|+|\frac{(x_{m+1}-x_{m-1})f''(x_m)\phi_m^{(k)}(x)}{4}|,\end{aligned}$$

where δ_j is a real number between γ_j and x_j. Applying estimation (2.16) leads to

$$I\leq\frac{h}{c^{k-1}}(\frac{1}{2}c_k(b-a)\|f'''\|_\infty+\frac{c_{k+1}}{2c}(b-a)\|f''\|_\infty+c_k\|f''\|_\infty). \quad (2.23)$$

Also, by using the mean value theorem besides estimation (2.16), we can deduce the following inequality

$$|\sum_{j=0}^{m}\{(x_{j+1}-x_j)^2f'''(\xi_{1_j})-(x_j-x_{j-1})^2f'''(\xi_{2_j})\}\phi_j^{(k)}(x)|\leq\frac{4h(b-a+h)c_k\|f'''\|_\infty}{c^{k-1}}. \quad (2.24)$$

If we define the function $f(x)$ in $\mathbb{R}\setminus[a,b]$ as $f(x)=0$, by using equation (2.19) and integration by parts formula, we can get

$$\|\int_a^b\frac{f''(t)\varphi^{(k)}(x-t)}{4}\,\mathrm{d}t-f^{(k)}(x)\|_\infty\leq O(c^2). \quad (2.25)$$

Now, utilizing equations (2.23-2.25) gives the following error estimation

$$\|(Lf)^{(k)}(x)-f^{(k)}(x)\|_\infty\leq O(\frac{h}{c^k})+O(\frac{h}{c^{k-1}})+O(c^2). \quad (2.26)$$

□

3. Construction of method using MQ quasi-interpolation and MOL

Consider the nonlinear gB-H equation

$$u_t + \alpha u^\delta u_x - u_{xx} = \beta u(1-u^\delta)(u^\delta - \gamma), \qquad x \in [a,b],\ t \geq 0 \tag{3.1}$$

with the following initial and boundary conditions

$$u(x,0) = f(x)$$
$$u(a,t) = g_a(t),\ u(b,t) = g_b(t).$$

In each node $x_i : i = 1, \cdots, m-1$ where $a = x_0 < x_1 < \cdots, x_m = b$ the nonlinear generalized Burgers-Huxley equation can be rewritten as

$$u_t(x_i,t) = -\alpha u^\delta(x_i,t)u_x(x_i,t) + u_{xx}(x_i,t) + \beta u(x_i,t)(1-u^\delta(x_i,t))(u^\delta(x_i,t) - \gamma). \tag{3.2}$$

According to the MQ quasi-interpolation $Lf(x)$ in the previous section, we can consider the following approximations for the spatial derivatives as

$$u_x(x_i,t) \simeq \frac{1}{2}\sum_{j=0}^{m-1} \frac{\phi_j'(x_i) - \phi_{j+1}'(x_i)}{x_{j+1}-x_j}(u(x_{j+1},t) - u(x_j,t)), \tag{3.3}$$

and

$$u_{xx}(x_i,t) \simeq \frac{1}{2}\sum_{j=0}^{m-1} \frac{\phi_j''(x_i) - \phi_{j+1}''(x_i)}{x_{j+1}-x_j}(u(x_{j+1},t) - u(x_j,t)), \tag{3.4}$$

where

$$\phi_m(x) = \phi_0(x) - 2x + x_m + x_0,$$
$$\phi_j(x) = \sqrt{(x-x_j)^2 + c^2},\ 0 \leq j \leq m-1.$$

If we set

$$C_{i,j} = \frac{\phi_j'(x_i) - \phi_{j+1}'(x_i)}{2(x_{j+1}-x_j)}, \tag{3.5}$$

$$D_{i,j} = \frac{\phi_j''(x_i) - \phi_{j+1}''(x_i)}{2(x_{j+1}-x_j)},\ i = 1, \cdots, m,\ j = 0, \cdots, m-1, \tag{3.6}$$

the equation (3.2) reduces to

$$\begin{aligned} u_t(x_i,t) = &-\alpha u^\delta(x_i,t)\sum_{j=0}^{m-1} C_{i,j}(u(x_{j+1},t) - u(x_j,t)) \\ &+ \sum_{j=0}^{m-1} D_{i,j}(u(x_{j+1},t) - u(x_j,t)) + \beta u(x_i,t)(1-u^\delta(x_i,t))(u^\delta(x_i,t) - \gamma). \end{aligned} \tag{3.7}$$

Since

$$\begin{aligned} \sum_{j=0}^{m-1} C_{i,j}(u(x_{j+1},t) - u(x_j,t)) = &- C_{i,0}u(x_0,t) + C_{i,m-1}u(x_m,t) \\ &+ (C_{i,0} - C_{i,1})u(x_1,t) + \cdots + (C_{i,m-2} - C_{i,m-1})u(x_{m-1},t), \end{aligned} \tag{3.8}$$

and

$$\begin{aligned} \sum_{j=0}^{m-1} D_{i,j}(u(x_{j+1},t) - u(x_j,t)) = &- D_{i,0}u(x_0,t) + D_{i,m-1}u(x_m,t) \\ &+ (D_{i,0} - D_{i,1})u(x_1,t) + \cdots + (D_{i,m-2} - D_{i,m-1})u(x_{m-1},t), \end{aligned} \tag{3.9}$$

so from equations (3.8) and (3.9), we have

$$\begin{aligned} u_t(x_i,t) =& -\alpha u^\delta(x_i,t)[(C_{i,0}-C_{i,1})u(x_1,t)+\cdots+(C_{i,m-2}-C_{i,m-1})u(x_{m-1},t)] \\ & -\alpha u^\delta(x_i,t)[-C_{i,0}u(x_0,t)+C_{i,m-1}u(x_m,t)] \\ & +(D_{i,0}-D_{i,1})u(x_1,t)+\cdots+(D_{i,m-2}-D_{i,m-1})u(x_{m-1},t) \\ & -D_{i,0}u(x_0,t)+D_{i,m-1}u(x_m,t)+\beta u(x_i,t)(1-u^\delta(x_i,t))(u^\delta(x_i,t)-\gamma). \end{aligned} \tag{3.10}$$

We now collocate equation (3.10) in $m-1$ points $x_i;\ i=1\cdots m-1$ and hence we can obtain a system of nonlinear ODEs as

$$\frac{dU}{dt} = G(U), \tag{3.11}$$

$$G(U) = -\alpha U*(AU+V)+BU+W+\beta U*(I-U^\delta)*(U^\delta-\Gamma) \tag{3.12}$$

where

$$A_{i,j}=C_{i,j-1}-C_{i,j},\ B_{i,j}=D_{i,j-1}-D_{i,j},$$
$$V_i=-C_{i,0}u(x_0,t)+C_{i,m-1}u(x_m,t),\ W_i=-D_{i,0}u(x_0,t)+D_{i,m-1}u(x_m,t),$$

and

$$I=[1,\cdots,1]^T,\ \Gamma=[\gamma,\cdots,\gamma]^T.$$

By solving this system of ODEs using numerical methods we get the approximate solution for the nonlinear gB-H equation.

4. Numerical Results

In this section, we provide the numerical results of MQQI-MOL method which was presented in previous section. To measure the accuracy of the described method, we use two norms

$$L_2 = \|u^{exact}-u^{approximate}\|_2 = \sqrt{h\sum_{j=0}^{m}|u_j^{exact}-u_j^{approximate}|^2}$$

$$L_\infty = \|u^{exact}-u^{approximate}\|_\infty = \max_{0\le j\le m}|u_j^{exact}-u_j^{approximate}|.$$

In all examples spatial domain is $[0,1]$, the shape parameter is considered as $c=0.02$ and in our computations we take $\Delta t=0.001$. For simplicity the nodes in spatial domains is selected as $x_i=i*\Delta x,\ i=0,\cdots,m$ where Δx is the spatial step size e.g. $\Delta x = x_i-x_{i-1},\ i=1,\cdots,m$ and m=20. Here, for the solution of the system of ODEs we use the classical fourth-order Runge-Kutta method. The computations have been performed in MAPLE 18 software.

Example 4.1. As the first example, we take the gB-H Eq. (1.1) with the coefficients $\alpha=0$, $\beta=1$, $\delta=1$, and $\gamma=0.001,\ 0.0001$. The L_2 and L_∞ errors of the presented method in different values of t are listed in Tables 1 and 2. In Figure 1, we have plotted the exact and numerical solutions at $t=1$. Figure 2 displays the absolute error of the presented method for the gB-H equation in domain $0\le x\le 1$ and $0\le t\le 1$. Moreover, to show the accuracy and efficiency of the presented method, comparisons among errors of MQQI-MOL method, fourth order improved finite difference Scheme [5], Gauss Chebyshev Galerkin method [18], El-Gendi Legendre Galerkin method [18], and modified cubic B-spline differential quadrature method [26] are given in Table 3.

Table 1. L_2 and L_∞ errors for the approximate solutions of Example 4.1 with $\gamma=0.001$ at some time levels.

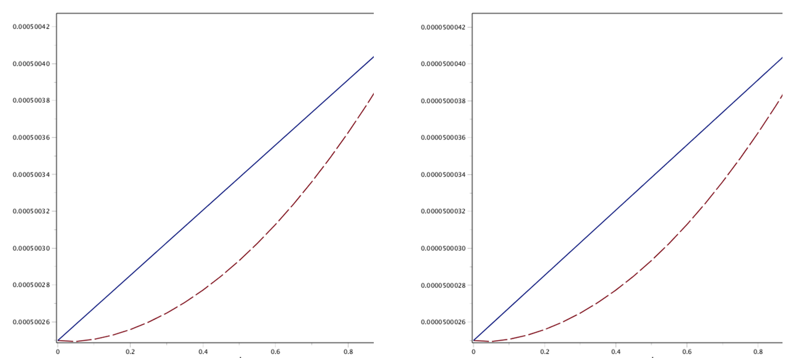
Time	t=0.1	t=0.3	t=0.5	t=0.7	t=1
L_∞	3.3220E-08	4.4251E-08	4.4963E-08	4.5009E-08	4.5011E-08
L_2	2.4614E-08	3.2434E-08	3.2939E-08	3.2971E-08	3.2973E-08

TABLE 2. L_2 and L_∞ errors for the approximate solutions of Example 4.1 with $\gamma = 0.0001$ at some time levels.

Time	t=0.1	t=0.3	t=0.5	t=0.7	t=1
L_∞	3.3231E-10	4.4265E-10	4.4974E-10	4.5022E-10	4.5025E-10
L_2	2.4622E-10	3.2444E-10	3.2947E-10	3.2979E-10	3.2984E-10

TABLE 3. Comparisons of the presented method for Example 4.1 with $\gamma = 0.001$

t	x	MQQI-MOL	MCB-DQM[26]	FDS4[5]	GCG[18]	ELG[18]
0.05	0.1	9.3519E-09	1.0044E-08	2.4988E-08	1.0698E-08	9.2752E-09
	0.5	2.1614E-08	2.3047E-08	2.4988E-08	9.2595E-09	9.2595E-09
	0.9	8.8872E-09	1.0044E-08	2.4987E-08	7.8921E-09	9.2845E-09
0.1	0.1	1.3152E-08	1.4790E-08	4.9975E-08	2.3188E-08	2.3173E-08
	0.5	3.3220E-08	3.8252E-08	4.9975E-08	2.1749E-08	2.1749E-08
	0.9	1.2450E-08	1.4790E-08	4.9975E-08	2.0382E-08	2.0360E-08
1	0.1	1.7009E-08	2.2205E-08	4.9975E-07	2.4872E-07	2.4729E-07
	0.5	4.5011E-08	6.2169E-08	4.9975E-07	2.4728E-07	2.4728E-07
	0.9	1.6065E-08	2.2205E-08	4.9975E-07	2.4591E-07	2.4530E-07



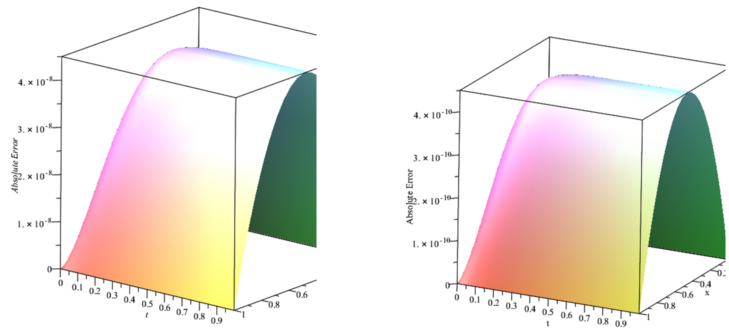
(a) Plot of the exact and numerical solutions for B-H equation with $\gamma = 0.001$.

(b) Plot of the exact and numerical solutions for B-H equation with $\gamma = 0.0001$.

FIGURE 1. The comparison between exact and approximate solution for B-H equation with the coefficients $\alpha = 0$, $\beta = 1$, $\delta = 1$ in interval $0 \leq x \leq 1$ at $t = 1$.

Example 4.2. Here, let's Consider the gB-H Eq. (1.1) with $\alpha = 0$, $\beta = 1$, $\delta = 2$ and $\gamma = 0.001, 0.0001$. In Table 4 and Table 5 the L_2 and L_∞ errors of our method at t=0.1, 0.3, 0.5, 0.7, and 1 are reported. According to Figures 3 and 4, we have plotted the graph of exact and approximate solutions of gB-H equation at some of the values of $0 \leq t \leq 1$. In Table 6 we compare the results of our method with some of the numerical methods to exhibit the reliability and effectiveness of the proposed method.

Example 4.3. We take the gB-H Eq. (1.1) with $\alpha = 1$, $\beta = 1$, $\delta = 1$, and $\gamma = 0.001, 0.0001$. The L_2 and L_∞ errors of the presented method in different values of t are listed in Tables 7 and 8. In Figure 5, we have plotted the



(a) Plot of the absolute error for B-H equation with $\gamma = 0.001$.

(b) Plot of the absolute error for B-H equation with $\gamma = 0.0001$.

FIGURE 2. The absolute error for B-H equation with the coefficients $\alpha = 0$, $\beta = 1$, $\delta = 1$ in domain $0 \le x \le 1$ and $0 \le t \le 1$.

TABLE 4. L_2 and L_∞ errors for the approximate solutions of Example 4.2 with $\gamma = 0.001$ at some time levels.

Time	t=0.1	t=0.3	t=0.5	t=0.7	t=1
L_∞	1.4857E-06	1.9790E-06	2.0105E-06	2.0126E-06	2.0125E-06
L_2	1.1008E-06	1.4505E-06	1.4729E-06	1.4743E-06	1.4743E-06

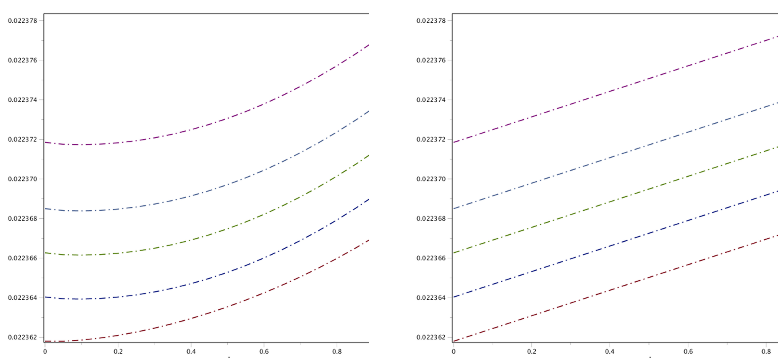
TABLE 5. L_2 and L_∞ errors for the approximate solutions of Example 4.2 with $\gamma = 0.0001$ at some time levels.

Time	t=0.1	t=0.3	t=0.5	t=0.7	t=1
L_∞	4.6993E-08	6.2599E-08	6.3601E-08	6.3648E-08	6.3649E-08
L_2	3.4818E-08	4.5883E-08	4.6592E-08	4.6626E-08	4.6625E-08

exact and numerical solutions at $t = 1$. In Figure 6, the error of the approximate solution at some of the values of t is given. Table 9 exhibits the results of MQQI-MOL method compared with some of the numerical methods such as fourth order improved finite difference Scheme [5], Adomian decomposition method [13, 14], and variational iteration method [3].

TABLE 6. Comparisons of the presented method for Example 4.2 with $\gamma = 0.001$

t	x	MQQI-MOL	MCB-DQM[26]	FDS4[5]	GCG[18]	ELG[18]
0.05	0.1	4.1828E-07	4.4924E-07	1.1176E-06	4.8110E-07	4.2020E-07
	0.5	9.6666E-07	1.0307E-06	1.1175E-06	3.9966E-07	3.9966E-07
	0.9	3.9745E-07	4.4917E-07	1.1174E-06	3.9240E-07	4.3297E-07
0.1	0.1	5.8819E-07	6.6147E-07	2.2353E-06	1.0397E-06	9.7883E-07
	0.5	1.4857E-06	1.7107E-06	2.2350E-06	9.5823E-07	9.5823E-07
	0.9	5.5674E-07	6.6139E-07	2.2347E-06	9.5091E-07	9.9147E-07
1	0.1	7.6055E-07	9.9267E-07	2.2353E-05	1.1021E-05	1.1008E-05
	0.5	2.0125E-06	2.7793E-06	2.2350E-05	1.1057E-05	1.1057E-05
	0.9	7.1824E-07	9.9260E-07	2.2347E-05	1.0841E-05	1.0955E-05



(a) Plot of approximate solution at the some of values of $0 \leq t \leq 1$.

(b) Plot of exact solution at the some of values of $0 \leq t \leq 1$.

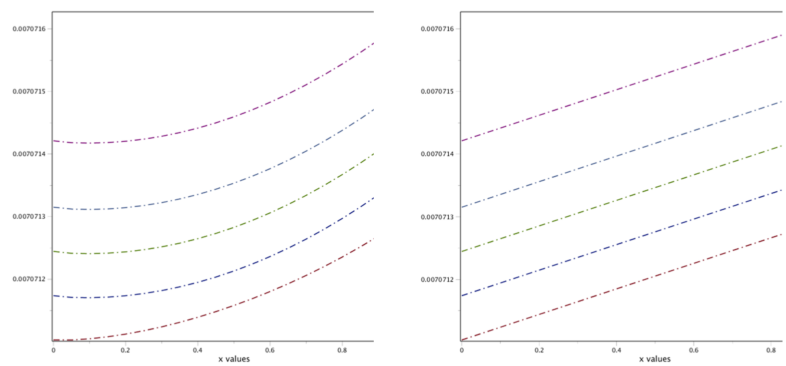
FIGURE 3. Comparison between the approximate and exact solutions for B-H equation with the coefficients $\alpha = 0$, $\beta = 1$, $\delta = 2$, and $\gamma = 0.001$.

TABLE 7. L_2 and L_∞ errors for the approximate solutions of Example 4.3 with $\gamma = 0.001$ at some time levels.

Time	t=0.1	t=0.3	t=0.5	t=0.7	t=1
L_∞	2.4936E-08	3.3219E-08	3.3753E-08	3.3788E-08	3.3792E-08
L_2	1.8482E-08	2.4354E-08	2.4733E-08	2.4757E-08	2.4760E-08

TABLE 8. L_2 and L_∞ errors for the approximate solutions of Example 4.3 with $\gamma = 0.0001$ at some time levels.

Time	t=0.1	t=0.3	t=0.5	t=0.7	t=1
L_∞	2.4943E-10	3.3220E-10	3.3750E-10	3.3765E-10	3.3775E-10
L_2	1.8486E-10	2.4357E-10	2.4729E-10	2.4741E-10	2.4748E-10



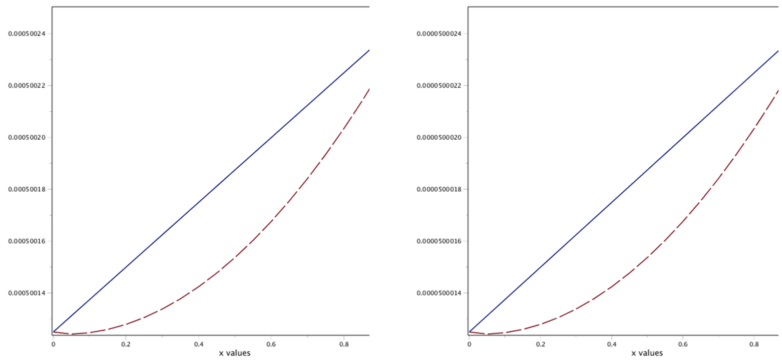
(a) Plot of approximate solution at the some of values of $0 \leq t \leq 1$.

(b) Plot of exact solution at the some of values of $0 \leq t \leq 1$.

FIGURE 4. Comparison between the approximate and exact solutions for B-H equation with the coefficients $\alpha = 0$, $\beta = 1$, $\delta = 2$, and $\gamma = 0.0001$.

TABLE 9. Comparisons of the presented method for Example 4.3 with $\gamma = 0.001$

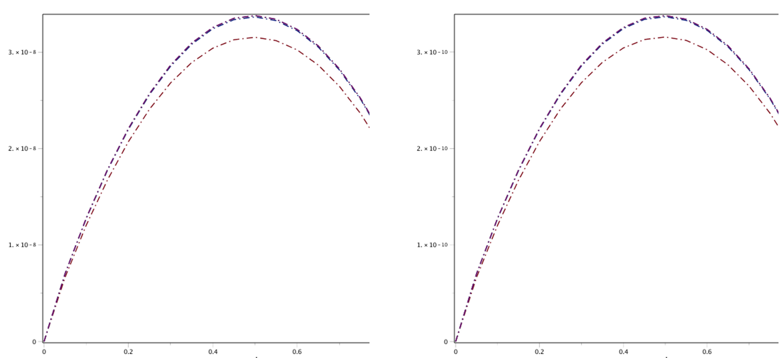
t	x	MQQI-MOL	FDS4[5]	ADM[14]	ADM[13]	VAM[3]
0.05	0.1	7.0382E-09	1.2646E-09	1.9372E-07	1.8741E-08	1.8741E-08
	0.5	1.6222E-08	1.9770E-08	1.9373E-07	1.8741E-08	1.8741E-08
	0.9	6.6906E-09	4.6018E-08	1.9375E-07	1.8741E-08	1.8741E-08
0.1	0.1	9.8913E-09	6.3953E-09	3.8743E-07	3.7481E-08	3.7481E-08
	0.5	2.4936E-08	3.9956E-08	3.8746E-07	3.7481E-08	1.3748E-08
	0.9	9.3660E-09	7.6633E-08	3.8749E-07	3.7481E-08	3.7481E-08
1	0.1	1.2788E-08	3.2922E-07	3.8750E-06	3.7481E-07	3.7481E-07
	0.5	3.3792E-08	3.7922E-07	3.8753E-06	3.7481E-07	3.7481E-07
	0.9	1.2081E-08	4.2922E-07	3.8756E-06	3.7481E-07	3.7481E-07



(a) Plot of the exact and numerical solutions for B-H equation with $\gamma = 0.001$.

(b) Plot of the exact and numerical solutions for B-H equation with $\gamma = 0.0001$.

FIGURE 5. The comparison between exact and approximate solution for B-H equation with the coefficients $\alpha = 1$, $\beta = 1$, $\delta = 1$ in interval $0 \leq x \leq 1$ at $t = 1$.



(a) The error of the approximate solution for B-H equation with $\gamma = 0.001$ at $t = 0.2, 0.4, 0.6, 0.8$ and 1.

(b) The error of the approximate solution for B-H equation with $\gamma = 0.001$ at $t = 0.2, 0.4, 0.6, 0.8$ and 1.

FIGURE 6. The error of the approximate solution for B-H equation with the coefficients $\alpha = 1$, $\beta = 1$, $\delta = 1$ in interval $0 \leq x \leq 1$ at different time levels.

5. Conclusion

The multiquadric quasi-interpolation method that uses the method of lines has been described. To improve the MQ quasi-interpolation accuracy, we use method of lines. In conventional methods for the solution of PDEs researchers use MQ quasi-interpolation and finite difference method for spatial and time derivatives respectively. Because these methods do not have satisfactory accuracy, we combine MQ quasi-interpolation approach with method of lines. In the first step the spatial derivatives are approximated by MQ quasi-interpolation operator. So the PDE reduced to a system of ODEs. Then we applied the classical fourth-order Runge-Kutta scheme for solving this system. From comparison the results of the presented method to other methods such as ADM, VIM, Galerkin method, finite difference method, and modified cubic B-spline differential quadrature method, it is observed that our method is reliable and posses high accuracy.

Acknowledgment

Authors would like to thank Islamic Azad University, professor Hesabi Branch for supporting this research project.

References

[1] N. Alinia and M. Zarebnia, *A numerical algorithm based on a new kind of tension B-spline function for solving Burgers-Huxley equation*, Numer. Algorithms., *82* (2019), 1121–1142.
[2] E. Ashpazzadeh, B. Han, and M. Lakestani, *Biorthogonal multiwavelets on the interval for numerical solutions of Burgers' equation*, J. Comput. Appl. Math., *317* (2017), 510–534.
[3] B. Batiha, M. S. M. Noorani, and I. Hashim, *Application of variational iteration method to the generalized Burgers–Huxley equation*, Chaos. Solitons. Fractals., *36* (2008), 660–663.
[4] R. K. Beatson and M. J. D. Powell, *Univariate multiquadric approximation: quasi interpolation to scattered data*, Constr. Approx., *8* (1992), 275–288.
[5] A. G. Bratsos, *A fourth order improved numerical scheme for the generalized Burgers–Huxley equation*, Am. J. Comput. Math., *1* (2011), 152–158.
[6] M. D. Buhmann, *Radial Basis Functions: Theory and implementations*, Cambridge University Press, 2003.
[7] R. Chen, X. Han, and Z. Wu, *A multiquadric quasi-interpolation with linear reproducing and preserving monotonicity*, J. Comput. Appl. Math., *231* (2009), 517–525.
[8] R. Chen and Z. Wu, *Applying multiquadric quasi-interpolation to solve Burgers' equation*, Appl. Math. Comput., *172* (2006), 472–484.
[9] M. Dehghan, B. Nemati Saray, and M. Lakestani, *Mixed finite difference and Galerkin methods for solving Burgers equations using interpolating scaling functions*, Math. Methods. Appl. Sci., *37* (2014), 894–912.
[10] F. Gao and C. Chi, em Numerical solution of nonlinear Burgers' equation using high accuracy multi-quadric quasi-interpolation, Appl. Math. Comput., *229* (2014), 414–421.
[11] R. L. Hardy, *Multiquadric equations of topography and other irregular surfaces*, J. Geophys. Res., *176* (1971), 1905–1915.
[12] R. L. Hardy, *Theory and applications of the multiquadric-biharmonic method, 20 years of discovery 1968-1988*, Comput. Math. Appl., *19* (1990), 163–208.
[13] I. Hashim, M. S. M. Noorani, and M. R. S. Al-Hadidi, *Solving the generalized Burgers-Huxley equation using the Adomian decomposition method*, Math. Comput. Model., *43* (2006), 1404–1411.
[14] H. N. A. Ismail, K. Raslan, and A. A. A. Rabboh, *Adomian Decomposition Method for Burger's Huxley and Burger's-Fisher Equations*, Appl. Math. Comput., *159* (2004), 291–301.
[15] M. Izadi, *Applications of the Newton-Raphson method in a SDFEM for inviscid Burgers equation*, Computational Methods for Differential Equations, *8* (2020), 708–732.
[16] Z. Jiang and R. Wang, *Numerical solution of one-dimensional Sine–Gordon equation using high accuracy multiquadric quasi-interpolation*, Appl. Math. Comput., *218* (2012), 7711–7716.
[17] Z. Jiang, R. Wang, C. Zhu and X. Min, *High accuracy multiquadricquasi-interpolation*, Appl. Math. Model., *35* (2011), 2185–2195.

[18] M. El-Kady, S. M. El-Sayed, and H. E. Fathy, *Development of Galerkin method for solving the generalized Burger's-Huxley equation*, Math. Probl. Eng., (2013), 1–9.

[19] L. Ling, *A univariate quasi-multiquadric interpolation with better smoothness*, Comput. Math. Appl., *48* (2004), 897–912.

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

[21] L. Ma and Z. Wu, *Approximation to the k-th derivatives by multiquadric quasi-interpolation*, J. Comput. Appl. Math., *231* (2009), 925–932.

[22] L. Ma, Z. Wu, *Stability of multiquadric quasi-interpolation to approximate high order derivatives*, Sci. China Math., *53* (2009), 985–992.

[23] M. J. D. Powell, *Univariate multiquadric approximation: reproduction of linear polynomials, in: W. Haussman, K. Jetter (Eds.), Multivariate Approximation and Interpolation*, Birkhauser Verlag, 1990, 227–240

[24] R. Schaback and Z. Wu, *Construction techniques for highly accurate quasi-interpolation operators*, J. Approx. theory., *91* (1997), 320–331.

[25] S. Shukla and M. Kumar, *Error analysis and numerical solution of Burgers-Huxley equation using 3-scale Haar wavelets*, Eng. Comput., *36* (2020).

[26] B. K. Singh, G. Arora, and M. K. Singh, *A numerical scheme for the generalized Burgers–Huxley equation*, J. Egyptian Math. Soc., *24* (2016), 629–637.

[27] L. Y. Sun and C. G. Zhu, *Cubic B-spline quasi-interpolation and an application to numerical solution of generalized Burgers-Huxley equation*, Adv. Mech. Eng., *12* (2020), 1–8.

[28] R. Wang, X. Min, and Q. Fang, *A kind of improved univariate multiquadric quasi-interpolation operators*, Comput. Math. Appl., *59* (2010), 451–456.

[29] X. Y. Wang, Z. S. Zhu, and Y. K. Lu, *Solitary wave solutions of the generalized Burgers-Huxley equation*, J. Phys. A: Math. Gen., *23* (1990), 271–274.

[30] H. Wu and Y. Duan, *Multi-quadric quasi-interpolation method coupled with FDM for the Degasperis–Procesi equations*, Appl. Math. Comput., *274* (2016), 83–92.

[31] R. Wu, T. Wu, and H. Li, *A family of multivariate multiquadric quasi-interpolation operators with higher degree polynomial reproduction*, J. Comput. Appl. Math., *274* (2015), 88–108.

[32] Z. Wu and J. P. Liu, *Generalized Strang-Fix condition for scattered data quasi-iterpolation*, Adv. Comput. Math., *23* (2005), 201–214.

[33] Z. Wu and R. Schaback, *Shape preserving properties and convergence of univariate multiquadric quasi-interpolation*, Acta. Math. Appl. Sin. Engl. Ser., *10* (1994), 441–446.

[34] Y. Zhang, X. Liang, and Q. Li, *A further research on the convergence of Wu-Schaback's multi-quadric quasi-interpolation*, J. Appl and Computat. Math., *2* (2013), 1–3.