Arab. J. Math. (2018) 7:303–315
https://doi.org/10.1007/s40065-018-0197-5

Arabian Journal of Mathematics

B. Sepehrian · Z. Shamohammadi

A high order method for numerical solution of time-fractional KdV equation by radial basis functions

Received: 19 March 2017 / Accepted: 28 January 2018 / Published online: 20 February 2018

Abstract A radial basis function method for solving time-fractional KdV equation is presented. The Caputo derivative is approximated by the high order formulas introduced in Buhman (Proc. Edinb. Math. Soc. 36:319–333, 1993). By choosing the centers of radial basis functions as collocation points, in each time step a nonlinear system of algebraic equations is obtained. A fixed point predictor–corrector method for solving the system is introduced. The efficiency and accuracy of our method are demonstrated through several illustrative examples. By the examples, the experimental convergence order is approximately $4-\alpha$, where α is the order of time derivative.

Mathematics Subject Classification 65D05 · 65M06 · 65N06 · 65N35

الملخص

يتم عرض طريقة الأساس الشعاعي للدوال وذلك لحل معادلة Kdv ذات مشتقات زمنية كسرية. نقوم بتقريب مشتقات كبوتو بصيغ ذات رتب عالية والتي قدمت في [4]. وباختيار مراكز الأساس الشعاعي للدوال كنقاط تجميع، نحصل في كل خطوة زمنية على نظام غير خطي لمعادلات جبرية. ولحل هذه الأنظمة، نستعمل طريقة النقطة الثابتة مع تنبأ وتصحيح. وقد ثبتت كفاءة ودقة طريقتنا بواسطة أمثلة توضيحية. ومن خلال هذه الأمثلة، تبين أن معدل التقارب التجريبي هو $4-\alpha$ تقريبا، حيث أن α هي رتبة المشتق الزمني.

1 Introduction

In this paper, we consider the time-fractional KdV equation

$$\frac{\partial^\alpha u(x,t)}{\partial t^\alpha} + \varepsilon\big(u(x,t)\big)^m \frac{\partial u(x,t)}{\partial x} + \nu \frac{\partial^3 u(x,t)}{\partial x^3} = f(x,t), \quad x \in \Omega = [a,b], \quad t \geq 0, \tag{1}$$

with the following initial condition:

$$u(x,0) = g(x), \quad x \in \Omega, \tag{2}$$

B. Sepehrian (✉) · Z. Shamohammadi
Department of Mathematics, Faculty of Science, Arak University, 38156-8-8349 Arak, Iran
E-mail: b-sepehrian@araku.ac.ir

Table 1 Some well-known functions that generate RBFs

Name of function	Definition
Gaussian (GA)	$\phi(r) = \exp(-cr^2)$
Hardy multiquadric (MQ)	$\phi(r) = \sqrt{r^2 + c^2}$
Inverse multiquadric (IMQ)	$\phi(r) = (\sqrt{r^2 + c^2})^{-1}$
Inverse quadric (IQ)	$\phi(r) = (r^2 + c^2)^{-1}$

and boundary conditions:

$$u(a,t) = h_1(t), \quad t \geq 0, \tag{3}$$
$$u(b,t) = h_2(t), \quad t \geq 0, \tag{4}$$

where ε and ν are the positive parameters, and $h_1(t)$, $h_2(t)$ and $g(x)$ are known functions. Also, $\frac{\partial^\alpha u}{\partial t^\alpha}$ is the Caputo fractional derivative and $0 < \alpha < 1$. The αth order Caputo fractional derivative of function F is defined as

$$D^\alpha_\star F(t) = \begin{cases} \frac{1}{\Gamma(k-\alpha)} \int_0^t (t-\xi)^{k-1-\alpha} F^{(k)}(\xi) d\xi, & k-1 < \alpha < k, \quad t > 0, \\ F^{(k)}(t), & \alpha = k. \end{cases} \tag{5}$$

The Korteweg–de Vries equation was initially introduced to describe the lossless propagation of shallow water waves [17]. Also, it has been applied in plasma physics, radar, rheology [25,32], optical-fiber communications [37], super conductors [10], etc. Several numerical and analytical methods for solving fractional KdV equations have been introduced. In [29], the variational iteration method for the space- and time-fractional KdV equation was applied. These equations were investigated by Adomian method in [39]. The homotopy perturbation method has been used for fractional KdV equations in [38]. A numerical technique based on generalized Taylor series formula has been presented in [13]. For more information, for example, see [1,18,28,33,35].

The radial basis functions methods are a family of meshless methods. There are other meshless methods. Some of them are as follows: element-free Galerkin method, reproducing kernel particle method, point interpolation method, boundary element-free method, moving least-square approximation, etc. The meshless methods are new and interesting numerical techniques. They can solve many problems in physics and engineering that are not suited to conventional numerical methods, with a minimum of meshing or no meshing at all [2]. For more study about these methods, for example, see [3,9,14,16,19–21,40].

The radial basis function methods are highly flexible and are efficient especially for solving problems with arbitrary geometry [30]. Furthermore, the RBF methods are performed without any mesh generation and they are usually more accurate than low order methods, such as finite differences, finite volumes and finite elements. To see the applications of RBFs in the numerical solution of partial differential equations (PDEs) and fractional PDEs, for example, see [7,11,15,24,27,36].

In Table 1, some well-known globally supported RBFs have been listed. Let r be the Euclidean distance between $x^* \in \mathbb{R}^d$ and any $x \in \mathbb{R}^d$, i.e., $\|x - x^*\|_2$. A radial function $\phi^* = \phi(\|x - x^*\|_2)$ depends only on the distance between $x \in \mathbb{R}^d$ and fixed point $x^* \in \mathbb{R}^d$. Hence, the RBF ϕ^* is radially symmetric about x^*. Clearly, the functions in Table 1 are globally supported, infinitely differentiable and depend on a free parameter c. The parameter c is called shape parameter. It allows us to broaden the solution space outside of the polynomial space. Many researches are done for finding the optimal values of c which produce most accurate interpolation (e.g., see [5,8]). But this is still an open problem. In practice, the shape parameter c is chosen experimentally. In fact, the problem is solved by the different values of c. Those values of c which give convergent sequences of approximations are suitable.

The global infinitely differentiable RBFs can be used for interpolating smooth data with a spectral accuracy [4,5,23,26]. More information about the accuracy of the approximations made by RBFs can be found in [12,41]. Let $x_1, x_2, \ldots, x_M$ be a given set of distinct points in $\mathbb{R}^d$. The idea behind the use of RBFs is interpolation by translations of a single function, i.e.,

$$h(x) = \sum_{i=1}^{M} \lambda_i \phi_i(x), \tag{6}$$

where $\phi_i(x) = \phi(\|x - x_i\|)$ and λ_i are unknown scalars for $i = 1, \ldots, M$. For interpolating the given values $f_i = f(x_i)$, $i = 1, \ldots, M$, the unknown scalars λ_i are found so that $h(x_j) = f_j$ for $j = 1, \ldots, M$. So, the

following linear system of equations is obtained

$$AY = b, \tag{7}$$

where $A = [a_{i,j}]$ with $a_{i,j} = \phi_i(x_j)$ for $1 \leq i, j \leq M$, $Y = [\lambda_1, \ldots, \lambda_M]^T$ and $b = [f_1, \ldots, f_M]^T$. Since all ϕ of the interest have global support, this method produces a dense matrix A. The matrix A corresponding to GS, IMQ and IQ for distinct interpolation points is positive definite, and therefore nonsingular [34].
In the above cases, the condition number of A

$$\kappa_s(A) = \|A\|_s \|A^{-1}\|_s, \quad s = 1, 2, \infty, \tag{8}$$

is usually a very large number and A is very ill-conditioned. Therefore, in our computations more precision arithmetics than the standard floating point arithmetic must be used.

In this work, the high order difference formulas introduced in [6] are applied for discretizing on time variable. In each time step, the solution of Eqs. (1)–(4) is approximated by a linear combination of RBFs with unknown coefficients. For finding these coefficients, we choose the centers of RBFs as collocation points, and consequently a nonlinear system of algebraic equations is obtained. The nonlinear system cannot be easily solved by the ordinary methods (for example, Newton method). So a fixed point iteration method for solving the resulted nonlinear system is proposed. By the fixed point method, the computations of the nonlinear system are reduced to some linear systems of equations.

The organization of the paper is as follows: In Sect. 2, a fixed point iteration method for solving the systems of nonlinear algebraic equations is introduced. In Sect. 3, the solution of Eq. (1) by RBFs is considered. Section 4 is devoted to numerical experiments.

2 Fixed point method

By the fixed point method first the system of nonlinear equations,

$$\mathbf{F(X)} = \mathbf{0}, \tag{9}$$

is rearranged as

$$\mathbf{G(X)} = \mathbf{X}, \tag{10}$$

for $\mathbf{F}, \mathbf{G} : \mathbb{R}^n \to \mathbb{R}^n$. Then by a suitable initial approximation $\mathbf{X}_0 \in \mathbb{R}^n$ and the following recursive formula, successive approximations for the solution of (9) are given

$$\mathbf{X}_{k+1} = \mathbf{G}(\mathbf{X}_k), \ \ k = 0, 1, \ldots \tag{11}$$

We will use the fixed point method as follows. First, we decompose (9) into an invertible linear ($\mathbf{F}_1$) and a nonlinear ($\mathbf{F}_2$) operators as

$$\mathbf{F}_1(\mathbf{X}) + \mathbf{F}_2(\mathbf{X}) = 0. \tag{12}$$

Then

$$\mathbf{F}_1(\mathbf{X}) = -\mathbf{F}_2(\mathbf{X}), \tag{13}$$

which results

$$\mathbf{X} = -\mathbf{F}_1^{-1}(\mathbf{F}_2(\mathbf{X})). \tag{14}$$

Therefore, with the formulation,

$$\mathbf{X}_{k+1} = -\mathbf{F}_1^{-1}(\mathbf{F}_2(\mathbf{X}_k)), \tag{15}$$

the desired sequence is obtained.
The method is a predictor–corrector method. There is a sufficient condition for the convergence of the iteration formula (11) [31].

Theorem 2.1 Suppose that $G : D \subset \mathbb{R}^n \to \mathbb{R}$ has a fixed point $\mathbf{X}^*$ in the interior of D and that G is continuously differentiable in a neighborhood of $\mathbf{X}^*$. Denote by J_G the Jacobian matrix of G and assume that $\rho(J_G(\mathbf{X}^*)) < 1$. Then there exists a neighborhood S of $\mathbf{X}^*$ such that $S \subset D$ and for any $\mathbf{X}_0 \in S$ the iterates defined by (11) all lie in D and converge to $\mathbf{X}^*$ [31].

3 Description of the method

First, we discretize Eq. (1) in the time direction as

$$\frac{\partial^\alpha u(x,t^n)}{\partial t^\alpha}+\varepsilon\Big(u(x,t^n)\Big)^m\frac{\partial u(x,t^n)}{\partial x}+\nu\frac{\partial^3 u(x,t^n)}{\partial x^3}=f(x,t^n),\quad n=1,2,3,\ldots, \tag{16}$$

where $u(x,t^n)=u^n, t^n=n\tau, n=1,2,\ldots,N$, the time step τ, and the time length $N\tau$. The value of $\frac{\partial^\alpha u(x,t^n)}{\partial t^\alpha}$ for $n=1$, $n=2$ and $n\geq 3$ is obtained as follows: [6]

$$\frac{\partial^\alpha u(x,t^1)}{\partial t^\alpha}=\mu a_0(u^1-u^0)+O(\tau^{2-\alpha}), \tag{17}$$

$$\frac{\partial^\alpha u(x,t^2)}{\partial t^\alpha}=\mu\Big[(b_0-a_1)u^0+(a_1-a_0-2b_0)u^1+(a_0+b_0)u^2\Big]+O(\tau^{3-\alpha}), \tag{18}$$

$$\frac{\partial^\alpha u(x,t^n)}{\partial t^\alpha}=\mu\Big[(b_{n-2}-a_{n-1})u^0+(a_{n-1}-a_{n-2}-2b_{n-2})u^1+(a_{n-2}+b_{n-2})u^2+\sum_{k=3}^{n-1}\Big(w_{1,n-k}u^k+$$
$$w_{2,n-k}u^{k-1}+w_{3,n-k}u^{k-2}+w_{4,n-k}u^{k-3}\Big)+w_{1,0}u^n+w_{2,0}u^{n-1}+w_{3,0}u^{n-2}+w_{4,0}u^{n-3}\Big]+O(\tau^{4-\alpha}), \tag{19}$$

in which

$$\begin{aligned}
\mu&=\frac{\tau^{-\alpha}}{\Gamma(2-\alpha)},\\
a_i&=(i+1)^{1-\alpha}-i^{1-\alpha},\\
b_i&=\frac{(i+1)^{2-\alpha}-i^{2-\alpha}}{2-\alpha}-\frac{(i+1)^{1-\alpha}+i^{1-\alpha}}{2},\\
w_{1,n-k}&=\frac{1}{6}\Big[2(n-k+1)^{1-\alpha}-11(n-k)^{1-\alpha}\Big]-\frac{1}{2-\alpha}\Big[2(n-k)^{2-\alpha}-(n-k+1)^{2-\alpha}\Big]\\
&\quad-\frac{1}{(2-\alpha)(3-\alpha)}\Big[(n-k)^{3-\alpha}-(n-k+1)^{3-\alpha}\Big],\\
w_{2,n-k}&=\frac{1}{2}\Big[6(n-k)^{1-\alpha}+(n-k+1)^{1-\alpha}\Big]+\frac{1}{2-\alpha}\Big[5(n-k)^{2-\alpha}-2(n-k+1)^{2-\alpha}\Big]\\
&\quad+\frac{3}{(2-\alpha)(3-\alpha)}\Big[(n-k)^{3-\alpha}-(n-k+1)^{3-\alpha}\Big],\\
w_{3,n-k}&=-\frac{1}{2}\Big[3(n-k)^{1-\alpha}+2(n-k+1)^{1-\alpha}\Big]-\frac{1}{2-\alpha}\Big[4(n-k)^{2-\alpha}-(n-k+1)^{2-\alpha}\Big]\\
&\quad-\frac{3}{(2-\alpha)(3-\alpha)}\Big[(n-k)^{3-\alpha}-(n-k+1)^{3-\alpha}\Big],
\end{aligned}$$

and

$$\begin{aligned}
w_{4,n-k}&=\frac{1}{6}\Big[2(n-k)^{1-\alpha}+(n-k+1)^{1-\alpha}\Big]+\frac{1}{2-\alpha}(n-k)^{2-\alpha}+\frac{1}{(2-\alpha)(3-\alpha)}\times\\
&\quad\times\Big[(n-k)^{3-\alpha}-(n-k+1)^{3-\alpha}\Big].
\end{aligned}$$

By Eqs. (16)–(19), the following finite differences equations are obtained:

$$\mu u^1+\nu\frac{\partial^3 u^1}{\partial x^3}=-\epsilon(u^1)^m\frac{\partial u^1}{\partial x}+\mu u^0+f^1, \tag{20}$$

$$\mu(a_0+b_0)u^2+\nu\frac{\partial^3 u^2}{\partial x^3}=-\epsilon(u^2)^m\frac{\partial u^2}{\partial x}-\mu\Big[(b_0-a_1)u^0+(a_1-a_0-2b_0)u^1\Big]+f^2, \tag{21}$$

and

$$\mu w_{1,0}u^n + \nu\frac{\partial^3 u^n}{\partial x^3} = -\epsilon(u^n)^m\frac{\partial u^n}{\partial x} - \mu\Big[(b_{n-2}-a_{n-1})u^0 + (a_{n-1}-a_{n-2}-2b_{n-2})u^1 + (a_{n-2}+b_{n-2})u^2 + \sum_{k=3}^{n-1}\Big(w_{1,n-k}u^k + w_{2,n-k}u^{k-1} + w_{3,n-k}u^{k-2} + w_{4,n-k}u^{k-3}\Big) + w_{2,0}u^{n-1} + w_{3,0}u^{n-2} + w_{4,0}u^{n-3}\Big] + f^n, \quad n = 3,4,5,\ldots \tag{22}$$

Now using the radial basis functions, we consider the solution $u(x,t^n)$ as

$$u^n(x) = \sum_{j=1}^{M}\lambda_j^n\phi(\|x-x_j\|), \tag{23}$$

where $\lambda_j^n,\ j=1,\ldots,M$ are unknown.

To construct the approximations for $u^1(x)$, first we substitute (23) in (20) and in boundary conditions (3) and (4), and then we collocate the resulted equations. For suitable collocation points, we choose the centers, $x_i,\ i=1,\ldots,M$ $\big(x_i = a+(i-1)\Delta x,\ \Delta x = \frac{b-a}{M-1}\big)$, as collocation points. Thus, a nonlinear system of M equations in M unknowns is obtained. By solving the system, $\lambda_j^1,\ j=1,\ldots,M$ is computed. Similarly, $u^2(x)$ is obtained by substituting (23) in (21), (3) and (4), and using the collocation method with the same collocation points. Inductively, $u^n(x),\ n=3,4,5,\ldots$ is approximated by Eqs. (23), (22), (3) and (4) and with the same technique.

We solve the resulted nonlinear systems by utilizing the fixed point method presented in Sect. 2, as follows: In nth, $n=1,2,\ldots$ time step, the unknown vector, $\Lambda^n = \left[\lambda_1^n,\lambda_2^n,\ldots,\lambda_M^n\right]^T$, is given by

$$\tilde{B}^n\Lambda_{k+1}^n = \tilde{F}^n(\Lambda_k^n), \quad k=0,1,2,\ldots, \tag{24}$$

with a suitable choice of Λ_0^n. In (24), $\tilde{F}^n = \left[\tilde{F}_1^n,\tilde{F}_2^n,\ldots,\tilde{F}_M^n\right]^T$ is a function of Λ^n and for $n=1$, $n=2$ and $n\geq 3$ is as follows:

For $i=2,\ldots,M-1$,

$$\tilde{F}_i^1 = -\varepsilon\Big(\sum_{j=1}^{M}\lambda_j^1\phi(\|x_i-x_j\|)\Big)^m\Big(\sum_{j=1}^{M}\lambda_j^1\frac{\partial\phi(\|x_i-x_j\|)}{\partial x}\Big) + \mu u_i^0 + f_i^1,$$

$$\tilde{F}_i^2 = -\varepsilon\Big(\sum_{j=1}^{M}\lambda_j^2\phi(\|x_i-x_j\|)\Big)^m\Big(\sum_{j=1}^{M}\lambda_j^2\frac{\partial\phi(\|x_i-x_j\|)}{\partial x}\Big) - \mu\Big[(b_0-a_1)u_i^0 + (a_1-a_0-2b_0)u_i^1\Big] + f_i^2,$$

$$\tilde{F}_i^l = -\varepsilon\Big(\sum_{j=1}^{M}\lambda_j^l\phi(\|x_i-x_j\|)\Big)^m\Big(\sum_{j=1}^{M}\lambda_j^l\frac{\partial\phi(\|x_i-x_j\|)}{\partial x}\Big) - \mu\Big[(b_{l-2}-a_{l-1})u_i^0 + (a_{l-1}-a_{l-2}-2b_{l-2})u_i^1 + (a_{l-2}+b_{l-2})u_i^2 + \sum_{k=3}^{l-1}\Big(w_{1,l-k}u_i^k + w_{2,l-k}u_i^{k-1} + w_{3,l-k}u_i^{k-2} + w_{4,l-k}u_i^{k-3}\Big) + w_{2,0}u_i^{l-1} + w_{3,0}u_i^{l-2} + w_{4,0}u_i^{l-3}\Big] + f_i^l, \quad l=3,4,5,\ldots,$$

and for i=1 and $i=M$, we have

$$\begin{aligned} F_1^n &= h_1(t^n),\\ F_M^n &= h_2(t^n), \end{aligned} \quad n=0,1,2,\ldots,$$

where h_1 and h_2 are the boundary conditions (3) and (4), respectively.

Also, in (24) the matrix $\tilde{B}^n = [\tilde{B}^n_{ij}]$ for n=1, n=2 and $n \geq 3$ is as follows: for $i = 2, \ldots, M-1$ and $j = 1, \ldots, M$,

$$\tilde{B}^1_{ij} = \mu\phi(\|x_i - x_j\|) + \nu\frac{\partial^3\phi(\|x_i - x_j\|)}{\partial x^3},$$
$$\tilde{B}^2_{ij} = \mu(a_0 + b_0)\phi(\|x_i - x_j\|) + \nu\frac{\partial^3\phi(\|x_i - x_j\|)}{\partial x^3},$$
$$\tilde{B}^l_{ij} = \mu w_{1,0}\phi(\|x_i - x_j\|) + \nu\frac{\partial^3\phi(\|x_i - x_j\|)}{\partial x^3}, \ l = 3, 4, \ldots,$$

and for i=1 and $i = M$,

$$\tilde{B}^n_{ij} = \phi(\|x_i - x_j\|), \quad j = 1, \ldots, M, \ n = 1, 2, \ldots$$

Clearly, for $n \geq 3$, the matrix $\tilde{B}^n$ is unchanged.

We solve (24) by LU decomposition of $\tilde{B}^n$. By the method, we write (24) as

$$LU\Lambda^n_{k+1} = \tilde{F}^n(\Lambda^n_k), \quad k = 0, 1, 2, \ldots,$$

where L and U are the lower and upper triangular matrices, respectively. This factorization needs $O(M^3)$ number of operations.

Using the forward substitution with $O(M^2)$ number of operations, we solve

$$Lz^n_{k+1} = \tilde{F}^n(\Lambda^n_k), \quad k = 0, 1, 2, \ldots,$$

and then we find Λ^n_{k+1} by solving

$$U\Lambda^n_{k+1} = z^n_{k+1}, \quad k = 0, 1, 2, \ldots,$$

using the backward substitution which also needs $O(M^2)$ number of operations.

By substituting the values of λ^n_j, $j = 1, 2, \ldots, M$ obtained by (24) in Equ. (23), the values of unknown function u^n, $n = 1, 2, \ldots$ are computed.

4 Numerical illustrations

In this section, three test examples have been presented to show the efficiency of the mentioned method. We performed our computations using **Maple 16** software. In all results, $\delta = 50$ floating point arithmetics are used in our computations. Solution of the resulted nonlinear systems is obtained via the corresponding fixed point iteration method. Here, we consider stop condition of fixed point iteration method as follows:

$$\|\Lambda_{k+1} - \Lambda_k\|_\infty < \xi,$$

where all results are obtained with $\xi = 10^{-5}$. Also, we have $\rho(J_G(X^*)) < 1$ and the fixed point method introduced in Sect. 2, gives convergent sequences. The errors and the experimental convergence order ($C-$Order) are calculated as follows:

$$E_\infty = \left\|u_{\text{exact}}(x, t^N) - u_{\text{approx}}(x, t^N)\right\|_\infty = \max_{1\leq i\leq M}\left|u_{\text{exact}}(x_i, t^N) - u_{\text{approx.}}(x_i, t^N)\right|,$$
$$E_2 = \sqrt{\sum_{i=1}^{M}\left(u_{\text{exact}}(x_i, t^N) - u_{\text{approx.}}(x_i, t^N)\right)^2},$$
$$\text{RMSE} = \sqrt{\frac{1}{M}\sum_{i=1}^{M}\left(u_{\text{exact}}(x_i, t^N) - u_{\text{approx.}}(x_i, t^N)\right)^2},$$
$$C - \text{Order} = \log_2\left(\frac{E_\infty(\Delta x, 2\tau)}{E_\infty(\Delta x, \tau)}\right).$$

Table 2 The E_∞ error and $C-$Order for various values of τ resulted by IMQ-RBFs and IQ-RBFs with $\Delta x = 0.05$ and $\alpha = 0.4$ for $[a, b] = [-2, 2]$ and $t = 1$ in Example 1

τ	IMQ ($c = 0.65$)		IQ ($c = 0.85$)	
	E_∞	$C-$Order	E_∞	$C-$Order
0.1	0.0030250581	–	0.0030908834	–
0.05	0.0002939023	3.36	0.0002998101	3.37
0.025	0.0000257615	3.51	0.0000270202	3.47
0.0125	0.0000021659	3.57	0.0000022385	3.60

4.1 Example 1

Consider the time-fractional KdV equation (1) with $m = 2$ as

$$\frac{\partial^\alpha u(x,t)}{\partial t^\alpha} + \varepsilon(u(x,t))^2\frac{\partial u(x,t)}{\partial x} + \nu\frac{\partial^3 u(x,t)}{\partial x^3} = f(x,t), \quad x \in \Omega = [a,b], \quad t \geq 0,$$

where

$$f(x,t) = t^5\text{sech}(x)\Big[\frac{\Gamma(6+\alpha)}{120} + \epsilon\Big(t^{5+\alpha}\text{sech}(x)\Big)^2\Big(-t^\alpha\tanh(x)\Big) + \nu\Big(-t^\alpha(\tanh(x))^3 + 5t^\alpha(\text{sech}(x))^2\tanh(x)\Big)\Big],$$

with the initial and boundary conditions,

$$\begin{aligned} u(x,0) &= 0, \quad a \leq x \leq b,\\ u(a,t) &= t^{5+\alpha}\text{sech}(a), \quad t \geq 0,\\ u(b,t) &= t^{5+\alpha}\text{sech}(b), \quad t \geq 0. \end{aligned}$$

The exact solution of this problem is $u(x,t) = t^{5+\alpha}\text{sech}(x)$.
We solved the problem by the new method and using GS-RBFs, IQ-RBFs, MQ-RBFs and IMQ-RBFs. Also, we put $\varepsilon = 6$, $\nu = 1$ (except in Table 6).
Table 2 presents the E_∞ errors and the experimental convergence orders ($C-$Order) obtained by our method with $M = 21$. In this case, $\rho(J_G(X^*)) < 0.37$. Table 2 shows the $C-$Order is approximately $4-\alpha$. Table 2, demonstrates that as τ becomes smaller the smaller errors are obtained. Also, with $\tau = 0.1, 0.05, 0.025$ and 0.0125, and respectively, 10, 20, 40 and 80 iterations the very small errors are obtained. So, we can conclude the method has good stability for solving the problem.
In Table 3, we report the E_∞, E_2 errors and $\rho(J_G(X^*))$ of present method using GS and MQ-RBFs for different values of α. Table 3 shows that the smaller the values of α, the more accurate results are obtained.

In Table 4, we report the E_∞, E_2, RMSE errors and condition number of matrix $B = \tilde{B}^n$, $n \geq 3$ resulted by 160 iterations of the new method, for some values of shape parameter c at $t = 1$. It seems that the optimal value of c is in the interval [2.3, 2.55]. But it cannot be found exactly.
Table 5 presents the E_∞, E_2, RMSE errors and $\kappa_\infty(B)$ for different values of M. In this case $\rho(J_G(X^*)) < 0.26$. Table 5 shows that as the number of RBFs becomes larger, the smaller errors are obtained while the condition number of B becomes larger. So, the dimension of matrix B should be small sufficiently to guarantee the stability of the solution.
Table 6 presents the E_∞ and RMSE errors for $\varepsilon = 1$ and various values of ν resulted by GS-RBFs and MQ-RBFs. As Table 6 shows, the larger the values of ν the more accurate results are obtained.
We plot the approximate solution resulted by 200 iterations of the mentioned method and $M = 41$ IQ-RBFs with $\tau = 0.005$ and $c = 1.25$ in Fig. 1. In Fig. 2, we plot the absolute error function resulted by $M = 61$ GS-RBFs with $c = 2$ for $[a, b] = [-6, 6]$.

Table 3 The E_∞, E_2 errors and $\rho(J_G(X^*))$ for different values of α using GS-RBFs and MQ-RBFs with $M = 51$ and $\tau = 0.01$ for $[a, b] = [-5, 5]$ and $t = 1$ in Example 1

α	GS ($c = 5$)			MQ ($c = 0.8$)		
	E_∞	E_2	$\rho(J_G(X^*))$	E_∞	E_2	$\rho(J_G(X^*))$
0.2	0.000022417	0.000058570	0.806610	0.000002820	0.000008631	0.807604
0.4	0.000027675	0.000070908	0.472866	0.000003212	0.000008830	0.472941
0.6	0.000034225	0.000077392	0.268719	0.000003926	0.000013131	0.268720
0.8	0.000041702	0.000094000	0.149252	0.000009820	0.000030618	0.149252

Table 4 The errors and condition number of matrix B for some values of shape parameter c with IMQ-RBFs, $M = 31, \tau = 0.00625$ and $\alpha = 0.25$ for $[a, b] = [-3, 3]$ and $t = 1$ in Example 1

c	E_∞	E_2	RMSE	κ_∞(B)
1.9	0.0000110685	0.0000315256	0.0000056622	3.768×10^{10}
2.1	0.0000079236	0.0000225640	0.0000040526	6.256×10^{11}
2.3	0.0000051073	0.0000145253	0.0000026088	1.019×10^{13}
2.5	0.0000008600	0.0000024029	0.0000004316	1.626×10^{14}
2.55	0.0000047404	0.0000129730	0.0000023300	3.264×10^{14}

Table 5 The errors and condition number of matrix B for different values of M with MQ-RBFs, $c = 0.5, \tau = 0.01, \alpha = 0.7$ and $[a, b] = [-1, 1]$ at $t = 1$ in Example 1

M	E_∞	E_2	RMSE	κ_2(B)
11	0.001665233	0.002781064	0.000838522	53218.73
21	0.000071393	0.000201506	0.000043972	5.5475×10^6
31	0.000013403	0.000045834	0.000008232	1.5048×10^9
41	0.000002098	0.000008872	0.000001386	2.1196×10^{12}

Table 6 The E_∞ and RMSE errors resulted by $M = 21$ MQ-RBFs and GS-RBFs for $\alpha = 0.3, \tau = 0.0125$ and $[a, b] = [-1, 1]$ at $t = 1$ in Example 1

ν	GS ($c = 10$)		MQ ($c = 0.35$)	
	E_∞	RMSE	E_∞	RMSE
0.8	0.0000838	0.0000496	0.0002283	0.0001309
0.08	0.0001624	0.0000792	0.0005567	0.0002542
0.008	0.0011131	0.0004075	0.0023742	0.0008331

4.2 Example 2

Consider the time-fractional KdV Eq. (1) with $\varepsilon = 1$, $\nu = 1$ and $m = 1$ as

$$\frac{\partial^\alpha u(x,t)}{\partial t^\alpha} + u(x,t)\frac{\partial u(x,t)}{\partial x} + \frac{\partial^3 u(x,t)}{\partial x^3} = f(x,t), \quad x \in \Omega = [a,b], \quad t \geq 0,$$

where

$$\begin{aligned} u(x,0) &= 0, \quad a \leq x \leq b, \\ u(a,t) &= \frac{1}{3000}\, t^5 e^{-a^2}, \quad t \geq 0, \\ u(b,t) &= \frac{1}{3000}\, t^5 e^{-b^2}, \quad t \geq 0, \end{aligned}$$

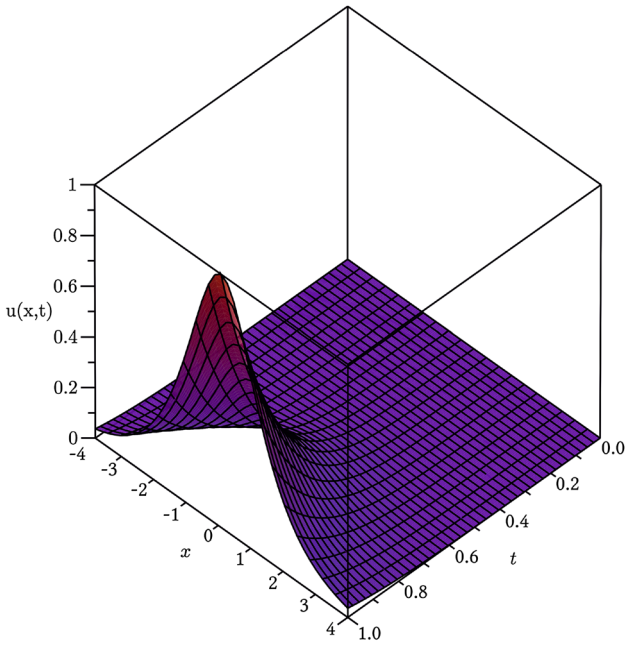

Fig. 1 Plot of approximate solution of Example 1 for $\alpha = 0.5$, $\tau = 0.005$ and $[a, b] = [-4, 4]$ obtained by IQ-RBFs

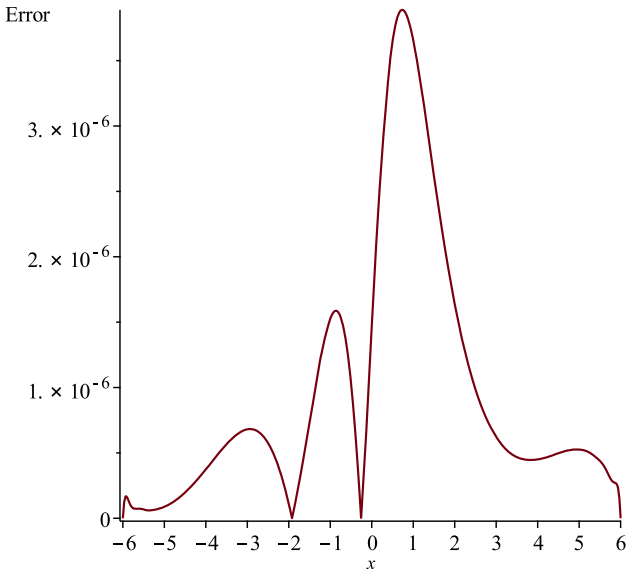

Fig. 2 Plot of the absolute error function for $\alpha = 0.25$, $\tau = 0.02$ and $t = 1$ resulted by GS-RBFs for Example 1

and

$$f(x,t) = \frac{t^5 e^{-x^2}}{25}\left(\frac{1}{\Gamma(6-\alpha)}t^{-\alpha} - \frac{1}{180000}\,t^2 x e^{-x^2} + \frac{1}{10}x - \frac{1}{15}x^3\right).$$

The exact solution is $u(x,t) = \frac{1}{3000}\,t^5 e^{-x^2}$. Table 7 demonstrates the errors of numerical approximations resulted by our method and $M = 31$ MQ-RBFs with $\tau = 0.2, 0.1, 0.05$ and 0.025, at $t = 1, 2, 3, 4$ and 5. The table shows that the method has good stability.

Table 8 depicts the errors of the approximations resulted by the new method with different values of M, further to listing the corresponding condition number of the matrix B. Table 8 shows that as the number of RBFs becomes larger, the more accurate approximations are obtained while $\kappa_\infty(B)$ becomes larger.

In Fig. 3, we have plotted the absolute error function in $-3 \le x \le 3$ for $0 \le t \le 1$ and $5 \le t \le 6$.

Table 7 The errors for various values of τ with $M = 31$ MQ-RBFs, $c = 1.65$, $\alpha = 0.2$ and $[a, b] = [-2, 2]$ for Example 2

t	τ	E_∞	E_2	RMSE
1	0.2	0.000001181903	0.000004718098	0.000000847395
	0.1	0.000000109650	0.000000437872	0.000000078644
	0.05	0.000000009263	0.000000037061	0.000000006656
	0.025	0.000000000848	0.000000003418	0.000000000613
2	0.2	0.000003507537	0.000014006351	0.000002515615
	0.1	0.000000296557	0.000001185333	0.000000212892
	0.05	0.000000027133	0.000000109052	0.000000019586
	0.025	0.000000006775	0.000000024663	0.000000004430
3	0.2	0.000006335980	0.000025152204	0.000004517469
	0.1	0.000000537880	0.000002141989	0.000000384712
	0.05	0.000000073507	0.000000284115	0.000000051029
	0.025	0.000000041402	0.000000145407	0.000000026116
4	0.2	0.000009682858	0.000037560465	0.000006746059
	0.1	0.000000883940	0.000003446326	0.000000618978
	0.05	0.000000209624	0.000000766061	0.000000137589
	0.025	0.000000164183	0.000000571970	0.000000102728
5	0.2	0.000013970715	0.000050850179	0.000009132962
	0.1	0.000001471858	0.000005391754	0.000000968388
	0.05	0.000000555648	0.000001929136	0.000000346483
	0.025	0.000000495057	0.000001683494	0.000000302364

Table 8 The errors and condition number of matrix B of Example 2 with IMQ-RBFs, $c = 1.65$, $\tau = 0.05$, $\alpha = 0.6$ for $t = 1$ and $[a, b] = [-1, 1]$

M	E_∞	E_2	RMSE	κ_∞(B)
6	0.0000123496	0.0000197114	0.0000080471	9798.70
8	0.0000078253	0.0000147976	0.0000052317	1.669×10^5
10	0.0000053096	0.0000114379	0.0000036170	4.659×10^6
12	0.0000034965	0.0000083600	0.0000024133	2.053×10^8
14	0.0000021502	0.0000055556	0.0000014848	1.150×10^{10}
16	0.0000011086	0.0000030738	0.0000007684	7.208×10^{11}

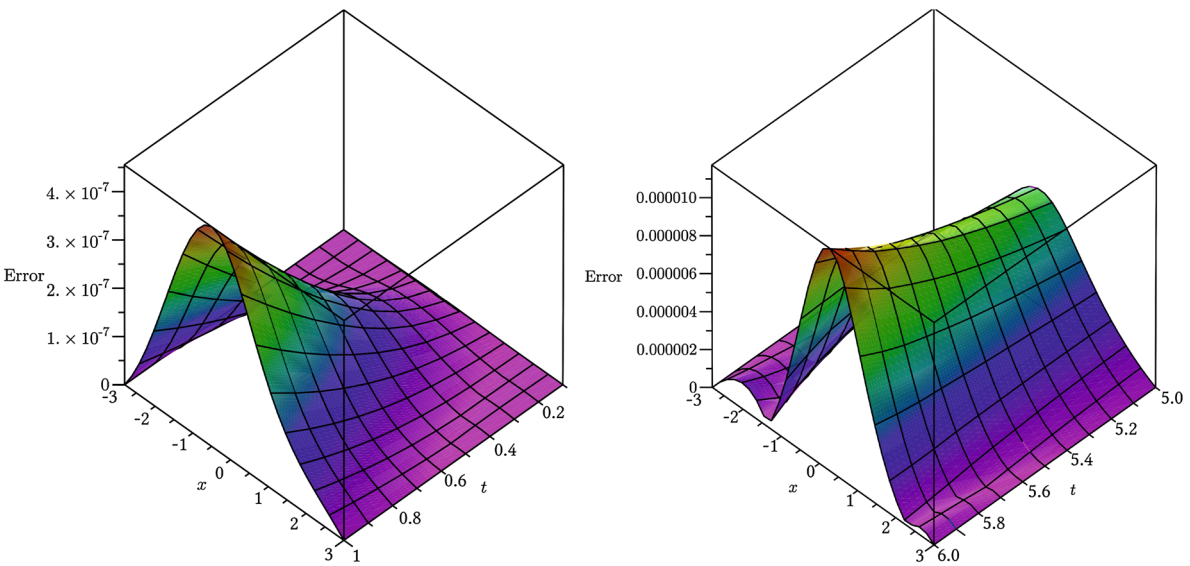

Fig. 3 Plot of the absolute error function in $-3 \leq x \leq 3$ for $0 \leq t \leq 1$ (left) and $5 \leq t \leq 6$ (right) with $M = 31$, $\alpha = 0.5$, $\tau = 0.1$ and $c = 3.5$ using GS-RBFs in Example 2

Table 9 The E_∞, RMSE errors and $C-$Order for various values of τ obtained by $M_1 = M_2 = 9$ and MQ-RBFs with $c = 1.95$ for $\alpha = 0.2$, $[a_1, a_2] = [b_1, b_2] = [-1, 1]$ and $t = 1$ in Example 3

τ	E_∞	E_2	RMSE	$C-$Order
0.2	0.0157580814	0.0685740388	0.0076193376	–
0.1	0.0016749121	0.0073042110	0.0008115790	3.23
0.05	0.0001501838	0.0006546922	0.0000727435	3.48
0.025	0.0000126669	0.0000548206	0.0000060912	3.57

Table 10 The errors and $\rho(J_G(X^\star))$ and condition number of B resulted by IQ-RBFs with $c = 1.95$ for $\tau = 0.02$, $\alpha = 0.35$, $[a_1, a_2] = [b_1, b_2] = [-1, 1]$ and $t = 1$ in Example 3

$M_1 = M_2$	E_∞	E_2	RMSE	$\rho(J_G(X^\star))$	κ_2(B)
5	0.0004712163	0.0006946442	0.0001389288	0.57	2.6187×10^6
7	0.0000270059	0.0000755743	0.0000107963	0.35	6.5033×10^9
9	0.0000134963	0.0000571351	0.0000063483	0.37	1.1075×10^{13}

4.3 Example 3

We consider

$$\frac{\partial^\alpha u(x,y,t)}{\partial t^\alpha} + 6(u(x,y,t))^2\frac{\partial u(x,y,t)}{\partial x} + \frac{\partial^3 u(x,y,t)}{\partial x^3} + \frac{\partial}{\partial x}\left(\frac{\partial^2 u(x,y,t)}{\partial y^2}\right) = f(x,y,t),$$

for $x \in [a_1, a_2]$, $y \in [b_1, b_2]$ and $t \geq 0$, where

$$f(x,t) = t^6\text{sech}(x)\text{sech}(y)\Big[\frac{720}{\Gamma(7-\alpha)}t^{-\alpha} + \tanh(x)\Big(-6\Big(t^6\text{sech}(x)\text{sech}(y)\Big)^2 - \tanh^2(x) + 5\text{sech}^2(x) - \tanh^2(y) + \text{sech}^2(y)\Big)\Big],$$

and the initial and boundary conditions are as

$$\begin{aligned}
&u(x,y,0) = 0, \quad a_1 \leq x \leq a_2, \quad b_1 \leq y \leq b_2,\\
&u(a_1,y,t) = t^{5+\alpha}\text{sech}(a_1)\text{sech}(y), \quad t \geq 0,\\
&u(a_2,y,t) = t^{5+\alpha}\text{sech}(a_2)\text{sech}(y), \quad t \geq 0,\\
&u(x,b_1,t) = t^{5+\alpha}\text{sech}(x)\text{sech}(b_1), \quad t \geq 0,\\
&u(x,b_2,t) = t^{5+\alpha}\text{sech}(x)\text{sech}(b_2), \quad t \geq 0.
\end{aligned}$$

The exact solution of this problem is $u(x,t) = t^6\text{sech}(x)\text{sech}(y)$.

We developed our method for solving this two-dimensional KdV problem. We chose the centers of RBFs (and the collocation points) as $(x_i, y_j) = (a_1 + (i-1)\Delta x, b_1 + (j-1)\Delta y)$ in which $\Delta x = \frac{a_2-a_1}{M_1-1}$ and $\Delta y = \frac{b_2-b_1}{M_2-1}$ for $i = 1, \ldots, M_1$ and $j = 1, \ldots, M_2$. Table 9 depicts the errors and the experimental convergence orders resulted by our method with $M_1 = M_2 = 9$ for four different time steps and $[a_1, a_2] = [b_1, b_2] = [-1, 1]$. In these cases, we got $\rho(J_G(X^*)) < 0.48$.

Table 10 presents the E_∞, E_2, RMSE errors, $\rho(J_G(X^*))$ and $\kappa_2(B)$ resulted by IQ-RBFs for the case $\alpha = 0.35$ and $c = 1.95$. In Fig. 4, we plot the error function $|u_{exact} - u_{approx.}|$ and the approximate solution resulted by 50 iterations of our method with MQ-RBFs and $\tau = 0.02$, at $t = 1$.

5 Conclusion

In this article, an RBF collocation method was used for finding the solution of the time-fractional KdV equation in Caputo sense. The RBFs were applied for discretization of the spatial variable and we chose the centers of

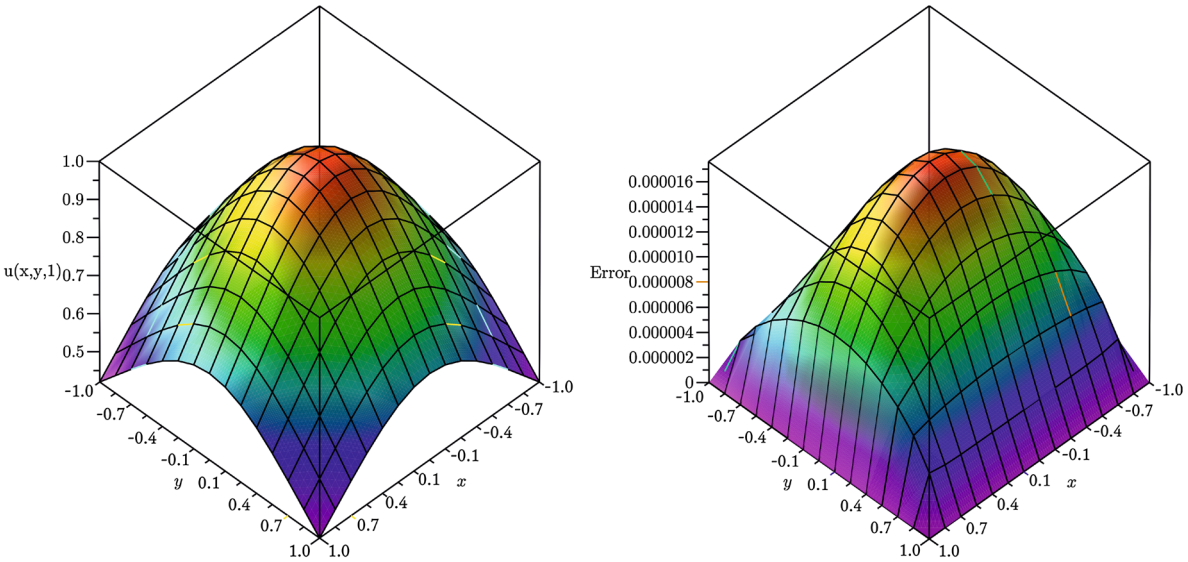

Fig. 4 The graphs of approximate solution (left) and absolute error (right) for $M_1 = M_2 = 15$, $\alpha = 0.35$, $\tau = 0.02$ and $t = 1$ by using MQ-RBFs with $c = 1.95$ in Example 3

RBFs as collocation points. The three formulas introduced in [6] were applied for approximating $\frac{\partial^\alpha u(X,t^n)}{\partial t^\alpha}$ for $n = 1$, $n = 2$ and $n \geq 3$. These formulas are of orders $2-\alpha$, $3-\alpha$ and $4-\alpha$, for, respectively, n=1, n=2 and $n \geq 3$, where α is the order of derivative. Instead of the formulas in [6], we can use for example the difference formula [22],

$$\frac{\partial^\alpha u(x,t^n)}{\partial t^\alpha} = a_0\Big(u^n - u^{n-1} + \sum_{k=1}^{n-1} b_k(u^{n-k} - u^{n-k-1})\Big) + O(\tau^{2-\alpha}), \quad n = 1, 2, 3, \ldots, \tag{25}$$

in which $a_0 = \frac{\tau^{-\alpha}}{\Gamma(2-\alpha)}$, $b_k = (k+1)^{1-\alpha} - k^\alpha, k = 0, 1, \ldots, n$. But the formulas in [6] have higher order of accuracy than the above formula, and consequently they lead to more accurate results. By our method, the computations of time-fractional KdV equation are reduced to some systems of nonlinear algebraic equations. A fixed point iteration method for solving the resulted nonlinear systems was introduced. Our method is computationally attractive and it can be generalized easily for two- and three-dimensional cases. In each time step, the method provides a closed form approximate solution. Three illustrative examples were included to demonstrate the accuracy, the stability and the convergence of the method and as we expected the experimental convergence order is approximately $4-\alpha$.

References

1. Abdulaziz, O.; Hashim, I.; Ismail, E.S.: Approximate analytical solution to fractional modified KdV equations. Math. Comput. Model. **49**, 36–45 (2009)
2. Belytschko, T.; Krongauz, Y.; Organ, D.; Fleming, M.; Krysl, P.: Meshless method: an overview and recent developments. Comput. Methods Appl. Mech. Eng. **139**, 3–47 (1996)
3. Belytschko, T.; Lu, Y.; Gu, L.: Element-free Galerkin methods. Int. J. Numer. Methods Eng. **37**, 229–256 (1994)
4. Buhman, M.D.: Spectral convergence of multiquadratic interpolation. Proc. Edinb. Math. Soc. **36**, 319–333 (1993)
5. Carlson, R.E.; Foley, T.A.: The parameter r^2 in multiquadratic interpolation. Comput. Math. Appl. **21**, 29–42 (1991)
6. Cao, J.X.; Li, C.P.; Chen, Y.Q.: High-order approximation to Caputo derivatives and Caputo-type advection-diffusion equations (II). Fract. Calcul. Appl. Anal. **18**(3), 735–761 (2015)
7. Chen, W.; Le, Y.; Sun, H.: Fractional diffusion equation by the Kansa method. Comput. Math. Appl. **59**, 1614–1620 (2010)

8. Cheng, R.; Cheng, Y.: Error estimates for the finite point method. Appl. Numer. Math. **58**, 884–898 (2008)
9. Cheng, Y.; Peng, M.: Boundary element-free method for elastodynamics. Sci. China Ser. G **48**(6), 641–657 (2005)
10. Coffey, M.W.: Nonlinear dynamics of vortices in ultraclean type-II superconductors: integrable wave equations in cylindrical geometry. Phys. Rev. B. **54**, 1270–1285 (1996)
11. Dehghan, M.; Shokri, A.: A numerical solution of the nonlinear Klein–Gordon equation using radial basis functions. J. Comput. Appl. Math. **230**, 400–410 (2009)
12. Duarte, M.; Oden, J.T.: An $h - p$ adaptive method using clouds. Comput. Methods Appl. Mech. Eng. **139**, 237–262 (1996)
13. El-Ajou, A.; Abu Arqub, O.; Momani, S.: Approximate analytical solution of the nonlinear fractional KdV–Burgers equation. J. Comut. Phys. **293**(c), 81–95 (2015)
14. Fausshauer, G.E.: Approximate moving least-square approximation with compactly supported radial weights. Lecture Notes in Computational Science and Engineering, vol. 26, pp. 105–116. Springer, New York (2002)
15. Gu, Y.T.; Zhuang, P.; Liu, F.: An advanced implicit meshless approach for the nonlinear anomalous subdiffusion equation. Comput. Model. Eng. Sci. **56**, 303–334 (2010)
16. Han, W.M.; Meng, X.P.: Error analysis of the reproducing kernel particle method. Comput. Methods Appl. Mech. Eng. **190**, 6157–6181 (2001)
17. Korteweg-de Vries, D.J.; de Vries, G.: On the change in form of long waves advancing in rectangular canal and on a new type of long stationary waves. Philos. Mag. **39**, 422–443 (1895)
18. Kurulay, M.; Bayram, M.: Approximate analytical solution for the fractional modified KdV by differential transform method. Commun. Nonlinear Sci. Numer. Simul. **15**, 1777–1782 (2010)
19. Li, X.: Meshless numerical analysis of a class of nonlinear generalized Klein–Gordon equations with a well-posed moving least squares approximation. Appl. Math. Model. **47**, 45–62 (2017)
20. Li, X.; Chen, H.; Wang, Y.: Error analysis in Sobolev spaces for the improved moving least-square approximation and the improved element-free Galerkin method. Appl. Math. Comput. **262**, 56–78 (2015)
21. Li, X.; Zhang, S.; Wang, Y.; Chen, H.: Analysis and application of the element-free Galerkin method for nonlinear Sine–Gordon and generalized Sinh–Gordon equations. Comput. Math. Appl. **71**, 1655–1678 (2016)
22. Lin, Y.; Xu, C.: Finite difference/spectral approximations for the time-fractional diffusion equation. J. Comput. Phys. **225**, 1533–1552 (2007)
23. Liu, W.K.; Han, W.M.: Reproducing Kernel element method. Part I: theoretical information. Comput. Methods Appl. Mech. Eng. **193**, 933–951 (2004)
24. Liu, Q.; Gu, Y.T.; Zhuang, P.; Liu, F.; Nie, Y.F.: An implicit RBF meshless approach for time fractional diffusion equations. Comput. Mech. **48**, 1–12 (2011)
25. Ludu, A.; Draayer, J.P.: Nonlinear modes of liquid drops as solitary waves. Phys. Rev. Lett. **80**, 2125–2128 (1998)
26. Melenk, J.M.; Babuska, I.: The partition of unity method: basic theory and applications. Comput. Methods Appl. Mech. Eng. **139**, 289–314 (1996)
27. Mohebbi, A.; Abbaszadeh, M.; Dehghan, M.: The use of meshless technique based on collocation and radial basis functions for solving the time fractional nonlinear Schrodinger equation arising in quantum mechanics. Eng. Anal. Bound. Elem. **37**, 475–485 (2013)
28. Momani, S.: An explicit and numerical solutions of the fractional KdV equation. Math. Comput. Simul. **70**, 110–118 (2005)
29. Momani, S.; Odibat, Z.; Alawneh, A.: Variational iteration method for solving the space- and the time-fractional KdV equation. Numer. Methods Partial Differ. Equ. **24**, 262–271 (2008)
30. Piret, C.; Hanert, E.: A radial basis functions method for fractional diffusion equations. J. Comput. Phys. **238**, 71–81 (2013)
31. Quarteroni, A.: Riccardo Sacco and Fausto Saleri Numerical Mathematics. Springer, Berlin (2000)
32. Reatto, L.; Galli, D.: What is a ROTON? Int. J. Mod. Phys. B. **13**, 607–616 (1999)
33. Sahoo, S.; Saha Ray, S.: Solitary wave solution for time fractional third order modified KdV equation using two reliable techniques (G'/G)-expansion method and improved (G'/G)-expansion method. Phys. A Stat. Mech. Appl. **448**, 265–282 (2016)
34. Schoenberg, I.J.: Metric spaces and completely monotone functions. Ann. Math. **39**, 811–841 (1938)
35. Song, L.; Zhang, H.Q.: Application of homotopy analysis method to fractional KdV–Burgers–Kuramoto equations. Phys. Lett. A **367**(1–2), 88–94 (2007)
36. Tatari, M.; Sepehrian, B.; Alibakhshi, M.: New implementation of radial basis functions for solving Burgers'–Fisher equation. Numer. Methods Partial Differ. Equ. **28**(1), 248–262 (2012)
37. Turitsyn, S.; Aceves, A.; Jones, C.; Zharnitsky, V.: Average dynamics of the optical soliton in communication lines with dispersion management: analytical results. Phys. Rev. E. **58**, R48–R51 (1998)
38. Wang, Q.: Homotopy perturbation method for fractional KdV equation. Appl. Math. Comput. **190**, 1795–1802 (2007)
39. Wang, Q.: Numerical solutions of the space and time fractional KdV–Burgers' equation by Adomian decomposition method. Appl. Math. Comput. **182**(2), 1048–1055 (2006)
40. Wang, J.G.; Liu, G.R.: A point interpolation meshless method based on radial basis functions. Int. J. Numer. Methods Eng. **54**, 1623–1648 (2002)
41. Zorzano, M.P.; Mais, H.; Vazquez, L.: Numerical solution of two dimensional Fokker–Planck equations. Appl. Math. Comput. **98**, 109–117 (1999)