A MESHFREE COLLOCATION METHOD FOR THE SOLUTION OF A CLASS OF PARTIAL INTEGRO-DIFFERENTIAL EQUATIONS

Arshed Ali[1], Syed Inayat Ali Shah[1], Shakeel Ahmad[1], Iltaf Hussain[2]
[1]Department of Mathematics, Islamia College Peshawar, Khyber Pakhtunkhwa, Pakistan.
[2]Department of Basic Sciences & Islamiat, University of Engineering & Technology, Peshawar, Pakistan.
Email addresses: arshad_math@hotmail.com, inayat64@gmail.com, shakeelicp60@gmail.com, iltafn@yahoo.com

ABSTRACT: *A meshfree collocation method using radial basis functions is developed for numerical solution of a parabolic type partial integro-differential equation with a weakly singular kernel. The scheme is constructed by approximating the time derivative using forward and central difference formulae while the spatial derivative is approximated using radial basis functions. Three types of radial basis functions are used for this purpose. Three test problems are provided to validate the proposed scheme. Numerical results are obtained using various numbers of collocation points and time step sizes. Accuracy of the method is assessed in terms of L_∞ and L_2 error norms. Remarkable accuracy is obtained and the results are also compared with cubic B-spline collocation method.*

Keywords: Partial integro-differential equation, Weakly singular kernel, Radial basis function, Collocation method.

1. INTRODUCTION

Partial integro-differential equations (PIDEs) arise as mathematical models of various physical processes including heat conduction, viscoelastic polymers, nuclear reactor dynamics, immunology, option pricing and electricity swaptions. Some applications of PIDE as models of different physical phenomena can be found in the literature [1-6]. Several numerical techniques have been developed in the literature for the approximate solution of various types of PIDEs including finite difference methods, finite element methods, wavelet methods, spline methods, chebyshev polynomials, variational iteration and homotopy perturbation methods and radial basis functions methods (see [7-23, 31-32] and the references therein).

In this paper, we develop a meshfree method based on collocation principle along with radial basis functions for solution of the following type PIDEs:

$$\int_0^t \beta(t-s)w_t(x,s)ds - w_{xx}(x,t) = g(x,t), x \in [a,b], \quad t>0, \qquad (1)$$

with initial condition

$$w(x,0) = h_0(x), \quad a \le x \le b, \qquad (2)$$

and boundary conditions

$$w(a,t) = h_1(t), w(b,t) = h_2(t), t \ge 0, \qquad (3)$$

where $\beta(t)$ is a weakly singular kernel which is given as $\beta(t) = \frac{t^{(\alpha-1)}}{\Gamma(\alpha)}$, $0<\alpha<1$, at $t=0$, Γ denotes the gamma function, $h_0(x)$, $h_1(t)$ and $h_2(t)$ are known functions.

Meshfree methods using radial basis functions (RBF) have become popular in approximation theory because of its simplicity, exponential accuracy, flexibility with respect to geometry, dimensional independence and straightforward implementation. Meshfree methods do not require a grid and only make use of a set of scattered data points regardless of the connectivity information between the points. Unlike finite-difference, finite element, finite volume and spectral methods these methods circumvent mesh generation which is main problem in mesh-based numerical methods. Also radial basis function based methods are domain type methods because the solution of the problem can be extended to whole domain In 1990, Kansa pioneered the radial basis function collocation method for the approximate solution of PDE [24]. Later on, Franke and Schaback gave a theoretical establishment to the method for the solution of partial differential equations [5]. Further applications of this method to partial differential equations can be found in [26-30]. Recently, the radial basis functions collocation method has been used for the solution of different types of PIDEs including nonlinear parabolic partial integro-differential equation with weakly singular kernel [22], nonlinear volterra partial integro-differential equations [31] and convection-diffusion integro-differential equations [32].

Radial basis functions are mainly divided in two categories:

(i) Infinitely smooth RBFs
(ii) Piecewise smooth RBFs

The infinitely smooth RBFs contain a parameter called shape parameter which affects both accuracy of solution and conditioning of collocation matrix, while piecewise smooth RBFs are free of shape parameter. To find that value of the shape parameter which leads to optimal accuracy is still an open problem (see [28-30]).

Some of commonly used RBFs are as follows:

Multiquadric (MQ)	$\psi(r) = \sqrt{r^2+c^2}$
Gaussian (GA)	$\psi(r) = \exp(-cr^2)$
Thin plate spline (TPS)	$\psi(r) = r^2 \ln r$
Quintic	$\psi(r) = r^5$
Spline of degree seven (SD)	$\psi(r) = r^7$

Where c is the shape parameter.

In this paper, MQ, GA and SD types of radial basis functions are used for computation.

2. CONSTRUCTION OF PROPOSED METHOD

Consider the partial integro-differential equation given in Eqs. (1)-(3). Let $t_n = nk$, where k is the time step and $t_{n+1} = t_n + k$, $n = 0,1,2 \ldots$. In Eq. (1), the integral term is approximated at $t = t_{n+1}$ as given in [19, 23]:

$$\int_0^{t_{n+1}} \frac{(t_{n+1}-s)^{\alpha-1} w_t(x,\ s)ds}{\Gamma(\alpha)} = \int_{t_0}^{t_1} \frac{(w^1(x)-w^0(x))(t_{n+1}-s)^{\alpha-1}ds}{k\Gamma(\alpha)} + \sum_{r=1}^{n} \int_{t_r}^{t_{r+1}} \frac{(w^{r+1}(x)-w^{r-1}(x))(t_{n+1}-s)^{\alpha-1}ds}{2k\ \Gamma(\alpha+1)} = b_n \frac{w^1(x)-w^0(x)}{2\Gamma(\alpha+1)k^{1-\alpha}} + \frac{1}{2\Gamma(\alpha+1)} \sum_{r=0}^{n-1} b_r \frac{w^{r+1}(x)-w^{r-1}(x)}{k^{1-\alpha}}, \qquad (4)$$

where $w^{n+1}(x) = w(x, t_{n+1}), b_n = (n+1)^\alpha - n^\alpha$, $n = 0,1,2,..,M$.

From Eqs. (1) and (4), we have

$b_0 w^{n+1}(x) - 2\Gamma(\alpha+1)k^{1-\alpha} w_{xx}^{n+1}(x) = b_0 w^{n-1}(x) - \sum_{r=1}^{n-1} b_r\left(w^{n-r+1}(x) - w^{n-r-1}(x)\right) - 2b_n\left(w^1(x) - w^0(x)\right) + 2\Gamma(\alpha+1)k^{1-\alpha} g^{n+1}(x),$
(5)

where $w_{xx}^{n+1}(x) = w_{xx}(x, t_{n+1})$, $g^{n+1}(x) = g(x, t_{n+1})$,
Assuming $d_0 = 2\Gamma(\alpha+1)k^{1-\alpha}$ with $b_0 = 1$.
Eq. (5) becomes

$w(x) - d_0 w_{xx}^{n+1}(x) = -b_1 w^n(x) + \sum_{r=1}^{n-1}(b_{r-1} - b_{r+1}) w^{n-r}(x) - b_n w^1(x) + (b_{n-1} + 2b_n) w^0(x) + d_0 g^{n+1}(x), \quad n \geq 1.$
(6)

Substituting $n = 0$, in Eq. (6), then u^1 can be obtained from the following:

$$w^1(x) - 0.5 d_0 {w_{xx}}^1 = w^0(x) + 0.5 d_0 g^1(x),$$

To obtain approximate solution of the problem (1)-(3) using radial basis functions collocation method, we choose $N+1$ distinct points x_i, $i = 0,1,2,\dots,N$, from the interval $[a,b]$ such that $a = x_0$ and $x_N = b$. The RBF approximation of the function $w(x,t)$ at nth time level is given by

$$w^n(x) = \sum_{m=0}^{N} \lambda_m^n \psi(e_m), \qquad (7)$$

where λ_m^n are unknown time dependent parameters to be determined from collocation conditions, ψ is a radial basis function and $e_m = |x - x_m|$ is Euclidean distance. Collocating Eq. (6) at $x = x_i, i = 1,2,\dots,N-1$, and using Eq. (7), we get

$$\sum_{m=0}^{N} \lambda_m^{n+1}\psi(e_{im}) - d_0 \sum_{m=0}^{N} \lambda_m^{n+1}\psi''(e_{im}) = -b_1 \sum_{m=0}^{N} \lambda_m^n \psi(e_{im}) + \sum_{r=1}^{n-1}\left((b_{r-1} - b_{r+1}) \sum_{m=0}^{N} \lambda_m^{n-r}\psi(e_{im})\right) - b_n \sum_{m=0}^{N} \lambda_m^1 \psi(e_{im}) + (b_{n-1} + 2b_n)\sum_{m=0}^{N} \lambda_m^0 \psi(e_{im}) + d_0 g^{n+1}(x_i). \qquad (8)$$

Also from Eqs. (3) and (7), we have

$$\left.\begin{aligned} \sum_{m=0}^{N} \lambda_m^{n+1}\psi(e_{0m}) &= h_1(t_{n+1}), \\ \sum_{m=0}^{N} \lambda_m^{n+1}\psi(e_{Nm}) &= h_2(t_{n+1}), \end{aligned}\right\} \qquad (9)$$

where $e_{im} = |x_i - x_m|$.

In matrix form Eqs. (8) and (9) can be expressed as

$$\boldsymbol{\lambda}^{n+1} = \mathbf{C}^{-1}\left[-b_1 \mathbf{A}\boldsymbol{\lambda}^n + \sum_{r=1}^{n-1}(b_{r-1} - b_{r+1})\mathbf{A}\boldsymbol{\lambda}^{n-r} - b_n \mathbf{A}\boldsymbol{\lambda}^1 + d_0 \boldsymbol{G}^{n+1}\right], \qquad (10)$$

where $\mathbf{A} = [a_{ij}]_{i,j=0}^{N}$, $\mathbf{B} = [b_{ij}]_{i,j=0}^{N}$ such that $a_{ij} = \psi(e_{ij})$,

$$b_{ij} = \begin{cases} \psi''(e_{ij}), & 1 \leq i \leq N-1, 0 \leq j \leq N, \\ 0, & i = 0, N, \ 0 \leq j \leq N. \end{cases}$$

$\mathbf{C} = [\mathbf{A} - d_0 \mathbf{B}]$ and $\boldsymbol{\lambda}^n = [\lambda_0^n, \lambda_1^n, \dots, \lambda_N^n]^T$. Inevitability of the matrix $\mathbf{C}$ is yet to be proved. Eq. (10) represents a system of $N+1$ equations in $N+1$ unknown parameters λ_j^{n+1}'s. The solution of Eq. (10) leads to the vector $\boldsymbol{\lambda}^{n+1}$ and then the approximate solution is obtained by using Eq. (7).

3. NUMERICAL TESTS AND PROBLEMS

In this section we provide some problems in order to test accuracy of the RBF collocation method (10) for the solution of the problem (1)-(3). For the sake of comparison all the three test problems are taken from the reference [19] with $\alpha = \frac{1}{2}$. Error norms L_∞ and L_2 are used for this purpose. Accuracy of the proposed scheme is compared with results of cubic B-spline collocation method [19]. The solution plotted in each figure is obtained using multiquadric in the RBF collocation method (10).

Example 1: Consider Eqs. (1)-(3) with $x \in [0,1]$, and $g(x,t)$ is chosen so that the exact solution [19] is

$w(x,t) = (t+1)\sin \pi x$.

The initial and boundary conditions are obtained from the exact solution. Computations are performed using MQ, SD and GA types of RBFs with parameters $N = 60,\ 100,\ k = 0.0001,\ 0.001$ and the error norms L_∞ and L_2 up to time level $M = 1000$ are reported in Tables (1)-(2) whereas Table (3) contains the error norms for different number of collocation points $N = 10, 20, 30, 40, 60$. Better accuracy of the present method using MQ and GA than cubic B-spline collocation method [19] is obvious from Tables 1-2. However, SD produces better results than [19] for larger values of N. Fig. 1 presents the RBF and exact solutions up to $M = 1000$. Fig. 2 shows the RBF solutions over the time interval [0, 1]. Fig. 3 shows L_2 errors versus $1/N$. Most accurate results are obtained for shape parameter $c = 0.21, 0.125$ corresponding to $N = 60, 100$ respectively for MQ and $c = 100$ for GA while SD is free of shape parameter c.

Example 2: In this example we consider Eqs. (1)-(3) and choose $g(x,t)$ such that the exact solution [19] is

$w(x,t) = (t+1)\cos \pi x,\ x \in [0,1]$.

The initial and boundary conditions are taken from the exact solution. Numerical simulations are done using MQ, SD and GA with parameters $N = 60$, $k = 0.001$, 0.0001 and the error norms L_∞ and L_2 up to time level $M = 1000$ are reported in Table (4) while Table 5 contains the error norms for different number of collocation points $N = 10,\ 20,\ 30,\ 40,\ 50,\ 60$. Better accuracy of the present method using MQ, SD and GA than B-spline collocation methods [19, 23] is evident from Table (4). However, SD requires larger values of N for better accuracy than [19, 23]. Fig. 4 presents the RBF and exact solutions up to time level $M = 500$. Fig. 5 shows the RBF solutions over the time interval [0, 1]. Most accurate results are obtained for shape parameter $c = 0.21$ for MQ and $c = 80, 90$ corresponding to $k = 0.001, 0.0001$ respectively for GA. The radial basis functions SD is independent of the shape parameter c.

Example 3: We consider Eqs. (1)-(3) and choose $g(x,t)$ such that the exact solution [19] is

$w(x,t) = (t+1)^2 \sin \pi x,\ x \in [-1,1]$.

The initial and boundary conditions are taken from the exact solution. Simulations are done using MQ, SD and GA with parameters $N = 40,\ k = 0.001,\ 0.00125$ and the error norms L_∞ and L_2 up to time level $M = 500$ are reported in Table (6) Better accuracy of the present method using MQ, SD and GA than cubic B-spline collocation method [19] can be seen from Table (6). However, SD requires larger values of N for better accuracy than [19]. Fig. 6 presents the RBF and exact solutions up to time level $M = 1000$. Fig. 7 shows the RBF solutions over the time interval [0,1]. Most accurate results are obtained for shape parameter $c = 0.4, 0.32$ corresponding to $k = 0.001, 0.00125$ respectively for MQ and $c = 75, 70$ corresponding to $k = 0.001, 0.00125$ respectively for GA. The radial basis function SD is independent of shape parameter c.

Table 1: L_∞ and L_2 for $k = 0.0001$.

		N =60		N =100	
	M	L_∞	L_2	L_∞	L_2
MQ	10	1.67×10^{-7}	8.48×10^{-9}	1.66×10^{-7}	7.13×10^{-9}
	50	1.74×10^{-7}	1.20×10^{-8}	1.71×10^{-7}	9.65×10^{-9}
	100	1.79×10^{-7}	1.43×10^{-8}	1.73×10^{-7}	1.07×10^{-8}
	500	1.87×10^{-7}	1.89×10^{-8}	1.82×10^{-7}	1.38×10^{-8}
	1000	1.98×10^{-7}	2.07×10^{-8}	1.91×10^{-7}	1.55×10^{-8}
SD	10	1.01×10^{-5}	5.54×10^{-7}	1.22×10^{-6}	5.11×10^{-8}
	50	1.04×10^{-5}	7.51×10^{-7}	1.25×10^{-6}	6.88×10^{-8}
	100	1.06×10^{-5}	8.52×10^{-7}	1.26×10^{-6}	7.79×10^{-8}
	500	1.12×10^{-5}	1.10×10^{-6}	1.33×10^{-6}	1.00×10^{-7}
	1000	1.22×10^{-5}	1.22×10^{-6}	1.40×10^{-6}	1.12×10^{-7}
GA	10	3.85×10^{-7}	2.21×10^{-8}	8.45×10^{-7}	2.98×10^{-8}
	50	3.94×10^{-7}	2.98×10^{-8}	7.91×10^{-7}	3.50×10^{-8}
	100	4.23×10^{-7}	3.46×10^{-8}	9.13×10^{-7}	5.09×10^{-8}
	500	4.56×10^{-7}	4.53×10^{-8}	9.51×10^{-7}	6.38×10^{-8}
	1000	4.82×10^{-7}	4.98×10^{-8}	9.82×10^{-7}	7.20×10^{-8}
[19]	10	4.12×10^{-5}	3.76×10^{-6}		
	50	4.06×10^{-6}	3.71×10^{-7}		

Table 2: L_∞ and L_2 for $k = 0.001$.

		N =60		N =100	
	M	L_∞	L_2	L_∞	L_2
MQ	10	2.88×10^{-8}	2.66×10^{-9}	1.13×10^{-7}	6.85×10^{-9}
	50	4.88×10^{-8}	4.63×10^{-9}	1.17×10^{-7}	9.03×10^{-9}
	100	4.71×10^{-8}	4.70×10^{-9}	1.29×10^{-7}	1.04×10^{-8}
	500	6.67×10^{-8}	7.29×10^{-9}	1.76×10^{-7}	1.53×10^{-8}
	1000	7.47×10^{-8}	7.74×10^{-9}	2.40×10^{-7}	2.17×10^{-8}
SD	10	1.06×10^{-5}	8.13×10^{-7}	1.26×10^{-6}	7.44×10^{-8}
	50	1.23×10^{-5}	1.08×10^{-6}	1.33×10^{-6}	9.87×10^{-8}
	100	1.18×10^{-5}	1.21×10^{-6}	1.40×10^{-6}	1.10×10^{-7}
	500	1.62×10^{-5}	1.80×10^{-6}	1.91×10^{-6}	1.64×10^{-7}
	1000	2.17×10^{-5}	2.45×10^{-6}	2.56×10^{-6}	2.23×10^{-7}
GA	10	4.63×10^{-7}	2.26×10^{-8}	5.30×10^{-8}	2.72×10^{-9}
	50	3.74×10^{-7}	2.75×10^{-8}	1.80×10^{-7}	8.20×10^{-9}
	100	3.06×10^{-7}	2.52×10^{-8}	1.20×10^{-7}	5.89×10^{-9}
	500	4.63×10^{-7}	4.50×10^{-8}	2.95×10^{-7}	1.78×10^{-8}
	1000	5.86×10^{-7}	6.08×10^{-8}	3.31×10^{-7}	2.29×10^{-8}
[19]	10	8.77×10^{-4}	8.01×10^{-5}		
	50	8.23×10^{-4}	7.51×10^{-5}		

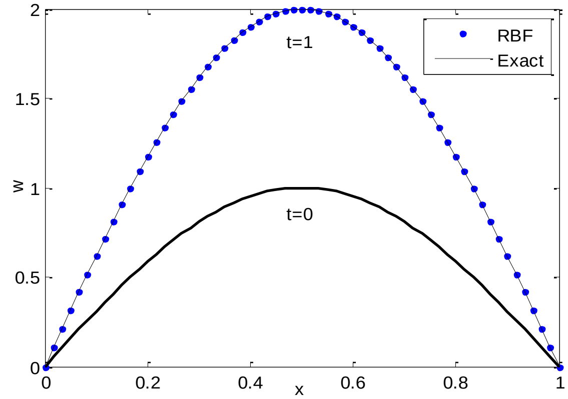

Figure 1: RBF and exact solutions for Example-1 for $k = 0.001$.

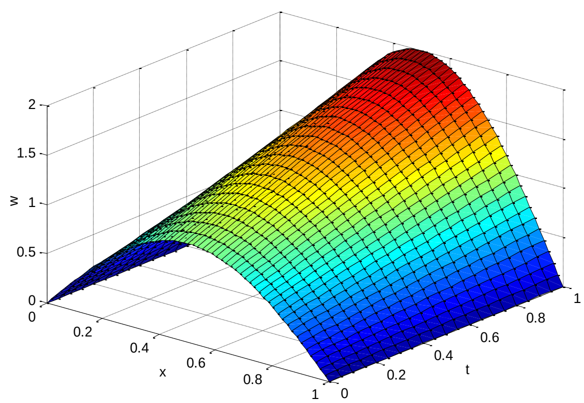

Figure 2: RBF solutions over time interval $[0, 1]$ corresponding to Example-1 for $N = 60, k = 0.001$.

Table 3: L_∞ and L_2 for $k = 0.0001, M = 10$.

	MQ		[19]	
N	L_∞	L_2	L_∞	L_2
10	9.33×10^{-8}	1.44×10^{-8}	8.58×10^{-4}	1.84×10^{-4}
20	3.18×10^{-7}	3.61×10^{-8}	1.62×10^{-4}	2.06×10^{-5}
30	1.14×10^{-7}	9.40×10^{-9}	4.09×10^{-5}	2.96×10^{-6}
40	4.14×10^{-9}	2.88×10^{-10}	1.32×10^{-6}	1.16×10^{-6}
60	1.67×10^{-7}	8.48×10^{-9}	9.14×10^{-5}	6.85×10^{-6}
	GA		SD	
10	1.30×10^{-8}	2.77×10^{-9}	4.90×10^{-3}	8.01×10^{-4}
20	6.39×10^{-8}	6.00×10^{-9}	2.14×10^{-4}	2.18×10^{-5}
30	7.09×10^{-8}	7.02×10^{-9}	3.88×10^{-5}	3.08×10^{-6}
40	1.55×10^{-7}	8.22×10^{-9}	1.18×10^{-5}	7.93×10^{-7}
60	3.85×10^{-7}	2.21×10^{-8}	2.34×10^{-6}	1.48×10^{-7}

Table 4: L_∞ and L_2 for $N = 60$.

	SD		[19]	
N	L_∞	L_2	L_∞	L_2
10	4.29×10^{-2}	7.20×10^{-3}	2.02×10^{-3}	4.51×10^{-4}
20	1.20×10^{-3}	1.25×10^{-4}	4.29×10^{-4}	6.78×10^{-5}
30	1.92×10^{-4}	1.57×10^{-5}	1.35×10^{-4}	1.74×10^{-5}
40	5.55×10^{-5}	3.84×10^{-6}	3.22×10^{-5}	3.60×10^{-6}
50	2.16×10^{-5}	1.32×10^{-6}	1.54×10^{-5}	1.54×10^{-6}
60	1.01×10^{-5}	5.54×10^{-7}	4.12×10^{-5}	3.76×10^{-6}

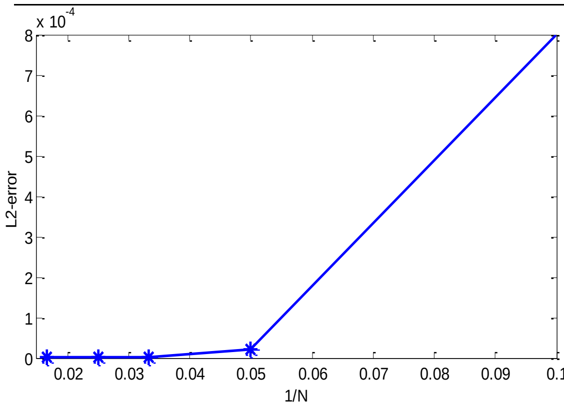

Figure 3: L_2 error versus $1/N$ Example-1 for $k = 0.001$.

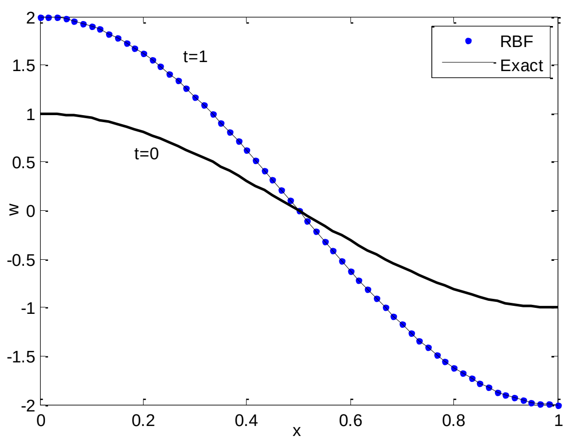

Figure 4: RBF and exact solutions corresponding to le-2 for $N = 60, k = 0.001$.

Table 5: L_∞ and L_2 for $k = 0.0001, M = 10$.

		k =0.001		k =0.0001	
	M	L_∞	L_2	L_∞	L_2
MQ	10	3.06×10^{-7}	2.00×10^{-8}	2.43×10^{-7}	9.18×10^{-9}
	20	3.18×10^{-7}	2.12×10^{-8}	4.14×10^{-7}	2.03×10^{-8}
	50	3.25×10^{-7}	2.25×10^{-8}	4.10×10^{-7}	1.64×10^{-8}
	100	3.39×10^{-7}	2.38×10^{-8}	3.70×10^{-7}	1.86×10^{-8}
	500	4.62×10^{-7}	3.32×10^{-8}	3.57×10^{-7}	2.12×10^{-8}
SD	10	2.34×10^{-6}	1.48×10^{-7}	2.25×10^{-6}	1.20×10^{-7}
	20	2.38×10^{-6}	1.56×10^{-7}	2.28×10^{-6}	1.32×10^{-7}
	50	2.46×10^{-6}	1.66×10^{-7}	2.32×10^{-6}	1.44×10^{-7}
	100	2.59×10^{-6}	1.77×10^{-7}	2.35×10^{-6}	1.51×10^{-7}
	500	3.54×10^{-6}	2.46×10^{-7}	2.47×10^{-6}	1.67×10^{-7}
GA	10	1.55×10^{-7}	8.18×10^{-9}	2.43×10^{-7}	1.16×10^{-8}
	20	1.59×10^{-7}	9.49×10^{-9}	3.22×10^{-7}	1.49×10^{-8}
	50	1.64×10^{-7}	8.73×10^{-9}	2.00×10^{-7}	1.16×10^{-8}
	100	1.71×10^{-7}	1.07×10^{-8}	3.24×10^{-7}	1.64×10^{-8}
	500	2.35×10^{-7}	1.26×10^{-8}	2.33×10^{-7}	1.43×10^{-8}
[19	10	9.90×10^{-4}	8.59×10^{-5}	9.14×10^{-5}	6.85×10^{-6}
	20	9.88×10^{-4}	8.58×10^{-5}	9.10×10^{-5}	5.40×10^{-6}
	50	9.77×10^{-4}	8.48×10^{-5}	8.86×10^{-5}	5.44×10^{-6}
[23	10	7.29×10^{-6}	8.15×10^{-7}	1.02×10^{-5}	1.14×10^{-6}
	20	1.16×10^{-5}	1.29×10^{-6}	1.02×10^{-5}	1.14×10^{-6}
	50	2.06×10^{-5}	2.30×10^{-6}	2.90×10^{-5}	3.24×10^{-6}

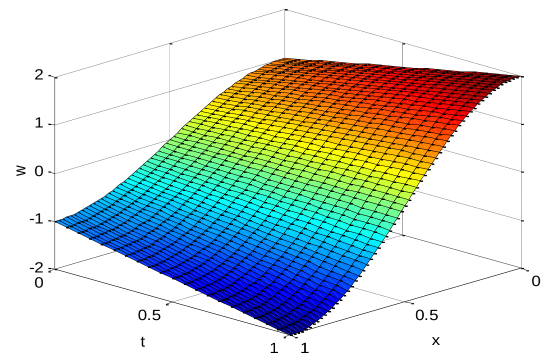

Figure 5: RBF solutions over time interval $[0, 1]$ for Example-2 for $N = 60, k = 0.001$.

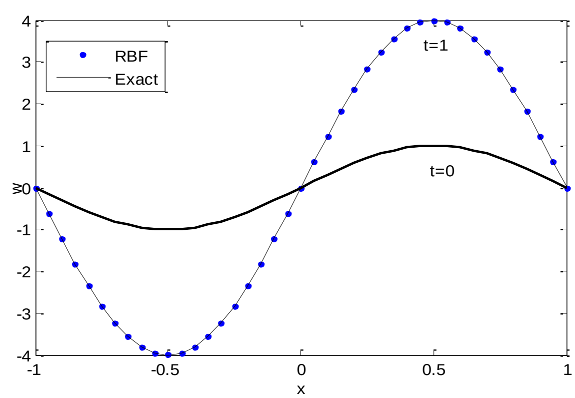

Figure 6: RBF and exact solutions corresponding to Example-3 for $N = 40, k = 0.001$.

Table 6: L_∞ and L_2 for $N = 60$.

		k =0.001		k =0.00125	
	M	L_∞	L_2	L_∞	L_2
MQ	10	5.98×10^{-6}	6.69×10^{-7}	8.82×10^{-6}	9.84×10^{-7}
	20	9.99×10^{-6}	1.12×10^{-6}	1.46×10^{-5}	1.63×10^{-6}
	50	1.86×10^{-5}	2.08×10^{-6}	2.68×10^{-5}	3.00×10^{-6}
	100	2.87×10^{-5}	3.21×10^{-6}	4.11×10^{-5}	4.59×10^{-6}
	500	7.27×10^{-5}	8.13×10^{-6}	1.03×10^{-4}	1.15×10^{-5}
SD	10	6.06×10^{-5}	6.26×10^{-6}	6.14×10^{-5}	6.78×10^{-6}
	20	6.31×10^{-5}	7.56×10^{-6}	6.43×10^{-5}	8.30×10^{-6}
	50	6.86×10^{-5}	9.97×10^{-6}	7.32×10^{-5}	1.10×10^{-5}
	100	8.22×10^{-5}	1.22×10^{-6}	9.84×10^{-5}	1.40×10^{-5}
	500	1.86×10^{-4}	2.62×10^{-5}	2.37×10^{-4}	3.28×10^{-5}
GA	10	9.08×10^{-6}	5.90×10^{-7}	6.31×10^{-6}	6.52×10^{-7}
	20	9.14×10^{-6}	6.63×10^{-7}	1.13×10^{-5}	1.16×10^{-6}
	50	1.16×10^{-5}	1.19×10^{-6}	2.25×10^{-5}	2.36×10^{-6}

	100	2.01×10^{-5}	2.05×10^{-6}	3.57×10^{-5}	3.81×10^{-6}
	500	5.46×10^{-5}	5.64×10^{-6}	9.02×10^{-5}	9.69×10^{-6}
[19]	10	5.99×10^{-4}	6.25×10^{-5}	1.00×10^{-3}	1.05×10^{-4}
	20	4.33×10^{-4}	4.66×10^{-5}	7.23×10^{-4}	7.71×10^{-5}
	50	2.00×10^{-3}	1.67×10^{-4}	2.87×10^{-3}	2.34×10^{-4}

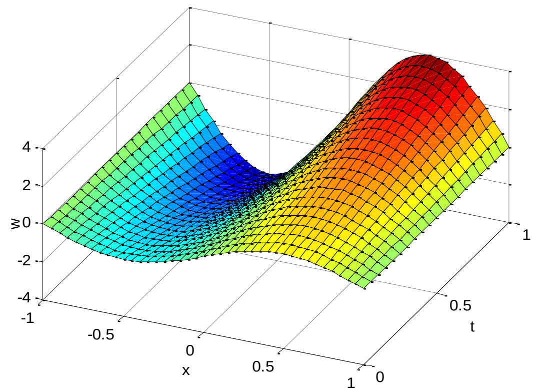

Figure 7: RBF solutions over time interval $[0,1]$ for Example-3 for $N = 40, k = 0.001$.

4. CONCLUSION

A collocation method coupled with radial basis functions is employed to approximate solution of a parabolic type integro-differential equation with a weakly singular kernel. The proposed method is validated by implementing three benchmark problems from literature. The errors are satisfactorily small and the results are in good agreement with exact solution. Implementation of the method is simple like finite difference method. Infinitely RBFs provided excellent accuracy. Numerical simulations suggest that this method can be used for numerical approximation of integral equations, partial differential equations and partial integro-differential equations of such type.

REFERENCES:

[1] Gurtin, M.E. and A.C. Pipkin, “A general theory of heat conduction with finite wave speed”, *Arch. Rational Mech. Anal.*, **31**: 113-126 (1968).

[2] Christensen, R.M., “Theory of viscoelasticity”, Academic Press, New York, (1971).

[3] Miller, R.K., “An integro-differential equation for rigid heat conductors with memory”, *J. Math. Anal. Appl.*, **66**: 313-332 (1978).

[4] Lodge, A.S., M. Renardy, and J.A. Nohel, “Viscoelasticity and rheology”, Academic Press, New York, (1985).

[5] Ortega, J.M., S.H. Davis, S. Rosemblat, and W.L. Kath, “Bifurcation with memory”, *SIAM J. Appl. Math.*, **46**:171-188 (1986).

[6] Renardy, M., “Mathematical analysis of viscoelastic flows”, *Ann. Rev. Fluid. Mech.*, **21**: 21-36 (1989).

[7] Lopez-Marcos, J.C., “A difference scheme for a nonlinear partial integro-differential equation”, *SIAM J. Numer. Anal.*, **27:** 20-31(1990).

[8] Le Roux, M.N. and V. Thomee, “Numerical solution of semilinear integro-differential equations of parabolic type with nonsmooth data”, *SIAM J. Numer. Anal.*, **26**: 1291-1309 (1989).

[9] Yanik, E.G. and G. Fairweather, “Finite element methods for parabolic and hyperbolic partial integro-differential equations”, *Nonlinear Anal. Theor.*, **12**: 785-809 (1988).

[10] Lin, Y. and T. Zhang, “The stability of 3 ritz-volterra projections and error estimates for finite element methods for a class of integro-differential equations of parabolic type”, *Appl. Math.*, **36**: 123-133 (1991).

[11] Pani, A.K., V. Thomee, and L.B. Wahlbin, “Numerical methods for hyperbolic and parabolic integro-differential equations”, *J. Integral Equat. Appl.*, **4**: 533-584 (1992).

[12] Tang, T., “A finite difference scheme for partial integro-differential equations with a weakly singular kernel”, *Appl. Numer. Math*., **11**: 309-319 (1993).

[13] Yang, X., D. Xu, and H. Zhang, “Quasi-wavelet based numerical method for fourth-order partial integro-differential equations with a weakly singular kernel”, *Int. J. Comput. Math.*, **81**: 3236-3254 (2011).

[14] Yang, C., “Numerical solution of nonlinear fredholm integro-differential equations of fractional order by using hybrid of block-pulse functions and chebyshev polynomials”, *Math. Probl. Eng.*, Volume 2011, Article ID 341989, 11 pages (2011).

[15] Nawaz, Y., “Variational iteration method and homotophy perturbation method for fourth-order fractional integro-differential equations”, *Appl. Math. Comput.*, **61**: 2330-2341 (2011).

[16] Yang, X., D. Xu, and H. Zhang, “Crank-Nicolson/quasi-wavelets method for solving fourth order partial integro-differential equation with a weakly singular kernel”, *J. Comput. Phys.*, **234**: 317-329 (2013).

[18] Zhang, H., X. Han, and X. Yang, “Quintic b-spline collocation method for fourth order partial integro differential equations with a weakly singular kernel”, *Appl. Math. Comput.*, **219**: 6565-6575 (2013).

[19] Siddiqi, S.S. and S. Arshed, “Cubic b-spline for the numerical solution of parabolic integro-differential equation with a weakly singular kernel”, *Res. J. Appl. Sci. Eng. Tech.*, **7**: 2065-2073 (2014).

[20] Luo, M., D. Xu, and L. Li, “A compact difference scheme for a partial integro-differential equation with a weakly singular kernel”, *Appl. Math. Model*., **39:** 947-954 (2015).

[21] Ali, A., S. Ahmad, S.I.A. Shah, and Fazal-i-Haq, “A quartic B-spline collocation technique for the solution of partial integro-differential equations with a weakly singular kernel”, *Science International*, **27**(5): 3971-3976 (2015).

[22] Biazar, J. and M.A. Asad, “FD-RBF for partial integro-differential equations with a weakly singular kernel”, *Appl. Comput. Math.*, **4**(6): 445-451 (2015).

[23] Ali, A., S. Ahmad, S.I.A. Shah, and F.I. Haq, "A computational technique for the solution of parabolic type integro-differential equation with a weakly singular kernel", *SURJ*, **48**(1): 71-74 (2016).

[24] Kansa, E.J., "Multiquadrics scattered data approximation scheme with applications to Computational fluid-dynamics I, surface approximations and partial derivative estimates", *Comp. Math. Appl.*, **19**(8-9): 147-161 (1990).

[25] Franke, C. and R. Schaback, "Solving partial differential equations by collocation with radial basis functions", *Appl. Math. Comput*., **93**: 73-82, (1998).

[26] Sarler, B. and R. Vertnik, "Meshfree explicit local radial basis function collocation method for diffusion problems", *Comp. Math. Appl.*, **51**(8): 1269-1282 (2006).

[27] Dehghan, M. and A. Shokri, "A numerical method for two-dimensional Schrödinger equation using collocation and radial basis functions", *Comp. Math. Appl.*, **54**(1): 136-146 (2007)

[28] Siraj-ul-Islam, A. Ali and S. Haq, "A Computational modeling of the behavior of the two dimensional reaction-diffusion Brusselator system", *Appl. Math. Model*., **34**, 3896–3909, (2010).

[29] Ali, A., F. Haq, and I. Hussain, "A numerical meshless technique for the solution of some Burgers' type equations", *W. Appl. Sci. J*., **14** (12): 1792-1798 (2011).

[30] Ali, A., Fazal Haq, I. Hussain and U. Shah, "A meshless method of lines for numerical solution of some coupled nonlinear evolution equations", *Int. J. Nonlinear Sci. Numer. Simul.*, **15**(2): 121–128 (2014).

[31] Avazzadeh, Z., Z. Beygi Rizi, F.M. Maalek Ghaini, and G. B. Loghmani, "A numerical solution of nonlinear parabolic-type volterra partial integro-differential equations using radial basis functions", *Eng. Anal. Bound. Elem.*, **36**: 881–893 (2012).

[32] Ali, A., E. Rahman, Z. Jan, I. Hussain, and S. Ahmad, "A meshless collocation method for the approximate solution of a partial integro-differential equation", *SURJ*, **48**(3): 589-594 (2016).