Iranian Journal of Numerical Analysis and Optimization
Vol. ??, No. ??, ??, pp ??
https://doi.org/??
https://ijnao.um.ac.ir/

Research Article

Space-time localized scheme to solve some partial integro-differential equations

M. Hamaidi*, M. Briki, A. Nouara and B. Hamdi

Abstract

It has been demonstrated that the space-time localized radial basis functions collocation method has very good accuracy in several research studies. In this paper, we extend the method to solve the partial integro-differential equations. Since the unknowns of the localized scheme are the values of the

*Corresponding author

Received ??? ; revised ??? ; accepted ???

Mohammed Hamaidi
Department of Mathematics, Faculty of exact sciences and computer science, Ziane Achour University, Djelfa, Algeria. e-mail: hamaidi@yahoo.com

Mabrouk Briki
Department of Mathematics, Faculty of exact sciences and computer science, Ziane Achour University, Djelfa, Algeria. e-mail: mabroukbriki@yahoo.fr

Ahmed Nouara
Department of Mathematics, Faculty of exact sciences and computer science, Ziane Achour University, Djelfa, Algeria. e-mail: ahmednouara150@gmail.com

Brahim Hamdi
Department of Mathematics, Faculty of exact sciences and computer science, Ziane Achour University, Djelfa, Algeria. e-mail: brahim.hamdi@univ-djelfa.dz

How to cite this article
Hamaidi, M., Briki, M., Nouara, A. and Hamdi, B., Space-time localized scheme to solve some partial integro-differential equations. *Iran. J. Numer. Anal. Optim.*, ??; ??(??): ??-??. ??

interpolated function, the method can be easily combined with the trapezoidal rule to find the numerical solution. The main advantages of such formulation are as follows: The time discretization is not applied; the time stability analysis is not discussed; and the recomputation of the resulting matrix at each time level is avoided because the matrix is computed once. Different examples are solved to show the accuracy of such a method.

AMS subject classifications (2020): Primary 45K05, 65M99; Secondary 65N22.

Keywords: Partial integro-differential equation; Localized radial basis functions; Space-time scheme; Collocation method; Trapezoidal rule.

1 Introduction

A partial integro-differential equation (PIDE) is an equation in which the unknown function appears under the sign of integration and contains the unknown function and its derivatives with respect to the space and time variables. Many problems in various fields of physical, engineering, biological, and epidemiology models are described by PIDEs.

The numerical solution of the PIDEs has recently gained much attention from researchers. To our best knowledge, in all published works, the solution methods are based on first discretizing the time variable by applying any time-stepping algorithms as implicit, explicit, Runge–Kutta or others, and seeking the approximate solution at each instant t in a space domain problem. Siddiqi and Arshed [14] employed cubic b-spline functions for spatial derivatives and the Euler backward formula for time derivatives to solve the PIDE. In [16, 15], it was used the 2-point Euler backward finite difference method was used for the discretization in time with a combination of the finite difference method and the trapezoidal rule to solve the PIDE. El-Sayed, Helal and El-Azab [4] implemented the implicit and explicit finite difference schemes for the time discretization. In most published works, these methods are based on differentiating between time and space variables. All methods start by discretizing the time variable using implicit, explicit, Runge–Kutta, or any other known method, and then solving the problem by

computing the approximate solution at each time t. The global radial basis function (RBF) method for solving the linear integro-differential equations was investigated by Golbabai and Seifollahi [6, 7]. Parand and Rad [12] presented the RBF collocation method for one-dimensional Volterra-Fredholm-Hammerstein integral equations. All works used the global formulation of RBF [6, 7, 12, 1, 17]. Therefore, in this paper, we develop an RBF-based space-time localized meshless collocation method combined with the trapezoidal rule to solve the space and time PIDE as space-time one, without differentiating between space and time variables. The posed problem can be solved once to approximating the solution at any space-time point (x, t). The main advantages of the considered technique are as follows:
(a). The discussion of the time stability analysis of the discrete system is avoided [8].
(b). The computational time when dealing with PIDEs with time-dependent coefficients is reduced as there is no need to recompute the matrix for the resulting algebraic system at each time level.
(c). The method uses the sparse matrices to store only the nonzero elements, so we save a significant amount of memory and speed up the resolution of the linear system.

The paper is organized as follows. In section 2, we introduce the formulation of the PIDE as a space-time problem and the space-time localized RBF method implementation. Section 3 is devoted to the discussion of results obtained by solving different PIDE examples. We conclude in Section 4.

2 Numerical details and discretization schemes

In this section, we describe the discretization scheme and the methodology used to solve the PIDE. The considered PIDE has the following form:

$$\begin{cases} \mathcal{D}_{(x,t)}u + \mathcal{I}_{(x,t)}u = f(x,t) & \text{for all } x \in (a,b), \text{ for all } t \in (0,T], \\ u(a,t) = g_1(t) & \text{for all } t \in (0,T), \\ u(b,t) = g_2(t) & \text{for all } t \in (0,T), \\ u(x,0) = u_0(x) & \text{for all } x \in [a,b], \end{cases} \tag{1}$$

where $\mathcal{D}_{(x,t)}$ is a differential operator of second order with variable coefficients defined by

$$\mathcal{D}_{(x,t)}u = \frac{\partial u}{\partial t} - a(x,t)\frac{\partial^2 u}{\partial x^2} + b(x,t)\frac{\partial u}{\partial x} + c(x,t)u, \tag{2}$$

$\mathcal{I}_{(x,t)}$ is an integral operator of the form

$$\mathcal{I}_{(x,t)}u = \int_0^t k(x,t,s)u(x,s)ds, \tag{3}$$

and f, g_1, g_2, u_0, and k are given smooth functions.

2.1 Space-time problem methodology

The formulation of the time-depends problem given by the system (1) as a space-time one starts by combining the space variable x and the time variable t in one vector $\hat{x} = (x,t)$. The constructed variable vector belongs to the space-time domain $\Omega_T = [a,b]\times[0,T]$ represented by Figure 1. The boundary of the new formulated domain Ω_T is given by $\partial\Omega_T = \Gamma_1 \cup \Gamma_2 \cup \Gamma_3 \cup \Gamma_4$, where $\Gamma_1 = \{a\} \times [0,T]$, $\Gamma_2 = \{b\} \times [0,T]$, $\Gamma_3 = [a,b] \times \{0\}$, and $\Gamma_4 = [a,b] \times \{T\}$.

Then, the problem has the new form:

$$\begin{cases} \mathcal{D}_{(x,t)}u + \mathcal{I}_{(x,t)}u = f(x,t) & \text{for all } (x,t) \in \Omega_T \\ u(x,t) = g_1(t) & \text{for all } (x,t) \in \Gamma_1, \\ u(x,t) = g_2(t) & \text{for all } (x,t) \in \Gamma_2, \\ u(x,t) = u_0(x) & \text{for all } (x,t) \in \Gamma_3, \\ \mathcal{D}_{(x,t)}u + \mathcal{I}_{(x,t)}u = f(x,t) & \text{for all } (x,t) \in \Gamma_4, \end{cases} \tag{4}$$

or in a reduced form, by setting $\Omega_T = \Omega_T \cup \Gamma 4$ and $\partial\Omega_T = \Gamma_1 \cup \Gamma_2 \cup \Gamma_3$, we have

$$\begin{cases} \mathcal{D}_{\hat{x}}u + \mathcal{I}_{\hat{x}}u = f(\hat{x}) & \text{for all } \hat{x} \in \Omega_T, \\ u(\hat{x}) = g(\hat{x}) & \text{for all } \hat{x} \in \partial\Omega_T, \end{cases} \tag{5}$$

where

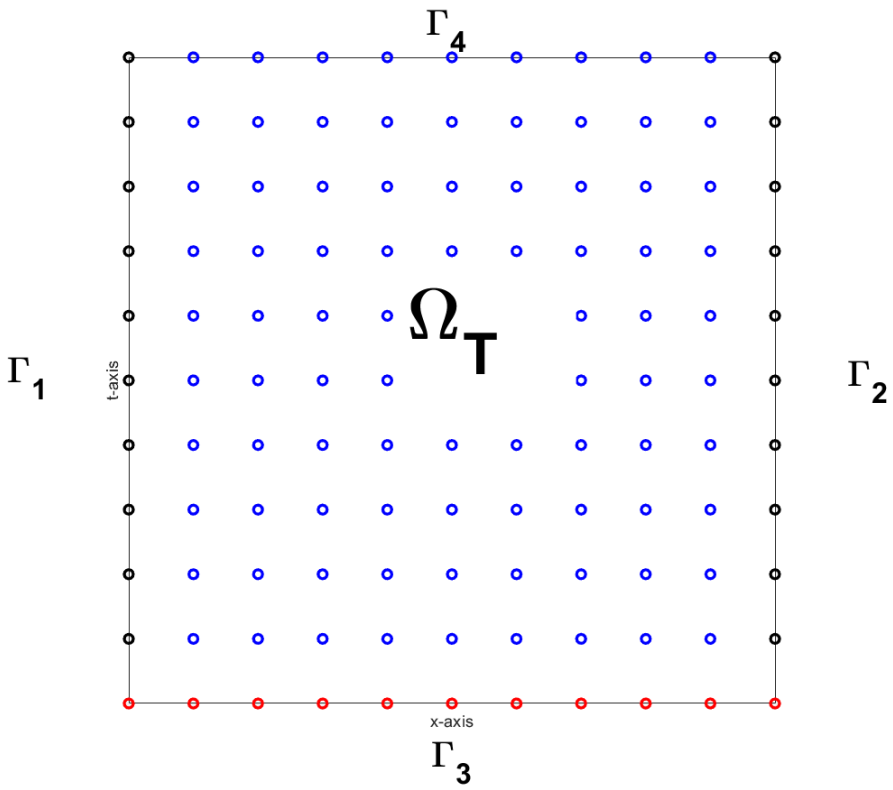

Figure 1: Space-time domain

$$\begin{cases} g(\hat{x}) = g_1(\hat{x}) & \text{for all } \hat{x} \in \Gamma_1, \\ g(\hat{x}) = g_2(\hat{x}) & \text{for all } \hat{x} \in \Gamma_2, \\ g(\hat{x}) = u_0(\hat{x}) & \text{for all } \hat{x} \in \Gamma_3. \end{cases} \tag{6}$$

2.2 The space-time localized RBFs scheme

To recall the technique, let $\{\hat{x}_i\}_{i=1}^{N_i}$ and $\{\hat{x}_i\}_{i=N_i+1}^{N}$ be center nodes in Ω_T and $\partial\Omega_T$, respectively (Interior and boundary nodes, where N is the total number of nodes in the space-time domain Ω_T). To approximate the differential operator, using the localized RBF method, we first need to derive the local approximation of the unknown function u [2, 3]. Then the local approx-

imation of $\mathcal{D}_{(x,t)}u(x,t)$ can be determined easily based on the components of the function u. So, the local approximation of u in an influence domain Ω_T^j associated with a selecting collocation point $\hat{x}_j = (x_j, t_j)$ and containing a number n_j of nearest neighboring points $\{\hat{x}_k^{[j]} = (x_k^{[j]}, t_k^{[j]})\}_{k=1}^{n_j} \in \Omega_T^j$, is given by

$$u(\hat{x}_j) \simeq \hat{u}(\hat{x}_j) = \sum_{k=1}^{n_j} \alpha_k \phi(\|\hat{x}_j - \hat{x}_k^{[j]}\|), \tag{7}$$

where $\{\alpha_k\}_{k=1}^{n_j}$ are the unknown coefficients, $\|\cdot\|$ is the Euclidean norm, and ϕ is the chosen RBF. There are many different RBFs to choose from. Among them we can mention the multiquadric function $\phi(r) = \sqrt{1+(\epsilon r)^2}$, Which has been proven in many references [11, 5, 13] to be the most effective over the past few decades (The real ϵ is the shape parameter of the RBF).

Using the collocation method, (7) is then applied to all collocation points $\{\hat{x}_k^{[j]}\}_{k=1}^{n_j}$ belonging to the influence domain Ω_T^j of $\hat{x}_j$. Then we have the following $n_j \times n_j$ linear system:

$$\widehat{\mathbf{u}}^{[j]} = {}^{[j]}\boldsymbol{\alpha}^{[j]}, \tag{8}$$

where ${}^{[j]} = \left[\phi(\|\hat{x}_m^{[j]} - \hat{x}_n^{[j]}\|)\right]_{1\le m,n\le n_j}$, $\boldsymbol{\alpha}^{[j]} = \left[\alpha_1^{[j]}, \alpha_2^{[j]}, \ldots, \alpha_{n_j}^{[j]}\right]$, and $\widehat{\mathbf{u}}^{[j]} = \left[u(\hat{x}_1^{[j]}), u(\hat{x}_2^{[j]}), \ldots, u(\hat{x}_{n_j}^{[j]})\right]$.

Then, the problem of seeking the expansion coefficients $\{\alpha_k\}_{k=1}^{n_j}$ is transformed into a determination of the values of solution $\hat{u}^{[j]}$ at each center point $\{\hat{x}_k^{[j]}\}_{k=1}^{n_j} \subset \Omega_T^j$ by using the equation

$$\boldsymbol{\alpha}^{[j]} = ({}^{[j]})^{-1} \cdot \widehat{\mathbf{u}}^{[j]}. \tag{9}$$

The local approximation of $\mathcal{D}_{\hat{x}}u$ can be determined by applying the differential operators $\mathcal{D}_{\hat{x}}$ to the equation (7) for any selected center point $\hat{x}_j$ in any sub-domain Ω_j (Figure 2). For $\hat{x}_j \in \Omega_j$, we obtain the following equation:

$$\begin{aligned} \mathcal{D}_{\hat{x}}\hat{u}(\hat{x}_j) &= \sum_{k=1}^{n_j} \alpha_k \mathcal{D}_{\hat{x}}\phi(\|\hat{x}_j - \hat{x}_k^{[j]}\|) \\ &= \mathbf{D}^{[\mathbf{j}]} \cdot \widehat{\mathbf{u}}^{[\mathbf{j}]} \\ &= \mathbf{D}^{[\mathbf{j}]} \cdot \widehat{\mathbf{u}}, \end{aligned} \tag{10}$$

where $\widehat{\mathbf{u}}^{[j]} = \left[u(\hat{x}_1^{[j]}), u(\hat{x}_2^{[j]}), \ldots, u(\hat{x}_{n_j}^{[j]})\right]$ and $\mathbf{D}^{[j]} = (\mathcal{D}_{\hat{x}}\ ^{[j]}) \cdot (\ ^{[j]})^{-1}$.

To switch from the local system (10) to the global one, the vector $\widehat{\mathbf{u}} = [u(\hat{x}_1), u(\hat{x}_2), \ldots, u(\hat{x}_N)]$ is incorporated in (10) by adding zeros at the proper locations based on the mapping of $\widehat{\mathbf{u}}^{[j]}$ to $\widehat{\mathbf{u}}$, and considering $\mathbf{D}^{[j]}_{1\times N}$ as the global expansions of $\mathbf{D}^{[j]}_{1\times n_j}$. For more details on space-time and localized RBFs collocation method, see [8, 9, 10].

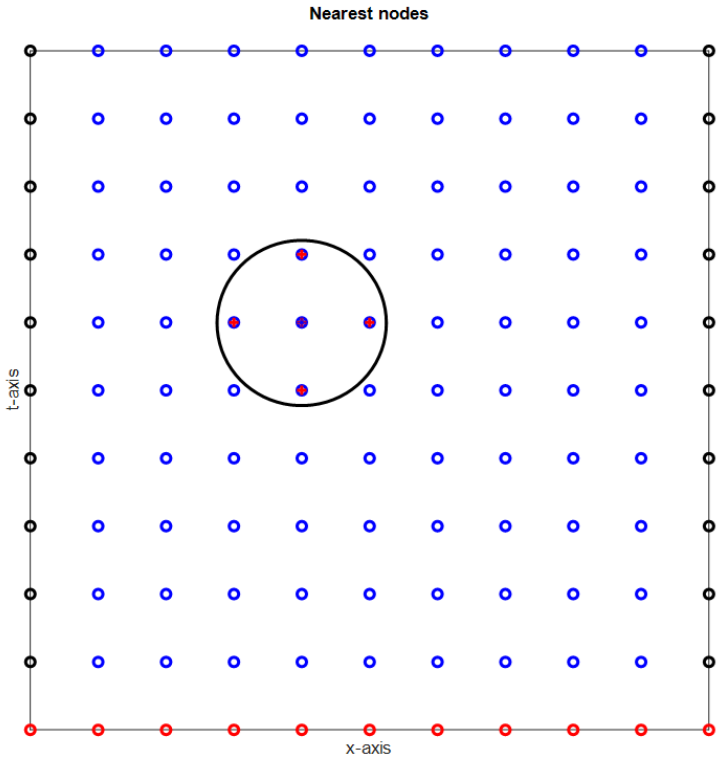

Figure 2: Nearest nodes

2.3 The trapezoidal rule on space-time

In the pace-time domain, the integral part of (1) can be achieved by the trapezoidal rule. First, for each node $\hat{x}_i = (x_i, t_i)$, we determine the in-

tegration nodes in a space-time influence domain I_T^i having p_i elements. With uniform distribution nodes, we have $I_T^i = \{(x_i, t_i), (x_i, t_i - h_t), (x_i, t_i - 2h_t), \ldots, (x_i, 0)\}$ and $h_t = \frac{T}{N_t}$, where N_t is the number of nodes on the time axis. Then the integral can be calculated by the trapezoidal rule. The integration nodes are shown in Figure 3.

We have

$$\mathcal{I}_{\hat{x}} u = \int_0^t k(x, t, s) u(x, s) ds. \tag{11}$$

Then, the discretization of $\mathcal{I}_{\hat{x}} u$ is done as follows:

$$\begin{aligned} \mathcal{I}_{\hat{x}} u(\hat{x}_i) &\approx \hat{\mathcal{I}}_{\hat{x}} u_i \\ &= \frac{h_t}{2} k(x_i, t_i, 0) u(x_i, 0) + h_t \sum_{k=1}^{p_i - 1} k(x_i, t_i, t_k) u(x_i, t_k) \\ &\quad + \tfrac{h_t}{2} k(x_i, t_i, t_i) u(x_i, t_i) \\ &= \frac{h_t}{2} k(x_i, t_i, 0) u_1^{[i]} + h_t \sum_{k=1}^{p_i - 1} k(x_i, t_i, t_k) u_k^{[i]} + \frac{h_t}{2} k(x_i, t_i, t_i) u_{p_i}^{[i]}. \end{aligned} \tag{12}$$

In reduced form, we have

$$\hat{\mathcal{I}}_{\hat{x}} u_i = \mathbf{I}^{[i]} \cdot \widehat{\mathbf{u}}^{[i]}, \tag{13}$$

where $\mathbf{I}^{[i]} = \left[\frac{h_t}{2} k(x_i, t_i, 0), \{h_t k(x_i, t_i, t_k)\}_{1 \le k \le p_i - 1}, \frac{h_t}{2} k(x_i, t_i, t_i)\right]$ and $\widehat{\mathbf{u}}^{[j]} = [u(\hat{x}_0^{[i]}), u(\hat{x}_1^{[i]}), \ldots, u(\hat{x}_{p_i}^{[i]})]$.

Similar to the localized scheme, the vector $\widehat{\mathbf{u}} = [u(\hat{x}_1), u(\hat{x}_2), \ldots, u(\hat{x}_N)]$ is incorporated in (13) by adding zeros at the proper locations based on the mapping of $\widehat{\mathbf{u}}^{[i]}$ to $\widehat{\mathbf{u}}$, and considering $\mathbf{I}_{1 \times N}$ as the global expansions of $\mathbf{I}_{1 \times p_i}^{[i]}$.

2.4 The combined scheme

For each node $\hat{x}_i$ in the space-time domain Ω_T, we determine the nearest nodes $\{\hat{x}_k \in \Omega_T^i\}$ and the integration nodes $\{\hat{x}_k \in \mathbf{I}^{[i]}\}$. Then we compute the coefficients of $\widehat{\mathbf{u}}$ by

$$\begin{cases} \mathcal{D}_{\hat{x}} \hat{u}_i = \mathbf{D}^{[\mathbf{i}]} \cdot \widehat{\mathbf{u}}, \\ \hat{\mathcal{I}}_{\hat{x}} u_i = \mathbf{I}^{[i]} \cdot \widehat{\mathbf{u}}, \end{cases} \tag{14}$$

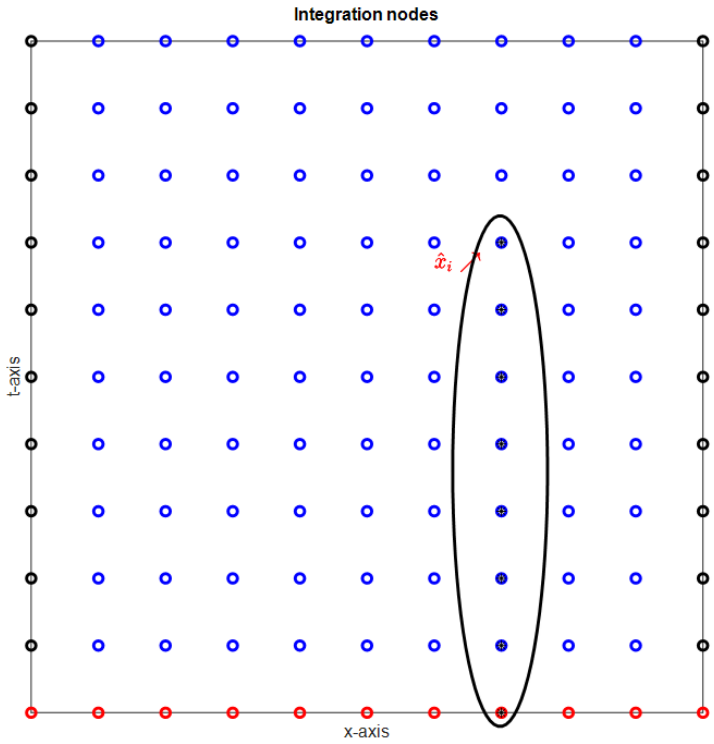

Figure 3: Trapezoidal rule nodes

and we get

$$\begin{cases} \mathcal{D}_{\hat{x}}\hat{u}_i + \hat{\mathcal{I}}_{\hat{x}}u_i = \mathbf{D}^{[\mathbf{i}]} \cdot \widehat{\mathbf{u}} + \mathbf{I}^{[i]} \cdot \widehat{\mathbf{u}} \\ \qquad\qquad\qquad = (\mathbf{D}^{[\mathbf{i}]} + \mathbf{I}^{[i]}) \cdot \widehat{\mathbf{u}}. \end{cases} \tag{15}$$

By substituting (15) into (5) for $x_i \in \Omega_T$, we obtain

$$f(\hat{x}_i) = \mathcal{D}_{\hat{x}}\hat{u}_i + \hat{\mathcal{I}}_{\hat{x}}u_i = (\mathbf{D}^{[\mathbf{i}]} + \mathbf{I}^{[i]}) \cdot \widehat{\mathbf{u}}. \tag{16}$$

For $x_i \in \partial\Omega_T$, we have

$$g(\hat{x}_i) = \widehat{\mathbf{u}}_\mathbf{i}. \tag{17}$$

By collocating all the interpolation points $\{x_j\}_{j=1}^{N}$ and using (16) and (17), we get the following sparse linear system:

$$\boldsymbol{\mathcal{A}}\mathbf{U} = \boldsymbol{\mathcal{B}}, \tag{18}$$

$$\text{where } \boldsymbol{\mathcal{A}} = \begin{bmatrix} (\mathbf{D}+\mathbf{I})(\hat{x}_1) \\ (\mathbf{D}+\mathbf{I})(\hat{x}_2) \\ \vdots \\ (\mathbf{D}+\mathbf{I})(\hat{x}_{N_i}) \\ \mathbb{1}_{N_i+1} \\ \vdots \\ \mathbb{1}_N \end{bmatrix}, \mathbf{U} = \begin{bmatrix} \hat{u}_1 \\ \hat{u}_2 \\ \vdots \\ \hat{u}_{N_i} \\ \hat{u}_{N_i+1} \\ \vdots \\ \hat{u}_N \end{bmatrix}, \text{ and } \boldsymbol{\mathcal{B}} = \begin{bmatrix} f(\hat{x}_1) \\ f(\hat{x}_2) \\ \vdots \\ f(\hat{x}_{N_i}) \\ g(\hat{x}_{N_i+1}) \\ \vdots \\ g(\hat{x}_N) \end{bmatrix}, \text{ where}$$

$$\mathbb{1}_j \cdot \hat{u} = \hat{u}_j, \quad \text{for all } j \in [N_i+1, N].$$

Note that the linear algebraic system (18) is square since the number of unknowns (the values of the approximate function) and the collocation points are equal. The approximate solution $\{\hat{u}(\hat{x_j})\}_{j=1}^{N}$ at the interpolation points $\{\hat{x_j}\}_{j=1}^{N}$ can be obtained by solving the above sparse linear system of equations.

In practice, the mapping from $\mathbf{D}^{[i]}$ to $\mathbf{D}$ and $\mathbf{I}^{[i]}$ to $\mathbf{I}$ is automatic without the need for inserting zeros, if we make good use of the index vector and store the sparse matrix properly. Figure 4 shows the sparse matrix of the system for some values of N. The **nz** value designates the nonzero elements in the matrix.

2.5 Rate of convergence of the scheme

It is well known that the trapezoidal rule has a quadratic accuracy, and we [8] demonstrated numerically that the rate of the localized space-time also has a quadratic convergence. So we can assert that the proposed scheme in this work also has a numerical quadratic rate of convergence.

The experimental rates of convergence with respect to the mesh size $h = \sup_{\hat{x}\in\Omega_T} \min_{\hat{x}_j} \|\hat{x} - \hat{x}_j\|$ are calculated using the following formula:

$$ROC = \frac{\log\left(\frac{E_{i+1}}{E_i}\right)}{\log\left(\frac{h_{i+1}}{h_i}\right)},$$

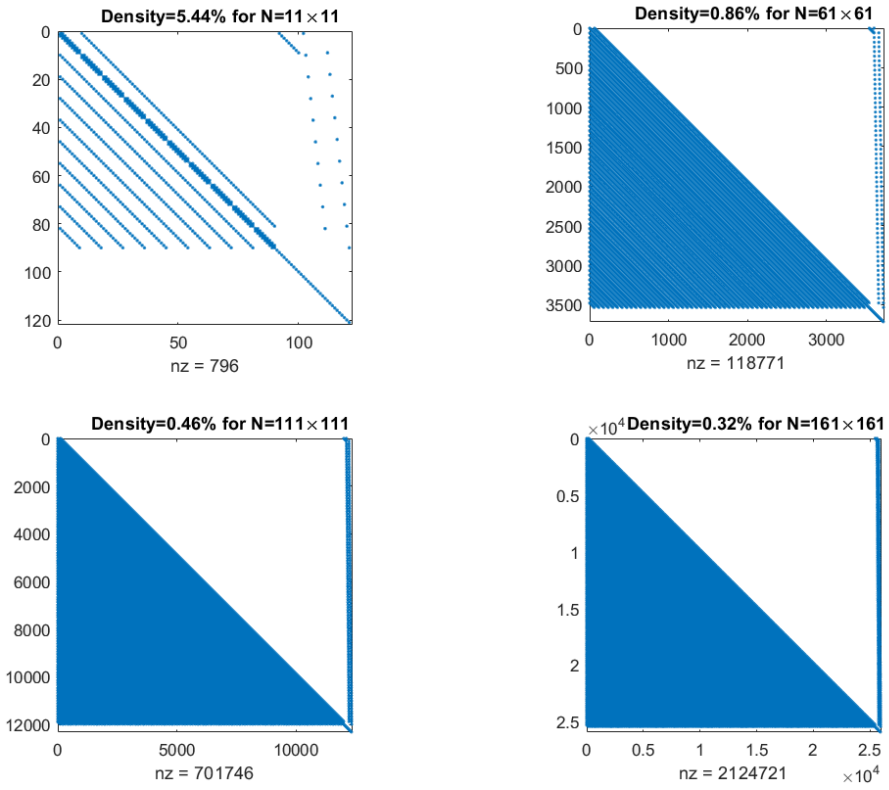

Figure 4: Density of the sparse matrix for some values of N

where E_i is one of the specified errors MAE, $RMSE$, or L_{er}^1 corresponding to the mesh size h_i.

3 Numerical results and discussions

In this section, we investigate the numerical solution of the PIDE using a spacetime localized RBF collocation method to show its efficiency and accuracy for solving such a problem.

To measure the numerical accuracy, we consider the maximum absolute error (MAE), the root mean squared error (RMSE), and the L_{er}^1 relative error defined as follows:

$$
\begin{aligned}
MAE &= \max_{1\le j\le N} |\hat{u}(\hat{x}_j) - u(\hat{x}_j)|,\\
RMSE &= \sqrt{\frac{1}{N}\sum_{j=1}^{N}(\hat{u}(\hat{x}_j) - u(\hat{x}_j))^2},\\
L^1_{er} &= \frac{\sum_{j=1}^{N} |\hat{u}(\hat{x}_j) - u(\hat{x}_j)|}{\sum_{j=1}^{N} |u(\hat{x}_j)|},
\end{aligned}
\tag{19}
$$

where $u(\hat{x}_j)$ and $\hat{u}(\hat{x}_j)$ are the exact and the approximate solutions at the node $\hat{x}_j$, respectively. We considered the maximum absolute error to show that there is no big error. The RMSE is more significant; it measures the average magnitude of the error. Moreover, L^1_{er} shows that even for "big" errors, it is relatively small. Since the norms are equivalent, RMSE suffices.

For all treated examples, the uniform node distribution is adopted. The choice of the shape parameter is not discussed, and it is fixed at $\epsilon = 1$. The number of nearest nodes is chosen $n_j = 5$ as the problems treated are in two-dimensional space-time domains [8].

3.1 Example 1

The first example treated is a diffusion integro-differential problem of the form:

$$\frac{\partial u}{\partial t}(x,t) - \frac{\partial^2 u}{\partial x^2}(x,t) - \int_0^t k(t,s)u(x,s)ds = f(x,t) \ on \ \ \Omega_T = (0,1)^2,$$

where $k(t,s) = st$. The function f and the boundary conditions on space-time domain are chosen according to the analytical solution:

$$u(x,t) = \sin(\pi x)e^{-\pi^2 t}.$$

Table 1 shows errors for different values of N, the total interpolation points in the space-time domain. The CPU time and matrix size are also given.

The experimental rate of convergence in this simulation shows that nearly quadratic convergence is achieved. Figure 5 shows the exact and numerical solution and the absolute error on the entire domain $\Omega_T = (0,1)^2$ with $N = 101^2$. It can be noted that the results have good accuracy.

Table 1: Errors for Example 3.1 for some values of $N = N_x \times N_t$

N	h	MAE	ROC	RMSE	ROC	L^1_{er}	ROC	Time	Size
11^2	0.100	3.76E-02	0.000	1.05E-02	0.000	6.40E-02	0.000	0.010	121^2
21^2	0.050	1.12E-02	1.751	3.27E-03	1.686	2.39E-02	1.418	0.016	441^2
31^2	0.033	5.15E-03	1.904	1.53E-03	1.872	1.20E-02	1.709	0.031	961^2
41^2	0.025	2.94E-03	1.950	8.80E-04	1.926	7.11E-03	1.812	0.094	1681^2
51^2	0.020	1.90E-03	1.970	5.69E-04	1.951	4.69E-03	1.862	0.125	2601^2
61^2	0.017	1.32E-03	1.980	3.98E-04	1.964	3.32E-03	1.891	0.250	3721^2
71^2	0.014	9.73E-04	1.985	2.94E-04	1.972	2.48E-03	1.910	0.422	5041^2
81^2	0.013	7.46E-04	1.989	2.26E-04	1.977	1.92E-03	1.924	0.563	6561^2
91^2	0.011	5.90E-04	1.991	1.79E-04	1.980	1.53E-03	1.934	0.766	8281^2
101^2	0.010	4.78E-04	1.994	1.45E-04	1.984	1.24E-03	1.942	0.969	10201^2
111^2	0.009	3.95E-04	1.997	1.20E-04	1.988	1.03E-03	1.950	1.328	12321^2
121^2	0.008	3.33E-04	1.988	1.01E-04	1.980	8.71E-04	1.946	1.703	14641^2
131^2	0.008	2.83E-04	2.008	8.60E-05	2.001	7.44E-04	1.969	2.203	17161^2
141^2	0.007	2.44E-04	1.982	7.43E-05	1.975	6.44E-04	1.946	2.719	19881^2
151^2	0.007	2.13E-04	2.006	6.47E-05	1.999	5.62E-04	1.972	3.391	22801^2
161^2	0.006	1.87E-04	2.015	5.68E-05	2.009	4.95E-04	1.984	4.109	25921^2

3.2 Example 2

As a second example, the following advection-diffusion integro-differential equation is considered [14]:

$$\frac{\partial u}{\partial t}(x,t) - a\frac{\partial^2 u}{\partial x^2}(x,t) + b\frac{\partial u}{\partial x}(x,t) - \int_0^t k(t,s)u(x,s)ds = f(x,t) \; on \; \Omega_T = (0,1)^2,$$

where $a = 0.4$, $b = 0.05$, and $k(t,s) = \sqrt{t-s}$. The other parameters are taken according to the analytical solution:

$$u(x,t) = (t^2+1)\sin(\pi x).$$

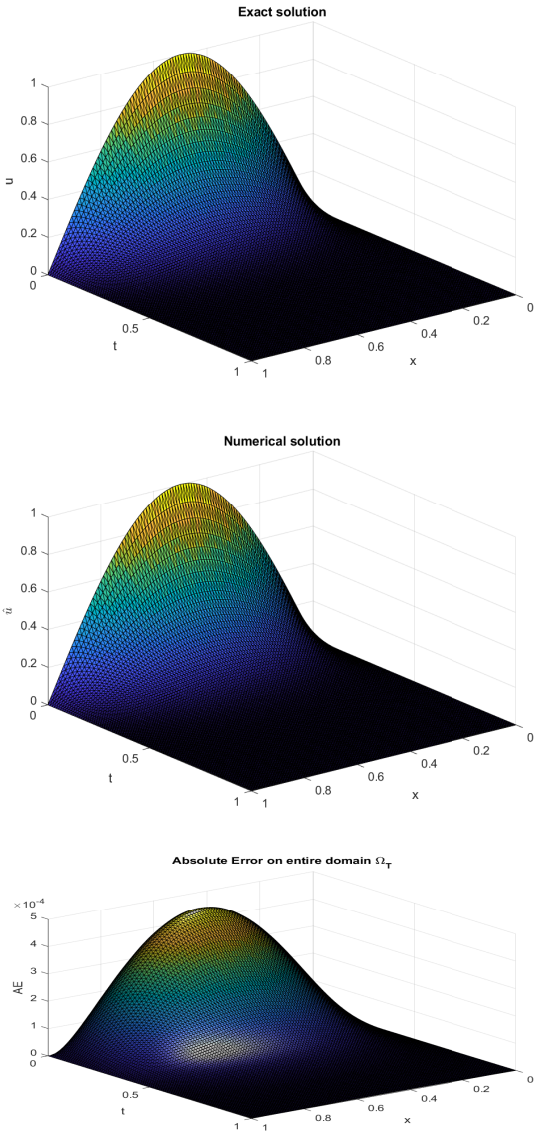

Figure 5: Exact and numerical solutions and the absolute error for the example 3.1 with $N = 101^2$.

Table 2 shows the results of some simulations for Example 3.2. Same remarks are noted as in Example 3.1 concerning the accuracy and the rate of convergence.

Table 2: Errors for Example 3.2 for some values of N

N	h	MAE	ROC	$RMSE$	ROC	L^1_{er}	ROC
21^2	0.0500	8.889E-03		2.25E-03		1.69E-03	
41^2	0.0250	2.474E-03	1.84508	6.01E-04	1.90682	4.49E-04	1.91278
61^2	0.0167	1.157E-03	1.87405	2.79E-04	1.89447	2.09E-04	1.88484
81^2	0.0125	6.740E-04	1.87914	1.62E-04	1.88032	1.22E-04	1.86126
101^2	0.0100	4.432E-04	1.87820	1.07E-04	1.86698	8.11E-05	1.83983
121^2	0.0083	3.148E-04	1.87711	7.63E-05	1.85763	5.82E-05	1.82456
141^2	0.0071	2.361E-04	1.86681	5.74E-05	1.83785	4.41E-05	1.79623
161^2	0.0063	1.842E-04	1.85918	4.50E-05	1.82052	3.48E-05	1.77233

3.3 Example 3

For this third example, more challenging problem is treated. The advection-diffusion-reaction integro-differential equation with variable coefficients is defined by

$$\frac{\partial u}{\partial t}(x,t) - a(x,t)\frac{\partial^2 u}{\partial x^2}(x,t) + b(x,t)\frac{\partial u}{\partial x}(x,t) + c(x,t)u(x,t) - \int_0^t k(t,s)u(x,s)ds = f(x,t),$$

where $a(x,t) = e^{-t}$, $b = \sin(x)$, $c(x,t) = e^t$, and $k(t,s) = (t-s)$. The other parameters are chosen according to the analytical solution:

$$u(x,t) = e^{-t}\cos(x)$$

and the space-time domain is $\Omega_T = [0, \frac{\pi}{2}] \times [0,1]$.

We can observe from Table 3 that even for this kind of complex problem, the results are very accurate. The same rate of convergence is observed. Because we arrived at $N = 201^2 = 40401$ nodes and the size of the matrix $\mathcal{A}$ defined in (18) is 40401×40401, without losing accuracy, we can assert that the scheme is also stable.

Table 3: Errors for Example 3.3 for some values of N

N	h	MAE	ROC	$RMSE$	ROC	L_{er}^1	ROC
11^2	0.1000	3.451E-03		1.65E-03		3.19E-03	
21^2	0.0500	9.272E-04	1.89609	4.65E-04	1.82568	9.47E-04	1.75046
31^2	0.0333	4.195E-04	1.95598	2.13E-04	1.92592	4.40E-04	1.88872
41^2	0.0250	2.374E-04	1.97869	1.21E-04	1.95451	2.53E-04	1.92983
51^2	0.0200	1.522E-04	1.99211	7.82E-05	1.96759	1.64E-04	1.94926
61^2	0.0167	1.060E-04	1.98399	5.46E-05	1.97497	1.14E-04	1.96039
71^2	0.0143	7.794E-05	1.99569	4.02E-05	1.97968	8.45E-05	1.96754
81^2	0.0125	5.972E-05	1.99384	3.09E-05	1.98292	6.49E-05	1.97237
91^2	0.0111	4.721E-05	1.99551	2.44E-05	1.98531	5.15E-05	1.97605
101^2	0.0100	3.824E-05	1.99960	1.98E-05	1.98753	4.18E-05	1.97932
111^2	0.0091	3.163E-05	1.99372	1.64E-05	1.98745	3.46E-05	1.98002
121^2	0.0083	2.657E-05	2.00062	1.38E-05	1.98999	2.91E-05	1.98332
131^2	0.0077	2.265E-05	1.99703	1.18E-05	1.99127	2.48E-05	1.98521
141^2	0.0071	1.952E-05	2.00184	1.01E-05	1.99476	2.14E-05	1.98928
151^2	0.0067	1.701E-05	1.99955	8.84E-06	1.99140	1.87E-05	1.98606
161^2	0.0063	1.495E-05	2.00023	7.77E-06	1.99566	1.64E-05	1.99084
171^2	0.0059	1.326E-05	1.98254	6.89E-06	1.97510	1.46E-05	1.96957
181^2	0.0056	1.181E-05	2.02565	6.14E-06	2.02247	1.30E-05	2.01969
191^2	0.0053	1.060E-05	1.98603	5.52E-06	1.98652	1.17E-05	1.98243
201^2	0.0050	9.585E-06	1.97164	4.99E-06	1.95884	1.06E-05	1.95271

4 Conclusion

In this paper, we presented a local space-time RBFs collocation method combined with the trapezoidal rule to solve the PIDE as space-time one without differentiating between space and time variables. The problem is solved once to approximate the solution at any point (x, t). The main advantages of the considered technique are as follows:

(1). The discussion of the time stability analysis of the discrete system is avoided.

(2). The computational time when dealing with PIDEs with time-dependent coefficients is reduced as there is no need to recompute the matrix for the resulting algebraic system at each time level.

(3). The method uses the sparse matrices to store only the nonzero elements, so we save a significant amount of memory and speed up the resolution of the linear system.
(4). The formulation is the same for any form of the linear differential operator of second order $\mathcal{D}_{\hat{x}}$ and any form of the function $k(x,t,s)$, only some changes in the programming script can be made according to the problem to be solved.

It has been demonstrated that our technique is simple, straightforward, and applicable to a large type of problems as it is shown in this paper. The application of the developed technique to equations with an integral boundary condition is under investigation. Further work will focus on developing a method without the trapezoidal rule to solve such a problem.

Conflicts of Interest

We have no conflicts of interest to disclose. All authors declare that they have no conflicts of interest.

Funding

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Data availability

No new data were created or analysed in this study. Data sharing is not applicable to this article.

Acknowledgements

Authors are grateful to there anonymous referees and editor for their constructive comments.

References

[1] Alipanah, A. and Esmaeili, S. *Numerical solution of the two-dimensional Fredholm integral equations using Gaussian radial basis function,* J. Comput. Appl. Math. 235(18) (2011) 5342–5347.

[2] Chen, C.S., Fan, C.M. and Wen, P.H. *The method of particular solutions for solving elliptic problems with variable coefficients,* Int. J. Comput. Method. 8 (2011) 545–559.

[3] Chen, C.S., Fan, C.M. and Wen, P.H. *The method of particular solutions for solving certain partial differential equations,* Numer. Method. Partial Differ. Equ. 28 (2012) 506–522.

[4] El-Sayed, A.M.A., Helal, S.M. and El-Azab, M.S.*Solution of a parabolic weakly-singular partial integro-differential equation with multi-point nonlocal boundary conditions,* J. Fract. Calc. Appl. 7(1) (2016) 1–11.

[5] Fornberg, B. and Flyer, N. *Accuracy of radial basis function interpolation and derivative approximations on 1-D infinite grids,* Adv. Comput. Math. 23 (2005) 5–20.

[6] Golbabai, A. and S. Seifollahi, *Numerical solution of the second-kind integral equations using radial basis function networks,* Appl. Math. Comput. 174 (2) (2006) 877–883.

[7] Golbabai, A. and Seifollahi, S. *Radial basis function networks in the numerical solution of linear integro-differential equations,* Appl. Math. Comput. 188 (1) (2007) 427–432.

[8] Hamaidi, M., Naji, A. and Charafi, A. *Space-time localized radial basis function collocation method for solving parabolic and hyperbolic equations,* Eng. Anal. Bound. Elem. 67 (2016) 152–163.

[9] Hamaidi, M., Naji, A., Ghafrani, F. and Jourhmane, M. *Noniterative localized and space-time localized RBF meshless method to solve the ill-posed and inverse problem,* Model. Simul. Eng. 2020 (2020) 5046286.

[10] Hamaidi, M., Naji, A. and Taik, A. *Solving parabolic and hyperbolic equations with variable coefficients using space-time localized radial basis function collocation method,* Model. Simul. Eng. 2021 (2021) 6688806.

[11] Jiang, Z.W., Wang, R.H., Zhu, C.G. and Xu, M. *High accuracy multiquadric quasi-interpolation,* Appl. Math. Model. 35 (5) (2011) 2185–2195.

[12] Parand, K. and Rad, J.A.*Numerical solution of nonlinear Volterra-Fredholm-Hammerstein integral equations via collocation method based on radial basis functions,* Appl. Math. Comput. 218 (9) (2012) 5292–5309.

[13] Scott, S. and Kansa, E.J. *Multiquadric radial basis function approximation methods for the numerical solution of partial differential equations,* Adv. Comput. Mech. 2(2) (2009) 220.

[14] Siddiqi, S.S. and Arshed, S. *Numerical solution of convection-diffusion integro-differential equations with a weakly singular kernel,* J. Basic Appl. Sci. Res. 3(11) (2013) 106–120.

[15] Soliman, A.F., El-Asyed, A.M.A. and El-Azab, M.S.*On the numerical solution of partial integro-differential equations,* Math. Sci. Lett. 1(1) (2012) 71–80.

[16] Soliman, A.F., El-Azab, M.S. and El-Asyed, A. *Fourth and sixth order compact finite difference Schemes for partial integro-differential equations,* J. Math. Comput. Sci. 2 (2) (2013) 206–225.

[17] Zhang, H.Q., Chen, Y., and Nie, X. *Solving the linear integral equations based on radial basis function interpolation,* J. Appl. Math. 2014 (2014), 793582.