JAMDA

J. Appl. Math. & Data. Anal.

Available online at https://jamda.abru.ac.ir/
**Journal of Applied Mathematics & Data Analytics** doi:
Vol. 1, No. 3, 2025, 15-34
*Research Article*

# Inverse heat conduction problem with a nonlinear source term through radial basis function partition of unity collocation method

Ahmad Jafarabadi*
Department of Applied Mathematics, Imam Khomeini International University,
Qazvin, 34149-16818, Iran

**Abstract**

**Abstract:** This study focuses on determining the numerical solution for the surface heat flux history and temperature distribution in an inverse heat conduction problem (IHCP) with a nonlinear source term. The problem is addressed using a temperature over-specification condition as an energy over-specification condition across the computational domain. The proposed approach employs the radial basis function partition of unity collocation method combined with the finite difference method for temporal discretization. This meshless method eliminates the need for mesh generation, and its local formulation at each time step results in a sparse coefficient matrix, significantly reducing computational cost. Despite the problem having a unique solution, it remains ill-posed, as small perturbations in the input data can lead to large errors in the output. Numerical results demonstrate that the proposed method provides accurate solutions for exact data and maintains stability when handling noisy data.
**Keywords:** Partition of Unity (PU) method; Meshless interpolation; Inverse heat conduction problem; Surface heat flux.

## 1 Introduction

Parabolic inverse problems play a crucial role in the modeling of various physical processes. They arise in numerous branches of physics and engineering, including the study of heat transfer, thermoplasticity, chemical diffusion, and control theory.

By nature, inverse problems are inherently unstable, since the unknown parameters or solutions must be reconstructed from indirect observational data that are often contaminated with measurement errors. The principal challenge in constructing any numerical scheme for approximating their solutions lies in the severe ill-posedness of the problem and the ill-conditioning of the resulting discretized systems [1].

*Corresponding author: jafarabdi.ahmad@yahoo.com

Among different types of parabolic inverse problems, as examined in [2, 3, 4, 5, 6, 7], the inverse heat conduction problem (IHCP) is particularly ill-posed, meaning that even a small perturbation in the input data may lead to a significant deviation in the solution. The IHCP appears in the analysis and regulation of processes involving heat propagation in thermophysics and continuum mechanics. It arises whenever surface temperatures or heat fluxes at inaccessible portions of a boundary must be inferred from measurements taken on accessible parts [1]. For instance, practical problems in physics and engineering include the estimation of boundary conditions and radiative properties in semi-transparent materials, as well as the reconstruction of boundary heat fluxes and inlet conditions within ducts subject to forced convection. In direct problems, the diffusion mechanism reduces the effect of measurement noise at the boundary or interior, whereas in inverse problems, the under-determined nature amplifies these errors, making the problem highly ill-posed. Moreover, mathematical approaches for handling heat transfer in anisotropic materials differ significantly from those in isotropic cases, due to the tensorial structure of heat flux. This leads to the presence of mixed partial derivatives in the governing differential equations, preventing the straightforward use of classical separation of variables. Similarly, integral transform techniques such as Green's functions or Fourier and Laplace transforms require semi-infinite or infinite spatial domains, restricting their applicability [8].

A variety of contributions have been made in recent decades to study different aspects of the inverse heat conduction problem, with a comprehensive survey presented in [8]. For example, the authors of [8] proposed a representation involving linear combinations of fundamental solutions and heat polynomials to address a two-dimensional anisotropic IHCP. Shidfar et al. employed Chebyshev polynomials for flux approximation in [9]. Dehghan et al. presented two procedures based on the Bernstein multi-scaling approximation and cubic B-spline scaling functions for nonlinear IHCPs in [10]. In [11], a sequential conjugate gradient technique was developed to reconstruct unknown boundary fluxes in nonlinear IHCPs, effectively merging the sequential function specification method (SFSM) with the conjugate gradient method (CGM). Fernandes et al. [12] proposed an impulse response (transfer function) approach rooted in Green's function theory, emphasizing analogies between thermal and dynamical systems. Ku et al. [13] applied a collocation scheme using space-time radial polynomial series functions (SRPSF) to tackle two-dimensional IHCPs in arbitrary geometries. Numerical strategies utilizing Laplace and Fourier transforms to solve boundary-value IHCPs are also discussed in [14].

In recent years, meshless methods have gained prominence as powerful tools for addressing IHCPs [1, 13, 15, 16, 17, 18]. Several studies, including [1, 13, 15], have concentrated on one-dimensional cases. Although numerous works have analyzed one-dimensional IHCPs via different strategies [1, 19, 20, 21, 22, 23, 24], we specifically focus here on the one-dimensional, time-dependent parabolic heat equation introduced in [9, 10, 25]:

$$\frac{\partial u(x,t)}{\partial t} = \frac{\partial^2 u(x,t)}{\partial x^2} + F(u(x,t)), \qquad x \in \Omega = (0,1), t \in (0,T], \tag{1}$$

with the initial condition

$$u(x,0) = u_0(x), \qquad x \in [0,1], \tag{2}$$

and Neumann boundary conditions

$$\frac{\partial u(x,t)}{\partial x}|_{x=0} = a(t), \qquad t \in [0,T], \tag{3}$$

$$\frac{\partial u(x,t)}{\partial x}|_{x=1} = g(t), \qquad t \in [0,T], \tag{4}$$

where $u(x,t)$ denotes the temperature distribution and $T$ is the final time of interest. The functions $F$, $u_0(x)$, and $g(t)$ are prescribed. Equations (1)–(3) form the Neumann direct problem, which is nonlinear but well-posed when the boundary flux $a(t)$ is specified. The function $F(u(x,t))$ may represent a heat source or, in chemical and biochemical contexts, a reaction term [9]. The conditions ensuring the existence of solutions to this direct problem are given in [2].

In contrast to the direct problem, in the inverse formulation the boundary heat flux $a(t)$ is unknown, and therefore supplementary information is necessary to guarantee the uniqueness of the solution. In this study, we focus on the case where the additional condition is given in the form of an integral constraint. Specifically, we assume that the $L^1$-norm of the solution over the spatial domain is available through measurement, namely

$$\int_0^1 u(x,t)\,dx = E(t), \qquad t \in [0,T], \tag{5}$$

where $E(t)$ denotes the experimentally observed data. Consequently, the system consisting of Eqs. (1)–(4) together with (5) defines the considered inverse problem in this work.

The partition of unity method (PUM) is a powerful meshfree approximation technique in which the computational domain is covered by overlapping subdomains, each equipped with a local approximation. A set of partition of unity weight functions is then employed to blend these local approximants into a globally smooth representation. This framework provides high flexibility, stability, and accuracy, especially for problems where traditional mesh-based methods or global approximants suffer from ill-conditioning.

In the present work, we employ the PUM to solve the inverse heat conduction problem with the integral-type additional condition (5). The main advantage of the PUM is its ability to maintain both local adaptability and global consistency, which is essential in handling the ill-posedness of inverse problems. Compared with the proposed approach in [25], our PUM-based formulation exhibits superior performance. Specifically, the numerical results demonstrate that the proposed method achieves higher accuracy and better stability in reconstructing the unknown boundary heat flux.

The remainder of this paper is organized as follows. In Section 2, we provide a brief review of the proposed partition of unity method and its main features. Section 3 is devoted to the algorithmic strategy for solving the direct heat conduction problem. In Section 4, the algorithmic procedure for the inverse problem with the integral additional condition is presented. Numerical experiments and computational outcomes are reported in Section 5 to illustrate the accuracy and efficiency of the proposed approach. Finally, concluding remarks is given in Section 6.

## 2 A brief review of the proposed method

### 2.1 Radial Basis Function Collocation Schemes

Let $N$ distinct points $x_1, \ldots, x_N \in \mathbb{R}^d$ be given, together with the corresponding scalar data values $u(x_1), u(x_2), \ldots, u(x_N)$. The objective of the Radial Basis Function (RBF) interpolation problem is to construct an interpolant of the form:

$$s(x) = \sum_{j=1}^{N} a_j \phi(\|x - x_j\|) + \sum_{k=1}^{Q} b_k p_k(x), \tag{6}$$

where $\phi$ denotes a radial basis function, $\{p_k(\mathbf{x})\}_{k=1}^{Q}$ is a basis for the $d$-variate polynomial space of total degree up to $m-1$ (where $m$ is the order of conditional positive definiteness of $\phi$), $\|\cdot\|$ is the Euclidean norm, $a_j \in \mathbb{R}$ and $b_k \in \mathbb{R}$ are the unknown coefficients. These coefficients are determined by enforcing the interpolation conditions $s(x_i) = u(x_i)$ alongside the additional constraints $\sum_{j=1}^{N} a_j p_k(\mathbf{x}_j) = 0$ for $k = 1, \ldots, Q$, to ensure uniqueness. This requirement yields the block linear system:

$$\begin{bmatrix} A & P \\ P^T & 0 \end{bmatrix} \begin{bmatrix} \underline{a} \\ \underline{b} \end{bmatrix} = \begin{bmatrix} \underline{u} \\ \underline{0} \end{bmatrix}, \tag{7}$$

with entries $A_{ij} = \phi(\|x_i - x_j\|)$ and $P_{ik} = p_k(\mathbf{x}_i)$, while $\underline{u} = [u(x_1), \ldots, u(x_N)]^{\mathrm{T}}$, $\underline{a} = [a_1, \ldots, a_N]^{\mathrm{T}}$, and $\underline{b} = [b_1, \ldots, b_Q]^{\mathrm{T}}$. For conditionally positive definite basis functions $\phi$ of order $m$, the system matrix is nonsingular provided that the data points are $\Pi_{m-1}^{d}$-unisolvent, ensuring the existence of a unique solution.

Alternatively, the interpolant can be represented by introducing cardinal basis functions $\psi_j(x)$ which incorporate the polynomial augmentation and satisfy the Kronecker delta property:

$$\psi_j(x_i) = \begin{cases} 1, & \text{if } i = j, \\ 0, & \text{if } i \neq j. \end{cases} \tag{8}$$

This leads to an equivalent expression for the interpolant:

$$s(x) = \sum_{j=1}^{N} \psi_j(x) u(x_j). \tag{9}$$

For a linear differential operator $\mathcal{L}$, we have:

$$\mathcal{L}s(x) = \sum_{j=1}^{N} \mathcal{L}\psi_j(x) u(x_j). \tag{10}$$

The differentiation process is then encoded in the matrix $\Psi_{\mathcal{L}}$, allowing us to compute:

$$\underline{s}_{\mathcal{L}} = \Psi_{\mathcal{L}} \underline{u}. \tag{11}$$

When dealing with time-dependent partial differential equations, the solution $u(x,t)$ is approximated in the form:

$$s(x,t) = \sum_{j=1}^{N} \psi_j(x) u_j(t), \tag{12}$$

where $u_j(t) \approx u(x_j, t)$ are the unknown time-dependent coefficients to be determined.

### 2.2 Radial Basis Function-based PUM

In the RBF-PUM framework, the global approximation $s(x)$ is formulated through a partition of unity representation:

$$s(x) = \sum_{j=1}^{M} w_j(x) s_j(x), \tag{13}$$

where each $s_j(x)$ denotes a local RBF interpolant defined over the subdomain (or patch) $\Omega_j$, and the functions $w_j(x)$ are partition weights constructed using the Shepard approach [26]:

$$w_j(x) = \frac{\varphi_j(x)}{\sum_{k \in I(x)} \varphi_k(x)}. \tag{14}$$

Here, the compactly supported functions $\varphi_j(x)$ are typically chosen as Wendland functions. By construction, the weight functions satisfy the partition of unity condition:

$$\sum_{j \in I(x)} w_j(x) = 1. \tag{15}$$

Therefore, the global interpolant can equivalently be expressed as:

$$s(x) = \sum_{j \in I(x)} w_j(x) s_j(x). \tag{16}$$

For time-dependent problems, the approximate solution $u(x,t)$ is represented by:

$$s(x,t) = \sum_{j \in I(x)} w_j(x) s_j(x,t), \tag{17}$$

where each local term $s_j(x,t)$ is further approximated as:

$$s_j(x,t) = \sum_{k \in J(\Omega_j)} \psi_k(x) u_k(t). \tag{18}$$

Combining these expressions, the global time-dependent approximation becomes:

$$s(x,t) = \sum_{j \in I(x)} \sum_{k \in J(\Omega_j)} w_j(x) \psi_k(x) u_k(t). \tag{19}$$

For spatial derivatives, applying the Leibniz rule yields:

$$\frac{\partial^{|\sigma|}}{\partial x^\sigma} s(x,t) = \sum_{j \in I(x)} \sum_{k \in J(\Omega_j)} \frac{\partial^{|\sigma|}}{\partial x^\sigma} \left( w_j(x) \psi_k(x) \right) u_k(t). \tag{20}$$

The density of nodes and patches is quantified through the local and global fill distances, denoted by $h$ and $H$, respectively:

$$h = \max\{h_j\}, \quad H = \sup_{x \in \Omega} \min_{1 \le j \le M} \|x - X_j\|.$$

For uniform discretizations, $h$ is proportional to the nodal spacing, while $H$ reflects the characteristic size of the patches.

## 3 Algorithmic strategy for solving the direct problem

In this section, we consider the direct initial boundary value problem (1)–(4). To discretize the time derivative, we adopt a finite difference approximation based on the time-stepping method. In particular, we use the following difference approximation

$$\frac{\partial u(x,t)}{\partial t} \simeq \frac{u^{k+1}(x) - u^k(x)}{\delta t}, \tag{21}$$

where $u^k(x) = u(x, k\delta t)$ and $\delta t$ is the time step size. Moreover, the Crank–Nicolson scheme is applied for approximating the Laplacian operator at two consecutive time levels as

$$\frac{\partial^2 u(x,t)}{\partial x^2} \simeq \frac{1}{2}\left(\frac{\partial^2 u^{k+1}(x)}{\partial x^2} + \frac{\partial^2 u^k(x)}{\partial x^2}\right). \tag{22}$$

Combining the above approximations, Eq. (1) can be rewritten in the form

$$\frac{u^{(k+1)}(x) - u^{(k)}(x)}{\delta t} = \frac{1}{2}\left(\frac{\partial^2 u^{(k+1)}(x)}{\partial x^2} + \frac{\partial^2 u^{(k)}(x)}{\partial x^2}\right) + F(\tilde{u}), \tag{23}$$

where $\tilde{u}$ denotes the most recent available approximation of $u$. Equation (23) can equivalently be written as

$$u^{(k+1)}(x) - \lambda\frac{\partial^2 u^{(k+1)}(x)}{\partial x^2} = u^{(k)}(x) + \lambda\frac{\partial^2 u^{(k)}(x)}{\partial x^2} + \delta t F(\tilde{u}), \tag{24}$$

with $\lambda = \delta t/2$.

Now, in accordance with the RBF-PUM approximation introduced in Section 2, the solution $u(x,t)$ is represented as

$$u(x,t) \approx s(x,t) = \sum_{j\in I(x)} \sum_{m\in J(\Omega_j)} w_j(x)\psi_m(x)u_m(t), \tag{25}$$

where $w_j(x)$ are the partition of unity weight functions, $\psi_m(x)$ are the local cardinal basis functions, and $u_m(t)$ are the unknown time-dependent coefficients. Applying the same representation for the spatial derivatives, we obtain

$$\frac{\partial^2 u(x,t)}{\partial x^2} \approx \sum_{j\in I(x)} \sum_{m\in J(\Omega_j)} \frac{\partial^2}{\partial x^2}\big(w_j(x)\psi_m(x)\big)\, u_m(t). \tag{26}$$

By substituting (25) and (26) into Eq. (24), we arrive at the following discrete relation

$$\sum_{m=1}^{N} \Phi_m(x)u_m^{(k+1)} - \lambda\sum_{m=1}^{N} D_{xm}^{(2)}(x)u_m^{(k+1)} = \sum_{m=1}^{N} \Phi_m(x)u_m^{(k)} + \lambda\sum_{m=1}^{N} D_{xm}^{(2)}(x)u_m^{(k)} + \delta t F(\tilde{u}), \tag{27}$$

where $\Phi_m(x) = \sum_j w_j(x)\psi_m(x)$ and

$$D_{xm}^{(2)}(x) = \sum_j \frac{\partial^2}{\partial x^2}\big(w_j(x)\psi_m(x)\big).$$

Evaluating Eq. (27) at the nodal points $x_i$, $i = 1, 2, \ldots, N_\Omega$, we obtain

$$u_i^{(k+1)} - \lambda \sum_{m=1}^{N} D_{x_{im}}^{(2)} u_m^{(k+1)} = u_i^{(k)} + \lambda \sum_{m=1}^{N} D_{x_{im}}^{(2)} u_m^{(k)} + \delta t F(\tilde{u}_i), \quad i = 1, 2, \ldots, N_\Omega. \tag{28}$$

For boundary nodes $x_i \in \partial\Omega = \{0, 1\}$, the boundary conditions (3)–(4) yield

$$\sum_{m=1}^{N} D_{x_{im}}^{(1)} u_m^{(k+1)} = a((k+1)\delta t), \qquad x_i = 0, \tag{29}$$

$$\sum_{m=1}^{N} D_{x_{im}}^{(1)} u_m^{(k+1)} = g((k+1)\delta t), \qquad x_i = 1, \tag{30}$$

where $D_{x_{im}}^{(1)}$ are the discrete derivative coefficients associated with the PUM basis.

Furthermore, the additional output (5) is computed by Simpson's composite integration rule as

$$E((k+1)\delta t) = \sum_{m=1}^{N} d_m u_m^{(k+1)}, \tag{31}$$

where $d_m$ are the Simpson rule coefficients.

In compact matrix form, the discrete system can be expressed as

$$\mathbf{A} U^{(k+1)} = \mathbf{B} U^{(k)} + \mathbf{C}, \tag{32}$$

where $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{N \times N}$, and $\mathbf{C}, U^{(k+1)} \in \mathbb{R}^N$ are defined as follows:

$$\mathbf{A}_{ij} = \begin{cases} \delta_{ij} - \lambda D_{x_{ij}}^{(2)}, & i = 1, 2, ..., N_\Omega, \\ D_{x_{ij}}^{(1)}, & x_i \in \partial\Omega = \{0, 1\}, \end{cases} \qquad j = 1, 2, \ldots, N, \tag{33}$$

$$\mathbf{B}_{ij} = \begin{cases} \delta_{ij} + \lambda D_{x_{ij}}^{(2)}, & i = 1, 2, ..., N_\Omega, \\ 0, & x_i \in \partial\Omega = \{0, 1\}, \end{cases} \qquad j = 1, 2, \ldots, N, \tag{34}$$

$$\mathbf{C}_i = \begin{cases} \delta t F(\tilde{u}_i), & i = 1, 2, ..., N_\Omega, \\ a((k+1)\delta t), & x_i = 0, \\ g((k+1)\delta t), & x_i = 1, \end{cases} \tag{35}$$

$$U^{(k+1)} = [u_1^{(k+1)}, u_2^{(k+1)}, \ldots, u_N^{(k+1)}]^{tr}, \tag{36}$$

and in Eqs. (33) and (34), $\delta_{ij}$ denotes the Kronecker delta function,

$$\delta_{ij} = \begin{cases} 1, & i = j, \\ 0, & i \neq j. \end{cases} \tag{37}$$

Here, $D^{(1)}_{x_{ij}}$ and $D^{(2)}_{x_{ij}}$ are the first- and second-order differentiation matrices constructed from the PUM basis functions according to formulas (19) and (20).

To deal with the nonlinear term in (32), we employ a simple predictor–corrector (P–C) iteration. At the first time step ($k = 0$), using the initial condition (2), we set

$$U^{(0)} = [u_0(x_1), u_0(x_2), \ldots, u_0(x_N)]^{tr}. \tag{38}$$

The algorithm proceeds as follows [**?**]:
**Step 1. Predictor:** set $\tilde{u} = U^{(k)}$ and solve system (32) as a linear algebraic system to obtain a first approximation $U^{(k+1)}$.
**Step 2. Corrector:** compute the average

$$U = \frac{1}{2}\big(U^{(k)} + U^{(k+1)}\big). \tag{39}$$

**Step 3. Iteration:** replace $\tilde{u}$ by $U$ and resolve system (32), producing a new iterate $U_l^{(k+1)}$ at the same time level. Then update

$$U = \frac{1}{2}\big(U^{(k)} + U_l^{(k+1)}\big). \tag{40}$$

This predictor–corrector cycle is repeated until the difference between two successive iterates is sufficiently small, i.e.

$$\|U_l^{(k+1)} - U_{l-1}^{(k+1)}\|_\infty < 10^{-15}. \tag{41}$$

Finally, we set $U^{(k+1)} = U_l^{(k+1)}$ and proceed to the next time level. In this way, the nonlinearity in system (32) is effectively handled at each step, and the numerical solution is advanced until the desired final time $T$.

# 4 Algorithmic strategy for solving the inverse problem

We now consider the inverse initial boundary value problem (1)–(4) with the additional condition (5), when both the surface heat flux $a(t)$ and the temperature distribution $u(x,t)$ are unknown. As in the direct problem, Eqs. (28) and (30) remain unchanged, but Eq. (29) is replaced by the following relation:

$$\sum_{j=1}^{N} D^{(1)}_{x_{ij}} u_j^{(k+1)} - a^{(k+1)} = 0, \qquad x_i = 0, \tag{42}$$

where $D^{(1)}_{x_{ij}}$ denotes the entries of the first-order differentiation matrix constructed from the PUM basis functions.

In practice, the additional observation (5) is obtained from measurements that are contaminated with errors. Therefore, instead of the exact data $E((k+1)\delta t)$, we use the noisy data

$$E_\gamma\big((k+1)\delta t\big) = E\big((k+1)\delta t\big)\Big(1 + \gamma \underline{\epsilon}\Big), \tag{43}$$

where $\gamma$ is the noise level and $\underline{\epsilon}$ are random variables uniformly distributed in $[-1, 1]$.

Thus, at each time step, we obtain $N_\Omega + N_{\partial\Omega} + 1$ equations, i.e., $N+1$ equations for the $N+1$ unknowns $\{u_1^{(k+1)}, u_2^{(k+1)}, \ldots, u_N^{(k+1)}, a^{(k+1)}\}$. This system can be written in matrix form as

$$\mathbf{A}U^{(k+1)} = \mathbf{B}U^{(k)} + \mathbf{C}, \tag{44}$$

where $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{(N+1)\times(N+1)}$ and $\mathbf{C}, U^{(k+1)} \in \mathbb{R}^{N+1}$ are defined as follows:

$$\mathbf{A}_{ij} = \begin{cases} \delta_{ij} - \lambda D^{(2)}_{x_{ij}}, & i = 1, 2, \ldots, N_\Omega, \\ D^{(1)}_{x_{ij}}, & x_i \in \partial\Omega = \{0, 1\}, \qquad j = 1, 2, \ldots, N, \\ d_j, & i = N+1, \end{cases} \tag{45}$$

$$\mathbf{A}_{i,N+1} = \begin{cases} 0, & i = 1, 2, \ldots, N_\Omega, \\ -1, & x_i = 0, \\ 0, & x_i = 1, \\ 0, & i = N+1, \end{cases} \tag{46}$$

$$\mathbf{B}_{ij} = \begin{cases} \delta_{ij} + \lambda D^{(2)}_{x_{ij}}, & i = 1, 2, \ldots, N_\Omega, \\ 0, & x_i \in \partial\Omega = \{0, 1\}, \qquad j = 1, 2, \ldots, N, \\ 0, & i = N+1, \end{cases} \tag{47}$$

$$\mathbf{B}_{i,N+1} = 0, \qquad i = 1, 2, \ldots, N+1, \tag{48}$$

$$\mathbf{C}_i = \begin{cases} \delta t F(\tilde{u}_i), & i = 1, 2, \ldots, N_\Omega, \\ 0, & x_i = 0, \\ g((k+1)\delta t), & x_i = 1, \\ E_\gamma((k+1)\delta t), & i = N+1. \end{cases} \tag{49}$$

Finally,

$$U^{(k+1)} = [u_1^{(k+1)}, u_2^{(k+1)}, \ldots, u_N^{(k+1)}, a^{(k+1)}]^{tr}. \tag{50}$$

At the initial step ($k = 0$), the value $a(0)$ is required. From Eqs. (2) and (3), we obtain

$$a(0) = u_0'(0). \tag{51}$$

## 5 Computational outcomes

This section is devoted to assessing the numerical performance of the proposed methodology through its application to two benchmark problems. The accuracy and stability of the presented scheme are evaluated by executing simulations across a range of values for the nodal spacing $h$, patch fill distance $H$, patch radius $R$, and the time step size $\delta t$. In the subsequent test cases, we use circular patches to ensure complete coverage of the domain and its boundaries. By adjusting the patch radii, it can be guaranteed that overlapping between patches occurs. The radius is set as $R = \frac{\sqrt{2}}{2} c_R H$, where the coefficient $c_R$ is determined based on the value of $h$ to achieve optimal results. Also, we use $M = 25$ circular patches in two test problems. Moreover, the final time in two test problems is $T = 1$.

In our implementation, the polyharmonic spline (PHS) is selected as the radial basis function in Eq. (6). This RBF is mathematically expressed by:

$$\phi(r) = r^{2\lambda}\ln(r), \qquad r = \|x - x_i\|, \qquad \lambda \in \mathbb{Z}^+, \tag{52}$$

and is employed within a support domain centered at the point $x_i$. For solving the second-order PDE Eq. (1), the parameter $\lambda = 2$ is chosen for the polyharmonic splines. A polynomial basis of order 5 is appended to the RBF approximation in Eq. (6), which is given by the set:

$$\{x^i \mid i = 0, 1, \ldots, 5\}. \tag{53}$$

## 5.1 Example 1

We examine Eqs. (1)–(4) with the exact solution

$$u(x,t) = xe^{2t}, \tag{54}$$

from which the remaining functions can be derived. In this case, the source term is given by $F(u) = 2u$. Furthermore, the target output (5) takes the form $E(t) = 0.5e^{2t}$ for $t \in [0,1]$. The absolute errors between the numerical approximation and the exact solution for both $u(x,t)$ and $E(t)$, computed with $h = 1/20$, $\delta t = 0.01$ $(a,b)$ and $h = 1/40$, $\delta t = 0.001$ $(c,d)$, are depicted in Fig. 1.

### 5.1.1 Exact data

In this subsection, we deal with the case of exact data, meaning that no noise is present in the input data (5). Hence, the exact solutions are $\{u(x,t), a(t)\} = \{xe^{2t}, e^{2t}\}$. Equivalently, this corresponds to setting $\gamma = 0$ in Eq. (43). The absolute errors for $a(t)$ and $u(x,T)$ obtained from solving the inverse problem are displayed in Fig. 2 with $h = 1/40, \delta t = 0.001$.

### 5.1.2 Noisy Data

To examine the stability of the numerical solution, we add $\gamma = 1\%$ and $\gamma = 5\%$ noise to the input data (5), as specified in Eq. (43). The numerical approximation of $a(t)$ obtained with $h = 1/26$, $\delta t = 1/30$, and without applying regularization turns out to be highly oscillatory and unstable, as illustrated in Fig. 3.

A variety of approaches for solving discrete ill-posed problems have been discussed in the literature, many of which rely on regularization techniques. The primary purpose of regularization is to stabilize the solution process and obtain a reliable approximation [27]. Among these, the most common is Tikhonov regularization. To apply the Tikhonov method at time level "$k+1$", the system (44) is reformulated as

$$\mathbf{A}\underline{\mathbf{a}} = \underline{\mathbf{b}}^{\epsilon}, \tag{55}$$

where $\underline{\mathbf{a}} = [u_1^{(k+1)}, u_2^{(k+1)}, ..., u_N^{(k+1)}, a^{(k+1)}]^{tr}$ and $\underline{\mathbf{b}}^{\epsilon} = \mathbf{B}U^{(k)} + \mathbf{C}$, in which the right-hand side of Eq. (55) contains the noisy data (43). The Tikhonov scheme then yields the regularized solution in the form

$$\underline{\mathbf{a}}_{\lambda} = \left(\mathbf{A}^{tr}\mathbf{A} + \lambda D_j^{tr} D_j\right)^{-1}\mathbf{A}^{tr}\underline{\mathbf{b}}^{\epsilon}, \tag{56}$$

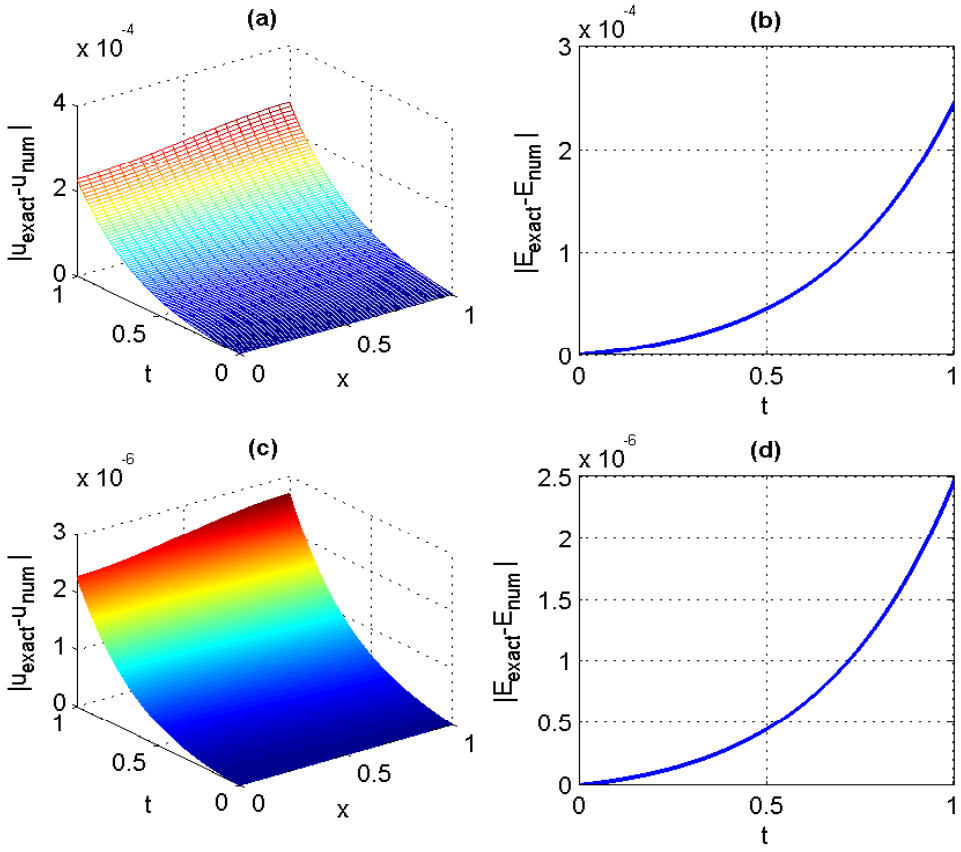


Figure 1: The absolute errors between the exact and numerical solutions $u(x,t)$ and $E(t)$ obtained by solving the direct problem with $h = 1/20$, $\delta t = 0.01$ (a, b), and $h = 1/40$, $\delta t = 0.001$ (c,d) for Example 1.

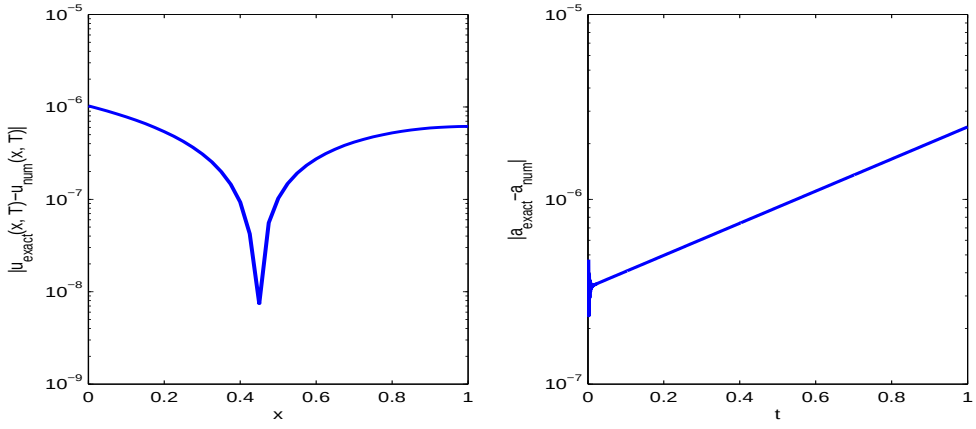


Figure 2: The absolute errors corresponding to $u(x,T)$ and $a(t)$ obtained by solving the inverse problem with $h = 1/40$, $\delta t = 0.001$, $\gamma = 0$, and no regularization for Example 1.

Table 1: The corresponding results to Figs. 4 and 5, i.e, the values of regularization parameters in the last iteration and the maximum norms for the zeroth-order Tikhonov regularization for different percentages of noise with $h = 1/40$ and $\delta t = 0.05$, for Example 1.

| | $num_\lambda = 70$ | | | $num_\lambda = 140$ | | |
|---|---|---|---|---|---|---|
| $\gamma(\%)$ | $\|a(t)\|_\infty$ | $\|u(x,T)\|_\infty$ | $\lambda_{opt}$ | $\|a(t)\|_\infty$ | $\|u(x,T)\|_\infty$ | $\lambda_{opt}$ |
| 1 | $6.3087e-01$ | $4.4439e-01$ | $1.2662e-03$ | $1.3971e-01$ | $3.4626e-02$ | $4.0961e-05$ |
| 3 | $7.2231e-01$ | $4.6272e-01$ | $1.2662e-03$ | $4.8981e-01$ | $9.5391e-02$ | $3.3344e-04$ |
| 5 | $6.9741e-01$ | $5.2376e-01$ | $1.2662e-03$ | $7.3192e-01$ | $1.3707e-01$ | $5.3700e-04$ |
| 10 | $2.8539e+00$ | $8.8401e-01$ | $1.2662e-03$ | $1.1023e+00$ | $2.6323e-01$ | $9.6585e-05$ |

where $D_j$ represents the regularization derivative operator of order $j \in \{0, 1, 2\}$ and $\lambda \geq 0$ is the regularization parameter. In the present work, we set $j = 0$, i.e., the zeroth-order Tikhonov regularization with $D_0 = I$. The choice of $\lambda$ can be guided by either the L-curve method [28] or the discrepancy principle [29]. The L-curve method selects $\lambda$ at the corner of the L-curve, which plots the residual norm $\|\mathbf{A}\underline{\mathbf{a}}_\lambda - \underline{\mathbf{b}}^\epsilon\|$ against the solution norm $\|\underline{\mathbf{a}}_\lambda\|$. In contrast, the discrepancy principle chooses $\lambda > 0$ such that the residual $\|\mathbf{A}\underline{\mathbf{a}}_\lambda - \underline{\mathbf{b}}^\epsilon\| \approx \epsilon$. Since in this study the exact solution is known, we adopt a simple strategy to determine the regularization parameter. Using the reference solution $\underline{\mathbf{a}}_{ex}$, we compute the error norm $\|\underline{\mathbf{a}}_{ex} - \underline{\mathbf{a}}_\lambda\|$ as a function of $\lambda$. The value $\lambda_{opt}$ corresponding to the minimum of this norm is taken as the optimal regularization parameter. This procedure is applied at each time level to compute the regularized solution. Moreover, we denote by $num_\lambda$ the number of candidate regularization parameters examined at each time step.

Table 1 reports the maximum errors for $a(t)$ and $u(x, T)$ obtained with $h = 1/40$, $\delta t = 0.05$, and noise levels $\gamma \in \{1, 3, 5, 10\}\%$, using two different choices of the number of regularization parameters at each time level, namely $num_\lambda = \{70, 140\}$. In this table, the optimal regularization parameters $\lambda_{opt}$ are those selected at the final iteration of each time level. Figs. 4 and 5 correspond to Table 1, providing a comparison between the exact $a(t)$ and its regularized approximations under various noise levels.

### 5.2 Example 2

As a further example, we begin with the direct problem (1)–(4) whose exact solution is

$$u(x, t) = \cos(x + t). \tag{57}$$

The necessary functions can be derived directly from Eq. (57). In this case, the nonlinear source term is $F(u) = u - \sqrt{1 - u^2}$. In addition, the desired output (5) is specified as $E(t) = \sin(1 + t) - \sin(t)$ for $t \in [0, 1]$. The absolute errors between the numerical and exact solutions for both $u(x, t)$ and

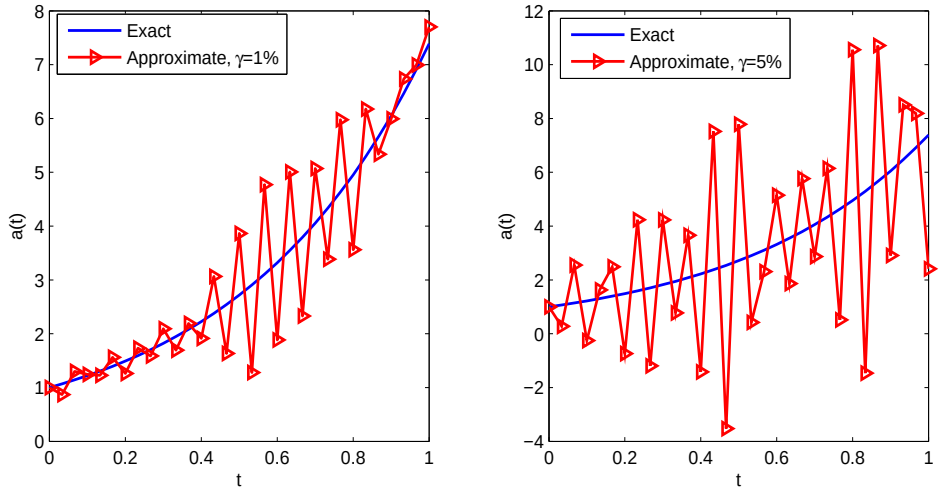


Figure 3: The exact solution for $a(t)$ in comparison with the numerical solution for $h = 1/26$, $\delta t = 1/30$, $\gamma = 1\%$ and $\gamma = 5\%$ noise, and no regularization, for the inverse problem in Example 1.

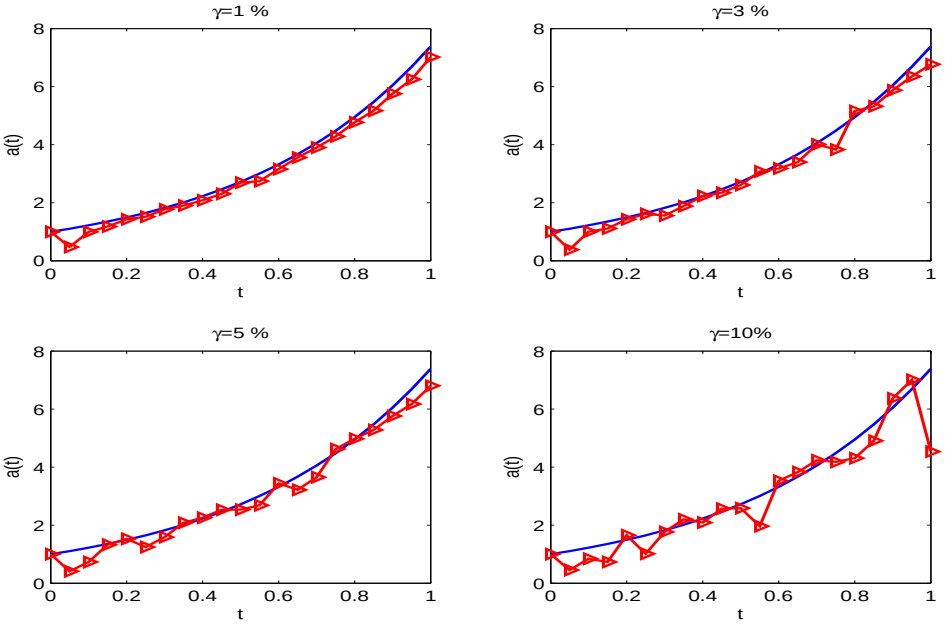


Figure 4: The exact $(-)$ and numerical $(-\triangle-)$ results of $a(t)$ obtained using the zeroth-order Tikhonov regularization for $\gamma \in \{1, 3, 5, 10\}\%$, by solving the inverse problem with $h = 1/40$, $\delta t = 0.05$ utilizing $num_\lambda = 70$ regularization parameters in every time level for Example 1.

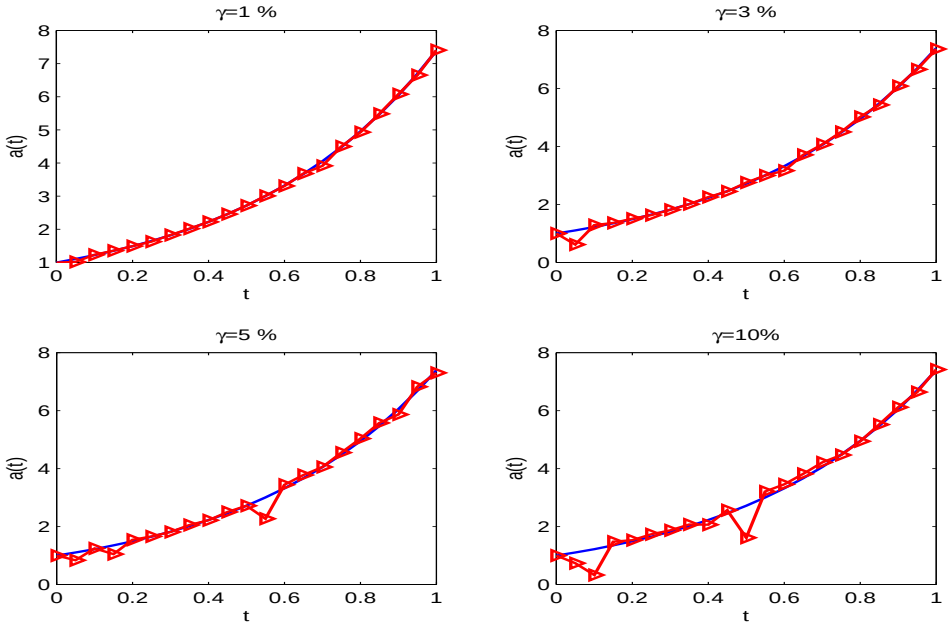


Figure 5: The exact $(-)$ and numerical $(-\triangle-)$ results of $a(t)$ obtained using the zeroth-order Tikhonov regularization for $\gamma \in \{1, 3, 5, 10\}\%$, by solving the inverse problem with $h = 1/40$, $\delta t = 0.05$ utilizing $num_\lambda = 140$ regularization parameters in every time level for Example 1.

$E(t)$, computed with $h = 1/20$, $\delta t = 0.01$ (a, b) and $h = 1/40$, $\delta t = 0.001$ (c, d), are illustrated in Fig. 6.

### 5.2.1 Exact data

The inverse problem is specified with the exact solutions

$$\{u(x,t), a(t)\} = \{\cos(x+t), -\sin(t)\}. \tag{58}$$

We first examine the case of exact data, that is, $\gamma = 0$ in Eq. (43). The absolute errors for $a(t)$ and $u(x,T)$ obtained by solving the inverse problem with $h = 1/40, \delta t = 0.001$ are shown in Fig. 7.

### 5.2.2 Noisy Data

To study the stability of the numerical solution, we add several different noise levels to the input data (5), as described in Eq. (43). Table 2 presents the maximum norms of $a(t)$ and $u(x,T)$ for $h = 1/40$, $\delta t = 0.05$, and noise levels $\gamma \in \{1, 3, 5, 10\}\%$, along with the optimal regularization parameters at the final time step using $num_\lambda = \{40, 140\}$. Based on Table 2, Figs. 8 and 9 display a comparison between the exact $a(t)$ and its regularized solution obtained from solving the inverse problem.

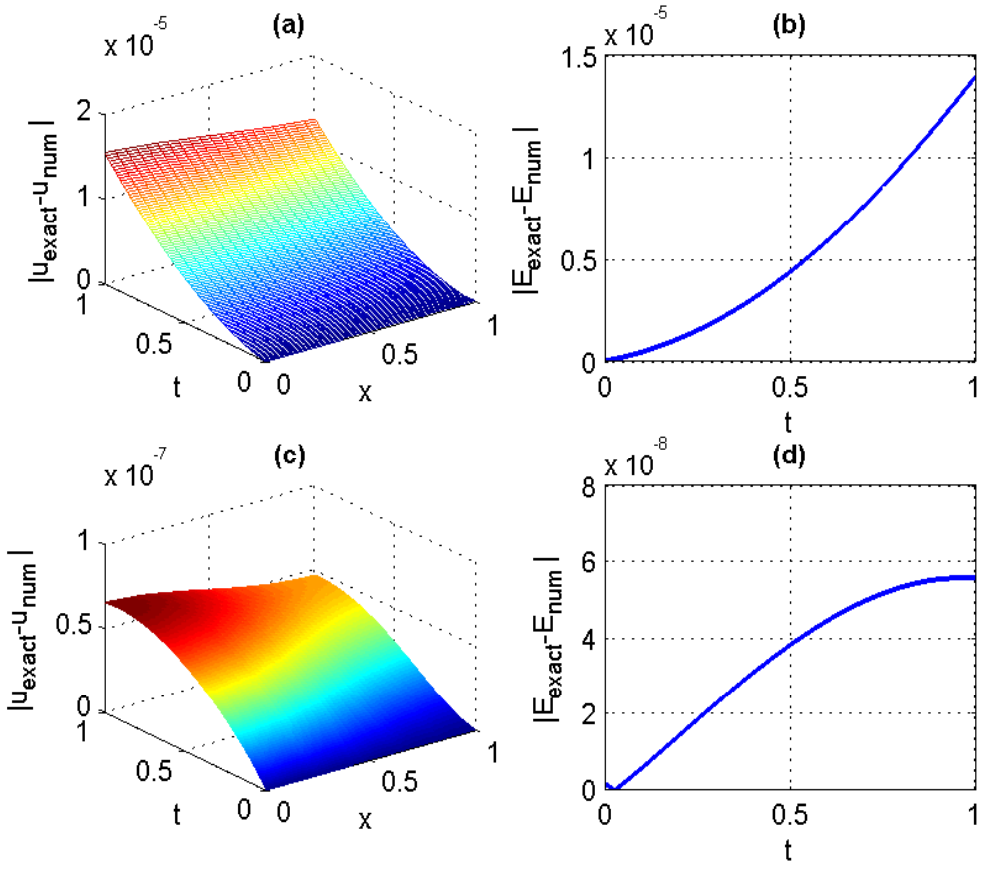


Figure 6: The absolute errors between the exact and numerical solutions $u(x,t)$ and $E(t)$ obtained by solving the direct problem with $h = 1/20$, $\delta t = 0.01$ (a, b), and $h = 1/40$, $\delta t = 0.001$ (c,d) for Example 2.

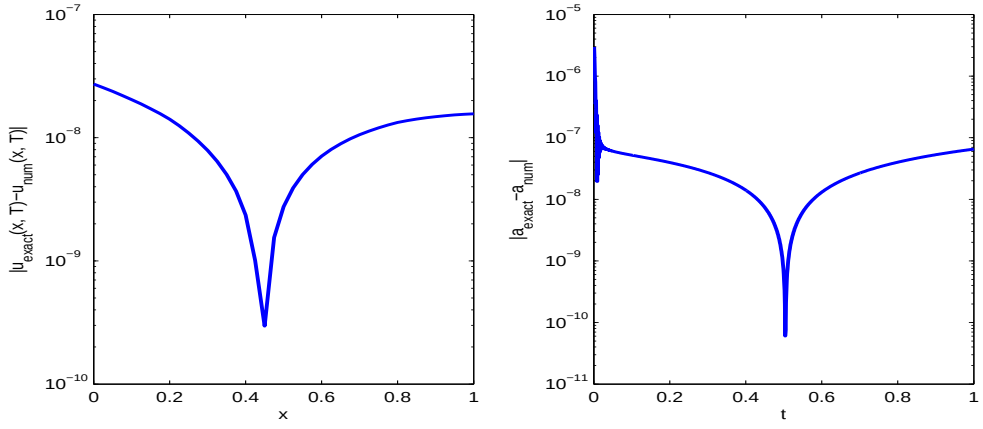


Figure 7: The absolute errors corresponding to $u(x,T)$ and $a(t)$ obtained by solving the inverse problem with $h = 1/40$, $\delta t = 0.001$, $\gamma = 0$, and no regularization for Example 2.

Table 2: The corresponding results to Figs. 8 and 9, i.e, the values of regularization parameters in the last iteration and the maximum norms for the zeroth-order Tikhonov regularization for different percentages of noise with $h = 1/40$ and $\delta t = 0.05$, for Example 1.

| | $num_\lambda = 40$ | | | $num_\lambda = 140$ | | |
|---|---|---|---|---|---|---|
| $\gamma(\%)$ | $\|a(t)\|_\infty$ | $\|u(x,T)\|_\infty$ | $\lambda_{opt}$ | $\|a(t)\|_\infty$ | $\|u(x,T)\|_\infty$ | $\lambda_{opt}$ |
| 1 | $2.6997e-01$ | $5.4867e-02$ | $3.2349e-02$ | $1.4507e-02$ | $2.8435e-04$ | $2.2432e-03$ |
| 3 | $3.3033e-01$ | $7.6958e-02$ | $3.9143e-02$ | $7.8897e-02$ | $9.6710e-03$ | $4.8085e-03$ |
| 5 | $1.9719e-01$ | $3.0647e-02$ | $2.4304e-02$ | $2.7097e-01$ | $5.1554e-04$ | $4.8085e-03$ |
| 10 | $4.1370e-01$ | $6.7983e-02$ | $3.5584e-02$ | $3.8509e-01$ | $6.2179e-03$ | $7.0401e-03$ |

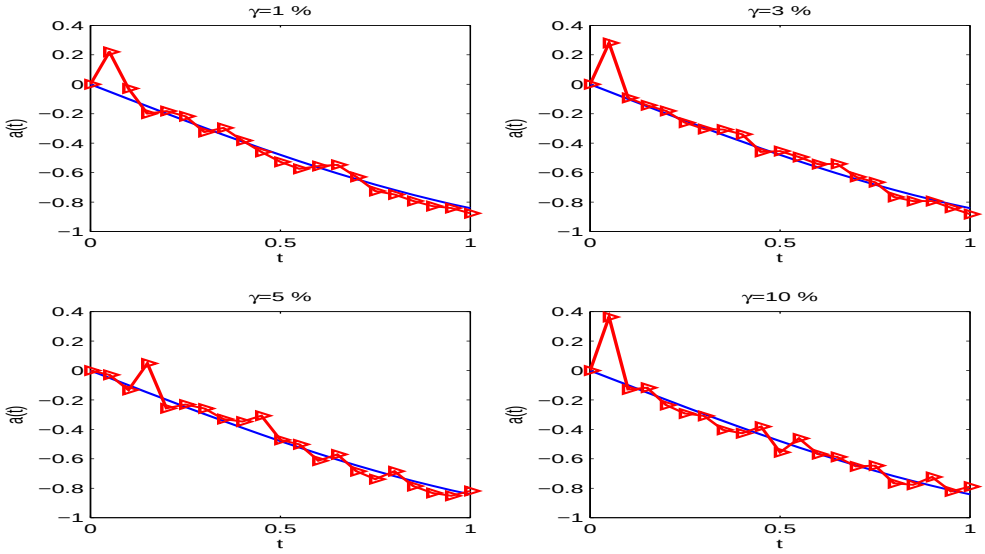


Figure 8: The exact $(-)$ and numerical $(-\triangle-)$ results of $a(t)$ obtained using the zeroth-order Tikhonov regularization for $\gamma \in \{1, 3, 5, 10\}\%$, by solving the inverse problem with $h = 1/40$, $\delta t = 0.05$ utilizing $num_\lambda = 40$ regularization parameters in every time level for Example 2.

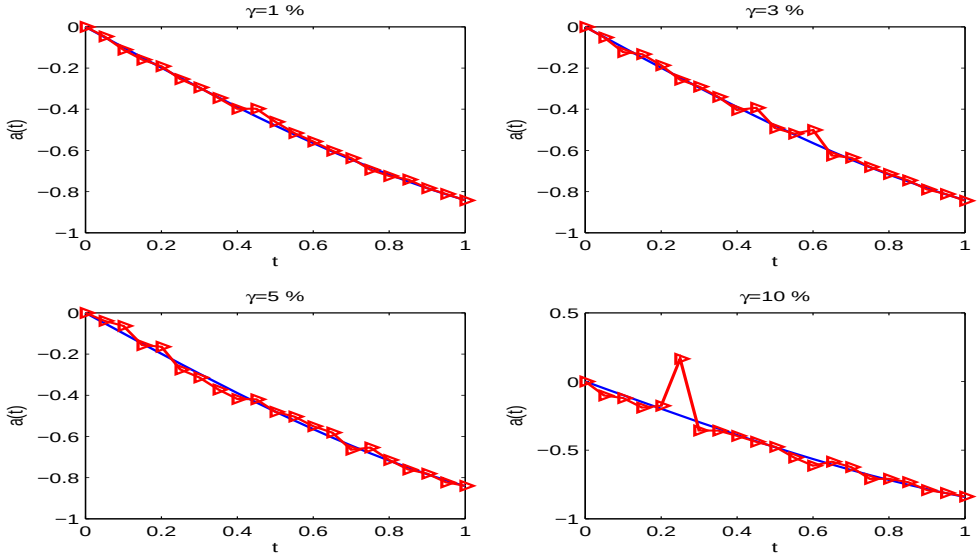


Figure 9: The exact $(-)$ and numerical $(-\triangle-)$ results of $a(t)$ obtained using the zeroth-order Tikhonov regularization for $\gamma \in \{1, 3, 5, 10\}\%$, by solving the inverse problem with $h = 1/40$, $\delta t = 0.05$ utilizing $num_\lambda = 140$ regularization parameters in every time level for Example 2.

## 6 Conclusion

This paper has successfully developed an efficient meshless approach based on the radial basis function partition of unity method (RBF-PUM) for solving the inverse heat conduction problem with a nonlinear source term. The proposed method effectively combines the flexibility of meshless techniques with the stability of finite difference time discretization. Numerical results demonstrate that the approach accurately reconstructs both the temperature distribution and the unknown boundary heat flux from integral over-specification data. The method shows considerable accuracy with exact data and maintains acceptable stability when handling noisy measurements through Tikhonov regularization. The local nature of the PUM approximation leads to sparse systems, enhancing computational efficiency. This work establishes RBF-PUM as a promising framework for solving ill-posed inverse problems in heat transfer and potentially other related fields.

## References


[1] YC Hon and T Wei. A fundamental solution method for inverse heat conduction problem. *Engineering analysis with boundary elements*, 28(5):489–495, 2004. doi:10.1016/j.enganabound.2004.01.002.

[2] John Rozier Cannon. *The one-dimensional heat equation*. Number 23. Cambridge University Press, 1984. doi:10.1017/CBO9781139086967.

[3] John Rozier Cannon and Paul DuChateau. Structural identification of an unknown source term in a heat equation. *Inverse problems*, 14(3):535, 1998. doi:10.1088/0266-5611/14/3/011.

[4] Fatma Kanca and Mansur I Ismailov. The inverse problem of finding the time-dependent diffusion coefficient of the heat equation from integral overdetermination data. *Inverse Problems in Science and Engineering*, 20(4):463–476, 2012. doi:10.1080/17415977.2011.649253.

[5] SA Yousefi, Daniel Lesnic, and Zahra Barikbin. Satisfier function in ritz–galerkin method for the identification of a time-dependent diffusivity. *Journal of Inverse and Ill-Posed Problems*, 20(5-6):701–722, 2012. doi:10.1515/jip-2011-0016.

[6] Dinh Nho Hao and H-J Reinhardt. Recent contributions to linear inverse heat conduction problems. 1996.

[7] Elyas Shivanian and Ahmad Jafarabadi. An inverse problem of identifying the control function in two and three-dimensional parabolic equations through the spectral meshless radial point interpolation. *Applied Mathematics and Computation*, 325:82–101, 2018. doi:10.1016/j.amc.2017.12.041.

[8] Surbhi Arora and Jaydev Dabas. Inverse heat conduction problem in two-dimensional anisotropic medium. *International Journal of Applied and Computational Mathematics*, 5(6):161, 2019. doi:10.1007/s40819-019-0749-1.

[9] A Shidfar, GR Karamali, and J Damirchi. An inverse heat conduction problem with a nonlinear source term. *Nonlinear Analysis: Theory, Methods & Applications*, 65(3):615–621, 2006. doi:10.1016/j.na.2005.09.032.

[10] Mehdi Dehghan, Sohrab Ali Yousefi, and Kamal Rashedi. Ritz–Galerkin method for solving an inverse heat conduction problem with a nonlinear source term via Bernstein multi-scaling functions and cubic B-spline functions. *Inverse Problems in Science and Engineering*, 21(3):500–523, 2013. doi:10.1080/17415977.2012.712527.

[11] Ping Xiong, Jian Deng, Tao Lu, Qi Lu, Yu Liu, and Yong Zhang. A sequential conjugate gradient method to estimate heat flux for nonlinear inverse heat conduction problem. *Annals of Nuclear Energy*, 149:107798, 2020. doi:10.1016/j.anucene.2020.107798.

[12] Ana Paula Fernandes, Marcelo Braga dos Santos, and Gilmar Guimarães. An analytical transfer function method to solve inverse heat conduction problems. *Applied Mathematical Modelling*, 39(22):6897–6914, 2015. doi:10.1016/j.apm.2015.02.032.

[13] Cheng-Yu Ku, Chih-Yu Liu, Jing-En Xiao, Shih-Meng Hsu, and Weichung Yeih. A collocation method with space–time radial polynomials for inverse heat conduction problems. *Engineering Analysis with Boundary Elements*, 122:117–131, 2021. doi:10.1016/j.enganabound.2020.10.011.

[14] Natalia Yaparova. Numerical methods for solving a boundary-value inverse heat conduction problem. *Inverse Problems in Science and Engineering*, 22(5):832–847, 2014. doi:10.1080/17415977.2013.856923.

[15] Hussen Gedefaw et al. Meshless and homotopy perturbation methods for one dimensional inverse heat conduction problem with neumann and robin boundary conditions. *Journal of applied mathematics & informatics*, 40(3_4):675–694, 2022. doi:10.14317/jami.2022.675.

[16] Samreen Ismail et al. Meshless collocation procedures for time-dependent inverse heat problems. *International Journal of Heat and Mass Transfer*, 113:1152–1167, 2017. doi:10.1016/j.ijheatmasstransfer.2017.06.029.

[17] Cheng Deng, Hui Zheng, Mingfu Fu, Jingang Xiong, and CS Chen. An efficient method of approximate particular solutions using polynomial basis functions. *Engineering Analysis with Boundary Elements*, 111:1–8, 2020. doi:10.1016/j.enganabound.2019.10.009.

[18] Jakub Krzysztof Grabski. Numerical solution of non-newtonian fluid flow and heat transfer problems in ducts with sharp corners by the modified method of fundamental solutions and radial basis function collocation. *Engineering analysis with boundary elements*, 109:143–152, 2019. doi:10.1016/j.enganabound.2019.09.019.

[19] JI Frankel and M Keyhani. A global time treatment for inverse heat conduction problems. 1997.

[20] P Jonas and AK Louis. Approximate inverse for a one-dimensional inverse heat conduction problem. *Inverse problems*, 16(1):175, 2000. doi:10.1088/0266-5611/16/1/314.

[21] D Lesnic and L Elliott. The decomposition approach to inverse heat conduction. *Journal of Mathematical Analysis and Applications*, 232(1):82–98, 1999. doi:10.1006/jmaa.1998.6253.

[22] D Lesnic, L Elliott, and DB Ingham. Application of the boundary element method to inverse heat conduction problems. *International Journal of Heat and Mass Transfer*, 39(7):1503–1517, 1996. doi:10.1016/0017-9310(95)00251-0.

[23] Jun Liu. A stability analysis on beck's procedure for inverse heat conduction problems. *Journal of Computational Physics*, 123(1):65–73, 1996. doi:10.1006/jcph.1996.0005.

[24] Shih-Yu Shen. A numerical study of inverse heat conduction problems. *Computers & Mathematics with applications*, 38(7-8):173–188, 1999. doi:10.1016/S0898-1221(99)00206-3.

[25] Abdollah Dinmohammadi and Ahmad Jafarabadi. Inverse heat conduction problem with a nonlinear source term by a local strong form of meshless technique based on radial point interpolation method. *Computational and Applied Mathematics*, 42(6):284, 2023. doi:10.1007/s40314-023-02416-5.

[26] Donald Shepard. A two-dimensional interpolation function for irregularly-spaced data. In *Proceedings of the 1968 23rd ACM national conference*, pages 517–524, 1968. doi:10.1145/800186.810616.

[27] Dorota Krawczyk-StańDo and Marek Rudnicki. Regularization parameter selection in discrete ill-posed problems—the use of the U-curve. *International Journal of Applied Mathematics and Computer Science*, 17(2):157–164, 2007. doi:10.2478/v10006-007-0014-3.

[28] Per Christian Hansen. The L-curve and its use in the numerical treatment of inverse problems. 1999.

[29] Vladimir Alekseevich Morozov. On the solution of functional equations by the method of regularization. In *Doklady Akademii Nauk*, volume 167, pages 510–512. Russian Academy of Sciences, 1966.