

Reconstruction of Two Time-Dependent Coefficients in 1D Parabolic Equations Using Tikhonov Regularization method

Authors Names	ABSTRACT
Ghiath Badrah ^a M. S. Hussein ^b Publication data: 2 / 10 /2026 Keywords Non-local condition, inverse problem, heat equation, nonlinear minimization, coefficient determination.	In this paper we solved a parabolic partial differential equation with nonlocal time condition, Neumann boundary condition and nonlocal integral condition. To solving direct problem, we applied Crank-Nicolson finite difference which insure stability and convergent. In order to solve the inverse problem and recovering two coefficients time-depend and the solution by using the two-point measurements, which it is ill-posed nonlinear problem, applied Tikhonov regularization by using Nonlinear Least Squares method in MATLAB's Optimization Toolbox "lsqnonlin" with Trust-Region-Reflective algorithm, the regularization parameter selected from the range $\{10^{-1}$ to $10^{-10}\}$ by using trial and error strategy. To insure the evaluated of numerical solution using root mean square error {RMSE} proving that our methodology recovers the composite coefficients with acceptable accuracy.

1. Introduction

Partial Differential Equations (PDEs) represent relationship between independent variables that describing physical and dynamic fields involve both time and space. As Walter Strauss elucidates in his book [1], these equations represent the mathematical model for natural phenomena ranging from heat transfer to wave propagation. Studying PDEs help as to understanding these phenomena and finding solutions means controlling it. In these cases, parabolic equations appear in large set of problems heat conduction in materials with varying properties, diffusion processes, chemical reaction processes governed by external constraints and others [3][5].

Some model includes more difficult form, like those problems including time coefficients and governed by non-local initial and boundary conditions, with nonlocal integral conditions [2,3,4]. These problems represent mathematical challenges regarding the interaction between these conditions and the behavior of the solution across the temporal and spatial domains.

Since finding analytical solutions for direct problem with these conditions is difficult, there is important to use numerical methods. Since the problem is a partial differential equation within a rectangular space and time domain, the finite difference Crank-Nicolson method is very suitable, due to it is a pivotal numerical scheme detailed widely in literature [6,7]. This method is selected because its unconditional stability and its second-order accuracy in both temporal and spatial discretizations. By treating the PDE as an average of explicit and implicit Euler schemes, the Crank-Nicolson framework transforms the continuous model into a linear system [6].

In the other hand solving inverse problems in order to recovering unknown coefficients and making the problem nonlinear which is more sophisticated ,such as the problems that Miglena N. Koleva and Lubin G. Vulkov[10] and Liu Yang, Yun Liu, and Zui-Cha Deng [11] where one time-dependent

^a Department of Mathematics, College of Science, University of Baghdad, Baghdad, Iraq., E-Mail: ghiath.exander2403@sc.uobaghdad.edu.iq(Ghiath Badrah),

^b Department of Mathematics, College of Science, University of Baghdad, Baghdad, Iraq., E-Mail: mmmsh@sc.uobaghdad.edu.iq

coefficient was recovered, M. A. J. Al-Shatrah and M. S. Hussein [8], El Mustapha Ait Ben Hassi, Salah-Eddine Chorfi, and Lahcen Maniar [12] where two time-dependent coefficients were recovered, while Mousa J. Huntul [9] where three time-dependent coefficients were recovered. The referenced literature show that these problems are very importance in applications where it depends on complementary measurements data known as overdetermination conditions [4,13,14].

Due to the nature of these inverse problems are ill-posed in the Hadamard sense [15], where minute perturbations in measurement data can lead to significant instabilities in the solution. Therefore constricting the numerical solution matrix was the first step of inverse solution then to ensure the quality of the solution added Gaussian noise to similitude the real data, with Tikhonov Regularization parameter " β " then in order the recovering the unknown coefficients we need to applied strong optimization routine as nonlinear least squares method with Trust-Region-Reflective (TRR) algorithm which MATLAB offers by function "lsqnonlin" in its toolbox[16][17][18].

Recently, to address optimization more efficiently, variational frameworks utilizing the adjoint-state method and conjugate gradient algorithms have been widely adopted to compute gradients with minimal computational cost [19,20]. Instead of meshless techniques, radial basis functions (RBFs) have become widely popular to avoid the geometric complexities of grid generation in complex domains [21,22]. The computational paradigm has expanded toward deep learning, where Physics-Informed Neural Networks (PINNs) including the parabolic governing equations directly into neural loss functions, offering a robust, framework without mesh for parameter estimation [23,24].

This paper is organized as follows section 2, present the mathematical module with the boundary conditions, section 3 the numerical method "finite difference C-N" that used to solve the direct problem, section 4 the inverse method to recover the missed data and formulation the objective function, section 5 show the data and results for inverse solved problem with comparison and evaluation of results.

2. Mathematical Setting of the inverse problem

Consider the problem of finding two time-varying coefficients in a parabolic equation of the form

$$a_1(t)u_t(x, t) = u_{xx}(x, t) - a(t)u(x, t) + b(t)g(x, t) + f(x, t), \quad (x, t) \in \mathcal{D}_T, \quad (1)$$

with the nonlocal time condition of the form

$$u(x, 0) + \delta u(x, T) = \varphi(x), \quad x \in [0, 1], \quad (2)$$

the Neumann type boundary condition (left side);

$$\frac{\partial u}{\partial x}(0, t) = 0, \quad t \in [0, T], \quad (3)$$

non-local integral condition of the first kind (right side),

$$\int_0^1 (x - 1)u(x, t) dx = 0, \quad 0 \leq t \leq T \quad (4)$$

and the two-points measurements are

$$u(x_\ell, t) = h_\ell(t), \quad \ell = 1, 2, \quad 0 < x_1, x_2 < 1, \quad x_1 \neq x_2, \quad t \in [0, T], \quad (5)$$

where, $\mathcal{D}_T$ is a rectangular solution domain in the xt -plane and T is a positive number that represents the final time. Also, $\delta \geq 0$ is a given number, and the functions $f, g, \varphi, h_\ell, \ell = 1, 2$ are known. Whilst, $a(t)$, $b(t)$ and $u(x, t)$ are sought functions. this mathematical module has proofed that the solution is exist and unique by Elvin I. Azizbayov [14].

3. Numerical approach for solving direct (Forward) Problem

In order to solve the direct problem (1)-(4) where $a(t)$ and $b(t)$ are known functions, we will use C-N method with $M_x \times N_t$ mesh with time step size $\Delta t = \frac{T}{N_t}$ and space step size $\Delta x = \frac{1}{M_x}$ the grid point $(x_i, t_j) = (i\Delta x, j\Delta t)$ where $i, j \in \mathbf{N}$ and $i = \overline{0, M_x}, j = \overline{0, N_t}$.

The equation:

$$a_1(t)u_t(x, t) = u_{xx}(x, t) - a(t)u(x, t) + b(t)g(x, t) + f(x, t), \quad (x, t) \in \mathcal{D}_T$$

Then we solve it by using the finite difference method and apply the C-N method

$$\frac{u_{i,j+1} - u_{i,j}}{\Delta t} = \frac{1}{2}(G_{i,j} + G_{i,j+1}) \quad (6)$$

where,

$$G_{i,j} = \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{a_1(t_j)(\Delta x)^2} - \frac{a(t_j)}{a_1(t_j)}u_{i,j} + \frac{1}{a_1(t_j)}(b(t_j)g_{i,j} + f_{i,j})$$

$$G_{i,j+1} = \frac{u_{i+1,j+1} - 2u_{i,j+1} + u_{i-1,j+1}}{a_1(t_{j+1})(\Delta x)^2} - \frac{a(t_{j+1})}{a_1(t_{j+1})}u_{i,j+1} + \frac{1}{a_1(t_{j+1})}(b(t_{j+1})g_{i,j+1} + f_{i,j+1})$$

then we have,

$$\begin{aligned} \frac{u_{i,j+1} - u_{i,j}}{\Delta t} = & \frac{1}{2} \left(\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{a_1(t_j)(\Delta x)^2} - \frac{a(t_j)}{a_1(t_j)}u_{i,j} + \frac{1}{a_1(t_j)}(b(t_j)g_{i,j} + f_{i,j}) \right. \\ & + \frac{u_{i+1,j+1} - 2u_{i,j+1} + u_{i-1,j+1}}{a_1(t_{j+1})(\Delta x)^2} - \frac{a(t_{j+1})}{a_1(t_{j+1})}u_{i,j+1} \\ & \left. + \frac{1}{a_1(t_{j+1})}(b(t_{j+1})g_{i,j+1} + f_{i,j+1}) \right) \quad (7) \end{aligned}$$

Multiplying (7) by (Δt) and rearrange the terms we have,

$$\begin{aligned} & -A_{j+1}u_{i+1,j+1} + \left(1 + 2A_{j+1} + \frac{\Delta t}{2a_1(t_{j+1})}a(t_{j+1}) \right)u_{i,j+1} - A_{j+1}u_{i-1,j+1} \\ & = A_j u_{i+1,j} + \left(1 - (2A_j + \frac{\Delta t}{2a_1(t_j)}a(t_j)) \right)u_{i,j} + A_j u_{i-1,j} \\ & + \frac{\Delta t}{2a_1(t_j)}(b(t_j)g_{i,j} + f_{i,j}) + \frac{\Delta t}{2a_1(t_{j+1})}(b(t_{j+1})g_{i,j+1} + f_{i,j+1}) \end{aligned}$$

Let $A_j = \frac{\Delta t}{2a_1(t_j)\Delta x^2}$, $B_j = \left(2A_j + \frac{\Delta t}{2a_1(t_j)}a(t_j) \right)$ and $kk_j = \frac{\Delta t}{2a_1(t_j)}$ then we get,

$$\begin{aligned}
& -A_{j+1}u_{i+1,j+1} + (1 + B_{j+1})u_{i,j+1} - A_{j+1}u_{i-1,j+1} \\
& = A_j u_{i+1,j} + (1 - B_j)u_{i,j} + A_j u_{i-1,j} + k k_j (b(t_j)g_{i,j} + f_{i,j}) + k k_{j+1} (b(t_{j+1})g_{i,j+1} + f_{i,j+1}) \quad (8)
\end{aligned}$$

Now, let us build the matrix form, we can represent the equation (8) in matrix form:

$$\Sigma U^{(j+1)} = \Gamma U^{(j)} + k k_j V_1 + k k_{j+1} V_2, \quad j = \overline{0, N_t} \quad (9)$$

for $i = \overline{0, M_x}$, $j = \overline{0, N_t}$. The problem is subject to the following conditions with $\delta = 0$, the initial condition approximated as

$$u_{i,0} = \varphi(x_i), \quad i = \overline{0, M_x} \quad (10)$$

from the boundary nonlocal type condition:

$$u_{1,j} = u_{-1,j}, \quad j = \overline{0, N_t} \quad (11)$$

If we want to discretize that non-local integral condition (4) directly using a numerical quadrature (such as the Trapezoidal rule) and use this condition as the last line in matrix system (9), we will destroy the tridiagonal nature of our finite difference system, making it computationally heavy. Furthermore, because at the end point $x = 1$, the coefficient for the right boundary node $u(1, t)$ vanishes entirely in the integral, meaning that one can't explicitly solve for the right boundary at each time loop which cause instability in CN-FDM scheme. However, in order to avoid this issue, we will calculate the value of the right boundary from the governing equation as next. From Equation (4), and differentiating the condition with respect to time once we get:

$$\int_0^1 (x - 1)u_t(x, t)dx = 0,$$

insert the main equation (1) into this integral after dividing on the non-zero coefficient $a_1(t) > 0$ we obtain;

$$\int_0^1 (x - 1) \left[\frac{1}{a_1(t)} (u_{xx} - a(t)u(x, t) + b(t)g(x, t) + f(x, t)) \right] dx = 0,$$

via condition (3) and using the integration by part once, we arrive to the following expression;

$$u(1, t) = u(0, t) + \int_0^1 (x - 1)(b(t)g(x, t) + f(x, t))dx,$$

using the trapezoidal rule for the integral part;

$$u_{M_x, j} = u_{0, j} + \frac{\Delta x}{2} [(x_0 - 1)(b_j g_{0, j} + f_{0, j}) + 2 \sum_{i=1}^{M_x-1} (x_i - 1)(b_j g_{i, j} + f_{i, j})], \quad (12)$$

That is, the first matrix Σ has the following structure without loss of generality, we assume $M = N$:

$$\Sigma = \begin{bmatrix} (1+B_{j+1}) & -2A_{j+1} & 0 & \cdots & 0 & 0 \\ -A_{j+1} & (1+B_{j+1}) & -A_{j+1} & \cdots & 0 & 0 \\ 0 & -A_{j+1} & (1+B_{j+1}) & -A_{j+1} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & -A_{j+1} & (1+B_{j+1}) & -A_{j+1} \\ -A_{j+1} & 0 & 0 & \cdots & -A_{j+1} & (1+B_{j+1}) \end{bmatrix}_{(M_x-1) \times (M_x-1)}$$

The second matrix Γ :

$$\Gamma = \begin{bmatrix} (1-B_j) & 2A_j & 0 & \cdots & 0 & 0 \\ A_j & (1-B_j) & A_j & \cdots & 0 & 0 \\ 0 & A_j & (1-B_j) & A_j & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & A_j & (1-B_j) & A_j \\ A_j & 0 & 0 & \cdots & A_j & (1-B_j) \end{bmatrix}_{(M_x-1) \times (M_x-1)}$$

The vectors V_1 and V_2 are as follows;

$$V_1 = \begin{bmatrix} b_j g_{0,j} + f_{0,j} \\ b_j g_{1,j} + f_{1,j} \\ \vdots \\ b_j g_{M_x-2,j} + f_{M_x,j} \\ b_j g_{M_x-1,j} + f_{M_x-1,j} + A_j S_j \end{bmatrix}_{(M_x-1) \times 1}$$

$$V_2 = \begin{bmatrix} b_{j+1} g_{0,j+1} + f_{0,j+1} \\ b_{j+1} g_{1,j+1} + f_{1,j+1} \\ \vdots \\ b_{j+1} g_{M_x-2,j+1} + f_{M_x,j+1} \\ b_{j+1} g_{M_x-1,j+1} + f_{M_x-1,j+1} + A_{j+1} S_{j+1} \end{bmatrix}_{(M_x-1) \times 1}$$

where,

$$S_j = \frac{\Delta x}{2} [(x_0 - 1)(b_j g_{0,j} + f_{0,j}) + 2 \sum_{i=1}^{M_x-1} (x_i - 1)(b_j g_{i,j} + f_{i,j})]$$

$$S_{j+1} = \frac{\Delta x}{2} [(x_0 - 1)(b_{j+1} g_{0,j+1} + f_{0,j+1}) + 2 \sum_{i=1}^{M_x-1} (x_i - 1)(b_{j+1} g_{i,j+1} + f_{i,j+1})]$$

3.1 Stability and Convergence Analysis

The adopted Crank-Nicolson (C-N) finite difference scheme used to solve the direct problem is unconditionally stable and second-order accurate in spatial and temporal discretizations as G.D. Smith mentioned in his book [6]. A brief von Neumann stability analysis shows that the amplification factor satisfies $|G| = 1$ "G is Amplification factor" for all mesh ratios, which guarantees the unconditional stability of the scheme. The local truncation error is of the order $\mathcal{O}((\Delta x)^2, (\Delta t)^2)$, which confirms that the method provides second-order accuracy and reliable convergence toward the analytical solution as the mesh is refined. Hence, the C-N formulation ensures both numerical stability and precision for the present problem.

3.2 Examples for direct problem (1) to (4)

3.2.1 Example 1

Assume an illustration for the direct problem given by (1)-(4) with $\mathcal{D}_T: [0,1] \times [0,T]$ and the input data are

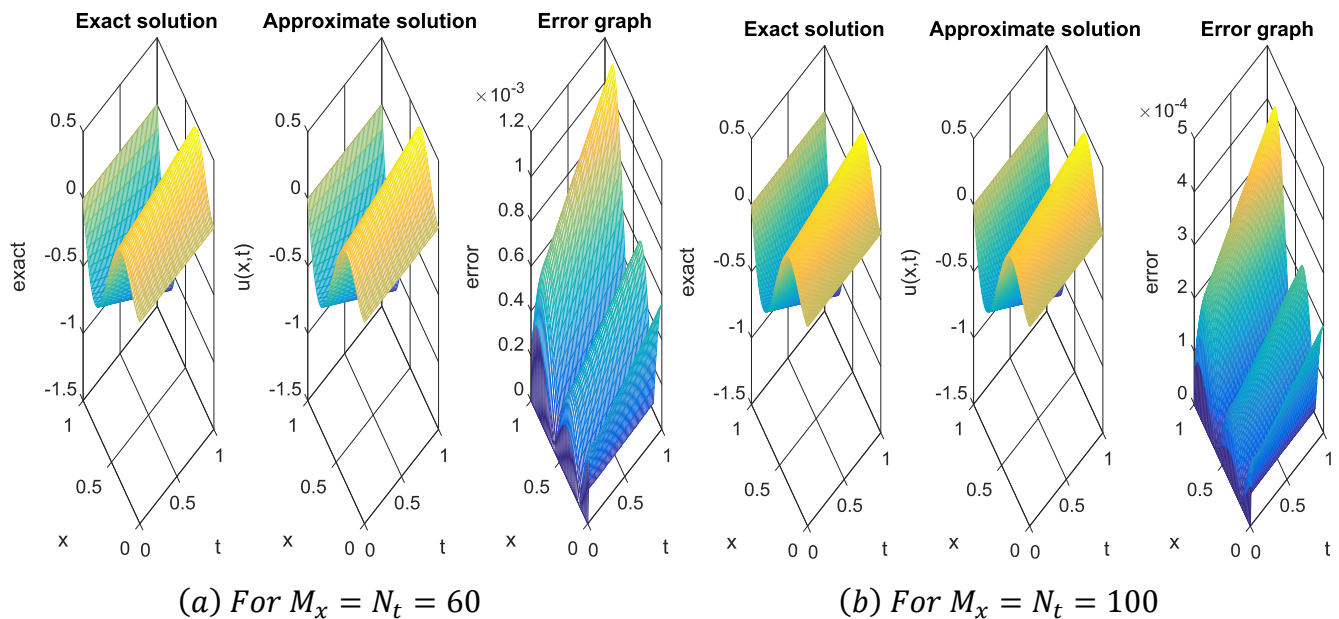
$$\begin{aligned} f(x,t) &= (t+2)(15x^7 - 77x^6 + 126x^5 - 70x^4 + 6x^2) \\ &+ (t^2+1)(1+t)(15x^7 - 77x^6 + 126x^5 - 70x^4 + 6x^2) \\ &- (2t+1)(21x^5 - 65x^4 + 60x^3 - 16x^2) \\ &- (t+1)(630x^5 - 2310x^4 + 2520x^3 - 840x^2 + 12) \\ a(t) &= t^2 + 1, \quad b(t) = 2t + 1, \quad a_1(t) = t + 1 \\ g(x,t) &= (21x^5 - 65x^4 + 60x^3 - 16x^2), \\ \varphi(x) &= (15x^7 - 77x^6 + 126x^5 - 70x^4 + 6x^2). \end{aligned}$$

and the analytical solution is

$$u(x,t) = (t+1)(15x^7 - 77x^6 + 126x^5 - 70x^4 + 6x^2)$$

whilst, desired output are calculated at $(x_1, x_2) = (0.25, 0.75)$,

$$h_1 = 0.2067(t+1), \quad h_2 = -0.5576(t+1), \quad t \in [0,1],$$



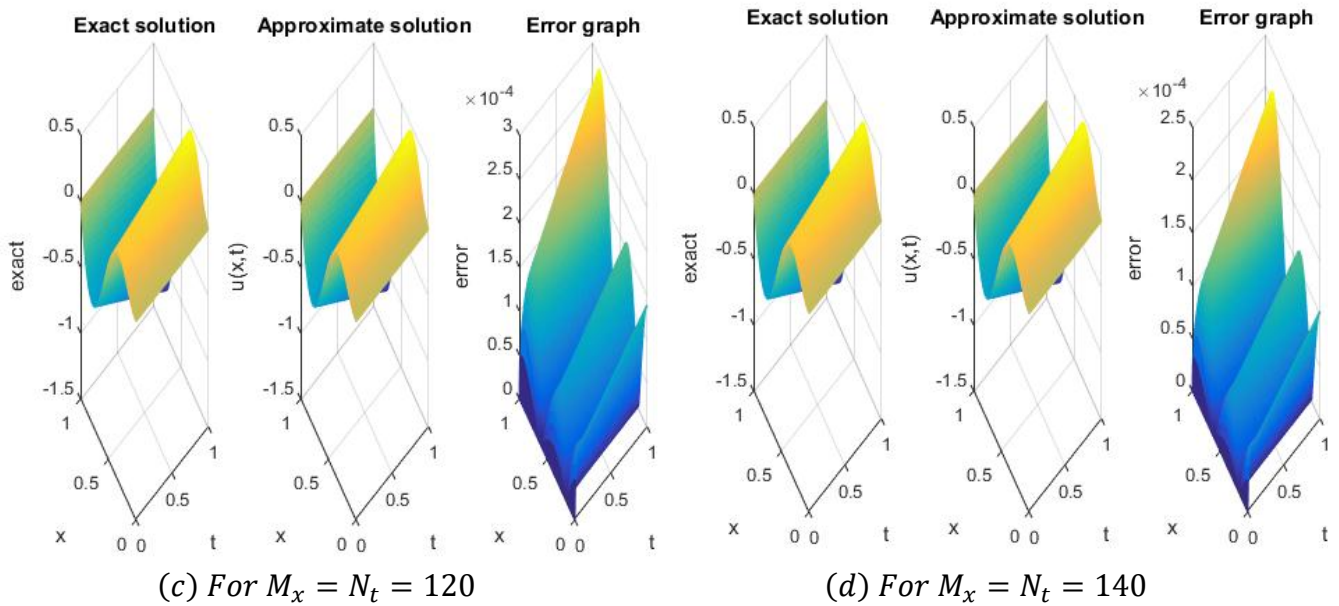
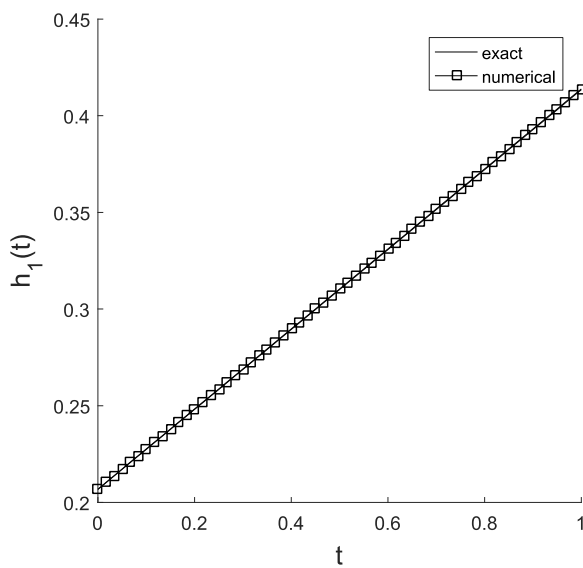
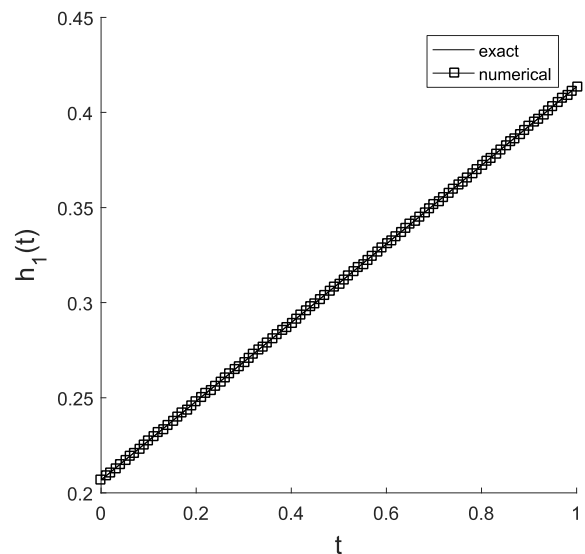


Figure 1: The graphs showing analytical and computational distribution for the function $u(x, t)$ with absolute error for the direct problem (1) - (4), for example 1 (a) - (d), represent the size of mesh $M_x = N_t \in \{60, 100, 120, 140\}$, respectively.



(a) For $M_x = N_t = 60$



(b) For $M_x = N_t = 100$

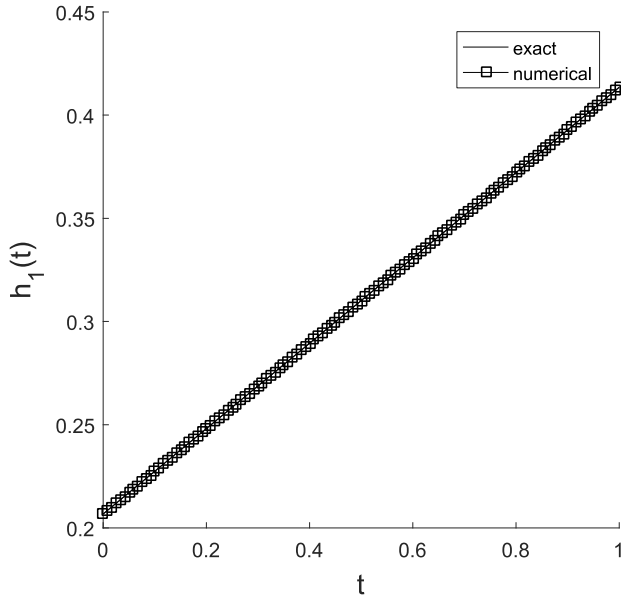
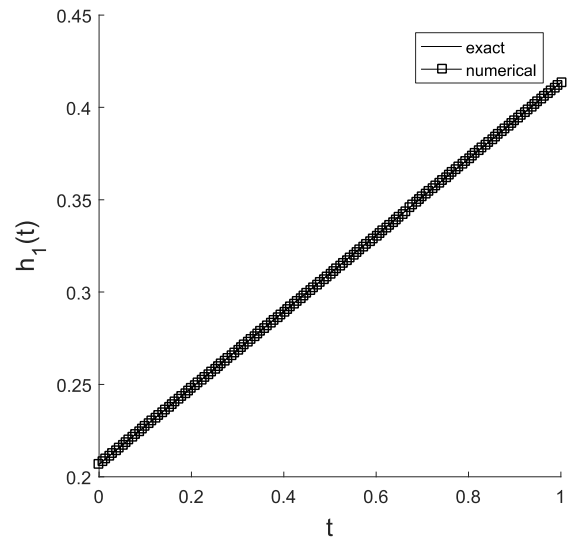
(c) For $M_x = N_t = 120$ (d) For $M_x = N_t = 140$

Figure 2: The graphs showing the exact and numerical values for $h_1 = u(0.25, t)$ equation (5) for example 1 with the size of mesh $M_x = N_t \in \{60, 100, 120, 140\}$

These Figure 2 (a) to (d) represent the numerical solution of the direct problem (1) to (4) using the Crank-Nicolson finite difference method. The spatial domain x and temporal domain t , where $x \in [0, 1]$ and $t \in [0, 1]$, are discretized into an $M_x \times N_t$ grid where $M_x = N_t \in \{60, 100, 120, 140\}$. It is observed that the accuracy of the solution increases as the number of grid subdivisions increases, yielding the highest accuracy at 140 compared to $\{60, 100, 120\}$. This progressive improvement in accuracy indicates that the solutions are stable and convergent, yielding an error rate of less than 10^{-4} for 140 subdivisions, which is both reasonable and acceptable.

It is observed that $h_1 = u(0.25, t)$ is identical in formulation at the examined point $x_1 = 0.25$. Therefore, it suffices to plot h_1 since both will take the similar shape shown in Figure (a) to (d). Furthermore, we also observe in this example that the solutions converge toward the exact solution as the number of grid subdivisions $\{60, 100, 120, 140\}$ increases, where the Crank-Nicolson finite difference method was employed as well. Furthermore, to empirically verify the theoretical second-order spatial and temporal accuracy of the adopted Crank-Nicolson finite difference scheme, the experimental order of convergence (EOC) was evaluated by successive mesh refinements, as reported in Table 1.

Based on the maximum absolute error (L_∞ -norm), the convergence rate q is calculated using the relation $q = \log_2 \left(\frac{E_{N_1}}{E_{N_2}} \right)$. As the mesh resolution increases from $N_t = 20$ to $N_t = 120$, the computational results yield experimental orders of convergence that asymptotically approach the theoretical value, reaching **1.9972** for the approximate solution $u(x, t)$ at the finest grid transition ($60 \rightarrow 120$). For the desired outputs at the examined points, the convergence rates for $h_1(t)$ consistently stabilize at **2.0002** and $h_2(t)$ at **1.9989**. These empirical values closely align with the theoretical truncation error of $\mathcal{O}((\Delta x)^2, (\Delta t)^2)$ (i.e., $q \approx 2$), confirming that the numerical treatment of the nonlocal integral condition via the trapezoidal rule successfully preserves both the second-order accuracy and the unconditional stability of the direct problem formulation across all tested mesh sizes.

Table 1: Experimental order of convergence (EOC) for Example 2 under successive mesh refinements ($M_x = N_t$).

Mesh Refinement	Order of Convergence (q)		
$(N_1 \rightarrow N_2)$	$u(x, t)$	$h_1(t)$	$h_2(t)$
20 $\rightarrow$ 40	1.9913	2.0021	1.9898
40 $\rightarrow$ 80	1.9967	2.0005	1.9975
60 $\rightarrow$ 120	1.9972	2.0002	1.9989

3.2.2 Example 2

Assume an illustration for the direct problem given by (1)-(4) with $\mathcal{D}_T: [0,1] \times [0, T]$ and the input data are

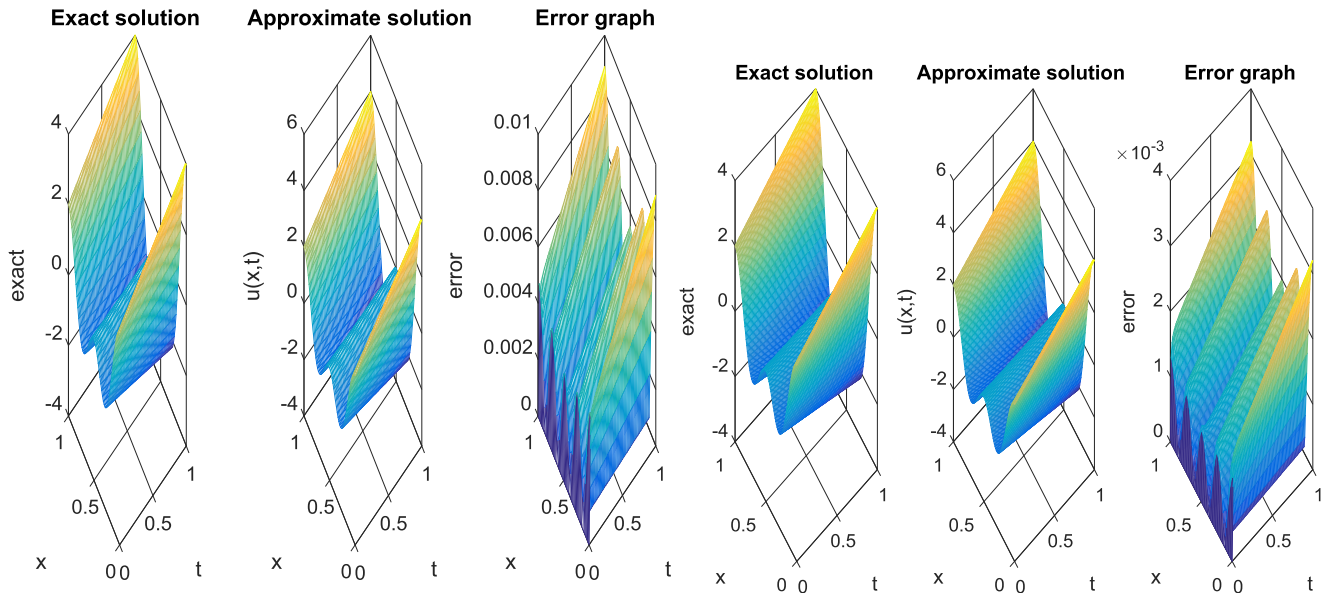
$$\begin{aligned}
 f(x, t) &= (t + 1)(\cos(4\pi x) + \cos(2\pi x)) + (t^2 + 1)(1 + t)(\cos(4\pi x) + \cos(2\pi x)) \\
 &\quad - e^t(21x^5 - 65x^4 + 60x^3 - 16x^2) + 4\pi^2(1 + t)(4\cos(4\pi x) + \cos(2\pi x)) \\
 a(t) &= t^2 + 1, \quad b(t) = e^t, \quad a_1(t) = t + 1 \\
 g(x, t) &= 21x^5 - 65x^4 + 60x^3 - 16x^2, \\
 \varphi(x) &= \cos(4\pi x) + \cos(2\pi x)
 \end{aligned}$$

and the analytical solution is

$$u(x, t) = (1 + t)(\cos(4\pi x) + \cos(2\pi x))$$

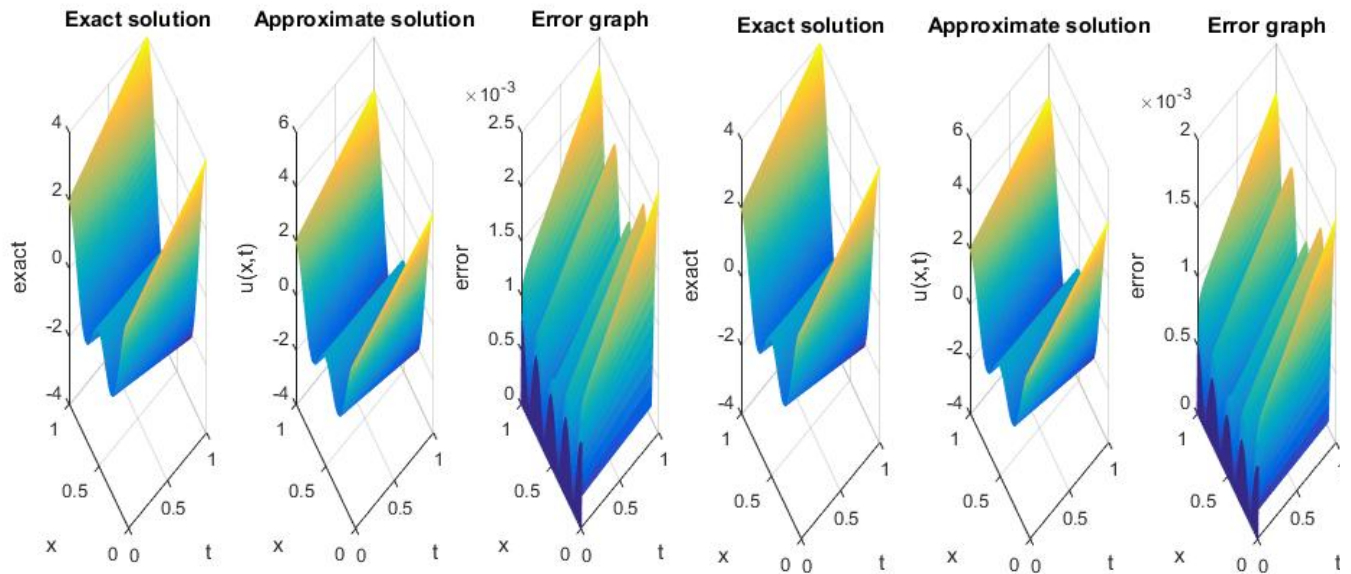
whilst, desired output are calculated at $(x_1, x_2) = (0.25, 0.75)$,

$$h_1 = -(t + 1), \quad h_2 = -(t + 1), \quad t \in [0, 1],$$



(a) For $M_x = N_t = 60$

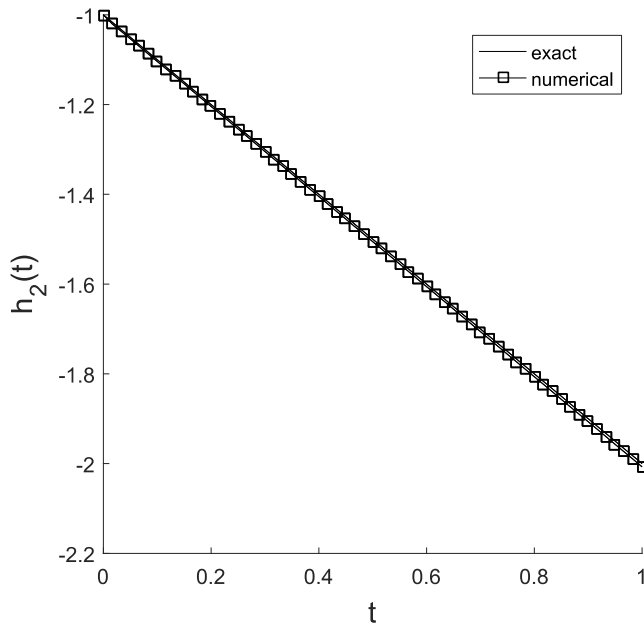
(b) For $M_x = N_t = 100$



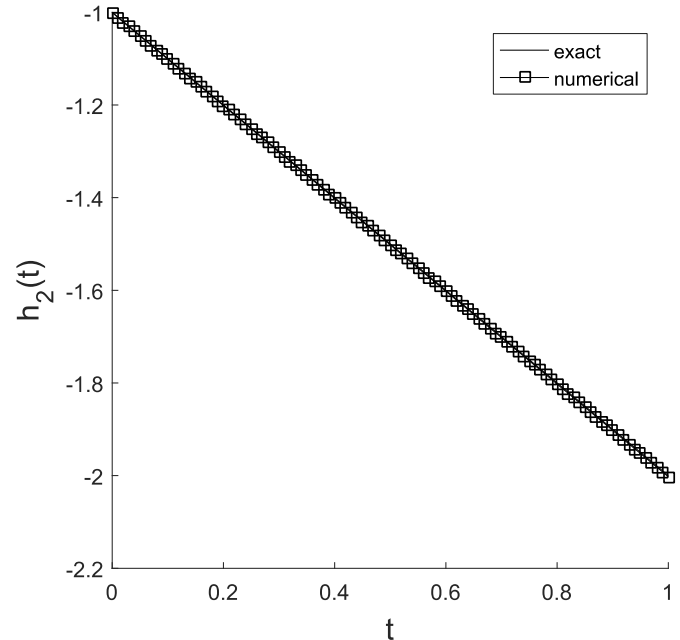
(c) For $M_x = N_t = 120$

(d) For $M_x = N_t = 140$

Figure 3: The graphs showing analytical and computational distribution for the function $u(x, t)$ with absolute error for the direct problem (1) - (4), for example 2 (a) - (d), represent the size of mesh $M_x = N_t \in \{60, 100, 120, 140\}$, respectively.



(a) For $M_x = N_t = 60$



(b) For $M_x = N_t = 100$

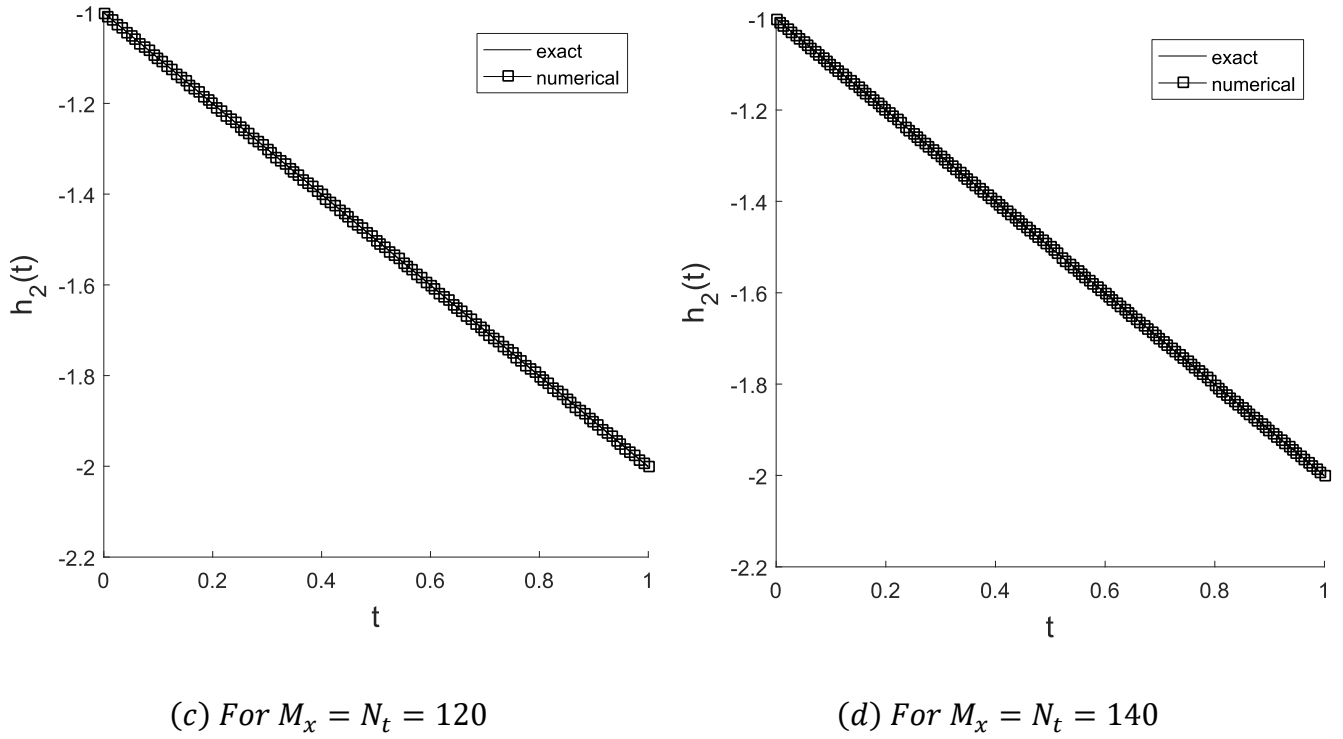


Figure 4: The graphs showing the exact and numerical values for $h_2 = u(0.75, t)$ equation(5) for example 2 with the size of mesh $M_x = N_t \in [60, 100, 120, 140]$.

These figure 3 (a) to (d) represent the numerical solution of the direct problem (1) to (4) using the Crank-Nicolson finite difference method. The spatial domain x and temporal domain t , where $x \in [0, 1]$ and $t \in [0, 1]$, are discretized into an $M_x \times N_t$ grid where $M_x = N_t \in \{60, 100, 120, 140\}$. It is observed that the accuracy of the solution increases as the number of grid subdivisions increases, yielding the highest accuracy at 140 compared to $\{60, 100, 120\}$. This progressive improvement in accuracy indicates that the solutions are stable and convergent, yielding an error rate of less than 10^{-4} for 140 subdivisions, which is both reasonable and acceptable.

Figure 4: It is observed that $h_1 = -(t + 1)$ and $h_2 = -(t + 1)$ are identical in formulation at the two examined points $x_1 = 0.25$ and $x_2 = 0.75$. Therefore, it suffices to plot h_2 since both will take the same shape shown in Figure 4. Furthermore, we also observe in this example that the solutions converge toward the exact solution as the number of grid subdivisions $\{60, 100, 120, 140\}$ increases, where the Crank-Nicolson finite difference method was employed as well, similar to the first example. Furthermore, to empirically verify the theoretical second-order spatial and temporal accuracy of the adopted Crank-Nicolson finite difference scheme, the experimental order of convergence (EOC) was evaluated by successive mesh refinements, as reported in Table 2. Based on the maximum absolute error (L_∞ -norm), the convergence rate q is calculated using the relation $q = \log_2 \left(\frac{E_{N_1}}{E_{N_2}} \right)$. As the mesh resolution increases from $N_t = 20$ to $N_t = 120$, the computational results yield experimental orders of convergence that asymptotically approach the theoretical value, reaching **2.0020** for the approximate solution $u(x, t)$ at the finest grid transition ($60 \rightarrow 120$). For the desired outputs at the examined points, the convergence rates for $h_1(t)$ and $h_2(t)$ consistently stabilize at **2.0023**. These empirical values closely align with the theoretical truncation error of $\mathcal{O}((\Delta x)^2, (\Delta t)^2)$ (i.e., $q \approx 2$), confirming that the numerical treatment of the nonlocal integral condition via the trapezoidal rule successfully preserves both the second-order accuracy and the unconditional stability of the direct problem formulation across all tested mesh sizes.

Table 2: Experimental order of convergence (EOC) for Example 2 under successive mesh refinements ($M_x = N_t$).

Mesh Refinement	Order of Convergence (q)		
$(N_1 \rightarrow N_2)$	$u(x, t)$	$h_1(t)$	$h_2(t)$
20 $\rightarrow$ 40	2.0178	2.0208	2.0208
40 $\rightarrow$ 80	2.0044	2.0052	2.0052
60 $\rightarrow$ 120	2.0020	2.0023	2.0023

4. Numerical approach for Solving Inverse problem (IP)

For the nonlinear inverse problem (IP) governed by Equations (1)–(5), the objective is to achieve a precise and stable identification of the unknown terms $a(t)$ and $b(t)$, alongside the state variable $u(x, t)$. Since the system is governed by a one-dimensional second-order parabolic equation, this inverse challenge is reformulated into a computationally trackable nonlinear least-squares minimization task.

4.1 The Numerical Optimization Approach

To solve the resulting minimization problem, a robust numerical solution is executed using the 'lsqnonlin' routine within MATLAB's Optimization Toolbox. This routine utilizes a trust-region reflective algorithm, which is highly efficient for handling nonlinear least-squares problems under specific bound constraints. The numerical procedure is iterative and begins from a suitably selected initial guess.

4.2 Stabilization and Computational Regularization

Because inverse problems are inherently ill-posed meaning they are highly sensitive to noise in the data Tikhonov Regularization (TR) is employed to stabilize the numerical findings.

The regularization parameter $\beta \geq 0$ is carefully selected to balance the trade-off between the raw measured data and the numerically computed solution. Using 'lsqnonlin', the numerical solver minimizes the following Tikhonov regularized objective functional:

$$F(a, b) = \sum_{i=1}^2 \|u(x_i, t) - h_i(t)\|_{L^2[0, t]}^2 + \beta \left(\|a(t)\|_{L^2[0, t]}^2 + \|b(t)\|_{L^2[0, t]}^2 \right) \quad (13)$$

where $\beta > 0$ is the regularization parameter. To implement this framework practically on a computer, the continuous functional is converted into its discretization form, yielding the discrete version of Equation (13):

$$F(\underline{a}, \underline{b}) = \sum_{i=1}^2 \sum_{j=1}^{N_t} [u(x_i, t_j) - h_i(t_j)]^2 + \beta \left(\sum_{j=1}^{N_t} a_j^2 + \sum_{j=1}^{N_t} b_j^2 \right) \quad (14)$$

5. Results and discussion

In this section, we present the numerical simulation results to demonstrate the accuracy and efficiency of the proposed method (using the Crank-Nicolson finite difference method combined with the LSQNONLIN algorithm and Tikhonov regularization) for the simultaneous retrieval of the time-dependent unknown coefficients $a(t)$ and $b(t)$. The spatial and temporal domains were discretized using a uniform grid with dimensions.

To evaluate the accuracy of the extracted numerical solutions, we used the Root Mean Square Error (RMSE) metric for both coefficients, given by the following formulas:

$$RMSE(a) = \sqrt{\frac{1}{N_t} \sum_{j=1}^{N_t} (a_{\text{exact}}(t_j) - a_{\text{approx}}(t_j))^2} \quad (15)$$

$$RMSE(b) = \sqrt{\frac{1}{N_t} \sum_{j=1}^{N_t} (b_{\text{exact}}(t_j) - b_{\text{approx}}(t_j))^2} \quad (16)$$

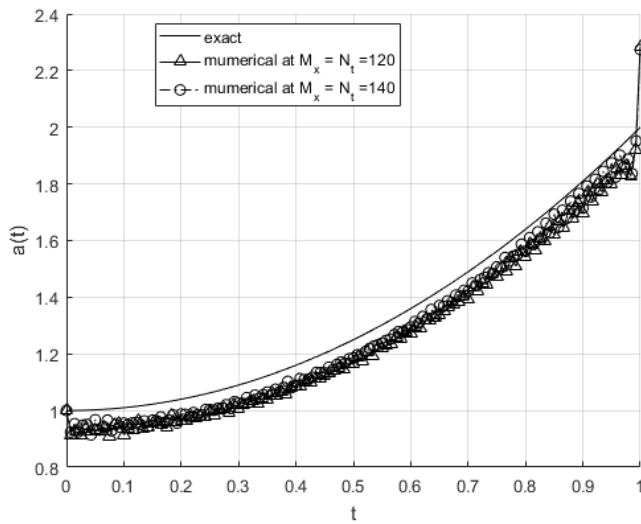
To test the stability of the method against measurement errors, the exact data $h_i(t)$ was perturbed by adding normally distributed random noise, according to the following relation:

$$h_i^\epsilon(t_j) = h_i(t_j) + \epsilon_i(t_j), \quad i = 1, 2 \quad (17)$$

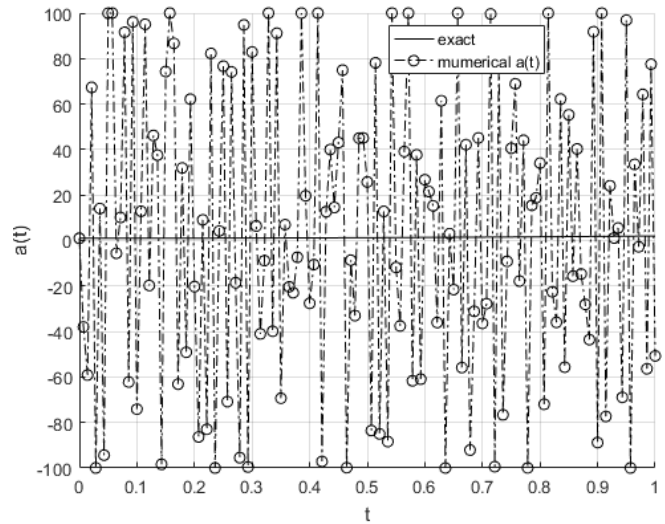
where $\epsilon_i(t_j)$ is a Gaussian random variable with zero mean and a standard deviation σ_i . The standard deviation is chosen based on the noise level for example $p = 1\%$ and is defined as $\sigma_i = \frac{1}{100} \max_j |h_i(t_j)|_{i=1,2}$ and $\epsilon = \text{normrnd}(0, \sigma, N_t)$. To ensure consistency across numerical experiments, the state of the random number generator was reset before adding the noise. To handle the ill-posedness caused by this noisy data, the Tikhonov regularization parameters β was selected from the range $\{10^{-1} \text{ to } 10^{-10}\}$ by trial and error technique.

5.1 Example 1

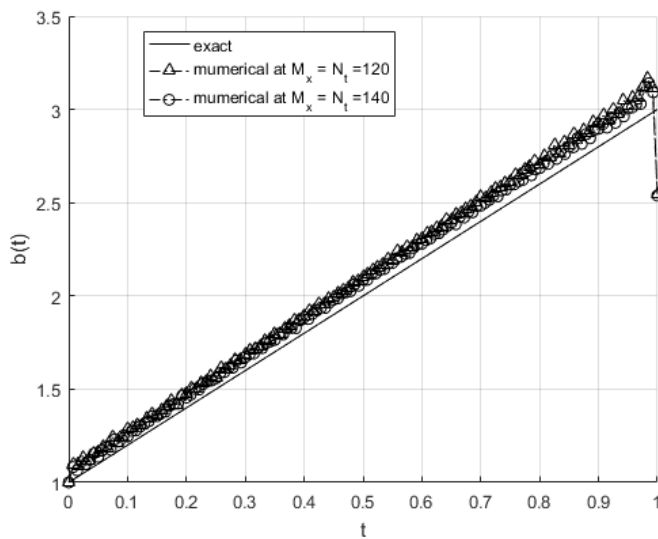
In this example, we test the algorithm's ability to retrieve time-dependent coefficients when the exact solution $u(x, t)$ is a polynomial function. We consider the problem with the same data Example 1 with the additional measurement points $(x_1, x_2) = (0.25, 0.75)$, within the time interval $t \in [0, 1]$ and the total domain $(x, t) \in \mathcal{D}_T$.



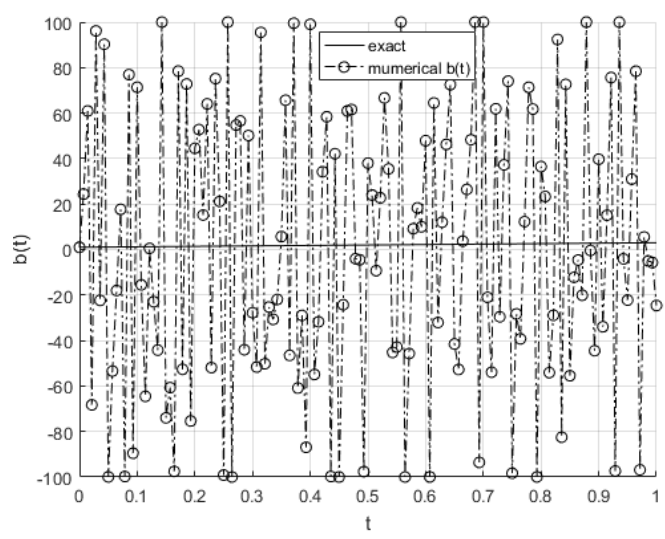
(a) For $M_x = N_t \in \{120, 140\}$ without noise



(b) For $M_x = N_t \in 140$ with noise 1%



(c) For $M_x = N_t \in \{120, 140\}$ without noise



(d) For $M_x = N_t \in 140$ with noise 1%

Figure 5: The graphs showing the exact and numerical inverse values for $a(t), b(t)$ example 1 without noise (a and c), with noise (b and d) $noise = 1\%$ and without regularization ($\beta = 0$).

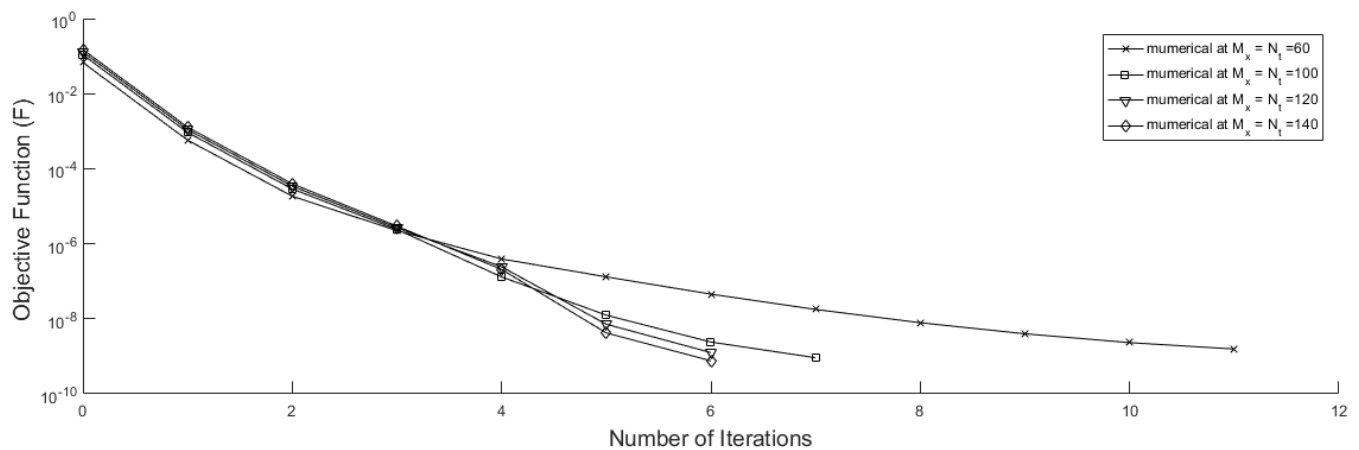


Figure 6: The objective function for example (1) when the size of mesh $M_x = N_t \in \{60, 100, 120, 140\}$ no noise, no regularization.

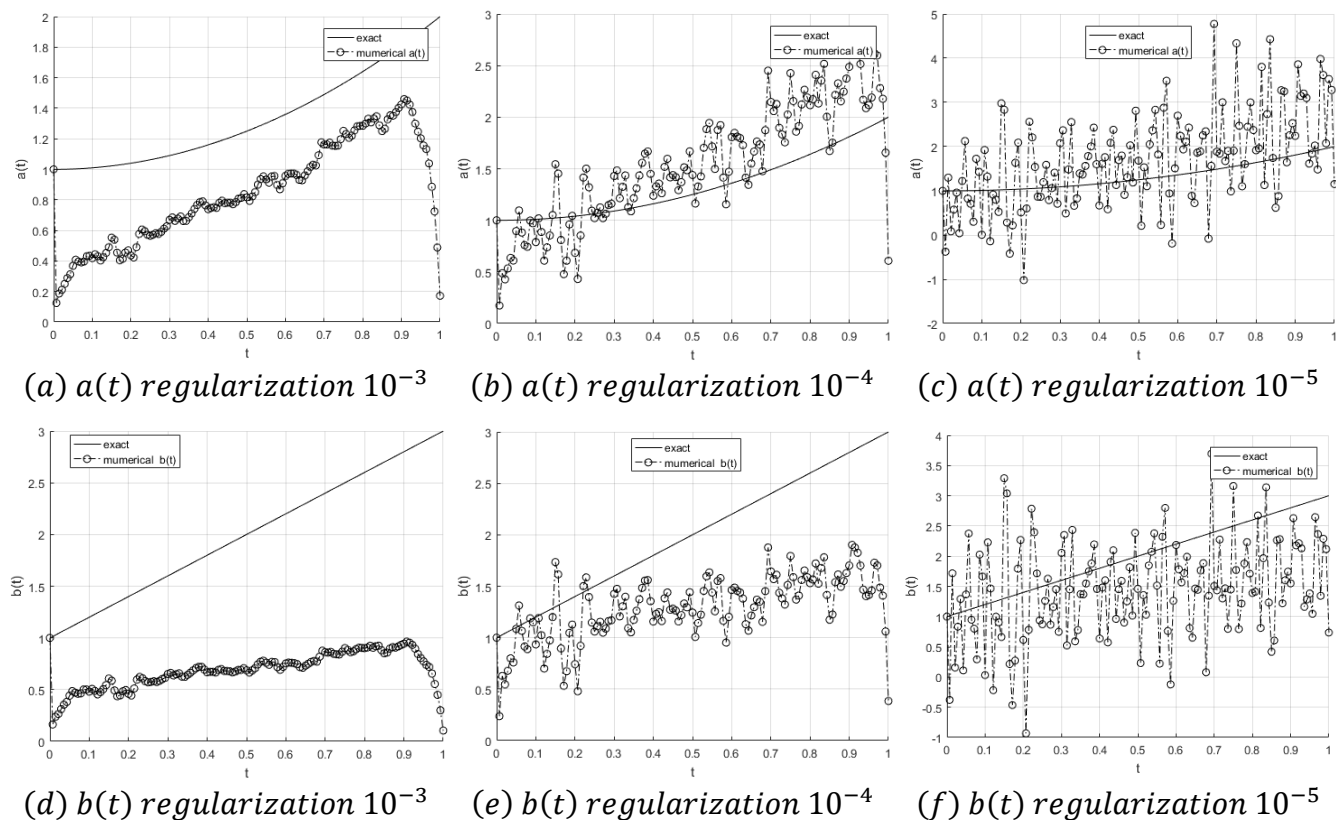


Figure 7: The reconstructed values of $a(t)$ and $b(t)$ for the example 1, in the presence of 1% noise and with regularization parameters $\{10^{-3}, 10^{-4}, 10^{-5}\}$.

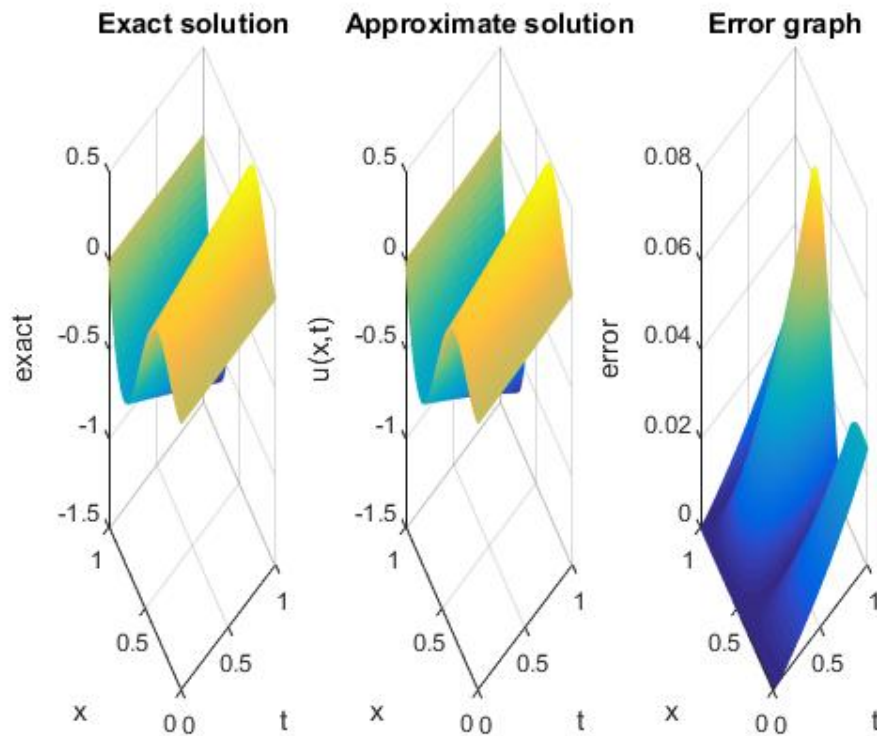


Figure 8: The graphs showing the exact and numerical inverse values for $u(x, t)$ with mesh $M_x = N_t = 140$ example 1 with noise 1% and with regularization $\beta = 10^{-4}$.

Table 3: Effect of the regularization parameter β for example 1

β	0	10^{-10}	10^{-9}	10^{-8}	10^{-7}	10^{-6}	10^{-5}	10^{-4}	10^{-3}	10^{-2}	10^{-1}
No. of iteration	402	402	402	402	402	402	402	402	402	402	402
No. of function evaluations	113766	113766	113766	3964	8207	12735	17263	113766	113766	113766	113766
Value of objective function	0.0013	0.0015	0.0023	0.0044	0.0066	0.0097	0.0205	0.0853	0.3852	0.7014	0.7702
RMSE (a)	63.4117	63.2802	50.2577	20.4687	6.1495	1.0057	2.5060	0.4224	0.5364	1.2009	1.3512
RMSE (b)	60.7396	60.7656	48.5021	17.6739	4.6001	2.0520	1.0208	0.8475	1.4063	1.9597	2.0743

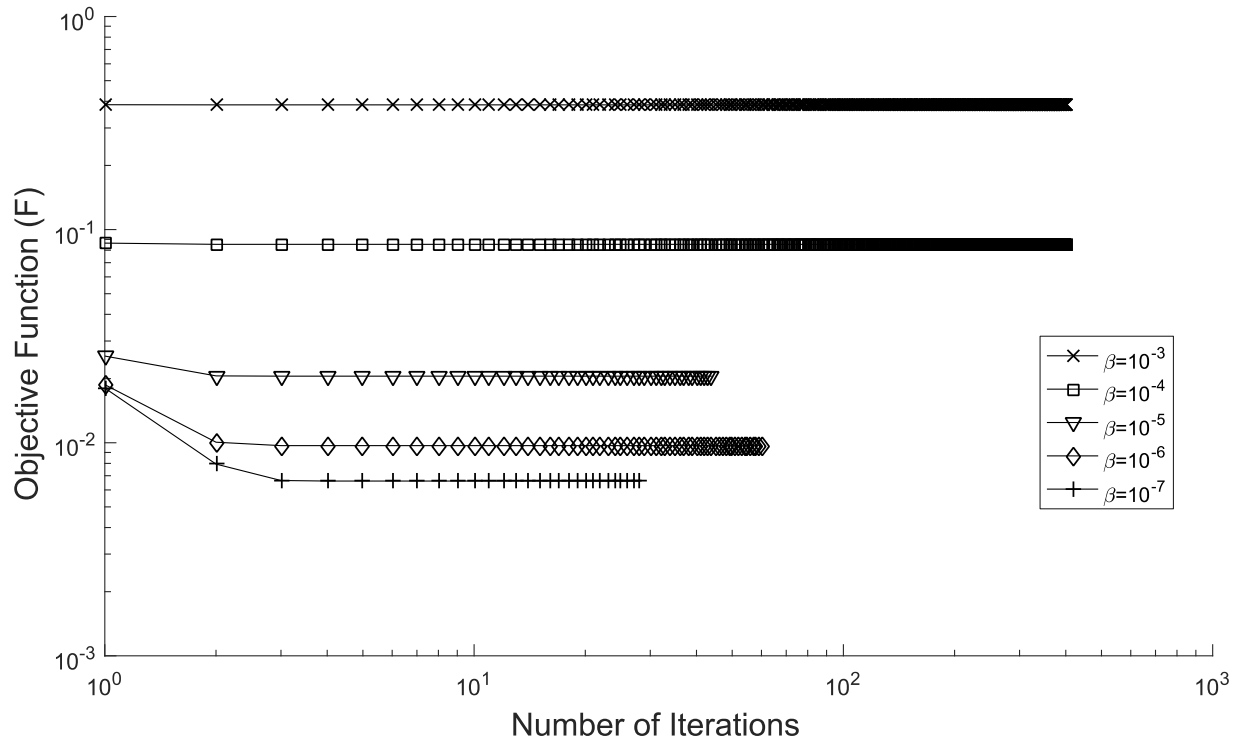
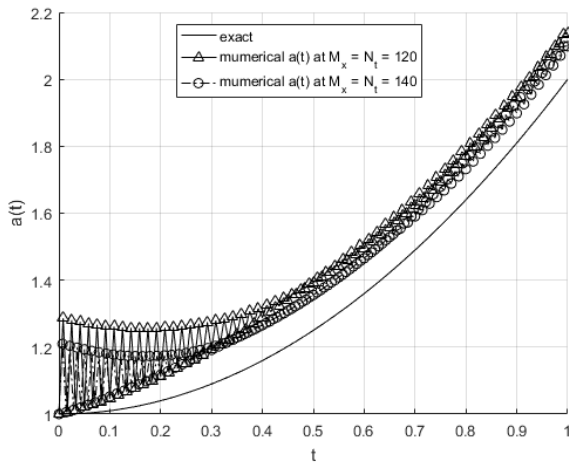


Figure 9: The figure showing the objective function for example 1 when the size of mesh $M = N = 140$, noise 1% and regularization $\beta = [10^{-3} \text{ to } 10^{-7}]$.

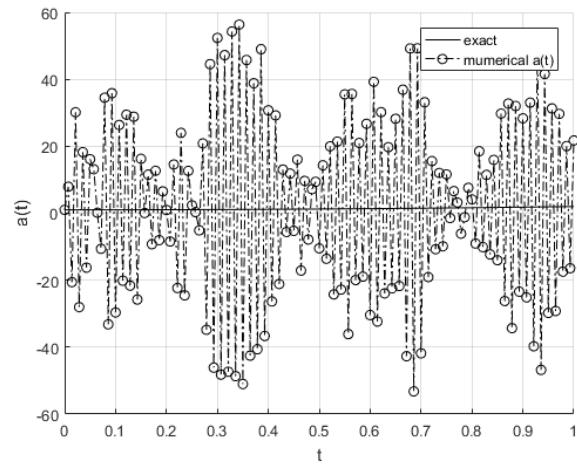
Figure 7 and Table 3 show a comparison between the analytical and approximate solutions for the coefficients $a(t)$ and $b(t)$. In the ideal case of exact data (noise-free, $p = 0\%$ and $\beta = 0$), the objective function stabilized at the value [0.1439], and the $RMSE(a)$ value reached approximately [0.0674] and the $RMSE(b)$ value was approximately [0.0893]. When testing the robustness of the method by adding noise at a level of $p = 1\%$, Tikhonov regularization was activated using optimal values $\beta = [10^{-4}]$ chosen from the set $\{10^{-1} \text{ to } 10^{-10}\}$. The results show that the error stabilized at $RMSE(a) = [0.4224]$, $RMSE(b) = [0.8475]$ and the objective function stabilized at the value [0.0853] after [402] iterations, reflecting excellent stability of the solution despite the presence of 1% noise.

5.2 Example 2

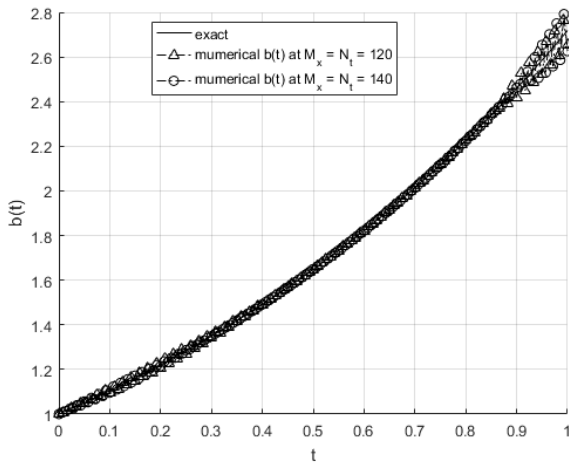
In this example, we make the problem slightly more complex by testing functions that combine trigonometric and exponential terms. The problem data is the same example 2 with the measurement points $(x_1, x_2) = (0.25, 0.75)$, and the time interval $t \in [0, 1]$ within the domain $(x, t) \in \mathcal{D}_T$.



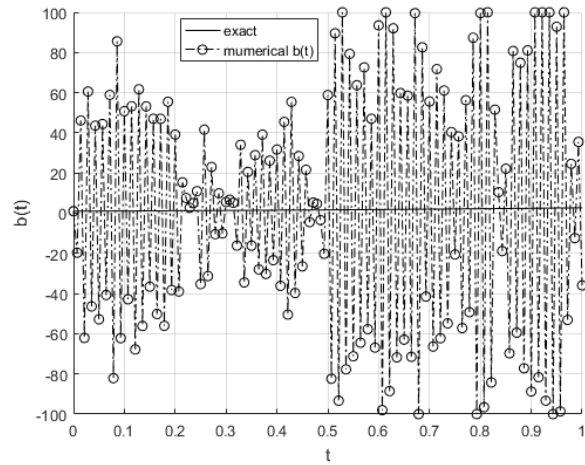
(a) For $M_x = N_t \in \{120,140\}$ without noise



(b) For $M_x = N_t = 140$ with noise 1%



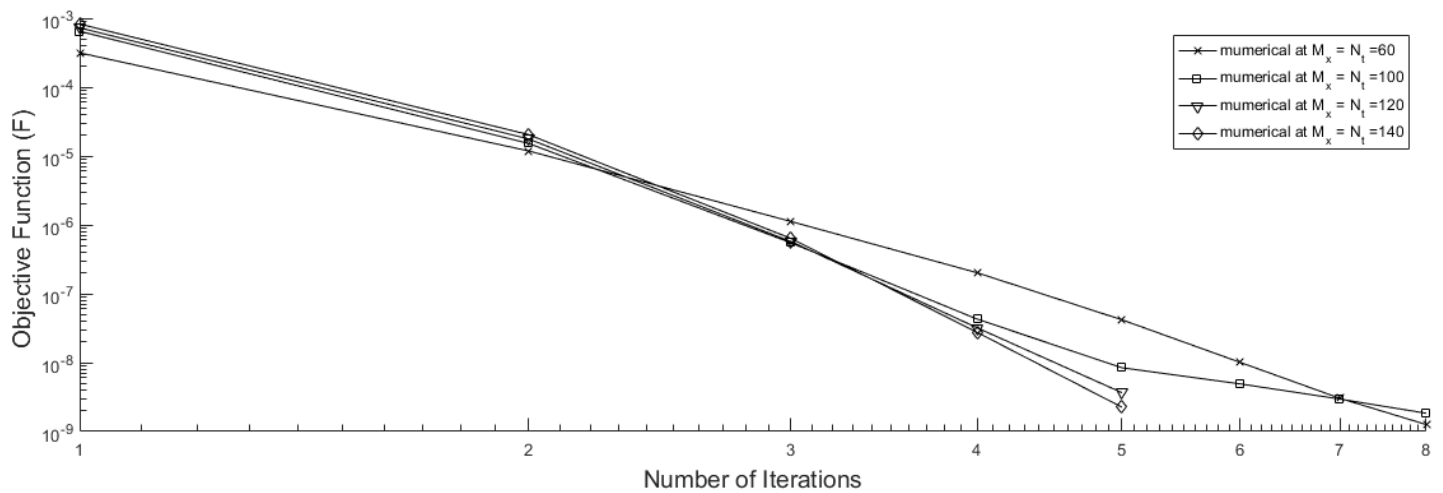
(c) For $M_x = N_t \in \{120,140\}$ without noise



(d) For $M_x = N_t = 140$ with noise 1%

Figure 10: The graphs showing the exact and numerical inverse values for $a(t), b(t)$ example 2 without noise (a and c) with noise (b and d) $noise = 1\%$ without regularization ($\beta = 0$).

Figure 11: The objective function for example 2 when the size of mesh $M = N = \{60,100,120,140\}$



no noise, no regularization.

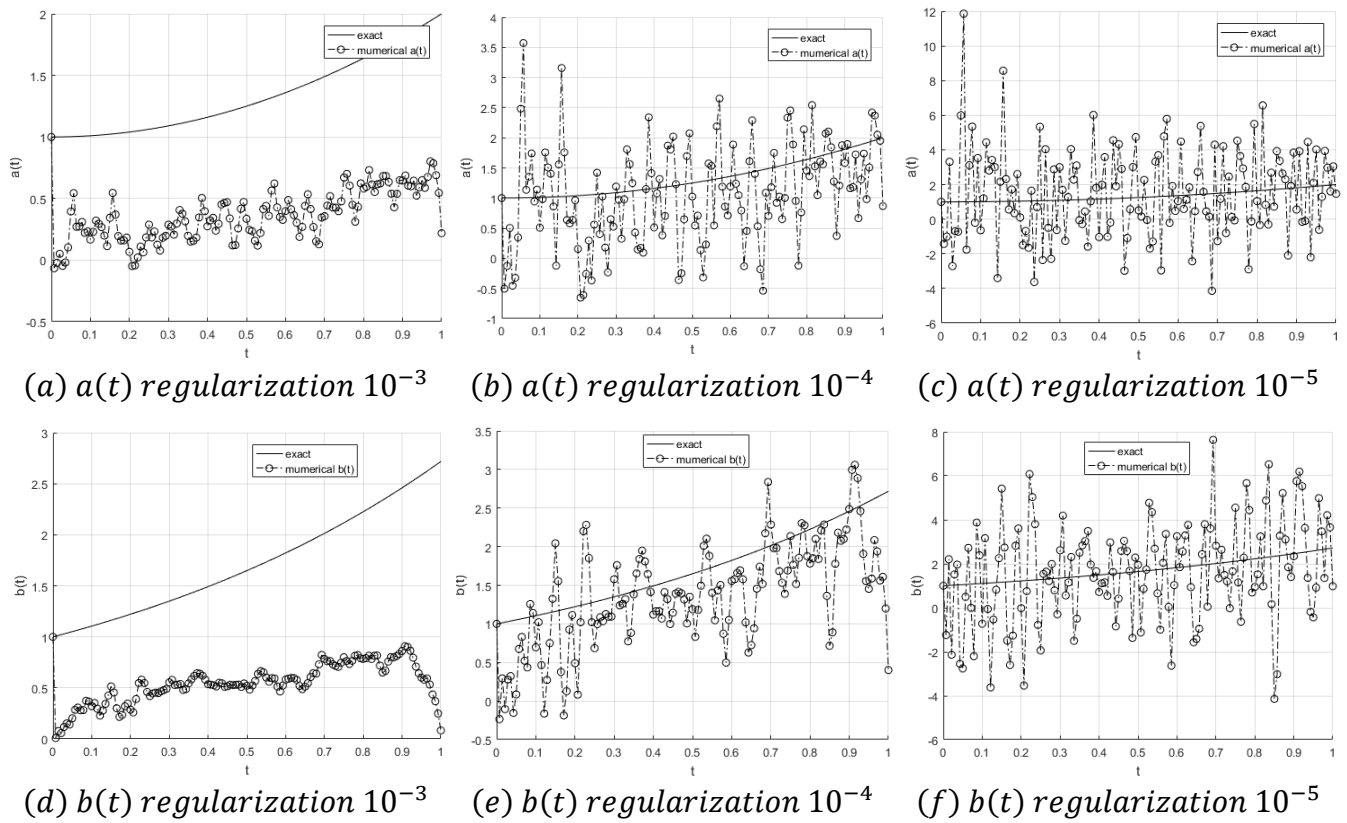


Figure 12: This figure presents the reconstructed values of $a(t)$ and $b(t)$ for the example 2, in the presence of 1% noise and with regularization parameters $\{10^{-3}, 10^{-4}, 10^{-5}\}$.

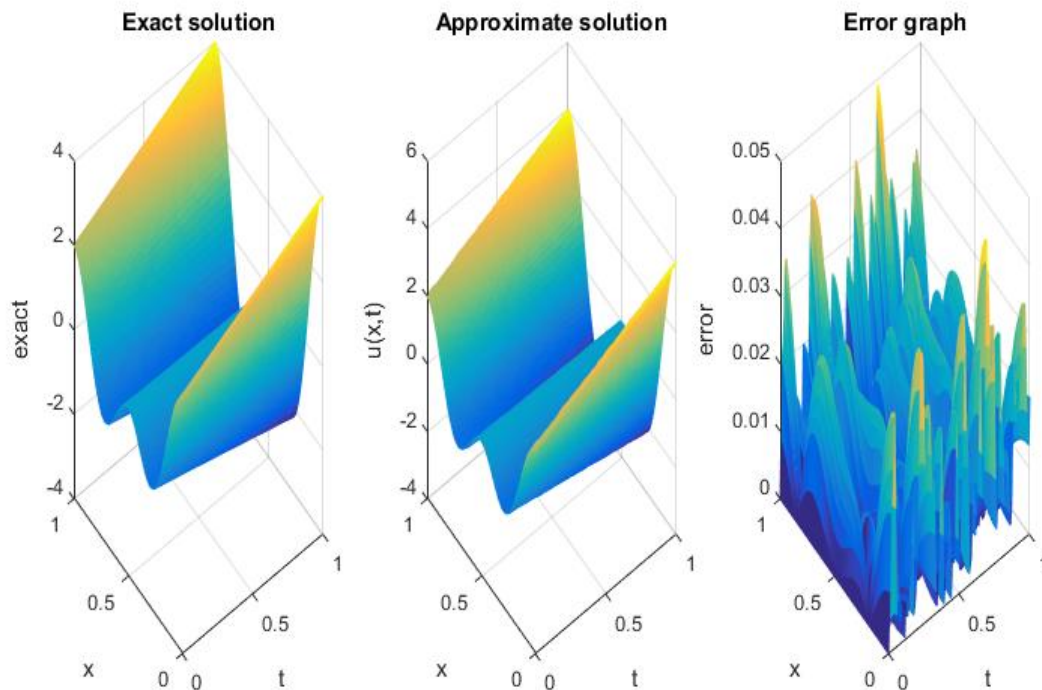
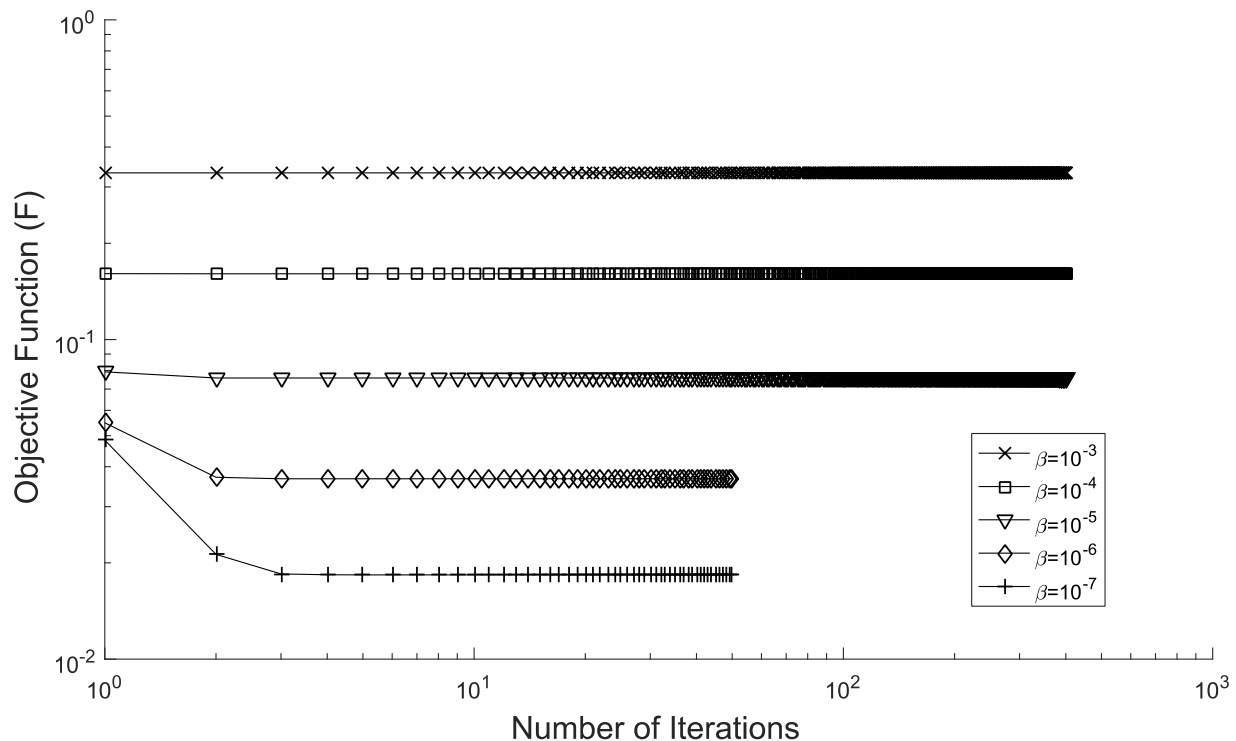


Figure 13: The graphs showing the exact and numerical inverse values for $u(x, t)$ with mesh $M_x = N_t = 140$ example 2 with noise 1% and with regularization $\beta = 10^{-4}$.

Table 4: Effect of the regularization parameter β for example 2

β	0	10^{-10}	10^{-9}	10^{-8}	10^{-7}	10^{-6}	10^{-5}	10^{-4}	10^{-3}	10^{-2}	10^{-1}
No. of iteration	83	88	84	63	51	51	402	402	402	402	402
No. of function evaluations	23489	24904	23772	17829	14433	14433	113766	113766	113766	113766	113766
Value of objective function	0.0058	0.0059	0.0064	0.0099	0.0184	0.0367	0.0758	0.1605	0.3320	0.4216	0.4350
RMSE (a)	27.7799	27.6863	26.9787	20.7536	10.6087	5.8790	2.5785	0.8274	0.9981	1.3161	1.3645
RMSE (b)	60.1477	59.6905	57.4521	38.8949	15.6026	6.6432	2.1969	0.6530	1.2441	1.7138	1.7853

Figure 14: The figure showing the objective function for example 2 when the size of mesh $M = N = 140$, noise 1% and regularization $\beta = [10^{-3} \text{ to } 10^{-7}]$.

As shown in Figure 12 and Table 4, the proposed method successfully captured the exponential behavior of the coefficient $b(t)$ and the quadratic behavior of the coefficient $a(t)$ simultaneously. At the constant noise level of $p = 1\%$, optimal regularization parameters $\beta = [10^{-4}]$ selected from the range $\{10^{-1} \text{ to } 10^{-10}\}$ were used to maintain the accuracy of the solution. Table [2] summarizes the error values, where we achieved $RMSE(a) = [0.8274]$ and $RMSE(b) = [0.6530]$, with the objective function reaching the value $[0.1605]$ after $[402]$ iterations.

These results in both examples confirm that the mathematical formulation and the implemented algorithm possess high robustness and numerical efficiency in solving inverse problems and reliably retrieving unknown coefficients, even under 1% noisy data conditions.

6. Conclusions

In this present, a robust and efficient numerical framework was successfully developed for the simultaneous identification of two time-dependent coefficients, $a(t)$, $b(t)$, time and space-dependent

function $u(x, t)$, in a second-order parabolic partial differential equation. The direct problem was discretized and solved using the unconditionally stable Crank-Nicolson finite difference method. To estimate the unknown coefficients, the inverse problem was formulated as a nonlinear least-squares optimization task and solved iteratively utilizing the 'lsqnonlin' routine based on the trust-region-reflective algorithm.

To rigorously evaluate the performance and stability of the proposed approach under realistic conditions, the exact measurement data were perturbed with 1% normally distributed (Gaussian) noise. We demonstrated that the inherent ill-posedness of the inverse problem can be effectively overcome by integrating Tikhonov regularization into the objective function. By appropriately selecting the regularization parameter β from the admissible range of $\{10^{-1}$ to $10^{-10}\}$, the algorithm successfully suppressed numerical oscillations and yielded stable approximate solutions.

The numerical experiments proved evaluated and accuracy of the method by using two deferent examples including polynomial, exponential, and trigonometric functions and successfully recovered the two time-dependent coefficients, $a(t)$, $b(t)$ and the solution $u(x, t)$, which the Root Mean Square Error (RMSE) was less than 1.

Therefore, this numerical strategy proves to be a highly reliable and practical tool for solving complex inverse parameter estimation problems.

References

- [1] Walter A. Strauss. Partial Differential Equations: An Introduction. John Wiley & Sons, Ltd, Chichester, West Sussex, 2nd edition, 2007.
- [2] W. S. Hailu and G. F. Duressa. Parameter-uniform numerical treatment of singularly perturbed parabolic delay differential equations with nonlocal boundary conditions. *Iranian Journal of Numerical Analysis and Optimization*, 15(1):255–283, 2025.
- [3] M.I. Ismailov, F. Kanca, and D. Lesnic. Determination of a time-dependent heat source under nonlocal boundary and integral overdetermination conditions. *Applied Mathematics and Computation*, 218(8):4138–4146, 2011.
- [4] Elvin Azizbayov and Yashar Mehraliyev. Solvability of nonlocal inverse boundary-value problem for a second-order parabolic equation with integral conditions. *Electronic Journal of Differential Equations*, 2017(125):1–14, 2017.
- [5] Rima Faizi and Atmania Rahima. Determination of an unknown time dependent coefficient for semilinear time-fractional parabolic equation. *International Journal of Nonlinear Analysis and Applications*, 16(8):112–125, 2025.
- [6] Gordon D. Smith. Numerical Solution of Partial Differential Equations: Finite Difference Methods. Oxford University Press, 3rd edition, 1985.
- [7] Richard L. Burden, J. Douglas Faires, and Annette M. Burden. Numerical Analysis. Cengage Learning, Boston, USA, 10th edition, 2015.
- [8] M. A. J. Al-Shatrah and M. S. Hussein. Simultaneous numerical determination of two time-dependent coefficients in second order parabolic equation with nonlocal initial and boundary conditions. *F1000Research*, 15:228, 2026.
- [9] Mousa J. Huntul. Uniqueness and cn–bell spectral reconstruction of three timedependent coefficients in a parabolic inverse problem with quadratic spatial diffusivity. *Mathematics*, 14(11), 2026.
- [10] Miglena N. Koleva and Lubin G. Vulkov. Reconstruction of time-dependent right-hand side in parabolic equations on disjoint domains. *Journal of Physics: Conference Series*, 2675(1):012025, dec 2023.
- [11] Liu Yang, Yun Liu, and Zui-Cha Deng. Multi-parameters identification problem for a degenerate parabolic equation. *Journal of Computational and Applied Mathematics*, 366:112422, 2020.
- [12] El Mustapha Ait Ben Hassi, Salah-Eddine Chorfi, and Lahcen Maniar. An inverse problem of radiative potentials and initial temperatures in parabolic equations with dynamic boundary conditions. *Journal of Inverse and Ill-posed Problems*, 30(3):363–378, 2022.
- [13] Elvin I. Azizbayov and Aynur N. Safarova. An inverse problem for a parabolic equation with nonlocal boundary conditions and two-point overdetermination. *European Journal of Pure and Applied Mathematics*, 18(4):6796, 2025.

- [14] Ya. T. Mehraliev and A. N. Safarova. On one nonlocal inverse boundary problem for the second-order parabolic equation. *Vestnik Yuzhno-Ural'skogo Gosudarstvennogo Universiteta. Seriya Matematika. Mekhanika. Fizika.*, 9(2):13–21, 2017.
- [15] Jacques Hadamard. *Lectures on Cauchy's Problem in Linear Partial Differential Equations*. Yale University Press, 1923.
- [16] J.V. Beck, B. Blackwell, and C.R.S. Clair. *Inverse Heat Conduction: Ill-Posed Problems*. Wiley-Interscience publication. Wiley, 1985.
- [17] Fan Yang and Chu-Li Fu. The method of simplified tikhonov regularization for dealing with the inverse time-dependent heat source problem. *Computers Mathematics with Applications*, 60(5):1228–1236, 2010.
- [18] A. Shidfar, A. Babaei, and A. Molabahrani. Solving the inverse problem of identifying an unknown source term in a parabolic equation. *Computers Mathematics with Applications*, 60(5):1209–1213, 2010.
- [19] O.M. Alifanov. *Inverse Heat Transfer Problems*. International series in heat and mass transfer. Springer-Verlag, 1994.
- [20] Yvon Jarny, M. Necati Ozisik, and Jean Pierre Bardon. A general optimization method using adjoint equation for solving multidimensional inverse heat conduction. *International Journal of Heat and Mass Transfer*, 34:2911–2919, 1991.
- [21] Majid Amirfakhrian. A meshless method based on the fundamental solution and radial basis function for solving an inverse heat conduction problem. *Advances in Mathematical Physics*, 2015, 2015.
- [22] Liviu Marin. The method of fundamental solutions for inverse problems associated with the steady-state heat conduction in the presence of sources. *CMESComputer Modeling in Engineering & Sciences*, 30(2):99–122, 2008.
- [23] M. Raissi, P. Perdikaris, and G.E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019.
- [24] Lu Lu, Xuhui Meng, Zhiping Mao, and George Em Karniadakis. Deepxde: A deep learning library for solving differential equations. *SIAM Review*, 63(1):208–228, 2021.