Computational Methods for Differential Equations
http://cmde.tabrizu.ac.ir
Vol. 7, No. 3, 2019, pp. 434-453

Application of cubic B-splines collocation method for solving non-linear inverse diffusion problem

Hamed Zeidabadi
Faculty of Engineering, Sabzevar University
of New Technology, Sabzevar, Iran.
E-mail: h.zeidabadi@yahoo.com

Reza Pourgholi*
School of Mathematics and Computer Science,
Damghan University, P. O. Box 36715-364, Damghan, Iran.
E-mail: pourgholi@du.ac.ir

Seyed Hashem Tabasi
School of Mathematics and Computer Science,
Damghan University, P. O. Box 36715-364, Damghan, Iran.
E-mail: tabasi@du.ac.ir

Abstract In this paper, we developed a collocation method based on cubic B-spline for solving nonlinear inverse parabolic partial differential equations as the following form

$$u_t = [f(u)\, u_x]_x + \varphi(x,t,u,u_x), \qquad 0 < x < 1,\ \ 0 \le t \le T,$$

where $f(u)$ and φ are smooth functions defined on $\mathbb{R}$. First, we obtained a time discrete scheme by approximating the first-order time derivative via forward finite difference formula, then we used cubic B-spline collocation method to approximate the spatial derivatives and Tikhonov regularization method for solving produced ill-posed system. It is proved that the proposed method has the order of convergence $O(k+h^2)$. The accuracy of the proposed method is demonstrated by applying it on three test problems. Figures and comparisons have been presented for clarity. The aim of this paper is to show that the collocation method based on cubic B-spline is also suitable for the treatment of the nonlinear inverse parabolic partial differential equations.

Keywords. Cubic B-spline, Collocation method, Inverse problems, Convergence analysis, Stability of solution, Tikhonov regularization method, Ill-posed problems, Noisy data.

2010 Mathematics Subject Classification. 65M32, 35K05.

1. Introduction

Inverse problems of parabolic type have been received much attention in various fields of science and technology. They arise for example, in the study of heat conduction processes, chemical diffusion, control theory, thermo-elasticity and etc. They have certainly been one of the fastest growing areas in applied mathematics and engineering over the last two decades due to their variety of applications and have been

Received: 4 November 2017 ; Accepted: 22 September 2018.
$*$ Corresponding author.

studied by many authors [3–6, 11, 12, 15, 16, 22–26, 29]. Inverse problems are usually difficult to solve analytically and therefore the numerical approaches are created to overcome the complexities of analytical methods. One of the well-known numerical approach is cubic B-spline collocation method. The theory of B-spline functions has attracted attention in the literature for the numerical solution of linear and nonlinear boundary value problems in science and engineering [2, 7, 14, 18, 19, 21, 28]. In this paper, the B-spline scaling functions are used to find the approximate solution of the surface heat flux histories and temperature distribution in an inverse heat conduction problem (IHCP) [8].

Recently, Pourgholi and Saeedi [24] used cubic B-spline collocation method to solve inverse partial differential equations as the following form

$$u_t = \varphi(x, t, u, u_x, u_{xx}), \qquad 0 < x < 1, \ 0 \le t \le T.$$

In this work which is an extension of [24], the cubic B-spline is used to solve the following inverse problem of parabolic type in the dimensionless form

$$u_t = [f(u)\, u_x]_x + \varphi(x, t, u, u_x), \qquad 0 < x < 1, \ 0 \le t \le T, \tag{1.1}$$

where $f(u)$ and φ are smooth functions defined on $\mathbb{R}$ such that

$$0 < \mu_1 \le f(u) \le \mu_2, \qquad |f'(u)| \le M \ \ for \ u \in \mathbb{R}, \tag{1.2}$$

and $\dfrac{\partial \varphi}{\partial u}$, $\dfrac{\partial \varphi}{\partial u_x}$ exist and are bounded with boundary conditions

$$u(0, t) = p(t), \qquad 0 \le t \le T, \tag{1.3}$$

$$u(1, t) = q(t), \qquad 0 \le t \le T, \tag{1.4}$$

initial condition

$$u(x, 0) = u_0(x), \qquad 0 \le x \le 1, \tag{1.5}$$

and the overspecified condition

$$u(\beta, t) = r(t), \qquad 0 \le t \le T, \tag{1.6}$$

where $0 < \beta < 1$ is a fixed point, $u_0(x)$ is a continuous known function, $q(t)$ and $r(t)$ are known functions and T represents the final time, while $p(t)$ and $u(x, t)$ are unknowns functions which remains to be determined from the overspecified data.

The outline of this study is as follows. In section 2, a description of the cubic B-splines collocation method is explained. Procedure for implementation of present method for equations (1.1)-(1.6) is described in section 3. In section 4, procedure for obtaining an initial vector which is required to start our method is explained. To regularize the resultant ill-posed linear system of equations, in section 5, we apply the Tikhonov regularization (of 2nd order) method to obtain the stable numerical approximation of our solution. The uniform convergence of the method is provided in section 6. Finally in section 7 numerical experiment is conducted computationally to demonstrate the viability and the efficiency of the proposed method and conclusion is given in section 8 that briefly summarizes the numerical outcomes.

2. Description of Method

In cubic B-splines collocation method the approximate solution can be written as a linear combination of basis functions which constitute a basis for the approximation space under consideration.

To construct numerical solution, we introduce a uniformly distributed set of nodes $0 = x_0 < x_1 < \ldots < x_N = 1$ over the spatial domain $[0, 1]$ and the spacial step length is denoted by $h = \frac{1}{N}$, $h = x_{i+1} - x_i$, $i = 0, 1, \ldots, N-1$. To construct the cubic B-spline, we need to extend the set of nodal points to

$$x_{-3} < x_{-2} < x_{-1} < x_0 \quad \textit{and} \quad x_N < x_{N+1} < x_{N+2} < x_{N+3},$$

where

$$\begin{aligned} x_{-3} &= -3h, & x_{N+1} &= (N+1)h, \\ x_{-2} &= -2h, & x_{N+2} &= (N+2)h, \\ x_{-1} &= -h, & x_{N+3} &= (N+3)h. \end{aligned}$$

The cubic B-spline B_i, $i = -1, 0, \ldots, N+1$, are defined as follows

$$B_i(x) = \frac{1}{h^3} \begin{cases} (x - x_{i-2})^3, & x \in [x_{i-2}, x_{i-1}], \\ h^3 + 3h^2(x - x_{i-1}) + 3h(x - x_{i-1})^2 - 3(x - x_{i-1})^3, & x \in [x_{i-1}, x_i], \\ h^3 + 3h^2(x_{i+1} - x) + 3h(x_{i+1} - x)^2 - 3(x_{i+1} - x)^3, & x \in [x_i, x_{i+1}], \\ (x_{i+2} - x)^3, & x \in [x_{i+1}, x_{i+2}], \\ 0, & \text{otherwise}, \end{cases} \tag{2.1}$$

where $B_i(x)\,(i = -1, \ldots, N+1)$ form a basis for functions defined on the interval $[0, 1]$.

Each cubic B-spline function covers four elements so that an element is covered by four cubic B-splines. All other B-splines are zero in this region. By using splines defined in (2.1), the value of $B_i(x)$ and its derivatives at the nodes x_i's are given by

$$B_m(x_i) = \begin{cases} 4, & \text{if } m = i, \\ 1, & \text{if } |m - i| = 1, \\ 0, & \text{if } |m - i| \geq 2, \end{cases} \quad B'_m(x_i) = \begin{cases} 0, & \text{if } m = i, \\ -\frac{3}{h}, & \text{if } m = i - 1, \\ \frac{3}{h}, & \text{if } m = i + 1, \\ 0, & \text{if } |m - 1| \geq 2, \end{cases}$$
$$B''_m(x_i) = \begin{cases} -\frac{12}{h^2}, & \text{if } m = i, \\ \frac{6}{h^2}, & \text{if } |m - 1| = 1, \\ 0, & \text{if } |m| \geq 2. \end{cases} \tag{2.2}$$

Our numerical scheme for above problem using the collocation method with the cubic B-spline is to find an approximate solution $U(x, t)$ to the exact solution $u(x, t)$ in the form

$$U(x, t) = \sum_{j=-1}^{N+1} c_j(t) B_j(x), \tag{2.3}$$

where B_j's are the cubic B-splines in our proposed method, and c_j's are time-dependent parameters which are determined by solving the problem. By using (2.2) and (2.3), the approximate values of $u(x,t)$ and its two derivatives at the knots are determined in terms of the time parameters c_j as follows:

$$U_j = c_{j-1} + 4c_j + c_{j+1}, \tag{2.4}$$

$$U_j^{'} = \frac{3}{h}(c_{j+1} - c_{j-1}), \tag{2.5}$$

$$U_j^{''} = \frac{6}{h^2}(c_{j-1} - 2c_j + c_{j+1}), \tag{2.6}$$

where $U_j = U(x_j, t)$. Using (2.3) and boundary condition (1.4), we get the approximate solution at the boundary points as

$$U(x_N, t) = \sum_{j=N-1}^{N+1} c_j B_j(x) = c_{N-1} + 4c_N + c_{N+1} = q(t), \tag{2.7}$$

and by using overspecified conditions (1.6), where $\beta = x_s$, $1 \leq s \leq N-1$ and (2.3) we have

$$U(x_s, t) = \sum_{j=s-1}^{s+1} c_j B_j(x) = c_{s-1} + 4c_s + c_{s+1} = r(t). \tag{2.8}$$

3. Implementation of Method

Our numerical scheme for solving equations (1.1)-(1.6) using the collocation method with cubic B-splines is to find approximate solutions $U(x,t)$ and $U(0,t)$, to the exact solution $u(x,t)$ and $p(t)$ are given in (2.3), where $c_j(t)$ are time dependent quantities which are determined from the boundary and overspecific conditions and the collocation from the differential equation. Now, using (2.3) in (1.1), we have

$$\begin{aligned} U_t = &\left[f\Big(\sum_{j=-1}^{N+1} c_j(t) B_j(x)\Big) \Big(\sum_{j=-1}^{N+1} c_j(t) B_j^{'}(x)\Big) \right]_x \\ &+ \varphi\left(x_j, t, \sum_{j=-1}^{N+1} c_j(t) B_j(x), \sum_{j=-1}^{N+1} c_j(t) B_j^{'}(x) \right). \end{aligned} \tag{3.1}$$

By using (2.4)-(2.5) in (3.1) at $x = x_j$, we have

$$\begin{aligned} U_t = &\left[f(c_{j-1} + 4c_j + c_{j+1}) \Big(\frac{3}{h}(c_{j+1} - c_{j-1})\Big) \right]_x \\ &+ \varphi\left(x_j,\, t,\, (c_{j-1} + 4c_j + c_{j+1}),\, \frac{3}{h}(c_{j+1} - c_{j-1}) \right), \end{aligned} \tag{3.2}$$

the time derivative is discretized in a forward finite difference scheme

$$(U_t)_j = \frac{U_j^{(n+1)} - U_j^{(n)}}{k},$$

where $U_j^{(n)} = U(x_j, t^{(n)})$ and $t^{(n)} = n\,k$, $n = 0, 1, \cdots$, where k is the time step $(k = t^{(n+1)} - t^{(n)})$. Then (3.2) becomes as

$$\begin{aligned} U_j^{(n+1)} - U_j^{(n)} = k\Bigg(&\Big[f\big(c_{j-1}^{(n)} + 4c_j^{(n)} + c_{j+1}^{(n)}\big)\,\big(\frac{3}{h}(c_{j+1}^{(n)} - c_{j-1}^{(n)})\big)\Big]_x \\ &+ \varphi\Big(x_j,\, t^{(n)},\, (c_{j-1}^{(n)} + 4c_j^{(n)} + c_{j+1}^{(n)}),\, \frac{3}{h}(c_{j+1}^{(n)} - c_{j-1}^{(n)})\Big)\Bigg). \end{aligned} \tag{3.3}$$

Introducing (2.4)-(2.6) into (3.3) yields

$$c_{j-1}^{(n+1)} + 4c_j^{(n+1)} + c_{j+1}^{(n+1)} = \psi_j^{(n)}, \tag{3.4}$$

where

$$\begin{aligned} \psi_j^{(n)} =&k\bigg(\Big[f(c_{j-1}^{(n)} + 4c_j^{(n)} + c_{j+1}^{(n)})\,(\frac{3}{h}(c_{j+1}^{(n)} - c_{j-1}^{(n)}))\Big]_x \\ &+ \varphi\Big(x_j,\, t^{(n)},\, (c_{j-1}^{(n)} + 4c_j^{(n)} + c_{j+1}^{(n)}),\, \frac{3}{h}(c_{j+1}^{(n)} - c_{j-1}^{(n)})\Big)\bigg) \\ &+ c_{j-1}^{(n)} + 4c_j^{(n)} + c_{j+1}^{(n)}, \qquad\qquad 0 \le j \le N. \end{aligned}$$

There for we have a system as follow

$$AC = \psi, \tag{3.5}$$

where

$$\mathbf{A} = \begin{pmatrix} 0 & \dots & 0 & 1 & 4 & 1 & 0 & \dots & 0 \\ 1 & 4 & 1 & & & & & & \\ & 1 & 4 & 1 & & & & & \\ & & \dots & \dots & \dots & & & & \\ & & & \dots & \dots & \dots & & & \\ & & & & \dots & \dots & \dots & & \\ & & & & & \dots & \dots & \dots & \\ \vdots & & & & & & 1 & 4 & 1 \\ 0 & \dots & & & & & 1 & 4 & 1 \end{pmatrix},$$

that $A[1, s+1] = 1$, $A[1, s+2] = 4$, $A[1, s+3] = 1$ by (2.8) and

$$\mathbf{C} = \begin{pmatrix} c_{-1}^{(n+1)} \\ c_0^{(n+1)} \\ c_1^{(n+1)} \\ \vdots \\ c_{N-1}^{(n+1)} \\ c_N^{(n+1)} \\ c_{N+1}^{(n+1)} \end{pmatrix}, \quad \psi = \begin{pmatrix} \psi_{-1}^{(n)} \\ \psi_0^{(n)} \\ \psi_1^{(n)} \\ \vdots \\ \psi_{N-1}^{(n)} \\ \psi_N^{(n)} \\ \psi_{N+1}^{(n)} \end{pmatrix},$$

where

$$\begin{aligned} \psi_{-1}^{(n)} =& r(t^{(n)}), \\ \psi_j^{(n)} =& k\Bigg(\Bigg[f(c_{j-1}^{(n)} + 4c_j^{(n)} + c_{j+1}^{(n)})\, (\frac{3}{h}(c_{j+1}^{(n)} - c_{j-1}^{(n)})) \Bigg]_x \\ &+ \varphi\Big(x_j,\, t^{(n)},\, (c_{j-1}^{(n)} + 4c_j^{(n)} + c_{j+1}^{(n)}),\, \frac{3}{h}(c_{j+1}^{(n)} - c_{j-1}^{(n)})\Big) \Bigg) \\ &+ c_{j-1}^{(n)} + 4c_j^{(n)} + c_{j+1}^{(n)}, \qquad\qquad 0 \le j \le N, \\ \psi_{N+1}^{(n)} =& q(t^{(n)}). \end{aligned}$$

Here A is a $(N+3) \times (N+3)$ matrix, ψ and C are $(N+3)$ order vectors, which depend on the boundary and overspesified conditions (1.4) and (1.6). With solving (3.5) by Tikhonov regularization method, the coefficients c_j are obtained and using these coefficients, we can obtain the approximate solution and finally

$$\begin{aligned} p(t^{(n)}) =& c_{-1}^{(n)} + 4c_0^{(n)} + c_1^{(n)}, \qquad n = 0,\, 1,\, ..., \\ U(x_j, t^{(n)}) =& c_{j-1}^{(n)} + 4c_j^{(n)} + c_{j+1}^{(n)}, \qquad n = 0,\, 1,\, ..., \qquad j = 0,\, 1,\, ...,\, N. \end{aligned}$$

4. The Initial Vector C^0

The initial vector C^0 can be obtained from the initial condition (1.5), boundary and overspesified conditions (1.4), (1.6) as the following expressions

$$\begin{aligned} u(x_s, 0) =& c_{s-1}^{(0)} + 4c_s^{(0)} + c_{s+1}^{(0)} = r(0), \\ \\ u(x_j, 0) =& c_{j-1}^{(0)} + 4c_j^{(0)} + c_{j+1}^{(0)} = u_0(x_j), \qquad 0 \le j \le N, \\ \\ u(x_N, 0) =& c_{N-1}^{(0)} + 4c_N^{(0)} + c_{N+1}^{(0)} = q(0). \end{aligned}$$

This yields a $(N+3) \times (N+3)$ system of equations, of the form

$$AC^0 = B, \tag{4.1}$$

where

$$\mathbf{A}=\begin{pmatrix} 0 & \dots & 0 & 1 & 4 & 1 & 0 & \dots & 0 \\ 1 & 4 & 1 & & & & & & \\ & 1 & 4 & 1 & & & & & \\ & & \dots & \dots & \dots & & & & \\ & & & \dots & \dots & \dots & & & \\ & & & & \dots & \dots & \dots & & \\ & & & & & \dots & \dots & \dots & \\ \vdots & & & & & & 1 & 4 & 1 \\ 0 & \dots & & & & & 1 & 4 & 1 \end{pmatrix},$$

$$\mathbf{C^0}=\begin{pmatrix} c_{-1}^{(0)} \\ c_{0}^{(0)} \\ c_{1}^{(0)} \\ \vdots \\ c_{N-1}^{(0)} \\ c_{N}^{(0)} \\ c_{N+1}^{(0)} \end{pmatrix}, \quad \mathbf{B}=\begin{pmatrix} r(0) \\ u_0(x_0) \\ u_0(x_1) \\ \vdots \\ u_0(x_{N-1}) \\ u_0(x_N) \\ q(0) \end{pmatrix},$$

that $A[1,s+1]=1$, $A[1,s+2]=4$, $A[1,s+3]=1$. The solution of (4.1) can be obtained by Tikhonov regularization method.

5. Stability of solution

Mathematically, inverse problems belong to the class of ill-posed problems. The matrix A is singular and ill-posed, thus the estimate of C^0 by (4.1) will be unstable so that the Tikhonov regularization method must be used to control this singularity. In our computations, we adapt the Tikhonov regularization method to solve the matrix system of equations (3.5) and (4.1). The Tikhonov regularized solutions to the systems of linear algebraic equations (3.5) and (4.1) are given by

$$F_\alpha(C)=\|A\,C-\psi\|_2^2+\alpha\|R^{(z)}\,C\,\|_2^2,$$

$$F_\alpha(C^0)=\|A\,C^0-B\|_2^2+\alpha\|R^{(z)}\,C^0\|_2^2.$$

On the case of the first-, second-Tikhonov regularization method the matrix $R^{(z)}$, for $z=1,\,2$, is given by, see e.g [20]

$$R^{(1)}=\begin{pmatrix} -1 & 1 & 0 & \dots & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & \dots & 0 & -1 & 1 \end{pmatrix}\in\mathbb{R}^{(M-1)\times(M)},$$

$$R^{(2)} = \begin{pmatrix} 1 & -2 & 1 & \dots & 0 & 0 & 0 \\ 0 & 1 & -2 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & -2 & 1 & 0 \\ 0 & 0 & 0 & \dots & 1 & -2 & 1 \end{pmatrix} \in \mathbb{R}^{(M-2)\times(M)},$$

where $M = N + 3$.

Definition 5.1. Let $A \in R^{m_1 \times n_1}$ and $B \in R^{m_2 \times n_1}$ with $m_1 \geq n_1$ and

$$r = rank\left(\begin{bmatrix} A \\ B \end{bmatrix} \right).$$

Then the generalized singular value decomposition of A and B are defined by

$$A = U\,\Gamma V^T, \qquad\qquad B = Y\,\Lambda V^T,$$

where $\Gamma \in R^{m_1 \times n_1}$, $\Lambda \in R^{m_2 \times n_1}$. Also, $U = (u_1, u_2, \dots, u_{m_1}) \in R^{m_1 \times m_1}$, $Y = (y_1, y_2, \dots, y_{m_2}) \in R^{m_2 \times m_2}$ are orthogonal and $V \in R^{n_1 \times n_1}$ is invertible such that

$$U^T\,A\,V = \Gamma = \begin{bmatrix} I_A & 0 & 0 \\ 0 & S_A & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$Y^T\,B\,V = \Lambda = \begin{bmatrix} 0 & 0 & 0 \\ 0 & S_B & 0 \\ 0 & 0 & I_B \end{bmatrix}.$$

Here, $I_A \in R^{p \times p}$, $I_B \in R^{(n_1-r)\times(n_1-r)}$. S_A and $S_B \in R^{(r-p)\times(r-p)}$ are defined by

$$\begin{aligned} S_A &= \text{diag}(\gamma_{p+1}, \gamma_{p+2}, \dots, \gamma_r), \\ S_B &= \text{diag}(\lambda_{p+1}, \lambda_{p+2}, \dots, \lambda_r), \end{aligned} \tag{5.1}$$

with $1 > \gamma_{p+1} \geq \gamma_{p+2} \geq \dots \geq \gamma_r > 0,\ 0 < \lambda_{p+1} \leq \lambda_{p+2} \leq \dots \leq \lambda_r < 1$, [10].

The matrices A and $R^{(z)}$ can be written as

$$A = U\,\Gamma V^T, \qquad\qquad R^{(z)} = Y\,\Lambda V^T.$$

The generalized singular values of A and $R^{(z)}$ are

$$\sigma_i = \frac{\gamma_i}{\lambda_i},$$

where

$$\gamma = \sqrt{diag(\Gamma^T\,\Gamma)}, \qquad\qquad \lambda = \sqrt{diag(\Lambda^T \Lambda)}.$$

Therefore the Tikhonov regularized solutions to the systems of linear algebraic equations (3.5) and (4.1) are given by

$$C_\alpha = [A^T A + \alpha (R^{(z)})^T R^{(z)}]^{-1} A^T \psi = \sum_{i=1}^{N+3} \frac{\gamma_i^2}{\gamma_i^2 + \alpha^2 \lambda_i^2} \frac{u_i^T(\psi)}{\gamma_i} v_i^{-T},$$

$$C_\alpha^0 = [A^T A + \alpha (R^{(z)})^T R^{(z)}]^{-1} A^T B = \sum_{i=1}^{N+3} \frac{\gamma_i^2}{\gamma_i^2 + \alpha^2 \lambda_i^2} \frac{u_i^T(B)}{\gamma_i} v_i^{-T}, \tag{5.2}$$

In our computation, we use the Generalized cross-validation (GCV) scheme to determine a suitable value of α ([1, 9, 13]).

6. Convergence Analysis

Let $u(x,t)$ be the exact solution of the problem (1.1) with the boundary condition, initial condition, overspesific condition and also $U(x,t) = \sum_{j=-1}^{N+1} c_j(t) B_j(x)$ be the B-spline collocation approximation to $u(x,t)$. Due to round off errors in computations we assume that $\hat{U}(x,t)$ be the computed spline for $U(x,t)$ so that $\hat{U}(x,t) = \sum_{j=-1}^{N+1} \hat{c}_j(t) B_j(x)$ where $\hat{C} = (\hat{c}_{-1}, \hat{c}_0, \ldots, \hat{c}_N, \hat{c}_{N+1},)$. To estimate the error $\|u(x,t) - U(x,t)\|_\infty$ we must estimate the errors $\|u(x,t) - \hat{U}(x,t)\|_\infty$ and $\|\hat{U}(x,t) - U(x,t)\|_\infty$ separately. Following (3.5) for $\hat{U}$ we have

$$A\hat{C} = \hat{\psi}, \tag{6.1}$$

where

$$\hat{\psi} = \Big(r(t), \hat{\psi}_0, \hat{\psi}_1, \ldots, \hat{\psi}_{N-1}, \hat{\psi}_N, q(t) \Big),$$

and

$$\begin{aligned} \hat{\psi}_j =& k \Bigg(\Big[f(\hat{c}_{j-1} + 4\hat{c}_j + \hat{c}_{j+1}) \, (\frac{3}{h}(\hat{c}_{j+1} - \hat{c}_{j-1})) \Big]_x \\ &+ \varphi\Big(x_j, \, t^{(n)}, \, (\hat{c}_{j-1} + 4\hat{c}_j + \hat{c}_{j+1}), \, \frac{3}{h}(\hat{c}_{j+1} - \hat{c}_{j-1}) \Big) \Bigg) \\ &+ \hat{c}_{j-1} + 4\hat{c}_j + \hat{c}_{j+1}, \qquad\qquad 0 \le j \le N. \end{aligned}$$

By subtracting (6.1) and (3.5), we have

$$A(C - \hat{C}) = (\psi - \hat{\psi}). \tag{6.2}$$

Now, first we need to recall some theorems.

Theorem 6.1. *Suppose $f \in C^4[a,b]$ and $|f^{(4)}(x)| \le L$ for $x \in [a,b]$. Let Δ be a partition $\Delta = \{a = x_0 < x_1 < \cdots < x_n = b\}$ of the interval $[a,b]$ with step size h. If S_Δ is the spline function which interpolates the values of the function f at the knots $x_0, \ldots, x_n \in \Delta$, then there exist constants $\lambda_j \le 2$, which do not depend on the partition Δ, such that for $x \in [a,b]$,*

$$\|f^{(j)}(x) - S_\Delta^{(j)}(x)\| \le \lambda_j \, L \, h^{4-j}, \qquad j = 0, 1, 2, 3, \tag{6.3}$$

where $\|.\|$ represents the ∞-norm.

Proof. For the proof see Stoer and Bulirsch [31]. □

Now, we find an upper bound for $\|\psi - \hat{\psi}\|_\infty$. For this, since

$$\begin{aligned}\Big|\psi(x_j) - \hat{\psi}(x_j)\Big| =& \Big|k\Big(\frac{\partial}{\partial x}\Big(\frac{\partial U}{\partial x}f(U) - \frac{\partial \hat{U}}{\partial x}f(\hat{U})\Big)\\&+\varphi\Big(x_j, U(x_j), U^{'}(x_j)\Big)\\&-\varphi\Big(x_j, \hat{U}(x_j), \hat{U}^{'}(x_j)\Big)\Big) + U(x_j) - \hat{U}(x_j)\Big|,\end{aligned}$$

by using the Cauchy-Schwarz inequality, we have

$$\begin{aligned}\Big|\psi(x_j) - \hat{\psi}(x_j)\Big| \leq& k\Big|\frac{\partial}{\partial x}\Big(\frac{\partial U}{\partial x}f(U) - \frac{\partial \hat{U}}{\partial x}f(\hat{U})\Big)\Big|\\&+k\Big|\varphi\Big(x_j, U(x_j), U^{'}(x_j)\Big) - \varphi\Big(x_j, \hat{U}(x_j), \hat{U}^{'}(x_j)\Big)\Big|\\&+\Big|U(x_j) - \hat{U}(x_j)\Big|\\=& k\Big|\frac{\partial}{\partial x}\Big(\frac{\partial U}{\partial x}f(U) - \frac{\partial U}{\partial x}f(\hat{U}) + \frac{\partial U}{\partial x}f(\hat{U}) - \frac{\partial \hat{U}}{\partial x}f(\hat{U})\Big)\Big|\\&+k\Big|\varphi(x_j, U(x_j), U^{'}(x_j)) - \varphi(x_j, \hat{U}(x_j), \hat{U}^{'}(x_j))\Big|\\&+\Big|U(x_j) - \hat{U}(x_j)\Big|\\\leq& k\Big|\frac{\partial}{\partial x}\Big(\frac{\partial U}{\partial x}(f(U) - f(\hat{U}))\Big)\Big|\\&+k\Big|\frac{\partial}{\partial x}(f(\hat{U}))\Big(\frac{\partial}{\partial x}(U - \hat{U})\Big)\Big|\\&+k\Big|\varphi(x_j, U(x_j), U^{'}(x_j)) - \varphi(x_j, \hat{U}(x_j), \hat{U}^{'}(x_j))\Big|\\&+\Big|U(x_j) - \hat{U}(x_j)\Big|\\=& k\Big|\frac{\partial}{\partial x}(f(U) - f(\hat{U}))\frac{\partial U}{\partial x} + \frac{\partial^2 U}{\partial x^2}(f(U) - f(\hat{U}))\Big|\\&+k\Big|\frac{\partial}{\partial x}(f(\hat{U}))(\frac{\partial}{\partial x}(U - \hat{U})) + f(\hat{U})\frac{\partial^2}{\partial x^2}(U - \hat{U})\Big|\\&+k\Big|\varphi(x_j, U(x_j), U^{'}(x_j)) - \varphi(x_j, \hat{U}(x_j), \hat{U}^{'}(x_j))\Big|\\&+\Big|U(x_j) - \hat{U}(x_j)\Big|\end{aligned}$$

$$
\begin{aligned}
&\le k\left|\frac{\partial U}{\partial x}\right|\left|\frac{\partial}{\partial x}\left(f(U)-f(\hat{U})\right)\right| + k\left|\frac{\partial^2 U}{\partial x^2}\right|\left|f(U)-f(\hat{U})\right| \\
&\quad + k\left|\frac{\partial}{\partial x}\left(f(\hat{U})\right)\right|\left|\frac{\partial}{\partial x}(U-\hat{U})\right| + k\left|f(\hat{U})\right|\left|\frac{\partial^2}{\partial x^2}(U-\hat{U})\right| \\
&\quad + k\left|\varphi\big(x_j, U(x_j), U^{'}(x_j)\big) - \varphi\big(x_j, \hat{U}(x_j), \hat{U}^{'}(x_j)\big)\right| \\
&\quad + \left|U(x_j)-\hat{U}(x_j)\right|.
\end{aligned}
$$

Now, by considering theorems (6.1) and [27, theorem 9.19], we obtain

$$
\begin{aligned}
\|\psi-\hat{\psi}\|_\infty \le & C\,k\left(\lambda_0\,L\,h^4\;+2\,\lambda_1\,L\,h^3\;+\lambda_2\,L\,h^2\right) \\
& + M\,k\left(\lambda_0\,L\,h^4+\lambda_1^2\,L\,h^3\right)\;+\;\lambda_0\,L\,h^4,
\end{aligned}
\tag{6.4}
$$

where $\|\varphi'\|_\infty \le M$. Thus we can rewrite (6.4) as follows

$$
\|\psi-\hat{\psi}\|_\infty \le M_1\,h^2, \tag{6.5}
$$

where $M_1 = C\,k\,(\lambda_0\,L\,h^2\;+2\,\lambda_1\,L\,h+\lambda_2\,L\,)+M\,k\,(\lambda_0\,L\,h^2+\lambda_1^2\,L\,h)\,+\,\lambda_0\,L\,h^2$. Hence, M_1h^2 is an upper bound for $\|\psi-\hat{\psi}\|_\infty$.

It is obvious that the matrix A in (6.2) is an ill-posed matrix, thus by Tikhonov regularization from (5.2) we have

$$
(C-\hat{C}) = [A^T\,A+\alpha(R^{(z)})^T\,R^{(z)}]^{-1}A^T\,(\psi-\hat{\psi}). \tag{6.6}
$$

Taking the infinity norm and then by using (6.5) we find

$$
\|C-\hat{C}\|_\infty \le \|(A^T\,A+\alpha(R^{(z)})^T\,R^{(z)})^{-1}A^T\|_\infty\,\|\psi-\hat{\psi}\|_\infty \le M_2h^2, \tag{6.7}
$$

where $M_2 = \|(A^T\,A+\alpha(R^{(z)})^T\,R^{(z)})^{-1}A^T\|_\infty\,M_1$. Now we will be able to prove the convergence of our present method.

Lemma 6.2. *The B-splines $\{B_{-1},\,B_0,\,\cdots,\,B_{N+1}\}$, are satisfies the following inequality*

$$
\Big|\sum_{i=-1}^{N+1} B_i(x)\Big| \le 10, \qquad 0\le x\le 1. \tag{6.8}
$$

Proof. For proof see [17]. □

Now, observe that we have

$$
U(x)-\hat{U}(x) = \sum_{i=-1}^{N+1}(c_i-\hat{c}_i)B_i(x),
$$

thus taking the infinity norm and using (6.7) and (6.8) we get

$$\begin{aligned}\|U(x)-\hat{U}(x)\|_\infty &= \|\sum_{i=-1}^{N+1}(c_i-\hat{c}_i)B_i(x)\|_\infty \\ &\le \|c_i-\hat{c}_i\|_\infty|\sum_{i=-1}^{N+1}B_i(x)| \le 10M_2\,h^2. \end{aligned} \tag{6.9}$$

Theorem 6.3. *Let $u(x)$ be the exact solution of the equation (1.1) with the boundary condition (1.4) and initial condition (1.5) and overspesific condition (1.6) and also $U(x)$ be the B-spline collocation approximation to $u(x)$ then the method has second order convergence*

$$\|u(x)-U(x)\| \le \omega\,h^2,$$

where $\omega = \lambda_0\,L\,h^2 + 10\,M_2$ is some finite constant.

Proof. From theorem (6.1) we have

$$\|u(x)-\hat{U}(x)\| \le \lambda_0\,L\,h^4. \tag{6.10}$$

Thus substituting from (6.9) and (6.10) we have

$$\begin{aligned}\|u(x)-U(x)\| &\le \|u(x)-\hat{U}(x)\| + \|U(x)-\hat{U}(x)\| \\ &\le \lambda_0\,L\,h^4 + 10\,M_2\,h^2 = \omega\,h^2,\end{aligned}$$

where $\omega = \lambda_0\,L\,h^2 + 10\,M_2$. □

Theorem 6.4. *The time discretization process (3.2) that we use to discretize equation (1.1) in time variable is of the one order convergence.*

Proof. See [30]. □

We suppose that $u(x,t)$ be the solution of equation (1.1) and $U(x,t)$ be the approximate solution by our present method then we have

$$\|u(x,t^n)-U(x,t^n)\| \le \Gamma(k+h^2),$$

(Γ is some finite constant), thus the order of convergence of our process is $O(k+h^2)$.

7. Numerical Results and Discussion

In this section, we are going to study numerically the inverse problems (1.1)-(1.6) with the unknown boundary condition. The main aim here is to show the applicability of the present method for solving the inverse problems (1.1)-(1.5). As expected the inverse problems are ill-posed and therefore it is necessary to investigate the stability of the present method by giving a test problem. Thus we compute L_2 error norm, by using following formula

$$L_2 = \sqrt{\frac{1}{n-1}\,(\sum_{i=1}^{n}|(u_{exact})_i-(U_{num})_i|^2)}\,,$$

where $u_i = u(x_j,\,t_i)$ and n is the total number of estimated values.

Example 7.1. In this example let us consider the following inverse problem

$$u_t = \Big[\frac{u}{100}\exp(u)\,u_x\Big]_x + \big(\frac{1}{10}\exp(u)\,+\,1\big)\,u_x, \quad 0 \le x \le 1, \quad 0 \le t \le 1,$$

with given data

$$u(x,\,0) = \ln(x+1),$$

$$u(1,\,t) = \ln(\frac{1}{-0.1\,t+1} + \frac{t+1}{-0.1\,t+1} - \frac{0.01\ln(-0.1\,t+1)}{-0.1\,t+1}),$$

$$u(0.5,\,t) = \ln(\frac{0.5}{-0.1\,t+1} + \frac{t+1}{-0.1\,t+1} - \frac{0.01\ln(-0.1\,t+1)}{-0.1\,t+1}).$$

The exact solution of this problem is

$$u(x,t) = \ln(\frac{x}{-0.1\,t+1} + \frac{t+1}{-0.1\,t+1} - \frac{0.01\ln(-0.1\,t+1)}{-0.1\,t+1}).$$

The results obtained for $u(0,t) = p(t)$ and $u(0.8,t)$ with $k = 0.001$, $h = 0.1$ and $\beta = 0.5$ with noisy data (noisy data=input data+(0.0001) rand(1)) are presented in Table 1 and Figures 1, 2.

Table 1. The comparison between exact solution and numerical solution for $p(t)$ with the noisy data by using cubic B-spline method and Tikhonov^{2nd} when $\beta = 0.5$ for Example 7.1.

	$p(t)$			$u(0.8,t)$		
t	*Exact*	*Numerical*	*Error*	*Exact*	*Numerical*	*Error*
0.1	0.105452	0.105870	0.000418	0.651329	0.652989	0.001032
0.2	0.202693	0.204502	0.001809	0.712848	0.715174	0.001723
0.3	0.293058	0.295909	0.002851	0.771962	0.774256	0.001714
0.4	0.377586	0.381223	0.003637	0.828906	0.830686	0.001221
0.5	0.457100	0.460759	0.003659	0.883885	0.885054	0.000628
0.6	0.532266	0.535159	0.002892	0.937078	0.937477	0.000124
0.7	0.603626	0.605826	0.002199	0.988644	0.988413	0.000738
0.8	0.671631	0.673657	0.002025	1.038720	1.038681	0.000532
0.9	0.736661	0.738828	0.002166	1.087431	1.088095	0.000183
1	0.799034	0.801513	0.002478	1.134887	1.136035	0.000679
L_2			0.002514			0.000997
Execution Time (second)						3.416
Regularization Parameter (α)						0.049759

FIGURE 1. The plots of approximate solution, exact solution and absolute error of $p(t)$ for Example 7.1 with the noisy data.

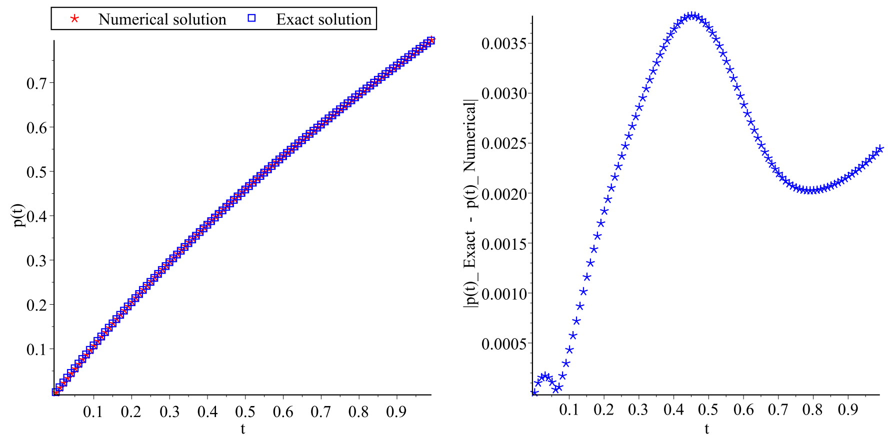
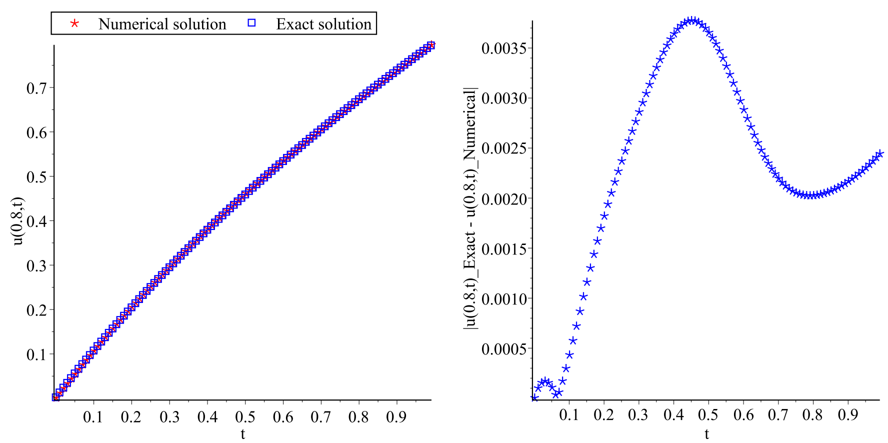


FIGURE 2. The plots of approximate solution, exact solution and absolute error of $u(0.8,t)$ for Example 7.1 with the noisy data.



Example 7.2. In this example, consider the following inverse problem:

$$u_t = 2[cosh^2(3u)\, u_x]_x, \qquad 0 \leq x \leq 1, \qquad 0 \leq t \leq 1,$$

with given data

$$u(x,\,0) = \frac{1}{3} arcsinh(\frac{x+0.25}{2\sqrt{2}}),$$

$$u(1,\,t) = \frac{1}{3} arcsinh(\frac{1.25}{2\sqrt{2-t}}),$$

$$u(0.5,\,t) = \frac{1}{3} arcsinh(\frac{0.75}{2\sqrt{2-t}}).$$

The exact solutions in a closed form are given by

$$u(x,t) = \frac{1}{3} arcsinh(\frac{x+0.25}{2\sqrt{2-t}}).$$

For numerical computation, we take with $\beta = 0.5$, $k = 0.001$ and $h = 0.1$ for estimate $u(0,t) = p(t)$ and $u(0.8,t)$ with noisy data (noisy data=input data+(0.0001) rand(1)) and results are reported in Table 2 and Figures 3, 4.

TABLE 2. The comparison between exact solution and numerical solution for $p(t)$ with the noisy data by using cubic B-spline method and Tikhonov^{2nd} when $\beta = 0.5$ for Example 7.2.

	$p(t)$			$u(0.8,t)$		
t	$Exact$	$Numerical$	$Error$	$Exact$	$Numerical$	$Error$
0.1	0.030186	0.032651	0.002464	0.124042	0.123961	0.000112
0.2	0.031011	0.034422	0.003411	0.127285	0.127061	0.000258
0.3	0.031908	0.035575	0.003667	0.130797	0.130553	0.000281
0.4	0.032887	0.036842	0.003955	0.134617	0.134350	0.000307
0.5	0.033961	0.038248	0.004286	0.138794	0.138501	0.000336
0.6	0.035149	0.039820	0.004670	0.143386	0.143064	0.000370
0.7	0.036471	0.041591	0.005120	0.148468	0.148112	0.000409
0.8	0.037954	0.043607	0.005653	0.154134	0.153739	0.000454
0.9	0.039634	0.045926	0.006292	0.160505	0.160065	0.000507
1	0.041558	0.048628	0.007069	0.167743	0.167249	0.000570
L_2			0.004577			0.0003608
Execution Time (second)						5.632
Regularization Parameter (α)						36.6301

FIGURE 3. The plots of approximate solution, exact solution and absolute error of $p(t)$ for Example 7.2 with the noisy data.

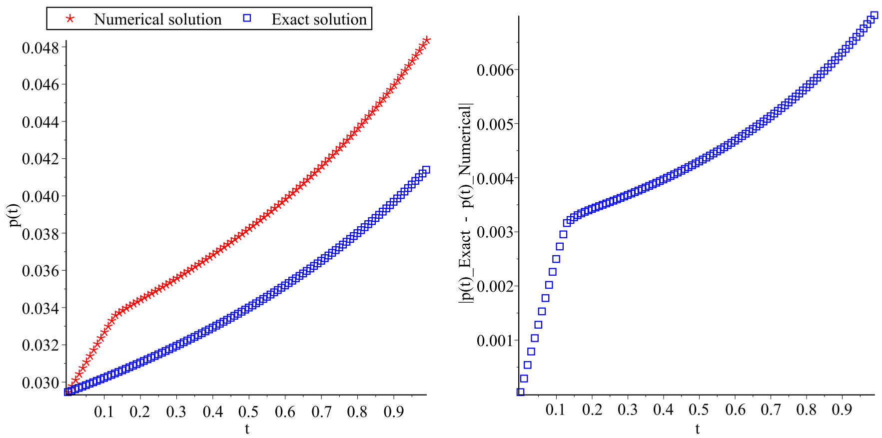
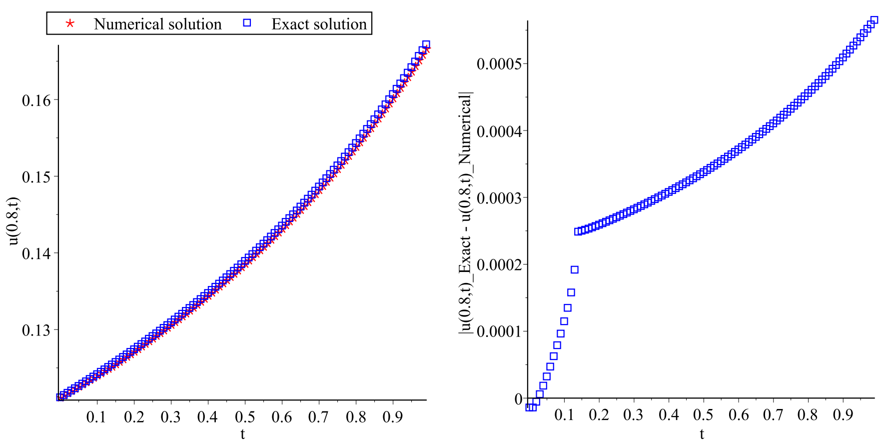


FIGURE 4. The plots of approximate solution, exact solution and absolute error of $u(0.8, t)$ for Example 7.2 with the noisy data.



Example 7.3. We consider the following inverse problem

$$u_t = 2[u^4 u_x]_x + (2u^4 - 2)\, u_x, \qquad 0 \le x \le 1, \qquad 0 \le t \le 1,$$

with given data

$$u(x,\,0) = \,(2-0.5x)^{\frac{1}{4}},$$

$$u(1,\,t) = \,(\frac{0.5}{t+1}+\frac{1}{8}\frac{\ln(t+1)}{(t+1)}+1)^{\frac{1}{4}},$$

$$u(0.5,\,t) = (\frac{1.5}{2t+2}+\frac{1}{8}\frac{\ln(t+1)}{(t+1)}+1)^{\frac{1}{4}}.$$

The exact solution of this problem is

$$u(x,t) = (\frac{2-x}{2t+2}+\frac{1}{8}\frac{\ln(t+1)}{(t+1)}+1)^{\frac{1}{4}},$$

For numerical computation, we take with $\beta = 0.5$, $k = 0.001$ and $h = 0.1$ for estimate $u(0,t) = p(t)$ and $u(0.8,t)$ with noisy data and results are reported in Table 3 and Figures 5, 6.

TABLE 3. The comparison between exact solution and numerical solution for $p(t)$ with the noisy data by using cubic B-spline method and Tikhonov^{2nd} when $\beta = 0.5$ for Example 7.3.

	$p(t)$			$u(0.8,t)$		
t	*Exact*	*Numerical*	*Error*	*Exact*	*Numerical*	*Error*
0.1	1.177120	1.182728	0.005607	1.116992	1.116657	0.000263
0.2	1.166619	1.174228	0.007609	1.110231	1.109882	0.000285
0.3	1.157399	1.166008	0.008609	1.104288	1.103936	0.000296
0.4	1.149230	1.158133	0.008902	1.099017	1.098672	0.000295
0.5	1.141934	1.150739	0.008804	1.094306	1.093974	0.000287
0.6	1.135375	1.143881	0.008506	1.090066	1.089752	0.000274
0.7	1.129441	1.137560	0.008119	1.086228	1.085932	0.000259
0.8	1.124044	1.131749	0.007704	1.082734	1.082457	0.000243
0.9	1.119113	1.126409	0.007296	1.07954	1.079280	0.000228
1	1.114587	1.121497	0.006910	1.076604	1.076362	0.000213
L_2			0.007731			0.000263
Execution Time (second)						7.004
Regularization Parameter (α)						8.1825

Figure 5. The plots of approximate solution, exact solution and absolute error of $p(t)$ for Example 7.3 with the noisy data.

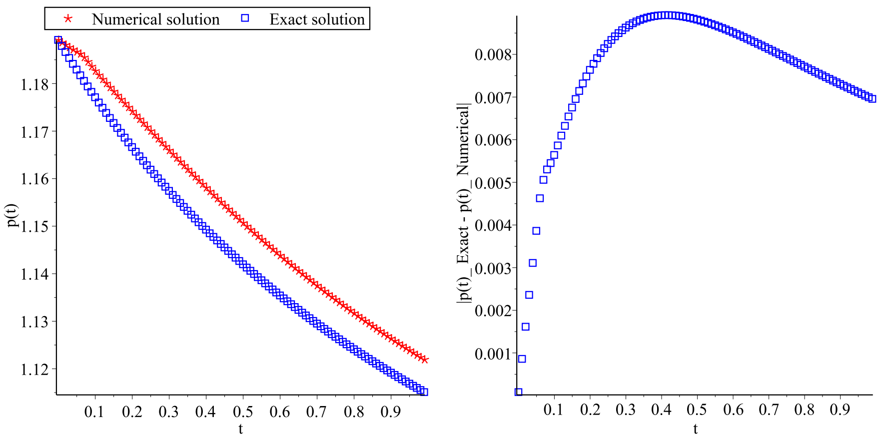
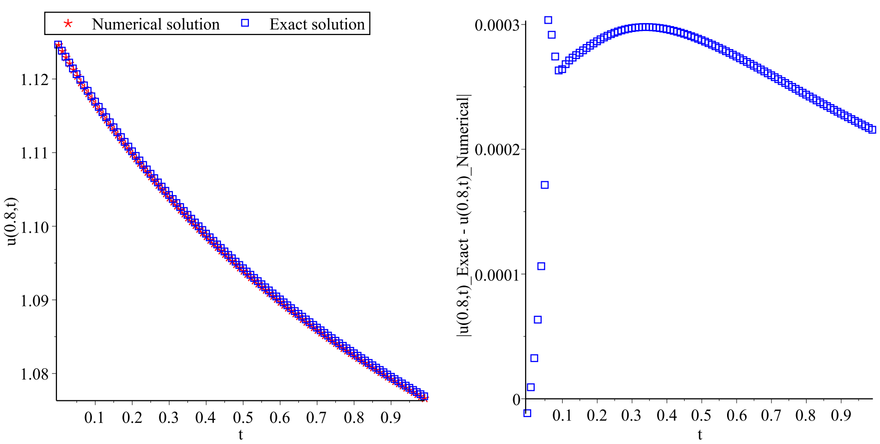


Figure 6. The plots of approximate solution, exact solution and absolute error of $u(0.8,t)$ for Example 7.3 with the noisy data.



8. Conclusion

A numerical method, to estimate unknown boundary conditions is proposed and the following results are obtained.

- The present study successfully applies the numerical method to inverse problems.
- Unlike some previous techniques using various transformations to reduce the equation in to more simple equation, the current method does not require

extra effort to deal with the nonlinear terms. Therefore, the equations are solved easily and elegantly using the present method.

- Numerical examples also verified the efficiency and accuracy of the method that can be obtained within a couple of minutes CPU time at Core($i5$)–2.67 GHz PC.
- The present method has been found stable with respect to small perturbation in the input data.

References

[1] R. C. Aster, B. Borchers, and C. Thurber, *Parameter estimation and inverse problems.* Textbook, New Mexico, Tech., 2003.

[2] F. Auricchio, L. Beirao da Veiga, T. Hughes, A. Reali, and G. Sangalli, *Isogeometric collocation methods,* Mathematical Models and Methods in Applied Sciences, *20*(11) (2010), 2075–2107.

[3] J. R. Cannon, *Determination of an unknown heat source from overspecified boundary data,* SIAM J. Numer. Anal., *5*(2) (1968), 275–286.

[4] J. R. Cannon, Y. Lin, and S. Wang, *Determination of a control parameter in a parabolic partial differential equation,* J. Aust. Math. Soc. Ser. B, *33* (1991), 149–163.

[5] M. Dehghan, *An inverse problem of finding a source parameter in a semilinear parabolic equation,* Appl. Math. Model., *25* (2001), 743–754.

[6] M. Dehghan, *parameter determination in partial differential equation from the overspecified data,* Math. Comput. Model., *41*(23) (2005), 196–213.

[7] M. Dehghan and M. Lakestani, *The Use of cubic B-spline scaling functions for solving the one-dimensional hyperbolic equation with a nonlocal conservation condition,* Numer. Methods Partial Differential Eq., *23*(6) (2007), 1277–1289.

[8] M. Dehghan, S. A. Yousefi, and K. Rashdi, *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 Prob. Sci. Eng., *21* (2013), 500–523.

[9] L. Elden, *A note on the computation of the generalized cross-validation function for ill-conditioned least squares problems,* BIT, *24* (1984), 467–472.

[10] K. Eric Chu, *Singular Value and Generalized Singular Value Decompositions and the Solution of Linear Matrix Equations,* Linear Algebra and its Applications, *88* (1987), 83–98.

[11] A. G. Fatullayev and S. Cula, *An iterative procedure for determining an unknown spacewise-dependent coeifficient in a parabolic equation,* Appl. Math. Lett., *22* (2009), 1033–1037.

[12] S. Foadian, R. Pourgholi, and S. H. Tabasi, *Cubic B-spline method for the solution of an inverse parabolic system,* Applicable Analysis, *97*(3) (2018), 438–465.

[13] G. H. Golub, M. Heath, and G. Wahba, *Generalized cross-validation as a method for choosing a good ridge parameter,* Technometrics, *21*(2) (1979), 215–223.

[14] H. Gomez and L. De Lorenzis, *The variational collocation method,* Computer Methods in Applied Mechanics and Engineering, *309* (2016), 152–181.

[15] Y. C. Hon and T. Wei, *A fundamental solution method for inverse heat conduction problem,* Eng. Anal. Bound. Elem., *28* (2004), 489–495.

[16] V. Isakov, *Inverse Problems for Partial Differential Equations,* Springer, New York, 1998.

[17] M. K. Kadalbajoo, V. Gupta, and A. Awasthi, *A uniformly convergent b-spline collocation method on a nonuniform mesh for singularly perturbed one-dimensional time-dependent linear convection diffusion problem,* Journal of Computational and Applied Mathematics, *220* (2008), 271–289.

[18] K. A. Khalid, K. R. Raslan, and T. S. El-Danaf, *Non-polynomial Spline Method for Solving Coupled Burgers Equations,* Comput. Methods Differ. Equ., *3*(3) (2015), 218–230.

[19] M. Lakestani and M. Dehghan, *Numerical solution of Fokker–Planck equation using the cubic B-spline scaling functions,* Numer Methods Partial Differential Eq., *25*(2) (2009), 418–429.

[20] L. Martin, L. Elliott, P. J. Heggs, D. B. Ingham, D. Lesnic, and X. Wen, *Dual reciprocity boundary element method solution of the Cauchy problem for Helmholtz-type equations with variable coefficients,* J. Sound Vib., *297* (2006), 89–105.

[21] M. Montardini, G. Sangalli, and L. Tamellini, *Optimal-order isogeometric collocation at Galerkin superconvergent points,* Computer Methods in Applied Mechanics and Engineering, *316* (2017), 741–757

[22] R. Pourgholi, N. Azizi, Y. S. Gasimov, F. Aliev, and H. K. Khalafi, *Removal of numerical instability in the solution of an inverse heat conduction problem,* Commun. Nonlinear Sci. Numer. Simul., *14*(6) (2009), 2664–2669.

[23] R. Pourgholi, H. Dana, and S. H. Tabasi, *Solving an inverse heat conduction problem using genetic algorithm: sequential and multi-core parallelization approach,* Appl. Math. Modelling, *38*(7) (2014), 1948–1958.

[24] R. Pourgholi and A. Saeedi, *Applications of cubic B-splines collocation method for solving non-linear inverse parabolic partial differential equations,* Numerical Methods for Partial Differential Equations, *34*(8) (2016), 1–17.

[25] A. G. Ramm, *An inverse problem for the heat equation,* J. Math. Anal. Appl., *264* (2004), 691–697.

[26] A. G. Ramm, *Inverse problems,* Springer, New York, 2005.

[27] W. Rudin, *Principles of mathematical analysis,* McGraw-Hill Inc., Third Edition, 1976.

[28] D. Schillinger, J. A. Evans, A. Reali, M. Scott, and T. Hughes, *Isogeometric collocation: cost comparison with Galerkin methods and extension to adaptive hierarchical NURBS discretizations,* Computer Methods in Applied Mechanics and Engineering, *267* (2013), 170–232.

[29] A. Shidfar, G. R. Karamali, and J. Damirchi, *An inverse heat conduction problem with a nonlinear source term,* Nonlinear Anal., *65* (2006), 615–621.

[30] G. D. Smith, *Numerical solution of partial differential equation: finite difference method,* Learendom Press, Oxford, 1978.

[31] J. Stoer and R. Bulrisch, *An introduction to numerical analysis,* Springer-Verlag, 1991.