Journal of Computational and Applied Mathematics 423 (2023) 114958

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

A direct parallel-in-time quasi-boundary value method for inverse space-dependent source problems

Yi Jiang [a], Jun Liu [a,*], Xiang-Sheng Wang [b]

[a] Department of Mathematics and Statistics, Southern Illinois University Edwardsville, Edwardsville, IL 62026, USA
[b] Department of Mathematics, University of Louisiana at Lafayette, Lafayette, LA 70503, USA

ARTICLE INFO

Article history:
Received 22 February 2022
Received in revised form 4 November 2022

Keywords:
Ill-posed
Inverse source problem
Quasi-boundary value method
Diagonalization
Parallel-in-time
Condition number

ABSTRACT

Inverse source problems arise often in real-world applications, such as localizing unknown groundwater contaminant sources. Being different from Tikhonov regularization, the quasi-boundary value method has been studied as an effective way for regularizing such inverse source problems, which was shown to achieve an optimal order convergence under suitable assumptions. However, fast direct or iterative solvers for the resulting large-scale all-at-once linear systems have been rarely studied in the literature. In this work, we propose and analyze a modified quasi-boundary value method that leads to a fast diagonalization-based parallel-in-time (PinT) direct solver, which can achieve a dramatic speedup in CPU times when compared with MATLAB's sparse direct solver. In particular, the time-discretization matrix B is shown to be diagonalizable, and the condition number of its eigenvector matrix V is proven to exhibit only quadratic growth, which guarantees that the roundoff errors due to diagonalization is well-conditioned. Several 1D and 2D examples are presented to demonstrate the very promising computational efficiency of our proposed method, where the CPU times in 2D cases can be speeded up by three orders of magnitude.

1. Introduction

Let $T > 0$ and $\Omega \subset \mathbb{R}^d (d = 1, 2, 3)$ be an open and bounded domain with a piecewise smooth boundary $\partial\Omega$. We consider the inverse source problem (ISP) [1–3] of reconstructing the unknown space-dependent source term $f \in L^2(\Omega)$ from the final condition $g = u(\cdot, T) \in H_0^1(\Omega)$, according to a heat equation

$$\begin{cases} u_t - \Delta u = f, & \text{in } \Omega \times (0, T), \\ u(\cdot, t) = 0, & \text{on } \partial\Omega \times (0, T), \\ u(\cdot, 0) = \phi, & \text{in } \Omega, \\ u(\cdot, T) = g, & \text{in } \Omega, \end{cases} \tag{1}$$

where $\phi \in H_0^1(\Omega)$ is a given initial condition. In practice, the final condition g is unknown and it is only available as a noisy measurement $g_\delta \in L^2(\Omega)$ satisfying $\|g - g_\delta\|_2 \le \delta$ for a noise level $\delta > 0$. This leads to an ill-posed inverse problem that requires effective numerical regularization techniques [4–7].

Much research has been dedicated to the inverse source problem since 1970s, where the desired source term is usually assumed to have *a priori* form. For f that depends on the state function u, the problem was investigated in [2,8,9]. For f that

* Corresponding author.
E-mail addresses: yjianaa@siue.edu (Y. Jiang), juliu@siue.edu (J. Liu), xswang@louisiana.edu (X.-S. Wang).

https://doi.org/10.1016/j.cam.2022.114958
0377-0427/

depends on space or time variable only, many regularization methods have been developed such as Fourier method [10], quasi-reversibility method [3], quasi-boundary value method [11] and simplified Tikhonov regularization method [12]. In particular, in [3,11,12], the original problem is perturbed by a regularization parameter and the unknown source term is expressed in the form of a series expansion tailored by a regularizing filter. Following appropriate convergence analysis, the regularization parameter in these work is determined to balance the approximation accuracy and the stability of the regularized problem. The Fourier method in [10] solves the problem in the frequency domain and alleviates the ill-posedness of the problem by cutting off the high frequency components in the source solution, where the cut-off frequency is also chosen based on a convergence analysis. Such idea that truncates the terms contributing to the ill-posedness is also seen in [13]. In this work, a finite difference method is used to solve the inverse problem and the resulting linear system is solved by the singular value decomposition, where the small singular values are filtered out based on generalized cross-validation criterion [14]. There are some other numerical methods adopted in the research on inverse source problems, usually in conjunction with a classical regularization technique like Tikhonov method. For example, the boundary element method [15], the method of fundamental solutions [16–18] and the finite element method [19]. Some iterative algorithms can be found in [20–23]. For f that is a function of both time and space variables but is additive or separable, we refer to [24–27]. All of these works are mainly focused on the convergence analysis of various regularization techniques, without addressing the practical implementation issue of how to efficiently solve the regularized systems in large-scale setting.

On the other hand, for solving direct (or forward) evolutionary PDEs, many efficient parallelizable numerical algorithms have been developed in the last few decades due to the advent of massively parallel computers. Beyond the achieved high parallelism in space, a lot of recent advances in various parallel-in-time (PinT) algorithms for solving forward time-dependent PDE problems were reviewed in [28]. However, the application of such PinT algorithms to ill-posed inverse PDE problems were rarely investigated, except in a short paper [29] about the *parareal* algorithm for a different parabolic inverse problem and another earlier paper [30] based on numerical (inverse) Laplace transform techniques in time. Our proposed fast diagonalization-based PinT direct solver in current work contributes to narrowing in the gap between two somewhat different communities. One obvious difficulty is how to address the underlying regularization treatment in the framework of PinT algorithms, which seems to be highly dependent on the regularized problem structure and discretization schemes. Inspired by several recent works [31–37] on diagonalization-based PinT algorithms, we propose to redesign the existing quasi-boundary value methods in a structured manner such that the diagonalization-based PinT direct solver can be successfully employed. Such a PinT direct solver can greatly speed up the quasi-boundary value methods while achieving a comparable reconstruction accuracy. Recently, such an interesting approach of integrating PinT direct solver with quasi-boundary method regularization was applied to backward heat conduction problems [38–40], where a block ω-circulant structure was exploited for developing a fast FFT-based direct PinT solver [41].

Besides the above mentioned ISPs for PDEs based on ordinary *integer-order* derivatives, there are also several recent works on solving ISPs in the framework of time-fractional PDEs, to name just a few [42–50]. Again, the majority of these contributions focuses on the convergence properties of the proposed regularization techniques without discussing their fast algorithms. For solving related time-fractional diffusion inverse source problems, the authors in [51] proposed a fast structured preconditioner based on approximate Schur complement and block ω-circulant matrix. However, as an iterative solver, the underlying convergence analysis for preconditioned GMRES in [51] is a daunting task (due to nonsymmetric systems) and the Schur complement-based preconditioner cannot be easily parallelized. Our proposed direct PinT solver does not have such limitations for its practical use. Due to different discretization schemes in time, our proposed direct PinT solver may not be directly applicable to such time-fractional ISPs. The generalization of our method to such time-fractional ISPs is interesting but is beyond our current scope.

In this paper we design and analyze a new parameterized quasi-boundary value method (PQBVM) for regularizing the ISPs, where a well-conditioned diagonalization-based PinT direct solver is developed for its efficient numerical implementation. The major goal is to improve the overall computational efficiency in terms of CPU times, without obviously degrading the convergence rates in comparison with existing methods. As major theoretical contributions, the condition number of the diagonalization of the time discretization matrix is rigorously estimated and the convergence rate of the new PQBVM is also shown with suitable a priori choice of the regularization parameter. For 2D problems with a small 64^3 mesh (see results in Table 4), our direct PinT solver can drastically speed up the CPU times of the standard QBVM based on sparse direct solver from over 2 mins to about 0.04 s (on a desktop PC).

The rest of this paper is organized as follows. In Section 2, we propose a new parameterized QBVM based on full finite difference discretization and present a diagonalization-based direct PinT solver based on the derived Kronecker product system structure. In Section 3, we derive a sufficient condition to assure that the time discretization matrix B can be diagonalized into $B = VDV^{-1}$ and, more importantly, we further rigorously estimate the growth rate of the 1-norm condition number of its eigenvector matrix V with a special choice of the free design parameter. The convergence analysis with suitable choices of the regularization parameter is given in Section 4. Several numerical examples are presented to illustrate the high efficiency of the proposed algorithm in Section 5. Finally, some conclusions are made in Section 6.

2. A new quasi-boundary value method and its PinT implementation

The QBVM in [11] for regularizing (1) solves the following well-posed regularized problem

$$\begin{cases} u_t - \Delta u = f, & \text{in } \Omega \times (0,T), \\ u(\cdot,t) = 0, & \text{on } \partial\Omega \times (0,T), \\ u(\cdot,0) = \phi, & \text{in } \Omega, \\ u(\cdot,T) + \beta f(\cdot) = g_\delta, & \text{in } \Omega, \end{cases} \tag{2}$$

where $\beta > 0$ is a regularization parameter to be chosen based on the noise level $\delta > 0$. Compared with the Tikhonov regularization [12] of minimizing a regularized functional $\|Kf - g_\delta\|_2^2 + \gamma\|f\|_2^2$ with K being a compact solution operator and γ being a regularization parameter, the QBVM provides a better control of the discretized system structure. In particular, the QBVM does not need to explicitly construct K (and its adjoint K^*) or use any eigenfunctions of the spatial differential operator that are unknown in general cases.

We consider a 2D square domain $\Omega = (0,L)^2$ with finite difference discretization in space and time, which can be easily adapted to 1D and 3D regular spatial domains. For an irregular spatial domain, finite element discretization can also be used in space. Let $\tau = T/n$ and $h = L/(M+1)$ for two given positive integers n and M, we partition the time interval $[0,T]$ uniformly into $0 = t_0 < t_1 < \cdots < t_n = T$ with $t_j = j\tau$, and discretize the 2D space domain $\overline{\Omega} = [0,L]^2$ into a uniform mesh of grid nodes $\overline{\Omega}_h = \{(x_i = ih, y_l = lh) | 0 \le i \le (M+1), 0 \le l \le (M+1)\}$. Denote all the interior grid nodes by $\Omega_h = \{(x_i, y_l) | 1 \le i \le M, 1 \le l \le M\}$. For any function $u(x,y,t)$, let u_{il}^j be the finite difference approximation of $u(x_i, y_l, t_j)$. Define the concatenated column vector $u^j = [u_{11}^j, u_{21}^j, \ldots, u_{M1}^j, \ldots, u_{1M}^j, u_{2M}^j, \ldots, u_{MM}^j]^{\mathsf{T}}$ in a lexicographic order. Similarly, let $f_{il} = f(x_i, y_l)$ and define the concatenated column vector $f_h = [f_{11}, f_{21}, \ldots, f_{M1}, \ldots, f_{1M}, f_{2M}, \ldots, f_{MM}]^{\mathsf{T}}$. Both ϕ_h and $g_{\delta,h}$ are defined analogously. Let Δ_h denote the discretized Laplacian matrix with the second-order accurate five-point central finite difference and the homogeneous Dirichlet boundary conditions. In matrix Kronecker product notation [52], we explicitly have $\Delta_h = (I_M \otimes Q_h + Q_h \otimes I_M) \in \mathbb{R}^{m\times m}$, where $m = M^2$ is the degree of freedom in space, $I_M \in \mathbb{R}^{M\times M}$ is an identity matrix, and $Q_h = (1/h^2)\text{tridiag}\{1, -2, 1\} \in \mathbb{R}^{M\times M}$ is the tridiagonal 1D discrete Laplacian matrix. With the central finite difference in space and a backward Euler scheme in time, the full discretization of (2) reads (with $u^0 = \phi_h$ being the given initial condition)

$$\begin{cases} (u^j - u^{j-1})/\tau - \Delta_h u^j - f_h = 0, \ j = 1, 2, \ldots, n, \\ u^n + \beta f_h = g_{\delta,h}, \end{cases} \tag{3}$$

which can be coupled into a nonsymmetric sparse linear system

$$\widehat{A}_h \boldsymbol{u}_h = \widehat{\boldsymbol{b}}_h \in \mathbb{R}^{m(n+1)}, \tag{4}$$

where (here $I_h \in \mathbb{R}^{m\times m}$ is an identity matrix of the same size as $\Delta_h \in \mathbb{R}^{m\times m}$)

$$\widehat{A}_h = \begin{bmatrix} \beta I_h & 0 & 0 & \cdots & 0 & I_h \\ -I_h & I_h/\tau - \Delta_h & 0 & \cdots & 0 & 0 \\ -I_h & -I_h/\tau & I_h/\tau - \Delta_h & 0 & \cdots & 0 \\ \vdots & 0 & \ddots & \ddots & \ddots & 0 \\ -I_h & 0 & \cdots & -I_h/\tau & I_h/\tau - \Delta_h & 0 \\ -I_h & 0 & \cdots & 0 & -I_h/\tau & I_h/\tau - \Delta_h \end{bmatrix}, \boldsymbol{u}_h = \begin{bmatrix} f_h \\ u^1 \\ u^2 \\ \vdots \\ u^{n-1} \\ u^n \end{bmatrix}, \widehat{\boldsymbol{b}}_h = \begin{bmatrix} g_{\delta,h} \\ \phi_h/\tau \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}.$$

Clearly, the $(1,1)$ block βI_h is different from the other diagonal blocks $(I_h/\tau - \Delta_h)$, which prevents a Kronecker product formulation of $\widehat{A}_h$ desired in direct PinT algorithm as shown in our new parameterized QBVM.

2.1. A new quasi-boundary value method based on finite difference scheme

To get a better structured linear system that allows a fast direct PinT solver upon finite difference discretization, we propose the following new parameterized QBVM (PQBVM)

$$\begin{cases} u_t - \Delta u = f, & \text{in } \Omega \times (0,T), \\ u(\cdot,t) = 0, & \text{on } \partial\Omega \times (0,T), \\ u(\cdot,0) = \phi, & \text{in } \Omega, \\ u(\cdot,T) + \beta(\alpha f(\cdot) - \Delta f(\cdot)) = g_\delta, & \text{in } \Omega, \end{cases} \tag{5}$$

where $\alpha \ge 0$ is a free design parameter to control the condition number of the subsequent direct PinT solver. In general with $\alpha \ne 0$, we expect the above new PQBVM to have a similar convergence rate as the standard QBVM in [11] due to the shared term f. We highlight that the special choice of $\alpha = 0$ in fact leads to the known modified QBVM (MQBVM) established in [44] within the framework of time-fractional diffusion equation. However, the authors in [44] focused on studying the improved convergence rates of MQBVM, without discussing fast algorithms for solving the regularized linear systems. In this paper, we propose the above PQBVM mainly from the perspective of designing regularized linear systems with better structures that are suitable for constructing direct PinT algorithms, while at the same time retaining the

convergence rates of QBVM. We emphasize that $\alpha \geq 0$ should not be treated as another regularization parameter like $\beta > 0$ and it will be chosen purely for facilitating the development of fast direct PinT system solvers.

With the same central finite difference scheme in space and backward Euler scheme in time as used in the above discretization (3), the full discretization of (5) leads to

$$\begin{cases} (u^j - u^{j-1})/\tau - \Delta_h u^j - f_h = 0, \ j = 1, 2, \ldots, n, \\ u^n + \beta(\alpha f_h - \Delta_h f_h) = g_{\delta,h}, \end{cases} \tag{6}$$

which, after dividing the last equation by β, can be reformulated into a nonsymmetric linear system

$$A_h \boldsymbol{u}_h = \boldsymbol{b}_h \in \mathbb{R}^{m(n+1)}, \tag{7}$$

where

$$A_h = \begin{bmatrix} \alpha I_h - \Delta_h & 0 & 0 & \cdots & 0 & I_h/\beta \\ -I_h & I_h/\tau - \Delta_h & 0 & \cdots & 0 & 0 \\ -I_h & -I_h/\tau & I_h/\tau - \Delta_h & 0 & \cdots & 0 \\ \vdots & 0 & \ddots & \ddots & \ddots & \\ -I_h & 0 & \cdots & -I_h/\tau & I_h/\tau - \Delta_h & 0 \\ -I_h & 0 & \cdots & 0 & -I_h/\tau & I_h/\tau - \Delta_h \end{bmatrix}, \boldsymbol{b}_h = \begin{bmatrix} g_{\delta,h}/\beta \\ \phi_h/\tau \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}.$$

We can now rewrite the block-structured matrix A_h in (7) into an elegant Kronecker product form

$$A_h = B \otimes I_h - I_t \otimes \Delta_h \tag{8}$$

where $I_t \in \mathbb{R}^{(n+1)\times(n+1)}$ denotes an identity matrix and the time discretization matrix B is given by

$$B = \begin{bmatrix} \alpha & 0 & 0 & \cdots & 0 & 1/\beta \\ -1 & 1/\tau & 0 & \cdots & 0 & 0 \\ -1 & -1/\tau & 1/\tau & 0 & \cdots & 0 \\ \vdots & 0 & \ddots & \ddots & \ddots & 0 \\ -1 & 0 & \cdots & -1/\tau & 1/\tau & 0 \\ -1 & 0 & \cdots & 0 & -1/\tau & 1/\tau \end{bmatrix} \in \mathbb{R}^{(n+1)\times(n+1)}. \tag{9}$$

Notice that the identity matrices I_h and I_t are of different size, due to the different discretization in space and time, respectively. Such a Kronecker product reformulation (8) is crucial to develop our following fast direct PinT solver, which requires the matrix B to be diagonalizable. Since B is nonsymmetric and has a nontrivial structure, its diagonalizability is not straightforward and will be discussed separately in Section 3.

2.2. A diagonalization-based direct PinT solver

Suppose B has a diagonalization $B = VDV^{-1}$, where $D = \mathrm{diag}(d_1, \ldots, d_{n+1})$ with d_j being the jth eigenvalue of B and the jth column of the invertible matrix V gives the corresponding eigenvector. Then we can factorize A_h into the product form

$$A_h = (VDV^{-1}) \otimes I_h - I_t \otimes \Delta_h = \underbrace{(V \otimes I_h)}_{\text{Step-(a)}} \underbrace{(D \otimes I_h - I_t \otimes \Delta_h)}_{\text{Step-(b)}} \underbrace{(V^{-1} \otimes I_h)}_{\text{Step-(c)}}.$$

Hence, let $Z = \mathtt{mat}(\boldsymbol{b}_h) \in \mathbb{R}^{m\times(n+1)}$, the solution $\boldsymbol{u}_h = A_h^{-1}\boldsymbol{b}_h$ can be computed via three steps:

$$\begin{aligned} &\text{Step-(a)} \ \ S_1 = ZV^{-\mathsf{T}}, \\ &\text{Step-(b)} \ \ S_2(:, j) = \left(d_j I_h - \Delta_h\right)^{-1} S_1(:, j), \quad j = 1, 2, \ldots, n+1, \\ &\text{Step-(c)} \ \ \boldsymbol{u}_h = \mathtt{vec}(S_2 V^{\mathsf{T}}), \end{aligned} \tag{10}$$

where $S_{1,2}(:, j)$ denotes the jth column of $S_{1,2}$ and V^{T} defines the non-conjugate transpose of V. Here we have used the efficient Kronecker product property $(C \otimes I_h)\mathtt{vec}(X) = \mathtt{vec}(XC^{\mathsf{T}})$ for any compatible matrices C and X. Clearly, the $(n+1)$ fully independent complex-shifted linear systems in Step-(b) can be computed in parallel. Notice that a different spatial discretization will only affect the matrix Δ_h in Step-(b).

Let $\kappa_p(V) = \|V\|_p \|V^{-1}\|_p$ with $p = 1, 2, \infty$ denotes the matrix p-norm condition number of V. Numerically, the overall round-off errors of such a three-step diagonalization-based PinT direct solver is proportional to the condition number of V, see Lemma 3.2 in [53] for a detailed round-off error analysis. Hence, it is essential to design the matrix B so that the condition number of V is well controlled for stable computations. In particular, it would become numerically unstable if

$\kappa(V)$ grows exponentially with respect to n. In view of the discretization errors in space and time, it is acceptable to have $\kappa(V) = \mathcal{O}(n^q)$ with a small q (say $q \le 3$).

3. The diagonalization of B and the condition number of V

In this section we will prove that the matrix B in (9) with a special choice of α is indeed diagonalizable and also provide explicit formulas for computing its eigenvector matrix V and estimating its condition number. More specifically, we will prove that $\kappa_1(V) = \mathcal{O}(cn)$ with $c = \beta/\tau^2$ under the special choice $\alpha = \alpha_* := 1/\tau + \tau/\beta$. Although the trivial choice of $\alpha = 0$ can be numerically used in the diagonalization-based direct PinT solver, there is no theoretical guarantee that the corresponding matrix B is diagonalizable and/or the eigenvector matrix V is well-conditioned for stable computation. In particular, the corresponding analysis based on the trivial choice of $\alpha = 0$ seems to be far too difficult to perform due to much more complicated eigenvalue/eigenvector expressions, which shows the necessity of introducing the free design parameter α.

Let λ be an eigenvalue of B with nonzero eigenvector $\boldsymbol{v} = [v_0, \ldots, v_n]^{\mathsf{T}}$. By $B\boldsymbol{v} = \lambda\boldsymbol{v}$ we have

$$\alpha v_0 + v_n/\beta = \lambda v_0, \tag{11}$$

$$-v_0 + v_1/\tau = \lambda v_1, \tag{12}$$

and

$$-v_0 - v_{k-1}/\tau + v_k/\tau = \lambda v_k, \quad k = 2, \ldots, n. \tag{13}$$

Obviously, $v_0 \neq 0$ and $\lambda \neq 1/\tau$ since otherwise it leads to $\boldsymbol{v} = \boldsymbol{0}$. Without loss of generality, we choose $v_0 = 1/\tau$. It is readily seen from (12) and (13) that

$$v_k = \mu + \cdots + \mu^k = \frac{\mu^{k+1} - \mu}{\mu - 1}, \quad k = 1, \ldots, n, \tag{14}$$

where $\mu = 1/(1 - \tau\lambda) \neq 1$. Assume $\alpha = \alpha_* := 1/\tau + \tau/\beta$ and denote $c = \beta/\tau^2$. Substituting $v_0 = 1/\tau$ and the above formula for v_n into Eq. (11) yields

$$\frac{1}{\tau^2} + \frac{1}{\beta} + \frac{1}{\beta}(\mu + \cdots + \mu^n) = \frac{\lambda}{\tau},$$

which, upon multiplying both sides by $\beta\mu$, reduces to

$$c + \mu + \cdots + \mu^{n+1} = 0. \tag{15}$$

The $(n+1)$ roots of (15) determine the $(n+1)$ eigenvalues of B. For convenience, we define

$$\psi(\mu) := (\mu - 1)(c + \mu + \cdots + \mu^{n+1}) = \mu^{n+2} + (c-1)\mu - c. \tag{16}$$

As a sufficient condition for guaranteeing the diagonalizability of B, we have the following result.

Lemma 3.1. *If $\alpha = \alpha_* := 1/\tau + \tau/\beta$ and $c := \beta/\tau^2 > 1$, then the matrix B has $n+1$ distinct eigenvalues. In particular, this implies the nonsymmetric matrix B is indeed diagonalizable.*

Proof. It suffices to prove that Eq. (15) has no repeated roots. Assume to the contrary that $\mu = \mu_0$ is a repeated root of (15). We then have $\psi(\mu_0) = \psi'(\mu_0) = 0$. From $\psi'(\mu_0) = 0$ we obtain

$$\mu_0^{n+1} = (1 - c)/(n + 2).$$

Substituting this into $\psi(\mu_0) = 0$ gives

$$\mu_0 = \frac{c(n+2)}{(c-1)(n+1)}.$$

Coupling the above two equations yields

$$c^{n+1}(n+2)^{n+2} + (c-1)^{n+2}(n+1)^{n+1} = 0,$$

which contradicts to the condition $c > 1$. This completes the proof. □

Denote by $\mu_1, \ldots, \mu_{n+1}$ the distinct roots of Eq. (15). The eigenvalues of B are $\lambda_k = (1 - 1/\mu_k)/\tau$ with $k = 1, \ldots, n+1$. The above eigenvector expression (14) implies that the eigenvector (after rescaled by $(\mu_k - 1)$) corresponding to the eigenvalue λ_k can be chosen as

$$\boldsymbol{v}^{(k)} = [(\mu_k - 1)/\tau, \mu_k^2 - \mu_k, \ldots, \mu_k^{n+1} - \mu_k]^{\mathsf{T}}.$$

Hence, we have the eigendecomposition $B = V\mathrm{diag}\{\lambda_1, \ldots, \lambda_{n+1}\}V^{-1}$ with the eigenvector matrix

$$V = \begin{bmatrix} (\mu_1 - 1)/\tau & \cdots & (\mu_{n+1} - 1)/\tau \\ \mu_1^2 - \mu_1 & \cdots & \mu_{n+1}^2 - \mu_{n+1} \\ \vdots & & \vdots \\ \mu_1^{n+1} - \mu_1 & \cdots & \mu_{n+1}^{n+1} - \mu_{n+1} \end{bmatrix}. \tag{17}$$

We remark that $V\Phi$ is also an eigenvector matrix for any nonsingular diagonal matrix Φ.

The following lemma shows that the roots of Eq. (15) are located in the annulus $1 < |\mu| < (2c-1)^{1/(n+1)}$ on the complex plane. This implies $|\tau\lambda_k - 1| = \frac{1}{|\mu_k|} \in ((2c-1)^{-1/(n+1)}, 1)$.

Lemma 3.2. *Let $\mu_1, \ldots, \mu_{n+1}$ be distinct roots of Eq. (15). If $c > 1$, then $|\mu_k| > 1$ and $|\mu_k|^{n+1} < 2c-1$ for $k = 1, \ldots, n+1$. In particular, this implies $\Re(\lambda_k) > 0$ for $k = 1, \ldots, n+1$.*

Proof. It is obvious from $c > 1$ that $\mu_k \neq 1$. We claim that $|\mu_k| > 1$; otherwise, we obtain from $\psi(\mu_k) = 0$ that

$$c = \mu_k^{n+2} + (c-1)\mu_k \leq |\mu_k|^{n+2} + (c-1)|\mu_k| \leq c,$$

which is satisfied if and only if $\mu_k = 1$, a contradiction. Next, it follows from $\psi(\mu_k) = 0$ and $|\mu_k| > 1$ that

$$|\mu_k^{n+1}| = |c/\mu_k - (c-1)| \leq c/|\mu_k| + |c-1| < 2c - 1.$$

Clearly, with $\mu_k = 1/(1 - \tau\lambda_k)$ we have

$$1 - \tau\Re(\lambda_k) = \Re(1 - \tau\lambda_k) \leq |1 - \tau\lambda_k| = 1/|\mu_k| < 1,$$

which implies $\Re(\lambda_k) > 0$ for $k = 1, \ldots, n+1$. The proof is completed. □

To find an explicit expression for the inverse matrix $W = V^{-1}$, we shall make use of the Lagrange interpolation polynomials (such that $L_j(\mu_l) = \delta_{j,l}$ with $\delta_{j,l}$ being the Kronecker delta)

$$L_j(\mu) = \prod_{1 \leq k \leq n+1, k \neq j} \frac{\mu - \mu_k}{\mu_j - \mu_k} = \sum_{k=1}^{n+1} L_{jk}\mu^{k-1}, \quad j = 1, \ldots, n+1, \tag{18}$$

where $L_{jk} = L_j^{(k-1)}(0)/(k-1)!$ is the coefficient of μ^{k-1} in the polynomial expression of $L_j(\mu)$. Let $U = [U_{kl}]_{k,l=1}^{n+1}$ be the Vandermonde matrix with $U_{kl} = \mu_l^{k-1}$ and $L = \left[L_{jk}\right]_{j,k=1}^{n+1}$. It follows from the identities $L_j(\mu_l) = \sum_{k=1}^{n+1} L_{jk}\mu_l^{k-1} = \delta_{j,l}$ that $LU = I$ with I being an identity matrix of size $(n+1)$.

Lemma 3.3. *Let $W = V^{-1} = \left[W_{jk}\right]_{j,k=1}^{n+1}$. We have the following expressions*

$$W_{jk} = \begin{cases} L_{j,k+1} - L_j(1)/(n+c+1), & 1 < k < n+1, \\ -L_j(1)/(n+c+1), & k = n+1, \\ c\tau L_j(1)/(n+c+1) - \tau L_{j1}, & k = 1. \end{cases} \tag{19}$$

Proof. First, we consider the case $k > 1$. It follows from (17) and $VW = I$ that

$$\sum_{j=1}^{n+1} (\mu_j - 1)W_{jk} = 0,$$

and

$$\sum_{j=1}^{n+1} (\mu_j^l - \mu_j)W_{jk} = \begin{cases} 1, & l = k, \\ 0, & l > 1,\ l \neq k. \end{cases}$$

For convenience, we denote $S_k = \sum_{j=1}^{n+1} W_{jk}$. It is readily seen from the above equations that

$$\sum_{j=1}^{n+1} \mu_j^l W_{jk} = \begin{cases} S_k + 1, & l = k, \\ S_k, & l \neq k. \end{cases} \tag{20}$$

Recall that $\mu_1, \ldots, \mu_{n+1}$ are the roots of the Eqs. (15); namely, $c + \sum_{l=1}^{n+1} \mu_j^l = 0$. We then obtain

$$0 = c\sum_{j=1}^{n+1} W_{jk} + \sum_{l=1}^{n+1}\sum_{j=1}^{n+1} \mu_j^l W_{jk} = cS_k + \left(\sum_{l=1, l\neq k}^{n+1} S_k\right) + (S_k + 1) = cS_k + (n+1)S_k + 1,$$

which implies $S_k = -1/(n+c+1)$ is independent of k. Now, we multiply both sides of (20) by $L_{m,l+1}$ and then add from $l=0$ to $l=n$ to find

$$W_{mk} = \sum_{j=1}^{n+1} L_m(\mu_j) W_{jk} = \sum_{l=0}^{n} L_{m,l+1} S_k + \begin{cases} L_{m,k+1}, & k < n+1, \\ 0 & , \quad k = n+1 \end{cases}$$
$$= -\frac{L_m(1)}{n+c+1} + \begin{cases} L_{m,k+1}, & k < n+1, \\ 0 & , \quad k = n+1. \end{cases}$$

Next, we consider the case $k=1$. It follows from (17) and $VW = I$ that

$$\sum_{j=1}^{n+1} (\mu_j - 1) W_{j1} = \tau,$$

and

$$\sum_{j=1}^{n+1} (\mu_j^l - \mu_j) W_{j1} = 0, \quad l > 1.$$

For convenience, we denote $S_1 = \sum_{j=1}^{n+1} W_{j1}$. It is readily seen from the above equations that

$$\sum_{j=1}^{n+1} \mu_j^l W_{jk} = \begin{cases} S_1, & l = 0, \\ S_1 + \tau, & l = 1, \ldots, n+1.. \end{cases} \tag{21}$$

Recall that $\mu_1, \ldots, \mu_{n+1}$ are the roots of the Eqs. (15). We then obtain

$$0 = c \sum_{j=1}^{n+1} W_{jk} + \sum_{l=1}^{n+1} \sum_{j=1}^{n+1} \mu_j^l W_{jk} = cS_1 + (n+1)(S_1 + \tau),$$

which implies $S_1 = -(n+1)\tau/(n+c+1)$. Now, we multiply both sides of (21) by $L_{m,l+1}$ and then add from $l=0$ to $l=n$ to find

$$W_{m1} = \sum_{j=1}^{n+1} L_m(\mu_j) W_{jk} = L_{m1} S_1 + \sum_{l=1}^{n} L_{m,l+1}(S_1 + \tau) = -\tau L_{m1} + \frac{c\tau L_m(1)}{n+c+1}.$$

This completes the proof. □

The following lemma gives an explicit formula for L_{jk}, which will be used to estimate $\|W\|_1$.

Lemma 3.4. *Let L_{jk} with $1 \le j, k \le n+1$ be the coefficient of μ^{k-1} in the polynomial expression of Lagrange interpolation polynomial $L_j(u)$ defined in (18). We have*

$$L_{jk} = \frac{\mu_j^{n+2-k} - 1}{(n+2)\mu_j^{n+1} + c - 1} = \frac{\mu_j^{1-k} - \mu_j^{-n-1}}{n+2+(c-1)\mu_j^{-n-1}}. \tag{22}$$

Proof. Recall from (16) that $\psi(\mu) = (\mu - 1)(c + \mu + \cdots + \mu^{n+1})$. Since $\mu_1, \ldots, \mu_{n+1}$ are distinct roots of the polynomial equation $c + \mu + \cdots + \mu^{n+1} = 0$, we can factor the polynomial as

$$c + \mu + \cdots + \mu^{n+1} = \prod_{k=1}^{n+1} (\mu - \mu_k).$$

Consequently, $\psi(\mu) = (\mu - 1)(\mu - \mu_1) \cdots (\mu - \mu_{n+1})$ and

$$\frac{\psi(\mu)}{(\mu - 1)(\mu - \mu_j)} = \prod_{1 \le k \le n+1, k \ne j} (\mu - \mu_k).$$

We denote

$$a_j = \prod_{1 \le k \le n+1, k \ne j} (\mu_j - \mu_k) = \lim_{\mu \to \mu_j} \frac{\psi(\mu)}{(\mu - 1)(\mu - \mu_j)} = \frac{\psi'(\mu_j)}{\mu_j - 1}.$$

On the other hand, we obtain from (18) that

$$c+\mu+\cdots+\mu^{n+1}=\prod_{k=1}^{n+1}(\mu-\mu_k)=a_j(\mu-\mu_j)L_j(u)=a_j(\mu-\mu_j)\sum_{k=1}^{n+1}L_{jk}\mu^{k-1}$$
$$=a_j[L_{j,n+1}\mu^{n+1}+(L_{jn}-\mu_jL_{j,n+1})\mu^n+\cdots+(L_{j1}-\mu_jL_{j2})\mu-\mu_jL_{j1}].$$

Comparing the polynomial coefficients on both side of the equation gives $c=-a_j\mu_jL_{j1}$, and

$$1=a_j(L_{j1}-\mu_jL_{j2})=\cdots=a_j(L_{jn}-\mu_jL_{j,n+1})=a_jL_{j,n+1}.$$

It is readily seen that

$$a_jL_{jk}=1+\mu_j+\cdots+\mu_j^{n+1-k}=\frac{\mu_j^{n+2-k}-1}{\mu_j-1}.$$

This together with $a_j=\psi'(\mu_j)/(\mu_j-1)$ and (16) proves (22). □

To show that $|L_{jk}|=O(1/n)$ uniformly for all $1\le j,k\le n+1$, we need the following lemma.

Lemma 3.5. *Assume $c>1$ and $n>11$. Let $\mu_1,\ldots,\mu_{n+1}$ be the distinct roots of (15). We have $|n+2+(c-1)\mu_j^{-n-1}|>n/2$ for all $j=1,\ldots,n+1$.*

Proof. We will prove by contradiction. Assume to the contrary that $|n+2+(c-1)\mu_j^{-n-1}|\le n/2$ for some $\mu_j=re^{i\theta}$ with $r>1$ and $\theta\in[0,\pi]$. Let $b=r^{n+1}/(c-1)>0$, we then have

$$|b(n+2)+\cos[(n+1)\theta]-i\sin[(n+1)\theta]|\le bn/2,$$

which gives

$$\cos[(n+1)\theta]<2b+\cos[(n+1)\theta]<-\frac{bn}{2},\qquad|\sin[(n+1)\theta]|\le\frac{bn}{2}.\tag{23}$$

Note from (15) and (16) that $\mu_j^{n+1}=c/\mu_j-(c-1)$, which upon dividing both sides by $(c-1)$ gives

$$b\,(\cos[(n+1)\theta]+i\sin[(n+1)\theta])=a(\cos\theta-i\sin\theta)-1,$$

where $a=c/[r(c-1)]>0$. Hence we have

$$b=\sqrt{1-2a\cos\theta+a^2},\quad b\cos[(n+1)\theta]=a\cos\theta-1,\quad b\sin[(n+1)\theta]=-a\sin\theta.\tag{24}$$

The third equality together with $\theta\in[0,\pi]$ implies $\sin[(n+1)\theta]=-(a/b)\sin\theta\le 0$, which gives $\theta\ge\pi/(n+1)$. We further obtain from the three equalities in (24) and two inequalities in (23) that

$$0<1-2a\cos\theta+a^2=b^2<\frac{b^2n}{2}<-b\cos[(n+1)\theta]=1-a\cos\theta<b<-\frac{2}{n}\cos[(n+1)\theta]<\frac{2}{n}$$

and hence (note the inequality $1-2a\cos\theta+a^2<1-a\cos\theta$ gives $a<\cos\theta$)

$$a\sin\theta=b|\sin[(n+1)\theta]|\le\frac{b^2n}{2}<\frac{2}{n},\qquad 1-\frac{2}{n}<a\cos\theta<a<\cos\theta.\tag{25}$$

Consequently, we obtain $(n-2)<na$, $\theta\in[\pi/(n+1),\pi/2)$ such that $\theta<\tan\theta$, and

$$\frac{\pi(n-2)}{n(n+1)}<\theta\cos\theta<\sin\theta<\frac{2}{na}<\frac{2}{n-2},\tag{26}$$

which is not true for $n>11$ and hence contradicts our assumption. This completes our proof. □

Finally, we are ready to estimate the condition number of the eigenvector matrix V in (17).

Theorem 3.1. *If $\alpha=\alpha_*:=1/\tau+\tau/\beta$ and $c=\beta/\tau^2>1$, then $\kappa_1(V)=\|V\|_1\|W\|_1=\mathcal{O}(cn)$.*

Proof. Lemma 3.2 implies $|\mu_k|^j<2c-1$ for any $1\le j,k\le n+1$. It is easily seen from (17) that

$$\sum_{j=1}^{n+1}|V_{jk}|=|\mu_k-1|/\tau+\sum_{j=2}^{n+1}|\mu_k^j-\mu_k|\le(2c)/\tau+(4c-2)n=(2c/T)n+(4c-2)n.$$

In particular, $\|V\|_1=\mathcal{O}(cn)$. Lemma 3.2 also implies $|\mu_j|>1$ for any $1\le j\le n+1$. Assume $n>11$. It then follows from Lemmas 3.4 and 3.5 that

$$|L_{jk}|\le\frac{|\mu_j|^{1-k}+|\mu_j|^{-n-1}}{|n+2+(c-1)\mu_j^{-n-1}|}<\frac{1+1}{n/2}=4/n$$

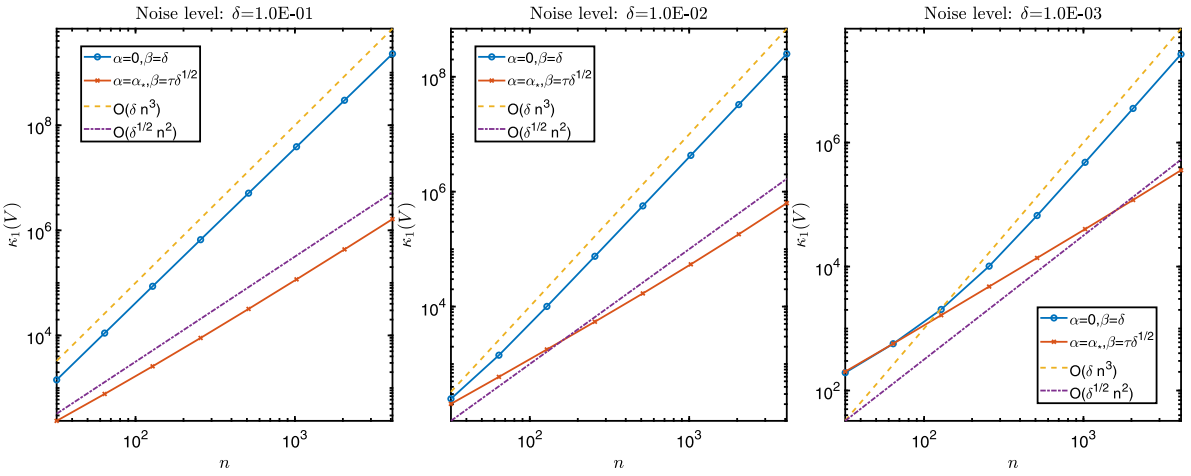

Fig. 1. Comparison of the condition number $\kappa_1(V)$ and its estimated bounds with two different choices of α and β.

for all $1 \le j, k \le n+1$. This together with Lemma 3.3 and $|L_j(1)| = |\sum_{k=1}^{n+1} L_{jk}| < \frac{4(n+1)}{n}$ yields

$$W_{jk} \le \begin{cases} |L_{j,k+1}| + \frac{|L_j(1)|}{n+c+1} \le \frac{4}{n} + \frac{4(n+1)}{n(n+c+1)} \le \frac{8}{n}, & 1 < k < n+1, \\ \frac{|L_j(1)|}{n+c+1} \le \frac{4(n+1)}{n(n+c+1)} < \frac{4}{n}, & k = n+1, \\ \frac{c\tau|L_j(1)|}{n+c+1} + \tau|L_{j1}| \le \frac{4c\tau(n+1)}{n(n+c+1)} + \frac{4\tau}{n} < \frac{4\tau(n+1)}{n} + \frac{4\tau}{n} = \frac{4\tau(n+2)}{n}, & k = 1. \end{cases}$$

The above inequalities can be combined into $|W_{jk}| < [8 + 4\tau(n+2)]/n$ for all $1 \le j, k \le n+1$ with $n > 11$. In particular, $\|W\|_1 = \mathcal{O}(1)$. Therefore, the condition number of the eigenvector matrix V with respect to the matrix 1-norm is $\kappa_1(V) = \|V\|_1\|W\|_1 = \mathcal{O}(cn)$. This completes the proof. □

Our subsequent convergence analysis shows that with $\alpha = \alpha_*$ the choice of regularization parameter $\beta = \tau\delta^{1/2}$ gives an $\mathcal{O}(\delta^{1/2})$ convergence rate, which yields a provable condition number estimate $\kappa_1(V) = \mathcal{O}(\delta^{1/2}n/\tau) = \mathcal{O}(\delta^{1/2}n^2)$. We remark that the trivial choice $\alpha = 0$ may also work well in numerical, but the corresponding condition number of V can be larger and it is more difficult to estimate due to very complicated characteristic equations for the eigenvalues of B.

Fig. 1 illustrates the two very different growth rates of the condition number of V corresponding to the MQBVM (with $\alpha = 0, \beta = \delta$) and our PQBVM (with $\alpha = \alpha_*, \beta = \tau\delta^{1/2}$), respectively. For a large mesh size n and noise level δ, the condition number of V with $\alpha = \alpha_*$ is indeed several order of magnitude smaller than the trivial case with $\alpha = 0$, which also numerically validated the estimated condition number growth rate with the optimized choice $\alpha = \alpha_*$. This improvement is achieved through the above rigorous detailed analysis.

4. Convergence analysis

In this section, we will analyze the convergence rate of our proposed PQBVM, where the optimized parameter $\alpha = \alpha_* = 1/\tau + \tau/\beta$ leads to a mesh-dependent regularization parameter $\beta > 0$. We emphasize that the presented convergence analysis is different from the MQBVM [44] case with $\alpha = 0$.

Let $\mathbb{A} = -\Delta$ and define a Hilbert function space $H = H_0^1(\Omega)$ equipped with the standard L^2 norm

$$\|f\|_2 := (f,f)^{1/2} = \left(\int_\Omega f^2 d\boldsymbol{x}\right)^{\frac{1}{2}}.$$

Then the self-adjoint operator $\mathbb{A}$ admits a set of orthonormal eigenfunctions $\{X_l\}_{l\ge1}$ in H, associated to a set of eigenvalues $\{\lambda_l\}_{l\ge1}$ such that $\mathbb{A}X_l = \lambda_l X_l$ with $0 < \lambda_1 < \lambda_2 < \cdots$ and $\lim_{l\to\infty}\lambda_l = +\infty$. Given any $g \in H$, it has a series expansion $g = \sum_{l=1}^{\infty}(g, X_l)X_l$, with $(g, X_l) := \int_\Omega g X_l d\boldsymbol{x}$ for all l. We assume the measured data $g_\delta(x) \in L^2(\Omega)$ and it satisfies

$$\|g - g_\delta\|_2 \le \delta. \tag{27}$$

We also impose *a priori* bound for the heat source, that is,

$$\|f\|_{H^p(\Omega)} := \left(\sum_{l=1}^{\infty}\lambda_l^p (f, X_l)^2\right)^{\frac{1}{2}} \le E_f, \quad p \ge 0, \tag{28}$$

where $E_f > 0$ is a constant. In particular, when $p = 0$, (28) is reduced to the L^2 norm, that is

$$\|f\|_{H^0(\Omega)} = \left(\sum_{l=1}^{\infty}(f, X_l)^2\right)^{1/2} = \|\sum_{l=1}^{\infty}(f, X_l)X_l\|_2 = \|f\|_2 = \left(\int_\Omega f^2 dx\right)^{\frac{1}{2}}. \tag{29}$$

Consider the exact noisy-free problem (1), by separation of variables and the initial condition $\phi = \sum_{l=1}^{\infty}(\phi, X_l)X_l$, the unknown solution function u can be expressed as (by solving the sequence of separated ODE initial value problem: $u_l'(t) + \lambda_l u_l(t) = (f, X_l)$ with $u_l(0) = (\phi, X_l)$)

$$u(\cdot, t) = \sum_{l=1}^{\infty} u_l(t)X_l = \sum_{l=1}^{\infty} e^{-\lambda_l t}\left((f, X_l)\frac{e^{\lambda_l t}}{\lambda_l} + c_l\right)X_l, \quad c_l = (\phi, X_l) - \frac{(f, X_l)}{\lambda_l}. \tag{30}$$

Applying the final time condition $u(\cdot, T) = g = \sum_{l=1}^{\infty}(g, X_l)X_l$, we further obtain

$$u(\cdot, T) = \sum_{l=1}^{\infty}\left(\frac{1 - e^{-\lambda_l T}}{\lambda_l}(f, X_l) + e^{-\lambda_l T}(\phi, X_l)\right)X_l = \sum_{l=1}^{\infty}(g, X_l)X_l = g, \tag{31}$$

which gives the exact source expression

$$f = \sum_{l=1}^{\infty}(f, X_l)X_l, \qquad \text{with} \quad (f, X_l) = \frac{\lambda_l}{1 - e^{-\lambda_l T}}\left((g, X_l) - e^{-\lambda_l T}(\phi, X_l)\right). \tag{32}$$

Clearly, this exact formula (32) is unstable for reconstructing f with a noisy g_δ since $\lambda_l \to \infty$ will magnify the noise, unless certain noise filters or regularization techniques are incorporated. Similarly, we can obtain the representation for the regularized solutions. See (37) and (38) below.

Now we give the error estimate between the regularized solution and the exact solution.

Theorem 4.1. *Let $f^\delta_{\alpha,\beta}(x)$ be the regularized solution of the problem (5) with the measured data g_δ satisfying (27). Let $f(x)$ be the exact solution of the problem (1) and satisfy a priori condition (28) for any $p \geq 0$. Then, by fixing $\alpha = \alpha_* = 1/\tau + \tau/\beta$, there holds*

(1) for $0 < p < 2$, if we choose $\beta = \tau\left(\frac{\delta}{E_f}\right)^{\frac{2}{p+2}}$, we have

$$\|f^\delta_{\alpha,\beta} - f\|_2 \leq C_1 E_f^{\frac{2}{p+2}}\delta^{\frac{p}{p+2}} + O(\tau^{\frac{p}{2}}); \tag{33}$$

(2) for $2 \leq p < 4$, if we choose $\beta = \tau\left(\frac{\delta}{E_f}\right)^{\frac{1}{2}}$, we have

$$\|f^\delta_{\alpha,\beta} - f\|_2 \leq \left(1 + C_2(E_f\delta)^{\frac{1}{2}}\max\left\{1, \left(\frac{\delta}{E_f}\right)^{\frac{p-2}{4}}\right\}\right)(E_f\delta)^{\frac{1}{2}} + O(\tau); \tag{34}$$

(3) for $p \geq 4$, if we choose $\beta = \frac{\tau}{\sqrt{\tau+1}}\left(\frac{\delta}{E_f}\right)^{\frac{1}{2}}$, we have

$$\|f^\delta_{\alpha,\beta} - f\|_2 \leq C_3\sqrt{\tau+1}(E_f\delta)^{\frac{1}{2}} + O(\tau); \tag{35}$$

where C_1, C_2, C_3 are positive constants that only depend on p, T, and λ_1.

Proof. Let $f_{\alpha,\beta}$ be the noise-free regularized solution. By the triangular inequality, we have

$$\|f^\delta_{\alpha,\beta} - f\|_2 \leq \|f^\delta_{\alpha,\beta} - f_{\alpha,\beta}\|_2 + \|f_{\alpha,\beta} - f\|_2, \tag{36}$$

where each term will be estimated separately based on the corresponding series expression.

By the separation of variables and the given side conditions, we can verify the following expressions

$$(f_{\alpha,\beta}, X_l) = \frac{\lambda_l}{1 - e^{-\lambda_l T} + \alpha\beta\lambda_l + \beta\lambda_l^2}\left((g, X_l) - e^{-\lambda_l T}(\phi, X_l)\right) \tag{37}$$

$$(f^\delta_{\alpha,\beta}, X_l) = \frac{\lambda_l}{1 - e^{-\lambda_l T} + \alpha\beta\lambda_l + \beta\lambda_l^2}\left((g_\delta, X_l) - e^{-\lambda_l T}(\phi, X_l)\right). \tag{38}$$

Then it holds that

$$\begin{aligned}\|f^{\delta}_{\alpha,\beta}-f_{\alpha,\beta}\|_2 &= \|\sum_{l=1}^{\infty}\frac{\lambda_l}{1-e^{-\lambda_l T}+\alpha\beta\lambda_l+\beta\lambda_l^2}(g_\delta-g,X_l)X_l\|_2\\ &\le \sup_{l\ge1}\left(\frac{\lambda_l}{1-e^{-\lambda_l T}+\alpha\beta\lambda_l+\beta\lambda_l^2}\right)\|g_\delta-g\|_2\\ &\le \sup_{l\ge1}\left(\frac{1}{\frac{\gamma_1}{\lambda_l}+\alpha\beta+\beta\lambda_l}\right)\delta \le \frac{\delta}{2\sqrt{\gamma_1\beta}+\alpha\beta},\end{aligned}$$

where $\gamma_1 := 1-e^{-\lambda_1 T}>0$. When $\alpha=\alpha_*=1/\tau+\tau/\beta$, we have

$$\|f^{\delta}_{\alpha,\beta}-f_{\alpha,\beta}\|_2 \le \frac{\delta}{2\sqrt{\gamma_1\beta}+\beta/\tau+\tau} \le \frac{\tau\delta}{\beta}. \tag{39}$$

Meanwhile, based on (32) and (37), the error between the noise-free regularized solution and the exact solution satisfies

$$\begin{aligned}\|f_{\alpha,\beta}-f\|_2 &\le \left\|\sum^{\infty}\left(\frac{\lambda_l}{1-e^{-\lambda_l T}+\alpha\beta\lambda_l+\beta\lambda_l^2}-\frac{\lambda_l}{1-e^{-\lambda_l T}}\right)\left((g,X_l)-(\phi,X_l)e^{-\lambda_l T}\right)X_l\right\|_2\\ &= \left(\sum_{l=1}^{\infty}\left(\frac{\lambda_l}{1-e^{-\lambda_l T}+\alpha\beta\lambda_l+\beta\lambda_l^2}-\frac{\lambda_l}{1-e^{-\lambda_l T}}\right)^2\left((g,X_l)-(\phi,X_l)e^{-\lambda_l T}\right)^2\right)^{\frac12}\\ &= \left(\sum_{l=1}^{\infty}\left(\frac{\alpha\beta\lambda_l+\beta\lambda_l^2}{1-e^{-\lambda_l T}+\alpha\beta\lambda_l+\beta\lambda_l^2}\right)^2\frac{1}{\lambda_l^p}\frac{\lambda_l^p\lambda_l^2}{\left(1-e^{-\lambda_l T}\right)^2}\left((g,X_l)-(\phi,X_l)e^{-\lambda_l T}\right)^2\right)^{\frac12}\\ &\le \left(\sup_{l\ge1}A_l\right)\left(\sum_{l=1}^{\infty}\lambda_l^p(f,X_l)^2\right)^{\frac12} = \left(\sup_{l\ge1}A_l\right)\|f\|_{H^p(\Omega)} \le \left(\sup_{l\ge1}A_l\right)E_f,\end{aligned}$$

where

$$A_l = \frac{\alpha\beta\lambda_l+\beta\lambda_l^2}{(1-e^{-\lambda_l T}+\alpha\beta\lambda_l+\beta\lambda_l^2)\lambda_l^{\frac p2}} = \frac{\alpha\beta\lambda_l^{1-\frac p2}+\beta\lambda_l^{2-\frac p2}}{1-e^{-\lambda_l T}+\alpha\beta\lambda_l+\beta\lambda_l^2} \le \frac{\alpha\beta\lambda_l^{1-\frac p2}}{\gamma_1+\alpha\beta\lambda_l}+\frac{\beta\lambda_l^{2-\frac p2}}{\gamma_1+\beta\lambda_l^2}.$$

According to Lemma 2.7 in [44], we can obtain

$$\frac{\alpha\beta\lambda_l^{1-\frac p2}}{\gamma_1+\alpha\beta\lambda_l} \le \begin{cases}C_4(\alpha\beta)^{\frac p2}, & 0<p<2,\\ C_5\alpha\beta, & p\ge2,\end{cases} \quad\text{and}\quad \frac{\beta\lambda_l^{2-\frac p2}}{\gamma_1+\beta\lambda_l^2} \le \begin{cases}C_6\beta^{\frac p4}, & 0<p<4,\\ C_7\beta, & p\ge4,\end{cases}$$

where C_i, $i=4,5,6,7$, are positive constants that only depend on p, T, and λ_1, which leads to

$$A_l \le \begin{cases}C_4(\alpha\beta)^{\frac p2}+C_6\beta^{\frac p4}, & 0<p<2,\\ C_5\alpha\beta+C_6\beta^{\frac p4}, & 2\le p<4,\\ C_5\alpha\beta+C_7\beta, & p\ge4.\end{cases} \tag{40}$$

For $\alpha=\alpha_*=1/\tau+\tau/\beta$, combining (39), (40) and the fact $\alpha\beta=\beta/\tau+\tau\ge2\sqrt{\beta}$, we show the desired error estimates in the following three different cases depending on the range of p:

Case (1): when $0<p<2$, we have (due to $(a+b)^p\le 2^p(a^p+b^p)$ for any $a>0, b>0, p>0$)

$$A_l \le C_4(\alpha\beta)^{\frac p2}+C_6 2^{-\frac p2}(\alpha\beta)^{\frac p2} \le \tilde C_1(\beta/\tau+\tau)^{\frac p2} \le \tilde C_1 2^{\frac p2}((\beta/\tau)^{\frac p2}+\tau^{\frac p2}),$$

then it holds that

$$\|f^{\delta}_{\alpha,\beta}-f\|_2 \le \frac{\tau\delta}{\beta}+\tilde C_1 2^{\frac p2}((\beta/\tau)^{\frac p2}+\tau^{\frac p2})E_f \le \frac{\tau\delta}{\beta}+\tilde C_2(\beta/\tau)^{\frac p2}E_f+O(\tau^{\frac p2}), \tag{41}$$

which, upon choosing $\beta=\tau\left(\frac{\delta}{E_f}\right)^{\frac{2}{p+2}}$ such that $\frac{\tau\delta}{\beta}=(\beta/\tau)^{\frac p2}E_f$, gives the desired error estimate as in (33) with $C_1=1+\tilde C_2$. Here $\tilde C_1$, $\tilde C_2$ are positive constants that only depend on p, T, and λ_1.

Case (2): when $2\le p<4$, we have

$$A_l \le C_5\alpha\beta+C_6 2^{-\frac p2}(\alpha\beta)^{\frac p2}, \tag{42}$$

and therefore

$$\begin{aligned}\|f_{\alpha,\beta}^{\delta}-f\|_2 \leq & \frac{\tau\delta}{\beta}+\left(C_5(\beta/\tau+\tau)+C_6 2^{-\frac{p}{2}}(\beta/\tau+\tau)^{\frac{p}{2}}\right)E_f\\ \leq & \frac{\tau\delta}{\beta}+\left(C_5(\beta/\tau+\tau)+C_6((\beta/\tau)^{\frac{p}{2}}+\tau^{\frac{p}{2}})\right)E_f\\ \leq & \frac{\tau\delta}{\beta}+C_2\max\{\beta/\tau,(\beta/\tau)^{\frac{p}{2}}\}E_f+O(\tau+\tau^{\frac{p}{2}}),\end{aligned}$$

where $C_2>0$ is a constant. By taking $\beta=\tau\left(\frac{\delta}{E_f}\right)^{\frac{1}{2}}$ such that $\frac{\tau\delta}{\beta}=(\beta/\tau)E_f$, we have

$$\begin{aligned}\|f_{\alpha,\beta}^{\delta}-f\|_2 \leq & (E_f\delta)^{\frac{1}{2}}+C_2\max\left\{\left(\frac{\delta}{E_f}\right)^{\frac{1}{2}},\left(\frac{\delta}{E_f}\right)^{\frac{p}{4}}\right\}E_f+O(\tau)\\ \leq & (E_f\delta)^{\frac{1}{2}}\left(1+C_2(E_f\delta)^{\frac{1}{2}}\max\left\{1,\left(\frac{\delta}{E_f}\right)^{\frac{p-2}{4}}\right\}\right)+O(\tau),\end{aligned}$$

which proves the estimate (34).

Case (3): when $p\geq 4$, we have

$$\|f_{\alpha,\beta}^{\delta}-f\|_2 \leq \frac{\tau}{\beta}\delta+(C_5(\beta/\tau+\tau)+C_7\beta)E_f \leq \frac{\tau}{\beta}\delta+\tilde{C}_4(1/\tau+1)\beta E_f+O(\tau), \tag{43}$$

where $\tilde{C}_4>0$ is a constant. The error estimate in (35) is achieved with $C_3=1+\tilde{C}_4$ if we choose $\beta=\frac{\tau}{\sqrt{\tau+1}}\left(\frac{\delta}{E_f}\right)^{\frac{1}{2}}$ such that $\frac{\tau}{\beta}\delta=(1/\tau+1)\beta E_f$. □

Remark 4.1. For $p>0$, Theorem 4.1 indicates that $\|f_{\alpha,\beta}^{\delta}-f\|_2\to 0$ as $\delta\to 0$, and the convergence rate depends on the regularity of f (i.e. $p>0$). In particular, for $p\geq 4$, the obtained convergence rate $O(\delta^{\frac{1}{2}})$ is slightly slower than the derived convergence rate $O(\delta^{\frac{2}{3}})$ of MQBVM [44]. This is reasonable since our PQBVM uses additional nonzero $\alpha f(\cdot)$ term to control the condition number of V.

Remark 4.2. For $p=0$, we have $A_l\leq 1$, then only the boundedness of $\|f_{\alpha,\beta}^{\delta}-f\|_2$ can be ensured.

Remark 4.3. In Theorem 4.1, the regularization parameter β (depends on δ and E_f) was chosen based on an *a priori* choice rule, which is less practical since E_f is in general difficult to obtain. Alternatively, for a given g_δ, an *a posterior* choice rule, i.e., Morozovs discrepancy principle (MDP) [4], can be used to choose the regularization parameter β (independent of E_f); see [54, Section 4.2] for related discussion. Within this framework, a nonlinear equation based on MDP usually needs to be solved for finding the *a posterior* regularization parameter β, for which our proposed direct PinT solver can also be incorporated. However, this requires different analysis and implementation, which is beyond the scope of our paper.

Remark 4.4. The error estimates in Theorem 4.1 are obtained in $L^2(\Omega)$ norm. In practical implementation, one has to discretize the spatial differential operator as shown in Section 2, leading to $O(h^2)$ discretization errors in space. However, such $O(h^2)$ discretization errors can be neglected since the magnitude of noise level δ is usually assumed to be significantly larger than the discretization errors. To provide a rational guide on the suitable choice of h when δ is small, let $f_{\alpha,\beta}^{\delta,h}(x)$ be the discrete regularized solution with α and β being chosen by Theorem 4.1. Due to the used second-order central finite difference in space and first-order backward Euler scheme in time, we formally have the error estimate $\|f_{\alpha,\beta}^{\delta,h}-f_{\alpha,\beta}^{\delta}\|_2\leq C_0(h^2+\tau)$ for some constant C_0. For simplicity, consider the third case with $p\geq 4$ in Theorem 4.1, we can easily obtain

$$\|f_{\alpha,\beta}^{\delta,h}-f\|_2 \leq \|f_{\alpha,\beta}^{\delta,h}-f_{\alpha,\beta}^{\delta}\|_2+\|f_{\alpha,\beta}^{\delta}-f\|_2 \leq C_0h^2+C_3\sqrt{\tau+1}(E_f\delta)^{\frac{1}{2}}+O(\tau), \tag{44}$$

which implies that $h=O(\delta^{1/4})$ is the most appropriate choice since it gives the coarsest mesh step size that achieves the $O(\delta^{1/2})$ total error. In particular, for a large δ, it is not recommended to use a too small $h\ll\delta^{1/4}$ in practice, since the total error is dominated by the δ term. Similar discussion applies to the time step size τ, which plays an additional role in determining the choice of α_*. This also explains the observed clear stagnation of approximation errors in numerical results (see below Tables 1–5) as the mesh is refined while fixing the noise level, and a smaller noise level gives smaller total errors for a sufficiently fine mesh.

5. Numerical examples

In this section, we present some numerical examples to illustrate the computational efficiency of our proposed PQBVM method. All simulations are implemented in serial with MATLAB on a Dell Precision 5820 Workstation with Intel(R)

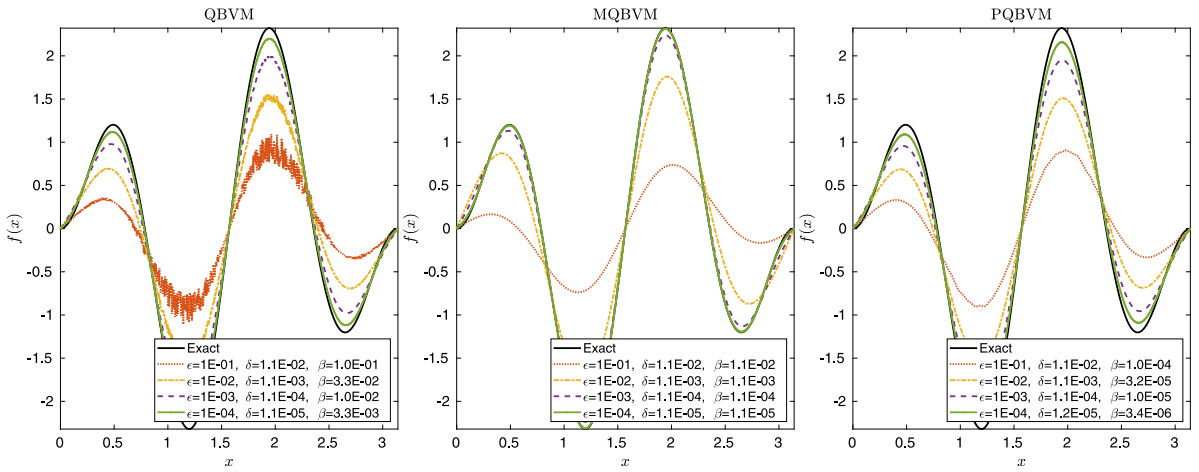

Fig. 2. Reconstructed $f(x)$ in Example 1 with different methods and noise levels $\epsilon \in \{10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}\}$ (using the mesh $h = \pi/1024$, $\tau = T/1024$, $\alpha = \delta^{1/2}$ for QBVM, $\alpha = \delta$ for MQBVM, and $\alpha = \tau\delta^{1/2}$ for PQBVM). The black solid curve is the exact solution.

Core(TM) i9-10900X CPU@3.70 GHz CPU and 64 GB RAM, where CPU times (in seconds) are estimated by the timing functions `tic/toc`. In QBVM, we directly solve the full sparse linear systems with MATLAB's backslash sparse direct solver, which runs very fast for several thousands (but not millions) of unknowns. Our proposed PQBVM (including MQBVM as a special case) will be solved by the three-step fast direct PinT solver (10), where the independent complex-shifted sparse linear systems in Step-(b) can be solved by fast direct solvers (Thomas' algorithm for 1D cases and FFT solver for 2D cases) for rectangular domains with regular grids. The diagonalization of $B = VDV^{-1}$ is computed with MATLAB's `eig` function and Step-(a) $ZV^{-\mathrm{T}}$ is done (with MATLAB code: `Z/(V.')`) by MATLAB's slash('/') direct solver.

To avoid inverse crimes, for a given exact source f we first solve the forward (direct) problem with Crank–Nicolson time-stepping scheme to compute g and then generate the noisy final condition measurement by $g_\delta = g \times (1 + \epsilon \times \mathrm{rand}(-1, 1))$, where $\epsilon > 0$ controls the noise level and $\mathrm{rand}(-1, 1)$ denotes random noise uniformly distributed within $[-1, 1]$. We then further compute the estimated noise bound $\delta := \|g_\delta - g\|_2$. In practice, the obtained noise bound δ may be over-estimated or under-estimated. Since E_f in Theorem 4.1 is unknown, we select more practical regularization parameters $\beta = \delta^{1/2}$, $\beta = \delta$, $\beta = \tau\delta^{1/2}$ for QBVM, MQBVM($\alpha = 0$), and our proposed PQBVM($\alpha = \alpha_*$), respectively. After solving the discretized full linear system, we obtain the approximate source f_h and then compute its discrete $L_2(\Omega)$ norm error as $e_h = \|f_h - f(\cdot)\|_2$. For a fixed mesh, we would expect e_h to decrease as the noise level δ gets smaller, but the discretization errors also affect the overall accuracy, especially for our PQBVM (with $\beta = \tau\delta^{1/2}$). The convergence rate also depends on the regularity of f, where a smoother f shows faster convergence rate.

5.1. 1D and 2D examples with space-dependent source term

Example 1. Choose $\Omega = (0, \pi)$, $T = 1$, $\phi(x) = 0$, and a smooth source function

$$f(x) = x(\pi - x)\sin(4x).$$

Table 1 reports the error results and CPU times with three different regularization methods, where the CPU times of both MQBVM and PQBVM with the PinT direct solver are significantly smaller than that of the QBVM based on MATLAB's backslash direct solver. With a very smooth f, the MQBVM in [44] indeed shows faster convergence rate than both QBVM and PQBVM. As shown in Fig. 2, the QBVM displays undesirable artificial oscillations for large noise levels, which was not visible in both MQBVM and PQBVM, mainly due to the introduced Laplacian regularization term Δf that smooths out the reconstructed approximation.

Example 2. Choose $\Omega = (0, \pi)$, $T = 1$, $\phi(x) = 0$, and a non-smooth source function

$$f(x) = \begin{cases} 2x, & 0 \le x \le \pi/2, \\ 2(\pi - x), & \pi/2 \le x \le \pi, \end{cases}$$

Table 2 reports the error results and CPU times with three different regularization methods, where again the CPU times of both MQBVM and PQBVM based on our PinT direct solver are much smaller than that of QBVM. Fig. 3 illustrates the reconstructed $f(x)$ with different noise levels, where the MQBVM shows only slightly better accuracy with a non-differentiable f. We mention that the QBVM-based regularization has a degraded approximation accuracy near the sharp non-smooth corner at $x = \pi/2$ due to the non-smoothness of f. In such non-smooth cases, total variation

Table 1
Error and CPU results for Example 1 with different mesh sizes and noise levels.

Method	$(m,n)\backslash\ \epsilon$	Errors in L_2 norm				CPU (in seconds)			
		10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-1}	10^{-2}	10^{-3}	10^{-4}
QBVM ($\beta=\delta^{1/2}$)	(256, 256)	1.43e+00	8.08e−01	3.43e−01	1.31e−01	0.5	0.5	0.5	0.5
	(512, 512)	1.41e+00	8.09e−01	3.57e−01	1.26e−01	2.6	2.5	2.6	2.7
	(1024,1024)	1.42e+00	7.97e−01	3.56e−01	1.28e−01	18.6	18.7	18.5	18.6
MQBVM ($\beta=\delta$)	(256, 256)	1.66e+00	6.16e−01	1.12e−01	1.78e−02	0.1	0.1	0.1	0.1
	(512, 512)	1.66e+00	6.22e−01	1.16e−01	1.75e−02	0.3	0.3	0.3	0.3
	(1024,1024)	1.65e+00	6.03e−01	1.11e−01	1.76e−02	1.3	1.3	1.2	1.2
PQBVM ($\beta=\tau\delta^{1/2}$)	(256, 256)	1.48e+00	8.94e−01	4.70e−01	2.67e−01	0.1	0.1	0.1	0.1
	(512, 512)	1.44e+00	8.54e−01	4.08e−01	2.00e−01	0.3	0.3	0.3	0.3
	(1024,1024)	1.42e+00	8.32e−01	3.83e−01	1.65e−01	1.2	1.2	1.2	1.2

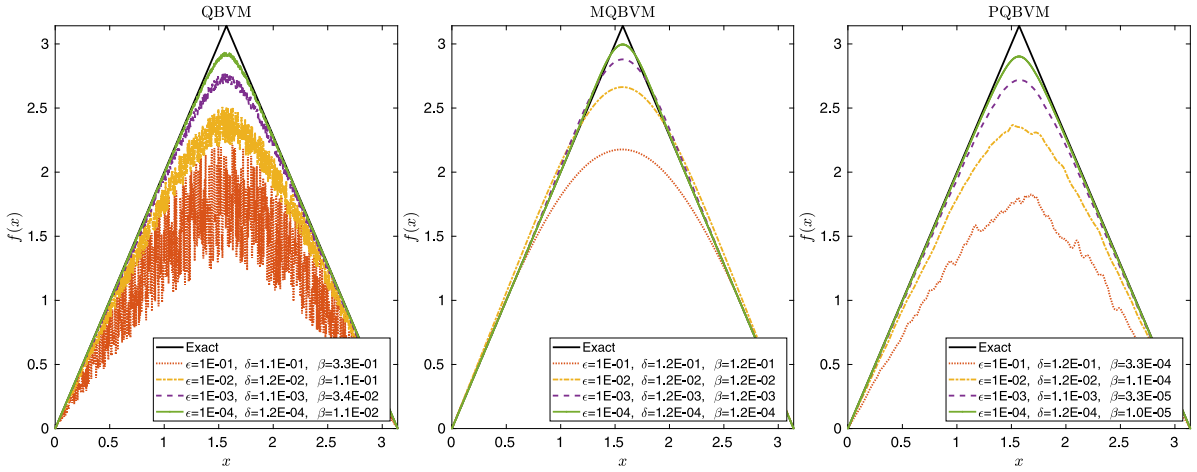

Fig. 3. Reconstructed $f(x)$ in Example 2 with different methods and noise levels $\epsilon \in \{10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}\}$ (using the mesh $h = \pi/1024$, $\tau = T/1024$, $\alpha = \delta^{1/2}$ for QBVM, $\alpha = \delta$ for MQBVM, and $\alpha = \tau\delta^{1/2}$ for PQBVM). The black solid curve is the exact solution.

Table 2
Error and CPU results for Example 2 with different mesh sizes and noise levels.

Method	$(m,n)\backslash\ \epsilon$	Errors in L_2 norm				CPU (in seconds)			
		10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-1}	10^{-2}	10^{-3}	10^{-4}
QBVM ($\beta=\delta^{1/2}$)	(256, 256)	1.21e+00	5.17e−01	2.05e−01	7.62e−02	0.5	0.5	0.5	0.5
	(512, 512)	1.19e+00	5.18e−01	2.06e−01	7.86e−02	2.7	2.6	2.6	2.6
	(1024,1024)	1.21e+00	5.26e−01	2.04e−01	7.93e−02	18.4	18.8	18.4	18.8
MQBVM ($\beta=\delta$)	(256, 256)	5.96e−01	2.31e−01	9.53e−02	4.05e−02	0.1	0.1	0.1	0.1
	(512, 512)	5.98e−01	2.35e−01	9.52e−02	3.97e−02	0.3	0.3	0.3	0.3
	(1024,1024)	6.21e−01	2.34e−01	9.53e−02	3.97e−02	1.3	1.3	1.3	1.3
PQBVM ($\beta=\tau\delta^{1/2}$)	(256, 256)	1.17e+00	5.26e−01	2.15e−01	1.01e−01	0.1	0.1	0.1	0.1
	(512, 512)	1.16e+00	5.16e−01	2.15e−01	8.95e−02	0.3	0.3	0.3	0.3
	(1024,1024)	1.16e+00	5.11e−01	2.08e−01	8.50e−02	1.3	1.2	1.2	1.2

regularization [55–57] may provide better reconstruction accuracy, which requires significantly different treatment and will be further studied in our future work.

Example 3. Choose $\Omega = (0, \pi)$, $T = 1$, $\phi(x) = 0$, and a discontinuous source function

$$f(x) = \begin{cases} 1, & \pi/3 \le x \le 2\pi/3, \\ 0, & \text{else}, \end{cases}$$

Table 3 reports the error results and CPU times with three different regularization methods, where the errors of all three methods are comparable but the CPU times of both MQBVM and PQBVM based on our PinT direct solver are much smaller. Fig. 4 illustrates the reconstructed $f(x)$ with different noise levels, where the MQBVM shows more clear Gibbs

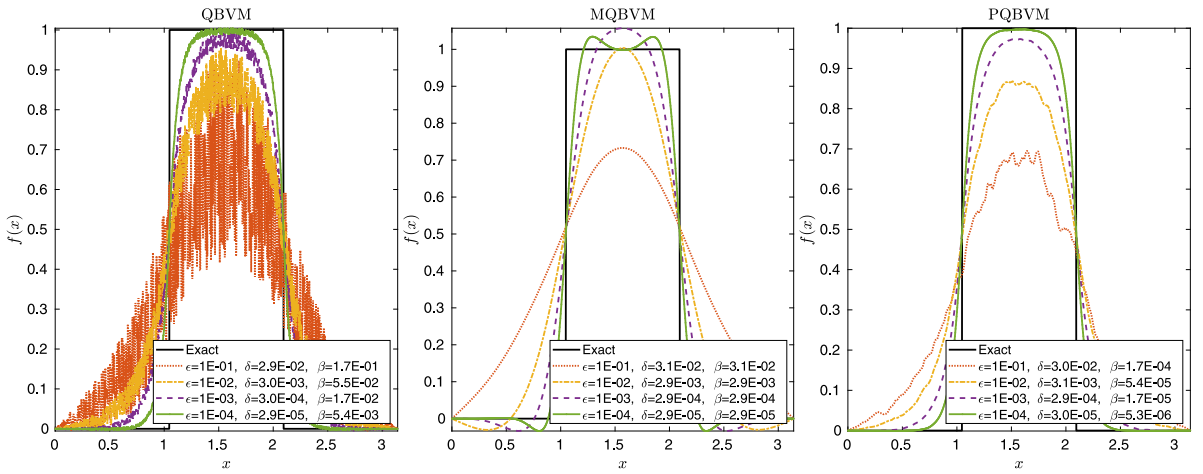

Fig. 4. Reconstructed $f(x)$ in Example 3 with different methods and noise levels $\epsilon \in \{10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}\}$ (using the mesh $h = \pi/1024$, $\tau = T/1024$, $\alpha = \delta^{1/2}$ for QBVM, $\alpha = \delta$ for MQBVM, and $\alpha = \tau\delta^{1/2}$ for PQBVM). The black solid curve is the exact solution.

Table 3
Error and CPU results for Example 3 with different mesh sizes and noise levels.

Method	$(m, n)\backslash\ \epsilon$	Errors in L_2 norm				CPU (in seconds)			
		10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-1}	10^{-2}	10^{-3}	10^{-4}
QBVM ($\beta = \delta^{1/2}$)	(256, 256)	5.04e−01	3.53e−01	2.55e−01	1.91e−01	0.5	0.5	0.5	0.5
	(512, 512)	5.21e−01	3.58e−01	2.58e−01	1.91e−01	2.6	2.5	2.6	2.5
	(1024,1024)	5.25e−01	3.57e−01	2.57e−01	1.92e−01	18.4	18.2	18.3	18.4
MQBVM ($\beta = \delta$)	(256, 256)	5.22e−01	3.42e−01	2.64e−01	1.97e−01	0.1	0.1	0.1	0.1
	(512, 512)	5.18e−01	3.40e−01	2.65e−01	1.97e−01	0.3	0.3	0.3	0.3
	(1024,1024)	5.16e−01	3.42e−01	2.63e−01	1.97e−01	1.3	1.3	1.3	1.3
PQBVM ($\beta = \tau\delta^{1/2}$)	(256, 256)	5.15e−01	3.68e−01	2.84e−01	2.34e−01	0.1	0.1	0.1	0.1
	(512, 512)	4.97e−01	3.61e−01	2.73e−01	2.19e−01	0.3	0.3	0.3	0.3
	(1024,1024)	4.97e−01	3.53e−01	2.65e−01	2.09e−01	1.2	1.2	1.3	1.2

phenomenon due to discontinuity and the PQBVM seems to provide the most stable approximation in the sense of avoiding oscillations and overshooting near the discontinuities.

Example 4. Choose $\Omega = (0, \pi)^2$, $T = 1$, $\phi(x, y) = 0$, and a smooth source function

$$f(x, y) = x(\pi - x)\sin(2x)y(\pi - y)\cos(y).$$

Table 4 reports the error results and CPU times with three different regularization methods, where the CPU times of both MQBVM and PQBVM based on our PinT direct solver are much smaller although the errors of QBVM are slightly smaller than MQBVM and PQBVM. Notice that even for a small mesh size $(m, n) = (64^2, 64)$ the CPU times are decreased from over 2 mins to about 0.04 s, let alone a larger mesh size (such as $(m, n) = (128^2, 128)$ with about 2.1 million unknowns). Here we used "–" to indicate the computation takes an excessively long time for MATLAB's backslash sparse direct solver. Fig. 5 illustrates the reconstructed $f(x)$ with different noise levels, where the differences between three methods are not visible. This example demonstrates the superior computational efficiency of our proposed PinT direct solver in treating more practical 2D/3D problems that are costly to solve by the sparse direct solver.

5.2. Application to separable space and time-dependent source term

Consider the following generalized model [51] with a given positive time-dependent source term $q(t) > 0$:

$$\begin{cases} u_t - \Delta u = f(x)q(t), & \text{in } \Omega \times (0, T), \\ u(\cdot, t) = 0, & \text{on } \partial\Omega \times (0, T), \\ u(\cdot, 0) = \phi, & \text{in } \Omega, \\ u(\cdot, T) + \beta(\alpha f(\cdot) - \Delta f(\cdot)) = g_\delta, & \text{in } \Omega. \end{cases} \tag{45}$$

With the same finite-difference discretization, we get a linear system of Kronecker product form

$$A_h = B_q \otimes I_h - I_t \otimes \Delta_h \tag{46}$$

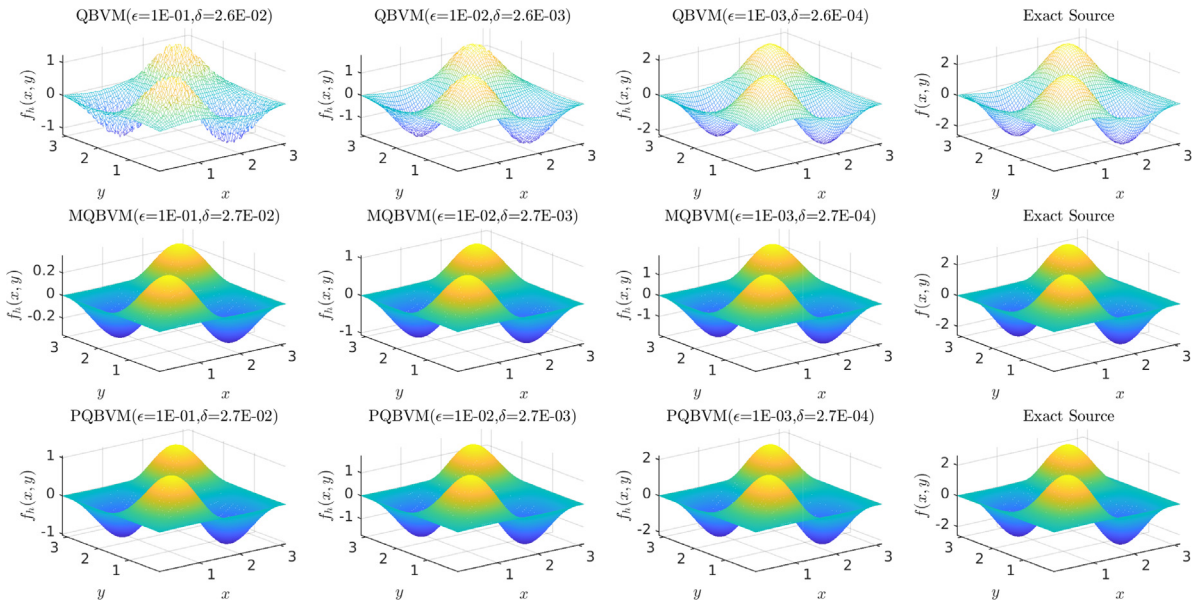

Fig. 5. Reconstructed $f(x)$ in Example 4 with different methods and noise levels $\epsilon \in \{10^{-1}, 10^{-2}, 10^{-3}\}$ (using the mesh $h = \pi/512$, $\tau = T/512$, $\alpha = \delta^{1/2}$ for QBVM, $\alpha = \delta$ for MQBVM, and $\alpha = \tau\delta^{1/2}$ for PQBVM).

Table 4
Error and CPU results for 2D Example 4 with different mesh sizes and noise levels.

Method	$(m, n)\backslash\ \epsilon$	Errors in L_2 norm			CPU (in seconds)		
		10^{-1}	10^{-2}	10^{-3}	10^{-1}	10^{-2}	10^{-3}
QBVM ($\beta = \delta^{1/2}$)	(32^2, 32)	2.21e+00	1.18e+00	4.98e−01	2.00	2.09	2.08
	(64^2, 64)	2.21e+00	1.18e+00	4.93e−01	132.01	127.05	132.84
	(128^2, 128)	–	–	–	–	–	–
MQBVM ($\beta = \delta$)	(32^2, 32)	3.26e+00	2.20e+00	9.83e−01	0.01	0.01	0.01
	(64^2, 64)	3.26e+00	2.21e+00	9.82e−01	0.04	0.04	0.05
	(128^2, 128)	3.26e+00	2.20e+00	9.85e−01	0.28	0.29	0.28
	(256^2, 256)	3.26e+00	2.20e+00	9.84e−01	2.83	2.90	2.91
	(512^2, 512)	3.26e+00	2.20e+00	9.84e−01	33.68	33.80	33.56
PQBVM ($\beta = \tau\delta^{1/2}$)	(32^2, 32)	2.53e+00	1.72e+00	1.19e+00	0.01	0.01	0.01
	(64^2, 64)	2.39e+00	1.49e+00	8.92e−01	0.04	0.04	0.04
	(128^2, 128)	2.31e+00	1.35e+00	7.05e−01	0.29	0.28	0.28
	(256^2, 256)	2.26e+00	1.27e+00	6.04e−01	3.41	2.92	3.44
	(512^2, 512)	2.23e+00	1.23e+00	5.49e−01	34.42	33.96	33.72

where the time discretization matrix B_q is given by (let $q_j = q(t_j)$)

$$B_q = \begin{bmatrix} \alpha & 0 & 0 & \cdots & 0 & 1/\beta \\ -q_1 & 1/\tau & 0 & \cdots & 0 & 0 \\ -q_2 & -1/\tau & 1/\tau & 0 & \cdots & 0 \\ \vdots & 0 & \ddots & \ddots & \ddots & 0 \\ -q_{n-1} & 0 & \cdots & -1/\tau & 1/\tau & 0 \\ -q_n & 0 & \cdots & 0 & -1/\tau & 1/\tau \end{bmatrix} \in \mathbb{R}^{(n+1)\times(n+1)}. \tag{47}$$

Hence, our proposed direct PinT solver can still be applied if assuming $B_q = V_q D_q V_q^{-1}$ is diagonalizable and V_q is well-conditioned (which are not proved yet). In this case, the diagonalizability of B_q and the estimate of $\kappa(V_q)$ are much more complicated to discuss as we did for the B with $q(t) \equiv 1$, which will be left as future work. The following numerical example shows numerically our proposed method indeed works very well.

Example 5. Choose $\Omega = (0, \pi)$, $T = 1$, $\phi(x) = 0$, and the smooth source functions

$$f(x) = x(\pi - x)\sin(4x), \qquad g(t) = e^{-t} + \ln(t + 1) + t^2.$$

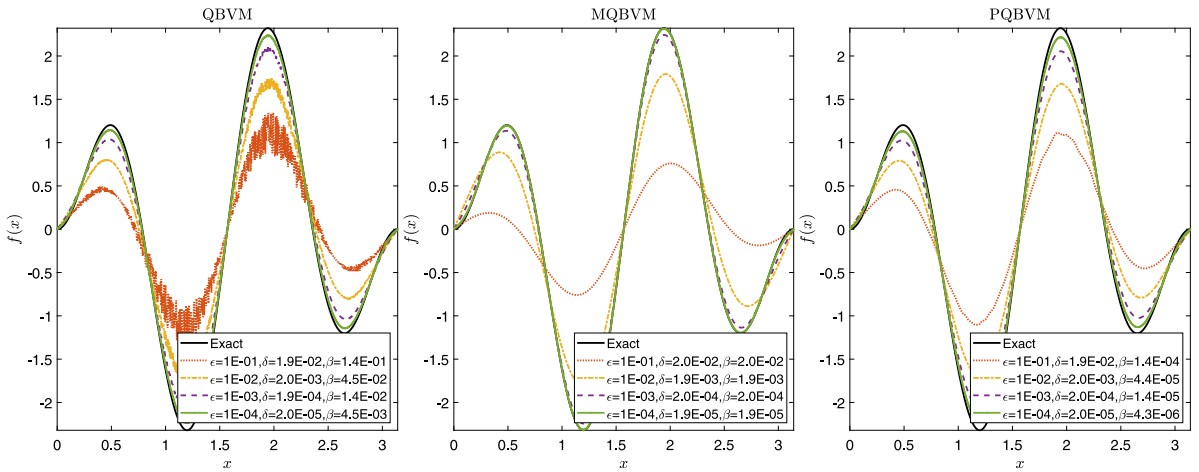

Fig. 6. Reconstructed $f(x)$ in Example 5 with different methods and noise levels $\epsilon \in \{10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}\}$ (using the mesh $h = \pi/1024$, $\tau = T/1024$, $\alpha = \delta^{1/2}$ for QBVM, $\alpha = \delta$ for MQBVM, and $\alpha = \tau\delta^{1/2}$ for PQBVM). The black solid curve is the exact solution.

Table 5
Error and CPU results for Example 5 with different mesh sizes and noise levels.

Method	$(m, n)\backslash\ \epsilon$	Errors in L_2 norm				CPU (in seconds)			
		10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-1}	10^{-2}	10^{-3}	10^{-4}
QBVM ($\beta = \delta^{1/2}$)	(256, 256)	1.23e+00	6.27e−01	2.60e−01	8.50e−02	0.5	0.5	0.5	0.5
	(512, 512)	1.24e+00	6.32e−01	2.58e−01	9.03e−02	2.5	2.5	2.5	2.5
	(1024,1024)	1.23e+00	6.39e−01	2.54e−01	9.13e−02	18.3	18.1	18.0	18.2
MQBVM ($\beta = \delta$)	(256, 256)	1.61e+00	5.29e−01	1.07e−01	1.44e−02	0.1	0.1	0.1	0.1
	(512, 512)	1.62e+00	5.73e−01	1.02e−01	1.56e−02	0.3	0.3	0.3	0.3
	(1024,1024)	1.62e+00	5.73e−01	1.03e−01	1.58e−02	1.3	1.3	1.3	1.3
PQBVM ($\beta = \tau\delta^{1/2}$)	(256, 256)	1.28e+00	6.94e−01	3.27e−01	1.61e−01	0.1	0.1	0.1	0.1
	(512, 512)	1.25e+00	6.62e−01	2.92e−01	1.27e−01	0.3	0.3	0.3	0.3
	(1024,1024)	1.25e+00	6.53e−01	2.79e−01	1.10e−01	1.3	1.2	1.4	1.3

Table 5 reports the error results and CPU times with three different regularization methods as before and Fig. 6 compares the reconstructed f, where similar conclusions can be made as in Example 1. The extra non-constant $q(t)$ term does not seem to affect the effectiveness of our proposed method, although our current analysis does not support this case yet.

6. Conclusions

Inverse space-dependent source problems are ill-posed and effective regularization is required for their stable numerical computation. The quasi-boundary value method and its variants are often used for regularizing such problems, which lead to large-scale ill-conditioned nonsymmetric sparse linear systems upon suitable finite difference discretization in space and time. Such nonsymmetric ill-conditioned all-at-once linear systems are costly to solve by either direct or iterative methods. In this paper we propose to modify the existing quasi-boundary value methods such that the full discretized system matrix admits a block Kronecker sum structure that can be solved by a fast diagonalization-based PinT direct solver. To control the roundoff errors of such a PinT direct solver, we carefully estimate the condition number of the eigenvector matrix of the time discretization matrix, where the free parameter $\alpha = \alpha_*$ is determined for this purpose. Convergence analysis (with a priori choice of regularization parameter β) for our proposed parameterized quasi-boundary value method (PQBVM) is given under the special choice of $\alpha = \alpha_*$. Both 1D and 2D examples show our proposed PinT methods can achieve a comparable accuracy with significantly smaller CPU times. It is interesting to generalize our idea of integrating regularization and fast solvers to other related inverse PDE problems, such as to simultaneously recover the source term and initial value [58–62].

Data availability

No data was used for the research described in the article.

Acknowledgments

The authors greatly appreciate the editor and two anonymous referees for their constructive and insightful suggestions that have greatly improved the quality of this paper. The third author XSW would like to thank National Science Foundation's support under the award NSF-2119688.

References

[1] E.G. Savateev, On problems of determining the source function in a parabolic equation, J. Inverse Ill-Posed Probl. 3 (1) (1995) 83–102.
[2] J.R. Cannon, P. DuChateau, Structural identification of an unknown source term in a heat equation, Inverse Problems 14 (3) (1998) 535–551.
[3] F.-F. Dou, C.-L. Fu, F. Yang, Identifying an unknown source term in a heat equation, Inverse Probl. Sci. Eng. 17 (7) (2009) 901–913.
[4] H. Engl, M. Hanke, A. Neubauer, Regularization of Inverse Problems, in: Mathematics and Its Applications, Springer Netherlands, 2000.
[5] S.I. Kabanikhin, Inverse and Ill-Posed Problems, de Gruyter, 2011.
[6] A. Kirsch, An Introduction to the Mathematical Theory of Inverse Problems, Vol. 120, Springer Nature, 2021.
[7] D. Lesnic, Inverse Problems with Applications in Science and Engineering, CRC Press, 2021.
[8] A. Fatullayev, Numerical solution of the inverse problem of determining an unknown source term in a heat equation, Math. Comput. Simulation 58 (3) (2002) 247–253.
[9] A. Fatullayev, Numerical solution of the inverse problem of determining an unknown source term in a two-dimensional heat equation, Appl. Math. Comput. 152 (3) (2004) 659–666.
[10] F.-F. Dou, C.-L. Fu, F.-L. Yang, Optimal error bound and Fourier regularization for identifying an unknown source in the heat equation, J. Comput. Appl. Math. 230 (2) (2009) 728–737.
[11] F. Yang, C.-L. Fu, X.-X. Li, A quasi-boundary value regularization method for determining the heat source, Math. Methods Appl. Sci. 37 (18) (2013) 3026–3035.
[12] F. Yang, C.-L. Fu, A simplified tikhonov regularization method for determining the heat source, Appl. Math. Model. 34 (11) (2010) 3286–3299.
[13] L. Yan, C.-L. Fu, F.-F. Dou, A computational method for identifying a spacewise-dependent heat source, Int. J. Numer. Methods Biomed. Eng. 26 (5) (2010) 597–608.
[14] G. Golub, M. Heath, G. Wahba, Generalized cross-validation as method for choosing a good ride parameter, Technometrics 2 (215–223) (1979).
[15] A. Farcas, D. Lesnic, The boundary-element method for the determination of a heat source dependent on one variable, J. Eng. Math. 54 (4) (2006) 375–388.
[16] L. Yan, C.-L. Fu, F.-L. Yang, The method of fundamental solutions for the inverse heat source problem, Eng. Anal. Bound. Elem. 32 (3) (2008) 216–222.
[17] M.N. Ahmadabadi, M. Arab, F.M. Ghaini, The method of fundamental solutions for the inverse space-dependent heat source problem, Eng. Anal. Bound. Elem. 33 (10) (2009) 1231–1235.
[18] L. Yan, F.-L. Yang, C.-L. Fu, A meshless method for solving an inverse spacewise-dependent heat source problem, J. Comput. Phys. 228 (1) (2009) 123–136.
[19] Z. Wang, W. Zhang, B. Wu, Regularized optimization method for determining the space-dependent source in a parabolic equation without iteration, J. Comput. Anal. Appl. 20 (1) (2016).
[20] B.T. Johansson, D. Lesnic, A variational method for identifying a spacewise-dependent heat source, IMA J. Appl. Math. 72 (6) (2007) 748–760.
[21] T. Johansson, D. Lesnic, Determination of a spacewise dependent heat source, J. Comput. Appl. Math. 209 (1) (2007) 66–80.
[22] L. Yang, M. Dehghan, J. Yu, G. Luo, Inverse problem of time-dependent heat sources numerical reconstruction, Math. Comput. Simul. 81 (8) (2011) 1656–1672.
[23] L. Yang, J. Yu, G. Luo, Z. Deng, Numerical identification of source terms for a two dimensional heat conduction problem in polar coordinate system, Appl. Math. Model. 37 (3) (2013) 939–957.
[24] Z. Yi, D. Murio, Source term identification in 1D IHCP, Comput. Math. Appl. 47 (12) (2004) 1921–1933.
[25] D. Trong, N. Long, P. Alain, Nonhomogeneous heat equation: Identification and regularization for the inhomogeneous term, J. Math. Anal. Appl. 312 (1) (2005) 93–104.
[26] D. Trong, P. Quan, P. Alain, Determination of a two-dimensional heat source: uniqueness, regularization and error estimate, J. Comput. Appl. Math. 191 (1) (2006) 50–67.
[27] Y. Ma, C. Fu, Y. Zhang, Identification of an unknown source depending on both time and space variables by a variational method, Appl. Math. Model. 36 (10) (2012) 5080–5090.
[28] M.J. Gander, 50 Years of time parallel time integration, in: Multiple Shooting and Time Domain Decomposition Methods, Springer, 2015, pp. 69–113.
[29] D.S. Daoud, Stability of the parareal time discretization for parabolic inverse problems, in: Domain Decomposition Methods in Science and Engineering XVI, Springer, 2007, pp. 275–282.
[30] J. Lee, D. Sheen, A parallel method for backward parabolic problems based on the Laplace transformation, SIAM J. Numer. Anal. 44 (4) (2006) 1466–1486.
[31] Y. Maday, E.M. Rønquist, Parallelization in time through tensor-product space-time solvers, C. R. Acad. Sci. Paris Sér. I Math. 346 (2008) 113–118.
[32] E. McDonald, J. Pestana, A. Wathen, Preconditioning and iterative solution of all-at-once systems for evolutionary partial differential equations, SIAM J. Sci. Comput. 40 (2018) A1012–A1033.
[33] M.J. Gander, L. Halpern, J. Rannou, J. Ryan, A direct time parallel solver by diagonalization for the wave equation, SIAM J. Sci. Comput. 41 (2019) A220–A245.
[34] S.-L. Wu, J. Liu, A parallel-in-time block-circulant preconditioner for optimal control of wave equations, SIAM J. Sci. Comput. 42 (3) (2020) A1510–A1540.
[35] J. Liu, S.L. Wu, A fast block α-circulant preconditoner for all-at-once systems from wave equations, SIAM J. Matrix Anal. Appl. 41 (2020) 1912–1943.
[36] J. Liu, Z. Wang, A ROM-accelerated parallel-in-time preconditioner for solving all-at-once systems in unsteady convection-diffusion PDEs, Appl. Math. Comput. 416 (2022) 126750.
[37] Y. Sun, S.-L. Wu, Y. Xu, A parallel-in-time implementation of the Numerov method for wave equations, J. Sci. Comput. 90 (20) (2022).
[38] T.I. Seidman, Optimal filtering for the backward heat equation, SIAM J. Numer. Anal. 33 (1) (1996) 162–170.
[39] U. Tautenhahn, T. Schröter, On optimal regularization methods for the backward heat equation, Z. Anal. Anwend. 15 (2) (1996) 475–493.
[40] J. Liu, M. Xiao, Quasi-boundary value methods for regularizing the backward parabolic equation under the optimal control framework, Inverse Problems 35 (12) (2019) 124003.

[41] Y. Jiang, J. Liu, Fast parallel-in-time quasi-boundary value methods for backward heat conduction problems, Appl. Numer. Math. 184 (2023) 325–339.
[42] B. Jin, W. Rundell, A tutorial on inverse problems for anomalous diffusion processes, Inverse Problems 31 (3) (2015) 035003.
[43] T. Wei, J.-G. Wang, A modified quasi-boundary value method for the backward time-fractional diffusion problem, ESAIM: Math. Model. Numer. Anal. 48 (2) (2014) 603–621.
[44] T. Wei, J. Wang, A modified quasi-boundary value method for an inverse source problem of the time-fractional diffusion equation, Appl. Numer. Math. 78 (2014) 95–111.
[45] F. Yang, C.-L. Fu, X.-X. Li, The inverse source problem for time-fractional diffusion equation: stability analysis and regularization, Inverse Probl. Sci. Eng. 23 (6) (2015) 969–996.
[46] H.T. Nguyen, D.L. Le, V.T. Nguyen, Regularized solution of an inverse source problem for a time fractional diffusion equation, Appl. Math. Model. 40 (19–20) (2016) 8244–8264.
[47] T. Wei, X. Li, Y. Li, An inverse time-dependent source problem for a time-fractional diffusion equation, Inverse Problems 32 (8) (2016) 085003.
[48] D.N. Hào, J. Liu, N.V. Duc, N.V. Thang, Stability results for backward time-fractional parabolic equations, Inverse Problems 35 (12) (2019) 125006.
[49] M. Ali, S. Aziz, S.A. Malik, Inverse source problems for a space–time fractional differential equation, Inverse Probl. Sci. Eng. 28 (1) (2020) 47–68.
[50] N.M. Dien, D.N.D. Hai, T.Q. Viet, D.D. Trong, On Tikhonov's method and optimal error bound for inverse source problem for a time-fractional diffusion equation, Comput. Math. Appl. 80 (1) (2020) 61–81.
[51] R. Ke, M.K. Ng, T. Wei, Efficient preconditioning for time fractional diffusion inverse source problems, SIAM J. Matrix Anal. Appl. 41 (4) (2020) 1857–1888.
[52] R. LeVeque, Finite Difference Methods for Ordinary and Partial Differential Equations: Steady-State and Time-Dependent Problems, SIAM, Philadelphia, PA, 2007.
[53] G. Caklovic, R. Speck, M. Frank, A parallel implementation of a diagonalization-based parallel-in-time integrator, 2021, arXiv preprint arXiv: 2103.12571.
[54] T. Wei, J. Wang, A modified quasi-boundary value method for an inverse source problem of the time-fractional diffusion equation, Appl. Numer. Math. 78 (2014) 95–111.
[55] L.I. Rudin, S. Osher, E. Fatemi, Nonlinear total variation based noise removal algorithms, Physica D 60 (1–4) (1992) 259–268.
[56] L. Wang, J. Liu, Total variation regularization for a backward time-fractional diffusion problem, Inverse Problems 29 (11) (2013) 115013.
[57] A. Chambolle, An algorithm for total variation minimization and applications, J. Math. Imaging Vision 20 (1) (2004) 89–97.
[58] B.T. Johansson, D. Lesnic, A procedure for determining a spacewise dependent heat source and the initial temperature, Appl. Anal. 87 (3) (2008) 265–276.
[59] Z. Wang, S. Qiu, Z. Ruan, W. Zhang, A regularized optimization method for identifying the space-dependent source and the initial value simultaneously in a parabolic equation, Comput. Math. Appl. 67 (7) (2014) 1345–1357.
[60] G.-H. Zheng, T. Wei, Recovering the source and initial value simultaneously in a parabolic equation, Inverse Problems 30 (6) (2014) 065013.
[61] B. Wang, B. Yang, M. Xu, Simultaneous identification of initial field and spatial heat source for heat conduction process by optimizations, Adv. Difference Equ. 2019 (1) (2019) 1–16.
[62] Z. Wang, S. Chen, S. Qiu, B. Wu, A non-iterative method for recovering the space-dependent source and the initial value simultaneously in a parabolic equation, J. Inverse Ill-Posed Probl. 28 (4) (2020) 499–516.