Efficient Preconditioning for Time Fractional Diffusion Inverse Source Problems

Rihuan Ke* Michael K. Ng† Ting Wei‡

September 28, 2020

Abstract

In this paper, we consider an inverse problem with quasi-boundary value regularization for recovering a source term of the time fractional diffusion equations from the final observation data. In particular, a two-by-two block linear system arising from the problem is studied. We propose a fast preconditioning technique by approximating the Schur complement in the system using a product of some factors, motivated by an approximate diagonalization of one of the blocks. The eigenvalues of the preconditioned system are shown to be clustered around 1, and the fast convergence of the methods is guaranteed theoretically. We also present an approach for selecting the regularization parameter of the quasi-boundary value method. Numerical experiments are carried out to demonstrate the effectiveness of our method.

1 Introduction

In the last few decades, time fractional and time-space fractional diffusion equations [37] have been successfully studied and employed in many applications, for example, fractional diffusion-wave equation [1], anomalous diffusion in statistical mechanisms, disordered media and physical applications [9, 20, 31], anomalous diffusion in a rotating flow [41].

Fast numerical methods have been investigated for solving fractional PDEs. Different from the integer order derivative cases, the discretization of the fractional order derivatives gives rise to systems with dense coefficients. Both iterative solvers and direct solvers, that take into account the structures of the discrete fractional derivatives, have been proposed for the discretized fractional PDE problems. Lei and Sun [26] propose a circulant preconditioner for space fractional diffusion equations (FDE) with variable diffusion coefficients. A preconditioning technique [35], built on an approximation of the diagonal-times-Toeplitz matrices, is developed for solving the FDE with a fast convergence rate. Donatelli et al. [14] propose two tridiagonal structure preserving preconditioners to approximate the matrices from the Meerschaert–Tadjeran discretization method. In [25], a fast direct method, based on a divide and conquer approach in the time domain, is applied to solving the time FDEs with time-varying diffusion coefficients. In this paper, we study the inverse source problem of FDEs and develop fast numerical methods based on the structure of the systems.

*Department of Applied Mathematics and Theoretical Physics, University of Cambridge. E-mail: krhuan@gmail.com

†Department of Mathematics, University of Hong Kong, Pokfulam, Hong Kong. M. Ng's research supported in part by the HKRGC GRF 12306616, 12200317, 12300218 and 12300519, and HKU 104005583. E-mail:mng@maths.hku.hk

‡School of Mathematics and Statistics, Lanzhou University.

Let Ω be an open bounded domain in $\mathbb{R}^d$ $(d = 1, 2)$. We consider the inverse problem of identifying the unknown source term component $f(x)$ from the time fractional diffusion equation (TFDE)

$$\begin{aligned} &D_t^\alpha u(x,t) = (Lu)(x,t) + f(x)q(t), && x \in \Omega,\ t \in (0,T), \\ &u(x,t) = 0, && x \in \partial\Omega,\ t \in (0,T), \\ &u(x,0) = 0, && x \in \Omega, \end{aligned} \tag{1}$$

with $\alpha \in (0,1)$, known time-dependent term $q(t) > 0$, and the additional data

$$u(x,T) = g(x), \quad x \in \bar{\Omega}. \tag{2}$$

In (1), D_t^α denotes the fractional differential operator of order α, and $-L$ is an elliptic operator. Different versions of fractional derivatives exist in the literature, see for instance [40, 20, 22, 31, 37]. Here the time derivative is the left Caputo fractional derivative given by

$$(D_t^\alpha u)(x,t) = \frac{1}{\Gamma(1-\alpha)} \int_0^t \frac{u_\tau(x,\tau)}{(t-\tau)^\alpha} d\tau,$$

in which $0 \le t \le T$, Γ is Euler's Gamma function, and $u_\tau(x,\tau)$ denotes the first order derivative of u with respect to the time τ. If D_t^α is replaced by the first order time derivative, then (1) becomes the classical diffusion equation. The operator L takes the form

$$(Lu)(x,t) = \sum_{i=1}^d \frac{\partial}{\partial x_i} \left(\sum_{j=1}^d a_{ij}(x) \frac{\partial}{\partial x_j} u(x,t) \right) + c(x)u(x,t),\ x \in \Omega, \tag{3}$$

in which the coefficients satisfy

$$\begin{cases} a_{ij} \in C^1\big(\bar{\Omega}\big), \quad a_{ij} = a_{ji}, & i,j = 1, \cdots, d, \\ \sum_{i,j=1}^d a_{ij}(x) y_i y_j \ge a_{\min} \sum_{i=1}^d y_i^2, & x \in \bar{\Omega},\ y_1, \cdots, y_d \in \mathbb{R}, \\ c \in C\big(\bar{\Omega}\big), \quad c(x) \le 0, & x \in \bar{\Omega}. \end{cases}$$

and $a_{\min} > 0$ is a constant.

1.1 Regularization for the Inverse Problems

Like many inverse problems for the classical diffusion equations or other fractional diffusion equations, the problem of finding $f(x)$ from (1)-(2) is found to be ill-posed (see, e.g., [52, 24]). It is a well known fact that, in such a problem, a naive solution computed by a direct inverse of the operator can be rather unreliable when the data is perturbed by small noise. Unfortunately, noise almost always exists in real world applications. To determine $f(x)$ in better accuracy, several regularization techniques have been developed. For instance, the truncation method [52], the Tikhonov methods [33], the simplified Tikhonov regularization method [45], generalized Tikhonov methods [30], the quasi-reversibility method [43], and the reproducing kernel space method with truncating [46].

The quasi-boundary value method (QBVM), also referred to as the non-local boundary value method, has been widely used in solving ill-posed inverse problems for parabolic equations, see e.g.,

[2, 13, 11, 19] and the references therein. Recently, it has been successfully generalized for solving the fractional diffusion inverse problems [42, 48, 49, 50]. Instead of working on (1) and (2), the method deals with a well-posed problem

$$\begin{aligned} &D_t^\alpha v(x,t) = (Lv)(x,t) + f_\mu(x)q(t), && x \in \Omega, \quad t \in (0,T), \\ &v(x,t) = 0, && x \in \partial\Omega, \quad t \in (0,T), \\ &v(x,0) = 0, && x \in \Omega \\ &v(x,T) = g(x) - \mu(Rf_\mu)(x), && x \in \bar{\Omega} \end{aligned} \tag{4}$$

in which $\mu > 0$ is a regularization parameter, and R is an elliptic operator that can be chosen as the identity mapping, $-L$, etc. Note that if $\mu = 0$, then it reduces to the original problem (1)-(2). The solutions v and f_μ of the well-posed problem (4) are viewed as approximations to u and f respectively.

1.2 Notations

Throughout this paper, boldface lowercase letters $\boldsymbol{a}$, $\boldsymbol{b}$, $\boldsymbol{c}$, $\cdots$ represent vectors, and boldface uppercase letters $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\cdots$ represent matrices. Matrix functions will be denoted by calligraphic letters (e.g., $\mathcal{S}(\mathbf{A})$ is a function of $\mathbf{A}$). We let $\mathbb{R}^{n\times m}$ and $\mathbb{R}^n$ denote the set of $n \times m$ matrices and the set of n-vectors, respectively. For a given real number x, we use $\lceil x \rceil$ to denote the smallest integer not less than x, and use $\lfloor x \rfloor$ to denote the largest number not bigger than x.

The superscript T denotes the transpose of a matrix or a vector (e.g., $\mathbf{A}^T$ and $\boldsymbol{a}^T$). We use $\mathbf{A} \otimes \mathbf{B}$ to denote the Kronecker product of matrices $\mathbf{A}$ and $\mathbf{B}$. We use $\mathrm{diag}(a_1, a_2, \cdots, a_k)$ to denote the diagonal matrix with diagonal entries $a_1, a_2, \cdots, a_k$ and use $\mathrm{diag}(\mathbf{A}_1, \mathbf{A}_2, \cdots, \mathbf{A}_k)$ to denote the block diagonal matrix whose diagonal blocks are $\mathbf{A}_1$, $\mathbf{A}_2$, $\cdots$, $\mathbf{A}_k$. Given a matrix $\mathbf{A}$, $\|\mathbf{A}\|_\infty$, $\|\mathbf{A}\|_2$, and $\|\mathbf{A}\|_1$ denote the infinity norm, the 2-norm, and the 1-norm of $\mathbf{A}$ respectively. We use the boldface letter $\mathbf{I}$ to denote the identity matrix and use a subscript to specify its number of rows (e.g., $\mathbf{I}_m$ is the identity matrix of size $m \times m$). Similarly, we use $\mathbf{F}_m$ to denote the Fourier transform matrix of size $m \times m$.

1.3 Structured Linear Systems

The discretization of the problem (4) will be discussed in Section 2. To solve (4), the discretization gives rise to a linear system of the form

$$\underbrace{\begin{pmatrix} \mathbf{B}_{11} & \mathbf{B}_{12} \\ \mathbf{B}_{21} & \mathbf{B}_{22} \end{pmatrix}}_{\mathbf{B}} \begin{pmatrix} \boldsymbol{v} \\ \boldsymbol{f}_\mu \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \boldsymbol{g} \end{pmatrix} \tag{5}$$

where $\boldsymbol{v}$, $\boldsymbol{f}_\mu$, and $\boldsymbol{g}$ are discrete versions of v, f_μ, and g respectively, $\mathbf{0}$ is a zero vector, both $\mathbf{B}_{11}$ and $\mathbf{B}_{22}$ are square and of full rank, and $\mathbf{B}_{21}$ and $\mathbf{B}_{12}$ are sparse. The matrix $\mathbf{B}_{22}$, corresponding to the regularization operator μR, is symmetric positive definite (SPD), while $\mathbf{B}_{11}$, a discrete operator from the forward fractional diffusion problem, is a nonsymmetric nonsingular matrix with block Toeplitz structure. If the numbers of grid points in the spatial domain and the time domain are n and m respectively, then $\mathbf{B}_{11} \in \mathbb{R}^{nm\times nm}$ and $\mathbf{B}_{22} \in \mathbb{R}^{n\times n}$, thus the size of $\mathbf{B}_{11}$ is larger than that of $\mathbf{B}_{22}$. Unlike other PDE related problems, $\mathbf{B}_{11}$ does not possess an extremely sparse structure because the fractional derivatives are nonlocal. A more detailed discussion on the structures of the

sub-matrices will be presented in Section 2. Clearly, the coupling of $\boldsymbol{v}$ and $\boldsymbol{f}_\mu$ makes the inverse source problem more difficult than its direct version. Though much attention has been given to the efficient algorithms for the time fractional diffusion equations in the last decades (see e.g., [25, 29, 23, 16]), little work has been done on fast computational methods for the associated inverse problems in the literature. In this work, we explore the structure of (5) and propose fast numerical solutions for the inverse problem.

The system (5) is highly structured and it can be stored efficiently in the computer memory. However, if we solve (5) using direct solvers such as block LU factorization, the structure is not preserved due to the fill-in of the matrices. In fact, both $\mathbf{B}_{11}^{-1}$ and $\mathbf{B}_{22}^{-1}$ are dense matrices, and the computation of block factorization will lead to dense matrices and have high memory requirements and time complexity. Therefore, these methods are not feasible for large problems. Iterative methods, on the other hand, can preserve the structure of the system and are hence preferred. Though the discretization of the time derivative operator with order $\alpha \in (0, 1)$ has dense coefficients (encoded in the matrix $\mathbf{B}_{11}$), we emphasize that the associated matrix-vector product can be carried out fast thanks to its Toeplitz structure. Thus, iterative methods such as Krylov subspace methods have a low cost at each iteration. Nevertheless, iterative methods often converge very slowly for this problem. This motivates us to develop preconditioners to exploit the structure of (5) and speed up the convergence.

The system (5) belongs to the generalized saddle point problems [4]. Many applications lead to sparse saddle point matrices, and thus iterative methods are often used for large systems. Preconditioning is essential when these methods are applied. One type of preconditioner for $\mathbf{B}$ in (5) is based on matrix splitting. As an example, the Hermitian and skew-Hermitian splitting (HSS) [3] yields the preconditioner $\mathbf{P}_s = (\gamma\mathbf{I} + \mathbf{H}_s)(\gamma\mathbf{I} + \mathbf{K}_s)$, where γ is a parameter, $\mathbf{H}_s := (\mathbf{B} + \mathbf{B}^T)/2$ is symmetric, and $\mathbf{K}_s := (\mathbf{B} - \mathbf{B}^T)/2$ is skew-symmetric. For the saddle point problems, the Schur-complement-based block preconditioners have also been developed. One typical choice is the block diagonal preconditioner $\mathbf{P}_d := \mathrm{diag}(\mathbf{B}_{11}, -\mathbf{S})$, where $\mathbf{S} := \mathbf{B}_{22} - \mathbf{B}_{21}\mathbf{B}_{11}^{-1}\mathbf{B}_{12}$ is the Schur complement of $\mathbf{B}_{11}$ in $\mathbf{B}$. If $\mathbf{B}_{22}$ is zero and the size $\mathbf{B}_{11}$ is not smaller than $\mathbf{B}_{22}$, then the GMRES applied to the preconditioned system will converge in three iterations (see [32] and Section 10 of [4]). Block triangular preconditioners, such as $\mathbf{P}_t := \begin{pmatrix} \mathbf{B}_{11} & \mathbf{B}_{12} \\ \mathbf{O} & -\mathbf{S} \end{pmatrix}$, have been also investigated [15, 5, 47]. It was shown that when preconditioned with $\mathbf{P}_t$, the GMRES method always converges in two steps [21, 4]. Though $\mathbf{P}_d$ and $\mathbf{P}_t$ have ideal theoretical convergence behaviour, they may not be practical because the computation of the Schur complement $\mathbf{S}^{-1}$ is not easy generally. As remedies for this, block triangular preconditioners based on approximate Schur complements are proposed [4]. In general, the approximation of the Schur complement is a crucial issue in the design of the block preconditioners (see e.g., [6, 47]), whereas a good Schur complement approximation is usually problem-dependent. A comparison of block preconditioners can be found in [34]. We refer the readers to [4, 47] for further introductions of the (generalized) saddle point problems and the associated computational methods.

1.4 The Contribution

In this paper, we study effective preconditioning techniques for determining the space-dependent component of the source term for the TFDE inverse problem which has the discretization (5). The techniques being developed can be extended to solving other closely related inverse problems, such as the backward time fractional diffusion problems with quasi-boundary value regularization

[48, 44], that exhibit similar structured linear systems as in (5).

We tackle the non-symmetric system (5) by investigating its block structure and approximating the Schur complement of $\mathbf{B}_{11}$ with a factorization. A preconditioner, which belongs to the category of block preconditioners, is proposed for the system. The preconditioning is efficient as the main computational cost is on inverting the matrix $\mathbf{B}_{11}$ for which fast computational methods exist, and the preconditioned system has spectrum clustered around 1. The convergence rate of Krylov subspace methods including GMRES is known to not completely be governed by the spectrum of the matrix when it is non-symmetric [17, 39]. With the careful construction of the preconditioner we show that the preconditioned system shares the same Krylov subspaces with a symmetric system, and hence the fast convergence behavior is guaranteed theoretically, thanks to the block factorization and the symmetric positive definiteness of the Schur complement. With the effectiveness in the computational time, the method is capable of solving the large scale problems (discretized in space and time), and outperforms the classical iterative methods by large margins. We also discuss the variants of the preconditioner that are based on inexact solvers of several subproblems and evaluate their performance under various parameter settings.

The rest of the paper is organized as follows. In Section 2, we give a discretization of the problem and establish the preconditioner. Section 3 includes theoretical analysis for the preconditioner and shows the fast convergence rate of the iterative method. In Section 4, numerical experiments are provided to demonstrate the efficiency of the methods. We also present a method for selecting the regularization parameters. Finally, concluding remarks are given in Section 5.

2 The Proposed Method

We start by deriving a discrete version of the problem. For simplicity, let us first consider a simple case where $d = 1$ and $\Omega = (0, 1)$, and the definition of L in (3) is rewritten as

$$(Lu)(x,t) = \frac{\partial}{\partial x}\left(a(x)\frac{\partial}{\partial x}u(x,t)\right) + c(x)u(x,t),$$

and the coefficient satisfies

$$a \in C^1\big(\bar{\Omega}\big),\ c \in C\big(\bar{\Omega}\big),\ \text{and } a(x) \geq a_{\min} > 0,\ c(x) \leq 0, \forall x \in \bar{\Omega}. \tag{6}$$

We divide $\bar{\Omega}$ into $n+1$ parts evenly and $[0, T]$ into m uniform time units, i.e., the spatial grid spacing and time grid spacing are taken as $\Delta x = \frac{1}{n+1}$ and $\Delta t = \frac{T}{m}$ respectively. The grid points on the x axis and on the time axis are denoted by $\{x_i \mid x_i = i(\Delta x),\ i = 0, 1, \cdots, n+1\}$ and $\{t_j \mid t_j = j(\Delta t),\ j = 0, 1, \cdots, m\}$ respectively. Let $a_{i+\frac{1}{2}} = a(x_i + \Delta x/2)$ and $c_i = c(x_i)$. With the second order central difference scheme, the operator $-L$ can be discretized as the following tridiagonal matrix

$$\mathbf{H} = \frac{1}{(\Delta x)^2}\begin{pmatrix} h_{1,1} & h_{1,2} & & \\ h_{2,1} & h_{2,2} & \ddots & \\ & \ddots & \ddots & h_{n-1,n} \\ & & h_{n,n-1} & h_{n,n} \end{pmatrix}, \tag{7}$$

in which $h_{i,i} = a_{i+\frac{1}{2}} + a_{i-\frac{1}{2}} - c_i$, $h_{i,i+1} = h_{i+1,i} = -a_{i+\frac{1}{2}}$ for $i \geq 1$, and $h_{i,j} = 0$ for $|i-j| > 1$. Based on (6), we have $a_{i+\frac{1}{2}} > 0$ and $c_j \leq 0$, and it is can be seen that

$$\sum_{j=1}^{n} h_{i,j} \geq 0, \quad h_{i,i} \geq -\sum_{j\neq i} h_{i,j} = \sum_{j\neq i} |h_{i,j}|, \quad \text{for } i = 1, 2, \cdots, n. \tag{8}$$

So the discrete version $\mathbf{H}$ of the elliptic operator $-L$ is symmetric positive definite (SPD). Other discretization methods, such as finite element methods, can lead to SPD operators like $\mathbf{H}$.

The fractional time derivative can also be discretized by a finite difference scheme. Let

$$\beta_j = \frac{(\Delta t)^{-\alpha}}{\Gamma(2-\alpha)}(w_{j+1} - w_j) \quad \text{and} \quad w_{j+1} = (j+1)^{1-\alpha} - j^{1-\alpha} \tag{9}$$

for $j = 0, 1, \cdots, m$, and $w_0 = 0$. Then one has the approximation of the time derivative (see e.g., [51, 28, 49])

$$D_t^{\alpha} v(x, t_k) \approx \sum_{j=0}^{k} \beta_{k-j} v(x, t_j), \quad k = 1, 2, \cdots, m. \tag{10}$$

The discretization parameters β_j and w_j have the following properties.

Lemma 1 ([51]). *The sequence $\{w_j \mid j = 1, 2, \cdots\}$ satisfies*

$$1 = w_1 > w_2 > \cdots w_k > 0, \quad \forall k > 0, \quad \text{and} \quad \lim_{k\to\infty} w_k = 0.$$

Consequently, $\beta_0 > 0$ and $\beta_j < 0$ for $j = 1, 2, \cdots$, and it holds that

$$|\beta_0| > \sum_{j=1}^{k} |\beta_j|, \quad \forall k \geq 1, \quad \text{and} \quad \lim_{k\to\infty} \sum_{j=0}^{k} \beta_j = \lim_{k\to\infty} \beta_0 w_{k+1} = 0.$$

In the following discussions, we fix R in (4) with $R := -L$ which has been investigated by Wei et al. [49]. The extension of the results to other cases, e.g., where the operator R is the identity mapping, is possible. If we let $\boldsymbol{f}_\mu = [f_\mu(x_1), f_\mu(x_2), \cdots, f_\mu(x_n)]^T$, $\boldsymbol{v}_k = [v(x_1, t_k), v(x_2, t_k), \cdots, v(x_n, t_k)]^T$, and $\boldsymbol{g} = [g(x_1), g(x_2), \cdots, g(x_n)]^T$, the problem (4) is then approximated by the numerical scheme

$$\mathbf{A}\boldsymbol{v}_k + \sum_{j=1}^{k-1} \beta_{k-j} \boldsymbol{v}_j - q_k \boldsymbol{f}_\mu = \mathbf{0}, \quad k = 1, \cdots, m, \tag{11}$$

$$\boldsymbol{v}_m + \mu \mathbf{H} \boldsymbol{f}_\mu = \boldsymbol{g}, \tag{12}$$

in which $\mathbf{A} = \mathbf{H} + \beta_0 \mathbf{I} \in \mathbb{R}^{n\times n}$, and $q_k = q(t_k) > 0$. In particular, if $a(x)$ is constant and $c(x) = 0$, then (11), as a discrete TFDE, is identical to the implicit finite difference scheme proposed by Zhuang et al. [51], which has been proven to be unconditionally stable.

Writing (11) and (12) into an all-in-one form, we get the structured system

$$\left(\begin{array}{ccccc|c} \mathbf{A} & & & & & -q_1\mathbf{I} \\ \beta_1\mathbf{I} & \mathbf{A} & & & & -q_2\mathbf{I} \\ \beta_2\mathbf{I} & \beta_1\mathbf{I} & \mathbf{A} & & & -q_3\mathbf{I} \\ \vdots & \ddots & \ddots & \ddots & & \vdots \\ \beta_{m-1}\mathbf{I} & \cdots & \beta_2\mathbf{I} & \beta_1\mathbf{I} & \mathbf{A} & -q_m\mathbf{I} \\ \hline \mathbf{O} & \cdots & \mathbf{O} & \mathbf{O} & \mathbf{I} & \mu\mathbf{H} \end{array}\right) \left(\begin{array}{c} \boldsymbol{v}_1 \\ \boldsymbol{v}_2 \\ \boldsymbol{v}_3 \\ \vdots \\ \boldsymbol{v}_m \\ \hline \boldsymbol{f}_\mu \end{array}\right) = \left(\begin{array}{c} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \hline \boldsymbol{g} \end{array}\right), \tag{13}$$

where $\mathbf{I}$ denotes the identity matrix of size $n \times n$ and $\mathbf{O}$ is the matrix of all zeros. Set

$$\boldsymbol{e}_m := [0, \cdots, 0, 1]^T \in \mathbb{R}^m, \text{ and } \boldsymbol{q} := [q_1, q_2, \cdots, q_m]^T \in \mathbb{R}^m. \tag{14}$$

Then the bottom-left block and the top-right block of the coefficient matrix in (13) are given as $\mathbf{B}_{21} := \boldsymbol{e}_m^T \otimes \mathbf{I}$, and $\mathbf{B}_{12} := -\boldsymbol{q} \otimes \mathbf{I}$, respectively. Furthermore, we write the diagonal blocks of (13) as

$$\mathbf{B}_{11} := \begin{pmatrix} \mathbf{A} & & & \\ \beta_1\mathbf{I} & \mathbf{A} & & \\ \vdots & \ddots & \ddots & \\ \beta_{m-1}\mathbf{I} & \cdots & \beta_1\mathbf{I} & \mathbf{A} \end{pmatrix}, \quad \text{and } \mathbf{B}_{22} := \mu\mathbf{H}. \tag{15}$$

Let $\boldsymbol{x}$ be the column stacking of $[\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots, \boldsymbol{v}_m, \boldsymbol{f}_\mu]$ and $\boldsymbol{b}$ be the column stacking of $[\mathbf{0}, \cdots, \mathbf{0}, \boldsymbol{g}]$. With these notations we have a compact form of (13),

$$\underbrace{\begin{pmatrix} \mathbf{B}_{11} & \mathbf{B}_{12} \\ \mathbf{B}_{21} & \mathbf{B}_{22} \end{pmatrix}}_{\mathbf{B}} \boldsymbol{x} = \boldsymbol{b}, \tag{16}$$

where $\boldsymbol{x}$ is the quantity of interest.

Since $\mathbf{B}_{11}$ has a block-Toeplitz structure and the sub-matrices $\mathbf{H}$, $\mathbf{A}$ and $\mathbf{I}$ are sparse, the matrix-vector multiplication for $\mathbf{B}$ can be carried out efficiently. In the following, we propose efficient preconditioners for $\mathbf{B}$. We start by introducing a block δ-circulant approximation to $\mathbf{B}_{11}$ in Subsection 2.1. Then with the block δ-circulant approximation, in Subsection 2.2, we construct an approximation of the Schur complement of $\mathbf{B}_{11}$ using a low dimensional space of rational functions. A preconditioner based on this approximation is proposed. Finally, we discuss the Krylov subspaces of the preconditioned system in Subsection 2.3.

2.1 Circulant Approximation

Circulant matrices and δ-circulant matrices [7] are often used to approximate Toeplitz matrices for fast computation. In particular, the inverses of triangular Toeplitz matrices can be computed approximately and in parallel using δ-circulant matrices [7, 27, 36]. Given the block lower-triangular Toeplitz structure of $\mathbf{B}_{11}$ in Equation (15), we approximate it with a block δ-circulant matrix defined by

$$\mathbf{B}_{\delta,11} = \begin{pmatrix} \mathbf{A} & \delta\beta_{m-1}\mathbf{I} & \cdots & \delta\beta_2\mathbf{I} & \delta\beta_1\mathbf{I} \\ \beta_1\mathbf{I} & \mathbf{A} & \cdots & \delta\beta_3\mathbf{I} & \delta\beta_2\mathbf{I} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \beta_{m-2}\mathbf{I} & \beta_{m-3}\mathbf{I} & \cdots & \mathbf{A} & \delta\beta_{m-1}\mathbf{I} \\ \beta_{m-1}\mathbf{I} & \beta_{m-2}\mathbf{I} & \cdots & \beta_1\mathbf{I} & \mathbf{A} \end{pmatrix}, \tag{17}$$

where δ is a small positive number. Clearly $\lim_{\delta\to 0} \|\mathbf{B}_{\delta,11} - \mathbf{B}_{11}\|_\infty = 0$. The block δ-circulant matrices have been investigated for fast solutions for block triangular Toeplitz systems [29] and the exponential of large block triangular Toeplitz matrices [8].

Recall that the diagonal blocks are $\mathbf{A} = \mathbf{H} + \beta_0\mathbf{I}$. One advantage of such an approximation is that, with the block δ-circulant structure, the matrix $\mathbf{B}_{\delta,11}$ has a closed form block decomposition

$$\mathbf{B}_{\delta,11} = \big(\mathbf{D}_\delta^{-1}\mathbf{F}_m^{-1} \otimes \mathbf{I}\big)\mathrm{diag}(\boldsymbol{\Sigma}_1, \boldsymbol{\Sigma}_2, \cdots, \boldsymbol{\Sigma}_m)(\mathbf{F}_m\mathbf{D}_\delta \otimes \mathbf{I}), \tag{18}$$

in which $\mathbf{F}_m$ denotes the discrete Fourier transform matrix of size $m \times m$, and

$$\mathbf{D}_\delta = \operatorname{diag}\Big(1, \delta^{1/m}, \cdots, \delta^{(m-1)/m}\Big), \tag{19a}$$

$$[\sigma_1, \sigma_2, \cdots, \sigma_m]^T = \mathbf{F}_m \mathbf{D}_\delta [\beta_0, \beta_1, \cdots, \beta_{m-1}]^T, \tag{19b}$$

$$\mathbf{\Sigma}_i = \sigma_i \mathbf{I} + \mathbf{H}. \tag{19c}$$

This block decomposition means that $\mathbf{B}_{\delta,11}$ can be inverted efficiently since there exist fast algorithms for computing $\mathbf{\Sigma}_i^{-1}$. The invertibility and a block-wise property of $\mathbf{B}_{\delta,11}$ are described in the following lemma.

Lemma 2. *For any $\delta \in [0,1]$, the matrices $\mathbf{B}_{\delta,11}$ and $\mathbf{\Sigma}_i$ $(i = 1, 2, \cdots, m)$ are invertible. Moreover, each of the $m \times m$ blocks of $\mathbf{B}_{\delta,11}^{-1}$ is symmetric positive definite, and they can be simultaneously diagonalized.*

Proof. Assume that $\delta \in [0,1]$. According to the definition of $\mathbf{H}$ in (7) and the inequalities in (8), $\mathbf{H}$ is diagonally dominant. This together with the fact that $\mathbf{A} = \beta_0 \mathbf{I} + \mathbf{H}$ and $\beta_0 > \sum_{k=1}^m |\beta_k|$ (see Lemma 1) shows that $\mathbf{B}_{\delta,11}$ is strictly diagonally dominant and hence nonsingular. As a consequence, $\mathbf{\Sigma}_i$ $(i = 1, 2, \cdots, m)$ are nonsingular.

To study the properties of the blocks of $\mathbf{B}_{\delta,11}^{-1}$, define

$$\mathbf{Z}_\delta = -\begin{pmatrix} 0 & \delta\beta_{m-1} & \cdots & \delta\beta_1 \\ \beta_1 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \delta\beta_{m-1} \\ \beta_{m-1} & \cdots & \beta_1 & 0 \end{pmatrix}, \tag{20}$$

then $\mathbf{B}_{\delta,11} = \mathbf{I}_m \otimes \mathbf{A} - \mathbf{Z}_\delta \otimes \mathbf{I}$ with $\mathbf{I}_m$ being the identity matrix of size $m \times m$. The inverse of $\mathbf{B}_{\delta,11}$ has an explicit form

$$\mathbf{B}_{\delta,11}^{-1} = \big(\mathbf{I}_m \otimes \mathbf{A}^{-1}\big) \sum_{i=0}^{\infty} \big[(\mathbf{Z}_\delta \otimes \mathbf{I})\big(\mathbf{I}_m \otimes \mathbf{A}^{-1}\big)\big]^i = \sum_{i=0}^{\infty} \mathbf{Z}_\delta^i \otimes \mathbf{A}^{-1-i}, \tag{21}$$

in which $\sum_{i=0}^{\infty} \big\|\mathbf{Z}_\delta^i \otimes \mathbf{A}^{-1-i}\big\|_\infty \le \sum_{i=0}^{\infty} \|\mathbf{Z}_\delta\|_\infty^i \big\|\mathbf{A}^{-1}\big\|_\infty^{i+1} \le \sum_{i=0}^{\infty} \|\mathbf{Z}_\delta\|_\infty^i (1/\beta_0)^{i+1} < \infty$. Since $\mathbf{A} = \beta_0 \mathbf{I} + \mathbf{H}$ is symmetric and strictly diagonally dominant, it is symmetric positive definite (SPD), and hence $\mathbf{A}^{-1}$ is also SPD. Besides, by Lemma 1, $\mathbf{Z}_\delta$ is a nonnegative matrix, hence $\mathbf{Z}_\delta^i$ is nonnegative and its $(k,l)^{\text{th}}$ entry $\big(\mathbf{Z}_\delta^i\big)_{kl}$ is nonnegative. Then following from Equation (21), the $(k,l)^{\text{th}}$ block of $\mathbf{B}_{\delta,11}^{-1}$ is equal to $\sum_{i=0}^{\infty} \big(\mathbf{Z}_\delta^i\big)_{kl} \mathbf{A}^{-1-i}$ which is SPD, for $k, l = 1, 2, \cdots, m$. Finally, since $\mathbf{A} = \beta_0 \mathbf{I} + \mathbf{H}$, the matrices $\mathbf{H}$ and $\mathbf{A}$ are simultaneously diagonalizable, which together with Equation (21) implies that $\mathbf{H}$, $\mathbf{A}^{-1}$ and the blocks of $\mathbf{B}_{\delta,11}^{-1}$ are simultaneously diagonalizable. □

Taking $\delta = 0$, one has $\mathbf{B}_{11} = \mathbf{B}_{\delta,11}$, so the properties given in Lemma 2 hold for $\mathbf{B}_{11}$. Equation (21) in the proof of Lemma 2 also confirms that

$$\lim_{\delta \to 0} \Big\|\mathbf{B}_{\delta,11}^{-1}\Big\|_\infty < \infty, \quad \text{and} \quad \lim_{\delta \to 0} \Big\|\mathbf{B}_{11}^{-1} - \mathbf{B}_{\delta,11}^{-1}\Big\|_\infty = 0. \tag{22}$$

2.2 The Proposed Preconditioner

The matrix $\mathbf{B}_{11}^{-1}$ in (15) can be well approximated by $\mathbf{B}_{\delta,11}^{-1}$ in (17) when $\delta > 0$ is small according to Equation (22). Besides, $\mathbf{B}_{\delta,11}^{-1}$ is easy to compute because $\mathbf{\Sigma}_i$ $(i = 1, \cdots, m)$ can be inverted easily. So

$$\mathbf{B}_\delta := \begin{pmatrix} \mathbf{B}_{\delta,11} & \mathbf{B}_{12} \\ \mathbf{B}_{21} & \mathbf{B}_{22} \end{pmatrix} \tag{23}$$

forms a preconditioner for the matrix $\mathbf{B}$ defined in (16). However, such a preconditioner could not be readily used for practical applications, as inverting the 2×2 block matrix $\mathbf{B}_\delta$ requires solving the inverse of the Schur complement of $\mathbf{B}_{\delta,11}$ which is dense. In the following, we further exploit the structure of the blocks of $\mathbf{B}_\delta$ to get an efficient preconditioner. The idea is to decompose $\mathbf{B}_\delta$ and approximate the associated Schur complement with a class of low degree rational functions whose inverses are easy to obtain.

Firstly, we convert $\mathbf{B}_\delta$ into an arrowhead matrix by diagonalizing its first block. With the decomposition of $\mathbf{B}_{\delta,11}$ in (18), the transform $\mathrm{diag}(\mathbf{F}_m\mathbf{D}_\delta \otimes \mathbf{I}, \mathbf{I})\mathbf{B}_\delta \mathrm{diag}(\mathbf{F}_m\mathbf{D}_\delta \otimes \mathbf{I}, \mathbf{I})^{-1}$ takes the following form

$$\left(\begin{array}{cccc|c} \mathbf{\Sigma}_1 & & & & \widehat{q}_1\mathbf{I} \\ & \mathbf{\Sigma}_2 & & & \widehat{q}_2\mathbf{I} \\ & & \ddots & & \vdots \\ & & & \mathbf{\Sigma}_m & \widehat{q}_m\mathbf{I} \\ \hline \widehat{\gamma}_1\mathbf{I} & \widehat{\gamma}_2\mathbf{I} & \cdots & \widehat{\gamma}_m\mathbf{I} & \mu\mathbf{H} \end{array}\right) \tag{24}$$

where $\widehat{q}_i$ and $\widehat{\gamma}_i$ $(i = 1, 2, \cdots, m)$ are defined respectively by

$$[\widehat{q}_1, \widehat{q}_2, \cdots, \widehat{q}_m]^T := -\mathbf{F}_m\mathbf{D}_\delta\boldsymbol{q} \quad \text{and} \quad [\widehat{\gamma}_1, \widehat{\gamma}_2, \cdots, \widehat{\gamma}_m] := \boldsymbol{e}_m^T\mathbf{D}_\delta^{-1}\mathbf{F}_m^{-1}, \tag{25}$$

$\boldsymbol{q}$ and $\boldsymbol{e}_m$ are given in (14), and $\mathbf{\Sigma}_i = \sigma_i\mathbf{I} + \mathbf{H}$ as defined in (19c).

Secondly, we formulate the Schur complement of the top-left block in $\mathbf{B}_\delta$ as a function of $\mathbf{H}$. Let $\mathbf{S}_\delta := \mathbf{B}_{22} - \mathbf{B}_{21}\mathbf{B}_{\delta,11}^{-1}\mathbf{B}_{12}$ be the Schur complement of $\mathbf{B}_{\delta,11}$ in $\mathbf{B}_\delta$, then it can be shown that $\mathbf{S}_\delta$ is equivalent to the Schur complement of the top-left block $\mathrm{diag}(\mathbf{\Sigma}_1, \cdots, \mathbf{\Sigma}_m)$ in the matrix (24). Based on this observation, for any $\delta > 0$, $\mathbf{S}_\delta$ has the following equivalent formulation

$$\mathbf{S}_\delta = \mathcal{S}(\mathbf{H}) := \mu\mathbf{H} - \sum_{i=1}^{m} \widehat{\gamma}_i\widehat{q}_i(\sigma_i\mathbf{I} + \mathbf{H})^{-1} \tag{26}$$

which is a function of $\mathbf{H}$. The Schur complement $\mathbf{S}_\delta$ is an SPD matrix as stated in the following lemma.

Lemma 3. *For any $\delta \in [0, 1]$, the matrix $\mathbf{S}_\delta$ is symmetric positive definite and is simultaneously diagonalizable with $\mathbf{H}$. Moreover, for $\delta > 0$, the function $\mathcal{S}(\cdot)$ is positive valued on $(0, \infty)$.*

Proof. According to the definition of $\mathbf{S}_\delta$ and Equation (21),

$$\begin{aligned} \mathbf{S}_\delta = \mathbf{B}_{22} - \mathbf{B}_{21}\mathbf{B}_{\delta,11}^{-1}\mathbf{B}_{12} &= \mathbf{B}_{22} + (\boldsymbol{e}_m^T \otimes \mathbf{I})\left(\sum_{i=0}^{\infty} \mathbf{Z}_\delta^i \otimes \mathbf{A}^{-1-i}\right)(\boldsymbol{q} \otimes \mathbf{I}) \\ &= \mathbf{B}_{22} + \sum_{i=0}^{\infty} (\boldsymbol{e}_m^T\mathbf{Z}_\delta^i\boldsymbol{q}) \otimes \mathbf{A}^{-1-i}, \end{aligned}$$

where $\mathbf{Z}_\delta$ is given by (20). So $\mathbf{S}_\delta$ is real and symmetric, and $\mathbf{S}_\delta$ and $\mathbf{H}$ are simultaneously diagonalizable. Observe that $\boldsymbol{e}_m$, $\boldsymbol{q}$, and $\mathbf{Z}_\delta$ are non-negative. The positive definiteness of $\mathbf{S}_\delta$ follows from the fact that both $\mathbf{B}_{22}$ and $\mathbf{A}^{-1}$ are SPD. Similarly, for $\delta > 0$, given any $\lambda > 0$, replacing $\mathbf{H}$ by $\lambda\mathbf{I}$ in the above proof, it can be shown that

$$\mathcal{S}(\lambda)\mathbf{I} = \mathcal{S}(\lambda\mathbf{I}) = \mu\lambda\mathbf{I} + \sum_{i=0}^{\infty} \left(\boldsymbol{e}_m^T \mathbf{Z}_\delta^i \boldsymbol{q}\right) \otimes (\beta_0\mathbf{I} + \lambda\mathbf{I})^{-1-i}$$

is positive definite, hence $\mathcal{S}(\lambda) > 0$ which completes the proof. □

Thirdly, we approximate $\mathbf{S}_\delta$ with functions from some lower dimensional spaces. A direct inversion of the right hand side of (26) can be numerically expensive, as its components $(\sigma_i\mathbf{I} + \mathbf{H})^{-1}$ are dense matrices and m is large. We therefore propose to approximate it by a rational function of $\mathbf{H}$ in the form

$$\begin{aligned} \widehat{\mathbf{S}}_\delta = \hat{\mathcal{S}}(\mathbf{H}) &:= \mu\mathbf{H} + \eta_1(\chi_1\mathbf{I} + \mathbf{H})^{-1} \\ &= (\mu\mathbf{H} + \tilde{\eta}_0\mathbf{I})(\mathbf{H} + \tilde{\eta}_1\mathbf{I})(\mathbf{H} + \chi_1\mathbf{I})^{-1}, \end{aligned} \tag{27}$$

where $\theta := \{\eta_1, \chi_1\}$ are parameters for the approximation, $\eta_1, \chi_1 \in \mathbb{R}$, and $\hat{\eta}_1$ and $\hat{\eta}_2$ are determined by μ and θ. The inverse of $\widehat{\mathbf{S}}_\delta$ can be computed efficiently by solving linear systems of $\mathbf{H} + \tilde{\eta}_i\mathbf{I}$. We expect $\widehat{\mathbf{S}}_\delta$ to be as close to $\mathbf{S}_\delta$ as possible. In Section 3, we will show that there exist parameters θ such that the eigenvalues of preconditioned matrix cluster around 1.

Finally, using the approximation of $\mathbf{S}_\delta$ we can construct a preconditioner for $\mathbf{B}$ in (15). Recall that $\mathbf{B}_{\delta,11}^{-1} \to \mathbf{B}_{11}^{-1}$ as $\delta \to 0$ following from Equation (22). Accordingly, the Schur complement $\mathbf{S}_\delta$ of $\mathbf{B}_{\delta,11}$ is an approximation of the Schur complement of $\mathbf{B}_{11}$ in $\mathbf{B}$ defined as

$$\mathbf{S} := \mathbf{B}_{22} - \mathbf{B}_{21}\mathbf{B}_{11}^{-1}\mathbf{B}_{12}. \tag{28}$$

The aim here is to obtain a variant of $\mathbf{B}$ whose inverse can be easily computed. Motivated by this, we do not replace individual blocks of $\mathbf{B}$ by their approximations, but instead modify the Schur complement in $\mathbf{B}$. To do this, define

$$\mathcal{L}(\mathbf{B}) := \begin{pmatrix} \mathbf{I}_{mn} & \\ \mathbf{B}_{21}\mathbf{B}_{11}^{-1} & \mathbf{I} \end{pmatrix}, \quad \mathcal{D}(\mathbf{B}) := \begin{pmatrix} \mathbf{I}_{mn} & \mathbf{O} \\ \mathbf{O}^T & \mathbf{S} \end{pmatrix}, \quad \text{and } \mathcal{R}(\mathbf{B}) := \begin{pmatrix} \mathbf{B}_{11} & \mathbf{B}_{12} \\ & \mathbf{I} \end{pmatrix}$$

where $\mathbf{I}_{mn}$ is the $mn \times mn$ identity matrix and $\mathbf{O}$ represents a zero matrix. Then $\mathbf{B}$ admits the following block factorization

$$\mathbf{B} = \mathcal{L}(\mathbf{B})\mathcal{D}(\mathbf{B})\mathcal{R}(\mathbf{B}). \tag{29}$$

Now we consider approximating the middle factor $\mathcal{D}(\mathbf{B})$ taking the advantage that the blocks are decoupled. If we let $\mathbf{S}$ be replaced by $\widehat{\mathbf{S}}_\delta$, a preconditioner based on $\widehat{\mathbf{S}}_\delta$ is then given as

$$\widehat{\mathbf{P}} = \mathcal{L}(\mathbf{B}) \begin{pmatrix} \mathbf{I}_{mn} & \mathbf{O} \\ \mathbf{O}^T & \widehat{\mathbf{S}}_\delta \end{pmatrix} \mathcal{R}(\mathbf{B}). \tag{30}$$

The inversions of $\mathcal{L}(\mathbf{B})$ and $\mathcal{R}(\mathbf{B})$ can be performed efficiently, while $\widehat{\mathbf{S}}_\delta$ defined in (27) is a product of some matrices that are easy to invert.

2.3 Right Preconditioning

In terms of fast computation, we introduce another preconditioner

$$\mathbf{P} = \begin{pmatrix} \mathbf{I}_{mn} & \mathbf{O} \\ \mathbf{O}^T & \widehat{\mathbf{S}}_\delta \end{pmatrix} \mathcal{R}(\mathbf{B}), \tag{31}$$

as an alternative to the preconditoiner $\widehat{\mathbf{P}}$ in (30). Compared to $\widehat{\mathbf{P}}$, the left factor $\mathcal{L}(\mathbf{B})$ is omitted in $\mathbf{P}$. However, their associated Krylov subspaces are the same if a zero initial guess is used, which means that less computation for $\mathbf{P}$ gives the same result as $\widehat{\mathbf{P}}$. To see the equivalence of the Krylov subspaces, first observe that the first m blocks of $\boldsymbol{b}$ are all zero, i.e., $\boldsymbol{b}$ takes the form $\boldsymbol{b} = [\mathbf{0}^T, \boldsymbol{g}^T]^T$, where $\mathbf{0} \in \mathbb{R}^{mn}$ is a zero vector. The preconditioned matrix and $\mathcal{L}(\mathbf{B})^{-1}$ have the block lower triangular structures, and $(\mathbf{B}\mathbf{P}^{-1})^i\boldsymbol{b}$ has m zero leading blocks, i.e.,

$$\mathbf{B}\mathbf{P}^{-1} = \begin{pmatrix} * & \mathbf{O} \\ * & * \end{pmatrix}, \quad \mathcal{L}(\mathbf{B})^{-1} = \begin{pmatrix} * & \mathbf{O} \\ * & \mathbf{I} \end{pmatrix}, \quad \text{and } (\mathbf{B}\mathbf{P}^{-1})^i\boldsymbol{b} = \begin{pmatrix} \mathbf{0} \\ * \end{pmatrix}, \quad i = 0, 1, 2\cdots, \tag{32}$$

where $*$ denotes a nonzero block. Second, from (32) it is straightforward to check that

$$\text{span}\Big\{\boldsymbol{b}, (\mathbf{B}\mathbf{P}^{-1})\boldsymbol{b}, \cdots, (\mathbf{B}\mathbf{P}^{-1})^{k-1}\boldsymbol{b}\Big\} = \text{span}\Big\{\boldsymbol{b}, \Big(\mathbf{B}\mathbf{P}^{-1}\mathcal{L}(\mathbf{B})^{-1}\Big)\boldsymbol{b}, \cdots, \Big(\mathbf{B}\mathbf{P}^{-1}\mathcal{L}(\mathbf{B})^{-1}\Big)^{k-1}\boldsymbol{b}\Big\}$$

and from the definition of $\widehat{\mathbf{P}}$ in (30), one has

$$\text{span}\Big\{\boldsymbol{b}, \Big(\mathbf{B}\mathbf{P}^{-1}\mathcal{L}(\mathbf{B})^{-1}\Big)\boldsymbol{b}, \cdots, \Big(\mathbf{B}\mathbf{P}^{-1}\mathcal{L}(\mathbf{B})^{-1}\Big)^{k-1}\boldsymbol{b}\Big\} = \text{span}\Big\{\boldsymbol{b}, \Big(\mathbf{B}\widehat{\mathbf{P}}^{-1}\Big)\boldsymbol{b}, \cdots, \Big(\mathbf{B}\widehat{\mathbf{P}}^{-1}\Big)^{k-1}\boldsymbol{b}\Big\}.$$

Finally, if the initial guess is the zero vector, then the two Krylov subspaces $\mathcal{K}_k\Big(\mathbf{B}\widehat{\mathbf{P}}^{-1}, \boldsymbol{b}\Big)$ and $\mathcal{K}_k\big(\mathbf{B}\mathbf{P}^{-1}, \boldsymbol{b}\big)$ are identical.

On the other hand, when the left preconditioning is used, the preconditioned system is given by

$$\widehat{\mathbf{P}}^{-1}\mathbf{B} = \mathcal{R}(\mathbf{B})^{-1}\mathbf{Q}^T\mathcal{R}(\mathbf{B}).$$

in which

$$\mathbf{Q} := \begin{pmatrix} \mathbf{I}_{mn} & \mathbf{O} \\ \mathbf{O}^T & \mathbf{S}\widehat{\mathbf{S}}_\delta^{-1} \end{pmatrix}. \tag{33}$$

Note that $\mathbf{S}\widehat{\mathbf{S}}_\delta^{-1} = \widehat{\mathbf{S}}_\delta^{-1}\mathbf{S}$ since $\mathbf{S}$, $\widehat{\mathbf{S}}_\delta$, and $\mathbf{H}$ are simultaneously diagonalizable (according to Lemma 3 and the definition of $\widehat{\mathbf{S}}_\delta$). Theoretically, the convergence of the GMRES on the preconditioned system $\widehat{\mathbf{P}}^{-1}\mathbf{B}$ depends on both $\mathbf{Q}$ and $\mathcal{R}(\mathbf{B})$.

However, it is interesting to note that the convergence for the right preconditioning depends only on $\mathbf{Q}$. In fact, according to (32), $\mathcal{K}_k\big(\mathbf{B}\mathbf{P}^{-1}, \boldsymbol{b}\big)$ is an invariant subspace of the block triangular matrices $\mathcal{L}(\mathbf{B})^{-1}$, and hence it is straightforward to show from the block decomposition of $\mathbf{B}$ and $\mathbf{P}$ that

$$\mathcal{K}_k\big(\mathbf{B}\mathbf{P}^{-1}, \boldsymbol{b}\big) = \mathcal{K}_k(\mathbf{Q}, \boldsymbol{b}), \quad k = 1, 2, \cdots. \tag{34}$$

It implies $\mathcal{L}(\mathbf{B})$ and $\mathcal{R}(\mathbf{B})$ do not play a role in the Krylov subspaces generated by the right preconditioning method. If we write the residual vector of right preconditioned GMRES for the system of $\mathbf{B}$ at iteration k as $\boldsymbol{r}^{(k)}$, then

$$\begin{aligned} \Big\|\boldsymbol{r}^{(k)}\Big\|_2 = \min_{\boldsymbol{y}\in\mathcal{K}_k(\mathbf{B}\mathbf{P}^{-1},\boldsymbol{b})}\big\|\boldsymbol{b} - \mathbf{B}\mathbf{P}^{-1}\boldsymbol{y}\big\|_2 &= \min_{\boldsymbol{y}\in\mathcal{K}_k(\mathbf{B}\mathbf{P}^{-1},\boldsymbol{b})}\Big\|\boldsymbol{b} - \mathcal{L}(\mathbf{B})^{-1}\big(\mathbf{B}\mathbf{P}^{-1}\boldsymbol{y}\big)\Big\|_2 \\ &= \min_{\boldsymbol{y}\in\mathcal{K}_k(\mathbf{Q},\boldsymbol{b})}\|\boldsymbol{b} - \mathbf{Q}\boldsymbol{y}\|_2, \end{aligned} \tag{35}$$

where the second equality follows from the fact that $\boldsymbol{b}-\mathbf{B}\mathbf{P}^{-1}\boldsymbol{y}$ has m zero leading blocks. Equation (35) implies that the convergence is determined by $\mathbf{Q}$ and $\mathbf{S}\widehat{\mathbf{S}}_\delta^{-1}$.

3 The Spectral Analysis

In this section, we analyze the preconditioner $\mathbf{P}$ given in (31). We will prove that there exists a parameter configuration of $\widehat{\mathbf{S}}_\delta^{-1}$, such that the eigenvalues of $\mathbf{Q}$ (defined in (33)) cluster around 1. Then by Equation (34), the convergence of Krylov methods is completely determined by $\mathbf{Q}$ if a zero initial guess is used, which implies the iterative methods like GMRES converge very quickly.

By Lemma 3 and the definition (27), the matrices $\mathbf{S}$ and $\widehat{\mathbf{S}}_\delta$ are real and symmetric and share the same set of eigenvectors, so the matrices $\mathbf{S}\widehat{\mathbf{S}}_\delta^{-1}$ and $\mathbf{Q}$ are real and symmetric as well. Consequently, the convergence properties of GMRES depend only on the eigenvalues of $\mathbf{S}\widehat{\mathbf{S}}_\delta^{-1}$. So it suffices to show that the spectrum of $\mathbf{S}\widehat{\mathbf{S}}_\delta^{-1}$ clusters around 1. We repeat the definition of $\hat{\mathcal{S}}$ here

$$\hat{\mathcal{S}}(\mathbf{H}) = \mu\mathbf{H} + \eta_1(\chi_1\mathbf{I} + \mathbf{H})^{-1}. \tag{36}$$

Now we derive a necessary condition for the clustered spectrum of $\mathbf{S}\widehat{\mathbf{S}}_\delta^{-1}$ for any $\mu \geq 0$ and arbitrarily large matrix $\mathbf{H}$. For the case of $\mu = 0$, and for any eigenvalue λ of $\mathbf{H}$, the associated eigenvalue of $\mathbf{S}\widehat{\mathbf{S}}_\delta^{-1}$ is

$$\mathcal{S}(\lambda)/\hat{\mathcal{S}}(\lambda) = -\left(\sum_{i=1}^{m} \widehat{\gamma}_i\widehat{q}_i(\sigma_i + \lambda)^{-1}\right) \Bigg/ \left(\eta_1(\chi_1 + \lambda)^{-1}\right).$$

So $\mathcal{S}(\lambda)/\hat{\mathcal{S}}(\lambda) \to -(\sum_{i=1}^m \widehat{\gamma}_i\widehat{q}_i)/\eta_1$ as $\lambda \to \infty$, and $\mathcal{S}(\lambda)/\hat{\mathcal{S}}(\lambda)$ converges to 1 only if $\eta_1 = -\sum_{i=1}^m \widehat{\gamma}_i\widehat{q}_i$. This is also a necessary condition for $\mathbf{S}\widehat{\mathbf{S}}_\delta^{-1}$ to have spectrum clustering around 1 for large sized $\mathbf{H}$. In fact, we will prove later in Lemma 4 that as the size of $\mathbf{H}$ increases most eigenvalues of $\mathbf{H}$ are larger than an arbitrarily given value, and hence the eigenvalues of $\mathbf{S}\widehat{\mathbf{S}}_\delta^{-1}$ cluster around 1 only if $\mathcal{S}(\lambda)/\hat{\mathcal{S}}(\lambda) \to 1$ as $\lambda \to \infty$.

In the subsequent, we let $\eta_1 = -\sum_{i=1}^m \widehat{\gamma}_i\widehat{q}_i$ and show that the eigenvalues of $\mathbf{S}\widehat{\mathbf{S}}_\delta^{-1}$ cluster around 1. We underline that the result holds for any size of matrices $\mathbf{H}$ and $\mathbf{B}$, and for any $\mu \geq 0$. The proofs are divided into three parts. We will first study the properties of the discretization matrices in Subsection 3.1. Then building on their properties, we further prove the clustered spectrum of the preconditioned system in Subsection 3.2. Finally, we extend the results to two-dimensional spatial domains in Subsection 3.3.

3.1 Properties of the Discretization Matrices

To start with, we study the properties of the matrix $\mathbf{H}$ (defined in (7)) and the discrete time fractional derivative. First we have the following lemma describing the distribution of the eigenvalues of $\mathbf{H}$.

Lemma 4. *For any $\lambda > 0$, there exists a number N_λ, independent of the size of $\mathbf{H}$, such that $\mathbf{H}$ has at most N_λ eigenvalues smaller than λ.*

Proof. Associated with $h_{i,i}$ and $h_{i,i+1}$ of the matrix $\mathbf{H}$ defined in (7), set

$$e_{i,i} := h_{i,i} - 2a_{\min},\ e_{i,i+1} = e_{i+1,i} := h_{i,i+1} + a_{\min},\ \text{and } e_{i,j} = 0 \text{ for } |i-j| > 1,$$

where $a_{\min}$ is given in (6). Recalling that $a_{i\pm\frac{1}{2}} \geq a_{\min}$, $h_{i,i} = a_{i+\frac{1}{2}} + a_{i-\frac{1}{2}} - c_i$, and $h_{i,i+1} = h_{i+1,i} = -a_{i+\frac{1}{2}}$, one has $e_{i,i} \geq 0$ and $e_{i,j} \leq 0$ for $j \neq i$. Furthermore, it is easy to check that $\sum_j e_{i,j} \geq -c_i \geq 0$ and hence $e_{i,i} \geq \sum_{j\neq i} |e_{i,j}|$. If we split $\mathbf{H}$ into $\mathbf{H} = \mathbf{H}_{\min} + \mathbf{E}$ where

$$\mathbf{H}_{\min} := \frac{1}{(\Delta x)^2}\begin{pmatrix} 2a_{\min} & -a_{\min} & & \\ -a_{\min} & 2a_{\min} & \ddots & \\ & \ddots & \ddots & -a_{\min} \\ & & -a_{\min} & 2a_{\min} \end{pmatrix}, \quad \mathbf{E} := \frac{1}{(\Delta x)^2}\begin{pmatrix} e_{1,1} & -e_{1,2} & & \\ -e_{2,1} & e_{2,2} & \ddots & \\ & \ddots & \ddots & -e_{n-1,n} \\ & & -e_{n,n-1} & e_{n,n} \end{pmatrix},$$

then it is clear that $\mathbf{E}$ is diagonally dominant with nonnegative diagonal entries and hence symmetric positive semi-definite. The eigenvalues of $\mathbf{H}_{\min}$ has the following closed form

$$\lambda_{\min,i} = \frac{4a_{\min}}{(\Delta x)^2}\left(\sin\frac{\pi i}{2n+2}\right)^2 \geq 4a_{\min}(n+1)^2\left(\frac{i}{n+1}\right)^2 \tag{37}$$

for $i = 1, 2, \cdots, n$. For any $\lambda > 0$, let $N_\lambda = \sqrt{\lambda/(4a_{\min})}$, then if $i \geq N_\lambda$, it holds that $\lambda_{\min,i} \geq \lambda$. This means at most N_λ eigenvalues of $\mathbf{H}_{\min}$ are smaller than λ. Note that here N_λ does not depend on n.

Denote the eigenvalues of $\mathbf{H}$ by $\lambda_1, \lambda_2, \cdots, \lambda_n$ with ordering $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$. Since the eigenvalues of $\mathbf{E}$ are non-negative, by Weyl's inequality, $\lambda_i \geq \lambda_{\min,i}$. Therefore, $\lambda_i \geq \lambda$ for $i \geq N_\lambda$, which implies at most N_λ eigenvalues of $\mathbf{H}$ are smaller than λ. □

Next, we discuss the norm of the inverse of the shifted time fractional operator. Following the discretization (10), let the discrete time fractional differential operator be denoted by

$$\mathbf{T} := \begin{pmatrix} \beta_0 & & & & \\ \beta_1 & \beta_0 & & & \\ \beta_2 & \beta_1 & \beta_0 & & \\ \vdots & \ddots & \ddots & \ddots & \\ \beta_{m-1} & \cdots & \beta_2 & \beta_1 & \beta_0 \end{pmatrix}$$

Then $\mathbf{B}_{11}$, i.e., the top-left block of $\mathbf{B}$ in (16), can be reformulated as follows:

$$\mathbf{B}_{11} = \mathbf{T} \otimes \mathbf{I} + \mathbf{I}_m \otimes \mathbf{H} \tag{38}$$

Therefore, to analyze $\mathbf{B}_{11}$ and $\mathbf{B}$ we first look into the structure of $\mathbf{T}$. The next lemma states a property of $\mathbf{T}$.

Lemma 5. *Given any $\epsilon > 0$, there exists $\lambda_\epsilon \in \mathbb{R}$, which is independent of the size of $\mathbf{T}$, such that for any $\lambda \geq \lambda_\epsilon$,*

$$1 - \epsilon \leq \left\|\lambda(\mathbf{T} + \lambda\mathbf{I}_m)^{-1}\right\|_\infty \leq 1.$$

Proof. By Lemma 1, the entries of $\mathbf{T}$ satisfy $|\beta_0| > \sum_{j=1}^{m-1} |\beta_j|$, so $\mathbf{T}$ is a diagonally dominant matrix with Toeplitz structure. If we write $\mathbf{T} = \beta_0\mathbf{I}_m - \mathbf{T}_L$ where $\mathbf{T}_L$ is a strictly lower triangular matrix, then for any $\lambda > 0$,

$$(\mathbf{T} + \lambda\mathbf{I}_m)^{-1} = \frac{1}{\beta_0 + \lambda}\sum_{k=0}^{m-1}\left(\frac{\mathbf{T}_L}{\beta_0 + \lambda}\right)^k. \tag{39}$$

where $\mathbf{T}_L$ is a nonnegative matrix. Taking the infinity norm of both sides of Equation (39), we get the following upper bound

$$\left\|\lambda(\mathbf{T}+\lambda\mathbf{I}_m)^{-1}\right\|_\infty \leq \frac{\lambda}{\beta_0+\lambda}\sum_{k=0}^{m-1}\left\|\frac{\mathbf{T}_L}{\beta_0+\lambda}\right\|_\infty^k \leq \frac{\lambda}{\beta_0+\lambda}\sum_{k=0}^{m-1}\left(\frac{\beta_0}{\beta_0+\lambda}\right)^k \leq 1.$$

Next, we derive a lower bound of $\left\|\lambda(\mathbf{T}+\lambda\mathbf{I}_m)^{-1}\right\|_\infty$. To do this, we make use of the following inequality from Zhuang and Liu [51] that for any $u \in C^2[0,T]$, there exists a constant $c_0 > 0$ not depending on m, such that

$$\left|(D_t^\alpha u)(t_k) - \frac{\Delta t^{-\alpha}}{\Gamma(2-\alpha)}\sum_{j=0}^{k-1} w_j\left[u(t_{k-j}) - u(t_{k-j-1})\right]\right| \leq c_0\Delta t \leq c_0 T \tag{40}$$

for $k = 1, 2, \cdots, m$. Here $t_k := k\frac{T}{m}$ denotes the grid point of the time domain. Define the vectors

$$\tilde{\boldsymbol{u}} := \left[(D_t^\alpha u)(t_1), (D_t^\alpha u)(t_2), \cdots, (D_t^\alpha u)(t_m)\right]^T \text{ and } \boldsymbol{u} := \left[u(t_1), u(t_2), \cdots, u(t_m)\right]^T,$$

then it follows from (40) that $\|\tilde{\boldsymbol{u}} - \mathbf{T}\boldsymbol{u}\|_\infty \leq c_0 T$. As a special case of u, we set $u(t) := t^2$, the fractional derivative of which is computed as

$$(D_t^\alpha u)(t) = \frac{\Gamma(3)}{\Gamma(3-\alpha)}t^{2-\alpha} \leq \frac{\Gamma(3)}{\Gamma(3-\alpha)}T^{2-\alpha}, \quad \forall t \in (0,T].$$

Consequently, $\|\tilde{\boldsymbol{u}}\|_\infty \leq \Gamma(3)T^{2-\alpha}/\Gamma(3-\alpha)$, and $\|\mathbf{T}\boldsymbol{u}\|_\infty$ is bounded as

$$\|\mathbf{T}\boldsymbol{u}\|_\infty \leq \|\tilde{\boldsymbol{u}}\|_\infty + \|\tilde{\boldsymbol{u}} - \mathbf{T}\boldsymbol{u}\|_\infty \leq \frac{\Gamma(3)}{\Gamma(3-\alpha)}T^{2-\alpha} + c_0 T. \tag{41}$$

Now, according to the definition of $\boldsymbol{u}$, $\|\boldsymbol{u}\|_\infty = u(t_m) = T^2$. Besides, one has the following equality

$$(\mathbf{T}+\lambda\mathbf{I}_m)^{-1}(\mathbf{T}\boldsymbol{u}+\lambda\boldsymbol{u}) = \boldsymbol{u},$$

which forces that

$$\left\|\lambda(\mathbf{T}+\lambda\mathbf{I}_m)^{-1}\right\|_\infty \geq \frac{\|\lambda\boldsymbol{u}\|_\infty}{\|\mathbf{T}\boldsymbol{u}+\lambda\boldsymbol{u}\|_\infty} \geq \frac{\lambda\|\boldsymbol{u}\|_\infty}{\|\mathbf{T}\boldsymbol{u}\|_\infty + \lambda\|\boldsymbol{u}\|_\infty} \geq \frac{\lambda T^2}{\frac{\Gamma(3)}{\Gamma(3-\alpha)}T^{2-\alpha} + c_0 T + \lambda T^2}, \tag{42}$$

where the last inequality follows from (41). For any given $\epsilon > 0$, let $\lambda_\epsilon = \frac{1-\epsilon}{\epsilon T^2}\left(\frac{\Gamma(3)}{\Gamma(3-\alpha)}T^{2-\alpha} + c_0 T\right)$, then the inequalities in (42) leads to

$$\left\|\lambda(\mathbf{T}+\lambda\mathbf{I}_m)^{-1}\right\|_\infty \geq 1-\epsilon, \quad \forall\lambda > \lambda_\epsilon.$$

We underline that λ_ϵ is not dependent on m. The proof is completed. □

Lemma 5 shows that the infinity norm of the matrix $\lambda(\mathbf{T}+\lambda\mathbf{I}_m)^{-1}$ is close to 1 for large λ. Based on this result, we can further show that, under some continuity assumption on $q(\cdot)$, the quantity $\frac{\lambda}{q_m}\boldsymbol{e}_m^T(\mathbf{T}+\lambda\mathbf{I}_m)^{-1}\boldsymbol{q}$ is also close to 1. Here the vectors $\boldsymbol{e}_m$ and $\boldsymbol{q}$ are defined in (14).

Lemma 6. *Assume that $q_{\max} \geq q(t) \geq q_{\min} > 0$ for any $t \in [0, T]$ and $q(t)$ is continuous at $t = T$. Given any $\epsilon > 0$, there exists $\lambda_\epsilon \in \mathbb{R}$, such that*

$$1 - \epsilon \leq \frac{\lambda}{q_m} \boldsymbol{e}_m^T (\mathbf{T} + \lambda \mathbf{I}_m)^{-1} \boldsymbol{q} \leq 1 + \epsilon, \ \forall \lambda \geq \lambda_\epsilon.$$

Proof. Without loss of generality, we assume that $0 < \epsilon < 1$ and $\lambda > 0$. Since $q(t)$ is continuous at $t = T$, there exists $l \in (0, 1)$ such that $|q(t) - q(T)| \leq \frac{\epsilon}{3} q(T)$ for any $t \in [lT, T]$. This means

$$|q_k - q_m| \leq \frac{\epsilon}{3} q_m, \quad \forall k \in \{\lfloor lm \rfloor + 1, \lfloor lm \rfloor + 2, \cdots, m\}. \tag{43}$$

Accordingly, we split the matrix $\mathbf{T} + \lambda \mathbf{I}_m$ into the following form of two-by-two blocks

$$\mathbf{T} + \lambda \mathbf{I}_m = \begin{pmatrix} \mathbf{T}_{\lfloor lm \rfloor} + \lambda \mathbf{I}_{\lfloor lm \rfloor} & \mathbf{O} \\ * & \mathbf{T}_{\hat{m}} + \lambda \mathbf{I}_{\hat{m}} \end{pmatrix},$$

where $\hat{m} := m - \lfloor lm \rfloor$, $*$ represents one of the non-zero blocks, $\mathbf{O}$ denotes a block of all zeros, $\mathbf{T}_{\lfloor lm \rfloor} + \lambda \mathbf{I}_{\lfloor lm \rfloor} \in \mathbb{R}^{\lfloor lm \rfloor \times \lfloor lm \rfloor}$ and $\mathbf{T}_{\hat{m}} + \lambda \mathbf{I}_{\hat{m}} \in \mathbb{R}^{\hat{m} \times \hat{m}}$. The lower triangular structure of $\mathbf{T} + \lambda \mathbf{I}_m$ enables the following block formulation of its inverse

$$(\mathbf{T} + \lambda \mathbf{I}_m)^{-1} = \begin{pmatrix} \left(\mathbf{T}_{\lfloor lm \rfloor} + \lambda \mathbf{I}_{\lfloor lm \rfloor}\right)^{-1} & \mathbf{O} \\ * & (\mathbf{T}_{\hat{m}} + \lambda \mathbf{I}_{\hat{m}})^{-1} \end{pmatrix}.$$

Besides, the lower triangular Toeplitz structure of $\mathbf{T} + \lambda \mathbf{I}_m$ implies that $(\mathbf{T} + \lambda \mathbf{I}_m)^{-1}$ is also lower triangular and Toeplitz. Therefore if we denote the last row of the matrix $\lambda(\mathbf{T} + \lambda \mathbf{I}_m)^{-1}$ by $\boldsymbol{r}^T := \left[\boldsymbol{r}_{\lfloor lm \rfloor}^T, \boldsymbol{r}_{\hat{m}}^T\right]$ with $\boldsymbol{r}_{\lfloor lm \rfloor} \in \mathbb{R}^{\lfloor lm \rfloor}$ and $\boldsymbol{r}_{\hat{m}} \in \mathbb{R}^{\hat{m}}$, then

$$\begin{aligned} \left\| \lambda(\mathbf{T} + \lambda \mathbf{I}_m)^{-1} \right\|_\infty &= \left\| \boldsymbol{r}_{\lfloor lm \rfloor}^T \right\|_\infty + \left\| \boldsymbol{r}_{\hat{m}}^T \right\|_\infty \\ &= \left\| \boldsymbol{r}_{\lfloor lm \rfloor}^T \right\|_\infty + \left\| \lambda(\mathbf{T}_{\hat{m}} + \lambda \mathbf{I}_{\hat{m}})^{-1} \right\|_\infty. \end{aligned} \tag{44}$$

Furthermore, it is easy to check that $\lambda \boldsymbol{e}_m^T (\mathbf{T} + \lambda \mathbf{I}_m)^{-1} = \boldsymbol{r}^T$ and $\boldsymbol{r} \geq 0$ (by Equation (39)). Based on this observation, using a split of vector $\boldsymbol{q}^T = [\boldsymbol{q}_{\lfloor lm \rfloor}^T, \boldsymbol{q}_{\hat{m}}^T]$ with $\boldsymbol{q}_{\lfloor lm \rfloor} \in \mathbb{R}^{\lfloor lm \rfloor}$, we get an upper bound of

$$\begin{aligned} &\left| \lambda \boldsymbol{e}_m^T (\mathbf{T} + \lambda \mathbf{I}_m)^{-1} \boldsymbol{q} / q_m - 1 \right| \\ \leq &\left| \boldsymbol{r}_{\lfloor lm \rfloor}^T \boldsymbol{q}_{\lfloor lm \rfloor} / q_m \right| + \left| \boldsymbol{r}_{\hat{m}}^T \boldsymbol{q}_{\hat{m}} / q_m - 1 \right| \\ \leq &q_{\max} / q_m \left\| \boldsymbol{r}_{\lfloor lm \rfloor} \right\|_1 + \left| \boldsymbol{r}_{\hat{m}}^T (\boldsymbol{q}_{\hat{m}} / q_m - \mathbb{1}) + \boldsymbol{r}_{\hat{m}}^T \mathbb{1} - 1 \right| \\ \leq &q_{\max} / q_m \left\| \boldsymbol{r}_{\lfloor lm \rfloor} \right\|_1 + \left| \boldsymbol{r}_{\hat{m}}^T (\boldsymbol{q}_{\hat{m}} / q_m - \mathbb{1}) \right| + \left| \|\boldsymbol{r}_{\hat{m}}\|_1 - 1 \right| \\ := &\mathrm{Err}_1 + \mathrm{Err}_2 + \mathrm{Err}_3, \end{aligned} \tag{45}$$

where $\mathbb{1}$ denotes a vector of all ones, and in the last inequality we have used the fact that $\boldsymbol{r} \geq 0$. In the rest of the proof, we show that each of Err_1, Err_2, and Err_3 is bounded by $\frac{\epsilon}{3}$ if λ is sufficiently large.

First, from Lemma 5 that there exist $\lambda_\epsilon^{(1)} > 0$ such that

$$\left\| \boldsymbol{r}_{\lfloor lm \rfloor}^T \right\|_\infty + \left\| \boldsymbol{r}_{\hat{m}}^T \right\|_\infty = \left\| \lambda(\mathbf{T} + \lambda \mathbf{I}_m)^{-1} \right\|_\infty \leq 1, \quad \forall \lambda > \lambda_\epsilon^{(1)}. \tag{46}$$

Second, we will use Lemma 5 once again to derive a bound for $\left\|\lambda(\mathbf{T}_{\hat{m}}+\lambda\mathbf{I}_{\hat{m}})^{-1}\right\|_\infty$. Define

$$\widehat{\mathbf{T}}_{\hat{m}} := \begin{pmatrix} \hat{\beta}_0 & & & \\ \hat{\beta}_1 & \hat{\beta}_0 & & \\ \vdots & \ddots & \ddots & \\ \hat{\beta}_{\hat{m}-1} & \cdots & \hat{\beta}_1 & \hat{\beta}_0 \end{pmatrix}, \quad \hat{\beta}_j := \frac{(T/\hat{m})^{-\alpha}}{\Gamma(2-\alpha)}(w_{j+1}-w_j).$$

Observe that the statement in Lemma 5 holds for any matrix size $m \times m$, and hence $\hat{m} \times \hat{m}$. Therefore there exist $\hat{\lambda}_\epsilon^{(2)} \in \mathbb{R}$ such that

$$1-\frac{\epsilon}{3}\frac{q_m}{q_{\max}} \le \left\|\hat{\lambda}\Big(\widehat{\mathbf{T}}_{\hat{m}}+\hat{\lambda}\mathbf{I}_{\hat{m}}\Big)^{-1}\right\|_\infty \le 1, \quad \forall \hat{\lambda} \ge \hat{\lambda}_\epsilon^{(2)}. \tag{47}$$

Here $\hat{\lambda}_\epsilon^{(2)}$ does not depend on $\hat{m}$. Besides, it is easy to check from the definition of $\widehat{\mathbf{T}}_{\hat{m}}$ that $\hat{\beta}_j = \frac{\hat{m}^\alpha}{m^\alpha}\beta_j$ and $\widehat{\mathbf{T}}_{\hat{m}} = \frac{\hat{m}^\alpha}{m^\alpha}\mathbf{T}_{\hat{m}}$. So if we set $\lambda := \frac{m^\alpha}{\hat{m}^\alpha}\hat{\lambda}$, then $\lambda(\mathbf{T}_{\hat{m}}+\lambda\mathbf{I}_{\hat{m}})^{-1} = \hat{\lambda}\Big(\widehat{\mathbf{T}}_{\hat{m}}+\hat{\lambda}\mathbf{I}_{\hat{m}}\Big)^{-1}$. Accordingly, let $\lambda_\epsilon^{(2)} := \max_{k\in\{1,2,\cdots\}}\frac{k^\alpha}{(k-\lfloor kl\rfloor)^\alpha}\hat{\lambda}_\epsilon^{(2)}$. Then $\lambda_\epsilon^{(2)}$ does not depend on m or $\hat{m}$, and it follows immediately from (47) that

$$1-\frac{\epsilon}{3}\frac{q_m}{q_{\max}} \le \left\|\lambda(\mathbf{T}_{\hat{m}}+\lambda\mathbf{I}_{\hat{m}})^{-1}\right\|_\infty = \left\|\boldsymbol{r}_{\hat{m}}^T\right\|_\infty \le 1, \quad \forall \lambda \ge \lambda_\epsilon^{(2)}. \tag{48}$$

Finally, let $\lambda_\epsilon := \max\{\lambda_\epsilon^{(1)}, \lambda_\epsilon^{(2)}\}$, then Equation (46) and the inequalities (48) force that

$$\mathrm{Err}_1 = \frac{q_{\max}}{q_m}\left\|\boldsymbol{r}_{\lfloor lm\rfloor}^T\right\|_\infty \le \frac{\epsilon}{3} \quad \text{and} \quad \mathrm{Err}_3 = \left|\left\|\boldsymbol{r}_{\hat{m}}^T\right\|_\infty - 1\right| \le \frac{\epsilon}{3}\frac{q_m}{q_{\max}} \le \frac{\epsilon}{3}, \quad \forall \lambda \ge \lambda_\epsilon.$$

Besides, according to the inequalities (43) and (48),

$$\mathrm{Err}_2 = \left|\boldsymbol{r}_{\hat{m}}^T(\boldsymbol{q}_{\hat{m}}/q_m - \mathbb{1})\right| \le \left\|\boldsymbol{r}_{\hat{m}}^T\right\|_\infty\left(\frac{\epsilon}{3}\right) \le \frac{\epsilon}{3}.$$

It then follows from Equation (45) that $\left|\frac{\lambda}{q_m}\boldsymbol{e}_m^T(\mathbf{T}+\lambda\mathbf{I}_m)^{-1}\boldsymbol{q}-1\right| \le \epsilon$ for all $\lambda \ge \lambda_\epsilon$. Note that λ_ϵ does not depend on m. The proof is completed. □

3.2 Clustered Spectrum of the Preconditioned System

With the properties of the matrices $\mathbf{H}$ and $\mathbf{T}$, we analyze the spectrum of $\mathbf{S}\widehat{\mathbf{S}}_\delta^{-1}$. We will show that most of the eigenvalues of $\mathbf{S}\widehat{\mathbf{S}}_\delta^{-1}$ are around 1 if the size of $\mathbf{S}$ is large. Based on this, we will discuss the convergence properties of Krylov subspace methods for the preconditioned systems for Problem (16).

Theorem 1. *Assume that $q_{\max} \ge q(t) \ge q_{\min} > 0$ for any $t \in [0,T]$ and $q(t)$ is continuous at $t = T$. Let $\eta_1 = -\sum_{i=1}^m \widehat{\gamma}_i\widehat{q}_i$. For any $\epsilon > 0$, there exists an integer $N > 0$, independent of m and n, and independent of the choice of regularization parameter μ, such that there are at most N eigenvalues of $\mathbf{S}\widehat{\mathbf{S}}_\delta^{-1}$ not belonging to $(1-\epsilon, 1+\epsilon)$.*

Proof. Without loss of generality, assume that $0 < \epsilon < 1$. As a symmetric positive definite matrix, $\mathbf{H}$ has the following decomposition

$$\mathbf{H} = \mathbf{U}^T\mathbf{\Lambda}_\mathbf{H}\mathbf{U},$$

where $\mathbf{U}$ is a unitary matrix, $\mathbf{\Lambda_H} := \mathrm{diag}(\lambda_{\mathbf{H},1}, \lambda_{\mathbf{H},2}, \cdots, \lambda_{\mathbf{H},n})$, and $\lambda_{\mathbf{H},i}$ are positive numbers. By the definition (28) and Equation (38), the Schur complement

$$\begin{aligned}
\mathbf{S} &= \mu\mathbf{H} - \mathbf{B}_{21}\mathbf{B}_{11}^{-1}\mathbf{B}_{12} \\
&= \mu\mathbf{H} + \big(\boldsymbol{e}_m^T \otimes \mathbf{I}\big)(\mathbf{T} \otimes \mathbf{I} + \mathbf{I}_m \otimes \mathbf{H})^{-1}(\boldsymbol{q} \otimes \mathbf{I}) \\
&= \mu\mathbf{H} + \big(\boldsymbol{e}_m^T \otimes \mathbf{I}\big)\big(\mathbf{T} \otimes \mathbf{I} + \mathbf{I}_m \otimes (\mathbf{U}^T\mathbf{\Lambda_H}\mathbf{U})\big)^{-1}(\boldsymbol{q} \otimes \mathbf{I}) \\
&= \mu\mathbf{H} + \big(\boldsymbol{e}_m^T \otimes \mathbf{I}\big)\big(\mathbf{I}_m \otimes \mathbf{U}^T\big)(\mathbf{T} \otimes \mathbf{I} + \mathbf{I}_m \otimes \mathbf{\Lambda_H})^{-1}(\mathbf{I}_m \otimes \mathbf{U})(\boldsymbol{q} \otimes \mathbf{I}) \\
&= \mu\mathbf{H} + \big(\boldsymbol{e}_m^T \otimes \mathbf{U}^T\big)(\mathbf{T} \otimes \mathbf{I} + \mathbf{I}_m \otimes \mathbf{\Lambda_H})^{-1}(\boldsymbol{q} \otimes \mathbf{U}) \\
&= \mu\mathbf{H} + \mathbf{U}^T\big(\boldsymbol{e}_m^T \otimes \mathbf{I}\big)(\mathbf{T} \otimes \mathbf{I} + \mathbf{I}_m \otimes \mathbf{\Lambda_H})^{-1}(\boldsymbol{q} \otimes \mathbf{I})\mathbf{U} \\
&= \mu\mathbf{H} + \mathbf{U}^T\big(\mathbf{I} \otimes \boldsymbol{e}_m^T\big)(\mathbf{I} \otimes \mathbf{T} + \mathbf{\Lambda_H} \otimes \mathbf{I}_m)^{-1}(\mathbf{I} \otimes \boldsymbol{q})\mathbf{U} = \mathbf{U}^T\mathbf{\Lambda_S}\mathbf{U},
\end{aligned}$$

in which $\mathbf{\Lambda_S} := \mu\mathbf{\Lambda_H} + \big(\mathbf{I} \otimes \boldsymbol{e}_m^T\big)(\mathbf{I} \otimes \mathbf{T} + \mathbf{\Lambda_H} \otimes \mathbf{I}_m)^{-1}(\mathbf{I} \otimes \boldsymbol{q})$ is a diagonal matrix because of the block diagonal structure of $(\mathbf{I} \otimes \mathbf{T} + \mathbf{\Lambda_H} \otimes \mathbf{I}_m)^{-1}$. Based on this decomposition, the eigenvalues of $\mathbf{S}$ are the diagonal entries of $\mathbf{\Lambda_S}$ which are given by

$$\lambda_{\mathbf{S},i} = \mu\lambda_{\mathbf{H},i} + \boldsymbol{e}_m^T(\mathbf{T} + \lambda_{\mathbf{H},i}\mathbf{I}_m)^{-1}\boldsymbol{q}, \quad i = 1, 2, \cdots, n. \tag{49}$$

On the other hand, the decomposition of $\mathbf{H}$ allows $\widehat{\mathbf{S}}_\delta$ defined in (36) to be decomposed as

$$\widehat{\mathbf{S}}_\delta = \mathbf{U}^T\mathbf{\Lambda}_{\widehat{\mathbf{S}}_\delta}\mathbf{U},$$

where $\mathbf{\Lambda}_{\widehat{\mathbf{S}}_\delta}$ is a diagonal matrix having diagonal entries

$$\begin{aligned}
\lambda_{\widehat{\mathbf{S}}_\delta,i} = \mu\lambda_{\mathbf{H},i} + \left(-\sum_{i=1}^{m}\widehat{\gamma}_i\widehat{q}_i\right)/(\chi_1 + \lambda_{\mathbf{H},i}) &= \mu\lambda_{\mathbf{H},i} + \big(\boldsymbol{e}_m^T\big(\mathbf{D}_\delta^{-1}\mathbf{F}_m^{-1}\big)\big)(\mathbf{F}_m\mathbf{D}_\delta\boldsymbol{q})/(\chi_1 + \lambda_{\mathbf{H},i}) \\
&= \mu\lambda_{\mathbf{H},i} + q_m/(\chi_1 + \lambda_{\mathbf{H},i})
\end{aligned} \tag{50}$$

for $i = 1, 2, \cdots, n$, where the second equality follows from the definition (25). Consequently, the eigenvalues of $\mathbf{S}\widehat{\mathbf{S}}_\delta^{-1}$ are given explicitly as $\lambda_{\mathbf{S}\widehat{\mathbf{S}}_\delta^{-1},i} = \lambda_{\mathbf{S},i}/\lambda_{\widehat{\mathbf{S}}_\delta,i}$. By Lemma 6, there exists $\lambda_\epsilon^{(1)} \in \mathbb{R}$, such that

$$\frac{\lambda}{q_m}\boldsymbol{e}_m^T(\mathbf{T} + \lambda\mathbf{I}_m)^{-1}\boldsymbol{q} \in \left(1 - \frac{\epsilon}{3}, 1 + \frac{\epsilon}{3}\right), \quad \forall\lambda \geq \lambda_\epsilon^{(1)}.$$

Let $\lambda_\epsilon = \max\Big(\lambda_\epsilon^{(1)}, \frac{3|\chi_1|}{\epsilon}\Big)$, then $\frac{\lambda+\chi_1}{\lambda} \in \big(1 - \frac{\epsilon}{3}, 1 + \frac{\epsilon}{3}\big)$ for all $\lambda > \lambda_\epsilon$. Therefore, by Equation (49) and (50), for any $\lambda_{\mathbf{H},i} > \lambda_\epsilon$,

$$\begin{aligned}
\lambda_{\mathbf{S}\widehat{\mathbf{S}}_\delta^{-1},i} &= \frac{\mu\lambda_{\mathbf{H},i} + \boldsymbol{e}_m^T(\mathbf{T} + \lambda_{\mathbf{H},i}\mathbf{I}_m)^{-1}\boldsymbol{q}}{\mu\lambda_{\mathbf{H},i} + q_m/\lambda_{\mathbf{H},i}}\frac{\mu\lambda_{\mathbf{H},i} + q_m/\lambda_{\mathbf{H},i}}{\mu\lambda_{\mathbf{H},i} + q_m/(\chi_1 + \lambda_{\mathbf{H},i})} \\
&\leq \max\left(1, \frac{\lambda_{\mathbf{H},i}}{q_m}\boldsymbol{e}_m^T(\mathbf{T} + \lambda_{\mathbf{H},i}\mathbf{I}_m)^{-1}\right) \cdot \max(1, (\chi_1 + \lambda_{\mathbf{H},i})/\lambda_{\mathbf{H},i}) \\
&\leq \left(1 + \frac{\epsilon}{3}\right)\left(1 + \frac{\epsilon}{3}\right) \leq 1 + \epsilon.
\end{aligned} \tag{51}$$

Note that λ_ϵ is independent of m and n. Using a similar argument, one can show that $1 - \epsilon < \lambda_{\mathbf{S}\widehat{\mathbf{S}}_\delta^{-1},i}$ if $\lambda_{\mathbf{H},i} > \lambda_\epsilon$. Finally, by Lemma 4, there exists an integer N (independent of m and n), such that at most N elements of $\{\lambda_{\mathbf{H},i} \mid i = 1, 2, \cdots, n\}$ are smaller than or equal to λ_ϵ. This implies that at most N eigenvalues of $\mathbf{S}\widehat{\mathbf{S}}_\delta^{-1}$ are not belonging to $(1 - \epsilon, 1 + \epsilon)$. The proof is completed. □

It follows immediately from Theorem 1 that the symmetric matrix $\mathbf{Q}$ has a spectrum clustered around 1. Moreover, Equation (37) together with the inequalities in (51) implies that $\mathbf{S}\widehat{\mathbf{S}}_\delta^{-1}$ (and hence $\mathbf{Q}$) has eigenvalues lower bounded by some positive number if $\chi_1 \geq -\lambda_{\mathbf{H},i}\ \forall i = 1, 2, \cdots, n$. Therefore, the GMRES has a superlinear rate of convergence on the problem $\mathbf{Q}\boldsymbol{z} = \boldsymbol{b}$. It follows from Equation (35) that residual vector of right preconditioned GMRES for $\mathbf{B}$ converges to zero superlinearly as well.

3.3 Analysis for the Two-dimensional Problems

A similar analysis to Theorem 1 can be applied to systems with 2D spatial domain $\Omega = (0, 1)^2$. For ease of presentation, in the following discussion we assume that for the differential operator L (defined in (3)) the coefficient function $a_{i,i} = a$ for $i \in \{1, 2\}$, and $c(\cdot) = a_{i,j}(\cdot) = 0$ for $i \neq j$. The operator $-L$ is then discretized as

$$\mathbf{H}_{2\mathrm{D}} = \mathbf{I} \otimes \mathbf{H} + \mathbf{H} \otimes \mathbf{I}$$

where $\mathbf{H}$ is the discrete version of the 1D negative Laplacian operator given in (7). Given that the matrix $\mathbf{H}$ satisfies the properties in Lemma 4, it is straightforward to show that

Lemma 7. *For any $\lambda > 0$, there exists an integer $N_\lambda > 0$, independent of the size of $\mathbf{H}_{2\mathrm{D}}$, such that $\mathbf{H}_{2\mathrm{D}}$ has at most N_λ eigenvalues smaller than λ.*

Proof. Let the eigenvalues of $\mathbf{H}$ be ordered as $\lambda_{\mathbf{H},1} \leq \lambda_{\mathbf{H},2} \leq \cdots \leq \lambda_{\mathbf{H},n}$. Given λ, by Lemma 4, there is a number N_λ which is not depending on n, such that $\lambda_{\mathbf{H},\lfloor\sqrt{N_\lambda}\rfloor} > \lambda$. According to the definition of $\mathbf{H}_{2\mathrm{D}}$, the set of its eigenvalues is $\{\lambda_{\mathbf{H},i} + \lambda_{\mathbf{H},j} \mid i, j = 1, 2, \cdots, n\}$. It holds that $\lambda_{\mathbf{H},i} + \lambda_{\mathbf{H},j} > \lambda$ if $i \geq \sqrt{N_\lambda}$ or $j \geq \sqrt{N_\lambda}$ which implies that at most N_λ eigenvalues of $\mathbf{H}_{2\mathrm{D}}$ are smaller than λ. □

To set up the problem for the two-dimensional case, we need to solve a system in the form of (13), but now the SPD matrix $\mathbf{H}$ is replaced by its 2D version $\mathbf{H}_{2\mathrm{D}}$, and accordingly, $\mathbf{B}_{11}$'s diagonal block is $\mathbf{A} := \mathbf{H}_{2\mathrm{D}} + \beta_0\mathbf{I}_{n^2}$ where $\mathbf{I}_{n^2} \in \mathbb{R}^{n^2\times n^2}$ denotes the identity matrix. Associated with $\mathbf{H}_{2\mathrm{D}}$ and $\mathbf{B}$, we define the 2D versions of the Schur complement $\mathbf{S} \in \mathbb{R}^{n^2\times n^2}$ and $\hat{\mathbf{S}}_\delta \in \mathbb{R}^{n^2\times n^2}$ (given in Equation (36)), respectively.

Corollary 2. *Under the assumption on $q(\cdot)$ and η_1 in Theorem 1, for any $\epsilon > 0$, there exists an integer $N > 0$, independent of m and n, and independent of the choice of regularization parameter μ, such that there are at most N eigenvalues of $\mathbf{S}\widehat{\mathbf{S}}_\delta^{-1}$ not belonging to $(1 - \epsilon, 1 + \epsilon)$.*

Proof. The proof is similar to the proof of Theorem 1. Since $\mathbf{H}_{2\mathrm{D}}$ is symmetric positive definite,

$$\mathbf{H}_{2\mathrm{D}} = \mathbf{U}_{2\mathrm{D}}^T\mathbf{\Lambda}_{2\mathrm{D}}\mathbf{U}_{2\mathrm{D}}$$

for some unitary matrix $\mathbf{U}_{2\mathrm{D}}$ and $\mathbf{\Lambda}_{2\mathrm{D}} = \mathrm{diag}\big(\lambda_{2\mathrm{D},1}, \lambda_{2\mathrm{D},2}, \cdots, \lambda_{2\mathrm{D},\mathrm{n}^2}\big)$. With some straightforward computation,

$$\begin{aligned} \lambda_{\mathbf{S},i} &= \mu\lambda_{2\mathrm{D},i} + \boldsymbol{e}_m^T(\mathbf{T} + \lambda_{2\mathrm{D},i}\mathbf{I}_m)^{-1}\boldsymbol{q}, \\ \lambda_{\widehat{\mathbf{S}}_\delta,i} &= \mu\lambda_{2\mathrm{D},i} + q_m/(\chi_1 + \lambda_{2\mathrm{D},i}). \end{aligned} \tag{52}$$

According to Lemma 6, for any $\epsilon > 0$, there exists $\lambda_\epsilon^{(1)}$, such that

$$\frac{\lambda}{q_m} \boldsymbol{e}_m^T (\mathbf{T} + \lambda \mathbf{I}_m)^{-1} \boldsymbol{q} \in \Big(1 - \frac{\epsilon}{3}, 1 + \frac{\epsilon}{3}\Big), \quad \forall \lambda \geq \lambda_\epsilon^{(1)}. \tag{53}$$

Based on the conclusions in (52) and (53), analogous to (51), for any $\lambda_{2\mathrm{D},i} > \lambda_\epsilon := \max\Big(\lambda_\epsilon^{(1)}, \frac{3|\chi_1|}{\epsilon}\Big)$

$$\begin{aligned}
\lambda_{\mathbf{S}\widehat{\mathbf{S}}_\delta^{-1},i} &\leq \max\left(1, \frac{\lambda_{2\mathrm{D},i}}{q_m} \boldsymbol{e}_m^T (\mathbf{T} + \lambda_{2\mathrm{D},i} \mathbf{I}_m)^{-1}\right) \cdot \max(1, (\chi_1 + \lambda_{2\mathrm{D},i})/\lambda_{2\mathrm{D},i}) \\
&\leq \Big(1 + \frac{\epsilon}{3}\Big)\Big(1 + \frac{\epsilon}{3}\Big) \leq 1 + \epsilon,
\end{aligned}$$

and $1 - \epsilon < \lambda_{\mathbf{S}\widehat{\mathbf{S}}_\delta^{-1},i}$. Therefore $1 - \epsilon \leq \lambda_{\mathbf{S}\widehat{\mathbf{S}}_\delta^{-1},i} \leq 1 + \epsilon$, for any i satisfying $\lambda_{2\mathrm{D},i} > \lambda_\epsilon$. Moreover, by Lemma 7, we can find an integer N not depending on m or n, such that $\{i \mid \lambda_{2\mathrm{D},i} \leq \lambda_\epsilon^{(1)}\}$ has at most N elements. This gives the desired result. □

For the 2D problems the matrix $\mathbf{B}$ in (16) and the preconditioner $\mathbf{P}$ in (31) have the same two-by-two block structure as in the 1D case, thus the equivalence of subspaces in (34) still holds. Therefore, Equation (35) together with the spectral analysis above indicates that the preconditioning technique constitutes iterative solvers with fast convergence rates. In the next section, we test the performance of the proposed preconditioner with numerical experiments and discuss the methods of selecting the regularization parameters.

4 Numerical Experiments

The numerical experiments in the paper are carried out on a machine equipped with Intel Xeon 6142 processors (2.60GHz) and MATLAB R2018b. The effectiveness of the proposed methods in both the speed and the accuracy will be evaluated over various factors such as the grid size n and m, the constant δ, the regularization parameters μ, etc. At the end of this section, we also discuss the selection of suitable regularization parameters.

4.1 Preconditioners Setting

We study the performance of $\mathbf{P}$ in (31) for the inverse problems in one-dimensional spatial domains and two-dimensional spatial domains respectively. In both cases the preconditioner $\mathbf{P}$ is defined with the same $\widehat{\mathbf{S}}_\delta$ given in (36) with $\eta_1 = -\sum_{i=1}^m \widehat{\gamma}_i \widehat{q}_i$ and $\chi_1 = 0$. Throughout all tests, we use the GMRES method with right preconditioning. Specifically, we solve the linear system

$$\mathbf{B}\mathbf{P}^{-1}\boldsymbol{y} = \boldsymbol{b}$$

with the GMRES method for $\boldsymbol{y}$, and $\boldsymbol{x}$ is computed based on $\boldsymbol{x} = \mathbf{P}^{-1}\boldsymbol{y}$. To perform the preconditioning with $\mathbf{P}$, each iteration of the iterative method requires an evaluation of the matrix-vector product $\hat{\boldsymbol{r}} = \mathbf{B}\mathbf{P}^{-1}\boldsymbol{r}$, which we decompose into the following steps

$$\boldsymbol{r}^{(1)} := \begin{pmatrix} \mathbf{I}_{mn} & \mathbf{O} \\ \mathbf{O}^T & \widehat{\mathbf{S}}_\delta^{-1} \end{pmatrix} \boldsymbol{r}, \quad \boldsymbol{r}^{(2)} := \begin{pmatrix} \mathbf{B}_{11} & \mathbf{B}_{12} \\ & \mathbf{I} \end{pmatrix}^{-1} \boldsymbol{r}^{(1)}, \quad \text{and } \hat{\boldsymbol{r}} := \mathbf{B}\boldsymbol{r}^{(2)}.$$

If we let the vectors be split into $\boldsymbol{r}^{(i)} = [(\boldsymbol{r}_1^{(i)})^T, (\boldsymbol{r}_2^{(i)})^T]^T$ for $i \in \{1, 2\}$, then to compute $\boldsymbol{r}^{(2)}$ one needs to solve the linear system,

$$\mathbf{B}_{11}\boldsymbol{r}_1^{(2)} = \boldsymbol{r}_1^{(1)} - \mathbf{B}_{12}\boldsymbol{r}_2^{(2)} \tag{54}$$

which is essentially the discrete time fractional diffusion equation (TFDE) and can be solved by fast direct methods e.g., in [25]. Alternatively, one may employ an inexact method. In particular, we have the approximation $\mathbf{B}_{\delta,11}$ (defined in (17)) of $\mathbf{B}_{11}$ introduced in Subsection 2.1. An approximated solution of (54) can be obtained by solving the linear system $\mathbf{B}_{\delta,11}\boldsymbol{r}_1^{(2)} = \boldsymbol{r}_1^{(1)} - \mathbf{B}_{12}\boldsymbol{r}_2^{(2)}$, for which the block diagonalization (18) can be exploited to implement fast solvers. The approximation of (54) results into a variant of the preconditioner $\mathbf{P}$, which has the form

$$\mathbf{P}_\delta = \begin{pmatrix} \mathbf{I}_{mn} & \mathbf{O} \\ \mathbf{O} & \widehat{\mathbf{S}}_\delta \end{pmatrix} \mathcal{R}(\mathbf{B}_\delta)$$

where $\mathbf{B}_\delta$ is defined in (23).

In this experiment, we test both $\mathbf{P}$ and $\mathbf{P}_\delta$ and compare their performance. To solve the $\mathbf{B}_{11}$ problem (54) for the preconditioner $\mathbf{P}$, we use the divide and conquer method proposed in [25]. Besides, the application of $\mathbf{P}$ and $\mathbf{P}_\delta$ requires solving the following subproblems.

Subproblems with shifted linear systems. The inversion of $\mathbf{B}_{11}$ (as required by $\mathbf{P}$) and the inversion of $\mathbf{B}_{11,\delta}$ (as required by $\mathbf{P}_\delta$) raise subproblems associated with $\mathbf{A}$ in (11) and with $\boldsymbol{\Sigma}_i$ defined in (19c), respectively. Both subproblems are with shifted linear systems of $\mathbf{H}$.

One of the advantages of our proposed preconditioners is that they are developed independently of the choice of solvers for the $\mathbf{A}$-subproblems and the $\boldsymbol{\Sigma}_i$-subproblems, therefore allowing the most efficient solvers to be applied. The matrix $\mathbf{H}$ being a negative discrete Laplacian, many sophisticated methods exist for solving these subproblems. We demonstrate the proposed preconditioners using two examples of existing solvers for the subproblems. Both direct methods and iterative methods are considered. We compare the convergence speed and the computational cost per iteration for these two cases, and discuss the trade-off between memory consumption and the time complexity. Next, we present the implementation details of the solvers for the $\mathbf{A}$-subproblems, and the solutions of the $\boldsymbol{\Sigma}_i$-subproblems are obtained similarly.

- *Re-ordered Cholesky factorization.* Typical direct solvers for linear systems in $\mathbf{A}$ are based on the LU factorization or the Cholesky factorization. However, the Cholesky factorization of sparse matrices often leads to dense matrices and is hence not efficient. To remedy this, the Re-ordered Cholesky Factorization (RCF) [12] method re-orders the row and columns of $\mathbf{A}$ to reduce the fill-in. In other words, it finds a permutation matrix $\mathbf{J}$ and a lower triangular matrix $\mathbf{L}$ such that

  $$\mathbf{J}^T\mathbf{A}\mathbf{J} = \mathbf{L}\mathbf{L}^T,$$

  and $\mathbf{L}$ is as sparse as possible. In our implementation, we obtain the permutation $\mathbf{J}$ with the MATLAB function `symamd`, and both $\mathbf{J}$ and $\mathbf{L}$ are computed offline. Optionally, one may store only $\mathbf{J}$ and compute $\mathbf{L}$ during each iteration to reduce the memory requirement. Once the factorization is obtained, the subproblem can be solved exactly using only one forward substitution and one backward substitution.
- *Multigrid method.* As an iterative method, the multigrid (MG) method [10] is known to be effective in solving elliptic PDEs and the time complexity scales linearly with the size of the

problems. In our implementation of the MG method, we use Galerkin coarse grid operators and use full weighting as restriction operators and their transpose multiplied by a constant (depending on the dimension of the spatial domain) as the interpolation operators. We refer the readers to [10] for the definitions of these operators. We use two Gauss–Seidel iterations pre-smoothing and two Gauss–Seidel iterations for post-smoothing. In the implementation, we use two V-cycles, and to optimize the speed for each V-cycle, some matrices are pre-computed and stored. More specifically, let $\mathbf{A}^{(1)} = \mathbf{A}$, and let $\mathbf{R}^{(k)}$ be the restriction operator at level k, then the coarse-grid operators and their splittings are given by

$$\mathbf{A}^{(k+1)} = \mathbf{R}^{(k)}\mathbf{A}^{(k)}\mathbf{I}^{(k)}, \quad \mathbf{L}^{(k)} + \mathbf{U}^{(k)} = \mathbf{A}^{(k)}, \quad \text{for } k = 1, 2, \cdots,$$

where $\mathbf{I}^{(k)} = c(\mathbf{R}^{(k)})^T$, $c \in \mathbb{R}$, $\mathbf{L}^{(k)}$ is the lower triangular component of $\mathbf{A}^{(k)}$, and $\mathbf{U}^{(k)}$ is the strictly upper triangular component of $\mathbf{A}^{(k)}$. We pre-compute the matrices $\mathbf{A}^{(k)}$, $\mathbf{R}^{(k)}$, $\mathbf{L}^{(k)}$, and $\mathbf{U}^{(k)}$ for every V-cycle level to speed up the iterations. The memory cost of storing these matrices scales linearly with the number of rows of $\mathbf{A}$. For solving the coarsest grid problem and the forward substitution of the Gauss–Seidel method, we use the MATLAB backslash operator " \".

We note that the choice of suitable solvers for the subproblems depends on the dimension of the spatial domain and the size of the discretization grids. In particular, for 1D problems, $\mathbf{A}$ being a tridiagonal matrix, its Cholesky factorization matrix has only two nonzero diagonals, thus the RCF based solvers require fewer arithmetic operations and are preferred over the MG method. More importantly, in the numerical results, we will show that the proposed preconditioning approach converges in a small number of iterations for different choices of subproblem solvers. In the following, the preconditioner $\mathbf{P}$ and $\mathbf{P}_\delta$ implemented with RCF will be denoted by $\mathbf{P}$-Chol and $\mathbf{P}_\delta$-Chol[1] respectively. Similarly, the ones based on the MG method will be denoted by $\mathbf{P}$-MG and $\mathbf{P}_\delta$-MG.

4.2 Computational Results

We start with a simulation on the one dimensional spatial domain (Example 1) and investigate the convergence speed for the proposed preconditioners as well as the accuracy of the QBVM regularization in the presence of noise in the observed final time data. Then we report similar results for the two dimensional case (Example 2). Based on these two examples, we further study the effect of the value of δ on the performance of $\mathbf{P}_\delta$. Finally, we discuss the selection of the best regularization parameter μ and show that the preconditioning techniques has fast convergence independent of the choice of the regularization parameter μ.

Example 1 (1D problem). Consider a one-dimensional spatial domain $\Omega = (0, 1)$ and $T = 1$. We set the diffusion coefficient $a(x) = x^2 + 1$ and $c(x) = -(x+1)$. The time-dependent source term is given as $q(t) = e^{-t}$. The order of time derivative is $\alpha = 0.6$. In this example we have the ground truth solution $f(x) = x(1-x)^\alpha \sin(5\pi x)$. The exact final time data is given by $g(x) = u(x, T)$ as defined in Equation (2) with u being a solution of (1). Letting $\boldsymbol{g}$ be the noise free data, we assume that only a noisy version of $\boldsymbol{g}$ is available as an input of the algorithm. In particular, the measured data is $\boldsymbol{g}_\upsilon = \boldsymbol{g} + \frac{\upsilon}{\sqrt{N}}\|\boldsymbol{g}\|_2\boldsymbol{\xi}$, where $\boldsymbol{\xi}$ is a vector of standard Gaussian white noise, N is

[1]For the $\boldsymbol{\Sigma}_i$-problem in $\mathbf{P}_\delta$-Chol, the matrices $\boldsymbol{\Sigma}_i$ have the same distribution of nonzero entries as $\mathbf{A}$, therefore we used the same reordering (as the one for $\mathbf{A}$) followed by a LU factorization.

the dimension of the vector $\boldsymbol{g}$. In the test, 1% noise is added, i.e., $\upsilon = 0.01$. The regularization parameter μ is 5.96×10^{-8}, which is selected based on a criterion of minimizing the l_2 distance between the computed solution and the ground truth $f(x)$. We delay the discussion on numerical methods for choosing μ until Subsection 4.2.2. To find $\boldsymbol{f}_\mu$ with the given data $\boldsymbol{g}_\upsilon$, the discrete system (13) is solved with the GMRES methods using the preconditioners $\mathbf{P}$ and $\mathbf{P}_\delta$ respectively. As for $\mathbf{P}_\delta$, we let $\delta = 0.2$. The stopping criterion is the relative residual error

$$\|\mathbf{B}\boldsymbol{x}^* - \boldsymbol{b}\|_2 / \|\boldsymbol{b}\|_2 \leq 10^{-6}$$

where $\boldsymbol{x}^*$ is the solution of the iterative methods.

The problems are solved on a variety of discretization grid sizes n and m. The number of iterations and the running time (excluding the precomputation time) of the methods are compared in Table 1. The running time is averaged over 8 independent runs for each method. In this example the precomputation time is much smaller than the overall running time of the iterations. In the table, "ITER" stands for the number of iterations at which the methods meet the stopping criterion. The term CPU means the running time of the GMRES methods. In addition to the proposed approaches, for comparison purposes, we also apply the restarted GMRES(k) method for solving the non-symmetric system, with inner iteration number $k = 50$. The results in Table 1 show that the method with preconditioner $\mathbf{P}$-Chol always converges in a few iterations, while the one with $\mathbf{P}_\delta$-Chol needs more iterations due to the fact that (54) is solved approximately. In terms of computational time, $\mathbf{P}$-Chol is faster for smaller m, and it is overtaken by $\mathbf{P}_\delta$-Chol when m is large. The baseline method GMRES(50), however, fails to converge at large numbers of iterations ($>$5000 total iterations) and is therefore much slower than the proposed methods.

For $\mathbf{P}$-MG and $\mathbf{P}_\delta$-MG, since the $\mathbf{A}$-subproblems and the $\mathbf{\Sigma}_i$-subproblems are not solved exactly (for both subproblems, 2 MG V-cycles are run), in general, they take more iterations than $\mathbf{P}$-Chol and $\mathbf{P}_\delta$-Chol which are based on exact solvers for the $\mathbf{A}$-subproblems and $\mathbf{\Sigma}_i$-subproblems. Nonetheless, for each of the four methods, the numbers of iterations are small for all n and m. It is not surprising that $\mathbf{P}$-MG and $\mathbf{P}_\delta$-MG take more time than $\mathbf{P}$-Chol and $\mathbf{P}_\delta$-Chol in this example, given the fact that $\mathbf{A}$ and $\mathbf{\Sigma}_i$ are tridiagonal matrices and the triangular factorization is of linear complexity in 1D problems.

To study the accuracy of the reconstructed results compared to the true $f(x)$, we fix $n = 2^7$ and $m = 2^{14}$ and solve the problem in a naive manner (i.e., without using regularization, or equivalently, $\mu = 0$). Having 1% Gaussian white noise in the observed data, the reconstruction without any regularization is severely corrupted by noise. In contrast, the computed solution matches the ground truth solution well if regularization is imposed (See the red line in Figure 1).

Example 2 (2D problem). In this example, we consider the two-dimensional case. Let $\Omega = (0,1) \times (0,1)$ and $T = 1$. For the diffusion coefficients, we let $a_{11}(x,y) = x^2 + 3$, $a_{12}(x,y) = a_{21}(x,y) = x + y + 1$, $a_{22}(x,y) = y^2 + 3$ for $(x,y) \in \Omega$. The order of time derivative is $\alpha = 0.8$. The time-dependent term is given by $q(t) = e^{-t} + t^\alpha + 1$ and $c(x,y) = -(x+y)^2$. Here, the ground truth solution is $f(x,y) = e^{-\frac{\alpha}{\sqrt{xy}} - \frac{\alpha}{\sqrt{(x-1)(y-1)}}}$. The discretization of the 2D problem is analogous to the 1D case discussed in Section 2. The grids for $\bar{\Omega}$ and $[0,T]$ are now given respectively by $\{(x_i,y_j) \mid (x_i,y_j) = (\Delta x)(i,j),\ i,j = 0,1,\cdots,n+1\}$ and $\{t_k \mid t_k = k(\Delta t),\ k = 0,1,\cdots,m\}$. The discrete system has the same form as (13) with the diagonal block $\mathbf{A}$ replaced by a 2D discrete elliptic operator $\beta_0\mathbf{I} + \mathbf{H}_{2\mathrm{D}}$ and $\mathbf{I}$ replaced by the $n^2 \times n^2$ identity matrix $\mathbf{I}_{nn}$. Therefore, we have a linear system of size $n^2 m$. The measured data $\boldsymbol{g}_\upsilon$ for this example is generated with 1% additive Gaussian white noise. For the quasi-boundary regularization, the parameter is set to

Table 1: The number of iterations (ITER) and CPU time (CPU) in second for Example 1. The running time of each proposed method is averaged over 8 runs. The "$>$" means that the algorithm does not converge up to the indicated number of iterations or running time.

Grid sizes		GMRES(50)		$\mathbf{P}$-Chol		$\mathbf{P}_\delta$-Chol		$\mathbf{P}$-MG		$\mathbf{P}_\delta$-MG	
		ITER	CPU	ITER	CPU	ITER	CPU	ITER	CPU	ITER	CPU
$m = 2^8$	$n = 2^6$	>5000	>50	3	0.07	4	0.15	7	0.43	8	0.69
	$n = 2^7$		>130	3	0.08	4	0.08	7	0.69	8	0.78
	$n = 2^8$		>180	3	0.11	4	0.11	7	0.96	8	1.35
	$n = 2^9$		>230	3	0.17	4	0.20	7	1.32	7	1.76
	$n = 2^{10}$		>300	3	0.28	4	0.35	7	2.06	7	2.91
$m = 2^{10}$	$n = 2^6$	>5000	>170	3	0.21	5	0.22	7	1.54	8	1.87
	$n = 2^7$		>220	3	0.31	5	0.31	7	2.45	8	2.77
	$n = 2^8$		>300	3	0.49	5	0.51	7	3.49	8	4.74
	$n = 2^9$		>600	3	0.79	5	0.83	7	4.84	8	7.16
	$n = 2^{10}$		>800	3	1.38	5	1.48	7	7.62	7	10.52
$m = 2^{12}$	$n = 2^6$	>5000	>400	3	0.96	5	0.89	7	6.27	8	7.28
	$n = 2^7$		>600	3	1.36	5	1.25	7	9.72	8	11.06
	$n = 2^8$		>1200	3	2.20	5	2.03	7	13.94	8	18.74
	$n = 2^9$		>2600	3	3.93	5	3.68	7	20.32	8	28.82
	$n = 2^{10}$		>4000	3	7.31	5	7.02	7	32.53	8	48.12
$m = 2^{14}$	$n = 2^6$	>5000	>1400	3	4.23	5	3.75	7	25.22	8	29.61
	$n = 2^7$		>2600	3	6.22	5	5.29	7	40.37	8	44.77
	$n = 2^8$		>5000	3	10.69	5	9.04	7	57.45	8	77.63
	$n = 2^9$		>11000	3	19.60	5	16.34	7	89.09	8	124.21
	$n = 2^{10}$		>14000	3	36.00	5	29.61	7	144.02	8	209.66
$m = 2^{16}$	$n = 2^6$	>5000	>6000	3	19.96	5	16.70	6	93.96	9	130.07
	$n = 2^7$		>12000	3	28.54	5	23.63	7	164.54	8	183.28
	$n = 2^8$		>23000	3	50.38	5	37.17	7	241.99	8	333.18
	$n = 2^9$		>40000	3	94.63	5	69.47	7	384.75	8	531.55
	$n = 2^{10}$		>70000	3	176.08	5	125.59	7	640.06	8	879.20

$\mu = 9.54 \times 10^{-7}$ based on a selection criterion of the l_2 distance between the result and the ground truth f. Again, we apply the GMRES method with the proposed preconditioning techniques on the discrete system. For the preconditioner $\mathbf{P}_\delta$-Chol and $\mathbf{P}_\delta$-MG, we set $\delta = 0.01$. The iterative methods are stopped when the relative residual error is less than or equal to 10^{-7}.

For the methods $\mathbf{P}$-Chol and $\mathbf{P}_\delta$-Chol, we first take a closer look at the sparseness of the triangular factorization matrices of the reordered $\mathbf{A}$ and $\mathbf{\Sigma}_i$. Let $n = 2^8$, then the factorization matrices are of size $2^{16} \times 2^{16}$. Figure 2(a) shows the distribution of the nonzero entries of the factorization matrix. As seen from the figure, the factorization matrix has a sparse structure with an average number of nonzero entries per row being 42.2 (while $\mathbf{A}$ and $\mathbf{\Sigma}_i$ have around 9 nonzero entries per row on average). Recall that the solutions of the $\mathbf{A}$-subproblem and the $\mathbf{\Sigma}_i$-subproblem are obtained by a backward substitution and a forward substitution. This implies that solving these subproblems requires around $84.4n^2$ multiplications if the factorization matrices are precomputed. It is worth mentioning that as n^2 increases the average number of nonzero entries per row also increases as shown in Figure 2(b), and therefore the factorization based direct solvers have relatively heavier memory consumption and need more computational time. The MG method, in contrast, scales linearly with n^2 in both memory and computational time.

For this example, the results of the preconditioning are shown in Table 2, in which the running

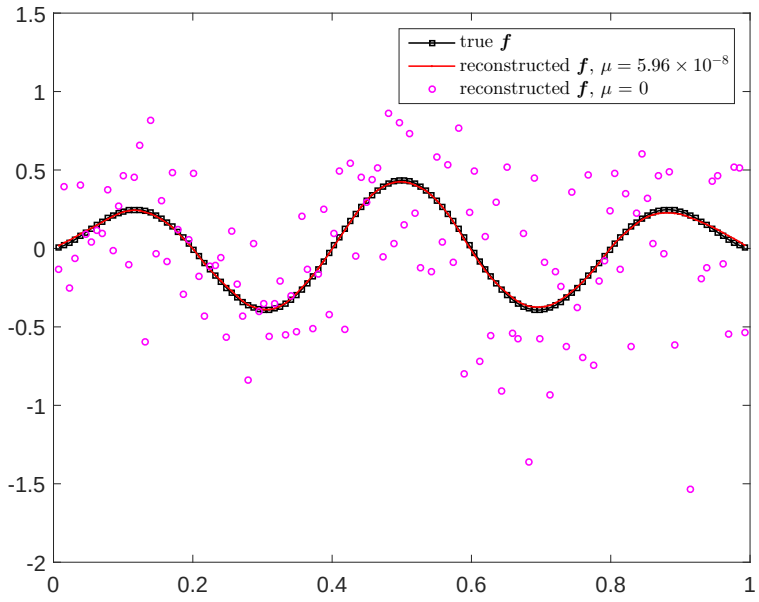
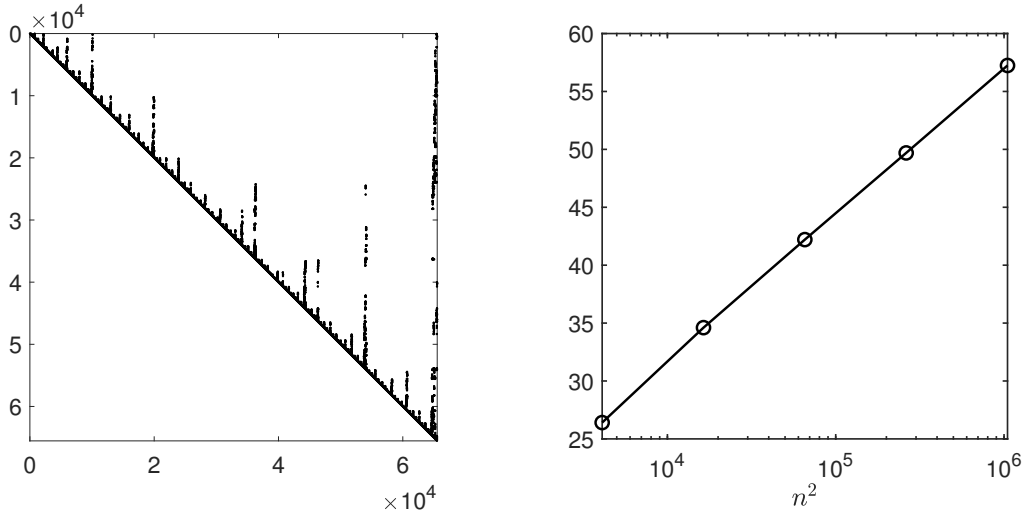


Figure 1: The true $\boldsymbol{f}$ and the reconstructed $\boldsymbol{f}$.



(a) the factorization matrix

(b) average number of nonzero entries per row

Figure 2: The nonzero entries of the matrices for Example 2. For (a), the factorization matrix ($n = 2^8$) have 42.2 nonzero entries per row on on average. For (b), the average number of nonzero entries per row is plotted versus n^2

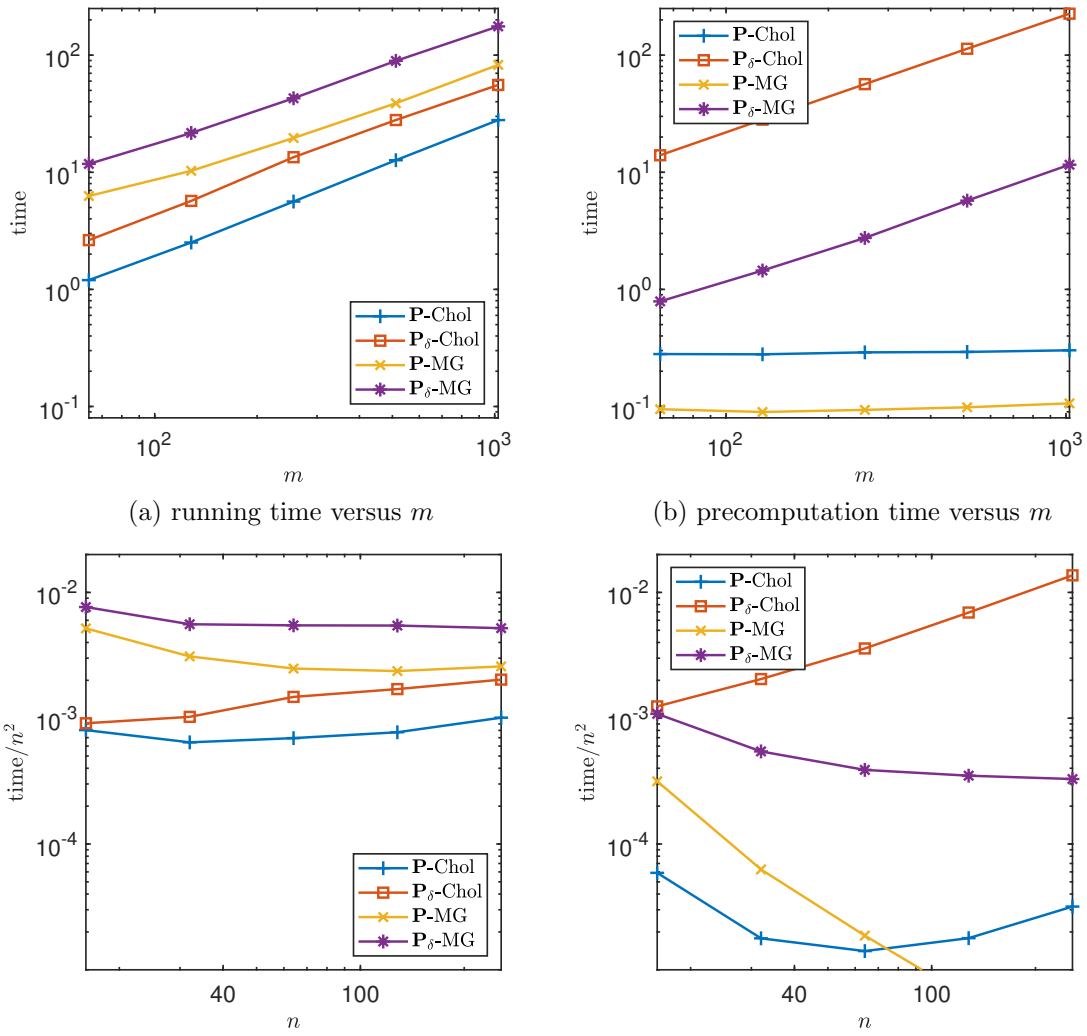
time for each method is averaged over 8 independent runs. We use "ITER" to stand for the number of iterations and "CPU" for the running time (excluding the precomputation time) of the methods. Results of the standard restarted GMRES method are also demonstrated for comparison. Similar

to the one-dimensional case in Example 1, both $\mathbf{P}$-Chol and $\mathbf{P}_\delta$-Chol converge in a few iterations, and the running time for these two methods is comparable. For larger m, $\mathbf{P}_\delta$-Chol needs slightly more iterations than $\mathbf{P}$-Chol due to the approximation errors in $\mathbf{B}_{11,\delta}$ for $\delta > 0$. Similar results hold for the preconditioners $\mathbf{P}_\delta$-MG and $\mathbf{P}$-MG. In general, $\mathbf{P}_\delta$-MG requires one or two more iterations than the $\mathbf{P}$-MG method. We observe that the increase in the running time of the former compared to the latter is not only due to the larger iteration numbers but also because of the complex diagonal entries of the matrices $\mathbf{\Sigma}_i$, which raises the computational time for the $\mathbf{\Sigma}_i$-subproblem by a small constant factor compared to solving the $\mathbf{A}$-subproblem (as needed by $\mathbf{P}$-MG). The methods discussed above are far more efficient than the baseline method GMRES(50) which never converges within 5000 iterations and is more than 50 times slower.

Table 2: The number of iterations "ITER" and the running time "CPU" (in second, averaged over 8 runs) of Example 2. The "$>$" means that the algorithm does not converge up to the indicated number of iterations or running time.

Grid sizes		GMRES(50)		$\mathbf{P}$-Chol		$\mathbf{P}_\delta$-Chol		$\mathbf{P}$-MG		$\mathbf{P}_\delta$-MG	
		ITER	CPU	ITER	CPU	ITER	CPU	ITER	CPU	ITER	CPU
$m = 2^6$	$n = 2^4$	>5000	>50	3	0.04	3	0.11	7	0.22	7	0.43
	$n = 2^5$		>130	3	0.08	3	0.11	7	0.52	7	0.78
	$n = 2^6$		>300	3	0.28	3	0.57	7	1.66	8	3.23
	$n = 2^7$		>1200	3	1.20	3	2.63	7	6.27	8	11.78
	$n = 2^8$		>5000	3	7.67	3	14.02	7	29.87	8	53.05
$m = 2^7$	$n = 2^4$	>5000	>120	3	0.06	3	0.07	7	0.37	7	0.49
	$n = 2^5$		>200	3	0.16	3	0.23	7	0.86	8	1.55
	$n = 2^6$		>600	3	0.55	3	1.11	7	2.66	8	5.65
	$n = 2^7$		>2600	3	2.51	3	5.69	7	10.29	8	21.65
	$n = 2^8$		>14000	3	14.66	3	27.97	7	46.45	8	91.21
$m = 2^8$	$n = 2^4$	>5000	>160	3	0.10	4	0.14	7	0.64	7	0.93
	$n = 2^5$		>300	3	0.30	4	0.51	7	1.63	8	2.92
	$n = 2^6$		>1300	3	1.28	4	2.94	7	5.04	8	11.04
	$n = 2^7$		>5000	3	5.62	4	13.44	7	19.60	8	42.83
	$n = 2^8$		>28000	3	32.51	4	66.77	7	86.05	8	169.92
$m = 2^9$	$n = 2^4$	>5000	>210	3	0.21	4	0.23	7	1.33	8	1.96
	$n = 2^5$		>600	3	0.66	4	1.05	7	3.17	8	5.72
	$n = 2^6$		>2600	3	2.84	4	6.05	7	10.16	8	22.41
	$n = 2^7$		>13000	3	12.67	4	27.91	7	38.83	8	89.30
	$n = 2^8$		>60000	3	66.17	4	132.77	7	169.12	8	340.54
$m = 2^{10}$	$n = 2^4$	>5000	>300	3	0.44	4	0.47	7	2.61	8	3.83
	$n = 2^5$		>1300	3	1.52	4	2.20	7	6.58	8	11.75
	$n = 2^6$		>5000	3	6.31	4	11.83	7	21.08	8	45.91
	$n = 2^7$		>29000	3	27.90	4	55.68	7	82.56	8	176.00
	$n = 2^8$		>130000	3	145.40	4	273.79	7	338.99	9	798.21

A further comparison of the performance of the methods is provided in Figure 3. In particular, Figure 3(a) and Figure 3(c) are the plots of the preconditioned iterative methods' running time (excluding the precomputation time) versus m (the grid size for the time domain) and versus n (the grid size for the spatial domain on each axis) respectively. The associated precomputation time of each method is displayed in Figure 3(b) and Figure 3(d). For Figure 3(a)-(b), n is fixed as 2^7, while for Figure 3(c)-(d), m is fixed as 2^9. We have the following remarks about the trade-off between the computational time and storage as well as the choice of the subproblem solvers.



(a) running time versus m

(b) precomputation time versus m

(c) running time versus n

(d) precomputation time versus n

Figure 3: The running time and the precomputation time with respect to m and n.

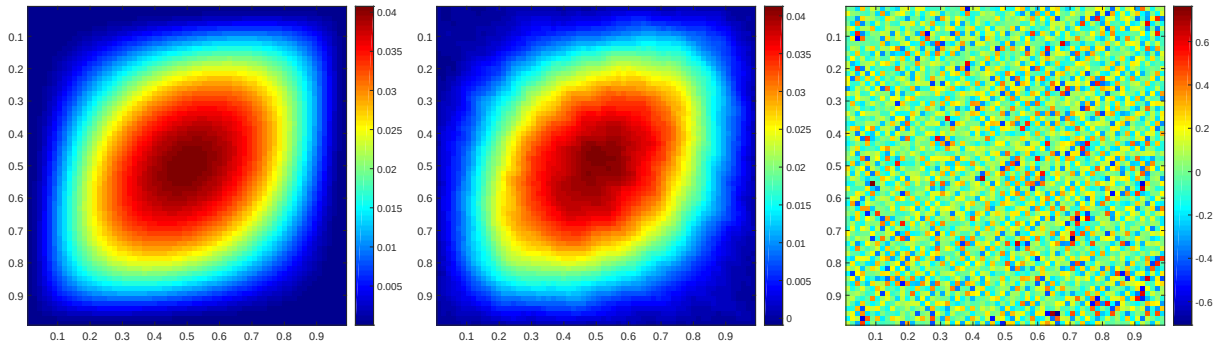
- *Precomputation.* Precomputation is often used to speed up iterative methods. For the $\mathbf{A}$- and $\mathbf{\Sigma}_i$-subproblems, the precomputation needs to be done once and hence saves the overall running time of the proposed methods. This comes at the cost of an increase in memory requirements. We note that for problems in 1D spatial domains, the precomputation time is always small, due to the fact that $\mathbf{A}$ and $\mathbf{\Sigma}_i$ are tridiagonal matrices and hence both subproblem solvers have linear time complexity and storage requirement. For 2D problems, the precomputation time of $\mathbf{P}_\delta$-Chol increases linearly with m. However, it does not scale linearly with the size of matrix $\mathbf{\Sigma}_i$, since the factorization matrices of the $\mathbf{\Sigma}_i$-subproblems have more nonzero entries per row for larger n (see Figure 2(b)). The memory consumption and precomputation time is thus more than the other three preconditioners as n and m increase. Therefore, for applications with restrictive memory, a more feasible option is to compute the factorization within the iteration or use the MG solvers instead. Notably, the precomputation

time for $\mathbf{P}$-MG, $\mathbf{P}_\delta$-MG, and $\mathbf{P}$-Chol is small compared to their running time, as shown in Figures 3(a)-(d). For $\mathbf{P}_\delta$-Chol, we observe from Figures 3(a)-(d) that its precomputation takes more time than the corresponding running time (see the red lines in the plots).

- *The subporblem solvers.* For the 1D problems, the tridiagonal structure of $\mathbf{A}$ and $\mathbf{\Sigma}_i$ makes the preconditioners $\mathbf{P}$-Chol and $\mathbf{P}_\delta$-Chol highly efficient. The MG solvers for the subproblems have linear complexity with a constant factor slower than the algorithms of the forward (backward) substitution associated with the triangular matrices. However, we note that the MG method can be easily parallelized if it is equipped with suitable smoothers. For 2D or higher dimensional problems, the direct subproblem solvers become less efficient compared to the MG methods as n increases. This fact is reflected by Figure 3(c) and Table 2, where the $\mathbf{P}$-Chol (resp. $\mathbf{P}_\delta$-Chol) needs less running time than that of $\mathbf{P}$-MG (resp. $\mathbf{P}_\delta$-MG) for small n, but their difference is getting smaller as n gets a bit larger. The experiment suggests that it is a good practice to choose $\mathbf{P}$-Chol or $\mathbf{P}_\delta$-Chol for small n and the MG based preconditioners for large scale problems.

In summary, the numbers of iterations required by the proposed preconditioners are small for all n and m, regardless of the choices of the subproblem solvers, as shown in Table 1 and Table 2. This implies that the preconditioner $\mathbf{P}$ is stable with respect to the errors from the block circulant approximation for (54) and also the errors from the inexact solutions by MG method for the $\mathbf{A}$-subproblems and $\mathbf{\Sigma}_i$-subproblems.

Next, we demonstrate the accuracy of the reconstructed solutions for the inverse problem. In this example, we select a grid size of $n = 2^6$ and $m = 2^9$. The reconstructed $\boldsymbol{f}$ from the noisy data $\boldsymbol{g}_v$ at the 3^{rd} iteration of the $\mathbf{P}$-Chol method is displayed in Figure 4(b). Though the computed $\boldsymbol{f}$ loses some local details of the ground truth $\boldsymbol{f}$, the improvement is dramatic from the naive reconstruction (without any regularization) which contains little information of the ground truth $\boldsymbol{f}$, as shown in Figure 4(c).



(a) true $\boldsymbol{f}$ (b) reconstructed $\boldsymbol{f}$ (c) reconstructed $\boldsymbol{f}$ $(\mu = 0)$.

Figure 4: The true $\boldsymbol{f}$ and its reconstructions. The image in (c) represents a reconstruction without regularization

4.2.1 The Effect of Different δ on the Convergence

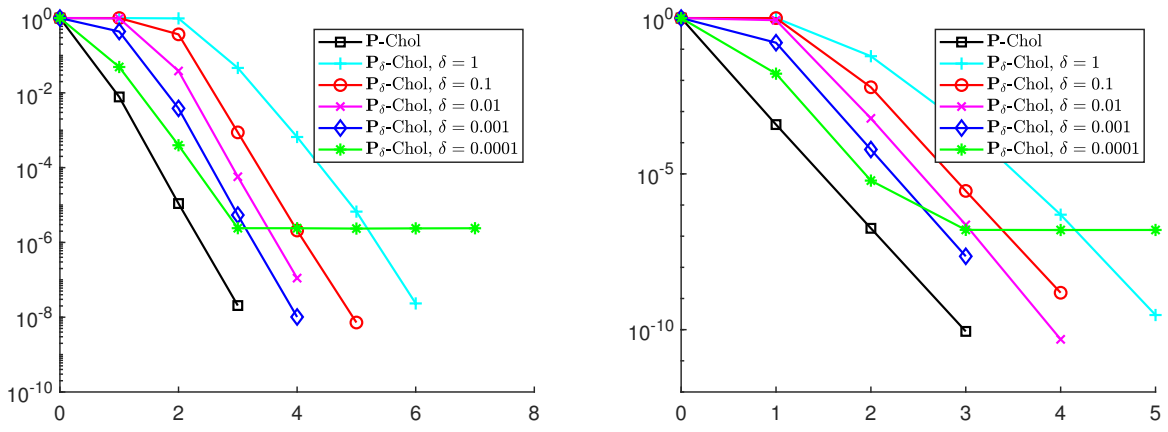
The performance of $\mathbf{P}_\delta$ depends on the values of δ. A smaller value of δ implies a better approximation $\mathbf{P}_\delta$ to $\mathbf{B}$, since the matrix $\mathbf{B}_{\delta,11}^{-1}$ converges to $\mathbf{B}_{11}^{-1}$ as $\delta \to 0$ which is indicated by Equation

(22). However, numerical instability of computing $\mathbf{P}_\delta^{-1}$ appears if δ is too small, and this reduces the quality of the preconditioner $\mathbf{P}_\delta$.

To study the influence of δ on the convergence speed of the preconditioning with $\mathbf{P}_\delta$, independent tests are carried out for $\delta = 10^0, 10^{-1}, 10^{-2}, 10^{-3}$, and 10^{-4} respectively. In this subsection, we choose the direct solvers for the $\mathbf{\Sigma}_i$-subproblems, i.e., the implementation of $\mathbf{P}_\delta$-Chol. The grid sizes (n, m) are set to $\left(2^7, 2^{14}\right)$ for Example 1 and $\left(2^6, 2^9\right)$ for Example 2, respectively. The convergence histories of the iterative methods for Example 1 and Example 2 are plotted in Figure 5(a) and Figure 5(b), respectively. For comparison, we also report the results of the preconditioner $\mathbf{P}$-Chol.

As shown in Figure 5, for $\delta = 1$, the $\mathbf{P}_\delta$-Chol method converges in 6 iterations for the 1D problem and 5 iterations for the 2D problem. As δ decreases from 1 to 10^{-4}, the convergence of the iterative method gets faster. However, when δ is very small (e.g., 10^{-4}), the method does not converge. As indicated by the green lines of the plots, the residual error does not decrease significantly after the solution reaches certain accuracy. This is due to the roundoff error during the numerical inversion of $\mathbf{P}_\delta$. In fact, to perform the preconditioning $\mathbf{P}_\delta$-Chol, we need to compute $\mathcal{R}(\mathbf{B}_\delta)^{-1}$ multiplying a given vector and hence the inverse of $\mathbf{B}_{11,\delta}$. The block diagonalization (18) has been used to establish the inverse $\mathbf{B}_{11,\delta}^{-1}$, where the involvement of $\mathbf{D}_\delta^{-1}$ causes numerical errors.

Based on the above observations and the numerical results, δ should not be chosen too small. However, for δ larger than 0.01, the number of iterations for the preconditioner $\mathbf{P}_\delta$-Chol does not increase dramatically as δ gets slightly larger. In practice, a typical choice of δ ranges from 0.01 to 1.



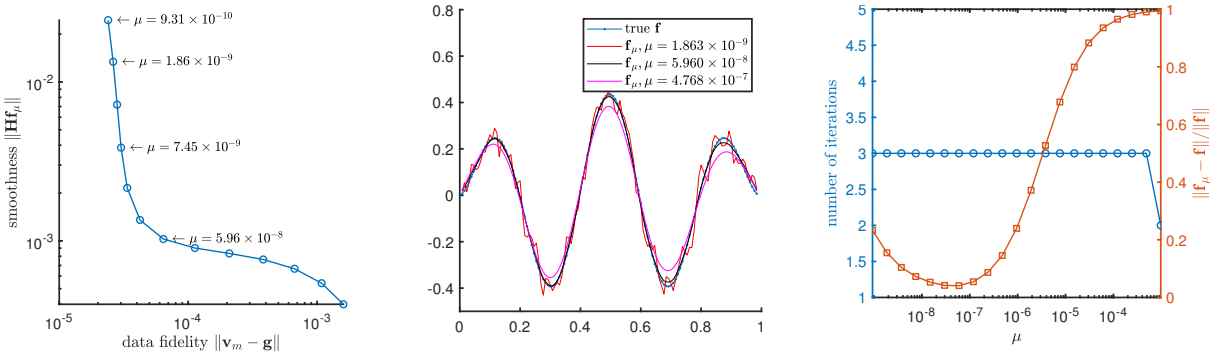
(a) Example 1 (b) Example 2

Figure 5: The relative residual errors measured in l_2 norm for different values of δ. The x-axis is the iteration number and the y-axis refers to the relative residual errors

4.2.2 The regularization parameter

Figure 1 and Figure 4 show that regularization ($\mu > 0$) is essential for obtaining accurate solutions to the inverse problem. However, one needs to avoid using a too large regularization parameter that over-smooths the solution. With a parameter that is too small, on the other hand, the solution is fitted to the noise in the measurements and therefore may not be accurate either. So a good

parameter should reflect a good trade-off point between the smoothness of the solution and the data fidelity. In the following discussions, we present a method for selecting the parameter μ without knowing the ground truth solutions or the level of the noise in the measured data. Besides, we investigate the convergence of the proposed preconditioning technique for different values of the parameter μ.



(a) smoothness versus data fidelity (b) reconstructions for different μ (c) convergences and errors

Figure 6: The selection of the parameter μ demonstrated from Example 1. (a). the data fidelity and smoothness as a function of μ. (b). Reconstructed $\boldsymbol{f}$ for three different choices of μ. (c). the number of iterations (in blue) and the reconstruction error (in orange) plotted versus μ

Various ways of choosing the regularization parameters exist. For determining the value of μ of the quasi-boundary regularization approach, we consider the pair

$$(\|\boldsymbol{v}_m - \boldsymbol{g}_v\|, \|\mathbf{H}\boldsymbol{f}_\mu\|) \tag{55}$$

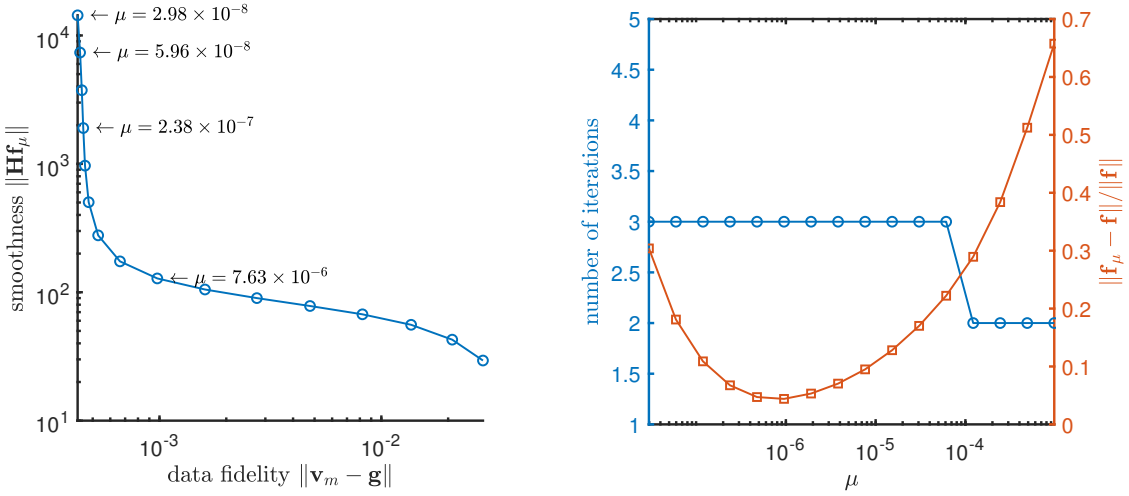
where $\boldsymbol{v}_m$ and $\boldsymbol{f}_\mu$ are solutions of Equation (13) given the parameter μ. The first entry of the pair reflects how well the computed solution fits the observed data $\boldsymbol{g}_v$, and the second entry measures the smoothness of the solution. Given a value of μ, the solutions $\boldsymbol{v}_m$ and $\boldsymbol{f}_\mu$ are computed numerically. We obtain the pair (55) for various μ and plot the curve associated with the pair parametrized by μ (see Figure 6(a)). The parameter μ is then selected to be the one at which the curve has maximum curvature. Note that this is similar to the L-curve method well-known for Tikhonov-type regularization, and the maximum curvature point is often referred to as the L-curve corner [38, 18].

Base on this criterion and the plot in Figure 6(a), in which solutions for $\boldsymbol{v}_m$ and $\boldsymbol{f}_\mu$ are computed with the preconditioner $\mathbf{P}$-Chol, we choose the parameter μ from between 2.98×10^{-8} and 5.96×10^{-8} for Example 1. Figure 6(b) shows that the solution at $\mu = 5.96 \times 10^{-8}$ fits the ground truth $\boldsymbol{f}$ well, while the larger μ imposes too much damping on the curve and the smaller μ suffers from heavy noise perturbation. The choice is in accordance with the minimum of the relative reconstruction errors $\|\boldsymbol{f}_\mu - \boldsymbol{f}\|_2/\|\boldsymbol{f}\|_2$ as shown by the orange curve in Figure 6(c). It is interesting to note that the convergence of the proposed preconditioner is stable with respect to the values of μ as shown by the blue curve in Figure 6(c). This result also agrees with the statements in Theorem 1 that the spectrum of the preconditioned system clusters around 1 regardless of the choice of μ.

We apply the same strategy to choosing the parameter in the two-dimensional problems. As shown in Figure 7(a), the curvature of the curve is maximized at a point with μ-value between 1.91×10^{-6} and 3.81×10^{-6}. The reconstruction errors demonstrated for different μ in Figure 7(b) suggest that the reconstruction associated with μ at this region has relative error $\|\boldsymbol{f}_\mu - \boldsymbol{f}\|_2/\|\boldsymbol{f}\|_2$

less than 0.1 and close to the optimal error at $\mu = 9.54 \times 10^{-7}$. Again, we observed that the values of μ do not significantly affect the number of iterations for the proposed preconditioner which is always within 3 steps as illustrated in Figure 7(b).

The regularization parameter selection method described above requires solving multiple systems associated with different values of μ. This is feasible only when we have a solver that is fast and robust to the choices of μ. The fast numerical methods developed in this paper significantly reduce the computation needed especially when solving multiple systems. In particular, the proposed preconditioning is shown to be very efficient for all μ theoretically in Section 3, and the preconditioned iterative methods converge in a few iterations in Subsection 4.2.



(a) smoothness versus data fidelity

(b) convergences and the reconstruction errors

Figure 7: The selection of the parameter μ and the iteration numbers for Example 2.

5 Conclusion

In this work, an inverse problem of identifying a source term depending on the spatial variables of the time fractional diffusion equation is studied. With a quasi-boundary value regularization, the problem is discretized into a 2-by-2 block structured linear system of equations. Numerical methods are developed for solving such a system. We propose a preconditioner by approximating the Schur complement with low degree rational functions related to the matrix in the diagonal blocks. The preconditioning can be carried out efficiently as fast TFDE solvers are available and the particular form of approximated Schur complement enables fast inversion. We also investigate the variants of the proposed preconditioner that reduce the computational cost at each iteration, based on approximate solutions of the TFDE and its subproblems. All preconditioners are tested and compared, and their time efficiency, memory requirements, and scalability are discussed. In addition, we present a method for choosing the regularization parameter which is demonstrated to work well in the 1D problems and the 2D problems.

Theoretically, we show that the GMRES method with the proposed preconditioning technique converges superlinearly on the non-symmetric system, based on the fact that the residual vectors lie

in an invariant subspace of a symmetric operator with spectrum clustered around 1. However, the analysis is valid only for the preconditioner with exact TFDE solvers. Though its variants show promising convergence results for different grid sizes and different choices of the regularization parameter, their convergence behavior is not theoretically understood. This issue will be of interest in our future work.

Acknowledgment. We thank the anonymous reviewers for their thorough review helping us to improve the paper.

References

[1] O. P. Agrawal, *Solution for a fractional diffusion-wave equation defined in a bounded domain*, Nonlinear Dynam., 29 (2002), pp. 145–155.

[2] K. Ames, G. Clark, J. Epperson, and S. Oppenheimer, *A comparison of regularizations for an ill-posed problem*, Math. Comp., 67 (1998), pp. 1451–1471.

[3] Z. Bai, G. Golub, and M. Ng, *Hermitian and skew-Hermitian splitting methods for non-Hermitian positive definite linear systems*, SIAM J. Matrix Anal. Appl., 24 (2003), pp. 603–626.

[4] M. Benzi, G. Golub, and J. Liesen, *Numerical solution of saddle point problems*, Acta Numer., 14 (2005), pp. 1–137.

[5] M. Benzi and M. Olshanskii, *Field-of-values convergence analysis of augmented lagrangian* preconditioners for the linearized navier–stokes problem, SIAM J. Numer. Anal., 49 (2011), pp. 770–788.

[6] M. Benzi and A. Wathen, *Some preconditioning techniques for saddle point problems*, In Model Order Reduction: Theory, Research Aspects and Applications, Math. Ind. 13, Springer-Verlag, Berlin, 2008, pp. 195–211.

[7] D. Bini, *Parallel solution of certain Toeplitz linear systems*, SIAM J. Comput., 13 (1984), pp. 268–276.

[8] D. Bini, S. Dendievel, G. Latouche, and B. Meini, *Computing the exponential of large block-triangular block-Toeplitz matrices encountered in fluid queues*, Linear Algebra Appl., 502 (2016), pp. 387–419.

[9] J. Bouchaud and A. Georges, *Anomalous diffusion in disordered media: statistical mechanisms, models and physical applications*, Phys. Rep., 195 (1990), pp. 127–293.

[10] W. Briggs, V. E. Henson, and S. F. McCormick, *A Multigrid Tutorial*, 2nd ed., SIAM, Philadelphia, 2000.

[11] G. Clark and S. Oppenheimer, *Quasireversibility methods for non-well-posed problems*, Electron. J. Differential Equations, 1994 (1994), pp. 1–9.

[12] T. Davis, J. Gilbert, S. Larimore, and E. Ng, *A column approximate minimum degree ordering algorithm*, ACM Trans. Math. Software, 30 (2004), pp. 353–376.

[13] M. Denche and K. Bessila, *A modified quasi-boundary value method for ill-posed problems*, J. Math. Anal. Appl., 301 (2005), pp. 419–426.

[14] M. Donatelli, M. Mazza, and S. Serra-Capizzano, *Spectral analysis and structure preserving preconditioners for fractional diffusion equations*, J. Comput. Phys., 307 (2016), pp. 262–279.

[15] H. Elman, D. Silvester, and A. Wathen, *Finite elements and fast iterative solvers: with applications in incompressible fluid dynamics*, 2nd ed., Numerical Mathematics and Scientific Computation, Oxford University Press, Oxford, 2014.

[16] F. Gaspar and C. Rodrigo, *Multigrid waveform relaxation for the time-fractional heat equation*, SIAM J. Sci. Comput., 39 (2017), pp. A1201–A1224.

[17] A. Greenbaum, V. Pták, and Z. Strakoš, *Any nonincreasing convergence curve is possible for gmres*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 465–469.

[18] P. Hansen and P. O'Leary, *The use of the L-curve in the regularization of discrete ill-posed problems*, SIAM J. Sci. Comput., 14 (1993), pp. 1487–1503.

[19] D. Hào, N. V. Duc, and H. Sahli, *A non-local boundary value problem method for parabolic equations backward in time*, J. Math. Anal. Appl., 345 (2008), pp. 805–815.

[20] R. Hilfer, *Applications of Fractional Calculus in Physics*, World Scientific, Singapore, 2000.

[21] I. Ipsen, *A note on preconditioning nonsymmetric matrices*, SIAM J. Sci. Comput., 23 (2001), pp. 1050–1051.

[22] H. Jiang, F. Liu, I. Turner, and K. Burrage, *Analytical solutions for the multi-term time–space Caputo–Riesz fractional advection–diffusion equations on a finite domain*, J. Math. Anal. Appl., 389 (2012), pp. 1117–1127.

[23] S. Jiang, J. Zhang, Q. Zhang, and Z. Zhang, *Fast evaluation of the caputo fractional derivative and its applications to fractional diffusion equations*, Commun. Comput. Phys., 21 (2017), pp. 650–678.

[24] B. Jin and W. Rundell, *A tutorial on inverse problems for anomalous diffusion processes*, Inverse Problems, 31 (2015), 035003.

[25] R. Ke, M. Ng, and H. Sun, *A fast direct method for block triangular Toeplitz-like with tri-diagonal block systems from time-fractional partial differential equations*, J. Comput. Phys., 303 (2015), pp. 203–211.

[26] S. Lei and H. Sun, *A circulant preconditioner for fractional diffusion equations*, J. Comput. Phys., 242 (2013), pp. 715–725.

[27] F. Lin, W. Ching, and M. Ng, *Fast inversion of triangular Toeplitz matrices*, Theoret. Comput. Sci., 315 (2004), pp. 511–523.

[28] Y. Lin and C. Xu, *Finite difference/spectral approximations for the time-fractional diffusion equation*, J. Comput. Phys., 225 (2007), pp. 1533–1552.

[29] X. Lu, H. Pang, and H. Sun, *Fast approximate inversion of a block triangular Toeplitz matrix with applications to fractional sub-diffusion equations*, Numer. Linear Algebra Appl., 22 (2015), pp. 866–882.

[30] Y. Ma, P. Prakash, and A. Deiveegan, *Generalized Tikhonov methods for an inverse source problem of the time-fractional diffusion equation*, Chaos Solitons Fractals, 108 (2018), pp. 39–48.

[31] R. Metzler and J. Klafter, *The random walk's guide to anomalous diffusion: a fractional dynamics approach*, Phys. Rep., 339 (2000), pp. 1–77.

[32] M. Murphy, G. Golub, and A. Wathen, *A note on preconditioning for indefinite linear systems*, SIAM J. Sci. Comput., 21 (2000), pp. 1969–1972.

[33] H. Nguyen, D. Le, and V. Nguyen, *Regularized solution of an inverse source problem for a time fractional diffusion equation*, Appl. Math. Model., 40 (2016), pp. 8244–8264.

[34] Y. Notay, *A new analysis of block preconditioners for saddle point problems*, SIAM J. Matrix Anal. Appl., 35 (2014), pp. 143–173.

[35] J. Pan, R. Ke, M. Ng, and H. Sun, *Preconditioning techniques for diagonal-times-Toeplitz matrices in fractional diffusion equations*, SIAM J. Sci. Comput., 36 (2014), pp. A2698–A2719.

[36] V. Pan, *Complexity of computations with matrices and polynomials*, SIAM Rev., 34 (1992), pp. 225–262.

[37] I. Podlubny, *Fractional Differential Equations: An Introduction to Fractional Derivatives, Fractional Differential Equations, to Methods of Their Solution and Some of Their Applications*, Math. Sci. Engrg. 198, Academic Press Inc., San Diego, CA, 1999.

[38] T. Regińska, *A regularization parameter in discrete ill-posed problems*, SIAM J. Sci. Comput., 17 (1996), pp. 740–749.

[39] Y. Saad, *Iterative Methods for Sparse Linear Systems*, 2nd ed., SIAM, Philadelphia, 2003.

[40] S. Samko, A. Kilbas, and O. Marichev, *Fractional Integrals and Derivatives: Theory and Applications*, Gordon and Breach Science Publishers, 1993.

[41] T. Solomon, E. Weeks, and H. Swinney, *Observation of anomalous diffusion and lévy flights in a two-dimensional rotating flow*, Phys. Rev. Lett., 71 (1993), pp. 3975–3978.

[42] N. Tuan, L. Long, V. Nguyen, and T. Tran, *On a final value problem for the time-fractional diffusion equation with inhomogeneous source*, Inverse Probl. Sci. Eng., 25 (2017), pp. 1367–1395.

[43] J. Wang and T. Wei, *Quasi-reversibility method to identify a space-dependent source for the time-fractional diffusion equation*, Appl. Math. Model., 39 (2015), pp. 6139–6149.

[44] J. Wang, Y. Zhou, and T. Wei, *A posteriori regularization parameter choice rule for the quasi-boundary value method for the backward time-fractional diffusion problem*, Appl. Math. Lett., 26 (2013), pp. 741–747.

[45] J. Wang, Y. Zhou, and T. Wei, *Two regularization methods to identify a space-dependent source for the time-fractional diffusion equation*, Appl. Numer. Math., 68 (2013), pp. 39–57.

[46] W. Wang, M. Yamamoto, and B. Han, *Numerical method in reproducing kernel space for an inverse source problem for the fractional diffusion equation*, Inverse Problems, 29 (2013), 095009.

[47] A. Wathen, *Preconditioning*, Acta Numer., 24 (2015), pp. 329–376.

[48] T. Wei and J. Wang, *A modified quasi-boundary value method for the backward time-fractional diffusion problem*, ESAIM Math. Model. Numer. Anal., 48 (2014), pp. 603–621.

[49] T. Wei and J. Wang, *A modified quasi-boundary value method for an inverse source problem of the time-fractional diffusion equation*, Appl. Numer. Math., 78 (2014), pp. 95–111.

[50] F. Yang, Y. Zhang, X. Li, and C. Huang, *The quasi-boundary value regularization method for identifying the initial value with discrete random noise*, Bound. Value Probl., 2018 (2018), pp. 108.

[51] P. Zhuang and F. Liu, *Implicit difference approximation for the time fractional diffusion equation*, J. Appl. Math. Comput., 22 (2006), pp. 87–99.

[52] Z. Zhang and T. Wei, *Identifying an unknown source in time-fractional diffusion equation by a truncation method*, Appl. Math. Comput., 219 (2013), pp. 5972–5983.