Journal of Computational Physics 303 (2015) 203–211

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

A fast direct method for block triangular Toeplitz-like with tri-diagonal block systems from time-fractional partial differential equations [☆]

Rihuan Ke [a], Michael K. Ng [b,*], Hai-Wei Sun [c]

[a] School of Mathematics, South China Normal University, China
[b] Department of Mathematics, Hong Kong Baptist University, Hong Kong
[c] Department of Mathematics, University of Macau, Macao

ARTICLE INFO

Article history:
Received 1 May 2015
Received in revised form 10 September 2015
Accepted 24 September 2015
Available online xxxx

Keywords:
Block triangular Toeplitz-like matrix
Direct methods
Divide-and-conquer strategy
Fast Fourier transform
Fractional partial differential equations

ABSTRACT

In this paper, we study the block lower triangular Toeplitz-like with tri-diagonal blocks system which arises from the time-fractional partial differential equation. Existing fast numerical solver (e.g., fast approximate inversion method) cannot handle such linear system as the main diagonal blocks are different. The main contribution of this paper is to propose a fast direct method for solving this linear system, and to illustrate that the proposed method is much faster than the classical block forward substitution method for solving this linear system. Our idea is based on the divide-and-conquer strategy and together with the fast Fourier transforms for calculating Toeplitz matrix–vector multiplication. The complexity needs $\mathcal{O}(MN\log^2 M)$ arithmetic operations, where M is the number of blocks (the number of time steps) in the system and N is the size (number of spatial grid points) of each block. Numerical examples from the finite difference discretization of time-fractional partial differential equations are also given to demonstrate the efficiency of the proposed method.

1. Introduction

Consider a linear system

$$\mathbf{A}\mathbf{u} = \mathbf{b}, \tag{1.1}$$

where $\mathbf{A}$ is the block lower triangular Toeplitz-like with tri-diagonal block (BL3TB-like) matrix of the form

$$\mathbf{A} = \begin{bmatrix} A_1^{(1)} & & & \\ A_2 & A_1^{(2)} & & \\ \vdots & \ddots & \ddots & \\ A_M & \cdots & A_2 & A_1^{(M)} \end{bmatrix}, \tag{1.2}$$

☆ The research was supported by research grants HKRGC GRF 12301214, MYRG102(Y2-L3)-FST13-SHW from University of Macau and 105/2012/A3 from FDCT of Macau.

* Corresponding author.
E-mail addresses: krhuan@gmail.com (R. Ke), mng@math.hkbu.edu.hk (M.K. Ng), HSun@umac.mo (H.-W. Sun).

http://dx.doi.org/10.1016/j.jcp.2015.09.042
0021-9991/

in which $A_1^{(1)}, A_1^{(2)}, \ldots, A_1^{(M)}$, and A_j $(j = 2, \ldots, M)$ are N-by-N tri-diagonal matrices, $\mathbf{u}$ is the unknown vector, and $\mathbf{b}$ is the right hand side vector. We remark that $A_1^{(1)}, A_1^{(2)}, \ldots, A_1^{(M)}$ are not necessarily the same. Such a linear system arises from the finite difference discretization of time-fractional partial differential equation; see [14,16,39,38,35,13,28,34] and Section 3. In the literature, there are many applications in time-fractional partial differential equations [26], for instances, chaotic dynamics of classical conservative systems [32], groundwater contaminant transport [2,3], turbulent flow [7,29], biology [23], finance [27], image processing [1], and physics [30]. Recent numerical methods for time-fractional partial differential equations can be found in [38,35,13,28,34,10–12,20,33,37].

Toeplitz matrices emerge from numerous topics such as signal and image processing, numerical solutions of partial differential equations and integral equations, as well as queueing networks; see [8,9] and the references therein. In particular, the triangular Toeplitz matrix plays a key role in the displacement representation of general Toeplitz matrices, which is fundamental in the study of structured matrices and polynomial computations; see [4,5,19,25].

Traditionally, the block forward substitution (BFS) method [18] can be straightforwardly applied to solve (1.1) in $\mathcal{O}(M^2N)$ arithmetic operations with $\mathcal{O}(MN)$ storage requirement. In order to reduce the computational cost, Lu, Pang, and Sun [22] recently proposed an approximate inversion method for solving (1.1) with $A_1^{(k)} = A_1$ for $1 \le k \le M$, where the coefficient matrix therefore becomes a block lower triangular Toeplitz with tri-diagonal block (BL3TB) matrix. Their idea is to make use of block-Toeplitz structure and approximate the BL3TB matrix by the block ϵ-circulant matrix that can be block-diagonalized by the fast Fourier transform (FFT) into a diagonal block matrix, in which each diagonal block is still tri-diagonal. The total computational complexity by their method is of $\mathcal{O}(MN\log M)$ arithmetic operations that is much cheaper than $\mathcal{O}(M^2N)$ arithmetic operations by the BFS method; see more details in [22]. Nevertheless, their approximate inversion method will be no longer available if $\mathbf{A}$ in (1.1) is not exact BL3TB; i.e., $A_1^{(k)}$ in (1.2) are different.

Another direct method for solving BL3TB system is the so-called block divide-and-conquer method [4,15]. Let $\mathbf{A}_{2k}$ be a BL3TB matrix and partitioned as

$$\mathbf{A}_{2k} = \begin{bmatrix} \mathbf{A}_k & \mathbf{0} \\ \mathbf{B}_k & \mathbf{A}_k \end{bmatrix}, \tag{1.3}$$

where $\mathbf{A}_k$ is still BL3TB and $\mathbf{B}_k$ is block Toeplitz. Thus, we have [4,15]

$$\mathbf{A}_{2k}^{-1} = \begin{bmatrix} \mathbf{A}_k^{-1} & \mathbf{0} \\ \mathbf{C}_k & \mathbf{A}_k^{-1} \end{bmatrix} \quad \text{with} \quad \mathbf{C}_k = -\mathbf{A}_k^{-1}\mathbf{B}_k\mathbf{A}_k^{-1}. \tag{1.4}$$

If $\mathbf{A}_k^{-1}$ is known, the only task is to compute $\mathbf{C}_k$, which is also a block Toeplitz matrix. Finally, $\mathbf{A}_M^{-1}$ can be recursively obtained from the inverse of a very small size matrix. In other words, the divide-and-conquer method is to compute the inverse of $\mathbf{A}_M$ exactly. The computational cost is of $\mathcal{O}(N^2M\log M + N^3M)$ and storage requirement is of $\mathcal{O}(N^2M)$ [4] since the inverse of a tri-diagonal matrix is usually dense. Therefore, the divide-and-conquer method may not be better than the BFS method for solving the BL3TB system if the block size N is large. Moreover, the divide-and-conquer method also cannot be applied for the BL3TB-like matrix if $A_1^{(k)}$ in (1.2) are different.

The main contribution of this paper is to propose a fast direct method for solving (1.2). Existing fast numerical solver (e.g., fast approximate inversion method) cannot handle such linear system as the main diagonal blocks are different. We illustrate that the proposed method is much faster than the classical block forward substitution method for solving this linear system. Our idea is to combine the BFS method with the divide-and-conquer strategy, to solve the BL3TB-like system in (1.1). As the divide-and-conquer method, the BL3TB-like matrix (1.2) is partitioned analogously to (1.3). Unlike the divide-and-conquer method, the inverse of the large size matrix is not calculated as that in (1.4). In our proposal, the partition is employed to reduce the original linear system into two half-size linear systems. Then the BFS method is exploited to solve both linear systems together. Meanwhile, the FFT can be applied to speed up the computation of the right hand side in the second half-size linear system as it contains a block Toeplitz matrix. Both half-size BL3TB-like matrices could be further reduced the matrix size by half until the matrices with small enough size are reached. Finally, the solution of (1.1) is recursively obtained. The computational complexity of the proposed method is of $\mathcal{O}(MN\log^2 M)$ operations which is cheaper than $\mathcal{O}(M^2N)$ operations of the classical BFS method. We remark that our method is of $\mathcal{O}(MN)$ storage requirement and it can be applied to the general BL3TB-like system. Numerical examples are given to illustrate the efficiency of the proposed method.

The outline of this paper is given as follows. In Section 2, we present the proposed algorithm. In Section 3, we consider coefficient matrices constructed by the finite difference discretization of time-fractional partial differential equations. In Section 4, we report experimental results and compare the proposed method with the other testing methods. Finally, concluding remarks are given in Section 5.