Applied Mathematics Letters 42 (2015) 53–58

Contents lists available at ScienceDirect

Applied Mathematics Letters

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

On k-step CSCS-based polynomial preconditioners for Toeplitz linear systems with application to fractional diffusion equations☆

Xian-Ming Gu [a,b], Ting-Zhu Huang [a,*], Hou-Biao Li [a], Liang Li [a], Wei-Hua Luo [a]

[a] School of Math. Sci., University of Electronic Science and Technology of China, Chengdu, 611731, China

[b] Institute of Math. Comput. Sci., University of Groningen, Nijenborgh 9, P.O. Box 407, 9700 AK Groningen, The Netherlands

ARTICLE INFO

Article history:
Received 24 September 2014
Received in revised form 4 November 2014
Accepted 4 November 2014
Available online 22 November 2014

Keywords:
Fractional diffusion equation
CSCS iteration method
Toeplitz matrix
Polynomial preconditioner
Fast Fourier transform

ABSTRACT

The implicit finite difference scheme with the shifted Grüwald formula for discretizing the fractional diffusion equations (FDEs) often results in the ill-conditioned non-Hermitian Toeplitz systems. In the present paper, we consider to solve such Toeplitz systems by exploiting the preconditioned GMRES method. A k-step polynomial preconditioner is designed based on the circulant and skew-circulant splitting (CSCS) iteration method proposed by Ng (2003). Theoretical and experimental results involving numerical solutions of FDEs demonstrate that the proposed k-step preconditioner is efficient to accelerate the GMRES solver for non-Hermitian Toeplitz systems.

1. Introduction

In this paper, we are interested in the iterative solvers for non-Hermitian Toeplitz systems arising in the numerical solutions of FDEs via finite difference method; see [1,2] and references therein. We also denote the Toeplitz system as the following form of matrices product

$$A\boldsymbol{u} = \boldsymbol{f}, \quad A \in \mathbb{C}^{n\times n} \text{ nonsingular, and } \boldsymbol{u}, \boldsymbol{f} \in \mathbb{C}^n, \tag{1}$$

where A is said to be Toeplitz if $A = [A_{ij}]_{i,j=1}^n = [a_{i-j}]$, $i \geq j$, i.e., A is constant along its diagonals; see [3,4]. Toeplitz systems also come from a variety of other applications in mathematics and engineering, see for instance the references in [3,4]. These applications have motivated both mathematicians and engineers to develop specific algorithms catering to solving Toeplitz systems. As we know, the Krylov subspace methods require in each iteration step only products of A with vector and since A is Toeplitz these products can be computed in $\mathcal{O}(n \log n)$ operations via using the fast Fourier transforms (FFTs). However, in order to reduce the number of iterations, iterative method must be chosen with suitable preconditioning in general. The construction of "efficient" preconditioners is the purpose of this paper. Although there exists a rich literature on Hermitian Toeplitz systems (see [3,4] and the references therein), only a few papers consider the non-Hermitian case [5–13]. For Hermitian positive definite Toeplitz matrices, the spectra of the circulant preconditioned Toeplitz matrices are shown to be clustered. It is clear how this affects the convergence of the PCG method. However, for the non-Hermitian case, it is not clear how the clustered eigenvalues affect the convergence of the Krylov subspace methods [3, pp. 74–79]. So searching the efficient preconditioners for non-Hermitian Toeplitz systems is still a promising topic.

☆ This research is supported by 973 Program (2013CB329404), NSFC (61370147, 61170311, 61170309 and 11301057) and the Fundamental Research Funds for Central Universities.

* Corresponding author.
E-mail addresses: guxianming@live.cn, x.m.gu@rug.nl (X.-M. Gu), tingzhuhuang@126.com (T.-Z. Huang).

http://dx.doi.org/10.1016/j.aml.2014.11.005
0893-9659/

The polynomial preconditioner is a kind of popular and interesting preconditioner in which the main issue is that efficient splitting of the coefficient matrix A is required and a suitable splitting iteration method needs to be used; see, for instance, [14–18]. Suppose that $A = M - N$ represents a splitting of matrix A and $\Omega = M^{-1}N$, where M is nonsingular. If the spectral radius of Ω, denoted by $\rho(\Omega)$, is less than one, i.e., $\rho(\Omega) < 1$, the inverse of A can be written as $A^{-1} = (\sum_{j=0}^{\infty} \Omega^j)M^{-1}$. We take $P_k = M(I+\Omega+\Omega^2+\cdots+\Omega^{k-1})^{-1}$ as an approximation to matrix A. Then P_k can be used as a preconditioner for linear system (1). We refer to such a preconditioner P_k as a polynomial preconditioner for A. The preconditioned matrix is given by

$$P_k^{-1}A = (I + \Omega + \Omega^2 + \cdots + \Omega^{k-1})M^{-1}A = I - \Omega^k. \tag{2}$$

In preconditioned Krylov subspace methods, the main computational cost is to solve the generalized residual equation $P_k\boldsymbol{z} = \boldsymbol{r}$ when P_k is applied as a preconditioner. It follows from (2) that $\boldsymbol{z} = (I + \Omega + \Omega^2 + \cdots + \Omega^{k-1})M^{-1}\boldsymbol{r}$. To obtain the vector $\boldsymbol{z}$, we perform a k-step iteration as follows:

$$M\boldsymbol{z}^{(j)} = N\boldsymbol{z}^{(j-1)} + \boldsymbol{r}, \quad j = 1, 2, \ldots, k. \tag{3}$$

Thus,

$$\boldsymbol{z}^{(m)} = \Omega^k\boldsymbol{z}^{(0)} + (I + \Omega + \Omega^2 + \cdots + \Omega^{k-1})M^{-1}\boldsymbol{r}. \tag{4}$$

If we choose $\boldsymbol{z}^{(0)} = \boldsymbol{0}$ in (4), then $\boldsymbol{z}^{(m)} = \left(I + \Omega + \Omega^2 + \cdots + \Omega^{k-1}\right)M^{-1}\boldsymbol{r} = \boldsymbol{z}$. A polynomial preconditioner is also called a k-step polynomial preconditioner, and it can also be derived from the two-stage iteration methods of trivial outer splittings, refer to [19] for this discussion. Based on the foregoing analysis, the effectiveness of a polynomial preconditioner depends upon a good splitting of matrix A, so that the iteration procedure (3) can be implemented efficiently. The k-step multisplitting preconditioners are commonly applied to parallel and vector machines to solve a parallel linear system [17, 18]. In this paper, by utilizing the efficient CSCS iteration method introduced in [20], we propose a CSCS-based polynomial preconditioner for GMRES method to solve the Toeplitz systems arising in numerical solutions of FDEs.

In Section 2, the CSCS iteration method is briefly reviewed. In Section 3, the polynomial preconditioner based on the CSCS iteration method is established and the spectrum of preconditioned matrix is theoretically analyzed. In Section 4, numerical experiments involving the numerical solutions of FDEs are reported to show the effectiveness of the proposed method. Finally, the paper closes with conclusions in Section 5.

2. The CSCS iteration method

Recently, Ng designed in [20] a circulant and skew-circulant splitting (CSCS) iteration method, which is very efficient for solving the non-Hermitian Toeplitz systems. Based on the fact that the Toeplitz matrix A in (1) naturally possesses circulant/skew-circulant splitting $A = C + S$, where

$$C = \frac{1}{2}\begin{bmatrix} a_0 & a_{-1}+a_{n-1} & \cdots & a_{2-n}+a_2 & a_{1-n}+a_1 \\ a_1+a_{1-n} & a_0 & \cdots & \cdots & a_{2-n}+a_2 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ a_{n-2}+a_2 & \cdots & \cdots & a_0 & a_{-1}+a_{n-1} \\ a_{n-1}+a_{-1} & a_{n-2}+a_{-2} & \cdots & a_1+a_{1-n} & a_0 \end{bmatrix} \tag{5}$$

and

$$S = \frac{1}{2}\begin{bmatrix} a_0 & a_{-1}-a_{n-1} & \cdots & a_{2-n}-a_2 & a_{1-n}-a_1 \\ a_1-a_{1-n} & a_0 & \cdots & \cdots & a_{2-n}-a_2 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ a_{n-2}-a_2 & \cdots & \cdots & a_0 & a_{-1}-a_{n-1} \\ a_{n-1}-a_{-1} & a_{n-2}-a_{-2} & \cdots & a_1-a_{1-n} & a_0 \end{bmatrix}. \tag{6}$$

Note that C is a circulant matrix and S is a skew-circulant matrix. A circulant matrix can be diagonalized by the discrete Fourier matrix F and a skew-circulant matrix can also be diagonalized via a discrete Fourier matrix with diagonal scaling, i.e., $\hat{F} = F\Omega$. That is to say, it holds that $C = F^*\Lambda_C F$ and $S = \hat{F}^*\Lambda_S\hat{F}$, where $F = (F)_{j,k} = \frac{1}{\sqrt{n}}e^{\frac{2\pi\iota}{n}(j-1)(k-1)}$, $\Omega = \mathrm{diag}(1, e^{-\frac{\pi\iota}{n}}, \ldots, e^{\frac{-(n-1)\pi\iota}{n}})$, $1 \le j, k \le n$ and ι is the imaginary unit [3, pp. 37–39]. Λ_C and Λ_S are diagonal matrices formed by the eigenvalues of C and S, respectively, which can be obtained in $\mathcal{O}(n\log n)$ operations by using the FFTs. Furthermore, the CSCS iteration method can be algorithmically described as follows:

Let $\boldsymbol{u}^{(0)} \in \mathbb{C}^n$ be an arbitrary initial guess. For $j = 0, 1, 2, \ldots$ until the sequence of iterates $\{\boldsymbol{u}^{(j)}\}_{j=0}^{\infty} \subset \mathbb{C}^n$ converges, compute the next iterate $\boldsymbol{u}^{(j+1)}$ according to the following procedure:

$$\begin{cases} (\alpha I + C)\boldsymbol{u}^{(j+\frac{1}{2})} = (\alpha I - S)\boldsymbol{u}^{(j)} + \boldsymbol{f}, \\ (\alpha I + S)\boldsymbol{u}^{(j+1)} = (\alpha I - C)\boldsymbol{u}^{(j+\frac{1}{2})} + \boldsymbol{f}, \end{cases}$$

where α is a given positive constant and I represents the identity matrix.