Fast Preconditioned Iterative Method for the Space Fractional Complex Ginzburg-Landau Equation

Lu Zhang[1(✉)], Lei Chen[2], and Wenyu Zhou[1]

[1] School of Mathematics and Statistics,
Xuzhou University of Technology, Xuzhou 221018, Jiangsu, China
yulu7517@126.com
[2] School of Information Engineering, Xuzhou University of Technology,
Xuzhou 221018, Jiangsu, China
chenlei@xzit.edu.cn

Abstract. In this work, we give an effective preconditioned numerical method to solve the discreted linear system, which is obtained from the space fractional complex Ginzburg-Landau equation. The coefficient matrix of the linear system is the sum of a symmetric tridiagonal matrix and a complex Toeplitz matrix. The preconditioned iteration method has computational superiority since we can use the fast Fourier transform (FFT) and the circulant preconditioner to solve the discreted linear system. Numerical examples are tested to illustrate the advantage of the proposed preconditioned numerical method.

Keywords: Space fractional Ginzburg-Landau equation · Toeplitz matrix · Preconditioned numerical method

1 Introduction

In this paper, we solve the space fractional complex Ginzburg-Landau equation as follows [39]

$$\frac{\partial v}{\partial t} + (\nu_1 + \mathbf{i}\eta_1)(-\Delta)^{\frac{\beta}{2}} v + (\kappa_1 + \mathbf{i}\zeta_1)|v|^2 v - \gamma_1 v = 0, \tag{1}$$

$$v(x,0) = v_0(x), \tag{2}$$

where $x \in \mathbb{R}$, $1 < \beta \leqslant 2$, $\mathbf{i}$ is the imaginary unit, $0 < t \leqslant T_1$, $v(x,t)$ is a complex-value function, $\nu_1 > 0$, $\kappa_1 > 0$, η_1, ζ_1, and γ_1 are real constants, and $v_0(x)$ is an initial function. Furthermore, the operator $(-\Delta)^{\frac{\beta}{2}} v(x,t)$ $(1 < \beta \leqslant 2)$ in (1) is defined [9] as follows

Supported by the "Peiyu" Project from Xuzhou University of Technology (Grant Number XKY2019104).

H. Song and D. Jiang (Eds.): SIMUtools 2020, LNICST 369, pp. 78–89, 2021.
https://doi.org/10.1007/978-3-030-72792-5_8

$$-(-\Delta)^{\frac{\beta}{2}}v(x,t) = -\frac{\frac{\partial^2}{\partial x^2}\int_{-\infty}^{\infty}|x-\xi|^{1-\beta}v(\xi,t)\mathrm{d}\xi}{2\cos(\frac{\beta\pi}{2})\Gamma(2-\beta)}. \tag{3}$$

The operator $(-\Delta)^{\frac{\beta}{2}}$ is equivalent to

$$-(-\Delta)^{\frac{\beta}{2}}v(x,t) = -\frac{-_{\infty}\hat{D}_x^{\beta}v(x,t) +_x \hat{D}_{+\infty}^{\beta}v(x,t)}{2\cos(\frac{\beta\pi}{2})},$$

where the two operators $_{-\infty}\hat{D}_x^{\beta}$ and $_x\hat{D}_{+\infty}^{\beta}$ are defined in [32].

The fractional Ginzburg-Landau equation has been used to describe a lot of physical phenomena; see [29,30,35]. However, there are few works on the numerical methods for the fractional complex Eq. (1)–(2) [10,13,33,38,39]. Based on the extensive application background of this equation, it is interesting to study the numerical methods for solving the fractional complex Eq. (1)–(2).

Recently, some new approaches are proposed to improve network routing and measurement [14,37,41]. Based on effective user behavior and traffic analysis methods [4,5,17,19], new scheduling strategies are designed to raise resources utilization [6,18,20,36] and energy-efficiency [21,22]. To test these new scheduling strategies, traffic Reconstruction is important [15,16,23,24,27,40]. Fluid model is effective model to reconstruct the bursty data traffic. Moreover, fractional differential equation can be used to build the fluid model. In our paper, we develop an effective preconditioned numerical method to solve the linear system, which is discreted from the fractional complex Eq. (1)–(2). Compared to the direct method, the complex linear systems can be fast solved by the circulant matrix and the FFT at each step due to the Toeplitz structure of coefficient matrices.

2 A Finite Difference Scheme

In this part, we exploit the fourth-order finite difference scheme [39] to discretize the fractional complex Eq. (1)–(2). For the two operators $_{-\infty}\hat{D}_x^{\beta}$ and $_x\hat{D}_{+\infty}^{\beta}$, the WSGD method [12] is used to approximate them. The shifted Grunwald formulae [28] is defined as

$${}_L\tilde{\mathcal{A}}_{\hat{h},p_1}^{\beta}v(x) = \frac{\sum_{i=0}^{+\infty}d_i^{(\beta)}v(x-(i-p_1)\hat{h})}{\hat{h}^{\beta}},$$

$${}_R\tilde{\mathcal{A}}_{\hat{h},q_1}^{\beta}v(x) = \frac{\sum_{i=0}^{+\infty}d_i^{(\beta)}v(x+(i-q_1)\hat{h})}{\hat{h}^{\beta}},$$

where p_1, q_1 are positive integers and the coefficients $d_i^{(\beta)}$ are computed as follows

$$d_0^{(\beta)} = 1,\ d_i^{(\beta)} = \frac{i-\beta-1}{i}d_{i-1}^{(\beta)},\ i \in \mathbb{Z}^+.$$

According to the reference [12] and using the shifted Grunwald formulae, the WSGD operator is of the following form:

$$ {}_L\tilde{\mathcal{D}}_h^{\beta} v(x) = \frac{\sum_{i=0}^{+\infty} z_i^{(\beta)} v(x-(i-1)\hat{h})}{\hat{h}^{\beta}}, $$

$$ {}_R\tilde{\mathcal{D}}_h^{\beta} v(x) = \frac{\sum_{i=0}^{+\infty} z_i^{(\beta)} v(x+(i-1)\hat{h})}{\hat{h}^{\beta}}, $$

where

$$ \begin{cases} z_0^{(\beta)} = \tilde{\lambda}_1 d_0^{(\beta)}, \ z_1^{(\beta)} = \tilde{\lambda}_1 d_1^{(\beta)} + \tilde{\lambda}_0 d_0^{(\beta)}, \\ z_i^{(\beta)} = \tilde{\lambda}_1 d_i^{(\beta)} + \tilde{\lambda}_0 d_{i-1}^{(\beta)} + \tilde{\lambda}_{-1} d_{i-2}^{(\beta)}, \ i \geqslant 2, \end{cases} \tag{4} $$

and

$$ \tilde{\lambda}_1 = \frac{\beta^2}{12} + \frac{\beta}{4} + \frac{1}{6}, \ \tilde{\lambda}_0 = \frac{2}{3} - \frac{\beta^2}{6}, \ \tilde{\lambda}_{-1} = \frac{\beta^2}{12} - \frac{\beta}{4} + \frac{1}{6}. $$

Let the operator $\mathcal{B}$ be

$$ \mathcal{B}v(x) = c^{\beta} v(x-\hat{h}) + (1-2c^{\beta}) v(x) + c^{\beta} v(x+\hat{h}), $$

where $c^{\beta} = -\frac{\beta^2}{24} + \frac{\beta}{24} + \frac{1}{6}$. Therefore, the fourth-order approximation to the operator $(-\Delta)^{\frac{\beta}{2}}$ can be obtained by

$$ \Delta_{\hat{h}}^{\beta} v(x) = \frac{{}_L\tilde{\mathcal{D}}_{\hat{h}}^{\beta} v(x) + {}_R\tilde{\mathcal{D}}_{\hat{h}}^{\beta} v(x)}{2\cos(\frac{\beta\pi}{2})} \tag{5} $$

$$ = \mathcal{B}(-\Delta)^{\frac{\beta}{2}} v(x) + \mathcal{O}(\hat{h}^4). \tag{6} $$

In the following, we will give the numerical discretization of (1)–(2) in the domain $\Pi = [a_1, b_1]$. Let $\hat{\tau} = \frac{T_1}{N_1}$ and denote $t_i = i\hat{\tau}$, where N_1 is a positive integer, $0 \leqslant i \leqslant N_1$. Given a grid function $u = \{u^i | 0 \leqslant i \leqslant N_1\}$, denote

$$ \tilde{D}_t u^{i+1} = \frac{3u^{i+1} - 4u^i + u^{i-1}}{2\hat{\tau}}, $$

$$ \tilde{u}^{i+1} = 2u^i - u^{i-1}. $$

Let $\hat{h} = \frac{b_1 - a_1}{M_1}$ and $x_i = a_1 + i\hat{h}$, where M_1 is a positive integer, $0 \leqslant i \leqslant M_1$. Moreover, we denote $\mathfrak{F}_{M_1} = \{i | i = 1, 2, \ldots, M_1 - 1\}$. According to the method of [39], we can obtain the following finite difference scheme for the fractional complex Eq. (1) and (2):

$$ \mathcal{B}\tilde{D}_t v_j^{i+1} + (\nu_1 + \mathbf{i}\eta_1)\Delta_{\hat{h}}^{\beta} v_j^{i+1} + (\kappa_1 + \mathbf{i}\zeta_1)\mathcal{B}|\tilde{v}_j^{i+1}|^2 \tilde{v}_j^{i+1} - \gamma_1 \mathcal{B}\tilde{v}_j^{i+1} = 0, $$

$$ j \in \mathfrak{F}_{M_1}, \ 1 \leqslant i \leqslant N_1 - 1, \tag{7} $$

$$ v_j^0 = v_0(x_j), j \in \mathbb{Z}, \tag{8} $$

$$ v_j^i = 0, j \in \mathbb{Z} \setminus \mathfrak{F}_{M_1}, \ 0 \leqslant i \leqslant N_1. \tag{9} $$

In the practical computation, we can calculate u^1 as follows [39]

$$\begin{cases} \mathcal{B}(\frac{v_j^1 - v_{0j}}{\hat{\tau}}) + (\nu_1 + \mathbf{i}\eta_1)\Delta_{\hat{h}}^{\beta} \frac{v_j^1 + v_{0j}}{2} + (\kappa_1 + \mathbf{i}\zeta_1)\mathcal{B}|v_j^{(1)}|^2 v_j^{(1)} = \gamma_1 \mathcal{B} v_j^{(1)}, \\ \mathcal{B}(\frac{v_j^{(1)} - v_{0j}}{\hat{\tau}/2}) + (\nu_1 + \mathbf{i}\eta_1)\Delta_{\hat{h}}^{\beta} v_{0j} + (\kappa_1 + \mathbf{i}\zeta_1)\mathcal{B}|v_{0j}|^2 v_{0j} = \gamma_1 \mathcal{B} v_{0j}, j \in \mathfrak{F}_{M_1}. \end{cases} \tag{10}$$

Let

$$v^{i+1} = [v_1^{i+1}, \ldots, v_{M_1-1}^{i+1}]^T,$$

$$D_1 = \hat{\tau}(\kappa_1 + \mathbf{i}\zeta_1) \begin{bmatrix} |v_{01}|^2 & 0 & \ldots & 0 \\ 0 & |v_{02}|^2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & |v_{0,M_1-1}|^2 \end{bmatrix} - (2 + \gamma_1 \hat{\tau}) I,$$

$$D_2 = (\kappa_1 + \mathbf{i}\zeta_1) \begin{bmatrix} |v_1^{(1)}|^2 & 0 & \ldots & 0 \\ 0 & |v_2^{(1)}|^2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & |v_{M_1-1}^{(1)}|^2 \end{bmatrix} - \gamma_1 I,$$

$$D_3 = (\kappa_1 + \mathbf{i}\zeta_1) \begin{bmatrix} |\tilde{v}_1^{i+1}|^2 & 0 & \ldots & 0 \\ 0 & |\tilde{v}_2^{i+1}|^2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & |\tilde{v}_{M_1-1}^{i+1}|^2 \end{bmatrix} - \gamma_1 I,$$

$$Z = \begin{bmatrix} z_1^{(\beta)} & z_0^{(\beta)} & 0 & \cdots & 0 & 0 \\ z_2^{(\beta)} & z_1^{(\beta)} & z_0^{(\beta)} & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ z_{M_1-2}^{(\beta)} & \ddots & \ddots & \ddots & z_1^{(\beta)} & z_0^{(\beta)} \\ z_{M_1-1}^{(\beta)} & z_{M_1-2}^{(\beta)} & \cdots & \cdots & z_2^{(\beta)} & z_1^{(\beta)} \end{bmatrix}, \tag{11}$$

and

$$A_\beta = \text{tridiag}\left(c^\beta, 1 - 2^\beta, c^\beta\right),$$

then the fourth-order finite difference scheme (7)–(10) has the following form

$$2A_\beta v^{(1)} = -A_\beta D_1 v^0 - \omega C v^0, \tag{12}$$

$$(A_\beta + \frac{\omega_1}{2} C) v^1 = (A_\beta - \frac{\omega_1}{2} C) v^0 - \hat{\tau} A_\beta D_2 v^{(1)}, \tag{13}$$

$$(\frac{3}{2} A_\beta + \omega_1 C) v^{i+1} = A_\beta (2v^i - \frac{1}{2} v^{i-1} - \hat{\tau} D_3 \tilde{v}^{i+1}), 1 \leqslant i \leqslant N_1 - 1. \tag{14}$$

where $C = Z + Z^T$ and $\omega_1 = \frac{(\nu_1 + \mathbf{i}\eta_1)\hat{\tau}}{2\hat{h}^\beta \cos(\frac{\beta\pi}{2})}$.

3 A Fast Preconditioned Numerical Method

In this part, we give an effective preconditioned generalized minimum residual (PGMRES) method [34] to solve the discreted linear system of the finite difference scheme (7)–(10), in which the preconditioned matrix is Strang's circulant preconditioner proposed in [2].

3.1 Toeplitz Matrix and GMRES Method

The Toeplitz linear system is as follows

$$B_{n_1} u = \tilde{b},$$

where B_{n_1} is a Toeplitz matrix, $\tilde{b}$ is a given vector. Toeplitz systems are widely used in various fields; see [1,3,7,11,25,26,31]. The elements of an $n_1 \times n_1$ Toeplitz matrix B_{n_1} satisfy $(B_{n_1})_{ij} = b_{i-j}$ for $i, j = 1, 2, \ldots, n_1$. The elements of a circulant matrix C_{n_1} satisfy $c_{-i} = c_{n_1 - i}$ for $1 \leqslant i \leqslant n_1 - 1$ [2].

It is well-known that [8] the computation cost will be $\mathcal{O}(n_1 \log n_1)$ operations if one wants to compute the matrix-vector products $C_{n_1} u$ and $C_{n_1}^{-1} u$ by the fast Fourier transform. In addition, we can calculate the matrix-vector product $B_{n_1} u$ in $\mathcal{O}(2n_1 \log(2n_1))$ by the FFT [2]. These important properties can be exploited to fast solve the discreted linear system in the form (12)–(14).

Consider the following non-Hermitian linear systems

$$Bu = \tilde{b},$$

where B is a non-Hermitian matrix. As we know, the GMRES method [34] is a very effective iterative method for solving these linear systems. Under normal circumstances, the convergent rate of the this method is very slow because of the very large condition number of the matrix B. To deal with this drawback, we could exploit the preconditioned matrix to speed up the convergent rate of the GMRES method. Please refer to [34] for the PGMRES method.

3.2 A Preconditioner for the Implicit-Explicit Difference Scheme

It can be seen that A_β and C are Toeplitz matrices in the matrix-vector form (12)–(14). According to Sect. 3.1, we can store an $M_1 \times M_1$ Toeplitz matrix B_{M_1} in $\mathcal{O}(M_1)$ of memory, and we can compute the matrix-vector product $B_{M_1} u$ in $\mathcal{O}(M_1 \log M_1)$ by the FFT. Moreover, the coefficient matrices of the complex linear systems (13) and (14) are non-Hermitian.

In this section, we exploit Strang's circulant matrix as a preconditioner to speed up the GMRES method. For the matrix A_β in (12), the preconditioned matrix is

$$S_1 = s(A_\beta),$$

where $s(A_\beta)$ is the Strang circulant matrix for the matrix A_β. For the matrix $A_\beta + \frac{\omega_1}{2} C$ in (13), the preconditioned matrix is

$$S_2 = s(A_\beta) + \frac{\omega_1}{2} s(C).$$

For the matrix $\frac{3}{2}A_\beta + \omega_1 C$ in (14), the preconditioned matrix is

$$S_3 = \frac{3}{2}s(A_\beta) + \omega_1 s(C).$$

It easily knows that S_1, S_2 and S_3 are circulant matrices. In the following, we will see that the proposed preconditioners are very efficient to speed up the GMRES method.

4 Numerical Experiments

In this part, we show the computational advantage of the PGMRES algorithm by two numerical examples for the fractional complex equation. We denote "GaE" by the direct method, which is implemented by left divide in MATLAB. For the PGMRES method with Strang's circulant preconditioner, we denote by "cPGMRES". We stop the cPGMRES method if the condition satisfies

$$\frac{\|\text{res1}^k\|_2}{\|\text{res1}^0\|_2} < 10^{-7},$$

where res1^k denotes the k-th residual vector for the cPGMRES method. In all tables, "Icpu" denotes the computational time in seconds for GaE and cPGMRES, and "Ite" is the iteration numbers for cPGMRES.

Example 1. In this example, the parameters in the fractional complex Eq. (1) and (2) are same as these in [39].

Furthermore, according to [39], the numerical exact solution v is calculated with $\hat{\tau} = 10^{-4}$ and $\hat{h} = 1.25 \times 10^{-2}$. Let $v_{\hat{h}}$ be the numerical solution. We compute the error $\text{ERR} = v - v_{\hat{h}}$ as the numerical accuracy at $T_1 = 2$ with the $l^\infty_{\hat{h}}$ norm.

We report the numerical results in Table 1. In this table, ERR_1 and ERR_2 denote the errors for the cPGMRES method and the GaE method, respectively. We can see that there is little difference between numerical errors of the two methods. But, if the size of the matrix in the complex linear systems (12)–(13) is large, the computational times of GaE are much more than the computational times of cPGMRES. Furthermore, Figs. 1, 2 and 3 show the distribution of the eigenvalues for the matrix $\frac{3}{2}A_\beta + \omega_1 C$ and $S_3^{-1}(\frac{3}{2}A_\beta + \omega_1 C)$ at $T_1 = 2$, respectively, when the size of the matrix is 320, and $\beta = 1.3,\ 1.6,\ 1.9$. In the figures, the blue points indicate that most of the eigenvalues of the matrix $S_3^{-1}(\frac{3}{2}A_\beta + \omega_1 C)$ approach to 1, while the eigenvalues of the matrix $\frac{3}{2}A_\beta + \omega_1 C$ do not approach to 1. Therefore, the figures show that our new preconditioner is very effective for solving the linear systems (12)–(14).

Table 1. Numerical results for Example 1

β	$(\hat{\tau}, \hat{h})$	*cPGMRES*			*GaE*	
		ERR_1	Ite	Icpu	ERR_2	Icpu
1.3	$(2^{-4}, 0.4)$	3.7500e − 2	3.1	0.0470	3.7500e − 2	0.0160
	$(2^{-6}, 0.2)$	2.4074e − 3	3.0	0.1090	2.4074e − 3	0.1870
	$(2^{-8}, 0.1)$	1.6404e − 4	2.9	0.4070	1.6405e − 4	3.1100
1.6	$(2^{-4}, 0.4)$	2.1789e − 2	3.0	0.0320	2.1789e − 2	0.0150
	$(2^{-6}, 0.2)$	1.3764e − 3	2.8	0.0780	1.3764e − 3	0.1570
	$(2^{-8}, 0.1)$	8.8752e − 5	2.0	0.3120	8.8693e − 5	3.0000
1.9	$(2^{-4}, 0.4)$	1.4653e − 2	2.1	0.0160	1.4653e − 2	0.0150
	$(2^{-6}, 0.2)$	9.1163e − 4	2.0	0.0620	9.1163e − 4	0.1570
	$(2^{-8}, 0.1)$	5.7246e − 5	2.0	0.2970	5.7246e − 5	3.1250

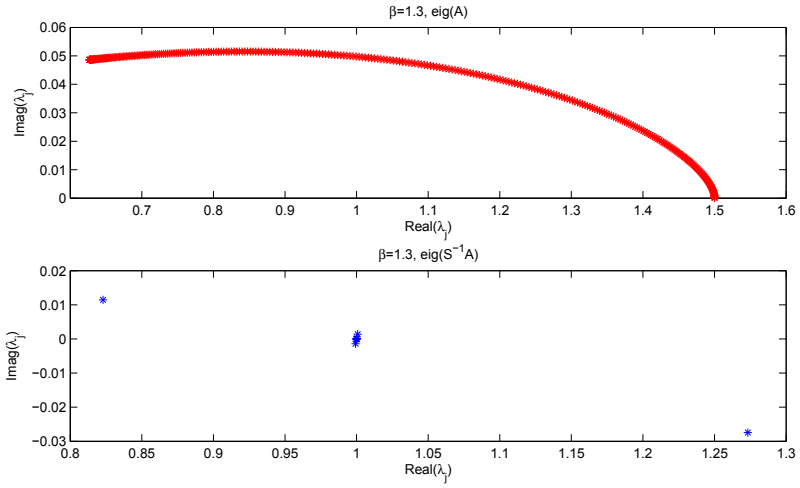

Fig. 1. Example 1: Spectrum of $\frac{3}{2}A_\beta + \omega_1 C$ (upper) and $S_3^{-1}(\frac{3}{2}A_\beta + \omega_1 C)$ (lower), when $\beta = 1.3$.

Example 2. In this example, we take the parameters which are same as these in [39]. Moreover, we compute the exact solution v with $\hat{\tau} = 10^{-4}$ and $\hat{h} = 1.25 \times 10^{-2}$.

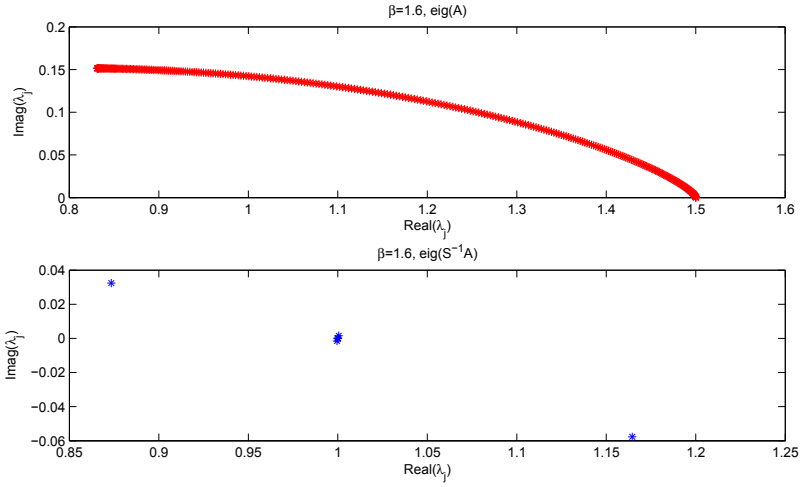

Fig. 2. Example 1: Spectrum of $\frac{3}{2}A_\beta + \omega_1 C$ (upper) and $S_3^{-1}(\frac{3}{2}A_\beta + \omega_1 C)$ (lower), when $\beta = 1.6$.

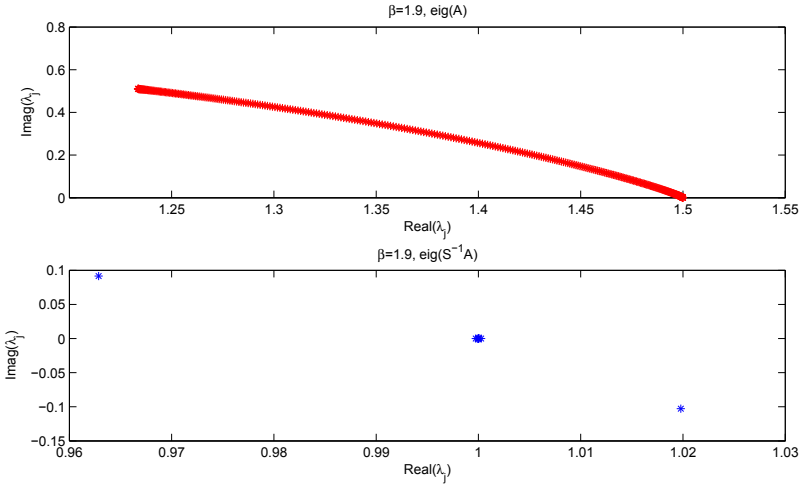

Fig. 3. Example 1: Spectrum of $\frac{3}{2}A_\beta + \omega_1 C$ (upper) and $S_3^{-1}(\frac{3}{2}A_\beta + \omega_1 C)$ (lower), when $\beta = 1.9$.

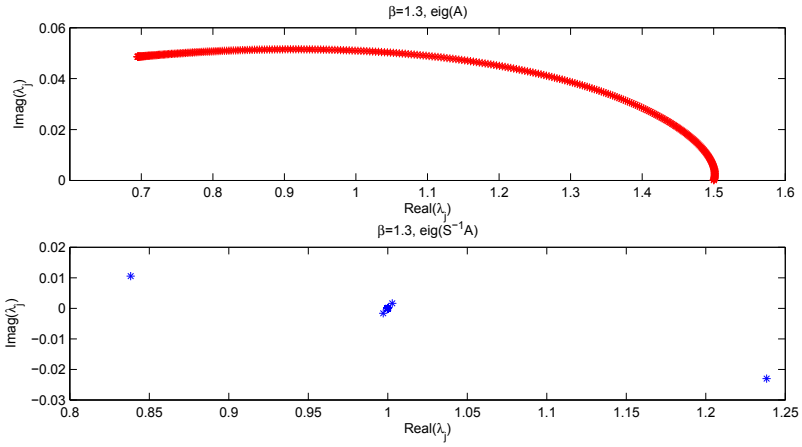

Fig. 4. Example 2: Spectrum of $\frac{3}{2}A_\beta + \omega_1 C$ (upper) and $S_3^{-1}(\frac{3}{2}A_\beta + \omega_1 C)$ (lower), when $\beta = 1.3$

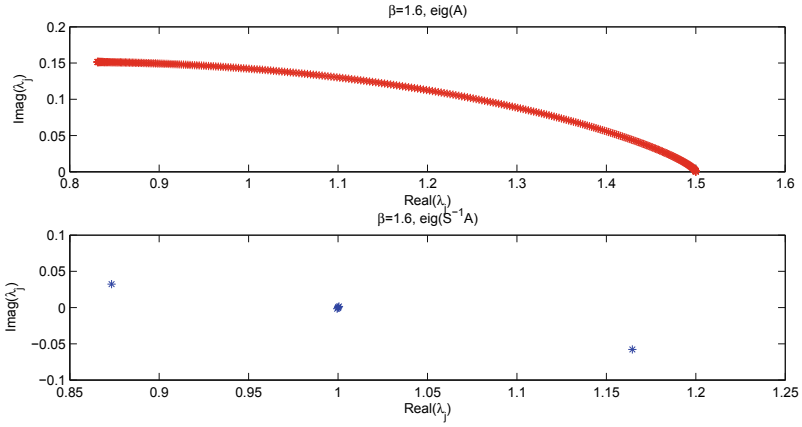

Fig. 5. Example 2: Spectrum of $\frac{3}{2}A_\beta + \omega_1 C$ (upper) and $S_3^{-1}(\frac{3}{2}A_\beta + \omega_1 C)$ (lower), when $\beta = 1.6$.

Table 2 gives the numerical results and Figs. 4, 5 and 6 show the distribution of the eigenvalues for the matrices $\frac{3}{2}A_\beta + \omega_1 C$ and $S_3^{-1}(\frac{3}{2}A_\beta + \omega_1 C)$ at $T_1 = 2$, respectively, when the size of the matrix is 320, and $\beta = 1.3,\ 1.6,\ 1.9$. Similar to Example 1, the computational results and figures indicate the superiority of the preconditioned numerical method.

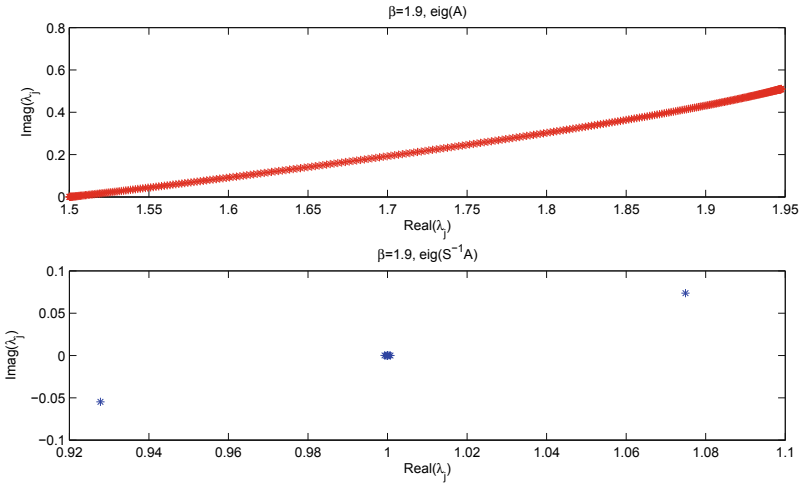

Fig. 6. Example 2: Spectrum of $\frac{3}{2}A_\beta + \omega_1 C$ (upper) and $S_3^{-1}(\frac{3}{2}A_\beta + \omega_1 C)$ (lower), when $\beta = 1.9$.

Table 2. Numerical results for Example 2

β	$(\hat{\tau}, \hat{h})$	*cPGMRES*			*GaE*	
		ERR_1	Ite	Icpu	ERR_2	Icpu
1.3	$(2^{-4}, 0.4)$	1.6529e − 4	3.9	0.0250	1.6529e − 4	0.0160
	$(2^{-6}, 0.2)$	1.1880e − 5	3.7	0.1160	1.1879e − 5	0.1870
	$(2^{-8}, 0.1)$	7.5537e − 7	3.0	0.4440	7.5333e − 7	3.2500
1.6	$(2^{-4}, 0.4)$	2.1495e − 4	3.4	0.0250	2.1495e − 4	0.0160
	$(2^{-6}, 0.2)$	1.6276e − 5	2.8	0.1010	1.6276e − 5	0.1400
	$(2^{-8}, 0.1)$	1.0397e − 6	2.6	0.3470	1.0396e − 6	3.0790
1.9	$(2^{-4}, 0.4)$	3.3256e − 4	2.9	0.0250	3.3256e − 4	0.0160
	$(2^{-6}, 0.2)$	2.5996e − 5	2.6	0.0960	2.5996e − 5	0.1560
	$(2^{-8}, 0.1)$	1.6857e − 6	2.2	0.3000	1.6857e − 6	2.9380

5 Conclusion and Future Work

In this work, we have given a fast preconditioned numerical method to solve the linear system, which is discreted from the space fractional complex Ginzburg-Landau equation. We propose a circulant preconditioner due to the Toeplitz structure of the coefficient matrix of the linear system. Numerical examples show that the preconditioned numerical method is very efficient.

References

1. Bunch, J.R.: Stability of methods for solving Toeplitz systems of equations. SIAM J. Sci. Stat. Comput. **6**, 349–364 (1985)
2. Chan, R., Jin, X.: An Introduction to Iterative Toeplitz Solvers. SIAM, Philadelphia (2007)
3. Chan, R., Ng, M.: Conjugate gradient methods for Toeplitz systems. SIAM Rev. **38**, 427–482 (1996)
4. Chen, L., Jiang, D., Bao, R., Xiong, J., Liu, F., Bei, L.: MIMO scheduling effectiveness analysis for bursty data service from view of QoE. Chin. J. Electron. **26**(5), 1079–1085 (2017)
5. Chen, L., et al.: A lightweight end-side user experience data collection system for quality evaluation of multimedia communications. IEEE Access **6**(1), 15408–15419 (2018)
6. Chen, L., Zhang, L.: Spectral efficiency analysis for massive MIMO system under QoS constraint: an effective capacity perspective. Mobile Netw. Appl. (2020). https://doi.org/10.1007/s11036-019-01414-4
7. Ching, W.-K.: Iterative Methods for Queuing and Manufacturing Systems. Springer, London (2001)
8. Davis, P.: Circulant Matrices, 2nd edn. AMS Chelsea, Providence, RI (1994)
9. Gorenflo, R., Mainardi, F.: Random walk models for space-fractional diffusion processes. Fractional Calc. Appl. Anal. **1**(2), 167–191 (1998)
10. Guo, B.-L., Huo, Z.-H.: Well-posedness for the nonlinear fractional Schrödinger equation and inviscid limit behavior of solution for the fractional Ginzburg-Landau equation. Fract. Calc. Appl. Anal. **16**(1), 226–242 (2012)
11. Hansen, P.C., Nagy, J.G., O'Leary, D.P.: Deblurring Images: Matrices, Spectra, and Filtering. SIAM, Philadelphia (2006)
12. Hao, Z.-P., Sun, Z.-Z., Cao, W.-R.: A fourth-order approximation of fractional derivatives with its applications. J. Comput. Phys. **281**, 787–805 (2015)
13. He, D., Pan, K.: An unconditionally stable linearized difference scheme for the fractional Ginzburg-Landau equation. Numer. Algor. **79**, 899–925 (2018)
14. Huo, L., Jiang, D., Lv, Z., et al.: An intelligent optimization-based traffic information acquirement approach to software-defined networking. Comput. Intell. **36**(1), 1–21 (2019)
15. Huo, L., Jiang, D., Qi, S., Song, H., Miao, L.: An AI-based adaptive cognitive modeling and measurement method of network traffic for EIS. Mob. Netw. Appl. 1–11 (2019). https://doi.org/10.1007/s11036-019-01419-z
16. Huo, L., Jiang, D., Zhu, X., et al.: An SDN-based fine-grained measurement and modeling approach to vehicular communication network traffic. Int. J. Commun. Syst. **2019**(9), 1–19 (2019)
17. Jiang, D., Huo, L., Song, H.: Rethinking behaviors and activities of base stations in mobile cellular networks based on big data analysis. IEEE Trans. Netw. Sci. Eng. **1**(1), 1–12 (2018)
18. Jiang, D., Huo, L., Lv, Z., et al.: A joint multi-criteria utility-based network selection approach for vehicle-to-infrastructure networking. IEEE Trans. Intell. Transp. Syst. **19**(10), 3305–3319 (2018)
19. Jiang, D., Wang, Y., Lv, Z., et al.: Big data analysis-based network behavior insight of cellular networks for industry 4.0 applications. IEEE Trans. Ind. Inf. **16**(2), 1310–1320 (2020)

20. Jiang, D., Zhang, P., Lv, Z., et al.: Energy-efficient multi-constraint routing algorithm with load balancing for smart city applications. IEEE Int. Things J. **3**(6), 1437–1447 (2016)
21. Jiang, D., Li, W., Lv, H.: An energy-efficient cooperative multicast routing in multi-hop wireless networks for smart medical applications. Neurocomputing **220**, 160–169 (2017)
22. Jiang, D., Wang, Y., Lv, Z., et al.: Intelligent optimization-based reliable energy-efficient networking in cloud services for IIoT networks. IEEE J. Sel. Areas Commun. **38**(5), 928–941 (2019)
23. Jiang, D., Wang, W., Shi, L., et al.: A compressive sensing-based approach to end-to-end network traffic reconstruction. IEEE Trans. Netw. Sci. Eng. **5**(3), 1–12 (2018)
24. Jiang, D., Huo, L., Li, Y.: Fine-granularity inference and estimations to network traffic for SDN. Plos One **13**(5), 1–23 (2018)
25. Jin, X.-Q.: Developments and Applications of Block Toeplitz Iterative Solvers. The Netherlands, and Science Press, Beijing, China, Kluwer Academic Publishers, Dordrecht (2002)
26. Kailath, T., Sayed, A.H. (eds.): Fast Reliable Algorithms for Matrices with Structure. SIAM, Philadelphia (1999)
27. Qi, S., Jiang, D., Huo, L.: A prediction approach to end-to-end traffic in space information networks. Mob. Netw. Appl. 1–10 (2019). https://doi.org/10.1007/s11036-019-01424-2
28. Meerschaert, M.-M., Tadjeran, C.: Finite difference approximations for fractional advection-dispersion flow equations. J. Comput. Appl. Math. **172**, 65–77 (2004)
29. Milovanov, A., Rasmussen, J.: Fractional generalization of the Ginzburg-Landau equation: an unconventional approach to critical phenomena in complex media. Phys. Lett. A **337**, 75–80 (2005)
30. Mvogo, A., Tambue, A., Ben-Bolie, G., Kofane, T.: Localized numerical impulse solutions in diffuse neural networks modeled by the complex fractional Ginzburg-Landau equation. Commun. Nonlinear Sci. **39**, 396–410 (2016)
31. Ng, M.K.: Iterative Methods for Toeplitz Systems. Oxford University Press, Oxford (2004)
32. Podlubny, I.: Fractional Differential Equations. Academic Press, New York (1999)
33. Pu, X., Guo, B.: Well-posedness and dynamics for the fractional Ginzburg-Landau equation. Appl. Anal. **92**, 318–334 (2013)
34. Saad, Y.: Iterative Methods for Sparse Linear Systems. SIAM, Philadelphia (2003)
35. Tarasov, V., Zaslavsky, G.: Fractional Ginzburg-Landau equation for fractal media. Physica A **354**, 249–261 (2005)
36. Wang, F., Jiang, D., Qi, S., Qiao, C., Shi, L.: A dynamic resource scheduling scheme in edge computing satellite networks. Mob. Netw. Appl. 1–12 (2020). https://doi.org/10.1007/s11036-019-01421-5
37. Wang, F., Jiang, D., Qi, S.: An adaptive routing algorithm for integrated information networks. China Commun. **7**(1), 196–207 (2019)
38. Wang, P., Huang, C.: An implicit midpoint difference scheme for the fractional Ginzburg-Landau equation. J. Comput. Phys. **312**, 31–49 (2016)
39. Wang, P., Huang, C.: An efficient fourth-order in space difference scheme for the nonlinear fractional Ginzburg-Landau equation. BIT **58**, 783–805 (2018)
40. Wang, Y., Jiang, D., Huo, L., Zhao, Y.: A new traffic prediction algorithm to software defined networking. Mob. Netw. Appl. 1–10 (2019). https://doi.org/10.1007/s11036-019-01423-3
41. Zhang, K., Chen, L., An, Y., et al.: A QoE test system for vehicular voice cloud services. Mob. Netw. Appl. (2019). https://doi.org/10.1007/s11036-019-01415-3