Journal of Scientific Computing (2021) 87:30
https://doi.org/10.1007/s10915-021-01431-0

A Dimensional Splitting Exponential Time Differencing Scheme for Multidimensional Fractional Allen-Cahn Equations

Hao Chen[1] · **Hai-Wei Sun**[2]

Received: 30 August 2020 / Revised: 15 January 2021 / Accepted: 8 February 2021

Abstract
This paper is concerned with numerical methods for solving the multidimensional Allen-Cahn equations with spatial fractional Riesz derivatives. A fully discrete numerical scheme is proposed using a dimensional splitting exponential time differencing approximation for the time integration with finite difference discretization in space. Theoretically, we prove that the proposed numerical scheme can unconditionally preserve the discrete maximum principle. The error estimate in maximum-norm of the proposed scheme is also established in the fully discrete sense. In practical computation, the proposed algorithm can be carried out by computing linear systems and the matrix exponential associated with only one dimensional discretized matrices that possess Toeplitz structure. Meanwhile, fast methods for inverting the Toeplitz matrix and computing the Toeplitz exponential multiplying a vector are exploited to reduce the complexity. Numerical examples in two and three spatial dimensions are given to illustrate the effectiveness and efficiency of the proposed scheme.

Keywords Fractional Allen-Cahn equation · Discrete maximum principle · Exponential time differencing · Dimensional splitting · Matrix exponential · Toeplitz matrix

Mathematics Subject Classification 65F10 · 65L05 · 65N22 · 65F15

1 Introduction

In this paper we consider the numerical solution of the following space fractional Allen-Cahn (FAC) equation [24]

$$u_t = \varepsilon^2 \mathcal{L}_{\mathbf{x}}^{\alpha} u + u - u^3, \quad \mathbf{x} \in \Omega, \ t \in (0, T],$$

✉ Hai-Wei Sun
HSun@umac.mo

Hao Chen
hch@cqnu.edu.cn

[1] College of Mathematics Science, Chongqing Normal University, Chongqing, China

[2] Department of Mathematics, University of Macau, Macao, China

Published online: 06 March 2021

$$\begin{aligned} u(\mathbf{x}, 0) &= u_0(\mathbf{x}), && \mathbf{x} \in \overline{\Omega}, \\ u(\mathbf{x}, t) &= 0, && \mathbf{x} \text{ on } \partial\Omega,\ t \in [0, T], \end{aligned} \tag{1.1}$$

where $u(\mathbf{x}, t)$ denotes the unknown function, $\Omega = (a, b)^d$ is the spatial domain in $\mathbb{R}^d$ ($d = 2, 3$), $\varepsilon > 0$ is an given constant, and $\mathcal{L}_{\mathbf{x}}^{\alpha}$ ($1 < \alpha < 2$) denotes the Riesz fractional operator in space. In the one dimensional (1D) case, we denote

$$\mathcal{L}_{\mathbf{x}}^{\alpha} u = \mathcal{L}_{x}^{\alpha} u := \frac{1}{-2\cos\frac{\alpha\pi}{2}} \left({}_a D_x^{\alpha} u + {}_x D_b^{\alpha} u \right),$$

where the left-sided and right-sided fractional derivative are respectively defined in Riemann-Liouville form as follows [34]

$$\begin{aligned} {}_a D_x^{\alpha} u &= \frac{1}{\Gamma(2-\alpha)} \frac{\partial^2}{\partial x^2} \int_a^x \frac{u(\xi, t)}{(x-\xi)^{\alpha-1}} d\xi, \\ {}_x D_b^{\alpha} u &= \frac{1}{\Gamma(2-\alpha)} \frac{\partial^2}{\partial x^2} \int_x^b \frac{u(\xi, t)}{(\xi-x)^{\alpha-1}} d\xi. \end{aligned}$$

Here $\Gamma(\cdot)$ is the Gamma function. In the two dimensional (2D) case, we denote $\mathcal{L}_{\mathbf{x}}^{\alpha} u = (\mathcal{L}_x^{\alpha} + \mathcal{L}_y^{\alpha}) u$, and similarly in the three dimensional (3D) case, we define $\mathcal{L}_{\mathbf{x}}^{\alpha} u = (\mathcal{L}_x^{\alpha} + \mathcal{L}_y^{\alpha} + \mathcal{L}_z^{\alpha}) u$.

The FAC equation (1.1) can be viewed as a fractional analogue of the classic Allen-Cahn equation, which is a well-known phase field model to describe the motion of antiphase boundaries in crystalline solids [1]. The Allen-Cahn equation possesses two important properties: one is the energy decay property and the other is the maximum principle, which means that the absolute value of the entire solution is bounded by 1 if the absolute values of initial and boundary data are bounded by 1. In the last few decades, there have been some studies on energy stable numerical methods for the classic Allen-Cahn equation; see [13–15,37,40,41]. Latterly, the maximum principle preserving numerical schemes are also studied [29,36,38].

There has been growing interest in studying the FAC equation (1.1) and the nonlocal Allen-Cahn equation in recent years. For example, implicit finite element method [4] and Fourier spectral method [3] were studied for solving the FAC equation. A semi-implicit linear scheme in time and Fourier spectral method in space [10] was discussed for the nonlocal Allen-Cahn equation. Nevertheless, the discrete maximum principle was not discussed for those methods. In 2017, Hou, Tang, and Yang proposed a temporal second order accurate method for solving the FAC equation (1.1) by the Crank-Nicolson method [24]. Their method can preserve the maximum principle in the discrete sense under a time step constraint, and this is the first work on the second order schemes preserving the maximum principle. However, their method included a nonlinear scheme which is computationally expensive, especially for the 3D problems. To reduce the complexity, He, Pan, and Hu combined the Strang splitting method and alternating direction implicit technique to proposed a fast algorithm [18] for solving the multidimensional FAC equation (1.1). The computational cost of their method is much cheaper than that of the method in [24]. Nevertheless, their method can only conditionally preserve the discrete maximum principle, which is under a time step constraint. Exponential time differencing (ETD) methods, also referred to as exponential integrators, are a well-known class of time-stepping methods for solving stiff partial differential equations, see [7,22]. In 2019, the temporal second order ETD Runge-Kutta (ETDRK2) scheme was proposed [11] for solving nonlocal Allen-Cahn equations with *periodic boundary condition*, where the proposed ETDRK2 scheme preserves the discrete maximum principle unconditionally. This is the first work to analyze the discrete maximum principle of the fully discrete ETD schemes. For problems with non-periodic boundary condition, however, the implementation

of ETDRK2 scheme requires to compute the products of matrix exponential-like functions with vectors, which could be very expensive for multidimensional problems; see Table 2 in Section 5.

In this paper, we propose a dimensional splitting ETDRK2 (DSETDRK2) method for solving the 2D and 3D FAC equations (1.1) with *non-periodic boundary conditions*. Our method is quite different from that proposed in [11] as we exploit the dimensional splitting technique to decompose the multi-dimensional problems into several 1D problems. More precisely, the proposed DSETDRK2 scheme only requires to compute matrix-vector products and matrix exponential associated with 1D discretized matrix, which is much cheaper than other existing methods. Theoretically, we show that the proposed scheme preserves the discrete maximum principle unconditionally. Moreover, the error estimate in maximum-norm of the proposed method is also established in the fully discrete sense. In practical computations, noting that the discretized matrix possesses the Toeplitz structure, we employ the shift-invert Lanczos method to calculate the Toeplitz matrix exponential multiplying a vector; see [32]. Meanwhile, the fast Fourier transform and Gohberg-Semencul formula are exploited to reduce the computational cost of the inverse Toeplitz matrix-vector multiplication. Numerical results are given to demonstrate the efficiency of the proposed method.

The rest of the paper is organized as follows. In section 2, we introduce the DSETDRK2 scheme for the FAC equation with a second order finite difference approximation used for spatial discretization. In section 3, we show that the proposed scheme preserves the discrete maximum principle and establish the maximum-norm error estimate. In section 4, efficient implementation issues are discussed. Numerical results are presented in section 5 and some concluding remarks are given in section 6.

2 Numerical Method

The goal of this section is to introduce the numerical method studied in this article, where a second order finite difference method in [39] is adopted for the spatial discretization of the Riesz fractional operator and an ETD-type integrator is constructed for the time integration of the semidiscrete FAC problem.

2.1 Space Discretization

We first introduce a spatial semidiscretization of the 2D FAC equation (1.1) through a second order finite difference method. Let us fix a positive integer M and define the following partition of Ω, i.e.,

$$x_i = y_i = a + ih, \quad h = \frac{b-a}{M+1}, \quad i = 0, 1, \ldots, M+1.$$

By the method of lines approach, we first discretize the fractional derivative in space by the following finite difference scheme [39]:

$$ {}_aD_x^\alpha u(x_i, y_j, t) = \frac{1}{h^\alpha} \sum_{k=0}^{i+1} \omega_k^{(\alpha)} u(x_{i-k+1}, y_j, t) + \mathcal{O}(h^2),$$

$$ {}_xD_b^\alpha u(x_i, y_j, t) = \frac{1}{h^\alpha} \sum_{k=0}^{M-i+2} \omega_k^{(\alpha)} u(x_{i+k-1}, y_j, t) + \mathcal{O}(h^2),$$

$$
\begin{aligned}
{}_aD_y^{\alpha}u(x_i,y_j,t) &= \frac{1}{h^{\alpha}}\sum_{l=0}^{j+1}\omega_l^{(\alpha)}u(x_i,y_{j-l+1},t)+\mathcal{O}(h^2),\\
{}_yD_b^{\alpha}u(x_i,y_j,t) &= \frac{1}{h^{\alpha}}\sum_{l=0}^{M-j+2}\omega_l^{(\alpha)}u(x_i,y_{j+l-1},t)+\mathcal{O}(h^2),
\end{aligned}
$$

and obtain

$$
\begin{aligned}
\frac{du_{i,j}(t)}{dt} &= \frac{\varepsilon^2}{-2h^{\alpha}\cos\frac{\alpha\pi}{2}}\left(\sum_{k=0}^{i+1}\omega_k^{(\alpha)}u_{i-k+1,j}(t)+\sum_{k=0}^{M-i+2}\omega_k^{(\alpha)}u_{i+k-1,j}(t)\right)\\
&\quad+\frac{\varepsilon^2}{-2h^{\alpha}\cos\frac{\alpha\pi}{2}}\left(\sum_{l=0}^{j+1}\omega_l^{(\alpha)}u_{i,j-l+1}(t)+\sum_{l=0}^{M-j+2}\omega_l^{(\beta)}u_{i,j+l-1}(t)\right),\\
&\quad+u_{i,j}(t)-u_{i,j}^3(t), \qquad 1\le i,j\le M,
\end{aligned}
\tag{2.1}
$$

where $u_{i,j}(t)$ is an approximation solution to $u(x_i,y_j,t)$ and the coefficients $\omega_k^{(\alpha)}$ are given by

$$
\omega_0^{(\alpha)}=\frac{\alpha}{2}g_0^{(\alpha)},\quad \omega_k^{(\alpha)}=\frac{\alpha}{2}g_k^{(\alpha)}+\frac{2-\alpha}{2}g_{k-1}^{(\alpha)},\quad k\ge 1,
\tag{2.2}
$$

with

$$
g_0^{(\alpha)}=1,\quad g_k^{(\alpha)}=\left(1-\frac{\alpha+1}{k}\right)g_{k-1}^{(\alpha)},\quad k\ge 1.
\tag{2.3}
$$

Denote the M^2-dimensional vectors

$$
\begin{aligned}
U(t) &= [u_{1,1}(t),\ldots,u_{M,1}(t),u_{1,2}(t),\ldots,u_{M,2}(t),\ldots,u_{1,M}(t),\ldots,u_{M,M}(t)]^T,\\
U(t)^{\cdot 3} &= [u_{1,1}^3(t),\ldots,u_{M,1}^3(t),u_{1,2}^3(t),\ldots,u_{M,2}^3(t),\ldots,u_{1,M}^3(t),\ldots,u_{M,M}^3(t)]^T,
\end{aligned}
$$

and the $M\times M$ Toeplitz matrices

$$
\tilde{B}=\begin{bmatrix}
\omega_1^{(\alpha)} & \omega_0^{(\alpha)} & 0 & \cdots & 0 & 0\\
\omega_2^{(\alpha)} & \omega_1^{(\alpha)} & \omega_0^{(\alpha)} & 0 & \cdots & 0\\
\vdots & \ddots & \ddots & \ddots & \ddots & \vdots\\
\vdots & \ddots & \ddots & \ddots & \ddots & 0\\
\omega_{M-1}^{(\alpha)} & \ddots & \ddots & \ddots & \omega_1^{(\alpha)} & \omega_0^{(\alpha)}\\
\omega_M^{(\alpha)} & \omega_{M-1}^{(\alpha)} & \cdots & \cdots & \omega_2^{(\alpha)} & \omega_1^{(\alpha)}
\end{bmatrix}
\tag{2.4}
$$

and

$$
K=\frac{\varepsilon^2}{-2h^{\alpha}\cos\frac{\pi\alpha}{2}}(\tilde{B}+\tilde{B}^T).
\tag{2.5}
$$

Then, by incorporating the homogeneous Dirichlet boundary conditions $u_{0,j}(t)=u_{M+1,j}(t)=u_{i,0}(t)=u_{i,M+1}(t)=0$ $(i,j=0,1,\ldots,M+1)$, the semi-discretization scheme (2.1) can be written in matrix form as

$$
\frac{dU}{dt}=\tilde{A}U+U-U^{\cdot 3},\quad U(0)=U_0,
\tag{2.6}
$$

where

$$\tilde{A} = I_M \otimes K + K \otimes I_M. \tag{2.7}$$

Here, I_M denotes the $M \times M$ identity matrix. Similarly, the semi-discretization system for the 3D FAC equation (1.1) can be also written in the form (2.6) with

$$\tilde{A} = I_M \otimes I_M \otimes K + I_M \otimes K \otimes I_M + K \otimes I_M \otimes I_M. \tag{2.8}$$

2.2 Time Discretization

For the sake of the stability of the time integration scheme constructed later, we introduce a stabilizing parameter $\kappa > 0$ and define

$$A := \tilde{A} - \kappa I_N, \quad f(U) := (\kappa + 1)U - U^{.3}, \tag{2.9}$$

where $N := M^d$. Then the ODE system (2.6) can be written as

$$\frac{dU}{dt} = AU + f(U), \quad U(0) = U_0. \tag{2.10}$$

By the variation-of-constant formula, the solution of (2.10) at time $t + \tau$ can be represented as

$$U(t+\tau) = e^{\tau A}U(t) + \int_0^\tau e^{(\tau-s)A} f(U(t+s))ds, \ \ \forall t \geq 0, \ \tau > 0. \tag{2.11}$$

For a positive integer N_t, we divide the time interval by $\{t_n = n\tau : n = 0, 1, \ldots, N_t\}$ with a uniform time step $\tau = T/N_t$. By setting $t = t_n$ in (2.11), we have

$$U(t_{n+1}) = e^{\tau A}U(t_n) + \int_0^\tau e^{(\tau-s)A} f(U(t_n+s))ds. \tag{2.12}$$

Various ETD schemes can be obtained from the approximations to the integral in (2.12). For example, by interpolating the integrand at the known value $f(U^n)$ only, we obtain the first order ETD scheme or exponential Euler method [7,22,30]

$$U^{n+1} = e^{\tau A}U^n + \int_0^\tau e^{(\tau-s)A} f(U^n)ds = e^{\tau A}U^n + \tau\varphi_1(\tau A) f(U^n),$$

where $\varphi_1(z) = \frac{e^z-1}{z}$. A second order ETDRK2 scheme is obtained by approximating $f(U(t_n + s))$ by a linear interpolation based on $f(U^n)$ and $f(\hat{U}^{n+1})$, where $\hat{U}^{n+1}$ is an approximation of $U(t_{n+1})$. The ETDRK2 scheme of (2.10) takes the following form [7,22,30]

$$\hat{U}^{n+1} = e^{\tau A}U^n + \int_0^\tau e^{(\tau-s)A} f(U^n)ds, \tag{2.13a}$$

$$U^{n+1} = e^{\tau A}U^n + \int_0^\tau e^{(\tau-s)A} \left[\left(1 - \frac{s}{\tau}\right) f(U^n) + \frac{s}{\tau} f(\hat{U}^{n+1})\right] ds. \tag{2.13b}$$

or, equivalently,

$$\hat{U}^{n+1} = e^{\tau A}U^n + \tau\varphi_1(\tau A) f(U^n), \tag{2.14a}$$

$$U^{n+1} = e^{\tau A}U^n + \tau\varphi_1(\tau A) f(U^n) + \tau\varphi_2(\tau A)[f(\hat{U}^{n+1}) - f(U^n)], \tag{2.14b}$$

where $\varphi_2(z) = \frac{e^z-1-z}{z^2}$. More sophisticated ETDRK methods can be derived by approximating the integral in (2.12) with high-order quadrature rules; see [22] and the references therein. We refer to [8,9] for discussions on the linear stabilities of some ETD and modified ETD schemes.

Note that the key ingredient in the effective implementation of an ETD scheme is the effective computation of matrix-vector products with φ-functions. For A arising in multidimensional problems with non-periodic boundary condition, exact evaluation of φ-functions and vector products is computationally expensive. To avoid expensive computational cost of the ETD schemes for multidimensional problems with non-periodic boundary condition, in this paper we introduce a new ETD-type method based on the idea of dimensional splitting.

For 2D problem, it is easy to show from (2.7) and (2.9) that A has the dimensional splitting

$$A = A_1 + A_2, \quad A_1 := I_M \otimes (K - \kappa I_M), \quad A_2 := K \otimes I_M. \tag{2.15}$$

Similarly, for 3D problem, we have the following splitting

$$A = A_1 + A_2, \quad A_1 := I_M \otimes I_M \otimes (K - \kappa I_M), \quad A_2 := I_M \otimes K \otimes I_M + K \otimes I_M \otimes I_M. \tag{2.16}$$

By the properties of the Kronecker product [19,23], we see that A_1 and A_2 commute, i.e., $A_1A_2 = A_2A_1$, and we have $e^{A_1+A_2} = e^{A_1}e^{A_2} = e^{A_2}e^{A_1}$.

Now we can write (2.12) into the following form

$$U(t_{n+1}) = e^{\tau A_1}e^{\tau A_2}U(t_n) + \int_0^\tau e^{(\tau-s)A_1}e^{(\tau-s)A_2}f(U(t_n+s))ds. \tag{2.17}$$

The second order DSETDRK2 comes from approximating $e^{(\tau-s)A_2}f(U(t_n+s))$ by a linear interpolation based on $e^{\tau A_2}f(U^n)$ and $f(\hat{U}^{n+1})$ and computing the resulting integral exactly, where $\hat{U}^{n+1}$ is an approximation of $U(t_{n+1})$. The DSETDRK2 scheme is as follows: for $n = 0, 1, \ldots, N_t$,

$$\hat{U}^{n+1} = e^{\tau A_1}e^{\tau A_2}U^n + \int_0^\tau e^{(\tau-s)A_1}e^{\tau A_2}f(U^n)ds, \tag{2.18a}$$

$$U^{n+1} = e^{\tau A_1}e^{\tau A_2}U^n + \int_0^\tau e^{(\tau-s)A_1}\left[\left(1-\frac{s}{\tau}\right)e^{\tau A_2}f(U^n) + \frac{s}{\tau}f(\hat{U}^{n+1})\right]ds, \tag{2.18b}$$

or, equivalently,

$$\hat{U}^{n+1} = e^{\tau A_1}e^{\tau A_2}U^n + \tau\varphi_1(\tau A_1)e^{\tau A_2}f(U^n), \tag{2.19a}$$

$$U^{n+1} = e^{\tau A_1}e^{\tau A_2}U^n + \tau\varphi_1(\tau A_1)e^{\tau A_2}f(U^n) + \tau\varphi_2(\tau A_1)\left[f(\hat{U}^{n+1}) - e^{\tau A_2}f(U^n)\right]. \tag{2.19b}$$

3 Numerical Analysis of the Scheme

This section is dedicated to demonstrating that the DSETDRK2 scheme (2.18) preserves the discrete maximum principle unconditionally. We also derive error estimate of the proposed scheme.

3.1 Discrete Maximum Principle

First, we present some lemmas that will be useful in the analysis of the proposed scheme.

Lemma 3.1 *Denote $B = \tilde{B} + \tilde{B}^T = (b_{ij}) \in \mathbb{R}^{M\times M}$, where $\tilde{B}$ is discretization matrix defined in (2.4), then there exists $\eta > 0$ such that*

$$b_{ii} < 0 \text{ and } |b_{ii}| \geq \sum_{\substack{j=1\\ j\neq i}}^{M} |b_{ij}| + \eta, \quad 1 \leq i \leq M.$$

Proof By the definition (2.4), it is easy to show that $B = \tilde{B} + \tilde{B}^T$ has the following form

$$B = \begin{bmatrix} 2\omega_1^{(\alpha)} & \omega_0^{(\alpha)} + \omega_2^{(\alpha)} & \omega_3^{(\alpha)} & \cdots & \omega_{M-1}^{(\alpha)} & \omega_M^{(\alpha)} \\ \omega_0^{(\alpha)} + \omega_2^{(\alpha)} & 2\omega_1^{(\alpha)} & \omega_0^{(\alpha)} + \omega_2^{(\alpha)} & \ddots & \cdots & \omega_{M-1}^{(\alpha)} \\ \omega_3^{(\alpha)} & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \omega_3^{(\alpha)} \\ \omega_{M-1}^{(\alpha)} & \ddots & \ddots & \ddots & 2\omega_1^{(\alpha)} & \omega_0^{(\alpha)} + \omega_2^{(\alpha)} \\ \omega_M^{(\alpha)} & \omega_{M-1}^{(\alpha)} & \cdots & \omega_3^{(\alpha)} & \omega_0^{(\alpha)} + \omega_2^{(\alpha)} & 2\omega_1^{(\alpha)} \end{bmatrix}.$$

From (2.2)-(2.3), we have (see [39])

$$\omega_0^{(\alpha)} = \frac{\alpha}{2}, \quad \omega_1^{(\alpha)} = \frac{2-\alpha-\alpha^2}{2}, \quad \omega_2^{(\alpha)} = \frac{\alpha(\alpha^2+\alpha-4)}{4},$$
$$1 \geq \omega_0^{(\alpha)} \geq \omega_3^{(\alpha)} \geq \omega_4^{(\alpha)} \geq \cdots \geq 0, \quad \sum_{k=0}^{\infty} \omega_k^{(\alpha)} = 0.$$

It follows from $1 < \alpha < 2$ that $b_{ii} = 2\omega_1^{(\alpha)} < 0$ and $\omega_0^{(\alpha)} + \omega_2^{(\alpha)} = \frac{\alpha(\alpha^2+\alpha-2)}{4} > 0$. Then we obtain that $b_{ij} \geq 0,\ i \neq j$. According to $\sum_{k=0}^{\infty} \omega_k^{(\alpha)} = 0$, we have

$$\begin{aligned} |b_{ii}| &= -2\omega_1^{(\alpha)} = 2(\omega_0^{(\alpha)} + \omega_2^{(\alpha)}) + 2\sum_{k=3}^{M} \omega_k^{(\alpha)} + 2\sum_{k=M+1}^{\infty} \omega_k^{(\alpha)} \\ &\geq \max_{1\leq i\leq M} \sum_{\substack{j=1\\ j\neq i}}^{M} b_{ij} + 2\sum_{k=M+1}^{\infty} \omega_k^{(\alpha)} = \max_{1\leq i\leq M} \sum_{\substack{j=1\\ j\neq i}}^{M} |b_{ij}| + 2\sum_{k=M+1}^{\infty} \omega_k^{(\alpha)}. \end{aligned}$$

The proof is then completed by setting $\eta := 2\sum_{k=M+1}^{\infty} \omega_k^{(\alpha)}$. □

Lemma 3.2 *(see [11]) Let $L = (l_{ij}) \in \mathbb{R}^{M\times M}$ with $l_{ii} < 0,\ 1 \leq i \leq M$, and there exists $\gamma > 0$ such that*

$$|l_{ii}| \geq \sum_{\substack{j=1\\ j\neq i}}^{M} |l_{ij}| + \gamma, \quad 1 \leq i \leq M.$$

Then

$$\|e^{\tau L}\|_\infty \le e^{-\tau\gamma}, \quad \forall \tau > 0.$$

The following results follow directly from the above two lemmas.

Lemma 3.3 *For any $\kappa > 0$ and $\tau > 0$, we always have $\|e^{\tau A_1}\|_\infty \le e^{-\tau\kappa}$ and $\|e^{\tau A_2}\|_\infty \le 1$.*

Proof Let $A_1 = (a_{ij}^{(1)}) \in R^{N\times N}$ and $A_2 = (a_{ij}^{(2)}) \in R^{N\times N}$. According to the definitions (2.5), (2.15)-(2.16) and the Lemma 3.1, we have

$$a_{ii}^{(1)} < 0 \text{ and } |a_{ii}^{(1)}| \ge \sum_{\substack{j=1\\ j\neq i}}^{N} |a_{ij}^{(1)}| + \zeta + \kappa, \quad 1 \le i \le M,$$

$$a_{ii}^{(2)} < 0 \text{ and } |a_{ii}^{(2)}| \ge \sum_{\substack{j=1\\ j\neq i}}^{N} |a_{ij}^{(2)}| + (d-1)\zeta, \quad 1 \le i \le M,$$

where d is the spatial dimension of the FAC equation and $\zeta = \frac{\varepsilon^2\eta}{-2h^\alpha \cos\frac{\pi\alpha}{2}} > 0$. Here η is defined in Lemma 3.1 as $\eta = 2\sum_{k=M+1}^{\infty} \omega_k^{(\alpha)}$. It then follows from Lemma 3.2 that

$$\|e^{\tau A_1}\|_\infty \le e^{-\tau(\zeta+\kappa)} \le e^{-\tau\kappa}, \quad \|e^{\tau A_2}\|_\infty \le e^{-\tau\zeta(d-1)} \le 1,$$

which completes the proof. □

Lemma 3.4 *(see [11]) Define $f(\xi) := (\kappa+1)\xi - \xi^3$ for any $\xi \in \mathbb{R}$. If $\kappa \ge 2$, then*

$$|f(\xi)| \le \kappa \text{ and } |f'(\xi)| \le \kappa + 1, \quad \forall \xi \in [-1, 1].$$

Lemma 3.5 *(see [11]) Let $\tau > 0$ and $\kappa > 0$, then*

$$e^{-\tau\kappa} + \int_0^\tau \kappa e^{-(\tau-s)\kappa} ds = 1.$$

Now we are in the position to give the following theorem.

Theorem 3.1 *Assume that the initial data satisfies $\|u_0\|_{L^\infty} \le 1$. For any time step size $\tau > 0$, the DSETDRK2 scheme (2.18) preserves the discrete maximum principle, i.e.,*

$$\|U^n\|_\infty \le 1, \quad \forall n \ge 0,$$

provided the stabilized parameter $\kappa \ge 2$.

Proof Following the proof of Theorem 3.1 in [11], we prove the theorem by induction. Obviously, $\|U^0\|_\infty \le \|u_0\|_{L^\infty} \le 1$. Now assume that the result holds for $n = k$, i.e., $\|U^k\|_\infty \le 1$. Next we prove the result holds for $n = k+1$. According to Lemma 3.3-3.4, we have

$$\|e^{\tau A_1}\|_\infty \le e^{-\tau\kappa}, \ \|e^{\tau A_2}\|_\infty \le 1, \ \|e^{(\tau-s)A_1}\|_\infty \le e^{-(\tau-s)\kappa}, \ \|f(U^k)\|_\infty \le \kappa,$$

where $s \in [0, \tau]$. From the DSETDRK2 scheme (2.18a), the bounds obtained above and Lemma 3.5, we find that

$$\|\hat{U}^{k+1}\|_\infty \le \|e^{\tau A_1}\|_\infty \|e^{\tau A_2}\|_\infty \|U^k\|_\infty + \int_0^\tau \|e^{(\tau-s)A_1}\|_\infty ds \cdot \|e^{\tau A_2}\|_\infty \|f(U^k)\|_\infty$$
$$\le e^{-\tau\kappa} + \int_0^\tau \kappa e^{-(\tau-s)\kappa} ds = 1.$$

Using Lemma 3.4, we have $\|f(\hat{U}^{k+1})\|_\infty \le \kappa$. Similarly, from (2.18b), we obtain

$$\|U^{k+1}\|_\infty \le \|e^{\tau A_1}\|_\infty \|e^{\tau A_2}\|_\infty \|U^k\|_\infty + \int_0^\tau \|e^{(\tau-s)A_1}\|_\infty \left\| \left(1 - \frac{s}{\tau}\right) e^{\tau A_2} f(U^k) + \frac{s}{\tau} f(\hat{U}^{k+1}) \right\|_\infty ds$$
$$\le e^{-\tau\kappa} + \int_0^\tau \left(\left(1 - \frac{s}{\tau}\right)\kappa + \frac{s}{\tau}\kappa \right) e^{-(\tau-s)\kappa} ds = 1,$$

which completes the proof. □

In the rest of the paper, we will always assume that $\kappa \ge 2$ for the proposed DSETDRK2 scheme so that it preserves the discrete maximum principle.

3.2 Convergence Analysis

In this section, we establish the L^∞ error estimate for the numerical solution produced by DSETDRK2 scheme (2.18) for the FAC equation (1.1). Denote $\tilde{U}(t)$ as the solution vector of (1.1) on the spatial mesh, i.e., $\tilde{U}(t) = [u(\mathbf{x}_1, t), u(\mathbf{x}_2, t), \ldots, u(\mathbf{x}_N, t)]^T$. Then we have

$$\frac{d\tilde{U}(t)}{dt} = A\tilde{U}(t) + f(\tilde{U}(t)) + R_h, \tag{3.1}$$

where R_h is the local truncation error. If the exact solution $u \in C^5(\overline{\Omega})$, it follows that [39]

$$\|R_h\|_\infty \le C_1 \varepsilon^2 h^2, \tag{3.2}$$

where C_1 is a positive constant depending on the exact solution u but not depending on h.

Theorem 3.2 *Assume that the exact solution u of the FAC equation (1.1) belongs to $C^2([0,T]; C^5(\overline{\Omega}))$ and $\|u_0\|_\infty \le 1$. Then we have the following error estimate for the DSETDRK2 scheme (2.18) as*

$$\|U^n - \tilde{U}(t_n)\|_\infty \le C e^{t_n}(h^2 + \tau^2), \quad t_n \le T,$$

where $C > 0$ is a constant depends on the $C^2([0,T]; C^5(\overline{\Omega}))$-norm of u, but independent of h and τ.

Proof Note that $A = A_1 + A_2$ and $e^A = e^{A_1} e^{A_2}$, then by applying the variation-of-constant formula to (3.1), we obtain

$$\tilde{U}(t_{n+1}) = e^{\tau A_1} e^{\tau A_2} \tilde{U}(t_n) + \int_0^\tau e^{(\tau-s)A_1} e^{(\tau-s)A_2} \left[f(\tilde{U}(t_n + s)) + R_h \right] ds. \tag{3.3}$$

Let $e^n = U^n - \tilde{U}(t_n)$ and $\hat{e}^{n+1} = \hat{U}^{n+1} - \tilde{U}(t_{n+1})$. Subtracting (3.3) from (2.18a) and taking norms, we have

$$\|\hat{e}^{n+1}\|_\infty \le \|e^{\tau A_1}\|_\infty \|e^{\tau A_2}\|_\infty \|e^n\|_\infty + \int_0^\tau \|e^{(\tau-s)A_1}\|_\infty (\|g_1\|_\infty + \|e^{(\tau-s)A_2} R_h\|_\infty) ds, \tag{3.4}$$

where $g_1 := e^{(\tau-s)A_2} f(\tilde{U}(t_n+s)) - e^{\tau A_2} f(U^n)$. By applying Lemma 3.3, we have, for $s \in (0,\tau)$, that

$$\|e^{\tau A_1}\|_\infty \le e^{-\tau\kappa},\ \|e^{\tau A_2}\|_\infty \le 1,\ \|e^{(\tau-s)A_1}\|_\infty \le e^{-(\tau-s)\kappa},\ \|e^{(\tau-s)A_2}\|_\infty \le 1. \quad (3.5)$$

Using (3.5), we have

$$\begin{aligned}\|g_1\|_\infty &\le \|e^{(\tau-s)A_2} f(\tilde{U}(t_n+s)) - e^{\tau A_2} f(\tilde{U}(t_n))\|_\infty + \|e^{\tau A_2}[f(\tilde{U}(t_n)) - f(U^n)]\|_\infty \\ &\le \|\psi(s)-\psi(0)\|_\infty + \|f(\tilde{U}(t_n)) - f(U^n)\|_\infty,\end{aligned}$$

where $\psi(s) := e^{(\tau-s)A_2} f(\tilde{U}(t_n+s))$. Since $u \in C^2([0,T]; C^5(\overline{\Omega}))$, we have that $\psi(s) \in C^2([0,T])$. Then according to the estimate for constant interpolation, we have, for $s \in (0,\tau)$, that $\|\psi(s)-\psi(0)\|_\infty \le C_2\tau$, where C_2 depends on the $C^1([0,T]; C^5(\overline{\Omega}))$-norm of u. Since $|f'(\theta)| \le \kappa+1$ for any $\theta \in [-1,1]$ and both the exact and numerical solutions satisfy the maximum principles, we get

$$\|f(\tilde{U}(t_n)) - f(U^n)\|_\infty \le (\kappa+1)\|\tilde{U}(t_n) - U^n\|_\infty = (\kappa+1)\|e^n\|_\infty.$$

Thus, we obtain

$$\|g_1\|_\infty \le (\kappa+1)\|e^n\|_\infty + C_2\tau.$$

It then follows from (3.2) and (3.4)-(3.5) that

$$\begin{aligned}\|\hat{e}^{n+1}\|_\infty &\le e^{-\tau\kappa}\|e^n\|_\infty + \left[(\kappa+1)\|e^n\|_\infty + C_2\tau + C_1\varepsilon^2h^2\right]\int_0^\tau e^{-(\tau-s)\kappa}ds \\ &= \left(1+\frac{1-e^{-\tau\kappa}}{\kappa}\right)\|e^n\|_\infty + \frac{1-e^{-\tau\kappa}}{\kappa}(C_2\tau + C_1\varepsilon^2h^2) \\ &\le (1+\tau)\|e^n\|_\infty + \tau(C_2\tau + C_1\varepsilon^2h^2), \end{aligned} \quad (3.6)$$

where we have used the fact that $1-e^{-z} \le z$ for any $z > 0$.

Now, subtracting (3.3) from (2.18b) and taking norms, we have

$$\begin{aligned}\|e^{n+1}\|_\infty &\le \|e^{\tau A_1}\|_\infty\|e^{\tau A_2}\|_\infty\|e^n\|_\infty + \int_0^\tau \|e^{(\tau-s)A_1}\|_\infty(\|g_2\|_\infty + \|e^{(\tau-s)A_2}R_h\|_\infty)ds, \\ &\le e^{-\tau\kappa}\|e^n\|_\infty + \int_0^\tau e^{-(\tau-s)\kappa}(\|g_2\|_\infty + C_1\varepsilon^2h^2)ds, \end{aligned} \quad (3.7)$$

where

$$g_2 = \left(1-\frac{s}{\tau}\right)e^{\tau A_2} f(U^n) + \frac{s}{\tau} f(\hat{U}^{n+1}) - e^{(\tau-s)A_2} f(\tilde{U}(t_n+s)).$$

For any $s \in (0,\tau)$, we have that

$$\begin{aligned}\|g_2\|_\infty &\le \left(1-\frac{s}{\tau}\right)\|e^{\tau A_2}[f(U^n) - f(U(t_n))]\|_\infty + \frac{s}{\tau}\|f(\hat{U}^{n+1}) - f(U(t_{n+1}))\|_\infty \\ &\quad + \left\|\left(1-\frac{s}{\tau}\right)e^{\tau A_2} f(U(t_n)) + \frac{s}{\tau} f(U(t_{n+1})) - e^{(\tau-s)A_2} f(\tilde{U}(t_n+s))\right\|_\infty \\ &\le \left(1-\frac{s}{\tau}\right)\|f(U^n) - f(U(t_n))\|_\infty + \frac{s}{\tau}\|f(\hat{U}^{n+1}) - f(U(t_{n+1}))\|_\infty \\ &\quad + \left\|\left(1-\frac{s}{\tau}\right)\psi(0) + \frac{s}{\tau}\psi(\tau) - \psi(s)\right\|_\infty.\end{aligned}$$

Since $\psi(s) \in C^2([0,T])$, then by the estimate for linear interpolation, we have

$$\left\|\left(1-\frac{s}{\tau}\right)\psi(0) + \frac{s}{\tau}\psi(\tau) - \psi(s)\right\|_\infty \le C_3\tau^2,$$

where C_3 depends on the $C^2([0,T];C^5(\overline{\Omega}))$-norm of u. It then follows from the Lipschitz continuity of f that

$$\|g_2\|_\infty \le \left(1-\frac{s}{\tau}\right)(\kappa+1)\|e^n\|_\infty + \frac{s}{\tau}(\kappa+1)\|\hat{e}^{n+1}\|_\infty + C_3\tau^2. \tag{3.8}$$

By combining (3.6),(3.7) and(3.8), we have

$$\begin{aligned}
\|e^{n+1}\|_\infty &\le e^{-\tau\kappa}\|e^n\|_\infty + \left[(\kappa+1)\|e^n\|_\infty + C_1\varepsilon^2h^2 + C_3\tau^2\right]\int_0^\tau e^{-(\tau-s)\kappa}ds \\
&\quad + \int_0^\tau \frac{s}{\tau}(\kappa+1)e^{-(\tau-s)\kappa}(\|\hat{e}^{n+1}\|_\infty - \|e^n\|_\infty)ds \\
&\le e^{-\tau\kappa}\|e^n\|_\infty + \frac{1-e^{-\tau\kappa}}{\kappa}\left[(\kappa+1)\|e^n\|_\infty + C_1\varepsilon^2h^2 + C_3\tau^2\right] \\
&\quad + (\kappa+1)(\|e^n\|_\infty + C_2\tau + C_1\varepsilon^2h^2)\int_0^\tau se^{-(\tau-s)\kappa}ds \\
&= \left(1+\tau+\frac{e^{-\tau\kappa}-1+\tau\kappa}{\kappa^2}\right)\|e^n\|_\infty + \frac{1-e^{-\tau\kappa}}{\kappa}(C_1\varepsilon^2h^2 + C_3\tau^2) \\
&\quad + \frac{e^{-\tau\kappa}-1+\tau\kappa}{\kappa^2}(\kappa+1)(C_1\varepsilon^2h^2 + C_2\tau) \\
&\le \left(1+\tau+\frac{\tau^2}{2}\right)\|e^n\|_\infty + \tau(C_1\varepsilon^2h^2 + C_3\tau^2) + \frac{\tau^2}{2}(\kappa+1)(C_1\varepsilon^2h^2 + C_2\tau) \\
&\le \left(1+\tau+\frac{\tau^2}{2}\right)\|e^n\|_\infty + C\tau(h^2+\tau^2),
\end{aligned}$$

where $C = \max\left\{\frac{C_2+2C_3}{2}, \frac{\kappa+3}{2}C_1\varepsilon^2\right\}$. We have used the condition $\tau \le 1$ in the last step of the above inequality. By induction it follows that

$$\begin{aligned}
\|e^{n+1}\|_\infty &\le \left(1+\tau+\frac{\tau^2}{2}\right)^n\|e^0\|_\infty + C\tau(h^2+\tau^2)\sum_{j=0}^{n-1}\left(1+\tau+\frac{\tau^2}{2}\right)^j \\
&\le e^{n\tau}(\|e^0\|_\infty + C(h^2+\tau^2)).
\end{aligned}$$

Thus, the desired result follows as $\|e^0\|_\infty = 0$. □

4 Implementation Issues

In this section, we illustrate the practical implementation of the proposed scheme (2.19). First, we present some properties related to Kronecker products.

Lemma 4.1 *(see [19,23]) The following properties related to Kronecker products hold:*

(1) $e^{B\otimes I} = e^B \otimes I$, $e^{I\otimes B} = I \otimes e^B$, $(e^B)^T = e^{B^T}$;
(2) $e^{B\otimes I + I\otimes C} = e^B \otimes e^C$;
(3) $(B\otimes C)(D\otimes E) = (BD)\otimes(CE)$;
(4) $(B\otimes C)^{-1} = B^{-1}\otimes C^{-1}$ *if* B *and* C *are nonsingular;*
(5) $(B\otimes C)w = vec(CWB^T)$, $vec(W) = w$.

Here I denotes the identity matrix and the vec operator transforms matrices into vectors by stacking columns as follows

$$W = [w_1, w_2, \ldots, w_m] \in \mathbb{R}^{n\times m} \Longleftrightarrow \mathrm{vec}(W) = [w_1^T, w_2^T, \ldots, w_m^T]^T \in \mathbb{R}^{mn}.$$

Lemma 4.2 *Let the φ-functions be defined by*

$$\varphi_k(z) = z\varphi_{k+1}(z) + \frac{1}{k!}, \quad \varphi_0(z) = e^z.$$

Assume that B is nonsingular, then we have

$$\varphi_k(I \otimes B) = I \otimes \varphi_k(B), \quad k = 0, 1, \ldots.$$

Proof We prove this lemma by induction. Obviously, by using the first property in Lemma 4.1 we have

$$\varphi_0(I \otimes B) = e^{I\otimes B} = I \otimes e^B = I \otimes \varphi_0(B).$$

Now assume that the result holds for $k = j$, i.e., $\varphi_j(I \otimes B) = I \otimes \varphi_j(B)$. Next we check whether this holds for $k = j + 1$. By using the definition of the φ-functions and Lemma 4.1, it is easy to show that

$$\begin{aligned}\varphi_{j+1}(I \otimes B) &= (I \otimes B)^{-1}\left(\varphi_j(I \otimes B) - \frac{1}{j!}I \otimes I\right) = (I \otimes B^{-1})\left(I \otimes \varphi_j(B) - \frac{1}{j!}I \otimes I\right)\\ &= (I \otimes B^{-1})\left(I \otimes \left(\varphi_j(B) - \frac{1}{j!}I\right)\right) = I \otimes \left(B^{-1}\left(\varphi_j(B) - \frac{1}{j!}I\right)\right)\\ &= I \otimes \varphi_{j+1}(B),\end{aligned}$$

which completes the proof. □

Here we focus on the implementation of the scheme (2.19) for solving 2D FAC equation (1.1), the implementation for 3D problems is similar and omitted here. Denote $\tilde{K} := K - \kappa I_M$ and we have that $e^{\tau K} = e^{\tau\kappa}e^{\tau\tilde{K}}$. Note that both K and $\tilde{K}$ are symmetric and negative definite Toeplitz matrices. Then by using (2.15), Lemma 4.1 and Lemma 4.2, we can rewrite the DSETDRK2 scheme (2.19) in the following form

$$\hat{U}^{n+1} = e^{\tau\kappa}\left(e^{\tau\tilde{K}} \otimes e^{\tau\tilde{K}}\right)U^n + \tau e^{\tau\kappa}\left(e^{\tau\tilde{K}} \otimes \varphi_1(\tau\tilde{K})\right)f(U^n), \tag{4.1a}$$

$$U^{n+1} = \hat{U}^{n+1} + \tau\left(I_M \otimes \varphi_2(\tau\tilde{K})\right)\left[f(\hat{U}^{n+1}) - e^{\tau\kappa}\left(e^{\tau\tilde{K}} \otimes I_M\right)f(U^n)\right], \tag{4.1b}$$

or, equivalently,

$$\hat{\mathcal{U}}^{n+1} = e^{\tau\kappa}e^{\tau\tilde{K}}\mathcal{U}^n e^{\tau\tilde{K}} + \tau e^{\tau\kappa}\varphi_1(\tau\tilde{K})\mathcal{F}^n e^{\tau\tilde{K}}, \tag{4.2a}$$

$$\mathcal{U}^{n+1} = \hat{\mathcal{U}}^{n+1} + \tau\varphi_2(\tau\tilde{K})\left[\hat{\mathcal{F}}^{n+1} - e^{\tau\kappa}\mathcal{F}^n e^{\tau\tilde{K}}\right], \tag{4.2b}$$

where

$$\mathrm{vec}(\hat{\mathcal{U}}^{n+1}) = \hat{U}^{n+1},\ \mathrm{vec}(\mathcal{U}^n) = U^n,\ \mathrm{vec}(\mathcal{F}^n) = f(U^n),\ \mathrm{vec}(\hat{\mathcal{F}}^{n+1}) = f(\hat{U}^{n+1}).$$

By the definition of the φ-functions, we can also write (4.2) in the following form

$$\hat{\mathcal{U}}^{n+1} = \left[e^{\tau\tilde{K}}\mathcal{U}^n + (e^{\tau\tilde{K}} - I_M)\tilde{K}^{-1}\mathcal{F}^n\right]e^{\tau\kappa}e^{\tau\tilde{K}}, \tag{4.3a}$$

$$\mathcal{U}^{n+1} = \hat{\mathcal{U}}^{n+1} + \tau^{-1}(e^{\tau\tilde{K}} - \tau\tilde{K} - I_M)\tilde{K}^{-2}\left[\hat{\mathcal{F}}^{n+1} - e^{\tau\kappa}\mathcal{F}^n e^{\tau\tilde{K}}\right]. \tag{4.3b}$$

Every step of the DSETDRK2 scheme requires the computation of matrix-vector products with φ-functions. In the last two decades a number of efficient algorithms have been developed for evaluating products of φ-functions with vectors. Here we mention contour integration methods [26,35], scaling and squaring methods [2,19,20] and Krylov subspace methods [16,21,31]. We remark that the explicit formulas for functions $\varphi_k(z)$ with $k > 0$ possess a removable discontinuity at $z = 0$, for small z a straightforward evaluation of $\varphi_k(z)$ suffers from cancellation errors [26]. Thus, straightforward computation of $\varphi_k(\tau\tilde{K})$ as in (4.3) could cause numerical roundoff error if the matrix $\tau\tilde{K}$ has eigenvalues near the origin. Note that $\tilde{K} = K - \kappa I_M$ and K is symmetric and negative definite, it follows that the maximum eigenvalue of $\tau\tilde{K}$ is smaller than $-\tau\kappa$. Therefore, we can use the formulae (4.3) when time step size τ is not too small.

From (4.3) we see that the implementation of the DSETDRK2 scheme requires computation of matrix-vector products with $\tilde{K}^{-1}$. Note that $\tilde{K}$ is a symmetric Toeplitz matrix and then $\tilde{K}^{-1}$ can be expressed explicitly in the Gohberg-Semencul-type formula [17]. Let the vector $p = [p_1, \ldots, p_M]^T$ satisfies the equation

$$\tilde{K}p = e_1, \tag{4.4}$$

where e_1 is the first column of the identity matrix I_M. Then by the Gohberg-Semencul-type formula [17], we have the following representation for $\tilde{K}^{-1}$:

$$\tilde{K}^{-1} = \frac{1}{2p_1}(S_1C_1 - S_2C_2), \tag{4.5}$$

where S_1, S_2 are skew-circulant matrices with $p, \bar{p} = [-p_1, p_M, \ldots, p_2]^T$ as their first columns, respectively; C_1, C_2 are circulant matrices with $\hat{p} = [p_1, p_M, \ldots, p_2]^T$, p as their first columns, respectively. From (4.4), we can see that p_1 is the first diagonal entry of $\tilde{K}^{-1}$. Since $\tilde{K}$ is symmetric and negative definite and so is $\tilde{K}^{-1}$, we obtain that $p_1 < 0$, which means (4.5) is applicable.

Let F_M be the $M \times M$ Fourier matrix and $\Omega_M = \mathrm{diag}[1, e^{-i\pi/M}, \ldots, e^{-i(M-1)\pi/M}]$, where $i = \sqrt{-1}$, then we have the following spectral decomposition:

$$C_j = F_M^* \Lambda_{C_j} F_M, \quad S_j = \Omega_M^* F_M^* \Lambda_{S_j} F_M \Omega_M, \quad j = 1, 2, \tag{4.6}$$

where Λ_{C_j}, Λ_{S_j} are diagonal matrices containing the eigenvalues of C_j and S_j, respectively. By (4.5)-(4.6), the matrix-vector multiplication associated with matrix $\tilde{K}^{-1}$ can be obtained by the fast Fourier transform (FFT) [5,6]. For solving Toeplitz linear systems in (4.4), we can employ the fast direct Toeplitz solver or preconditioned conjugate gradient (PCG) method with Strang-type circulant preconditioner [5,6].

The implementation of the DSETDRK2 scheme (4.3) also requires the computation of the matrix exponential $e^{\tau\tilde{K}}$ or products of $e^{\tau\tilde{K}}$ with vectors. Since $\tilde{K}$ is 1D discretization matrix of size $M \times M$, we can use scaling and squaring methods [19] to compute $e^{\tau\tilde{K}}$ when M is not too large. In this case, we need to compute matrix exponential $e^{\tau\tilde{K}}$ just for once in the whole simulation since the matrix $\tau\tilde{K}$ arising in all the time steps are the same. If the dimension of the matrix $\tilde{K}$ is too large, we will use the shift-invert Lanczos method [32] to approximate the products of the matrix exponential $e^{\tau\tilde{K}}$ with vectors. We refer the interest readers to [27,28,33,42–44] for related studies on fast algorithms for Toeplitz matrix exponential.

A Krylov subspace method for approximating $e^{\tau\tilde{K}}v$ first constructs a basis of a Krylov subspace with respect to the vector v and the matrix $\tilde{K}$. Instead of using $\tilde{K}$ as the generating

matrix of the Krylov subspace, the shift-invert Lanczos method uses $(I_M - \gamma\tilde{K})^{-1}$ as the generating matrix and the Krylov subspace is denoted by

$$\mathcal{K}_m = \text{span}\{v, (I_M - \gamma\tilde{K})^{-1}v, \ldots, (I_M - \gamma\tilde{K})^{1-m}v\},$$

where $\gamma > 0$ is a shift parameter.

The Lanczos method constructs an orthonormal basis $Q_m = [q_1, q_2, \ldots, q_m] \in \mathbb{R}^{M\times m}$ of $\mathcal{K}_m$ and a symmetric tridiagonal matrix $D_m = (d_{ij}) \in \mathbb{R}^{m\times m}$ satisfying

$$(I_M - \gamma\tilde{K})^{-1}Q_m = Q_m D_m + d_{m+1,m}q_{m+1}e_m^T.$$

Here e_m is the mth vector of the canonical basis of $\mathbb{R}^m$. Then we can approximate $e^{\tau\tilde{K}}v$ by [32]

$$e^{\tau\tilde{K}}v \approx \|v\|_2 Q_m e^{\frac{\tau}{\gamma}(I_m - D_m^{-1})}e_1.$$

The computation of the smaller matrix exponential $e^{\frac{\tau}{\gamma}(I_m - D_m^{-1})}$ can be done by standard methods such as Padé approximation or scaling and squaring method [19].

In each iteration step of the shift-invert Lanczos method, we need to solve a linear system

$$(I_M - \gamma\tilde{K})w = r.$$

Note that $I_M - \gamma\tilde{K}$ is a symmetric positive definite (SPD) Toeplitz matrix. Similar to (4.4)-(4.6), $(I_M - \gamma\tilde{K})^{-1}r$ can be computed efficiently by the Gohberg-Semencul-type formula and FFT. For details on the implementation of the shift-invert Lanczos algorithm we refer to [32].

We remark that the solution of fractional models may possess boundary singularities, and hence graded meshes are preferred. In this case, the stiffness matrix is dense and not Toeplitz-like, and hence the FFT-based methods introduced above can not be used. One promising strategy for efficiently computing matrix exponential or φ-functions associated with dense and unstructured matrices (or their product with vectors) is to apply scaling and squaring method together with a truncated Taylor series approximation to the exponential [2]. In addition, Krylov subspace-based method KIOPS is also a powerful tool for computing linear combinations of φ-functions that appear in exponential integrators [16].

5 Numerical Experiments

In this section, we provide some numerical experiments conducted in order to demonstrate the effectiveness and efficiency of the DSETDRK2 scheme (2.19) for solving the FAC equation (1.1). In the first part we verify the convergence rate of the proposed scheme with a smooth initial data, while in the second part we investigate the discrete maximum principle beginning with a random initial state. For numerical comparison, we also present numerical results obtained by applying the ETDRK2 scheme (2.14), which can be written as (for 2D problem)

$$\hat{U}^{n+1} = e^{\tau\kappa}(e^{\tau\tilde{K}} \otimes e^{\tau\tilde{K}})U^n + (e^{\tau\kappa}e^{\tau\tilde{K}} \otimes e^{\tau\tilde{K}} - I_N)A^{-1}f(U^n), \tag{5.1a}$$

$$U^{n+1} = \hat{U}^{n+1} + \tau^{-1}(e^{\tau\kappa}e^{\tau\tilde{K}} \otimes e^{\tau\tilde{K}} - \tau A - I_N)A^{-2}[f(\hat{U}^{n+1}) - f(U^n)], \tag{5.1b}$$

where $A = I_M \otimes K + K \otimes I_M - \kappa I_M \otimes I_M$. Note that A is a block Toeplitz matrix with Toeplitz blocks (BTTB). Unfortunately, there is no Gohberg-Semencul-type formula for the inverse of the BTTB matrix. We will use PCG method with Strang-type block circulant

preconditioner [25] to solve the linear systems associated with A in the implementation of the ETDRK2 scheme. We take the stabilizing parameter $\kappa = 2$ for both the DSETDRK2 and ETDRK2 schemes in all experiments.

We also compare our method with the Crank-Nicolson scheme [24]:

$$\left(I_N - \frac{\tau}{2}\tilde{A}\right) U^{n+1} = \left(I_N + \frac{\tau}{2}\tilde{A}\right) U^n + \frac{\tau}{2}\left(U^n - (U^n)^{\cdot 3}\right) + \frac{\tau}{2}\left(U^{n+1} - (U^{n+1})^{\cdot 3}\right) \quad (5.2)$$

which is solved by the linear iterative method

$$\left(I_N - \frac{\tau}{2}\tilde{A}\right) V_{k+1} = G + \frac{\tau}{2}\left(V_k - (V_k)^{\cdot 3}\right), \quad k = 0, 1, \ldots, \quad (5.3)$$

with $G = (I_N + \frac{\tau}{2}\tilde{A})U^n + \frac{\tau}{2}(U^n - (U^n)^{\cdot 3})$ and initial guess $V_0 = U^n$. Here $\tilde{A}$ is as defined in (2.7) for 2D problem and (2.8) for 3D problem. We remark that a nonlinear iterative method has been proposed for solving (5.2) in [24], where one needs to compute real root of cubic equations N times in every iteration. However, numerical experiments show that the nonlinear iterative method works only for very small step size τ for our examples. So we use the linear iterative method (5.3) for the Crank-Nicolson scheme in our tests.

We use PCG method with Strang-type block circulant preconditioner [25] to solve the linear systems in (5.3). We use the MATLAB built-in function *pcg* as the PCG solver and choose a zero initial guess and a stopping tolerance tol $= 10^{-8}$ based on the reduction in relative residual norms. The iteration (5.3) is stopped when $\|V_{k+1} - V_k\|_\infty \le 10^{-8}$. All simulations are implemented using MATLAB on a laptop with Intel(R) Core(TM) i7-8565U CPU@1.80GHz and 16GB RAM.

5.1 Convergence Tests

Example 5.1 We consider a 2D FAC equation (1.1) with $\varepsilon = 0.1$ on the domain $\Omega = (0, 1)^2$ up to the time $T = 0.5$. We set the smooth initial data as $u_0(x, y) = \sin(2\pi x)\sin(2\pi y)$.

First, we test the convergence rates in space of the DSETDRK2 scheme. To eliminate the time-marching effect, we consider the DSETDRK2 scheme with $\tau = T$. In other words, we only test the convergence of the spatial finite difference scheme applied to an elliptic equation. We take the numerical solution of the FAC equation obtained by the DSETDRK2 scheme with $M = 1024$ as the benchmark solution. The numerical errors in the maximum-norm are presented in Table 1. We see that the numerical convergence rates with respect to h are approximately second order in most cases. However, we observe the order reduction when h becomes smaller ($h = \frac{1}{64}$). Figure 1 plots the errors of the numerical solutions with $h = \frac{1}{32}, \frac{1}{64}, \frac{1}{128}$ for the case $\alpha = 1.8$. We find that the errors near the boundary become relatively larger than the errors away from the boundary when h decreases. It appears that the order reduction is due to the possible boundary singularities of the FAC equation. We remark that graded meshes are more suitable for fractional models due to the possible singularities near the boundary and this topic is beyond the scope of this paper and will be carried out in our future work.

Second, we test the convergence rates in time. We fix $h = 2^{-9}$ ($M = 511$) and compute the numerical solutions of the 2D FAC equation using the DSETDRK2, ETDRK2 and Crank-Nicolson schemes with various time step sizes. The approximation solution obtained by using the Crank-Nicolson scheme with $\tau = 10^{-4}$ is taken as the benchmark solution for computing errors. The maximum norms of the numerical errors and corresponding CPU times (in seconds) are reported in Table 2. We observe that the desired second order accuracy for the DSETDRK2 scheme is attained. In addition, for fixed τ, we see that the errors of the

Table 1 Spatial convergence rates in the maximum-norm for Example 5.1

h	$\alpha = 1.2$		$\alpha = 1.5$		$\alpha = 1.8$	
	Error	Rate	Error	Rate	Error	Rate
1/8	6.8984e-3	-	1.2464e-2	-	1.6021e-2	-
1/16	2.9896e-3	1.2063	4.0434e-3	1.6241	4.5916e-3	1.8029
1/32	6.5915e-4	2.1813	1.0873e-3	1.8948	1.3240e-3	1.7941
1/64	1.0847e-3	*	1.6651e-3	*	6.2423e-4	1.0848

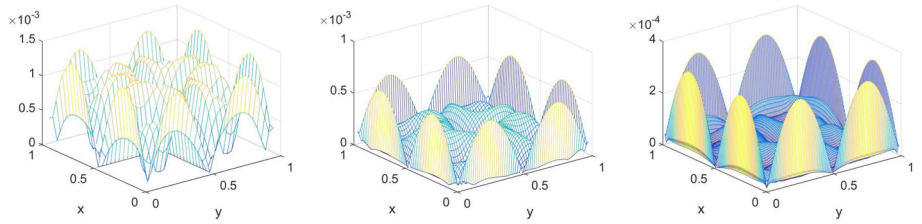

Fig. 1 Error of the numerical solution for Example 5.1 with $\alpha = 1.8$. From left to right: $h = \frac{1}{32}, \frac{1}{64}, \frac{1}{128}$

Table 2 Comparison of errors and CPU times of the DSETDRK2, ETDRK2 and Crank-Nicolson schemes for Example 5.1

α	τ	DSETDRK2			ETDRK2			Crank-Nicolson		
		Error	Rate	CPU	Error	Rate	CPU	Error	Rate	CPU
1.2	1/10	2.170e-3	-	1.3	1.976e-3	-	8.6	1.532e-4	-	12.4
	1/20	5.926e-4	1.873	2.6	5.395e-4	1.873	21.2	3.812e-5	2.007	23.2
	1/40	1.550e-4	1.935	5.0	1.411e-4	1.935	43.3	9.530e-6	2.000	36.1
	1/80	3.964e-5	1.967	10.5	3.608e-5	1.967	88.7	2.375e-6	2.005	60.6
	1/160	1.002e-5	1.984	20.1	9.125e-6	1.984	182.0	5.956e-7	1.995	118.4
1.5	1/10	1.874e-3	-	1.3	1.561e-3	-	19.0	5.931e-4	-	12.7
	1/20	5.104e-4	1.877	2.6	4.251e-4	1.876	31.6	6.967e-5	3.090	30.0
	1/40	1.334e-4	1.936	5.0	1.111e-4	1.936	64.1	1.739e-5	2.002	44.9
	1/80	3.409e-5	1.966	10.4	2.840e-5	1.968	128.4	4.346e-6	2.001	73.1
	1/160	8.619e-6	1.984	19.9	7.180e-6	1.984	258.0	1.100e-6	1.982	138.5
1.8	1/10	1.350e-3	-	1.3	7.378e-4	-	22.4	1.249e-3	-	27.6
	1/20	3.685e-4	1.873	2.5	1.994e-4	1.888	48.2	4.483e-4	1.478	44.7
	1/40	9.642e-5	1.934	5.0	5.195e-5	1.941	97.8	5.119e-5	3.130	65.7
	1/80	2.467e-5	1.966	10.1	1.327e-5	1.969	201.9	7.377e-6	2.795	88.1
	1/160	6.240e-6	1.983	23.6	3.352e-6	1.984	399.0	1.843e-6	2.001	165.1

ETDRK2 scheme are slightly smaller than those of the DSETDRK2 scheme and the errors of the Crank-Nicolson scheme are smallest in most cases. For a given level of accuracy, it is observed that the DSETDRK2 scheme performs better than the ETDRK2 and Crank-Nicolson schemes in terms of computational cost.

Table 3 Spatial convergence rates in the maximum-norm for Example 5.2

h	$\alpha = 1.2$		$\alpha = 1.5$		$\alpha = 1.8$	
	Error	Rate	Error	Rate	Error	Rate
1/4	6.1570e-3	-	9.8831e-3	-	1.2780e-2	-
1/8	1.8399e-3	1.7426	3.1847e-3	1.6338	4.3548e-3	1.5532
1/16	5.9493e-4	1.6288	1.2526e-3	1.3462	1.4004e-3	1.6368
1/32	3.4559e-4	0.7837	5.3134e-4	1.2372	2.8145e-4	2.3149
1/64	2.0497e-4	0.7536	2.2085e-4	1.2666	2.5861e-4	0.1221

Example 5.2 Next, we consider a 3D FAC equation (1.1) with $\varepsilon = 0.1$ on the domain $\Omega = (0, 1)^3$ up to the time $T = 0.1$. The initial condition is chosen to be $u_0(x, y, z) = \sin(2\pi x)\sin(2\pi y)\sin(2\pi z)$.

In Table 3, we test the convergence with respect to the spatial size h by fixing $\tau = T$. The numerical solution obtained by the DSETDRK2 scheme with $M = 512$ is taken as the benchmark solution. Similar to Example 5.1, we see the order reduction due to the possible insufficient regularity near the boundary for the solution of the FAC equation (1.1).

Similar to Example 5.1, in Table 4, we compare the performance of the DSETDRK2, ETDRK2 and Crank-Nicolson schemes with respect to accuracy and computational time. Here, we fix $h = 2^{-7}$ ($M = 127$) and take the solution obtained by the Crank-Nicolson scheme with $\tau = 10^{-4}$ as the benchmark. Again, we observe that the DSETDRK2 scheme achieves second order accuracy in time. For fixed τ, we also see that the errors of the ETDRK2 are slightly smaller than those of the DSETDRK2 scheme, while the errors of the DSETDRK2 scheme are slightly smaller than those of the Crank-Nicolson scheme when $\alpha = 1.5, 1.8$. In addition, for a given level of accuracy, the DSETDRK2 scheme is fastest.

5.2 Stability Tests

Example 5.3 Let $\Omega = (0, 1)^2$ and $\varepsilon = 0.01$. Consider the 2D FAC equation (1.1) with a random initial data ranging from -0.05 to 0.05 uniformly generated on the 127×127 mesh ($h = 1/128$). The numerical solutions are computed by the DSETDRK2 scheme with time step $\tau = 0.01$ for all cases.

In Figure 2, we display the evolutions of the solutions of the FAC equation with $\alpha = 1.2, 1.6, 1.8$ at times $t = 5, 20, 100$, respectively. From Figure 2, we can see the effects of fractional diffusion on the phase separation and coarsening process. When the order α of the fractional diffusion decreases, we find that the interfaces generated by diffusion propagation becomes thinner that allow for smaller spots and a much more heterogeneous phase pattern. Moreover, the reduction of the fractional order generates a decrease in the velocity of propagation of the phase coarsening. The observations are consistent with the results of [4,24].

Figure 3 shows the evolutions of the corresponding maximum norms and the energies of the numerical solutions, respectively. The discrete energy is defined by

$$E_h(U) = h^d \left(\frac{1}{4} \sum_{j=1}^{N} (U_j^2 - 1)^2 - \frac{1}{2} U^T \tilde{A} U \right),$$

Table 4 Comparison of errors and CPU times of the DSETDRK2, ETDRK2 and Crank-Nicolson schemes for Example 5.2

α	τ	DSETDRK2			ETDRK2			Crank-Nicolson		
		Error	Rate	CPU	Error	Rate	CPU	Error	Rate	CPU
1.2	1/10	4.317e-4	-	2.0	3.495e-4	-	16.2	1.584e-4	-	33.7
	1/20	1.176e-4	1.876	3.8	9.538e-5	1.874	35.6	3.915e-5	2.016	44.3
	1/40	3.072e-5	1.937	7.6	2.493e-5	1.936	72.3	9.760e-6	2.004	74.3
	1/80	7.853e-6	1.968	15.0	6.376e-6	1.968	145.5	2.436e-6	2.002	97.6
	1/160	1.985e-6	1.984	33.8	1.612e-6	1.984	301.2	6.090e-7	2.000	208.0
1.5	1/10	4.188e-4	-	2.0	3.781e-4	-	20.0	9.655e-4	-	33.2
	1/20	1.143e-4	1.874	3.9	9.817e-5	1.945	46.8	1.819e-4	2.408	53.1
	1/40	2.988e-5	1.935	7.6	2.560e-5	1.939	101.3	4.365e-5	2.059	77.4
	1/80	7.639e-6	1.967	16.3	6.550e-6	1.967	210.6	1.081e-5	2.014	125.3
	1/160	1.932e-6	1.984	34.9	1.657e-6	1.983	417.6	2.695e-6	2.004	245.7
1.8	1/10	4.087e-4	-	2.0	2.652e-4	-	33.1	2.669e-3	-	44.7
	1/20	1.124e-4	1.862	3.8	6.821e-5	1.959	75.8	3.554e-4	2.909	65.9
	1/40	2.940e-5	1.935	7.6	1.754e-5	1.959	155.1	6.022e-5	2.561	101.8
	1/80	7.516e-6	1.968	16.9	4.469e-6	1.973	328.4	1.502e-5	2.004	159.2
	1/160	1.900e-6	1.984	35.6	1.129e-6	1.985	636.0	3.761e-6	1.997	243.4

which is a discrete analogue of the energy functional

$$E(u) = \int_\Omega \left(\frac{1}{4}(u^2 - 1)^2 - \frac{\varepsilon^2}{2} u \mathcal{L}_{\mathbf{x}}^{\alpha} u \right) d\mathbf{x}.$$

It is known that the FAC equation (1.1) satisfies the energy dissipation law [24], i.e., $E(u(t))$ decreases with respect to t. The numerical results show that the discrete maximum principle is preserved and the discrete energy decays monotonically.

Example 5.4 Now, we consider a 3D FAC equation (1.1) with $\Omega = (0, 1)^3$ and $\varepsilon = 0.01$. We choose a random initial data ranging from -0.05 to 0.05 uniformly generated on the $63 \times 63 \times 63$ mesh ($h = 1/64$). The time step is set to be $\tau = 0.1$ for all cases.

Similar to Example 5.3, as shown in Figures 4 and 5, the numerical solutions obtained by the DSETDRK2 scheme preserve the discrete maximum principle and the corresponding discrete energy decays monotonically. In addition, reducing the fractional order leads to thinner interface and slower phase coarsening process.

We remark that although numerical tests show that the DSETDRK2 scheme is energy stable, the theoretical analysis of the energy stability of the DSETDRK2 scheme is still missing.

5.3 FAC Equation with a Logarithmic Nonlinear Term

In this section, we test the performance of the DSETDRK2 scheme for solving the following FAC equation with a logarithmic nonlinear term

$$u_t = \varepsilon^2 \mathcal{L}_{\mathbf{x}}^{\alpha} u + f_0(u), \quad \mathbf{x} \in \Omega, \ t \in (0, T],$$

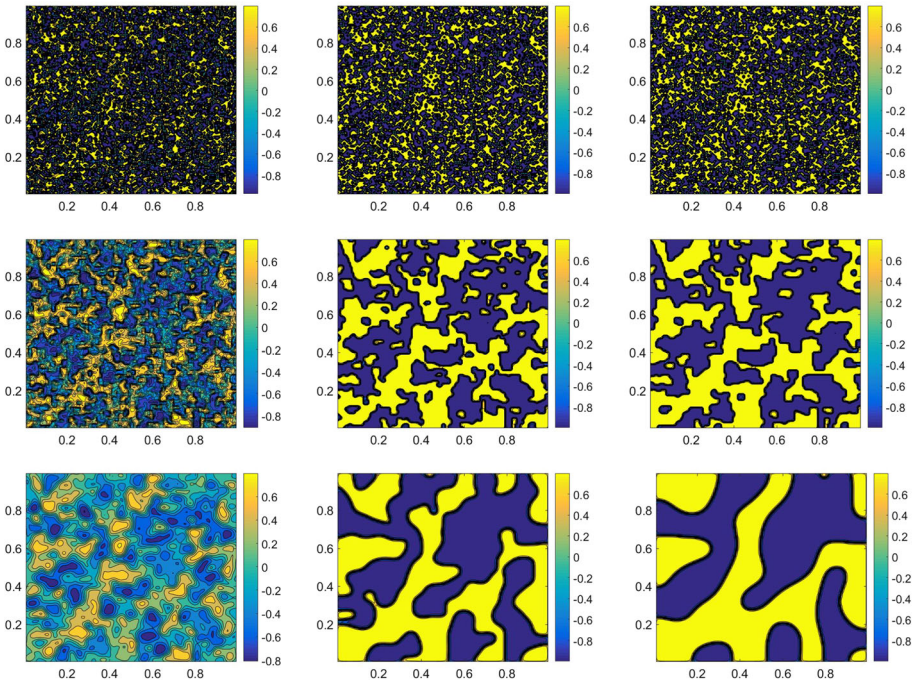

Fig. 2 Evolutions of the solutions in Example 5.3 with $\alpha = 1.2$ (top row), $\alpha = 1.6$ (middle row) and $\alpha = 1.8$ (bottom row). From left to right: $t = 5, 20, 100$

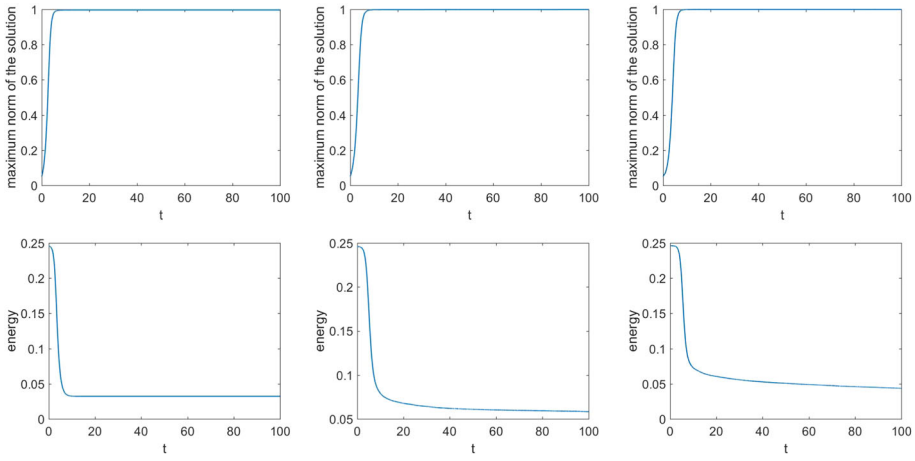

Fig. 3 Evolutions of the maximum norms (top row) and the energies (bottom row) of the numerical solutions in Example 5.3 From left to right: $\alpha = 1.2, 1.6, 1.8$

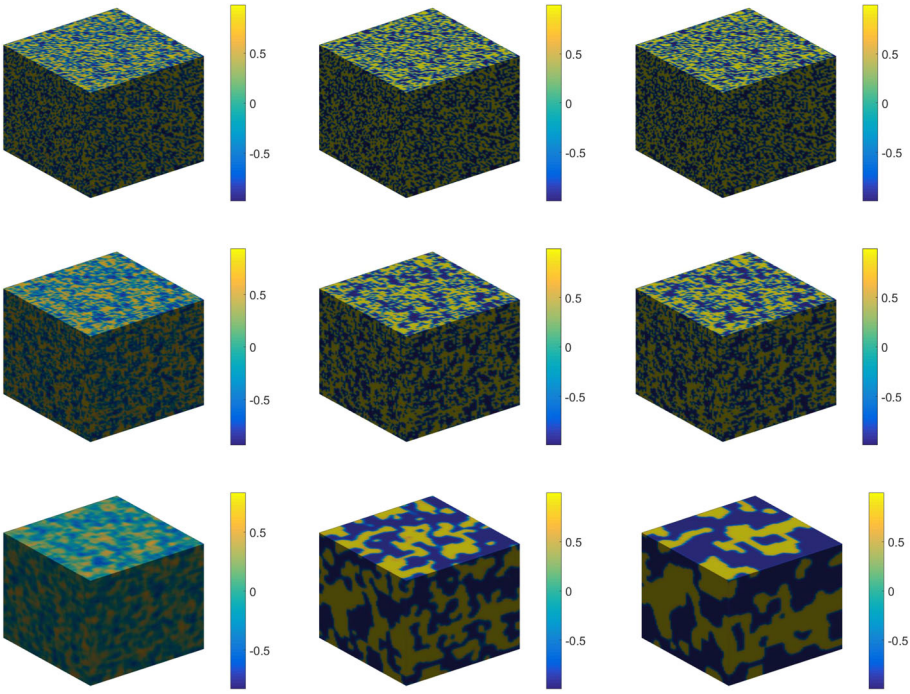

Fig. 4 Evolutions of the solutions in Example 5.4 with $\alpha = 1.2$ (top row), $\alpha = 1.6$ (middle row) and $\alpha = 1.8$ (bottom row). From left to right: $t = 5, 20, 100$

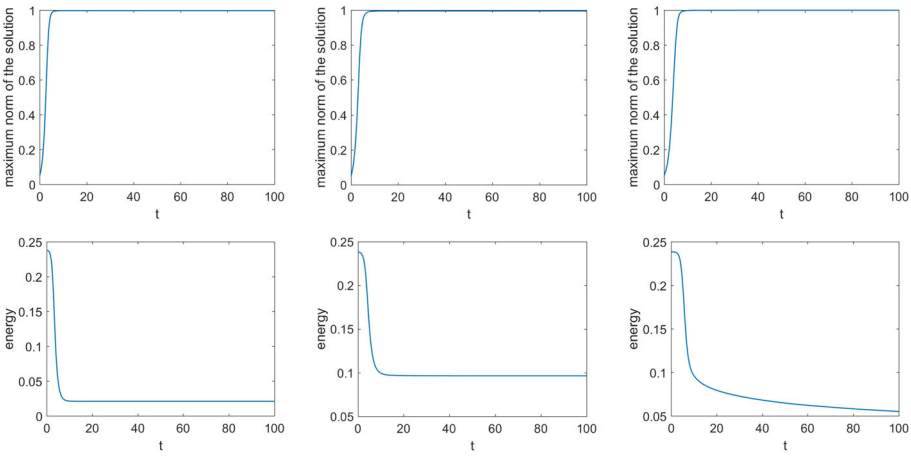

Fig. 5 Evolutions of the maximum norms (top row) and the energies (bottom row) of the numerical solutions in Example 5.4 From left to right: $\alpha = 1.2, 1.6, 1.8$

Table 5 Spatial convergence rates in the maximum- norm for Example 5.5

h	$\alpha = 1.2$		$\alpha = 1.5$		$\alpha = 1.8$	
	Error	Rate	Error	Rate	Error	Rate
1/8	4.0984e-3	-	6.9851e-3	-	9.4293e-3	-
1/16	1.1800e-3	1.7963	1.9599e-3	1.8335	2.5165e-3	1.9057
1/32	3.1252e-4	1.9168	5.1221e-4	1.9360	6.4243e-4	1.9698
1/64	2.4715e-4	0.3386	3.9140e-4	0.3881	1.6146e-4	1.9924
1/128	4.0394e-4	*	2.2303e-4	0.8114	7.1901e-5	1.1671

$$\begin{aligned} u(\mathbf{x}, 0) &= u_0(\mathbf{x}), && \mathbf{x} \in \overline{\Omega}, \\ u(\mathbf{x}, t) &= 0, && \mathbf{x} \text{ on } \partial\Omega,\ t \in [0, T], \end{aligned} \tag{5.4}$$

where $f_0(u) = \frac{\theta}{2} \ln \frac{1-u}{1+u} + \theta_c u$. The energy functional associated with (5.4) is given by

$$E(u) = \int_\Omega \left(F(u) - \frac{\varepsilon^2}{2} u \mathcal{L}_{\mathbf{x}}^{\alpha} u \right) d\mathbf{x}, \tag{5.5}$$

where $F(u)$ is the Flory-Huggins potential

$$F(u) = \frac{\theta}{2}[(1+u)\ln(1+u) + (1-u)\ln(1-u)] - \frac{\theta_c}{2}u^2.$$

Here θ and θ_c are two constants satisfying $0 < \theta < \theta_c$ and we have that $f_0(u) = -F'(u)$.

Denote by η the positive root of $f_0(\eta) = 0$. It is shown in [12] that equation (5.4) satisfies the maximum bound principle (MBP), which means that the absolute value of the entire solution is bounded by η if the absolute values of initial and boundary data are bounded by η. It is also shown in [12] that ETDRK2 scheme preserves the MBP when the stabilizing constant κ satisfies $\kappa \geq \frac{\theta}{1-\eta^2} - \theta_c$. We remark that the DSETDRK2 scheme for (5.4) also preserves the discrete MBP and is second order accurate. Since the analysis could be carried out in a quite similar way as done in [12], we omit the theoretical results of the DSETDRK2 scheme for solving (5.4) and only present numerical results.

In the rest of this paper, we set $\theta = 0.8$ and $\theta_c = 1.6$ in (5.4). The positive root of $f_0(\eta) = 0$ is approximately $\eta \approx 0.9575$ and the stabilizing constant is then chosen as $\kappa = 8.02$. To our best knowledge, whether the Crank-Nicolson scheme could preserve (or conditionally preserve) the discrete MBP of equation (5.4) has not been discussed yet, so we do not test the Crank-Nicolson scheme for (5.4) in this section.

Example 5.5 We consider a 2D equation (5.4) with $\varepsilon = 0.1$ on the domain $\Omega = (0, 1)^2$ up to the time $T = 0.5$. We set the smooth initial data as $u_0(x, y) = 0.2 \sin(3\pi x) \sin(3\pi y)$.

First, we test the convergence of the DSETDRK2 scheme with respect to the spatial size h by fixing $\tau = T$. We take the numerical solution of the equation (5.4) obtained by the DSETDRK2 scheme with $M = 1024$ as the benchmark solution. The numerical errors in the maximum-norm are presented in Table 5. We observe that the spatial convergence rates are approximately second order in most cases. As h becomes smaller ($h = \frac{1}{64}, \frac{1}{128}$), we find the order reduction, which may be caused by the possible boundary singularities of the fractional model (5.4).

Next, we test the convergence rates in time. We fix $h = 2^{-9}$ ($M = 511$) and compute the numerical solutions of the 2D equation (5.4) using the DSETDRK2 and ETDRK2 schemes

Table 6 Comparison of errors and CPU times of the DSETDRK2 and ETDRK2 schemes for Example 5.5

α	τ	DSETDRK2			ETDRK2		
		Error	Rate	CPU	Error	Rate	CPU
1.2	1/10	8.8860e-3	-	1.2	7.0688e-3	-	9.2
	1/20	2.9244e-3	1.6034	2.4	2.3203e-3	1.6072	18.4
	1/40	8.4388e-4	1.7930	4.9	6.6885e-4	1.7946	36.7
	1/80	2.2691e-4	1.8949	9.6	1.7978e-4	1.8955	69.7
	1/160	5.8833e-5	1.9474	19.2	4.6605e-5	1.9477	143.6
1.5	1/10	6.2444e-3	-	1.2	3.0030e-3	-	10.7
	1/20	2.0513e-3	1.6060	2.5	9.8253e-4	1.6118	25.0
	1/40	5.9151e-4	1.7941	4.8	2.8288e-4	1.7963	53.6
	1/80	1.5902e-4	1.8952	9.9	7.6002e-5	1.8961	110.7
	1/160	4.1227e-5	1.9475	23.2	1.9697e-5	1.9481	223.7
1.8	1/10	2.3305e-3	-	1.2	3.3334e-3	-	16.4
	1/20	7.6317e-4	1.6106	2.4	1.0753e-3	1.6323	39.6
	1/40	2.1980e-4	1.7958	4.8	3.0795e-4	1.8040	82.5
	1/80	5.9067e-5	1.8958	9.7	8.2600e-5	1.8985	166.7
	1/160	1.5312e-5	1.9477	21.5	2.1406e-5	1.9481	337.1

with various time step sizes. The approximation solution obtained by using the ETDRK2 scheme with $\tau = 10^{-4}$ is taken as the benchmark solution for computing errors. The maximum norms of the numerical errors and corresponding CPU times are reported in Table 6 where the second order accuracy of the DSETDRK2 and ETDRK2 schemes are seen obviously. For a given level of accuracy, we find that the time consumption of the DSETDRK2 scheme is much less than the ETDRK2 scheme.

Example 5.6 Let $\Omega = (0, 1)^2$ and $\varepsilon = 0.01$. Consider the 2D equation (5.4) with a random initial data ranging from -0.9 to 0.9 uniformly generated on the 127×127 mesh ($h = 1/128$). The numerical solutions are computed by the DSETDRK2 scheme with time step $\tau = 0.1$ for all cases.

In Figure 6, we display the evolutions of the solutions of equation (5.4) with $\alpha = 1.3, 1.6, 1.9$ at times $t = 5, 20, 200$, respectively. Again, we observe that equation (5.4) with a smaller fractional order has a thinner interface and slower phase coarsening process. Figure 7 shows the evolutions of the corresponding maximum norms and the energies of the numerical solutions, respectively. The dash horizonal line shows the theoretical upper bound η of the numerical solutions. The discrete energy is defined by

$$E_h(U) = h^2 \left(\sum_{j=1}^{N} \left(\frac{\theta}{2} [(1+U_j)\ln(1+U_j) + (1-U_j)\ln(1-U_j)] - \frac{\theta_c}{2} U_j^2 \right) - \frac{1}{2} U^T \tilde{A} U \right),$$

which is a discrete analogue of the energy functional $E(u)$ (5.5). The numerical results show that the discrete MBP is preserved and the discrete energy decays monotonically.

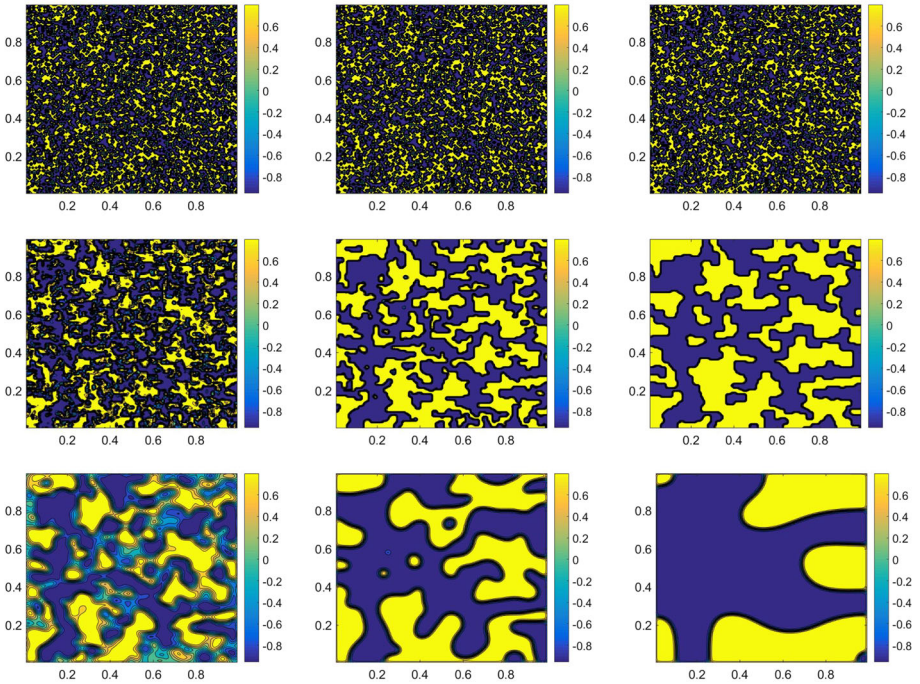

Fig. 6 Evolutions of the solutions in Example 5.6 with $\alpha = 1.3$ (top row), $\alpha = 1.6$ (middle row) and $\alpha = 1.9$ (bottom row). From left to right: $t = 5, 20, 200$

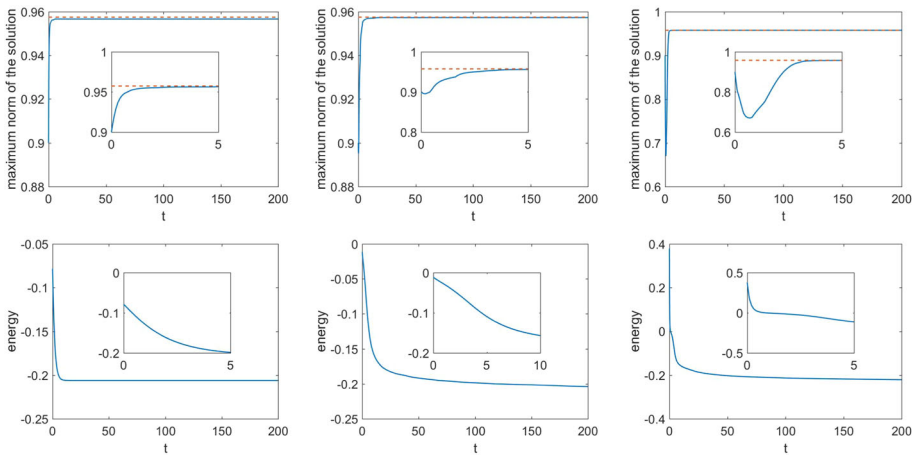

Fig. 7 Evolutions of the maximum norms (top row) and the energies (bottom row) of the numerical solutions in Example 5.6 From left to right: $\alpha = 1.3, 1.6, 1.9$

6 Conclusion

In this paper, we are concerned with efficient numerical methods for solving multidimensional space FAC equations. Based on a natural dimensional splitting of the spatial discretization matrix, we propose a new ETD-type integrator for multidimensional FAC equations. We prove that the proposed DSETDRK2 scheme preserves the maximum principle in the discrete level. We also prove that the DSETDRK2 scheme is second order accurate and give maximum-norm error estimate. The implementation of the scheme requires the computation of linear systems and matrix exponential associated with only 1D discretization matrix which possesses Toeplitz structure. Fast algorithms for Toeplitz inverse and Toeplitz matrix exponential are used in the implementation of the scheme. Numerical experiments support our theoretical analysis and demonstrate efficiency of the DSETDRK2 scheme.

Acknowledgements The authors are very grateful to the referees for their constructive comments and valuable suggestions, which greatly improved the quality of this paper. The first author was partially supported by the National Natural Science Foundation of China (Grant No.11971085), the Program of Chongqing Innovation Research Group Project in University (No. CXQT19018), and the Scientific and Technological Research Program of Chongqing Municipal Education Commission (No. KJQN202000543).

References

1. Allen, S.M., Cahn, J.W.: A microscopic theory for antiphase boundary motion and its application to antiphase domain coarsening. Acta Metall. **27**, 1085–1095 (1979)
2. Al-Mohy, A.H., Higham, N.J.: Computing the action of the matrix exponential, with an application to exponential integrators. SIAM J. Sci. Comput. **33**, 488–511 (2011)
3. Bueno-Orovio, A., Kay, D., Burrage, K.: Fourier spectral methods for fractional-in-space reaction-diffusion equations. BIT **54**, 937–954 (2014)
4. Burrage, K., Hale, N., Kay, D.: An efficient implicit FEM scheme for fractional-in-space reaction-diffusion equations. SIAM J. Sci. Comput. **34**, A2145–A2172 (2012)
5. Chan, R., Jin, X.: An Introduction to Iterative Toeplitz Solvers. SIAM, Philadelphia (2007)
6. Chan, R., Ng, M.: Conjugate gradient methods for Toeplitz system. SIAM Rev. **38**, 427–482 (1992)
7. Cox, S.M., Matthews, P.C.: Exponential time differencing for stiff systems. J. Comput. Phys. **176**, 430–455 (2002)
8. Du, Q., Zhu, W.: Stability analysis and application of the exponential time differencing schemes. J. Comput. Math. **22**, 200–209 (2004)
9. Du, Q., Zhu, W.: Analysis and applications of the exponential time differencing schemes and their contour integration modifications. BIT **45**, 307–328 (2005)
10. Du, Q., Yang, J.: Asymptotic compatible Fourier spectral approximations of nonlocal Allen-Cahn equations. SIAM J. Numer. Anal. **54**, 1899–1919 (2016)
11. Du, Q., Ju, L., Li, X., Qiao, Z.: Maximum principle preserving exponential time differencing schemes for the nonlocal Allen-Cahn equation. SIAM J. Numer. Anal. **57**, 875–898 (2019)
12. Du, Q., Ju, L., Li, X., Qiao, Z.: Maximum bound principles for a class of semilinear parabolic equations and exponential time differencing schemes. SIAM Rev. accepted, 2020
13. Feng, X., Prohl, A.: Numerical analysis of the Allen-Cahn equation and approximation for mean curvature flows. Numer. Math. **94**, 33–65 (2003)
14. Feng, X., Song, H., Tang, T., Yang, J.: Nonlinear stability of the implicit-explicit methods for the Allen-Cahn equation. Inverse Probl. Imaging **7**, 679–695 (2013)
15. Feng, X., Tang, T., Yang, J.: Stabilized Crank-Nicolson/Adams-Bashforth schemes for phase field models. East Asian J. Appl. Math. **3**, 59–80 (2013)
16. Gaudreault, S., Rainwater, G., Tokman, M.: KIOPS: A fast adaptive Krylov subspace solver for exponential integrators. J. Comput. Phys. **372**, 236–255 (2018)
17. Gohberg, I., Olshevsky, V.: Circulants, displacements and decompositions of matrices. Integral Equ. Oper. Theory. **15**, 730–743 (1992)
18. He, D., Pan, K., Hu, H.: A spatial fourth-order maximum principle preserving operator splitting scheme for the multi-dimensional fractional Allen-Cahn equation. Appl. Numer. Math. **151**, 44–63 (2020)

19. Higham, N.J.: Functions of Matrices: Theory and Computation. SIAM, Philadelphia (2008)
20. Higham, N.J., Al-Mohy, A.H.: Computing matrix functions. Acta Numerica **19**, 159–208 (2010)
21. Hochbruck, M., Lubich, C.: On Krylov subspace approximations to the matrix exponential operator. SIAM J. Numer. Anal. **34**, 1911–1925 (1997)
22. Hochbruck, M., Ostermann, A.: Exponential integrators. Acta Numerica **19**, 209–286 (2010)
23. Horn, R., Johnson, C.: Topics in Matrix Analysis. Cambridge University Press, Cambridge (1991)
24. Hou, T., Tang, T., Yang, J.: Numerical analysis of fully discretized Crank-Nicolson scheme for fractional-in-space Allen-Cahn equations. J. Sci. Comput. **72**, 1214–1231 (2017)
25. Jin, X.: Developments and Applications of Block Toeplitz Iterative Solvers. Kluwer, Dordrecht (2002)
26. Kassam, A.K., Trefethen, L.N.: Fourth-order time-stepping for stiff PDEs. SIAM J. Sci. Comput. **26**, 1214–1233 (2005)
27. Lee, S., Liu, X., Sun, H.-W: Fast exponential time integration scheme for option pricing with jumps. Numer. Linear Algebra Appl. 19, 87–101 (2012)
28. Shift-invert Arnoldi approximation to the Toeplitz matrix exponential: Lee, S., Pang, H., Sun, H.-W. SIAM J. Sci. Comput. **32**, 774–792 (2010)
29. Liao, H.-L., Tang, T., Zhou, T.: On energy stable, maximum-principle preserving, second-order BDF scheme with variable steps for the Allen-Cahn equation. SIAM J. Numer. Anal. **58**, 2294–2314 (2020)
30. Minchev, B.V., Wright, W.M.: A review of exponential integrators for first order semi-linear problems. preprint, NTNU Trondheim, Trondheim, Norway, 2005
31. Niesen, J., Wright, W.M.: Algorithm 919: A Krylov subspace algorithm for evaluating the φ-functions appearing in exponential integrators. ACM Trans. Math. Softw. **38**, 22 (2012)
32. Pang, H., Sun, H.-W.: Shift-invert Lanczos method for the symmetric positive semidefinite Toeplitz matrix exponential. Numer. Linear Algebra Appl. **18**, 603–614 (2011)
33. Pang, H., Sun, H.-W.: Fast exponential time integration for pricing options in stochastic volatility jump diffusion models. East Asian J. Appl. Math. **4**, 53–68 (2014)
34. Podlubny, I.: Fractional Differential Equations. Academic Press, New York (1999)
35. Schmelzer, T., Trefethen, L.N.: Evaluating matrix functions for exponential integrators via Carathéodory-Fejér approximation and contour integrals. Electron. Trans. Numer. Anal. **29**, 1–18 (2007)
36. Shen, J., Tang, T., Yang, J.: On the maximum principle preserving schemes for the generalized Allen-Cahn equation. Commun. Math. Sci. **14**, 1517–1534 (2016)
37. Shen, J., Yang, X.: Numerical approximations of Allen-Cahn and Cahn-Hilliard equations. Discret. Contin. Dyn. Syst. **28**, 1669–1691 (2010)
38. Tang, T., Yang, J.: Implicit-explicit scheme for the Allen-Cahn equation preserves the maximum principle. J. Comput. Math. **34**, 471–481 (2016)
39. Tian, W., Zhou, H., Deng, W.: A class of second order difference approximation for solving space fractional diffusion equations. Math. Comput. **84**, 1703–1727 (2015)
40. Yang, X.: Error analysis of stabilized semi-implicit method of Allen-Cahn equation. Discrete Contin. Dyn. Syst. Ser. B **11**, 1057–1070 (2009)
41. Zhang, J., Du, Q.: Numerical studies of discrete approximations to the Allen-Cahn equation in the sharp interface limit. SIAM J. Sci. Comput. **31**, 3042–3063 (2009)
42. Zhang, L., Sun, H., Pang, H.: Fast numerical solution for fractional diffusion equations by exponential quadrature rule. J. Comput. Phys. **299**, 130–143 (2015)
43. Zhang, L., Sun, H.: Numerical solution for multi-dimensional Riesz fractional nonlinear reaction-diffusion equation by exponential Runge-Kutta method. J. Appl. Math. Comput. **62**, 449–472 (2020)
44. Zhang, L., Zhang, Q., Sun, H.: Exponential Runge-Kutta method for two-dimensional nonlinear fractional complex Ginzburg-Landau equations. J. Sci. Comput. **83**, 59 (2020)