A Fast Compact Finite Difference Method for Fractional Cattaneo Equation Based on Caputo–Fabrizio Derivative

Haili Qiao, Zhengguang Liu, and Aijie Cheng

1School of Mathematics, Shandong University, Jinan, Shandong, China
2School of Mathematics and Statistics, Shandong Normal University, Jinan, Shandong, China

Correspondence should be addressed to Aijie Cheng; aijie@sdu.edu.cn

Received 15 October 2019; Accepted 16 January 2020; Published 19 March 2020

Copyright © 2020 Haili Qiao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The Cattaneo equations with Caputo–Fabrizio fractional derivative are investigated. A compact finite difference scheme of Crank–Nicolson type is presented and analyzed, which is proved to have temporal accuracy of second order and spatial accuracy of fourth order. Since this derivative is defined with an integral over the whole passed time, conventional direct solvers generally take computational complexity of $O(MN^2)$ and require memory of $O(MN)$, with $M$ and $N$ the number of space steps and time steps, respectively. We develop a fast evaluation procedure for the Caputo–Fabrizio fractional derivative, by which the computational cost is reduced to $O(MN)$ operations; meanwhile, only $O(M)$ memory is required. In the end, several numerical experiments are carried out to verify the theoretical results and show the applicability of the fast compact difference procedure.

1. Introduction

Fractional diffusion equations have become a strong and forceful tool to describe the phenomenon of anomalous diffusion, and more research works have been obtained in the last decades [1–6]. However, since the fractional derivative is nonlocal and has weak singularity, it is impossible to solve fractional diffusion equations analytically in most cases. Instead, seeking numerical solutions is becoming an indispensable tool for research work about fractional equations.

Different from the traditional derivative of the integer order, the fractional derivative depends on the total information in the correlative region, and this is the so-called nonlocal properties. Just because of this, it consumes computational time extremely to solve fractional equations. We hope to develop effective numerical schemes, which not only have better stability and higher accuracy but also require less storage memory and save computational cost.

About stability and convergence analysis of the numerical schemes for fractional equations, the readers can refers to [7, 8] for spatial fractional order equation, [9–18] for temporal fractional diffusion equations, and [19–22] for space-time-fractional equations. About the complexity, i.e., storage requirement and computation cost of an algorithm, researchers devote themselves to reduce storage requirement and computational time by analyzing the particular structure of coefficient matrices arising from the discretization system or reutilizing the intermediate data skillfully. We call these algorithms fast methods, including fast finite difference methods [23–28], fast finite element methods [29], and fast collocation methods [30, 31]. A fast method for Caputo fractional derivatives is proposed [32, 33]. Lu et al. [34] presented a fast method of approximate inversion for triangular Toeplitz tridiagonal block matrix, which is successfully applied to the fractional diffusion equations. Comparatively, there is less research work about the fast method for temporal fractional derivative than that for spatial fractional operators.
A time-fractional Cattaneo equation is considered with the following form:

\[
\frac{\partial u(x, t)}{\partial t} + \frac{\partial^\alpha u(x, t)}{\partial t^\alpha} - \frac{\partial^2 u(x, t)}{\partial x^2} + f(x, t), \quad (x, t) \in \Omega \times (0, T],
\]

\[
u(x, 0) = \phi(x), \quad \frac{\partial u(x, t)}{\partial t} = \psi(x), \quad x \in \Omega,
\]

\[
u(x, t) = 0, \quad x \in \partial\Omega,
\]

where \(1 < \alpha < 2; \Omega = (a, b)\) for one-dimensional case, and \(\Omega = (a, b) \times (c, d)\) for two-dimensional case; \(f(x, t)\) is the source term; \(\phi(x)\) and \(\psi(x)\) are the prescribed functions for initial conditions, and \(\partial^\alpha u/\partial t^\alpha\) is a new Caputo fractional derivative without singular kernel, which is defined in the next section.

Our purpose is to establish a fast finite difference scheme of high order for this equation. We will extract the recursive relation between the \((k + 1)\) time step and the \(k\) time step of the finite difference solution. The computational work is significantly reduced from \(O(MN^2)\) to \(O(MN)\), and the memory requirement from \(O(MN)\) to \(O(M)\), where \(M\) and \(N\) are the total numbers of points for space and time steps, respectively. For improving the accuracy, a compact finite difference scheme is established. Theoretical analysis shows that the fast compact difference scheme has spatial accuracy of fourth order and temporal accuracy of second order. Several numerical experiments are implemented, which verify the effectiveness, applicability, and convergence rate of the proposed scheme.

This paper is organized as follows: some definitions and notations are prepared in Section 2. The compact finite difference scheme is described and then the stability and convergence rates are rigorously analyzed for the scheme in Section 3. The compact finite difference scheme is extended to the case of two space dimensions in Section 4. Fast evaluation and efficient storage are established skillfully in Section 5. Some numerical experiments are carried out in Section 6. In the end, we summarize the major contribution of this paper in Section 7.

2. Some Notations and Definitions

We provide some definitions which will be used in the following analysis.

First, let us recall the usual Caputo fractional derivative of order \(\alpha\) with respect to time variable \(t\), which is given by

\[
^C_0 D_t^\alpha u(x, t) = \frac{1}{\Gamma(n - \alpha)} \sum_{a}^t (t - s)^{n - \alpha - 1} u^{(n)}(x, s) ds,
\]

where \(n - 1 < \alpha < n\).

By replacing the kernel function \((t - s)^{-\alpha}\) with the exponential function \(\exp(-\alpha(t - s/(1 - \alpha)))\) and \(1/(1/(1 - \alpha))\) with \(M(\alpha)/1 - \alpha\), Caputo and Fabrizio [35] proposed the following definition of fractional time derivative.

\[
\text{Definition 1 (see [35]). Let } u(t, t) \in H^1(a, b), \text{ then the new Caputo derivative of the fractional order is defined as}
\]

\[
D_t^\alpha u(x, t) = \frac{M(\alpha)}{1 - \alpha} \int_a^t u'(x, s) \exp\left(-\alpha \frac{t - s}{1 - \alpha}\right) ds,
\]

where \(M(\alpha)\) is a normalization function satisfying \(M(0) = M(1) = 1\). When the function \(u\) does not belong to \(H^1(a, b)\), this derivative can be reformulated as

\[
D_t^\alpha u(x, t) = \frac{\alpha M(\alpha)}{1 - \alpha} \int_a^t (u(x, t) - u(x, s)) \exp\left(-\alpha \frac{t - s}{1 - \alpha}\right) ds.
\]

\[
\text{Definition 2 (see [36]). The above new Caputo derivative of order } 0 < \alpha < 1 \text{ can also be reformulated as}
\]

\[
\text{CF } D_t^\alpha u(x, t) = \frac{1}{1 - \alpha} \sum_a^t u'(x, s) \exp\left(-\alpha \frac{t - s}{1 - \alpha}\right) ds.
\]

\[
\text{Definition 3 (see [35, 36]). Let } u(t, t) \in H^1(a, b), \text{ if } n \geq 1, \text{ and } \alpha \in [0, 1], \text{ the fractional time derivative } 0 \text{CF } D_t^\alpha u(x, t) \text{ of order } (n + \alpha) \text{ is defined by}
\]

\[
0 \text{CF } D_t^\alpha u(x, t) = 0 \text{CF } D_t^{n+\alpha} u(x, t),
\]

\[
\text{Particularly, for } 1 < \alpha < 2, \text{ we have}
\]

\[
0 \text{CF } D_t^\alpha u(x, t) = \frac{M(\alpha)}{2 - \alpha} \sum_a^t u''(x, s) \exp\left((1 - \alpha) \frac{t - s}{2 - \alpha}\right) ds.
\]

Remark 1. The Caputo–Fabrizio (CF) operator was proposed with a nonsingular kernel for describing material heterogeneities that do not exhibit power-law behavior [35].

Remark 2. An open discussion is ongoing about the mathematical construction of the CF operator. Ortigueira and Tenreiro Machado [37] indicated that the CF fractional derivative is neither a fractional operator nor a derivative operator, the authors of [38, 39] showed that this operator cannot describe dynamic memory, and Giusti [40] indicated that this operator can be expressed as an infinite linear combination of Riemann–Liouville integrals with integer
powers. As responses to these criticisms, Atangana and Gómez-Aguilar [41] pointed out the need to account for a fractional calculus approach without an imposed index law and with nonsingular kernels. Furthermore, Hristov [42] indicated that the CF operator is not applicable for explaining the physical examples in [37, 40]; instead, he suggested that the CF operator can be used for the analysis of materials that do not follow a power-law behavior. The authors of [43] believe that models with CF operators produce a better representation of physical behaviors than do integer-order models, providing a way to model the intermediate (between elliptic and parabolic or between parabolic and hyperbolic) behaviors.

To obtain the accuracy of the fourth order in spatial directions, the following lemma is necessary.

Lemma 1 (see [44]). Denote \( \delta(s) = (1 - s)^2 [5 - 3(1 - s)^2] \). If \( f(x) \in C^2[a, b], h = (b - a)/M, x_i = a + ih (0 \leq i \leq M) \), then, it holds that

\[
\frac{1}{12} [f^{(2)}(x_{i-1}) + 10 f^{(2)}(x_i) + f^{(2)}(x_{i+1})] = \frac{1}{h^2} \left[ f(x_{i+1}) - 2 f(x_i) + f(x_{i-1}) \right]
\]

\[
+ \frac{h^4}{360} \int_0^1 \left[ f^{(6)}(x_i - s h) + f^{(6)}(x_i + s h) \right] (x_i + s h) \cdot \delta(s) ds, \quad 1 \leq i \leq M - 1.
\]

3. Compact Finite Difference Scheme for One-Dimensional Fractional Cattaneo Equation

In order to construct the finite difference schemes, the interval \([a, b]\) is divided into subintervals with \( x_i = a + ih, (0 \leq i \leq M) \), and \([0, T]\) is discretized with \( t_k = k\Delta t, (0 \leq k \leq N) \), where \( h = (b - a)/M \) and \( \Delta t = T/N \) are the spatial grid size and temporal step size, respectively. Denote \( \Omega_h = \{ x_i: 0 \leq i \leq M \}, \Omega_{\Delta t} = \{ t_k: 0 \leq k \leq N \} \), then \( \Omega_h \times \Omega_{\Delta t} \) becomes a discretization of the practical computational domain \([a, b] \times [0, T]\). The values of the function \( u \) at the grid points are denoted as \( u^h = u(x_i, t_k) \), and the approximate solution at the point \((x_j, t_{j+k})\) is denoted as \( u^h_j \).

Denote \( V_h = \{ v = (v_0, v_1, \ldots, v_M) \} \). We also introduce the following notations for any mesh function \( v \in V_h \):

\[
\delta^1_x v_j = \frac{v_{j+1} - v_{j-1}}{h},
\]

\[
\delta^2_x v_j = \frac{v_{j+1} - 2v_j + v_{j-1}}{h^2}, \quad 1 \leq j \leq M - 1,
\]

and define the average operator

\[
\mathcal{A} v_j = \begin{cases} 
\frac{1}{12} (v_{j+1} + 10v_j + v_{j-1}), & 1 \leq j \leq M - 1, \\
v_j, & j = 0, M.
\end{cases}
\]

It is easy to see that

\[
\mathcal{A} v_j = \left( I + \frac{h^2}{12} \right) v_j,
\]

where \( I \) is the identical operator. We also denote \( \mathcal{A} v = (\mathcal{A} v_1, \mathcal{A} v_2, \ldots, \mathcal{A} v_M) \) for vector \( v = (v_1, v_2, \ldots, v_M) \), and \( \mathcal{A} (u, v) = (\mathcal{A} u, v) \).

For any two grid functions \( u, v \in V_h = \{ v | v \in V_h, v_0 = v_M = 0 \} \), the discrete inner products and norms are defined as

\[
\langle u, v \rangle = h \sum_{j=1}^{M-1} u_j v_j,
\]

\[
\| u \|^2_2 = (u, u),
\]

\[
\| u \|_{\infty} = \max_{1 \leq j \leq M-1} |u_j|.
\]

By summation by parts, it is easy to see that

\[
\delta^1_x u, v = -(\delta^1_x u, \delta^1_x v) = -\langle u, v \rangle = \langle u, \delta^1_x v \rangle.
\]

For the average operator \( \mathcal{A} \), define

\[
\mathcal{A} (v, v) \triangleq (\mathcal{A} v, v) = \| v \|^2_2.
\]

Additionally, let \( V_{\Delta t} = \{ v = (v^0, v^1, \ldots, v^N) \} \) be the space of grid function defined on \( \Omega_{\Delta t} \). For any function \( v \in V_{\Delta t} \), a difference operator is introduced as follows:

\[
\delta^k_x v = v^{k+1} - v^{k-1} \Delta t.
\]

3.1. Compact Finite Difference Scheme. We will consider the time-fractional Cattaneo equation equipped with the Caputo–Fabrizio derivative. Vivas-Cruz et al. [43] gave the theoretical analysis of a model of fluid flow in a reservoir with the Caputo–Fabrizio operator. They proved that this model cannot be used to describe nonlocal processes since it can be represented as an equivalent differential equation with a finite number of integer-order derivatives.

The finite difference methods usually lead to stencils through the whole history passed by the solution which consume too much computational work. In this paper, we will establish a high-order finite difference scheme and propose a procedure to reduce the computational cost. In [43], the authors proposed a recurrence formula of discretized CF operator and obtained an algorithm which can be considered a stencil with a one-step expression without the need of integrals over the whole history. It seems that the procedure in our paper and the algorithm in [43] are different in approach but equally satisfactory in result.

For obtaining effective approximation with high order, we introduce the numerical discretization for the fractional
the Cattaneo equation by means of compact finite difference methods.

At the node \((x_i, t_{k+1/2})\), the differential equation takes the following form:

\[
\frac{\partial u}{\partial t}(x_i, t_{k+1/2}) + \frac{\partial^2 u}{\partial x^2}(x_i, t_{k+1/2}) = \phi_i + f(x_i, t_{k+1/2}), \quad 1 \leq i \leq M - 1, 1 \leq k \leq N - 1.
\]  

(16)

The approximation of the fractional derivative is given by [45]

\[
\begin{align*}
\mathcal{D}_0^{\alpha} \Delta_t^a u(x_i, t_{k+1/2}) &= \frac{1}{(\alpha - 1) \Delta t} \left[ M_0^c \delta_i u_{k+1}^i - \sum_{n=1}^{k} (M_{k-n} u_{n+1}) \delta_i u_{j}^i - M_k \psi_i \right] + R_i^{k+1/2},
\end{align*}
\]  

(17)

with truncation error \(R_i^{k+1/2} = O(\Delta t^2)\), and

\[
M_n = \exp \left[ \frac{1 - \alpha}{2} \Delta t n \right] - \exp \left[ \frac{1 - \alpha}{2} \Delta t (n + 1) \right].
\]  

(18)

Furthermore, by Lemma 1 and equation (15), the space and time derivative are approximated by

\[
\begin{align*}
\frac{\partial^2 u}{\partial x^2}(x_i, t_{k+1/2}) &= \frac{1}{2} \left( \frac{\partial^2 u_{k+1}}{\partial x^2} + \frac{\partial^2 u_{k}}{\partial x^2} \right) + O(\Delta t^2 + h^4),
\end{align*}
\]  

(19)

\[
\begin{align*}
\frac{\partial u}{\partial t}(x_i, t_{k+1/2}) &= \delta_i u_{k+1}^i + O(\Delta t^2).
\end{align*}
\]  

(20)

Substituting (17) and (19)–(20) into (16), we get

\[
\begin{align*}
\mathcal{D}_0^{\alpha} \delta_i u_{k+1}^i + \frac{1}{(\alpha - 1) \Delta t} \mathcal{D}_0^{\alpha} \left[ M_0^c \delta_i u_{k+1}^i - \sum_{n=1}^{k} (M_{k-n} u_{n+1}) \delta_i u_{j}^i - M_k \psi_i \right]
\end{align*}
\]  

\[
= \frac{1}{2} \left( \delta_i u_{k+1}^i + \delta_i u_{k}^i \right) + \mathcal{D}_0^{\alpha} f_i^{k+1/2} + R_i^{k+1/2},
\]  

\[1 \leq i \leq M - 1, 0 \leq k \leq N - 1,
\]  

(21)

and there exists a constant \(C\), depending on the function \(u\) and its derivatives such that

\[
R_i^{k+1/2} \leq C(\Delta t^2 + h^4).
\]  

(22)

By the initial and boundary value conditions, we have

\[
\begin{align*}
u_i^0 &= \phi_i, \quad 1 \leq i \leq M - 1, \\
u_0^k &= u_M^k = 0, \quad 0 \leq k \leq N.
\end{align*}
\]  

(23)

A compact finite difference scheme can be established by omitting the truncation term \(R_i^{k+1/2}\) and replacing the exact solution \(u_i^k\) in equation (21) with numerical solution \(u_i^k\):

\[
\begin{align*}
(\alpha - 1) \Delta t \mathcal{D}_0^{\alpha} \delta_i u_{k+1}^i + M_0^c \delta_i u_{k+1}^i - \frac{\alpha - 1}{2} \left( \delta_i u_{k+1}^i, u_{k+1}^i \right)
\end{align*}
\]  

\[
+ \frac{\alpha - 1}{2} \left( \delta_i u_{k}^i, U^k \right)
\]  

\[
= \frac{\alpha - 1}{2} \left( \delta_i u_{k}^i, U^k \right) - \frac{\alpha - 1}{2} \left( \delta_i u_{k}^i, U^k \right)
\]  

\[
+ \sum_{n=1}^{k} (M_{k-n} - M_{k-n+1}) \left( \mathcal{D}_0^{\alpha} \delta_i u_{n}^i, \delta_i u_{k+1}^i \right)
\]  

\[
+ M_k \left( \mathcal{D}_0^{\alpha} \delta_i u_{k+1}^i \right) + (\alpha - 1) \Delta t (\mathcal{D}_0^{\alpha} f_i^{k+1/2}, \delta_i u_{k+1}^i).
\]  

(24)

3.2. Stability Analysis and Optimal Error Estimates

3.2.1. Stability Analysis. The following Lemma about \(M_n\) is useful for the analysis of stability.

Lemma 2 (see [45]). For the definition of \(M_n\), \(M_n > 0\) and \(M_n < M_n, \forall n \leq k, \) are held.

Multiplying \(\mathcal{D}_0^{\alpha} \delta_i u_{k+1}^i\) on both sides of equation (24) and summing up with respect to \(i\) from 1 to \(M - 1\), the following equation is obtained:

\[
(\alpha - 1) \Delta t \left\| \delta_i U_{k+1}^i \right\|^2_{A} + M_0^c \left\| \delta_i U_{k+1}^i \right\|^2_{A} - \frac{\alpha - 1}{2} \left( \delta_i U_{k+1}^i, U_{k+1}^i \right)
\]  

\[
+ \frac{\alpha - 1}{2} \left( \delta_i U_{k}^i, U^k \right)
\]  

\[
= \frac{\alpha - 1}{2} \left( \delta_i U_{k}^i, U^k \right) - \frac{\alpha - 1}{2} \left( \delta_i U_{k}^i, U^k \right)
\]  

\[
+ \sum_{n=1}^{k} (M_{k-n} - M_{k-n+1}) \left( \mathcal{D}_0^{\alpha} \delta_i u_{n}^i, \delta_i u_{k+1}^i \right)
\]  

\[
+ M_k \left( \mathcal{D}_0^{\alpha} \delta_i u_{k+1}^i \right) + (\alpha - 1) \Delta t (\mathcal{D}_0^{\alpha} f_i^{k+1/2}, \delta_i U_{k+1}^i).
\]  

(25)

Observing equation (13), we have

\[
\left( \delta_i U_{k+1}^i, U_{k+1}^i \right) = -\left( \delta_i U_{k+1}^i, \delta_i U_{k+1}^i \right) = \left( \delta_i U_{k+1}^i \right)^2 \leq 0,
\]  

\[
\left( \delta_i U_{k}^i, U^k \right) = -\left( \delta_i U_{k}^i, \delta_i U_{k}^i \right) = \left( \delta_i U_{k}^i \right)^2 \leq 0,
\]  

\[
\left( \delta_i U_{k+1}^i, U_{k+1}^i \right) = -\left( \delta_i U_{k+1}^i, \delta_i U_{k+1}^i \right) = \left( \delta_i U_{k+1}^i \right)^2 \leq 0,
\]  

\[
\left( \delta_i U_{k}^i, U^k \right) = -\left( \delta_i U_{k}^i, \delta_i U_{k}^i \right) = \left( \delta_i U_{k}^i \right)^2 \leq 0.
\]  

(26)

By the triangle inequality and Lemma 2, we obtain
\[ \sum_{n=1}^{k} (M_{k-n} - M_{k-n+1}) (\alpha \delta U^n, \delta U^{k+1}) + M_{k} (\alpha \Psi, \delta U^{k+1}) \]
\[ \leq \sum_{n=1}^{k} \frac{1}{2} (M_{k-n} - M_{k-n+1}) \left( \| \delta U^n \|_A^2 + \| \delta U^{k+1} \|_A^2 \right) \]
\[ + \frac{1}{2} M_k \left( \| \Psi \|_A^2 + \| \delta U^{k+1} \|_A^2 \right) \]
\[ = \sum_{n=1}^{k} \frac{1}{2} (M_{k-n} - M_{k-n+1}) \| \delta U^n \|_A^2 + \frac{1}{2} (M_0 - M_k) \]
\[ \| \delta U^{k+1} \|_A^2 + \frac{1}{2} M_k \| \Psi \|_A^2. \]

Combining equation (25) with (26)~(27), we get
\[ \sum_{n=1}^{k+1} M_{k-n} \| \delta U^n \|_A^2 + (\alpha - 1) \| \delta U^{k+1} \|_A^2 \]
\[ \leq \sum_{n=1}^{k} M_{k-n} \| \delta U^n \|_A^2 + (\alpha - 1) \| \delta U^{k+1} \|_A^2 + M_k \| \Psi \|_A^2 \]
\[ + (\alpha - 1) \Delta t \| f^{k+1(1/2)} \|_A^2. \]

Let
\[ Q(U^k) = \sum_{n=1}^{k} M_{k-n} \| \delta U^n \|_A^2 + (\alpha - 1) \| \delta U^{k+1} \|_A^2, \]

Summing up with respect to \( k \) from 0 to \( N - 1 \) leads to
\[ Q(U^N) \leq Q(U^0) + \sum_{k=0}^{N-1} M_k \| \Psi \|_A^2 + (\alpha - 1) \Delta t \sum_{k=0}^{N-1} \| f^{k+1(1/2)} \|_A^2. \]

The initial condition \( U^0 = \phi \) implies that \( Q(U^0) = (\alpha - 1) \| \delta \phi \|_A^2 \), and then
\[ Q(U^N) \leq (\alpha - 1) \| \delta \phi \|_A^2 + \sum_{k=0}^{N-1} M_k \| \Psi \|_A^2 \]
\[ + (\alpha - 1) \Delta t \sum_{k=0}^{N-1} \| f^{k+1(1/2)} \|_A^2. \]

**Theorem 1.** For scheme (24), we have the following stable conclusion:
\[ Q(U^m) \leq (\alpha - 1) \| \delta \phi \|_A^2 + \sum_{k=0}^{m-1} M_k \| \Psi \|_A^2 \]
\[ + (\alpha - 1) \Delta t \sum_{k=0}^{m-1} \| f^{k+1(1/2)} \|_A^2 \]
\[ \forall 0 \leq m \leq N. \]

3.2.2. **Optimal Error Estimates.** Combining equations (21) and (23) with (24), we get an error equation as follows:
\[ (\alpha - 1) \Delta t \alpha \delta u^{k+1} + M_0 \alpha \delta u^{k+1} - \frac{(\alpha - 1) \Delta t}{2} \delta v^{k+1} \]
\[ = \frac{(\alpha - 1) \Delta t}{2} \delta v^k + \sum_{m=1}^{k} \left( (M_{k-m} - M_{k-m+1}) \delta u^{k+1} \right) \]
\[ + (\alpha - 1) \Delta t R^{k+1(1/2)}, \]
where \( R^{k+1(1/2)} = O(\Delta t^2 + h^2) \) and \( e^i = u^i - u^k, \forall k \geq 0 \).

Multiplying \( h \delta u^{k+1} \) on both sides of equation (33) and summing up with respect to \( i \) from 1 to \( M - 1 \), we get
\[ (\alpha - 1) \Delta t \| \delta u^{k+1} \|_A^2 + M_0 \| \delta u^{k+1} \|_A^2 + \frac{\alpha - 1}{2} \| \delta u^{k+1} \|_A^2 \]
\[ = \frac{\alpha - 1}{2} \| \delta u^i \|_A^2 + \sum_{m=1}^{k} (M_{k-m} - M_{k-m+1}) \| \delta u^{k+1} \|_A^2 \]
\[ + (\alpha - 1) \Delta t \| R^{k+1(1/2)} \|_A^2. \]

By the triangle inequality and Lemma 2, we obtain
\[ \sum_{n=1}^{k} (M_{k-n} - M_{k-n+1}) (\alpha \delta u^e, \delta u^{k+1}) \]
\[ \leq \sum_{n=1}^{k} (M_{k-n} - M_{k-n+1}) \left( \| \delta u^e \|_A^2 + \| \delta u^{k+1} \|_A^2 \right) \]
\[ = \sum_{n=1}^{k} (M_{k-n} - M_{k-n+1}) \| \delta u^e \|_A^2 + \frac{1}{2} (M_0 - M_k) \| \delta u^{k+1} \|_A^2 \]
\[ \leq \sum_{n=1}^{k} M_{k-n} \| \delta u^e \|_A^2 + \sum_{n=1}^{k} \left( M_{k-n} \| \delta u^e \|_A^2 + \frac{1}{2} M_0 \| \delta u^{k+1} \|_A^2 \right) \]
\[ \leq \sum_{n=1}^{k} M_{k-n} \| \delta u^e \|_A^2 + (\alpha - 1) \| \delta u^{k+1} \|_A^2 \]
\[ + (\alpha - 1) \| \delta u^i \|_A^2 + (\alpha - 1) \Delta t \| R^{k+1(1/2)} \|_A^2. \]

By the definition of \( Q \) in stability analysis, the inequality (36) can be rearranged as
Summing up with respect to \( k \) from 0 to \( N - 1 \), we get
\[
Q(e^N) \leq C(\Delta t^2 + h^4).
\] (38)

Observing that the initial error \( e^0 = 0 \) implies \( Q(e^0) = 0 \). Then, we have
\[
(a - 1)|\delta e|^2 \leq Q(e^N) \leq C(\Delta t^2 + h^4),
\] (39)
where
\[
\delta e = e^N - e^1.
\]

### 4. Compact Finite Difference Scheme in Two Dimensions

In this section, the following fractional Cattaneo equation in two dimensions will be considered:
It is clear that
\[
\mathcal{A}_x v_{i,j} = \left( I + \frac{h_x^2 \Delta^2}{12} \right) v_{i,j},
\]
\[
\mathcal{A}_y v_{i,j} = \left( I + \frac{h_y^2 \Delta^2}{12} \right) v_{i,j}.
\] (44)

We also denote \(\mathcal{A}_x \mathcal{A}_y (u, v) = (\mathcal{A}_x \mathcal{A}_y) u, v\). It is easy to see that \(\mathcal{A}_x \mathcal{A}_y = \mathcal{A}_y \mathcal{A}_x\).

For any grid function \(u, v \in V^0\), the discrete inner product and norms are defined as follows:
\[
(u, v) = h_x h_y \sum_{i=1}^{M_x-1} \sum_{j=1}^{M_y-1} u_{i,j} v_{i,j},
\]
\[
\|u\|^2 = (u, u).
\] (45)

For the average operator \(\mathcal{A}_x \mathcal{A}_y\), define
\[
\mathcal{A}_x \mathcal{A}_y (v, v) \equiv (\mathcal{A}_x \mathcal{A}_y v, v) = \|v\|^2.
\] (46)

4.1. Compact Finite Difference Scheme. At the node \((x_i, y_j, t_{k+1})\), the differential equation is rewritten as
\[
\frac{\partial u}{\partial t}(x_i, y_j, t_{k+1}) + \mathcal{A}_x \mathcal{A}_y f(x_i, y_j, t_{k+1}) = 0
\]
\[
\frac{\partial^2 u}{\partial x^2}(x_i, y_j, t_{k+1}) + \frac{\partial^2 u}{\partial y^2}(x_i, y_j, t_{k+1})
\]
\[
+ f(x_i, y_j, t_{k+1}),
\]
\[
1 \leq i \leq M_x, 1 \leq j \leq M_y - 1, 0 \leq k \leq N - 1.
\] (47)

For the approximation of the time-fractional derivative, we have the following approximation [45]:
\[
\frac{\partial^\alpha u}{\partial t^\alpha}(x_i, y_j, t_{k+1}) = \frac{1}{(\alpha - 1) \Delta t} \left[ M_0 \delta_i u_{i,j}^{k+1} - \sum_{n=1}^{k} (M_{k-n}) \delta_i u_{i,j}^{k-n} - M_{k-n+1} \delta_i u_{i,j}^{k-n+1} - M_{k-n+1} \delta_i u_{i,j}^{k-n+1} - M_{k-n+1} \delta_i u_{i,j}^{k-n+1} \right]
\]
\[
+ R_{i,j}^{k+1/2},
\] (48)

where the truncation error \(R_{i,j}^{k+1/2} = O(\Delta t^2)\) and
\[
M_n = \exp \left[ \frac{1 - \alpha}{2 - \alpha} \Delta t \right] - \exp \left[ \frac{1 - \alpha}{2 - \alpha} \Delta t (n + 1) \right].
\] (49)

Furthermore, we also have
\[
\frac{\partial^2 u}{\partial x^2}(x_i, y_j, t_{k+1/2}) = \frac{1}{2} \left( \frac{\partial^2 u_{i,j}^{k+1}}{\partial x^2} + \frac{\partial^2 u_{i,j}^{k}}{\partial x^2} \right) + O(\Delta t^2 + h_x^2),
\]
\[
\frac{\partial^2 u}{\partial y^2}(x_i, y_j, t_{k+1/2}) = \frac{1}{2} \left( \frac{\partial^2 u_{i,j}^{k+1}}{\partial y^2} + \frac{\partial^2 u_{i,j}^{k}}{\partial y^2} \right) + O(\Delta t^2 + h_y^2),
\] (50)

\[
\frac{\partial u}{\partial t}(x_i, y_j, t_{k+1}) = \delta_i u_{i,j}^{k+1} + O(\Delta t^2).
\] (52)

Substituting (48) and (50)–(52) into (47) leads to
\[
\mathcal{A}_x \mathcal{A}_y \delta_i u_{i,j}^{k+1} + \frac{1}{(\alpha - 1) \Delta t} \mathcal{A}_x \mathcal{A}_y \left[ M_0 \delta_i u_{i,j}^{k+1} + \sum_{n=1}^{k} (M_{k-n} - M_{k-n+1}) \delta_i u_{i,j}^{k-n} - M_k \psi_{i,j} \right]
\]
\[
+ \frac{1}{2} \left( \delta_x^2 u_{i,j}^{k+1} + \delta_x^2 u_{i,j}^{k} \right) + \mathcal{A}_x \mathcal{A}_y f_{i,j}^{k+1/2}
\]
\[
+ R_{i,j}^{k+1/2},
\]
\[
1 \leq i \leq M_x - 1, 1 \leq j \leq M_y - 1, 0 \leq k \leq N - 1,
\] (53)

and there exists a constant \(C\), depending on the function \(u\) and its derivatives such that
\[
R_{i,j}^{k+1/2} \leq C(\Delta t^2 + h_x^2 + h_y^2).
\] (54)

By the initial and boundary conditions, we have
\[
u_{i,j}^0 = \phi_{i,j}, \quad 1 \leq i \leq M_x - 1, 1 \leq j \leq M_y - 1,
\]
\[
u_{i,j}^k = u_{M_x, j}^k = u_{i, M_y}^k = 0, \quad 0 \leq k \leq N.
\] (55)

Omitting the truncation error \(R_{i,j}^{k+1/2}\) and replacing the true solution \(u_{i,j}^k\) with numerical solution \(u_{i,j}^k\), a compact finite difference scheme can be obtained as follows:
\[
(a - 1) \Delta t \mathcal{A}_x \mathcal{A}_y \delta_i U_{i,j}^{k+1} + M_0 \mathcal{A}_x \mathcal{A}_y \delta_i \tilde{U}_{i,j}^{k+1}
\]
\[
- \mathcal{A}_y \left( \frac{a - 1}{2} \Delta t \delta_x U_{i,j}^{k+1} - \mathcal{A}_x \left( \frac{a - 1}{2} \Delta t \delta_x U_{i,j}^{k+1} \right) \right)
\]
\[
= \frac{a - 1}{2} \Delta t \delta_x U_{i,j}^{k} + \mathcal{A}_x \left( \frac{a - 1}{2} \Delta t \delta_x U_{i,j}^{k} \right)
\]
\[
+ \sum_{n=1}^{k} (M_{k-n} - M_{k-n+1}) \mathcal{A}_x \mathcal{A}_y \delta_i u_{i,j}^{k-n} + M_k \mathcal{A}_x \mathcal{A}_y \psi_{i,j}
\]
\[
+ \mathcal{A}_x \mathcal{A}_y f_{i,j}^{k+1/2},
\]
\[
1 \leq i \leq M_x - 1, 1 \leq j \leq M_y - 1, 1 \leq k \leq N - 1.
\] (56)
4.2. Stability Analysis and Optimal Error Estimates

4.2.1. Stability Analysis

**Definition 4** (see [46]). For any grid function \( u \in V_h^0 \), define the norm
\[
\| \nabla_h u \|_A = \left( \| \delta_x u \|_2^2 - \frac{h_y^2}{12} \| \delta_y u \|_2^2 \right) + \left( \| \delta_y u \|_2^2 - \frac{h_x^2}{12} \| \delta_x u \|_2^2 \right).
\]

(57)

The lemmas below is useful in the subsequent analysis of stability.

**Lemma 3** (see [46]). For any grid function \( u \in V_h^0 \), the following equation is held:
\[
\frac{1}{3} \| u \|_A^2 \leq \| u \|_A^2 \leq \| u \|_A^2.
\]

(58)

\[
(\alpha + 1) \Delta t \| \delta_x U^{k+1} \|_A^2 + M_0 \| \delta_x U^{k+1} \|_A^2 = (\alpha + 1) \Delta t \left( \mathcal{A}_x \delta_x^2 U^{k+1(1/2)} + \mathcal{A}_x \delta_x \delta_y U^{k-1(1/2)} + \delta_x U^{k+1} \right) + \sum_{n=1}^{k} (M_{k-n} - M_{k-n+1}) \left( \mathcal{A}_x \delta_x \psi, \delta_x U^{n} + \delta_x U^{k+1} \right) + M_k \left( \mathcal{A}_x \delta_x \psi, \delta_x U^{k+1} \right) + (\alpha - 1) \Delta t \left( \mathcal{A}_x \delta_y f^{k+1(1/2)}, \delta_y U^{k+1} \right).
\]

Observing Lemma 4, we have
\[
(\alpha - 1) \Delta t \left( \mathcal{A}_x \delta_y f^{k+1(1/2)}, \delta_y U^{k+1} \right) \leq \frac{(\alpha - 1) \Delta t}{2} \left( \| f^{k+1(1/2)} \|_A^2 + \| \delta_y U^{k+1} \|_A^2 \right).
\]

(62)

By the triangle inequality and Lemma 2, we obtain
\[
\sum_{n=1}^{k} (M_{k-n} - M_{k-n+1}) \left( \mathcal{A}_x \delta_x \psi, \delta_x U^{n} + \delta_x U^{k+1} \right) + M_k \left( \mathcal{A}_x \delta_x \psi, \delta_x U^{k+1} \right)
\]
\[
\leq \sum_{n=1}^{k} \frac{1}{2} (M_{k-n} - M_{k-n+1}) \left( \| \delta_x U^n \|_A^2 + \| \delta_x U^{k+1} \|_A^2 \right) + \frac{1}{2} M_k \left( \| \delta_x U^{k+1} \|_A^2 + \| \delta_x U^{k+1} \|_A^2 \right)
\]
\[
= \frac{1}{2} \sum_{n=1}^{k} (M_{k-n} - M_{k-n+1}) \| \delta_x U^n \|_A^2 + \frac{1}{2} \left( M_0 - M_k \right) \| \delta_x U^{k+1} \|_A^2 + \frac{1}{2} M_k \left( \| \delta_x U^{k+1} \|_A^2 + \| \delta_x U^{k+1} \|_A^2 \right)
\]
\[
+ \sum_{n=1}^{k} \frac{1}{2} M_{k-n} \| \delta_x U^n \|_A^2 + \frac{1}{2} \sum_{n=1}^{k} M_{k-n+1} \| \delta_x U^n \|_A^2 + \frac{1}{2} M_0 \| \delta_x U^{k+1} \|_A^2 + \frac{1}{2} M_k \| \psi \|_A^2.
\]

(63)

Combining equation (60) with (61)–(63), we get
\[
\sum_{n=1}^{k} M_{k-n+1} \| \delta_x U^n \|_A^2 + (\alpha - 1) \| \nabla_h U^{k+1} \|_A^2
\]
\[
\leq \sum_{n=1}^{k} M_{k-n} \| \delta_x U^n \|_A^2 + (\alpha - 1) \| \nabla_h U^{k+1} \|_A^2 + M_k \| \psi \|_A^2 + (\alpha - 1) \Delta t \| f^{k+1(1/2)} \|_A^2.
\]

(64)
Let
\[ Q(U^k) = \sum_{m=1}^{k} M_{k-m} \| \delta_i U^m \|_A^2 + (\alpha - 1) \| \nabla U^k \|_A^2, \]
(65)

Summing up with respect to \( k \) from 0 to \( N - 1 \), we get
\[ Q(U^N) \leq Q(U^0) + \sum_{k=0}^{N-1} M_k \| \nabla U^k \|_A^2 + (\alpha - 1) \Delta t \sum_{k=0}^{N-1} \| f^{k+1/2} \|_A^2. \]
(66)

Noting that \( U^0 = \phi \), we have \( Q(U^0) = (\alpha - 1) \| \nabla U^1 \|_A^2 \). It follows that
\[ Q(U^N) \leq (\alpha - 1) \| \nabla U^1 \|_A^2 + \sum_{k=0}^{N-1} M_k \| \nabla U^k \|_A^2 + (\alpha - 1) \Delta t \sum_{k=0}^{N-1} \| f^{k+1/2} \|_A^2. \]
(67)

**Theorem 3.** For the compact finite difference scheme (56), the following stability inequality holds:
\[ Q(U^m) \leq (\alpha - 1) \| \nabla U^1 \|_A^2 + \sum_{k=0}^{m-1} M_k \| \nabla U^k \|_A^2 + (\alpha - 1) \Delta t \sum_{k=0}^{m-1} \| f^{k+1/2} \|_A^2, \quad \forall 0 \leq m \leq N. \]
(68)

Similar to the stability, the convergence can also be analyzed.

**Theorem 4.** Suppose that the exact solution of the fractional Cattaneo equation is sufficiently smooth, then there exists a positive constant \( C \) independent of \( h, k, \) and \( \Delta t \) such that
\[ |e^{k}_1| \leq C (\Delta t^2 + h^2), \quad \forall 0 \leq k \leq N, \]
(69)
where \( e^{k}_1 = u^{k}_1 - U^k_1 \) and \( h = \max \{ h_x, h_y \} \).

5. Efficient Storage and Fast Evaluation of the Caputo–Fabrizio Fractional Derivative

Since time-fractional derivative operator is nonlocal, the traditional direct method for numerically solving the fractional Cattaneo equations generally requires total \( O(MN) \) memory units and \( O(MN^2) \) computational complexity, where \( N \) and \( M \) are the total number of time steps and space steps, respectively.

In this section, we develop a fast solution method for the finite difference scheme of the time-fractional Cattaneo equation.

Let
\[ N^{k+1}_n = M_{k-n} - M_{k-n+1} \]
\[ = \exp \left( \frac{1 - \alpha}{2 - \alpha} (k-n) \Delta t \right) \cdot \left[ 1 - 2 \exp \left( \frac{1 - \alpha}{2 - \alpha} \Delta t \right) + \exp \left( \frac{1 - \alpha}{2 - \alpha} 2 \Delta t \right) \right], \]
(70)

then
\[ N^{k+1}_n = M_{k+1-n} - M_{k+1-n+1} \]
\[ = \exp \left( \frac{1 - \alpha}{2 - \alpha} (k-n) \Delta t \right) \cdot \left[ 1 - 2 \exp \left( \frac{1 - \alpha}{2 - \alpha} \Delta t \right) + \exp \left( \frac{1 - \alpha}{2 - \alpha} 2 \Delta t \right) \right] \exp \left( \frac{1 - \alpha}{2 - \alpha} \Delta t \right), \]
(71)

So we have
\[ \sum_{n=1}^{k+1} N^{k+1}_n \delta_{i,n} u^n_i = \exp \left( \frac{1 - \alpha}{2 - \alpha} \Delta t \right) \sum_{n=1}^{k} N^{k}_n \delta_{i,n} u^n_i + N^{k+1}_i \delta_{i,n} u^{k+1}_i, \]
\[ \sum_{n=1}^{k+1} N^{k+1}_n \delta_{j,n} u^n_j = \exp \left( \frac{1 - \alpha}{2 - \alpha} \Delta t \right) \sum_{n=1}^{k} N^{k}_n \delta_{j,n} u^n_j + N^{k+1}_j \delta_{j,n} u^{k+1}_j. \]
(72)

**Remark.** We find that at the \( k \)-th level, only \( O(1) \) operations are needed to compute the \( k \)-th level since the \((k-1)\)-th level is known at that point. Thus, the total operations are reduced from \( O(N^2) \) to \( O(N) \), and the memory requirement decreases from \( O(N) \) to \( O(1) \). We conclude that this fast method significantly reduces the total computational cost from \( O(MN^2) \) to \( O(MN) \) and the memory requirement from \( O(MN) \) to \( O(M) \).

6. Numerical Experiments

In this section, we carry out several numerical experiments to check the effectiveness of the proposed scheme. The convergence rate and CPU consumption are all compared in the simulations. We take the space-time domain \( \Omega = [0, 1] \times [0, 1], T = 1 \) for one-dimensional case and \( \Omega = [0, 1] \times [0, 1], T = 1 \) for two-dimensional case. These simulations are implemented in Matlab, and the numerical experiments are run on a computer with 4GB memory. The time-fractional Cattaneo equation of the following forms is considered.

**Example 1.** We provide the exact solution \( u(x, t) = e^\delta \sin(\pi x), \) and for different \( \alpha \), we have different \( f(x, t) \) accordingly.
Table 1: Considering $\Delta t = h^2$, the discrete $\ell^\infty$ error and convergence rates of $u$ with different $\alpha$ for Example 1.

| $h$   | $\| e^N \|_{\infty}$, Order | $\| e^N \|_{\infty}$, Order | $\| e^N \|_{\infty}$, Order |
|-------|-----------------------------|-----------------------------|-----------------------------|
| $2^{-3}$ | 1.5320e – 04                | —                           | 1.5941e – 04                | —                           |
| $2^{-4}$ | 9.5109e – 06                | 4.0097                      | 9.8981e – 06                | 4.0094                      | 1.1230e – 05                | 4.0086                      |
| $2^{-5}$ | 5.9343e – 07                | 4.0024                      | 6.1762e – 07                | 4.0024                      | 7.0079e – 07                | 4.0022                      |
| $2^{-6}$ | 3.7074e – 08                | 4.0006                      | 3.8585e – 08                | 4.0006                      | 4.3783e – 08                | 4.0005                      |

Table 2: Considering $h = \sqrt{\Delta t}$, the discrete $l^2$ error and convergence rates of $u$ with different $\alpha$ for Example 1.

| $\Delta t$ | $\| e^N \|_{2}$, Order | $\| e^N \|_{2}$, Order | $\| e^N \|_{2}$, Order |
|-------------|-----------------------------|-----------------------------|-----------------------------|
| $2^{-4}$ | 0.0017                      | —                           | 0.0018                      | —                           |
| $2^{-6}$ | 1.0832e – 04                | 4.0382                      | 1.1271e – 04                | 4.0373                      | 1.2781e – 04                | 4.0344                      |
| $2^{-8}$ | 6.7252e – 06                | 4.0097                      | 6.9990e – 06                | 4.0094                      | 7.9405e – 06                | 4.0086                      |
| $2^{-10}$ | 4.1961e – 07                | 4.0024                      | 4.3672e – 07                | 4.0023                      | 4.9553e – 07                | 4.0021                      |
| $2^{-12}$ | 2.6215e – 08                | 4.0006                      | 2.7284e – 08                | 4.0005                      | 3.0959e – 08                | 4.0005                      |

Table 3: The CPU time consumption of the fast compact difference scheme and direct difference scheme for Example 1.

| $\Delta t$ | DCD | FCD | DCD | FCD | DCD | FCD |
|-------------|-----|-----|-----|-----|-----|-----|
| 1/1000      | 3.4008 | 1.8876 | 3.4632 | 1.9032 | 3.3540 | 1.9188 |
| 1/2500      | 14.2740 | 4.6800 | 14.1960 | 4.7580 | 13.8840 | 4.7424 |
| 1/5000      | 47.0967 | 9.3600 | 46.3010 | 9.4692 | 46.1450 | 9.3756 |
| 1/7500      | 99.4194 | 13.9464 | 95.6754 | 14.0088 | 96.7206 | 13.8684 |
| 1/10,000    | 169.9318 | 18.6733 | 164.1130 | 18.6421 | 165.8290 | 18.4393 |
| 1/25,000    | 981.6362 | 46.9101 | 950.9820 | 46.9251 | 959.1409 | 46.0046 |
| 1/50,000    | 3826.9853 | 93.4133 | 3692.3096 | 94.0062 | 3754.7412 | 91.9625 |

Figure 1: The CPU time (a) and the log-log CPU time (b) versus the total number of time steps $N$ for Example 1.
Table 4: Considering $\Delta t = 2^{-13}$, the discrete $\ell^\infty$ error and convergence rates of $u$ with different $\alpha$ for Example 1.

| $h$  | $\|e^N\|_{\ell^\infty}$ | Order | $\|e^N\|_{\ell^\infty}$ | Order | $\|e^N\|_{\ell^\infty}$ | Order |
|------|-----------------|-------|-----------------|-------|-----------------|-------|
| $2^{-3}$ | $2.2486 \times 10^{-4}$ | — | $2.2733 \times 10^{-4}$ | — | $2.3705 \times 10^{-4}$ | — |
| $2^{-4}$ | $1.3985 \times 10^{-4}$ | 4.0071 | $1.4140 \times 10^{-4}$ | 4.0069 | $1.4743 \times 10^{-4}$ | 4.0069 |
| $2^{-5}$ | $8.6897 \times 10^{-5}$ | 4.0084 | $8.7882 \times 10^{-5}$ | 4.0081 | $9.1726 \times 10^{-5}$ | 4.0068 |
| $2^{-6}$ | $5.0194 \times 10^{-5}$ | 4.1137 | $5.1024 \times 10^{-5}$ | 4.1063 | $5.4092 \times 10^{-5}$ | 4.0838 |

Table 5: Considering $h = 0.001$, the discrete $\ell^\infty$ error and convergence rates of $u$ with different $\alpha$ for Example 1.

| $\Delta t$ | $\|e^N\|_{\ell^\infty}$ | Order | $\|e^N\|_{\ell^\infty}$ | Order | $\|e^N\|_{\ell^\infty}$ | Order |
|------------|-----------------|-------|-----------------|-------|-----------------|-------|
| $2^{-3}$ | 0.0046 | — | 0.0043 | — | 0.0036 | — |
| $2^{-4}$ | 0.0011 | 2.0641 | 0.0011 | 1.9668 | 0.0063 | 1.9990 |
| $2^{-5}$ | 2.8662 | 1.9403 | 2.7171 | 2.0174 | 2.2518 | 1.9999 |
| $2^{-6}$ | 7.1658 | 1.9999 | 6.7928 | 2.0000 | 5.6295 | 2.0000 |
| $2^{-7}$ | 1.7915 | 2.0000 | 1.6982 | 2.0000 | 1.4074 | 2.0000 |

Figure 2: Considering $y = 0.001$, $\alpha = 1.5$, $M = N = 100$, real and numerical solutions of $u$ at $T = 1$ for Example 2 with the value of $x_0$ equal to (a) 0, (b) 0.2, (c) 0.4, (d) 0.6, (e) 0.8, and (f) 1.
\[
\begin{align*}
\frac{\partial u(x, t)}{\partial t} + \frac{\partial^\alpha u(x, t)}{\partial t^\alpha} &= \frac{\partial^2 u(x, t)}{\partial x^2} + f(x, t), \\
u(x, 0) &= \sin(\pi x), \quad \frac{\partial u}{\partial t}\big|_{t=0} = \sin(\pi x), \\
u(0, t) &= u(1, t) = 0.
\end{align*}
\] (73)

In Tables 1 and 2, we take \(\Delta t = h^2\) and \(h = \sqrt{\Delta t}\) to examine the discrete \(L^\infty\)-norm (\(L^2\)-norm) errors and corresponding spatial and temporal convergence rates, respectively. We list the errors and convergence rates (order) of the proposed compact finite difference (CD) scheme, which is almost \(O(\Delta t^2 + h^4)\) for different \(\alpha\). Additionally, Table 3 shows the CPU time (CPU) consumed by direct compact (DCD) scheme and fast compact difference (FCD) scheme, respectively. It is obvious that the FCD scheme has a significantly reduced CPU time over the DCD scheme. For instance, when \(\alpha = 1.5\), we choose \(h = 0.1\) and \(\Delta t = 1/50,000\) and observe that the FCD scheme consumes only 94 seconds, while the DCD scheme consumes 3692 seconds. We can find

**Figure 3:** Considering \(x_0 = 0.5, \alpha = 1.5, M = N = 100\), real and numerical solutions of \(u\) at \(T = 1\) for Example 2 with the value of \(\gamma\) equal to (a) 1, (b) 0.1, (c) 0.01, and (d) 0.001.
that the performance of the FCD scheme will be more conspicuous as the time step size \( \Delta t \) decreases.

In Figure 1, we set \( h = 0.1 \) and \( \alpha = 1.75 \) and change the total number of time steps \( N \) to plot out the CPU time (in seconds) of the FCD scheme and DCD scheme. We can observe that the CPU time increases almost linearly with respect to \( N \) for the FCD scheme, while the DCD scheme scales like \( O(N^2) \).

Tables 4 and 5 show the discrete \( \ell^\infty \) errors and convergence rates of the compact finite difference scheme for Example 1. The space rates are almost \( O(h^2) \) for fixed \( \Delta t = 2^{-13} \), and the time convergence rates are always \( O(\Delta t^2) \) for fixed \( h = 0.001 \). We can conclude that the numerical convergence rates of our scheme approach almost to \( O(\Delta t^2 + h^4) \).

### Example 2

The example is described by

\[
\frac{\partial u(x,t)}{\partial t} + \frac{\partial^\alpha u(x,t)}{\partial x^\alpha} = \frac{\partial^2 u(x,t)}{\partial x^2} + f(x,t),
\]

\[u(x,0) = 0, \quad \left. \frac{\partial u}{\partial t} \right|_{t=0} = 0,
\]

\[u(0,t) = t^2\exp\left(-\frac{x_0^2}{\gamma}\right), \quad u(1,t) = t^2\exp\left(-\frac{(1-x_0)^2}{\gamma}\right).
\]

(74)

Note that the exact solution of the above problem is

\[u(x,t) = t^2\exp\left(-(x-x_0)^2/\gamma\right), \text{ where } x_0 \in [0,1] \text{ and } \gamma \text{ is a.
Table 7: Considering $h = \sqrt{\Delta t}$ and $y = 0.01$, the discrete $L^2$ error and convergence rates of $u$ with different $\alpha$ and $x_0$ for Example 2.

| $x_0$ | $\Delta t$ | $\alpha = 1.25$ | Order | $\|e^N\|_2$ | Order | $\|e^N\|_2$ | Order |
|-------|-------------|-----------------|-------|---------------|-------|---------------|-------|
| 0     | $2^{-8}$    | 0.0044          | —     | 0.0044        | —     | 0.0042        | —     |
|       | $2^{-10}$   | 2.4205e-04      | 4.1841| 2.3721e-04    | 4.2133| 2.3098e-04    | 4.1845|
|       | $2^{-12}$   | 1.4684e-05      | 4.0430| 1.4390e-05    | 4.0430| 1.4013e-05    | 4.0429|
| 0.25  | $2^{-8}$    | 0.0027          | —     | 0.0027        | —     | 0.0027        | —     |
|       | $2^{-10}$   | 1.4902e-04      | 4.1794| 1.4894e-04    | 4.1802| 1.4892e-04    | 4.1803|
|       | $2^{-12}$   | 9.0680e-06      | 4.0386| 9.0633e-06    | 4.0386| 9.0616e-06    | 4.0386|
| 0.5   | $2^{-8}$    | 0.0027          | —     | 0.0027        | —     | 0.0027        | —     |
|       | $2^{-10}$   | 1.4848e-04      | 4.1846| 1.4842e-04    | 4.1852| 1.4842e-04    | 4.1852|
|       | $2^{-12}$   | 9.0323e-06      | 4.0390| 9.0288e-06    | 4.0390| 9.0286e-06    | 4.0390|
| 0.75  | $2^{-8}$    | 0.0027          | —     | 0.0027        | —     | 0.0027        | —     |
|       | $2^{-10}$   | 1.4902e-04      | 4.1794| 1.4894e-04    | 4.1802| 1.4892e-04    | 4.1803|
|       | $2^{-12}$   | 9.0680e-06      | 4.0386| 9.0633e-06    | 4.0386| 9.0616e-06    | 4.0386|
| 1     | $2^{-8}$    | 0.0044          | —     | 0.0044        | —     | 0.0042        | —     |
|       | $2^{-10}$   | 2.4205e-04      | 4.1841| 2.3721e-04    | 4.2133| 2.3098e-04    | 4.1845|
|       | $2^{-12}$   | 1.4684e-05      | 4.0430| 1.4390e-05    | 4.0430| 1.4013e-05    | 4.0429|

Figure 5: The CPU time (a) and the log-log CPU time (b) versus the total number of time steps $N$ for Example 3.

Table 8: Considering $\Delta t = h^2$, the discrete $L^\infty$ error and convergence rates of $u$ with different $\alpha$ for Example 3.

| $h$   | $\alpha = 1.25$ | Order | $\alpha = 1.5$ | Order | $\alpha = 1.75$ | Order |
|-------|-----------------|-------|-----------------|-------|-----------------|-------|
| $2^{-2}$ | 0.0028          | —     | 0.0028          | —     | 0.0030          | —     |
| $2^{-3}$ | 1.6948e-04      | 4.0462| 1.7328e-04      | 4.0142| 1.8411e-04      | 4.0263|
| $2^{-4}$ | 1.0522e-05      | 4.0096| 1.0759e-05      | 4.0095| 1.1434e-05      | 4.0092|
| $2^{-5}$ | 6.5652e-07      | 4.0024| 6.7133e-07      | 4.0024| 7.1346e-07      | 4.0024|
constant, and for different $\alpha$, we have different $f(x,t)$ accordingly.

We apply the fast compact difference scheme to discretize the equation. In Figure 2, we set $\gamma = 0.001, \alpha = 1.5$, and $M = N = 100$ and plot exact and numerical solutions at time $T = 1$ for Example 2 with different $x_0$. For $x_0 = 0.5, \alpha = 1.5$, and $M = N = 100$, we also plot exact and numerical solutions with the different $\gamma$ in Figure 3. In Figure 4, for $h = 0.1$ and $\alpha = 1.5$, we vary the total number of time steps $N$ to plot out the CPU time (in seconds) of the FCD scheme and DCD scheme. The numerical experiments verified our theoretical results. In Table 6, by equating $\Delta t = h^2$ and fixing $x_0 = 0.5$, we compute the discrete $L^2$ error and convergence rates with different fractional derivative orders $\alpha$ and different $\gamma$. It shows that the compact finite difference scheme has space accuracy of fourth order and temporal accuracy of second order. We set $h = \sqrt{\Delta t}$ and fix $\gamma = 0.01$, and the discrete $L^2$ error and convergence rates with different $\alpha$ and $x_0$ are displayed in Table 7. These numerical convergence rates are almost approaching $O(\Delta t^2 + h^4)$ as Example 1.

Example 3. If the exact solution is given by $u(x,t) = e^t \sin(\pi x) \sin(\pi y)$, we have different $f(x,t)$ for different $\alpha$ accordingly:

\[
\frac{\partial u}{\partial t} + \frac{\partial^\alpha u}{\partial t^\alpha} = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + f(x,t),
\]

\[
\begin{align*}
\frac{\partial u}{\partial t}(x,0) &= \sin(\pi x)\sin(\pi y), \\
\frac{\partial u}{\partial t}(x,0) &= \sin(\pi x)\sin(\pi y), \\
\end{align*}
\]

\[
\begin{align*}
u(0,x) &= u(1,x) = u(x,0) = u(x,1) = 0.
\end{align*}
\]

In Figure 5, $h = 0.1$ and $\alpha = 1.75$ are fixed, and the total number of time steps $N$ vary to plot out the CPU time (in seconds) of the FCD procedure and DCD procedure, and it presents an approximately linear computation complexity for FCD procedure. We set $\Delta t = h^2$ in Table 8, and $h = \sqrt{\Delta t}$ in Table 9, the discrete $F^\alpha$ error, discrete $L^2$ error, and convergence rates with different derivative orders $\alpha$ are presented. The fourth-order space accuracy and second-order temporal accuracy can be observed clearly.

7. Conclusion

In this paper, we develop and analyze a fast compact finite difference procedure for the Cattaneo equation equipped with time-fractional derivative without singular kernel. The time-fractional derivative is of Caputo–Fabrizio type with the order of $\alpha (1 < \alpha < 2)$. Compact difference discretization is applied to obtain a high-order approximation for spatial derivatives of integer order in the partial differential equation, and the Caputo–Fabrizio fractional derivative is discretized by means of Crank–Nicolson approximation. It has been proved that the proposed compact finite difference scheme has spatial accuracy of fourth order and temporal accuracy of second order. Since the fractional derivatives are history dependent and nonlocal, huge memory for storage and computational cost are required. This means extreme difficulty especially for a long-time simulation. Enlightened by the treatment for Caputo fractional derivative [32], we develop an effective fast evaluation procedure for the new Caputo–Fabrizio fractional derivative for the compact finite difference scheme. Several numerical experiments have been carried out to show the convergence orders and applicability of the scheme.

Inspired by the work [43], the topic about modelling and numerical solutions of porous media flow equipped with fractional derivatives is very interesting and challenging and will be our main research direction in the future.

Data Availability

All data generated or analyzed during this study are included in this article.
Conflicts of Interest
The authors declare that they have no conflicts of interest.

Acknowledgments
This work was supported in part by the National Natural Science Foundation of China under Grant nos. 91630207 and 11971272.

References
[1] A. Chaves, “A fractional diffusion equation to describe Lévy flights,” Physics Letters A, vol. 239, no. 1-2, pp. 13–16, 1998.
[2] M. Giona and H. E. Roman, “Fractional diffusion equation for transport phenomena in random media,” Physica A: Statistical Mechanics and Its Applications, vol. 185, no. 1-4, pp. 87–97, 1992.
[3] R. Metzler and J. Klafter, “The random walk’s guide to anomalous diffusion: a fractional dynamics approach,” Physics Reports, vol. 339, no. 1, pp. 1-77, 2000.
[4] R. L. Magin, Fractional Calculus in Bioengineering, Begell House Redding, Redding, CT, USA, 2006.
[5] R. L. Magin, O. Abdullah, D. Baleanu, and X. J. Zhou, “Anomalous diffusion expressed through fractional order differential operators in the Bloch-Torrey equation,” Journal of Magnetic Resonance, vol. 190, no. 2, pp. 255–270, 2008.
[6] I. Podlubny, Fractional Differential Equations: An Introduction to Fractional Derivatives, Fractional Differential Equations, to Methods of Their Solution and Some of Their Applications, Vol. 198, Elsevier, Amsterdam, Netherlands, 1998.
[7] M. Ran and C. Zhang, “Compact difference scheme for a class of fractional-in-space nonlinear damped wave equations in two space dimensions,” Computers & Mathematics with Applications, vol. 71, no. 5, pp. 1151–1162, 2016.
[8] A. H. Bhrawy and M. A. Zaky, “Highly accurate numerical schemes for multi-dimensional space variable-order fractional Schrödinger equations,” Computers & Mathematics with Applications, vol. 73, no. 6, pp. 1100–1117, 2017.
[9] S. B. Yuste, “Weighted average finite difference methods for fractional diffusion equations,” Journal of Computational Physics, vol. 216, no. 1, pp. 264–274, 2006.
[10] T. A. M. Langlands and B. I. Henry, “The accuracy and stability of an implicit solution method for the fractional diffusion equation,” Journal of Computational Physics, vol. 205, no. 2, pp. 719–736, 2005.
[11] C.-M. Chen, F. Liu, I. Turner, and V. Anh, “A Fourier method for the fractional diffusion equation describing sub-diffusion,” Journal of Computational Physics, vol. 227, no. 2, pp. 886–897, 2007.
[12] Y. Lin and C. Xu, “Finite difference/spectral approximations for the time-fractional diffusion equation,” Journal of Computational Physics, vol. 225, no. 2, pp. 1533–1552, 2007.
[13] G.-H. Gao and Z.-Z. Sun, “A compact finite difference scheme for the fractional sub-diffusion equations,” Journal of Computational Physics, vol. 230, no. 3, pp. 586–595, 2011.
[14] F. Zeng, C. Li, F. Liu, and I. Turner, “Numerical algorithms for time-fractional subdiffusion equation with second-order accuracy,” SIAM Journal on Scientific Computing, vol. 37, no. 1, pp. A55–A78, 2015.
[15] G.-H. Gao, Z.-Z. Sun, and H.-W. Zhang, “A new fractional numerical differentiation formula to approximate the Caputo fractional derivative and its applications,” Journal of Computational Physics, vol. 259, no. 2, pp. 33–50, 2014.
[16] H. Li, J. Cao, and C. Li, “High-order approximation to Caputo derivatives and Caputo-type advection-diffusion equations (III),” Journal of Computational and Applied Mathematics, vol. 299, pp. 159–175, 2016.
[17] A. A. Alikhanov, “A new difference scheme for the time fractional diffusion equation,” Journal of Computational Physics, vol. 280, pp. 424–438, 2015.
[18] Z. Wang and S. Vong, “Compact difference schemes for the modified anomalous fractional sub-diffusion equation and the fractional diffusion-wave equation,” Journal of Computational Physics, vol. 277, pp. 1–15, 2014.
[19] H.-K. Pang and H.-W. Sun, “Fourth order finite difference schemes for time-space fractional sub-diffusion equations,” Computers & Mathematics with Applications, vol. 71, no. 6, pp. 1287–1302, 2016.
[20] M. Dehghan, M. Abbaszadeh, and W. Deng, “Fourth-order numerical method for the space-time tempered fractional diffusion-wave equation,” Applied Mathematics Letters, vol. 73, no. 11, pp. 120–127, 2017.
[21] H. Sun, Z.-Z. Sun, and G.-H. Gao, “Some high order difference schemes for the space and time fractional Bloch-Torrey equations,” Applied Mathematics and Computation, vol. 281, pp. 356–380, 2016.
[22] S. Arshad, J. Huang, A. Q. M. Khaliq, and Y. Tang, “Trapezoidal scheme for time-space fractional diffusion equation with Riesz derivative,” Journal of Computational Physics, vol. 350, pp. 1–15, 2017.
[23] H. Wang, K. Wang, and T. Sircar, “A direct O(Nlog2N) finite difference method for fractional diffusion equations,” Journal of Computational Physics, vol. 229, no. 21, pp. 8095–8104, 2010.
[24] H. Wang and T. S. Basu, “A fast finite difference method for two-dimensional space-fractional diffusion equations,” SIAM Journal on Scientific Computing, vol. 34, no. 5, pp. A2444–A2458, 2012.
[25] H. Wang and N. Du, “A superfast-preconditioned iterative method for steady-state space-fractional diffusion equations,” Journal of Computational Physics, vol. 240, pp. 49–57, 2013.
[26] H. Wang and N. Du, “A fast finite difference method for three-dimensional time-dependent space-fractional diffusion equations and its efficient implementation,” Journal of Computational Physics, vol. 253, pp. 50–63, 2013.
[27] H. Wang and N. Du, “Fast alternating-direction finite difference methods for three-dimensional space-fractional diffusion equations,” Journal of Computational Physics, vol. 258, pp. 305–318, 2014.
[28] K. Wang and H. Wang, “A fast characteristic finite difference method for fractional advection-diffusion equations,” Advances in Water Resources, vol. 34, no. 7, pp. 810–816, 2011.
[29] H. Wang and H. Tian, “A fast Galerkin method with efficient matrix assembly and storage for a peridynamic model,” Journal of Computational Physics, vol. 231, no. 23, pp. 7730–7738, 2012.
[30] H. Tian, H. Wang, and W. Wang, “An efficient collocation method for a non-local diffusion model,” International Journal of Numerical Analysis & Modeling, vol. 10, no. 4, pp. 815–825, 2013.
[31] H. Wang and H. Tian, “A fast and faithful collocation method with efficient matrix assembly for a two-dimensional nonlocal diffusion model,” Computer Methods in Applied Mechanics and Engineering, vol. 273, no. 5, pp. 19–36, 2014.
[32] S. Jiang, J. Zhang, Q. Zhang, and Z. Zhang, “Fast evaluation of the Caputo fractional derivative and its applications to fractional diffusion equations,” Communications in Computational Physics, vol. 21, no. 3, pp. 650–678, 2017.
Y. Yan, Z.-Z. Sun, and J. Zhang, "Fast evaluation of the Caputo fractional derivative and its applications to fractional diffusion equations: a second-order scheme," *Communications in Computational Physics*, vol. 22, no. 4, pp. 1028–1048, 2017.

X. Lu, H.-K. Pang, and H.-W. Sun, "Fast approximate inversion of a block triangular Toeplitz matrix with applications to fractional sub-diffusion equations," *Numerical Linear Algebra with Applications*, vol. 22, no. 5, pp. 866–882, 2015.

M. Caputo and M. Fabrizio, "A new definition of fractional derivative without singular kernel," *Progress in Fractional Differentiation and Applications*, vol. 1, no. 2, pp. 73–85, 2015.

A. Atangana, "On the new fractional derivative and application to nonlinear Fisher's reaction-diffusion equation," *Applied Mathematics and Computation*, vol. 273, pp. 948–956, 2016.

M. D. Ortigueira and J. Tenreiro Machado, "A critical analysis of the Caputo-Fabrizio operator," *Communications in Nonlinear Science and Numerical Simulation*, vol. 59, pp. 608–611, 2018.

V. E. Tarasov, "No nonlocality no fractional derivative," *Communications in Nonlinear Science and Numerical Simulation*, vol. 62, pp. 157–163, 2018.

V. Tarasov, "Caputo-Fabrizio operator in terms of integer derivatives: memory or distributed lag?," *Computational and Applied Mathematics*, vol. 38, p. 113, 2019.

A. Giusti, "A comment on some new definitions of fractional derivative," *Nonlinear Dynamics*, vol. 93, no. 3, pp. 1757–1763, 2018.

A. Atangana and J. F. Gómez-Aguilar, "Fractional derivatives with no-index law property: application to chaos and statistics," *Chaos, Solitons & Fractals*, vol. 114, pp. 516–535, 2018.

J. Hristov, "Response functions in linear viscoelastic constitutive equations and related fractional operators," *Mathematical Modelling of Natural Phenomena*, vol. 14, no. 3, pp. 1–34, 2019.

L. X. Vivas-Cruz, A. Gonzalez-Calderon, M. A. Taneco-Hernandez, and D. P. Luis, "Theoretical analysis of a model of fluid flow in a reservoir with the Caputo-Fabrizio operator," *Communications in Nonlinear Science and Numerical Simulation*, vol. 84, Article ID 105186, 2020.

H.-L. Liao and Z.-Z. Sun, "Maximum norm error bounds of ADI and compact ADI methods for solving parabolic equations," *Numerical Methods for Partial Differential Equations*, vol. 26, no. 1, pp. 37–60, 2010.

Z. Liu, A. Cheng, and X. Li, "A second order Crank-Nicolson scheme for fractional Cattaneo equation based on new fractional derivative," *Applied Mathematics and Computation*, vol. 311, pp. 361–374, 2017.

Z. Sun and G. Gao, *Finite Difference Method for Fractional Differential Equations*, Science Press, Beijing, China, 2015.