



Article

Fast Compact Difference Scheme for Solving the Two-Dimensional Time-Fractional Cattaneo Equation

Lijuan Nong¹, Qian Yi¹, Jianxiong Cao² and An Chen^{3,*} ¹ College of Mathematics and Statistic, Guangxi Normal University, Guilin 541004, China² College of Science, Lanzhou University of Technology, Lanzhou 730050, China³ College of Science, Guilin University of Technology, Guilin 541004, China

* Correspondence: chena@glut.edu.cn

Abstract: The time-fractional Cattaneo equation is an equation where the fractional order $\alpha \in (1, 2)$ has the capacity to model the anomalous dynamics of physical diffusion processes. In this paper, we consider an efficient scheme for solving such an equation in two space dimensions. First, we obtain the space's semi-discrete numerical scheme by using the compact difference operator in the spatial direction. Then, the semi-discrete scheme is converted to a low-order system by means of order reduction, and the fully discrete compact difference scheme is presented by applying the $L2-1_\sigma$ formula. To improve the computational efficiency, we adopt the fast discrete Sine transform and sum-of-exponentials techniques for the compact difference operator and $L2-1_\sigma$ difference operator, respectively, and derive the improved scheme with fast computations in both time and space. That aside, we also consider the graded meshes in the time direction to efficiently handle the weak singularity of the solution at the initial time. The stability and convergence of the numerical scheme under the uniform meshes are rigorously proven, and it is shown that the scheme has second-order and fourth-order accuracy in time and in space, respectively. Finally, numerical examples with high-dimensional problems are demonstrated to verify the accuracy and computational efficiency of the derived scheme.

Keywords: time-fractional Cattaneo equation; compact difference operator; $L2-1_\sigma$ method; fast discrete Sine transform; sum of exponentials



Citation: Nong, L.; Yi, Q.; Cao, J.; Chen, A. Fast Compact Difference Scheme for Solving the Two-Dimensional Time-Fractional Cattaneo Equation. *Fractal Fract.* **2022**, *6*, 438. <https://doi.org/10.3390/fractalfract6080438>

Academic Editors: Arsen V. Pskhu and Roman Parovik

Received: 26 June 2022

Accepted: 9 August 2022

Published: 11 August 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Let $\Omega \subset \mathbb{R}^2$ be a bounded rectangle domain and T be the fixed time. In this paper, we study the efficient numerical scheme for solving the following time-fractional Cattaneo equation:

$$\partial_t u(\mathbf{x}, t) + \kappa {}_C D_{0,t}^\alpha u(\mathbf{x}, t) = \mu \Delta u(\mathbf{x}, t) + f(\mathbf{x}, t), \text{ in } \Omega \times (0, T], \quad (1)$$

where the boundary and initial conditions are $u(\mathbf{x}, t) = 0$ on $\partial\Omega \times (0, T]$ and $u(\mathbf{x}, 0) = u_0(\mathbf{x})$, $\partial_t u(\mathbf{x}, 0) = u_1(\mathbf{x})$ in Ω . Here, $\mathbf{x} = (x_1, x_2)$. The parameters κ and μ are the positive constants related to the relaxation time and diffusion properties, respectively. The source term f and the initial conditions u_0 and u_1 are given suitable functions. The Laplacian Δ is defined by $\Delta = \partial_{x_1}^2 + \partial_{x_2}^2$. The Caputo derivative ${}_C D_{0,t}^\alpha$ with $\alpha \in (1, 2]$ is given by

$${}_C D_{0,t}^\alpha v(t) = \frac{1}{\Gamma(2-\alpha)} \int_0^t (t-s)^{1-\alpha} v''(s) ds,$$

where $\Gamma(\cdot)$ is the Gamma function.

The time-fractional Cattaneo Equation (1) provides a flexible and efficient way to model the anomalous dynamics of physical diffusion processes, especially the general dynamics crossover behaviors [1,2]. In recent years, many efforts have been made in the

theoretical and numerical study of such a time-fractional equation or its variants. See [2,3] for the theoretical case and [1,4–6] for the numerical case, just to name a few. One may refer to these two review papers [7,8] or this book [9] for further details.

In [4], Zhao and Sun proposed a compact Crank–Nicolson scheme for solving the time-fractional Cattaneo Equation (1). Ren and Gao also constructed an alternating direction implicit (ADI) compact difference scheme by adding a small term to improve the computational efficiency for the two-dimensional problem [5]. The temporal convergence order of both of the above two methods is the order $3 - \alpha$, where $1 < \alpha < 2$. Later on, Chen and Li proposed a temporal second-order ADI Galerkin scheme by applying the Galerkin finite element method in space and the L_2-1_σ method in time [6]. In [1], Chen and Nong developed two efficient, fully discrete schemes for solving Equation (1) based on the Galerkin finite method in space and the convolution quadrature in time. Error estimates and numerical experiments show that their schemes are efficient when dealing with different data regularity in Equation (1). It seems that the schemes mentioned above still have some limitations, especially in the high-dimensional and non-smooth solution problems. This motivated us to develop an efficient numerical scheme to cope with such situations.

For the high-dimensional time-fractional model, there are two ways to improve the computational efficiency of the numerical scheme. One involves the spatial direction, such as with the ADI technique, fast discrete Sine transforms (DSTs), and so on (see [6,10]). The other one involves the temporal direction, such as with the fast convolution quadrature and sum-of-exponentials (SOE) techniques (see [11–14]). In this paper, we focus on the DST and SOE techniques.

The DST technique has attracted the interest of many scholars in recent years, mainly because it avoids the direct inversion for the coefficient matrix of the linear discrete systems and, at the same time, ensures the convergence accuracy of the difference operators [15,16]. In [15], Wang, et al. proposed an efficient fourth-order compact finite difference scheme for solving the Poisson equation based on the fast discrete Sine transform and greatly reduced the computational cost by avoiding matrix inversion for the discretized system. There has been some work in recent years on the application of the DST technique in numerically solving high-dimensional time-fractional equations (see [10,17,18]).

The SOE technique is an efficient way to reduce the computational complexity caused by the non-locality of fractional derivatives [13,19]. In [13], Yan et al. presented the fast L_2-1_σ (denoted by FL_2-1_σ) formula by employing the SOE technique with the kernel function in the Caputo derivative. Subsequently, some numerical studies of time-fractional models based on this approach have emerged (see [10,19] and the corresponding references therein). In [19], Liang et al. proposed a fast difference scheme for solving the one-dimensional time-fractional telegraph equation based on the FL_2-1_σ formula. In [10], Li et al. derived a fast compact difference scheme for solving subdiffusion equations by applying the FL_2-1_σ in time and the compact difference operator in space. It is worth mentioning that the compact difference operator is implemented by the DST via the fast Fourier transform (FFT) algorithm. Therefore, their scheme is computationally efficient in both the time direction and spatial dimensions.

Motivated by the work in [10], we aim to develop a fast compact difference scheme for solving Equation (1). To this end, we derive a compact difference scheme by applying the L_2-1_σ formula in time and compact difference operator in space. Then, the SOE and DST techniques are both adopted to improve the computational performance of the derived scheme. The contributions of our paper are as follows. First, we construct a compact difference scheme with fast computation in both the temporal and spatial directions, which effectively handles the high computational cost caused by the high spatial dimension and the non-locality of the Caputo derivative (see the numerical scheme (10)). Second, we present the rigorous stability and error estimate for the uniform mesh-based fast compact difference scheme (see Theorems 1 and 2). Third, extensive numerical experiments are designed to verify the accuracy and efficiency of the fast compact difference scheme (see the Section 5, Tables 1 and 2 for the accuracy; Table 5 and Figure 1 for efficiency). Besides,

the fast compact difference scheme based on graded meshes in time is adopted to handle non-smooth solution problems (see Tables 3 and 4 in Section 5).

This paper is organized as follows. In Section 2, we present the fully discrete difference scheme based on the compact difference operator for the Laplacian in space and the L2-1 σ formula for the Caputo derivative in time. In order to improve the computational performance of the derived scheme, we further adopt the DST and SOE techniques to obtain the fast compact difference scheme in Section 3. In Section 4, the stability and error estimate of the fast compact difference scheme based on the uniform meshes are provided rigorously. Numerical examples and the conclusions of this paper are given in Sections 5 and 6, respectively.

2. The Compact Difference Scheme

In this section, we first approximate the Laplacian Δ in Equation (1) by applying the compact difference operator and then find the low-order system for the space’s semi-discrete scheme with the aid of the reduced-order method so that the fully discrete scheme of the equation can be obtained by using the second-order L2-1 σ formula.

To this end, we introduce some useful notations in the following. Suppose the rectangle domain $\Omega = [x_1^L, x_1^R] \times [x_2^L, x_2^R]$. Let the spatial step size be $h_k = (x_k^R - x_k^L) / M_k$ and the grid points be $x_{k,j_k} = x_k^L + j_k h_k$ ($j_k = 0, 1, \dots, M_k$), where M_k ($k = 1, 2$) is a positive integer. The fully discrete grids on Ω are given by $\bar{\Omega}_h = \{\mathbf{x}_h = (x_{1,j_1}, x_{2,j_2}) | 0 \leq j_k \leq M_k (k = 1, 2)\}$. We will use M without a subscript to denote M_1 (or M_2) when $M_1 = M_2$, unless otherwise specified. The inner and boundary grids are denoted by $\Omega_h = \bar{\Omega}_h \cap \Omega$ and $\partial\Omega_h = \bar{\Omega}_h \cap \partial\Omega$, respectively. We denote the space of the grid function as $V_h = \{v | v = (v_h)_{\mathbf{x}_h} \text{ and } v_h = 0 \text{ for } \mathbf{x}_h \in \partial\Omega_h\}$.

For the grid function $v_h = v(\mathbf{x}_h)$ with the index vector $h = (i_1, i_2)$ at the k th position ($k = 1, 2$), the compact difference operator is given by $\bar{\Delta}_k v_{i_k} = \frac{\delta_k^2}{h_k^2} v_{i_k}$, where

$$\mathcal{H}_k v_{i_k} = \left(I + \frac{h_k^2}{12} \delta_k^2 \right) v_{i_k}.$$

Here, we have

$$\delta_k^2 v_{i_k} = \frac{\delta_k v_{i_k+\frac{1}{2}} - \delta_k v_{i_k-\frac{1}{2}}}{h_k} \quad \text{and} \quad \delta_k v_{i_k+\frac{1}{2}} = \frac{v_{i_k+1} - v_{i_k}}{h_k}.$$

Therefore, for $\mathbf{x}_h \in \Omega_h$, we have $\Delta v(\mathbf{x}_h) = \bar{\Delta}_h v_h + O(\mathbf{h}^4)$ with $\bar{\Delta}_h v_h = (\bar{\Delta}_1 + \bar{\Delta}_2) v_h$ and $\mathbf{h} = (h_1, h_2)$.

When employing the compact difference operator $\bar{\Delta}_h$ for the discretization of the Laplacian Δ in (1), one has

$$\partial_t u(\mathbf{x}_h, t) + \kappa {}_C D_{0,t}^\alpha u(\mathbf{x}_h, t) = \mu \bar{\Delta}_h u(\mathbf{x}_h, t) + f(\mathbf{x}_h, t) + R_x(t),$$

from which we obtain the following space semi-discrete scheme:

$$\partial_t u_h(t) + \kappa {}_C D_{0,t}^\alpha u_h(t) = \mu \bar{\Delta}_h u_h(t) + f_h(t), \tag{2}$$

where $u_h(t) \approx u(\mathbf{x}_h, t)$ and the truncation error $R_x(t) = O(\mathbf{h}^4)$.

Next, we introduce the L2-1 σ formula for the approximation of the Caputo derivative ${}_C D_{0,t}^\beta$ with $\beta \in (0, 1)$. Let the temporal step size be $\tau = T / N_t$. Here, N_t is a positive integer. Denote $t_n = n\tau$, where $n \geq 0$ and $t_{n+\frac{1}{2}} = (t_n + t_{n+1}) / 2$. For a given function $g(t)$, denote

$$\begin{aligned} \delta_t g^{n+\frac{1}{2}} &= \frac{g(t_{n+1}) - g(t_n)}{\tau}, & g^{n+\frac{1}{2}} &= \frac{g(t_{n+1}) + g(t_n)}{2}, \\ \widehat{\delta}_t g^n &= \frac{1}{2\tau} ((2\sigma + 1)g(t_{n+1}) - 4\sigma g(t_n) + (2\sigma - 1)g(t_{n-1})), \end{aligned}$$

and $g^{n+\sigma} = \sigma g(t_{n+1}) + (1 - \sigma)g(t_n)$, where $\sigma = 1 - \beta/2$. For more details about the difference operator $\widehat{\delta}_t$, one may refer to Lemma 2.1 of [20].

We state the L2-1 $_{\sigma}$ formula as follows [21]. If $g(t) \in C^3[0, T]$, then

$${}_C D_{0,t}^{\beta} g(t)|_{t=t_{n+\sigma}} = \overline{D}_{\tau}^{\beta} g^{n+\sigma} + O(\tau^{3-\beta}), \tag{3}$$

where $\overline{D}_{\tau}^{\beta} g^{n+\sigma} = \sum_{k=0}^n w_{n-k}^{(n+1)}(g(t_{k+1}) - g(t_k))$ and for $n = 0, w_0^{(1)} = a_0 \tau^{-\beta} / \Gamma(2 - \beta)$, where $n \geq 1$, we have

$$w_k^{(n+1)} = \frac{\tau^{-\beta}}{\Gamma(2 - \beta)} \begin{cases} a_0 + b_1, & k = 0, \\ a_k + b_{k+1} - b_k, & k = 1, 2, \dots, n - 1, \\ a_n - b_n, & k = n. \end{cases} \tag{4}$$

Here, $a_0 = \sigma^{1-\beta}, a_k = (k + \sigma)^{1-\beta} - (k - 1 + \sigma)^{1-\beta}, k \geq 1$, and

$$b_k = \frac{(k + \sigma)^{2-\beta} - (k - 1 + \sigma)^{2-\beta}}{2 - \beta} - \frac{(k + \sigma)^{1-\beta} - (k - 1 + \sigma)^{1-\beta}}{2}, \quad k \geq 1.$$

By setting $\phi_h(t) = \partial_t u_h(t)$ for (2), we find the following lower-order system:

$$\phi_h(t) + \kappa {}_C D_{0,t}^{\alpha-1} \phi_h(t) = \mu \overline{\Delta}_h u_h(t) + f_h(t). \tag{5}$$

Therefore, by replacing ${}_C D_{0,t}^{\alpha-1}$ with the difference operator $\overline{D}_{\tau}^{\alpha-1}$ of the L2-1 $_{\sigma}$ Formula (3) in the space semi-discrete scheme (2), one has

$$\begin{cases} \phi_h(t_{n+\sigma}) + \kappa \overline{D}_{\tau}^{\alpha-1} \phi_h^{n+\sigma} = \mu \overline{\Delta}_h u_h(t_{n+\sigma}) + f_h(t_{n+\sigma}) + R_{xt}^n, \\ \widehat{\delta}_t u_h^{1/2} = \phi_h(t_{1/2}) + r^0, \\ \widehat{\delta}_t u_h^n = \phi_h(t_{n+\sigma}) + r^n, \end{cases}$$

where the truncation error $R_{xt}^n = O(\tau^2 + h^4)$ and $r^0 = r^n = O(\tau^2)$. By dropping the small terms R_{xt}^n, r^0 , and r^n , we have the following compact difference scheme for solving Equation (1). We find the numerical solution ϕ_h^n of $\phi(\mathbf{x}_h, t_n)$ for $n \geq 1$ such that

$$\begin{cases} \phi_h^{n+\sigma} + \kappa \overline{D}_{\tau}^{\alpha-1} \phi_h^{n+\sigma} = \mu \overline{\Delta}_h u_h^{n+\sigma} + f_h^{n+\sigma}, \\ \delta_t u_h^{1/2} = \phi_h^{1/2}, \\ \widehat{\delta}_t u_h^n = \phi_h^{n+\sigma}, \end{cases} \tag{6}$$

where $u_h^0 = u_0(\mathbf{x}_h), \phi_h^0 = u_1(\mathbf{x}_h)$, and $u(\mathbf{x}_h)|_{\mathbf{x}_h \in \partial \Omega_h} = 0$. From (6), we obtain the numerical solution u_h^n of $u(\mathbf{x}_h, t_n)$ to (1): $u_h^1 = u_h^0 + \tau(\phi_h^1 + \phi_h^0)/2$ and for $n \geq 1$:

$$u_h^{n+1} = \frac{2\tau}{2\sigma + 1} (\sigma \phi_h^{n+1} + (1 - \sigma) \phi_h^n) + \frac{1}{2\sigma + 1} (4\sigma u_h^n - (2\sigma - 1) u_h^{n-1}).$$

3. The Derivation of the Fast Compact Difference Scheme

In this part, we use the DST and SOE techniques to improve the computational performance of the compact difference scheme (6).

By utilizing the discrete sine transform, one can derive that

$$\widehat{v}_{jk}^n \approx \widehat{v}_{jk} \frac{12}{h_k^2} \cdot \frac{s_{jk} - 1}{s_{jk} + 5} := \widehat{v}_{jk} \lambda^{(j_k, M_k)},$$

where $s_{jk} = \cos(\frac{j_k \pi}{M_k})$ and $1 \leq j_k \leq M_k - 1 (k = 1, 2)$ (see [15] for more details about the derivation).

Therefore, the scheme in Equation (6) is equivalent to

$$\begin{cases} \widehat{\phi}_\nu^{n+\sigma} + \kappa \overline{D}_\tau^{\alpha-1} \widehat{\phi}_\nu^{n+\sigma} = \mu \left(\lambda^{(j_1, M_1)} + \lambda^{(j_2, M_2)} \right) \widehat{u}_\nu^{n+\sigma} + \widehat{f}_\nu^{n+\sigma}, \\ \delta_t \widehat{u}_\nu^{1/2} = \widehat{\phi}_\nu^{1/2}, \\ \widehat{\delta}_t \widehat{u}_\nu^n = \widehat{\phi}_\nu^{n+\sigma}, \end{cases} \tag{7}$$

where ν is the index set defined by $\nu = \{(j_1, j_2) | 1 \leq j_k \leq M_k - 1 (k = 1, 2)\}$. We can obtain the numerical solution to Equation (1) by the following steps:

- (a) Compute $\widehat{\phi}_\nu^0, \widehat{u}_\nu^0,$ and \widehat{f}_ν^n from $\phi_h^0, u_h^0,$ and f_h^n by means of DST, where $n \geq 1$;
- (b) Compute $\widehat{\phi}_\nu^n$ by solving the system (7), and then obtain $\widehat{u}_\nu^n,$ where $n \geq 1$;
- (c) Obtain the numerical solutions ϕ_h^n and u_h^n from $\widehat{\phi}_\nu^n$ and $\widehat{u}_\nu^n,$ respectively, by means of the inverse of DST, where $n \geq 1$.

It is worth mentioning that if we solve numerical scheme (6) directly, the inverse of the coefficient matrix needs to be solved at each time layer, and this leads to a computational cost of $O(M_1^2 M_2^2)$. If the DST technique based on FFT, that is the scheme (7) is used, then the computational cost will be reduced to $O(M_1 M_2 \log(M_1 M_2))$.

Next, we consider the SOE technique to further reduce the computational complexity of the scheme (7) caused by the non-locality of the Caputo derivative.

Without loss of generality, we adopt the graded meshes $t_n = T(\frac{n}{N_t})^\gamma$ in time, where $n = 0, \dots, N_t$ and $\gamma \geq 1$ is a proper chosen temporal mesh grading parameter. Denote the time step size $\tau_n = t_n - t_{n-1}$. Set $t_{n+\sigma} = t_n + \sigma \tau_{n+1}$ and $D_\tau g^k = \frac{g^k - g^{k-1}}{\tau_k}$. The fast L2-1 σ formula based on the graded meshes is defined as

$$F\overline{D}_{\tau_n}^\beta g^{n+\sigma} = \omega_0 D_\tau g^{n+1} + \sum_{k=1}^{N_{\text{exp}}} \omega_k e^{-s_k \sigma \tau_{n+1}} H_k(t_n), \tag{8}$$

where $\omega_0 = \frac{(\sigma \tau_{n+1})^{1-\beta}}{\Gamma(2-\beta)}$ and $H_k(t_j)$ ($j \geq 0, k \geq 1$) is obtained by the following recurrence:

$$H_k(t_j) = e^{-s_k \tau_j} H_k(t_{j-1}) + a_{k,j} D_\tau g^j + b_{k,j} (D_\tau g^{j+1} - D_\tau g^j), \tag{9}$$

with the initial value $H_k(t_0) = 0$ [10]. Here, the coefficients $a_{k,j} = \frac{1}{s_k} (1 - e^{-s_k \tau_j})$ and $b_{k,j} = \frac{2}{\tau_j + \tau_{j+1}} \int_{t_{j-1}}^{t_j} (t - t_{j-\frac{1}{2}}) e^{-s_k(t_j-t)} dt$. The weights ω_k ($k \geq 1$) and the points s_k ($k \geq 1$) in Formula (8) are chosen to satisfy

$$\left| \frac{t^{-\beta}}{\Gamma(1-\beta)} - \sum_{k=1}^{N_{\text{exp}}} \omega_k e^{-s_k t} \right| \leq \epsilon, \quad \forall t \in [t_1, T],$$

where ϵ is the absolute tolerance error and the number of exponentials N_{exp} satisfies

$$N_{\text{exp}} = O\left(\log \frac{1}{\epsilon} \left(\log \log \frac{1}{\epsilon} + \log \frac{T}{\tau_1} \right) + \log \frac{1}{\tau_1} \left(\log \log \frac{1}{\epsilon} + \log \frac{1}{\tau_1} \right) \right).$$

We remark that the letter F in the difference operator $F\overline{D}_{\tau_n}^\beta$ (see the left-hand side of the Formula (8)) represents the word ‘‘fast’’, which refers to the fact that the corresponding formula uses the SOE technique to improve the computational performance. It is shown in [13] that the number of exponentials $N_{\text{exp}} = O(\log N_t)$ when $T \gg 1$ and $N_{\text{exp}} = O(\log^2 N_t)$ when $T \approx 1$. The total computational cost for the fast L2-1 σ Formula (8) is $O(N_t N_{\text{exp}})$ with storage $O(N_{\text{exp}})$.

By replacing the classical L2-1 σ formula in the numerical scheme (7) with the above fast L2-1 σ formula based on graded meshes, we have the following fast compact difference scheme:

$$\begin{cases} \widehat{\phi}_v^{n+\sigma} + \kappa {}^F\overline{D}_{\tau_n}^{\alpha-1} \widehat{\phi}_v^{n+\sigma} = \mu \left(\lambda^{(j_1, M_1)} + \lambda^{(j_2, M_2)} \right) \widehat{u}_v^{n+\sigma} + \widehat{f}_v^{n+\sigma}, \\ \widetilde{\delta}_t \widehat{u}_v^{1/2} = \widehat{\phi}_v^{1/2}, \\ \widetilde{\delta}_t \widehat{u}_v^n = \widehat{\phi}_v^{n+\sigma}, \end{cases} \tag{10}$$

where $\widetilde{\delta}_t \widehat{u}_v^{1/2} = \frac{\widehat{u}_v^1 - \widehat{u}_v^0}{\tau_1}$ and

$$\widetilde{\delta}_t \widehat{u}_v^n = \frac{\tau_n + 2\sigma\tau_{n+1}}{\tau_n + \tau_{n+1}} D_\tau \widehat{u}_v^{n+1} - \frac{\tau_{n+1}(2\sigma - 1)}{\tau_n + \tau_{n+1}} D_\tau \widehat{u}_v^n. \tag{11}$$

We will simply denote the fast compact difference scheme (10) as the FL2-1 σ (γ) scheme in what follows if no ambiguity arises.

One can observe that, compared with the numerical scheme (6), the FL2-1 σ (γ) scheme (10) reduces the overall computational cost from $O(N_t^2 M_1^2 M_2^2)$ to $O(N_t N_{\text{exp}} M_1 M_2 \log(M_1 M_2))$, which greatly improves the computational efficiency of the original scheme (6).

When $\gamma = 1$, the graded meshes recover the uniform one. Thus, in this case, we shall use τ instead of τ_n in (10) unless otherwise specified. The corresponding scheme based on uniform meshes is called the FL2-1 σ (1) scheme. When $\gamma > 1$, the grid points in the time direction are concentrated near the origin, which is beneficial for the numerical solution to capture the weak singularity solution, thus improving the accuracy loss of the scheme caused by the non-smooth solution. This will be verified by numerical examples in Section 5.

4. Stability and Error Estimates

We studied the stability and error estimates for the FL2-1 σ (1) scheme (10) in this section.

For any given grid function $v \in V_h$, the discrete L^2 norm is given by $\|v\| = \sqrt{(v, v)_h}$ with the discrete inner product $(u, v)_h = (h_1 h_2) \sum_{x_h \in \Omega_h} u_h v_h$. The discrete H^1 semi-norm and H^1 norm are denoted as $|v|_1 = \sqrt{\|\nabla_h v_h\|^2} = \sqrt{\|\delta_1 v_h\|^2 + \|\delta_2 v_h\|^2}$ and $\|v\|_1 = \sqrt{\|v\|^2 + |v|_1^2}$, respectively. Here, $\nabla_h = (\delta_1, \delta_2)$. In light of the embedding theorem, we can conclude that for any given $v \in V_h$, the discrete H^1 semi-norm $|v|_1$ and H^1 norm $\|v\|_1$ are equivalent.

Using the recursive relationship of the historical term $H_k(t_j)$ in Equation (9), one may reformulate the expression of the difference operator ${}^F\overline{D}_\tau^\beta$ based on the uniform meshes as follows (or see [10,13] more details on the derivation):

$${}^F\overline{D}_\tau^\beta g^{n+\sigma} = \sum_{k=0}^n W_{n-k}^{(n+1)} (g(t_{k+1}) - g(t_k)),$$

where $W_{n-k}^{(n+1)}$ are defined as $W_0^{(1)} = w_0^{(1)}$ when $n = 0$, and for $n \geq 1$, we have

$$W_k^{(n+1)} = \begin{cases} A_0 + B_1, & k = 0, \\ A_k + B_{k+1} - B_k, & k = 1, 2, \dots, n - 1, \\ A_n - B_n, & k = n. \end{cases} \tag{12}$$

Here, $A_0 = W_0^{(1)}$, $A_{n-k} = \frac{1}{\tau} \int_{t_k}^{t_{k+1}} \sum_{j=0}^{N_{\text{exp}}} \omega_j e^{-s_j(t_{n+\sigma-s})} ds$, and

$$B_{n-k} = \frac{1}{\tau^2} \int_{t_k}^{t_{k+1}} \sum_{j=0}^{N_{\text{exp}}} \omega_j e^{-s_j(t_{n+\sigma-s})} (s - t_{k+\frac{1}{2}}) ds.$$

We need the following lemma:

Lemma 1. The weights $W_k^{(n+1)}$ in Equation (12) have the following properties:

$$\sum_{k=1}^{m-1} (W_{k-1}^{(k+1)} - W_k^{(k+1)}) \leq \frac{9\epsilon(m-1)}{4\Gamma(1-\beta)} + \frac{4-3\beta}{\tau^\beta \Gamma(3-\beta)} (m-1+\sigma)^{1-\beta},$$

and

$$\sum_{k=1}^{m-1} W_k^{(k+1)} \leq \frac{\epsilon(m-1)}{\Gamma(1-\beta)} + \frac{(m-1+\sigma)^{1-\beta}}{\tau^\beta \Gamma(2-\beta)}.$$

Proof. In light of Lemma 3.3 in [19] and the definitions of $W_k^{(n+1)}$ in Equation (12), one may readily obtain the desired results. \square

Let the symbol c be a positive constant which is independent of the temporal and spatial step sizes. The stability of the scheme (10) is described as follows:

Theorem 1. The FL2-1 $_{\sigma}(\gamma)$ compact difference scheme (10) with $\gamma = 1$ is stable in the sense that

$$\begin{aligned} \|u_h^m\|_1^2 \leq & c \left(\|\phi_h^0\|^2 + \|\nabla_h u_h^0\|^2 \right. \\ & \left. + \tau \sum_{n=1}^{m-1} \|\nabla_h \varepsilon_h^{n+\sigma}\|^2 + \tau \|\nabla_h \varepsilon_h^{\frac{1}{2}}\|^2 + \tau \sum_{n=0}^{m-1} \|f_h^{n+\sigma}\|^2 \right), \end{aligned}$$

where $\varepsilon_h^{n+\sigma}$ and $\varepsilon_h^{\frac{1}{2}}$ are the perturbations of $\widehat{\delta}_t u_h^n = \phi_h^{n+\sigma} + \varepsilon_h^{n+\sigma}$ and $\delta_t u_h^{\frac{1}{2}} = \phi_h^{\frac{1}{2}} + \varepsilon_h^{\frac{1}{2}}$, respectively.

Proof. In light of the equivalence of schemes (6) and (7), one only needs to consider the following scheme instead of (10):

$$\begin{cases} \phi_h^{n+\sigma} + \kappa \overline{D}_\tau^{\alpha-1} \phi_h^{n+\sigma} = \mu \overline{\Delta}_h u_h^{n+\sigma} + f_h^{n+\sigma}, \\ \delta_t u_h^{1/2} = \phi_h^{1/2}, \\ \widehat{\delta}_t u_h^n = \phi_h^{n+\sigma}. \end{cases} \tag{13}$$

We first consider the case $n \geq 1$ for (13). By taking the discrete inner product on both sides of (13) with $\phi_h^{n+\sigma}$, we have

$$(\phi_h^{n+\sigma}, \phi_h^{n+\sigma})_h + (\kappa \overline{D}_\tau^{\alpha-1} \phi_h^{n+\sigma}, \phi_h^{n+\sigma})_h = (\mu \overline{\Delta}_h u_h^{n+\sigma}, \phi_h^{n+\sigma})_h + (f_h^{n+\sigma}, \phi_h^{n+\sigma})_h.$$

We can use Lemma 1 and the estimate for the discrete operator $\overline{\Delta}_h$ presented in (3.14) of [10] such that

$$\frac{3}{2} (\Delta_h u_h^{n+\sigma}, \phi_h^{n+\sigma})_h < (\overline{\Delta}_h u_h^{n+\sigma}, \phi_h^{n+\sigma})_h < (\Delta_h u_h^{n+\sigma}, \phi_h^{n+\sigma})_h, \tag{14}$$

From this, we derive that

$$\|\phi_h^{n+\sigma}\|^2 + \frac{\kappa}{2} \overline{D}_\tau^{\alpha-1} \|\phi_h^{n+\sigma}\|^2 \leq -\mu (\nabla_h u_h^{n+\sigma}, \nabla_h \phi_h^{n+\sigma})_h + (f_h^{n+\sigma}, \phi_h^{n+\sigma})_h. \tag{15}$$

Note that $\widehat{\delta}_t u_h^n = \phi_h^{n+\sigma} + \varepsilon_h^{n+\sigma}$, so it follows from Lemma 3.5 of [20] and the Cauchy–Schwarz inequality that

$$\begin{aligned} -\mu (\nabla_h u_h^{n+\sigma}, \nabla_h \phi_h^{n+\sigma})_h &= -\mu (\nabla_h u_h^{n+\sigma}, \nabla_h \widehat{\delta}_t u_h^n)_h + \mu (\nabla_h u_h^{n+\sigma}, \nabla_h \varepsilon_h^{n+\sigma})_h \\ &\leq -\frac{\mu}{4\tau} (E^{n+1} - E^n) + \mu \|\nabla_h u_h^{n+\sigma}\| \|\nabla_h \varepsilon_h^{n+\sigma}\|, \end{aligned}$$

where $E^{n+1} = (2\sigma + 1) \|\nabla_h u_h^{n+1}\|^2 - (2\sigma - 1) \|\nabla_h u_h^n\|^2 + (2\sigma^2 + \sigma - 1) \|\nabla_h u_h^{n+1} - \nabla_h u_h^n\|^2$.

By using the inequality $ab \leq \epsilon a^2 + \frac{1}{4\epsilon} b^2$ with $\epsilon > 0$, we obtain

$$\|\nabla_h u_h^{n+\sigma}\| \|\nabla_h \epsilon_h^{n+\sigma}\| \leq \frac{1}{4} \|\nabla_h u_h^{n+\sigma}\|^2 + \|\nabla_h \epsilon_h^{n+\sigma}\|^2.$$

Therefore, the inequality (15) has the following estimate:

$$\begin{aligned} & \|\phi_h^{n+\sigma}\|^2 + \frac{\kappa}{2} \overline{D}_\tau^{\alpha-1} \|\phi_h^{n+\sigma}\|^2 + \frac{\mu}{4\tau} E^{n+1} \\ & \leq \frac{\mu}{4\tau} E^n + \frac{\mu}{4} \|\nabla_h u_h^{n+\sigma}\|^2 + \mu \|\nabla_h \epsilon_h^{n+\sigma}\|^2 + (f_h^{n+\sigma}, \phi_h^{n+\sigma})_h, \end{aligned}$$

In other words, we have

$$\begin{aligned} & \|\phi_h^{n+\sigma}\|^2 + \frac{\kappa}{2} W_0^{(n+1)} \|\phi_h^{n+1}\|^2 + \frac{\kappa}{2} \sum_{k=1}^n (W_{n-k+1}^{(n+1)} - W_{n-k}^{(n+1)}) \|\phi_h^k\|^2 + \frac{\mu}{4\tau} E^{n+1} \\ & \leq \frac{\kappa}{2} W_n^{(n+1)} \|\phi_h^0\|^2 + \frac{\mu}{4\tau} E^n + \frac{\mu}{4} \|\nabla_h u_h^{n+\sigma}\|^2 + \mu \|\nabla_h \epsilon_h^{n+\sigma}\|^2 + (f_h^{n+\sigma}, \phi_h^{n+\sigma})_h. \end{aligned}$$

We sum up n from $n = 1$ to $n = m - 1$ for the above inequality and obtain

$$\begin{aligned} & \sum_{n=1}^{m-1} \|\phi_h^{n+\sigma}\|^2 + \frac{\kappa}{2} \sum_{n=1}^{m-1} W_0^{(n+1)} \|\phi_h^{n+1}\|^2 + \frac{\kappa}{2} \sum_{n=1}^{m-1} \sum_{k=1}^n (W_{n-k+1}^{(n+1)} - W_{n-k}^{(n+1)}) \|\phi_h^k\|^2 + \frac{\mu}{4\tau} E^m \\ \leq & \frac{\kappa}{2} \sum_{n=1}^{m-1} W_n^{(n+1)} \|\phi_h^0\|^2 + \frac{\mu}{4\tau} E^1 + \frac{\mu}{4} \sum_{n=1}^{m-1} \|\nabla_h u_h^{n+\sigma}\|^2 \\ & + \mu \sum_{n=1}^{m-1} \|\nabla_h \epsilon_h^{n+\sigma}\|^2 + \sum_{n=1}^{m-1} (f_h^{n+\sigma}, \phi_h^{n+\sigma})_h. \end{aligned} \tag{16}$$

Since

$$\begin{aligned} & \sum_{n=1}^{m-1} \sum_{k=1}^n (W_{n-k+1}^{(n+1)} - W_{n-k}^{(n+1)}) \|\phi_h^k\|^2 \\ = & \sum_{n=1}^{m-1} \left(\sum_{k=2}^n (W_{n-k+1}^{(n+1)} - W_{n-k}^{(n+1)}) \|\phi_h^k\|^2 + (W_n^{(n+1)} - W_{n-1}^{(n+1)}) \|\phi_h^1\|^2 \right) \\ = & \sum_{k=2}^{m-1} \sum_{n=k}^{m-1} (W_{n-k+1}^{(n+1)} - W_{n-k}^{(n+1)}) \|\phi_h^k\|^2 + \sum_{n=1}^{m-1} (W_n^{(n+1)} - W_{n-1}^{(n+1)}) \|\phi_h^1\|^2 \\ = & \sum_{k=2}^{m-1} (W_{m-k}^{(m)} - W_0^{(k+1)}) \|\phi_h^k\|^2 + \sum_{n=1}^{m-1} (W_n^{(n+1)} - W_{n-1}^{(n+1)}) \|\phi_h^1\|^2, \end{aligned}$$

we have

$$\begin{aligned} & \sum_{n=1}^{m-1} W_0^{(n+1)} \|\phi_h^{n+1}\|^2 + \sum_{n=1}^{m-1} \sum_{k=1}^n (W_{n-k+1}^{(n+1)} - W_{n-k}^{(n+1)}) \|\phi_h^k\|^2 \\ = & \sum_{k=2}^m W_{m-k}^{(m)} \|\phi_h^k\|^2 + \sum_{n=1}^{m-1} (W_n^{(n+1)} - W_{n-1}^{(n+1)}) \|\phi_h^1\|^2 \\ \geq & C^f \sum_{k=2}^m \|\phi_h^k\|^2 - \left(\frac{9\epsilon(m-1)}{4\Gamma(1-\beta)} + \frac{4-3\beta}{\tau^\beta \Gamma(3-\beta)} (m-1+\sigma)^{1-\beta} \right) \|\phi_h^1\|^2, \end{aligned}$$

where Lemma 1 and the boundness of $W_{m-k}^{(m)}$ presented in Lemma 4.1 of [13] are used. Here, $C^f = \min\{W_0^{(1)}, W_n^{(n+1)}\}$ and $\beta = \alpha - 1$.

The term E^m has the estimate $E^m \geq \frac{1}{\sigma} \|\nabla_h u_h^m\|^2$ (see Lemma 3.5 of [20]). Thus, one has the further estimate for the inequality (16):

$$\begin{aligned} & \sum_{n=1}^{m-1} \|\phi_h^{n+\sigma}\|^2 + \frac{\kappa C^f}{2} \sum_{k=2}^m \|\phi_h^k\|^2 + \frac{\mu \sigma^{-1}}{4\tau} \|\nabla_h u_h^m\|^2 \\ \leq & \left(\frac{9\epsilon(m-1)}{4\Gamma(1-\beta)} + \frac{4-3\beta}{\tau^\beta \Gamma(3-\beta)} (m-1+\sigma)^{1-\beta} \right) \|\phi_h^1\|^2 \\ & + \left(\frac{\epsilon(m-1)}{\Gamma(1-\beta)} + \frac{(m-1+\sigma)^{1-\beta}}{\tau^\beta \Gamma(2-\beta)} \right) \|\phi_h^0\|^2 + \frac{\mu}{4\tau} E^1 \\ & + \frac{\mu}{4} \sum_{n=1}^{m-1} \|\nabla_h u_h^{n+\sigma}\|^2 + \mu \sum_{n=1}^{m-1} \|\nabla_h \epsilon_h^{n+\sigma}\|^2 + \sum_{n=1}^{m-1} (f_h^{n+\sigma}, \phi_h^{n+\sigma})_h. \end{aligned} \tag{17}$$

Since

$$\begin{aligned} & \sum_{n=1}^{m-1} (f_h^{n+\sigma}, \phi_h^{n+\sigma})_h \\ \leq & \frac{1}{2} \sum_{n=1}^{m-1} \left(\frac{1}{4\epsilon_1} \|f_h^{n+\sigma}\|^2 + \epsilon_1 \|\phi_h^{n+\sigma}\|^2 + \frac{1}{4\epsilon_2} \|f_h^{n+\sigma}\|^2 + \epsilon_2 \|\phi_h^{n+\sigma}\|^2 \right) \\ \leq & c \left(\sum_{n=1}^{m-1} \frac{1}{\epsilon_1} \|f_h^{n+\sigma}\|^2 + \epsilon_1 \sum_{n=1}^{m-1} \|\phi_h^{n+\sigma}\|^2 + \epsilon_2 \sum_{n=2}^m \|\phi_h^n\|^2 + \|\phi_h^1\|^2 \right), \end{aligned}$$

by choosing suitable ϵ_1 and ϵ_2 values, the inequality (17) yields

$$\begin{aligned} \|\nabla_h u_h^m\|^2 \leq & c \left(\|\phi_h^0\|^2 + \|\phi_h^1\|^2 + \|\nabla_h u_h^0\|^2 + \|\nabla_h u_h^1\|^2 \right. \\ & \left. + \tau \sum_{n=1}^{m-1} \|\nabla_h u_h^n\|^2 + \tau \sum_{n=1}^{m-1} \|\nabla_h \epsilon_h^{n+\sigma}\|^2 + \tau \sum_{n=1}^{m-1} \|f_h^{n+\sigma}\|^2 \right). \end{aligned} \tag{18}$$

Next, we consider the bound of $\|\phi_h^1\|^2 + \|\nabla_h u_h^1\|^2$ on the right-hand side of (18). To this end, we operate the discrete inner product on both sides of the numerical scheme (6) with $\phi_h^{\frac{1}{2}}$ for $n = 0$ and find

$$(\phi_h^\sigma, \phi_h^{\frac{1}{2}})_h + \kappa ({}^F D_\tau^{\alpha-1} \phi_h^\sigma, \phi_h^{\frac{1}{2}})_h = \mu (\bar{\Delta}_h u_h^\sigma, \phi_h^{\frac{1}{2}})_h + (f_h^\sigma, \phi_h^{\frac{1}{2}})_h.$$

Observing that ${}^F D_\tau^{\alpha-1} \phi_h^\sigma = W_0^{(1)} (\phi_h^1 - \phi_h^0)$, and using the equality $\delta_t u_h^{\frac{1}{2}} = \phi_h^{\frac{1}{2}} + \epsilon_h^{\frac{1}{2}}$, we obtain

$$(\phi_h^\sigma, \phi_h^{\frac{1}{2}})_h + \frac{\kappa W_0^1}{2} (\|\phi_h^1\|^2 - \|\phi_h^0\|^2) = -\mu (\nabla_h u_h^\sigma, \nabla_h \delta_t u_h^{\frac{1}{2}})_h + \mu (\nabla_h u_h^\sigma, \nabla_h \epsilon_h^{\frac{1}{2}})_h + (f_h^\sigma, \phi_h^{\frac{1}{2}})_h. \tag{19}$$

Following the idea in the proof of Theorem 4.1 of [6] for the case $n = 0$, we further have

$$\begin{aligned} (\phi_h^\sigma, \phi_h^{\frac{1}{2}})_h &= \frac{1}{2} (\sigma \phi_h^1 + (1-\sigma) \phi_h^0, \phi_h^1 + \phi_h^0)_h \\ &= \frac{1}{2} (\sigma \|\phi_h^1\|^2 + (1-\sigma) \|\phi_h^0\|^2 + (\phi_h^0, \phi_h^1)_h) \\ &\geq \frac{1}{2} (\sigma \|\phi_h^1\|^2 + (1-\sigma) \|\phi_h^0\|^2) - \frac{1}{2} (\epsilon_1 \|\phi_h^1\|^2 + \frac{1}{4\epsilon_1} \|\phi_h^0\|^2) \\ &= \frac{1}{4} \|\phi_h^1\|^2 + \frac{1}{2} \left[1 - \sigma - \frac{1}{4(\sigma - 0.5)} \right] \|\phi_h^0\|^2, \end{aligned}$$

where we choose $\epsilon_1 = \sigma - \frac{1}{2} > 0$.

For the first term on the right-hand side of Equation (19), we have

$$\begin{aligned}
 & -\mu(\nabla_h u_h^\sigma, \nabla_h \delta_t u_h^{\frac{1}{2}})_h \\
 = & -\frac{\mu}{\tau}(\sigma \nabla_h u_h^1 + (1 - \sigma) \nabla_h u_h^0, \nabla_h u_h^1 - \nabla_h u_h^0)_h \\
 = & -\frac{\mu}{\tau}(\sigma \|\nabla_h u_h^1\|^2 - (1 - \sigma) \|\nabla_h u_h^0\|^2) + \frac{\mu}{\tau}(2\sigma - 1)(\nabla_h u_h^0, \nabla_h u_h^1)_h \\
 \leq & -\frac{\mu}{\tau}(\sigma \|\nabla_h u_h^1\|^2 - (1 - \sigma) \|\nabla_h u_h^0\|^2) + \frac{\mu}{\tau}(2\sigma - 1)\left(\epsilon_2 \|\nabla_h u_h^1\|^2 + \frac{1}{4\epsilon_2} \|\nabla_h u_h^0\|^2\right) \\
 = & -\frac{1}{2\tau} \|\nabla_h u_h^1\|^2 + \frac{\mu}{\tau} \left(1 - \sigma + \frac{(2\sigma - 1)^2}{4\mu(\sigma - 0.5)}\right) \|\nabla_h u_h^0\|^2,
 \end{aligned}$$

where we let $\epsilon_2 = \frac{\sigma - 0.5}{2\sigma - 1} > 0$.

The second term on the right-hand side of Equation (19) has the following estimate:

$$\begin{aligned}
 & \mu(\nabla_h u_h^\sigma, \nabla_h \epsilon_h^{\frac{1}{2}})_h \\
 = & \mu(\sigma \nabla_h u_h^1 + (1 - \sigma) \nabla_h u_h^0, \nabla_h \epsilon_h^{\frac{1}{2}})_h \\
 \leq & \mu(\sigma \|\nabla_h u_h^1\|^2 + (1 - \sigma) \|\nabla_h u_h^0\|^2) \|\nabla_h \epsilon_h^{\frac{1}{2}}\|^2 \\
 \leq & \mu\sigma\epsilon_3 \|\nabla_h u_h^1\|^2 + \mu\sigma \frac{1}{4\epsilon_3} \|\nabla_h \epsilon_h^{\frac{1}{2}}\|^2 + \mu(1 - \sigma)\epsilon_4 \|\nabla_h u_h^0\|^2 + \mu(1 - \sigma) \frac{1}{4\epsilon_4} \|\nabla_h \epsilon_h^{\frac{1}{2}}\|^2.
 \end{aligned}$$

By suitably choosing the parameters ϵ_3 and ϵ_4 (e.g., let $\epsilon_3 = \frac{1}{\sigma}(\frac{1}{2\tau} - 1) > 0$ and $\epsilon_4 = \frac{1 - \sigma}{4} > 0$), and noting that $W_0^{(1)} = w_0^{(1)} = \frac{t_\sigma^{1-\beta}}{\tau\Gamma(2-\beta)}$, we can derive that

$$\|\phi_h^1\|^2 + \|\nabla_h u_h^0\|^2 \leq c \left(\|\phi_h^0\|^2 + \|\nabla_h u_h^0\|^2 + \tau \|\nabla_h u_h^0\|^2 + \tau \|f_h^\sigma\|^2 + \tau \|\nabla_h \epsilon_h^{\frac{1}{2}}\|^2 \right).$$

This together with (18) leads to

$$\begin{aligned}
 \|\nabla_h u_h^m\|^2 \leq & c \left(\|\phi_h^0\|^2 + \|\nabla_h u_h^0\|^2 \right. \\
 & \left. + \tau \sum_{n=0}^{m-1} \|\nabla_h u_h^n\|^2 + \tau \sum_{n=1}^{m-1} \|\nabla_h \epsilon_h^{n+\sigma}\|^2 + \tau \|\nabla_h \epsilon_h^{\frac{1}{2}}\|^2 + \tau \sum_{n=0}^{m-1} \|f_h^{n+\sigma}\|^2 \right).
 \end{aligned}$$

By applying the Gronwall inequality to the above estimate and the equivalence of the H^1 norm and H^1 semi-norm, one has the desired result. All of this ends the proof. \square

Finally, we present the convergence result for the FL2-1 $_\sigma$ (1) scheme (10). Let $e_h^n = u(x_h, t_n) - u_h^n$ and $\bar{e}_h^n = \phi_h(x_h, t_n) - \phi_h^n$. From Lemma 2.3 of [13], we have

$$cD_{0,t}^\beta g(t)|_{t=t_{n+\sigma}} = {}^F\bar{D}_\tau^\beta g^{n+\sigma} + O(\tau^{3-\beta} + \epsilon).$$

Therefore, in light of (2), (6), and (13), one may find the following error equations:

$$\begin{cases} \bar{e}_h^{n+\sigma} + \kappa {}^F\bar{D}_\tau^{\alpha-1} \bar{e}_h^{n+\sigma} = \mu \bar{\Delta}_h e_h^{n+\sigma} + \tilde{R}_h^{n+\sigma}, \\ \delta_t e_h^{1/2} = \bar{e}_h^{1/2} + \tilde{r}_h^{1/2}, \\ \widehat{\delta}_t e_h^n = \bar{e}_h^{n+\sigma} + \tilde{r}_h^{n+\sigma}, \end{cases}$$

where $\tilde{R}_h^{n+\sigma} = O(\tau^2 + \mathbf{h}^4 + \epsilon)$, $\tilde{r}_h^{1/2} = O(\tau^2)$, and $\tilde{r}_h^{n+\sigma} = O(\tau^2)$. By applying the stability result of Theorem 1, we obtain the error bound in the H^1 norm as follows:

$$\begin{aligned} \|e_h^n\|_1^2 &\leq c \left(\|\bar{e}_h^0\|^2 + \|\nabla_h e_h^0\|^2 \right. \\ &\quad \left. + \tau \sum_{n=1}^{m-1} \|\nabla_h \tilde{r}_h^{n+\sigma}\|^2 + \tau \|\nabla_h \tilde{r}_h^{1/2}\|^2 + \tau \sum_{n=0}^{m-1} \|\tilde{R}_h^{n+\sigma}\|^2 \right) \\ &\leq c(\tau^2 + \mathbf{h}^4 + \epsilon)^2. \end{aligned}$$

The above estimate leads to the following error estimate of the FL2- $1_\sigma(1)$ scheme (10). The corresponding proof is thus omitted:

Theorem 2. Let $u(\mathbf{x}_h, t_n)$ and u_h^n be the solutions to Equation (1) and numerical scheme (10), respectively. Suppose the solution $u(\mathbf{x}, t)$ belongs to the function space $C_{\mathbf{x},t}^{6,3}(\Omega \times [0, T])$. Then, for $\gamma = 1$, we have

$$\|(u(\mathbf{x}_h, t_n) - u_h^n)\|_1^2 \leq c(\tau^2 + \mathbf{h}^4 + \epsilon).$$

Remark 1. In practice, the absolute tolerance error ϵ in the scheme (10) is always set to a very small number so as not to affect the temporal or spatial convergence accuracy. In this sense, the convergence result in Theorem 2 can be understood as $O(\tau^2 + \mathbf{h}^4)$.

Remark 2. Although the nonuniform mesh-based FL2- $1_\sigma(\gamma)$ scheme (i.e., $\gamma > 1$) can effectively handle non-smooth solution problems, its stability and error estimates are more difficult. In [10], Li et al. successfully gave a rigorous proof of the stability and convergence of the FL2- 1_σ scheme based on the nonuniform meshes, but their derived scheme mainly deals with the subdiffusion problem. It seems that the application of their ideas here is not obvious, and further investigation is needed.

5. Numerical Examples

Two numerical examples will be presented to verify the convergence theoretical results and efficiency of the FL2- 1_σ scheme (10). In the numerical examples, the computational domain is $\Omega = (0, 1)^2$, and the parameters κ and μ are both one. Using the equivalence of the H^1 norm and H^1 semi-norm, we calculated the H^1 norm error at $t = t_n$ by $e(n, h) = |u(\mathbf{x}_h, t_n) - u_h^n|_1$. The corresponding convergence orders in time and in space were obtained by $\log(e(n, h)/e(2n, h))$ and $\log(e(n, h)/e(n, h/2))$, respectively. We always set the absolute tolerance error $\epsilon = 10^{-10}$ in (10) so that it did not contaminate the temporal or spatial convergence orders.

Example 1 (Accuracy). Consider the following problem with zero Dirichlet boundary conditions:

$$\begin{cases} \partial_t u(x, y, t) + {}_C D_{0,t}^\alpha u(x, y, t) = \Delta u(x, y, t) + f(x, y, t), & (x, y, t) \in \Omega \times (0, 1], \\ u(x, y, 0) = \sin(\pi x) \sin(\pi y), & \partial_t u(x, y, 0) = 0, \end{cases}$$

where

$$f(x, y, t) = \sin(\pi x) \sin(\pi y) \left(\eta t^{\eta-1} + 2\pi^2(1+t^\eta) + \frac{\Gamma(\eta+1)}{\Gamma(\eta+1-\alpha)} t^{\eta-\alpha} \right).$$

The exact solution is $u = (1+t^\eta) \sin(\pi x) \sin(\pi y)$ with the fixed parameter $\eta > 1$.

We considered two cases, $\eta = 3.5$ and $\eta = 1.5$, to verify the accuracy of the FL2- $1_\sigma(\gamma)$ scheme (10). For these two cases, we applied the uniform mesh-based and nonuniform mesh-based FL2- $1_\sigma(\gamma)$ scheme (10), respectively, to obtain the numerical results. See Tables 1 and 2 for the first case and Tables 3 and 4 for the second case. We remark that the H^1 norm errors here and below were obtained at the final time T .

Table 1. The H^1 -norm errors in time for the smooth case in Example 1 with $h = 1/64$.

N_t	$\alpha = 1.1$		$\alpha = 1.5$		$\alpha = 1.9$	
	H^1 Error	Rate	H^1 Error	Rate	H^1 Error	Rate
40	2.66×10^{-4}	-	6.73×10^{-4}	-	1.29×10^{-3}	-
80	6.69×10^{-5}	1.99	1.69×10^{-4}	2.00	3.23×10^{-4}	2.00
160	1.68×10^{-5}	1.99	4.22×10^{-5}	2.00	8.07×10^{-5}	2.00
320	4.27×10^{-6}	1.98	1.05×10^{-5}	2.01	2.01×10^{-5}	2.00

Table 2. The H^1 -norm errors in space for the smooth case in Example 1 with $\tau = 1/4000$.

M	$\alpha = 1.1$		$\alpha = 1.5$		$\alpha = 1.9$	
	H^1 Error	Rate	H^1 Error	Rate	H^1 Error	Rate
4	2.31×10^{-3}	-	2.22×10^{-3}	-	2.15×10^{-3}	-
8	2.72×10^{-4}	3.09	2.62×10^{-4}	3.09	2.53×10^{-4}	3.09
16	2.05×10^{-5}	3.73	1.96×10^{-5}	3.74	1.89×10^{-5}	3.74
32	1.38×10^{-6}	3.89	1.26×10^{-6}	3.96	1.16×10^{-6}	4.03

Table 3. The H^1 -norm errors in time for the non-smooth case in Example 1 with $h = 1/64$ and $\alpha = 1.5$.

N_t	$\gamma = 1$		$\gamma = 1.5$		$\gamma = 2$	
	H^1 Error	Rate	H^1 Error	Rate	H^1 Error	Rate
40	5.86×10^{-5}	-	1.85×10^{-4}	-	3.70×10^{-4}	-
80	3.98×10^{-5}	0.56	4.42×10^{-5}	2.06	9.28×10^{-5}	2.00
160	2.22×10^{-5}	0.84	1.03×10^{-5}	2.10	2.31×10^{-5}	2.00
320	1.15×10^{-5}	0.94	2.27×10^{-6}	2.18	5.52×10^{-6}	2.07

Table 4. The H^1 -norm errors in space for the non-smooth case in Example 1 with $\tau = 1/1000$ and $\alpha = 1.5$.

M	$\gamma = 1$		$\gamma = 1.5$		$\gamma = 2$	
	H^1 Error	Rate	H^1 Error	Rate	H^1 Error	Rate
4	2.52×10^{-3}	-	2.52×10^{-3}	-	2.52×10^{-3}	-
8	3.00×10^{-4}	3.07	2.97×10^{-4}	3.09	2.96×10^{-4}	3.09
16	2.56×10^{-5}	3.55	2.22×10^{-5}	3.74	2.06×10^{-5}	3.84
32	5.05×10^{-6}	2.34	1.33×10^{-6}	4.07	3.41×10^{-7}	5.92

When $\eta = 3.5$, the solution of the equation satisfied the smoothness requirement of Theorem 2. From the numerical results in Tables 1 and 2, it can be observed that the accuracy of the scheme (10) was $O(\tau^2 + h^4)$ for different fractional orders α , which is consistent with the theoretical result of the convergence accuracy.

When $\eta = 1.5$, the second-order partial derivative of u with respect to t was $\partial_t^2 u(x, y, t) = \eta(\eta - 1)t^{\eta-2} \sin(\pi x) \sin(\pi y)$, which was unbounded at $t = 0$ when $\eta < 2$. Therefore, the solution was not sufficiently smooth in this case. In order to capture the weak singularity solution at the initial point, we let the temporal mesh grading parameter be $\gamma > 1$. From the numerical results in Tables 3 and 4, one can see that by selecting a different γ value (i.e., $\gamma = 1.5$ and $\gamma = 2$), the theoretical convergence of the scheme (10) was restored, which shows that the scheme (10) based on graded meshes indeed captured the weak singularity solution and could significantly improve the convergence accuracy.

Example 2 (Computational efficiency). *In this example, we studied the computational performance for the FL2-1 σ scheme (10). For the sake of simplicity, the case $\eta = 3.5$ in the above example was used. To better reveal that the scheme (10) had improved computational efficiency, we compared it with the two schemes (6) and (7). The numerical results are demonstrated in Table 5 and Figure 1.*

Table 5. Comparison of CPU time of three schemes (6), (7), and (10) for Example 1 with $h = 1/64$ and $\tau = 0.004/2^k$.

α	Scheme	$k = 1$		$k = 2$		$k = 3$	
		H^1 Error	CPU (s)	H^1 Error	CPU (s)	H^1 Error	CPU (s)
1.1	(6)	1.66×10^{-6}	3.86	3.69×10^{-7}	9.85	4.78×10^{-8}	28.03
	(7)	1.80×10^{-6}	1.59	5.18×10^{-7}	5.50	1.96×10^{-7}	19.56
	(10)	1.81×10^{-6}	1.03	5.04×10^{-7}	1.99	2.08×10^{-7}	3.98
1.5	(6)	4.39×10^{-6}	3.67	1.14×10^{-6}	9.59	3.28×10^{-7}	27.68
	(7)	4.25×10^{-6}	1.43	9.97×10^{-7}	5.23	1.85×10^{-7}	19.22
	(10)	4.25×10^{-6}	0.94	9.97×10^{-7}	1.96	1.85×10^{-7}	3.98
1.9	(6)	8.33×10^{-6}	3.77	2.12×10^{-6}	9.77	5.72×10^{-7}	28.13
	(7)	8.19×10^{-6}	1.56	1.98×10^{-6}	5.38	4.34×10^{-7}	19.66
	(10)	8.19×10^{-6}	0.97	1.98×10^{-6}	1.98	4.34×10^{-7}	4.07

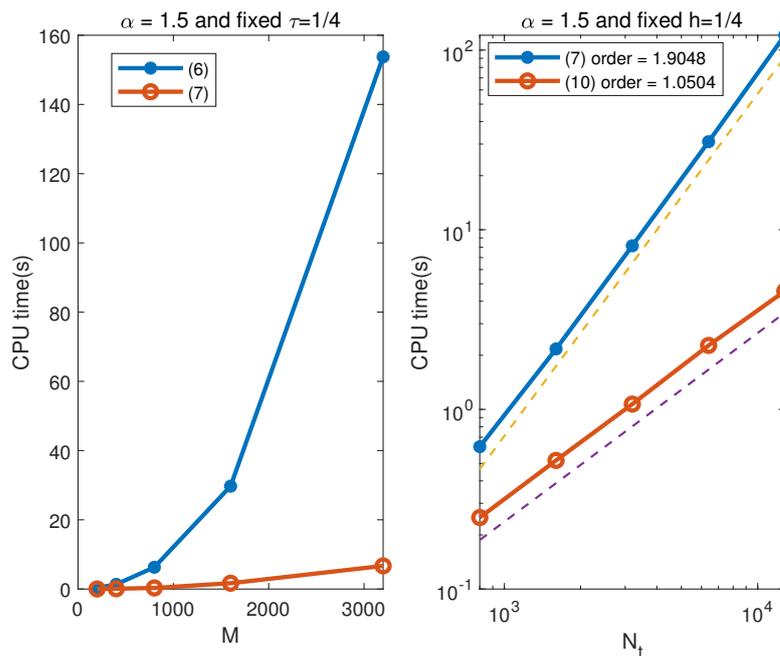


Figure 1. Comparison of computational costs with fixed $\alpha = 1.5$. (The dashed lines below the numerical curves are the corresponding fitted curves, which are shifted down here for the convenience of observation.)

From Table 5, we can observe that when the spatial step size h was fixed, the accuracy of all three schemes increased as the number of temporal nodes increased. However, their computational cost also increased. With the fixed same number of spatial and temporal nodes, the accuracies of these three schemes were almost the same, but the scheme (6) based on the direct solver took the most time, followed by the scheme (7), while the FL2-1 σ (γ) scheme (10) took the least time, and this time consumption gap increased with the increase in the number of temporal nodes.

For the case where $\alpha = 1.5$, when the time step size τ was fixed to $1/4$, and the spatial grid became denser, similar phenomena to those in Table 5 could be observed in the left subplot of Figure 1. In particular, we compared the computational efficiency of the two numerical schemes (7) and (10) by

plotting the CPU time versus N_t in a log-log style. In order to numerically quantify the complexity of these two schemes, we expressed the CPU time in the form of N_t^r using least squares fitting and reported the corresponding order r in the legend of the right subplot of Figure 1. One can see that the order r of the FL2-1 $_{\sigma}(\gamma)$ scheme (10) was about one, while that of the DST-based scheme (7) was near two. This indicates that the proposed scheme (10) based on the DST and SOE techniques is more competitive in solving high-dimensional time-fractional problems. For other cases, such as $\alpha = 1.1$ and 1.9, similar numerical results were also obtained, and they are not presented here for the sake of brevity.

6. Conclusions

In this paper, we proposed a fast compact difference scheme with $O(\tau^2 + h^4)$ convergence accuracy to solve the two-dimensional time-fractional Cattaneo Equation (1). This scheme is based on the DST and SOE techniques and has the ability of fast calculation in both time and space. The stability and error estimate of the numerical scheme (10) based on uniform meshes are presented rigorously. Numerical examples show that this scheme is efficient and suitable for numerically solving high-dimensional time-fractional problems.

We remark that the results obtained in this paper can easily be generalized to other high-dimensional problems. Aside from that, one may notice that although numerical experiments have shown that the scheme (10) based on graded meshes can improve the accuracy of solving high-dimensional non-smooth solution problems, the analysis of its corresponding numerical theory is not an easy job. Such an issue deserves further study and will be one of our upcoming research directions.

We note that the scheme presented in this paper is only for linear problems with constant coefficients, while in practical problems, the equations are often accompanied by variable coefficients or nonlinear terms. In the following work, we will try to extend the methods of this paper to solve variable coefficients and nonlinear problems.

Author Contributions: Methodology, L.N., Q.Y., J.C. and A.C.; validation, L.N., Q.Y., J.C. and A.C.; formal analysis, L.N., Q.Y., J.C. and A.C.; investigation, L.N., Q.Y., J.C. and A.C.; writing—original draft preparation, L.N., Q.Y., J.C. and A.C.; writing—review and editing, L.N., Q.Y., J.C. and A.C.; funding acquisition, L.N., Q.Y., J.C. and A.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Guangxi Natural Science Foundation grant numbers 2021GXNSFBA196027, 2020GXNSFBA297121, 2018GXNSFBA281020, and 2018GXNSFAA138121, National Natural Science Foundation of China grant number 11901266, and Gansu Natural Science Foundation grant number 21JR7RA253.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data of the numerical simulation used to support the findings of this study are included within the paper.

Acknowledgments: The authors wish to express their appreciation to the reviewers for their valuable suggestions, which greatly improved the presentation of this paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Chen, A.; Nong, L. Efficient Galerkin finite element methods for a time-fractional Cattaneo equation. *Adv. Differ. Equ.* **2020**, *2020*, 545. [CrossRef]
2. Awad, E.; Metzler, R. Crossover dynamics from superdiffusion to subdiffusion: Models and solutions. *Fract. Calc. Appl. Anal.* **2020**, *23*, 55–102. [CrossRef]
3. Compte, A.; Metzler, R. The generalized Cattaneo equation for the description of anomalous transport processes. *J. Phys. Math. Gen.* **1997**, *30*, 7277–7289. [CrossRef]
4. Zhao, X.; Sun, Z. Compact Crank–Nicolson Schemes for a Class of Fractional Cattaneo Equation in Inhomogeneous Medium. *J. Sci. Comput.* **2015**, *62*, 747–771. [CrossRef]

5. Ren, J.; Gao, G. Efficient and stable numerical methods for the two-dimensional fractional Cattaneo equation. *Numer. Algorithms* **2015**, *69*, 795–818. [[CrossRef](#)]
6. Chen, A.; Li, C. An alternating direction Galerkin method for a time-fractional partial differential equation with damping in two space dimensions. *Adv. Differ. Equ.* **2017**, *2017*, 356. [[CrossRef](#)]
7. Li, C.; Chen, A. Numerical methods for fractional partial differential equations. *Int. J. Comput. Methods* **2018**, *95*, 1048–1099. [[CrossRef](#)]
8. Jin, B.; Lazarov, R.; Zhou, Z. Numerical methods for time-fractional evolution equations with nonsmooth data: A concise overview. *Comput. Methods Appl. Mech. Eng.* **2019**, *346*, 332–358. [[CrossRef](#)]
9. Li, C.; Zeng, F. *Numerical Methods for Fractional Calculus*; Chapman and Hall/CRC: Boca Raton, FL, USA, 2015.
10. Li, X.; Liao, H.; Zhang, L. A second-order fast compact scheme with unequal time-steps for subdiffusion problems. *Numer. Algorithms* **2021**, *86*, 1011–1039. [[CrossRef](#)]
11. Zeng, F.; Turner, I.; Burrage, K. A stable fast time-stepping method for fractional integral and derivative operators. *J. Sci. Comput.* **2018**, *77*, 283–307. [[CrossRef](#)]
12. Yin, B.; Liu, Y.; Li, H.; Zeng, F. A class of efficient time-stepping methods for multi-term time-fractional reaction-diffusion-wave equations. *Appl. Numer. Math.* **2021**, *165*, 56–82. [[CrossRef](#)]
13. Yan, Y.; Sun, Z.; Zhang, J. Fast evaluation of the Caputo fractional derivative and its applications to fractional diffusion equations: A second-order scheme. *Commun. Comput. Phys.* **2017**, *22*, 1028–1048. [[CrossRef](#)]
14. Liao, H.; Tang, T.; Zhou, T. A second-order and nonuniform time-stepping maximum-principle preserving scheme for time-fractional Allen-Cahn equations. *J. Comput. Phys.* **2020**, *414*, 109473. [[CrossRef](#)]
15. Wang, H.; Zhang, Y.; Ma, X.; Qiu, J.; Liang, Y. An efficient implementation of fourth-order compact finite difference scheme for Poisson equation with Dirichlet boundary conditions. *Comput. Math. Appl.* **2016**, *71*, 1843–1860. [[CrossRef](#)]
16. Ren, Y.; Feng, H.; Zhao, S. A FFT accelerated high order finite difference method for elliptic boundary value problems over irregular domains. *J. Comput. Phys.* **2022**, *448*, 110762. [[CrossRef](#)]
17. Nong, L.; Chen, A.; Yi, Q.; Li, C. Fast Crank-Nicolson compact difference scheme for the two-dimensional time-fractional mobile/immobile transport equation. *AIMS Math.* **2021**, *6*, 6242–6254. [[CrossRef](#)]
18. Nong, L.; Chen, A. Fast high-order difference scheme for the modified anomalous subdiffusion equation based on fast discrete Sine transform. *J. Funct. Spaces* **2021**, *2021*, 9918955. [[CrossRef](#)]
19. Liang, Y.; Yao, Z.; Wang, Z. Fast high order difference schemes for the time fractional telegraph equation. *Numer. Methods Partial Differ. Equ.* **2020**, *36*, 154–172. [[CrossRef](#)]
20. Sun, H.; Sun, Z.; Gao, G. Some temporal second order difference schemes for fractional wave equations. *Numer. Methods Partial Differ. Equ.* **2016**, *32*, 970–1001. [[CrossRef](#)]
21. Alikhanov, A.A. A new difference scheme for the time fractional diffusion equation. *J. Comput. Phys.* **2015**, *280*, 424–438. [[CrossRef](#)]