



Article

A Bound-Preserving Numerical Scheme for Space–Time Fractional Advection Equations

Jing Gao  and Huaiguang Chen *

School of Science, Shandong Jianzhu University, Jinan 250101, China; gaojing20@sdjzu.edu.cn

* Correspondence: chenhuaguang20@sdjzu.edu.cn or chgy1w@163.com

Abstract: We develop and analyze an explicit finite difference scheme that satisfies a bound-preserving principle for space–time fractional advection equations with the orders of $0 < \alpha$ and $\beta \leq 1$. The stability (and convergence) of the method is discussed. Due to the nonlocal property of the fractional operators, the numerical method generates dense coefficient matrices with complex structures. In order to increase the effectiveness of the method, we use Toeplitz-like structures in the full coefficient matrix in a sparse form to reduce the costs of computation, and we also apply a fast evaluation method for the time–fractional derivative. Therefore, an efficient solver is constructed. Numerical experiments are provided for the utility of the method.

Keywords: finite difference scheme; space–time fractional advection equation; bound-preserving; stability



Citation: Gao, J.; Chen, H. A Bound-Preserving Numerical Scheme for Space–Time Fractional Advection Equations. *Fractal Fract.* **2024**, *8*, 89. <https://doi.org/10.3390/fractalfract8020089>

Academic Editors: Haci Mehmet Baskonus, Xian-Ming Gu, Hongbin Chen, Shulin Wu and Xiangcheng Zheng

Received: 21 December 2023

Revised: 22 January 2024

Accepted: 26 January 2024

Published: 30 January 2024



Copyright: © 2024 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

It has been shown that fractional partial differential equations (FPDEs) [1,2] provide a powerful way through which to simulate challenging phenomena—including long-range spatial interactions [3–7] and anomalously diffusive transport [8–11] and memory effects [12]—and they have attracted extensive research [13–18]. The majority of the works in the literature have focused on FPDEs that model anomalously diffusive transport, including both subdiffusive transport and superdiffusive transport, or a combination of both [19–23]. Here, we discuss space–time fractional advection equations (stFAEs) in two space dimensions:

$$\begin{aligned} \partial_t^\alpha u + v_x [\kappa_x \partial_x^{\beta_{x,-}} u + (1 - \kappa_x) \partial_x^{\beta_{x,+}} u] + v_y [\kappa_y \partial_y^{\beta_{y,-}} u + (1 - \kappa_y) \partial_y^{\beta_{y,+}} u] \\ = f, \quad \text{in } \Omega \times (0, T], \\ u(x, y, 0) = u_0(x, y), \quad (x, y) \in \bar{\Omega}, \\ u(x, y, t) = 0, \quad (x, y, t) \in \partial\Omega \times [0, T]. \end{aligned} \quad (1)$$

Here, v_x and v_y are the components of the velocity field along the coordinate directions, which are assumed to be non-negative constants for simplicity of exposition. $0 \leq \kappa_x, \kappa_y \leq 1$ represent the relative weights of the forward versus backward probabilities of the underlying particle jumps along coordinate directions, respectively. The rectangular domain $\Omega := (x_d, x_u) \times (y_d, y_u)$, and fractional derivatives $\partial_x^{\gamma,\pm} u$ of order $0 < \gamma \leq 1$ are defined by [2]

$$\begin{aligned}\partial_x^{\gamma,-} u &:= \begin{cases} {}_{x_d}I_x^{1-\gamma} \partial_x u, & \gamma \in (0,1), \\ \partial_x u, & \gamma = 1, \end{cases} \\ \partial_x^{\gamma,+} u &:= \begin{cases} -{}_xI_{x_u}^{1-\gamma} \partial_x u, & \gamma \in (0,1), \\ -\partial_x u, & \gamma = 1, \end{cases} \\ {}_aI_x^\gamma g(x) &:= \frac{1}{\Gamma(\gamma)} \int_a^x \frac{g(z)}{(x-z)^{1-\gamma}} dz, \quad {}_xI_b^\gamma g(x) := \frac{1}{\Gamma(\gamma)} \int_x^b \frac{g(z)}{(z-x)^{1-\gamma}} dz.\end{aligned}\quad (2)$$

If $\alpha = 1$, then Equation (1) reduces to a space-fractional advection equation (sFAE)

$$\begin{aligned}\partial_t u + v_x [\kappa_x \partial_x^{\beta_{x,-}} u + (1 - \kappa_x) \partial_x^{\beta_{x,+}} u] + v_y [\kappa_y \partial_y^{\beta_{y,-}} u + (1 - \kappa_y) \partial_y^{\beta_{y,+}} u] \\ = f, \quad \text{in } \Omega \times (0, T], \\ u(x, y, 0) = u_0(x, y), \quad (x, y) \in \Omega, \\ u(x, y, t) = 0, \quad (x, y, t) \in \partial\Omega \times (0, T].\end{aligned}\quad (3)$$

To understand the properties of the governing equations, we begin with the problem (3) in one-dimension and drop the subscript x as follows:

$$\begin{aligned}\partial_t u + v [\kappa \partial_x^{\beta_{x,-}} u + (1 - \kappa) \partial_x^{\beta_{x,+}} u] = f(x, t), \quad (x, t) \in (x_d, x_u) \times (0, T], \\ u(x, 0) = u_0(x), \quad x \in [x_d, x_u], \quad u(x_d, t) = u(x_u, t) = 0, \quad t \in [0, T].\end{aligned}\quad (4)$$

We consider the problem (4) with $(x_d, x_u) = (0, 1)$, $v = 1$, $\kappa = 1$ and f being constant.

$$\partial_x^{\beta_{x,-}} u = {}_0I_x^{1-\beta} \partial_x u = f, \quad x \in (0, 1). \quad (5)$$

We utilized the semigroup property of ${}_0I_x^\beta$ [2] and the formula

$${}_0I_x^\gamma x^\mu = \frac{\Gamma(\mu+1)}{\Gamma(\gamma+\mu+1)} x^{\gamma+\mu}, \quad 0 < \gamma < 1, \quad \mu > -1 \quad (6)$$

to obtain a closed-form expression of the solution u as

$$u(x) - u(0) = {}_0I_x \partial_x u = {}_0I_x^\beta {}_0I_x^{1-\beta} \partial_x u = {}_0I_x^\beta f = \frac{fx^{1-\beta}}{\Gamma(2-\beta)}. \quad (7)$$

To uniquely determine a solution u to problem (5), one has to enforce a boundary condition $u(0)$ for Problem (5). In particular, $u(x) = u(0)$ for $f \equiv 0$ shows the advective nature of the problem.

We can similarly find a closed-form expression for the one-sided fractional diffusion equation of an order $1 < \gamma < 2$

$$\partial_x^{\gamma,-} u = {}_0I_x^{2-\gamma} \partial_{xx} u = 1, \quad x \in (0, 1) \quad (8)$$

as follows:

$$u(x) = \frac{x^\gamma}{\gamma(\gamma-1)} + C_1 x + C_2, \quad x \in (0, 1). \quad (9)$$

To uniquely determine a solution for Problem (8), one has to enforce the boundary condition for $u(0)$ and $u(1)$.

This partially explains why an implicit finite-difference approximation to Problem (8) is obtained by directly truncating the fractional derivative, which is unstable, as proved in [24]. The reason for this is that the numerical approximation is uniquely determined by only the left-sided boundary condition, while the solution to the continuous Problem (9)

can only be uniquely determined with the boundary condition at both of the endpoints of the interval.

We observed from (7) that the solution to the left-sided FAE (5) can be uniquely determined by the boundary condition at $x = x_d$. Via symmetry, we anticipated that the solution to the right-sided analog of Problem (5) can be uniquely determined by the right-end boundary condition. Consequently, the two-sided Problem (4) needed to be closed by boundary conditions at both the endpoints of the interval (x_d, x_u) . Unlike fractional diffusion problems, the two boundary conditions in the two-sided FAE Problem (4) work against each other in the sense that the two fronts generated by the two boundary conditions move toward each other as in the turning point problems in integer-order, advection–diffusion equations (which always produce numerical approximations with spurious oscillations, under and over shoot, as well as produce other numerical difficulties [25]).

Inspired by these considerations, we followed the ideas from [26–30] to develop a bound-preserving numerical method for fractional advection Equations (1) and (3), so that the resulting numerical approximations could remove the spurious numerical oscillations, as well as the under and over shoot. As we shall see, Problems (1) and (3) exhibited a combination of an advective and diffusive nature. This required further numerical and mathematical analysis. For instance, diffusion-dominated problems typically prefer an implicit temporal discretization, while advection-dominated problems prefer an explicit temporal discretization. We intended to develop an explicit numerical scheme for these problems.

Numerical discretizations of fractional differential operators involve dense matrices for their nonlocal nature, even if an explicit numerical discretization is used [31,32]. Further, the numerical discretizations of time-fractional differential operators contain numerical solutions at each time step. Their numerical solutions have a computational complexity of $O(MN^2 + M^2N)$, as well as an $O(MN^2)$ memory requirement, where N is the number of spatial unknowns and M is the number of time steps. To improve the method, we followed the ideas from [33–35] to develop fast numerical methods for Problems (1) and (3).

The rest of the paper is organized as follows. An explicit finite difference method (EFDM) for the sFDE (3) is provided in Section 2, where we prove its boundary-preserving principle and error estimate. A fast implementation is also derived. In Section 3, we present an EFDM for stFDE (1). In a similar way, we prove its boundary-preserving principle and error estimate, as well as the fast implementation. In Section 4, we conduct numerical experiments to confirm the theoretical analysis, as well as study the performance of its fast implementation.

2. Derivation of Explicit Numerical Schemes

We derive numerical schemes for stFAEs (1) and sFAE (3) in this section. Let M, N_x and N_y be positive integers. We defined a space–time partition on $\Omega \times [0, T]$ by the following: $t_m := m\tau$ for $m = 0, 1, \dots, M$ with $\tau := T/M$ $x_i := x_d + ih_x$, as well as $y_j := y_d + jh_y$ for $i = 0, 1, \dots, N_x$ and $j = 0, 1, \dots, N_y$ with $h_x := (x_u - x_d)/N_x$ and $h_y := (y_u - y_d)/N_y$, respectively.

2.1. An Explicit Numerical Scheme for sFAE (3)

Let $\delta_h g(x) := (g(x+h) - g(x))/h$ and $\delta_\tau g(t) := (g(t+\tau) - g(t))/\tau$. We used an explicit Euler function to discretize $\partial_t u(x_i, t_m)$, as well as used the L-1 discretization to discretize $\partial_x^{\beta_{x,-}} u(x_i, t_m)$ and $\partial_x^{\beta_{x,+}} u(x_i, t_m)$ (where we have dropped the dummy variable y_j for conciseness), as follows:

$$\begin{aligned}
\partial_t u(x_i, t_m) &= \delta_\tau u(x_i, t_m) - R_{i,m}^t \\
\partial_x^{\beta_{x,-}} u(x_i, t_m) &= \frac{1}{\Gamma(1-\beta_x)} \sum_{k=0}^{i-1} \int_{x_k}^{x_{k+1}} \frac{\delta_{h_x} u(x_k, t_m) + (\partial_x u(z, t_m) - \delta_{h_x} u(x_k, t_m))}{(x_i - z)^{\beta_x}} dz \\
&= \frac{1}{\Gamma(2-\beta_x) h_x^{\beta_x}} \sum_{k=0}^{i-1} (u(x_{k+1}, t_m) - u(x_k, t_m)) \omega_{i-k}^{\beta_x} + R_{i,m}^{\beta_{x,-}}, \\
&=: \delta_{h_x}^{\beta_{x,-}} u(x_i, t_m) + R_{i,m}^{\beta_{x,-}}, \tag{10}
\end{aligned}$$

$$\begin{aligned}
\partial_x^{\beta_{x,+}} u(x_i, t_m) &= \frac{-1}{\Gamma(1-\beta_x)} \sum_{k=i}^{N_x-1} \int_{x_k}^{x_{k+1}} \frac{\delta_{h_x} u(x_k, t_m) + (\partial_x u(z, t_m) - \delta_{h_x} u(x_k, t_m))}{(z - x_i)^{\beta_x}} dz \\
&= \frac{-1}{\Gamma(2-\beta_x) h_x^{\beta_x}} \sum_{k=i}^{N_x-1} (u(x_{k+1}, t_m) - u(x_k, t_m)) \omega_{k-i+1}^{\beta_x} + R_{i,m}^{\beta_{x,+}},
\end{aligned}$$

where $\omega_k^\beta := k^{1-\beta} - (k-1)^{1-\beta}$ and

$$\begin{aligned}
R_{i,m}^t &:= \frac{1}{\tau} \int_{t_m}^{t_{m+1}} \partial_{tt} u(x_i, t) (t_{m+1} - t) dt = \mathcal{O}(\tau), \\
R_{i,m}^{\beta_{x,-}} &:= \frac{1}{\Gamma(1-\beta)} \sum_{k=0}^{i-1} \int_{x_k}^{x_{k+1}} \partial_{xx} u(s, t_m) \frac{(x_{k+1} + x_k - 2s)}{(x_i - s)^\beta} ds + \mathcal{O}(h^2), \\
R_{i,m}^{\beta_{x,+}} &:= \frac{1}{\Gamma(1-\beta)} \sum_{k=i}^{N-1} \int_{x_k}^{x_{k+1}} \partial_{xx} u(s, t_m) \frac{(x_{k+1} + x_k - 2s)}{(s - x_i)^\beta} ds + \mathcal{O}(h^2). \tag{11}
\end{aligned}$$

It was proved in [36] that $|R_{i,m}^{\beta_{x,-}}| \leq \mathcal{O}(h^{2-\beta})$. In fact, for the first term in the RHS of $R_{i,m}^{\beta_{x,+}}$ given above, we have

$$\begin{aligned}
&\frac{1}{\Gamma(1-\beta)} \sum_{k=i}^{N-1} \int_{x_k}^{x_{k+1}} \frac{(x_{k+1} + x_k - 2s)}{(s - x_i)^\beta} ds \\
&= \frac{h^{2-\beta}}{\Gamma(2-\beta)} \sum_{k=i}^{N-1} (2k+1) [(k-i+1)^{1-\beta} - (k-i)^{1-\beta}] \\
&\quad - \frac{2h^{2-\beta}}{\Gamma(2-\beta)} \sum_{k=i}^{N-1} [(k+1)(k-i+1)^{1-\beta} - k(k-i)^{1-\beta}] \\
&\quad + \frac{2h^{2-\beta}}{\Gamma(3-\beta)} \sum_{k=i}^{N-1} [(k-i+1)^{2-\beta} - (k-i)^{2-\beta}] \\
&= \frac{h^{2-\beta}}{\Gamma(2-\beta)} [(2N-1)(N-i)^{1-\beta} - 2(1^{1-\beta} + 2^{1-\beta} + 3^{1-\beta} + \dots \\
&\quad + (N-i-1)^{1-\beta})] - \frac{2h^{2-\beta}}{\Gamma(2-\beta)} N(N-i)^{1-\beta} + \frac{2h^{2-\beta}}{\Gamma(3-\beta)} (N-i)^{2-\beta} \\
&= \frac{h^{2-\beta}}{\Gamma(2-\beta)} [-(N-i)^{1-\beta} - 2(1^{1-\beta} + 2^{1-\beta} + 3^{1-\beta} + \dots \\
&\quad + (N-i-1)^{1-\beta}) + \frac{2}{2-\beta} (N-i)^{2-\beta}] \\
&= \frac{-h^{2-\beta}}{\Gamma(2-\beta)} S(N-i).
\end{aligned}$$

It was shown in [36] that $S(N-i) \leq C$; thus, we obtained $|R_{i,m}^{\beta_{x,+}}| \leq \mathcal{O}(h^{2-\beta})$.

We discretized $\partial_y^{\beta_y, \pm} u$ by symmetry and reformulate sFAE (3) as the following reference equation:

$$\begin{aligned} \delta_\tau u(x_i, y_j, t_m) + v_x [\kappa_x \delta_{h_x}^{\beta_x, -} + (1 - \kappa_x) \delta_{h_x}^{\beta_x, +}] u(x_i, y_j, t_m) \\ + v_y [\kappa_y \delta_{h_y}^{\beta_y, -} + (1 - \kappa_y) \delta_{h_y}^{\beta_y, +}] u(x_i, y_j, t_m) = f(x_i, y_j, t_m) + R_{i,j,m}, \end{aligned} \quad (12)$$

where $R_{i,j,m}$ is given by

$$\begin{aligned} R_{i,j,m} := -R_{i,j,m}^t - v_x [\kappa_x R_{i,j,m}^{\beta_x, -} + (1 - \kappa_x) R_{i,j,m}^{\beta_x, +}] \\ - v_y [\kappa_y R_{i,j,m}^{\beta_y, -} + (1 - \kappa_y) R_{i,j,m}^{\beta_y, +}] = \mathcal{O}(\tau + h_x^{2-\beta_x} + h_y^{2-\beta_y}). \end{aligned} \quad (13)$$

Let $u_{i,j}^m$ be the numerical approximation to $u(x_i, y_j, t_m)$ and $f_{i,j}^m := f(x_i, y_j, t_m)$. We replaced $u(x_i, y_j, t_m)$ by $u_{i,j}^m$ in (13), and we dropped the local truncation error term to arrive at the following numerical scheme:

$$\begin{aligned} \delta_\tau u_{i,j}^m + v_x [\kappa_x \delta_{h_x}^{\beta_x, -} + (1 - \kappa_x) \delta_{h_x}^{\beta_x, +}] u_{i,j}^m \\ + v_y [\kappa_y \delta_{h_y}^{\beta_y, -} + (1 - \kappa_y) \delta_{h_y}^{\beta_y, +}] u_{i,j}^m = f_{i,j}^m. \end{aligned} \quad (14)$$

2.2. An Explicit Numerical Scheme for stFAE (1)

To derive a numerical scheme for stFAE (1), we simply need to discretize $\partial_t^\alpha u$ at time step t_m by an L_1 discretization as follows:

$$\begin{aligned} \partial_t^\alpha u(x_i, y_j, t_m) &= \frac{1}{\Gamma(1-\alpha)} \sum_{k=0}^{m-1} \int_{t_k}^{t_{k+1}} \frac{\partial_t u(x_i, y_j, t)}{(t_m - t)^\alpha} dt \\ &= \frac{1}{\Gamma(1-\alpha)} \sum_{k=0}^{m-1} \int_{t_k}^{t_{k+1}} \frac{\delta_\tau u(x_i, y_j, t_k)}{(t_m - t)^\alpha} dt + R_{i,j,m}^{t,\alpha} \\ &= \frac{1}{\Gamma(2-\alpha)\tau^\alpha} \sum_{k=0}^{m-1} (u(x_i, y_j, t_{k+1}) - u(x_i, y_j, t_k)) \omega_{m-k}^\alpha + R_{i,j,m}^{t,\alpha} \\ &=: \delta_\tau^\alpha u(x_i, y_j, t_m) + R_{i,j,m}^{t,\alpha}, \end{aligned} \quad (15)$$

where the local truncation error $R_{i,j,m}^{t,\alpha}$ can be expressed as

$$\begin{aligned} R_{i,j,m}^{t,\alpha} &= \frac{1}{\Gamma(1-\alpha)} \sum_{k=0}^{m-1} \int_{t_k}^{t_{k+1}} \frac{\delta_\tau u(x_i, y_j, t_k) - \partial_t u(x_i, y_j, t)}{(t_m - t)^\alpha} dt \\ &= \frac{1}{\Gamma(1-\alpha)\tau} \sum_{k=0}^{m-1} \int_{t_k}^{t_{k+1}} \frac{1}{(t_m - t)^\alpha} \int_t^{t_{k+1}} \int_t^s \partial_{\theta\theta} u(x_i, y_j, \theta) d\theta ds dt. \end{aligned} \quad (16)$$

$$\begin{aligned}
\delta_{h_x}^{\beta_x, -} u_{i,j}^m &:= \frac{h_x^{-\beta_x}}{\Gamma(2-\beta_x)} \left[\omega_1^{\beta_x} u_{i,j}^m - \sum_{k=1}^{i-1} (\omega_k^{\beta_x} - \omega_{k+1}^{\beta_x}) u_{i-k,j}^m - \omega_i^{\beta_x} u_{0,j}^m \right], \\
\delta_{h_x}^{\beta_x, +} u_{i,j}^m &:= \frac{h_x^{-\beta_x}}{\Gamma(2-\beta_x)} \left[\omega_1^{\beta_x} u_{i,j}^m - \sum_{k=1}^{N_x-i-1} (\omega_k^{\beta_x} - \omega_{k+1}^{\beta_x}) u_{i+k,j}^m - \omega_{N_x-i}^{\beta_x} u_{N_x,j}^m \right], \\
\delta_{h_y}^{\beta_y, -} u_{i,j}^m &:= \frac{h_y^{-\beta_y}}{\Gamma(2-\beta_y)} \left[\omega_1^{\beta_y} u_{i,j}^m - \sum_{k=1}^{j-1} (\omega_k^{\beta_y} - \omega_{k+1}^{\beta_y}) u_{i,j-k}^m - \omega_j^{\beta_y} u_{i,0}^m \right], \\
\delta_{h_y}^{\beta_y, +} u_{i,j}^m &:= \frac{h_y^{-\beta_y}}{\Gamma(2-\beta_y)} \left[\omega_1^{\beta_y} u_{i,j}^m - \sum_{k=1}^{N_y-i-1} (\omega_k^{\beta_y} - \omega_{k+1}^{\beta_y}) u_{i,j+k}^m - \omega_{N_y-i}^{\beta_y} u_{i,N_y}^m \right].
\end{aligned} \tag{17}$$

Here, $\omega_k^\beta := k^{1-\beta} - (k-1)^{1-\beta}$, with $0 < \beta < 1$ having the following properties:

$$\begin{cases} \omega_k^\beta > 0, k = 1, 2, \dots, N, \\ 1 = \omega_1^\beta > \omega_2^\beta > \dots > \omega_k^\beta, \omega_k^\beta \rightarrow 0 \text{ as } k \rightarrow \infty, \\ \sum_{k=1}^{i-1} (\omega_k^\beta - \omega_{k+1}^\beta) + \omega_i^\beta = 1. \end{cases} \tag{18}$$

From the above discretization, we can obtain the following EFDM for sFDE (3), which satisfies the bound-preserving principle as follows:

$$\begin{aligned}
u_{i,j}^m &= (1 - \lambda_x - \lambda_y) u_{i,j}^{m-1} + \lambda_x \kappa_x \left[\sum_{k=1}^{i-1} (\omega_k^{\beta_x} - \omega_{k+1}^{\beta_x}) u_{i-k,j}^m + \omega_i^{\beta_x} u_{0,j}^m \right] \\
&\quad + \lambda_x (1 - \kappa_x) \left[\sum_{k=1}^{N_x-i-1} (\omega_k^{\beta_x} - \omega_{k+1}^{\beta_x}) u_{i+k,j}^m + \omega_{N_x-i}^{\beta_x} u_{N_x,j}^m \right] \\
&\quad + \lambda_y \kappa_y \left[\sum_{k=1}^{j-1} (\omega_k^{\beta_y} - \omega_{k+1}^{\beta_y}) u_{i,j-k}^m + \omega_j^{\beta_y} u_{i,0}^m \right] \\
&\quad + \lambda_y (1 - \kappa_y) \left[\sum_{k=1}^{N_y-i-1} (\omega_k^{\beta_y} - \omega_{k+1}^{\beta_y}) u_{i,j+k}^m + \omega_{N_y-i}^{\beta_y} u_{i,N_y}^m \right] + f_{i,j}^m,
\end{aligned} \tag{19}$$

where $\lambda_x = v_x \frac{\tau}{\Gamma(2-\beta_x) h_x^{\beta_x}}$ and $\lambda_y = v_y \frac{\tau}{\Gamma(2-\beta_y) h_y^{\beta_y}}$.

2.3. Stability and Error Analysis of the EFDM for sFDE

Now, we are in the position to analyze the bound-preserving principle and the error estimate of the EFDM for sFDE (3). We begin with the bound-preserving principle, i.e., if $u_{i,j}^0$ is in the range $[l, L]$, then $u_{i,j}^m \in [l, L]$.

Theorem 1. *If all the values of $u_{i,j}^0$ are in the range $[l, L]$, then $u_{i,j}^m \in [l, L]$ for any i, j, m under the Courant–Friedrichs–Lewy (CFL) condition is $0 < \lambda_x + \lambda_y < 1$.*

Proof. The theorem is proved by a mathematical induction. Firstly, when $m = 1$, according to the explicit scheme (19), we have

$$\begin{aligned}
 u_{i,j}^1 &= (1 - \lambda_x - \lambda_y)u_{i,j}^0 + \lambda_x \kappa_x \left[\sum_{k=1}^{i-1} (\omega_k^{\beta_x} - \omega_{k+1}^{\beta_x})u_{i-k,j}^0 + \omega_i^{\beta_x}u_{0,j}^0 \right] \\
 &\quad + \lambda_x(1 - \kappa_x) \left[\sum_{k=1}^{N_x-i-1} (\omega_k^{\beta_x} - \omega_{k+1}^{\beta_x})u_{i+k,j}^0 + \omega_{N_x-i}^{\beta_x}u_{N_x,j}^0 \right] \\
 &\quad + \lambda_y \kappa_y \left[\sum_{k=1}^{j-1} (\omega_k^{\beta_y} - \omega_{k+1}^{\beta_y})u_{i,j-k}^0 + \omega_j^{\beta_y}u_{i,0}^0 \right] \\
 &\quad + \lambda_y(1 - \kappa_y) \left[\sum_{k=1}^{N_y-j-1} (\omega_k^{\beta_y} - \omega_{k+1}^{\beta_y})u_{i,j+k}^0 + \omega_{N_y-j}^{\beta_y}u_{i,N_y}^0 \right] \\
 &=: \sum_{i=0}^{N_x} \sum_{j=0}^{N_y} \eta_{i,j} u_{i,j}^0.
 \end{aligned} \tag{20}$$

We can conclude that all coefficients $\eta_{i,j} \geq 0$ are such using the properties of coefficients ω_k^β in (18) and the CFL condition [37] $0 < \lambda_x + \lambda_y < 1$. Then, we have

$$\begin{aligned}
 \sum_{i=0}^{N_x} \sum_{j=0}^{N_y} \eta_{i,j} &= (1 - \lambda_x - \lambda_y) + \lambda_x \kappa_x \left[\sum_{k=1}^{i-1} (\omega_k^{\beta_x} - \omega_{k+1}^{\beta_x}) + \omega_i^{\beta_x} \right] \\
 &\quad + \lambda_x(1 - \kappa_x) \left[\sum_{k=1}^{N_x-i-1} (\omega_k^{\beta_x} - \omega_{k+1}^{\beta_x}) + \omega_{N_x-i}^{\beta_x} \right] \\
 &\quad + \lambda_y \kappa_y \left[\sum_{k=1}^{j-1} (\omega_k^{\beta_y} - \omega_{k+1}^{\beta_y}) + \omega_j^{\beta_y} \right] \\
 &\quad + \lambda_y(1 - \kappa_y) \left[\sum_{k=1}^{N_y-j-1} (\omega_k^{\beta_y} - \omega_{k+1}^{\beta_y}) + \omega_{N_y-j}^{\beta_y} \right] \\
 &= 1.
 \end{aligned} \tag{21}$$

We combined (21) into (20) to bound $u_{i,j}^1$ as follows:

$$\min_{i,j} u_{i,j}^0 \leq u_{i,j}^1 = \sum_{i=0}^{N_x} \sum_{j=0}^{N_y} \eta_{i,j} u_{i,j}^0 \leq \max_{i,j} u_{i,j}^0.$$

As $u_{i,j}^0 \in [l, L]$ for all i, j , then we have for all i, j ,

$$u_{i,j}^1 \in [l, L].$$

Secondly, suppose that all the values of $u_{i,j}^{m-1}$ are in the range $[l, L]$, then according to the explicit scheme (19), we have

$$u_{i,j}^m = \sum_{i=0}^{N_x} \sum_{j=0}^{N_y} \eta_{i,j} u_{i,j}^{m-1}.$$

The above equation has the same coefficients $\eta_{i,j}$ with Equation (20). Hence, we can conclude that $u_{i,j}^m \in [l, L]$ under the CFL condition is $0 < \lambda_x + \lambda_y < 1$. $\square$

An error estimate of the explicit finite difference scheme (19) is proved as follows. Let

$$e_{i,j}^m := u_{i,j}^m - u(x_i, y_j, t_m), \quad (x_i, y_j) \in \Omega, \quad 0 \leq m \leq M. \tag{22}$$

Theorem 2. Assume $u(x, y, t) \in L^\infty(0, T; H^3)$, then there is a constant C that is positive and independent of the mesh parameters h_x, h_y and τ , such that

$$\sup_{i,j} |e_{i,j}^m| \leq C(\tau + h_x^{2-\beta_x} + h_y^{2-\beta_y}),$$

under the CFL condition $0 < \lambda_x + \lambda_y < 1$.

Proof. From the truncation errors of the discretization in Equation (16) and the explicit scheme, we determined that the errors $e_{i,j}^m$ satisfy the following formulation:

$$\begin{aligned} & e_{i,j}^m - e_{i,j}^{m-1} + \lambda_x \kappa_x \left[\omega_1^{\beta_x} e_{i,j}^{m-1} - \sum_{k=1}^{i-1} (\omega_k^{\beta_x} - \omega_{k+1}^{\beta_x}) e_{i-k,j}^{m-1} - \omega_i^{\beta_x} e_{0,j}^{m-1} \right] \\ & + \lambda_x (1 - \kappa_x) \left[\omega_1^{\beta_x} e_{i,j}^{m-1} - \sum_{k=1}^{N_x-i-1} (\omega_k^{\beta_x} - \omega_{k+1}^{\beta_x}) e_{i+k,j}^{m-1} - \omega_{N_x-i}^{\beta_x} e_{N_x,j}^{m-1} \right], \\ & + \lambda_y \kappa_y \left[\omega_1^{\beta_y} e_{i,j}^{m-1} - \sum_{k=1}^{j-1} (\omega_k^{\beta_y} - \omega_{k+1}^{\beta_y}) e_{i,j-k}^{m-1} - \omega_j^{\beta_y} e_{i,0}^{m-1} \right] \\ & + \lambda_y (1 - \kappa_y) \left[\omega_1^{\beta_y} e_{i,j}^{m-1} - \sum_{k=1}^{N_y-j-1} (\omega_k^{\beta_y} - \omega_{k+1}^{\beta_y}) e_{i,j+k}^{m-1} - \omega_{N_y-j}^{\beta_y} e_{i,N_y}^{m-1} \right], \\ & = O(\tau^2 + \tau(h_x^{2-\beta_x} + h_y^{2-\beta_y})), \end{aligned} \quad (23)$$

and all $e_{i,j}^0 = 0$.

When the CFL condition is $0 < \lambda_x + \lambda_y < 1$, according to Theorem 1, we have

$$\begin{aligned} |e_{i,j}^m| & \leq (1 - \lambda_x - \lambda_y) |e_{i,j}^{m-1}| + \lambda_x \kappa_x \sum_{k=1}^{i-1} (\omega_k^{\beta_x} - \omega_{k+1}^{\beta_x}) |e_{i-k,j}^{m-1}| + \lambda_x \kappa_x \omega_i^{\beta_x} |e_{0,j}^{m-1}| \\ & + \lambda_x (1 - \kappa_x) \sum_{k=1}^{N_x-i-1} (\omega_k^{\beta_x} - \omega_{k+1}^{\beta_x}) |e_{i+k,j}^{m-1}| + \lambda_x (1 - \kappa_x) \omega_{N_x-i}^{\beta_x} |e_{N_x,j}^{m-1}| \\ & + \lambda_y \kappa_y \sum_{k=1}^{j-1} (\omega_k^{\beta_y} - \omega_{k+1}^{\beta_y}) |e_{i,j-k}^{m-1}| + \lambda_y \kappa_y \omega_j^{\beta_y} |e_{i,0}^{m-1}| \\ & + \lambda_y (1 - \kappa_y) \sum_{k=1}^{N_y-j-1} (\omega_k^{\beta_y} - \omega_{k+1}^{\beta_y}) |e_{i,j+k}^{m-1}| + \lambda_y (1 - \kappa_y) \omega_{N_y-j}^{\beta_y} |e_{i,N_y}^{m-1}| \\ & + O(\tau^2 + \tau(h_x^{2-\beta_x} + h_y^{2-\beta_y})) \\ & \leq \sup_{i,j} |e_{i,j}^{m-1}| + O(\tau^2 + \tau(h_x^{2-\beta_x} + h_y^{2-\beta_y})) \end{aligned}$$

i.e.,

$$\sup_{i,j} |e_{i,j}^m| \leq \sup_{i,j} |e_{i,j}^{m-1}| + O(\tau^2 + \tau(h_x^{2-\beta_x} + h_y^{2-\beta_y})). \quad (24)$$

By induction,

$$\sup_{i,j} |e_{i,j}^m| \leq \sup_{i,j} |e_{i,j}^0| + m \cdot O(\tau^2 + \tau(h_x^{2-\beta_1} + h_y^{2-\beta_2})). \quad (25)$$

For $m \leq M$, $T = M\tau$, we can easily obtain $m\tau \leq T$.

Therefore, $\sup_{i,j} |e_{i,j}^m| \leq C(\tau + h_x^{2-\beta_x} + h_y^{2-\beta_y})$ and C is a constant independent of h_x, h_y

and τ . $\square$

2.4. A Fast Implementation

We provide a demonstration of the complexity of computation of the fast finite difference method for sFDE (3) in this subsection.

We set

$$\mathbf{u}^m = [u_{1,1}^m, \dots, u_{N_x,1}^m, u_{1,2}^m, \dots, u_{N_x,2}^m, \dots, u_{1,N_y}^m, \dots, u_{N_x,N_y}^m]^T,$$

$$\mathbf{f}^m = [f_{1,1}^m, \dots, f_{N_x,1}^m, f_{1,2}^m, \dots, f_{N_x,2}^m, \dots, f_{1,N_y}^m, \dots, f_{N_x,N_y}^m]^T.$$

From the explicit finite difference scheme of sFDE (3), we can get obtain linear algebraic system

$$\mathbf{u}^m = \mathbf{u}^{m-1} + \mathbf{A}\mathbf{u}^{m-1} + \tau\mathbf{f}^{m-1}. \quad (26)$$

Here, the stiffness of Matrix $\mathbf{A}$ is expressed by

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_0 & \mathbf{A}_1 & \dots & \mathbf{A}_{N_y-2} & \mathbf{A}_{N_y-1} \\ \mathbf{A}_{-1} & \mathbf{A}_0 & \mathbf{A}_1 & \ddots & \mathbf{A}_{N_y-2} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{A}_{2-N_y} & \ddots & \mathbf{A}_{-1} & \mathbf{A}_0 & \mathbf{A}_1 \\ \mathbf{A}_{1-N_y} & \mathbf{A}_{2-N_y} & \dots & \mathbf{A}_{-1} & \mathbf{A}_0 \end{pmatrix}, \quad (27)$$

where

$$\mathbf{A}_0 = \begin{pmatrix} a_0 & a_1 & \dots & a_{N_x-2} & a_{N_x-1} \\ a_{-1} & a_0 & a_1 & \ddots & a_{N_x-2} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ a_{2-N_x} & \ddots & a_{-1} & a_0 & a_1 \\ a_{1-N_x} & a_{2-N_x} & \dots & a_{-1} & a_0 \end{pmatrix},$$

$$\begin{aligned} \mathbf{A}_s(i, j) &= b_s \mathbf{E}, \\ \mathbf{A}_{-s}(i, j) &= b_{-s} \mathbf{E}, \quad s = 1, 2, \dots, N_y - 1. \end{aligned}$$

with $\mathbf{E}$ being the N_x -by- N_x dimensional unit matrix and

$$\begin{aligned} a_0 &= -\lambda_x - \lambda_y, \\ a_{-s} &= \lambda_x \kappa_x (\omega_s^{\beta_x} - \omega_{s+1}^{\beta_x}), \quad s = 1, 2, \dots, N_x - 1 \\ a_s &= \lambda_x (1 - \kappa_x) (\omega_s^{\beta_x} - \omega_{s+1}^{\beta_x}), \quad s = 1, 2, \dots, N_x - 1 \\ b_{-s} &= \lambda_y \kappa_y (\omega_s^{\beta_y} - \omega_{s+1}^{\beta_y}), \quad s = 1, 2, \dots, N_y - 1 \\ b_s &= \lambda_y (1 - \kappa_y) (\omega_s^{\beta_y} - \omega_{s+1}^{\beta_y}), \quad s = 1, 2, \dots, N_y - 1. \end{aligned}$$

Through studying the structure of the stiffness Matrix $\mathbf{A}$, we found that the stiffness Matrix $\mathbf{A}$ had a block-Toeplitz-Toeplitz-block structure [38,39]. It was proved in [33,34] that, for any vector $\mathbf{u} \in \mathbb{R}^N$, the computational cost of the matrix-vector multiplication $\mathbf{A}\mathbf{u}$ has $O(N \log N)$ operations, with $N = N_x N_y$ being the spatial points number, which is obtained by embedding $\mathbf{A}$ into a $2N_y$ -by- $2N_y$ block-circulant Matrix $\mathbf{C}$ and using a fast Fourier transform (FFT). Combining the linear algebraic system, we can easily obtain the following theorem.

Theorem 3. *The fast EFDM can be carried out in $O(MN \log N)$ operations and has an $O(MN)$ requirement by the fast Fourier transform.*

3. A Finite Difference Approximation for stFDE (1)

We constructed an explicit monotone numerical scheme to solve stFDE (1), which satisfied the bound-preserving principle. And the error estimate of the EFDM for the stFDE was analyzed in the maximum-norm form. We also showed that a fast EFDM (FFDM) with the stiffness matrix of the EFDM has a Toeplitz-like structure, and the matrix-vector computation can be carried out by the FFT.

Combining the discretizations (17) with (15), a bound-principle-satisfying EFDM for the stFDE (1) can be given as

$$\begin{aligned}
 u_{i,j}^m = & (1 - \lambda_3 - \lambda_4 - \omega_1^\alpha) u_{i,j}^{m-1} + \sum_{s=2}^{m-1} (\omega_{s-1}^\alpha - \omega_s^\alpha) u_{i,j}^{m-s} + \omega_{m-1}^\alpha u_{i,j}^0 \\
 & + \lambda_3 \kappa_x \left[\sum_{k=1}^{i-1} (\omega_{k-1}^{\beta_x} - \omega_k^{\beta_x}) u_{i-k,j}^{m-1} + \omega_{i-1}^{\beta_x} u_{0,j}^{m-1} \right] \\
 & + \lambda_3 (1 - \kappa_x) \left[\sum_{k=1}^{N_x-i-1} (\omega_{k-1}^{\beta_x} - \omega_k^{\beta_x}) u_{i+k,j}^{m-1} + \omega_{N_x-i-1}^{\beta_x} u_{N_x,j}^{m-1} \right] \\
 & + \lambda_4 \kappa_y \left[\sum_{k=1}^{j-1} (\omega_{k-1}^{\beta_y} - \omega_k^{\beta_y}) u_{i,j-k}^{m-1} + \omega_{j-1}^{\beta_y} u_{i,0}^{m-1} \right] \\
 & + \lambda_4 (1 - \kappa_y) \left[\sum_{k=1}^{N_y-j-1} (\omega_{k-1}^{\beta_y} - \omega_k^{\beta_y}) u_{i,j+k}^{m-1} + \omega_{N_y-j-1}^{\beta_y} u_{i,N_y}^{m-1} \right] + f_{i,j}^m,
 \end{aligned} \quad (28)$$

$$\text{Here } \lambda_3 = v_x \frac{\Gamma(2-\alpha)\tau^\alpha}{\Gamma(2-\beta_x)h_x^{\beta_x}} \text{ and } \lambda_4 = v_y \frac{\Gamma(2-\alpha)\tau^\alpha}{\Gamma(2-\beta_y)h_y^{\beta_y}}.$$

3.1. Error Analysis and Stability of the EFDM for stFDE

We analyze the bound-preserving principle and the error estimate of the explicit finite difference scheme for stFDE (1) in this subsection. We begin with the bound-preserving principle.

Theorem 4. *If the initial values $u_{i,j}^0$ are in the range $[l, L]$, then $u_{i,j}^m \in [l, L]$ for any i, j, m under the CFL condition is $0 < \lambda_3 + \lambda_4 < 1 - \omega_2^\alpha$.*

Proof. The theorem is proved by mathematical induction. Firstly, the EFDM (28) of stFDE (1) can be formulated as

$$\begin{aligned}
 u_{i,j}^1 = & (1 - \lambda_3 - \lambda_4) u_{i,j}^0 + \lambda_3 \kappa_x \left[\sum_{k=1}^{i-1} (\omega_k^{\beta_x} - \omega_{k+1}^{\beta_x}) u_{i-k,j}^0 + \omega_i^{\beta_x} u_{0,j}^0 \right] \\
 & + \lambda_3 (1 - \kappa_x) \left[\sum_{k=1}^{N_x-i-1} (\omega_k^{\beta_x} - \omega_{k+1}^{\beta_x}) u_{i+k,j}^0 + \omega_{N_x-i}^{\beta_x} u_{N_x,j}^0 \right], \\
 & + \lambda_4 \kappa_y \left[\sum_{k=1}^{j-1} (\omega_k^{\beta_y} - \omega_{k+1}^{\beta_y}) u_{i,j-k}^0 + \omega_j^{\beta_y} u_{i,0}^0 \right] \\
 & + \lambda_4 (1 - \kappa_y) \left[\sum_{k=1}^{N_y-j-1} (\omega_k^{\beta_y} - \omega_{k+1}^{\beta_y}) u_{i,j+k}^0 + \omega_{N_y-j}^{\beta_y} u_{i,N_y}^0 \right],
 \end{aligned} \quad (29)$$

This formulation is similar with (20) in Theorem 1, where λ_x and λ_y are replaced by λ_3 and λ_4 , respectively. According to Theorem 1, we can obtain all of the $u_{i,j}^1 \in [l, L]$ if all the initial values $u_{i,j}^0$ are in the range of $[l, L]$.

Now, if we suppose that $u_{i,j}^{m-1} \in [l, L]$ and all the values at each time steps are in the range $[l, L]$, then we need prove $u_{i,j}^m \in [l, L]$ for Scheme (28).

$$\begin{aligned}
u_{i,j}^m &= (1 - \omega_2^\alpha - \lambda_3 - \lambda_4)u_{i,j}^{m-1} + \sum_{s=2}^{m-1} (\omega_s^\alpha - \omega_{s+1}^\alpha)u_{i,j}^{m-s} + \omega_m^\alpha u_{i,j}^0 \\
&\quad + \lambda_3 \kappa_x (\omega_1^{\beta_x} - \omega_2^{\beta_x})u_{i-1,j}^{m-1} + \cdots + \lambda_3 \kappa_x \omega_i^{\beta_x} u_{0,j}^{m-1} \\
&\quad + \lambda_3 (1 - \kappa_x) (\omega_1^{\beta_x} - \omega_2^{\beta_x})u_{i+1,j}^{m-1} + \cdots + \lambda_3 (1 - \kappa_x) \omega_{N_x-i}^{\beta_x} u_{N_x,j}^{m-1} \\
&\quad + \lambda_4 \kappa_y (\omega_1^{\beta_y} - \omega_2^{\beta_y})u_{i,j-1}^{m-1} + \cdots + \lambda_4 \kappa_y \omega_{i,j}^{\beta_y} u_{i,0}^{m-1} \\
&\quad + \lambda_4 (1 - \kappa_y) (\omega_1^{\beta_y} - \omega_2^{\beta_y})u_{i,j+1}^{m-1} + \cdots + \lambda_4 (1 - \kappa_y) \omega_{N_y-i}^{\beta_y} u_{i,N_y}^{m-1}. \\
&= \sum_{k=0}^{N_x} \sum_{n=0}^{N_y} \eta_{k,n} u_{k,n}^{m-1} + \sum_{s=0}^{m-2} \zeta_s u_{i,j}^s.
\end{aligned} \tag{30}$$

According to the CFL condition and the properties of the coefficients ω_k^β , we can obtain all the coefficients $\eta_{k,l} \geq 0$ and $\zeta_s \geq 0$; as such, we can only analyze the coefficients. From Equation (18), we have

$$\begin{aligned}
\sum_{k=0}^{N_x} \sum_{n=0}^{N_y} \eta_{k,n} &= (1 - \omega_2^\alpha - \lambda_3 - \lambda_4) + \lambda_3 \kappa_x \left[\sum_{k=0}^{i-1} (\omega_k^{\beta_x} - \omega_{k+1}^{\beta_x}) + \omega_{i-1}^{\beta_x} \right] \\
&\quad + \lambda_3 (1 - \kappa_x) \left[\sum_{k=0}^{N_x-i} (\omega_{k-1}^{\beta_x} - \omega_k^{\beta_x}) + \omega_{N_x-i}^{\beta_x} \right] \\
&\quad + \lambda_4 \kappa_y \left[\sum_{k=0}^{i-1} (\omega_k^{\beta_y} - \omega_{k+1}^{\beta_y}) + \omega_{i-1}^{\beta_y} \right] \\
&\quad + \lambda_4 (1 - \kappa_y) \left[\sum_{k=0}^{N_y-i} (\omega_{k-1}^{\beta_y} - \omega_k^{\beta_y}) + \omega_{N_y-i}^{\beta_y} \right] \\
&= (1 - \omega_2^\alpha - \lambda_3 - \lambda_4) + \lambda_3 \kappa_x + \lambda_3 (1 - \kappa_x) + \lambda_4 \kappa_y + \lambda_4 (1 - \kappa_y) \\
&= 1 - \omega_2^\alpha,
\end{aligned}$$

$$\sum_{s=0}^{m-2} \zeta_s = \sum_{s=2}^{m-1} (\omega_s^\alpha - \omega_{s+1}^\alpha) + \omega_m^\alpha = \omega_2^\alpha.$$

Hence, $\sum_{k=0}^{N_x} \sum_{l=0}^{N_y} \eta_{k,l} + \sum_{s=0}^{m-2} \zeta_s = 1$ is a convex combination and has

$$l \leq \sum_{k=0}^{N_x} \sum_{l=0}^{N_y} \eta_{k,l} u_{k,l}^{m-1} + \sum_{s=0}^{m-2} \zeta_s u_{i,j}^s \leq L.$$

Thus, the $u_{i,j}^m \in [l, L]$ under the CFL condition is $0 < \lambda_3 + \lambda_4 < 1 - \omega_2^\alpha$. $\square$

Next, we analyzed the error estimate of the EFDM for stFDE (2). The error $e_{i,j}^m$ was defined in the same way as in Section 2.

Theorem 5. If we assume $u(x, y, t) \in L^\infty(0, T; H^2)$, then there is a positive constant C , which is independent of the mesh parameters h_x, h_y and τ , such that

$$\sup_{i,j} |e_{i,j}^m| \leq C(\tau^{2-\alpha} + h_x^{2-\beta_x} + h_y^{2-\beta_y})$$

under the CFL condition is $0 < \lambda_3 + \lambda_4 < 1 - \omega_2^\alpha$.

Proof. The error $e_{i,j}^m$ satisfies the following error equation:

$$\begin{aligned}
 e_{i,j}^m &= (1 - \lambda_3 - \lambda_4 - \omega_2^\alpha) e_{i,j}^{m-1} + \sum_{s=2}^{m-1} (\omega_s^\alpha - \omega_{s+1}^\alpha) e_{i,j}^{m-s} + \omega_{m-1}^\alpha e_{i,j}^0 \\
 &+ \lambda_3 \kappa_x \left[\sum_{k=1}^{i-1} (\omega_k^{\beta_x} - \omega_{k+1}^{\beta_x}) e_{i-k,j}^{m-1} + \omega_i^{\beta_x} e_{0,j}^{m-1} \right] \\
 &+ \lambda_3 (1 - \kappa_x) \left[\sum_{k=1}^{N_x-i} (\omega_k^{\beta_x} - \omega_{k+1}^{\beta_x}) e_{i+k,j}^{m-1} + \omega_{N_x-i}^{\beta_x} e_{N_x,j}^{m-1} \right], \\
 &+ \lambda_4 \kappa_y \left[\sum_{k=1}^{j-1} (\omega_k^{\beta_y} - \omega_{k+1}^{\beta_y}) e_{i,j-k}^{m-1} + \omega_j^{\beta_y} e_{i,0}^{m-1} \right] \\
 &- \lambda_4 (1 - \kappa_y) \left[\sum_{k=1}^{N_y-j-1} (\omega_k^{\beta_y} - \omega_{k+1}^{\beta_y}) e_{i,j+k}^{m-1} + \omega_{N_y-j}^{\beta_y} e_{i,N_y}^{m-1} \right], \\
 &+ \Gamma(2 - \alpha) \Delta t^\alpha O(\Delta t^{2-\alpha} + \Delta t(h_x^{2-\beta_x} + h_y^{2-\beta_y})),
 \end{aligned} \tag{31}$$

where all $e_{i,j}^0 = 0$.

Firstly, when $m = 1$, the above error equation can be formulated as follows:

$$\begin{aligned}
 e_{i,j}^1 &= (1 - \lambda_3 - \lambda_4 - \omega_2^\alpha) e_{i,j}^0 + \lambda_3 \kappa_x \left[\sum_{k=1}^{i-1} (\omega_k^{\beta_x} - \omega_{k+1}^{\beta_x}) e_{i-k,j}^0 + \omega_i^{\beta_x} e_{0,j}^0 \right] \\
 &+ \lambda_3 (1 - \kappa_x) \left[\sum_{k=1}^{N_x-i-1} (\omega_k^{\beta_x} - \omega_{k+1}^{\beta_x}) e_{i+k,j}^0 + \omega_{N_x-i}^{\beta_x} e_{N_x,j}^0 \right], \\
 &+ \lambda_4 \kappa_y \left[\sum_{k=1}^{j-1} (\omega_k^{\beta_y} - \omega_{k+1}^{\beta_y}) e_{i,j-k}^0 + \omega_j^{\beta_y} e_{i,0}^0 \right] \\
 &+ \lambda_4 (1 - \kappa_y) \left[\sum_{k=1}^{N_y-j-1} (\omega_k^{\beta_y} - \omega_{k+1}^{\beta_y}) e_{i,j+k}^0 + \omega_{N_y-j}^{\beta_y} e_{i,N_y}^0 \right] \\
 &+ \Gamma(2 - \alpha) \Delta t^\alpha O(\Delta t^{2-\alpha} + h_x^{2-\beta_x} + h_y^{2-\beta_y}).
 \end{aligned}$$

The above formulation is similar to (20) in Theorem 1, with λ_x and λ_y being replaced by λ_3 and λ_4 , respectively. According to (18), all of the coefficients of errors are positive and the sum of all the coefficients is 1; thus, we can obtain

$$\begin{aligned}
 \sup_{i,j} |e_{i,j}^1| &\leq \sup_{i,j} |e_{i,j}^0| + \Gamma(2 - \alpha) \Delta t^\alpha O(\Delta t^{2-\alpha} + h_x^{2-\beta_x} + h_y^{2-\beta_y}) \\
 &\leq \sup_{i,j} |e_{i,j}^0| + \Gamma(1 - \alpha) t_1^\alpha O(\Delta t^{2-\alpha} + h_x^{2-\beta_x} + h_y^{2-\beta_y}).
 \end{aligned} \tag{32}$$

Secondly, suppose that, for all $s \leq m - 1$, the error e_i^s satisfies the following formulation:

$$\sup_{i,j} |e_{i,j}^s| \leq \sup_{i,j} |e_{i,j}^0| + \Gamma(1 - \alpha) t_1^\alpha O(\Delta t^{2-\alpha} + h_x^{2-\beta_x} + h_y^{2-\beta_y}). \tag{33}$$

Next, we proved that the error estimate holds when $s = m$. Furthermore, if we note that the coefficients $\omega_{k+1}^\alpha = (1 - \alpha) \int_k^{k+1} x^{-\alpha} dx$ and $x^{-\alpha}$ monotonically decrease for $x > 0$, then

$$(1 - \alpha)(k + 1)^{-\alpha} < \omega_{k+1}^\alpha < (1 - \alpha)k^{-\alpha}. \tag{34}$$

We can observe that the coefficients of errors are positive and the sum of all the coefficients is 1 from (18) and

$$|e_{i,j}^m| \leq (1 - \omega_1^\alpha) \sup_{i,j} |e_{i,j}^{m-1}| + \sum_{s=1}^{m-1} (\omega_s^\alpha - \omega_{s+1}^\alpha) \sup_{i,j} |e_{i,j}^{m-s}| + \omega_{m-1}^\alpha \sup_{i,j} |e_{i,j}^0| \\ + \Gamma(2 - \alpha) \Delta t^\alpha O(\Delta t^{2-\alpha} + h_x^{2-\beta_x} + h_y^{2-\beta_y}).$$

With the supposed Equation (33) and Equation (18), we can obtain

$$\sup_{i,j} |e_{i,j}^m| \leq \sup_{i,j} |e_{i,j}^0| + \Gamma(2 - \alpha) \Delta t^\alpha O(\Delta t^{2-\alpha} + h_x^{2-\beta_x} + h_y^{2-\beta_y}) \\ + \sum_{s=1}^{m-1} (\omega_s^\alpha - \omega_{s+1}^\alpha) \Gamma(1 - \alpha) t_{m-s}^\alpha O(\Delta t^{2-\alpha} + h_x^{2-\beta_x} + h_y^{2-\beta_y}). \quad (35)$$

For the coefficients ω_{m-1}^α that satisfy $(1 - \alpha)m^{-\alpha} < \omega_{m-1}^\alpha$ and the time nodes $t_n^\alpha < t_m^\alpha, n < m$, we obtain the result

$$\sup_{i,j} |e_{i,j}^m| \leq \sup_{i,j} |e_{i,j}^0| + \omega_{m-1}^\alpha \Gamma(1 - \alpha) t_m^\alpha O(\Delta t^{2-\alpha} + h_x^{2-\beta_x} + h_y^{2-\beta_y}) \\ + \sum_{s=1}^{m-1} (\omega_{s-1}^\alpha - \omega_s^\alpha) \Gamma(1 - \alpha) t_{m-s}^\alpha O(\Delta t^{2-\alpha} + h_x^{2-\beta_x} + h_y^{2-\beta_y}) \\ \leq \sup_{i,j} |e_{i,j}^0| + \Gamma(1 - \alpha) t_m^\alpha O(\Delta t^{2-\alpha} + h_x^{2-\beta_x} + h_y^{2-\beta_y}).$$

That is

$$\sup_{i,j} |e_{i,j}^m| \leq C(\Delta t^{2-\alpha} + h_x^{2-\beta_x} + h_y^{2-\beta_y})$$

and C are constant independent of h_x, h_y and τ . $\square$

3.2. A Fast-Evaluation-in-Time Implementation

Analogous to Section 2.4 and the EFDM for stFDE (1), we can rewrite the EFDM for stFDE into the matrix from the following:

$$\mathbf{u}^m = (1 - \omega_2^\alpha) \mathbf{u}^{m-1} + \sum_{s=2}^{m-1} (\omega_s^\alpha - \omega_{s+1}^\alpha) \mathbf{u}^{m-s} + \omega_m^\alpha \mathbf{u}^0 + \mathbf{A} \mathbf{u}^{m-1} \\ + \tau^\alpha \Gamma(2 - \alpha) \mathbf{f}^{m-1}. \quad (36)$$

The stiffness Matrix $\mathbf{A}$ is identified with $\mathbf{A}$ in Section 2.4—where λ_x and λ_y are replaced by λ_3 and λ_4 , respectively—such that the matrix–vector multiplication can be carried out in $O(N \log N)$ computation by FFT. Furthermore, we used the fast method in time developed in [40] to approximate the RHS of (36).

To obtain the definition of the time fractional derivative $\partial_t^\alpha u(x, y, t_m)$ at time step t_m , we decomposed the integrating range $(0, t_m)$ into the two integrating ranges of (t_{m-1}, t_m) and $(0, t_{m-1})$. We evaluated the integral on the first range by the standard L^1 approximation and approximated the second term via integration by parts as follows:

$$\int_0^{t_{m-1}} \frac{\partial u(x, y, s)}{\partial s} (t_m - s)^{-\alpha} ds = \frac{u(x, y, t_{m-1})}{\tau^\alpha} - \frac{u_0}{t_m^\alpha} - \alpha \int_0^{t_{m-1}} \frac{u(x, y, s)}{(t_m - s)^{\alpha+1}} ds.$$

We can find that the integral on the RHS can be approximated by the following recursive relation:

$$\begin{aligned} & \sum_{k=1}^{M_{exp}} w_k U_{hist,k}(x_i, y_j, t_m) \\ & \approx \sum_{k=1}^{M_{exp}} w_k \left\{ e^{-s_k \tau} U_{hist,k}(x_i, y_j, t_{m-1}) + \frac{e^{s_k \tau}}{s_k^2 \tau} [(e^{-s_k \tau} - 1 + s_k \tau) u(x_i, y_j, t_{m-1}) \right. \\ & \quad \left. + (1 - e^{-s_k \tau} (1 + s_k \tau)) u(x_i, y_j, t_{m-2})] \right\}. \end{aligned}$$

Here, $\{s_k\}_{k=1}^{M_{exp}}$ and $\{w_k\}_{k=1}^{M_{exp}}$ are positive.

A fast-evaluation-in-time EFDM(TFFDM) is formulated as follows:

$$\begin{aligned} \mathbf{u}^m &= \mathbf{A} \mathbf{u}^{m-1} + \Gamma(2 - \alpha) \tau^\alpha \mathbf{f}^m + \alpha \mathbf{u}^{m-1} + (1 - \alpha) \tau^\alpha \mathbf{u}^0 \\ &+ \alpha(1 - \alpha) \tau^\alpha \sum_{k=1}^{M_{exp}} (a_k \mathbf{U}_{hist,k}^{m-1} + b_k \mathbf{u}^{m-1} + c_k \mathbf{u}^{m-2}), \end{aligned} \quad (37)$$

with

$$\begin{aligned} a_k &:= w_k e^{-s_k \tau}, \\ b_k &:= w_k f e^{s_k \tau} s_k^2 \tau (e^{-s_k \tau} - 1 + s_k \tau), \\ c_k &:= w_k \frac{e^{s_k \tau}}{s_k^2 \tau} (1 - e^{-s_k \tau} (1 + s_k \tau)), \\ \mathbf{U}_{hist,k}^{m-1} &:= \mathbf{U}_{hist,k}(\cdot, t_{m-1}). \end{aligned} \quad (38)$$

Theorem 6. The TFFDM requires a memory of $O(N \log M)$ and costs a complexity of computation $O(MN \log(MN))$.

The detail proof can be find in [40], so we will not reiterate it here.

4. Numerical Experiments

We provide some numerical examples to illustrate the performance of the finite difference method in this section. We will also investigate the fast algorithm, which significantly reduces the computational complexity of the numerical method from $O(MN^2 + M^2N)$ to $O(MN \log(MN))$ when compared with the traditional algorithm for the finite difference method. Furthermore, the TFFDM was found to reduce the memory requirement from $O(MN^2)$ to $O(N \log M)$.

4.1. The Convergence and Performance of Both Methods

We considered the problems in Equations (3) and (1) with the spacial fractional order $\beta_x = \beta_y = \beta$ and $\kappa = 0.5$, as well as the diffusion coefficient $v_x = v_y = 0.5$. The spatial domain was $\Omega = (0, 1) \times (0, 1)$ and the time interval was $[0, T] = [0, 1]$. The true solution was chosen to be $u(x, y, t) = t^2 x^2 (1 - x)^2 y^2 (1 - y)^2$. The source term f , the initial condition u_0 and the boundary data could be computed accordingly. For convenience, we chose $N_x = N_y$, such that $h := h_x = h_y$. The convergence rates r_t and r_s were measured such that

$$\|u - u_h\|_{L_\infty(0,T;L_2(\Omega))} \leq C(\tau^{r_t} + h^{r_s}).$$

We present the results of both the FDM and FFDM for the sFDE in Tables 1–3 and the stFDE in Tables 4–6, respectively. The performance of the FDM and FFDM for each example with $\alpha = 1$, as well as $\alpha = 0.3, 0.5$ and 0.7 at fixed spatial meshes and the successively refined time steps are in Table 2 and Table 5, respectively. We present the performance of the FDM and FFDM for each an example with $\beta = 0.3, 0.5$ and 0.7 at successively refined meshes in space and time in Table 2 and Table 5, respectively. In Table 3 and in Table 6, we respectively show the computational performance of both the methods with $\beta = 0.5$,

as well as $\alpha = 0.5$ and 0.7 , at successively refined temporal meshes with fixed spatial grid points. And the results confirmed the theory analyzed in the Sections 2 and 3.

Table 1. Results and convergence rates of r_t for (3), with $\beta_x = \beta_y = 0.5$ and $N_x = N_y = 2^8$.

M	FDM			FFDM		
	L^2 Error	r_t	CPU Time	L^2 Error	r_t	CPU Time
2^4	0.0021	-	14.25 s	0.0021	-	8.78 s
2^5	0.0010	1.0704	3 min 52 s	0.0010	1.0704	17.05 s
2^6	5.2135×10^{-4}	0.9397	10 min 27 s	5.2135×10^{-4}	0.9397	33.63 s
2^7	-	-	Out of memory	2.6067×10^{-4}	1.0000	66.67 s
2^8	-	-	Out of memory	1.3033×10^{-4}	1.0001	2 min 14 s

Table 2. Results and convergence rates r_s for (3) with $M = 2^{12}$.

$\beta_x = \beta_y$	$N_x = N_y$	FDM		FFDM	
		L^2 -Error	r_s	L^2 -Error	r_s
0.3	2^3	2.3313×10^{-4}	-	2.3313×10^{-4}	-
	2^4	7.9127×10^{-5}	1.5589	7.9127×10^{-5}	1.5589
	2^5	2.0332×10^{-5}	1.9604	2.0332×10^{-5}	1.9604
	2^6	3.1985×10^{-6}	2.6683	3.1985×10^{-6}	2.6683
0.5	2^3	6.6565×10^{-4}	-	6.6565×10^{-4}	-
	2^4	2.6633×10^{-4}	1.3215	2.6633×10^{-4}	1.3215
	2^5	9.5791×10^{-5}	1.4753	9.5791×10^{-5}	1.4753
	2^6	2.8968×10^{-5}	1.7254	2.8968×10^{-5}	1.7254
	2^7	4.9270×10^{-6}	2.5557	4.9270×10^{-6}	2.5557
0.7	2^3	0.0016	-	0.0016	-
	2^4	7.3537×10^{-4}	1.1215	7.3537×10^{-4}	1.1215
	2^5	3.1433×10^{-4}	1.2262	3.1433×10^{-4}	1.2262
	2^6	1.2657×10^{-4}	1.3123	1.2657×10^{-4}	1.3123
	2^7	4.6550×10^{-5}	1.4431	4.6550×10^{-5}	1.4431

Table 3. Results and CPU times for (3) with $\beta_x = \beta_y = 0.5$ ($M = N_x = N_y$).

M	FDM			FFDM		
	L^2 Error	Rate	CPU Time	L^2 Error	Rate	CPU Time
2^5	0.0013	-	1.43 s	0.0013	-	1.35 s
2^6	6.7777×10^{-4}	0.9392	15.0 s	6.7777×10^{-4}	0.9392	10.6 s
2^7	3.4326×10^{-4}	0.9820	4 min 38 s	3.4326×10^{-4}	0.9820	1 min 53 s
2^8	-	-	Out of memory	1.7335×10^{-4}	0.9856	10 min 56 s
2^9	-	-	Out of memory	8.7323×10^{-5}	0.9893	1 h 33 min 49 s
2^{10}	-	-	Out of memory	4.3901×10^{-5}	0.9921	12 h 51 min 6 s

Table 4. Results and convergence rates r_t for (1) with $\beta_x = \beta_y = 0.5$ and $N_x = N_y = 2^8$.

α	M	FDM		FFDM	
		L^2 Error	r_t	L^2 Error	r_t
0.3	2^3	2.7251×10^{-4}	-	2.7251×10^{-4}	-
	2^4	9.1584×10^{-5}	1.5669	9.1584×10^{-5}	1.5669
	2^5	3.0983×10^{-5}	1.5699	3.0983×10^{-5}	1.5699
	2^6	1.0531×10^{-5}	1.5568	1.0531×10^{-5}	1.5568
	2^7	3.6480×10^{-6}	1.5295	3.6480×10^{-6}	1.5295
0.5	2^3	6.9408×10^{-4}	-	6.9408×10^{-4}	-
	2^4	2.5789×10^{-4}	1.4283	2.5789×10^{-4}	1.4283
	2^5	9.4645×10^{-5}	1.4462	9.4645×10^{-5}	1.4462
	2^6	3.4528×10^{-5}	1.4548	3.4528×10^{-5}	1.4548
	2^7	1.2575×10^{-5}	1.4572	1.2575×10^{-5}	1.4572
0.7	2^3	0.0015	-	0.00154	-
	2^4	6.3110×10^{-4}	1.2490	6.3110×10^{-4}	1.2490
	2^5	2.6111×10^{-4}	1.2732	2.6111×10^{-4}	1.2732
	2^6	1.0735×10^{-4}	1.2823	1.0735×10^{-4}	1.2823
	2^7	4.3973×10^{-5}	1.2876	4.3973×10^{-5}	1.2876

Table 5. Results and convergence rates r_s for (1) with $\alpha = 0.5$ and $M = 2^{12}$.

$\beta_x = \beta_y$	$N_x = N_y$	FDM		FFDM	
		L^2 -Error	r_s	L^2 -Error	r_s
0.3	2^3	3.4169×10^{-4}	-	3.4169×10^{-4}	-
	2^4	1.2196×10^{-4}	1.4863	1.2196×10^{-4}	1.4863
	2^5	3.7410×10^{-5}	1.7049	3.7410×10^{-5}	1.7049
	2^6	7.9030×10^{-6}	2.2430	7.9030×10^{-6}	2.2430
0.5	2^3	9.6137×10^{-4}	-	9.6137×10^{-4}	-
	2^4	3.9324×10^{-4}	1.2897	3.9324×10^{-4}	1.2897
	2^5	1.4755×10^{-4}	1.4142	1.4755×10^{-4}	1.4142
	2^6	5.0551×10^{-5}	1.5454	5.0551×10^{-5}	1.5454
0.7	2^3	0.0023	-	0.0023	-
	2^4	0.0011	1.0641	0.0011	1.0641
	2^5	4.7942×10^{-4}	1.1981	4.7942×10^{-4}	1.1981
	2^6	1.9961×10^{-4}	1.2641	1.9961×10^{-4}	1.2641

Table 6. Results and CPU times for (1) with $\beta_x = \beta_y = 0.5$ ($M = N_x = N_y$).

α	M	FDM			FFDM		
		L^2 Error	Rate	CPU Time	L^2 Error	Rate	CPU Time
0.5	2^5	8.2859×10^{-4}	-	1.80 s	8.2859×10^{-4}	-	1.88 s
	2^6	4.2568×10^{-4}	0.9598	15.3 s	4.2568×10^{-4}	0.9598	12.3 s
	2^7	2.1763×10^{-4}	0.9690	3 min 59 s	2.1763×10^{-4}	0.9690	1 min 39 s
	2^8	1.1066×10^{-4}	0.9757	6 h 37 min 7 s	1.1066×10^{-4}	0.9757	12 min 43 s
	2^9	-	-	Out of memory	5.6020×10^{-5}	0.9821	1 h 42 min 2 s
	2^{10}	-	-	Out of memory	2.8261×10^{-5}	0.9871	14 h 45 min 9 s
0.7	2^5	8.8750×10^{-4}	-	1.56 s	8.8750×10^{-4}	-	1.47 s
	2^6	4.4614×10^{-4}	0.9923	15.9 s	4.4614×10^{-4}	0.9923	12.3 s
	2^7	2.2392×10^{-4}	0.9945	3 min 59 s	2.2392×10^{-4}	0.9945	1 m 39 s
	2^8	1.1212×10^{-4}	0.9979	6 h 34 min 35 s	1.1212×10^{-4}	0.9979	12 min 39 s
	2^9	-	-	Out of memory	5.6031×10^{-5}	1.0007	1 h 45 min 6 s
	2^{10}	-	-	Out of memory	2.7954×10^{-5}	1.0032	15 h 38 min 33 s

By comparing the results that are shown in the tables, we obtained the following observations:

- (i) FFDM can effectively reduce the costs of computation and memory requirement, which is to say that FFDM has a much better efficiency than FDM.
- (ii) Both methods have the same accuracy and the convergence results that verify the theory analyzed in Sections 2 and 3.
- (iii) In Table 4, one can see that we only show the two cases where $\alpha = 0.5$ and $\alpha = 0.7$ because the result is unsteady when $\alpha = 0.3$ is for $\lambda > 1$ as they did not satisfy the CFL condition, which strongly confirm the theories.

4.2. Anomalous Diffusive Transport

We simulated the anomalous diffusive transport described by a one-dimensional analog of Problems (1) and (3) with the homogeneous Dirichlet boundary condition, as well as with $\Omega = (0, 1)$, $[0, T] = [0, 1]$, $h = 1/256$, and $\tau = 1/250$.

Example 1. In order to figure out the connections and differences between the integer equation $\partial_t u + K\partial_x u = 0$ and the fractional-order differential equation $\partial_t u + \mathcal{L}_x^\beta u = 0$, K can be 1 or -1 and $\mathcal{L}_x^\beta$ can be the left-sided fractional operator $\partial_x^{\beta,-}$ or the right-sided fractional operator $\partial_x^{\beta,+}$, as defined in (2). We set the same function $u_0 = \exp(-x^2)$ as our initial condition. We used the Euler scheme to respectively obtain the numerical solutions.

In Figures 1 and 2, we display the numerical solutions of the different models evaluated at $t = 0, 0.3, 0.6$ and 0.9 . We can see that the moving direction is toward the coordinate directions in both figures. The difference is that the integer model moves in parallel, while the fractional model moves in a downward trend.

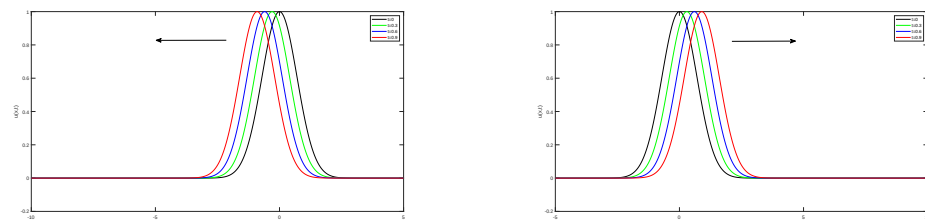


Figure 1. The $\partial_t u - \partial_x u = 0$ expression is on the (left) and $\partial_t u + \partial_x u = 0$ is on the (right).

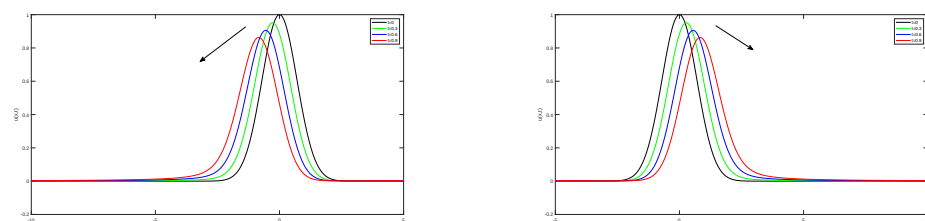


Figure 2. The $\partial_t u + \partial_x^{\beta,+} u = 0$ expression is on the (left) and $\partial_t u + \partial_x^{\beta,-} u = 0$ is on the (right) with $\beta = 0.9$.

In Figure 3, we display a plot of the exact solutions (g-), the FDM solutions (bd) and the FFDM solutions (r*) in x and t ($t = 1$) with $N = 2^6$ and $M = N$ to show the accuracy of the method. We can see that the numerical solutions fit the exact solution accurately.

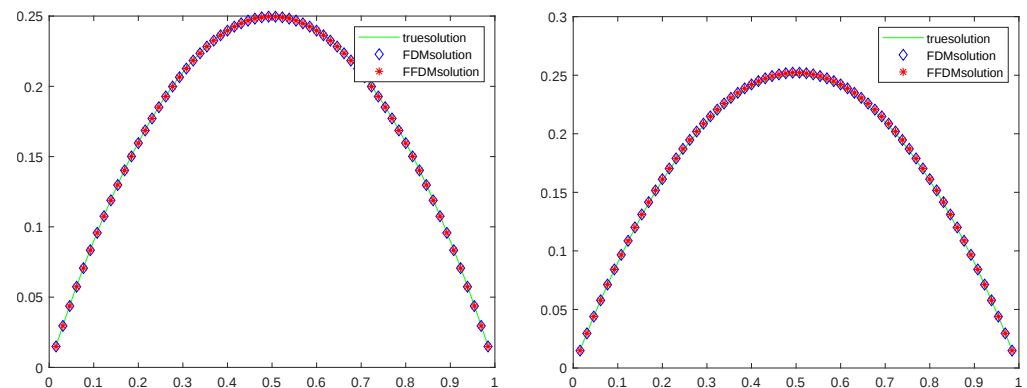


Figure 3. The accuracy of both methods (sFAD on the (left) and tsFADE on the (right)).

5. Discussion and Conclusions

We developed and analyzed an EFDM that satisfied a bound-preserving principle for space–time fractional advection equations with orders of $0 < \alpha$ and $\beta \leq 1$. Under the CFL condition, the EFDM is bound-preserved and convergent. And the FFDM for stFAE has the same accuracy with FDM. But, FFDM can reduce the costs of computation and the requirements of memory effectively, which is to say that the FFDM has a much better efficiency than FDM. However, both methods have a rate of time convergence of 1. To obtain a higher accuracy in the numerical method, the discontinuous Galerkin method [41–43] can be adopted for space and/or time derivatives, and this can lead to interesting outcomes. It could be that a compact finite difference method [44,45] can be adopted for time derivatives in our future work, which could thus lead to a better convergence.

Author Contributions: Conceptualization, J.G. and H.C.; methodology, J.G.; software, J.G.; validation, J.G. and H.C.; formal analysis, J.G.; investigation, H.C.; resources, H.C.; data curation, H.C.; writing—original draft preparation, J.G.; writing—review and editing, H.C.; visualization, J.G.; supervision, H.C.; project administration, H.C.; funding acquisition, H.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Natural Science Foundation of the Shandong Province of China (grant number ZR2021QA062).

Data Availability Statement: The data underlying the results presented in this paper are not publicly available at this time but may be obtained from the authors upon reasonable request.

Acknowledgments: The authors are grateful to H. Wang for their help in completing the article.

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

References

- Metzler, R.; Klafter, J. The restaurant at the end of the random walk: Recent developments in the description of anomalous transport by fractional dynamics. *J. Phys. A Math. Gen.* **2004**, *37*, R161–R208. [\[CrossRef\]](#)
- Podlubny, I. *Fractional Differential Equations*; Academic Press: Cambridge, MA, USA, 1999.
- Diethelm, K.; Ford, N. A note on the well-posedness of terminal value problems for fractional differential equations. *J. Integral Equ. Appl.* **2017**, *30*, 371–376. [\[CrossRef\]](#)
- Zhang, X.; Yu, L.; Jiang, J.; Wu, Y.; Cui, Y. Positive Solutions for a Weakly Singular Hadamard-Type Fractional Differential Equation with Changing-Sign Nonlinearity. *J. Funct. Spaces* **2020**, *2020*, 5623589.
- Zhang, X.; Yu, L.; Jiang, J.; Wu, Y.; Cui, Y. Solutions for a Singular Hadamard-Type Fractional Differential Equation by the Spectral Construct Analysis. *J. Funct. Spaces* **2020**, *2020*, 8392397. [\[CrossRef\]](#)
- He, J.; Zhang, X.; Liu, L.; Wu, Y.; Cui, Y. A singular fractional Kelvin-Voigt model involving a nonlinear operator and their convergence properties. *Bound. Value Probl.* **2019**, *2019*, 112. [\[CrossRef\]](#)
- Wu, J.; Zhang, X.; Liu, L.; Wu, Y.; Cui, Y. The convergence analysis and error estimation for unique solution of a p-Laplacian fractional differential equation with singular decreasing nonlinearity. *Bound. Value Probl.* **2018**, *2018*, 82. [\[CrossRef\]](#)
- Chen, H.; Wang, H. Numerical simulation for conservative fractional diffusion equations by an expanded mixed formulation. *J. Comput. Appl. Math.* **2016**, *296*, 480–498. [\[CrossRef\]](#)
- Ervin, V.; Heuer, N.; Roop, J. Regularity of the solution to 1-D fractional order diffusion equations. *Math. Comput.* **2018**, *87*, 2273–2294. [\[CrossRef\]](#)

10. Zheng, X.; Wang, H. An error estimate of a numerical approximation to a hidden-memory variable-order space-time fractional diffusion equation. *SIAM J. Numer. Anal.* **2020**, *58*, 2492–2514. [\[CrossRef\]](#)
11. Zheng, X.; Wang, H. An optimal-order numerical approximation to variable-order space-fractional diffusion equations on uniform or graded meshes. *SIAM J. Numer. Anal.* **2020**, *58*, 330–352. [\[CrossRef\]](#)
12. Deng, W.; Li, C.; Lü, J. Stability analysis of linear fractional differential system with multiple time delays. *Nonlinear Dyn.* **2007**, *48*, 409–416. [\[CrossRef\]](#)
13. Zheng, X.; Wang, H. A hidden-memory variable-order fractional optimal control model: Analysis and approximation. *SIAM J. Control Optim.* **2021**, *59*, 1851–1880. [\[CrossRef\]](#)
14. Yu, B.; Zheng, X.; Zhang, P.; Zhang, L. Computing solution landscape of nonlinear space-fractional problems via fast approximation algorithm. *J. Comput. Phys.* **2022**, *468*, 111513. [\[CrossRef\]](#)
15. Gu, X.; Huang, T.; Ji, C.; Carpentieri, B.; Alikhanov, A.A. Fast iterative method with a second-order implicit difference scheme for time-space fractional convection-diffusion equation. *J. Sci. Comput.* **2017**, *72*, 957–985. [\[CrossRef\]](#)
16. Ren, T.; Li, S.; Zhang, X.; Liu, L. Maximum and minimum solutions for a nonlocal p-Laplacian fractional differential system from eco-economical processes. *Bound. Value Probl.* **2017**, *2017*, 118. [\[CrossRef\]](#)
17. Wu, J.; Zhang, X.; Liu, L.; Wu, Y.; Cui, Y. Convergence analysis of iterative scheme and error estimation of positive solution for a fractional differential equation. *Math. Model. Anal.* **2018**, *23*, 611–626.
18. Garrappa, R.; Moret, I.; Popolizio, M. Solving the time-fractional Schrödinger equation by Krylov projection methods. *J. Comput. Phys.* **2015**, *293*, 115–134. [\[CrossRef\]](#)
19. Jia, L.; Chen, H.; Wang, H. Mixed-type Galerkin variational principle and numerical simulation for a generalized nonlocal elastic model. *J. Sci. Comput.* **2017**, *71*, 660–681. [\[CrossRef\]](#)
20. Liu, F.; Anh, V.; Turner, I. Numerical solution of the space fractional Fokker-Planck equation. *J. Comput. Appl. Math.* **2004**, *166*, 209–219. [\[CrossRef\]](#)
21. Jin, B.; Li, B.; Zhou, Z. Discrete maximal regularity of time-stepping schemes for fractional evolution equations. *Numer. Math.* **2018**, *138*, 101–131. [\[CrossRef\]](#)
22. Wang, H.; Yang, D. Wellposedness of variable-coefficient conservative fractional elliptic differential equations. *SIAM Numer. Anal.* **2013**, *51*, 1088–1107. [\[CrossRef\]](#)
23. Gao, J.; Zhao, M.; Du, N.; Guo, X.; Wang, H.; Zhang, J. A finite element method for space-time directional fractional diffusion partial differential equations in the plane and its error analysis. *J. Comput. Appl. Math.* **2019**, *362*, 354–365. [\[CrossRef\]](#)
24. Meerschaert, M.M.; Tadjeran, C. Finite difference approximations for two-sided space-fractional partial differential equations. *Appl. Numer. Math.* **2006**, *56*, 80–90. [\[CrossRef\]](#)
25. Roos, H.G.; Stynes, M.; Tobiska, L. *Robust Numerical Methods for Singularly Perturbed Differential Equations*; Springer: Berlin/Heidelberg, Germany, 2008.
26. Zhang, X.; Shu, C. On maximum-principle-satisfying high order schemes for scalar conservation laws. *J. Comput. Phys.* **2010**, *229*, 3091–3120. [\[CrossRef\]](#)
27. Zhang, X.; Shu, C. Maximum-principle-satisfying and positivity-preserving high order schemes for conservation laws: Survey and new developments. *Proc. R. Soc. A* **2011**, *467*, 2752–2776. [\[CrossRef\]](#)
28. Zhang, X.; Shu, C. Positivity-preserving high order discontinuous Galerkin schemes for compressible Euler equations with source terms. *J. Comput. Phys.* **2011**, *230*, 1238–1248. [\[CrossRef\]](#)
29. Du, J.; Yang, Y. High-order bound-preserving discontinuous Galerkin methods for multicomponent chemically reacting flows. *J. Comput. Phys.* **2022**, *469*, 111548. [\[CrossRef\]](#)
30. Zhang, Y.; Zhang, X.; Shu, C. Maximum-principle-satisfying second order discontinuous Galerkin schemes for convection-diffusion equations on triangular meshes. *J. Comput. Phys.* **2003**, *234*, 295–316. [\[CrossRef\]](#)
31. Jia, J.; Wang, H. A fast finite volume method for conservative space-time fractional diffusion equations discretized on space-time locally refined meshes. *Comput. Math. Appl.* **2019**, *78*, 1345–1356. [\[CrossRef\]](#)
32. Jia, J.; Wang, H. A fast finite volume method on locally refined meshes for fractional diffusion equations. *East Asian J. Appl. Math.* **2019**, *9*, 755–779.
33. Wang, H.; Basu, T.S. A fast finite difference method for two-dimensional space-fractional diffusion equations. *SIAM Sci. Comput.* **2012**, *34*, A2444–A2458. [\[CrossRef\]](#)
34. Du, N.; Wang, H. A Fast Finite Element Method for Space-Fractional Dispersion Equations on Bounded Domains in R^2 . *SIAM J. Sci. Comput.* **2015**, *37*, A1614–A1635. [\[CrossRef\]](#)
35. Wang, H.; Wang, K.; Sircar, T. A direct $O(N \log^2 N)$ finite difference method for fractional diffusion equations. *J. Comput. Phys.* **2010**, *229*, 8095–8104. [\[CrossRef\]](#)
36. Lin, Y.; Xu, C. Finite difference/spectral approximation for the time-fractional diffusion equation. *J. Comput. Phys.* **2007**, *225*, 1533–1552. [\[CrossRef\]](#)
37. De Moura, C.A.; Kubrusly, C.S. *The Courant-Friedrichs-Lewy Condition*; AMC: Leawood, UK, 2013.
38. Chan, R.H.; Ng, M.K. Conjugate gradient methods for Toeplitz systems. *SIAM Rev.* **1996**, *38*, 427–482. [\[CrossRef\]](#)
39. Davis, P.J. *Circulant Matrices*; American Mathematical Society: Providence, RI, USA, 2012.
40. Jiang, S.; Zhang, J.; Zhang, Q.; Zhang, Z. Fast evaluation of the Caputo fractional derivative and its applications to fractional diffusion equations. *arXiv* **2015**, arXiv:1511.03453.

41. Xu, Q.; Hesthaven, J.S.; Discontinuous Galerkin Method For Fractional Convection-Diffusion Equations. *SIAM J. Numer. Anal.* **2014**, *52*, 405–423. [[CrossRef](#)]
42. Baccouch, M.; Temimi, H. Analysis of Optimal Error Estimates and Superconvergence of the Discontinuous Galerkin Method for Convection-Diffusion Problems in one Space Dimension. *Int. J. Numer. Anal. Model.* **2016**, *13*, 403–434.
43. Deng, W.; Hesthaven, J.S. Local discontinuous Galerkin methods for fractional diffusion equations. *ESAIM Math. Modell. Numer. Anal.* **2013**, *47*, 1845–1864. [[CrossRef](#)]
44. Ren, J.; Sun, Z.; Zhao, X. Compact difference scheme for the fractional sub-diffusion equation with Neumann boundary conditions. *J. Comput. Phys.* **2012**, *232*, 456–467. [[CrossRef](#)]
45. Zhang, Y.; Sun, Z.; Zhao, X. Compact alternating direction implicit scheme for the two-dimensional fractional diffusion-wave equation. *SIAM J. Numer. Anal.* **2012**, *50*, 1535–1555. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.