



## Article

# A Fast Preconditioned Semi-Implicit Difference Scheme for Strongly Nonlinear Space-Fractional Diffusion Equations

Yu-Yun Huang <sup>1,\*</sup>, Xian-Ming Gu <sup>1,\*</sup> , Yi Gong <sup>2,\*</sup>, Hu Li <sup>3</sup> , Yong-Liang Zhao <sup>4</sup> and Bruno Carpentieri <sup>5</sup><sup>1</sup> Institute of Mathematics, School of Economic Mathematics, Southwestern University of Finance and Economics, Chengdu 611130, China; 219070100003@swufe.edu.cn<sup>2</sup> Fundamental Department, Shanghai Customs College, Shanghai 201204, China<sup>3</sup> School of Mathematics, Chengdu Normal University, Chengdu 611130, China; 063029@cdnu.edu.cn<sup>4</sup> School of Mathematical Sciences, Sichuan Normal University, Chengdu 610068, China; ylzhaos@sicnu.edu.cn<sup>5</sup> Faculty of Computer Science, Free University of Bozen-Bolzano, 39100 Bolzano, Italy;

Bruno.Carpentieri@unibz.it

\* Correspondence: guxm@swufe.edu.cn (X.-M.G.); gongyi@shcc.edu.cn (Y.G.)

**Abstract:** In this paper, we propose a semi-implicit difference scheme for solving one-dimensional nonlinear space-fractional diffusion equations. The method is first-order accurate in time and second-order accurate in space. It uses a fractional central difference formula and the backward Euler method to approximate its space and time derivatives, respectively. Stability and convergence properties of the proposed scheme are proved with the help of a discrete Grönwall inequality. Moreover, we extend the method to the solution of two-dimensional nonlinear models. A fast matrix-free implementation based on preconditioned Krylov subspace methods is presented for solving the discretized linear systems. The resulting fast preconditioned semi-implicit difference scheme reduces the memory requirement of conventional semi-implicit difference schemes from  $\mathcal{O}(N_s^2)$  to  $\mathcal{O}(N_s)$  and the computational complexity from  $\mathcal{O}(N_s^3)$  to  $\mathcal{O}(N_s \log N_s)$  in each iterative step, where  $N_s$  is the number of space grid points. Experiments with two numerical examples are shown to support the theoretical findings and to illustrate the efficiency of our proposed method.

**Keywords:** nonlinear space fractional diffusion equation; semi-implicit scheme; stability analysis; Krylov subspace methods



**Citation:** Huang, Y.-Y.; Gu, X.-M.; Gong, Y.; Li, H.; Zhao, Y.-L.; Carpentieri, B. A Fast Preconditioned Semi-Implicit Difference Scheme for Strongly Nonlinear Space-Fractional Diffusion Equations. *Fractal Fract.* **2021**, *5*, 230. <https://doi.org/10.3390/fractalfract5040230>

Academic Editor: Manuel Ortigueira

Received: 26 October 2021

Accepted: 13 November 2021

Published: 18 November 2021

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



**Copyright:** © 2021 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

In this paper, we are interested in developing efficient numerical solutions of strongly nonlinear space-fractional diffusion equations (NSFDEs) of the following form:

$$\begin{cases} \frac{\partial u(x,t)}{\partial t} = a(u) \frac{\partial^\alpha u(x,t)}{\partial |x|^\alpha} + f(x,t), & (x,t) \in (0, L_x) \times (0, T], \\ u(x,0) = \phi(x), & x \in [0, L_x], \\ u(0,t) = \varphi(t), \quad u(L_x,t) = \psi(t), & t \in [0, T], \end{cases} \quad (1)$$

where  $a(u)$ ,  $f(x,t)$ ,  $\phi(x)$ ,  $\varphi(t)$ ,  $\psi(t)$  are known functions. Function  $u(x,t)$  may represent, for example, the concentration of a particle plume undergoing anomalous diffusion. The diffusion coefficient  $a(u)$  is assumed to be positive, and the forcing function  $f(x,t)$  represents the source or sink term [1,2]. In addition, we assume that Equation (1) has a smooth solution  $u(x,t)$ , and that there exist real positive numbers  $\epsilon_1, \epsilon_2, \epsilon, a_0, a_1$  and  $L$  satisfying the following:

$$0 < a_0 \leq a(u(x,t) + \epsilon_1) \leq a_1, \quad (2)$$

$$|a(u(x,t) + \epsilon_1) - a(u(x,t) + \epsilon_2)| \leq L|\epsilon_1 - \epsilon_2|, \quad (3)$$

where  $|\epsilon_1| \leq \epsilon$ ,  $|\epsilon_2| \leq \epsilon$ , and  $(x, t) \in [0, L_x] \times [0, T]$ . Moreover, the symbol  $\frac{\partial^\alpha}{\partial |x|^\alpha}$  denotes the Riesz fractional derivative operator [3–5] for  $1 < \alpha \leq 2$ , defined as follows:

$$\frac{\partial^\alpha u(x, t)}{\partial |x|^\alpha} = -\frac{1}{2 \cos(\frac{\alpha\pi}{2}) \Gamma(2 - \alpha)} \frac{\partial^2}{\partial x^2} \int_{-\infty}^{\infty} |x - \xi|^{1-\alpha} u(\xi, t) d\xi, \quad (4)$$

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

In the past few decades, a significant amount of research work has been devoted to the analysis and solution of fractional partial differential equations (fPDEs) especially for modeling anomalous diffusion [6,7], subdiffusion and superdiffusion processes that can be observed in many real physical systems, such as heat transfer problems and wave analysis [8,9]. Space fractional equations are obtained by replacing the traditional space derivative with a fractional derivative in the integer-order diffusion equation and can describe heterogeneous substances with memory and genetic properties very effectively [10]. Closed-form analytic solutions are available only for limited classes of fPDEs. For solving more general equations, the only viable approach is to use discretization methods such as finite difference [11–15], finite element [16,17], finite volume [18,19] and spectral [20] methods.

The Riemann–Liouville fractional derivative is often used to model the dynamics of discrete systems with long-range interaction. Several studies have applied the weighted and shifted Grünwald difference (WSGD) formula to approximate the Riemann–Liouville derivative; see [14,21,22] for details. By using the WSGD formula for two-sided fractional derivatives combining the compact technique, Hao et al. presented a fourth-order difference approximation for the space fractional derivatives [23]. In addition, Çelik and Duman used a fractional centered difference operator to approximate the Riesz fractional derivative with second-order accuracy and to solve the Riesz SFDE (RSFDE) on a finite domain [24]. Later, Yang et al. [25] used the Diethelm method [15] to derive a new  $\mathcal{O}(\Delta x^{3-\alpha})$  approximation for discretizing the Riesz fractional derivatives of order  $\alpha \in (1, 2)$ , where  $\Delta x$  is the step size. Then, a new finite difference scheme for solving the RSFDE was obtained by discretizing the first-order time derivative using the Crank–Nicolson (CN) method. In contrast with the classical diffusion operator  $\Delta$ , Riesz fractional derivative [3,5] is a special linear combination of left- and right-sided Riemann–Liouville fractional derivatives. By exploiting this property, Ding and Li [4] established a novel class of high-order numerical algorithms for Riesz derivatives through constructing new generating functions. In 2018, Lin et al. [26] studied the CN temporal discretization with various high-order spatial difference schemes for RSFDEs with variable diffusion coefficients and gave the unconditional stability and convergence analysis for temporally second-order and spatially  $j$ th-order ( $j = 1, 2, 3, 4$ ) difference schemes for such equations with variable coefficients. In 2019, Lin et al. [27] studied the CN alternative direction implicit (CN-ADI) method for two-dimensional RSFDEs with nonseparable coefficients. They showed under mild assumptions the unconditional stability of the CN-ADI method in discrete  $\ell^2$ -norm and the consistency of cross perturbation terms arising from the CN-ADI method. We point the reader to [3,12,13,28] for other recent works on the numerical solution of fPDEs with the Riesz fractional derivative.

Due to their non-local nature, numerical approximations of space fractional operators generally lead to dense matrices. Conventional direct solution methods for linear systems based on variants of the Gaussian elimination algorithm may not be affordable to use because of their high computational complexity and large memory costs. The quest for fast algorithms for solving linear systems originated in the numerical discretization of fPDEs has become a topic of great interest in the recent literature on numerical methods for fPDEs. Lin et al. [29] proposed a splitting preconditioner for fast solution of Toeplitz-like linear systems arising from one- and two-dimensional time-dependent RSFDEs with variable diffusion coefficients [27]. Some other preconditioners are being applied also to solving the

discretized (non-)linear systems from the solution of linear and non-linear problems; see, for example, [1,2,30,31] and the references therein.

In this paper, we study the efficient numerical solution of strongly NSFDEs because they are more suitable than linear models to describe some difficult physical processes, such as fractional distillation in nonlinear space, and when the diffusion coefficient is related to the concentration of a particle plume [31]. However, to the best of our knowledge, only a few studies on fast numerical schemes for nonlinear or semi-linear problems have been published in the literature; see, for example, refs. [32–34]. Our main contribution to the subject can be summarized as follows: first, for this class of strongly NSFDEs, we propose a semi-implicit difference scheme that can avoid solving the discretized nonlinear systems [35], and analyze its stability and convergence properties. Then, we extend the method to the solution of the two-dimensional formulation of Equation (1). In order to compute an efficient iterative solution of the discretized systems, a suitable preconditioning technique is developed and presented in the paper. The complexity and memory requirement of the resulting semi-implicit difference scheme are fairly lower, compared to the fully implicit difference scheme that requires to solve the discretized nonlinear systems, motivating our choice of the semi-implicit difference scheme for Equation (1) [36]. Moreover, unlike the traditional (semi-)implicit difference schemes, our proposed semi-implicit difference scheme can be implemented as a matrix-free method because it does not store any of the coefficient matrices involved in the whole solution process.

The rest of this article is organized as follows. In Section 2, we propose a semi-implicit difference scheme for Equation (1), and then we analyze its stability and convergence properties. In Section 3, we extend our method to the solution of the two-dimensional formulation of Equation (1). A fast matrix-free preconditioned iterative algorithm for solving the discretized linear system arising in the analysis of the two-dimensional case of Equation (1) is developed, and it is presented in Section 4. In Section 5, we report on numerical experiments to illustrate the efficiency of the proposed semi-implicit difference method. Some conclusions arising from this work are drawn in Section 6.

## 2. The Semi-Implicit Difference Scheme and Its Stability and Convergence Analysis

In this section, we propose a semi-implicit difference method to approximate the solution of Equation (1). The proposed method uses a fractional central difference formula to discretize the Riesz fractional derivative; it is stable and achieves first-order convergence in time and second-order convergence in space, respectively.

### 2.1. The Semi-Implicit Difference Scheme

We introduce the following notation. Let  $M, N$  be positive integers,  $h = L_x/N$ ,  $x_i = ih$ ,  $t_j = j\Delta t$  and  $\Delta t = T/M$ . We consider the sets of grid points  $\Omega_h := \{x_i | 0 \leq i \leq N\}$ ,  $\Omega_{\Delta t} := \{t_j | 0 \leq j \leq M\}$ ,  $\Omega_{h\Delta t} := \{(x_i, t_j) | 0 \leq i \leq N, 0 \leq j \leq M\}$ , and let  $v = \{v_i^j | 0 \leq i \leq N, 0 \leq j \leq M\}$  be a grid function defined on  $\Omega_{h\Delta t}$ . To simplify the notation, we write Equation (1) at the generic points  $(x_i, t_j)$  as follows:

$$\frac{\partial u(x_i, t_j)}{\partial t} = a(u(x_i, t_j)) \frac{\partial^\alpha u(x_i, t_j)}{\partial |x|^\alpha} + f(x_i, t_j), \quad 1 \leq i \leq N-1, 1 \leq j \leq M, \quad (5)$$

or, equivalently, as follows:

$$\frac{\partial u(x_i, t_j)}{\partial t} - a(u(x_i, t_{j-1})) \frac{\partial^\alpha u(x_i, t_j)}{\partial |x|^\alpha} = F'(x_i, t_j), \quad 1 \leq i \leq N-1, 1 \leq j \leq M, \quad (6)$$

where  $F'(x_i, t_j) = f(x_i, t_j) + [a(u(x_i, t_j)) - a(u(x_i, t_{j-1}))] \frac{\partial^\alpha u(x_i, t_j)}{\partial |x|^\alpha}$ . Additionally, we introduce a grid function  $U$  defined as follows:

$$U_i^j := u(x_i, t_j), \quad f_i^j := f(x_i, t_j), \quad 0 \leq i \leq N, 0 \leq j \leq M. \quad (7)$$

By these notations, the partial derivative of  $u(x, t)$  with respect to  $t$  evaluated at point  $(x_i, t_j)$  is expressed as follows:

$$\frac{\partial u(x_i, t_j)}{\partial t} = \frac{u(x_i, t_j) - u(x_i, t_{j-1})}{\Delta t} + R_{ij}^{(1)} \triangleq \frac{U_i^j - U_i^{j-1}}{\Delta t} + \mathcal{O}(\Delta t), \quad (8)$$

and the Riesz space fractional derivative as follows:

$$\frac{\partial^\alpha u(x_i, t_j)}{\partial |x|^\alpha} = -h^{-\alpha} \Delta_h^\alpha u(x_i, t_j) + R_{ij}^{(2)} \triangleq \delta_h^\alpha U_i^j + \mathcal{O}(h^2), \quad (9)$$

where  $\Delta_h^\alpha U_i^j = \sum_{k=i-N}^i g_k^{(\alpha)} U_{i-k}^j$  and  $g_k^{(\alpha)} = \frac{(-1)^k \Gamma(\alpha + 1)}{\Gamma(\alpha/2 - k + 1) \Gamma(\alpha/2 + k + 1)}$ ; refer to [24] for details. For brevity, we denote the following:

$$R_{ij} = -R_{ij}^{(1)} - R_{ij}^{(2)} + [a(u(x_i, t_j)) - a(u(x_i, t_{j-1}))] \frac{\partial^\alpha u(x_i, t_j)}{\partial |x|^\alpha}. \quad (10)$$

Finally, Equation (1) at grid points  $(x_i, t_j)$  is written as follows:

$$\frac{U_i^j - U_i^{j-1}}{\Delta t} - a(U_i^{j-1}) \delta_h^\alpha U_i^j = f_i^j + \widehat{R}_{ij}, \quad 1 \leq i \leq N-1, 1 \leq j \leq M, \quad (11)$$

where the terms  $\{\widehat{R}_{ij}\}$  are small and satisfy the inequality as follows:

$$|\widehat{R}_{ij}| \leq c_1(\Delta t + h^2), \quad 1 \leq i \leq N-1, 1 \leq j \leq M. \quad (12)$$

Since the quantities  $\{\widehat{R}_{ij}\}$  are small, we can omit them. By imposing the initial-boundary value conditions,

$$U_i^0 = \phi(x_j), \quad 0 \leq i \leq N, \quad (13)$$

$$U_0^j = \varphi(t_j), \quad U_N^j = \psi(t_j), \quad 1 \leq j \leq M, \quad (14)$$

the following semi-implicit difference scheme is obtained:

$$\begin{cases} \frac{u_i^j - u_i^{j-1}}{\Delta t} - a(u_i^{j-1}) \delta_h^\alpha u_i^j = f_i^j, & 1 \leq i \leq N-1, 1 \leq j \leq M, \\ u_i^0 = \phi(x_j), & 0 \leq i \leq N, \\ u_0^j = \varphi(t_j), \quad u_N^j = \psi(t_j), & 1 \leq j \leq M, \end{cases} \quad (15)$$

that requires to solve the discretized linear system at each time step.

## 2.2. Stability and Convergence Analysis

In this subsection, we study the stability and convergence properties of the numerical scheme given by Equation (15). Let  $u_i^j$  be the approximate solution of Equation (15), and introduce the error function as follows:

$$e_i^j = U_i^j - u_i^j, \quad i = 0, \dots, N, j = 0, \dots, M,$$

defined as

$$\begin{cases} \frac{e_i^j - e_i^{j-1}}{\Delta t} - a(u_i^{j-1}) \delta_h^\alpha e_i^j = [a(U_i^{j-1}) - a(u_i^{j-1})] \delta_h^\alpha U_i^j + \widehat{R}_{ij}, & 1 \leq i \leq N-1, 1 \leq j \leq M, \\ e_i^0 = 0, & 0 \leq i \leq N, \\ e_0^j = 0, \quad e_N^j = 0, & 1 \leq j \leq M, \end{cases} \quad (16)$$

where  $|\widehat{R}_{ij}| \leq c_2(\Delta t + h^2)$ . Denote  $E_j = [e_1^j, e_2^j, \dots, e_{N-1}^j]^\top$ , and assume the following:

$$\|E^j\|_\infty = |e_{\ell_j}| = \max_{1 \leq \ell \leq N-1} |e_\ell^j| \quad (0 \leq j \leq M, \quad 1 \leq \ell_j \leq N-1). \tag{17}$$

To prove the stability and assess the convergence order of the scheme (15), we need some preliminary theoretical results given in [24,35,37] that are recalled below.

**Lemma 1.** *Let*

$$g_k^{(\alpha)} = \frac{(-1)^k \Gamma(\alpha + 1)}{\Gamma(\alpha/2 - k + 1) \Gamma(\alpha/2 + k + 1)}$$

be the coefficients of the centered finite difference approximation (9) for  $k = 0, \pm 1, \pm 2, \dots$ , and  $\alpha > -1$ . Then, we have the following:

- (1)  $g_0^{(\alpha)} \geq 0$  and  $g_{-k}^{(\alpha)} = g_k^{(\alpha)} \leq 0$  for all  $|k| \geq 1$ ;
- (2)  $\sum_{-\infty}^{\infty} g_k^{(\alpha)} = 0$  and  $\sum_{k=i-N, k \neq 0}^i g_k^{(\alpha)} \geq -g_0$ .

**Lemma 2.** (Discrete Grönwall inequality) *Let  $\{F^k \mid k \leq 0\}$  be a sequence of non-negative numbers and let  $\Delta t > 0$  be such that the following is true:*

$$F^{k+1} \leq (1 + c\Delta t)F^k + (\Delta t)g, \quad k = 0, 1, 2, \dots, \tag{18}$$

where both  $c$  and  $g$  are non-negative scalars. Then, the following inequality holds:

$$F^k \leq e^{ck\Delta t} \left( F^0 + \frac{g}{c} \right) \tag{19}$$

with  $k = 0, 1, \dots$

At this stage, the following result on the numerical stability of the proposed one dimensional semi-implicit difference scheme can be proved:

**Theorem 1.** *Suppose that  $\{u_i^j \mid 0 \leq j \leq M, 0 \leq i \leq N\}$  is the solution of the semi-implicit difference scheme (15), and define the following:*

$$C_1 = \frac{c_1}{LM_\alpha} \exp(LM_\alpha T), \quad M_\alpha = \left| \max_{\substack{0 \leq x \leq L, \\ 1 \leq t \leq T}} \frac{\partial^\alpha u(x, t)}{\partial |x|^\alpha} \right|. \tag{20}$$

Then, if  $\Delta t$  and  $h$  satisfy the following condition

$$\Delta t \leq \frac{\epsilon}{2C_1}, \quad h \leq \sqrt{\frac{\epsilon}{2C_1}}, \tag{21}$$

the semi-implicit difference scheme (15) is stable, and the following inequality holds:

$$\|e^k\|_\infty \leq C_1(\Delta t + h^2), \quad 0 \leq k \leq n. \tag{22}$$

**Proof.** According to Equation (15), we know that  $\|e^k\|_\infty \leq C_1(\Delta t + h^2)$  holds for  $k = 0$ . Suppose that the inequality (22) holds for  $0 \leq k \leq l$ , when  $\Delta t$  and  $h$  satisfy the condition (21) and

$$\|e^k\|_\infty \leq \epsilon, \quad 0 \leq k \leq l. \tag{23}$$

Therefore, the following holds:

$$a_0 \leq a(u_i^j) \leq a_1, \quad |a(U_i^j) - a(u_i^j)| \leq L|e_i^j|, \quad 1 \leq i \leq N-1, \quad 0 \leq j \leq l. \tag{24}$$

From Equation (17), we have the following:

$$e_i^j - e_i^{j-1} = \Delta t a(u_i^{j-1}) \delta_h^\alpha e_i^j + \Delta t [a(U_i^{j-1}) - a(u_i^{j-1})] \delta_h^\alpha U_i^j + \Delta t \widehat{R}_{ij},$$

where  $\delta_h^\alpha e_i^j = -h^{(-\alpha)} \sum_{k=i-N}^i g_k^{(\alpha)} e_{i-k}^j$ . Thus we obtain the following:

$$e_i^j - e_i^{j-1} = -\frac{\Delta t}{h^\alpha} a(u_i^{j-1}) \left[ \sum_{k=i-N, k \neq 0}^i g_k^{(\alpha)} e_{i-k}^j + g_0^{(\alpha)} e_i^j \right] + \Delta t [a(U_i^{j-1}) - a(u_i^{j-1})] \delta_h^\alpha U_i^j + \Delta t \widehat{R}_{ij}.$$

So, we rewrite Equation (17) to obtain the following:

$$\left[ 1 + \mu g_0^{(\alpha)} a(u_i^{j-1}) \right] e_i^j = \mu a(u_i^{j-1}) \sum_{k=i-N, k \neq 0}^i g_k^{(\alpha)} e_{i-k}^j + e_i^{j-1} + \Delta t [a(U_i^{j-1}) - a(u_i^{j-1})] \delta_h^\alpha U_i^j + \Delta t \widetilde{R}_{ij}, \quad (25)$$

where  $\mu = \Delta t/h^\alpha$ . If we take infinite norms on both sides of Equation (25), and we set  $i = \ell_i$  obtained so that

$$\left[ 1 + \mu g_0 a(u_i^{j-1}) \right] \|e^j\|_\infty \leq \mu a(u_i^{j-1}) g_0^{(\alpha)} \|e^j\|_\infty + \|e^{j-1}\|_\infty + \Delta t LM_\alpha \|e^{j-1}\|_\infty + c_1 \Delta t (\Delta t + h^2)$$

for  $1 \leq i \leq N-1$ ,  $1 \leq j \leq l+1$ . Thus, we have the following:

$$\|e^j\|_\infty = (1 + \Delta t LM_\alpha) \|e^{j-1}\|_\infty + c_1 \Delta t (\Delta t + h^2), \quad 1 \leq j \leq l+1.$$

By applying the discrete Grönwall inequality (cf. Lemma 2) to  $F^k = e^{k-1}$ ,  $c = LM_\alpha$  and  $g = c_1(\Delta t + h^2)$ , we obtain the inequality as follows:

$$\begin{aligned} \|e^{l+1}\|_\infty &= \exp[LM_\alpha(l+1)\Delta t] \cdot \left[ \|e^0\|_\infty + \frac{c_1(\Delta t + h^2)}{LM_\alpha} \right] \\ &\leq \frac{c_1}{LM_\alpha} \exp(LM_\alpha T) (\Delta t + h^2) \\ &\triangleq C_1 (\Delta t + h^2). \end{aligned}$$

The above proof shows that the semi-implicit difference scheme (15) is stable in the discrete  $L_\infty$ -norm.  $\square$

### 3. Extensions to the Two-Dimensional Problem

In this section, we extend our method to the solution of two-dimensional problems. For this purpose, let  $N_x, N_y$  be positive integers and define a spatial partition of nodes  $h = L/N$ ,  $x_i = ih_x$  ( $i = 1, 2, \dots, N_x$ ),  $y_j = jh_y$  ( $j = 1, 2, \dots, N_y$ ) with step sizes  $h_x = \frac{L_x}{N_x}$ ,  $h_y = \frac{L_y}{N_y}$  in the  $x$ -axis and  $y$ -axis, respectively, and a uniform time partition  $t_k = k\Delta t$  of the interval  $[0, T]$  with time step  $\Delta t = T/M$ . We consider the sets of grid points  $\Omega_h := \{(x_i, y_j) \mid 0 \leq i \leq N_x, 0 \leq j \leq N_y\}$ ,  $\Omega_{\Delta t} := \{t_k \mid 0 \leq k \leq M\}$ ,  $\Omega_{h\Delta t} := \{(x_i, y_j, t_k) \mid 0 \leq i \leq N_x, 0 \leq j \leq N_y, 0 \leq k \leq M\}$ , and let  $u = \{u_{ij}^k \mid 0 \leq i \leq N_x, 0 \leq j \leq N_y, 0 \leq k \leq M\}$  be a grid function defined on  $\Omega_{h\Delta t}$ .

We consider the following two-dimensional problem:

$$\begin{cases} \frac{\partial u(x, y, t)}{\partial t} = a(u) \left[ \frac{\partial^\alpha u(x, y, t)}{\partial |x|^\alpha} + \frac{\partial^\alpha u(x, y, t)}{\partial |y|^\alpha} \right] + f(x, y, t), & (x, y, t) \in \Omega \times (0, T], \\ u(x, y, 0) = \phi(x, y), & (x, y) \in \Omega, \\ u(x, y, t) = 0, & (x, y, t) \in \partial\Omega \times (0, T]. \end{cases} \quad (26)$$

where  $\Omega = [0, L_x] \times [0, L_y]$ . Upon evaluating Equation (26) at the points  $(x_i, y_j, t_k)$ , we obtain the following discrete equation:

$$\frac{\partial u(x_i, y_j, t_k)}{\partial t} = a(u(x_i, y_j, t_k)) \left[ \frac{\partial u(x_i, y_j, t_k)}{\partial |x|^\alpha} + \frac{\partial u(x_i, y_j, t_k)}{\partial |y|^\alpha} \right] + f(x_i, y_j, t_k), \tag{27}$$

$$1 \leq i \leq N_x - 1, 1 \leq j \leq N_y - 1, 1 \leq k \leq M,$$

which can be written equivalently as follows:

$$\frac{\partial u(x_i, y_j, t_k)}{\partial t} - a(u(x_i, y_j, t_{k-1})) \left[ \frac{\partial^\alpha u(x_i, y_j, t_k)}{\partial |x|^\alpha} + \frac{\partial^\alpha u(x_i, y_j, t_k)}{\partial |y|^\alpha} \right] =$$

$$[a(u(x_i, y_j, t_k)) - a(u(x_i, y_j, t_{k-1}))] \left[ \frac{\partial^\alpha u(x_i, y_j, t_k)}{\partial |x|^\alpha} + \frac{\partial^\alpha u(x_i, y_j, t_k)}{\partial |y|^\alpha} \right] + f(x_i, y_j, t_k). \tag{28}$$

Let  $U$  be the grid function defined as follows:

$$U_{i,j}^k := u(x_i, y_j, t_k), \quad f_{i,j}^k := f(x_i, y_j, t_k), \quad 0 \leq i \leq N_x, 0 \leq j \leq N_y, 0 \leq k \leq M.$$

Thus, the partial derivative of  $t$  at the point  $(x_i, y_j, t_k)$  is as follows:

$$\frac{\partial u(x_i, y_j, t_k)}{\partial t} = \frac{u(x_i, y_j, t_k) - u(x_i, y_j, t_{k-1})}{\Delta t} + R_{i,j,k}^{(1)} = \frac{u_{i,j}^k - u_{i,j}^{k-1}}{\Delta t} + \mathcal{O}(\Delta t),$$

and

$$\frac{\partial^\alpha u(x_i, y_j, t_k)}{\partial |x|^\alpha} = -h_x^{-\alpha} \Delta_{h_x}^\alpha u(x_i, y_j, t_k) + R_{i,j,k}^{(2)} \triangleq \delta_{h_x}^\alpha U_i^j + \mathcal{O}(h_x^2),$$

$$\frac{\partial^\alpha u(x_i, y_j, t_k)}{\partial |y|^\alpha} = -h_y^{-\alpha} \Delta_{h_y}^\alpha u(x_i, y_j, t_k) + R_{i,j,k}^{(3)} \triangleq \delta_{h_y}^\alpha U_i^j + \mathcal{O}(h_y^2),$$

where  $\Delta_{h_x}^\alpha U_{i,j}^k = \sum_{\ell=i-N_x}^i g_\ell^{(\alpha)} U_{i-\ell,j}^k$ ,  $\Delta_{h_y}^\alpha U_{i,j}^k = \sum_{\ell=j-N_y}^j g_\ell^{(\alpha)} U_{i,j-\ell}^k$  and  $g_\ell^{(\alpha)} = \frac{(-1)^\ell \Gamma(\alpha+1)}{\Gamma(\alpha/2-\ell+1)\Gamma(\alpha/2+\ell+1)}$ .

We denote for simplicity the following:

$$R_{i,j,k} = -R_{i,j,k}^{(1)} - a(u(x_i, y_j, t_{k-1}))R_{i,j,k}^{(2)} - a(u(x_i, y_j, t_{k-1}))R_{i,j,k}^{(3)} + R', \tag{29}$$

where  $R' = [a(u(x_i, y_j, t_k)) - a(u(x_i, y_j, t_{k-1}))] \left[ \frac{\partial^\alpha u(x_i, y_j, t_k)}{\partial |x|^\alpha} + \frac{\partial^\alpha u(x_i, y_j, t_k)}{\partial |y|^\alpha} \right]$ , so that the equation can be rewritten as follows:

$$\frac{U_{i,j}^k - U_{i,j}^{k-1}}{\Delta t} - a(U_{i,j}^{k-1}) \left[ \delta_{h_x}^\alpha U_{i,j}^k + \delta_{h_y}^\alpha U_{i,j}^k \right] = f_{i,j}^k + R_{i,j,k}, \tag{30}$$

with  $1 \leq i \leq N_x - 1, 1 \leq j \leq N_y - 1, 1 \leq k \leq M$ .

Observe that the terms  $R_{i,j,k}$  are small. Omitting the quantities  $R_{i,j,k}$  in the previous expression, and imposing the initial-boundary value conditions,

$$U_{i,j}^0 = \phi(x_i, y_j), \quad 0 \leq i \leq N_x, 0 \leq j \leq N_y;$$

$$U_{0,j}^k = U_{N_x,j}^k = \varphi(t_k), \quad U_{i,0}^k = U_{i,N_y}^k = \varphi(t_k), \quad 0 \leq k \leq M, \tag{31}$$

the following semi-implicit difference scheme is obtained, where by  $u_{i,j}^k$  we denote the approximate solution of Equation (31) at the points  $(x_i, y_j, t_k)$ :

$$\begin{cases} \frac{u_{i,j}^k - u_{i,j}^{k-1}}{\Delta t} - a(u_{i,j}^{k-1}) [\delta_{h_x}^\alpha u_{i,j}^k + \delta_{h_y}^\alpha u_{i,j}^k] = f_{i,j}^k, & 1 \leq i \leq N_x - 1, 1 \leq j \leq N_y - 1, 1 \leq k \leq M \\ u_{i,j}^0 = \phi(x_i, y_j), & 1 \leq i \leq N_x - 1, 1 \leq j \leq N_y - 1, \\ u_{0,j}^k = u_{N_x,j}^k = \varphi(t_k), \quad u_{i,0}^k = u_{i,N_y}^k = \psi(t_k), & 1 \leq k \leq M. \end{cases} \tag{32}$$

Below, we analyze stability and convergence properties of the two-dimensional semi-implicit difference scheme given by Equation (32). We define the following:

$$e_{i,j}^k = U_{i,j}^k - u_{i,j}^k, \quad i = 0, \dots, N_x, j = 1, \dots, N_y, k = 0, \dots, M,$$

the error satisfying the following equations:

$$\begin{cases} \frac{e_{i,j}^k - e_{i,j}^{k-1}}{\Delta t} - a(u_{i,j}^{k-1}) [\delta_{h_x}^\alpha e_{i,j}^k + \delta_{h_y}^\alpha e_{i,j}^k] = [a(U_{i,j}^{k-1}) - a(u_{i,j}^{k-1})] [\delta_{h_x}^\alpha U_{i,j}^k + \delta_{h_y}^\alpha U_{i,j}^k] + R_{i,j,k}, & 1 \leq i \leq N_x - 1, 1 \leq j \leq N_y - 1, 1 \leq k \leq M, \\ e_{i,j}^0 = 0, & 0 \leq i \leq N_x, 0 \leq j \leq N_y, \\ e_{0,j}^k = e_{N_x,j}^k = 0, \quad e_{i,0}^k = e_{i,N_y}^k = 0, & 1 \leq k \leq M. \end{cases} \tag{33}$$

By a similar proof to Theorem 1, the following result is established about the numerical stability of the semi-implicit difference scheme given by Equation (26).

**Theorem 2.** Suppose that  $\{u_{i,j}^k | 0 \leq i \leq N_x, 0 \leq j \leq N_y, 0 \leq k \leq M\}$  is the solution of the semi-implicit difference scheme (25), and define the following:

$$C_2 = \frac{c_2}{LM_\alpha} \exp(LM_\alpha T), \quad M_\alpha = \left| \max_{\substack{0 \leq x, y \leq L, \\ 1 \leq t \leq T}} \left[ \frac{\partial^\alpha u(x, t)}{\partial |x|^\alpha} + \frac{\partial^\alpha u(x, y, t)}{\partial |y|^\alpha} \right] \right|. \tag{34}$$

Then, when constants  $\Delta t, h$  ( $= h_x = h_y$ ) satisfy the following condition,

$$\Delta t \leq \frac{\epsilon}{2C_2}, \quad h \leq \sqrt{\frac{\epsilon}{2C_2}}, \tag{35}$$

the semi-implicit difference scheme (32) is stable and as follows:

$$\|e^k\|_\infty \leq C_2(\Delta t + h^2), \quad 0 \leq k \leq M. \tag{36}$$

#### 4. Fast Implementation of the Semi-Implicit Difference Scheme

In this section, we present a fast matrix-free implementation of the proposed one-dimensional and two-dimensional semi-implicit difference schemes that use the preconditioned biconjugate gradients stabilized (BiCGSTAB) method [38] to iteratively solve the nonsymmetric linear system with a Toeplitz-like structure arising from the discretization. Such an implementation enables us to reduce both the algorithmic complexity and the memory requirements for the solution significantly.

First, we analyze the properties of the coefficient matrices of the discretized linear systems.

##### 4.1. One-Dimensional Coefficient Matrix

By rewriting Equation (15) in the following form,

$$\frac{w^j - w^{j-1}}{\Delta t} = \frac{1}{h^\alpha} D^{(j-1)} A w^j + \frac{1}{h^\alpha} D^{(j-1)} v^{j-1} + f^j, \tag{37}$$

where  $v^{j-1} = [g_1^{(\alpha)} u_0^{j-1} + g_{-(N-1)}^{(\alpha)} u_N^j, g_2^{(\alpha)} u_0^{j-1} + g_{-(N-2)}^{(\alpha)} u_N^j, \dots, g_{N-1}^{(\alpha)} u_0^{j-1} + g_{-1}^{(\alpha)} u_N^j]^\top$ , we derive the following matrix form of the semi-implicit difference scheme:

$$\left(I - \frac{\Delta t}{h^\alpha} D^{(j-1)} A\right) u^j = u^{j-1} + \Delta t F^j, \quad (38)$$

where  $I$  is an identity matrix of suitable order,  $F^j = \frac{1}{h^\alpha} D^{(j-1)} v^{j-1} + f^j$ ,  $u^j = [u_1^j, u_2^j, \dots, u_N^j]^\top$ ,

$$D^{(j-1)} = \begin{pmatrix} a(u_1^{j-1}) & & & \\ & a(u_2^{j-1}) & & \\ & & \ddots & \\ & & & a(u_{N-1}^{j-1}) \end{pmatrix} \in \mathbb{R}^{(N-1) \times (N-1)},$$

is a diagonal matrix and

$$A = \begin{pmatrix} g_0^{(\alpha)} & g_{-1}^{(\alpha)} & \cdots & g_{-(N-2)}^{(\alpha)} \\ g_1^{(\alpha)} & g_0^{(\alpha)} & \cdots & g_{-(N-3)}^{(\alpha)} \\ \vdots & \vdots & \ddots & \vdots \\ g_{N-2}^{(\alpha)} & g_{N-3}^{(\alpha)} & \cdots & g_0^{(\alpha)} \end{pmatrix}$$

is a symmetric negative definite Toeplitz matrix [3,35], which can be represented by a vector containing the elements of the first column, i.e., its storage requirement is of  $\mathcal{O}(N)$ .

#### 4.2. Two-Dimensional Coefficient Matrix

The structure of the two-dimensional coefficient matrix is more complex than in the one-dimensional case. By writing Equation (26) in the following equivalent form

$$\frac{u^k - u^{k-1}}{\Delta t} = \tilde{D}^{(k-1)} [W_x^{(\alpha)} \otimes I_y + I_x \otimes W_y^{(\alpha)}] u^k + f_{i,j}^k, \quad (39)$$

in which  $u^k = [u_{11}^k, \dots, u_{N_x-1,1}^k, u_{1,2}^k, \dots, u_{N_x-1,2}^k, \dots, u_{1,N_y-1}^k, \dots, u_{N_x-1,N_y-1}^k]^\top$ , we can derive the discretized linear system as follows:

$$\left(I - \frac{\Delta t}{h^\alpha} \tilde{D}^{(k-1)} \tilde{A}\right) u^k = u^{k-1} + \Delta t f^k, \quad (40)$$

where  $\tilde{A} = W_x^{(\alpha)} \otimes I_y + W_y^{(\alpha)} \otimes I_x$ ,  $I_q$  ( $q \in \{x, y\}$ ) is also an identity matrix of suitable order and  $f^k = [f_{11}^k, \dots, f_{N_x-1,1}^k, f_{12}^k, \dots, f_{N_x-1,2}^k, \dots, f_{1,N_y-1}^k, \dots, f_{N_x-1,N_y-1}^k]^\top$  and

$$\tilde{D}^{(k-1)} = \begin{pmatrix} a(u_{11}^{k-1}) & & & & \\ & \ddots & & & \\ & & a(u_{N_x-1,1}^{k-1}) & & \\ & & & \ddots & \\ & & & & a(u_{1,N_y-1}^{k-1}) \\ & & & & & \ddots \\ & & & & & & a(u_{N_x-1,N_y-1}^{k-1}) \end{pmatrix}.$$

is a diagonal matrix. In order to simplify the calculation, we choose  $h = h_x = h_y$  (i.e.,  $N_x = N_y = N$ ) and thus  $W_x^{(\alpha)} = W_y^{(\alpha)} = W^{(\alpha)} = A$ .

According to Lemma 2,  $W^{(\alpha)}$  is a symmetric Toeplitz matrix of order  $(N - 1)$ . We convert  $W^{(\alpha)}$  into a  $\tau$ -matrix defined as follows:

$$\tau(W^{(\alpha)}) = W^{(\alpha)} - H_{N-1},$$

where  $H_{N-1}$  is a Hankel matrix with constant entries along antidiagonals, namely

$$\left[ g_2^{(\alpha)}, g_3^{(\alpha)}, \dots, g_{N-2}^{(\alpha)}, 0, 0, 0, g_{N-2}^{(\alpha)}, \dots, g_3^{(\alpha)}, g_2^{(\alpha)} \right].$$

Notice that the  $\tau$ -matrix can be diagonalized by the sine transform matrix  $S_{N-1}$  [39] as follows:

$$\tau(W^{(\alpha)}) = S_{N-1} \Lambda S_{N-1}, \tag{41}$$

where  $\Lambda$  is the diagonal matrix consisting of the eigenvalues of  $\tau(W^{(\alpha)})$ . The entries of  $S_{N-1}$  are given by the following:

$$[S_{N-1}]_{i,j} = \sqrt{\frac{2}{N}} \sin\left(\frac{\pi ij}{N}\right), \quad 1 \leq i, j \leq N - 1.$$

A simple calculation shows that  $S_{N-1}$  is a symmetric orthogonal matrix, that is,

$$S_{N-1}^{-1} = S_{N-1}^\top = S_{N-1}.$$

Upon replacing  $W^{(\alpha)}$  with  $\tau(W^{(\alpha)})$  in the expression of  $\tilde{A}$ , and using the properties of the Kronecker product, we derive the following:

$$\begin{aligned} P &= \tau(W^{(\alpha)}) \otimes I + I \otimes \tau(W^{(\alpha)}) \\ &= (S_{N-1} \otimes I)[(\Lambda \otimes I) + (I \otimes \Lambda)](S_{N-1} \otimes I). \end{aligned} \tag{42}$$

Finally, we introduce matrices  $\mathcal{A}$  and  $\mathcal{M}$  defined as follows:

$$\mathcal{A} = I - \Delta t \tilde{D}^{(k-1)} \tilde{A} \quad \text{and} \quad P_\tau = I - \Delta t d^{(k-1)} P, \tag{43}$$

where constant  $d^{(k-1)}$  is defined as follows:

$$d^{(k-1)} := \frac{a(u_{11}^{k-1}) + \dots + a(u_{1,N-1}^{k-1}) + \dots + a(u_{N-1,1}^{k-1}) + \dots + a(u_{N-1,N-1}^{k-1})}{(N - 1)^2} > 0, \tag{44}$$

due to the condition (2).

Under these assumptions, we can prove the following property of the preconditioner  $P_\tau$  that guarantees the invertibility of  $P_\tau$ :

**Proposition 1.** *The preconditioner  $P_\tau$  defined in (43) is symmetric positive definite.*

**Proof.** From Equation (41) and Lemma 3.2 in [30], we know that all the eigenvalues of  $\tau(W^{(\alpha)})$  are negative as are those of matrix  $W^{(\alpha)}$ ; that is, the matrix  $P$  defined in (42) is negative definite. Recalling the definition of the preconditioner  $P_\tau$  given by Equation (43), and that  $\Delta t, d^{(k-1)} > 0$ , we can conclude that the preconditioner  $P_\tau$  is symmetric positive definite.  $\square$

Henceforth, for clarity, we rewrite Equation (40) as the following linear system:

$$\mathcal{A}u = b, \tag{45}$$

where the right-hand side vector is  $b = u^{k-1} + \Delta t f^k$ . Since the coefficient matrix  $\mathcal{A}$  is non-symmetric, we can solve it by the preconditioned BiCGSTAB method described below: (Algorithm 1).

**Algorithm 1** The preconditioned BiCGSTAB method.

- |   |  |
|---|--|
| 1: $x_0$ is an initial guess; $r_0 = b - \mathcal{A}u_0$ ;                                    | 10: $\alpha = \rho_i / (\bar{r}_0, v_i)$ ;                                     |
| 2: $\bar{r}_0$ is an arbitrary vector, such that  | 11: $s = r_{i-1} - \alpha v_i$ ;   |
| 3: $(\bar{r}_0, r_0) \neq 0$ , e.g., $\bar{r}_0 = r_0$ ; $\rho_0 = \alpha =$                  | 12: Solve $z$ from $P_\tau z = s$ ;  |
| $\omega_0 = 1$ ;  | 13: $t = \mathcal{A}z$ ;   |
| 4: $v_0 = p_0 = 0$ ;  | 14: $\omega_i = (P_\tau^{-1}t, P_\tau^{-1}s) / (P_\tau^{-1}t, P_\tau^{-1}t)$ ; |
| 5: <b>for</b> $i = 1, 2, \dots$ , <b>do</b>   | 15: $u_i = u_{i-1} + \alpha y + \omega_i z$ ;                                  |
| 6: $\rho_i = (\bar{r}_0, r_{i-1})$ ; $\beta = (\rho_i / \rho_{i-1})(\alpha / \omega_{i-1})$ ; | 16: <b>if</b> $u_i$ is accurate enough then quit;                              |
| 7: $p_i = r_{i-1} + \beta(p_{i-1} - \omega_{i-1}v_{i-1})$ ;                                   | 17: $r_i = s - \omega_i t$ ;   |
| 8: Solve $y$ from $P_\tau y = p_i$ ;  | 18: <b>end for</b>   |
| 9: $v_i = \mathcal{A}y$   |  |

At each iteration of the preconditioned BiCGSTAB algorithm, two matrix-vector products need to be computed, i.e.,  $\mathcal{A}v$  and  $P_\tau^{-1}v$ , where  $v$  represents an arbitrary vector. The matrix-vector product

$$\mathcal{A}v = (I - \Delta t \tilde{D}^{(k-1)} \tilde{A})v = v - \Delta t \tilde{D}^{(k-1)}(\tilde{A}v), \tag{46}$$

is calculated as follows:

$$\begin{aligned} \tilde{A}v &= (W^{(\alpha)} \otimes I + I \otimes W^{(\alpha)})v \\ &= (W^{(\alpha)} \otimes I)v + (I \otimes W^{(\alpha)})v \\ &= W^{(\alpha)} \cdot V + V \cdot W^{(\alpha)}, \end{aligned} \tag{47}$$

where  $V = [v_1, v_2, \dots, v_{N-1}]$ ,  $v_i \in \mathbb{R}^{N-1}$  and  $v = \text{vec}(V)$ . Obviously, this problem is transformed into a matrix-vector operation with matrix  $W^{(\alpha)}$  and vector  $v$ . On the one hand, the calculation  $W^{(\alpha)} \cdot V$  is expressed as  $W^{(\alpha)} \cdot [v_1, v_2, \dots, v_{N-1}]$  one by one and costs  $\mathcal{O}(n \log n)$  arithmetic operations, where  $n = (N - 1)^2$ . On the other hand, the operation  $V \cdot W^{(\alpha)}$  is computed as  $W^{(\alpha)} \cdot V^\top$ , and thus, it can be carried out at the same with  $\mathcal{O}(n \log n)$  complexity. Similarly, since matrix  $P$  can be diagonalized, the preconditioner  $P_\tau^{-1}$  can be diagonalized too, so the inverse of  $P_\tau$  exists. The preconditioning operation  $P_\tau^{-1}v$  is as follows:

$$\begin{aligned} P_\tau^{-1}v &= [I - \Delta t d^{(k-1)} P]^{-1}v \\ &= (S_{N-1} \otimes I) [I - \Delta t d^{(k-1)} (\Lambda \otimes I + I \otimes \Lambda)]^{-1} (S_{N-1} \otimes I)v. \end{aligned}$$

In conclusion, each iteration of the preconditioned BiCGSTAB iterative algorithm to solve Equation (45) requires  $\mathcal{O}(n \log n)$  arithmetic operations in total and can be implemented as a matrix-free manner.

It is worth noting that, although the preconditioner  $P_\tau^{-1}$  is well defined according to Proposition 1, due to the presence of nonlinear diffusion coefficients and the nonsymmetry of the discretized linear systems (40), it is not simple to derive theoretically the eigenvalue distribution of the preconditioned matrix [33]. However, in the next section, we provide numerical evidence of the favorable clustering of the eigenvalues of the preconditioned matrices for some problems.

**5. Numerical Experiments**

In this section, two numerical experiments are presented to illustrate the efficiency of the proposed preconditioned semi-implicit difference scheme, and to support our theoretical findings. All experiments are performed in MATLAB R2019b on Intel(R) Core(TM)

i5-10210U CPU @ 1.60~2.11 GHz and 8.00 GB of RAM. On occasion, one can reduce non-homogeneous Dirichlet boundary conditions to homogeneous Dirichlet boundary conditions by using the following transformation:

$$u(x, t) = W(x, t) + V(x, t), \tag{48}$$

where  $V(x, t) = \phi(t) + \frac{[\Phi(t) - \phi(t)]x}{L_x}$  and  $W(x, t)$  is an unknown function such that  $W(0, t) = 0, W(L_x, t) = 0$ .

**Example 1.** For the first problem, we consider Equation (1) with  $L_x = T = 1, a(u) = u^2, \phi(x) = x^2(1 - x)^2, \varphi(t) = \psi(t) = 0$ , and the source term  $f(x, t)$  is computed as follows:

$$f(x, t) = \alpha(1 + t)^{\alpha-1}x^2(1 - x)^2 + \frac{a(u)(1 + t)^\alpha}{2 \cos(\pi\alpha/2)} \left\{ \frac{\Gamma(3)}{\Gamma(3 - \alpha)} [x^{2-\alpha} + (1 - x)^{2-\alpha}] - \frac{2\Gamma(4)}{\Gamma(4 - \alpha)} [x^{3-\alpha} + (1 - x)^{3-\alpha}] + \frac{\Gamma(5)}{\Gamma(5 - \alpha)} [x^{4-\alpha} + (1 - x)^{4-\alpha}] \right\}.$$

This problem can be viewed as a modification of [1,24], and its exact solution is  $u(x, t) = (1 + t)^\alpha x^2(1 - x)^2$ . Denoting by  $u_j^n$  the numerical solution, we define the  $L_\infty$ -norm errors and the  $L_2$ -norm errors as follows:

$$E_\infty(\Delta t, h) = \max_{1 \leq j \leq M} \|u(ih, j\Delta t) - u_i^j\|_\infty, \quad E_2(\Delta t, h) = \max_{1 \leq j \leq M} \|u(ih, j\Delta t) - u_i^j\|,$$

respectively, and the space and time convergence orders as follows:

$$\begin{aligned} rate_1^t &= \log_2 \left( \frac{E_\infty(2\Delta t, h)}{E_\infty(\Delta t, h)} \right), & rate_1^x &= \log_2 \left( \frac{E_\infty(4h^2, 2h)}{E_\infty(h^2, h)} \right), \\ rate_2^t &= \log_2 \left( \frac{E_2(2\Delta t, h)}{E_2(\Delta t, h)} \right), & rate_2^x &= \log_2 \left( \frac{E_2(4h^2, 2h)}{E_2(h^2, h)} \right). \end{aligned}$$

The data reported in Tables 1 and 2 refer to the convergence results obtained by the fast preconditioned iterative algorithm and by a direct method (i.e., the backslash “\” operator in MATLAB) for solving the non-symmetric discretized linear system, respectively. Because of the small size of the linear systems presented in Table 2, the cost of direct solvers is affordable [16]. Table 1 presents the maximum errors and the corresponding convergence orders in time of the semi-implicit difference scheme when  $N = 512$ , for  $\alpha = 1.20, 1.50, 1.90$ , respectively. The numerical results show a convergence order approximately equal to 1 in time for the semi-implicit difference scheme. Meanwhile, the experiments reported in Table 2 confirm that the semi-implicit difference scheme is convergent with second-order accuracy in space. These results are consistent with the theoretical analysis presented in Section 2.

**Table 1.** The errors and temporal convergence orders when  $N = 512$ .

$\alpha$	$M$	$E_\infty(\Delta t, h)$	$rate_1^t$	$E_2(\Delta t, h)$	$rate_2^t$
1.20	8	1.6125e − 03	–	8.6746e − 04	–
	16	7.9547e − 04	1.0194	4.2894e − 04	1.0160
	32	3.9507e − 04	1.0097	2.1329e − 04	1.0080
	64	1.9688e − 04	1.0048	1.0635e − 04	1.0040
	128	9.8288e − 05	1.0022	5.3109e − 05	1.0018

**Table 1.** Cont.

$\alpha$	$M$	$E_\infty(\Delta t, h)$	$rate_1^t$	$E_2(\Delta t, h)$	$rate_2^t$
1.50	8	4.7362e − 03	–	2.6365e − 03	–
	16	2.3256e − 03	1.0261	1.2984e − 03	1.0219
	32	1.1525e − 03	1.0128	6.4439e − 04	1.0107
	64	5.7374e − 04	1.0063	3.2102e − 04	1.0053
	128	2.8628e − 04	1.0030	1.6023e − 04	1.0025
1.90	8	1.2993e − 02	–	7.4538e − 03	–
	16	6.3112e − 03	1.0417	3.6319e − 03	1.0373
	32	3.1120e − 03	1.0201	1.7935e − 03	1.0179
	64	1.5455e − 03	1.0098	8.9131e − 04	1.0088
	128	7.7025e − 04	1.0047	4.4437e − 04	1.0042

**Table 2.** The errors and spatial convergence orders when  $M = N^2$ .

$\alpha$	$N$	$E_\infty(\Delta t, h)$	$rate_1^x$	$E_2(\Delta t, h)$	$rate_2^x$
1.20	4	1.1988e − 03	–	6.3875e − 04	–
	8	2.9694e − 04	2.0133	1.5674e − 04	2.0269
	16	7.4054e − 05	2.0035	3.9116e − 05	2.0025
	32	1.8502e − 05	2.0009	9.7746e − 06	2.0006
	64	4.6248e − 06	2.0002	2.4434e − 06	2.0001
1.50	4	3.5560e − 03	–	1.9379e − 03	–
	8	8.7005e − 04	2.0489	4.7737e − 04	2.0213
	16	2.1633e − 04	2.0120	1.1889e − 04	2.0055
	32	5.4010e − 05	2.0030	2.9695e − 05	2.0013
	64	1.3498e − 05	2.0007	7.4219e − 06	2.0004
1.90	4	1.0923e − 02	–	6.1965e − 03	–
	8	2.6106e − 03	2.0649	1.5174e − 03	1.9285
	16	6.4524e − 04	2.0165	3.7556e − 04	1.9873
	32	1.6085e − 04	2.0041	9.3655e − 05	1.9954
	64	4.0184e − 05	2.0010	2.3399e − 05	1.9992

**Example 2.** For the second problem, we consider the two-dimensional nonlinear Riesz space-fractional diffusion problem, which is modified from [34] as follows:

$$\begin{cases} \frac{\partial u(x, y, t)}{\partial t} = a(u) \left[ \frac{\partial u(x, y, t)^\alpha}{\partial |x|^\alpha} + \frac{\partial u(x, y, t)^\alpha}{\partial |y|^\alpha} \right] + f(x, y, t), & (x, y) \in \Omega, 0 < t \leq 1 \\ u(0, y, t) = u(1, y, t) = u(x, 0, t) = u(x, 1, t) = 0, & 0 \leq t \leq 1 \\ u(x, y, 0) = x^3(1-x)^3y^3(1-y)^3, & (x, y) \in \Omega, \end{cases} \quad (49)$$

where  $\Omega = (0, 1) \times (0, 1)$ ,  $a(u) = |u| + 10$ , and

$$\begin{aligned} f(x, y, t) = & -u(x, y, t) + \frac{a(u)}{2 \cos(\frac{\alpha\pi}{2})} e^{-t} \left\{ y^3(1-y)^3 [g_1(x) - 3g_2(x) + 3g_3(x) - g_4(x)] \right. \\ & \left. + x^3(1-x)^3 [g_1(y) - 3g_2(y) + 3g_3(y) - g_4(y)] \right\} \end{aligned} \quad (50)$$

where

$$g_i(x) = \frac{\Gamma(i+3)}{\Gamma(i+3-\alpha)} \left[ x^{i+2-\alpha} + (1-x)^{i+2-\alpha} \right], \quad i = 1, 2, 3, 4. \quad (51)$$

The exact solution for this problem is  $u(x, y, t) = e^{-t} x^3(1-x)^3 y^3(1-y)^3$ .

The data reported in Tables 3 and 4 are convergence results obtained with the preconditioned iterative algorithm. Table 3 presents the maximum errors and the corresponding convergence orders in time of the semi-implicit difference scheme when  $N = 32$ , for  $\alpha = 1.20, 1.50, 1.90$ , respectively. The numerical results exhibit a convergence order approximately equal to 1 in time for the semi-implicit scheme. Meanwhile, Table 4 shows that the semi-implicit scheme is convergent with second-order accuracy in space. These results are consistent with the theoretical analysis presented in Section 3.

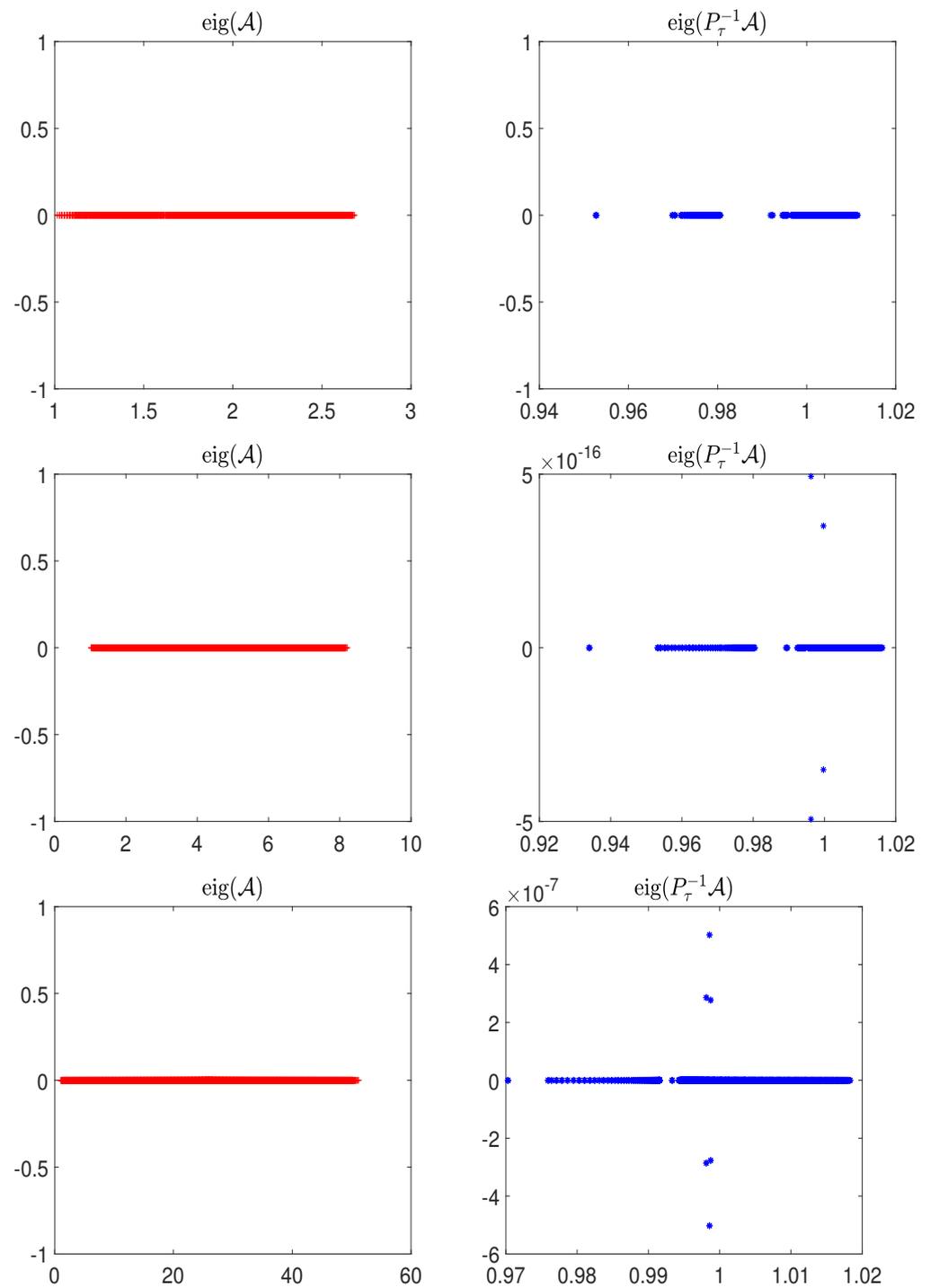
**Table 3.** The errors and temporal convergence orders when  $N = 32$ .

$\alpha$	$M$	$E_{\infty}(\Delta t, h)$	$rate_1^t$	$E_2(\Delta t, h)$	$rate_2^t$
1.20	4	1.8489e − 05	–	6.3044e − 06	–
	8	9.4453e − 06	0.9690	3.2208e − 06	0.9689
	16	4.7734e − 06	0.9846	1.6277e − 06	0.9846
	32	2.3998e − 06	0.9921	8.1828e − 07	0.9922
1.50	4	1.8488e − 05	–	6.3044e − 06	–
	8	9.4452e − 06	0.9689	3.2207e − 06	0.9690
	16	4.7732e − 06	0.9846	1.6276e − 06	0.9846
	32	2.3996e − 06	0.9922	8.1822e − 07	0.9922
1.90	4	1.8489e − 05	–	6.3045e − 06	–
	8	9.4456e − 06	0.9690	3.2208e − 06	0.9690
	16	4.7737e − 06	0.9845	1.6278e − 06	0.9845
	32	2.4001e − 06	0.9920	8.1838e − 07	0.9921

**Table 4.** The errors and spatial convergence orders when  $M = N^2$ .

$\alpha$	$N$	$E_{\infty}(\Delta t, h)$	$rate_1^x$	$E_2(\Delta t, h)$	$rate_2^x$
1.20	4	4.7734e − 06	–	1.6181e − 06	–
	8	1.2036e − 06	1.9877	4.1036e − 07	1.9793
	16	3.0226e − 07	1.9943	1.0299e − 07	1.9943
	32	7.6388e − 08	1.9844	2.6021e − 08	1.9855
1.50	4	4.7732e − 06	–	1.6180e − 06	–
	8	1.2034e − 06	1.9878	4.1030e − 07	1.9795
	16	3.0205e − 07	1.9943	1.0298e − 07	1.9943
	32	7.6178e − 08	1.9873	2.5954e − 08	1.9883
1.90	4	4.7738e − 06	–	1.6182e − 06	–
	8	1.2039e − 06	1.9874	4.1046e − 07	1.9791
	16	3.0264e − 07	1.9920	1.0314e − 07	1.9926
	32	7.6769e − 08	1.9790	2.6120e − 08	1.9814

Table 5 shows comparative experiments between a direct method and two fast iterative algorithms for the solution of the discretized linear system at each step of the implicit difference scheme for  $\alpha = 1.20, 1.60, 1.70, 1.90$ . It presents the total number of iterations (Iter) and the solution time (CPU in seconds) when  $M = N^2$ . Obviously, we can see from the table that the two iterative algorithms are significantly faster than a direct algorithm, motivating our quest for fast solvers. As seen from Figure 1, the proposed preconditioner  $P_{\tau}$  can be indeed efficient to accelerate the convergence of the preconditioned BiCGSTAB method in terms of the clustering eigenvalues, i.e., most eigenvalues of the preconditioned matrices increasingly gather around 1 when we take  $\alpha \rightarrow 2$ .



**Figure 1.** The eigenvalue distributions of both original and preconditioned matrices with  $N = 2^6$ ,  $M = N^2$ , first line:  $\alpha = 1.2$ ; second line:  $\alpha = 1.5$ ; third line:  $\alpha = 1.9$ ).

**Table 5.** Total number of iterations and CPU time (in seconds) when  $M = N^2$ .

$\alpha$	N	Direct	I		$P_\tau$	
		CPU	Iter	CPU	Iter	CPU
1.20	16	0.026	15.0	0.041	4.0	0.021
	32	1.635	24.0	0.350	4.0	0.094
	64	130.982	37.0	2.354	5.0	0.532
	128	23,599.059	50.4	41.207	5.0	5.248
	256	out of memory	59.9	844.715	5.0	38.040
1.60	16	0.024	18.0	0.044	4.0	0.023
	32	1.614	33.0	0.479	4.0	0.094
	64	117.905	59.4	3.764	4.0	0.430
	128	24,804.137	95.1	78.685	4.8	4.962
	256	out of memory	154.3	706.824	5.0	38.935
1.70	16	0.020	19.0	0.050	3.0	0.016
	32	1.562	36.0	0.522	4.0	0.096
	64	751.896	64.0	4.065	4.0	0.428
	128	32,105.020	109.4	89.902	4.0	4.185
	256	out of memory	188.2	858.853	4.0	30.393
1.90	16	0.021	20.0	0.051	3.0	0.017
	32	1.689	43.7	0.636	3.0	0.071
	64	263.742	75.3	4.785	4.0	0.426
	128	22,645.044	144.7	387.991	4.0	4.211
	256	out of memory	273.0	5633.458	4.0	30.324

Since it is proved that the convergence order of the proposed semi-implicit different scheme is  $\mathcal{O}(\Delta t + h^2)$ , the Richardson extrapolation can be used to improve the temporal convergence order. For the sake of clarity, we do the following:

- (1) We apply the numerical scheme (24) with time step  $\Delta t$ , and we compute the solution  $U_{ij}^n(h_x, h_y, \Delta t)$ . According to the error analysis of the method, the following holds:

$$u(x_i, y_j, t_n) - U_{ij}^n(h_x, h_y, \Delta t) = \mathcal{O}(h_x^2 + h_y^2) + c\Delta t + \mathcal{O}(\Delta t^2), \quad (52)$$

- (2) Another run with the scheme (24) is carried out using  $\frac{\Delta t}{2}$ , and a new solution  $U_{ij}^{2n}(h_x, h_y, \frac{\Delta t}{2})$  is computed on the fine temporal grid. According to the error analysis, we have the following:

$$u(x_i, y_j, t_n) - U_{ij}^{2n}(h_x, h_y, \frac{\Delta t}{2}) = \mathcal{O}(h_x^2 + h_y^2) + c\frac{\Delta t}{2} + \mathcal{O}\left(\left(\frac{\Delta t}{2}\right)^2\right), \quad (53)$$

- (3) We compute  $(5.6) \times 2 - (5.5)$  and define  $\tilde{U}_{ij}^n = 2U_{ij}^{2n}(h_x, h_y, \frac{\Delta t}{2}) - U_{ij}^n(h_x, h_y, \frac{\Delta t}{2})$ :

$$u(x_i, y_j, t_n) - \tilde{U}_{ij}^n = \mathcal{O}(h_x^2 + h_y^2 + \Delta t^2). \quad (54)$$

This means that Richardson extrapolation indeed can improve the temporal convergence order of the proposed semi-implicit difference scheme from 1 to 2. However, it is always sensitive to the parameter selection used for solving the model problem and thus, it is numerically unstable (we omit the specific numerical results here).

## 6. Concluding Remarks

In this paper, a fast preconditioned semi-implicit difference scheme for a one-dimensional strongly NSFDE is first presented, which is also extended to solve a two-dimensional NSFDE. In addition, the stability and convergence properties of the new numerical scheme are proved. At the same time, a fast preconditioned iterative algorithm is designed to solve the discretized linear systems on each time step efficiently and rapidly without necessarily storing the coefficient matrix of the linear system at each time level. In the future work, we will design the fast numerical methods and explore their stability and convergence analyses for solving a class of strongly NSFDEs. Moreover, due to the nonlocality of the Riesz fractional derivative, developing the fast matrix-free iterative methods accelerated by suitable preconditioners (e.g., the recent parallel-in-time preconditioners [36,40]) for solving the discretized linear systems should be also very meaningful.

**Author Contributions:** Funding acquisition, X.-M.G. and H.L.; investigation, Y.-Y.H. and Y.G.; methodology, X.-M.G., Y.G. and B.C.; project administration, X.-M.G., H.L. and B.C.; software, Y.-L.Z.; supervision, X.-M.G. and Y.-L.Z.; visualization, Y.-Y.H.; writing—original draft, Y.-Y.H.; writing—review and editing, Y.-Y.H., X.-M.G., Y.-L.Z., H.L., Y.G. and B.C. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was supported by NSFC (Nos. 11801463 and 12101089) and the Applied Basic Research Program of Sichuan Province (No. 2020YJ0007). The last author is a member of the *Gruppo Nazionale per il Calcolo Scientifico* (GNCS) of the Istituto Nazionale di Alta Matematica (INdAM) and this work was partially supported by INdAM-GNCS under Progetti di Ricerca 2020.

**Data Availability Statement:** The MATLAB codes for numerical experiments will be available by corresponding authors' emails.

**Acknowledgments:** The authors would like to thank anonymous referees for their many valuable comments and suggestions for improving the presentation of this work.

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

## References

- Moroney, T.; Yang, Q. A banded preconditioner for the two-sided, nonlinear space-fractional diffusion equation. *Comput. Math. Appl.* **2013**, *66*, 659–667. [[CrossRef](#)]
- Moroney, T.; Yang, Q. Efficient solution of two-sided nonlinear space-fractional diffusion equations using fast Poisson preconditioners. *J. Comput. Phys.* **2013**, *246*, 304–317. [[CrossRef](#)]
- Yang, Q.; Liu, F.; Turner, I. Numerical methods for fractional partial differential equations with Riesz space fractional derivatives. *Appl. Math. Model.* **2010**, *34*, 200–218. [[CrossRef](#)]
- Ding, H. High-order numerical algorithms for Riesz derivatives via constructing new generating functions. *J. Sci. Comput.* **2017**, *71*, 759–784. [[CrossRef](#)]
- Cai, M.; Li, C. On Riesz derivative. *Fract. Calc. Appl. Anal.* **2019**, *22*, 287–301. [[CrossRef](#)]
- Metzler, R.; Klafter, J. The random walk's guide to anomalous diffusion: A fractional dynamics approach. *Phys. Rep.* **2000**, *399*, 1–77. [[CrossRef](#)]
- 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. [[CrossRef](#)]
- Schneider, W.R.; Wyss, W. Fractional diffusion and wave equations. *J. Math. Phys.* **1989**, *30*, 134–144. [[CrossRef](#)]
- Wyss, W. The fractional diffusion equation. *J. Math. Phys.* **1986**, *27*, 2782–2785. [[CrossRef](#)]
- Kilbas, A.A.; Srivastava, H.M.; Trujillo, J.J. *Theory and Applications of Fractional Differential Equations*; Elsevier: Amsterdam, The Netherlands, 2006.
- Ail, U.; Sohail, M.; Abdullah, F.A. An efficient numerical scheme for variable-order fractional sub-diffusion equation. *Symmetry* **2020**, *13*, 1437.
- Ilic, M.; Liu, F.; Turner, I.; Anh, V. Numerical approximation of a fractional-in-space diffusion equation, I. *Fract. Calc. Appl. Anal.* **2005**, *8*, 323–341. [[CrossRef](#)]
- Ilic, M.; Liu, F.; Turner, I.; Anh, V. Numerical approximation of a fractional-in-space diffusion equation (II)—with nonhomogeneous boundary conditions. *Fract. Calc. Appl. Anal.* **2006**, *9*, 333–349.
- Sousa, E.; Li, C. A weighted finite difference method for the fractional diffusion equation based on the Riemann-Liouville derivative. *Appl. Numer. Math.* **2015**, *90*, 22–37.
- Diethelm, K. An algorithm for the numerical solution of differential equations of fractional order. *Electron. Trans. Numer. Anal.* **1997**, *5*, 1–6. [[CrossRef](#)]

16. Li, M.; Gu, X.-M.; Huang, C.; Fei, M.; Zhang, G. A fast linearized conservative finite element method for the strongly coupled nonlinear fractional Schrödinger equations. *J. Comput. Phys.* **2018**, *358*, 256–282.
17. Zhao, X.; Hu, X.; Cai, W.; Karniadakis, G.E. Adaptive finite element method for fractional differential equations using hierarchical matrices. *Comput. Methods Appl. Mech. Eng.* **2017**, *325*, 56–76. [[CrossRef](#)]
18. Fu, H.; Liu, H.; Zheng, X. A preconditioned fast finite volume method for distributed-order diffusion equation and applications. *East. Asian J. Appl. Math.* **2019**, *9*, 28–44. [[CrossRef](#)]
19. Simmons, A.; Yang, Q.; Moroney, T. A finite volume method for two-sided fractional diffusion equations on non-uniform meshes. *J. Comput. Phys.* **2017**, *335*, 747–759. [[CrossRef](#)]
20. Zeng, F.; Liu, F.; Li, C.; Burrage, K.; Turner, I.; Anh, V. A Crank-Nicolson ADI spectral method for a two-dimensional Riesz space fractional nonlinear reaction-diffusion equation. *SIAM J. Numer. Anal.* **2014**, *52*, 2599–2622. [[CrossRef](#)]
21. Tian, W.Y.; Zhou, H.; Deng, W. A class of second order difference approximations for solving space fractional diffusion equations. *Math. Comp.* **2015**, *84*, 1703–1727. [[CrossRef](#)]
22. Li, C.; Deng, W. A new family of difference schemes for space fractional advection diffusion equation. *Adv. Appl. Math. Mech.* **2017**, *9*, 282–306. [[CrossRef](#)]
23. Hao, Z.-P.; Sun, Z.-Z.; Cao, W.-R. A fourth-order approximation of fractional derivatives with its applications. *J. Comput. Phys.* **2015**, *281*, 787–805. [[CrossRef](#)]
24. Çelik, C.; Duman, M. Crank-Nicolson method for the fractional diffusion equation with the Riesz fractional derivative. *J. Comput. Phys.* **2012**, *231*, 1743–1750. [[CrossRef](#)]
25. Yang, J.; Li, Z.; Yan, Y. A new numerical method for solving Riesz space-fractional diffusion equation. *Math. Numer. Sin.* **2019**, *41*, 170–190. (In Chinese) [[CrossRef](#)]
26. Lin, X.-L.; Ng, M.K.; Sun, H.-W. Stability and convergence analysis of finite difference schemes for time-dependent space-fractional diffusion equations with variable diffusion coefficients. *J. Sci. Comput.* **2018**, *75*, 1102–1127.
27. Lin, X.-L.; Ng, M.K.; Sun, H.-W. Crank-Nicolson alternative direction implicit method for space-fractional diffusion equations with nonseparable coefficients. *SIAM J. Numer. Anal.* **2019**, *57*, 997–1019. [[CrossRef](#)]
28. Farquhar, N.E.; Moroney, T.J.; Yang, Q.; Turner, I.W. GPU accelerated algorithms for computing matrix function vector products with applications to exponential integrators and fractional diffusion. *SIAM J. Sci. Comput.* **2016**, *38*, C127–C149. [[CrossRef](#)]
29. Lin, X.-L.; Ng, M.K.; Sun, H.-W. A splitting preconditioner for Toeplitz-like linear systems arising from fractional diffusion equations. *SIAM J. Matrix Anal. Appl.* **2017**, *38*, 1580–1614. [[CrossRef](#)]
30. Huang, X.; Lin, X.-L.; Ng, M.K.; Sun, H.-W. Spectral analysis for preconditioning of multi-dimensional Riesz fractional diffusion equations. *arXiv* **2021**, arXiv:2102.01371. [[CrossRef](#)]
31. Simmons, A.; Yang, Q.; Moroney, T. A preconditioned numerical solver for stiff nonlinear reaction-diffusion equations with fractional Laplacians that avoids dense matrices. *J. Comput. Phys.* **2015**, *287*, 254–268.
32. Bertaccini, D.; Durastante, F. Efficient Preconditioner Updates for Semilinear Space-Time Fractional Reaction-Diffusion Equations. In *Structured Matrices in Numerical Linear Algebra*; Bini, D., Di Benedetto, F., Tyrtyshnikov, E., Van Barel, M., Eds.; Springer INdAM Series; Springer: Cham, Switzerland, 2019; Volume 30, pp. 285–302. [[CrossRef](#)]
33. Huang, X.; Sun, H.-W. A preconditioner based on sine transform for two-dimensional semi-linear Riesz space fractional diffusion equations in convex domains. *Appl. Numer. Math.* **2021**, *169*, 289–302.
34. Jian, H.-Y.; Huang, T.-Z.; Gu, X.-M.; Zhao, Y.-L. Fast compact implicit integration factor method with non-uniform meshes for the two-dimensional nonlinear Riesz space-fractional reaction-diffusion equation. *Appl. Numer. Math.* **2020**, *156*, 346–363. [[CrossRef](#)]
35. Sun, Z.-Z. *Numerical Solutions of Partial Differential Equations*, 2nd ed.; Science Press: Beijing, China, 2012. (In Chinese) [[CrossRef](#)]
36. Zhao, Y.-L.; Zhu, P.-Y.; Gu, X.-M.; Zhao, X.-L.; Jian, H.-Y. A preconditioning technique for all-at-once system from the nonlinear tempered fractional diffusion equation. *J. Sci. Comput.* **2020**, *83*, 10. [[CrossRef](#)]
37. Zhou, Y. *Application of Discrete Functional Analysis to the Finite Difference Methods*; International Academic Publishers: Beijing, China, 1990.
38. Van der Vorst, H.A. Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.* **1992**, *13*, 631–644.
39. Bini, D.; Di Benedetto, F. A new preconditioner for the parallel solution of positive definite Toeplitz systems. In *Proceedings of the Second Annual ACM Symposium on Parallel Algorithms and Architectures, SPAA '90, Island of Crete, Greece, 2–6 July 1990*; AMC: New York, NY, USA, 1990; pp. 220–223. [[CrossRef](#)]
40. Gu, X.-M.; Zhao, Y.-L.; Zhao, X.-L.; Carpentieri, B.; Huang, Y.-Y. A note on parallel preconditioning for the all-at-once solution of Riesz fractional diffusion equations. *Numer. Math. Theor. Meth. Appl.* **2021**, *14*, 893–919.