



Article

A Fast High-Order Compact Difference Scheme for Time-Fractional KS Equation with the Generalized Burgers' Type Nonlinearity

Huifa Jiang ^{1,*} and Da Xu ^{2,*}

¹ School of Science, Hunan University of Technology, Zhuzhou 412007, China

² School of Mathematics and Statistics, Hunan Normal University, Changsha 410081, China

* Correspondence: huifajiang98@163.com (H.J.); daxu@hunnu.edu.cn (D.X.)

Abstract: This work integrates the fast Alikhanov method with a compact scheme to solve the time-fractional Kuramoto–Sivashinsky (KS) equation with the generalized Burgers' type nonlinearity. Initially, the Alikhanov algorithm, designed to handle the Caputo fractional derivative on non-uniform time grids, effectively avoids the initial singularity. Additionally, the combination of the Alikhanov method with the sum-of-exponentials (SOE) technique significantly reduces both computational cost and memory requirements. By discretizing the spatial direction using a compact finite difference method, a fully discrete scheme is developed, achieving fourth-order convergence in the spatial domain. Stability and convergence are analyzed through energy methods. Several numerical examples are provided to validate the theoretical framework, demonstrating that the proposed algorithm is accurate, stable, and efficient.

Keywords: time-fractional Kuramoto–Sivashinsky equation; fast Alikhanov method; sum-of-exponentials technique; non-uniform time steps; compact difference method



Academic Editor: Luis Manuel Sánchez Ruiz

Received: 17 December 2024

Revised: 22 March 2025

Accepted: 28 March 2025

Published: 30 March 2025

Citation: Jiang, H.; Xu, D. A Fast High-Order Compact Difference Scheme for Time-Fractional KS Equation with the Generalized Burgers' Type Nonlinearity. *Fractal Fract.* **2025**, *9*, 218. <https://doi.org/10.3390/fractalfract9040218>

Copyright: © 2025 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

The Kuramoto–Sivashinsky (KS) equation belongs to an important class of chaotic partial differential equations that serve as a bridge between the partial differential equation and dynamical system frameworks. Kuramoto [1] originally proposed the KS equation in his study of phase turbulence in reaction–diffusion systems, while Sivashinsky [2] introduced it in his analysis of flame combustion propagation models. The KS equation has a wide range of applications in simulation science and engineering, such as the evolution of the flow of fluid films on inclined planes [3], flame front instability [4], interfacial turbulence [5], etc.

In this paper, we study the time-fractional KS equation with the generalized Burgers' type nonlinearity:

$${}^C_0 D_t^\alpha u(x, t) + u^p(x, t)u_x(x, t) - \mu u_{xx}(x, t) + \lambda u_{xxxx}(x, t) = f(x, t), \quad (1)$$

$$0 < x < L, \quad 0 < t \leq T,$$

with the initial condition (IC)

$$u(x, 0) = u_0(x), \quad 0 < x < L, \quad (2)$$

and the boundary conditions (BCs)

$$\begin{aligned} u(0, t) = u(L, t) = 0, \quad 0 \leq t \leq T, \\ u_{xx}(0, t) = u_{xx}(L, t) = 0, \quad 0 \leq t \leq T, \end{aligned} \quad (3)$$

where $\mu > 0$ is the viscosity coefficient, $p \geq 1$ is a positive integer, perturbation parameters $0 < \lambda \leq 1$, $L > 0$, $T > 0$ are fixed values, and $u_0(x)$ and $f(x, t)$ are prescribed real functions. The equations are complex, and their combination of fractional-order derivative KS equations and Burgers' nonlinear terms provides them with greater flexibility to capture the intricate dynamical behaviors observed in various physical systems.

- When the fourth-order derivative u_{xxxx} and the source term are not included in the equation, Equation (1) transforms into a time-fractional generalized Burgers equation [6] as follows:

$${}_0^C D_t^\alpha u(x, t) + u^p(x, t)u_x(x, t) - \mu u_{xx}(x, t) = 0, \quad 0 < x < L, \quad 0 < t \leq T, \quad (4)$$

where $u^p u_x$ is the generalized Burgers' type nonlinearity.

- When $\alpha = 1$, $p = 1$, Equation (1) is the usual KS equation as follows:

$$u_t(x, t) + u(x, t)u_x(x, t) - \mu u_{xx}(x, t) + u_{xxxx}(x, t) = 0, \quad 0 < x < L, \quad 0 < t \leq T, \quad (5)$$

and many pioneering studies have been carried out [7–9].

- When $p = 1$, Equation (1) is the generalized fractional KS equation [10] as follows:

$${}_0^C D_t^\alpha u(x, t) + u(x, t)u_x(x, t) - \mu u_{xx}(x, t) + u_{xxxx}(x, t) = 0, \quad 0 < x < L, \quad 0 < t \leq T, \quad (6)$$

In recent years, numerical computational algorithms for solving the time-fractional KS equation have been developed and become imperative. Many numerical methods are already available for studying KS equations, including those of the form presented in Equation (5). Mittal [11] proposed a quintic B-spline collocation method based on the Crank–Nicolson scheme for temporal discretization, aimed at approximating the solution to the KS equation. In [12], a locally discontinuous Galerkin method is implemented to obtain the numerical solution to the KS equation. The implicit–explicit BDF method was developed by Akrivis and Smyrlis [13] for simulating the KS equation.

However, there are few research studies on effective numerical algorithms for solving fractional KS equations. For the fractional KS equation of the type Equation (6), the weak singularity of the solution generated by the fractional derivative presents a difficult problem to solve. At present, the effective method to solve this problem is to use the variable step numerical method, which concentrates more grid points around the weak singularity and employs a sparse grid where the solution changes slowly. Indeed, the $L1$ formula for the construction of piecewise linear interpolation was proposed by Langlands [14], which leads to a temporal convergence order of $1 + \gamma$ for $0 < \gamma < 1$. Since then, the $L1$ formula has been widely used to deal with Caputo fractional derivatives [15–20]. Furthermore, for the numerical solution to pure sub-diffusion equations, there are several high-order discrete convolution forms for the approximation of the Caputo fractional-order time derivatives, such as the $L1-2$ scheme [21–23] and the $L2-1_\sigma$ formula, which is based on linear interpolation on the last subinterval $[t_{k-1}, t_k]$ and quadratic polynomial interpolation on the rest of the subintervals $[t_{k-1}, t_k]$ (for details, please refer to [24]). These formulas can achieve second-order temporal accuracy for sufficiently smooth solutions.

Due to the advantages of the $L2-1_\sigma$ formula, we will study the approximation of the Caputo fractional derivative by the variable time-step Alikhanov formula. Influenced by [25,26], we also give the regularity assumptions on the exact solution u as follows:

$$\left\| \frac{\partial^k u}{\partial t^k} \right\|_{H^2(0,L)} \leq C(1 + t^{\sigma-k}), \quad k = 0, 1, 2, 3, \quad (7)$$

$$\left\| {}_0^C D_t^\alpha u \right\|_{H^6(0,L)} + \|u\|_{H^8(0,L)} \leq C, \quad (8)$$

where $0 < t \leq T$, and $\sigma \in (0, 1) \cup (1, 2)$ is a regularity parameter. In this paper, we only consider the case of $\lambda = 1$.

In order to obtain a more efficient algorithm for solving the problem and avoid the issue of large computational costs caused by the historical dependence of the fractional derivative, one of the most effective methods is the SOE technique [27–29], which is applied in approximating the convolution kernel $t^{-\alpha}$ with a uniform absolute error ϵ . There have been many research studies on the nonlinear KS equation with a spatial fourth derivative, employing various methods such as the quintic B-spline collocation method [30], cubic Hermite collocation [31], Chebyshev collocation method [10], and Sinc–Galerkin method [32]. The compact difference scheme has good characteristics and high accuracy and is widely used to solve a variety of equations [33–37]. The application of a compact difference scheme to the spatial discretization of fractional KS equations is exactly the starting point of our present work. The theoretical part of the proof makes use of the classical energy method [38–40]. Moreover, the main contributions of this work can be summarized as follows:

- For the complex KS equation, we first apply the non-uniform Alikhanov formula to approximate the Caputo fractional derivative, and then treat the generalized Burgers' nonlinear term. Then, we combine the second-order time scheme and the SOE technique to establish the fast Alikhanov second-order scheme. This approach improves numerical accuracy and reduces computational costs.
- The high-order compact difference scheme is proposed for the first time to be applied to the KS equation with the generalized Burgers' nonlinear term, which can improve the spatial convergence rate to the fourth order.
- We demonstrate in detail the convergence and stability of the fully discrete scheme by the energy argument. In addition, we provide numerical examples to validate the theoretical analysis and demonstrate the efficiency of the algorithm by comparing fast and non-fast Alikhanov schemes.

The outline of this paper is as follows: In Section 2, we introduce some useful notations and lemmas. Additionally, some properties of compact difference formulas and fast Alikhanov formulas are analyzed. In Section 3, the high-order fast Alikhanov compact difference scheme is constructed. The stability and convergence of a full-discrete scheme are investigated. Numerical examples verify the stability and efficiency of the algorithm and the correctness of the theory in Section 4. The conclusion is summarized in Section 5.

2. Preliminaries

In this section, we first recall some basic formulations and useful lemmas. For ease of exposition and proof, we define some notations. To make our analysis extendable, the conditions and properties of the discrete convolution kernel in the Alikhanov formula are also introduced in this section.

2.1. Properties of Compact Difference Scheme

Let the spatial step size $h := L/M$, where M is a given positive integer. We denote $x_j = jh$ ($0 \leq j \leq M$) and $\bar{U}_h := \{u|u = (u_0, u_1, \dots, u_M), 0 \leq j \leq M\}$. For any $u, v \in \bar{U}_h$, we denote the following notations:

$$\begin{aligned}\nabla^+ u_{j+1} &= \frac{1}{h}(u_{j+1} - u_j), \quad \nabla^- u_{j-1} = \frac{1}{h}(u_j - u_{j-1}), \\ \nabla_h u_j &= \frac{1}{2h}(u_{j+1} - u_{j-1}), \quad \Delta_x u_j = \frac{u_{j+1} - 2u_j + u_{j-1}}{h^2}.\end{aligned}$$

We also need to introduce the subspace of $\bar{\mathcal{U}}_h$ as $\mathcal{U}_h := \{u \in \bar{\mathcal{U}}_h | u_0 = u_M = 0\}$, and define the inner product and norm as follows:

$$\begin{aligned}\langle u, v \rangle &= h \sum_{j=1}^{M-1} u_j v_j, \quad (u, v) = h \sum_{j=1}^M (\nabla^+ u_j) (\nabla^+ v_j), \\ \|u\| &= \sqrt{\langle u, u \rangle}, \quad |u|_1 = \sqrt{(u, u)}, \quad \|u\|_\infty = \max_{0 \leq j \leq M} |u_j|.\end{aligned}$$

Furthermore, we denote the following:

$$\mathcal{A}_1 u_j := (1 + \frac{h^2}{6} \Delta_x) u_j, \quad \mathcal{A}_2 u_j := (1 + \frac{h^2}{12} \Delta_x) u_j.$$

Thus, for the discretization of the second and fourth derivatives of the function $u = u(x, t)$, we have the following formulas:

$$u_x(x_j) = \mathcal{A}_1^{-1} \nabla_h U_j + \mathcal{O}(h^4), \quad u_{xx}(x_j) = \mathcal{A}_2^{-1} \Delta_x U_j + \mathcal{O}(h^4). \quad (9)$$

The following transformation can be advanced for the nonlinear term:

$$u^p u_x = v u_x = \frac{1}{p+2} [v u_x + (v u)_x], \quad (10)$$

where $v := u^p$, and

$$v_j \frac{\partial u_j}{\partial x} + \frac{\partial u_j v_j}{\partial x} = v_j \mathcal{A}_1^{-1} \nabla_h u_j + \mathcal{A}_1^{-1} \nabla_h (v_j u_j) + \mathcal{O}(h^4) = \varphi(u_j, v_j) + \mathcal{O}(h^4). \quad (11)$$

Lemma 1 ([41]). For any grid functions $u, v \in \mathcal{U}_h$, then the following holds:

$$\langle \nabla_h u, v \rangle = -\langle u, \nabla_h v \rangle, \quad \langle \Delta_x u, v \rangle = -(u, v).$$

Lemma 2 ([41]). For any grid functions $u \in \mathcal{U}_h$, then the following holds:

$$\|u\|_\infty \leq \frac{\sqrt{L}}{2} |u|_1, \quad \|u\| \leq \frac{L}{\sqrt{6}} |u|_1, \quad |u|_1 \leq \frac{2}{h} \|u\|.$$

Lemma 3 ([42]). For real symmetric positive-definite matrices $\mathcal{A}_1^{-1}$, $\mathcal{A}_2^{-1}$, and $\omega, u \in \mathcal{U}_h$, we obtain the following:

$$(\mathcal{A}_1^{-1} \omega, u) = (\mathcal{V}_1 \omega, \mathcal{V}_1 u) = (\omega, \mathcal{A}_1^{-1} u), \quad (\mathcal{A}_2^{-1} \omega, u) = (\mathcal{V}_2 \omega, \mathcal{V}_2 u) = (\omega, \mathcal{A}_2^{-1} u),$$

where $\mathcal{V}_1$ and $\mathcal{V}_2$ are upper triangular matrices, which can be obtained by the Cholesky factorization of $\mathcal{A}_1^{-1}$ and $\mathcal{A}_2^{-1}$, namely, $\mathcal{A}_1^{-1} = (\mathcal{V}_1)^T \mathcal{V}_1$, $\mathcal{A}_2^{-1} = (\mathcal{V}_2)^T \mathcal{V}_2$.

Proof. The proof of the theorem can be referred to in ([42], Lemma 2.1c). $\square$

Lemma 4 ([43]). Let $\omega, u \in \mathcal{U}_h$, then we can obtain the following:

$$\begin{aligned}(\mathcal{A}_1^{-1} \nabla_h \omega, u) &= (\nabla_h \mathcal{A}_1^{-1} \omega, u) = -(\mathcal{A}_1^{-1} \omega, \nabla_h u) = -(\omega, \mathcal{A}_1^{-1} \nabla_h u), \\ (\mathcal{A}_2^{-1} \Delta_x \omega, u) &= (\Delta_x \mathcal{A}_2^{-1} \omega, u) = -(\nabla^+ \mathcal{A}_2^{-1} \omega, \nabla^+ u) = -(\mathcal{A}_2^{-1} \nabla^+ \omega, \nabla^+ u).\end{aligned}$$

Lemma 5. Let $u, v \in \mathcal{U}_h$, we arrive at

$$\langle \varphi(u, v), u \rangle = 0.$$

Proof. For $u, v \in \mathcal{U}_h$, using Lemmas 1 and 4, we can obtain the following:

$$\begin{aligned} \langle \varphi(u, v), u \rangle &= \langle v_j \mathcal{A}_1^{-1} \nabla_h u_j + \mathcal{A}_1^{-1} \nabla_h (v_j u_j), u_j \rangle \\ &= \langle v_j \mathcal{A}_1^{-1} \nabla_h u_j, u_j \rangle + \langle \mathcal{A}_1^{-1} \nabla_h (v_j u_j), u_j \rangle, \\ &= \langle \mathcal{A}_1^{-1} \nabla_h u_j, v_j u_j \rangle - \langle v_j u_j, \mathcal{A}_1^{-1} \nabla_h u_j \rangle, \\ &= 0. \end{aligned}$$

□

Lemma 6 ([43]). For real symmetric positive-definite matrices $\mathcal{A}_1^{-1}$, $\mathcal{A}_2^{-1}$, and $\omega \in \mathcal{U}_h$, we obtain the following:

$$\|\omega\|^2 \leq (\mathcal{A}_1^{-1} \omega, \omega) = \|\mathcal{V}_1 \omega\|^2 \leq 3\|\omega\|^2, \quad \|\omega\|^2 \leq (\mathcal{A}_2^{-1} \omega, \omega) = \|\mathcal{V}_2 \omega\|^2 \leq \frac{3}{2}\|\omega\|^2.$$

2.2. Properties of the Fast Alikhanov Scheme

In order to construct the fast Alikhanov scheme, we first consider the time levels $0 = t_0 < t_1 < t_2 < \dots < t_N = T$, where N is a given positive integer. We define the n th step size $\tau_n := t_n - t_{n-1}$ for $1 \leq n \leq N$, the maximum step size $\tau := \max_{0 \leq n \leq N} \tau_n$. We set the time level $t_{n-\theta} := \theta t_{n-1} + (1-\theta)t_n$ with a fixed parameter $\theta = \alpha/2$. It is necessary to set the local step-size ratios:

$$\rho_n := \frac{\tau_n}{\tau_{n+1}} \text{ for } 1 \leq n \leq N-1 \text{ and } \rho := \max_{1 \leq n \leq N-1} \rho_n.$$

For any time sequence $(v^n)_{n=0}^N$, we define $\nabla_\tau v^n := v^n - v^{n-1}$ and the interpolated value $v^{n-\theta} := \theta v^{n-1} + (1-\theta)v^n$. Thus, it is advisable to denote $\Pi_{1,k}u$ as the linear interpolant function of a function, u , with respect to t_{k-1}, t_k , and to denote the quadratic interpolant of u with respect to t_{k-1}, t_k and t_{k+1} as $\Pi_{2,k}u$, respectively. Recalling the definition of $\nabla_\tau u^k$, it follows that for $k \geq 1$, we have the following:

$$\begin{aligned} (\Pi_{1,k}u)'(t) &= \frac{\nabla_\tau u^k}{\tau_k}, \\ (\Pi_{2,k}u)'(t) &= \frac{\nabla_\tau u^k}{\tau_k} + \frac{2(t - t_{k-1/2})}{\tau_k(\tau_k + \tau_{k+1})} (\rho_k \nabla_\tau u^{k+1} - \nabla_\tau u^k). \end{aligned}$$

Next, the Caputo fractional derivative can be approximated by the defined interpolant polynomial at $t_{n-\theta}$ to obtain the Alikhanov approximation:

$$\begin{aligned} (D_\tau^\alpha u)^{n-\theta} &:= \int_{t_{n-1}}^{t_{n-\theta}} \omega_{1-\alpha}(t_{n-\theta} - s) (\Pi_{1,n}u)'(s) ds + \sum_{k=1}^{n-1} \int_{t_{k-1}}^{t_k} \omega_{1-\alpha}(t_{n-\theta} - s) (\Pi_{2,k}u)'(s) ds \\ &= a_0^{(n)} \nabla_\tau u^n + \sum_{k=1}^{n-1} \left(a_{n-k}^{(n)} \nabla_\tau u^k + \rho_k b_{n-k}^{(n)} \nabla_\tau u^{k+1} - b_{n-k}^{(n)} \nabla_\tau u^k \right), \\ &= \sum_{k=1}^n A_{n-k}^{(\alpha, n)} \nabla_\tau v^k, \end{aligned} \tag{12}$$

we have $A_0^{(1)} := a_0^{(1)}$ if $n = 1$, and for $n \geq 2$,

$$A_{n-k}^{(n)} := \begin{cases} a_0^{(n)} + \rho_{n-1}b_1^{(n)}, & k = n, \\ a_{n-k}^{(n)} + \rho_{k-1}b_{n-k+1}^{(n)} - b_{n-k}^{(n)}, & 2 \leq k \leq n-1, \\ a_{n-1}^{(n)} - b_{n-1}^{(n)}, & k = 1. \end{cases} \quad (13)$$

Here, $A_{n-k}^{(n)}$ is the discrete convolution kernel of the Alikhanov formula; next, the SOE technique is applied to derive the discrete convolution kernel of the fast Alikhanov formula.

Lemma 7 ([27]). (SOE) Let $\alpha \in (0, 1)$, $\epsilon \ll 1$, and $\Delta t > 0$. Assume there exists a positive integer N_q , and positive quadrature nodes s^l and $\omega^l > 0$ ($1 \leq l \leq N_q$), such that we have the following:

$$\left| \omega_{1-\alpha}(t) - \sum_{l=1}^{N_q} \omega^l e^{-s^l t} \right| \leq \epsilon, \quad t \in [\Delta t, T],$$

and the quadrature node number N_q satisfies

$$N_q = O\left(\log \frac{1}{\epsilon} \left(\log \log \frac{1}{\epsilon} + \log \frac{T}{\Delta t}\right) + \log \frac{1}{\Delta t} \left(\log \log \frac{1}{\epsilon} + \log \frac{1}{\Delta t}\right)\right).$$

Now, the fractional derivative can be approximated in combination with the SOE technique in Lemma 7 at the time point $t_{n-\theta}$ for $1 \leq n \leq N$

$$\begin{aligned} {}^C_0 D_f^\alpha u(t_{n-\theta}) &\approx \int_{t_{n-1}}^{t_{n-\theta}} \omega_{1-\alpha}(t_{n-\theta} - s) (\Pi_{1,n} u)'(s) ds + \sum_{k=1}^{n-1} \int_{t_{k-1}}^{t_k} \sum_{l=1}^{N_q} \omega^l e^{-s^l(t_{n-\theta}-s)} u'(s) ds \\ &= \hat{\mathbf{a}}_0^{(n)} \nabla_\tau u^n + \sum_{k=1}^{n-1} \left(\hat{\mathbf{a}}_{n-k}^{(n)} \nabla_\tau u^k + \rho_k \hat{\mathbf{b}}_{n-k}^{(n)} \nabla_\tau u^{k+1} - \hat{\mathbf{b}}_{n-k}^{(n)} \nabla_\tau u^k \right) \\ &= \sum_{k=1}^n \hat{\mathbf{A}}_{n-k}^{(n)} \nabla_\tau u^k, \end{aligned} \quad (14)$$

where coefficients $\hat{\mathbf{a}}_{n-k}^{(n)}$ and $\hat{\mathbf{b}}_{n-k}^{(n)}$ are defined by the following:

$$\hat{\mathbf{a}}_0^{(n)} := a_0^{(n)}, \quad \hat{\mathbf{a}}_{n-k}^{(n)} := \frac{1}{\tau_k} \int_{t_{k-1}}^{t_k} \sum_{l=1}^{N_q} \omega^l e^{-s^l(t_{n-\theta}-s)} ds, \quad 1 \leq k \leq n-1,$$

$$\hat{\mathbf{b}}_{n-k}^{(n)} := \frac{2}{\tau_k(\tau_k + \tau_{k+1})} \int_{t_{k-1}}^{t_k} \sum_{l=1}^{N_q} \omega^l e^{-s^l(t_{n-\theta}-s)} e^{-s^l(t_{k+1}-\theta-s)} (s - t_{k-\frac{1}{2}}) ds, \quad 1 \leq k \leq n-1.$$

We reformulate the discrete convolution kernel $\hat{\mathbf{A}}_{n-k}^{(n)}$ as follows: $\hat{\mathbf{A}}_0^{(1)} := a_0^{(1)}$ if $n = 1$ and for $n \geq 2$

$$\hat{\mathbf{A}}_{n-k}^{(n)} := \begin{cases} \hat{\mathbf{a}}_0^{(n)} + \rho_{n-1} \hat{\mathbf{b}}_1^{(n)}, & \text{for } k = n, \\ \hat{\mathbf{a}}_{n-k}^{(n)} + \rho_{k-1} \hat{\mathbf{b}}_{n-k+1}^{(n)} - \hat{\mathbf{b}}_{n-k}^{(n)}, & \text{for } 2 \leq k \leq n-1, \\ \hat{\mathbf{a}}_{n-1}^{(n)} - \hat{\mathbf{b}}_{n-1}^{(n)}, & \text{for } k = 1. \end{cases} \quad (15)$$

In order to effectively assess some necessary properties of the discrete convolution kernel $\hat{\mathbf{A}}_{n-k}^{(n)}$, the time step needs to satisfy the following two conditions:

M1. The maximum time-step ratio is $\rho = 7/4$.

M2. There exists a positive constant C_γ , such that $\tau_k \leq C_\gamma \tau \min\{1, t_k^{1-1/\gamma}\}$ for $1 \leq k \leq N$, with $t_k \leq C_\gamma t_{k-1}$ and $\tau_k/t_k \leq C_\gamma \tau_{k-1}/t_{k-1}$ for $2 \leq k \leq N$.

Lemma 8 ([27]). Assume that the tolerance error ϵ of the SOE approximation satisfies

$$\epsilon \leq \min \left\{ \frac{\alpha_p}{2(1-\alpha_p)} \omega_{1-\alpha_p}(T), \frac{1}{26} \omega_{1-\alpha_p}(T) \right\},$$

then the discrete kernel $\hat{\mathbf{A}}_{n-k}^{(n)}$ satisfies the following:

(I) $\hat{\mathbf{A}}_{n-k}^{(n)}$ are bounded, $\hat{\mathbf{A}}_{n-k}^{(n)} \leq \frac{2}{\tau_n} \omega_{2-\alpha_p}(\tau_n)$ and

$$\hat{\mathbf{A}}_{n-k}^{(n)} \geq \frac{1}{2\tau_k} \int_{t_{k-1}}^{t_k} \omega_{1-\alpha}(t_n - s) ds, \quad 1 \leq k \leq n;$$

(II) $\hat{\mathbf{A}}_{n-k}^{(n)}$ are monotone, and

$$\hat{\mathbf{A}}_{n-k-1}^{(n)} - \hat{\mathbf{A}}_{n-k}^{(n)} \geq (1 + \rho_k) \left(\frac{1}{5\rho_k} + 1 \right) b_{n-k}^{(n)}, \quad 1 \leq k \leq n-1;$$

(III) $\hat{\mathbf{A}}_0^{(n)} - \hat{\mathbf{A}}_1^{(n)} > \theta (2\hat{\mathbf{A}}_0^{(n)} - \hat{\mathbf{A}}_1^{(n)})$ for $n \geq 2$.

In this case, for the later analysis of global consistency, we introduce the complementary discrete convolution kernels $P_{n-k}^{(n)}$ of $\hat{\mathbf{A}}_{n-k}^{(n)}$ that satisfy

$$\sum_{i=k}^n P_{n-i}^{(n)} \hat{\mathbf{A}}_{i-k}^{(i)} \equiv 1. \quad (16)$$

which yields the following recursive formulas:

$$P_0^{(n)} := \frac{1}{\hat{\mathbf{A}}_0^{(n)}}, \quad P_i^{(n)} := \frac{1}{\hat{\mathbf{A}}_0^{(i)}} \sum_{k=i+1}^n \left(\hat{\mathbf{A}}_{k-i-1}^{(k)} - \hat{\mathbf{A}}_{k-i}^{(k)} \right) P_k^{(n)}, \quad 1 \leq i \leq n-1.$$

In fact, according to Lemma 8, the complementary discrete kernels are non-negative and satisfy the following:

$$\sum_{i=1}^n P_{n-i}^{(n)} \omega_{1+(m-1)\alpha}(t_i) \leq \pi_A \omega_{1+m\alpha}(t_n), \quad m = 0, 1, \quad 1 \leq n \leq N. \quad (17)$$

Lemma 9 ([44]). Let λ_s be nonnegative constants with $0 < \sum_{s=0}^n \lambda_s \leq \Lambda$ for $n \geq 1$, where Λ is some positive constant independent of n . Suppose that the nonnegative sequences ξ_n and η_n are bounded, and the nonnegative grid function $\{v_n \mid n \geq 0\}$ satisfies the following:

$$\sum_{k=1}^n A_{n-k}^{(n)} \nabla_{\tau}(v^k)^2 \leq \sum_{s=1}^n \lambda_{n-s} (v^{s,\theta_1})^2 + \xi^n v^{n,\theta_2} + (\eta^n)^2, \quad n \geq 1, \quad (18)$$

where for $i = 1, 2$ and each n , we set $v^{n,\theta_1} := \theta_i v^{n-1} + (1 - \theta_i) v^n$ for some constant $\theta_i \in [0, 1]$. If the nonuniform grid satisfies the maximum time-step criterion $\tau_N \leq [\frac{11}{2} \Gamma(2 - \alpha) \Lambda]^{-1/\alpha}$, then we have the following:

$$v^n \leq 2E_{\alpha} \left(\frac{11}{2} \Lambda t_n^{\alpha} \right) \left[v^0 + \max_{1 \leq k \leq n} \sum_{j=1}^k P_{k-j}^{(k)} (\xi^j + \eta^j) + \max_{1 \leq j \leq n} \eta^j \right], \quad 1 \leq n \leq N,$$

where $E_{\alpha}(z) := \sum_{k=0}^{\infty} z^k / \Gamma(1 + k\alpha)$, and $E_{\alpha}(\cdot)$ represents the Mittag-Leffler function.

Lemma 10 ([45]). Under the condition M1 and Lemma 8, the fast Alikhanov Formula (14) satisfies the following:

$$\left\langle (\mathcal{D}_\tau^\alpha v)^{n-\theta}, v^{n-\theta} \right\rangle \geq \frac{1}{2} \sum_{k=1}^n A_{n-k}^{(n)} \nabla_\tau (\|v^k\|^2), \quad 1 \leq n \leq N.$$

3. Establishment and Analysis of Compact Scheme

In this section, we shall derive the second-order fast compact difference scheme for problems (1)–(3). After that, the convergence and stability of the fast compact difference algorithm are rigorously proved.

3.1. Numerical Scheme

First, combined with (11), the temporal direction of Equation (1) at the grid points $(x_j, t_{n-\theta})$ is discretized by the fast nonuniform Alikhanov Formula (14) and the spatial direction is discretized by the compact difference Formula (9), where we obtain the following:

$$\begin{cases} (\mathcal{D}_f^\alpha U_j)^{n-\theta} + \frac{1}{p+2} \varphi(V_j^{n-\theta}, U_j^{n-\theta}) - \mu \mathcal{A}_2^{-1} \Delta_x U_j^{n-\theta} + \mathcal{A}_2^{-1} \Delta_x W_j^{n-\theta} = f_j^{n-\theta} + \Lambda_j^{n-\theta}, \\ 1 \leq j \leq M-1, \quad 1 \leq n \leq N, \end{cases} \quad (19)$$

$$V_j^n = (U_j^n)^p, \quad 0 \leq j \leq M, \quad 0 \leq n \leq N, \quad (20)$$

$$W_j^{n-\theta} = \Delta_x U_j^{n-\theta} + R_3^{n-\theta}, \quad 0 \leq j \leq M, \quad 0 \leq n \leq N, \quad (21)$$

where $\Lambda_j^{n-\theta}$ and $R_3^{n-\theta}$ denote the truncation errors.

Omitting the small terms $\Lambda_j^{n-\theta}$ and $R_3^{n-\theta}$ in (19), we replace the functions U_j^n , W_j^n , and V_j^n with their numerical approximations u_j^n , ω_j^n , and v_j^n , respectively. Meanwhile, combining IC (2) and BC (3), we can obtain the fully discrete difference scheme:

$$\begin{cases} (\mathcal{D}_f^\alpha u_j)^{n-\theta} + \frac{1}{p+2} \varphi(v_j^{n-\theta}, u_j^{n-\theta}) - \mu \mathcal{A}_2^{-1} \Delta_x u_j^{n-\theta} + \mathcal{A}_2^{-1} \Delta_x \omega_j^{n-\theta} = f_j^{n-\theta}, \\ 1 \leq j \leq M-1, \quad 1 \leq n \leq N, \end{cases} \quad (22)$$

$$v_j^n = (u_j^n)^p, \quad 0 \leq j \leq M, \quad 0 \leq n \leq N, \quad (23)$$

$$\omega_j^{n-\theta} = \Delta_x u_j^{n-\theta}, \quad 0 \leq j \leq M, \quad 0 \leq n \leq N, \quad (24)$$

$$u_j^0 = u_0(x_j), \quad 1 \leq j \leq M-1, \quad (25)$$

$$u_0^n = u_M^n = \omega_0^n = \omega_M^n = 0, \quad 0 \leq n \leq N. \quad (26)$$

In what follows, the stability and convergence analysis of the above fully discrete schemes (22)–(26) are carried out by the energy method.

3.2. Stability

Here, we present the stability result as follows:

Theorem 1. Suppose that $\{u^n\}_{n=0}^N$ is the solution to (22)–(26), and under the conditions of Lemma 8 and M2, then we have the following:

$$\|u^n\| \leq \|u_0\| + \frac{11}{2} \Gamma(1-\alpha) \max_{1 \leq k \leq n} t_k^\alpha \|f^{k-\theta}\|.$$

Proof. Taking an inner product of (22) with $u^{n-\theta}$, we obtain the following:

$$\begin{aligned} \left\langle \left(\mathcal{D}_f^\alpha u \right)^{n-\theta}, u^{n-\theta} \right\rangle + \frac{1}{p+2} \left\langle \varphi(v^{n-\theta}, u^{n-\theta}), u^{n-\theta} \right\rangle - \mu \left\langle \mathcal{A}_2^{-1} \Delta_x u^{n-\theta}, u^{n-\theta} \right\rangle + \left\langle \mathcal{A}_2^{-1} \Delta_x \omega^{n-\theta}, u^{n-\theta} \right\rangle \\ = \left\langle f^{n-\theta}, u^{n-\theta} \right\rangle. \end{aligned} \quad (27)$$

By Lemmas 5 and 10, we have the following:

$$\left\langle \left(\mathcal{D}_f^\alpha u \right)^{n-\theta}, u^{n-\theta} \right\rangle \geq \frac{1}{2} \sum_{k=1}^n A_{n-k}^{(n)} \nabla_\tau (\|u^k\|^2), \quad (28)$$

$$\frac{1}{p+2} \left\langle \varphi(v^{n-\theta}, u^{n-\theta}), u^{n-\theta} \right\rangle = 0, \quad (29)$$

and

$$-\mu \left\langle \mathcal{A}_2^{-1} \Delta_x u^{n-\theta}, u^{n-\theta} \right\rangle = \mu \left\langle \mathcal{A}_2^{-1} \nabla^+ u^{n-\theta}, \nabla^+ u^{n-\theta} \right\rangle \geq \mu \|u^{n-\theta}\|_1^2, \quad (30)$$

$$\left\langle \mathcal{A}_2^{-1} \Delta_x \omega^{n-\theta}, u^{n-\theta} \right\rangle = \left\langle \mathcal{A}_2^{-1} \omega^{n-\theta}, \Delta_x u^{n-\theta} \right\rangle = \left\langle \mathcal{A}_2^{-1} \omega^{n-\theta}, \omega^{n-\theta} \right\rangle \geq 0. \quad (31)$$

Under the relations (28)–(31), we have the following:

$$\sum_{k=1}^n A_{n-k}^{(n)} \nabla_\tau (\|u^k\|^2) \leq 2 \|f^{n-\theta}\| \|u^{n-\theta}\|.$$

Taking $\lambda_s = 0$ for $s \geq 0$, $\eta^n = 0$ and $\xi^n = 2 \|f^{n-\theta}\|$ for $n \geq 1$ in Lemma 9 and $\pi_A = \frac{11}{4}$ in Lemma 8, it is easy to obtain

$$\|u^n\| \leq \|u_0\| + \max_{1 \leq k \leq n} \sum_{i=1}^k 2P_{k-i}^{(k)} \|f^{k-\theta}\| \leq \|u_0\| + 2\pi_A \Gamma(1-\alpha) \max_{1 \leq k \leq n} t_k^\alpha \|f^{k-\theta}\|.$$

Consequently, we infer the following:

$$\|u^n\| \leq \|u_0\| + \frac{11}{2} \Gamma(1-\alpha) \max_{1 \leq k \leq n} t_k^\alpha \|f^{k-\theta}\|.$$

□

3.3. Convergence

Before analyzing the convergence of the fast Alikhanov schemes (22)–(26), we first analyze the truncation errors $\Lambda_j^{n-\theta}$ and $(R_3)_j^{n-\theta}$. In fact, $\Lambda_j^{n-\theta}$ is composed of the following four parts:

$$\Lambda_j^{n-\theta} := (R_{t1})_j^{n-\theta} + (R_{t2})_j^{n-\theta} + (R_3)_j^{n-\theta} + (R_s)_j^{n-\theta}, \quad 1 \leq j \leq M-1, \quad 1 \leq n \leq N,$$

where $(R_s)_j^{n-\theta}$ means that the error is caused by approximating space derivatives; it can easily be seen that $(R_s)_j^{n-\theta} = \mathcal{O}(h^4)$ from (9) and (11). Moreover, we have the following:

$$\begin{aligned}(R_{t1})_j^{n-\theta} &:= \left(\mathcal{D}_f^\alpha u_j\right)^{n-\theta} - {}^C_0 D_t^\alpha u(x_j, t_{n-\theta}), \\(R_{t2})_j^{n-\theta} &:= -\frac{1}{p+2} \varphi((R_v)_j^{n-\theta}, (R_u)_j^{n-\theta}) + \mu \mathcal{A}_2^{-1} \Delta_x (R_u)_j^{n-\theta} - \mathcal{A}_2^{-1} \Delta_x (R_\omega)_j^{n-\theta},\end{aligned}$$

where $(R_u)_j^{n-\theta}$, $(R_v)_j^{n-\theta}$ and $(R_\omega)_j^{n-\theta}$ denote the errors of the weighted time approximation at $t_{n-\theta}$, namely, we have the following:

$$\begin{aligned}(R_u)_j^{n-\theta} &:= u(x_j, t_{n-\theta}) - [\theta U_j^{n-1} + (1-\theta)U_j^n], \\(R_v)_j^{n-\theta} &:= v(x_j, t_{n-\theta}) - [\theta V_j^{n-1} + (1-\theta)V_j^n], \\(R_\omega)_j^{n-\theta} &:= \omega(x_j, t_{n-\theta}) - [\theta W_j^{n-1} + (1-\theta)W_j^n].\end{aligned}$$

Based on the mesh condition **M1** and the regularity (7), and consulting Lemma 3.3 and Lemma 3.4 in [25,27], we arrive at the following:

$$\sum_{k=1}^n P_{n-k}^{(n)} |(R_{t1})_j^{k-\theta}| \leq C \left(\frac{\tau_1^\sigma}{\sigma} + \frac{\epsilon}{\sigma} t_n^\alpha \hat{t}_{n-1}^2 + \frac{1}{1-\alpha} \max_{2 \leq k \leq n} t_k^\alpha t_{k-1}^{\sigma-3} \tau_k^3 / \tau_{k-1}^\alpha \right), \quad 1 \leq n \leq N, \quad (32)$$

$$\sum_{k=1}^n P_{n-k}^{(n)} |(R_u)_j^{k-\theta}| \leq C \left(\frac{\tau_1^{\sigma+\alpha}}{\sigma} + t_n^\alpha \max_{2 \leq k \leq n} t_{k-1}^{\sigma-2} \tau_k^2 \right), \quad 1 \leq n \leq N, \quad (33)$$

where $\hat{t}_n = \max\{1, t_n\}$.

Combining (17) and $(R_s)_j^{n-\theta} = \mathcal{O}(h^4)$, we can also obtain the following:

$$\max_{1 \leq k \leq n} \sum_{i=1}^k P_{k-i}^{(k)} |(R_s)_j^{i-\theta}| \leq Ch^4, \quad 1 \leq n \leq N. \quad (34)$$

Therefore, combining (20)–(22), the following holds:

$$\max_{1 \leq k \leq n} \sum_{i=1}^k P_{k-i}^{(k)} \|\Lambda^{i-\theta}\| \leq C \left(\frac{\tau_1^\sigma}{\sigma} + \frac{\epsilon}{\sigma} t_n^\alpha \hat{t}_{n-1}^2 + \max_{2 \leq i \leq k} t_i^\alpha t_{i-1}^{\sigma-3} \tau_i^3 / \tau_{i-1}^\alpha + t_k^\alpha \max_{2 \leq i \leq k} t_{i-1}^{\sigma-2} \tau_i^2 + h^4 \right), \quad 1 \leq n \leq N. \quad (35)$$

To obtain a pointwise error estimate of (22)–(26), we need to apply an important method, namely, the truncated function method. Based on the regularity assumption (8), we denote

$Q := \max_{(x,t) \in [0,L] \times [0,T]} |u(x,t)|$. We can find a second-order smooth function as follows:

$$g(u) := \begin{cases} u^p, & |u| \leq Q+1, \\ 0, & |u| \geq Q+2. \end{cases}$$

Furthermore, let $c_1 := \max_{u \in R} |g(u)|$, $c_2 := \max_{u \in R} |g'(u)|$. It is not difficult to verify that the constants c_1 and c_2 depend only on Q and p by the Hermite polynomial interpolant. Clearly, (20) and (23) are equivalent to the following:

$$\begin{cases} V_j^n = g(U_j^n), & 0 \leq j \leq M, \quad 0 \leq n \leq N, \\ v_j^n = g(u_j^n), & 0 \leq j \leq M, \quad 0 \leq n \leq N. \end{cases} \quad (36)$$

$$(37)$$

The convergence of the full discrete schemes (22)–(26) is studied below. Subtracting (19)–(21) from (22)–(24), respectively, the error equation can be written as follows:

$$\begin{cases} (\mathcal{D}_t^\alpha e_j)^{n-\theta} + \frac{1}{p+2} [\varphi(V_j^{n-\theta}, U_j^{n-\theta}) - \varphi(v_j^{n-\theta}, u_j^{n-\theta})] - \mu \mathcal{A}_2^{-1} \Delta_x e_j^{n-\theta} + \mathcal{A}_2^{-1} \Delta_x \eta_j^{n-\theta} = \Lambda_j^{n-\theta}, \\ 1 \leq j \leq M-1, \quad 1 \leq n \leq N, \end{cases} \quad (38)$$

$$\rho_j^{n-\theta} = g(U_j^{n-\theta}) - g(u_j^{n-\theta}), \quad 0 \leq j \leq M, \quad 0 \leq n \leq N, \quad (39)$$

$$\eta_j^{n-\theta} = \mathcal{A}_2^{-1} \Delta_x e_j^{n-\theta} + (R_3)_j^{n-\theta}, \quad 0 \leq j \leq M, \quad 0 \leq n \leq N, \quad (40)$$

$$e_j^0 = 0, \quad 1 \leq j \leq M-1, \quad (41)$$

$$e_0^n = e_M^n = \eta_0^n = \eta_M^n = 0, \quad 0 \leq n \leq N, \quad (42)$$

where $e_j^n := U_j^n - u_j^n$, $\varrho_j^n := V_j^n - v_j^n$, $\eta_j^n := W_j^n - \omega_j^n$.

Theorem 2. Suppose that the exact solution, u , satisfies the regularity property in (7) and (8) with the parameter $\sigma \in (0, 1) \cup (1, 2)$. Let **M1**, **M2** hold, and $\hat{t}_n = \max\{1, t_n\}$ for $1 \leq n \leq N$. If $\tau \leq ((11/2)\Gamma(2-\alpha)c_3)^{-1/\alpha}$, the following estimate holds:

$$\|e^n\| \leq C(\tau^{\min\{\gamma\sigma, 2\}} + h^4 + \epsilon), \quad 0 \leq n \leq N. \quad (43)$$

Proof. Firstly, according to the differential mean value theorem, from (39), it is clear that

$$|\varrho_j^n| \leq c_2 |e_j^n|, \quad 0 \leq j \leq M, \quad 0 \leq n \leq N,$$

moreover, we obtain the following:

$$\|\varrho^n\| \leq c_2 \|e^n\|, \quad 0 \leq n \leq N. \quad (44)$$

The next thing to do in the proof is to take an inner product of (38) with $e^{n-\theta}$; we obtain the following:

$$\begin{aligned} & \left\langle (\mathcal{D}_t^\alpha e)^{n-\theta}, e^{n-\theta} \right\rangle + \frac{1}{p+2} \left\langle \varphi(V^{n-\theta}, U^{n-\theta}) - \varphi(v^{n-\theta}, u^{n-\theta}), e^{n-\theta} \right\rangle \\ & - \mu \left\langle \mathcal{A}_2^{-1} \Delta_x e^{n-\theta}, e^{n-\theta} \right\rangle + \left\langle \mathcal{A}_2^{-1} \Delta_x \eta^{n-\theta}, e^{n-\theta} \right\rangle = \left\langle \Lambda^{n-\theta}, e^{n-\theta} \right\rangle. \end{aligned} \quad (45)$$

Furthermore, taking an inner product of (40) with $\eta^{n-\theta}$, we obtain the following:

$$\left\langle \eta^{n-\theta}, \eta^{n-\theta} \right\rangle = \left\langle \mathcal{A}_2^{-1} \Delta_x e^{n-\theta}, \eta^{n-\theta} \right\rangle + \left\langle R_3^{n-\theta}, \eta^{n-\theta} \right\rangle. \quad (46)$$

Another step is to analyze the above equation term by term. Using Lemma 10, we have the following:

$$\left\langle (\mathcal{D}_t^\alpha e)^{n-\theta}, e^{n-\theta} \right\rangle \geq \frac{1}{2} \sum_{k=1}^n A_{n-k}^{(n)} \nabla_\tau (\|e^k\|^2). \quad (47)$$

By the regularity assumption (7), let $c_0 := \max_{(x,t) \in [0,L] \times [0,T]} \{|u(x,t)|, |u_x(x,t)|\}$. Applying Lemmas 2–6 and the Cauchy–Schwarz inequality, and noting that $v^{n-\theta} = V^{n-\theta} - \varrho^{n-\theta}$ and $u^{n-\theta} = U^{n-\theta} - e^{n-\theta}$, we have the following:

$$\begin{aligned}
 & \left\langle \varphi(V^{n-\theta}, U^{n-\theta}) - \varphi(v^{n-\theta}, u^{n-\theta}), e^{n-\theta} \right\rangle \\
 &= \left\langle \varphi(\varrho^{n-\theta}, U^{n-\theta}) + \varphi(V^{n-\theta}, e^{n-\theta}) - \varphi(\varrho^{n-\theta}, e^{n-\theta}), e^{n-\theta} \right\rangle \\
 &= \left\langle \varphi(\varrho^{n-\theta}, U^{n-\theta}), e^{n-\theta} \right\rangle \\
 &= \left\langle \varrho^{n-\theta} \mathcal{A}_1^{-1} \nabla_h U^{n-\theta} + \mathcal{A}_1^{-1} \nabla_h (\varrho^{n-\theta} U^{n-\theta}), e^{n-\theta} \right\rangle \\
 &= \left(\varrho^{n-\theta} \mathcal{A}_1^{-1} \nabla_h U^{n-\theta}, e^{n-\theta} \right) + \left(\mathcal{A}_1^{-1} \nabla_h (\varrho^{n-\theta} U^{n-\theta}), e^{n-\theta} \right) \\
 &= \left(\nabla_h U^{n-\theta}, \mathcal{A}_1^{-1} (\varrho^{n-\theta} e^{n-\theta}) \right) - \left(\varrho^{n-\theta} U^{n-\theta}, \mathcal{A}_1^{-1} \nabla_h e^{n-\theta} \right) \\
 &\leq \| \mathcal{V}_1 \nabla_h U^{n-\theta} \| \| \mathcal{V}_1 (\varrho^{n-\theta} e^{n-\theta}) \| + \| \mathcal{V}_1 (\varrho^{n-\theta} U^{n-\theta}) \| \| \mathcal{V}_1 \nabla_h e^{n-\theta} \| \\
 &\leq 3Lc_0 \| \varrho^{n-\theta} e^{n-\theta} \| + 3c_0 \| e^{n-\theta} \|_1 \| \varrho^{n-\theta} \| \\
 &\leq 3Lc_0 \| \varrho^{n-\theta} \| \| e^{n-\theta} \|_\infty + 3c_0 c_2 \| e^{n-\theta} \|_1 \| e^{n-\theta} \| \\
 &\leq 3Lc_0 c_2 \| e^{n-\theta} \| \frac{\sqrt{L}}{2} \| e^{n-\theta} \|_1 + 3c_0 c_2 \| e^{n-\theta} \|_1 \| e^{n-\theta} \| \\
 &\leq 3c_0 c_2 (L^{3/2} + 1) \| e^{n-\theta} \| \| e^{n-\theta} \|_1 \\
 &\leq \frac{9c_0^2 c_1^2 (L^{3/2} + 1)^2}{4\mu} \| e^{n-\theta} \|^2 + \mu \| e^{n-\theta} \|_1^2,
 \end{aligned} \tag{48}$$

as well as

$$-\mu \left\langle \mathcal{A}_2^{-1} \Delta_x e^{n-\theta}, e^{n-\theta} \right\rangle = \mu \left\langle \mathcal{A}_2^{-1} \nabla^+ e^{n-\theta}, \nabla^+ e^{n-\theta} \right\rangle \geq \mu \| e^{n-\theta} \|_1^2, \tag{49}$$

$$\left\langle \Lambda^{n-\theta}, e^{n-\theta} \right\rangle \leq \| \Lambda^{n-\theta} \| \| e^{n-\theta} \|. \tag{50}$$

Substituting the results of (47)–(49) into (34), and adding (46), we have the following:

$$\begin{aligned}
 \frac{1}{2} \sum_{k=1}^n A_{n-k}^{(n)} \nabla_\tau (\| e^k \|^2) + \left\langle \eta^{n-\theta}, \eta^{n-\theta} \right\rangle &\leq \frac{9c_0^2 c_1^2 (L^{3/2} + 1)^2}{4\mu} \| e^{n-\theta} \|^2 + \| \Lambda^{n-\theta} \| \| e^{n-\theta} \| + \left\langle R_3^{n-\theta}, \eta^{n-\theta} \right\rangle. \\
 \sum_{k=1}^n A_{n-k}^{(n)} \nabla_\tau (\| e^k \|^2) &\leq \frac{9c_0^2 c_1^2 (L^{3/2} + 1)^2}{2\mu} \| e^{n-\theta} \|^2 + 2 \| \Lambda^{n-\theta} \| \| e^{n-\theta} \| + \frac{1}{2} \| R_3^{n-\theta} \|^2.
 \end{aligned}$$

According to Lemma 9, $\lambda_0 = \frac{9c_0^2 c_1^2 (L^{3/2} + 1)^2}{2\mu} < c_3$, $\tilde{\zeta}^n = 2 \| \Lambda^{n-\theta} \|$, $\eta^n = \frac{\sqrt{2}}{2} \| R_3^{n-\theta} \|$, we can obtain the following:

$$\| e^n \| \leq 2E_\alpha \left(\frac{11}{2} c_3 t_n^\alpha \right) \left[\| e^0 \| + \max_{1 \leq k \leq n} \sum_{j=1}^k P_{k-j}^{(k)} (2 \| \Lambda^{j-\theta} \| + \frac{\sqrt{2}}{2} \| R_3^{j-\theta} \|) + \frac{\sqrt{2}}{2} \max_{1 \leq j \leq n} R_3^j \right], \quad 1 \leq n \leq N. \tag{51}$$

Due to $\| e^0 \| = 0$, $2E_\alpha (\frac{11}{2} c_3 t_n^\alpha) \leq C$, and combining (35), we have the following:

$$\| e^n \| \leq C \left(\frac{\tau_1^\sigma}{\sigma} + \frac{\epsilon}{\sigma} t_n^\alpha \hat{t}_{n-1}^2 + \max_{2 \leq i \leq k} t_i^\alpha t_{i-1}^{\sigma-3} \tau_i^3 / \tau_{i-1}^\alpha + t_k^\alpha \max_{2 \leq i \leq k} t_{i-1}^{\sigma-2} \tau_i^2 + h^4 \right). \tag{52}$$

On the other hand, if **M1** and **M2** hold, $\beta = \min\{2, \gamma\sigma\}$, we have the following:

$$\begin{aligned} t_k^\alpha t_{k-1}^{\sigma-3} \tau_k^3 \tau_{k-1}^{-\alpha} &\leq C_\gamma t_k^{\alpha+\sigma-3} \tau_k^{3-\alpha} \\ &\leq C_\gamma t_k^{\alpha+\sigma-3} \tau_k^{3-\alpha-\beta} \left(\tau \min\{1, t_k^{1-1/\gamma}\} \right)^\beta \\ &\leq C_\gamma t_k^{\sigma-\beta/\gamma} (\tau_k/t_k)^{3-\alpha-\beta} \tau^\beta \\ &\leq C_\gamma t_k^{\max\{0, \sigma-(3-\alpha)/\gamma\}} \tau^\beta \quad \text{for } 2 \leq k \leq n, \end{aligned} \quad (53)$$

$$\begin{aligned} t_{k-1}^{\sigma-2} \tau_k^2 &\leq C_\gamma t_k^{\sigma-2} \tau_k^{2-\beta} \left(\tau \min\{1, t_k^{1-1/\gamma}\} \right)^\beta \\ &\leq C_\gamma t_k^{\sigma-\beta/\gamma} (\tau_k/t_k)^{2-\beta} \tau^\beta \\ &\leq C_\gamma t_k^{\max\{0, \sigma-2/\gamma\}} \tau^\beta, \quad \text{for } 2 \leq k \leq n. \end{aligned} \quad (54)$$

As a result, the discrete solution $u_j^{n-\theta}$ is convergent in the sense that

$$\|e^n\| \leq C(\tau^{\min\{\gamma\sigma, 2\}} + h^4 + \epsilon), \quad 0 \leq n \leq N. \quad (55)$$

This finishes the proof. $\square$

4. Numerical Experiment

In this section, we test the stability and convergence of the fast compact schemes (22)–(26) for three different experiments on a nonuniform grid $\{t_n = T(n/N)^\gamma, n = 1, \dots, N\}$. All experiments were performed on a Windows server with an AMD Ryzen 5 4600H processor of 16GB RAM and 3.00 GHz CPU. In the following experiments, the tolerance accuracy of the SOE algorithm is limited to $\epsilon = 10^{-12}$ in order to balance accuracy and efficiency [6]. The following suitable formulas are given to calculate the error and convergence order of the following numerical solution:

$$\begin{aligned} E^{SOE}(M, N) &= \|U^N - u^N\|_2 = \sqrt{h \sum_{j=1}^{M-1} (U_j^N - u_j^N)^2}, \\ Rate_h^{SOE} &= \log_2 \left(\frac{E^{SOE}(M, N)}{E^{SOE}(2M, N)} \right), \quad Rate_\tau^{SOE} = \log_2 \left(\frac{E^{SOE}(M, N)}{E^{SOE}(M, 2N)} \right), \end{aligned}$$

where U^N and u^N represent the exact and numerical solutions at the N th time level, respectively.

In addition, we define the following notation to illustrate the numerical implementation of the proposed scheme:

$$U^* = \max_{1 \leq n \leq N} \|U^n\|.$$

Before proceeding further, let us remark on the numerical scheme setting of the fast Alikhanov approach and Alikhanov approach.

- Alikhanov scheme: The Caputo fractional derivative discretization scheme is (12).
- Fast Alikhanov scheme: The Caputo fractional derivative discretization scheme is (14).

Example 1. In this example, we consider the following problem:

$${}_0^C D_t^\alpha u + u^p u_x - \mu u_{xx} + u_{xxx} = f(x, t), \quad 0 < x < 1, \quad 0 \leq t \leq 1, \quad (56)$$

$$u(x, 0) = u_0(x), \quad 0 < x < 1, \quad (57)$$

$$u_{xx}(0, t) = u_{xx}(1, t), \quad 0 \leq t \leq 1, \quad (58)$$

and the exact solution is set as $u(x, t) = \left(1 + \frac{t^\alpha}{\Gamma(1+\alpha)}\right) \sin(\pi x)$. Accordingly, the initial condition $u_0 = \sin(\pi x)$ and the source term $f(x, t)$ are as follows:

$$f(x, t) = \sin(\pi x) \left(1 + \mu\pi^2 + \mu\pi^2 \frac{t^\alpha}{\Gamma(1+\alpha)} + \pi^4 + \pi^4 \frac{t^\alpha}{\Gamma(1+\alpha)} \right) + \pi \left(1 + \frac{t^\alpha}{\Gamma(1+\alpha)} \right)^{p+1} (\sin(\pi x))^{p+1} \cos(\pi x).$$

For Example 1, in Table 1, we fix $M = 256$, $\mu = 1$, and $\gamma = 2/\alpha$ to explore the performances of the fast Alikhanov and Alikhanov schemes in terms of errors in the L^2 -norm and the convergence order for different α and p cases. It can be found that the errors in the L^2 -norm of the Alikhanov and fast Alikhanov schemes are very close and both of them can reach second-order convergence in the temporal direction with $\alpha \in \{0.3, 0.5, 0.7\}$ and $p \in \{2, 3, 4\}$. When $p \in \{2, 3, 4\}$, the value of α keeps on changing, the errors in Alikhanov and fast Alikhanov schemes have the same effect and also achieve second-order convergence. Table 2 shows the errors and convergence orders in the spatial direction for different μ and p when fixing $N = 512$, $\alpha = 0.5$, and $\gamma = 4$. It can be observed that regardless of whether μ changes or p changes, the spatial errors in the L^2 -norm of the two schemes are similar and both can achieve fourth-order convergence in agreement with the theoretical prediction.

Table 3 considers the cases where $N = 512$, $p = 2$, $\mu = 1$, and $\gamma = 2$ are fixed, with α changing, which indicates that the fast Alikhanov and Alikhanov schemes can achieve similar spatial errors and fourth-order convergence. The CPU times of the two algorithms are given; it is obvious that the fast Alikhanov scheme is much faster than the Alikhanov scheme. In order to further confirm whether the temporal and spatial convergence orders of the fast Alikhanov scheme remain stable as p increases, the temporal convergence and spatial convergence orders are shown in Figures 1 and 2. Table 4 shows that with the gradual increase in N , the value of U^* tends to stabilize and shows no increasing trend. These results confirm the numerical stability of the proposed scheme in the time direction.

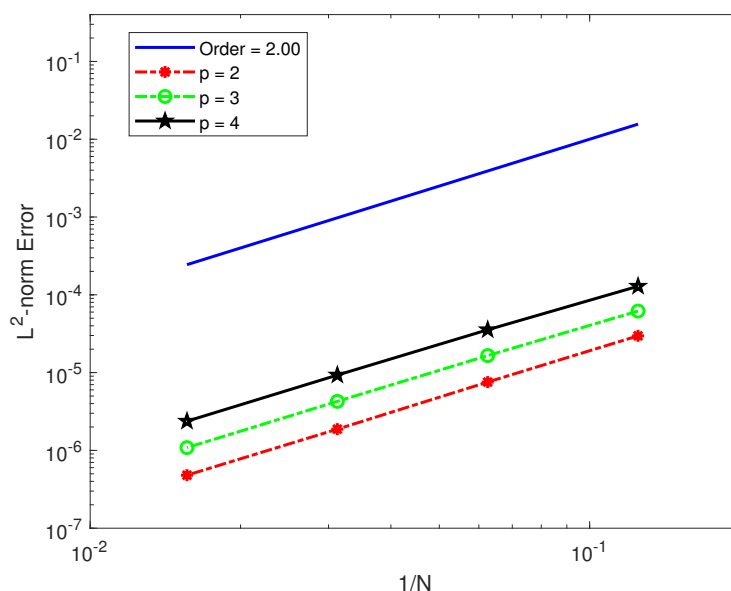


Figure 1. The temporal convergence orders of the fast compact difference scheme when $\alpha = 0.5$, $\mu = 1$, $\gamma = 4$ and $N = 256$.

Table 1. Temporal error and convergence order of Example 1 for different α and p with $M = 256$, $\mu = 1$, $\gamma = 2/\alpha$.

	N	$E^{SOE}(M, N)$	$Rate_{\tau}^{SOE}$	$E^{SOE}(M, N)$	$Rate_{\tau}^{SOE}$	$E^{SOE}(M, N)$	$Rate_{\tau}^{SOE}$
$\alpha = 0.3$	Alikhanov		$p = 2$	$p = 3$		$p = 4$	
	8	2.6080×10^{-5}	-	4.2294×10^{-5}	-	8.0635×10^{-5}	-
	16	6.2591×10^{-6}	2.0589	1.1197×10^{-5}	1.9173	2.2588×10^{-5}	1.8358
	32	1.4056×10^{-6}	2.1548	2.8265×10^{-6}	1.9861	5.9572×10^{-6}	1.9228
	64	3.2266×10^{-7}	2.1230	7.0951×10^{-7}	1.9941	1.5306×10^{-6}	1.9605
	Fast Alikhanov		$p = 2$	$p = 3$		$p = 4$	
	8	2.6081×10^{-5}	-	4.2298×10^{-5}	-	8.0632×10^{-5}	-
	16	6.2657×10^{-6}	2.0575	1.1195×10^{-5}	1.9177	2.2588×10^{-5}	1.8358
	32	1.4042×10^{-6}	2.1578	2.8276×10^{-6}	1.9852	5.9571×10^{-6}	1.9229
	64	3.2485×10^{-7}	2.1119	7.0925×10^{-7}	1.9952	1.5302×10^{-6}	1.9609
	Alikhanov		$p = 2$	$p = 3$		$p = 4$	
	8	2.9692×10^{-5}	-	6.1797×10^{-5}	-	1.2884×10^{-4}	-
	16	7.5590×10^{-6}	1.9738	1.6557×10^{-5}	1.9001	3.5529×10^{-5}	1.8585
	32	1.8727×10^{-6}	2.0131	4.2675×10^{-6}	1.9560	9.2949×10^{-6}	1.9345
	64	4.8342×10^{-7}	1.9537	1.0864×10^{-6}	1.9739	2.3738×10^{-6}	1.9693
	Fast Alikhanov		$p = 2$	$p = 3$		$p = 4$	
	8	2.9697×10^{-5}	-	6.1799×10^{-5}	-	1.2884×10^{-4}	-
	16	7.5559×10^{-6}	1.9746	1.6555×10^{-5}	1.9003	3.5529×10^{-5}	1.8585
	32	1.8753×10^{-6}	2.0105	4.2675×10^{-6}	1.9558	9.2949×10^{-6}	1.9345
	64	4.8013×10^{-7}	1.9657	1.0853×10^{-6}	1.9752	2.3737×10^{-6}	1.9693
$\alpha = 0.5$	Alikhanov		$p = 2$	$p = 3$		$p = 4$	
	8	3.2325×10^{-5}	-	7.1925×10^{-5}	-	1.5150×10^{-4}	-
	16	8.3560×10^{-6}	1.9517	1.8960×10^{-5}	1.9235	4.0683×10^{-5}	1.8968
	32	2.1056×10^{-6}	1.9886	4.8485×10^{-6}	1.9674	1.0492×10^{-5}	1.9552
	64	5.3111×10^{-7}	1.9872	1.2273×10^{-6}	1.9820	2.6629×10^{-6}	1.9782
	Fast Alikhanov		$p = 2$	$p = 3$		$p = 4$	
	8	3.2332×10^{-5}	-	7.1927×10^{-5}	-	1.5149×10^{-4}	-
	16	8.3605×10^{-6}	1.9513	1.8961×10^{-5}	1.9235	4.0681×10^{-5}	1.8968
	32	2.1158×10^{-6}	1.9824	4.8494×10^{-6}	1.9672	1.0485×10^{-5}	1.9560
	64	5.2915×10^{-7}	1.9995	1.2265×10^{-6}	1.9833	2.6618×10^{-6}	1.9779
	Alikhanov		$p = 2$	$p = 3$		$p = 4$	
	8	3.2325×10^{-5}	-	7.1925×10^{-5}	-	1.5150×10^{-4}	-
	16	8.3560×10^{-6}	1.9517	1.8960×10^{-5}	1.9235	4.0683×10^{-5}	1.8968
	32	2.1056×10^{-6}	1.9886	4.8485×10^{-6}	1.9674	1.0492×10^{-5}	1.9552
	64	5.3111×10^{-7}	1.9872	1.2273×10^{-6}	1.9820	2.6629×10^{-6}	1.9782
	Fast Alikhanov		$p = 2$	$p = 3$		$p = 4$	
	8	3.2332×10^{-5}	-	7.1927×10^{-5}	-	1.5149×10^{-4}	-
	16	8.3605×10^{-6}	1.9513	1.8961×10^{-5}	1.9235	4.0681×10^{-5}	1.8968
	32	2.1158×10^{-6}	1.9824	4.8494×10^{-6}	1.9672	1.0485×10^{-5}	1.9560
	64	5.2915×10^{-7}	1.9995	1.2265×10^{-6}	1.9833	2.6618×10^{-6}	1.9779

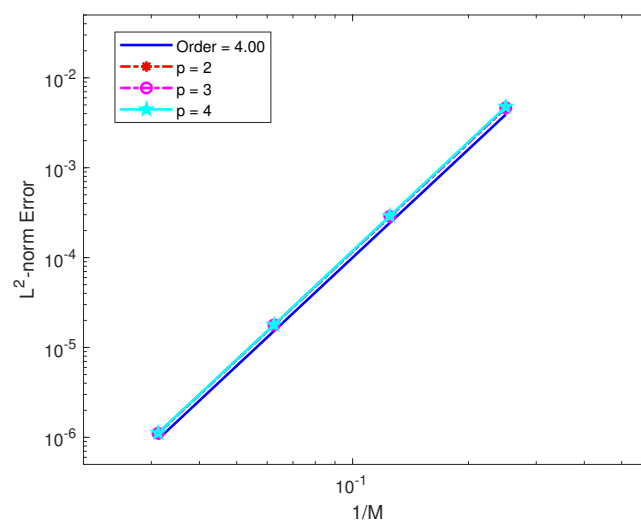
**Figure 2.** The spatial convergence orders of the fast compact difference scheme when $\alpha = 0.5$, $\mu = 1$, $\gamma = 4$ and $N = 512$.

Table 2. Space error and convergence order of Example 1 for different μ and p with $N = 512$, $\alpha = 0.5$, $\gamma = 4$.

M		$E^{SOE}(M, N)$	$Rate_h^{SOE}$	$E^{SOE}(M, N)$	$Rate_h^{SOE}$	$E^{SOE}(M, N)$	$Rate_h^{SOE}$
$\mu = 0.1$	Alikhanov	$p = 2$		$p = 3$		$p = 4$	
	4	4.8317×10^{-3}	-	4.8272×10^{-3}	-	4.9532×10^{-3}	-
	8	2.9702×10^{-4}	4.0239	2.9925×10^{-4}	4.0118	3.0394×10^{-4}	4.0265
	16	1.8486×10^{-5}	4.0060	1.8543×10^{-5}	4.0124	1.8659×10^{-5}	4.0258
	32	1.1542×10^{-6}	4.0015	1.1572×10^{-6}	4.0021	1.1689×10^{-6}	3.9966
	Fast Alikhanov	$p = 2$		$p = 3$		$p = 4$	
	4	4.8317×10^{-3}	-	4.8272×10^{-3}	-	4.9532×10^{-3}	-
	8	2.9702×10^{-4}	4.0239	2.9925×10^{-4}	4.0118	3.0394×10^{-4}	4.0265
	16	1.8486×10^{-5}	4.0060	1.8543×10^{-5}	4.0124	1.8659×10^{-5}	4.0258
	32	1.1542×10^{-6}	4.0015	1.1572×10^{-6}	4.0021	1.1689×10^{-6}	3.9966
	Alikhanov	$p = 2$		$p = 3$		$p = 4$	
	4	4.6355×10^{-3}	-	4.6325×10^{-3}	-	4.7695×10^{-3}	-
	8	2.8495×10^{-4}	4.0239	2.8705×10^{-4}	4.0124	2.9150×10^{-4}	4.0323
	16	1.7735×10^{-5}	4.0060	1.7788×10^{-5}	4.0123	1.7899×10^{-5}	4.0256
	32	1.1073×10^{-6}	4.0015	1.1102×10^{-6}	4.0021	1.1212×10^{-6}	3.9967
	Fast Alikhanov	$p = 2$		$p = 3$		$p = 4$	
	4	4.6355×10^{-3}	-	4.6325×10^{-3}	-	4.7695×10^{-3}	-
	8	2.8495×10^{-4}	4.0239	2.8705×10^{-4}	4.0124	2.9150×10^{-4}	4.0323
	16	1.7735×10^{-5}	4.0060	1.7788×10^{-5}	4.0123	1.7899×10^{-5}	4.0256
	32	1.1073×10^{-6}	4.0015	1.1102×10^{-6}	4.0021	1.1212×10^{-6}	3.9967
$\mu = 1$	Alikhanov	$p = 2$		$p = 3$		$p = 4$	
	4	3.6473×10^{-3}	-	3.6500×10^{-3}	-	3.8049×10^{-3}	-
	8	2.2412×10^{-4}	4.0245	2.2550×10^{-4}	4.0167	2.2849×10^{-4}	4.0576
	16	1.3948×10^{-5}	4.0062	1.3981×10^{-5}	4.0116	1.4058×10^{-5}	4.0227
	32	8.7088×10^{-7}	4.0014	8.7269×10^{-7}	4.0019	8.8006×10^{-7}	3.9976
	Fast Alikhanov	$p = 2$		$p = 3$		$p = 4$	
	4	3.6473×10^{-3}	-	3.6500×10^{-3}	-	3.8049×10^{-3}	-
	8	2.2412×10^{-4}	4.0245	2.2550×10^{-4}	4.0167	2.2849×10^{-4}	4.0576
	16	1.3948×10^{-5}	4.0062	1.3981×10^{-5}	4.0116	1.4058×10^{-5}	4.0227
	32	8.7088×10^{-7}	4.0014	8.7269×10^{-7}	4.0019	8.8006×10^{-7}	3.9976
$\mu = 10$	Alikhanov	$p = 2$		$p = 3$		$p = 4$	
	4	3.6473×10^{-3}	-	3.6500×10^{-3}	-	3.8049×10^{-3}	-
	8	2.2412×10^{-4}	4.0245	2.2550×10^{-4}	4.0167	2.2849×10^{-4}	4.0576
	16	1.3948×10^{-5}	4.0062	1.3981×10^{-5}	4.0116	1.4058×10^{-5}	4.0227
	32	8.7088×10^{-7}	4.0014	8.7269×10^{-7}	4.0019	8.8006×10^{-7}	3.9976
	Fast Alikhanov	$p = 2$		$p = 3$		$p = 4$	
	4	3.6473×10^{-3}	-	3.6500×10^{-3}	-	3.8049×10^{-3}	-
	8	2.2412×10^{-4}	4.0245	2.2550×10^{-4}	4.0167	2.2849×10^{-4}	4.0576
	16	1.3948×10^{-5}	4.0062	1.3981×10^{-5}	4.0116	1.4058×10^{-5}	4.0227
	32	8.7088×10^{-7}	4.0014	8.7269×10^{-7}	4.0019	8.8006×10^{-7}	3.9976

Table 3. Errors, convergence orders in space, and CPU times for the Alikhanov scheme and fast Alikhanov scheme by fixing $N = 512$, $p = 2$, $\mu = 1$, and $\gamma = 2$ for Example 1.

		$\alpha = 0.3$		$\alpha = 0.5$		$\alpha = 0.7$	
Alikhanov	M	$E^{SOE}(M, N)$	$Rate_h^{SOE}$	$E^{SOE}(M, N)$	$Rate_h^{SOE}$	$E^{SOE}(M, N)$	$Rate_h^{SOE}$
	4	4.6004×10^{-3}	-	4.6355×10^{-3}	-	4.5792×10^{-3}	-
	8	2.8278×10^{-4}	4.0240	2.8495×10^{-4}	4.0239	2.8149×10^{-4}	4.0240
	16	1.7600×10^{-5}	4.0061	1.7735×10^{-5}	4.0060	1.7555×10^{-5}	4.0031
	32	1.0988×10^{-6}	4.0016	1.1073×10^{-6}	4.0015	1.1327×10^{-6}	3.9541
CPU(s)		82.777		75.941		82.308	
		$\alpha = 0.3$		$\alpha = 0.5$		$\alpha = 0.7$	
Fast Alikhanov	M	$E^{SOE}(M, N)$	$Rate_h^{SOE}$	$E^{SOE}(M, N)$	$Rate_h^{SOE}$	$E^{SOE}(M, N)$	$Rate_h^{SOE}$
	4	4.6004×10^{-3}	-	4.6355×10^{-3}	-	4.5792×10^{-3}	-
	8	2.8278×10^{-4}	4.0240	2.8495×10^{-4}	4.0239	2.8145×10^{-4}	4.0241
	16	1.7600×10^{-5}	4.0060	1.7735×10^{-5}	4.0060	1.7517×10^{-5}	4.0061
	32	1.0997×10^{-6}	4.0005	1.1073×10^{-6}	4.0015	1.0939×10^{-6}	4.0011
CPU(s)		3.9295		3.8685		3.9980	

Table 4. Numerical solutions when $M = 256$, $p = 2$, $\mu = 1$, and $\gamma = 2/\alpha$ for Example 1.

	$\alpha = 0.3$	$\alpha = 0.5$	$\alpha = 0.7$
N	U^*	U^*	U^*
4	1.4949	1.5049	1.4853
8	1.4950	1.5050	1.4853
16	1.4950	1.5050	1.4853
32	1.4950	1.5050	1.4853
64	1.4950	1.5050	1.4853
128	1.4950	1.5050	1.4853

Example 2. In order to verify the wide range of applications of the numerical scheme, the case where the exact solution is unknown is tested in this numerical example. We consider problems (1)–(3) in $\Omega = (0, \pi)$ to verify the effectiveness and high accuracy of the numerical scheme. Let $T = 1$, the initial condition is $u_0(x) = \sin(x)$, and the source term is $f(x, t) = 0$.

In Table 5, the errors in the L^2 -norm, spatial convergence orders, and CPU times of the Alikhanov scheme and the fast Alikhanov numerical scheme are presented for fixed values of $p = 2$, $\mu = 1$, $\gamma = 2$, and $N = 2048$, with $\alpha \in \{0.3, 0.5, 0.7\}$. It is worth mentioning that both numerical schemes can achieve the same approximation effect, but the CPU time of the fast scheme is much shorter. Table 6 demonstrates that for fixed values of $M = 512$, $\mu = 1$, $\alpha = 0.8$, and $\gamma = 2/\alpha$, the L^2 -norm error decreases as the number of temporal subintervals increases. It is expected that the convergence order of the fast approximate scheme is $\mathcal{O}(\tau^2)$. Meanwhile, the error and convergence order remain as expected for changing values of p , suggesting that the approximation scheme is stable for different p . Table 7 shows the errors, temporal convergence orders, and CPU times for different μ , where we take $M = 512$, $p = 2$, and $\gamma = 2/\alpha$. In each row of the table, the errors and convergence orders change very little as μ varies, so the method is stable for the coefficient μ , as predicted by our theoretical analysis. In addition, in order to demonstrate more intuitively the speedup effect of the SOE approximation, taking $\alpha = 0.5$, $p = 2$, $\mu = 1$, $\gamma = 2$, and $M = 4$, we plot the computation time curves of the two schemes for the N increase, as shown in Figure 3. It is clear that the fast Alikhanov scheme saves a lot of computational costs compared to the Alikhanov scheme.

Table 5. Errors, convergence orders in space, and CPU times for the Alikhanov scheme and fast Alikhanov scheme by fixing $p = 2$, $\mu = 1$, $\gamma = 2$, and $N = 2048$ for Example 2.

		$\alpha = 0.3$		$\alpha = 0.5$		$\alpha = 0.7$	
		$E^{SOE}(M, N)$	$Rate_h^{SOE}$	$E^{SOE}(M, N)$	$Rate_h^{SOE}$	$E^{SOE}(M, N)$	$Rate_h^{SOE}$
Alikhanov	M						
	4	4.3605×10^{-4}	-	4.3568×10^{-4}	-	4.5247×10^{-4}	-
	8	2.6588×10^{-5}	4.0356	2.6570×10^{-5}	4.0354	2.7511×10^{-5}	4.0398
	16	1.6529×10^{-6}	4.0077	1.6517×10^{-6}	4.0078	1.7094×10^{-6}	4.0084
	32	1.0316×10^{-7}	4.0020	1.0308×10^{-7}	4.0021	1.0667×10^{-7}	4.0022
CPU(s)		1253.8		1197.3		1279.9	
		$\alpha = 0.3$		$\alpha = 0.5$		$\alpha = 0.7$	
		$E^{SOE}(M, N)$	$Rate_h^{SOE}$	$E^{SOE}(M, N)$	$Rate_h^{SOE}$	$E^{SOE}(M, N)$	$Rate_h^{SOE}$
Fast Alikhanov	M						
	4	4.3605×10^{-4}	-	4.3568×10^{-4}	-	4.5247×10^{-4}	-
	8	2.6588×10^{-5}	4.0356	2.6570×10^{-5}	4.0354	2.7511×10^{-5}	4.0398
	16	1.6529×10^{-6}	4.0077	1.6517×10^{-6}	4.0078	1.7094×10^{-6}	4.0084
	32	1.0316×10^{-7}	4.0020	1.0308×10^{-7}	4.0021	1.0667×10^{-7}	4.0022
CPU(s)		44.704		45.165		45.322	

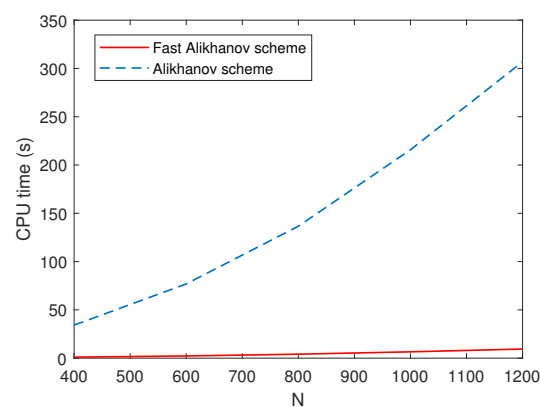
Table 6. The E^{SOE} -norm errors and temporal convergence orders with $M = 512$, $\mu = 1$, $\alpha = 0.8$, and $\gamma = 2/\alpha$.

N	$p = 1$		$p = 2$		$p = 3$		$p = 4$		$p = 5$	
	$E^{SOE}(M, N)$	$Rate_{\tau}^{SOE}$	$E^{SOE}(M, N)$	$Rate_{\tau}^{SOE}$	$E^{SOE}(M, N)$	$Rate_{\tau}^{SOE}$	$E^{SOE}(M, N)$	$Rate_{\tau}^{SOE}$	$E^{SOE}(M, N)$	$Rate_{\tau}^{SOE}$
10	1.3260×10^{-3}	-	1.3265×10^{-3}	-	1.3266×10^{-3}	-	1.3266×10^{-3}	-	1.3266×10^{-3}	-
20	3.4271×10^{-4}	1.9520	3.4279×10^{-4}	1.9522	3.4281×10^{-4}	1.9522	3.4281×10^{-4}	1.9522	3.4281×10^{-4}	1.9523
40	9.0337×10^{-5}	1.9236	9.0353×10^{-5}	1.9237	9.0356×10^{-5}	1.9237	9.0354×10^{-5}	1.9238	9.0351×10^{-5}	1.9238
80	2.3914×10^{-5}	1.9174	2.3917×10^{-5}	1.9176	2.3917×10^{-5}	1.9176	2.3914×10^{-5}	1.9177	2.3914×10^{-5}	1.9177
160	6.3120×10^{-6}	1.9217	6.3132×10^{-6}	1.9216	6.3131×10^{-6}	1.9216	6.3145×10^{-6}	1.9211	6.3158×10^{-6}	1.9208

Table 7. The E^{SOE} -norm errors, temporal convergence orders, and CPU time (seconds) with $M = 512$, $p = 2$, and $\gamma = 2/\alpha$.

N	$\mu = 0.05$			N	$\mu = 0.1$		
	$E^{SOE}(M, N)$	$Rate_{\tau}^{SOE}$	CPU(s)		$E^{SOE}(M, N)$	$Rate_{\tau}^{SOE}$	CPU(s)
4	2.8796×10^{-3}	-	0.4605	4	3.2656×10^{-3}	-	0.4406
8	6.8144×10^{-4}	2.0792	0.7595	8	7.7751×10^{-4}	2.0704	0.8029
16	1.7604×10^{-4}	1.9527	1.5174	16	2.0089×10^{-4}	1.9525	1.5480
32	4.8379×10^{-5}	1.8634	2.5469	32	5.4828×10^{-5}	1.8734	2.7098
64	1.3632×10^{-5}	1.8273	5.0930	64	1.5262×10^{-5}	1.8450	5.3880

N	$\mu = 1$			N	$\mu = 2$		
	$E^{SOE}(M, N)$	$Rate_{\tau}^{SOE}$	CPU(s)		$E^{SOE}(M, N)$	$Rate_{\tau}^{SOE}$	CPU(s)
4	8.7476×10^{-3}	-	0.4646	4	9.7976×10^{-3}	-	0.5341
8	2.0697×10^{-3}	2.0794	0.7903	8	2.3022×10^{-3}	2.0894	0.7326
16	5.2813×10^{-4}	1.9705	1.4359	16	5.8600×10^{-4}	1.9741	1.3253
32	1.3868×10^{-4}	1.9291	2.6138	32	1.5258×10^{-4}	1.9413	2.5373
64	3.6684×10^{-5}	1.9185	5.0402	64	3.9897×10^{-5}	1.9353	4.7893

**Figure 3.** Comparison of the CPU time between the fast Alikhanov and Alikhanov schemes with $\alpha = 0.5$, $p = 2$, $\mu = 1$, $\gamma = 2$, and $M = 4$.

Example 3. In this example, we consider problem (1) over domains $\Omega = (0, 1)$ and $T = 1$. The analytic solution is $u(x, t) = \left(\frac{t^\alpha}{\Gamma(1+\alpha)}\right) \sin(\pi x) \cos(\pi x)$, such that the source term $f(x, t)$ is as follows:

$$f(x, t) = \sin(\pi x) \cos(\pi x) + 2\pi^2 \mu \frac{t^\alpha}{\Gamma(1+\alpha)} \sin(2\pi x) + 8\pi^4 \frac{t^\alpha}{\Gamma(1+\alpha)} \sin(2\pi x) + \pi \left(\frac{t^\alpha}{\Gamma(1+\alpha)}\right)^{p+1} (\sin(\pi x))^p (\cos(\pi x))^p \cos(2\pi x).$$

Tables 8 and 9 display the temporal and spatial errors, spatial-temporal convergence orders, and CPU times for various α values. Furthermore, the results show that the proposed scheme can reach the second order of convergence in time and the fourth order of convergence in space. Next, we test the stability by fixing $M = 256$, $\mu = 1$, $p = 2$, and

$\gamma = 2/\alpha$ and selecting different T values, which means the time-step sizes are different. The numerical results are listed in Table 10, which shows that the fast Alikhanov scheme is stable and has a relaxed stability restriction. In Table 11, we tested the errors for $\alpha = 0.95, \alpha = 0.99, \alpha = 0.999$, and $\alpha = 0.9999$ by making $M = 256, \mu = 1, p = 2$, and $\gamma = 2/\alpha$, which also shows that the numerical scheme is stable when $\alpha \rightarrow 1^-$.

Table 8. Errors, temporal convergence orders, and CPU times by fixing $p = 2, \mu = 1, \gamma = 4$, and $M = 256$ for Example 3.

N	$\alpha = 0.3$			$\alpha = 0.5$			$\alpha = 0.7$		
	$E^{SOE}(M, N)$	$Rate_{\tau}^{SOE}$	CPU	$E^{SOE}(M, N)$	$Rate_{\tau}^{SOE}$	CPU	$E^{SOE}(M, N)$	$Rate_{\tau}^{SOE}$	CPU
16	9.5519×10^{-7}	-	0.2818	3.9181×10^{-6}	-	0.2457	8.5622×10^{-6}	-	0.2060
32	2.4356×10^{-7}	1.9715	0.4170	1.0148×10^{-6}	1.9490	0.3583	2.2256×10^{-6}	1.9438	0.4410
64	5.8573×10^{-8}	2.0560	0.7595	2.5816×10^{-7}	1.9748	0.7026	5.6804×10^{-7}	1.9701	0.7078
128	1.6584×10^{-8}	1.8205	1.8818	6.6561×10^{-8}	1.9555	1.4596	1.4255×10^{-7}	1.9946	1.5336

Table 9. Errors, convergence orders in space, and CPU times by fixing $p = 2, \mu = 1, \gamma = 4$, and $N = 1000$ for Example 3.

M	$\alpha = 0.3$			$\alpha = 0.5$			$\alpha = 0.7$		
	$E^{SOE}(M, N)$	$Rate_h^{SOE}$	CPU	$E^{SOE}(M, N)$	$Rate_h^{SOE}$	CPU	$E^{SOE}(M, N)$	$Rate_h^{SOE}$	CPU
16	7.7546×10^{-5}	-	13.587	7.8530×10^{-5}	-	13.533	7.6592×10^{-5}	-	13.994
32	4.8243×10^{-6}	4.0067	15.856	4.8854×10^{-6}	4.0067	15.493	4.7648×10^{-6}	4.0067	15.609
64	3.0131×10^{-7}	4.0010	19.515	3.0497×10^{-7}	4.0017	19.445	2.9744×10^{-7}	4.0017	19.734
128	1.9031×10^{-8}	3.9848	28.411	1.9111×10^{-8}	3.9962	28.621	1.8807×10^{-8}	3.9833	28.510

Table 10. The E_2 -norm errors and temporal convergence orders with $M = 256, \mu = 1, p = 2$, and $\gamma = 2/\alpha$.

N	$T = 0.5$		$T = 1$		$T = 4$		$T = 10$	
	$E^{SOE}(M, N)$	$Rate_{\tau}^{SOE}$	$E^{SOE}(M, N)$	$Rate_{\tau}^{SOE}$	$E^{SOE}(M, N)$	$Rate_{\tau}^{SOE}$	$E^{SOE}(M, N)$	$Rate_{\tau}^{SOE}$
16	1.3810×10^{-6}	-	3.9181×10^{-6}	-	3.1501×10^{-5}	-	1.2475×10^{-4}	-
32	3.5931×10^{-7}	1.9424	1.0148×10^{-6}	1.9490	8.1468×10^{-6}	1.9511	3.2269×10^{-5}	1.9507
64	9.0207×10^{-8}	1.9939	2.5816×10^{-7}	1.9748	2.0664×10^{-6}	1.9791	8.2114×10^{-6}	1.9745
128	2.3438×10^{-8}	1.9444	6.6561×10^{-8}	1.9555	5.2247×10^{-7}	1.9837	2.0765×10^{-6}	1.9835
256	6.0047×10^{-9}	1.9647	1.5317×10^{-8}	2.1196	1.3334×10^{-7}	1.9703	5.2527×10^{-7}	1.9830

Table 11. The E_2 -norm errors and temporal convergence orders with $M = 256, \mu = 1, p = 2, \alpha = 0.5$, and $\gamma = 2/\alpha$.

N	$\alpha = 0.95$		$\alpha = 0.99$		$\alpha = 0.999$		$\alpha = 0.9999$	
	$E^{SOE}(M, N)$	$Rate_{\tau}^{SOE}$	$E^{SOE}(M, N)$	$Rate_{\tau}^{SOE}$	$E^{SOE}(M, N)$	$Rate_{\tau}^{SOE}$	$E^{SOE}(M, N)$	$Rate_{\tau}^{SOE}$
16	4.0608×10^{-6}	-	3.8864×10^{-6}	-	3.8417×10^{-6}	-	3.8366×10^{-6}	-
32	1.0171×10^{-6}	1.9973	9.7447×10^{-6}	1.9957	9.6418×10^{-7}	1.9944	9.6071×10^{-7}	1.9977
64	2.5443×10^{-7}	1.9992	2.4199×10^{-7}	2.0097	2.4196×10^{-7}	1.9946	2.3883×10^{-7}	2.0081
128	6.5139×10^{-8}	1.9657	6.0890×10^{-8}	1.9907	6.0593×10^{-8}	1.9975	5.8879×10^{-8}	2.0201
256	1.6078×10^{-8}	2.0184	1.5477×10^{-8}	1.9761	1.7048×10^{-8}	1.8295	1.4500×10^{-8}	2.0217

5. Conclusions

In this work, we propose a fast high-order compact scheme for solving the fractional KS equation with the generalized Burgers' type nonlinearity. To deal with the weak singularity at the initial time, the Alikhanov scheme is implemented to approximate the Caputo fractional derivative in the time direction. With the aid of the SOE technique, the computational efficiency is improved and the storage requirement is reduced. In the space direction, the developed compact difference formulas have been successfully applied for approximating the spatial derivatives. As a result, a fully discrete scheme is constructed.

Meanwhile, the stability and convergence with $\tau^2 + h^4$ of the proposed scheme are proved. Finally, the results of the numerical experiments demonstrate that the convergence orders for temporal and spatial convergence achieve second-order and fourth-order accuracies, respectively. This is highly consistent with the theoretical predictions and effectively verifies the theoretical analysis.

Author Contributions: Conceptualization, D.X.; methodology, H.J.; software, H.J.; validation, H.J.; formal analysis, H.J.; investigation, H.J.; re-sources, H.J.; data curation, H.J.; writing—original draft preparation, H.J.; writing—review and editing, D.X.; visu-alization, D.X.; supervision, D.X. ; project administration, D.X. All authors have read and agreed to the published version of the manuscript.

Funding: The work was supported by Scientific Research Fund of Hunan Provincial Education Department (grant number No. 24C0265).

Data Availability Statement: The manuscript has no associated data.

Conflicts of Interest: The authors declare that they have no conflicts of interest.

References

1. Kuramoto, Y.; Tsuzuki, T. On the Formation of Dissipative Structures in Reaction-Diffusion Systems. *Prog. Theor. Phys.* **1975**, *54*, 687–699.
2. Sivashinsky, G.I. Nonlinear Analysis of Hydrodynamic Instability in Laminar Flames-I. Derivation of Basic Equations. *Acta Astronaut.* **1977**, *4*, 1177–1206.
3. Sivashinsky, G.I.; Michelson, D.M. On Irregular Wavy Flow of a Liquid Film Down a Vertical Plane. *Prog. Theor. Phys.* **1980**, *63*, 2112–2114.
4. Sivashinsky, G.I. Instabilities, Pattern-Formation, and Turbulence in Flames. *Ann. Rev. Fluid Mech.* **1983**, *15*, 179–199.
5. Hyman, J.M.; Nicolaenko, B.; Zaleski, S. Order and Complexity in the Kuramoto-Sivashinsky Model of Weakly Turbulent Interfaces. *Phys. D* **1986**, *23*, 265–292.
6. Peng, X.; Qiu, W.; Hendy, A.; Zaky, M. Temporal Second-Order Fast Finite Difference/Compact Difference Schemes for Time-Fractional Generalized Burgers' Equations. *J. Sci. Comput.* **2024**, *99*, 52.
7. Tadmor, E. The Well-Posedness of the Kuramoto-Sivashinsky Equation. *SIAM J. Math. Anal.* **1986**, *17*, 884–893.
8. Zgliczynski, P. Attracting Fixed Points for the Kuramoto-Sivashinsky Equation: A Computer Assisted Proof. *SIAM J. Appl. Dyn. Syst.* **2002**, *1*, 215–235.
9. Sakthivel, R.; Ito, H. Non-Linear Robust Boundary Control of the Kuramoto-Sivashinsky Equation. *IMA J. Math. Control Inf.* **2007**, *24*, 47–55.
10. Khater, A.H.; Tamsah, R.S. Numerical Solutions of the Generalized Kuramoto-Sivashinsky Equation by Chebyshev Spectral Collocation Methods. *Comput. Math. Appl.* **2008**, *56*, 1465–1472.
11. Mittal, R.C.; Arora, G. Quintic B-Spline Collocation Method for Numerical Solution of the Kuramoto-Sivashinsky Equation. *Commun. Nonlinear Sci.* **2010**, *15*, 2798–2808.
12. Xu, Y.; Shu, C.W. Local Discontinuous Galerkin Methods for the Kuramoto-Sivashinsky Equations and the Ito-Type Coupled KdV Equations. *Comput. Methods Appl. Mech. Eng.* **2006**, *195*, 3430–3447.
13. Akrivis, G.; Smyrlis, Y. Implicit—Explicit BDF Methods for the Kuramoto-Sivashinsky Equation. *Appl. Numer. Math.* **2004**, *51*, 151–169.
14. Langlands, T.A.M.; Henry, B.I. The Accuracy and Stability of an Implicit Solution Method for the Fractional Diffusion Equation. *J. Comput. Phys.* **2005**, *205*, 719–736.
15. Sun, Z.Z.; Wu, X.N. A Fully Discrete Difference Scheme for a Diffusion-Wave System. *Appl. Numer. Math.* **2006**, *56*, 193–209.
16. Li, C.; Deng, W.H.; Wu, Y.J. Numerical Analysis and Physical Simulations for the Time Fractional Radial Diffusion Equation. *Comput. Math. Appl.* **2011**, *62*, 1024–1037.
17. Li, L.M.; Xu, D.; Luo, M. Alternating Direction Implicit Galerkin Finite Element Method for the Two-Dimensional Fractional Diffusion-Wave Equation. *J. Comput. Phys.* **2013**, *255*, 471–485.
18. Lin, Y.M.; Xu, C.J. Finite Difference/Spectral Approximations for the Time-Fractional Diffusion Equation. *J. Comput. Phys.* **2007**, *225*, 1533–1552.
19. Qiu, W.L.; Chen, H.B.; Xuan, X. An Implicit Difference Scheme and Algorithm Implementation for the One-Dimensional Time-Fractional Burgers Equations. *Math. Comput. Simulat.* **2019**, *166*, 298–314.
20. Liu, T.; Liu, H.; Ma, Y. Approximate Solution of a Kind of Time-Fractional Evolution Equations Based on Fast L1 Formula and Barycentric Lagrange Interpolation. *Fractal Fract.* **2024**, *8*, 675. [\[CrossRef\]](#)

21. Gao, G.H.; Sun, Z.Z.; Zhang, H.W. A New Fractional Differentiation Formula to Approximate the Caputo Fractional Derivative and Its Applications. *J. Comput. Phys.* **2014**, *259*, 33–50. [\[CrossRef\]](#)
22. Liao, H.L.; Lyu, P.; Vong, S.; Zhao, Y. Stability of Fully Discrete Schemes with Interpolation-Type Fractional Formulas for Distributed-Order Subdiffusion Equations. *Numer. Algor.* **2017**, *75*, 845–878. [\[CrossRef\]](#)
23. Lv, C.W.; Xu, C.J. Error Analysis of a High Order Method for Time-Fractional Diffusion Equations. *SIAM J. Sci. Comput.* **2016**, *38*, 2699–2724. [\[CrossRef\]](#)
24. Alikhanov, A.A. A New Difference Scheme for the Time Fractional Diffusion Equation. *J. Comput. Phys.* **2015**, *280*, 424–438.
25. Liao, H.L.; McLean, W.; Zhang, J.W. A Second-Order Scheme with Nonuniform Time Steps for a Linear Reaction-Subdiffusion Problem. *Commun. Comput. Phys.* **2021**, *30*, 567–601.
26. Zhang, H.X.; Yang, X.H.; Xu, D. An Efficient Spline Collocation Method for a Nonlinear Fourth-Order Reaction Subdiffusion Equation. *J. Sci. Comput.* **2020**, *85*, 7. [\[CrossRef\]](#)
27. Li, X.; Liao, H.L.; Zhang, L.M. A Second-Order Fast Compact Scheme with Unequal Time-Steps for Subdiffusion Problems. *Numer. Algorithms* **2021**, *86*, 1011–1039.
28. Jiang, S.D.; Zhang, J.W.; Zhang, Q.; Zhang, Z.M. Fast Evaluation of the Caputo Fractional Derivative and Its Applications to Fractional Diffusion Equations. *Commun. Comput. Phys.* **2017**, *21*, 650–678. [\[CrossRef\]](#)
29. Lee, S.; Kim, H.; Jang, B. A Novel Numerical Method for Solving Nonlinear Fractional-Order Differential Equations and Its Applications. *Fractal Fract.* **2024**, *8*, 65. [\[CrossRef\]](#)
30. Yang, S.; Yang, X.H. A time two-grid difference method for nonlinear generalized viscous Burgers' equation. *J. Math. Chem.* **2024**, *62*, 1323–1356.
31. Ganaie, I.A.; Arora, S.; Kukreja, V.K. Cubic Hermite Collocation Solution of the Kuramoto–Sivashinsky Equation. *Int. J. Comput. Math.* **2016**, *93*, 223–235. [\[CrossRef\]](#)
32. Al-Khaled, K.; Abu-Irwaq, I. Computational Sinc-Scheme for Extracting Analytical Solution for the Model Kuramoto–Sivashinsky Equation. *Int. J. Electr. Comput. Eng.* **2019**, *9*, 3720–3731.
33. Wang, Y.M.; Zhang, H.B. Higher-Order Compact Finite Difference Method for Systems of Reaction-Diffusion Equations. *J. Comput. Appl. Math.* **2009**, *233*, 502–518.
34. Sari, M.; Gürarslan, G. A Sixth-Order Compact Finite Difference Scheme to the Numerical Solutions of Burgers' Equation. *Appl. Math. Comput.* **2009**, *208*, 475–483.
35. Sutmann, G. Compact Finite Difference Schemes of Sixth Order for the Helmholtz Equation. *J. Comput. Appl. Math.* **2007**, *203*, 15–31.
36. Tian, Z.F.; Liang, X.; Yu, P.X. A Higher Order Compact Finite Difference Algorithm for Solving the Incompressible Navier-Stokes Equations. *Int. J. Numer. Meth. Eng.* **2011**, *88*, 511–532.
37. Kuzenov, V.V.; Ryzhkov, S.V.; Varaksin, A.Y. Development of a Method for Solving Elliptic Differential Equations Based on a Nonlinear Compact-Polynomial Scheme. *J. Comput. Appl. Math.* **2024**, *451*, 116098.
38. Qiu, W.L.; Nikan, O.; Avazzadeh, Z. Numerical Investigation of Generalized Tempered-Type Integrodifferential Equations with Respect to Another Function. *Fract. Calc. Appl. Anal.* **2023**, *60*, 41.
39. Xu, D.; Guo, J.; Qiu, W.L. Time Two-Grid Algorithm Based on Finite Difference Method for Two-Dimensional Nonlinear Fractional Evolution Equations. *Appl. Numer. Math.* **2020**, *152*, 169–184.
40. Qiu, W.L.; Fairweather, G.; Yang, X.H.; Zhang, H.X. ADI Finite Element Galerkin Methods for Two-Dimensional Tempered Fractional Integro-Differential Equations. *Calcolo* **2023**, *60*, 41.
41. Sun, Z.Z. *Numerical Methods of Partial Differential Equations*; Science Press: Beijing, China, 2012.
42. Wang, T.C. Optimal Point-Wise Error Estimate of a Compact Difference Scheme for the Klein–Gordon–Schrödinger Equation. *J. Math. Anal. Appl.* **2014**, *412*, 155–167.
43. Dimitrienko, Y.I.; Li, S.G.; Niu, Y. Study on the Dynamics of a Nonlinear Dispersion Model in Both 1D and 2D Based on the Fourth-Order Compact Conservative Difference Scheme. *Math. Comput. Simulat.* **2020**, *182*, 661–689.
44. Huang, C.B.; Stynes, M. A Sharp α -Robust $L^\infty(H^1)$ Error Bound for a Time-Fractional Allen-Cahn Problem Discretised by the Alikhanov $L2 - 1_\sigma$ Scheme and a Standard FEM. *J. Sci. Comput.* **2022**, *91*, 43.
45. Liao, H.L.; McLean, W.; Zhang, J.W. A Discrete Grönwall Inequality with Applications to Numerical Schemes for Subdiffusion Problems. *SIAM J. Numer. Anal.* **2019**, *57*, 218–237.

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.