SIAM J. NUMER. ANAL.
Vol. 57, No. 6, pp. 2829–2849

A FAST HIGH ORDER METHOD FOR THE TIME-FRACTIONAL DIFFUSION EQUATION*

HONGYI ZHU† AND CHUANJU XU†‡

Abstract. In this paper, we present a fast $(3-\alpha)$-order numerical method for the Caputo fractional derivative based on the L2 scheme and the sum-of-exponentials (SOE) approximation to the convolution kernel involved in the fractional derivative. This work can be regarded as a continuation of previous works reported by one of the authors in [C.W. Lv and C.J. Xu, *SIAM J. Sci. Comput.*, 38 (2016), pp. A2699–A2724], which constructed and analyzed a $(3-\alpha)$-order L2 time stepping scheme for the time-fractional diffusion equation. It is now extended to take into account the fast SOE evaluation method, which allows us to reduce the storage and overall computational cost from $\mathcal{O}(N_T)$ and $\mathcal{O}(N_T^2)$ for the L2 scheme to $\mathcal{O}(N_\varepsilon)$ and $\mathcal{O}(N_T N_\varepsilon)$, respectively, with N_T being the number of time steps and N_ε being the number of fast evaluation terms. The proposed method is then used for the time-fractional diffusion equation in bounded domains. The stability as well as the accuracy of the resulting scheme are rigorously analyzed. Several numerical examples are provided to validate the theoretical results and to demonstrate the efficiency of the proposed method. Finally, we extend the discussion to a graded mesh to make the scheme more suitable for problems having weakly singular solutions at the initial time.

Key words. time-fractional diffusion equation, time stepping scheme, sum-of-exponentials approximation, error estimate

AMS subject classifications. 65M12, 65M06, 65M70, 35S10

DOI. 10.1137/18M1231225

1. Introduction.

Many methods have been developed so far for solving fractional differential equations, in particular the time-fractional diffusion equation (TFDE); see, e.g., [24, 14, 7, 6, 18, 1, 19, 5, 29] for various time discretization methods. Among the existing approaches, the L1 scheme is probably the best known; see, e.g., Sun and Wu [24] and Lin and Xu [14]. The L1 scheme makes use of a piecewise linear approximation and attains a $(2-\alpha)$-order convergence rate for the α-order fractional derivative [14, 7, 18]. Since then, attempts have been made to design and analyze high order schemes. Based on piecewise linear approximation at the closest interval to the time point and piecewise quadratic interpolation at the previous time intervals, Alikhanov [1] constructed an L2-1_σ scheme and proved it is second order accurate when applied to the TFDE. Lv and Xu [19] developed a $(3-\alpha)$-order scheme by using piecewise quadratic interpolation, which will be referred to as the L2 scheme hereafter. A rigorous proof of the convergence order was given in that paper. It is noted that other types of $(3-\alpha)$-order schemes can also be found; see, e.g., [6]. While the aforementioned work has focused on the accuracy of numerical solutions, other works address the storage reduction issue. It is well known that due to the nonlocality of the fractional derivatives, traditional approaches to them unavoidably result in high storage cost. More precisely, the above-mentioned schemes require $\mathcal{O}(N_T)$

*Received by the editors December 7, 2018; accepted for publication (in revised form) September 4, 2019; published electronically December 5, 2019.
https://doi.org/10.1137/18M1231225

Funding: The work of the authors was partially supported by the National Natural Science Foundation of China grants 11971408, 51661135011, 11421110001, and 91630204.

†School of Mathematical Sciences and Fujian Provincial Key Laboratory of Mathematical Modeling and High Performance Scientific Computing, Xiamen University, 361005 Xiamen, China (hyzhu@stu.xmu.edu.cn, cjxu@xmu.edu.cn).
‡Corresponding author.

storage and $\mathcal{O}(N_T^2)$ computational costs for N_T time steps, which is too expensive, especially in the case of several spatial variables. To overcome this difficulty, several authors have proposed approximating the weakly singular kernel function by a sum-of-exponentials (SOE), resulting in some new time stepping schemes with reduced storage. The work in this direction seems to have been inspired by Lubich [16, 17], who expressed the kernel function by the integral of its inverse Laplace transform on a suitable contour and approximated the integral term by quadrature. Baffet and Hesthaven [2, 3] followed this idea and used the multipole expansion to approximate the kernel function. Jiang et al. [9, 30] represented the kernel $t^{-\alpha-1}$ by an integral on a semi-infinite interval and employed the Gauss-type quadratures to approximate the resulting integral. They combined this approximation with the L1 scheme to derive a fast memory-saving scheme. A similar technique was used to accelerate the L2-1_σ scheme in Yan, Sun, and Zhang [26]. Some related work includes Zeng, Turner, and Burrage [28] on Gauss–Laguerre approximation to fractional integral operators, McLean [20] on degenerate kernel approximation, and Lu, Pang, and Sun [15] and Ke, Ng, and Sun [11] on approximation methods based on the block triangular Toeplitz-like matrix.

This paper aims at constructing a high order fast evaluation scheme to discretize the fractional derivative. Precisely, we focus on improving the L2 scheme and seek a $(3-\alpha)$-order scheme with reduced memory and provable stability. Starting with the traditional L2 scheme, we propose making use of the SOE approximation to the kernel function. This leads to a modified L2 scheme having reduced storage and calculation cost while preserving the same convergence order. Precisely, compared to the L2 scheme, the modified scheme reduces the storage and computational cost from $\mathcal{O}(N_T)$ and $\mathcal{O}(N_T^2)$ to $\mathcal{O}(N_\varepsilon)$ and $\mathcal{O}(N_T N_\varepsilon)$, respectively, where N_ε stands for the number of SOE approximations. In this paper we (1) introduce in section 2 a fast finite difference operator for the Caputo fractional derivative; (2) derive a sharp estimate for the approximation error of the fast finite difference operator in section 3, construct a cheaper $(3-\alpha)$-order scheme based on the L2 formula and SOE approximation for the TFDE in a bounded domain, and prove the unconditional stability and convergence order for the full discrete solution in the framework of the spectral collocation method for the spatial discretization; (3) discuss the issue of how high order convergence can be achieved for nonsmooth solutions—an interesting point since the solution of the TFDE is known to be nonsmooth with respect to the time variable for smooth data (right-hand side function, boundary, and boundary condition, etc.); and (4) extend in section 4 the cheaper high order scheme to suitable graded meshes so that the high order accuracy is attainable for typical solutions of the TFDE, which are weakly singular at the starting time.

2. Fast finite difference operator based on the L2 scheme. In this section, we present a high order fast evaluation scheme for the Caputo fractional derivative, defined by

$$\partial_t^\alpha u(t) = \frac{1}{\Gamma(1-\alpha)} \int_0^t \frac{\partial_s u(s)}{(t-s)^\alpha} ds, \quad 0 < \alpha < 1, \tag{2.1}$$

in the interval $I = (0, T]$ over a set grid points $t_k := k\Delta t,\ k = 0, 1, \ldots, N_T$, with the time step $\Delta t := \frac{T}{N_T}$. Let $I_k = [t_{k-1}, t_k], k = 1, \ldots, N_T$.

2.1. The L2 scheme. We briefly recall the L2 scheme, which was constructed and analyzed in [19]. The scheme makes use of the linear interpolation of the function

u at the first time step and quadratic interpolation from the second one. Precisely, we first define the quadratic interpolation operator $\Pi_{2,j}$, $j=1,\dots,N_T-1$, by

$$\Pi_{2,j}u(s) = u(t_{j-1})\frac{(s-t_j)(s-t_{j+1})}{2\Delta t^2} - u(t_j)\frac{(s-t_{j-1})(s-t_{j+1})}{\Delta t^2} + u(t_{j+1})\frac{(s-t_{j-1})(s-t_j)}{2\Delta t^2},$$
$$s\in[t_{j-1},t_{j+1}],$$

which is a polynomial of degree 2 interpolating u at the points $\{t_{j-1},t_j,t_{j+1}\}$. Then we define the finite difference operators:

(2.2)
$$\begin{aligned}
L_t^\alpha u(t_1) &= \frac{1}{\nu_\alpha}[u(t_1)-u(t_0)],\\
L_t^\alpha u(t_k) &= \frac{1}{\Gamma(1-\alpha)}\Big[\int_{t_{k-1}}^{t_k}\frac{\partial_s\Pi_{2,k-1}u(s)}{(t_k-s)^\alpha}ds + \sum_{j=1}^{k-1}\int_{t_{j-1}}^{t_j}\frac{\partial_s\Pi_{2,j}u(s)}{(t_k-s)^\alpha}ds\Big]\\
&= \frac{1}{\Gamma(3-\alpha)\Delta t^\alpha}\Big\{\frac{\alpha}{2}u(t_{k-2}) - 2u(t_{k-1}) + \frac{4-\alpha}{2}u(t_k)\\
&\quad+\sum_{j=1}^{k-1}\big[a_ju(t_{k-j-1})+b_ju(t_{k-j})+c_ju(t_{k-j+1})\big]\Big\},\quad 2\le k\le N_T,
\end{aligned}$$

where $\nu_\alpha=\Gamma(2-\alpha)\Delta t^{2-\alpha}$,

$$\begin{aligned}
a_j &= -\frac{3}{2}(2-\alpha)(j+1)^{1-\alpha}+\frac{1}{2}(2-\alpha)j^{1-\alpha}+(j+1)^{2-\alpha}-j^{2-\alpha},\\
b_j &= 2(2-\alpha)(j+1)^{1-\alpha}-2(j+1)^{2-\alpha}+2j^{2-\alpha},\\
c_j &= -\frac{1}{2}(2-\alpha)\big((j+1)^{1-\alpha}+j^{1-\alpha}\big)+(j+1)^{2-\alpha}-j^{2-\alpha}.
\end{aligned}\tag{2.3}$$

The scheme (2.2) can be rewritten under a more compact form:

$$L_t^\alpha u(t_k) = \frac{1}{\kappa_\alpha p_\alpha^{-1}}\Big[u(t_k)-\sum_{j=1}^{k}d_{k-j}^{k,\alpha}u(t_{k-j})\Big],\quad 2\le k\le N_T,\tag{2.4}$$

where

$$\kappa_\alpha=\Gamma(3-\alpha)\Delta t^\alpha,\quad p_\alpha=c_1+2-\frac{\alpha}{2},\tag{2.5}$$

(2.6)
$$\begin{aligned}
&d_1^{2,\alpha}=-(b_1-2)p_\alpha^{-1},\quad d_0^{2,\alpha}=\Big(-a_1-\frac{\alpha}{2}\Big)p_\alpha^{-1};\\
&d_2^{3,\alpha}=-(b_1+c_2-2)p_\alpha^{-1},\quad d_1^{3,\alpha}=\Big(-a_1-b_2-\frac{\alpha}{2}\Big)p_\alpha^{-1},\quad d_0^{3,\alpha}=-a_2p_\alpha^{-1};\\
\text{for }\ k\ge 4,\quad &d_{k-1}^{k,\alpha}=-(b_1+c_2-2)p_\alpha^{-1},\quad d_{k-2}^{k,\alpha}=\Big(-a_1-b_2-c_3-\frac{\alpha}{2}\Big)p_\alpha^{-1};\\
&d_{k-j}^{k,\alpha}=(-a_{j-1}-b_j-c_{j+1})p_\alpha^{-1},\quad j=3,\dots,k-2;\\
&d_1^{k,\alpha}=(-a_{k-2}-b_{k-1})p_\alpha^{-1},\quad d_0^{k,\alpha}=-a_{k-1}p_\alpha^{-1}.
\end{aligned}$$

It has been proved [19] that the truncation error, defined by

$$r_{\Delta t}^{k,\alpha}:=\partial_t^\alpha u(t_k)-L_t^\alpha u(t_k),\tag{2.7}$$

has the following error bounds:

$$|r_{\Delta t}^{1,\alpha}|\le c_\alpha\widetilde{M}(u)\Delta t^{2-\alpha},\qquad |r_{\Delta t}^{k,\alpha}|\le c_\alpha M(u)\Delta t^{3-\alpha},\ k=2,\dots,N_T,\tag{2.8}$$

where c_α depends on α, $\widetilde{M}(u)=\max_{t\in I_1}|\partial_t^2u(t)|$, $M(u)=\max_{t\in[0,t_k]}|\partial_t^3u(t)|$.

2.2. Construction of the fast finite difference operator. Note that the right-hand side of (2.4) involves a sum of all previous solutions $\{u(t_j)\}_{j=0}^{k}$, which reflects the memory effect of the nonlocal fractional operator. This makes both the computation and storage expensive. In order to overcome this difficulty, we propose a further approach to the fractional derivative, which makes use of an idea stemming from [2, 3, 9]. The idea consists in first splitting the Caputo derivative into a local term $L(t_k)$ and a history term $H(t_k)$, defined respectively by

(2.9)
$$L(t_k) := \frac{1}{\Gamma(1-\alpha)} \int_{t_{k-1}}^{t_k} \frac{\partial_s u(s)}{(t_k - s)^\alpha} ds, \qquad H(t_k) := \frac{1}{\Gamma(1-\alpha)} \int_0^{t_{k-1}} \frac{\partial_s u(s)}{(t_k - s)^\alpha} ds.$$

The local term is approximated as follows:

(2.10)
$$L(t_k) \approx \frac{1}{\Gamma(1-\alpha)} \int_{t_{k-1}}^{t_k} \frac{\partial_s \Pi_{2,k-1} u(s)}{(t_k - s)^\alpha} ds = \frac{1}{\kappa_\alpha} \Big[\frac{\alpha}{2} u(t_{k-2}) - 2u(t_{k-1}) + \frac{4-\alpha}{2} u(t_k) \Big].$$

For the history term $H(t_k)$, using integration by parts gives

$$H(t_k) = \frac{1}{\Gamma(1-\alpha)} \Big[\frac{u(t_{k-1})}{\Delta t^\alpha} - \frac{u(t_0)}{t_k^\alpha} - \alpha \int_0^{t_{k-1}} \frac{u(s)ds}{(t_k - s)^{1+\alpha}} \Big]. \tag{2.11}$$

It is known that there exists an approximation to the last term in (2.11) using the fact that the power function $t^{-1-\alpha}, \forall \alpha \in (0,1)$, can be approximated by an SOE in the interval $[\Delta t, T]$; see, e.g., [9]. Precisely, for any prescribed error $\varepsilon > 0$, there exist positive real numbers s_i^α and $\omega_i^\alpha, i = 1, \ldots, N_\varepsilon$, such that

$$\left| \frac{1}{t^{1+\alpha}} - \sum_{i=1}^{N_\varepsilon} \omega_i^\alpha e^{-s_i^\alpha t} \right| \le \varepsilon \quad \forall t \in [\Delta t, T], \tag{2.12}$$

where

$$N_\varepsilon = \mathcal{O}\bigg(\log \frac{1}{\varepsilon} \bigg(\log\log \frac{1}{\varepsilon} + \log \frac{T}{\Delta t} \bigg) + \log \frac{1}{\Delta t} \bigg(\log\log \frac{1}{\varepsilon} + \log \frac{1}{\Delta t} \bigg) \bigg). \tag{2.13}$$

Roughly, for fixed ε, $N_\varepsilon = \mathcal{O}(\log(N_T))$ if $T \gg 1$; $N_\varepsilon = \mathcal{O}(\log^2(N_T))$ if $T \approx 1$.

Approximating the kernel $(t_k - s)^{-\alpha-1}$ in (2.11) by $\sum_{i=1}^{N_\varepsilon} \omega_i^\alpha e^{-s_i^\alpha (t_k - s)}$ gives

$$H(t_k) \approx \frac{1}{\Gamma(1-\alpha)} \left[\frac{u(t_{k-1})}{\Delta t^\alpha} - \frac{u(t_0)}{t_k^\alpha} - \alpha \sum_{i=1}^{N_\varepsilon} \omega_i^\alpha U_{h,i}^\alpha(t_k) \right], \tag{2.14}$$

where

$$U_{h,i}^\alpha(t_k) = \int_0^{t_{k-1}} e^{-s_i^\alpha (t_k - s)} u(s) ds. \tag{2.15}$$

The key for the success of the new approach lies in the fact that there exists the recurrence

$$U_{h,i}^\alpha(t_k) = e^{-s_i^\alpha \Delta t} U_{h,i}^\alpha(t_{k-1}) + \int_{t_{k-2}}^{t_{k-1}} e^{-s_i^\alpha (t_k - s)} u(s) ds \tag{2.16}$$

to evaluate $U_{h,i}^\alpha(t_k)$ without the need to perform the integration over the whole interval $(0, t_{k-1})$ in (2.15). That is, thanks to the recurrence (2.16), the calculation

$\int_0^{t_{k-1}} e^{-s_i^\alpha(t_k-s)}u(s)ds$ is reduced to $\int_{t_{k-2}}^{t_{k-1}} e^{-s_i^\alpha(t_k-s)}u(s)ds$, which can be computed by using the quadratic interpolation $\Pi_{2,k-1}u(s)$ of u as follows:

(2.17)
$$\int_{t_{k-2}}^{t_{k-1}} e^{-s_i^\alpha(t_k-s)}u(s)ds \approx \int_{t_{k-2}}^{t_{k-1}} e^{-s_i^\alpha(t_k-s)}\Pi_{2,k-1}u(s)ds = \hat{a}_i u(t_{k-2}) - \hat{b}_i u(t_{k-1}) + \hat{c}_i u(t_k),$$

where

$$\begin{aligned} \hat{a}_i &= \frac{e^{-2s_i^\alpha \Delta t}}{2s_i^\alpha (s_i^\alpha \Delta t)^2}\big[2e^{s_i^\alpha \Delta t} - 2(s_i^\alpha \Delta t)^2 - s_i^\alpha \Delta t(3 - e^{s_i^\alpha \Delta t}) - 2\big], \\ \hat{b}_i &= \frac{e^{-2s_i^\alpha \Delta t}}{s_i^\alpha (s_i^\alpha \Delta t)^2}\big[2e^{s_i^\alpha \Delta t} - 2s_i^\alpha \Delta t - (s_i^\alpha \Delta t)^2 e^{s_i^\alpha \Delta t} - 2\big], \\ \hat{c}_i &= \frac{e^{-2s_i^\alpha \Delta t}}{2s_i^\alpha (s_i^\alpha \Delta t)^2}\big[2e^{s_i^\alpha \Delta t} - s_i^\alpha \Delta t(1 + e^{s_i^\alpha \Delta t}) - 2\big]. \end{aligned} \tag{2.18}$$

We now introduce the finite difference operator F_t^α: for the discrete function $\{v^k\}_{k=0}^{N_T}$, define

$$\begin{aligned} F_t^\alpha v^1 &= L_t^\alpha v^1, \\ F_t^\alpha v^k &= \frac{1}{\kappa_\alpha}\Big(\frac{\alpha}{2}v^{k-2} - 2v^{k-1} + \frac{4-\alpha}{2}v^k\Big) \\ &\quad + \frac{1}{\Gamma(1-\alpha)}\Big(\frac{v^{k-1}}{\Delta t^\alpha} - \frac{v^0}{t_k^\alpha} - \alpha\sum_{i=1}^{N_\varepsilon}\omega_i^\alpha U_{h,i}^{k,\alpha}\Big), \quad 2 \le k \le N_T, \end{aligned} \tag{2.19}$$

where

(2.20)
$$U_{h,i}^{1,\alpha} = 0, \quad U_{h,i}^{k,\alpha} = e^{-s_i^\alpha \Delta t}U_{h,i}^{k-1,\alpha} + \hat{a}_i v^{k-2} - \hat{b}_i v^{k-1} + \hat{c}_i v^k,\ i = 1, \dots, N_\varepsilon.$$

It is clearly seen from (2.10)–(2.17) that $F_t^\alpha u(t_k)$ is an approximation to $\partial_t^\alpha u(t_k)$. Comparing $L_t^\alpha u(t_k)$ in (2.2) with $F_t^\alpha u(t_k)$ in (2.19), we see that both are approximations to $\partial_t^\alpha u(t_k)$; the former requires all the previous time step values of u, while the latter only needs $u(t_k)$, $u(t_{k-1})$, $u(t_{k-2})$, $u(t_0)$, and $U_{h,i}^\alpha(t_k)$, $i = 1, \dots, N_\varepsilon$. This means that approximating $\partial_t^\alpha u(t_k)$ by $F_t^\alpha u(t_k)$ considerably reduces the storage and computational costs as compared to $L_t^\alpha u(t_k)$, especially in the case of long time integration. Roughly replacing $L_t^\alpha u(t_k)$ by $F_t^\alpha u(t_k)$ allows us to reduce the storage cost from $\mathcal{O}(N_T)$ to $\mathcal{O}(N_\varepsilon)$ and the computational cost from $\mathcal{O}(N_T^2)$ to $\mathcal{O}(N_T N_\varepsilon)$, where N_ε is the SOE approximation number defined in (2.13).

The main purpose of this work is to use the above-defined finite difference operator $F_t^\alpha u(t_k)$ to construct efficient schemes for solving the time-fractional diffusion equation, and prove the stability and convergence of the constructed schemes.

3. Fast scheme for the fractional diffusion equation in bounded domains.

We consider the time-fractional diffusion equation as follows:

$$\partial_t^\alpha u(x,t) - \partial_x^2 u(x,t) = f(x,t), \quad x \in \Lambda,\ t \in I, \tag{3.1}$$

subject to the following initial and boundary conditions:

$$u(x,0) = u_0(x), \quad x \in \Lambda, \tag{3.2}$$
$$u(-1,t) = u(1,t) = 0, \quad 0 \le t \le T, \tag{3.3}$$

where $\Lambda = (-1, 1)$, and u_0 is the initial condition. We now consider the following time semidiscrete scheme:

$$\begin{cases} F_t^\alpha u^k - \partial_x^2 u^k = f(t_k),\ 1 \le k \le N_T, \\ u^0 = u_0, \end{cases} \tag{3.4}$$

where the finite difference operator F_t^α is as defined in (2.19). In the next subsection, we will analyze the stability and convergence of this scheme in the absence of the source term f, which are stated in Theorems 3.1 and 3.2.

3.1. Stability and convergence of the semidiscrete problem. We first establish the stability estimate for the scheme (3.4). Let $L^2(\Lambda), H^1(\Lambda),\ H_0^1(\Lambda)$ be usual Sobolev spaces endowed with standard inner products and norms. We rewrite (3.4) without the source term f in the following weak form: given $\{u^j\}_{j=0}^{k-1}$, find $u^k \in H_0^1(\Lambda)$, $1 \le k \le N_T$, such that

$$(F_t^\alpha u^k, v) + (\partial_x u^k, \partial_x v) = 0 \quad \forall v \in H_0^1(\Lambda), \tag{3.5}$$

where $(\cdot,\cdot)$ is the usual L^2–inner product.

To prove the stability of (3.5), we need some preparations, which are presented through a number of lemmas.

First, by comparing the two operators F_t^α and L_t^α, we have

$$\begin{aligned} F_t^\alpha u(t_k) - L_t^\alpha u(t_k) &= \frac{\alpha}{\Gamma(1-\alpha)} \sum_{j=1}^{k-1} \int_{t_{j-1}}^{t_j} \Big[(t_k - s)^{-1-\alpha} - \sum_{i=1}^{N_\varepsilon} \omega_i^\alpha e^{-s_i^\alpha (t_k - s)}\Big] \Pi_{2,j} u(s) ds \\ &= \frac{1}{\kappa_\alpha q_\alpha^{-1}} \sum_{j=1}^{k-1} \big[\tilde{a}_{k-j} u(t_{j-1}) + \tilde{b}_{k-j} u(t_j) + \tilde{c}_{k-j} u(t_{j+1})\big], \end{aligned} \tag{3.6}$$

where

$$q_\alpha = \alpha(1-\alpha)(2-\alpha), \tag{3.7}$$

and the coefficients $\tilde{a}_{k-j}$, $\tilde{b}_{k-j}$, and $\tilde{c}_{k-j}$ are defined by

$$\tilde{a}_{k-j} = \Delta t^\alpha \int_{t_{j-1}}^{t_j} \Big[(t_k - s)^{-1-\alpha} - \sum_{i=1}^{N_\varepsilon} \omega_i^\alpha e^{-s_i^\alpha (t_k - s)}\Big] \frac{(s - t_j)(s - t_{j+1})}{2\Delta t^2} ds, \tag{3.8}$$

$$\tilde{b}_{k-j} = -\Delta t^\alpha \int_{t_{j-1}}^{t_j} \Big[(t_k - s)^{-1-\alpha} - \sum_{i=1}^{N_\varepsilon} \omega_i^\alpha e^{-s_i^\alpha (t_k - s)}\Big] \frac{(s - t_{j-1})(s - t_{j+1})}{\Delta t^2} ds, \tag{3.9}$$

$$\tilde{c}_{k-j} = \Delta t^\alpha \int_{t_{j-1}}^{t_j} \Big[(t_k - s)^{-1-\alpha} - \sum_{i=1}^{N_\varepsilon} \omega_i^\alpha e^{-s_i^\alpha (t_k - s)}\Big] \frac{(s - t_{j-1})(s - t_j)}{2\Delta t^2} ds. \tag{3.10}$$

It can be checked using (2.12) that the coefficients $\tilde{a}_{k-j}$, $\tilde{b}_{k-j}$, $\tilde{c}_{k-j}$, $1 \le j \le k-1, 2 \le k \le N_T$ defined in (3.8)–(3.10) have bounds as follows:

$$|\tilde{a}_{k-j}| \le \frac{5}{12} \varepsilon \Delta t^{\alpha+1}, \quad |\tilde{b}_{k-j}| \le \frac{2}{3} \varepsilon \Delta t^{\alpha+1}, \quad |\tilde{c}_{k-j}| \le \frac{1}{12} \varepsilon \Delta t^{\alpha+1}. \tag{3.11}$$

Furthermore, combining (2.4) and (3.6) gives

$$F_t^\alpha u^k = \frac{1}{\kappa_\alpha p_\alpha^{-1}} \Big[u^k - \sum_{j=1}^{k} d_{k-j}^{k,\alpha} u^{k-j}\Big] + \frac{1}{\kappa_\alpha q_\alpha^{-1}} \sum_{j=1}^{k-1} \big[\tilde{a}_{k-j} u^{j-1} + \tilde{b}_{k-j} u^j + \tilde{c}_{k-j} u^{j+1}\big], \tag{3.12}$$

in which the first part corresponds to the traditional L2 scheme, while the second part comes from the SOE approximation to the convolution. It is convenient to rewrite (3.12) in the following compact form:

$$F_t^\alpha u^k = \frac{1}{\kappa_\alpha p_\alpha^{-1}}\Big[u^k - \sum_{j=1}^{k} d_{k-j}^{k,\alpha} u^{k-j}\Big] + \frac{1}{\kappa_\alpha q_\alpha^{-1}} \sum_{j=0}^{k} \xi_{k-j}^{k,\alpha} u^{k-j}, \tag{3.13}$$

where
(3.14)

$$\begin{aligned}
&\xi_2^{2,\alpha} = \tilde{c}_1,\ \xi_1^{2,\alpha} = \tilde{b}_1,\ \xi_0^{2,\alpha} = \tilde{a}_1;\\
&\xi_3^{3,\alpha} = \tilde{c}_1,\ \xi_2^{3,\alpha} = \tilde{b}_1 + \tilde{c}_2,\ \xi_1^{3,\alpha} = \tilde{a}_1 + \tilde{b}_2,\ \xi_0^{3,\alpha} = \tilde{a}_2;\\
\text{for } k \geq 4,\quad &\xi_k^{k,\alpha} = \tilde{c}_1,\quad \xi_{k-1}^{k,\alpha} = \tilde{b}_1 + \tilde{c}_2,\quad \xi_{k-j}^{k,\alpha} = \tilde{a}_{j-1} + \tilde{b}_j + \tilde{c}_{j+1},\ j = 2, \dots, k-2;\\
&\xi_1^{k,\alpha} = \tilde{a}_{k-2} + \tilde{b}_{k-1},\quad \xi_0^{k,\alpha} = \tilde{a}_{k-1}.
\end{aligned}$$

Some properties of the coefficients $d_{k-j}^{k,\alpha}$ in the L2 scheme have been given in [19]. Here we need some more to establish the stability of the fast scheme, which is provided in the next lemma.

LEMMA 3.1. *For any* $0 < \alpha < 1$, $k \geq 3$, $n \geq k$, *the coefficients* $d_{k-j}^{k,\alpha}$ *and* p_α *in* (3.13) *satisfy*
(1) $0 < d_{k-1}^{k,\alpha} < \frac{4}{3}$; (2) $-\frac{1}{2} < d_{k-2}^{k,\alpha} < \frac{1}{3}$; (3) $d_{k-j}^{k,\alpha} > 0, j = 3, \dots, k$; (4) $\sum_{j=1}^{k} d_{k-j}^{k,\alpha} = 1$;
(5) $p_\alpha \in (\frac{3}{2}, 2)$; (6) $\sum_{k=2}^{n} d_0^{k,\alpha} \leq (2-\alpha)(n-1)^{1-\alpha}$; (7) $\sum_{k=2}^{n} d_1^{k,\alpha} \leq \frac{4}{3} + (2-\alpha)(n-2)^{1-\alpha}$;
(8) $\sum_{k=j+1}^{n} d_j^{k,\alpha} \leq 1,\ j = 2, \dots, n-2.$

Proof. See Appendix A. ∎

Following the technique used in [19], we reformulate the L2 part in (3.13) by introducing a parameter η to be determined:
(3.15)

$$\begin{aligned}
u^k - \sum_{j=1}^{k} d_{k-j}^{k,\alpha} u^{k-j} &= (u^k - \eta u^{k-1}) - (d_{k-1}^{k,\alpha} - \eta)(u^{k-1} - \eta u^{k-2}) - \cdots\\
&- (d_2^{k,\alpha} + \eta d_3^{k,\alpha} + \cdots + \eta^{k-3} d_{k-1}^{k,\alpha} - \eta^{k-2})(u^2 - \eta u^1)\\
&- (d_1^{k,\alpha} + \eta d_2^{k,\alpha} + \eta^2 d_3^{k,\alpha} + \cdots + \eta^{k-2} d_{k-1}^{k,\alpha} - \eta^{k-1})(u^1 - \eta u^0)\\
&- (d_0^{k,\alpha} + \eta d_1^{k,\alpha} + \eta^2 d_2^{k,\alpha} + \eta^3 d_3^{k,\alpha} + \cdots + \eta^{k-1} d_{k-1}^{k,\alpha} - \eta^k) u^0\\
&= \overline{u}^k - \sum_{j=1}^{k} \overline{d}_{k-j}^{k,\alpha} \overline{u}^{k-j},
\end{aligned}$$

where η^j means power of η, and

$$\overline{u}^0 = u^0,\quad \overline{u}^j = u^j - \eta u^{j-1},\quad \overline{d}_{k-j}^{k,\alpha} = \sum_{l=1}^{j} \eta^{j-l} d_{k-l}^{k,\alpha} - \eta^j,\quad j = 1, \dots, k. \tag{3.16}$$

The SOE approximation part in (3.13) can be reformulated in a similar way:

$$\frac{1}{\kappa_\alpha q_\alpha^{-1}} \sum_{j=0}^{k} \xi_{k-j}^{k,\alpha} u^{k-j} = \frac{1}{\kappa_\alpha q_\alpha^{-1}} \sum_{j=0}^{k} \overline{\xi}_{k-j}^{k,\alpha} \overline{u}^{k-j},$$

where

$$\overline{\xi}_{k-j}^{k,\alpha} = \sum_{l=0}^{j} \eta^{j-l} \xi_{k-l}^{k,\alpha}, \quad j = 0, \dots, k. \tag{3.17}$$

Thus, $F_t^\alpha u^k$ in (3.13) can be equivalently written as

$$\overline{F}_t^\alpha \overline{u}^k = F_t^\alpha u^k := \frac{1}{\kappa_\alpha p_\alpha^{-1}} \Big[\overline{u}^k - \sum_{j=1}^{k} \overline{d}_{k-j}^{k,\alpha} \overline{u}^{k-j} \Big] + \frac{1}{\kappa_\alpha q_\alpha^{-1}} \sum_{j=0}^{k} \overline{\xi}_{k-j}^{k,\alpha} \overline{u}^{k-j}, \quad 2 \le k \le N_T. \tag{3.18}$$

Consequently, the problem (3.5) is equivalent to the following: find $u^k \in H_0^1(\Lambda), \overline{u}^j = u^j - \eta u^{j-1}, j = 1, \dots, k$, such that

$$(\overline{F}_t^\alpha \overline{u}^k, v) + (\partial_x u^k, \partial_x v) = 0 \quad \forall v \in H_0^1(\Lambda). \tag{3.19}$$

The new coefficients of $\overline{F}_t^\alpha$ in (3.18) possess better properties, as shown in the next lemma.

LEMMA 3.2. *Let $0 < \alpha < 1$, $\eta = \frac{1}{2}\big[3 - \frac{\alpha+4}{\alpha+2}(\frac{3}{2})^{1-\alpha}\big]$. Then the coefficients $\overline{d}_{k-j}^{k,\alpha}$ and $\overline{\xi}_{k-j}^{k,\alpha}$ in (3.18) satisfy the following: for $k \ge 2$, $n \ge k$,*

(1) $\overline{d}_{k-j}^{k,\alpha} > 0$, $j = 1, \dots, k$; (2) $\dfrac{1}{\overline{d}_0^{k,\alpha}} < \dfrac{1}{d_0^{k,\alpha}} < \dfrac{2k^\alpha}{(1-\alpha)(2-\alpha)}$; (3) $\sum_{j=1}^{k} \overline{d}_{k-j}^{k,\alpha} \le 1 - \frac{\eta(1-\alpha)(2-\alpha)}{2(1-\eta)k^\alpha}$;

(4) $|\overline{\xi}_k^{k,\alpha}| \le \dfrac{1}{12}\varepsilon \Delta t^{\alpha+1}$; (5) $|\overline{\xi}_{k-j}^{k,\alpha}| \le \dfrac{7}{2}\varepsilon \Delta t^{\alpha+1}$, $j = 1, \dots, k$.

Proof. See Appendix A. ∎

Remark 3.1. It can be verified by using (2.3) and (2.6) that the value of η taken in Lemma 3.2 is $\frac{1}{2} d_{k-1}^{k,\alpha}, k \ge 3$. However, the choice of η is not unique. Indeed, the results in Lemma 3.2 remain true for any η such that $\frac{d_{k-1}^{k,\alpha} - \sqrt{(d_{k-1}^{k,\alpha})^2 + 4 d_{k-2}^{k,\alpha}}}{2} < \eta < \frac{d_{k-1}^{k,\alpha} + \sqrt{(d_{k-1}^{k,\alpha})^2 + 4 d_{k-2}^{k,\alpha}}}{2}$, but the value taken in Lemma 3.2 makes the coefficient $\overline{d}_{k-2}^{k,\alpha}$ maximal.

THEOREM 3.1. *If the prescribed error ε given in (2.12) is small enough, i.e., satisfies (3.25) and (3.26), then the semidiscrete problem (3.5) is unconditionally stable, and its solution satisfies the following estimate for any $\Delta t > 0$:*

$$\|u^k\|_0 + \sqrt{\kappa_\alpha p_\alpha^{-1}}\; \|\partial_x u^k\|_0 \le 4 \|u^0\|_0, \quad 1 \le k \le N_T, \tag{3.20}$$

where κ_α and p_α are as given in (2.5).

Proof. (1) For $k = 1$, taking $v = u^1$ in (3.5) and noting that $\kappa_\alpha p_\alpha^{-1} \le \nu_\alpha$, we immediately get (3.20).

(2) For $k \geq 2$, we first prove by induction the following result:

$$\|\overline{u}^k\|_0^2 + \kappa_\alpha p_\alpha^{-1}\|\partial_x u^k\|_0^2 \leq \|u^0\|_0^2, \quad 2 \leq k \leq N_T. \tag{3.21}$$

Setting $v = 2\overline{u}^k$ in (3.19), we obtain

$$2(\overline{u}^k, \overline{u}^k) + 2\kappa_\alpha p_\alpha^{-1}(\partial_x u^k, \partial_x \overline{u}^k) + 2p_\alpha^{-1} q_\alpha \sum_{j=0}^{k} \overline{\xi}_{k-j}^{k,\alpha}(\overline{u}^{k-j}, \overline{u}^k) = 2\sum_{j=1}^{k} \overline{d}_{k-j}^{k,\alpha}(\overline{u}^{k-j}, \overline{u}^k). \tag{3.22}$$

Using the identity $2(\partial_x u^k, \partial_x \overline{u}^k) = \|\partial_x u^k\|_0^2 + \|\partial_x \overline{u}^k\|_0^2 - \eta^2\|\partial_x u^{k-1}\|_0^2$, the Schwarz inequality, and Lemma 3.2(3), we get

$$\begin{aligned} &\|\overline{u}^k\|_0^2 + \kappa_\alpha p_\alpha^{-1}\|\partial_x u^k\|_0^2 + \kappa_\alpha p_\alpha^{-1}\|\partial_x \overline{u}^k\|_0^2 \\ \leq\ & \eta\|\overline{u}^{k-1}\|_0^2 + \kappa_\alpha p_\alpha^{-1}\eta\|\partial_x u^{k-1}\|_0^2 + \sum_{j=2}^{k} \overline{d}_{k-j}^{k,\alpha}\|\overline{u}^{k-j}\|_0^2 - 2p_\alpha^{-1} q_\alpha \sum_{j=0}^{k} \overline{\xi}_{k-j}^{k,\alpha}(\overline{u}^{k-j}, \overline{u}^k). \end{aligned} \tag{3.23}$$

Now the key is to find the right way to control the last term on the right-hand side. Using Lemma 3.1(5), Lemma 3.2(4), and the Poincaré inequality, we have

$$\left| -2q_\alpha \overline{\xi}_k^{k,\alpha}(\overline{u}^k, \overline{u}^k)\right| \leq \frac{1}{6} q_\alpha \varepsilon \Delta t^{\alpha+1}\|\overline{u}^k\|_0^2 \leq \frac{1}{2}\kappa_\alpha\|\partial_x \overline{u}^k\|_0^2 \tag{3.24}$$

if ε satisfies

$$\varepsilon \leq \frac{3\Gamma(1-\alpha)}{2\alpha\Delta t}. \tag{3.25}$$

Furthermore, setting $\theta_k = 1 - \sum_{j=1}^{k} \overline{d}_{k-j}^{k,\alpha}$, it follows from Young's inequality that

$$\left| -2q_\alpha \sum_{j=1}^{k} \overline{\xi}_{k-j}^{k,\alpha}(\overline{u}^{k-j}, \overline{u}^k)\right| \leq \frac{\theta_k}{k}\sum_{j=1}^{k}\|\overline{u}^{k-j}\|_0^2 + \frac{kq_\alpha^2}{\theta_k}\sum_{j=1}^{k}(\overline{\xi}_{k-j}^{k,\alpha})^2\|\overline{u}^k\|_0^2.$$

By virtue of Lemma 3.2(3) and 3.2(5), it is verified that $\theta_k \geq \frac{\eta(1-\alpha)(2-\alpha)}{2(1-\eta)k^\alpha}$. If we furthermore assume the condition

$$\varepsilon \leq \frac{1}{7\alpha}\sqrt{\frac{\eta\Gamma(1-\alpha)}{2(1-\eta)T^{\alpha+2}}}, \tag{3.26}$$

then we have

$$\begin{aligned} &\left| -2q_\alpha \sum_{j=1}^{k} \overline{\xi}_{k-j}^{k,\alpha}(\overline{u}^{k-j}, \overline{u}^k)\right| \\ \leq\ & \frac{\theta_k}{k}\sum_{j=1}^{k}\|\overline{u}^{k-j}\|_0^2 + \frac{4(1-\eta)k^{\alpha+2}q_\alpha^2}{\eta(1-\alpha)(2-\alpha)}\left[\frac{7}{2}\varepsilon\Delta t^{\alpha+1}\right]^2\|\partial_x \overline{u}^k\|_0^2 \\ \leq\ & \frac{\theta_k}{k}\sum_{j=1}^{k}\|\overline{u}^{k-j}\|_0^2 + \frac{1}{2}\kappa_\alpha\|\partial_x \overline{u}^k\|_0^2. \end{aligned} \tag{3.27}$$

Combining (3.24) and (3.27) together, and plugging the result into (3.23), we get

$$\|\overline{u}^k\|_0^2 + \kappa_\alpha p_\alpha^{-1}\|\partial_x u^k\|_0^2 \leq \left(\eta + \frac{\theta_k}{k}\right)\|\overline{u}^{k-1}\|_0^2 + \kappa_\alpha p_\alpha^{-1}\eta\|\partial_x u^{k-1}\|_0^2 + \sum_{j=2}^{k}\left(\overline{d}_{k-j}^{k,\alpha} + \frac{\theta_k}{k}\right)\|\overline{u}^{k-j}\|_0^2. \tag{3.28}$$

Using the fact that $\eta+\theta_2+\overline{d}_0^{2,\alpha}<1$, we obtain (3.21) for $k=2$.

Assuming (3.21) is true for all $k=2,3,\dots,n-1$, we deduce from (3.28)

$$\|\overline{u}^n\|_0^2+\kappa_\alpha p_\alpha^{-1}\|\partial_x u^n\|_0^2 \le \Big[\eta+\frac{\theta_n}{n}+\sum_{j=2}^{n}\overline{d}_{n-j}^{n,\alpha}+\frac{\theta_n}{n}(n-1)\Big]\|u^0\|_0^2=\|u^0\|_0^2 \ \ \forall n=2,\dots,N_T.$$

This proves (3.21) for $k=n$, and thus for all $2\le k\le N_T$.

(3) Finally, recall that $\overline{u}^k=u^k-\eta u^{k-1}$; it follows from (3.21) that

$$\begin{aligned}\|u^k\|_0&=\|\overline{u}^k+\eta u^{k-1}\|_0\le\|u^0\|_0+\eta\|u^{k-1}\|_0\le(1+\eta)\|u^0\|_0+\eta^2\|u^{k-2}\|_0\\&\le(1+\eta+\eta^2+\cdots+\eta^k)\|u^0\|_0\le 3\|u^0\|_0,\quad 2\le k\le N_T.\end{aligned}$$

Combining the above estimate with (3.21) gives (3.20). The proof is completed. ∎

Remark 3.2. The condition on ε, i.e., (3.25) and (3.26), is not a real restriction since in most applications ε is taken as small as machine accuracy. This is confirmed by Example 5.1, Figure 5.3, in section 5.

Now we turn to perform the error analysis for the semidiscrete problem (3.5). We first estimate the truncation error of the finite difference operator F_t^α, which is defined by

$$R_{\Delta t}^{k,\alpha}:=\partial_t^\alpha u(t_k)-F_t^\alpha u(t_k),\quad 2\le k\le N_T. \tag{3.29}$$

Lemma 3.3. *Suppose that $u\in C^3[0,T]$; then for any $\alpha\in(0,1)$, it holds that*

$$|R_{\Delta t}^{k,\alpha}|\le c_\alpha\max_{t\in[0,t_k]}|\partial_t^3u(t)|\Delta t^{3-\alpha}+\max_{t\in[0,t_k]}|u(t)|\frac{7\alpha t_{k-1}}{6\Gamma(1-\alpha)}\varepsilon,\quad k=2,\dots,N_T. \tag{3.30}$$

Proof. We split $R_{\Delta t}^{k,\alpha}$ into $\partial_t^\alpha u(t_k)-L_t^\alpha u(t_k)$ and $L_t^\alpha u(t_k)-F_t^\alpha u(t_k)$. By (3.6), we have

$$R_{\Delta t}^{k,\alpha}=r_{\Delta t}^{k,\alpha}-\frac{1}{\kappa_\alpha q_\alpha^{-1}}\sum_{j=1}^{k-1}\big[\tilde{a}_{k-j}u(t_{j-1})+\tilde{b}_{k-j}u(t_j)+\tilde{c}_{k-j}u(t_{j+1})\big], \tag{3.31}$$

where $r_{\Delta t}^{k,\alpha}$ is as given in (2.7). Then it follows from (2.8) and (3.11) that

$$\begin{aligned}|R_{\Delta t}^{k,\alpha}|&\le c_\alpha\max_{t\in[0,t_k]}|\partial_t^3u(t)|\Delta t^{3-\alpha}+\frac{\varepsilon\Delta t^{\alpha+1}}{\kappa_\alpha q_\alpha^{-1}}\sum_{j=1}^{k-1}\Big[\frac{5}{12}|u(t_{j-1})|+\frac{2}{3}|u(t_j)|+\frac{1}{12}|u(t_{j+1})|\Big]\\&\le c_\alpha\max_{t\in[0,t_k]}|\partial_t^3u(t)|\Delta t^{3-\alpha}+\frac{7\alpha\varepsilon t_{k-1}}{6\Gamma(1-\alpha)}\max_{t\in[0,t_k]}|u(t)|.\end{aligned}$$

This proves the lemma. ∎

It can be easily checked that the error at the first time step is only of second order. In order to obtain global high accuracy, we can follow the idea in [19], which consists in constructing a substepping scheme for the first step. We omit the details of this technique here and only give the error estimate starting with the second step under the assumption that the error at the first time step has the desired accuracy.

Theorem 3.2. *Let u be the exact solution of (3.1)–(3.3), and let $\{u^k\}_{k=1}^{N_T}$ be the semidiscrete solution of (3.5) with the initial $u^0(x)=u(x,0)$. Suppose $\partial_t^3u\in L^\infty((0,T];L^2(\Lambda))$; then the following error estimate holds for $2\le k\le N_T$:*

$$\|u(t_k)-u^k\|_0+\sqrt{\kappa_\alpha p_\alpha^{-1}}\ \|\partial_x(u(t_k)-u^k)\|_0\le c_{\alpha,T}(\Delta t^{3-\alpha}\|\partial_t^3u\|_{L^\infty(L^2)}+\varepsilon\|u\|_{L^\infty(L^2)}),$$

where $c_{\alpha,T}$ depends only on α and T.

Proof. Let $e^k = u(t_k) - u^k$. Combining (3.1), (3.4), and (3.29) gives the error equation for $k \geq 2$,

$$(F_t^\alpha e^k, v) + (\partial_x e^k, \partial_x v) = (R_{\Delta t}^{k,\alpha}, v) \quad \forall v \in H_0^1(\Lambda). \tag{3.32}$$

By following the reformulation technique in (3.15)–(3.19) and the same lines in Theorem 3.1, and making use of Lemma 3.2(2), we obtain

$$\|\bar{e}^k\|_0^2 + \kappa_\alpha p_\alpha^{-1} \|\partial_x e^k\|_0^2 \leq 2\|e^0\|_0^2 + \frac{24 T^\alpha \Gamma(1-\alpha)}{\kappa_\alpha} \max_{2 \leq i \leq N_T} \|R_{\Delta t}^{i;\alpha}\|_0^2, \quad k \geq 2, \tag{3.33}$$

where $\bar{e}^k = u(t_k) - \bar{u}^k$. Then we apply Lemma 3.3 to conclude. ∎

3.2. Spectral collocation discretization in space and error estimates. Let $\mathbb{P}_N(\Lambda)$ be the space of all polynomials of degree less than or equal to N, $\mathbb{P}_N^0(\Lambda) = H_0^1(\Lambda) \cap \mathbb{P}_N(\Lambda)$. $L_N(x)$ denotes the Legendre polynomial of degree N, and $\{\xi_j\}_{j=0}^N$ and $\{\omega_j\}_{j=0}^N$ are the Legendre–Gauss–Lobatto (LGL) points and weights, respectively. The discrete inner product is defined by $(\phi, \psi)_N := \sum_{i=0}^N \phi(\xi_i)\psi(\xi_i)\omega_i$, and $\|\phi\|_N := (\phi, \phi)_N^{1/2}$ is the associated discrete norm. We use $\pi_N^{1,0}$ to denote the H_0^1-orthogonal projection operator from $H_0^1(\Lambda)$ into $\mathbb{P}_N^0(\Lambda)$. The following results are well known (see, e.g., [4, 21]):

$$\|\varphi\|_0 \leq \|\varphi\|_N \leq \sqrt{3}\|\varphi\|_0 \quad \forall \varphi \in \mathbb{P}_N(\Lambda). \tag{3.34}$$

$$|(\varphi, v_N) - (\varphi, v_N)_N| \leq cN^{-m}\|\varphi\|_m \|v_N\|_0 \quad \forall \varphi \in H^m(\Lambda),\ \forall v_N \in \mathbb{P}_N(\Lambda),\ m \geq 1. \tag{3.35}$$

$$\|\varphi - \pi_N^{1,0}\varphi\|_l \leq cN^{l-m}\|\varphi\|_m \quad \forall \varphi \in H^m(\Lambda) \cap H_0^1(\Lambda),\ m \geq 1,\ l = 0, 1. \tag{3.36}$$

We consider the spectral collocation method in space as follows: find $u_N^k \in \mathbb{P}_N^0(\Lambda)$, such that

$$(F_t^\alpha u_N^k, v_N)_N + (\partial_x u_N^k, \partial_x v_N)_N = 0 \quad \forall v_N \in \mathbb{P}_N^0(\Lambda). \tag{3.37}$$

The error analysis for this full discrete problem is given in the following theorem.

THEOREM 3.3. *Let u be the exact solution of (3.1)–(3.3), and let $\{u_N^k\}_{k=1}^{N_T}$ be the solution of problem (3.37) with the initial condition $u_N^0 = \pi_N^{1,0}u^0$. Suppose $\partial_t^3 u \in L^\infty((0,T]; H^m(\Lambda))$, $m \geq 1$; then for $k = 2, \dots, N_T$ it holds that*

$$\begin{aligned} &\|u(t_k) - u_N^k\|_0 + \sqrt{\kappa_\alpha p_\alpha^{-1}}\ \|\partial_x(u(t_k) - u_N^k)\|_0 \\ \leq\ & c_{\alpha,T}\big(\Delta t^{3-\alpha}\|\partial_t^3 u\|_{L^\infty(L^2)} + N^{-m}\Delta t^{3-\alpha}\|\partial_t^3 u\|_{L^\infty(H^m)} + N^{-m}\|\partial_t^\alpha u\|_{L^\infty(H^m)} \\ & + N^{1-m}\|u\|_{L^\infty(H^m)} + \varepsilon\|u\|_{L^\infty(L^2)} + \varepsilon N^{-m}\|u\|_{L^\infty(H^m)}\big). \end{aligned} \tag{3.38}$$

Proof. Let $e_N^k = u_N^k - \pi_N^{1,0}u(t_k)$. A straightforward calculation using (3.37) gives

$$\begin{aligned} & (F_t^\alpha e_N^k, v_N)_N + (\partial_x e_N^k, \partial_x v_N)_N \\ = \ & (F_t^\alpha u_N^k, v_N)_N + (\partial_x u_N^k, \partial_x v_N)_N - (F_t^\alpha \pi_N^{1,0}u(t_k), v_N)_N - (\partial_x \pi_N^{1,0}u(t_k), \partial_x v_N)_N \\ = \ & (F_t^\alpha u(t_k) - F_t^\alpha \pi_N^{1,0}u(t_k), v_N)_N - (F_t^\alpha u(t_k), v_N)_N - (\partial_x \pi_N^{1,0}u(t_k), \partial_x v_N)_N \\ =: \ & \varepsilon_1^k(v_N) + \varepsilon_2^k(v_N) \quad \forall v_N \in \mathbb{P}_N^0(\Lambda), \end{aligned} \tag{3.39}$$

where

$$\varepsilon_1^k(v_N) = (F_t^\alpha u(t_k) - F_t^\alpha \pi_N^{1,0} u(t_k), v_N)_N, \tag{3.40}$$

$$\varepsilon_2^k(v_N) = -\big(F_t^\alpha u(t_k), v_N\big)_N - (\partial_x \pi_N^{1,0} u(t_k), \partial_x v_N)_N. \tag{3.41}$$

By virtue of (3.18), the equation (3.39) can be reformulated as

$$(\overline{F}_t^\alpha \overline{e}_N^k, v_N)_N + (\partial_x e_N^k, \partial_x v_N)_N = \varepsilon_1^k(v_N) + \varepsilon_2^k(v_N) \quad \forall v_N \in \mathbb{P}_N^0(\Lambda),$$

where $\overline{e}_N^k := e_N^k - \eta e_N^{k-1}$.

Now we estimate $\varepsilon_1^k(v_N)$ and $\varepsilon_2^k(v_N)$ one by one. Using (3.29) and (3.35), we have

$$\begin{aligned}
|\varepsilon_1^k(v_N)| &= \big|((I_d - \pi_N^{1,0})(\partial_t^\alpha u(t_k) - R_{\Delta t}^{k,\alpha}), v_N)_N\big| \\
&\le \big|((I_d - \pi_N^{1,0})(\partial_t^\alpha u(t_k) - R_{\Delta t}^{k,\alpha}), v_N)\big| + cN^{-1}\big\|(I_d - \pi_N^{1,0})(\partial_t^\alpha u(t_k) - R_{\Delta t}^{k,\alpha})\big\|_1 \|v_N\|_0 \\
&\le \Big[\big\|(I_d - \pi_N^{1,0})(\partial_t^\alpha u(t_k) - R_{\Delta t}^{k,\alpha})\big\|_0 + cN^{-1}\big\|(I_d - \pi_N^{1,0})(\partial_t^\alpha u(t_k) - R_{\Delta t}^{k,\alpha})\big\|_1\Big] \|v_N\|_0.
\end{aligned} \tag{3.42}$$

Furthermore, it follows from Lemma 3.3 that, for $l = 0, 1$,

$$\|(I_d - \pi_N^{1,0}) R_{\Delta t}^{k,\alpha}\|_l \le c_\alpha \max_{t\in I} \|(I_d - \pi_N^{1,0})\partial_t^3 u(\cdot,t)\|_l \Delta t^{3-\alpha} + \varepsilon c_{\alpha,T} \max_{t\in I} \|(I_d - \pi_N^{1,0}) u(\cdot,t)\|_l. \tag{3.43}$$

Then applying (3.36) to (3.42) and (3.43) yields

$$|\varepsilon_1^k(v_N)| \le c_{\alpha,T}(N^{-m}\|\partial_t^\alpha u\|_{L^\infty(H^m)} + N^{-m}\Delta t^{3-\alpha}\|\partial_t^3 u\|_{L^\infty(H^m)} + \varepsilon N^{-m}\|u\|_{L^\infty(H^m)})\|v_N\|_0. \tag{3.44}$$

On the other side, according to the definition of the discrete norm $\|\cdot\|_N$ and the projection operator $\pi_N^{1,0}$, we have

$$(\partial_x \pi_N^{1,0} u(t_k), \partial_x v_N)_N = (\partial_x \pi_N^{1,0} u(t_k), \partial_x v_N) = (\partial_x u(t_k), \partial_x v_N) \ \forall v_N \in \mathbb{P}_N^0(\Lambda).$$

Using this result and (3.1) in (3.41), we obtain

$$\varepsilon_2^k(v_N) = -\big(F_t^\alpha u(t_k), v_N\big)_N - (\partial_x u(t_k), \partial_x v_N) = (F_t^\alpha u(t_k), v_N) - \big(F_t^\alpha u(t_k), v_N\big)_N + (R_{\Delta t}^{k,\alpha}, v_N).$$

Then applying (3.35) and Lemma 3.3 gives

$$\begin{aligned}
\big|\varepsilon_2^k(v_N)\big| &\le \big(cN^{-m}\|\partial_t^\alpha u(t_k) - R_{\Delta t}^{k,\alpha}\|_m + \|R_{\Delta t}^{k,\alpha}\|_0\big)\|v_N\|_0 \\
&\le c_{\alpha,T}\big(N^{-m}\|\partial_t^\alpha u\|_{L^\infty(H^m)} + N^{-m}\Delta t^{3-\alpha}\|\partial_t^3 u\|_{L^\infty(H^m)} + \varepsilon N^{-m}\|u\|_{L^\infty(H^m)} \\
&\quad + \Delta t^{3-\alpha}\|\partial_t^3 u\|_{L^\infty(L^2)} + \varepsilon\|u\|_{L^\infty(L^2)}\big)\|v_N\|_0.
\end{aligned} \tag{3.45}$$

Combining (3.44) and (3.45), together with the Poincaré inequality, we get

$$\begin{aligned}
\big|\varepsilon_1^k(v_N)\big| + \big|\varepsilon_2^k(v_N)\big| &\le c_{\alpha,T}\big(N^{-m}\|\partial_t^\alpha u\|_{L^\infty(H^m)} + N^{-m}\Delta t^{3-\alpha}\|\partial_t^3 u\|_{L^\infty(H^m)} \\
&\quad + \varepsilon N^{-m}\|u\|_{L^\infty(H^m)} + \Delta t^{3-\alpha}\|\partial_t^3 u\|_{L^\infty(L^2)} + \varepsilon\|u\|_{L^\infty(L^2)}\big)\|\partial_x v_N\|_0.
\end{aligned}$$

Proceeding analogously to the proof of Theorem 3.1, we subsequently have

$$\begin{aligned}
\|e_N^k\|_0 + \sqrt{\kappa_\alpha p_\alpha^{-1}}\ \|\partial_x e_N^k\|_0 \le\ & c_{\alpha,T}\big(N^{-m}\|\partial_t^\alpha u\|_{L^\infty(H^m)} + N^{-m}\Delta t^{3-\alpha}\|\partial_t^3 u\|_{L^\infty(H^m)} \\
& + \varepsilon N^{-m}\|u\|_{L^\infty(H^m)} + \Delta t^{3-\alpha}\|\partial_t^3 u\|_{L^\infty(L^2)} + \varepsilon\|u\|_{L^\infty(L^2)}\big).
\end{aligned}$$

Finally, we conclude with help of the triangle inequality and (3.36). □

4. Extension to nonuniform mesh for nonsmooth solutions. The singularity issue of the TFDE has been subject of much research, both from a numerical and a theoretical viewpoint. It has been known that the solution to the TFDE may exhibit a weak singularity near $t = 0$ for a general smooth force function, i.e., the time derivatives of the exact solution are constrained by negative power of t; see, e.g., [22, 10, 23]. Recently, Hou and Xu [8] proposed a general framework using fractional polynomials for some weakly singular integro-differential equations and fractional differential equations. It was proved that the exponential convergence rate can be achieved for solutions which are smooth after the variable change $t \to t^{1/\lambda}$ for suitable λ. Some other authors have proposed using nonuniform graded mesh to capture the starting point singularity; see Liao, Li, and Zhang [12] and Stynes, O'Riordan, and Gracia [23] for an L1 scheme on the graded mesh and Liao, McLean, and Zhang [13] for a refined L2-1_σ scheme. It is also worth noting that there exist some other techniques, such as the starting steps correction method in Yan et al. [5, 25, 27], to capture the singularity near the origin.

Here we seek to construct an accelerated L2 scheme on the graded mesh for (3.1)–(3.3). Let us consider the graded mesh $t_k := T(\frac{k}{N_T})^r$, $k = 0, 1, \dots, N_T$, with the mesh grading constant $r \geq 1$. Note that the uniform mesh corresponds to $r = 1$. Denote the time step size $\Delta t_k := t_k - t_{k-1}$, $k = 1, 2, \dots, N_T$. Following the same idea as for the the uniform mesh (see (2.19)), the graded mesh version of the fast L2 finite difference operator, denoted by $F^\alpha_{t,G}$, reads as follows: for $2 \leq k \leq N_T$,

(4.1)
$$F^\alpha_{t,G} v^k = \frac{1}{\kappa^1_\alpha}\big(\beta_k v^{k-2} - \delta_k v^{k-1} + \eta_k v^k\big) + \frac{1}{\Gamma(1-\alpha)}\Big(\frac{v^{k-1}}{\Delta t^\alpha_k} - \frac{v^0}{t^\alpha_k} - \alpha\sum_{i=1}^{N_\varepsilon} \omega^\alpha_i \hat{U}^{k,\alpha}_{h,i}\Big),$$

where

$$\kappa^1_\alpha = \Gamma(3-\alpha)\Delta t^\alpha_1, \quad \beta_k = \frac{\alpha[k^r - (k-1)^r]^{2-\alpha}}{[(k-1)^r - (k-2)^r][k^r - (k-2)^r]},$$
$$\delta_k = \frac{\alpha k^r + (2-2\alpha)(k-1)^r - (2-\alpha)(k-2)^r}{[(k-1)^r - (k-2)^r][k^r - (k-1)^r]^\alpha}, \quad \eta_k = \frac{2k^r - \alpha(k-1)^r - (2-\alpha)(k-2)^r}{[k^r - (k-1)^r]^\alpha [k^r - (k-2)^r]},$$
$$\hat{U}^{1,\alpha}_{h,i} = 0, \quad \hat{U}^{k,\alpha}_{h,i} = e^{-s^\alpha_i \Delta t_k}\hat{U}^{k-1,\alpha}_{h,i} + \hat{a}^k_i v^{k-2} - \hat{b}^k_i v^{k-1} + \hat{c}^k_i v^k,$$
$$\hat{a}^k_i = \frac{e^{-s^\alpha_i \Delta t_k}}{\Delta t_{k-1}(\Delta t_{k-1} + \Delta t_k)(s^\alpha_i)^3}\Big\{s^\alpha_i \Delta t_k + 2 - e^{-s^\alpha_i \Delta t_{k-1}}\big[(s^\alpha_i)^2 \Delta t_{k-1}(\Delta t_{k-1} + \Delta t_k) + s^\alpha_i(2\Delta t_{k-1} + \Delta t_k) + 2\big]\Big\},$$
$$\hat{b}^k_i = \frac{e^{-s^\alpha_i \Delta t_k}}{\Delta t_{k-1}\Delta t_k (s^\alpha_i)^3}\Big[s^\alpha_i(\Delta t_k - \Delta t_{k-1}) - (s^\alpha_i)^2 \Delta t_{k-1}\Delta t_k + 2 - e^{-s^\alpha_i \Delta t_{k-1}}\big(s^\alpha_i(\Delta t_k + \Delta t_{k-1}) + 2\big)\Big],$$
$$\hat{c}^k_i = \frac{e^{-s^\alpha_i \Delta t_k}}{\Delta t_k(\Delta t_{k-1} + \Delta t_k)(s^\alpha_i)^3}\Big[-s^\alpha_i \Delta t_{k-1} + 2 - e^{-s^\alpha_i \Delta t_{k-1}}\big(-s^\alpha_i \Delta t_{k-1} + 2\big)\Big], \quad i = 1, \dots, N_\varepsilon.$$

It can be deduced from an analysis similar that in Lemma 3.3 and [23, Lemma 5.2] that the following truncation error estimate holds for the graded mesh operator $F^\alpha_{t,G}$:

$$|\partial^\alpha_t u(t_k) - F^\alpha_{t,G} u(t_k)| \leq c\big(k^{-\min\{3-\alpha, r\alpha\}} + \varepsilon\big), \quad k = 2, \dots, N_T, \tag{4.2}$$

provided that the exact solution u satisfies

$$|\partial^l_t u(t)| < c(1 + t^{\alpha - l}) \quad \text{for } t \in (0, T],\ l = 0, 1, 2, 3. \tag{4.3}$$

The above error estimate suggests use of the grading constant $r = (3-\alpha)/\alpha$, which makes the scheme possess maximal convergence order, i.e., $3 - \alpha$. In Example 5.2 in the next section we will numerically investigate the impact of the grading constant on the accuracy.

Unfortunately, for the time being we are unable to offer a proof for the stability of the graded mesh scheme. This is due to the complexity of the coefficients in the scheme. In particular, a difficulty in establishing the stability comes from the fact that the negative coefficient can't be bounded in a small range as the grading constant r varies.

It is also worth pointing out that the assumption (4.3) is reasonable. In fact, Stynes, O'Riordan, and Gracia [23] proved that (4.3) holds for $l = 0, 1, 2$ provided $u_0(x) \in D(\mathcal{L}^{5/2})$, $f(\cdot, t) \in D(\mathcal{L}^{5/2})$, $f_t(\cdot, t)$ and $f_{tt}(\cdot, t)$ in $D(\mathcal{L}^{1/2})$ with $\|f(\cdot, t)\|_{\mathcal{L}^{5/2}} + \|f_t(\cdot, t)\|_{\mathcal{L}^{1/2}} + t^\rho \|f_{tt}(\cdot, t)\|_{\mathcal{L}^{1/2}} \le C$ for $t \in (0, T]$ and some constant $\rho < 1$, where the operator $\mathcal{L} = -\Delta$,

$$D(\mathcal{L}^\theta) = \Big\{ g \in L^2(\Lambda) : \|g\|_{\mathcal{L}^\theta} < \infty \Big\}, \ \|g\|_{\mathcal{L}^\theta} = \Big(\sum_{i=1}^{\infty} \lambda_i^{2\theta} |(g, \psi_i)|^2 \Big)^{1/2} \quad \forall \theta \in \mathbb{R},$$

and $\{(\lambda_j, \psi_j)\}_{j=1}^\infty$ are the eigenvalues and L^2-orthonormal eigenfunctions of $\mathcal{L}$ associated with the Dirichlet boundary condition. Inequality (4.3) for $l = 3$ can also be proved by following same lines as in [23] under an additional assumption that $t^\rho \|f_{ttt}(\cdot, t)\|_{\mathcal{L}^{1/2}}$ is bounded for some constant $\rho < 1$.

5. Numerical results.

In this section, we present some numerical examples to verify the theoretical result. We first give some implementation details. By the definition of F_t^α, i.e., (2.19), the fully discrete problem (3.37) with the forcing f added can be rewritten as

$$p(u_N^k, v_N)_N + \kappa_\alpha(\partial_x u_N^k, \partial_x v_N)_N = \mathcal{F}_N(v_N), \tag{5.1}$$

where

$$\mathcal{F}_N(v_N) = \sum_{i=1}^{2} p_i(u_N^{k-i}, v_N)_N + p_0(u_N^0, v_N)_N + \sum_{i=1}^{N_\varepsilon} q_i(U_{h,i}^{k-1,\alpha}, v_N)_N + \kappa_\alpha(f(\cdot, t_k), v_N)_N.$$

The coefficients p, p_1, p_2, and q_j are defined as follows:

$$p = \frac{4-\alpha}{2} - q_\alpha \Delta t^\alpha \sum_{i=1}^{N_\varepsilon} \omega_i^\alpha \hat{c}_i, \quad p_0 = \frac{(1-\alpha)(2-\alpha)}{k^\alpha}, \quad p_1 = 2 - \frac{q_\alpha}{\alpha} - q_\alpha \Delta t^\alpha \sum_{i=1}^{N_\varepsilon} \omega_i^\alpha \hat{b}_i,$$
$$p_2 = -\frac{\alpha}{2} + q_\alpha \Delta t^\alpha \sum_{i=1}^{N_\varepsilon} \omega_i^\alpha \hat{a}_i, \quad q_j = q_\alpha \Delta t^\alpha \omega_j^\alpha e^{-s_j^\alpha \Delta t}, \ j = 1, \dots, N_\varepsilon.$$

Then by using the Lagrangian polynomials $\{h_j\}_{j=0}^N$ based on the LGL points as the basis functions of $\mathbb{P}_N(\Lambda)$, we arrive at the following linear system at each time step:

$$(pB + \kappa_\alpha A)\underline{u}^k = \underline{f}, \tag{5.2}$$

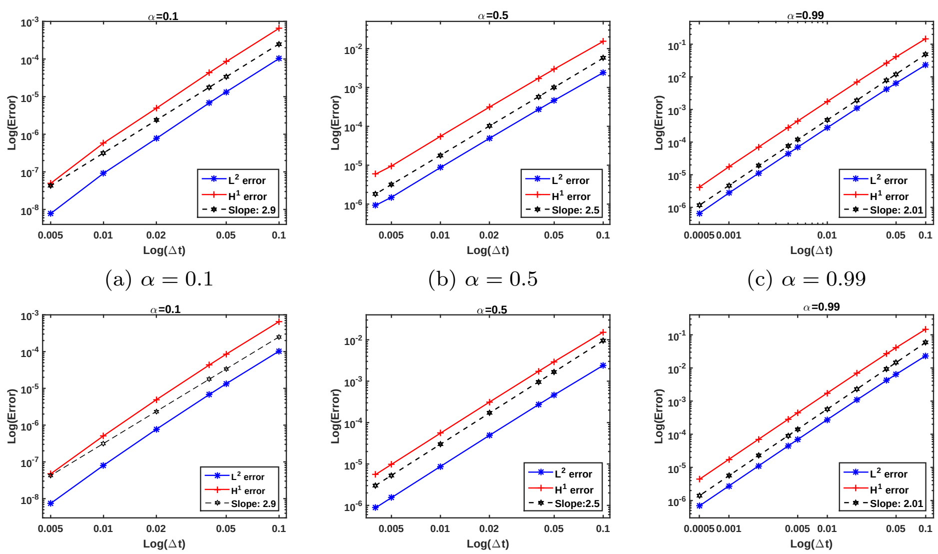
where B is the mass matrix with the entries $B_{ij} := \omega_i \delta_{ij}, i, j = 1, \dots, N-1$, A is the stiffness matrix having the entries $A_{ij} := \sum_{q=0}^N D_{qi} D_{qj} \omega_q, \ D_{ij} := h_j'(\xi_i), \ i, j = 1, \dots, N-1$, $\underline{u}^k$ is the nodal unknown vector $(u_N^k(\xi_j))_{j=1}^{N-1}$, and $\underline{f}$ is the right-hand side vector $(\mathcal{F}_N(h_j))_{j=1}^{N-1}$. In our calculation, the conventional conjugate gradient method is used to solve the linear system (5.2), which is symmetric positive definite.

As mentioned in section 3, in order to achieve the overall high accuracy, we will employ a substepping scheme in the first step in all numerical examples that follow. The substep size used in the calculation is $\Delta t_s = \frac{\Delta t}{n_s}$, where $n_s = \lceil \frac{\Delta t}{\Delta t^{2-\alpha}} \rceil$; see [19] for more details.

Example 5.1 (smooth solution). We consider the problem (3.1) with a fabricated forcing function f such that the exact solution is $u(x,t) = e^{3t}\cos(2\pi x + \frac{\pi}{2})$. Set $\Lambda = (0,1)$, $T = 1$, and $\varepsilon = 10^{-10}$. Notice that when the space domain Λ is not the reference domain $(-1,1)$, it should be rescaled to apply the spectral method described in section 3.

The purpose of this example is to demonstrate the accuracy and the efficiency of the proposed fast L2 scheme in term of the convergence rate and CPU time. For purposes of comparison, the results obtained by using the standard L2 scheme are also presented. All calculations are done on a computer with Intel Dual-Core i7-5500U using MATLAB.

We first investigate the convergence rate of the time stepping scheme. Figure 5.1 shows the errors in the H^1 and L^2 norms as a function of the time step size in log-log scale, obtained from the fast scheme and standard L2 scheme, respectively. The polynomial degree N is taken to be 24, large enough such that the spatial discretization does not pollute the temporal error. It is observed that the error decay rates for the two schemes are almost the same, i.e., $3-\alpha$ order with respect to the time step size for the three tested values of α. This confirms that the SOE approximation to the convolution in the L2 scheme does not lower the convergence order.



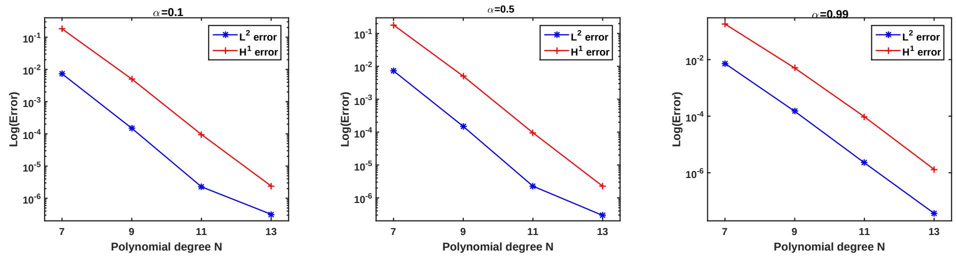
(a) $\alpha = 0.1$ (b) $\alpha = 0.5$ (c) $\alpha = 0.99$

(d) $\alpha = 0.1$ (e) $\alpha = 0.5$ (f) $\alpha = 0.99$

FIG. 5.1. *Comparison of the convergence rates for the fast L2 (top) and standard L2 (bottom) schemes.*

Then we turn to investigate the spatial error behavior. In Figure 5.2, we plot the errors in semilog scale versus the polynomial degree N for the proposed fast L2 scheme, where the time step size is fixed to $\Delta t = 10^{-4}$. As expected for spectral approximations to smooth solutions, the straight error lines in these semilog plots indicate that the numerical solutions are exponentially convergent when the polynomial degree increases.

Finally, we compare the computational costs of the two schemes. The main advantage of using the accelerated L2 scheme is the reduction of the computational complexity as compared to the original L2 scheme. We plot in Figure 5.3 the CPU



(a) $\alpha = 0.1$ (b) $\alpha = 0.5$ (c) $\alpha = 0.99$

FIG. 5.2. *Errors versus the polynomial degree N for the fast L2 scheme.*

time with respect to the total number of time steps N_T for both schemes with $N = 8$. We observe that the CPU time of the fast scheme increases linearly with respect to N_T, while the cost for the original L2 scheme increases almost quadratically. This observation is in a very good agreement with the analysis presented at the end of the second section. The influence of the prescribed error ε on the SOE number N_ε and CPU time is also investigated. We present in the right figure of Figure 5.3 the computation time and N_ε of the accelerated L2 scheme as functions of ε in the range from 10^{-5} to 10^{-10}. We see from this figure that N_ε is nearly a linear function of $\log \varepsilon$, and the computation time remains almost constant as ε varies. This is due to the fact that the SOE summation cost is negligible as compared to the overall cost, especially when N_T is large. This observation suggests using ε as small as possible in practical applications.

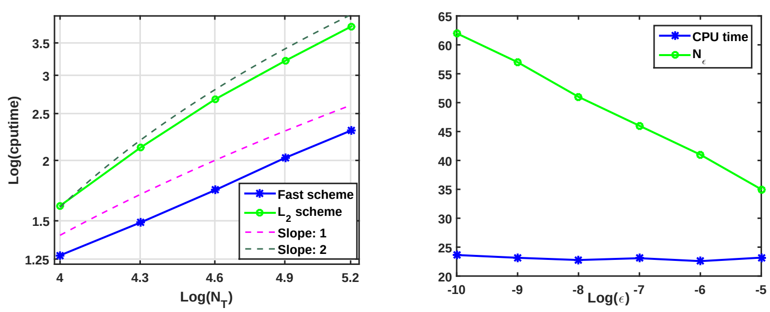


FIG. 5.3. *(Left) CPU time in seconds as a function of time steps N_T. (Right) CPU time and SOE number N_ε versus the prescribed error ε.*

Example 5.2 (singular solution). Consider the nonsmooth solution $u = (t^\alpha + t^{1+\alpha} + t^{2+\alpha}) \sin x$ in $(0, \pi) \times (0, 1)$, which is typical for the time-fractional diffusion equation (3.1) with general smooth forcing. This example is used to show that the schemes constructed on the uniform mesh may fail to produce numerical solutions of desired accuracy, while the use of the graded mesh helps in recovering the convergence order.

We first compute the numerical solution by using the fast L2 scheme (2.19) in uniform meshes. Table 1 displays the maximum L^2 errors with $\varepsilon = 10^{-7}$ for the SOE approximation precision and $N = 20$ for the polynomial degree in the space discretization. It is observed that the numerical solution has only α-order accuracy.

This is not surprising since the proved convergence order in Theorem 3.2 is only valid for solutions having the regularity $\partial_t^3 u \in L^\infty$, while the regularity of the exact solution tested in this example is not more than C^1 in time.

TABLE 1
Maximum L^2 error and convergence rate of the scheme (2.19) *for the solution $u = (t^\alpha + t^{1+\alpha} + t^{2+\alpha}) \sin x$.*

N_T	$\alpha = 0.4$		$\alpha = 0.6$		$\alpha = 0.8$	
	Error	Order	Error	Order	Error	Order
64	4.17e-02	–	1.95e-02	–	5.93e-03	–
128	3.29e-02	0.34	1.33e-02	0.55	3.56e-03	0.73
256	2.57e-02	0.36	8.99e-03	0.57	2.10e-03	0.77
512	1.99e-02	0.37	6.01e-03	0.58	1.22e-03	0.78
1024	1.54e-02	0.38	4.00e-03	0.59	7.04e-04	0.79

We then demonstrate the efficiency of the fast L2 scheme constructed on the graded mesh for the nonsmooth solution. We discretize (3.1) in time using the scheme (4.1) and in space using the spectral method. The obtained numerical results are listed in Tables 2–5. According to the maximum L^2 error behavior with respect to the time step size, it is confirmed that the convergence rates are strictly $O(N_T^{-\min\{3-\alpha, r\alpha\}})$ for the tested mesh grading constants $r = 1/\alpha, 2/\alpha, (3-\alpha)/\alpha$, and $5/\alpha$. This is in perfect agreement with the theoretical prediction given in (4.2).

TABLE 2
Maximum L^2 error and convergence rate of the graded mesh fast scheme (4.1) *with $r = 1/\alpha$.*

N_T	$\alpha = 0.4$		$\alpha = 0.6$		$\alpha = 0.8$	
	Error	Order	Error	Order	Error	Order
64	7.16e-03	–	4.00e-03	–	2.74e-03	–
128	3.64e-03	0.97	2.01e-03	0.99	1.40e-03	0.97
256	1.84e-03	0.99	1.01e-03	0.99	7.04e-04	0.99
512	9.24e-04	0.99	5.06e-04	1.00	3.53e-04	0.99
1024	4.63e-04	1.00	2.53e-04	1.00	1.77e-04	1.00

TABLE 3
Maximum L^2 error and convergence rate of the graded mesh fast scheme (4.1) *with $r = 2/\alpha$.*

N_T	$\alpha = 0.4$		$\alpha = 0.6$		$\alpha = 0.8$	
	Error	Order	Error	Order	Error	Order
64	2.86e-03	–	7.92e-04	–	4.01e-04	–
128	7.18e-04	2.00	1.98e-04	2.00	8.89e-05	2.17
256	1.80e-04	2.00	4.97e-05	2.00	1.96e-05	2.18
512	4.49e-05	2.00	1.24e-05	2.00	4.89e-06	2.00
1024	1.12e-05	2.00	3.10e-06	2.00	1.22e-06	2.00

6. Concluding remarks. We have proposed and analyzed a $(3-\alpha)$-order accelerated L2 scheme for the time-fractional diffusion equation. Basically the proposed scheme made use of the L2 scheme for discretizing the Caputo fractional derivative and the sum-of-exponentials approximation to the convolution kernel involved in the fractional derivative. The stability and convergence of the scheme were rigorously

Table 4
Maximum L^2 error and convergence rate of the graded mesh fast scheme (4.1) *with* $r = (3-\alpha)/\alpha$.

N_T	$\alpha = 0.4$		$\alpha = 0.6$		$\alpha = 0.8$	
	Error	Order	Error	Order	Error	Order
64	1.67e-03	–	4.40e-04	–	4.61e-04	–
128	2.76e-04	2.60	8.36e-05	2.40	1.02e-04	2.18
256	4.56e-05	2.60	1.58e-05	2.40	2.22e-05	2.19
512	7.52e-06	2.60	3.00e-06	2.40	4.84e-06	2.20
1024	1.24e-06	2.60	5.69e-07	2.40	1.05e-06	2.20

Table 5
Maximum L^2 error and convergence rate of the graded mesh fast scheme (4.1) *with* $r = 5/\alpha$.

N_T	$\alpha = 0.4$		$\alpha = 0.6$		$\alpha = 0.8$	
	Error	Order	Error	Order	Error	Order
64	1.36e-03	–	1.15e-03	–	2.10e-03	–
128	1.23e-04	3.47	2.28e-04	2.33	4.81e-04	2.13
256	2.02e-05	2.61	4.39e-05	2.38	1.07e-04	2.17
512	3.27e-06	2.63	8.35e-06	2.39	2.36e-05	2.18
1024	5.26e-07	2.64	1.58e-06	2.40	5.17e-06	2.19

proved for the uniform mesh. A series of numerical examples were provided to confirm the theoretical results and to demonstrate the efficiency of the proposed method. Finally, the accelerated L2 scheme was extended to some nonuniform meshes to handle the singularity of the solution. Numerical results showed that the scheme constructed on graded meshes with suitable grading constants has the capability to recover the convergence order for some kinds of nonsmooth solutions.

Appendix A.

Proof of Lemma 3.1. The properties Lemma 3.1(1)–(5) can be found in [19]; we only need to prove Lemmas 3.1(6), 3.1(7), and 3.1(8).

First, it follows from the mean value theorem that

$$
\begin{aligned}
-a_{j-1} &= \frac{3}{2}(2-\alpha)j^{1-\alpha} - \frac{1}{2}(2-\alpha)(j-1)^{1-\alpha} - j^{2-\alpha} + (j-1)^{2-\alpha} \\
&\leq \frac{1}{2}(1-\alpha)(2-\alpha)(j-1)^{-\alpha} + (2-\alpha)j^{1-\alpha} - (2-\alpha)(j-1)^{1-\alpha} \\
&\leq \frac{3}{2}(1-\alpha)(2-\alpha)(j-1)^{-\alpha}.
\end{aligned} \tag{A.1}
$$

This, together with Lemma 3.1(5), leads to

$$
\begin{aligned}
\sum_{k=2}^{n} d_0^{k,\alpha} &= \left(-a_1 - \frac{\alpha}{2}\right)p_\alpha^{-1} + \sum_{k=3}^{n}(-a_{k-1})p_\alpha^{-1} \leq \sum_{k=2}^{n}(-a_{k-1})p_\alpha^{-1} \\
&\leq (1-\alpha)(2-\alpha)\sum_{k=2}^{n}(k-1)^{-\alpha} \\
&\leq (2-\alpha)\sum_{k=2}^{n}[(k-1)^{1-\alpha} - (k-2)^{1-\alpha}] = (2-\alpha)(n-1)^{1-\alpha}.
\end{aligned}
$$

This proves Lemma 3.1(6). Then we deduce from (2.3) and (A.1)

$$\begin{aligned}\sum_{k=2}^{n} d_1^{k,\alpha} &= \Big[(2-b_1)+\Big(-a_1-b_2-\frac{\alpha}{2}\Big)+\sum_{k=4}^{n}(-a_{k-2}-b_{k-1})\Big]p_\alpha^{-1}\\ &\le \frac{4\alpha}{2+\alpha}-\sum_{k=3}^{n}a_{k-2}p_\alpha^{-1}\\ &\le \frac{4}{3}+(1-\alpha)(2-\alpha)\sum_{k=3}^{n}(k-2)^{-\alpha}\le \frac{4}{3}+(2-\alpha)(n-2)^{1-\alpha},\end{aligned}$$

which gives Lemma 3.1(7). Finally, by the definition of $d_{k-j}^{k,\alpha}$ in (2.6), we find

$$d_{k-j}^{k,\alpha}=d_{k-j+1}^{k+1,\alpha},\ j=1,\dots,k-2,\ k\ge 3,\quad \forall\alpha\in(0,1); \tag{A.2}$$

this implies that, for $2\le j\le n-2$, $j+1\le k\le n$, $d_j^{k,\alpha}=d_{j+1}^{k+1,\alpha}=d_{j+2}^{k+2,\alpha}=\cdots=d_{n-k+j}^{n,\alpha}$. Recalling Lemma 3.1(3) and (4), we derive

$$\sum_{k=j+1}^{n} d_j^{k,\alpha}=\sum_{k=j+1}^{n} d_{n-k+j}^{n,\alpha}=\sum_{k=j}^{n-1} d_k^{n,\alpha}\le\sum_{k=2}^{n-1} d_k^{n,\alpha}\le 1,$$

which proves Lemma 3.1(8). ∎

Proof of Lemma 3.2. The inequalities Lemma 3.2(1) and (2) were given in [19]. We deduce from (3.16) that

$$\begin{aligned}\sum_{j=1}^{k}\overline{d}_{k-j}^{k,\alpha} &= d_{k-1}^{k,\alpha}(1+\eta+\eta^2+\cdots+\eta^{k-1})+d_{k-2}^{k,\alpha}(1+\eta+\eta^2+\cdots+\eta^{k-2})\\ &\quad+\cdots+d_2^{k,\alpha}(1+\eta+\eta^2)+d_1^{k,\alpha}(1+\eta)+d_0^{k,\alpha}-(\eta+\eta^2+\cdots+\eta^k)\\ &= d_{k-1}^{k,\alpha}\frac{1-\eta^k}{1-\eta}+d_{k-2}^{k,\alpha}\frac{1-\eta^{k-1}}{1-\eta}+\cdots+d_2^{k,\alpha}\frac{1-\eta^3}{1-\eta}+d_1^{k,\alpha}\frac{1-\eta^2}{1-\eta}+d_0^{k,\alpha}-\eta\frac{1-\eta^k}{1-\eta}.\end{aligned}$$

Then using Lemma 3.1(4), the definition of $\overline{d}_0^{k,\alpha}$ in (3.16), and Lemma 3.2(2), we obtain

$$\begin{aligned}\sum_{j=1}^{k}\overline{d}_{k-j}^{k,\alpha} &= \frac{1}{1-\eta}\big[(d_{k-1}^{k,\alpha}+d_{k-2}^{k,\alpha}+\cdots+d_2^{k,\alpha}+d_1^{k,\alpha}+d_0^{k,\alpha})-\eta(1-\eta^k)\\ &\quad-\eta(d_{k-1}^{k,\alpha}\eta^{k-1}+d_{k-2}^{k,\alpha}\eta^{k-2}+\cdots+d_2^{k,\alpha}\eta^2+d_1^{k,\alpha}\eta+d_0^{k,\alpha})\big]\\ &= 1-\frac{\eta}{1-\eta}\overline{d}_0^{k,\alpha}\le 1-\frac{\eta(2-\alpha)(1-\alpha)}{2(1-\eta)k^\alpha}.\end{aligned}$$

This proves Lemma 3.2(3).

Note that $\overline{\xi}_k^{k,\alpha}=\xi_k^{k,\alpha}=\tilde{c}_1$; Lemma 3.2(4) follows directly from (3.11).

Finally, we deduce from (3.11) and (3.14) that $|\xi_{k-j}^{k,\alpha}|\le\frac{7}{6}\varepsilon\Delta t^{\alpha+1}$, $j=1,\dots,k$. Thus

$$|\overline{\xi}_{k-j}^{k,\alpha}|=\left|\sum_{l=0}^{j}\eta^{j-l}\xi_{k-l}^{k,\alpha}\right|\le\frac{7}{6}\varepsilon\Delta t^{\alpha+1}\sum_{l=0}^{j}\eta^l\le\frac{7}{6(1-\eta)}\varepsilon\Delta t^{\alpha+1}\le\frac{7}{2}\varepsilon\Delta t^{\alpha+1}.$$

This proves Lemma 3.2(5). ∎

REFERENCES

[1] A.A. Alikhanov, *A new difference scheme for the time fractional diffusion equation*, J. Comput. Phys., 280 (2015), pp. 424–438.

[2] D. Baffet and J.S. Hesthaven, *High-order accurate adaptive kernel compression time-stepping schemes for fractional differential equations*, J. Sci. Comput., 72 (2017), pp. 1169–1195.

[3] D. Baffet and J.S. Hesthaven, *A kernel compression scheme for fractional differential equations*, SIAM J. Numer. Anal., 55 (2017), pp. 496–520, https://doi.org/10.1137/15M1043960.

[4] C. Bernardi and Y. Maday, *Approximations spectrales de problèmes aux limites elliptiques*, Springer-Verlag, Paris, 1992.

[5] N.J. Ford and Y.B. Yan, *An approach to construct higher order time discretisation schemes for time fractional partial differential equations with nonsmooth data*, Fract. Calc. Appl. Anal., 20 (2017), pp. 1076–1105.

[6] G.H. Gao, Z.Z. Sun, and H.W. Zhang, *A new fractional numerical differentiation formula to approximate the Caputo fractional derivative and its applications*, J. Comput. Phys., 259 (2014), pp. 33–50.

[7] G.H. Gao, Z.Z. Sun, and Y.N. Zhang, *A finite difference scheme for fractional sub-diffusion equations on an unbounded domain using artificial boundary conditions*, J. Comput. Phys., 231 (2012), pp. 2865–2879.

[8] D.M. Hou and C.J. Xu, *A fractional spectral method with applications to some singular problems*, Adv. Comput. Math., 43 (2017), pp. 911–944.

[9] S.D. Jiang, J.W. Zhang, Q. Zhang, and Z.M. Zhang, *Fast evaluation of the Caputo fractional derivative and its applications to fractional diffusion equations*, Commun. Comput. Phys., 21 (2017), pp. 650–678.

[10] B.T. Jin, R. Lazarov, and Z. Zhou, *An analysis of the L1 scheme for the subdiffusion equation with nonsmooth data*, IMA J. Numer. Anal., 36 (2016), pp. 197–221.

[11] R.H. Ke, M.K. Ng, and H.W. Sun, *A fast direct method for block triangular Toeplitz-like with tri-diagonal block systems from time-fractional partial differential equations*, J. Comput. Phys., 303 (2015), pp. 203–211.

[12] H.-L. Liao, D.F. Li, and J.W. Zhang, *Sharp error estimate of the nonuniform L1 formula for linear reaction-subdiffusion equations*, SIAM J. Numer. Anal., 56 (2018), pp. 1112–1133, https://doi.org/10.1137/17M1131829.

[13] H.-L. Liao, W. McLean, and J.W. Zhang, *A Second-Order Scheme with Nonuniform Time Steps for a Linear Reaction-Subdiffusion Problem*, preprint, https://arxiv.org/abs/1803.09873, 2018.

[14] Y.M. Lin and C.J. Xu, *Finite difference/spectral approximations for the time-fractional diffusion equation*, J. Comput. Phys., 225 (2007), pp. 1533–1552.

[15] X. Lu, H.K. Pang, and H.W. Sun, *Fast approximate inversion of a block triangular Toeplitz matrix with applications to fractional sub-diffusion equations*, Numer. Linear Algebra Appl., 22 (2015), pp. 866–882.

[16] Ch. Lubich, *Discretized fractional calculus*, SIAM J. Math. Anal., 17 (1986), pp. 704–719, https://doi.org/10.1137/0517050.

[17] Ch. Lubich, *Convolution quadrature and discretized operational calculus.* I, Numer. Math., 52 (1988), pp. 129–145.

[18] C.W. Lv and C.J. Xu, *Improved error estimates of a finite difference/spectral method for time-fractional diffusion equations*, Int. J. Numer. Anal. Model., 12 (2015), pp. 384–400.

[19] C.W. Lv and C.J. Xu, *Error analysis of a high order method for time-fractional diffusion equations*, SIAM J. Sci. Comput., 38 (2016), pp. A2699–A2724, https://doi.org/10.1137/15M102664X.

[20] W. McLean, *Fast summation by interval clustering for an evolution equation with memory*, J. Sci. Comput., 34 (2012), pp. A3039–A3056.

[21] A. Quarteroni and A. Valli, *Numerical Approximation of Partial Differential Equations*, Springer-Verlag, Berlin, 1994.

[22] K. Sakamoto and M. Yamamoto, *Initial value/boundary value problems for fractional diffusion-wave equations and applications to some inverse problems*, J. Math. Anal. Appl., 382 (2011), pp. 426–447.

[23] M. Stynes, E. O'Riordan, and J. Gracia, *Error analysis of a finite difference method on graded meshes for a time-fractional diffusion equation*, SIAM J. Numer. Anal., 55 (2017), pp. 1057–1079, https://doi.org/10.1137/16M1082329.

[24] Z.Z. Sun and X.N. Wu, *A fully discrete difference scheme for a diffusion-wave system*, Appl. Numer. Math., 56 (2006), pp. 193–209.
[25] Y.Y. Xing and Y.B. Yan, *A higher order numerical method for time fractional partial differential equations with nonsmooth data*, J. Comput. Phys., 357 (2018), pp. 305–323.
[26] Y.G. Yan, Z.Z. Sun, and J.W. Zhang, *Fast evaluation of the Caputo fractional derivative and its applications to fractional diffusion equations: A second-order scheme*, Commun. Comput. Phys., 22 (2017), pp. 1028–1048.
[27] Y. Yang, Y.B. Yan, and N.J. Ford, *Some time stepping methods for fractional diffusion problems with nonsmooth data*, Comput. Methods Appl. Math., 18 (2018), pp. 129–146.
[28] F.H. Zeng, I. Turner, and K. Burrage, *A stable fast time-stepping method for fractional integral and derivative operators*, J. Sci. Comput., 77 (2018), pp. 283–307.
[29] F.H. Zeng, Z.Q. Zhang, and G.E. Karniadakis, *Second-order numerical methods for multi-term fractional differential equations: Smooth and non-smooth solutions*, Comput. Methods Appl. Mech. Engrg., 327 (2017), pp. 478–502.
[30] Q. Zhang, J.W. Zhang, S.D. Jiang, and Z.M. Zhang, *Numerical solution to a linearized time fractional KdV equation on unbounded domains*, Math. Comp., 87 (2018), pp. 693–719.