A Fast Compact Difference Scheme for the Fourth-Order Multi-Term Fractional Sub-Diffusion Equation With Non-smooth Solution

Dakang Cen\textsuperscript{a}, Zhibo Wang\textsuperscript{a}, Yan Mo\textsuperscript{a}

\textsuperscript{a}School of Mathematics and Statistics, Guangdong University of Technology, Guangdong, Guangzhou 510006, China.

Abstract. In this paper, we develop a fast compact difference scheme for the fourth-order multi-term fractional sub-diffusion equation with Neumann boundary conditions. Combining L1 formula on graded meshes and the efficient sum-of-exponentials approximation to the kernels, the proposed scheme recovers the losing temporal convergence accuracy and spares the computational costs. Meanwhile, difficulty caused by the Neumann boundary conditions and fourth-order derivative is also carefully handled. The unique solvability, unconditional stability and convergence of the proposed scheme are analyzed by the energy method. At last, the theoretical results are verified by numerical experiments.

1. Introduction

Recently, fractional differential equations (FDEs) have been widely studied by many researchers, which have become powerful tools in model simulation about wave propagation, fluid flows and financial markets, see [1–3]. Under the fact that the exact solutions of FDEs are hardly to obtain, investigating efficient numerical methods for FDEs is urgent. Different from traditional PDE problems, the solutions of FDEs are usually non-smooth and it will cost much more computation in numerical approximation. Stynes \textit{et al.} considered a reaction-diffusion problem with the Caputo time fractional derivative and analyzed a standard finite difference method for the problem on nonuniform grid [4]. Yan \textit{et al.} presented an efficient algorithm for the evaluation of the Caputo fractional derivative based on sum-of-exponentials approximation and applied it to solve the fractional diffusion equations [5]. Liao \textit{et al.} studied the stability and convergence of L1 formula on nonuniform mesh for linear reaction-subdiffusion equations based on a novel fractional Gronwall inequality [7]. More details and other research work can be found in [8]–[15]. Nonetheless, the above work mentioned here only contain a single time-fractional derivative. Actually multi-term fractional models are applied in many fields, such as visco-elastic damping, frequency-dependent loss and dispersion [16–18]. Jin \textit{et al.} considered the initial/boundary value problem...
for a diffusion equation involving multiple time-fractional derivatives on a bounded convex polyhedral domain [19]. Gao et al. used $L2-1_\sigma$ formula to numerically solve the multi-term and distributed-order time fractional sub-diffusion equations [20]. Zeng et al. applied one special case of the modified weighted shifted Grünwald-Letnikov formula to solve the multi-term fractional ordinary and partial differential equations [21]. Sun et al. derived two temporal second-order schemes for the multi-term time fractional diffusion-wave equation based on the order reduction technique [22]. Feng et al. considered a novel two-dimensional multi-term time-fractional mixed sub-diffusion and diffusion-wave equation on convex domains [23]. Lyu et al. studied a fast and linearized finite difference method to solve the nonlinear time-fractional wave equation involving multiple fractional derivatives [24]. Qiao et al. proposed a new numerical approximation for the two dimensional multi-term time fractional integro-differential equation based on the high order orthogonal spline collocation method for the spatial discretization and the classical $L1$ approximation for the Caputo fractional derivatives [25]. However, most of the results mentioned above are valid under the smooth solution assumption and only consider second-order space derivative in related equations.

In fact, fractional sub-diffusion models with the fourth-order space derivative have some important practical applications including ice formation, brain warping and wave propagation in beams, see [26, 27]. Some related research results are as follows. Hu and Zhang constructed a new implicit compact difference scheme for the fourth-order fractional diffusion-wave system by the method of order reduction [28]. Vong and Wang proposed a high-order compact difference scheme for the fourth-order fractional sub-diffusion system with the first kind of Dirichlet boundary conditions [29]. By using $L2-1_\sigma$ formula, Zhang and Pu derived a temporal second-order compact difference scheme for the fourth-order fractional sub-diffusion equations [30]. Yao and Wang considered the numerical method for the similar fourth-order fractional sub-diffusion equations under Neumann boundary conditions [31]. Based on orthogonal spline collocation method in spatial direction and classical $L1$ approximation in temporal direction, Yang et al. established a fully discrete scheme for a class of fourth-order fractional reaction-diffusion equations [32]. By an effective numerical quadrature rule based on boundary value method, Ran et al. presented a class of new compact difference schemes for solving the fourth-order time fractional sub-diffusion equation of the distributed order [33].

But the studies for the fourth-order multi-term time-fractional problems with non-smooth solutions under Neumann boundary conditions are still limited. Therefore, in this paper, we study the efficient finite difference method for the following equation:

$$
\sum_{p=0}^{q} \lambda_p D_0^\theta u(x,t) + \frac{\partial^4 u(x,t)}{\partial x^4} + u(x,t) = f(x,t), \quad 0 < x < L, \quad 0 < t \leq T,
$$

$$
u(x,0) = \phi(x), \quad 0 \leq x \leq L,
$$

$$
\frac{\partial u(0,t)}{\partial x} = \beta_0(t), \quad \frac{\partial u(L,t)}{\partial x} = \beta_1(t), \quad \frac{\partial^3 u(0,t)}{\partial x^3} = \gamma_0(t), \quad \frac{\partial^3 u(L,t)}{\partial x^3} = \gamma_1(t), \quad 0 < t \leq T,
$$

where $0 < a_q < \cdots < a_0 < 1$ and $\lambda_0, \lambda_1, \ldots, \lambda_q$ are positive weights. The symbol $D_0^\theta$ means the Caputo fractional derivative of order $\theta$, i.e.

$$
D_0^\theta u(x,t) = \frac{1}{\Gamma(1-\theta)} \int_0^t \frac{\partial u(s,t)}{\partial s} \, ds, \quad 0 < \theta < 1.
$$

Refer to [7, 37], assume that the solutions satisfy the following regularity conditions:

$$
|\partial^\sigma u(x,t)/\partial t^\sigma| \leq C(1 + |x|^{-l}), \quad \sigma \in (0, 1), \quad l = 0, 1, 2,
$$

$$
|\partial^k u(x,t)/\partial x^k| \leq C, \quad k = 1, 2, \ldots, 8,
$$

where $(x,t) \in [0, L] \times (0, T]$. Throughout this paper, we use $C$, with or without subscript, to denote positive constants independent of mesh parameters and it may takes different values at different places.

This work may be considered as a continuation of [31], in which a compact finite difference scheme on uniform grids is derived for the fourth-order fractional sub-diffusion equations. In this paper, by
using $L_1$ formula on nonuniform grid and the sum-of-exponentials (SOEs) technique, we develop a fast compact difference scheme for the problem (1)-(3), and present the corresponding rigorous error estimate under the reasonable regularity conditions mentioned above. The sharp theoretical results can be easily extended to the case of distributed order sub-diffusion equation, which can be approximated by the multi-term sub-diffusion equation. In fact, the core of the fast algorithm is approximating the kernel function $t^{-\beta-1}$ ($0 < \beta < 1$) on the internal $[0, T]$ by using SOEs, where $\delta$ is cut-off time restriction and $T$ is the final time. It shows that the fast algorithm has nearly optimal complexity - $O(MN_{\exp})$ work and $O(MN_{\exp})$ storage, where $M, N$ are the total numbers of grids in spatial direction and in temporal direction, $N_{\exp}$ is the number of exponentials [5].

The structure of the paper is as follows. In Section 2, we do some preliminary work and the fast compact difference scheme for (6)-(9) is also established. The proof of the stability and convergence will be presented in Section 3. In Section 4, numerical experiments are carried out to verify the theoretical claims. The article ends with a brief conclusion.

2. Preliminaries

Firstly, by the order of reduction, we introduce the following equivalent form for the problem:

$$
\frac{\partial}{\partial t}v(x, t) + \frac{\partial^2 v(x, t)}{\partial x^2} + u(x, t) = f(x, t), \quad 0 < x < L, \quad 0 < t \leq T,
$$

$$
\frac{\partial v(x, 0)}{\partial x} = \phi(x), \quad 0 < x < L,
$$

$$
\frac{\partial u(0, t)}{\partial x} = \beta_0(t), \quad \frac{\partial u(L, t)}{\partial x} = \beta_1(t), \quad \frac{\partial \gamma_0(t)}{\partial x} = \gamma_0(t), \quad \frac{\partial \gamma_1(t)}{\partial x} = \gamma_1(t), \quad 0 < t \leq T.
$$

Some useful notations are now defined. Let $h = \frac{L}{M}$, $x_i = ih, \quad 0 \leq i \leq M$, $t_n = T(n/M), \quad 0 \leq n \leq N$, $\tau_k = t_k - t_{k-1}, \quad 1 \leq k \leq N$ ($M, N \in \mathbb{N}^+$, $r \geq 1$). For a grid function $u = \{u_i\}_{0 \leq i \leq M}$, denote

$$
\delta_k u_i = \begin{cases} 
\frac{1}{h} (u_i - u_{i-1}), & 1 \leq i \leq M, \\
\frac{1}{h} (\partial_x u_i), & i = 0,
\end{cases}
$$

$$
\delta_k^2 u_i = \begin{cases} 
\frac{1}{h^2} (\partial_x^2 u_i), & 1 \leq i \leq M - 1,
\frac{1}{h^2} (\partial_x^2 u_M), & i = M,
\end{cases}
$$

$$
\mathcal{H} u_i = \begin{cases} 
\frac{1}{h} (5u_i + u_{i+1}), & i = 0,
\frac{1}{h} (u_{i-1} + 10u_i + u_{i+1}), & 1 \leq i \leq M - 1,
\frac{1}{h} (u_{M-1} + 5u_M), & i = M.
\end{cases}
$$

For grid functions $u, v$, the notations of discrete inner products and norms are defined as follows:

$$(u, v) = h \left( \frac{1}{2} u_0 v_0 + \sum_{i=1}^{M-1} u_i v_i + \frac{1}{2} u_M v_M \right), \quad \|u\|^2 = (u, u).$$

Then, we review the fast approximation to the Caputo derivative $\frac{\partial}{\partial t} D_\alpha u(t_n)$, $\alpha \in (0, 1)$, via Lemma 2.1, see [5, 6].

**Lemma 2.1.** Let $\epsilon$ denote tolerance error, $\delta$ cut-off time restriction and $T$ final time. Then there is a natural number $N_{\exp}$ and positive numbers $s_j$ and $w_j$, $j = 1, 2, \ldots, N_{\exp}$ such that

$$
\left| t^{-\beta} - \sum_{j=1}^{N_{\exp}} w_j e^{-s_j t} \right| \leq \epsilon, \quad t \in [\delta, T],
$$

$$
\sum_{j=1}^{N_{\exp}} w_j e^{-s_j t} \leq \epsilon, \quad t \in [\delta, T].
$$
where

\[ N_{exp} = O\left( (\log e^{-1})(\log \log e^{-1} + \log(T\delta^{-1})) + (\log \delta^{-1})(\log \log e^{-1} + \log\delta^{-1}) \right). \]

According to the linear polynomial interpolation and Lemma 2.1, one has

\[
\frac{\zeta}{\alpha} D_t^n u(t_n) = \frac{1}{\Gamma(1-\alpha)} \left[ \int_0^{t_{n+1}} \frac{u'(s)}{(t_n-s)^\alpha} ds + \int_{t_{n+1}}^{t_n} \frac{u'(s)}{(t_n-s)^\alpha} ds \right] - \int_0^{t_{n+1}} \frac{u(t_n) - u(t_{n-1})}{\tau_n} \frac{1}{(t_n-s)^\alpha} ds
\]

\[
\approx \frac{1}{\Gamma(1-\alpha)} \left[ \int_0^{t_{n+1}} u'(s) \sum_{j=1}^{N_{exp}} w_j e^{-\gamma_j (t_n-s)} ds + \int_{t_{n+1}}^{t_n} \frac{u(t_n) - u(t_{n-1})}{\tau_n} \frac{1}{(t_n-s)^\alpha} ds \right]
\]

\[
= \frac{1}{\Gamma(1-\alpha)} \left\{ \sum_{j=1}^{N_{exp}} w_j \int_0^{t_{n+1}} u'(s) e^{-\gamma_j (t_n-s)} ds + \frac{\tau_n^{-\alpha}}{1-\alpha} \left[ u(t_n) - u(t_{n-1}) \right] \right\}
\]

\[
:= F^\alpha D_t^n u(t_n), \quad 1 \leq n \leq N. \tag{10}
\]

Follow the idea in [5], denote \( F_j^n = \int_0^{t_{n+1}} u'(s) e^{-\gamma_j (t_n-s)} ds \), it is easy to check that

\[
F_j^n \approx e^{-\gamma_j} F_j^{n-1} + B_j^n [u(t_{n-1}) - u(t_{n-2})], \quad n \geq 2, \tag{11}
\]

where

\[
F_j^1 = 0, \quad B_j^n = \int_{t_{n-1}}^{t_n} e^{-\gamma_j (t_n-s)} ds, \quad 1 \leq j \leq N_{exp}. \tag{12}
\]

Combining (10)-(12), one arrives that

\[
F^\alpha D_t^n u(t_n) = \frac{1}{\Gamma(1-\alpha)} \left\{ \sum_{j=1}^{N_{exp}} w_j F_j^n + \frac{\tau_n^{-\alpha}}{1-\alpha} \left[ u(t_n) - u(t_{n-1}) \right] \right\}, \quad n \geq 1, \tag{13}
\]

\[
F_j^n = e^{-\gamma_j} F_j^{n-1} + B_j^n [u(t_{n-1}) - u(t_{n-2})], \quad n \geq 2, \tag{14}
\]

\[
F_j^1 = 0. \tag{15}
\]

For the convenience of stability and convergence analysis, an equivalent form is proposed as follow:

\[
F^\alpha D_t^n u(t_n) = \frac{1}{\Gamma(1-\alpha)} \left[ b_j^{(n)} u(t_n) - \sum_{k=1}^{n-1} \left( b_j^{(n)}(k+1) - b_j^{(n)}(k) \right) u(t_k) - b_j^{(n)}(1) u(t_0) \right], \quad 1 \leq n \leq N, \tag{16}
\]

with

\[
b_j^{(n)} = \begin{cases} \sum_{j=1}^{N_{exp}} w_j \int_{t_{j-1}}^{t_j} e^{-\gamma_j (t_n-s)} ds, & k = 1, 2, \ldots, n-1, \\ \frac{\tau_n^{-\alpha}}{1-\alpha}, & k = n. \end{cases} \]

In the practical numerical computation, \( N_{exp} \) is usually much smaller than \( N \), see [5]. It shows that the fast algorithm (13)-(15) effectively reduces the computation costs compared to the direct method in [4, 7]:

\[
D_t^n u(t_n) = \frac{1}{\Gamma(1-\alpha)} \left[ a_j^{(n)} u(t_n) - \sum_{k=1}^{n-1} \left( a_j^{(n)}(k+1) - a_j^{(n)}(k) \right) u(t_k) - a_j^{(n)}(1) u(t_0) \right], \quad 1 \leq n \leq N, \tag{17}
\]

where \( a_j^{(n)} = \frac{1}{\tau_n} \int_{t_{j-1}}^{t_j} \frac{ds}{(t_n-s)^\alpha} \).

Now, we proceed to derive our numerical scheme for problem (6)-(9). The truncation error of our scheme based on the following two lemmas.
Lemma 2.2. ([37]) Under the assumption (4), one has
\[ \frac{5}{6} \frac{D^2_t}{D_t} u(t_n) = D^2_t u(t_n) + O(t_n^{-n}N^{-\min\{n, 2, n\}}) \]
\[ = \frac{5}{6} D^2_t u(t_n) + O(t_n^{-n}N^{-\min\{n, 2, n\}} + e), \quad 1 \leq n \leq N. \]

Lemma 2.3. ([34, 36])
(I) Suppose \( u \in C^2[x_0, x_1] \), then we have
\[ \left[ \frac{1}{6} u''(x_0) + \frac{5}{6} u''(x_1) \right] - \frac{1}{6} \left[ \frac{u(x_1) - u(x_0)}{h} - u'(x_0) \right] = -\frac{h}{6} u'''(x_0) + \frac{h^3}{90} u^{(5)}(x_0) + O(h^4). \]

(II) Suppose \( u \in C^2[x_{M-1}, x_M] \), then we get
\[ \left[ \frac{1}{6} u''(x_{M-1}) + \frac{5}{6} u''(x_M) \right] - \frac{1}{6} \left[ \frac{u(x_M) - u(x_{M-1})}{h} - u'(x_M) \right] = \frac{h}{6} u'''(x_M) - \frac{h^3}{90} u^{(5)}(x_M) + O(h^4). \]

(III) Suppose \( u \in C^2[x_{i-1}, x_{i+1}] \), \( 1 \leq i \leq M - 1 \), then it holds that
\[ \frac{1}{12} \left[ u''(x_{i-1}) + 10u''(x_i) + u''(x_{i+1}) \right] - \frac{1}{12} \left[ u(x_{i-1}) - 2u(x_i) + u(x_{i+1}) \right] = O(h^4). \]

Following the idea in [31, 36], we differentiate equation (6) with respect to \( x \) and let \( x \to 0^+ \), under the boundary conditions (9), it arrives that,
\[ \frac{\partial^3 v(0, t)}{\partial x^3} = -\left[ \sum_{\nu=0}^{q} \lambda_\nu C D^\nu_t \beta_\nu(t) + \beta_0(t) - f_x(0, t) \right]. \quad (18) \]

In a similar way, differentiating equation (6) three times with respect to \( x \) yields
\[ \frac{\partial^3 v(0, t)}{\partial x^3} = -\left[ \sum_{\nu=0}^{q} \lambda_\nu C D^\nu_t \gamma_\nu(t) + \gamma_0(t) - f_{xxx}(0, t) \right]. \quad (19) \]

Repeat above operations at the other end of the boundary, we obtain
\[ \frac{\partial^3 v(L, t)}{\partial x^3} = -\left[ \sum_{\nu=0}^{q} \lambda_\nu C D^\nu_t \beta_\nu(t) + \beta_1(t) - f_x(L, t) \right]. \quad (20) \]

and
\[ \frac{\partial^3 v(L, t)}{\partial x^3} = -\left[ \sum_{\nu=0}^{q} \lambda_\nu C D^\nu_t \gamma_\nu(t) + \gamma_1(t) - f_{xxx}(L, t) \right]. \quad (21) \]

Denote \( u^n_i \) and \( v^n_i \) is the numerical solution of (6)-(9) at grid point \((x_i, t_n)\). We proposed the following fast
3. Stability and convergence analysis

At first, we introduce some useful lemmas, which will be used in stability and convergence analysis.

Lemma 3.1. ([35]) Let $u$ be a grid function, then it holds that

$$\frac{5}{12} ||u||^2 \leq ||Hu||^2 \leq ||u||^2.$$ 

Lemma 3.2. ([31]) For any grid function $u, v$, one has

$$(\delta^2 u, Hu) = (\delta^2 v, Hu).$$

Lemma 3.3. Suppose $\varepsilon \leq \min\{C_pN^{\alpha}, T^{-\alpha}/2\}$ with $C_p$ being a positive constant, for $b_k^{(n,\alpha)}|1 \leq n \leq N, 1 \leq k \leq n|$, where $\alpha_p \in (0, 1)$, defined by (16), we have

(I) $b_1^{(n,\alpha)} \geq \frac{1}{2} l_n^{-\alpha}$,

(II) $0 < b_1^{(n,\alpha)} < \cdots < b_k^{(n,\alpha)} < \cdots < b_n^{(n,\alpha)}$.

Proof. By the mean-value theorem, there exists a number $\xi_k \in (t_{k-1}, t_k)$, such that

$$a_k^{(n,\alpha)} = (t_n - \xi_k)^{-\alpha},$$
Based on (35), it is easy to check that

\[
\text{Lemma 3.3.}
\]

By Lemma 3.3, we have

\[
\text{and}
\]

Proof. Making the inner product of (29) and (30) with \( \mathcal{H} v^p \) respectively, we obtain

\[
\left( \sum_{p=0}^{q} \lambda_p F D_t^{(n_\alpha)} H u^p + \delta_t^2 v^p, \mathcal{H} v^p \right) + \| \mathcal{H} v^p \|^2 = (p^n, \mathcal{H} v^p),
\]

and

\[
\| \mathcal{H} v^p \|^2 = (\delta_t^2 u^p, \mathcal{H} v^p) + (Q^p, \mathcal{H} v^p).
\]

By Lemma 3.3, we have

\[
2F D_t^{(n_\alpha)} u^p, u^p = \frac{2b^{(n_\alpha)}}{\alpha_p} \| u^p \|^2 - \sum_{k=1}^{n-1} \frac{b^{(n_\alpha)}}{\alpha_p} \| u^p \|^2 - \frac{2b^{(n_\alpha)}}{\alpha_p} \| u^p \|^2
\]

Based on (35), it is easy to check that

\[
\left( \sum_{p=0}^{q} \lambda_p F D_t^{(n_\alpha)} H u^p, H u^p \right) \geq \frac{1}{2} \sum_{p=0}^{q} \lambda_p F D_t^{(n_\alpha)} \| H u^p \|^2.
\]
Adding (33) and (34), by Lemma 3.2 and (36), we get
\[
\frac{1}{2} \sum_{p=0}^{q} \lambda_p D_1 \left\| \mathcal{H} u^p \right\|^2 + \left\| \mathcal{H} u^0 \right\|^2 + \left\| \mathcal{H} \nu^0 \right\|^2 \leq (P^0, \mathcal{H} u^0) + (Q^0, \mathcal{H} \nu^0).
\]

Using Cauchy-Schwarz inequality, it holds that
\[
\sum_{p=0}^{q} \lambda_p D_1 \left\| \mathcal{H} u^p \right\|^2 \leq 2 \| P^0 \| \| \mathcal{H} u^0 \| + \frac{1}{2} \| Q^0 \|^2 := \left\| \bar{P}^0 \right\| \| \mathcal{H} u^0 \| + \| \bar{Q}^0 \|^2,
\]

(37)

where \( \bar{P}^0 = 2P^0 \), \( \bar{Q}^0 = \frac{\lambda^2}{2} Q^0 \), 1 \( \leq n \leq N \).

We prove the main results by using mathematical induction. Considering the estimate (32) in the case of \( k = 1 \). If
\[
\| \mathcal{H} u^1 \| \leq \frac{\| \bar{Q}^1 \|}{\sqrt{\sum_{p=0}^{q} \lambda_p b_p \frac{1}{1-\alpha_p} G(1-\alpha_p)}},
\]
the proof is completed. Otherwise,
\[
\| \mathcal{H} u^1 \| > \frac{\| \bar{Q}^1 \|}{\sqrt{\sum_{p=0}^{q} \lambda_p b_p \frac{1}{1-\alpha_p} G(1-\alpha_p)}},
\]
there are two cases.

Case 1. If \( \| \mathcal{H} u^1 \| \leq \| \mathcal{H} u^0 \| \), the proof is finished.

Case 2. If \( \| \mathcal{H} u^1 \| > \| \mathcal{H} u^0 \| \), it follows that
\[
\sum_{p=0}^{q} \frac{\lambda_p b_p \frac{1}{1-\alpha_p}}{G(1-\alpha_p)} \| \mathcal{H} u^1 \|^2 \leq \sum_{p=0}^{q} \frac{\lambda_p b_p \frac{1}{1-\alpha_p}}{G(1-\alpha_p)} \| \mathcal{H} u^0 \|^2 + \| P^1 \| \| \mathcal{H} u^0 \| + \| Q^1 \|^2
\]
\[
\leq \sum_{p=0}^{q} \frac{\lambda_p b_p \frac{1}{1-\alpha_p}}{G(1-\alpha_p)} \| \mathcal{H} u^0 \|^2 \| \mathcal{H} u^0 \| + \| P^1 \| \| \mathcal{H} u^1 \| + \sqrt{\sum_{p=0}^{q} \frac{\lambda_p b_p \frac{1}{1-\alpha_p}}{G(1-\alpha_p)} \| \bar{Q}^1 \| \| \mathcal{H} u^1 \|}.
\]

Dividing both sides of the above result by \( \| \mathcal{H} u^1 \| \), we obtain
\[
\sum_{p=0}^{q} \frac{\lambda_p b_p \frac{1}{1-\alpha_p}}{G(1-\alpha_p)} \| \mathcal{H} u^1 \| \leq \sum_{p=0}^{q} \frac{\lambda_p b_p \frac{1}{1-\alpha_p}}{G(1-\alpha_p)} \| \mathcal{H} u^0 \| + \| P^1 \| \| \mathcal{H} u^0 \| + \sqrt{\sum_{p=0}^{q} \frac{\lambda_p b_p \frac{1}{1-\alpha_p}}{G(1-\alpha_p)} \| \bar{Q}^1 \| \| \mathcal{H} u^1 \|},
\]
that is
\[
\| \mathcal{H} u^1 \| \leq \| \mathcal{H} u^0 \| + \frac{\| P^1 \|}{\sum_{p=0}^{q} \frac{\lambda_p b_p \frac{1}{1-\alpha_p}}{G(1-\alpha_p)}} + \frac{\| \bar{Q}^1 \|}{\sqrt{\sum_{p=0}^{q} \frac{\lambda_p b_p \frac{1}{1-\alpha_p}}{G(1-\alpha_p)}}}.
\]

the estimate (32) holds for \( k = 1 \). Assume that the estimate is valid for \( k = 1, \ldots, n-1 \) with \( n \leq N \), i.e.
\[
\| \mathcal{H} u^k \| \leq \| \mathcal{H} u^0 \| + \max_{1 \leq m \leq k} \frac{\| P^m \|}{\sum_{p=0}^{q} \frac{\lambda_p b_p \frac{1}{1-\alpha_p}}{G(1-\alpha_p)}} + \max_{1 \leq m \leq k} \frac{\| \bar{Q}^m \|}{\sqrt{\sum_{p=0}^{q} \frac{\lambda_p b_p \frac{1}{1-\alpha_p}}{G(1-\alpha_p)}}}, \quad 1 \leq k \leq n-1.
\]

(38)
Theorem 3.1. The fast compact finite scheme (22)-(28) is uniquely solvable.
Proof. Denote \( u^n = (u^n_0, u^n_1, \ldots, u^n_M) \), \( v^n = (v^n_0, v^n_1, \ldots, v^n_M) \). The initial values \( u^0 \) is determined by (28). The linear system in \( u^1 \), \( v^1 \) can be obtained from scheme (22)-(27). To show their unique solvability, consider the corresponding homogeneous system:

\[
\left[ \sum_{p=0}^{a} \lambda_p \beta_{p,1}^{(1, n, i)} \right] + 1 \right] H u^1_i + \delta_i^2 v^1_i = 0, \quad (39) \\
- \delta_i^2 u^1_i + H v^1_i = 0. \quad (40)
\]

Taking the inner product of (39) and (40) with \( H u^1 \) and \( H v^1 \), respectively, we obtain

\[
\left( \sum_{p=0}^{a} \lambda_p \beta_{p,1}^{(1, n, i)} \right) \| H u^1 \|^2 + (\delta_i^2 v^1, Hu^1) = 0, \quad (41)
\]

and

\[
\| H v^1 \|^2 - (\delta_i^2 u^1, Hv^1) = 0. \quad (42)
\]

Adding (41) and (42), noting Lemma 3.2, we obtain

\[
\left( \sum_{p=0}^{a} \lambda_p \beta_{p,1}^{(1, n, i)} \right) \| H u^1 \|^2 + \| H v^1 \|^2 = 0. \quad (43)
\]

By Lemma 3.1, we deduce that

\[
\| u^1 \| = 0, \quad \| v^1 \| = 0,
\]

which means \( u^1 = 0, v^1 = 0 \). Thus the unique solvability to \( u^1, v^1 \) is confirmed. If \( u^1, \ldots, u^{n-1}, v^1, \ldots, v^{n-1} \) have been uniquely determined, then we get a linear system with respect to \( u^n, v^n \). One has \( u^n, v^n \) are uniquely determined and the process of argument is similar to (39)-(43). The proof is completed by the principle of induction. □

From Lemma 3.4, we obtain the following stability statement.

Theorem 3.2. The fast compact finite scheme (22)-(28) is unconditionally stable.

Theorem 3.3. (convergence estimate) Assume that \( u(x, t), v(x, t) \) is the solution of (6)-(9) and \( u^n_i, v^n_i, 0 \leq i \leq M, 0 \leq n \leq N \) is the solution of the finite difference scheme (22)-(28), respectively. Denote

\[
e_i^n = u(x_i, t_n) - u^n_i, \quad e_i^n = v(x_i, t_n) - v^n_i, \quad 0 \leq i \leq M, 0 \leq n \leq N.
\]

If \( \epsilon \leq \min_{0 \leq p \leq 1} \{ C_p N^\alpha, T^{-\alpha_r} \} \) with \( C_p \) being positive constants, then there exists a positive constant \( C \) such that

\[
\| e_i^n \| \leq C (N^{- \min\{n, 2 - \alpha_r \}} + h + \epsilon), \quad 0 \leq n \leq N.
\]

Proof. It is easy to get the following error equation:

\[
\sum_{p=0}^{a} \lambda_p \beta_{p,1}^{(1, n, i)} H e_i^n + \delta_i^2 e_i^n + H e_i^n = R_i^n, \quad 0 \leq i \leq M, 1 \leq n \leq N, \quad (44)
\]

\[
H e_i^n = \delta_i^2 e_i^n + S_i^n, \quad 0 \leq i \leq M, 1 \leq n \leq N, \quad (45)
\]

\[
e_i^0 = 0, \quad 0 \leq i \leq M, \quad (46)
\]

where \( R_i^n = O(\sum_{p=0}^{a} \epsilon p N^{- \min\{n, 2 \alpha_r \}} + h + \epsilon), S_i^n = O(h^4) \).
Lemma 3.3 and Lemma 3.4 imply that

\[
\|H^p\| \leq \max_{1 \leq m \leq n} \frac{2\|R^m\|}{\sum_{p=0}^{q} \lambda^m_{p}(1-\omega)} + \max_{1 \leq m \leq n} \frac{\|S^m\|}{\sqrt{2} \sum_{p=0}^{q} \lambda^m_{p}(1-\omega)}
\]

\[
\leq C_1 \left[ \max_{1 \leq m \leq n} \frac{1}{\sum_{p=0}^{q} \lambda^m_{p}(1-\omega)} \left( \sum_{p=0}^{q} \Gamma(1-\omega) + \epsilon \right) \right] + \max_{1 \leq m \leq n} \frac{1}{\sqrt{2} \sum_{p=0}^{q} \lambda^m_{p}(1-\omega)} h^4
\]

\[
\leq C_1 \left[ 2 \max_{1 \leq m \leq n} \left( \sum_{p=0}^{q} \Gamma(1-\omega) + \epsilon \right) \right] + 2C_2(N^{-\min\{\nu,2-\alpha\}} + h^4)
\]

where \(C_1\) and \(C_2\) are positive constants. The desired result then follows by Lemma 3.1. \(\square\)

4. Numerical experiments

In this section, we carry out numerical experiments to illustrate our theoretical statements and all our tests are done in MATLAB with a laptop. The \(L^2\) norm errors between the exact and the numerical solutions

\[
E_2(M, N) = \max_{0 \leq t \leq N} \|e^p\|
\]

are shown in the following tables. Furthermore, the temporal convergence order and spatial convergence order, denoted by

\[
\text{Rate1} = \log_2 \left( \frac{E_2(M, N/2)}{E_2(M, N)} \right) \quad \text{and} \quad \text{Rate2} = \log_2 \left( \frac{E_2(M/2, N)}{E_2(M, N)} \right)
\]

respectively, are reported.

**Example 4.1.** The following problem is considered:

\[
\sum_{p=0}^{q} \lambda^m_{p} x^p \frac{\partial^m}{\partial t^m} u(x, t) + u_{xxxx}(x, t) + u(x, t) = f(x, t), \quad 0 < x < 1, \quad 0 < t \leq 1,
\]

\[
u(x, 0) = \cos(\pi x), \quad 0 \leq x \leq 1,
\]

\[
u_x(0, t) = u_x(1, t) = u_{xxxx}(0, t) = u_{xxxx}(1, t) = 0, \quad 0 < t \leq 1,
\]

where

\[
f(x, t) = \cos(\pi x) \left( \sum_{p=0}^{q} \lambda^m_{p} \frac{\Gamma(1+\alpha_0)}{\Gamma(1+\alpha_0-\alpha_p)} x^{\alpha_p-\alpha_p} (1 + \pi^2)(1 + \pi^4) \right).
\]

The exact solution for this problem is \(u(x, t) = \cos(\pi x) (1 + t^\alpha)\).

We set the tolerance error \(\epsilon = 10^{-8}\) and cut-off time \(\delta = 10^{-12}\) in Fast Scheme of Example 4.1. Moreover, to verify the efficiency of the proposed scheme, we compare it with Direct Scheme. In Table 1, the temporal convergence order \(\text{Rate1} \approx \min\{\alpha, 2-\alpha_0\}\) with regularity parameter \(\alpha = \alpha_0\). Table 2 shows that the spatial convergence order is equal to \(O(M^{-4})\). Under the tolerance error condition, Fast Scheme shows its powerful efficiency compared to Direct Scheme, see Table 5.
Table 1: Numerical convergence orders in temporal direction for Example 4.1 with $M = 100$.

| $(r, \alpha_0, \alpha_1, \lambda_0, \lambda_1)$ | $N$   | Fast Scheme |  | Direct Scheme |  |
|-----------------------------------------------|------|-------------|---------------------|---------------------|
|                                               |      | $E_2(M, N)$ | Rate 1             | $E_2(M, N)$ | Rate 1             |
| $(1,0.8,0.3,2,3)$                             | 1024 | 3.317e-04   | *                   | 3.317e-04 | *                   |
|                                               | 2048 | 2.048e-04   | 0.6953              | 2.048e-04 | 0.6953              |
|                                               | 4096 | 1.272e-04   | 0.6879              | 1.272e-04 | 0.6879              |
|                                               | 8192 | 7.663e-05   | 0.7307              | 7.663e-05 | 0.7307              |
| $(1,0.8,0.3,3,2)$                             | 1024 | 3.536e-04   | *                   | 3.536e-04 | *                   |
|                                               | 2048 | 2.199e-04   | 0.6854              | 2.199e-04 | 0.6854              |
|                                               | 4096 | 1.330e-04   | 0.7249              | 1.330e-04 | 0.7249              |
|                                               | 8192 | 7.894e-05   | 0.7531              | 7.894e-05 | 0.7531              |
| $(2,0.5,0.3,1,2)$                             | 1024 | 1.313e-04   | *                   | 1.313e-04 | *                   |
|                                               | 2048 | 6.874e-05   | 0.9341              | 6.874e-05 | 0.9341              |
|                                               | 4096 | 3.531e-05   | 0.9611              | 3.531e-05 | 0.9611              |
|                                               | 8192 | 1.794e-05   | 0.9766              | 1.794e-05 | 0.9766              |
| $(2,0.5,0.3,2,1)$                             | 1024 | 1.405e-04   | *                   | 1.405e-04 | *                   |
|                                               | 2048 | 7.188e-05   | 0.9669              | 7.188e-05 | 0.9669              |
|                                               | 4096 | 3.644e-05   | 0.9803              | 3.644e-05 | 0.9803              |
|                                               | 8192 | 1.849e-05   | 0.9790              | 1.849e-05 | 0.9790              |
| $(3,0.8,0.5,1,3)$                             | 1024 | 2.775e-06   | *                   | 2.778e-06 | *                   |
|                                               | 2048 | 1.193e-06   | 1.2182              | 1.195e-06 | 1.2165              |
|                                               | 4096 | 5.125e-07   | 1.2188              | 5.151e-07 | 1.2147              |
|                                               | 8192 | 2.197e-07   | 1.2221              | 2.222e-07 | 1.2127              |
| $(3,0.8,0.5,3,1)$                             | 1024 | 5.578e-06   | *                   | 5.581e-06 | *                   |
|                                               | 2048 | 2.425e-06   | 1.2015              | 2.428e-06 | 1.2005              |
|                                               | 4096 | 1.053e-06   | 1.2035              | 1.056e-06 | 1.2012              |
|                                               | 8192 | 4.563e-07   | 1.2066              | 4.593e-07 | 1.2014              |

Table 2: Numerical convergence orders in spatial direction for Example 4.1 with $N = 10000$.

| $(r, \alpha_0, \alpha_1, \lambda_0, \lambda_1)$ | $M$ | Fast Scheme |  | Direct Scheme |  |
|------------------------------------------------|----|-------------|---------------------|---------------------|
|                                               |      | $E_2(M, N)$ | Rate 2             | $E_2(M, N)$ | Rate 2             |
| $(3,0.5,0.3,1,3)$                             | 4   | 4.403e-03   | *                   | 4.403e-03 | *                   |
|                                               | 8   | 2.699e-04   | 4.0277              | 2.699e-04 | 4.0277              |
|                                               | 16  | 1.679e-05   | 4.0069              | 1.679e-05 | 4.0069              |
|                                               | 32  | 1.045e-06   | 4.0102              | 1.045e-06 | 4.0102              |
| $(3,0.5,0.3,3,1)$                             | 4   | 4.415e-03   | *                   | 4.415e-03 | *                   |
|                                               | 8   | 2.707e-04   | 4.0278              | 2.707e-04 | 4.0278              |
|                                               | 16  | 1.683e-05   | 4.0072              | 1.683e-05 | 4.0072              |
|                                               | 32  | 1.045e-06   | 4.0102              | 1.044e-06 | 4.0104              |
Example 4.2. Moreover, another example with nonzero initial and boundary conditions is considered:

$$\sum_{p=0}^{1} \lambda_p \phi^0_p u(x, t) + u_{xxxx}(x, t) + u(x, t) = f(x, t), \quad 0 < x < 1, \quad 0 < t \leq 1,$$

$$u(x, 0) = \cos(\pi x), \quad 0 \leq x \leq 1,$$

$$u_0(0, t) = u_{xxx}(0, t) = t^{\alpha_0}, \quad 0 < t \leq 1,$$

$$u_1(1, t) = u_{xxx}(1, t) = e^{t^{\alpha_0}}, \quad 0 < t \leq 1,$$

where

$$f(x, t) = \sum_{p=0}^{1} \lambda_p \frac{\Gamma(1 + \alpha_0)}{\Gamma(1 + \alpha_0 - \alpha_p)} e^{x^{2\beta_{\alpha_p}} + 2e^{x^{2\beta_{\alpha_p}}} + (\pi^2 + 1) \cos(\pi x)}.$$

The exact solution for this problem is

$$u(x, t) = \cos(\pi x) + e^{t^{\alpha_0}}.$$

Table 3: Numerical convergence orders in temporal direction for Example 4.2 with $M = 100$.

| $(r, \alpha_0, \alpha_1, \lambda_0, \lambda_1)$ | $N$ | Fast Scheme | Direct Scheme |
|---------------------------------------------|-----|-------------|---------------|
|                                             | $E_2(M, N)$ | Rate 1 | $E_2(M, N)$ | Rate 1 |
| $(1,0.5,0.3,1,2)$ | 1024 | $9.779e-03$ | * | $9.779e-03$ | * |
|                             | 2048 | $7.066e-03$ | 0.4687 | $7.066e-03$ | 0.4687 |
|                             | 4096 | $5.098e-03$ | 0.4711 | $5.098e-03$ | 0.4711 |
|                             | 8192 | $3.672e-03$ | 0.4732 | $3.672e-03$ | 0.4732 |
| $(1,0.5,0.3,2,1)$ | 1024 | $1.094e-02$ | * | $1.094e-02$ | * |
|                             | 2048 | $7.835e-03$ | 0.4822 | $7.835e-03$ | 0.4822 |
|                             | 4096 | $5.602e-03$ | 0.4839 | $5.602e-03$ | 0.4839 |
|                             | 8192 | $4.001e-03$ | 0.4855 | $4.001e-03$ | 0.4855 |
| $(2,0.6,0.3,1,3)$ | 1024 | $1.137e-04$ | * | $1.137e-04$ | * |
|                             | 2048 | $5.090e-05$ | 0.1591 | $5.090e-05$ | 0.1591 |
|                             | 4096 | $2.259e-05$ | 0.1721 | $2.259e-05$ | 0.1721 |
|                             | 8192 | $9.958e-06$ | 0.1817 | $9.958e-06$ | 0.1817 |
| $(2,0.6,0.3,3,1)$ | 1024 | $1.221e-04$ | * | $1.221e-04$ | * |
|                             | 2048 | $5.341e-05$ | 0.1591 | $5.341e-05$ | 0.1591 |
|                             | 4096 | $2.332e-05$ | 0.1953 | $2.332e-05$ | 0.1953 |
|                             | 8192 | $1.018e-05$ | 0.1967 | $1.018e-05$ | 0.1967 |
| $(3,0.9,0.3,2,3)$ | 1024 | $5.973e-05$ | * | $5.973e-05$ | * |
|                             | 2048 | $2.778e-05$ | 0.1046 | $2.778e-05$ | 0.1046 |
|                             | 4096 | $1.293e-05$ | 0.1029 | $1.293e-05$ | 0.1029 |
|                             | 8192 | $6.028e-06$ | 0.1003 | $6.020e-06$ | 0.1023 |
| $(3,0.9,0.3,3,2)$ | 1024 | $9.288e-05$ | * | $9.288e-05$ | * |
|                             | 2048 | $4.328e-05$ | 0.1017 | $4.328e-05$ | 0.1017 |
|                             | 4096 | $2.018e-05$ | 0.1011 | $2.017e-05$ | 0.1012 |
|                             | 8192 | $9.410e-06$ | 0.1009 | $9.405e-06$ | 0.1009 |

In Example 4.2, for Fast Scheme, we set the tolerance error $\epsilon = 10^{-8}$ and cut-off time $\delta = 5 \times 10^{-14}$. It is easy to check that the temporal convergence order $\text{Rate1} \approx \min\{r \sigma, 2 - \alpha_0\}$ with regularity parament $\sigma = \alpha_0$, while the spatial convergence order $\text{Rate2} \approx 4$, reported in Table 3 and Table 4, respectively. What’s more, for both examples, the proposed scheme takes less CPU time than Direct Scheme in Table 5. These practical computation confirm the theoretical analysis.

5. Conclusion

In this paper, we study a fast compact difference scheme for the fourth-order multi-term fractional sub-diffusion equation with Neumann boundary conditions. After a equivalent transformation, based on
Table 4: Numerical convergence orders in spatial direction for Example 4.2 with \(N = 25000\).

| \((r, \alpha_0, \alpha_1, \lambda_0, \lambda_1)\) | \(M\) | Fast Scheme | Direct Scheme |
|---|---|---|---|
| | | \(E_2(M, N)\) | Rate2 | \(E_2(M, N)\) | Rate2 |
| \((3,0.5,0.3,2,3)\) | 4 | 2.199e-03 | * | 2.199e-03 | * |
| | 8 | 1.348e-04 | 4.0277 | 1.348e-04 | 4.0277 |
| | 16 | 8.389e-06 | 4.0065 | 8.389e-06 | 4.0065 |
| | 32 | 5.316e-07 | 3.9801 | 5.304e-07 | 3.9835 |
| \((3,0.5,0.3,3,2)\) | 4 | 2.203e-03 | * | 2.203e-03 | * |
| | 8 | 1.351e-04 | 4.0277 | 1.351e-04 | 4.0277 |
| | 16 | 8.406e-06 | 4.0064 | 8.406e-06 | 4.0064 |
| | 32 | 5.382e-07 | 3.9653 | 5.380e-07 | 3.9657 |

Table 5: CPU in seconds of fast scheme (F-S) and direct scheme (D-S) with \(M = 100\).

| \((r, \alpha_0, \alpha_1, \lambda_0, \lambda_1)\) | \(N\) | Example 4.1 | Example 4.2 |
|---|---|---|---|
| | | F-S | D-S | F-S | D-S |
| \((1,0.8,0.5,1,3)\) | 4096 | 5.50 | 24.84 | 6.53 | 23.95 |
| | 8192 | 10.84 | 90.24 | 12.71 | 92.54 |
| | 16384 | 23.32 | 348.53 | 25.83 | 357.59 |
| | 32768 | 43.99 | 1388.85 | 52.60 | 1435.65 |
| \((2,0.5,0.3,1,2)\) | 4096 | 5.93 | 24.06 | 6.31 | 24.39 |
| | 8192 | 12.40 | 92.38 | 12.90 | 91.20 |
| | 16384 | 21.51 | 354.70 | 24.80 | 355.02 |
| | 32768 | 45.09 | 1398.59 | 50.94 | 1423.70 |

the sum-of-exponentials technic, we derive a fast compact scheme for (6)-(9) via \(L_1\) formula on graded meshes. Meanwhile, Neumann boundary conditions are carefully handled. The unconditional stability and convergence of the proposed scheme are analyzed by energy method based on \(L_2\) norm. At last, numerical experiments are carried out to confirm our theoretical results.

Acknowledgment

The authors would like to thank the editor and reviewers for their constructive comments and suggestions, which helped the authors to improve the quality of the paper significantly.

References

[1] W. Wyss, The fractional diffusion equation, Journal of Mathematical Physics 27 (1986) 2782–2785.
[2] N. Laskin, Fractional market dynamics, Physica A: Statistical Mechanics and its Applications 287 (2000) 482–492.
[3] T. Chung, Computational fluid dynamics, Cambridge, UK: Cambridge University Press, 2002.
[4] M. Stynes, E. O’Riordan, J. Gracia, Error analysis of a finite difference method on graded meshes for a time-fractional diffusion equation, SIAM Journal Numerical Analysis 55 (2017) 1057–1079.
[5] Y. Yan, Z. Sun, J. Zhang, Fast evaluation of the Caputo fractional derivative and its applications to fractional diffusion equations: a second-order scheme, Communications in Computational Physics 22 (2017) 1028–1048.
[6] Z. Sun, C. Gao, Fractional Differential Equation. Finite Difference Methods, Science Press Beijing & de Cruyter, Berlin/Boston, 2020.
[7] H. Liao, D. Li, J. Zhang, Sharp error estimate of the nonuniform \(L_1\) formula for linear reaction-subdiffusion equations, SIAM Journal Numerical Analysis 56 (2018) 1112–1133.
[8] W. Deng, Smoothness and stability of the solutions for nonlinear fractional differential equations, Nonlinear Analysis: Theory, Methods & Applications 72 (2010) 1766–1777.
[9] J. Jia, H. Wang, Fast finite difference methods for space-fractional diffusion equations with fractional derivative boundary conditions, Journal of Computational Physics 293 (2015) 359–369.

[10] C. Li, Q. Yi, A. Chen, Finite difference methods with non-uniform meshes for nonlinear fractional differential equations, Journal of Computational Physics 316 (2016) 614–631.

[11] D. Li, J. Zhang, Z. Zhang, Unconditionally optimal error estimates of a linearized galerkin method for nonlinear time fractional reaction-subdiffusion equations, Journal of Scientific Computing 76 (2018) 848–866.

[12] D. Li, W. Sun, C. Wu, A novel numerical approach to time-fractional parabolic equations with nonsmooth solutions, Numerical Mathematics: Theory, Methods and Applications 14 (2021) 355–376.

[13] D. Cen, Z. Wang, Y. Mo, Second order difference schemes for time-fractional KdV-Burgers equation with initial singularity, Applied Mathematics Letters 112 (2021) 108829.

[14] Z. Wang, D. Cen, Y. Mo, Sharp error estimate of a compact L1-ADI scheme for the two-dimensional time-fractional integro-differential equation with singular kernels, Applied Numerical Mathematics 159 (2021) 190–203.

[15] P. Lu, S. Vong, A high-order method with a temporal nonuniform mesh for a time-fractional Benjamin-Bona-Mahony equation, Journal of Scientific Computing 80 (2019) 1607–1628.

[16] E. Hesameddini, A. Rahimi, E. Asadollahifard, On the convergence of a new reliable algorithm for solving multi-order fractional differential equations, Communications in Nonlinear Science and Numerical Simulation 34 (2016) 154–164.

[17] F. Liu, M. Meerschaert, R. McGough, P. Zhuang, Q. Liu, Numerical methods for solving the multi-term time-fractional wave-diffusion equation, Fractional Calculus and Applied Analysis 16 (2013) 9–25.

[18] L. Qiao, D. Xu, Z. Wang, Orthogonal spline collocation method for the two-dimensional time fractional mobile-immobile equation, J. Appl. Math. Comput. (2021). https://doi.org/10.1007/s12190-021-01661-3

[19] B. Jin, R. Lazarov, Y. Liu, Z. Zhou, The Galerkin finite element method for a novel 2D multi-term time-fractional mixed sub-diffusion and wave-equation on convex domains, Communications in Nonlinear Science and Numerical Simulation 70 (2019) 354–371.

[20] F. Zeng, Z. Zhang, G. Karniadakis, Second-order numerical methods for multi-term fractional differential equations: Smooth and non-smooth solutions, Computer Methods in Applied Mechanics and Engineering 327 (2017) 478–502.

[21] H. Sun, X. Zhao, Z. Sun, The temporal second order difference schemes based on the interpolation approximation for solving the time multi-term and distributed-order fractional sub-diffusion equations, Journal of Scientific Computing 73 (2017) 93–121.

[22] L. Feng, F. Liu, I. Turner, Finite difference/finite element method for a novel 2D multi-term time-fractional mixed sub-diffusion and diffusion-wave equation with singular kernels, Applied Numerical Mathematics 159 (2021) 190–203.

[23] X. Hu, L. Zhang, A new implicit compact difference scheme for the fourth-order fractional diffusion-wave equation, Communications in Nonlinear Science and Numerical Simulation 70 (2019) 467–498.

[24] L. Qiao, Z. Wang, Compact finite difference scheme for the fourth-order fractional sub-diffusion equation, Advances in Applied Mathematics and Mechanics 6 (2014) 419–435.

[25] P. Zhang, H. Pu, A second-order compact difference scheme for the fourth-order fractional sub-diffusion equation, Numerical Algorithms 76 (2017) 573–598.

[26] Z. Yao, Z. Wang, A compact difference scheme for fourth-order fractional sub-diffusion equations with Neumann boundary conditions, Applied Numerical Mathematics 129 (2018) 58–70.

[27] M. Cui, Compact finite difference method for the fractional diffusion equation, Journal of Computational Physics 228 (2009) 7792–7804.

[28] J. Li, Z. Sun, X. Zhao, A three level linearized compact difference scheme for the Cahn-Hilliard equation, Science China Mathematics 55 (2012) 805–826.

[29] J. Ren, Z. Sun, Numerical algorithm with high spatial accuracy for the fractional diffusion-wave equation with Neumann boundary conditions, Journal of Scientific Computing 56 (2013) 381–408.

[30] J. Ren, H. Chen, J. Zhang, Z. Zhang, Error analysis of a fully discrete scheme for a multi-term time fractional diffusion equation with singularly (in Chinese), Scientia Sinica Mathematica 50 (2020) 1–18.

[31] J. Shen, Z. Sun, R. Du, Fast finite difference schemes for time-fractional diffusion equations with a weak singularity at the initial time, East Asian Journal on Applied Mathematics 8 (2018) 834–858.