THE USE OF FINITE DIFFERENCE/ELEMENT APPROACHES FOR SOLVING THE TIME-FRACTIONAL SUBDIFFUSION EQUATION

FANHAI ZENG, CHANGPIN LI, FAWANG LIU, AND IAN TURNER

Abstract. In this paper, two finite difference/element approaches for the time-fractional subdiffusion equation with Dirichlet boundary conditions are developed, in which the time direction is approximated by the fractional linear multistep method and the space direction is approximated by the finite element method. The two methods are unconditionally stable and convergent of order \( O(\tau^q + h^r + 1) \) in the \( L^2 \) norm, where \( q = 2 - \beta \) or 2 when the analytical solution to the subdiffusion equation is sufficiently smooth, \( \beta \) (\( 0 < \beta < 1 \)) is the order of the fractional derivative, \( \tau \) and \( h \) are the step sizes in time and space, respectively, and \( r \) is the degree of the polynomial space. The corresponding schemes for the subdiffusion equation with Neumann boundary conditions are presented as well, where the stability and convergence are shown. Numerical examples are provided to verify the theoretical analysis. Comparisons between the algorithms derived in this paper and the existing algorithms are given, which show that our numerical schemes exhibit better performances than the existing ones.

Key words. finite difference method, finite element method, fractional derivative, subdiffusion, unconditional stability, convergence

AMS subject classifications. 26A33, 65M06, 65M12, 65M15, 35R11

DOI. 10.1137/130910865

1. Introduction. In recent years, fractional calculus has become a hot topic due to its wide applications in science and engineering, such as in physics, material science, control, and biology; see, for example, [1, 2, 23, 26, 33, 37, 44, 46, 56]. In physics, fractional derivatives are used to model anomalous diffusion (i.e., subdiffusion and superdiffusion), where particles spread in a power-law manner [37, 45]. It is important to seek efficient methods to solve fractional differential equations (FDEs). Of course, there are some analytical methods available for solving some linear and special FDEs, such as the Fourier transform method, the Laplace transform method, the Mellin transform method, and the Green function method [44]. In real applications, the analytical methods do not work well on the majority of FDEs. Therefore, it is of great importance to find efficient and reliable numerical methods to solve these FDEs.

Similar to the numerical methods for the classical differential equations, there also exist finite difference methods [3, 10, 12, 13, 20, 24, 26, 27, 31, 34, 51, 52, 54, 55, 62], finite element methods (FEMs) [16, 48, 59, 61], and spectral methods [6, 25, 28] for numerically solving the FDEs. There are other techniques such as the
homotopy perturbation method, the variational method, and the matrix approach; see, for example, [38, 45, 49, 50, 53].

Fractional kinetic equations of diffusion, diffusion-advection, and Fokker–Planck type are presented as a useful approach for the description of transport dynamics in complex systems that are governed by anomalous diffusion and nonexponential relaxation patterns [37].

This paper mainly considers the time-fractional subdiffusion equation [37]

\[
\begin{aligned}
\begin{cases}
\partial_t^{\beta} u + \mu \partial_x^2 u + f(x, t), & (x, t) \in I \times (0, T], I = (a, b), T > 0, \\
\partial_t u = 0, & (x, t) \in \partial I \times (0, T],
\end{cases}
\end{aligned}
\]  

\(1.1\)

where \(0 < \beta < 1, \mu > 0, \partial_x^l = \frac{\partial^l}{\partial x^l}, l = 1, 2, \ldots, \) and \(\partial_t^{\beta}\) is the \(\beta\)-th-order Caputo derivative operator defined by

\[
\begin{aligned}
\partial_t^{\beta} u(x, t) = D_{0,t}^{-(1-\beta)} \left[ \partial_t u(x, t) \right] = \frac{1}{\Gamma(1-\beta)} \int_0^t (t-s)^{-\beta} \partial_s u(x, s) \, ds,
\end{aligned}
\]  

\(1.2\)

in which \(D_{0,t}^\beta\) is the fractional integral operator defined by [44]

\[
\begin{aligned}
D_{0,t}^{\beta} u(x, t) = RL D_{0,t}^{-\beta} u(x, t) = \frac{1}{\Gamma(\beta)} \int_0^t (t-s)^{\beta-1} u(x, s) \, ds, \quad \beta > 0.
\end{aligned}
\]  

\(1.3\)

The order \(\beta \in (0, 1)\) in the Caputo derivative in (1.1) indicates a subdiffusion equation. Another commonly used fractional derivative is the Riemann–Liouville derivative. The \(\beta\)-th-order Riemann–Liouville derivative operator \(RL D_{0,t}^\beta\) is defined by [44]

\[
\begin{aligned}
RL D_{0,t}^\beta u(x, t) &= \partial_t \left[ D_{0,t}^{-(1-\beta)} u(x, t) \right] \\
&= \frac{1}{\Gamma(1-\beta)} \frac{\partial}{\partial t} \left[ \int_0^t (t-s)^{-\beta} u(x, s) \, ds \right], \quad \beta \in (0, 1).
\end{aligned}
\]

If \(u(x, t)\) is suitably smooth in time, then we have the following relationship [44]:

\[
RL D_{0,t}^{\beta} \left[ u(x, t) - u(x, 0) \right] = C D_{0,t}^{\beta} u(x, t).
\]

We can also obtain the following relation [44]:

\[
RL D_{0,t}^{1-\beta} \left[ C D_{0,t}^{\beta} u(x, t) \right] = RL D_{0,t}^{1-\beta} \left[ RL D_{0,t}^\beta \left( u(x, t) - u(x, 0) \right) \right] = \partial_t u(x, t).
\]

Therefore, if we apply the operator \(RL D_{0,t}^{1-\beta}\) to both sides of (1.1) and impose suitably smooth conditions on \(u(x, t)\), then (1.1) can be transformed into the following equivalent form [20, 37]:

\[
\begin{aligned}
\begin{cases}
\partial_t u = \mu RL D_{0,t}^{1-\beta} \partial_x^2 u + g(x, t), & (x, t) \in I \times (0, T], I = (a, b), T > 0, \\
\partial_t u = 0, & (x, t) \in \partial I \times (0, T],
\end{cases}
\end{aligned}
\]  

\(1.4\)

where \(0 < \mu < 1, \partial_x^l = \frac{\partial^l}{\partial x^l}, l = 1, 2, \ldots, \) and \(\partial_t^{\beta}\) is the \(\beta\)-th-order Riemann–Liouville derivative operator defined by [44]

\[
\begin{aligned}
\partial_t^{\beta} u(x, t) = D_{0,t}^{-(1-\beta)} \left[ \partial_t u(x, t) \right] = \frac{1}{\Gamma(1-\beta)} \int_0^t (t-s)^{-\beta} \partial_s u(x, s) \, ds,
\end{aligned}
\]  

\(1.2\)

in which \(D_{0,t}^\beta\) is the fractional integral operator defined by [44]

\[
\begin{aligned}
D_{0,t}^{\beta} u(x, t) = RL D_{0,t}^{-\beta} u(x, t) = \frac{1}{\Gamma(\beta)} \int_0^t (t-s)^{\beta-1} u(x, s) \, ds, \quad \beta > 0.
\end{aligned}
\]  

\(1.3\)
where $g(x,t) = RL D_{t}^{1-\beta} f(x,t)$. The derivative order $1-\beta \in (0,1)$ in the Riemann–Liouville derivative in (1.4) also indicates a subdiffusion equation.

Throughout this paper, we suppose that (1.1) has a unique solution: the analytical solution $u$, the source term $f$, the initial data $\phi_0$, and the boundary conditions satisfy the suitable smooth conditions to meet the theoretical analysis when needed.

For the subdiffusion equation (1.4), there are limited works on the numerical solutions to such subdiffusion problems. An earlier work can be found in [24], where the Riemann–Liouville derivative was approximated by the L1 method. Yuste and Acedo [54] proposed an explicit finite difference method for the subdiffusion equation with first-order accuracy in time. A class of fractional weighted average finite difference methods were developed in [55] by Yuste, in which the stability was investigated by using the fractional von Neumann stability analysis. Chen et al. [7] developed an implicit algorithm for the subdiffusion equation with first-order accuracy in time. Cui [10] derived a compact implicit finite difference method for (1.4) with first-order in time and fourth-order in space. In 2008, Zhuang et al. [62] developed a new implicit numerical method for the subdiffusion equation (1.4) with first-order in time. It appears that until now no strict analysis has been developed for this improved algorithm. In [57], two Crank–Nicolson-type difference schemes were proposed to solve (1.4). An implicit finite-difference Crank–Nicolson method was also developed in [40] with spatial discretization carried out by the FEM. A piecewise-linear, discontinuous Galerkin method for the time discretization of (1.4) was presented in [41] with space being discretized by the FEM, and uniform convergence and superconvergence were investigated in [42] and [43], respectively. There are other related works, such as the convolution quadrature, which was used to study a class of integro-partial differential equations [9]. The Laplace transform method with high accuracy by numerical quadrature was used for the time discretization of (1.4) in [36]. The finite difference methods for the modified fractional diffusion equation and the fractional cable equation were studied in [29, 30]. There are also numerical methods for high-dimensional fractional diffusion equations; see, for example, [8, 11, 35, 58, 63]. For more details, see the recent review article [26].

This paper aims to construct numerical methods with high-order accuracy in time to solve (1.1). As is known, some studies have been carried out on the numerical simulations of the subdiffusion equation (1.1). The commonly used approach to time discretization is the L1 method with the convergence order $2-\beta$. The space coordinate of (1.1) is often discretized by the finite difference method or Galerkin method. Readers can refer to [20, 21, 22, 28, 47, 60] for further details. In [21, 22, 28], the FEM and the spectral method were used in space, and the lump mass Galerkin FEM was also studied in [22]. The numerical methods for the subdiffusion equation with Neumann boundary conditions were presented in [24, 47, 60], in which the L1 method is used to approximate the time fractional derivative.

In this paper, we construct two kinds of time discretization approaches to solve the subdiffusion equation (1.1) with the spatial discretization performed by the FEM, in which the time fractional derivative is approximated by the fractional linear multistep method (FLMM). We give rigorous stability and convergence analysis for the established methods, which shows that the two methods are unconditionally stable with $q$th-order accuracy in time, where $q$ is related to the smoothness of the analytical solution. Theoretical analysis shows that $q = 2 - \beta$ if the analytical solution $u(x,t)$ is twice differentiable in time and $\partial_t u(x,0) \neq 0$, and $q = 2$ if $u(x,t)$ is twice differentiable in time and $\partial_t u(x,0) = 0$. Even if $u(x,t)$ is not sufficiently smooth in time, the present
methods still show second-order accuracy in time if \(C \cdot D^\beta_a u(x, t) = t^\mu \tilde{u}(x, t)\), where \(\mu \geq 1\) and \(\tilde{u}(x, t)\) is sufficiently smooth in time. The optimal error estimates in space are also derived. We also present corresponding FEMs for the subdiffusion equation (1.1) with Neumann boundary conditions; the stability and convergence are also presented. Numerical examples are presented to verify the theoretical analysis. We find that the numerical results exhibit somewhat better performance than the theoretical results. Comparisons are made between the derived algorithms in this paper and existing algorithms [17, 20, 21, 47, 57, 60], which show that our algorithms are much better than the existing ones in the numerical experiments.

The remainder of this paper is outlined as follows. In section 2, some necessary notation and lemmas are introduced. In section 3, the two FEMs for the subdiffusion equation (1.1) are established, and the stability and error estimate are given. Afterward, the two corresponding FEMs for the subdiffusion equation (1.1) with Neumann boundary conditions are also constructed. The numerical results are presented in section 4, and the conclusion is included in the last section.

2. Preliminaries. In this section, we introduce some notation and lemmas that are needed in the following sections.

Let \(I = (a, b)\) be a finite domain, and denote by \((\cdot, \cdot)\) the inner product defined on the space \(L^2(I)\) with the \(L^2\) norm \(\| \cdot \|\) and the maximum norm \(\| \cdot \|_{\infty}\). Denote \(H^r(I)\) and \(H^r_0(I)\) as the commonly used Sobolev spaces with the norm \(\| \cdot \|_r\) and seminorm \(\| \cdot \|_r\), respectively. Define \(P_r(I)\) as the space of polynomials defined on \(I\) with degree no greater than \(r\), \(r \in \mathbb{Z}^+\). Let \(S_h\) be a uniform partition of \(I\), which is given by

\[a = x_0 < x_1 < \cdots < x_N - 1 < x_N = b, \quad N \in \mathbb{Z}^+.
\]

Denote \(h = (b - a)/N = x_i - x_{i-1}\) and \(I_i = [x_{i-1}, x_i]\) for \(i = 1, 2, \ldots, N\). We define the finite element space \(X_h^r\) as the set of piecewise polynomials with degree at most \(r\) \((r \geq 1)\) on the mesh \(S_h\), which can be expressed by

\[X_h^r = \{ v : v|_{I_i} \in P_r(I_i), v \in C(I) \}.
\]

Introduce the piecewise interpolation operator \(I_h : C(\bar{I}) \rightarrow X_h^r\) as

\[I_h u|_{I_i} = \sum_{k=0}^{r} u(x_i^k) F_k^r(x), \quad u \in C(\bar{I}),
\]

where \(F_k^r(x)\) are Lagrangian basis functions defined by

\[F_k^r(x) = \prod_{i=0,i \neq k}^{r} \frac{x - x_i^k}{x_i^k - x_i}, \quad i = 1, 2, \ldots, N,
\]

and \(\{x_i^k, k = 0, 1, \ldots, r\}\) are the interpolation nodes on the interval \(I_i\) with \(x_0^k = x_{i-1}\) and \(x_r^k = x_i\).
Define \( \varphi^i (i = 0, 1, \ldots, N) \) and \( \varphi_k^i (k = 1, 2, \ldots, r - 1; i = 1, 2, \ldots, N) \) as

\[
\varphi_k^i(x) = \begin{cases} 
F_k^i(x), & x \in [x_{i-1}, x_i], \quad k = 1, 2, \ldots, r - 1, i = 1, \ldots, N, \\
0, & \text{otherwise},
\end{cases}
\]

\[
\varphi^i(x) = \begin{cases} 
F_i^1(x), & x \in [x_{i-1}, x_i], \quad i = 1, \ldots, N - 1, \\
F_0^{i+1}(x), & x \in [x_i, x_{i+1}], \quad i = 1, \ldots, N - 1, \\
0, & \text{otherwise},
\end{cases}
\]

\[
\varphi^0(x) = \begin{cases} 
F_0^1(x), & x \in [x_0, x_1], \\
0, & \text{otherwise},
\end{cases}
\]

\[
\varphi^N(x) = \begin{cases} 
F_N^N(x), & x \in [x_{N-1}, x_N], \\
0, & \text{otherwise}.
\end{cases}
\]

Let \( X_{h0}^r = X_h \cap H_0^1(I) \). Then the spaces \( X_{h0}^r \) and \( X_h^r \) can be expressed as

\[
X_{h0}^r = \text{span} \left\{ \varphi_k^i, k = 1, 2, \ldots, r - 1, i = 1, 2, \ldots, N \right\} \cup \left\{ \varphi^i, i = 1, 2, \ldots, N - 1 \right\},
\]

\[
X_h^r = \text{span} \left\{ \varphi_k^i, k = 1, 2, \ldots, r - 1, i = 1, 2, \ldots, N \right\} \cup \left\{ \varphi^i, i = 0, 1, \ldots, N \right\}.
\]

The basis functions \( \{ \varphi_k^1 \} \cup \{ \varphi^i \} \) will be used in the numerical simulation with grid points \( x_k^i = x_0 + kh/r, k = 0, 1, \ldots, r \).

The orthogonal projection operator \( \Pi_{h}^{1,0} : H_0^1(I) \rightarrow X_{h0}^r \) is defined as

\[
(\partial_t (u - \Pi_{h}^{1,0} u), \partial_t v) = 0, \quad u \in H_0^1(I) \forall v \in X_{h0}^r.
\]

Next, we introduce the properties of the projector \( \Pi_{h}^{1,0} \) and interpolation operator \( I_h \) that will be used later on.

**Lemma 2.1** (see [5]). Let \( m, r \in \mathbb{Z}^+, \ r \geq 1, \) and \( u \in H^m(I) \cap H_0^1(I) \). If \( 1 \leq m \leq r + 1 \), then there exists a positive constant \( C \) independent of \( h \) such that

\[
\| u - \Pi_{h}^{1,0} u \| \leq C h^{m-1} \| u \|_m, \quad l = 0, 1.
\]

**Lemma 2.2** (see p. 108 in [4]). Let \( m, r \in \mathbb{Z}^+, \ r \geq 1, \) and \( u \in H^m(I) \). If \( 0 \leq m \leq r + 1 \), then there exists a positive constant \( C \) independent of \( h \) such that

\[
\| u - I_h u \| \leq C h^m \| u \|_m.
\]

3. The numerical schemes. In this section, we first present the finite difference/element methods for the subdiffusion equation (1.1), and we prove the stability and convergence. Then the corresponding finite difference/element methods with Neumann boundary conditions are analyzed.

Let \( \tau \) be the time step size and \( n_T \) be a positive integer with \( \tau = T/n_T \) and \( t_n = n\tau \) for \( n = 0, 1, \ldots, n_T \). For function \( u(x,t) \in C([0,T]; L^2(I)) \), denote \( u^n = u^n(\cdot) = u(\cdot, t_n) \).

3.1. Semidiscrete finite difference approximations in time. In [32], Lubich proposed FLMMs to discretize the fractional integral and Riemann–Liouville derivative. The \( p \)-th order FLMMs for the fractional integral read as
THE TIME-FRACTIONAL SUBDIFFUSION EQUATION

(3.1) \[ D_{0,t}^{-\beta} u(t) \big|_{t=t_n} = \tau^\beta \sum_{k=0}^n \omega_{n-k}^{(\beta)} u(t_k) + \tau^\beta \sum_{k=0}^n u_{n,k}^{(\beta)} u(t_k) + O(\tau^p), \]

where \( \{\omega_k^{(\beta)}\} \) can be the coefficients of Taylor expansions of the generating function

(3.2) \[ w^{(\beta)}(z) = \left[ \sum_{j=1}^p \frac{1}{j!} (1-z)^j \right]^{-\beta}, \quad p = 1, 2, \ldots, 6, \]

(3.3) \[ w^{(\beta)}(z) = (1-z)^{-\beta} \left[ \gamma_0 + \gamma_1 (1-z) + \gamma_2 (1-z)^2 + \cdots + \gamma_{p-1} (1-z)^{p-1} \right], \]

(3.4) \[ w^{(\beta)}(z) = \left( \frac{1 + z}{2 - z} \right)^\beta, \]

in which \( \{\gamma_k\} \) in (3.3) satisfy the following relation:

\[ \left( \frac{\ln z}{z - 1} \right)^{-\beta} = \sum_{k=0}^\infty \gamma_k (1-z)^k. \]

One can obtain \( \gamma_0 = 1, \gamma_1 = -\beta/2; \) see also [18, 19] for more details. The starting weights \( \{w_{n,k}^{(\beta)}\} \) are chosen such that the asymptotic behavior of the function \( u(t) \) near the origin are taken into account [15]. In this paper, we will use (3.1) with the term \( \tau^\beta \sum_{k=0}^n w_{n,k}^{(\beta)} u(t_k) \) being dropped for the time discretization of (1.1). Therefore, the integer number \( s \) and the starting weights \( w_{n,k}^{(\beta)} \) in (3.1) are independent of the numerical schemes for the subdiffusion equation (1.1), so we omit the corresponding details of \( s \) and \( w_{n,k}^{(\beta)} \); see [15, 18, 19, 32] for more detailed information.

The FLMM (3.1) has second-order accuracy if the generating function (3.4) is used. The FLMMs (3.1) with the generating functions (3.2), (3.3), and (3.4) are called the fractional backward difference formulas, the generalized Newton–Gregory formulas, and the fractional trapezoidal rule, respectively [32].

Remark 3.1. \( \beta \) in (3.1) can be negative. In this case, (3.1) is just the \( p \)th-order FLMMs for the fractional derivative \( RL \, D_{0,t}^{-\beta} u(t) \big|_{t=t_n}, \beta < 0. \)

Next, we consider the time discretization for (1.4). Consider the following fractional ordinary differential equation (FODE):

(3.5) \[ CD_{0,t}^{\beta} y(t) = G(t, y(t)), \quad y(0) = y_0, \quad 0 < \beta < 1. \]

The above FODE is equivalent to the Volterra integral equation

(3.6) \[ y(t) - y_0 = \frac{1}{\Gamma(\beta)} \int_0^t (t-s)^{\beta-1} G(s, y(s)) \, ds = D_{0,t}^{-\beta} G(t, y(t)) \]

in the sense that a continuous function is a solution of (3.5) if and only if it is a solution of (3.6); see Lemma 2.3 in [14].
Let \( y_n \) be the approximate solution to \( y(t_n) \). Using (3.1), one can obtain the convolution quadrature with \( p \)-th order accuracy for (3.6) below [18]:

\[
y_n - y_0 = \tau^\beta \sum_{k=0}^s w_{n,k}^{(\beta)} G(t_k, y_k) + \tau^\beta \sum_{k=0}^n \omega_{n-k}^{(\beta)} G(t_k, y_k). \tag{3.7}
\]

We are now in a position to introduce the following two lemmas.

**Lemma 3.2 (see [32]).** If \( y(t) = t^{\nu-1}, \nu > 0 \), then

\[
D_{0,t}^{-\beta} y(t) \big|_{t=t_n} = \tau^\beta \sum_{k=0}^n \omega_{n-k}^{(\beta)} y(t_k) + O(\tau^p) + O(\tau^\nu),
\]

where \( \omega_{n-k}^{(\beta)} \) can be the coefficients of the Taylor series of the generating functions defined as (3.2)–(3.4), and \( p = 2 \) if (3.4) is used.

**Lemma 3.3.** Suppose that \( 0 < \beta < 1, y(t) \in C^m([0,T]), m > 0 \), is a positive integer. Then

\[
c D_{0,t}^\beta y(t) = \sum_{k=0}^{m-1} \frac{y^{(k)}(0)}{\Gamma(k+1-\beta)} t^{k-\beta} + c D_{0,t}^{\beta-m} y^{(m)}(t).
\]

**Proof.** By Taylor series expansion, one has

\[
y(t) = \sum_{k=0}^{m-1} \frac{y^{(k)}(0)}{\Gamma(k+1)} t^k + \frac{1}{\Gamma(m)} \int_0^t (t-s)^{m-1} y^{(m)}(s) \, ds = \sum_{k=0}^{m-1} \frac{y^{(k)}(0)}{\Gamma(k+1)} t^k + D_{0,t}^{\beta-m} y^{(m)}(t).
\]

Applying the fractional derivative operator \( c D_{0,t}^\beta \) to both sides of the above equation yields

\[
c D_{0,t}^\beta y(t) = \sum_{k=0}^{m-1} \frac{y^{(k)}(0)}{\Gamma(k+1)} c D_{0,t}^\beta t^k + c D_{0,t}^{\beta-m} y^{(m)}(t)
= \sum_{k=0}^{m-1} \frac{y^{(k)}(0)}{\Gamma(k+1-\beta)} t^{k-\beta} + D_{0,t}^{\beta-m} y^{(m)}(t).
\]

The proof is completed. \( \Box \)

Next we present the discretization for (3.5) that will be used for the time discretization of (1.1). We omit the term \( \sum_{k=0}^s w_{n,k}^{(\beta)} G(t_k, y_k) \) in (3.7) or apply Lemma 3.2 to \( D_{0,t}^{-\beta} G(t, y(t)) \) in (3.6) to obtain the following convolution quadrature:

\[
y_n - y_0 = \tau^\beta \sum_{k=0}^n \omega_{n-k}^{(\beta)} G(t_k, y_k). \tag{3.8}
\]

Now, we analyze the convergence order of (3.8), which we show is \( q \). Suppose that the solution \( y(t) \) to (3.5) is sufficiently smooth, say, \( y(t) \in C^2[0,T] \). Then, by Lemma 3.3, \( G(t, y(t)) \) can be written in the form \( G(t, y(t)) = c D_{0,t}^\beta y(t) = \frac{y^{(0)}(0)}{\Gamma(1-\beta)} t^{1-\beta} + D_{0,t}^{(2-\beta)} y''(t) \). By Lemma 3.2 and Theorem 2.4 in [32], one knows that the convolution
quadrature (3.8) has $q$th-order accuracy if $y \in C^2[0,T]$ and the generating function (3.3) with $p = 2$ or (3.4) is used. Specially, $q = 2$ if $y \in C^2[0,T]$, $y'(0) = 0$; $q = 2 - \beta$ if $y \in C^2[0,T]$, $y'(0) \neq 0$. Even if $y(t)$ is not sufficiently smooth but satisfies $CD^\beta_0 y(t) = t^\mu \tilde{y}(t)$ ($\mu \geq 0$) with $\tilde{y}(t)$ being sufficiently smooth, then the convolution quadrature (3.8) has $q$th-order accuracy, $q = \min\{\mu + 1, 2\}$. Refer to [32] for more details. All these will be illustrated by the numerical examples in the following section.

Next, we give the equivalent form of (3.8) in the following lemma.

**Lemma 3.4.** The discrete convolution quadrature (3.8) can be written into the equivalent form

$$
\sum_{k=0}^{n} \alpha_k (y_{n-k} - y_0) = \tau^\beta \sum_{k=0}^{n} \theta_{n-k} G(t_k, y_k),
$$

where $\alpha_k$ and $\theta_k$ are the coefficients of Taylor expansions of $\alpha(z)$ and $\theta(z)$ satisfying $w^{(\beta)}(z) = \theta(z)/\alpha(z)$, and $w^{(\beta)}(z)$ can be defined by (3.2), (3.3), or (3.4).

**Proof.** We first rewrite $w^{(\beta)}(z) = \theta(z)/\alpha(z)$ into the form

$$
\left( \sum_{k=0}^{\infty} \alpha_k z^k \right) \left( \sum_{k=0}^{\infty} \omega_k^{(\beta)} z^k \right) = \sum_{k=0}^{\infty} \theta_k z^k,
$$

which yields

$$
\theta_m = \sum_{k=0}^{m} \omega_k^{(\beta)} \alpha_{m-k}, \quad m = 0, 1, \ldots, n.
$$

By (3.8), one obtains $y(t_m) - y_0 = \tau^\beta \sum_{k=0}^{m} \omega^{(\beta)}_{m-k} G(t_k, y(t_k)) + \tau^q r_m$, where $|r_m|$ is bounded. Hence, we have

$$
\sum_{m=0}^{n} \alpha_{n-m} (y(t_m) - y_0) = \sum_{m=0}^{n} \alpha_{n-m} \left[ \tau^\beta \sum_{k=0}^{m} \omega^{(\beta)}_{m-k} G(t_k, y(t_k)) \right] + \tau^q \sum_{m=0}^{n} \alpha_{n-m} r_m.
$$

Rearranging the right-hand side of (3.11), using (3.10), letting $r^n = 0$, and replacing $y(t_k)$ with $y_k$ yields the desired result. The proof is completed. \[\Box\]

We can see that $\tau^q \sum_{m=0}^{n} \alpha_{n-m} r_m$ is the local truncation error of (3.9) if $\tau^q r_m$ is the truncation error of (3.8).

Next, we present the time discretization to (1.1). For simplicity, we introduce the following notation:

$$
D^{(\beta)} u^n = \frac{1}{\tau^\beta} \sum_{k=0}^{n} \omega_k (u^{n-k} - u^0) = \frac{1}{\tau^\beta} \left[ \sum_{k=0}^{n} \omega_k u^{n-k} - b_n u^0 \right],
$$

$$
L_1^{(\beta)} u^n = (1/2)^\beta \sum_{k=0}^{n} \omega_k (-1)^k u^{n-k},
$$

$$
L_2^{(\beta)} u^n = (1 - \beta/2) u^n + \frac{\beta}{2} u^{n-1},
$$
where $\omega_k$ and $b_n$ are defined by

\begin{align}
\omega_k &= (-1)^k \left( \frac{\beta}{k} \right) \frac{\Gamma(k - \beta)}{\Gamma(-\beta) \Gamma(k + 1)}, \\
b_n &= \sum_{k=0}^{n} \omega_k = \frac{\Gamma(n + 1 - \beta)}{\Gamma(1 - \beta) \Gamma(n + 1)}, \quad n \geq 0.
\end{align}

Suppose that $u(x,t)$ is smooth enough. Then, by Lemma 3.4, we get two types of time discretization for the subdiffusion equation (1.1). The first approach is to use the generating function (3.4), where $\alpha(z)$ and $\theta(z)$ in Lemma 3.4 can be chosen as $\alpha(z) = (1 - z)\beta$, $\theta(z) = (1 + z)\beta$. The second approach is to use the generating function (3.2) with $p = 2$, where $\alpha(z)$ and $\theta(z)$ in Lemma 3.4 can be chosen as $\alpha(z) = (1 - z)\beta$, $\theta(z) = 1 - \frac{q}{2}(1 - z)$. Hence, we obtain the local truncation error of (3.9) from (3.11), satisfying

$$
\left| \tau^q \sum_{k=0}^{n} \alpha_{n-k} r_k \right| = \tau^{q+\beta} \left| \frac{1}{\tau^z} \sum_{k=0}^{n} \omega_{n-k} r_k \right| \leq C \tau^{q+\beta} \left( \left| \left[ RL D_0^\beta r(t) \right]_{t=n} \right| + \tau \right) \leq C \tau^{q+\beta},
$$

where the Grünwald–Letnikov definition is used. Therefore, the local truncation error of (3.9) is $O(\tau^{q+\beta})$ when the generating function (3.4) or (3.2) with $p = 2$ is used.

Before applying the time discretization of (1.1), we define

$$
q = \begin{cases} 
2 - \beta & \text{if } u(t) \in C^2[0,T], u'(0) \neq 0, \\
2 & \text{if } u(t) \in C^2[0,T], u'(0) = 0,
\end{cases}
$$

where $u(t) = u(x,t)$ is the analytical solution of (1.1).

Next, we provide the two approaches to the time discretization of (1.1), as follows:

- **Time discretization I.** Applying the FLMM (3.9) based on the generating function (3.4) with $\alpha(z) = (1 - z)\beta$, $\theta(z) = (1 + z)\beta$ yields

$$
D^{(\beta)} u^n = \mu L_1^{(\beta)} (\partial_x^2 u^n) + L_1^{(\beta)} f^n + O(\tau^q),
$$

where $q$ is defined by (3.17), and $D^{(\beta)}$ and $L_1^{(\beta)}$ are defined by (3.12) and (3.13), respectively.

- **Time discretization II.** Applying the FLMM (3.9) based on the generating function (3.3) with $p = 2$ and $\alpha(z) = (1 - z)\beta$, $\theta(z) = 1 - \frac{q}{2}(1 - z)$ leads to

$$
D^{(\beta)} u^n = \mu L_2^{(\beta)} (\partial_x^2 u^n) + L_2^{(\beta)} f^n + O(\tau^q),
$$

where $q$ is defined by (3.17), and $D^{(\beta)}$ and $L_2^{(\beta)}$ are defined by (3.12) and (3.14), respectively.

Therefore, the semidiscrete approximations for (1.1) are given as follows:

- **Semidiscrete scheme I.** Find $U^n \in H_0^1(I)$ for $n = 1, 2, \ldots, n_T - 1$ such that

$$
\begin{cases} 
(D^{(\beta)} U^n, v) = -\mu (L_1^{(\beta)} \partial_x U^n, \partial_x v) + (L_1^{(\beta)} f^n, v) & \forall v \in H_0^1, \\
U^0 = \phi_0.
\end{cases}
$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.
3.2. Fully discrete schemes. In this subsection, we present two fully discrete schemes for (1.1) with Dirichlet boundary conditions. From the weak forms (3.20) and (3.21), we present the corresponding fully discrete schemes for (1.1) with Dirichlet boundary conditions (FDS-D) as follows:

- FDS-D I. Find \( u_h^n \in X_{k0}^r \) for \( n = 1, 2, \ldots, n_T - 1 \) such that

\[
(D^{(\beta)} u_h^n, v) = -\mu(L_2^{(\beta)} \partial_x u_h^n, \partial_x v) + (f^n, v) \quad \forall v \in H_0^1,
\]

where \( F^n = I_h(L_1^{(\beta)} f^n) \), and \( D^{(\beta)} \) and \( L_1^{(\beta)} \) are defined by (3.12) and (3.13), respectively.

- FDS-D II. Find \( u_h^n \in X_{k0}^r \) for \( n = 1, 2, \ldots, n_T - 1 \) such that

\[
(D^{(\beta)} u_h^n, v) = -\mu(L_2^{(\beta)} \partial_x u_h^n, \partial_x v) + (f^n, v) \quad \forall v \in H_0^1,
\]

where \( F^n = I_h(L_2^{(\beta)} f^n) \), and \( D^{(\beta)} \) and \( L_1^{(\beta)} \) are defined by (3.12) and (3.13), respectively.

Remark 3.5. If \( \beta = 1 \), the methods (3.22) and (3.23) are reduced to the classical Crank–Nicolson method for the corresponding PDE as follows:

\[
\begin{align*}
(D^{(\beta)} u_h^n, v) &= -\mu(L_2^{(\beta)} \partial_x u_h^n, \partial_x v) + (f^n, v) \quad \forall v \in H_0^1, \\
&= -\mu(L_2^{(1)} \partial_x u_h^n, \partial_x v) + (f^n, v) \quad \forall v \in H_0^1,
\end{align*}
\]

where \( u_{ht}^n \) and \( u_{h}^{n+1/2} \) are given by \( u_{ht}^n = \frac{u_{ht}^{n+1} - u_{ht}^n}{\tau} \) and \( u_{h}^{n+1/2} = \frac{u_{h}^{n+1} + u_{h}^n}{2} \), respectively.

Next, we give a brief method of the implementation strategy for the scheme (3.22); the implementation of the scheme (3.23) is similar. The scheme (3.22) can be rewritten into the following equivalent form:

\[
\sum_{k=0}^{n} \omega_k (u_h^{n-k}, v) - b_n (u_h^0, v)
= -\left( \frac{\tau}{2} \right)^\beta \sum_{k=0}^{n} \omega_k (-1)^k (\partial_x u_h^{n-k}, \partial_x v) + \tau^\beta (F^n, v) \quad \forall v \in X_{k0}^r.
\]

The unknown function \( u_h^n \) can be expressed as

\[
\begin{align*}
u_h^n &= \sum_{j=0}^{N_r} \tilde{u}_j \phi_j (x),
\end{align*}
\]
where
\[
\dot{\phi}_j(x) = \begin{cases} 
\varphi_k^1(x), & j = (i-1)r + k, \ k = 1, 2, \ldots, r-1, i = 1, 2, \ldots, N, \\
\varphi^1(x), & j = ir, \ i = 0, 1, \ldots, N.
\end{cases}
\]

Denote \( \mathbf{u}^n = (\hat{u}_0^n, \hat{u}_1^n, \ldots, \hat{u}_{Nr-1}^n)^T, \hat{\mathbf{u}}^n = (\hat{u}_0^n, \hat{u}_1^n, \ldots, \hat{u}_{N}^n)^T. \) The matrices \( A \in \mathbb{R}^{(N\tau-1) \times (N\tau-1)} \) and \( M, S \in \mathbb{R}^{(N\tau-1) \times (N\tau+1)} \) are given by
\[
(A)_{i,j} = \omega_0 \left( M + (\tau/2)^\beta S \right)_{i,j}, \quad i, j = 1, 2, \ldots, N\tau - 1, \\
(M)_{i,j} = (\phi_i, \dot{\phi}_j), \quad (S)_{i,j} = (\partial_x \phi_i, \partial_x \dot{\phi}_j), \quad i = 1, 2, \ldots, N\tau - 1, \quad j = 0, 1, \ldots, N\tau.
\]

Inserting the unknown function \( u_n^\omega \) defined by (3.25) into (3.24) and letting \( v = \dot{\phi}_j(x), \ j = 1, 2, \ldots, N\tau - 1, \) we obtain the matrix representation of the scheme (3.24) as
\[
A\mathbf{u}^n = \mathbf{b}^n,
\]
in which \( A \) is a tridiagonal (or seven-diagonal) matrix if a linear (or cubic) element is used, \( \mathbf{b}^n \) can be simply calculated by the formula
\[
\mathbf{b}^n = -\sum_{k=1}^{n} \omega_k \left[ M - (\tau/2)^\beta (-1)^k S \right] \hat{\mathbf{u}}^{n-k} + b_n(M\hat{\mathbf{u}}^0) + \tau^\beta \mathbf{F}_1^n - \omega_0(B\mathbf{u}_{\text{bound}}^n),
\]
where \( \mathbf{u}_{\text{bound}}^n = (\hat{u}_0^n, \hat{u}_1^n, \ldots, \hat{u}_N^n)^T, \mathbf{F}_1^n = (u(a, t_n), u(b, t_n))^T, \) \( (\mathbf{F}_1^n, \dot{\phi}_j), j = 1, 2, \ldots, N\tau - 1, \) and \( B \in \mathbb{R}^{(N\tau-1) \times 2} \) satisfies
\[
B_{1,i} = \omega_0 \left[ M_{i,0} + (\tau/2)^\beta S_{i,0} \right], \quad B_{i,2} = \omega_0 \left[ M_{i,N\tau} + (\tau/2)^\beta S_{i,N\tau} \right], \quad i = 1, 2, \ldots, N\tau - 1.
\]

Algorithm 3.1 gives the implementation of FDS-D I (3.22). The implementation of FDS-D II (3.23) is very similar, and we omit the details here.

### 3.3. Stability and convergence

This subsection deals with stability and convergence for the schemes (3.22) and (3.23). Next, we introduce a lemma.

**Lemma 3.6** (see [19]). Let \( \{\omega_k\} \) be given by (3.15). Then we have
\[
\omega_0 = 1, \quad \omega_n < 0, \quad |\omega_{n+1}| < |\omega_n|, \quad n = 1, 2, \ldots;
\]
\[
\omega_0 = -\sum_{k=1}^{\infty} \omega_k > -\sum_{k=1}^{n} \omega_k > 0, \quad n = 1, 2, \ldots;
\]
\[
b_{n-1} = \sum_{k=0}^{n-1} \omega_k = \frac{\Gamma(n-\beta)}{\Gamma(1-\beta)\Gamma(n)} = \frac{n^{-\beta}}{\Gamma(1-\beta)} + O(n^{-1-\beta}), \quad n = 1, 2, \ldots.
\]

Furthermore, \( b_n - b_{n-1} = \omega_n < 0 \) for \( n > 0 \), i.e., \( b_n \leq b_{n-1}. \)

For convenience, we define the norms \( ||| \cdot |||_1 \) and \( ||| \cdot |||_2 \) as
\[
|||u|||_1 = \left( \|u\|^2 + \mu \tau^\beta (1/2)^\beta \|\partial_x u\|^2 \right)^{1/2},
\]
\[
|||u|||_2 = \left( \|u\|^2 + \mu \tau^\beta (2-3\beta/2) \|\partial_x u\|^2 \right)^{1/2}.
\]

The following theorem states that the method (3.22) is unconditionally stable. Unconditional stability here means that the perturbation of the initial value will not
such that
\[
\tau D (3.28) \quad (f \in C(0,T;C(\bar{I})) \quad \text{for} \quad \tau > 0)
\]
with the property of norm, and $v = u^n$.

We rewrite (3.28) as
\[
(3.27)
\]
for time step $n = 1 : n_T$ do

Compute $F_1^n = (\tau / 2)^{\beta} \sum_{k=0}^{n} (-1)^k \omega_k (M f^{n-k})$;

Set $u_0^{\text{bound}} = (u_0^a, u_0^b)^T$, $u_0^n = u(a, t_n)$, $u_b^n = u(b, t_n)$;

Compute
\[
b^n = - \sum_{k=1}^{n} \omega_k (M - (\tau / 2)^{\beta} (-1)^k S) \tilde{u}^{n-k} + b_n(M\tilde{u}^0) + F_1^n - \omega_0 (Bu_0^{\text{bound}});
\]

Solve the system $Au^n = b^n$ to get $u^n$;

Set $\tilde{u}^n = \begin{pmatrix} u_a^n \\ u_b^n \end{pmatrix}$.

end

be amplified, i.e., $\|\tilde{u}^n_h\|_A \leq C\|\tilde{u}^0_h\|_A$, where $\tilde{u}^n_h$ is a perturbation of $u^n_h$, $\|\cdot\|_A$ is a kind of norm, and $C$ is a positive constant independent of $\tau, N$ and $T$.

**Theorem 3.7.** Suppose that $u^n_h$ ($k = 1, 2, \ldots, n_T$) are solutions of (3.22), $f \in C(0,T;C(\bar{I}))$. Then, there exists a positive constant $C$ independent of $n, h$, and $\tau$ such that

\[
(3.27) \\
\|u^n_h\|_1^2 \leq \|u^0_h\|_1^2 + 2\|u^0_h\|^2 + C \max_{0 \leq k \leq n_T} \|f^k\|^2.
\]

Inequality (3.27) means that the method (3.22) is unconditionally stable.

**Proof.** We prove (3.27) by using the mathematical induction method. Letting $v = u^n_h$ in (3.22) yields

\[
\begin{aligned}
(D^{\beta}) u^n_h, u^n_h & = -\mu(L_1^{(\beta)} \partial_x u^n_h, \partial_x u^n_h) + (F_1^n, u^n_h).
\end{aligned}
\]

We rewrite (3.28) as

\[
\begin{aligned}
\|u^n_h\|_1^2 & = (u^n_h, u^n_h) + \mu(\tau/2)^{\beta} (\partial_x u^n_h, \partial_x u^n_h) \\
& = \sum_{k=1}^{n} (b_k - b_{k-1}) \big[ (u_h^{n-k}, u_h^n) - \mu(\tau/2)^{\beta} (-1)^k (\partial_x u_h^{n-k}, \partial_x u_h^n) \big] \\
& \quad + b_n(u_0^n, u_h^n) + \tau^\beta (F_1^n, u_h^n),
\end{aligned}
\]

where the property $b_k - b_{k-1} = \omega_k \leq 0, k > 0$ has been used from Lemma 3.6.
Using (3.29), \( b_k \leq b_{k-1} \), and the Cauchy–Schwarz inequality yields

\begin{equation}
\|u_h^n\|^2 \leq \frac{1}{2} \sum_{k=1}^{n-1} (b_{k+1} - b_k) \left[ \|\nabla_h u_h^{n-k}\|^2 + \|u_h^n\|^2 + \mu(\tau/2)^\beta (\|\partial_x u_h^{n-k}\|^2 + \|\partial_x u_h^n\|^2) \right] + b_n \|u_h^n\|^2 + \frac{\tau^{2\beta}}{b_n} \|F_h^n\|^2 + \frac{b_n}{4} \|u_h^n\|^2
\end{equation}

\begin{equation}
= \frac{1}{2} \sum_{k=1}^{n-1} (b_{k+1} - b_k) \|u_h^{n-k}\|^2 + \frac{\tau^{2\beta}}{b_n} \|F_h^n\|^2 + b_n \|u_h^n\|^2
\end{equation}

\begin{equation}
- \frac{1}{2} b_n \mu(\tau/2)^\beta \|\partial_x u_h^n\|^2.
\end{equation}

It immediately follows that

\begin{equation}
\|u_h^n\|^2 \leq \sum_{k=1}^{n-1} (b_{k+1} - b_k) \|u_h^{n-k}\|^2 + \frac{2\tau^{2\beta}}{b_n} \|F_h^n\|^2 + 2b_n \|u_h^n\|^2.
\end{equation}

By Lemma 3.6, we have \( 1/b_n \leq C_\beta n^\beta \), where \( C_\beta \) is only dependent on \( \beta \). So

\begin{equation}
\frac{\tau^{2\beta}}{b_n} = b_n \frac{T^{2\beta}}{b_n} \leq b_n C_\beta^2 T^{2\beta} \left( \frac{n}{n_T} \right)^2 \leq (C_\beta T)^2 b_n.
\end{equation}

Noticing that \( \sum_{k=0}^{n} |\omega_k| = \omega_0 + \sum_{k=1}^{n} |\omega_k| < 2\omega_0 = 2 \), we have

\begin{equation}
\|F_h^n\| = \left\| \sum_{k=0}^{n} (-1)^k \omega_k I_h f^{n-k} \right\| \leq \sum_{k=0}^{n} |\omega_k| \max_{0 \leq k \leq n_T} \|I_h f^k\| \leq C \max_{0 \leq k \leq n_T} \|f^k\|,
\end{equation}

where we have used \( \|I_h f^k\| \leq \tilde{C} \|f^k\| \), \( \tilde{C} > 0 \), from Lemma 2.2 when \( h \) is suitably small. Denote by \( C = 2(C_1 C_\beta T^\beta)^2 \) and

\[ E = \|u_h^0\|^2 + 2\|u_h^0\|^2 + C \max_{0 \leq k \leq n_T} \|f^k\|^2. \]

Combining (3.31)–(3.33) yields

\begin{equation}
\|u_h^n\|^2 \leq \sum_{k=1}^{n-1} (b_{k+1} - b_k) \|u_h^{n-k}\|^2 + b_n E.
\end{equation}

Setting \( n = 0 \) in (3.34), and noticing that \( \|u_h^0\|^2 \leq E \), one has

\begin{equation}
\|u_h^0\|^2 \leq (1 - b_1) \|u_h^0\|^2 + b_1 E \leq (1 - b_1) E + b_1 E = E.
\end{equation}

Obviously, the inequality (3.27) holds for \( n = 1 \). Suppose that the inequality (3.27) holds for \( 0 \leq n \leq m - 1 \), i.e., \( \|u_h^n\|^2 \leq E \) (\( 0 \leq n \leq m - 1 \)). Next, we just need to prove that the inequality (3.27) still holds for \( n = m \).
Letting $v = u^n_h$ in (3.28) yields (3.34) with $n = m$. Considering the inequality (3.34) with $n = m$ and using the assumption $|||u^n_h|||^2 \leq E$ for $0 \leq n \leq m - 1$, one has

\begin{equation}
|||u^n_h|||^2 \leq \sum_{k=1}^{m}(b_{k-1} - b_k)|||u^{m-k}_h|||^2 + b_mE \leq \sum_{k=1}^{m}(b_{k-1} - b_k)E + b_mE = E,
\end{equation}

which means that (3.27) holds for $n = m$. Hence, (3.27) is true for any $0 \leq n \leq n_T$.

In order to illustrate that the method (3.22) is unconditionally stable, we suppose that $u^n_h$ has the perturbation $\tilde{u}_n^h$, so we can get the perturbation equation as

\begin{equation}
(D^{(\beta)}\tilde{u}_n^h, v) = -\mu(L_1^{(\beta)} \partial_x\tilde{u}_n^h, \partial_x v),
\end{equation}

from which we observe that

\begin{equation}
|||\tilde{u}_n^h|||^2 \leq |||\tilde{u}_0^n|||^2 + 2|||\tilde{u}_0^n|||^2 \leq 3|||u_0^n|||^2
\end{equation}

by letting $f^k = 0$ and replacing $u^n_h$ with $\tilde{u}_n^h$ in (3.27). Hence, the method (3.22) is unconditionally stable, which completes the proof. □

Next, we consider the convergence analysis for the scheme (3.22).

Suppose that $u \in C^2(0, T; H^m(I) \cap H^n_0(I)), m > r$. Denote $u_* = \Pi_{h+1}^{1,0} u$, $e = u_* - u_h$, and $\eta = u - u_*$. Noticing that $(\partial_x\eta, \partial_x v) = 0$ from (2.1), we obtain the error equation for (3.22),

\begin{equation}
(D^{(\beta)} e^n, v) = -\mu(L_1^{(\beta)} \partial_x e^n, \partial_x v) + (G^n, v) \quad \forall v \in X^r_{0, h},
\end{equation}

where $G^n = \sum_{i=1}^3 G^n_i$ and

\begin{equation}
G_1^n = O(\tau^q), \quad G_2^n = F^n - \Pi_{h+1}^{1,0} F^n = L_1^{(\beta)}(f^n - \Pi_{h+1}^{1,0} f^n), \quad G_3^n = -D^{(\beta)}\eta^n.
\end{equation}

By Theorem 3.7 and (3.37), we obtain the following convergence theorem.

**Theorem 3.8.** Suppose that $r \geq 1$, $u$ and $u^n_h (1 \leq n \leq n_T)$ are the solutions to (1.1) and (3.22), respectively. If $m \geq r + 1$, $u \in C^2(0, T; H^m(I) \cap H^n_0(I)), f \in C(0, T; H^m(I)), \text{ and } \phi_0 \in H^m(I)$, then there exists a positive constant $C$ independent of $u, h$, and $\tau$ such that

\begin{equation}
|||u^n_h - u(t_n)||| \leq C(\tau^q + h^{r+1}).
\end{equation}

**Proof.** According to Theorem 3.7, we only need to estimate

\begin{equation}
|||e^n|||^2 + 2|||\tilde{e}_0^n|||^2 + \max_{0 \leq k \leq n_T} \{|||G_1^n|||^2 + |||G_2^n|||^2 + |||G_3^n|||^2\}
\end{equation}

to get the error bound. By (3.38) and Lemmas 2.1 and 2.2, we can get the error bounds

\begin{equation}
|||G_1^n|||^2 \leq C\tau^q,
\end{equation}

\begin{equation}
|||G_2^n||| = (1/2)^\beta \left\| \sum_{k=1}^{n}(-1)^k \omega_k (f^{n-k} - I_h f^{n-k}) \right\| \leq (1/2)^\beta \max_{0 \leq n \leq n_T} \| f^n - I_h f^n \| \leq C\tau^{r+1},
\end{equation}

\begin{equation}
|||G_3^n||| = \frac{1}{\tau^q} \left\| - \sum_{k=0}^{n} \omega_k \eta^{n-k} - \eta^0 \right\| \leq C\tau^{r+1},
\end{equation}

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.
where we have used the Grünwald–Letnikov formula, i.e.,
\[
[RLD^\beta_{0,t}(\eta(t) - \eta^0)]_{t=t_n} = \frac{1}{\tau^\beta} \sum_{k=0}^{n} \omega_k \eta^{n-k} - \eta^0 + O(\tau),
\]
which indicates \(|\frac{1}{\tau^\beta} \sum_{k=0}^{n} \omega_k \eta^{n-k} - \eta^0| \leq C\left[|\left[RLD^\beta_{0,t}(\eta(t) - \eta^0)\right]|_{t=t_n} + C \max_{0 \leq t \leq t_n} \{|\eta(t)| + |\eta'(t)| + |\eta''(t)|\}\right\) (see also (3.20) and (3.21) in [32]) that yields a bound for \(G^3_\beta\).

For the initial errors \(e^0\), we have \(e^0 = 0\). Hence, one obtains
\[
\|e^n\| \leq \|e^n\|_{1} \leq C(r^q + h^{r+1}).
\]

By using Lemma 2.1 again, one has
\[
\|u_h^n - u(t_n)\| = \|u_h^n - \Pi^1_0 u^n + \Pi^1_0 u(t_n) - u(t_n)\| \\
\leq \|e^n\| + \|\Pi^1_0 u(t_n) - u(t_n)\| \\
\leq C(r^q + h^{r+1}).
\]

The proof is completed. \(\Box\)

Similar to Theorem 3.7, we can immediately deduce the stability and convergence analysis for the scheme (3.23).

**Theorem 3.9.** Suppose that \(u_h^n (n = 1, 2, \ldots, n_T)\) are solutions to (3.23), \(f \in C(0, T; C(I))\). Then, there exists a positive constant \(C\) independent of \(n, h, \) and \(\tau\) such that
\[
\|u_h^n\|^2 \leq \|u_h^0\|^2 + 2\|u_h^0\|^2 + C \max_{0 \leq k \leq n_T} \|f^k\|^2.
\]

The inequality (3.41) means that method FDS-D II (3.23) is unconditionally stable.

By Theorem 3.9, one obtains the error estimate for the method (3.23).

**Theorem 3.10.** Suppose that \(r \geq 1, u\) and \(u_h^n (1 \leq n \leq n_T)\) are the solutions to (1.1) and (3.23), respectively. If \(m \geq r + 1, u \in C^2(0, T; H^m(I)), f \in C(0, T; H^m(I) \cap H^1_0(I)), \) and \(\phi_0 \in H^m(I)\), then there exists a positive constant \(C\) independent of \(n, h, \) and \(\tau\) such that
\[
\|u_h^n - u(t_n)\| \leq C(r^q + h^{r+1}).
\]

### 3.4. Neumann boundary conditions

This subsection considers the FEMs for the subdiffusion equation as the form (1.1) with Neumann boundary conditions [60]
\[
\begin{align*}
C D_0^\beta u &= \mu \partial_x^2 u + f(x, t), \quad (x, t) \in I \times (0, T], T > 0, \\
u(x, 0) &= \phi_0(x), \quad x \in I, \\
\partial_x u(a, t) &= \partial_x u(b, t) = 0, \quad (x, t) \in \partial I \times (0, T].
\end{align*}
\]

The numerical methods for (3.43) have forms similar to FDS-D I and FDS-D II; we list the two fully discrete schemes with Neumann (FDS-N) boundary conditions below:

- **FDS-N I.** Find \(u_h^n \in X_h^n\) for \(n = 1, 2, \ldots, n_T - 1\) such that
\[
\begin{align*}
(D^{(\beta)} u_h^n, v) &= -\mu (L^1_1 \partial_x u_h^n, \partial_x v) + (L^1_1 I_h f^n, v) \quad \forall v \in X_h^n, \\
u_h^0 &= \Pi^1_0 \phi_0,
\end{align*}
\]

where \(D^{(\beta)}\) and \(L^1_1\) are defined by (3.12) and (3.13), respectively.
THE TIME-FRACTIONAL SUBDIFFUSION EQUATION

4. Numerical examples. In this section, we present several numerical examples. For convenience, we use the interpolation operator \( I_h \) to replace the projectors \( \Pi_{h,2}^1 \) and \( \Pi_{h,0}^1 \) for the computation. We first numerically verify the error estimates and the convergence orders of the FEMs FDS-D I (see (3.22)) and FDS-D II (see (3.23)).

Example 4.1. Consider the following subdiffusion equation [21, 28]:

\[
\begin{cases}
\begin{align*}
\partial_t^\alpha u(x,t) & = \partial_x^2 u + f(x,t), \quad (x,t) \in (0,1) \times (0,1], \\
u(x,0) & = 0, \quad x \in [0,1], \\
u(0,t) & = u(1,t) = 0, \quad t \in (0,1],
\end{align*}
\end{cases}
\tag{4.1}
\]

where \( f(x,t) = \frac{2t^{1-\alpha}}{\Gamma(2-\alpha)} \sin(2\pi x) + 4\pi^2 t^2 \sin(2\pi x) \). The exact solution to (4.1) is

\[
u = t^2 \sin(2\pi x).
\]

Let \( E(\tau, h) = u_h - u \). The convergence orders in time and space in the sense of the \( L^2 \) norm are defined as

\[
\text{order} = \begin{cases}
\frac{\log(\|E(\tau_1, h)/\|E(\tau_2, h)\|)}{\log(\tau_1/\tau_2)} & \text{in time,} \\
\frac{\log(\|E(\tau, h_1)/\|E(\tau, h_2)\|)}{\log(h_1/h_2)} & \text{in space,}
\end{cases}
\tag{4.2}
\]

where \( \tau, \tau_1, \tau_2 (\tau_1 \neq \tau_2) \) and \( h, h_1, h_2 (h_1 \neq h_2) \) are the time and space step sizes, respectively.

The cubic element \( (r = 3) \) is used in this example.

Table 4.1 displays the \( L^2 \) errors at \( t = 1 \) and convergence orders in time and space for the cubic element methods FDS-D I (3.22) and FDS-D II (3.23) with \( \alpha = 0.5 \). Clearly, numerical solutions fit well with the exact solutions, and the second-order convergence in time and fourth-order convergence in space are observed, which is in agreement with the theoretical analysis.
Table 4.1

The $L^2$ errors at $t = 1$ and convergence orders in time for Example 4.1, $\beta = 0.5$.

<table>
<thead>
<tr>
<th>Method</th>
<th>$1/\tau$</th>
<th>$N$</th>
<th>Error</th>
<th>Order</th>
<th>$1/\tau$</th>
<th>$N$</th>
<th>Error</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>FDS-D I</td>
<td>10</td>
<td>100</td>
<td>8.6592e-6</td>
<td>1.9918</td>
<td>1000</td>
<td>20</td>
<td>6.8017e-7</td>
<td>4.0041</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td></td>
<td>3.5097e-7</td>
<td>2.0342</td>
<td></td>
<td>25</td>
<td>2.7835e-7</td>
<td>4.0027</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td></td>
<td>1.5268e-7</td>
<td>2.0175</td>
<td></td>
<td>30</td>
<td>1.3417e-7</td>
<td>4.0019</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td></td>
<td>1.9306e-7</td>
<td>2.0026</td>
<td></td>
<td>40</td>
<td>4.2432e-8</td>
<td>4.0013</td>
</tr>
<tr>
<td>FDS-D II</td>
<td>10</td>
<td>100</td>
<td>1.4875e-5</td>
<td>2.0050</td>
<td>1000</td>
<td>20</td>
<td>6.8017e-7</td>
<td>4.0041</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td></td>
<td>9.5026e-7</td>
<td>2.0015</td>
<td></td>
<td>25</td>
<td>2.7835e-7</td>
<td>4.0027</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td></td>
<td>4.8253e-7</td>
<td>2.0031</td>
<td></td>
<td>30</td>
<td>1.3417e-7</td>
<td>4.0019</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td></td>
<td>1.9725e-7</td>
<td>2.0019</td>
<td></td>
<td>40</td>
<td>4.2432e-8</td>
<td>4.0013</td>
</tr>
</tbody>
</table>

Table 4.2

Comparison of the $L^2$ errors $\max_{0 \leq n \leq n_T} \| E^n(\tau, h) \|$ for Example 4.1.

<table>
<thead>
<tr>
<th>$\beta$</th>
<th>$N$</th>
<th>$1/\tau$</th>
<th>FDS-D I</th>
<th>FDS-D II</th>
<th>Method [21]</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2</td>
<td>200</td>
<td>1000</td>
<td>5.0220e-10</td>
<td>1.1177e-09</td>
<td>1.5657e-08</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3000</td>
<td>7.0446e-11</td>
<td>1.4651e-10</td>
<td>2.1762e-08</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5000</td>
<td>5.1233e-11</td>
<td>8.6250e-11</td>
<td>8.6694e-09</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7000</td>
<td>4.8253e-11</td>
<td>7.4574e-11</td>
<td>4.7265e-09</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9000</td>
<td>4.7551e-11</td>
<td>6.9367e-11</td>
<td>2.0932e-10</td>
</tr>
<tr>
<td></td>
<td></td>
<td>11000</td>
<td>4.7551e-11</td>
<td>6.8865e-11</td>
<td>1.5487e-10</td>
</tr>
<tr>
<td>0.5</td>
<td>80</td>
<td>1000</td>
<td>2.0384e-09</td>
<td>3.2480e-09</td>
<td>2.5717e-08</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3000</td>
<td>1.8596e-09</td>
<td>2.6838e-09</td>
<td>4.9093e-08</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5000</td>
<td>1.8575e-09</td>
<td>2.6614e-09</td>
<td>2.2700e-08</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7000</td>
<td>1.8573e-09</td>
<td>2.6559e-09</td>
<td>1.3650e-08</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9000</td>
<td>1.8573e-09</td>
<td>2.6537e-09</td>
<td>9.3326e-09</td>
</tr>
<tr>
<td></td>
<td></td>
<td>11000</td>
<td>1.8573e-09</td>
<td>2.6526e-09</td>
<td>6.8876e-09</td>
</tr>
<tr>
<td></td>
<td></td>
<td>13000</td>
<td>1.8573e-09</td>
<td>2.651e-09</td>
<td>5.3476e-09</td>
</tr>
<tr>
<td>0.8</td>
<td>40</td>
<td>1000</td>
<td>2.9737e-08</td>
<td>4.2556e-08</td>
<td>3.3741e-06</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3000</td>
<td>2.9737e-08</td>
<td>4.2444e-08</td>
<td>8.9630e-06</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5000</td>
<td>2.9734e-08</td>
<td>4.2435e-08</td>
<td>8.3393e-07</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7000</td>
<td>2.9734e-08</td>
<td>4.2433e-08</td>
<td>3.2157e-07</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9000</td>
<td>2.9734e-08</td>
<td>4.2432e-08</td>
<td>2.3712e-07</td>
</tr>
<tr>
<td></td>
<td></td>
<td>11000</td>
<td>2.9734e-08</td>
<td>4.2431e-08</td>
<td>1.8587e-07</td>
</tr>
<tr>
<td></td>
<td></td>
<td>13000</td>
<td>2.9734e-08</td>
<td>4.2431e-08</td>
<td>1.5174e-07</td>
</tr>
</tbody>
</table>

Next, we compare the present FEMs FDS-D I and FDS-D II with the high-order method proposed in [21], where time was discretized by the L1 method and space was approximated by the FEM as given in this paper. We choose the same parameters as in [21], the results are shown in Table 4.2. We find that the two methods FDS-D I and FDS-D II in this paper show better performance than that in [21] for this example, since we use higher-order approximation in time.

Example 4.2. Consider the following subdiffusion equation [20]:

\[
\begin{align*}
C \frac{D_0^\beta}{D_t^\beta} u & = \partial_x^2 u + f(x, t), \quad (x, t) \in (0, 1) \times (0, 1], \\
\partial_x u & = \exp(x), \quad x \in (0, 1), \\
(0, 0) & = t^{1+\beta}, \quad u(1, t) = t^{1+\beta} \exp(1) \quad t \in (0, 1],
\end{align*}
\]
where $0 < \beta < 1$, and

$$f(x, t) = (\Gamma(2 + \beta)t - t^{2\beta + 1}) \exp(x).$$

The exact solution of (4.3) is

$$u = t^{1+\beta} \exp(x).$$

In this example, the $L^\infty$ error on the grid points $\{x_i\}$ at $t = t_n$ is defined as

$$E_{\infty}(\tau, h, t_n) = \max_{0 \leq i \leq N} |u(x_i, t_n) - u_h^n(x_i)|.$$

The convergence orders are similarly defined as (4.2).

We mainly compare the numerical results obtained by the cubic element methods FDS-D I and FDS-D II with the compact finite difference method proposed by Gao and Sun [20], in which time was discretized by the L1 method and space was approximated by the fourth-order compact finite difference method.

Table 4.3 displays the $L^\infty$ errors at $t = 1$ and the convergence orders with $\beta = 0.75$; the last two columns present the numerical results obtained in [20]. The parameters $N$ and $\tau$ are chosen as $N = 10000$ and $\tau = 5e-6$ in [20] (see Table 1 in [20]), while we choose $N = 100$ and $\tau = 1e-4$. From Table 4.3, we can see that we obtain much better results. In the following, we will give illustrations in Table 4.5 to show that second-order accuracy is obtained even if $u$ is not sufficiently smooth in time. Choosing the same parameters, we also compare the methods FDS-D I and FDS-D II with the method proposed by Gao and Sun [20]; the results are shown in Table 4.4. One can observe that our methods exhibit much better numerical results since the methods FDS-D I and FDS-D II achieve higher convergence rates in this example.

Next, we choose the suitable right-hand-side function $f(x, t)$ and the suitable initial and boundary conditions such that (4.3) has the following analytical solution:

### Table 4.3

Comparison of the $L^\infty$ errors at $t = 1$ for Example 4.2, $\beta = 0.75$, $N = 100$.

<table>
<thead>
<tr>
<th>$1/\tau$</th>
<th>FDS-D I Order</th>
<th>FDS-D II Order</th>
<th>LIC [20] Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>5.5002e-05</td>
<td>3.9580e-07</td>
<td>5.519e-03</td>
</tr>
<tr>
<td>20</td>
<td>7.5823e-06</td>
<td>2.8588</td>
<td>2.318e-03</td>
</tr>
<tr>
<td>40</td>
<td>1.0496e-06</td>
<td>3.5706e-09</td>
<td>3.8513</td>
</tr>
<tr>
<td>80</td>
<td>1.4782e-07</td>
<td>9.3856e-10</td>
<td>9.742e-04</td>
</tr>
<tr>
<td>160</td>
<td>2.1243e-08</td>
<td>4.8269e-10</td>
<td>1.2500</td>
</tr>
</tbody>
</table>

### Table 4.4

Comparison of the $L^\infty$ errors at $t = 1$ for Example 4.2.

<table>
<thead>
<tr>
<th>$1/\tau$</th>
<th>$\beta = 0.25$</th>
<th>$\beta = 0.75$</th>
</tr>
</thead>
<tbody>
<tr>
<td>64 8 2.7301e-06 2.8460e-07 6.0359e-06</td>
<td>1.3147e-07 1.2932e-07 5.3739e-04</td>
<td></td>
</tr>
<tr>
<td>1028 16 6.8502e-09 8.367e-09 3.8460e-08</td>
<td>8.3028e-09 8.3022e-09 1.6874e-05</td>
<td></td>
</tr>
<tr>
<td>8 8 2.1647e-04 1.9277e-05 2.2328e-04</td>
<td>1.0239e-04 7.6851e-07 7.0000e-03</td>
<td></td>
</tr>
<tr>
<td>128 16 5.6815e-07 4.9091e-08 1.8709e-06</td>
<td>3.1043e-08 8.2790e-09 2.2716e-04</td>
<td></td>
</tr>
<tr>
<td>2048 32 9.7178e-10 5.3225e-10 1.4588e-08</td>
<td>5.2618e-10 5.2606e-10 7.1085e-06</td>
<td></td>
</tr>
</tbody>
</table>
\[ u = (t^\alpha + 1) \exp(x), \quad (x,t) \in (0,1) \times (0,1], \quad \alpha \geq \beta. \]

In the numerical simulation, the cubic element \((r = 3)\) is used, and \(\beta = 0.25, N = 1000\). We choose several values of \(\alpha (\alpha = 0.25, 0.5, 0.75, 1)\); the \(L^\infty\) errors at \(t = 1\) are shown in Table 4.5. Since \(C D_t^\alpha u(x,t) = t^{\alpha - \beta} \exp(x)\), we can predict that the convergence rate is \(\min\{\alpha - \beta + 1, 2\}\) in time for such a solution. From Table 4.5, we can see that a slightly better experimental convergence rate is achieved. For \(\alpha \geq 1.25\), the second-order experimental convergence rate can be observed; we do not list all the results here. Table 4.6 presents the \(L^\infty\) error at \(t = 1\) and corresponding convergence orders for different \(\beta (\beta = 0.1, 0.4, 0.6, 0.9)\) and \(\alpha = 1, N = 1000\).

### Table 4.5

The \(L^\infty\) errors at \(t = 1\) and convergence orders in time for Example 4.2 with exact solution \(u = (t^\alpha + 1) \exp(x), \beta = 0.25, N = 1000\).

<table>
<thead>
<tr>
<th>Method</th>
<th>(1/\tau)</th>
<th>(\alpha = 0.25)</th>
<th>Order (\alpha = 0.5)</th>
<th>Order (\alpha = 0.75)</th>
<th>Order (\alpha = 1)</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>FDS-D I</td>
<td>10</td>
<td>2.50e-3</td>
<td>1.44e-3</td>
<td>7.00e-4</td>
<td>3.20e-4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>2.00e-3</td>
<td>1.57e-4</td>
<td>1.56e-4</td>
<td>1.80e-4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>5.00e-4</td>
<td>1.13e-4</td>
<td>1.65e-5</td>
<td>1.90e-5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>2.40e-4</td>
<td>1.20e-5</td>
<td>1.73e-6</td>
<td>1.98e-6</td>
<td></td>
</tr>
<tr>
<td></td>
<td>160</td>
<td>1.00e-4</td>
<td>2.06e-5</td>
<td>1.52e-6</td>
<td>2.03e-6</td>
<td></td>
</tr>
<tr>
<td></td>
<td>320</td>
<td>4.00e-5</td>
<td>9.09e-6</td>
<td>3.68e-7</td>
<td>2.05e-7</td>
<td></td>
</tr>
<tr>
<td></td>
<td>640</td>
<td>1.63e-5</td>
<td>3.13e-6</td>
<td>8.76e-8</td>
<td>2.07e-8</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1280</td>
<td>6.49e-6</td>
<td>1.09e-6</td>
<td>2.05e-8</td>
<td>2.10e-8</td>
<td></td>
</tr>
<tr>
<td>FDS-D H</td>
<td>10</td>
<td>8.20e-4</td>
<td>3.37e-4</td>
<td>1.40e-4</td>
<td>5.85e-5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>3.53e-4</td>
<td>1.22e-4</td>
<td>4.18e-5</td>
<td>1.50e-5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>1.53e-4</td>
<td>4.42e-5</td>
<td>1.26e-5</td>
<td>3.89e-6</td>
<td></td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>6.69e-5</td>
<td>1.63e-5</td>
<td>3.87e-6</td>
<td>1.01e-6</td>
<td></td>
</tr>
<tr>
<td></td>
<td>160</td>
<td>2.94e-5</td>
<td>6.07e-6</td>
<td>1.20e-6</td>
<td>2.67e-7</td>
<td></td>
</tr>
<tr>
<td></td>
<td>320</td>
<td>1.30e-5</td>
<td>2.28e-6</td>
<td>3.77e-7</td>
<td>7.13e-8</td>
<td></td>
</tr>
<tr>
<td></td>
<td>640</td>
<td>5.80e-6</td>
<td>8.67e-7</td>
<td>1.21e-7</td>
<td>2.09e-8</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1280</td>
<td>2.40e-6</td>
<td>3.31e-7</td>
<td>3.87e-8</td>
<td>5.66e-9</td>
<td></td>
</tr>
</tbody>
</table>

### Table 4.6

The \(L^\infty\) errors at \(t = 1\) and convergence orders in time for Example 4.2 with exact solution \(u = (t + 1) \exp(x), N = 1000\).

<table>
<thead>
<tr>
<th>Method</th>
<th>(1/\tau)</th>
<th>(\beta = 0.1)</th>
<th>Order (\beta = 0.4)</th>
<th>Order (\beta = 0.6)</th>
<th>Order (\beta = 0.9)</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>FDS-D I</td>
<td>10</td>
<td>6.38e-5</td>
<td>6.71e-4</td>
<td>9.00e-4</td>
<td>1.36e-3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>1.81e-5</td>
<td>1.59e-4</td>
<td>2.15e-4</td>
<td>3.28e-4</td>
<td>2.05</td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>5.07e-6</td>
<td>3.71e-5</td>
<td>5.23e-5</td>
<td>8.07e-5</td>
<td>2.02</td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>1.41e-6</td>
<td>8.90e-6</td>
<td>1.27e-6</td>
<td>2.00e-5</td>
<td>2.01</td>
</tr>
<tr>
<td></td>
<td>160</td>
<td>3.87e-7</td>
<td>2.17e-6</td>
<td>3.12e-6</td>
<td>5.44e-6</td>
<td>1.88</td>
</tr>
<tr>
<td></td>
<td>320</td>
<td>1.05e-7</td>
<td>5.30e-7</td>
<td>7.69e-7</td>
<td>2.50e-6</td>
<td>1.12</td>
</tr>
<tr>
<td></td>
<td>640</td>
<td>2.82e-8</td>
<td>1.30e-7</td>
<td>2.25e-7</td>
<td>1.15e-6</td>
<td>1.11</td>
</tr>
<tr>
<td></td>
<td>1280</td>
<td>6.59e-9</td>
<td>3.17e-8</td>
<td>8.35e-8</td>
<td>5.37e-7</td>
<td>1.11</td>
</tr>
<tr>
<td>FDS-D H</td>
<td>10</td>
<td>1.94e-5</td>
<td>1.05e-4</td>
<td>1.72e-4</td>
<td>1.89e-4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>4.88e-6</td>
<td>2.84e-5</td>
<td>5.18e-5</td>
<td>6.56e-5</td>
<td>1.53</td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>1.23e-6</td>
<td>7.78e-6</td>
<td>1.63e-5</td>
<td>2.76e-5</td>
<td>1.25</td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>3.09e-7</td>
<td>2.18e-6</td>
<td>5.38e-6</td>
<td>1.21e-5</td>
<td>1.19</td>
</tr>
<tr>
<td></td>
<td>160</td>
<td>7.93e-8</td>
<td>6.22e-7</td>
<td>1.83e-6</td>
<td>5.46e-6</td>
<td>1.15</td>
</tr>
<tr>
<td></td>
<td>320</td>
<td>2.06e-8</td>
<td>1.81e-7</td>
<td>6.64e-7</td>
<td>2.50e-6</td>
<td>1.13</td>
</tr>
<tr>
<td></td>
<td>640</td>
<td>3.99e-9</td>
<td>5.44e-8</td>
<td>2.34e-7</td>
<td>1.16e-6</td>
<td>1.11</td>
</tr>
<tr>
<td></td>
<td>1280</td>
<td>1.14e-9</td>
<td>1.72e-8</td>
<td>8.50e-8</td>
<td>5.37e-7</td>
<td>1.11</td>
</tr>
</tbody>
</table>
The next example considers (1.1) with Neumann boundary conditions.

**Example 4.3.** Consider the following subdiffusion equation with Neumann boundary conditions [47, 60]:

\[
\begin{aligned}
C D^\beta_{0,t} u &= \partial^2_x u + f(x,t), \quad (x,t) \in (0,1) \times (0,1], \quad 0 < \beta < 1, \\
\quad u(x,0) &= 0, \quad x \in (0,1), \\
\quad \partial_x u(0, t) &= 0, \quad \partial_x u(1, t) = 0 \quad t \in (0,1],
\end{aligned}
\]

where

\[
f(x,t) = \frac{\Gamma(3 + \beta)}{2} \exp(x)x^2(1-x)^2t^2 - t^{\beta+2} \exp(x)(2 - 8x + x^2 + 6x^3 + x^4).
\]

The exact solution to (4.4) is

\[
u = t^{\beta+2} \exp(x)x^2(1-x)^2.
\]

The maximum \(L^\infty\) error on the grid points is defined as

\[
E_{\infty}(\tau, h) = \max_{0 \leq i \leq N, 0 \leq n \leq n_T} |u(x_i, t_n) - u_h(x_i, t_n)|.
\]

We first choose the parameters \(\beta = 0.3, 0.5, 0.7\) and \(h = 1/N = 1/20000\) as in [60]. Table 4.7 displays the maximum \(L^\infty\) errors and convergence orders in time direction of

<table>
<thead>
<tr>
<th>(\beta)</th>
<th>1/(\tau)</th>
<th>FDS-N I</th>
<th>Order</th>
<th>FDS-N II</th>
<th>Order</th>
<th>Method [60]</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.3</td>
<td>10</td>
<td>4.2659e-5</td>
<td></td>
<td>8.6912e-5</td>
<td></td>
<td>4.076e-4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>1.0670e-5</td>
<td></td>
<td>2.1812e-5</td>
<td></td>
<td>1.341e-4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>2.6606e-6</td>
<td></td>
<td>5.4561e-6</td>
<td></td>
<td>4.337e-5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>6.5688e-7</td>
<td></td>
<td>1.3568e-6</td>
<td></td>
<td>1.385e-5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>160</td>
<td>1.5600e-7</td>
<td></td>
<td>3.3066e-7</td>
<td></td>
<td>1.6469</td>
<td></td>
</tr>
<tr>
<td></td>
<td>320</td>
<td>3.3059e-8</td>
<td></td>
<td>7.363e-8</td>
<td></td>
<td>1.6915</td>
<td></td>
</tr>
<tr>
<td></td>
<td>640</td>
<td>6.8788e-9</td>
<td></td>
<td>1.5367e-8</td>
<td></td>
<td>1.7536</td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>10</td>
<td>8.9181e-5</td>
<td></td>
<td>1.5477e-4</td>
<td></td>
<td>4.07e-4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>2.2318e-5</td>
<td></td>
<td>3.8888e-5</td>
<td></td>
<td>4.543e-4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>5.5751e-6</td>
<td></td>
<td>9.7396e-6</td>
<td></td>
<td>1.9974</td>
<td></td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>1.3884e-6</td>
<td></td>
<td>2.4324e-6</td>
<td></td>
<td>1.9977</td>
<td></td>
</tr>
<tr>
<td></td>
<td>160</td>
<td>3.4090e-7</td>
<td></td>
<td>6.0192e-7</td>
<td></td>
<td>2.142e-5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>320</td>
<td>8.1489e-8</td>
<td></td>
<td>1.4664e-7</td>
<td></td>
<td>2.142e-5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>640</td>
<td>1.7182e-8</td>
<td></td>
<td>2.2457</td>
<td></td>
<td>2.498e-6</td>
<td></td>
</tr>
<tr>
<td>0.7</td>
<td>10</td>
<td>1.5080e-4</td>
<td></td>
<td>2.2020e-4</td>
<td></td>
<td>2.967e-3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>3.8306e-5</td>
<td></td>
<td>5.5299e-5</td>
<td></td>
<td>1.237e-3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>9.5742e-6</td>
<td></td>
<td>1.3853e-5</td>
<td></td>
<td>5.11e-4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>2.3065e-6</td>
<td></td>
<td>3.4641e-6</td>
<td></td>
<td>1.9997</td>
<td></td>
</tr>
<tr>
<td></td>
<td>160</td>
<td>5.9277e-7</td>
<td></td>
<td>8.6198e-7</td>
<td></td>
<td>2.098e-5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>320</td>
<td>1.4459e-7</td>
<td></td>
<td>2.1208e-7</td>
<td></td>
<td>8.577e-5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>640</td>
<td>3.2598e-8</td>
<td></td>
<td>4.9506e-8</td>
<td></td>
<td>1.422e-5</td>
<td></td>
</tr>
</tbody>
</table>

The \(L^\infty\) errors at \(t = 1\) for Example 4.3, \(N = 20000\).
the linear element methods FDS-N I and FDS-N II. The last two columns of Table 4.7 present the results of the box-type scheme [60] with convergence order $O(\tau^{2-\beta} + h^2)$. Obviously, the second-order experimental convergence rates in time are observed for the methods FDS-N I and FDS-N II, and much better performances are shown over the box-type scheme [60] in this example.

Next, we compare the cubic element methods FDS-N I and FDS-N II with the compact finite difference method developed in [47] with convergence order $O(\tau^{2-\beta} + h^4)$. We first choose the parameters $\beta = 0.3, 0.5, 0.7$, and $h = 1/N = 1/1000$ in the computation. The maximum $L^\infty$ errors are displayed in Table 4.8. Then, the time step $\tau$ and space step $h$ are chosen such that $\tau^{2-\beta} \approx h^2$ as in [47], and $\beta = 0.3, 0.5, 0.7$; the maximum $L^\infty$ errors are shown in Table 4.9. From Tables 4.8 and 4.9, one can easily find that the more accurate numerical results are derived by the present methods FDS-N I and FDS-N II.

### Table 4.8

<table>
<thead>
<tr>
<th>$\beta$</th>
<th>$1/\tau$</th>
<th>FDS-N I Order</th>
<th>FDS-N II Order</th>
<th>Method [47] Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.3</td>
<td>10</td>
<td>4.2669e-5</td>
<td>8.6920e-5</td>
<td>4.076e-4</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>1.0682e-5</td>
<td>2.1822e-5</td>
<td>1.341e-4</td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>6.6800e-7</td>
<td>1.3681e-6</td>
<td>1.389e-5</td>
</tr>
<tr>
<td></td>
<td>160</td>
<td>1.6691e-7</td>
<td>3.4216e-7</td>
<td>4.404e-6</td>
</tr>
<tr>
<td>0.5</td>
<td>10</td>
<td>8.9190e-5</td>
<td>1.4578e-4</td>
<td>2.072e-3</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>2.2327e-5</td>
<td>3.8897e-5</td>
<td>4.544e-4</td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>1.3964e-6</td>
<td>2.4405e-6</td>
<td>5.979e-5</td>
</tr>
<tr>
<td></td>
<td>160</td>
<td>3.4905e-7</td>
<td>6.1047e-7</td>
<td>1.4794e-5</td>
</tr>
<tr>
<td>0.7</td>
<td>10</td>
<td>1.5309e-4</td>
<td>2.9021e-4</td>
<td>2.9021e-3</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>3.8312e-5</td>
<td>5.5305e-5</td>
<td>1.237e-3</td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>9.5860e-6</td>
<td>3.4684e-6</td>
<td>2.098e-4</td>
</tr>
<tr>
<td></td>
<td>160</td>
<td>3.4905e-7</td>
<td>8.6752e-7</td>
<td>1.2903e-5</td>
</tr>
</tbody>
</table>

### Table 4.9

<table>
<thead>
<tr>
<th>$\beta$</th>
<th>$1/\tau$</th>
<th>$N$</th>
<th>FDS-N I</th>
<th>FDS-N II</th>
<th>Method [47]</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.3</td>
<td>25</td>
<td>10</td>
<td>1.1580e-5</td>
<td>1.1493e-5</td>
<td>1.684e-4</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>15</td>
<td>2.3279e-6</td>
<td>2.3151e-6</td>
<td>3.335e-5</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>20</td>
<td>7.4030e-7</td>
<td>7.3974e-7</td>
<td>1.056e-5</td>
</tr>
<tr>
<td></td>
<td>150</td>
<td>25</td>
<td>3.0596e-7</td>
<td>3.0480e-7</td>
<td>4.330e-6</td>
</tr>
<tr>
<td></td>
<td>200</td>
<td>30</td>
<td>1.4807e-7</td>
<td>1.4758e-7</td>
<td>2.089e-6</td>
</tr>
<tr>
<td>0.5</td>
<td>25</td>
<td>10</td>
<td>9.7797e-6</td>
<td>9.7494e-6</td>
<td>1.526e-4</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>15</td>
<td>1.9712e-6</td>
<td>1.9678e-6</td>
<td>3.022e-5</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>20</td>
<td>6.2992e-7</td>
<td>6.2917e-7</td>
<td>9.571e-6</td>
</tr>
<tr>
<td></td>
<td>150</td>
<td>25</td>
<td>2.5935e-7</td>
<td>2.5931e-7</td>
<td>3.922e-6</td>
</tr>
<tr>
<td></td>
<td>200</td>
<td>30</td>
<td>1.2565e-7</td>
<td>1.2556e-7</td>
<td>1.892e-6</td>
</tr>
<tr>
<td>0.7</td>
<td>25</td>
<td>10</td>
<td>8.2205e-6</td>
<td>8.2158e-6</td>
<td>1.386e-4</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>15</td>
<td>1.6608e-6</td>
<td>1.6604e-6</td>
<td>2.747e-5</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>20</td>
<td>5.3131e-7</td>
<td>5.3124e-7</td>
<td>8.701e-6</td>
</tr>
<tr>
<td></td>
<td>150</td>
<td>25</td>
<td>2.1905e-7</td>
<td>2.1904e-7</td>
<td>3.566e-6</td>
</tr>
<tr>
<td></td>
<td>200</td>
<td>30</td>
<td>1.0610e-7</td>
<td>1.0609e-7</td>
<td>1.720e-6</td>
</tr>
</tbody>
</table>
Example 4.4. Consider the following problem [22]:

\[
\begin{aligned}
-cD_0^\beta u &= \partial_2^2 u, \quad (x, t) \in (0, 1) \times (0, 1], \quad 0 < \beta < 1, \\
u(0, t) &= u(1, t) = 0 \quad t \in (0, 1], \\
u(x, 0) &= \phi_0(x), \quad x \in (0, 1).
\end{aligned}
\] (4.6)

In [22], the authors tested the problem (4.6) in several cases of initial data with different smoothness. In the present paper, we perform numerical tests on the following three different cases:

(a) Smooth initial data: \( \phi_0(x) = 4x(1 - x) \). In this case the initial data \( \phi_0 \in H^2(I) \cap H_0^1(I) \) and the exact solution can be represented by the following rapidly converging Fourier series:

\[
u(x, t) = \frac{16}{\pi^3} \sum_{k=1}^{\infty} \frac{1}{k^3} E_{\beta,1}(-k^2 \pi^2 t^\beta)(1 - (-1)^k) \sin(k\pi x),
\]

where \( E_{\beta,\gamma}(z) \) is the Miggag–Leffler function defined by

\[
E_{\beta,\gamma}(z) = \sum_{k=0}^{\infty} \frac{z^k}{\Gamma(k\beta + \gamma)}.
\]

(b) Initial data in \( H^1 \) (intermediate smoothness):

\[
\phi_0(x) = \begin{cases} x, & x \in [0, 0.5], \\
1 - x, & x \in (0.5, 1].
\end{cases}
\]

(c) Nonsmooth initial data: (1) \( \phi_0(x) = 1 \), (2) \( \phi_0(x) = x \). In this case, \( \phi_0 \) is not compatible with the homogeneous Dirichlet boundary conditions. However, \( \phi_0 \in H^s, 0 < s < 1/2 \).

We choose the same parameters as those in [22], except that the time step size \( \tau = 1 \times 10^{-4} \) is used in this paper, while the time step size in [22] is chosen as \( \tau = 1 \times 10^{-6} \). In this example, the space step sizes are always chosen as \( h \) (\( h = 2^{-k}, k = 2, 4, 5, 6, 7 \)), the accuracy is measured by the normalized \( L^2 \) error \( \frac{\|u(t) - u_h\|}{\|u_0\|} \), and a linear element is used as the Galerkin FEM in [22]. For cases (b) and (c), the exact solutions are not easy to obtain, so we use the space step size \( h = 2^{-7} \) with cubic element as the reference solution.

Table 4.10 displays the normalized \( L^2 \) error at \( t = 1 \) with different \( \beta (\beta = 0.1, 0.5, 0.95) \) for case (a). We can see that the methods FDS-D I and FDS-D II have similar numerical solutions as in [22]. Table 4.11 shows the normalized \( L^2 \) error

<table>
<thead>
<tr>
<th>( \beta )</th>
<th>FDS-D I</th>
<th>FDS-D II</th>
<th>FEM [22]</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \beta = 0.1 )</td>
<td>( \beta = 0.5 )</td>
<td>( \beta = 0.95 )</td>
<td>( \beta = 0.1 )</td>
</tr>
<tr>
<td>8</td>
<td>9.98e-4</td>
<td>7.13e-4</td>
<td>1.11e-4</td>
</tr>
<tr>
<td>16</td>
<td>2.44e-4</td>
<td>1.79e-4</td>
<td>2.83e-5</td>
</tr>
<tr>
<td>32</td>
<td>5.30e-5</td>
<td>4.46e-5</td>
<td>7.09e-6</td>
</tr>
<tr>
<td>64</td>
<td>8.89e-6</td>
<td>1.07e-5</td>
<td>1.76e-6</td>
</tr>
<tr>
<td>128</td>
<td>6.08e-6</td>
<td>2.23e-6</td>
<td>4.29e-7</td>
</tr>
</tbody>
</table>
Table 4.11
The normalized $L^2$ errors at $t = 1$ for cases (b) and (c), $\tau = 1 \times 10^{-4}, \beta = 0.5$.

<table>
<thead>
<tr>
<th>$N$</th>
<th>8</th>
<th>16</th>
<th>32</th>
<th>64</th>
<th>128</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case (b)</td>
<td>FDS-D I</td>
<td>5.99e-6</td>
<td>2.45e-6</td>
<td>1.73e-6</td>
<td>3.20e-6</td>
</tr>
<tr>
<td></td>
<td>FDS-D II</td>
<td>7.06e-6</td>
<td>1.80e-6</td>
<td>4.53e-7</td>
<td>1.13e-7</td>
</tr>
<tr>
<td></td>
<td>FEM [22]</td>
<td>8.08e-4</td>
<td>2.00e-4</td>
<td>5.00e-5</td>
<td>1.26e-5</td>
</tr>
<tr>
<td>Case (c) (1)</td>
<td>FDS-D I</td>
<td>7.44e-6</td>
<td>2.65e-6</td>
<td>1.46e-6</td>
<td>1.26e-6</td>
</tr>
<tr>
<td></td>
<td>FDS-D II</td>
<td>6.44e-6</td>
<td>1.64e-6</td>
<td>4.11e-7</td>
<td>1.03e-7</td>
</tr>
<tr>
<td></td>
<td>FEM [22]</td>
<td>8.07e-4</td>
<td>2.02e-4</td>
<td>5.03e-5</td>
<td>1.25e-5</td>
</tr>
<tr>
<td>Case (c) (2)</td>
<td>FDS-D I</td>
<td>6.44e-6</td>
<td>2.30e-6</td>
<td>1.31e-6</td>
<td>1.21e-6</td>
</tr>
<tr>
<td></td>
<td>FDS-D II</td>
<td>5.58e-6</td>
<td>1.42e-6</td>
<td>3.56e-7</td>
<td>8.92e-8</td>
</tr>
<tr>
<td></td>
<td>FEM [22]</td>
<td>8.05e-4</td>
<td>2.01e-4</td>
<td>5.03e-5</td>
<td>1.25e-5</td>
</tr>
</tbody>
</table>

at $t = 1$ for cases (b) and (c) with $\beta = 0.5$. The results of our methods exhibit a slightly better accuracy than the published results.

5. Conclusion. In this paper, we propose finite difference/element methods for the subdiffusion equation (1.1) subject to Dirichlet and Neumann boundary conditions. We give strict stability and convergence analysis. All the numerical methods presented in this paper are unconditionally stable, and the convergence orders in time are between $(2 - \beta)$ and 2 for the suitably smooth solutions. Even if the exact solution is not smooth enough, the present methods can attain second-order accuracy in time. To the best knowledge of the authors, there are few works on numerical methods with convergence order greater than $(2 - \beta)$ with unconditional stability for the subdiffusion equation (1.1), while numerical methods with $(2 - \beta)$-order accuracy can be found in several papers; see, for instance, [17, 20, 21, 28]. One can also see [24, 39] for the corresponding works.

We present enough numerical experiments to verify the theoretical analysis, and the comparisons with other methods are also given, which exhibit better accuracy than many of the existing numerical methods.

Obviously, the methods FDS-D I and FDS-D II can be easily extended to the corresponding two- and three-dimensional problems. The stability and convergence analysis is very similar to that given here. In future work, we would extend the present methods with the alternating direct implicit technique to deal with high-order problems with high-order accuracy in both time and space.

Acknowledgments. The authors wish to thank the referees for their constructive comments and suggestions to improve the present paper.

REFERENCES

THE TIME-FRACTIONAL SUBDIFFUSION EQUATION


Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.


