

**AN APPROACH TO CONSTRUCT HIGHER ORDER
TIME DISCRETISATION SCHEMES FOR TIME
FRACTIONAL PARTIAL DIFFERENTIAL EQUATIONS
WITH NONSMOOTH DATA
IN “FCAA” JOURNAL**

Neville J. Ford ¹, Y.Yan ²

Abstract

In this paper, we shall review an approach by which we can seek higher order time discretisation schemes for solving time fractional partial differential equations with nonsmooth data. The low regularity of the solutions of time fractional partial differential equations implies standard time discretisation schemes only yield first order accuracy. To obtain higher order time discretisation schemes when the solutions of time fractional partial differential equations have low regularities, one may correct the starting steps of the standard time discretisation schemes to capture the singularities of the solutions. We will consider these corrections of some higher order time discretisation schemes obtained by using Lubich’s fractional multistep methods, L1 scheme and its modification, discontinuous Galerkin methods, etc. Numerical examples are given to show that the theoretical results are consistent with the numerical results.

MSC 2010: Primary 65M06; Secondary 65M12, 65M15, 26A33, 35R11.

Key Words and Phrases: time fractional partial differential equations, finite difference method, convolution quadrature, error estimates

1. Introduction

In this paper, we review an approach to looking for higher order time discretisation methods for solving the following time fractional partial differential equations, with $0 < \alpha < 1$,

$${}_0^C D_t^\alpha u(t) + Au(t) = f(t), \quad \text{for } 0 < t \leq T, \quad \text{with } u(0) = u_0, \quad (1.1)$$

where ${}_0^C D_t^\alpha u(t)$ denotes the Caputo fractional derivative defined by

$${}_0^C D_t^\alpha u(t) = \frac{1}{\Gamma(1-\alpha)} \int_0^t (t-s)^{-\alpha} u'(s) ds,$$

and $u'(s) = \partial u / \partial s$ and A is a selfadjoint positive definite second order elliptic partial differential operator in $\Omega \subset \mathbb{R}^d, d = 1, 2, 3$, with $\mathcal{D}(A) = H_0^1(\Omega) \cap H^2(\Omega)$, where $H_0^1(\Omega), H^2(\Omega)$ denote the standard Sobolev spaces. For example, we may choose $A = -\Delta$ subject to the Dirichlet boundary condition, where Δ denotes the Laplacian.

The assumption that A is positive definite implies that A generates an analytic semigroup, so that for some $\pi/2 < \theta_0 < \pi$ and with $C = C_{\theta_0}$ we have the resolvent estimate, see Lubich *et al.* [33], Thomée [48],

$$\|(zI + A)^{-1}\| \leq C|z|^{-1} \quad \text{for } z \in \Sigma_{\theta_0} = \{z \neq 0 : |\arg z| < \theta_0\}. \quad (1.2)$$

In our analysis, we will choose $\theta > \pi/2$ close to $\pi/2$ such that $\theta < \theta_0$ which implies that $z^\alpha \in \Sigma_{\theta_0}$ for any $z \in \Sigma_\theta$ since $\arg(z^\alpha) = \alpha\theta < \theta < \theta_0$ for $0 < \alpha < 1$. Hence there exists a constant C which depends only on θ and α such that, see Jin *et al.* [21, (2.3)],

$$\|(z^\alpha I + A)^{-1}\| \leq C|z|^{-\alpha}, \quad \forall z \in \Sigma_\theta = \{z \neq 0 : |\arg z| < \theta\}. \quad (1.3)$$

Under the assumptions that the solutions of (1.1) are sufficiently smooth, for example $u \in C^2[0, T]$, there are many time discretisation schemes in the literature, for example, Lubich's fractional multistep methods [7], [30], [55], [65], [61], [62], [64], the L1 scheme and its modification [28], [26], [18], [29], [47], the spectral method [58] [63], [4], [68], [57], nonpolynomial collocation method [14], discontinuous Galerkin method [39], etc.

However, the solutions of (1.1) are not smooth in general. The regularity of the solution of (1.1) is restrictive. For example, for the homogeneous equation with initial data $u_0 \in L_2(\Omega)$, we have the following stability estimate [44]

$$\|{}_0^C D_t^\alpha u\|_{L_2} \leq Ct^{-\alpha} \|u_0\|_{L_2}.$$

That is, the α th order Caputo derivative is already unbounded when $t \rightarrow 0$. Hence, the C^2 -regularity assumption generally does not hold for (1.1). Numerical experiments indicate that the convergence rate actually does not hold uniformly in t even for smooth data u_0 [20]. Stynes *et al.* [46] showed that the solution u of (1.1) behaves as $t^\alpha, 0 < \alpha < 1$ which is not in $C^2[0, T]$, see also Stynes [45]. Hence error estimates of the time discretisation schemes of (1.1) obtained under the assumptions that the solutions of (1.1) are sufficiently smooth do not hold when the solutions of (1.1) are not smooth, for example $u \notin C^2[0, T]$. In other words, the optimal convergence rates of the time discretisation schemes cannot be achieved when the solutions are not smooth. Therefore it is interesting to consider

error estimates of time discretisation schemes for solving (1.1) when the solutions of (1.1) are not smooth. There are several alternative approaches to construct higher order time discretisation schemes for solving (1.1) when the solutions of (1.1) are not smooth.

- (1) Use non-uniform meshes and approximate the solution near the singular point $t = 0$ by using finer meshes, see, [36], [39], [43], [56], [66].
- (2) Use some nonpolynomial (or singular) basis functions or collocation spectral methods to capture the singularity of the solutions of (1.1), see [1], [5], [13], [14], [34], [27], [59], [63], [67].
- (3) Separate the solution into two parts: smooth and nonsmooth parts. The nonsmooth part of the solution has the form $u(t) = t^{\lambda_l}, l = 0, 1, 2, \dots, s$ with $0 \leq \lambda_0 < \lambda_1 < \dots < \lambda_s$. Approximate the nonsmooth part by introducing some starting weights such that the time discretisation scheme holds exactly for the nonsmooth part $u(t) = t^{\lambda_l}, l = 0, 1, 2, \dots, s$ with $0 \leq \lambda_0 < \lambda_1 < \dots < \lambda_s$, see [30], [60], [62].
- (4) Introduce a *Correction* to the starting steps of the time discretisation schemes for solving (1.1) to capture the singularity of the solution of (1.1), see [33], [22], [23], [50], [52], [53].

Approach 4 is the main focus of this paper. Approach 3 is most closely related to it and we make the following remarks: Let $0 < t_0 < t_1 < \dots < t_N = T$ be a partition of $[0, T]$ and τ the time step size. To construct a time discretisation scheme for solving (1.1), we first need to approximate the Caputo fractional derivative ${}_0^C D_t^\alpha u(t)$ at $t = t_n, n = 1, 2, \dots, N$. Note that the Caputo fractional derivative has the following relation with the Riemann-Liouville fractional derivative, ${}_0^C D_t^\alpha u(t) = {}_0^R D_t^\alpha (u(t) - u(0)), 0 < \alpha < 1$, where ${}_0^R D_t^\alpha u(t)$ denotes the Riemann-Liouville fractional derivative defined by

$${}_0^R D_t^\alpha u(t) = \frac{1}{\Gamma(1 - \alpha)} \frac{d}{dt} \int_0^t (t - s)^{-\alpha} u(s) ds.$$

We may consider the approximation of the Riemann-Liouville fractional derivative instead. Based on Lubich's convolution quadrature [30], [31], [32], the Riemann-Liouville fractional derivative ${}_0^R D_t^\alpha u(t)$ can be approximated by, with $0 < \alpha < 1$,

$${}_0^R D_t^\alpha u(t) \approx \tau^{-\alpha} \sum_{j=0}^n w_{n-j} u(t_j). \tag{1.4}$$

Here $w_j, j = 0, 1, 2, \dots$ are the weights generated by the characteristic polynomial $\delta(\zeta)$ where, [23, (2.2)]

$$\delta(\zeta)^\alpha = \sum_{j=0}^{\infty} w_j \zeta^j \quad \text{with} \quad \delta(\zeta) = \sum_{j=1}^p \frac{1}{j} (1 - \zeta)^j, \quad p = 1, 2, \dots, 6. \quad (1.5)$$

The approximation method (1.4) generated by (1.5) is also called the backward difference method with order p (BDF $_p$) where $p = 1, 2, \dots, 6$, [30], [31], [32]. When $p = 1$, it is also called the Grünwald-Letnikov approximation method [42].

If the solution u is sufficiently smooth and has many vanishing derivatives at 0, then we have, [30], [31], [32],

$${}^R D_t^\alpha u(t_n) = \tau^{-\alpha} \sum_{j=0}^n w_{n-j} u(t_j) + O(\tau^p), \quad p = 1, 2, \dots, 6. \quad (1.6)$$

However the solutions of (1.1) has low regularity [44]. We only obtain first-order accuracy when solving (1.1) by using the approximation scheme (1.6) with $p \geq 2$. This has been observed numerically [7], [20].

To obtain the convergence rate $O(\tau^p), p = 2, \dots, 6$ for the approximation scheme (1.6) when $u(t)$ has low regularity, one may use the starting weights to correct the approximate scheme (1.6). More precisely, we may have, for $n \geq 1$,

$${}^R D_t^\alpha u(t_n) = \tau^{-\alpha} \sum_{j=0}^n w_{n-j} u(t_j) + \sum_{j=0}^s w_{n,j} u(t_j) + O(\tau^p), \quad p = 2, \dots, 6, \quad (1.7)$$

for some suitable starting weights $w_{n,j}, j = 0, 1, \dots, s$ with some $s \in \mathbb{Z}^+$. To determine the starting weights, we assume that (1.7) holds exactly for $u(t) = t^{\lambda_l}, l = 0, 1, \dots, s$, where $0 \leq \lambda_0 < \lambda_1 < \dots < \lambda_s$ are some numbers depending on $0 < \alpha < 1$ and p . Hence we may obtain the starting weights by solving the following linear system, with $u(t) = t^{\lambda_l}, l = 0, 1, \dots, s$,

$${}^R D_t^\alpha u(t_n) = \tau^{-\alpha} \sum_{j=0}^n w_{n-j} u(t_j) + \sum_{j=0}^s w_{n,j} u(t_j), \quad p = 2, \dots, 6. \quad (1.8)$$

For each fixed $n \geq 1, n \geq s$, we need to solve the linear system (1.8) to determine the starting weights $w_{n,j}, j = 0, 1, \dots, s$ in order to calculate the approximate solutions of the considered fractional differential equations. This idea works well for solving the following nonlinear fractional differential equation, with $0 < \alpha < 1$,

$${}^C D_t^\alpha y(t) = f(t, y(t)), \quad \text{with } y(0) = y_0, \quad (1.9)$$

since the solution of (1.9) is well understood and it can be expressed as the following form, for sufficiently smooth f ,

$$u(t) = c_0 t^{\lambda_0} + c_1 t^{\lambda_1} + \dots + c_s t^{\lambda_s} + \psi(t),$$

for some $0 = \lambda_0 < \lambda_1 < \dots < \lambda_s$. Here $\psi(t)$ is some smooth function and $c_i \in \mathbb{R}, i = 0, 1, \dots, s, s \in \mathbb{Z}^+$. Diethelm *et al.* [12] and Weilbeer [49] discussed the implementation of this idea in detail for solving (1.9), see also [3], [54], [6], [58], [61].

More recently, Zeng *et al.* [65] extended this idea to construct the higher order time discretisation scheme for solving time fractional partial differential equation (1.1). Under the assumption that the solution $u(t, x)$ of (1.1) satisfies

$$u(t, x) - u(0, x) = \sum_{j=0}^s c_j(x) t^{\gamma_j} + c_{s+1}(x) t^{\gamma_{s+1}} + \dots, \quad (1.10)$$

for some $0 < \gamma_0 < \gamma_1 < \dots < \gamma_s, s \in \mathbb{Z}^+$, Zeng *et al.* [65] constructed a second order numerical method for solving (1.1) by using starting weights and the weighted shifted Grünwald-Letnikov formula for both smooth and nonsmooth solutions. However Jin *et al.* [23] mentioned that the assumption (1.10) requires imposing certain compatibility conditions on the source term f which may not be satisfied for the general f in (1.1).

Without using the assumption (1.10) and the starting weights, we may also obtain higher order time discretisation schemes by using Approach 4, *i.e.*, by correcting the starting steps of the standard time discretisation schemes when the solutions of (1.1) are not smooth. This idea was first used in Lubich *et al.* [33] to construct a second order time discretisation scheme for solving diffusion wave equation, see (3.30) and (3.31) in Section 3 below. They proved the error estimates by using the Laplace transform method. Recently Jin *et al.* [22], [23] and Yan *et al.* [50], [52], [53] used the similar ideas as in Lubich *et al.* [33] to construct higher order time discretisation methods for solving (1.1). In this paper, we will review some recent results for constructing the time discretisation scheme for solving (1.1) by using Approach 4 when the solutions of (1.1) are not smooth.

The paper is organised as follows. In Section 2, we review the corrected Lubich's fractional multistep methods for solving time fractional partial differential equations. In Section 3, we review the corrected L1 scheme (or Diethelm's finite difference method) and its modifications for solving time fractional partial differential equations. In Section 4, we consider the discontinuous Galerkin method for solving time fractional partial differential equations. Finally we propose some open problems and issues for further work.

Throughout, the notations C and c , with or without a subscript, denote generic constants, which may differ at different occurrences, but are always independent of the step size k .

2. Correction of the Lubich fractional multistep methods

To obtain a higher order time discretisation method for solving (1.1), Jin *et al.* [23] introduced some corrected backward difference formula (BDF) for solving (1.1).

To see this, we first note that (1.1) is equivalent to the following form

$${}_0^R D_t^\alpha (u(t) - u(0)) + Au(t) = f(t), \quad \text{for } 0 < t \leq T, \quad \text{with } u(0) = u_0. \quad (2.11)$$

Let $U^n \approx u(t_n)$ denote the approximate solution of $u(t_n)$. We define the following time discretisation scheme for solving (2.11), with $n \geq 1$,

$$\tau^{-\alpha} \sum_{j=0}^n w_{n-j} (U^j - U^0) + AU^n = f(t_n), \quad \text{with } U^0 = u_0. \quad (2.12)$$

Jin *et al.* [23] showed that $\|U^n - u(t_n)\|$ has only first order accuracy even if the initial data u_0 and the source term f are sufficiently smooth.

To restore the p th accuracy with $p = 2, 3, \dots, 6$, Jin *et al.* [23] corrected the BDF $_p$ at the starting $p-1$ steps by (as usual, the summation disappears if the upper index is smaller than the lower one)

$$\begin{aligned} & \tau^{-\alpha} \sum_{j=0}^n w_{n-j} (U^j - U^0) + AU^n \\ &= f(t_n) + a_n^{(p)} (Au_0 + f(0)) + \tau^l \sum_{l=1}^{p-2} b_{l,n}^{(p)} f^{(l)}(0), \quad 1 \leq n \leq p-1, \end{aligned} \quad (2.13)$$

$$\tau^{-\alpha} \sum_{j=0}^n w_{n-j} (U^j - U^0) + AU^n = f(t_n), \quad p \leq n \leq N, \quad (2.14)$$

$$U^0 = u_0, \quad (2.15)$$

where $a_n^{(p)}$ and $b_{l,n}^{(p)}$ are coefficients determined in Tables 1 and 2, respectively. They are constructed so as to improve the accuracy of the overall scheme to $O(\tau^p)$ for general initial data $u_0 \in \mathcal{D}(A)$ and a possibly incompatible right-hand side f .

order of BDF	$a_1^{(p)}$	$a_2^{(p)}$	$a_3^{(p)}$	$a_4^{(p)}$	$a_5^{(p)}$
$p = 2$	$\frac{1}{2}$				
$p = 3$	$\frac{11}{12}$	$-\frac{5}{12}$			
$p = 4$	$\frac{31}{24}$	$-\frac{7}{6}$	$\frac{3}{8}$		
$p = 5$	$\frac{1181}{720}$	$-\frac{177}{80}$	$\frac{341}{240}$	$-\frac{251}{720}$	
$p = 6$	$\frac{2837}{1440}$	$-\frac{2543}{720}$	$\frac{17}{5}$	$-\frac{1201}{720}$	$\frac{95}{288}$

TABLE 1. The coefficients $a_j^{(p)}$ in (2.13)-(2.15) given in [23, Table 1]

order of BDF		$b_{l,1}^{(p)}$	$b_{l,2}^{(p)}$	$b_{l,3}^{(p)}$	$b_{l,4}^{(p)}$	$b_{l,5}^{(p)}$
$p = 3$	$l = 1$	$\frac{1}{12}$	0			
$p = 4$	$l = 1$	$\frac{1}{6}$	$-\frac{1}{12}$	0		
	$l = 2$	0	0	0		
$p = 5$	$l = 1$	$\frac{59}{240}$	$-\frac{29}{120}$	$\frac{19}{240}$	0	
	$l = 2$	$\frac{1}{240}$	$-\frac{1}{240}$	0	0	
	$l = 3$	$\frac{1}{720}$	0	0	0	
$p = 6$	$l = 1$	$\frac{77}{240}$	$-\frac{7}{15}$	$\frac{73}{240}$	$-\frac{3}{14}$	0
	$l = 2$	$\frac{1}{96}$	$-\frac{1}{60}$	$\frac{1}{160}$	0	0
	$l = 3$	$-\frac{1}{360}$	$\frac{1}{720}$	0	0	0
	$l = 4$	0	0	0	0	0

TABLE 2. The coefficients $b_{l,j}^{(p)}$ in (2.13)-(2.15) given in [23, Table 2]

THEOREM 2.1. ([23, Theorem 2.2]) Let $u(t_n)$ and U^n be the solutions of (2.11) and (2.13)-(2.15), respectively. Then we have, with $p = 1, 2, \dots, 6$,

$$\begin{aligned} \|U^n - u(t_n)\| \leq & C\tau^p \left(t_n^{\alpha-p} \|f(0) + Au_0\| + \sum_{l=1}^{p-1} t_n^{\alpha+l-p} \|f^{(l)}(0)\| \right. \\ & \left. + \int_0^{t_n} (t_n - s)^{\alpha-1} \|f^{(p)}(s)\| ds \right). \end{aligned}$$

REMARK 2.1. ([23, Remark 2.3]) For any fixed $t_n > 0$, by Theorem 2.1, the corrected BDFp method (2.13)-(2.15) has convergence rate $O(\tau^p)$, $p = 1, 2, \dots, 6$. To get a uniform rate $O(\tau^p)$, one may need the following compatibility conditions [32]

$$f(0) + Au(0) = 0, \quad \text{and } f^{(l)}(0) = 0, \quad l = 1, 2, \dots, p-1. \quad (2.16)$$

When these conditions do not hold, the error estimates may deteriorate as $t \rightarrow 0$ and we cannot obtain the uniform rate $O(\tau^p)$, $p = 1, 2, \dots, 6$.

REMARK 2.1. ([23, Remark 2.4]) When the initial value $u_0 \in H = L_2(\Omega)$, we may obtain similar error estimates as in Theorem 2.1. More precisely, we have

$$\begin{aligned} \|U^n - u(t_n)\| \leq & C\tau^p \left(t_n^{-p} \|f(0) + u_0\| + \sum_{l=1}^{p-1} t_n^{\alpha+l-p} \|f^{(l)}(0)\| \right. \\ & \left. + \int_0^{t_n} (t_n - s)^{\alpha-1} \|f^{(p)}(s)\| ds \right). \end{aligned}$$

We see that if the initial value u_0 is nonsmooth, *i.e.*, $u_0 \in H$, then in the error estimates, $t_n^{\alpha-p}$ reduces to t_n^{-p} , in other words, t_n loses α th power if u_0 is nonsmooth.

3. Correction of L1 scheme and its modification

3.1. Correction of the L1 scheme

Another approach to approximate the fractional derivative is by using the L1 scheme (or the Diethelm finite difference method). The L1 scheme first appeared in the book [41] for the approximation of the Caputo fractional derivative. The L1 scheme may be obtained by direct approximation of the derivative in the definition of the Caputo fractional derivative, *e.g.*, [28], [26], [18], [29], [47], [19], or by the approximation of the Hadamard finite-part integral, *e.g.*, [9], [11], [15], [16], [17], [24], [51].

Let us first consider how to obtain the L1 scheme by the direct approximation of the derivative in the definition of the Caputo fractional derivative. Note that, with $0 < \alpha < 1$,

$$\begin{aligned} {}_0^C D_t^\alpha u(t_n) &= \frac{1}{\Gamma(1-\alpha)} \int_0^{t_n} (t_n - s)^{-\alpha} u'(s) ds \\ &= \frac{1}{\Gamma(1-\alpha)} \sum_{k=0}^{n-1} \int_{t_k}^{t_{k+1}} (t_n - s)^{-\alpha} u'(s) ds, \end{aligned}$$

where $u'(s) = \frac{du}{ds}$.

Approximating $u'(s)$ on $[t_k, t_{k+1}]$ by using the finite difference quotient $(u(t_{k+1}) - u(t_k))/\tau$, we obtain the following so-called L1 scheme

$${}_0^C D_t^\alpha u(t_n) \approx \frac{1}{\Gamma(1-\alpha)} \int_0^{t_n} (t_n - s)^{-\alpha} \frac{u(t_{k+1}) - u(t_k)}{\tau} ds = \tau^{-\alpha} \sum_{j=0}^n w_{n-j,n} u(t_j), \quad (3.17)$$

where $w_{j,n}, j = 0, 1, 2, \dots, n$ are given by

$$\Gamma(2-\alpha)w_{j,n} = \begin{cases} 1, & \text{for } j = 0, \\ -2j^{1-\alpha} + (j-1)^{1-\alpha} + (j+1)^{1-\alpha}, & \text{for } j = 1, 2, \dots, n-1, \\ (j-1)^{1-\alpha} - j^{1-\alpha}, & \text{for } j = n. \end{cases}$$

Lin *et al.* [28] and Sun *et al.* [47] independently proved that the truncation error of the L1 scheme is $O(\tau^{2-\alpha})$ for a sufficiently smooth function u .

We next consider how to obtain the L1 scheme by approximating the Hadamard finite-part integral. Noting that Riemann-Liouville fractional derivative can be expressed by the Hadamard finite-part integral, Diethelm [9], [10] approximated the Hadamard finite-part integral by using the piecewise interpolation polynomials and obtained the approximate schemes to the Riemann-Liouville fractional derivatives. More precisely, we have, with $0 < \alpha < 1$,

$${}_0^R D_t^\alpha u(t) = \frac{1}{\Gamma(-\alpha)} \oint_0^t (t-s)^{-\alpha-1} u(s) ds, \quad (3.18)$$

where the integral \oint is in the Hadamard sense [9], [10]. At $t = t_n$, we have

$${}_0^R D_t^\alpha u(t_n) = \frac{1}{\Gamma(-\alpha)} \sum_{k=0}^{n-1} \oint_{t_k}^{t_{k+1}} (t_n - s)^{-\alpha-1} u(s) ds.$$

Approximating $u(s)$ on $[t_k, t_{k+1}]$ by using the linear interpolation polynomial

$$P_1(s) = \frac{s - t_{k+1}}{t_k - t_{k+1}} u(t_k) + \frac{s - t_k}{t_{k+1} - t_k} u(t_{k+1}),$$

we get

$$\begin{aligned} {}_0^C D_t^\alpha u(t_n) &= {}_0^R D_t^\alpha (u(t_n) - u(0)) \\ &\approx \frac{1}{\Gamma(-\alpha)} \sum_{k=0}^{n-1} \oint_{t_k}^{t_{k+1}} (t_n - s)^{-\alpha-1} P_1(s) ds - {}_0^R D_t^\alpha u(0) = \tau^{-\alpha} \sum_{j=0}^n w_{n-j,n} u(t_j), \end{aligned} \quad (3.19)$$

where the weights $w_{j,n}, j = 0, 1, \dots, n$ are exactly the same as the weights defined in (3.17). Hence we conclude that, when using the piecewise linear interpolation polynomial to approximate the Hadamard finite-part integral,

the scheme (3.19) for approximating the Riemann-Liouville fractional derivative is the same as the L1 scheme (3.17) for approximating the Caputo fractional derivative.

We now define the L1 scheme for solving (2.11). From now on, we only consider the homogeneous case with $f = 0$. The inhomogeneous case can be discussed similarly by using the argument in Jin *et al.* [20], [22], [23].

Let $u(t) - u_0 = V(t)$. Then solving (1.1) with $f = 0$ is equivalent to find $V(t)$ such that, noting that ${}_0^C D_t^\alpha u_0 = 0$,

$${}_0^C D_t^\alpha V(t) + AV(t) = -Au_0, \quad \text{with } V(0) = 0. \quad (3.20)$$

It proves more convenient to consider the error estimates of the time discretisation scheme of (3.20), see [33].

Let $V^j \approx V(t_j)$ be the approximate solution of $V(t_j)$ in (3.20). We define the following L1 scheme for solving (3.20),

$$\tau^{-\alpha} \sum_{j=0}^n w_{n-j,n} V^j + AV^n = -Au_0, \quad n \geq 1, \quad \text{with } V^0 = 0, \quad (3.21)$$

or

$$\tau^{-\alpha} \sum_{j=1}^n w_{n-j,n} V^j + AV^n = -Au_0, \quad n \geq 1. \quad (3.22)$$

For any fixed $n \geq 1$, we observe that $w_{j,n}, j = 0, 1, \dots, n-1$ only depend on $j = 0, 1, 2, \dots, n-1$. For example, we have $w_{0,n} = 1/\Gamma(2-\alpha)$ for any $n \geq 1$, $w_{1,n} = 1/\Gamma(2-\alpha)((-2)1^{1-\alpha} + (1-1)^{1-\alpha} + (1+1)^{1-\alpha})$ for any $n \geq 2, \dots$. Therefore, it is reasonable to write $w_{j,n}, j = 0, 1, 2, \dots, n-1$ with w_j since $w_{j,n}, j = 0, 1, 2, \dots, n-1$ are independent of n . In other words, we denote $w_0 = w_{0,n}, w_1 = w_{1,n}, w_2 = w_{2,n}, \dots, w_{n-1} = w_{n-1,n}$ for any fixed $n \geq 1$. Further we define w_j for all $j = 0, 1, 2, \dots$ by the following

$$\Gamma(2-\alpha)w_j = \begin{cases} 1, & \text{for } j = 0, \\ -2j^{1-\alpha} + (j-1)^{1-\alpha} + (j+1)^{1-\alpha}, & \text{for } j = 1, 2, \dots \end{cases} \quad (3.23)$$

Using the notations $w_j, j = 0, 1, 2, \dots$ defined in (3.23), the time discretisation scheme (3.22) can now be written as

$$\tau^{-\alpha} \sum_{j=1}^n w_{n-j} V^j + AV^n = -Au_0, \quad n \geq 1. \quad (3.24)$$

THEOREM 3.1. ([21, Theorem 3.16]) Let $V(t_n)$ and V^n be the solutions of (3.20) and (3.24), respectively. Let $u_0 \in H = L_2(\Omega)$. Then we have,

with $0 < \alpha < 1$,

$$\|V(t_n) - V^n\| \leq C\tau t_n^{-1} \|u_0\|. \quad (3.25)$$

To obtain a higher order method for solving (1.4), we correct the starting steps of the L1 scheme as used in Jin *et al.* [23]. Let $c_0 = 1/2$, we correct the L1 scheme for solving (3.20) as follows

$$\tau^{-\alpha} \sum_{j=1}^n w_{n-j} V^j + AV^n = (-Au_0)(1 + c_0), \text{ for } n = 1, \quad (3.26)$$

$$\tau^{-\alpha} \sum_{j=1}^n w_{n-j} V^j + AV^n = -Au_0, \text{ for } n \geq 2, \quad (3.27)$$

$$V^0 = 0. \quad (3.28)$$

where the weights $w_{n-j}, j = 1, 2, \dots, n$ are given by (3.23).

The idea of introducing the correction term in the first step $n = 1$ in (3.26) comes from the idea in Lubich *et al.* [33] where the authors introduced a modified scheme to construct a higher order time discretisation scheme for solving an evolution equation with a positive-type memory term. To see this, let us first note that (3.20) is equivalent to the following form, with $0 < \alpha < 1$,

$$V(t) + {}_0^R D_t^{-\alpha}(AV(t)) = -{}_0^R D_t^{-\alpha}(Au_0), \quad \text{with } V(0) = 0, \quad (3.29)$$

where ${}_0^R D_t^{-\alpha}V(t)$ denotes the Riemann-Liouville fractional integral. To obtain a higher order time discretisation scheme for solving (3.29), following the idea in Lubich *et al.* [33], we may introduce the following modified time discretisation scheme to approximate (3.29),

$$V^n + q_n^c(AV) = -q_n^c(Au_0), \quad \text{with } V^0 = 0, \quad (3.30)$$

where $q_n^c(\varphi)$ is the modification of the quadrature formula approximating the Riemann-Liouville fractional integral ${}_0^R D_t^{-\alpha}\varphi$, defined by

$$q_n^c(\varphi) = k^{-\alpha} \sum_{k=1}^n \beta_{n-k} \varphi^k + c_0 \beta_{n-1} \varphi^0, \quad \text{with } c_0 = 1/2. \quad (3.31)$$

Here β_0, β_1, \dots are generated by some function $\hat{\beta}(\zeta) = \sum_{j=0}^{\infty} \beta_j \zeta^j$.

LEMMA 3.1. ([52]) Assume that $\hat{\beta}(\zeta) = (\sum_{j=0}^{\infty} w_j \zeta^j)^{-1}$, where $w_j, j = 0, 1, 2, \dots$ are defined in (3.23). Then the modified L1 scheme (3.26)-(3.28) is equivalent to (3.30).

REMARK 3.1. From Lemma 3.1, we note that the correction on the first step $n = 1$ in (3.26)-(3.28) is equivalent to the correction in (3.30). Therefore we see that the corrected L1 scheme (3.26)-(3.28) is actually equivalent to the modified scheme (3.30) which has been used to improve the convergence rate of the time discretisation scheme for solving an evolution equation with a positive-type memory term in Lubich *et al.* [33].

THEOREM 3.2. ([52]) Let $V(t_n)$ and V^n be the solutions of (3.20) and (3.26)-(3.28), respectively. Let $u_0 \in H = L_2(\Omega)$. We have

$$\|V(t_n) - V^n\| \leq C\tau^{2-\alpha}t_n^{\alpha-2}\|u_0\|.$$

3.2. Numerical examples

In this section, we will consider the experimentally determined convergence rates of the L1 and the corrected L1 schemes for smooth and nonsmooth data. We only show the numerical results for the homogeneous problem here. Similarly we may consider the numerical results for the inhomogeneous problem with $f \neq 0$ as discussed in Jin *et al.* [20], [23]. Let us consider

$${}_0^C D_t^\alpha u(x, t) - u_{xx} = 0, \quad 0 < x < 1, \quad t > 0, \quad (3.32)$$

$$u(0, t) = u(1, t) = 0, \quad (3.33)$$

$$u(x, 0) = u_0(x), \quad (3.34)$$

where $u_0(x) = x(1-x)$ or $u_0(x) = \chi_{(0,1/2)}$.

Let $0 < t_0 < t_1 < \dots < t_N = T$ be the time partition and τ the time step size. Let N_h be a positive integer. Let $0 = x_0 < x_1 < x_2 < \dots < x_{N_h} = 1$ be the space partitions and h the space step size. We will use finite element methods to consider the spatial discretisation.

We first consider the scheme (3.24) and the convergence rate was proved to be $O(\tau)$ for both smooth and nonsmooth data in [21].

To observe this convergence order, we first calculate the reference exact solution $u_{ref}(t)$ at $T = 1$ with $h_{ref} = 2^{-6}$ and $\tau_{ref} = 2^{-10}$. We then use $h = 2^{-6}$ and $\tau = \text{kappa} * \tau_{ref}$ with $\text{kappa} = [2^2, 2^3, 2^4, 2^5, 2^6]$ to obtain the approximate solutions $u(t)$ at $t = 1$. We choose the smooth and nonsmooth initial data (a) $u_0 = x(1-x)$ and the nonsmooth data (b) $u_0 = \chi_{(0,1/2)}$ we obtain the following results which are consistent with the Table 1 in [21]. In Table 3, we observe that the convergence rate indeed is almost $O(\tau)$ for the different $\alpha \in (0, 1)$ in both smooth and nonsmooth data cases.

We next consider the corrected L1 scheme (3.26)-(3.28). By Theorem 3.2, the convergence rate of the corrected L1 scheme (3.26)-(3.28) is $O(\tau^{2-\alpha})$ for smooth and nonsmooth data. To observe this convergence

α		$\tau = 2^{-8}$	$\tau = 2^{-7}$	$\tau = 2^{-6}$	$\tau = 2^{-5}$	$\tau = 2^{-4}$	Rate
0.1	(a)	0.0212e-4	0.0496e-4	0.1067e-4	0.2218e-4	0.4564e-4	1.1063
	(b)	0.0055e-3	0.0127e-3	0.0274e-3	0.0570e-3	0.1172e-3	1.1063
0.3	(a)	0.0056e-3	0.0130e-3	0.0280e-3	0.0585e-3	0.1209e-3	1.1100
	(b)	0.0143e-3	0.0333e-3	0.0718e-3	0.1479e-3	0.3094e-3	1.1099
0.8	(a)	0.0078e-3	0.0185e-3	0.0403e-3	0.0857e-3	0.1824e-3	1.1359
	(b)	0.0198e-3	0.0466e-3	0.1017e-3	0.2160e-3	0.4595e-3	1.1350
0.9	(a)	0.0054e-3	0.0128e-3	0.0284e-3	0.0621e-3	0.1404e-3	1.1766
	(b)	0.0134e-3	0.0320e-3	0.0708e-3	0.1546e-3	0.3490e-3	1.1757

TABLE 3. Time convergence rates with the different α for the L1 scheme (3.24)

rate, we first calculate the reference exact solution $uref(t)$ at $T = 1$ with $h_{ref} = 2^{-6}$ and $\tau_{ref} = 2^{-10}$. We then use $h = 2^{-6}$ and $\tau = kappa * \tau_{ref}$ with $kappa = [2^2, 2^3, 2^4, 2^5, 2^6]$ to obtain the approximate solutions $u(t)$ at $t = 1$. We choose the smooth and nonsmooth initial data (a) $u_0 = \sin(2\pi x)$ and (b) $u_0 = \chi_{(0,1/2)}$ we obtain the following results.

We found that in Table 4 the corrected L1 scheme has the better accuracy than the L1 scheme and the errors are about $1e - 05$ or $1e - 04$ for all $\alpha \in (0, 1)$. The errors of the L1 scheme are only $1e - 03$. For the convergence rates, when $\alpha < 1/2$, we observe that, in Table 4, the convergence rates are almost 2 which is better than the theoretical results $2 - \alpha$. However when $\alpha > 1/2$, the convergence rates are almost $2 - \alpha$ as we expected.

3.3. Correction of the modified L1 scheme

In this section we will introduce a corrected higher order time discretisation scheme (which we call the modified L1 scheme) for solving (3.20) and show the error estimates for the nonsmooth data [50]. Let us first introduce a higher order scheme (the modified L1 scheme) to approximate the Riemann-Liouville derivative by using Diethelm’s approach [9], [10].

LEMMA 3.2. Let $0 < \alpha < 1$ and assume that $u \in C^3[0, T]$. Let $n \geq 5$, then we have

$${}_0^R D_t^\alpha u(t_n) = \tau^{-\alpha} \sum_{j=0}^n w_{j,n} u(t_{n-j}) + O(\tau^{3-\alpha}),$$

α		$\tau = 2^{-8}$	$\tau = 2^{-7}$	$\tau = 2^{-6}$	$\tau = 2^{-5}$	$\tau = 2^{-4}$	Rate
0.1	(a)	0.0007e-5	0.0030e-5	0.0125e-5	0.0517e-5	0.2197e-5	2.0674
	(b)	0.0018e-5	0.0078e-5	0.0322e-5	0.1333e-5	0.5658e-5	2.0668
0.3	(a)	0.0013e-5	0.0064e-5	0.0291e-5	0.1302e-5	0.5891e-5	2.1914
	(b)	0.0004e-4	0.0017e-4	0.0076e-4	0.0339e-4	0.1527e-4	2.1839
0.8	(a)	0.0079e-4	0.0201e-4	0.0462e-4	0.0981e-4	0.1782e-4	1.1223
	(b)	0.0196e-4	0.0496e-4	0.1140e-4	0.2421e-4	0.4407e-4	1.1230
0.9	(a)	0.0141e-4	0.0345e-4	0.0778e-4	0.1687e-4	0.3484e-4	1.1573
	(b)	0.0347e-4	0.0851e-4	0.1920e-4	0.4162e-4	0.8597e-4	1.1572

TABLE 4. Time convergence rates with the different α for the modified L1 scheme (3.26)-(3.28)

where

$$\Gamma(3-\alpha)w_{j,n} = \begin{cases} (1 + \frac{\alpha}{2})2^{1-\alpha}, & j = 0, \\ (2 + \frac{\alpha}{2})3^{1-\alpha} - (3 + \frac{3}{2}\alpha)2^{1-\alpha}, & j = 1, \\ (3 + \frac{\alpha}{2})4^{1-\alpha} - (6 + \frac{3}{2}\alpha)3^{1-\alpha} + (3 + \frac{3}{2}\alpha)2^{1-\alpha}, & j = 2, \end{cases}$$

and, with $j = 3, 4, \dots, n-2$,

$$\begin{aligned} \Gamma(3-\alpha)w_{j,n} &= (j+2)^{2-\alpha} - (1 - \frac{\alpha}{2})(j+2)^{1-\alpha} - 3(j+1)^{2-\alpha} \\ &\quad + (3 - \frac{3}{2}\alpha)(j+1)^{1-\alpha} + 3j^{2-\alpha} - (3 - \frac{3\alpha}{2})j^{1-\alpha} - (j-1)^{2-\alpha} \\ &\quad + (1 - \frac{\alpha}{2})(j-1)^{1-\alpha}, \quad j = 3, 4, \dots, n-2, \end{aligned}$$

and

$$\begin{aligned} \Gamma(3-\alpha)w_{n-1,n} &= (4 - 2n - 2\alpha)n^{1-\alpha} + (3n - 6 + \frac{3\alpha}{2})(n-1)^{1-\alpha} - (n + \frac{\alpha}{2} - 3)(n-2)^{1-\alpha}, \\ \Gamma(3-\alpha)w_{n,n} &= (n - 3 + \frac{3\alpha}{2})n^{1-\alpha} + (1 - \alpha)(2 - \alpha)n^{-\alpha} - (n - 2 + \frac{\alpha}{2})(n-1)^{1-\alpha}. \end{aligned}$$

P r o o f. Note that the Riemann-Liouville fractional derivative ${}^R_0D_t^\alpha u(t)$, $0 < \alpha < 1$ may be written as

$${}^R_0D_t^\alpha u(t) = \frac{1}{\Gamma(1-\alpha)} \frac{d}{dt} \int_0^t (t-s)^{-\alpha} u(s) ds = \frac{1}{\Gamma(-\alpha)} \oint_0^t (t-s)^{-\alpha-1} u(s) ds,$$

where the integral \oint must be interpreted as a Hadamard finite-part integral, see *e.g.*, [9], [10],

At $t = t_n, n = 5, 6, \dots, N$, we have

$$\begin{aligned} {}_0^R D_t^\alpha u(t_n) &= \frac{1}{\Gamma(-\alpha)} \oint_0^{t_n} (t_n - s)^{-\alpha-1} u(s) ds = \frac{t_n^{-\alpha}}{\Gamma(-\alpha)} \oint_0^1 w^{-\alpha-1} u(t_n - t_n w) dw \\ &= \frac{t_n^{-\alpha}}{\Gamma(-\alpha)} \sum_{j=1}^n \oint_{w_{j-1}}^{w_j} w^{-\alpha-1} u(t_n - t_n w) dw. \end{aligned}$$

Denote $g(w) = u(t_n - t_n w)$ and approximate $g(w)$ by the piecewise quadratic interpolation polynomial $g_2(w)$ defined on the nodes $w_l = \frac{l}{n}, l = 0, 1, 2, \dots, n$ by

$$\begin{aligned} g_2(w) &= \frac{(w - w_1)(w - w_2)}{(w_0 - w_1)(w_0 - w_2)} g(w_0) + \frac{(w - w_0)(w - w_2)}{(w_1 - w_0)(w_1 - w_2)} g(w_1) \\ &\quad + \frac{(w - w_0)(w - w_1)}{(w_2 - w_0)(w_2 - w_1)} g(w_2), \quad \text{for } w \in [w_0, w_1], \end{aligned}$$

and

$$\begin{aligned} g_2(w) &= \frac{(w - w_{j-1})(w - w_j)}{(w_{j-2} - w_{j-1})(w_{j-2} - w_j)} g(w_{j-2}) + \frac{(w - w_{j-2})(w - w_j)}{(w_{j-1} - w_{j-2})(w_{j-1} - w_j)} g(w_{j-1}) \\ &\quad + \frac{(w - w_{j-2})(w - w_{j-1})}{(w_j - w_{j-2})(w_j - w_{j-1})} g(w_j), \quad \text{for } w \in [w_{j-1}, w_j], j = 2, 3, \dots, n. \end{aligned}$$

More precisely, on $[w_{j-1}, w_j], j = 2, 3, \dots, n$, $g(w)$ is approximated by the quadratic interpolation polynomial $g_2(w)$ determined by $g(w_{j-2}), g(w_{j-1}), g(w_j)$, and on $[w_0, w_1]$, $g(w)$ is approximated by the quadratic interpolation polynomial $g_2(w)$ determined by $g(w_0), g(w_1), g(w_2)$.

By [10, Theorem 2.4], we have the following error estimates

$${}_0^R D_t^\alpha u(t_n) = \frac{t_n^{-\alpha}}{\Gamma(-\alpha)} \oint_0^1 w^{-\alpha-1} g(w) dw = \frac{t_n^{-\alpha}}{\Gamma(-\alpha)} \oint_0^1 w^{-\alpha-1} g_2(w) dw + O(k^{3-\alpha}). \quad (3.35)$$

We next calculate the integral on the right side of (3.35). Note that

$$\oint_0^1 w^{-\alpha-1} g_2(w) dw = \oint_0^{w_1} w^{-\alpha-1} g_2(w) dw + \sum_{j=2}^n \int_{w_{j-1}}^{w_j} w^{-\alpha-1} g_2(w) dw, \quad (3.36)$$

where only the first integral on the right side of (3.36) is in the sense of the Hadamard finite-part integral and the other integrals on the right side of (3.36) are in the usual sense. By using the definition of the Hadamard

finite-part integral [9], [10], we have

$$\begin{aligned} \oint_0^{w_1} w^{-\alpha-1} g_2(w) dw &= \oint_0^{w_1} w^{-\alpha-1} (g_2(0) + \int_0^{w_1} g_2'(y) dy) dw \\ &= \frac{w_1^{-\alpha}}{-\alpha} g_2(0) + \int_0^{w_1} w^{-\alpha-1} \left(\int_0^{w_1} g_2'(y) dy \right) dw, \end{aligned}$$

where

$$\begin{aligned} g_2'(y) &= (n^2/2)(2y - (w_1 + w_2))g(0) - n^2(2y - w_2)g(w_1) \\ &\quad + (n^2/2)(2y - w_1)g(w_2). \end{aligned}$$

Hence

$$\begin{aligned} \oint_0^{w_1} w^{-\alpha-1} g_2(w) dw &= \frac{1}{(-\alpha)(-\alpha+1)(-\alpha+2)n^{-\alpha}} \\ &\quad \cdot ((2 - \alpha/2)g(0) + (-\alpha)(3 - \alpha)g(w_1) + (\alpha/2)g(w_2)). \end{aligned}$$

For the other integrals in (3.36), we have, with $j = 2, 3, \dots, n$,

$$\begin{aligned} \int_{w_{j-1}}^{w_j} w^{-\alpha-1} g_2(w) dw &= (n^2/2)g(w_{j-2}) \int_{w_{j-1}}^{w_j} w^{-\alpha-1} (w - w_{j-1})(w - w_j) dw \\ &\quad - n^2g(w_{j-1}) \int_{w_{j-1}}^{w_j} w^{-\alpha-1} (w - w_{j-2})(w - w_j) dw \\ &\quad + (n^2/2)g(w_j) \int_{w_{j-1}}^{w_j} w^{-\alpha-1} (w - w_{j-2})(w - w_{j-1}) dw \\ &= \frac{1}{(-\alpha)(-\alpha+1)(-\alpha+2)n^{-\alpha}} \left(\frac{1}{2}E_j g(w_{j-2}) - F_j g(w_{j-1}) + \frac{1}{2}G_j g(w_j) \right), \end{aligned}$$

where

$$\begin{aligned} E_j &= 2j^{-\alpha+2} - (-\alpha+2)j^{-\alpha+1} - 2(j-1)^{-\alpha+2} - (-\alpha+2)(j-1)^{-\alpha+1}, \\ F_j &= 2j^{-\alpha+2} - 2(-\alpha+2)j^{-\alpha+1} - 2(j-1)^{-\alpha+2} + (-\alpha+1)(-\alpha+2)(j-1)^{-\alpha}, \\ G_j &= 2j^{-\alpha+2} - 3(-\alpha+2)j^{-\alpha+1} - 2(j-1)^{-\alpha+2} \\ &\quad + (-\alpha+2)(j-1)^{-\alpha+1} + 2(-\alpha+1)(-\alpha+2)j^{-\alpha}. \end{aligned}$$

Thus we get

$$\oint_0^1 w^{-\alpha-1} g_2(w) dw = \sum_{j=0}^n \alpha_{jn} g(w_j),$$

where

$$(-\alpha)(-\alpha+1)(-\alpha+2)n^{-\alpha}\alpha_{jn} = \begin{cases} 2 - \frac{1}{2}\alpha + \frac{1}{2}E_2, & \text{for } j = 0, \\ (-\alpha)(3 - \alpha) + \frac{1}{2}E_3 - F_2, & \text{for } j = 1, \\ \frac{1}{2}\alpha + \frac{1}{2}E_4 - F_3 + \frac{1}{2}G_2, & \text{for } j = 2, \\ \frac{1}{2}E_{j+2} - F_{j+1} + \frac{1}{2}G_j, & \text{for } j = 3, \dots, n-2, \\ -F_n + \frac{1}{2}G_{n-1}, & \text{for } j = n-1, \\ \frac{1}{2}G_n. & \text{for } j = n. \end{cases}$$

Therefore we have

$$\begin{aligned} {}_0^R D_t^\alpha u(t_n) &= \frac{t_n^{-\alpha}}{\Gamma(-\alpha)} \int_0^1 w^{-\alpha-1} g_2(w) dw + O(k^{3-\alpha}) \\ &= \frac{t_n^{-\alpha}}{\Gamma(-\alpha)} \sum_{j=0}^n \alpha_{jn} g(w_j) + O(k^{3-\alpha}) \\ &= k^{-\alpha} \sum_{j=0}^n \frac{(-\alpha)(-\alpha+1)(-\alpha+2)n^{-\alpha}\alpha_{jn}}{\Gamma(3-\alpha)} g(w_j) + O(k^{3-\alpha}) \\ &= k^{-\alpha} \sum_{j=0}^n w_{jn} u(t_{n-j}) + O(k^{3-\alpha}), \end{aligned}$$

where, with $j = 0, 1, 2, \dots, n$, $n = 5, 6, \dots, N$,

$$\Gamma(3-\alpha)w_{j,n} = (-\alpha)(-\alpha+1)(-\alpha+2)n^{-\alpha}\alpha_{jn}.$$

More precisely, for $j = 0$, we have

$$\Gamma(3-\alpha)w_{0,n} = 2 - \frac{\alpha}{2} + \frac{1}{2}E_2 = \left(1 + \frac{\alpha}{2}\right)2^{1-\alpha}.$$

For $j = 1$, we have

$$\Gamma(3-\alpha)w_{1,n} = (-\alpha)(3-\alpha) + \frac{1}{2}E_3 - F_2 = \left(2 + \frac{\alpha}{2}\right)3^{1-\alpha} - \left(3 + \frac{3}{2}\alpha\right)2^{1-\alpha}.$$

For $j = 2$, we have

$$\begin{aligned} \Gamma(3-\alpha)w_{2,n} &= \frac{1}{2}\alpha + \frac{1}{2}E_4 - F_3 + \frac{1}{2}G_2 = \left(3 + \frac{\alpha}{2}\right)4^{1-\alpha} \\ &\quad - \left(6 + \frac{3}{2}\alpha\right)3^{1-\alpha} + \left(3 + \frac{3}{2}\alpha\right)2^{1-\alpha}. \end{aligned}$$

For $j = 3, 4, \dots, n-2$,

$$\begin{aligned}\Gamma(3-\alpha)w_{j,n} &= \frac{1}{2}E_{j+2} - F_{j+1} + \frac{1}{2}G_j = (j+2)^{2-\alpha} - (1-\frac{\alpha}{2})(j+2)^{1-\alpha} \\ &\quad - 3(j+1)^{2-\alpha} + (3-\frac{3}{2}\alpha)(j+1)^{1-\alpha} + 3j^{2-\alpha} - (3-\frac{3\alpha}{2})j^{1-\alpha} \\ &\quad - (j-1)^{2-\alpha} + (1-\frac{\alpha}{2})(j-1)^{1-\alpha}.\end{aligned}$$

For $j = n-1$, we have

$$\Gamma(3-\alpha)w_{j,n} = (4-2n-2\alpha)n^{1-\alpha} + (3n-6+\frac{3\alpha}{2})(n-1)^{1-\alpha} - (n+\frac{\alpha}{2}-3)(n-2)^{1-\alpha},$$

For $j = n$, we have

$$\Gamma(3-\alpha)w_{n,n} = (n-3+\frac{3\alpha}{2})n^{1-\alpha} + (1-\alpha)(2-\alpha)n^{-\alpha} - (n-2+\frac{\alpha}{2})(n-1)^{1-\alpha}.$$

Together these estimates complete the proof of Lemma 3.2. \square

We remark that for any fixed $n \geq 5$, $w_{j,n}, j = 0, 1, \dots, n-2$ are independent on n and only $w_{n-1,n}$ and $w_{n,n}$ depend on n . For example, we have $w_{0,n} = (1+\frac{\alpha}{2})2^{1-\alpha}$ for any $n \geq 5$, $w_{1,n} = (2+\frac{\alpha}{2})3^{1-\alpha} - (3+\frac{3}{2}\alpha)2^{1-\alpha}$ for any $n \geq 5, \dots$. Based on this observation, we may define $w_j, j = 0, 1, 2, \dots, n-1$ which we need in the numerical scheme (3.38)-(3.41) as follows. For $j = 0, 1, 2$, we define $w_0 = w_{0,n}, w_1 = w_{1,n}, w_2 = w_{2,n}$ with $n \geq 5$. For $w_j, j = 3, 4, \dots, n-2$, we define $w_j = w_{j,n}, j = 3, 4, \dots, n-2$, where

$$\begin{aligned}\Gamma(3-\alpha)w_{j,n} &= \frac{1}{2}E_{j+2} - F_{j+1} + \frac{1}{2}G_j = (j+2)^{2-\alpha} - (1-\frac{\alpha}{2})(j+2)^{1-\alpha} \\ &\quad - 3(j+1)^{2-\alpha} + (3-\frac{3}{2}\alpha)(j+1)^{1-\alpha} + 3j^{2-\alpha} - (3-\frac{3\alpha}{2})j^{1-\alpha} \\ &\quad - (j-1)^{2-\alpha} + (1-\frac{\alpha}{2})(j-1)^{1-\alpha}.\end{aligned}$$

For $w_j, j \geq n-1$, we also define w_j by

$$\begin{aligned}\Gamma(3-\alpha)w_j &= \frac{1}{2}E_{j+2} - F_{j+1} + \frac{1}{2}G_j = (j+2)^{2-\alpha} - (1-\frac{\alpha}{2})(j+2)^{1-\alpha} \\ &\quad - 3(j+1)^{2-\alpha} + (3-\frac{3}{2}\alpha)(j+1)^{1-\alpha} + 3j^{2-\alpha} - (3-\frac{3\alpha}{2})j^{1-\alpha} \\ &\quad - (j-1)^{2-\alpha} + (1-\frac{\alpha}{2})(j-1)^{1-\alpha}.\end{aligned}$$

Now for any fixed $n \geq 5$, we have defined all $w_j, j = 0, 1, 2, \dots, n-2, n-1, n, \dots$. We remark that in our numerical method (3.38)-(3.41), we only need the weights $w_j, j = 0, 1, 2, \dots, n-1$, where $w_j = w_{j,n}, j = 0, 1, 2, \dots, n-2$, but $w_{n-1} \neq w_{n-1,n}$. We shall see that in the proof of Theorem 3.3, we need to define all $w_j, j = 0, 1, 2, \dots, n-2, n-1, n, \dots$ in

order to use the discrete Laplace transform. For $w_j, j = 0, 1, 2, \dots$ defined above, we may introduce the characteristic function $\delta(\zeta)$

$$\delta(\zeta)^\alpha = \sum_{j=0}^{\infty} w_j \zeta^j. \quad (3.37)$$

We now introduce the following corrected higher order time discretisation scheme for solving (2.11), with $c_0 = 11/12, c_1 = -5/12,$

$$\tau^{-\alpha} \sum_{j=1}^n w_{n-j} V^j + AV^n = (-Au_0)(1 + c_0), \quad n = 1, \quad (3.38)$$

$$\tau^{-\alpha} \sum_{j=1}^n w_{n-j} V^j + AV^n = (-Au_0)(1 + c_1), \quad n = 2, \quad (3.39)$$

$$\tau^{-\alpha} \sum_{j=1}^n w_{n-j} V^j + AV^n = -Au_0, \quad n = 3, 4, \dots, N, \quad (3.40)$$

$$V^0 = 0, \quad (3.41)$$

where $w_j, j = 0, 1, 2, \dots$ are determined by (3.37) below.

REMARK 3.2. Note that the weights $w_j, j = 0, 1, 2, \dots$ in (3.38)-(3.41) are deduced based on Lemma 3.2 for the approximation of Riemann-Liouville fractional derivative ${}_0^R D_t^\alpha u(t_n)$ with $n \geq 5$. Therefore one needs to choose at least $N \geq 5$ for the numerical method (3.38)-(3.41).

THEOREM 3.3. ([50]) Let $c_0 = 11/12, c_1 = -5/12$ and let $V(t_n)$ and V^n be the solutions of (3.20) and (3.38)-(3.41), respectively. Let $u_0 \in L_2(\Omega)$. Then there exists a positive constant C such that

$$\|V^n - V(t_n)\| \leq Ck^{3-\alpha}t_n^{\alpha-3}\|u_0\|, \quad n \geq 5.$$

4. Correction of the discontinuous Galerkin method in McLean and Mustapha [35]

In this section, we will consider the discontinuous Galerkin method for solving (1.1) introduced by McLean and Mustapha [35]. We first obtain the same time discretisation scheme as in McLean and Mustapha [35] by approximating the Riemann-Liouville fractional derivative with the L1 method (or Diethelm's finite difference method). We then introduce a higher order numerical method for solving (1.1) based on this scheme.

4.1. The discontinuous Galerkin scheme in McLean and Mustapha [35]

In this subsection, we obtain the same time discretisation scheme as in McLean and Mustapha [35] by approximating the Riemann-Liouville fractional derivative by using the L1 method. We first write (1.1) as the following equivalent form

$$u_t + {}_0^R D_t^{1-\alpha} Au = 0, \quad \text{for } 0 < t \leq T, \quad \text{with } u(0) = u_0, \quad (4.42)$$

where u_t denotes the time derivative and ${}_0^R D_t^\alpha u(t)$ denotes the Riemann-Liouville fractional derivative.

At $t = t_n$, we approximate the time derivative by using the backward Euler method

$$u_t(t_n) = (u(t_n) - u(t_{n-1}))/\tau + O(\tau), \quad \text{as } \tau \rightarrow 0.$$

To approximate the Riemann-Liouville fractional derivative ${}_0^R D_t^{1-\alpha} Au(t_n)$, we shall use the following Diethelm finite difference method [9], with $u \in C^2[0, T; \mathcal{D}(A)]$

$${}_0^R D_t^{1-\alpha} Au(t_n) = \tau^{\alpha-1} \sum_{j=0}^n w_{n-j} Au(t_j) + O(\tau^{1+\alpha}), \quad \text{as } \tau \rightarrow 0, \quad (4.43)$$

where $w_j, j = 0, 1, 2, \dots, n-1$ are given again by (3.23) and w_n satisfies

$$\Gamma(1+\alpha)w_n = (n-1)^\alpha - n^\alpha + \alpha n^{\alpha-1}. \quad (4.44)$$

With $U^n \approx u(t_n)$, we define the following time discretisation problem for solving (4.42)

$$U^n - U^{n-1} + \tau^\alpha \sum_{j=0}^n w_{n-j} AU^j = 0, \quad n \geq 1, \quad \text{with } U^0 = u_0, \quad (4.45)$$

where $w_j, j = 0, 1, 2, \dots, n-1$ are given by (3.23) and w_n is corrected as

$$\Gamma(1+\alpha)w_n = -2n^\alpha + (n-1)^\alpha + (n+1)^\alpha. \quad (4.46)$$

Now all $w_j, j = 0, 1, 2, \dots, n$ are defined by (3.23). The reason for correcting w_n is that we shall use the discrete Laplace transform for $(w_0, w_1, w_2, \dots, w_n, \dots)$ in the proof of the error estimates.

THEOREM 4.1. ([53]) Let $u(t_n)$ and U^n be the solutions of (4.42) and (4.45), respectively. Let $u_0 \in L_2(\Omega)$. Then we have, with $0 < \alpha < 1$,

$$\|u(t_n) - U^n\| \leq C(\tau^\alpha t_n^{-\alpha} + \tau t_n^{-1}) \|u_0\|. \quad (4.47)$$

We note that the convergence rate of the time discretisation scheme (4.45) is only $O(\tau^\alpha)$, $0 < \alpha < 1$ for smooth and nonsmooth initial data. To derive a time discretisation method for solving (4.42) with the convergence rate $O(\tau)$ for smooth and nonsmooth initial data u_0 , we will approximate ${}_0^R D_t^{1-\alpha} Au(t_n)$ by

$${}_0^R D_t^{1-\alpha} Au(t_n) \approx \tau^{\alpha-1} \sum_{j=1}^n w_{n-j} Au(t_j),$$

where we ignore the term $Au(t_0)$ in (4.43). More precisely, we choose $w_n = 0$ in the summation $\sum_{j=0}^n w_{n-j} Au(t_j)$ in (4.43). It is easy to show that

$${}_0^R D_t^{1-\alpha} Au(t_n) = \tau^{\alpha-1} \sum_{j=1}^n w_{n-j} Au(t_j) + O(\tau), \text{ as } \tau \rightarrow 0. \quad (4.48)$$

To see this, by (4.43), it suffices to show that, for the fixed $t_n = n\tau$,

$$\tau^{\alpha-1} w_n = O(\tau), \text{ as } \tau \rightarrow 0. \quad (4.49)$$

In fact, let t_n be fixed, for example, assume that $t_n = 1, n = 1/\tau$, we have

$$\begin{aligned} \Gamma(1+\alpha)\tau^{\alpha-1} w_n &= \tau^{\alpha-1} (\alpha n^{\alpha-1} + (n-1)^\alpha - n^\alpha) \\ &= \alpha t_n^{\alpha-1} + (n-1)^\alpha \tau^{\alpha-1} - n^\alpha \tau^{\alpha-1} = t_n^{\alpha-1} \left(\alpha + \frac{(n-1)^\alpha}{n^{\alpha-1}} - \frac{n^\alpha}{n^{\alpha-1}} \right) \\ &= t_n^{\alpha-1} \left(\alpha + \frac{(1/\tau - 1)^\alpha}{(1/\tau)^{\alpha-1}} - \frac{1}{\tau} \right) = t_n^{\alpha-1} \left(\alpha + \frac{(1-\tau)^\alpha - 1}{\tau} \right) \\ &= t_n^{\alpha-1} \left(\alpha + \frac{(1-\tau)^\alpha - 1}{\tau} \right) = t_n^{\alpha-1} O(\tau) = O(\tau), \text{ as } \tau \rightarrow 0, \end{aligned}$$

which implies (4.49) and therefore (4.48) follows.

Now we may define the time discretisation scheme for solving (4.42), with $U^0 = u_0$,

$$U^n - U^{n-1} + k^\alpha \sum_{j=1}^n w_{n-j} AU^j = 0, \quad n \geq 1, \quad (4.50)$$

where $w_j, j = 0, 1, 2, \dots, n-1, n \geq 1$ are given by

$$\Gamma(1+\alpha)w_j = \begin{cases} 1, & \text{for } j = 0, \\ -2j^\alpha + (j-1)^\alpha + (j+1)^\alpha, & \text{for } j = 1, 2, \dots, n-1. \end{cases} \quad (4.51)$$

The time stepping method (4.50) was first introduced in McLean and Mustapha [35] for solving (4.42) by using the piecewise discontinuous Galerkin method, see other related works for the discontinuous Galerkin method [8], [37], [38], [40].

THEOREM 4.2. ([35]) Let $u(t_n)$ and U^n be the solutions of (4.42) and (4.50), respectively. Let $u_0 \in L_2(\Omega)$. Then we have, with $0 < \alpha < 1$,

$$\|u(t_n) - U^n\| \leq C\tau t_n^{-1} \|u_0\|. \quad (4.52)$$

4.2. Correction of the discontinuous Galerkin method of McLean and Mustapha [35]

In this subsection, we shall introduce a modified piecewise constant discontinuous Galerkin method for solving (4.42). We show that this method has the convergence rate $O(\tau^{1+\alpha})$ for smooth and nonsmooth initial data u_0 .

Following the idea in Lubich *et al.* [33], we approximate the time derivative $u_t(t_n)$ by using a second order backward difference method

$$u_t(t_n) = \frac{\frac{3}{2}u(t_n) - 2u(t_{n-1}) + \frac{1}{2}u(t_{n-2})}{\tau} + O(\tau^2), \quad \text{as } \tau \rightarrow 0.$$

We define the following finite difference method for solving (4.42), with $U^n \approx u(t_n)$ and $c_0 = 1/2$,

$$\bar{D}U^n + \tau^{\alpha-1} \left(\sum_{j=1}^n w_{n-j} AU^j + c_0 w_{n-1} Au_0 \right) = 0, \quad n \geq 2, \quad (4.53)$$

$$\bar{D}U^n + \tau^{\alpha-1} \left(\sum_{j=1}^n w_{n-j} AU^j + c_0 w_{n-1} Au_0 \right) = 0, \quad n = 1, \quad (4.54)$$

$$U^0 = u_0, \quad U^{-1} = u_0, \quad (4.55)$$

where

$$\bar{D}U^n = \frac{\frac{3}{2}U^n - 2U^{n-1} + \frac{1}{2}U^{n-2}}{\tau}, \quad n \geq 1,$$

and $w_j, j = 0, 1, 2, \dots, n-1$ are defined by (4.51). Here we use a modification term $c_0 w_{n-1} Au_0$ as in Lubich *et al.* [33, (1.18)].

THEOREM 4.3. ([53]) Let $u(t_n)$ and U^n be the solutions of (4.42) and (4.53)-(4.55), respectively. Let $u_0 \in L_2(\Omega)$. Then we have, with $0 < \alpha < 1$,

$$\|u(t_n) - U^n\| \leq C\tau^{1+\alpha} t_n^{1-\alpha} \|u_0\|, \quad n \geq 1. \quad (4.56)$$

4.3. Numerical examples

In this section, we will consider the experimentally determined convergence rates of the different numerical methods discussed in Section 4 for solving (4.42).

Let us consider the following time fractional partial differential equation in the one dimensional case.

$$\begin{aligned} {}_0^C D_t^\alpha u(x, t) - u_{xx} &= 0, \quad 0 < x < 1, \quad 0 < t \leq T, \\ u(0, t) = u(1, t) &= 0, \\ u(x, 0) &= u_0(x). \end{aligned}$$

Let $0 < t_0 < t_1 < \dots < t_N = T$ be the time partition of $[0, T]$ with $T = 1$ and τ the time step size. Let N_h be a positive integer. Let $0 = x_0 < x_1 < x_2 < \dots < x_{N_h} = 1$ be the space partition and h the space step size. The space is discretised by using the standard linear finite element method.

α		$k = 2^{-4}$	$k = 2^{-5}$	$k = 2^{-6}$	$k = 2^{-7}$	$k = 2^{-8}$
0.1	(a)	1.57e-01	1.27e-01	9.84e-02	7.13e-02	4.59e-02
		0.308	0.370	0.464	0.635	
	(b)	1.65e-01	1.33e-01	1.03e-01	7.45e-02	4.80e-02
		0.308	0.370	0.465	0.635	
0.3	(a)	1.64e-01	1.21e-01	8.62e-02	5.76e-02	3.43e-02
		0.434	0.492	0.581	0.747	
	(b)	1.69e-01	1.25e-01	8.88e-02	5.93e-02	3.54e-02
		0.435	0.492	0.582	0.747	
0.8	(a)	1.90e-02	1.06e-02	5.75e-03	2.99e-03	1.42e-03
		0.848	0.877	0.939	1.076	
	(b)	1.77e-02	9.49e-03	5.37e-03	2.80e-03	1.33e-03
		0.843	0.875	0.938	1.075	
0.9	(a)	7.73e-03	3.99e-03	2.04e-03	1.00e-03	4.52e-04
		0.953	0.970	1.023	1.151	
	(b)	6.87e-03	3.58e-03	1.83e-03	9.03e-04	4.07e-04
		0.942	0.965	1.020	1.149	

TABLE 5. Time convergence rates with the different $\alpha \in (0, 1)$ for the numerical method (4.45)

We first consider the scheme (4.45): the convergence rate was proved to be $O(\tau^\alpha)$ for both smooth and nonsmooth data in Theorem 4.1. To observe this convergence rate, we first calculate the reference solution $uref(t)$ at $T = 1$ with $h_{ref} = 2^{-6}$ and $\tau_{ref} = 2^{-10}$. We then use $h = 2^{-6}$ and $\tau = kappa * \tau_{ref}$ with $kappa = [2^2, 2^3, 2^4, 2^5, 2^6]$ to obtain the approximate solution at $u(T)$ with $T = 1$. Let e_τ denote the error of $u(T)$ at $T = 1$ with the time step size τ and the fixed space step size $h = 2^{-6}$. By Theorem

4.1, we have $\|e_\tau\| \leq C\tau^\alpha$. Thus the convergence rate α is determined experimentally by $\alpha \approx \log 2 \left(\frac{\|e_{2\tau}\|}{\|e_\tau\|} \right)$.

Choosing the smooth and nonsmooth initial data (a) $u_0 = \sin(\pi x)$ and (b) $u_0 = \chi_{[0,1/2]}$, we observe, in Table 5, that the experimentally determined convergence rate is indeed almost $O(k^\alpha)$ for the different $\alpha \in (0, 1)$ in both smooth and nonsmooth data cases.

We next consider the numerical method (4.50) proposed by McLean and Mustapha [35] which has the convergence rate $O(\tau)$ for both smooth and nonsmooth data. Using the same notations and the same initial data as in Table 5, we found, in Table 6, that the experimentally determined convergence rate of this method is indeed approximately 1.

α		$k = 2^{-4}$	$k = 2^{-5}$	$k = 2^{-6}$	$k = 2^{-7}$	$k = 2^{-8}$
0.1	(a)	1.20e-04	6.53e-05	3.43e-05	1.71e-05	7.73e-06
		0.876	0.929	1.01	1.14	
	(b)	7.99e-05	4.36e-05	2.29e-05	1.14e-05	5.16e-06
		0.876	0.929	1.004	1.145	
0.3	(a)	6.99e-04	3.57e-04	1.77e-04	8.46e-05	3.69e-05
		0.972	1.01	1.07	1.19	
	(b)	4.65e-04	2.37e-04	1.18e-04	5.61e-05	2.45e-05
		0.972	1.008	1.069	1.199	
0.8	(a)	1.29e-03	6.01e-04	2.83e-04	1.30e-04	5.54e-05
		1.103	1.088	1.120	1.233	
	(b)	8.36e-04	3.89e-04	1.83e-04	8.43e-05	3.58e-05
		1.103	1.088	1.120	1.233	
0.9	(a)	9.66e-04	4.35e-04	2.02e-04	9.22e-05	3.91e-05
		1.151	1.109	1.130	1.238	
	(b)	6.21e-04	2.80e-04	1.30e-04	5.93e-05	2.52e-05
		1.150	1.109	1.130	1.237	

TABLE 6. Time convergence rates with the different $\alpha \in (0, 1)$ for the numerical method (4.50)

Finally we consider the improved numerical method (3.38)-(3.40) which has the convergence rate $O(\tau^{1+\alpha})$ for both smooth and nonsmooth data. Using the same notations and the same initial data as in Tables 5 and 6, we found, in Table 7, that the experimentally determined convergence rate is approximately $\tau^{1+\alpha}$ (actually the experimentally determined convergence rate is better than $1 + \alpha$) as we expected.

α		$k = 2^{-4}$	$k = 2^{-5}$	$k = 2^{-6}$	$k = 2^{-7}$	$k = 2^{-8}$
0.1	(a)	2.08e-04	9.34e-05	4.17e-05	1.82e-05	7.35e-06
		1.159	1.162	1.198	1.307	
	(b)	1.39e-04	6.22e-05	2.78e-05	1.21e-05	4.89e-06
		1.159	1.162	1.198	1.307	
0.3	(a)	2.16e-04	7.77e-05	2.88e-05	1.07e-05	3.78e-06
		1.472	1.431	1.428	1.502	
	(b)	1.43e-04	5.14e-05	1.90e-05	7.07e-06	2.50e-6
		1.473	1.432	1.429	1.502	
0.8	(a)	1.04e-04	2.25e-05	4.95e-06	1.08e-06	2.22e-07
		2.199	2.187	2.203	2.273	
	(b)	6.67e-05	1.45e-05	3.18e-06	6.91e-07	1.43e-07
		2.200	2.188	2.204	2.274	
0.9	(a)	9.87e-05	2.27e-05	5.26e-06	1.23e-06	2.80e-07
		2.122	2.107	2.096	2.134	
	(b)	6.32e-05	1.45e-05	3.68e-06	7.88e-07	1.79e-07
		2.122	2.108	2.096	2.134	

TABLE 7. Time convergence rates with the different $\alpha \in (0, 1)$ for the numerical methods (3.38)-(3.40)

5. Open problems and issues for further work

In this paper, we have reviewed an approach for constructing the higher order time discretisation schemes for solving (1.1). We propose some open problems and issues for further works.

- To use the starting weights to construct higher order time discretisation schemes for solving (1.1), one needs to assume that the solution $u(t, x)$ has the series expansion form (1.10) which implies that the source term f should satisfy some compatibility conditions. We may need to ask: which types of compatibility conditions should f satisfy? For which types of f in (1.1) may we use the starting weights techniques to improve the convergence rates of the numerical methods for solving (1.1)?
- To approximate the Hadamard finite-part integral with piecewise p th order polynomials with $p \geq 3$ to obtain the approximate scheme to the Riemann-Liouville fractional derivative with the convergence order $p+1-\alpha$. Can one use such a scheme to construct the corrected higher order time discretisation method for solving (1.1)?

- To construct the corrected higher order numerical methods for solving (1.1) by using the higher order approximate schemes to the Riemann-Liouville fractional derivatives developed in Li *et al.* [2], [25].
- To construct the corrected higher order time discretisation method for solving

$$u(t) - u(0) + {}_0^R D_t^{-\alpha}(Au(t)) = 0, \quad (5.57)$$

by approximating the Riemann-Liouville integral ${}_0^R D_t^{-\alpha}(Au(t))$ with the Lubich convolution quadratures discussed in Jin *et al.* [23].

References

- [1] Y. Cao, T. Herdman, and Y. Xu, A hybrid collocation method for Volterra integral equations with weakly singular kernels, *SIAM J. Numer. Anal.* **41**, (2003), 364-381.
- [2] J. X. Cao, C. P. Li, and Y. Chen, High-order approximation to Caputo derivatives and Caputo-type advection-diffusion equations (II), *Fract. Calc. Appl. Anal.* **18**, (2015), 735-761.
- [3] W. Cao, F. Zeng, Z. Zhang, and G. E. Karniadakis, Implicit-explicit difference schemes for nonlinear fractional differential equations with nonsmooth solutions, *SIAM J. Sci. Comput.* **38**, (2016), A3070-A3093.
- [4] F. Chen, Q. Xu, and J. S. Hesthaven, A multi-domain spectral method for time-fractional differential equations, *J. Comput. Phys.* **293**, (2015), 157-172.
- [5] S. Chen, J. Shen, and L.-L. Wang, Generalized Jacobi functions and their applications to fractional differential equations, *Math. Comp.* **85**, (2016), 1603-1638.
- [6] X. Chen, F. Zeng, and G.E.Karniadakis, A tunable finite difference method for fractional differential equations with non-smooth solutions, *Comput. Methods Appl. Mech. Engrg.* **318**, (2017), 193-214.
- [7] E. Cuesta, C. Lubich, and C. Palencia, Convolution quadrature time discretization of fractional diffusion-wave equations, *Math. Comp.* **75**, (2006), 673-696.
- [8] W. Deng, J. S. Hesthaven, Local discontinuous Galerkin methods for fractional ordinary differential equations, *BIT* **55**, (2015), 967-985.
- [9] K. Diethelm, An algorithm for the numerical solution of differential equations of fractional order, *Electronic Transactions on Numerical Analysis* **5**, (1997), 1-6.
- [10] K. Diethelm, Generalized compound quadrature formulae for finite part integral, *IMA J. of Numer. Anal.* **17**, (1997), 479- 493.
- [11] K. Diethelm, N. J. Ford, and A. D. Freed, Detailed error analysis for a fractional Adams method, *Numer. Algorithms* **36**, (2004), 31-52.
- [12] K. Diethelm, J. M. Ford, N. J. Ford, and M. Weilbeer, Pitfalls in fast numerical solution of fractional differential equations, *J. Comput. Appl. Math.* **186**, (2006), 482-503.
- [13] S. Esmaili, M. Shamsi, and Y. Luchko, Numerical solution of fractional differential equations with a collocation method based on Müntz polynomials, *Comput. Math. Appl.* **62**, (2011), 918-929.
- [14] N. J. Ford, M. L. Morgado, and M. Rebelo, Nonpolynomial collocation approximation of solutions to fractional differential equations, *Fract. Calc. Appl. Anal.* **16**, (2013), 874-891.

- [15] N. J. Ford, K. Pal, and Y. Yan, An algorithm for the numerical solution of space-fractional partial differential equations, *Computational Methods in Applied Mathematics* **15**, (2015), 497-514.
- [16] N. J. Ford, J. Xiao, and Y. Yan, Stability of a numerical method for a space-time-fractional telegraph equation, *Computational Methods in Applied Mathematics* **12**, (2012), 1-16.
- [17] N. J. Ford, J. Xiao, and Y. Yan, A finite element method for time-fractional partial differential equations, *Fract. Calc. Appl. Anal.* **14**, (2011), 454-474.
- [18] G.-H. Gao, Z.-Z. Sun, and H.-W. Zhang, A new fractional numerical differentiation formula to approximate the Caputo fractional derivative and its applications, *J. Comput. Phys.* **259**, (2014), 33-50.
- [19] B. Jin, R. Lazarov, D. Sheen, and Z. Zhou, Error estimates for approximations of distributed order time fractional diffusion with nonsmooth data, *Fract. Calc. Appl. Anal.* **19**, (2016), 69-93.
- [20] B. Jin, R. Lazarov, and Z. Zhou, Two fully discrete schemes for fractional diffusion and diffusion-wave equations with nonsmooth data, *SIAM J. Sci. Comput.* **38**, (2016), A146-A170.
- [21] B. Jin, R. Lazarov, and Z. Zhou, An analysis of the L1 scheme for the subdiffusion equation with nonsmooth data, *IMA J. of Numer. Anal.* **36**, (2016), 197-221.
- [22] B. Jin, B. Li, and Z. Zhou, An analysis of the Crank-Nicolson method for subdiffusion, *IMA J. of Numer. Anal.*, in press, (2017), arXiv:1607.06948.
- [23] B. Jin, B. Li, and Z. Zhou, Correction of high-order BDF convolution quadrature for fractional evolution equations, *preprint*, (2017), arXiv:1703.08808.
- [24] Z. Li, Z. Liang, and Y. Yan, High-order numerical methods for solving time fractional partial differential equations, *J. Sci. Comput.* **71**, (2017), 785-803.
- [25] H. F. Li, J. X. Cao, and C. P. Li, Higher-order approximation to Caputo derivatives and Caputo-type advection-diffusion equations (III), *Journal of Computational and Applied Mathematics* **299**, (2016), 159-175.
- [26] C. Li, H. Ding, Higher order finite difference method for the reaction and anomalous-diffusion equation, *Appl. Math. Model.* **38**, (2014), 3802-3821.
- [27] C. Li, F. Zeng, and F. Liu, Spectral approximations to the fractional integral and derivative, *Fract. Calc. Appl. Anal.* **15**, (2012), 383-406.
- [28] Y. Lin and C. Xu, Finite difference/spectral approximations for the time-fractional diffusion equation, *J. Comput. Phys.* **225**, (2007), 1533-1552.
- [29] C. Lv, C. Xu, Error analysis of a high order method for time-fractional diffusion equations, *SIAM J. Sci. Comput.* **38**, (2016), A2699-A2724.
- [30] C. Lubich, Discretized fractional calculus, *SIAM J. Math. Anal.* **17**, (1986), 704-719.
- [31] C. Lubich, Convolution quadrature and discretized operational calculus. I., *Numer. Math.* **52**, (1988), 129-145.
- [32] C. Lubich, Convolution quadrature revisited, *BIT* **44**, (2004), 503-514.
- [33] C. Lubich, I. H. Sloan, and V. Thomée, Nonsmooth data error estimates for approximations of an evolution equation with a positive-type memory term, *Math. Comp.* **65**, (1996), 1-17.
- [34] Z. P. Mao, J. Shen, Efficient spectral-Galerkin methods for fractional partial differential equations with variable coefficients, *J. Comput. Phys.* **307**, (2016), 243-261.
- [35] W. McLean, K. Mustapha, Time-stepping error bounds for fractional diffusion problems with non-smooth initial data, *J. Comput. Phys.* **293**, (2015), 201-217.
- [36] W. McLea, K. Mustapha, A second-order accurate numerical method for a fractional wave equation, *Numer. Math.* **105**, (2007), 481-510.

- [37] K. Mustapha, Time-stepping discontinuous Galerkin methods for fractional diffusion problems, *Numer. Math.* **130**, (2015), 497-516.
- [38] K. Mustapha, B. Abdallah, and K. M. Furati, A discontinuous Petrov-Galerkin method for time-fractional diffusion equations, *SIAM J. Numer. Anal.* **52**, (2014), 2512-2529.
- [39] K. Mustapha, W. Mclean, Superconvergence of a discontinuous Galerkin method for fractional diffusion and wave equations, *SIAM J. Numer. Anal.* **51**, (2013), 491-515.
- [40] K. Mustapha, D. Schötzau, Well-posedness of hp-version discontinuous Galerkin methods for fractional diffusion wave equations, *IMA J. Numer. Anal.* **34**, (2014), 1426-1446.
- [41] K. B. Oldham, J. Spanier, *The Fractional Calculus*, Academic, New York, (1974).
- [42] I. Podlubny *Fractional Differential Equations*, Academic press, San Diego, (1999).
- [43] J. Quintana-Murillo, S. B. Yuste, A finite difference method with non-uniform timesteps for fractional diffusion and diffusion-wave equations, *The European Physical Journal Special Topics* **222**, (2013), 1987-1998.
- [44] K. Sakamoto, M. Yamamoto, Initial value/boundary value problems for fractional diffusion-wave equations and applications to some inverse problems, *J. Math. Anal. Appl.* **382**, (2011), 426-447.
- [45] M. Stynes, Too much regularity may force too much uniqueness, *Fractional Calculus and Applied Analysis*, **19**, (2016), 1554-1562.
- [46] M. Stynes, E. O'riordan and J. L. Gracia, Error analysis of a finite difference method on graded meshes for a time-fractional diffusion equation, *SIAM J. Numer. Anal.* **55**, (2017), 1057-1079.
- [47] Z.-Z. Sun, X. Wu, A fully discrete scheme for a diffusion wave system, *Appl. Numer. Math.* **56**, (2011), 193-209.
- [48] V. Thomée, *Galerkin Finite Element Methods for Parabolic Problems*, Springer-Verlag, Berlin, (2007).
- [49] M. Weillbeer, *Efficient Numerical Methods for Fractional Differential Equations and their Analytical Background*, Dissertation, Technischen Universität Braunschweig, (2005).
- [50] Y. Xing, Y. Yan, A higher order numerical method for time fractional partial differential equations with nonsmooth data, *preprint*, (2017).
- [51] Y. Yan, K. Pal, and N. J. Ford, Higher order numerical methods for solving fractional differential equations, *BIT Numer Math* **54**, (2014), 555-584.
- [52] Y. Yan, M. Khan, and N. J. Ford, An analysis of the modified L1 scheme for the time-fractional partial differential equations with nonsmooth data, *preprint*, (2016).
- [53] Y. Yang, Y. Yan, and N. J. Ford, The modified piecewise constant discontinuous Galerkin methods for fractional diffusion problems with nonsmooth data, *Computational Methods in Applied Mathematics*, in press, (2017).
- [54] Y. Yu, P. Perdikaris, and G.E. Karniadakis, Fractional modeling of viscoelasticity in 3d cerebral arteries and aneurysms, *J. Comput. Phys.* **323**, (2016), 219-242.
- [55] S. B. Yuste, Weighted average finite difference methods for fractional diffusion equations, *J. Comput. Phys.* **216**, (2006), 264-274.
- [56] S. B. Yuste, J. Quintana-Murillo, Fast, accurate and robust adaptive finite difference methods for fractional diffusion equations, *Numer. Algor.* **71**, (2016), 207-228.
- [57] M. Zayernouri, M. Ainsworth, and G. E. Karniadakis, A unified Petrov-Galerkin spectral method for fractional pdes, *Comput. Methods Appl. Mech. Engrg.* **283**, (2015), 1545-1569.

- [58] M. Zayernouri, A. Matzavinos, Fractional Adams-Bashforth/Moulton methods: An application to the fractional Keller-Segel chemotaxis system, *J. Comput. Phys.* **317**, (2016), 1-14.
- [59] M. Zayernouri, G. E. Karniadakis, Fractional spectral collocation method, *SIAM J. Sci. Comput.* **36**, (2014), A40-A62.
- [60] F. Zeng, Second-order stable finite difference schemes for the time-fractional diffusion-wave equation, *J. Sci. Comput.* **65**, (2015), 411-430.
- [61] F. Zeng, C. Li, F. Liu, and I. Turner, The use of finite difference/element approaches for solving the time-fractional subdiffusion equation, *SIAM J. Sci. Comput.* **35**, (2013), A2976-A3000.
- [62] F. Zeng, C. Li, F. Liu, and I. Turner, Numerical algorithms for time-fractional subdiffusion equation with second-order accuracy, *SIAM J. Sci. Comput.* **37**, (2015), A55-A78.
- [63] F. Zeng, Z. Zhang, and G. E. Karniadakis, A generalized spectral collocation method with tunable accuracy for variable-order fractional differential equations, *SIAM J. Sci. Comput.* **37**, (2015), A2710-A2732.
- [64] F. Zeng, Z. Zhang, and G. E. Karniadakis, Fast difference schemes for solving high-dimensional time-fractional subdiffusion equations, *J. Comput. Phys.* **307**, (2016), 15-33.
- [65] F. Zeng, Z. Zhang, and G. E. Karniadakis, Second-order numerical methods for multi-term fractional differential equations: Smooth and non-smooth solutions, arXiv: 1701.00996, (2017).
- [66] Y. -N. Zhang, Z. -Z. Sun, and H. -L. Liao, Finite difference methods for the time fractional diffusion equation on non-uniform meshes, *J. Comput. Phys.* **265**, (2014), 195-210.
- [67] Z. Zhang, F. Zeng, and G. E. Karniadakis, Optimal error estimates for spectral Petrov-Galerkin and collocation methods for initial value problems for fractional differential equations, *SIAM J. Numer. Anal.* **53**, (2015), 2074-2096.
- [68] M. Zheng, F. Liu, V. Anh and I. Turner, A high order spectral method for the multi-term time-fractional diffusion equations, *Appl. Math. Modelling* **40**, 2016, pp. 4970-4985.

¹ *Department of Mathematics*
University of Chester
Parkgate Road
Chester, CH1 4BJ, UK
e-mail: njford@chester.ac.uk

Received: June, 2017

² *Department of Mathematics*
University of Chester
Parkgate Road
Chester, CH1 4BJ, UK
e-mail: y.yan@chester.ac.uk