

Article

A New Diffusive Representation for Fractional Derivatives, Part II: Convergence Analysis of the Numerical Scheme [†]

Kai Diethelm 

Faculty of Applied Natural Sciences and Humanities, University of Applied Sciences Würzburg-Schweinfurt, Ignaz-Schön-Str. 11, 97421 Schweinfurt, Germany; kai.diethelm@fhws.de

[†] Dedicated to my colleague and friend Neville J. Ford on the occasion of his 65th birthday.

Abstract: Recently, we have proposed a new diffusive representation for fractional derivatives and, based on this representation, suggested an algorithm for their numerical computation. From the construction of the algorithm, it is immediately evident that the method is fast and memory-efficient. Moreover, the method's design is such that good convergence properties may be expected. In this paper, we commence a systematic investigation of these convergence properties.

Keywords: fractional derivative; Caputo derivative; diffusive representation; numerical method; convergence

MSC: primary 65D25; secondary 26A33; 65L05; 65R20



Citation: Diethelm, K. A New Diffusive Representation for Fractional Derivatives, Part II: Convergence Analysis of the Numerical Scheme. *Mathematics* **2022**, *10*, 1245. <https://doi.org/10.3390/math10081245>

Academic Editor: Nikolai Kosmatov

Received: 2 March 2022

Accepted: 7 April 2022

Published: 10 April 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Fractional differential operators have proven to be very useful tools for the numerical modeling of various phenomena in science, engineering, and economics, cf., e.g., [1] and the literature cited therein. Since differential equations with fractional derivatives are very difficult, and often impossible, to solve exactly in closed form, there is a great need for efficient numerical methods.

A large number of different approaches has been proposed but many of these are not sufficiently fast or require too much memory whereas others are more efficient in these respects but their convergence behaviour is not well understood (see Section 2 for more details on this topic). In this paper, we pick up a recently developed approach [2] that is very efficient with respect to both runtime and memory requirements and that has explicitly been designed in a way that allows a thorough analysis. While [2] merely indicated that such an error analysis is possible, in the present paper we actually describe this analysis and its outcomes. Our main result, Theorem 1, contains an error estimate that clearly shows the influence of all the parameters on which the algorithm depends. Therefore, in combination with the results of [2], we now have a complete picture of the method's properties, both from the theoretical and from the practical perspective.

We explicitly stress that this paper is exclusively devoted to (theoretical) error analysis. Numerical aspects of the algorithm under consideration have already been discussed in [2]; that paper in particular also contains some numerical examples. Therefore, it is not necessary to repeat these here.

2. A Diffusive-Representation-Based Numerical Scheme for Computing Fractional Derivatives

The topic of this paper is the analysis of a recently developed numerical method for the approximate calculation of fractional derivatives [2]. Specifically, we assume a function $y \in C^{[\alpha]}[a, a + T]$ to be given with some $T > 0$ and some $a \in \mathbb{R}$, and the task is to compute the Caputo derivative $D_a^\alpha y(t_j)$ of order $\alpha \in \mathbb{R}_+ \setminus \mathbb{N}$ of the function y ([3], Chapter 3) at some

points $t_0 < t_1 < t_2 < \dots < t_N$, with $t_0 = a$ and $t_N = a + T$ being the two end points of the interval of interest. The algorithm is based on the diffusive representation

$$D_a^\alpha y(t) = \int_{-\infty}^{\infty} \phi_D(w, t) dw \tag{1}$$

where the function $\phi_D(w, t)$ in the integrand on the right-hand side of Equation (1) is the solution to the initial value problem

$$\frac{\partial \phi_D}{\partial t}(w, t) = -e^w \phi_D(w, t) + (-1)^{[\alpha]} \frac{\sin \alpha \pi}{\pi} e^{w q_D} y^{([\alpha])}(t), \quad \phi_D(w, a) = 0, \tag{2}$$

cf. ([2], Theorem 1). In Equation (2), $q_D = \alpha - [\alpha] + 1 \in (0, 1)$; thus, if $\alpha \in (0, 1)$, i.e., in the case that is prevalent in most known technical applications of fractional derivatives, we have $q_D = \alpha$.

Based on Equations (1) and (2), the algorithm developed in [2] comprises two elements:

1. approximate the integral in Equation (1) by means of a suitable quadrature formula, and
2. compute the integrand function that needs to be evaluated by this quadrature formula by means of a numerical solver for the first-order initial value problem (2).

More precisely, there are two variants of the approach. Both variants use the quadrature formula

$$\int_{-\infty}^{\infty} \phi_D(w, t) dw = \int_0^{\infty} e^{-u} \hat{\phi}(w, t) dw \approx \sum_{k=1}^K a_{k,K}^{GLa} \hat{\phi}(x_{k,K}^{GLa}, t) \tag{3}$$

with

$$\hat{\phi}(w, t) = e^w \left(\frac{1}{q_D} \phi_D(-w/q_D, t) + \frac{1}{1 - q_D} \phi_D(w/(1 - q_D), t) \right) \tag{4}$$

where $a_{k,K}^{GLa}$ and $x_{k,K}^{GLa}$ for $k = 1, 2, \dots, K$ denote the weights and nodes, respectively, of the K -point Gauss–Laguerre quadrature formula, i.e., the Gaussian quadrature formula for the weight function e^{-w} . The difference between the two methods is that, in order to compute the function values $\hat{\phi}(x_{k,K}^{GLa}, t)$ for some given $t = t_n$ via the representation (4), one of them uses the backward Euler formula for the solution ϕ_D of the initial value problem (2) whereas the other one employs the trapezoidal method.

The basic motivation for the development of this technique was to use it as a building block for a numerical solver for fractional differential equations. If this is done, then, like other numerical schemes based on diffusive representations or similar techniques (cf., e.g., [4–10]) this approach has some significant advantages over traditional schemes such as the fractional Adams method [11,12] or fractional linear multistep methods [13,14]:

1. The computational complexity is $O(N)$ where N is the number of time steps over which the solution to the fractional differential equation is sought, whereas traditional methods usually have a cost of $O(N^2)$ when implemented in a straightforward manner [11,13] or of $O(N \log N)$ or $O(N \log^2 N)$ [15–18] if more sophisticated implementations are used.
2. Due to the completely different way in which the inherent memory of the fractional differential operators is dealt with, the active memory requirements of the method are only $O(1)$ and not $O(N)$ for the traditional methods or at best $O(\log N)$ for their modified versions [15,19].
3. Whereas some (but not all) traditional schemes require the use of a uniform mesh, this approach gives the user complete freedom to use any discretization whatsoever of the interval on which the fractional differential equation is to be solved.

3. Convergence Properties of the Numerical Method

The paper [2] in which the abovementioned algorithm was developed contains some numerical results. It also provides a qualitative convergence analysis and a heuristic

argumentation as to why the method should have satisfactory convergence properties, and it describes how to avoid certain potential pitfalls in the implementation of the algorithm in finite precision arithmetic. However, it does not provide a thorough convergence analysis in the quantitative sense, so we shall now address this matter here. To this end, we assume that the function values of $D_a^\alpha y$ are to be approximately computed at the points t_n , $n = 0, 1, 2, \dots, N$, with $a = t_0 < t_1 < \dots < t_N = a + T$. We then proceed as follows.

First of all, we note that our approximation formula takes the final form

$$D_a^\alpha y(t_n) \approx \sum_{k=1}^K a_{k,K}^{\text{GLa}} \hat{\phi}_{k,K,n} \tag{5}$$

where $\hat{\phi}_{k,K,n}$ denotes the approximation for $\hat{\phi}(x_{k,K}^{\text{GLa}}, t_n)$ obtained by the ODE solver in question using the grid $\{t_n : n = 0, 1, \dots, N\}$. It thus follows that the error of this approximation is

$$\begin{aligned} R_{\alpha,K,n}(y) &= D_a^\alpha y(t_n) - \sum_{k=1}^K a_{k,K}^{\text{GLa}} \hat{\phi}_{k,K,n} \\ &= \int_0^\infty e^{-u} \hat{\phi}(w, t) dw - \sum_{k=1}^K a_{k,K}^{\text{GLa}} \hat{\phi}_{k,K,n} \\ &= R_{\alpha,K}^Q(y; t_n) + R_{\alpha,K,n}^{\text{ODE}}(y) \end{aligned} \tag{6}$$

where

$$R_{\alpha,K}^Q(y; t_n) = \int_0^\infty e^{-u} \hat{\phi}(w, t_n) dw - \sum_{k=1}^K a_{k,K}^{\text{GLa}} \hat{\phi}(x_{k,K}^{\text{GLa}}, t_n) \tag{7}$$

is the error induced by the numerical quadrature, and

$$R_{\alpha,K,n}^{\text{ODE}}(y) = \sum_{k=1}^K a_{k,K}^{\text{GLa}} \left(\hat{\phi}(x_{k,K}^{\text{GLa}}, t_n) - \hat{\phi}_{k,K,n} \right) \tag{8}$$

is the error induced by the ODE solver.

Remark 1. In the decomposition (6) of the total error, it is clear from Equation (7) that the component $R_{\alpha,K}^Q(y; t_n)$ —as indicated by the notation—depends only on the function y whose fractional derivative we want to compute, the order α of this derivative, the number K of nodes of the Gauss–Laguerre quadrature formula that we use, and the point t_n . Thus, there is an indirect dependency on the exact solution of the initial value problem (2) at the point $t = t_n$ but not on the numerical solution of this problem. Hence, for the analysis of $R_{\alpha,K}^Q(y; t_n)$, it does not matter by which numerical method and with which grid we solve this initial value problem.

The component $R_{\alpha,K,n}^{\text{ODE}}(y)$, on the other hand, does not only depend on the ODE solver and its grid but also on the number K of quadrature nodes and on the location of these nodes because these quantities appear as parameters in the differential equation and thus have an influence on the ODE solver’s error.

Our main goal now is to analyze and estimate the expressions $R_{\alpha,K}^Q(y; t_n)$ and $R_{\alpha,K,n}^{\text{ODE}}(y)$ under reasonable assumptions on the given data. The first result in this context reads as follows.

Lemma 1. Let $y \in C^m[a, a + T]$ for some $m \in \mathbb{N}$ and some $T > 0$. Then, for all $p > 0$, we have uniformly for all $n \in \{0, 1, 2, \dots, N\}$ that

$$R_{\alpha,K}^Q(y; t_n) = O(K^{-p})$$

as $K \rightarrow \infty$.

Therefore, we can conclude that the quadrature error converges to zero with a faster-than-algebraic rate.

For the proof of Lemma 1, we require an auxiliary result on the asymptotic behaviour of $\phi_D(w, t)$ for $w \rightarrow \pm\infty$ that refines the findings of ([2], Theorem 1(e)):

Lemma 2. *Let $y \in C^m[a, a + T]$ for some $m \in \mathbb{N}$ and some $T > 0$. Then, for all $r \in \mathbb{N}_0$, we have*

$$\frac{\partial^r \phi_D}{\partial w^r}(w, t) = O(e^{w(q_D-1)}) \cdot |\sin \alpha \pi| \quad \text{for } w \rightarrow \infty$$

and

$$\frac{\partial^r \phi_D}{\partial w^r}(w, t) = O(e^{wq_D}) \cdot |\sin \alpha \pi| \quad \text{for } w \rightarrow -\infty.$$

The implied constants in the O -terms can be bounded uniformly for all $t \in [a, a + T]$ and all $\alpha \in (m - 1, m)$.

Proof. The claims for $r = 0$ have already been shown in ([2], Theorem 1(e)). We thus only discuss the cases $r = 1, 2, \dots$ here. To this end, we first recall from ([2], Theorem 1) that

$$\phi_D(w, t) = (-1)^{|\alpha|} \frac{\sin \alpha \pi}{\pi} e^{wq_D} \int_a^t y^{(m)}(\tau) \exp(-(t - \tau)e^w) d\tau \tag{9}$$

holds for any fixed $t \in [a, a + T]$. This implies, in particular, that $\phi_D(w, a) = 0$ for all w , and so the claims clearly hold in the case $t = a$.

For $t > a$, an application of the Leibniz rule to (9) shows that

$$\begin{aligned} \frac{\partial^r \phi_D}{\partial w^r}(w, t) &= (-1)^{|\alpha|} \frac{\sin \alpha \pi}{\pi} \\ &\times \sum_{s=0}^r \binom{r}{s} q_D^{r-s} e^{wq_D} \int_a^t y^{(m)}(\tau) \frac{\partial^s}{\partial w^s} \exp(-(t - \tau)e^w) d\tau \end{aligned}$$

which, upon setting

$$c(\alpha) := \frac{|\sin \alpha \pi|}{\pi} \|y^{(m)}\|_{L^\infty[a, a+T]} < \infty, \tag{10}$$

implies for all $w \in \mathbb{R}$ that

$$\begin{aligned} \left| \frac{\partial^r \phi_D}{\partial w^r}(w, t) \right| &\leq c(\alpha) \sum_{s=0}^r \binom{r}{s} q_D^{r-s} e^{wq_D} \int_a^t \frac{\partial^s}{\partial w^s} \exp(-(t - \tau)e^w) d\tau \\ &= c(\alpha) \sum_{s=0}^r \binom{r}{s} q_D^{r-s} e^{wq_D} \frac{\partial^s}{\partial w^s} \int_a^t \exp(-(t - \tau)e^w) d\tau \\ &= c(\alpha) \sum_{s=0}^r \binom{r}{s} q_D^{r-s} e^{wq_D} \frac{\partial^s}{\partial w^s} [e^{-w}(1 - \exp(-(t - a)e^w))]. \end{aligned} \tag{11}$$

For the last factor in each summand, we can again use the Leibniz rule and find

$$\begin{aligned} &\frac{\partial^s}{\partial w^s} [e^{-w}(1 - \exp(-(t - a)e^w))] \\ &= \sum_{\sigma=1}^s \binom{s}{\sigma} (-1)^{s-\sigma} e^{-w} \frac{\partial^\sigma}{\partial w^\sigma} \exp(-(t - a)e^w) \\ &\quad + (-1)^s e^{-w}(1 - \exp(-(t - a)e^w)) \\ &= \sum_{\sigma=1}^s \binom{s}{\sigma} (-1)^{s-\sigma} J_\sigma(w) + (-1)^s J_0(w) \end{aligned} \tag{12}$$

with

$$J_0(w) = e^{-w}(1 - \exp(-(t - a)e^w))$$

and

$$J_\sigma(w) = e^{-w} \frac{\partial^\sigma}{\partial w^\sigma} \exp(-(t-a)e^w) \quad (\sigma = 1, 2, 3, \dots).$$

It is then immediately clear that

$$|J_0(w)| \leq e^{-w} \text{ for all } w,$$

so

$$\lim_{w \rightarrow \infty} e^w |J_0(w)| \leq 1. \tag{13}$$

Furthermore, by de l’Hospital’s rule,

$$\lim_{w \rightarrow -\infty} J_0(w) = t - a.$$

Together with the fact that $J_0(w)$ is monotonically decreasing in w , this shows that

$$|J_0(w)| \leq T \text{ for all } w. \tag{14}$$

Moreover, a straightforward mathematical induction yields for $\sigma = 1, 2, 3, \dots$ that there exist polynomials $\pi_{\sigma-1}$ of degree $\sigma - 1$ such that

$$J_\sigma(w) = \exp(-(t-a)e^w) \pi_{\sigma-1}(e^w).$$

Denoting the polynomials $\pi_{\sigma-1}$ by $\pi_{\sigma-1}(z) = \sum_{\rho=0}^{\sigma-1} \psi_\rho z^\rho$, it can also be seen via the induction process that the coefficients ψ_ρ depend on $t - a$ in a continuous way and so they may be bounded in absolute value independently of t . Hence, we conclude

$$J_\sigma(w) = \exp(-(t-a)e^w) \sum_{\rho=0}^{\sigma-1} \psi_\rho e^{\rho w} = \sum_{\rho=0}^{\sigma-1} \psi_\rho \exp(\rho w - (t-a)e^w).$$

Thus,

$$\lim_{w \rightarrow -\infty} |J_\sigma(w)| \leq \lim_{w \rightarrow -\infty} \sum_{\rho=0}^{\sigma-1} |\psi_\rho| \exp(\rho w - (t-a)e^w) = |\psi_0| \tag{15}$$

and

$$\lim_{w \rightarrow \infty} e^w |J_\sigma(w)| \leq \lim_{w \rightarrow \infty} \sum_{\rho=0}^{\sigma-1} |\psi_\rho| \exp((\rho + 1)w - (t-a)e^w) = 0 \tag{16}$$

because (since we only need to deal with those values of t for which $t > a$) the argument of each of the exponential functions is less than $-2w$ for sufficiently large w . Plugging (13) and (16) into (12), we then derive

$$\frac{\partial^s}{\partial w^s} [e^{-w}(1 - \exp(-(t-a)e^w))] = O(e^{-w}) \text{ for } w \rightarrow \infty,$$

and inserting (14) and (15) into (12), we find

$$\frac{\partial^s}{\partial w^s} [e^{-w}(1 - \exp(-(t-a)e^w))] = O(1) \text{ for } w \rightarrow -\infty.$$

The two claims of the Lemma then follow upon combining these two relations, both of which evidently hold uniformly for all admissible values of t and α , with Equation (11) and the observation that $c(\alpha) = \tilde{c} \cdot |\sin \alpha \pi|$ with some constant $\tilde{c} \in \mathbb{R}$. \square

Remark 2. Clearly, the bounds of Lemma 2 imply the slightly simpler relationships

$$\frac{\partial^r \phi_D}{\partial w^r}(w, t) = O(e^{w(q_D-1)}) \text{ for } w \rightarrow \infty$$

and

$$\frac{\partial^r \phi_D}{\partial w^r}(w, t) = O(e^{wq_D}) \quad \text{for } w \rightarrow -\infty$$

uniformly for all $t \in [a, a + T]$ and all $\alpha \in (m - 1, m)$ that, however, provide weaker bounds than Lemma 2 itself if α is close to an integer.

Proof of Lemma 1. We begin by noting that $\phi_D(\cdot, t) \in C^\infty(\mathbb{R})$ (this is an immediate consequence of the representation (9); see also ([2], Theorem 1(d))). Therefore, it follows from Equation (4) that $\hat{\phi}(\cdot, t) \in C^\infty[0, \infty)$. Moreover, defining the function κ_r for $r \in \mathbb{N}_0$ on $[0, \infty)$ by

$$\kappa_r(w) = e^{-w} w^{r/2} \frac{\partial^r \hat{\phi}}{\partial w^r}(w, t),$$

we can see by definition of $\hat{\phi}$, cf. Equation (4), that $\kappa_r \in C[0, \infty)$. In addition, by the Leibniz formula we conclude

$$\begin{aligned} \kappa_r(w) &= e^{-w} w^{r/2} \sum_{s=0}^r \binom{r}{s} e^{w} \frac{\partial^s}{\partial w^s} \left[\frac{1}{q_D} \phi_D(-w/q_D, t) + \frac{1}{1 - q_D} \phi_D(w/(1 - q_D), t) \right] \\ &= w^{r/2} \sum_{s=0}^r \binom{r}{s} \left[\frac{(-1)^s}{q_D^{s+1}} \frac{\partial^s}{\partial u^s} \phi_D(u, t) \Big|_{u=-w/q_D} + \frac{1}{(1 - q_D)^{s+1}} \frac{\partial^s}{\partial u^s} \phi_D(u, t) \Big|_{u=w/(1-q_D)} \right]. \end{aligned}$$

An application of Lemma 2 then tells us that, for arbitrary $r \in \mathbb{N}$ and $w \rightarrow \infty$,

$$\begin{aligned} |\kappa_r(w)| &\leq w^{r/2} c |\sin \alpha \pi| \sum_{s=0}^r \binom{r}{s} \left[\frac{1}{q_D^{s+1}} e^{-w} + \frac{1}{(1 - q_D)^{s+1}} e^{-w} \right] \\ &\leq c |\sin \alpha \pi| w^{r/2} e^{-w} \sum_{s=0}^r \binom{r}{s} \left[\frac{1}{q_D^{s+1}} + \frac{1}{(1 - q_D)^{s+1}} \right] \\ &= c |\sin \alpha \pi| \left[\frac{(q_D + 1)^r}{q_D^{r+1}} + \frac{(2 - q_D)^r}{(1 - q_D)^{r+1}} \right] w^{r/2} e^{-w} \end{aligned} \tag{17}$$

with some constant c that is independent of t and α . Therefore, $\kappa_r \in L_1[0, \infty)$ for all $r \in \mathbb{N}$, so we may invoke ([20], Theorem 1) with any such r , and the claim of Lemma 1 follows. \square

Remark 3. An inspection of the proof of Lemma 1 reveals that the implied constant in the O -term of the final estimate cannot be bounded uniformly for all $\alpha \in (m - 1, m)$. This is due to the presence of the positive powers of q_D and $(1 - q_D)$ in the denominators of the right-hand side of Equation (17) and the fact that q_D tends to zero for $\alpha \rightarrow m - 1$ and that $1 - q_D$ tends to zero for $\alpha \rightarrow m$. The factor $|\sin \alpha \pi|$ on the right-hand side of Equation (17) (that originates from the bounds of Lemma 2 in its present form) can only compensate a first power of these expressions but not a higher one. Whether or not a uniform bound can be obtained with a more sophisticated estimation technique remains an open question. The uniformity of the bound with respect to n is no problem though.

For the other component of the total error, we may also derive a bound. We assume here that the backward Euler formula is chosen as the numerical solver for the initial value problems.

Lemma 3. Assume that the grid $\{t_n : n = 0, 1, \dots, N\}$ is uniform, i.e., that $t_n = a + nh$ with $h = T/N$. If $y \in C^{[\alpha]+1}[a, a + T]$ and the differential equations are solved by means of the backward Euler method, then

$$|R_{\alpha, K, n}^{\text{ODE}}(y)| \leq C(K) t_n h \leq C(K) Th$$

with

$$C(K) = \frac{|\sin \alpha \pi|}{2\pi} e^{x_{K,K}^{GLa} q_D / (1 - q_D)} \times \left(\left\| y^{([\alpha]+1)}(t) \right\|_{L^\infty[a, a+T]} + 2e^{x_{K,K}^{GLa} / (1 - q_D)} \left\| y^{([\alpha])}(t) \right\|_{L^\infty[a, a+T]} \right).$$

Proof. According to Equations (3) and (4), we need to solve the differential equation in Equation (2) for $w \in W_- \cup W_+$ where

$$W_- = \{-x_{k,K}^{GLa} / q_D : k = 1, 2, \dots, K\} \tag{18}$$

and

$$W_+ = \{x_{k,K}^{GLa} / (1 - q_D) : k = 1, 2, \dots, K\}. \tag{19}$$

Let us introduce the notation

$$f(t, z) = -e^w z + (-1)^{[\alpha]} \frac{\sin \alpha \pi}{\pi} e^{w q_D} y^{([\alpha])}(t),$$

so that the differential equation in question takes the form

$$\frac{\partial \phi_D}{\partial t}(w, t) = f(t, \phi_D(w, t)).$$

For any $w \in W_+ \cup W_-$, we have $-e^w \leq -\exp(-x_{K,K}^{GLa} / q_D)$ and so, for all $z_1, z_2 \in \mathbb{R}$, we can see that

$$(f(t, z_1) - f(t, z_2))(z_1 - z_2) = -e^w (z_1 - z_2)^2 \leq -\exp(-x_{K,K}^{GLa} / q_D) \cdot (z_1 - z_2)^2. \tag{20}$$

Therefore, in the terminology of ([21], Definition 8.58), the initial value problem satisfies an upper Lipschitz condition with constant $-\exp(-x_{K,K}^{GLa} / q_D) < 0$, and hence the initial value problem is dissipative.

This property allows us to estimate the local truncation error of the backward Euler method in the following essentially standard way: By our smoothness assumption on the function y , a Taylor expansion shows that the local truncation error in the n -th step has the form

$$\delta_n = h(f(t_n, \phi_D(w, t_n)) - f(t_n, \phi_{D,n}(w))) + \frac{1}{2} h^2 \frac{d}{dt} f(t, \phi_D(w, t))|_{t=\xi}$$

where $\phi_{D,n}(w)$ is the approximation for $\phi_D(w, t_n)$ (so that $\delta_n = \phi_D(w, t_n) - \phi_{D,n}(w)$) and $\xi \in [t_{n-1}, t_n]$. Hence,

$$\begin{aligned} \left(\delta_n - \frac{1}{2} h^2 \frac{d}{dt} f(t, \phi_D(w, t))|_{t=\xi} \right) \delta_n &= h(f(t_n, \phi_D(w, t_n)) - f(t_n, \phi_{D,n}(w))) \delta_n \\ &\leq -h \exp(-x_{K,K}^{GLa} / q_D) \delta_n^2 \end{aligned}$$

in view of Equation (20). A rearrangement of terms then yields

$$\delta_n^2 (1 + h \exp(-x_{K,K}^{GLa} / q_D)) \leq \frac{1}{2} h^2 \frac{d}{dt} f(t, \phi_D(w, t))|_{t=\xi} \cdot \delta_n.$$

Evidently, the left-hand side of this inequality is nonnegative, and so the right-hand side must be nonnegative too. Hence, we may replace both sides of the inequality by their respective absolute values without changing anything. Having done that, we may divide both sides by $|\delta_n|$ to obtain

$$|\delta_n| (1 + h \exp(-x_{K,K}^{GLa} / q_D)) \leq \frac{1}{2} h^2 \left| \frac{d}{dt} f(t, \phi_D(w, t))|_{t=\xi} \right|. \tag{21}$$

In view of the chain rule, the concrete structure of the function f and the differential equation under consideration, we have

$$\begin{aligned} \left| \frac{d}{dt} f(t, \phi_D(w, t)) \right| &\leq \left| \frac{\partial f}{\partial t}(t, \phi_D(w, t)) \right| + \left| \frac{\partial f}{\partial z}(t, \phi_D(w, t)) \cdot \frac{\partial \phi_D}{\partial t}(w, t) \right| \\ &\leq \frac{|\sin \alpha \pi|}{\pi} e^{wq_D} \left| y^{([\alpha]+1)}(t) \right| + e^w |f(t, \phi_D(w, t))|. \end{aligned}$$

Using the explicit representation of ϕ_D given in Equation (9), it can be shown that the last expression in this inequality satisfies

$$|f(t, \phi_D(w, t))| \leq 2e^{wq_D} \frac{|\sin \alpha \pi|}{\pi} \left| y^{([\alpha]}(t) \right|.$$

Thus, we can continue the estimation of Equation (21) as

$$\begin{aligned} &|\delta_n| (1 + h \exp(-x_{K,K}^{GLa} / q_D)) \\ &\leq \frac{1}{2} h^2 \frac{|\sin \alpha \pi|}{\pi} e^{wq_D} \left(\left\| y^{([\alpha]+1)}(t) \right\|_{L_\infty[a, a+T]} + 2e^w \left\| y^{([\alpha]}(t) \right\|_{L_\infty[a, a+T]} \right) \\ &\leq \frac{1}{2} h^2 \frac{|\sin \alpha \pi|}{\pi} e^{x_{K,K}^{GLa} q_D / (1-q_D)} \\ &\quad \times \left(\left\| y^{([\alpha]+1)}(t) \right\|_{L_\infty[a, a+T]} + 2e^{x_{K,K}^{GLa} / (1-q_D)} \left\| y^{([\alpha]}(t) \right\|_{L_\infty[a, a+T]} \right). \end{aligned}$$

Since the second factor on the left-hand side is greater than 1, we deduce

$$\begin{aligned} |\delta_n| &\leq \frac{1}{2} h^2 \frac{|\sin \alpha \pi|}{\pi} e^{x_{K,K}^{GLa} q_D / (1-q_D)} \\ &\quad \times \left(\left\| y^{([\alpha]+1)}(t) \right\|_{L_\infty[a, a+T]} + 2e^{x_{K,K}^{GLa} / (1-q_D)} \left\| y^{([\alpha]}(t) \right\|_{L_\infty[a, a+T]} \right). \end{aligned} \tag{22}$$

for all admissible values of the parameters.

This estimate, together with the dissipativity of the differential equation, allows us to derive our claim from ([21], Theorem 8.68). □

Thus, combining Lemmas 1 and 3 with the fact that

$$x_{K,K}^{GLa} < 4K + 2$$

(cf. [22], Equation (6.32.2)), we obtain the following overall error estimate:

Theorem 1. *If $y \in C^{[\alpha]+1}[a, a + T]$, if the grid $\{t_n : n = 0, 1, \dots, N\}$ is uniform and if the differential equations are solved using the backward Euler method, then, for all $p > 0$,*

$$\max_{n=1,2,\dots,N} |R_{\alpha,K,n}(y)| = O(K^{-p}) + O\left(h \cdot \exp\left(\frac{1 + q_D}{1 - q_D} (4K + 2)\right)\right)$$

as $K \rightarrow \infty$ and/or $N \rightarrow \infty$, where $h = T/N$.

Remark 4. *If the backward Euler method for solving the differential Equation (2) for $w \in W_- \cup W_+$ is replaced by the trapezoidal method, a similar analysis may be performed. It is then possible to replace the factor h in the error estimate by h^2 , reflecting the higher order of the trapezoidal scheme, but this requires even stronger differentiability assumptions on the function y .*

4. Comments and Further Remarks

The results indicated above give rise to a number of observations that raise new questions and indicate some directions for additional research work. We intend to address these issues in the future.

4.1. The Stiffness of the Differential Equation for the Integrand

A short look at the differential equation in the initial value problem (2) reveals that it is an inhomogeneous linear differential equation with constant coefficients (note that it is a differential equation with respect to the variable t , so the value w that arises in the coefficient of ϕ_D on the right-hand side is a constant for any given differential equation of this form). The Lipschitz constant of the function on its right-hand side is e^w with some $w \in W_- \cup W_+$ (cf. Equations (18) and (19)). Since $x_{k,K} > 0$ for all k and $q_D \in (0, 1)$, it is clear that $W_- \subset (-\infty, 0)$, and so $e^w \in (0, 1)$ for $w \in W_-$. Therefore, the differential equations associated to these values of w have small Lipschitz constants and thus do not exhibit any stiffness which, in turn, means that they do not pose significant challenges for the numerical solvers.

For the case $w \in W_+$, however, the situation is completely different. Here, we encounter Lipschitz constants of up to $L_{K,+} = \exp(x_{K,K}^{\text{GLa}} / (1 - q_D))$. Bearing in mind the well known result ([22], Equation (6.32.8)) that

$$x_{K,K}^{\text{GLa}} = 4K(1 + o(1)) \text{ as } K \rightarrow \infty,$$

it is clear that this Lipschitz constant may be an extremely large number if q_D is close to 1 (i.e., if $\alpha = A - \varepsilon$ with some $A \in \mathbb{N}$ and a positive number ε close to 0) or if K is large. Therefore, these differential equations may be extremely stiff and hence difficult to handle numerically. This, in fact, also explains why one should only use A-stable solvers for the differential equation.

4.2. The Error Bounds for the ODE Solver

The error estimates of Lemma 3 and hence also of Theorem 1 indicate that the error of the ODE solver depends on the maximum of the Lipschitz constants of the differential equations under consideration and hence, in view of the observation from Section 4.1, on the number K of quadrature nodes or, more precisely, on the location of the largest node $x_{K,K}^{\text{GLa}}$ of the quadrature formula in use. Although this is only an upper bound that in fact may drastically overestimate the true error ([23], p. 7), it nevertheless clarifies the importance of keeping the value $x_{K,K}^{\text{GLa}}$ as small as possible.

4.3. Choice of the Quadrature Formula

From the results of Mastroianni and Monegato [24] one can conclude that it might be useful to modify the quadrature formula in use in our algorithm. To be precise, they suggest truncating the summation in Equation (5) prematurely, i.e., letting the summation index k run only from 1 to some number K^* with $K^* < K$. This concept has two obvious advantages, namely, it reduces the computational cost and it improves the approximation quality of the ODE solver (due to the fact that the largest nodes of the quadrature formula, i.e., the nodes of which the associated differential equations have the right-hand sides with the largest Lipschitz constants and are thus most difficult to solve accurately—cf. Section 4.2—are left out). Intuitively, one is likely led into the belief that one has to pay for this improvement on the ODE solver component that this approach generates in terms of a loss of accuracy on the numerical integration component. It has been shown in [24], however, that, in reality, the convergence rate of the quadrature formula actually becomes better, at least when the number K^* is chosen in a proper way.

4.4. The Smoothness of the Function y

The assumptions of Lemma 3 are relatively strong in the sense that they require some degree of smoothness of the function y whose fractional derivative we want to compute. If an algorithm of the type under consideration is to be used in the context of numerically solving fractional initial value problems of the form

$$D_a^\alpha y(t) = f(t, y(t)), \quad y(a) = y_0 \tag{23}$$

with some $\alpha \in (0, 1)$, it is well known ([3], Theorem 6.27) that these smoothness properties can only be expected to be present in rare and exceptional situations. If the function is less smooth, then the convergence order of Lemma 3 and Theorem 1 cannot be achieved. This expectation was confirmed by the numerical experiments in [2] for this particular algorithm and in [25] for a different algorithm based on the same fundamental principle (i.e., on diffusive representations). A precise estimation of the convergence behaviour under less restrictive (and hence more realistic, in the context of solutions to fractional differential equations) conditions remains a topic for future research.

4.5. Diffusive Representations for Fractional Integrals

As a possible remedy for the problem discussed in Section 4.4, one may rewrite the given initial value problem (23) in the equivalent form ([3], Lemma 6.2)

$$y(t) = y_0 + J_a^\alpha [f(\cdot, y(\cdot))](t)$$

where J_a^α is the Riemann–Liouville integral operator of order α with starting point a , find a diffusive representation analog to (1) for this integral operator and proceed in a corresponding manner. It is conceivable that the functions needing to be approximated in this approach possess more favorable smoothness properties, thus possibly leading to better convergence properties of the ODE solver.

4.6. The Discretization of the Interval $[a, a + T]$

In the error analysis of Theorem 1, we had assumed a uniform discretization of the interval $[a, a + T]$ on which the fractional derivative of y was to be computed. However, as indicated above, the basic construction principle of our approach actually admits a completely arbitrary discretization $a = t_0 < t_1 < t_2 < \dots < t_N = a + T$. In such a case, the methods applied in the proof of Theorem 1 can still be applied and the same result may be obtained, wherein the parameter h must then be interpreted as $h = \max_{n=1,2,\dots,N} (t_n - t_{n-1})$.

If one chooses a graded mesh of the form $t_n = a + (n/N)^\mu T$ with a mesh grading coefficient $\mu > 0$ (most commonly actually with $\mu \geq 1$), then we conjecture that it may be possible to obtain an error bound of the form given in Theorem 1 under weaker smoothness conditions on the function y .

5. Conclusions

In [2], a new algorithm for the approximate calculation of fractional derivatives of Caputo’s type has been developed. The algorithm is based on a novel diffusive representation of such operators. By construction, with respect to run time and memory requirements the method described in [2] behaves in the same way as other methods based on diffusive representations, i.e., in order to compute the fractional derivative at N points, the computational complexity is $O(N)$ and the memory requirements are $O(1)$. This is optimal and, in particular, significantly smaller than the corresponding requirements of traditional methods which are not based on diffusive representations.

The main difference between the approach of [2] and other diffusive-representation-based schemes is that, also by construction, the numerical integration phase of the algorithm has been specifically designed in a way that allows to invoke traditional well understood, fast and highly accurate quadrature methods. There is no need to use ad hoc quadrature methods with many numerical parameters whose behaviour is hardly understood.

Rather, the only parameters of the algorithm that have an influence on the error are the discretization grid and the number of quadrature nodes. The simplicity of this structure has enabled us in this paper to provide a comprehensive error analysis which clearly exhibits the influence of each parameter on the accuracy of the final result and which is much more complete than the error analysis that has been conducted for many other methods based on diffusive representations.

Combining the new error analysis developed here with the numerical results shown in [2], we can conclude that the algorithm described in Equation (5)—which is based on the new diffusive representation, comprising Equations (1) and (2)—provides a promising tool for the approximate calculation of Caputo-type fractional derivatives and that, as such, it is also a good candidate to be a fundamental building block in numerical schemes for solving fractional differential equations.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Diethelm, K.; Kiryakova, V.; Luchko, Y.; Machado, J.A.T.; Tarasov, V.E. Trends, directions for further research, and some open problems of fractional calculus. *Nonlinear Dyn.* **2022**, *107*, 3245–3270. [[CrossRef](#)]
2. Diethelm, K. A new diffusive representation for fractional derivatives, Part I: Construction, implementation and numerical examples. In *Fractional Differential Equations: Modeling, Discretization, and Numerical Solvers*; Springer: Heidelberg, Germany, 2022.
3. Diethelm, K. *The Analysis of Fractional Differential Equations*; Springer: Berlin, Germany, 2010.
4. Baffet, D. A Gauss-Jacobi kernel compression scheme for fractional differential equations. *J. Sci. Comput.* **2019**, *79*, 227–248. [[CrossRef](#)]
5. Hinze, M.; Schmidt, A.; Leine, R.I. Numerical solution of fractional-order ordinary differential equations using the reformulated infinite state representation. *Fract. Calc. Appl. Anal.* **2019**, *22*, 1321–1350. [[CrossRef](#)]
6. Li, J.-R. A fast time stepping method for evaluating fractional integrals. *SIAM J. Sci. Comput.* **2010**, *31*, 4696–4714. [[CrossRef](#)]
7. McLean, W. Exponential sum approximations for $t^{-\beta}$. In *Contemporary Computational Mathematics*; Dick, J., Kuo, F.Y., Woźniakowski, H., Eds.; Springer: Cham, Switzerland, 2018; pp. 911–930.
8. Singh, S.J.; Chatterjee, A. Galerkin projections and finite elements for fractional order derivatives. *Nonlinear Dyn.* **2006**, *45*, 183–206. [[CrossRef](#)]
9. Yuan, L.; Agrawal, O.P. A numerical scheme for dynamic systems containing fractional derivatives. *J. Vib. Acoust.* **2002**, *124*, 321–324. [[CrossRef](#)]
10. Zhang, W.; Capilnasiu, A.; Sommer, G.; Holzapfel, G.A.; Nordsletten, D. An efficient and accurate method for modeling nonlinear fractional viscoelastic biomaterials. *Comput. Methods Appl. Mech. Eng.* **2020**, *362*, 112834. [[CrossRef](#)] [[PubMed](#)]
11. Diethelm, K.; Ford, N.J.; Freed, A.D. A predictor-corrector approach for the numerical solution of fractional differential equations. *Nonlinear Dyn.* **2002**, *29*, 3–22. [[CrossRef](#)]
12. Diethelm, K.; Ford, N.J.; Freed, A.D. Detailed error analysis for a fractional Adams method. *Numer. Algorithms* **2004**, *36*, 31–52. [[CrossRef](#)]
13. Lubich, C. Fractional linear multistep methods for Abel-Volterra integral equations of the second kind. *Math. Comput.* **1985**, *45*, 463–469. [[CrossRef](#)]
14. Lubich, C. Discretized fractional calculus. *SIAM J. Math. Anal.* **1986**, *17*, 704–719. [[CrossRef](#)]
15. Ford, N.J.; Simpson, A.C. The numerical solution of fractional differential equations: Speed versus accuracy. *Numer. Algorithms* **2001**, *26*, 333–346. [[CrossRef](#)]
16. Garrappa, R. Numerical solution of fractional differential equations: A survey and a software tutorial. *Mathematics* **2018**, *6*, 16. [[CrossRef](#)]
17. Hairer, E.; Lubich, C.; Schlichte, M. Fast numerical solution of nonlinear Volterra convolution equations. *SIAM J. Sci. Stat. Comput.* **1985**, *6*, 532–541. [[CrossRef](#)]
18. Hairer, E.; Lubich, C.; Schlichte, M. Fast numerical solution of weakly singular Volterra integral equations. *J. Comput. Appl. Math.* **1988**, *23*, 87–98. [[CrossRef](#)]
19. Diethelm, K.; Freed, A.D. An efficient algorithm for the evaluation of convolution integrals. *Comput. Math. Appl.* **2006**, *51*, 51–72. [[CrossRef](#)]

20. Mastroianni, G.; Monegato, G. Error Estimates for Gauss-Laguerre and Gauss-Hermite Quadrature Formulas. In *Approximation and Computation*; Zahar, R.V.M., Ed.; Int. Ser. Numer. Math. 119; Birkhäuser: Boston, MA, USA, 1994; pp. 421–434.
21. Plato, R. *Concise Numerical Mathematics*; American Mathematical Society: Providence, RI, USA, 2003.
22. Szegő, G. *Orthogonal Polynomials*, 4th ed.; American Mathematical Society: Providence, RI, USA, 1975.
23. Iserles, A. *A First Course in the Numerical Analysis of Differential Equations*; Cambridge University Press: Cambridge, UK, 1996.
24. Mastroianni, G.; Monegato, G. Truncated quadrature rules over $(0, \infty)$ and Nyström-type methods. *SIAM J. Numer. Anal.* **2003**, *41*, 1870–1892. [[CrossRef](#)]
25. Diethelm, K. Fast solution methods for fractional differential equations in the modeling of viscoelastic materials. In Proceedings of the 9th International Conference on Systems and Control (ICSC 2021), Caen, France, 24–26 November 2021.