fractal and fractional

MDPI

Article

High-Order Schemes for Nonlinear Fractional Differential Equations

Omar Alsayyed [1,†], Fadi Awawdeh [1,*,†], Safwan Al-Shara' [2,†] and Edris Rawashdeh [3,†]

[1] Department of Mathematics, Faculty of Science, The Hashemite University, P.O. Box 330127, Zarqa 13133, Jordan
[2] Department of Mathematics, Faculty of Sciences, Al al-Bayt University, P.O. Box 130095, Mafraq 25113, Jordan
[3] Department of Mathematics, Yarmouk University, Irbid 21163, Jordan
* Correspondence: awawdeh@hu.edu.jo
† These authors contributed equally to this work.

Abstract: We propose high-order schemes for nonlinear fractional initial value problems. We split the fractional integral into a history term and a local term. We take advantage of the sum of exponentials (SOE) scheme in order to approximate the history term. We also use a low-order quadrature scheme to approximate the fractional integral appearing in the local term and then apply a spectral deferred correction (SDC) method for the approximation of the local term. The resulting one-step time-stepping methods have high orders of convergence, which make adaptive implementation and accuracy control relatively simple. We prove the convergence and stability of the proposed schemes. Finally, we provide numerical examples to demonstrate the high-order convergence and adaptive implementation.

Keywords: fractional differential equations; numerical algorithms; time-stepping schemes; high-order methods

Citation: Alsayyed, O.; Awawdeh, F.; Al-Shara', S.; Rawashdeh, E. High-Order Schemes for Nonlinear Fractional Differential Equations. *Fractal Fract.* **2022**, *6*, 748. https://doi.org/10.3390/fractalfract6120748

Academic Editors: Carlo Cattani and Paul Eloe

Received: 2 November 2022
Accepted: 16 December 2022
Published: 19 December 2022

1. Introduction

Fractional differential equations (FDEs) have gained mathematical significance in recent decades due to the growing interest in dynamics of anomalous diffusion processes in amorphous materials. They have governed significant models in many fields of applied sciences and engineering, such as viscoelastic materials, image processing, option pricing models, control theory, etc. Surveys or collections of applications can be found in [1–5].

This paper is concerned with approximating the solution $y(t)$ to the fractional initial-value problem

$$D_t^\alpha y(t) = f(t, y(t)), \qquad 0 \le t \le T,$$
$$y(0) = y_0,$$

where $y_0, y(t) \in \mathbb{R}^d$, $f : [0, T] \times \mathbb{R}^d \to \mathbb{R}^d$, and $D_t^\alpha y$ denotes the Caputo fractional derivative of order α ($0 < \alpha < 1$). Throughout this paper, we assume without loss of generality that $d = 1$.

A number of numerical methods have been proposed for solving FDEs [6–11]. Two common approaches have been considered by many authors. The first approach is based on a direct discretization of the fractional derivative operators in the considered FDEs [6,7,12]. The second approach to solving FDEs is based on discretizing the integral in the equivalent forms [13–16]. There are also several numerical methods for FDEs, for example see [17–20].

Low-order approaches are the most frequently employed methods for FDEs, and it has been proven to be difficult to construct high-order and adaptive schemes [13,15,21,22]. In this paper, we present high-order computational schemes for solving nonlinear FDEs. The proposed schemes deal with integral forms of the FDEs and then split the fractional integral into a history term and a local term. We employ a sum of exponentials scheme

(SOE) for the approximation of the history term and low-order quadrature approximations of the fractional integral. We further modify a spectral deferred correction (SDC) scheme to increase the order of the local approximation. In contrast to the methods inspired by [13], here we use an accurate sum of exponentials method as a kernel compression scheme and a completely different deferred correction scheme as a high-order adaptive method for FDEs.

The remainder of the paper is organized as follows. In Section 2, we provide an overview of the proposed methods. Section 3 discusses the sum of exponentials method. Section 4 introduces high-order time-stepping techniques for solving FDEs and discuses the relevant error analysis. Some numerical results are presented in Section 5 before the conclusion is given in Section 6.

2. Overview

We discuss a numerical method for the fractional initial value problem

$$\begin{cases} D_t^\alpha y(t) = f(t, y(t)), & 0 \le t \le T, \\ y(0) = y_0, \end{cases} \tag{1}$$

where y_0 may be an arbitrary real number, and where $0 < \alpha < 1$. In (1), D_t^α denotes the Caputo differential operator, defined by

$$D_t^\alpha u(t) = J^{n-\alpha} D^n u(t),$$

where $n := \lceil \alpha \rceil$ is the smallest integer $\ge \alpha$. Here, D^n is the usual differential operator of (integer) order n, and for $\mu > 0$, J^μ is the integral operator of order μ in the sense of Riemann–Liouville, defined by

$$J^\mu u(t) = \frac{1}{\Gamma(\alpha)} \int_0^t (t-s)^{\mu-1} u(s) ds.$$

We assume the function f to be such that a unique solution to (1) exists on some interval $[0, T]$. The initial-value problem (1) is known to be equivalent to the Volterra integral equation

$$y(t) = y(0) + \frac{1}{\Gamma(\alpha)} \int_0^t (t-s)^{\alpha-1} f(s, y(s)) ds. \tag{2}$$

Consider the fractional integral

$$I^\alpha f(t) = \int_0^t (t-s)^{\alpha-1} f(s) ds.$$

Fix $t \ge 0$ and $h > 0$, and let $\gamma \in (0, h]$. It is clear that

$$I^\alpha f(t+\gamma) = \int_0^t (t-s+\gamma)^{\alpha-1} f(s) ds + \int_0^\gamma (\gamma - s)^{\alpha-1} f(t+s) ds. \tag{3}$$

Owing to (2) and (3), we obtain

$$y(t+\gamma) = H(t, \gamma) + L(t, \gamma), \tag{4}$$

where

$$H(t, \gamma) = y(0) + \frac{1}{\Gamma(\alpha)} \int_0^t (t-s+\gamma)^{\alpha-1} f(s, y(s)) ds \tag{5}$$

and

$$L(t,\gamma) = \frac{1}{\Gamma(\alpha)} \int_0^\gamma (\gamma - s)^{\alpha-1} f(t+s, y(t+s)) ds. \tag{6}$$

We call H and L the history and local parts, respectively.

We can observe that Equation (4) can be written as a Volterra integral equation (VIE),

$$Y(\gamma) = \frac{1}{\Gamma(\alpha)} \int_0^\gamma (\gamma - s)^{\alpha-1} F(s, Y(s)) ds + H(\gamma), \tag{7}$$

where $Y(\gamma) = y(t+\gamma)$, $F(s,y) = f(t+s,y)$ and $H(\gamma) = H(t,\gamma)$.

3. Approximation of the History Term

In this section, we describe a kernel compression scheme for the approximation of the history term. This will reduce the cost of evaluating the history term.

It is proved in [6] that for any specified error $\epsilon > 0$, we are able to find positive real numbers s_i^α and w_i^α, $i = 1, \ldots, N_\epsilon$, such that

$$\left| \frac{1}{t^{1-\alpha}} - \sum_{i=1}^{N_\epsilon} w_i^\alpha e^{-s_i^\alpha t} \right| < \epsilon, \qquad \forall t \in [h, T],$$

where

$$N_\epsilon = \mathcal{O}\left(\log \frac{1}{\epsilon} \left(\log\log \frac{1}{\epsilon} + \log \frac{T}{h} \right) + \log \frac{1}{h} \left(\log\log \frac{1}{\epsilon} + \log \frac{1}{h} \right) \right).$$

Roughly, for fixed ϵ, the number of exponentials N_ϵ needed is of order $N_\epsilon = \mathcal{O}(\log N_T)$ if $T \gg 1$ or $\mathcal{O}\left(\log^2 N_T\right)$ if $T \approx 1$, where $N_T = \frac{T}{h}$ is the number of time-steps.

Approximating the kernel $(t - s + \gamma)^{\alpha-1}$ in $H(t,\gamma)$ by the sum of exponentials (SOE), we can write the history term (5) as

$$\begin{aligned} H(t,\gamma) &\approx y(0) + \tfrac{1}{\Gamma(\alpha)} \int_0^t \left(\sum_{i=1}^{N_\epsilon} w_i^\alpha e^{-s_i^\alpha (t-s+\gamma)} \right) f(s, y(s)) ds \\ &= y_0 + \sum_{i=1}^{N_\epsilon} \sigma_i^{\alpha,\gamma} U_i^\alpha(t), \end{aligned} \tag{8}$$

where

$$\lambda_i = -s_i^\alpha, \;\; \sigma_i^{\alpha,\gamma} = w_i^\alpha e^{\lambda_i \gamma} / \Gamma(\alpha)$$

and

$$U_i^\alpha(t) = \int_0^t e^{\lambda_i (t-s)} f(s, y(s)) ds.$$

Observe that each $U_i^\alpha(t)$ is the solution to the IVP

$$\begin{aligned} \frac{d}{dt} U_i^\alpha(t) &= \lambda_i U_i^\alpha(t) + f, \\ U_i^\alpha(0) &= 0. \end{aligned}$$

To simplify the notation, we consider $U_1^\alpha(t), \ldots, U_{N_\epsilon}^\alpha(t)$ to be components of a vector $\Phi = \Phi(t)$; similarly, $\Lambda = diag(\lambda_1, \lambda_2, \ldots, \lambda_{N_\epsilon})$, $\mathbf{0}_{N_\epsilon} = (0, \ldots, 0) \in \mathbb{R}^{N_\epsilon}$, and $\mathbf{1}_{N_\epsilon} = (1, \ldots, 1) \in \mathbb{R}^{N_\epsilon}$. Thus we recover

$$\Phi' = \Phi\Lambda + f\mathbf{1}_{N_\epsilon}, \qquad \Phi(0) = \mathbf{0}_{N_\epsilon}. \tag{9}$$

A stable Runge-Kutta (RK) methods may be used for approximating (9).

4. Approximation of the History Term

Let N be a positive integer and T be the given time. We divide the interval $[0, T]$ into

$$0 = t_0 < t_1 < \ldots < t_N = T,$$

with step sizes $\tau_i = t_i - t_{i-1}$, $1 \leq i \leq N$. Let y_k be the approximate solution of $y(t_k)$, $k = 0, 1, \ldots, N$.

At $t = t_n$, we consider (7),

$$Y(\gamma) = \frac{1}{\Gamma(\alpha)} \int_0^{\gamma} (\gamma - s)^{\alpha-1} F(s, Y(s)) ds + H(\gamma), \tag{10}$$

in $(0, \tau_n]$, where $H(\gamma)$ is given. Divide the interval $[0, \tau_n]$ into $0 = \gamma_0 \leq \gamma_1 < \ldots < \gamma_p \leq \tau_n$, and let V be the numerical approximation to Y. Suppose that we know the approximate solutions V_k at γ_k, for each $k = 0, \ldots, j-1$, and let $\mathcal{P}_j = (\gamma_1, \ldots, \gamma_j)$. We compute V_j by solving

$$V_j = \sum_{k=0}^{j} \omega_{jk} F(\gamma_k, V_k) + H_j, \tag{11}$$

for V_j, where $H_j = H(\gamma_j)$, and $\omega_{jk} = \omega_{jk}(\mathcal{P}_j)$ are the weights characterizing the scheme.

4.1. Quadrature Rules on Non-Uniform Meshes

We define

$$t_{n,j} = t_n + \gamma_j, \quad Y_{n,j} = Y_n(\gamma_j) = y_{n,j} = y(t_n + \gamma_j), \quad H_n(\gamma_j) = H(t_n, \gamma_j).$$

To obtain

$$y(t_{n,j}) = H(t_n, \gamma_j) + L(t_n, \gamma_j),$$

we need to approximate the fractional integral

$$L(t_n, \gamma_j) = \frac{1}{\Gamma(\alpha)} \int_0^{\gamma_j} (\gamma_j - s)^{\alpha-1} f(t_n + s, y(t_n + s)) ds.$$

To do so, we use the following approach

$$L(t_n, \gamma_j) = \frac{1}{\Gamma(\alpha)} \sum_{k=0}^{j-1} \int_{\gamma_k}^{\gamma_{k+1}} (\gamma_j - s)_k^{\alpha-1} \tilde{f}_k (t_n + s, y(t_n + s)) ds,$$

where $\tilde{f}_k (t_n + s, y(t_n + s))$, $k = 0, 1, \ldots, j-1$ is an approximation of $f(t_n + s, y(t_n + s))$, where $s \in [\gamma_k, \gamma_{k+1})$. Notice that different choices of $\tilde{f}_k$ lead to different schemes.

(i) By choosing $\tilde{f}_k$ as

$$\tilde{f}_k (t_n + s, y(t_n + s)) = f (t_n + \gamma_k, y(t_n + \gamma_k)) = f (t_{n,k}, y(t_{n,k})),$$

the fractional rectangle method is derived as

$$L(t_n, \gamma_j) = \sum_{k=0}^{j-1} \omega_{jk} f(t_{n,k}, y(t_{n,k})),$$

where

$$\omega_{jk} = \frac{1}{\Gamma(\alpha)} \int_{\gamma_k}^{\gamma_{k+1}} (\gamma_j - s)_k^{\alpha-1} ds$$
$$= \frac{(\gamma_j - \gamma_k)^{\alpha} - (\gamma_j - \gamma_{k+1})^{\alpha}}{\Gamma(1+\alpha)}, \qquad k = 0, 1, \ldots, j-1.$$

(ii) If $\tilde{f}_k$ is selected as

$$\tilde{f}_k\,(t_n + s, y(t_n + s)) = \frac{t_n + s - \gamma_{k+1}}{\gamma_k - \gamma_{k+1}} f(t_{n,k}, y(t_{n,k})) + \frac{t_n + s - \gamma_k}{\gamma_{k+1} - \gamma_k} f(t_{n,k+1}, y(t_{n,k+1})),$$

then the fractional trapezoid method is given by

$$L(t_n, \gamma_j) = \sum_{k=0}^{j} \tilde{\omega}_{jk} f(t_{n,k}, y(t_{n,k}))$$

in which

$$\tilde{\omega}_{jk} = \frac{1}{\Gamma(2+\alpha)} \begin{cases} \frac{1}{\gamma_1} A_0, & \text{if } k = 0, \\ \frac{1}{\gamma_{k+1} - \gamma_k} A_k + \frac{1}{\gamma_{k-1} - \gamma_k} B_k, & \text{if } k = 1, 2, \ldots, j-1, \\ (\gamma_j - \gamma_{j-1})^{\alpha}, & \text{if } k = j, \end{cases}$$

and

$$\begin{cases} A_0 = (\gamma_j - \gamma_1)^{\alpha+1} - \gamma_j^{\alpha+1} + (\alpha+1)\gamma_1 \gamma_j^{\alpha}, \\ A_k = (\gamma_j - \gamma_{k+1})^{\alpha+1} - (\gamma_j - \gamma_k)^{\alpha+1} + (\alpha+1)(\gamma_{k+1} - \gamma_k)(\gamma_j - \gamma_k)^{\alpha}, \\ B_k = (\gamma_j - \gamma_k)^{\alpha+1} - (\gamma_j - \gamma_{k-1})^{\alpha+1} + (\alpha+1)(\gamma_k - \gamma_{k-1})(\gamma_j - \gamma_k)^{\alpha}. \end{cases}$$

It is proved in [8] that for $g \in C^1[0, T]$, we have that

$$\left| \frac{1}{\Gamma(\alpha)} \int_0^{\gamma_j} (\gamma_j - s)^{\alpha-1} g(s) ds - \sum_{k=0}^{j-1} \omega_{jk} g(\gamma_k) \right| \leq \frac{\|g'\|_{\infty}}{\Gamma(1+\alpha)} T^{\alpha} \max_k \tau_k$$

and for $g \in C^2[0, T]$,

$$\left| \frac{1}{\Gamma(\alpha)} \int_0^{\gamma_j} (\gamma_j - s)^{\alpha-1} g(s) ds - \sum_{k=0}^{j} \tilde{\omega}_{jk} g(\gamma_k) \right| \leq \frac{\|g''\|_{\infty}}{2\Gamma(1+\alpha)} T^{\alpha} \left(\max_k \tau_k \right)^2.$$

4.2. The Spectral Deferred Correction Framework

Let τ_n denote the desired time-step size. Then, (4) can be written as

$$y(t+\gamma) = H(t, \gamma) + \frac{1}{\Gamma(\alpha)} \int_0^{\gamma} (\gamma - s)^{\alpha-1} f(t+s, y(t+s)) ds, \tag{12}$$

where $H(t, \gamma)$ is given. Let $\mathcal{P} = \{\gamma_k\}_{k=1}^{p}$ be a set of distinct points in the closed interval $[0, \tau_n]$, where $0 = \gamma_0 \leq \gamma_1 < \ldots < \gamma_p \leq \tau_n$.

Replacing the integral in (12) by a quadrature from Section 4.1 yields the collocation approximation

$$y(t+\gamma_j) = H(t,\gamma_j) + \sum_{k=0}^{j} \omega_{jk} f(t+\gamma_k, y(t+\gamma_k)).$$

This can be written as

$$Y(\gamma_j) = H(\gamma_j) + \sum_{k=0}^{j} \omega_{jk} F(\gamma_k, Y(\gamma_k)), \tag{13}$$

where $Y(\gamma) = y(t+\gamma)$, $F(\gamma, y) = f(t+\gamma, y)$ and $H(\gamma)$ substitutes for $H(t,\gamma)$. At $t = t_n$, the values $Y_{n,j}$ are defined explicitly by the formula

$$Y_{n,j} = H_n(\gamma_j) + \sum_{k=0}^{j-1} \omega_{jk} f(t_{n,k}, Y_{n,k})$$

or implicitly through the nonlinear system

$$Y_{n,j} = H_n(\gamma_j) + \sum_{k=0}^{j} \tilde{\omega}_{jk} f(t_{n,k}, Y_{n,k}).$$

To obtain higher-order schemes, we can apply a spectral deferred correction (SDC) scheme [23–25].

While the time is stepping from t_n to $t_{n+1} = t_n + \tau_n$, an SDC method subdivides the interval $[t_n, t_{n+1}]$ into p substeps

$$t_n \le t_{n,1} < t_{n,2} < \ldots < t_{n,p} \le t_{n+1}.$$

At each substep $t_{n,j}$, the method computes a provisional solution using a low-order method with initial condition y_n. We will refer to the provisional solution as

$$Y_{n,j}^{[1]} \approx y(t_{n,j}), \qquad j = 1, 2, \ldots, p.$$

To improve the accuracy of the provisional solution, we perform a process of iterated corrections of the form

$$Y_{n,j}^{[k+1]} \approx Y_{n,j}^{[k]} + E_{n,j}^{[k]}, \tag{14}$$

where

$$E_{n,j}^{[k]} \approx y(t_{n,j}) - Y_{n,j}^{[k]}.$$

Consider again the integral formulation of the solution,

$$Y(\gamma) = H(\gamma) + \int_0^{\gamma} \frac{1}{\Gamma(\alpha)} (\gamma - s)^{\alpha-1} F(s, Y(s)) ds. \tag{15}$$

Let $\tilde{Y}(\gamma)$ be an approximate solution to (15). Then, plugging $Y(\gamma) = E(\gamma) + \tilde{Y}(\gamma)$ into (15) yields the integral equation

$$E(\gamma) = -\tilde{Y}(\gamma) + H(\gamma) + \int_0^{\gamma} \frac{1}{\Gamma(\alpha)} (\gamma - s)^{\alpha-1} F(s, \tilde{Y}(s) + E(s)) ds.$$

Consequently, the error function can be written as

$$E(\gamma) = \underbrace{\int_0^{\gamma} \frac{1}{\Gamma(\alpha)}(\gamma - s)^{\alpha-1} G(s, E(s))ds}_{\text{Error Term}} + \underbrace{\left[H(\gamma) + \int_0^{\gamma} \frac{1}{\Gamma(\alpha)}(\gamma - s)^{\alpha-1} F(s, \tilde{Y}(s))ds\right] - \tilde{Y}(\gamma)}_{\text{Residual Term } R(\gamma, H(\gamma))}, \tag{16}$$

where

$$G(s, E(s)) = F(s, \tilde{Y}(s) + E(s)) - F(s, \tilde{Y}(s)).$$

To compute the error

$$E_{n,j}^{[k]} \approx y(t_{n,j}) - Y_{n,j}^{[k]},$$

we replace $\tilde{Y}(s)$ with the Lagrange interpolating polynomial $L^{[k]}(s)$,

$$L^{[k]}(t) = \sum_{i=1}^{p} Y_{n,i}^{[k]} l_i(t), \qquad \text{where} \quad l_i(t) = \prod_{\substack{m=1 \\ m \neq k}}^{p} \frac{t - t_{n,m}}{t_{n,i} - t_{n,m}}.$$

To approximate the error, we first use a quadrature formula with nodes at $\{\gamma_j\}_{j=1}^{p}$ to approximate the integral appearing in the residual term $R(\gamma, H(\gamma))$, and then we use a fractional implicit or explicit Euler approximation to approximate the remaining integral term in (16). This process leads to the implicit ($\eta = 1$) or the explicit ($\eta = 0$) time-stepping method

$$E_{n,j+1}^{[k]} = E_{n,j}^{[k]} + \frac{\left(h_{n,j}\right)^{\alpha}}{\Gamma(1+\alpha)} G_{n,j+\eta}^{[k]} + R_{n,j}^{[k]}, \tag{17}$$

where

$$h_{n,j} = t_{n,j+1} - t_{n,j},$$
$$G_{n,j+\eta}^{[k]} = f(t_{n,j+\eta}, Y_{n,j+\eta}^{[k]} + E_{n,j+\eta}^{[k]}) - f(t_{n,j+\eta}, Y_{n,j+\eta}^{[k]}),$$
$$R_{n,j}^{[k]} = Y_{n,j}^{[k]} - Y_{n,j+1}^{[k]} + H_{n,j+1} - H_{n,j} + \sum_{i=1}^{p} \omega_{ji} f(t_{n,i}, Y_{n,i}^{[k]}),$$

and

$$H_{n,j} = H(t_n, \gamma_j).$$

During the iteration process, we always use the initial conditions,

$$t_{n,0} = t_n, \quad Y_{n,0}^{[k]} = y_n, \quad E_{n,0}^{[k]} = 0.$$

From now on, we will refer to the implicit method as IM-SDC and the explicit method as EX-SDC.

Practically, we start by using the fractional Euler's method of order α to compute the provisional solution and then repeat the iteration (17) followed by (14) a total of m times to obtain solutions with an increasingly high order of accuracy

$$\left\{Y_{n,j}^{[1]}\right\}_{j=1}^{p}, \left\{Y_{n,j}^{[2]}\right\}_{j=1}^{p}, \ldots, \left\{Y_{n,j}^{[m+1]}\right\}_{j=1}^{p}.$$

The scheme can be rewritten under a more compact form

$$Y_{n,j+1}^{[k+1]} = Y_{n,j}^{[k+1]} + \frac{\left(h_{n,j}\right)^{\alpha}}{\Gamma(1+\alpha)}\left(f_{n,j+\eta}^{[k+1]} - f_{n,j}^{[k]}\right) + \left(H_{n,j+1} - H_{n,j}\right) + \sum_{i=1}^{p}\omega_{ji} f_{n,i}^{[k]},$$

where $f_{n,i}^{[k]} = f(t_{n,i}, Y_{n,i}^{[k]})$.

4.3. Order of Convergence

Suppose that the integrator used to construct the provisional solution has order p_0, the integrator used to compute the correction approximation at level k has order p_k, and m correction steps are taken. If we have p groups of N uniformly distributed nodes, then the SDC methods can attain order [26,27]

$$O(h^{\min(P_m, p+1)}),$$

where $P_m = \sum_{k=0}^{m} p_k$.

The convergence rate of our fully discrete time-stepping SDC method is given in the following theorem.

Theorem 1. *For any sufficiently smooth function $f : [0, T] \times \mathbb{R} \to \mathbb{R}$, the SDC method outlined in the previous section using $p + 1$ distinct quadrature nodes and m correction steps converges to the exact solution $(Y(\gamma_0), \ldots, Y(\gamma_p))$ with order of accuracy*

$$O(h^{\min\{p+1, (m+1)\alpha\}}).$$

5. Adaptive Implementation

Adaptive marching and accuracy control play a significant role in practical applications involving numerical schemes for solving differential equations. The schemes proposed in this paper are essentially one-step methods, making adaptive implementation relatively simple. Additionally, the schemes have a high order of convergence, which makes accuracy control much easier.

In what follows in this section, we describe details of an adaptive implementation of the SDC schemes and illustrate a number of numerical applications we have performed.

5.1. Accuracy Control

Given a fractional IVP (1) on the interval $[0, T]$, an approximate solution $Y_n^{[m+1]}$,

$$Y_n^{[m+1]} = \left(Y_{n,1}^{[m+1]}, Y_{n,2}^{[m+1]}, \ldots, Y_{n,p}^{[m+1]}\right) \approx \left(y(t_{n,1}), y(t_{n,2}), \ldots, y(t_{n,p})\right)$$

and a positive ϵ, we would like to determine whether

$$\left\| Y_n^{[m+1]} - y \right\| < \epsilon. \tag{18}$$

The structure of the proposed SDC method provides us with a number of completely reliable conditions which can be used as stopping criteria:

1. We verify that $\left\| E_n^{[m+1]} \right\| < \epsilon$, where the vector $E_n^{[m+1]}$ (see (17)),

$$E_n^{[m+1]} = \left(E_{n,1}^{[m+1]}, E_{n,2}^{[m+1]}, \ldots, E_{n,p}^{[m+1]}\right) \tag{19}$$

   is obtained during the last correction.
2. Once $Y_n^{[m+1]}$ has been computed; the value of the approximate solution at an arbitrary time $t \in [t_n, t_{n+1}]$ can be obtained by the Lagrange interpolant. We apply the interpo-

lation process to both $Y_n^{[m+1]}$ and $Y_n^{[m]}$ and demand that the difference be less than ϵ. This indicates that both the correction process and the discretization have converged to precision.

We can also control the step-size h. We start with an arbitrary step-size h and attempt to apply the SDC scheme. The step-size is halved and the procedure is repeated if the obtained precision is insufficient (per criteria (18) and (19)). However, the step-size is unchanged if the precision is acceptable and doubled if the precision is acceptable two steps in a row.

5.2. Computational Details and Numerical Tests

In what follows, we present some numerical results of our numerical method applied to nonlinear fractional differential equations.

We solve all test problems using the proposed SDC methods: implicit SDC (IM-SDC) or explicit SDC (EX-SDC). While using the implicit scheme, we use Newton's method to solve the resulting system of nonlinear equations. Moreover, if the exact solution to the given problem is not known, we use reference solutions computed by the IM-SDC with error tolerance $\epsilon = 10^{-10}$ instead.

As a first example, we study the nonlinear fractional differential equation

$$\begin{cases} D_t^{\alpha} y(t) = \frac{40320}{\Gamma(9-\alpha)} t^{8-\alpha} - \frac{3\Gamma(5+\alpha/2)}{\Gamma(5-\alpha/2)} t^{4-\alpha/2} + \left(\frac{3}{2} t^{\alpha/2} - t^4\right)^3 \\ \qquad -y(t)^{3/2} + \frac{9\Gamma(\alpha+1)}{4}, \\ y(0) = 0. \end{cases} \tag{20}$$

The exact solution is $y(t) = t^8 - 3t^{4+\alpha/2} + \frac{9}{4} t^{\alpha}$.

We apply the IM-SDC method to (20) with $\alpha = 0.75$, $p = 5$, $m = 3$, $\epsilon = 10^{-6}$, and initial step-size $h_0 = 2^{-4}$. When $m = 0$, the local order of the method is $1 + \alpha = 3/2$. When we perform correction iterations, i.e., $m \geq 1$, we measure a high order of convergence. For example, with $m = 3$, we obtain order three which is in agreement with the expected theoretical results. We present our numerical results in Figure 1. The results show that high accuracy can be obtained using few correction iterations after applying high-order SDC methods.

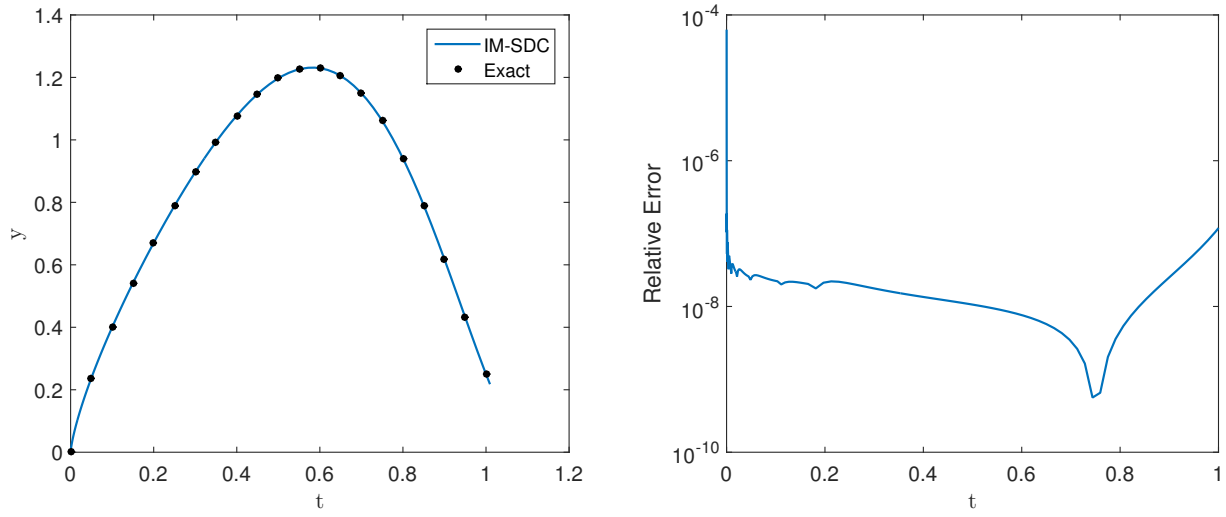

Figure 1. *Cont.*

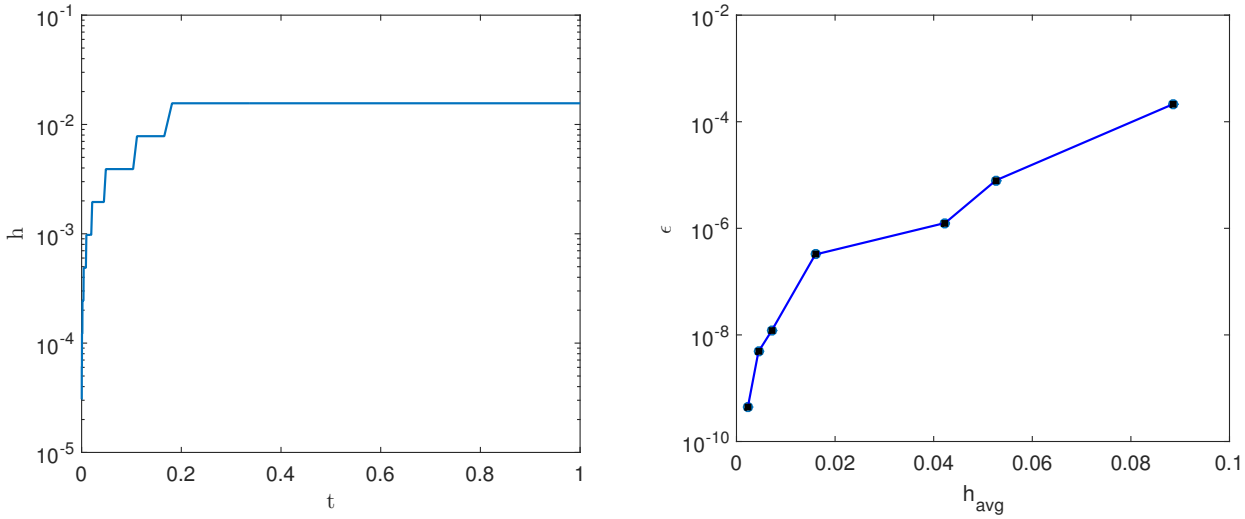

Figure 1. (**top left**)—solution plot; (**top right**)—accuracy vs. time; (**bottom left**)—step-size vs. time; (**bottom right**)—accuracy vs. h_{avg}.

We also studied the fractional-order Chua's circuit system [28]

$$\begin{aligned} D_t^{\alpha} x(t) &= \beta_1 (y(t) - H(x(t))), \\ D_t^{\alpha} y(t) &= x(t) - y(t) + z(t), \\ D_t^{\alpha} z(t) &= -\beta_2 y(t), \end{aligned}$$

where β_1 and β_2 are constant parameters, and $H(\cdot)$ is given by

$$H(v) = -\frac{1}{7}v + \frac{2}{7}v^3.$$

In this test, $\alpha = 0.9, \beta_1 = 12.5, \beta_2 = 100/7, p = 6, m = 5, \epsilon = 10^{-6}$, $(x_1(0), x_2(0), x_3(0)) = (0, 0.1, 0)$, and $T = 50$. The approximations $Y_n^{[m+1]}$ are compared with a reference solution Y_{ref} obtained with $\epsilon = 10^{-10}$. Figure 2 shows the reference solutions $x_{ref}, y_{ref}, z_{ref}$ and the curve $\left(x_{ref}, z_{ref}\right)$ in the (x, z) plane.

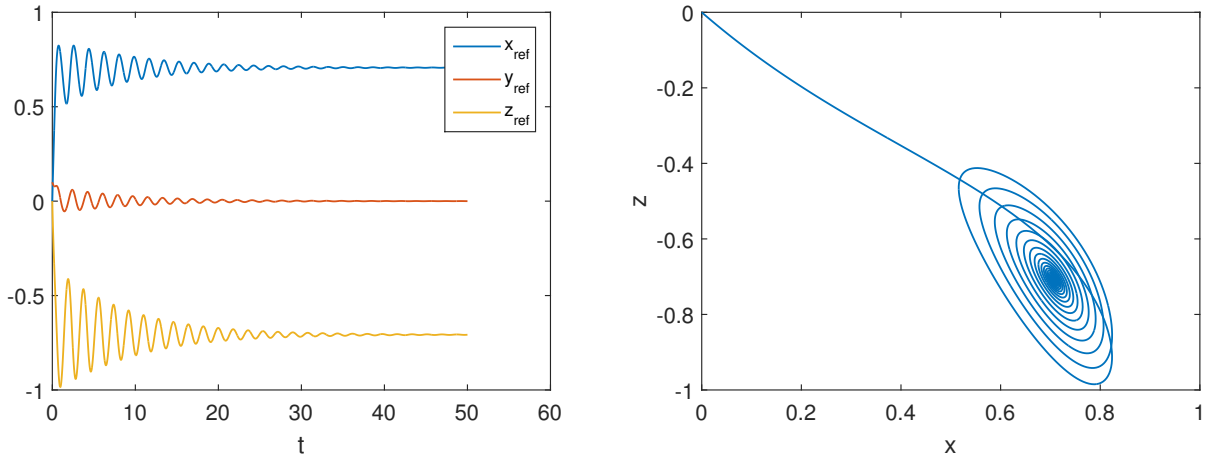

Figure 2. Chua's circuit: (**left**)—solution plot; (**right**)—the curve $\left(x_{ref}, z_{ref}\right)$ in the (x, z) plane.

In Figure 3, the accuracy is plotted as a function of the average time-step h_{avg}.

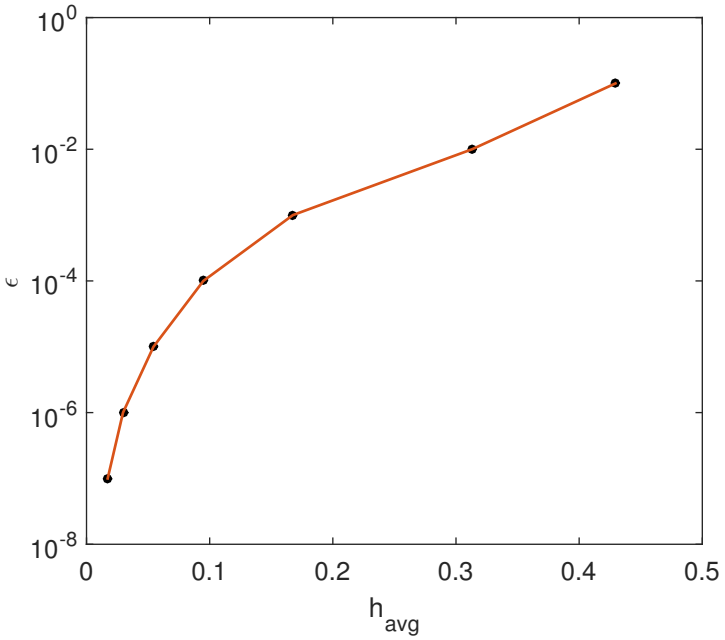

Figure 3. The accuracy as a function of the average time-step h_{avg}.

6. Conclusions

We proposed high-order adaptive time-stepping schemes for nonlinear FDEs by using suitable kernel compression and SDC methods. We discussed the accuracy and local convergence of the proposed schemes. We also discussed the extension of these schemes to construct other higher-order methods for nonlinear fractional initial-value problems.

We provided numerical examples to verify the efficiency of the proposed schemes, which shows that applying suitable correction iterations can increase the accuracy in a significant way.

In future work, we will focus on proposing high-order schemes for the numerical solutions of fractional boundary value problems and time-fractional diffusion equations.

Author Contributions: Conceptualization, O.A. and F.A.; methodology, F.A., S.A.-S. and E.R.; software, F.A.; validation, S.A.-S., O.A. and E.R.; formal analysis, F.A.; investigation, S.A.-S. and O.A.; resources, E.R.; data curation, E.R.; writing—original draft preparation, F.A., O.A., E.R. and S.A.-S.; writing—review and editing, S.A.-S. and O.A.; supervision, E.R.; project administration, F.A.; funding acquisition, F.A. All authors have read and agreed to the published version of the manuscript.

Funding: Fadi Awawdeh work was supported by The Hashemite University through project 12/2021.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors are very grateful to the anonymous referees for their careful reading and valuable suggestions.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Dastgerdi, M.; Bastani, A. Solving Parametric Fractional Differential Equations Arising from the Rough Heston Model Using Quasi-Linearization and Spectral Collocation. *SIAM J. Financ. Math.* **2020**, *11*, 1063–1097. [CrossRef]
2. Li, C.; Zeng, F. Numerical Methods for Fractional Calculus. In *Chapman & Hall/CRC Numerical Analysis and Scientific Computing*; CRC Press: Boca Raton, FL, USA, 2015.
3. Metzler, R.; Klafter, J. The restaurant at the end of the random walk: Recent developments in the description of anomalous transport by fractional dynamics. *J. Phys. A* **2004**, *37*, 161–208. [CrossRef]

4. Perdikaris, P.; Karniadakis, G.E. Fractional-order viscoelasticity in one-dimensional blood flow models. *Ann. Biomed. Eng.* **2014**, *42*, 1012–1023. [CrossRef]
5. Wang, W.; Chen, X.; Ding, D.; Lei, S.L. Circulant preconditioning technique for barrier options pricing under fractional diffusion models. *Int. J. Comput. Math.* **2015**, *92*, 2596–2614. [CrossRef]
6. Gao, G.H.; Sun, Z.Z.; Zhang, H.W. A new fractional numerical differentiation formula to approximate the Caputo fractional derivative and its applications. *J. Comput. Phys.* **2014**, *259*, 33–50. [CrossRef]
7. Jiang, S.; Zhang, J.; Zhang, Q.; Zhang, Z. Fast evaluation of the Caputo fractional derivative and its applications to fractional diffusion equations. *Commun. Comput. Phys.* **2017**, *21*, 650–678. [CrossRef]
8. Li, C.; Yi, Q.; Chen, A. Finite difference methods with non-uniform meshes for nonlinear fractional differential equations. *J. Comput. Phys.* **2016**, *316*, 614–631. [CrossRef]
9. Lv, C.W.; Xu, C.J. Error analysis of a high order method for time-fractional diffusion equations. *SIAM J. Sci. Comput.* **2016**, *38*, A2699–A2724. [CrossRef]
10. Zayernouri, M.; Karniadakis, G.E. Exponentially accurate spectral and spectral element methods for fractional ODEs. *J. Comput. Phys.* **2014**, *257*, 460–480. [CrossRef]
11. Zeng, F.; Turner, I.; Burrage, K. A Stable Fast Time-Stepping Method for Fractional Integral and Derivative Operators. *J. Sci. Comput.* **2018**, *77*, 283–307. [CrossRef]
12. Yan, Y.G.; Sun, Z.Z.; Zhang, J.W. Fast evaluation of the Caputo fractional derivative and its applications to fractional diffusion equations: A second-order scheme. *Commun. Comput. Phys.* **2017**, *22*, 1028–1048. [CrossRef]
13. Baffet, D.; Hesthaven, J.S. High-order accurate adaptive kernel compression time-stepping schemes for fractional differential equations. *J. Sci. Comput.* **2019**, *72*, 1169–1195. [CrossRef]
14. Baffet, D.; Hesthaven, J.S. A kernel compression scheme for fractional differential equations. *SIAM J. Numer. Anal.* **2017**, *55*, 496–520. [CrossRef]
15. Cao, W.; Zhang, Z.; Karniadakis, G.E. Time-splitting schemes for fractional differential equations I: Smooth solutions. *SIAM J. Sci. Comput.* **2015**, *37*, A1752–A1776. [CrossRef]
16. Li, J.R. A fast time stepping method for evaluating fractional integrals. *SIAM J. Sci. Comput.* **2010**, *31*, 4696–4714. [CrossRef]
17. Jin, B.; Lazarov, R.; Pasciak, J.; Zhou, Z. Error analysis of a finite element method for the space-fractional parabolic equation. *SIAM J. Numer. Anal.* **2014**, *52*, 2272–2294. [CrossRef]
18. Tian, W.Y.; Deng, W.; Wu, Y. Polynomial spectral collocation method for space fractional advection-diffusion equation. *Numer. Methods Partial. Differ. Equ.* **2014**, *30*, 514–535. [CrossRef]
19. Podlubny, I.; Chechkin, A.; Skovranek, T.; Chen, Y.; Vinagre Jara, B.M. Matrix approach to discrete fractional calculus. II. Partial fractional differential equations. *J. Comput. Phys.* **2009**, *228*, 3137–3153. [CrossRef]
20. Stynes, M.; O'Riordan, E.; Gracia, J. Error analysis of a finite difference method on graded meshes for a time-fractional diffusion equation. *SIAM J. Numer. Anal.* **2017**, *55*, 1057–1079. [CrossRef]
21. Rauh, A.; Jaulin, L. Novel Techniques for a Verified Simulation of Fractional-Order Differential Equations. *Fractal Fract.* **2021**, *5*, 17. [CrossRef]
22. Jin, S.; Xie, J.; Qu, J.; Chen, Y. A Numerical Method for Simulating Viscoelastic Plates Based on Fractional Order Model. *Fractal Fract.* **2022**, *6*, 150. [CrossRef]
23. Bohmer, K.; Stetter, H.J. *Defect Correction Methods Theory and Applications*; Springer: New York, NY, USA, 1984.
24. Buvoli, T. A Class of Exponential Integrators Based on Spectral Deferred Correction. *SIAM J. Sci. Comput.* **2020**, *42*, A1–A27. [CrossRef]
25. Dutt, A.; Greengard, L.; Rokhlin, V. Spectral deferred correction methods for ordinary differential equations. *BIT* **2000**, *40*, 241–266. [CrossRef]
26. Causley, M.F.; Seal, D.C. On the convergence of spectral deferred correction methods. *Commun. Appl. Math. Comput. Sci.* **2019**, *14*, 33–64. [CrossRef]
27. Ong, B.W.; Spiteri, R.J. Deferred Correction Methods for Ordinary Differential Equations. *J. Sci. Comput.* **2020**, *38*, 60–83. [CrossRef]
28. Cafagna, D.; Grassi, G. Fractional-Oder Chua's Circuit: Time-Domain Analysis, Bifurcation, Chaotic Behavior and Test for Chaos. *Int. J. Bifurcat Chaos* **2008**, *18*, 615–639. [CrossRef]