High order algorithms for numerical solution of fractional differential equations

Mohammad Shahbazi Asl[a], Mohammad Javidi[a,*], Yubin Yan[b]

[a]*Department of Mathematics, University of Tabriz, Tabriz, Iran.*
[b]*Department of Mathematics, University of Chester, Chester CH1 4BJ, UK.*

Abstract

In this paper, two novel high order numerical algorithms are proposed for solving fractional differential equations where the fractional derivative is considered in the Caputo sense. The total domain is discretized into a set of small subdomains and then the unknown functions are approximated using the piecewise Lagrange interpolation polynomial of degree three and degree four. The detailed error analysis is presented, and it is analytically proven that the proposed algorithms are of orders 4 and 5. The stability of the algorithms is rigorously established and the stability region is also achieved. Numerical examples are provided to check the theoretical results and illustrate the efficiency and applicability of the novel algorithms.

Keywords: Fractional differential equation, Caputo fractional derivative, Stability analysis, Error estimates

2010 MSC: 26A33, 65D05, 65D30

1. Introduction

The subject of fractional calculus (theory of integration and differentiation of arbitrary order) can be considered as an old yet novel topic. It is an ongoing topic for more than 300 years, however since the 1970s, it has been gaining increasing attention [1]. Firstly, there were almost no practical applications of fractional calculus (FC),

*Corresponding author
Email addresses: mshahbazia@yahoo.com (Mohammad Shahbazi Asl), mo_javidi@tabrizu.ac.ir (Mohammad Javidi), y.yan@chester.ac.uk (Yubin Yan)

and it was considered by many as an abstract area containing only mathematical manipulations of little or no use [2, 3, 4]. Recently, FC has been widely used in various applications in almost every field of science, engineering, and mathematics and it have gained considerable importance due to their frequent appearance applications in fluid flow, polymer rheology, economics, biophysics, control theory, psychology and so on [5, 6, 7].

The main reason that fractional differential equations (FDEs) are being used to modeling real phenomena is that they are nonlocal in nature, that is, a realistic model of a physical phenomenon depends not only on the time instant but also the previous time history [8]. In other words, fractional derivative provides a perfect tool when it is used to describe the memory and hereditary properties of various materials and processes [9, 10]. Some of the other main differences between fractional calculus and classical calculus are: (1) FDEs are, at least, as stable as their integer order counterpart [11, 12]; (2) Using FDEs can help to reduce the errors arising from the neglected parameters in modeling real-life phenomena [13, 14]; (3) In some situations, FDEs models seem more consistent with the real phenomena than the integer-order models [15, 16]; (4) Fractional order models are more general [17] and in the limit results obtained from FC coincide with those obtained from classical calculus [18] and so on.

The wide applicability of FC in the field of science and engineering motivates researchers to try to find out the analytical or numerical solutions for the FDEs. It is well known that the analytical and closed solutions of FDEs cannot generally be obtained and if luckily obtained always contain some infinite series (such as Mittag-Leffler function) which make evaluation very expensive [19**?** , 20]. For this reason, necessarily, one may need an efficient approximate and numerical technique for the solution of FDEs [21].

Odibat et al. constructed a numerical scheme for the numerical solution of FDEs based on the modified trapezoidal rule and the fractional Euler's method [22]. To obtain a numerical solution scheme for the fractional differential equations, authors of [23] divided the time interval into a set of small subintervals, and utilized quadratic interpolation polynomial between two successive intervals to approximate unknown functions. Cao and Xu applied quadratic interpolation polynomial to construct a high

order scheme based on the so-called block-by-block approach, for the fractional ordinary differential equations [24]. The convergence order of this scheme is $3+\alpha$ for $0<\alpha\leq 1$, and 4 for $\alpha>1$. Diethelm proposed an implicit numerical algorithm for solving FDEs by using piecewise linear interpolation polynomials to approximated the Hadamard finite-part integral [25]. Yan et al. designed a high order numerical scheme for solving a linear fractional differential equation by approximating the Hadamard finite-part integral with the quadratic interpolation polynomials [26]. This method is based on a direct discretisation of the fractional differential operator and the order of convergence of the method is $O(h^{3-\alpha})$. A high order fractional Adams-type method for solving a nonlinear FDEs is also obtained in this paper. Pal et al. designed an extrapolation algorithm for solving linear FDEs based on the direct discretization of the fractional differential operator [27].

In this paper, we will introduce two new numerical algorithms for solving the nonlinear FDEs, which are expressed in terms of Caputo type fractional derivatives. In these algorithms properties of the Caputo derivative are used to reduce the FDE into a Volterra type integral equation of the second kind. We then use the Lagrange interpolation polynomials of degree three and four to approximate the integral and the proposed numerical algorithms has the truncation error $O(h^4)$ and $O(h^5)$ for all $\alpha>0$. The stability of the numerical method is proved based on the properties of the weights in the numerical algorithm under the assumption that the time $T>0$ is sufficiently small. Such properties are used in the first time to prove the stability of the numerical methods for solving fractional differential equations. To our best knowledge, there is no numerical algorithm for solving nonlinear fractional differential equation with the convergence order greater than 4 in the literature. We also introduce a new way to analyze the stability of the numerical methods for solving fractional differential equations.

The outline of the paper is as follows. Numerical algorithms are presented in Section 2 by using the piecewise Lagrange interpolation polynomial of degree three and degree four. Section 3 deals with the error analysis of the presented algorithms and stability analysis of these algorithms is given in Section 4. Linear stability analysis of the proposed schemes is given in Section 5 to achieve stability region of these methods. To demonstrate the effectiveness and high accuracy of the proposed methods some

numerical examples are provided in Section 6. Finally, Some concluding remarks are given in Section 7.

2. Numerical algorithms

Consider the nonlinear fractional differential equation, with $\alpha > 0$,

$$\begin{cases} {}_0^C D_t^\alpha y(t) = f(t, y(t)), \quad 0 \le t \le T, \\ y^{(k)}(t_0) = y_0^{(k)}, \quad k = 0, 1, ..., \lceil \alpha \rceil - 1 \end{cases} \tag{1}$$

where ${}_0^C D_t^\alpha$ denotes the Caputo fractional derivative and $f(t,u)$ satisfies the Lipschitz condition with respect to the second variable, i.e., there exists a constant $L > 0$ such that

$$|f(t,u_1) - f(t,u_2)| \le L|u_1 - u_2|, \quad \forall u_1, u_2 \in \mathbf{R}. \tag{2}$$

It is well known that the initial value problem (1) is equivalent to the Volterra integral equation

$$y(t) = h(t) + \frac{1}{\Gamma(\alpha)} \int_0^t (t-\tau)^{\alpha-1} f(\tau, y(\tau)) d\tau, \tag{3}$$

where $h(t) = \sum_{j=0}^{\lceil \alpha \rceil - 1} \frac{t^j}{j!} y^{(j)}(0)$, in the sense that a continuous function solves (3) if and only if it solves (1). The piecewise Lagrange interpolation polynomial of degree three and degree four are used to approximate the integral in (3). For an integer N and the given time T, the interval $[0,T]$ is divided into $t_j = jh,\ j = 0, \ldots, N$ where $h = T/N$ is the step length. The numerical solution of Eq. (1) at the point t_j is denoted by y_j. For notational convenience, let $F(\tau) = f(\tau, y(\tau))$ and $F_j = f(t_j, y_j)$.

2.1. Numerical algorithm I

We start with computing the value of $y(t)$ at t_1, t_2 and t_3, simultaneously. Consider the following integral for the first three steps ($k = 0, 1, 2$)

$$\begin{aligned} I_{k+1} &= \int_0^{t_{k+1}} (t_{k+1} - \tau)^{\alpha-1} F(\tau) d\tau = \sum_{j=0}^{k} \int_{t_j}^{t_{j+1}} (t_{k+1} - \tau)^{\alpha-1} F(\tau) d\tau \\ &\approx \sum_{j=0}^{k} \int_{t_j}^{t_{j+1}} (t_{k+1} - \tau)^{\alpha-1} \tilde{F}(\tau) d\tau = \sum_{j=0}^{3} d_j^{k+1} F(t_j) \end{aligned} \tag{4}$$

where $\tilde{F}(\tau)$ is chosen to be the piecewise Lagrange cubic interpolation polynomial of $F(\tau)$ associated with the nodes t_0, t_1, t_2 and t_3. In this way, we have

$$\hat{d}_j^1 = \begin{cases} 6\alpha^3+25\alpha^2+23\alpha, & j=0 \\ 6(3\alpha^2+10\alpha+6), & j=1 \\ -9\alpha^2-21\alpha, & j=2 \\ -2\alpha^2-4\alpha, & j=3 \end{cases}, \quad \hat{d}_j^2 = \begin{cases} 2^{\alpha+1}\alpha(\alpha+2)(3\alpha+1), & j=0 \\ 3\times 2^{\alpha+2}\left(3\alpha^2+5\alpha\right), & j=1 \\ 3\times 2^{\alpha+1}\left(-3\alpha^2+\alpha+6\right), & j=2 \\ 2^{\alpha+2}\alpha^2-\alpha & j=3 \end{cases}$$

$$\hat{d}_j^3 = \begin{cases} 3^{\alpha+1}(2\alpha^3+\alpha^2+3\alpha), & j=0 \\ 2\times 3^{\alpha+3}\alpha^2, & j=1 \\ -3^{\alpha+3}(\alpha^2-3\alpha), & j=2 \\ 2\times 3^{\alpha+1}(\alpha^2-4\alpha+6) & j=3 \end{cases}, \quad d_j^{k+1} = \frac{h^\alpha}{6\alpha(\alpha+1)(\alpha+2)(\alpha+3)}\hat{d}_j^{k+1}. \tag{5}$$

For this reason, after some elementary calculations y_{k+1} for the first three steps $k = 0,1,2$ can be approximated as follows:

$$y_{k+1} = h(t_{k+1}) + \frac{1}{\Gamma(\alpha)}\sum_{j=0}^{3} d_j^{k+1} f(t_j, y_j), \quad k=0,1,2. \tag{6}$$

As it is mentioned above, the first three step solutions y_1, y_2 and y_3 are coupled in (6), thus need to be solved simultaneously. An explicit solution of these three equations is given in Appendix A section.

To construct the scheme for the next steps, I_{k+1}, $k \geq 3$ is descritized as follows

$$I_{k+1} \approx \left[\sum_{j=0}^{2}\int_{t_j}^{t_{j+1}} \tilde{F}(\tau) + \sum_{j=3}^{k}\int_{t_j}^{t_{j+1}} \tilde{F}_{j+1}(\tau)\right](t_{k+1}-\tau)^{\alpha-1}d\tau = \sum_{j=0}^{k+1} d_j^{k+1} F(t_j), \tag{7}$$

in which like as (4), for the first three integrals ($j = 0,1,2,3$), $\tilde{F}$ is the piecewise Lagrange cubic interpolation polynomial of $F(\tau)$ associated with the nodes t_0, t_1, t_2 and t_3. For the reminder integrals ($j = 3,4,\ldots,k+1$), $\tilde{F}_{j+1}$ is chosen to be the piecewise Lagrange cubic interpolation polynomial of $F(\tau)$ associated with the nodes t_{j-2}, t_{j-1}, t_j and t_{j+1}. In this way, for $k \geq 3$ we have

$$\begin{aligned} \hat{d}_0^{k+1} =& (k+1)^\alpha\left[6\alpha^3+25\alpha^2+23\alpha+12\alpha k^2-k\left(11\alpha^2+31\alpha+6(k-2)(k-1)\right)\right] \\ &+2P(1)(k-2)^{\alpha+1}, \end{aligned}$$

$$\begin{aligned}\hat{d}_1^{k+1} =&6(k+1)^{\alpha+1}\left[3k(k-3)-5k\alpha+3\alpha^2+10\alpha+6\right]+\frac{6}{3}P(2)(k-3)^{\alpha+1}\\&-\frac{24}{3}P(1)(k-2)^{\alpha+1},\end{aligned}$$

$$\begin{aligned}\hat{d}_2^{k+1} =&-3(k+1)^{\alpha+1}\left[6k(k-2)-8k\alpha+3\alpha^2+7\alpha\right]+2P(3)(k-4)^{\alpha+1}\\&-\frac{24}{3}P(2)(k-3)^{\alpha+1}+12P(1)(k-2)^{\alpha+1},\end{aligned}$$

$$\begin{aligned}\hat{d}_3^{k+1} =&2(k+1)^{\alpha+1}\left[\alpha^2+2\alpha+3k^2-3\alpha k-3k\right]+2P(4)(k-5)^{\alpha+1}-8P(3)(k-4)^{\alpha+1}\\&+12P(2)(k-3)^{\alpha+1}-8P(1)(k-2)^{\alpha+1},\end{aligned}$$

$$\begin{aligned}\hat{d}_j^{k+1} =&2P(j-3)(k-j+2)^{\alpha+1}-8P(j-2)(k-j+1)^{\alpha+1}+12P(j-1)(k-j)^{\alpha+1}\\&-8P(j)(k-j-1)^{\alpha+1}+2P(j+1)(k-j-2)^{\alpha+1},\qquad 4\le j\le k-2,\end{aligned}$$

$$\hat{d}_{k-1}^{k+1} =6\hat{d}_{k+1}^{k+1}-2^{\alpha+4}\varphi+2\times 3^{\alpha+1}\left[\alpha^2+14\alpha+60\right],\quad \hat{d}_k^{k+1}=2^{\alpha+2}\varphi-4\hat{d}_{k+1}^{k+1},$$

$$\hat{d}_{k+1}^{k+1} =2\alpha^2+16\alpha+36,$$

in which $d_j^{k+1}=\frac{h^\alpha}{6\alpha(\alpha+1)(\alpha+2)(\alpha+3)}\hat{d}_j^{k+1}$ and

$$\varphi=\alpha^2+11\alpha+36,\quad P(j)=\alpha(\alpha+2)+3j^2-3j(\alpha+2k+1)+3k^2+3(\alpha+1)k.$$

The following special cases should be excluded:

$$\begin{cases}\hat{d}_2^4=-3\left[4^{\alpha+1}(3\alpha^2-17\alpha+18)-2\hat{d}_{k+1}^{k+1}\right],\\ \hat{d}_3^4=2\left[4^{\alpha+1}(\alpha^2-7\alpha+18)-2\hat{d}_{k+1}^{k+1}\right],\end{cases}\tag{8}$$

$$\hat{d}_3^5=2\left[5^{\alpha+1}(\alpha^2-10\alpha+36)-2^{\alpha+3}\varphi+3\hat{d}_{k+1}^{k+1}\right].\tag{9}$$

For this reason, after some explicit calculations y_{k+1} for $k\ge 3$ can be approximated as follows:

$$y_{k+1}=h(t_{k+1})+\frac{1}{\Gamma(\alpha)}\sum_{j=0}^{k+1}d_j^{k+1}f(t_j,y_j),\quad k\ge 3.\tag{10}$$

To summarize, we obtain the following novel scheme:

$$y_{k+1}=h(t_{k+1})+\frac{1}{\Gamma(\alpha)}\sum_{j=0}^{k+1}d_j^{k+1}f(t_j,y_j),\quad k\ge 1,\tag{11}$$

where d_j^{k+1} are defined as above.

2.2. Numerical algorithm II

Consider the following integral for the first four steps ($k = 0, 1, 2, 3$)

$$I_{k+1} = \int_0^{t_{k+1}} (t_{k+1} - \tau)^{\alpha-1} F(\tau) d\tau = \sum_{j=0}^{k} \int_{t_j}^{t_{j+1}} (t_{k+1} - \tau)^{\alpha-1} F(\tau) d\tau$$
$$= \sum_{j=0}^{k} \int_{t_j}^{t_{j+1}} (t_{k+1} - \tau)^{\alpha-1} \tilde{F}(\tau) d\tau = \sum_{j=0}^{4} b_j^{k+1} F(t_j) \tag{12}$$

where $\tilde{F}(\tau)$ is the piecewise Lagrange interpolation polynomial of degree four associated with the nodes t_0, t_1, t_2, , t_3 and t_4. Therefore, one can achieve the following weights

$$\begin{cases} \hat{b}_0^1 = \alpha \left[12\alpha^3 + 95\alpha^2 + 230\alpha + 165\right], \\ \hat{b}_1^1 = 4\left[12\alpha^3 + 82\alpha^2 + 157\alpha + 72\right], \\ \hat{b}_2^1 = -6\alpha\left[6\alpha^2 + 35\alpha + 47\right], \\ \hat{b}_3^1 = 4\alpha\left[4\alpha^2 + 22\alpha + 27\right], \\ \hat{b}_4^1 = -\alpha\left[3\alpha^2 + 16\alpha + 19\right], \end{cases} \quad \begin{cases} \hat{b}_0^2 = 2^{\alpha+1}\alpha\left[6\alpha^3 + 35\alpha^2 + 55\alpha + 20\right], \\ \hat{b}_1^2 = 2^{\alpha+5}\alpha\left[3\alpha^2 + 14\alpha + 14\right], \\ \hat{b}_2^2 = 3 \times 2^{\alpha+3}(\alpha+1)(\alpha+3)(4-3\alpha), \\ \hat{b}_3^2 = 2^{\alpha+5}\alpha\left[\alpha^2 + 2\alpha - 2\right], \\ \hat{b}_4^2 = 2^{\alpha+1}\alpha\left[-3\alpha^2 - 5\alpha + 4\right], \end{cases}$$

$$\begin{cases} \hat{b}_0^3 = 3^{\alpha+1}\alpha\left[4\alpha^3 + 15\alpha^2 + 20\alpha + 15\right], \\ \hat{b}_1^3 = 4 \times 3^{\alpha+2}\alpha\left[4\alpha^2 + 10\alpha + 3\right], \\ \hat{b}_2^3 = 2 \times 3^{\alpha+3}\alpha\left[-2\alpha^2 + \alpha + 9\right], \\ \hat{b}_3^3 = 4 \times 3^{\alpha+1}\left[4\alpha^3 - 6\alpha^2 - \alpha + 24\right], \\ \hat{b}_4^3 = -3^{\alpha+2}\alpha\left[\alpha^2 - 2\alpha + 3\right], \end{cases} \quad \begin{cases} \hat{b}_0^4 = 4^{\alpha+1}\alpha^2\left[3\alpha^2 + 5\alpha + 20\right], \\ \hat{b}_1^4 = 4^{\alpha+3}\alpha\left[3\alpha^2 + \alpha + 4\right], \\ \hat{b}_2^4 = 3 \times 4^{\alpha+2}\alpha\left[-3\alpha^2 + 11\alpha - 4\right], \\ \hat{b}_3^4 = 4^{\alpha+3}\alpha\left[\alpha^2 - 5\alpha + 12\right], \\ \hat{b}_4^4 = -4^{\alpha+1}(\alpha-2)(3\alpha^2 - 11\alpha + 36), \end{cases} \tag{13}$$

here $b_j^{k+1} = \frac{h^\alpha}{12\alpha(\alpha+1)(\alpha+2)(\alpha+3)(\alpha+4)} \hat{b}_j^{k+1}$. Hence y_{k+1} for the first four steps k=0,1,2,3 can be determined as follows:

$$y_{k+1} = h(t_{k+1}) + \frac{1}{\Gamma(\alpha)} \sum_{j=0}^{4} b_j^{k+1} f(t_j, y_j), \quad 0 \le k \le 3. \tag{14}$$

It is obvious that, the first four step solutions y_1, y_2, y_3 and y_4 are coupled in (14), thus need to be solved simultaneously. An explicit solution of these four equations is given in Appendix B section.

To design the schema for the next steps, $I_{k+1},\ k \geq 4$ is descritized as follows

$$I_{k+1} = \left[\sum_{j=0}^{3} \int_{t_j}^{t_{j+1}} \tilde{F}(\tau) + \sum_{j=4}^{k} \int_{t_j}^{t_{j+1}} \tilde{F}_{j+1}(\tau) \right] (t_{k+1} - \tau)^{\alpha-1} d\tau = \sum_{j=0}^{k+1} b_j^{k+1} F(t_j), \quad (15)$$

in which like as (12), for the first four integrals ($j = 0,1,2,3$), $\tilde{F}$ is the piecewise Lagrange interpolation polynomial of degree four associated with the nodes t_0, t_1, t_2, , t_3 and t_4. For the reminder integrals ($j = 4,5,\ldots,k+1$), $\tilde{F}_{j+1}$ is the piecewise Lagrange interpolation polynomial of degree four associated with the nodes $t_{j-3}, t_{j-2}, t_{j-1}, t_j$ and t_{j+1}. In this way, for $k \geq 4$ we have the following weights

$$\begin{aligned}
\tilde{b}_0^{k+1} =& (k+1)^{\alpha} \left[12\alpha^4 + 5\alpha^2 \left(7k^2 - 31k + 46\right) - 5\alpha(k-3)\left(6k^2 - 13k + 11\right) \right. \\
&\left. + 5\alpha^3(19 - 5k) + 12k(k-3)(k-2)(k-1) \right] - P(2)(k-3)^{\alpha+1},
\end{aligned}$$

$$\begin{aligned}
\tilde{b}_1^{k+1} =& -4(k+1)^{\alpha+1} \left[h_2(-2) - 15(\alpha+4)\left(\alpha^2 + 4\alpha + 3k^2 - \alpha k + 3k + 6\right) \right] \\
&- P(3)(k-4)^{\alpha+1} + 5P(2)(k-3)^{\alpha+1},
\end{aligned}$$

$$\begin{aligned}
\tilde{b}_2^{k+1} =& 6(k+1)^{\alpha+1} \left[h_2(-2) - (\alpha+4)\left(9\alpha^2 + 37\alpha + 42k^2 - 8\alpha k + 60k + 72\right) \right] \\
&- P(4)(k-5)^{\alpha+1} + 5P(3)(k-4)^{\alpha+1} - 10P(2)(k-3)^{\alpha+1},
\end{aligned}$$

$$\begin{aligned}
\tilde{b}_3^{k+1} =& -4(k+1)^{\alpha+1} \left[h_2(-2) - (\alpha+4)\left(7\alpha^2 + 32\alpha + 39k^2 - 3\alpha k + 69k + 72\right) \right] \\
&- P(5)(k-6)^{\alpha+1} + 5P(4)(k-5)^{\alpha+1} - 10P(3)(k-4)^{\alpha+1} + 10P(2)(k-3)^{\alpha+1},
\end{aligned}$$

$$\begin{aligned}
\tilde{b}_4^{k+1} =& (k+1)^{\alpha+1} \left[h_2(-2) - 6(\alpha+4)\left(\alpha^2 + 5\alpha + 6k^2 + 12k + 12\right) \right] - P(6)(k-7)^{\alpha+1} \\
&+ 5P(5)(k-6)^{\alpha+1} - 10P(4)(k-5)^{\alpha+1} + 10P(3)(k-4)^{\alpha+1} - 5P(2)(k-3)^{\alpha+1},
\end{aligned}$$

$$\begin{aligned}
\tilde{b}_j^{k+1} =& -P(j+2)(-j+k-3)^{\alpha+1} + P(j-3)(-j+k+2)^{\alpha+1} - 5P(j-2)(-j+k+1)^{\alpha+1} \\
&+ 10P(j-1)(k-j)^{\alpha+1} + 5P(j+1)(-j+k-2)^{\alpha+1} - 10P(j)(-j+k-1)^{\alpha+1}, \\
& 5 \leq j \leq k-3,
\end{aligned}$$

$$\tilde{b}_{k-2}^{k+1} = 2^{\alpha+2} \left[5\psi_2 + 2^{\alpha} \left(3\alpha^3 + 71\alpha^2 + 674\alpha + 2520\right) \right] - 5 \times 3^{\alpha+2} \psi_1 - 10\tilde{b}_{k+1}^{k+1},$$

$$\tilde{b}_{k-1}^{k+1} = 10\left[\tilde{b}_{k+1}^{k+1} - 2^{\alpha}\psi_2\right] + 3^{\alpha+2}\psi_1, \quad \tilde{b}_{k}^{k+1} = 2^{\alpha+1}\psi_2 - 5\tilde{b}_{k+1}^{k+1},$$
$$\tilde{b}_{k+1}^{k+1} = 3\alpha^3 + 38\alpha^2 + 173\alpha + 288,$$

where $b_j^{k+1} = \frac{h^\alpha}{12\alpha(\alpha+1)(\alpha+2)(\alpha+3)(\alpha+4)}\tilde{b}_j^{k+1}$ and

$$\begin{aligned} P(j) =& 3\alpha^3 + \alpha\left(18j^2 - 36jk - 41j + 18k^2 + 41k + 19\right) + \alpha^2(-11j + 11k + 16) \\ & - 12(j-k-2)(j-k-1)(j-k), \end{aligned}$$

$$\psi_1 = \alpha^3 + 20\alpha^2 + 157\alpha + 480, \quad \psi_2 = 3\alpha^3 + 49\alpha^2 + 304\alpha + 720.$$

The following special cases should be excluded:

$$\begin{cases} \hat{b}_2^5 = -10\left[3\times 5^{\alpha}\left(6\alpha^3 - 41\alpha^2 + 91\alpha - 96\right) + \tilde{b}_{k+1}^{k+1}\right], \\ \hat{b}_3^5 = 10\left[5^{\alpha}\left(8\alpha^3 - 68\alpha^2 + 278\alpha - 288\right) + \tilde{b}_{k+1}^{k+1}\right], \\ \hat{b}_4^5 = -5\left[5^{\alpha}\left(3\alpha^3 - 28\alpha^2 + 143\alpha - 288\right) + \tilde{b}_{k+1}^{k+1}\right], \end{cases} \tag{16}$$

$$\begin{cases} \hat{b}_3^6 = 10(2^{\alpha+1}\psi_2 - \tilde{b}_{k+1}^{k+1}) + 2^{\alpha+5}3^{\alpha+1}\left(\alpha^3 - 12\alpha^2 + 68\alpha - 120\right), \\ \hat{b}_4^6 = 10\left(-2^{\alpha}\psi_2 + \tilde{b}_{k+1}^{k+1}\right) - 2^{\alpha+1}3^{\alpha+2}\left(\alpha^3 - 13\alpha^2 + 88\alpha - 240\right), \end{cases} \tag{17}$$

$$\hat{b}_4^7 = 7^{\alpha+1}\left(-3\alpha^3 + 50\alpha^2 - 421\alpha + 1440\right) - 5\times 3^{\alpha+2}\psi_1 + 5\times 2^{\alpha+2}\psi_2 - 10\tilde{b}_{k+1}^{k+1}. \tag{18}$$

Therefore y_{k+1} for $k \geq 4$ can be approximated as follows:

$$y_{k+1} = h(t_{k+1}) + \frac{1}{\Gamma(\alpha)}\sum_{j=0}^{k+1} b_j^{k+1} f(t_j, y_j), \quad k \geq 4. \tag{19}$$

Thus a new numerical algorithm II is described by (14) and (19) with the weights b_j^{k+1} defined as above.

3. Error analysis

For the numerical algorithm I the truncation error at the step $k+1$ is defined by [24]

$$r_{k+1}(h) := y(t_{k+1}) - \tilde{y}_{k+1} \tag{20}$$

where $\tilde{y}_{k+1}$ is an approximation to $y(t_{k+1})$, evaluated by using the algorithm I (11) with exact previous solutions, i.e. for $k \geq 3$,

$$\tilde{y}_{k+1} = h(t_{k+1}) + \frac{1}{\Gamma(\alpha)} \sum_{j=0}^{k+1} d_j^{k+1} F(t_j). \tag{21}$$

For the numerical algorithm II (19), the definition of truncation error is the same as (20), where $\tilde{y}_{k+1}$ for $k \geq 4$ is as follows:

$$\tilde{y}_{k+1} = h(t_{k+1}) + \frac{1}{\Gamma(\alpha)} \sum_{j=0}^{k+1} b_j^{k+1} F(t_j) \tag{22}$$

Theorem 1. *Let $r_{k+1}(h)$ be the truncation error defined in (20). If $F(\tau) \in C^4[0,T]$ for some suitable chosen T, then for the numerical algorithm I (11) there exist a positive constant $C > 0$, independent of h, such that*

$$|r_{k+1}(h)| \leq Ch^4$$

Proof. We have, by Eqs. (3), (7) and (21),

$$\begin{aligned}
|r_{k+1}(h)| = |y(t_{k+1}) - \tilde{y}_{k+1}| = \Bigg| & h(t_{k+1}) + \frac{1}{\Gamma(\alpha)} \int_0^{t_{k+1}} (t_{k+1}-\tau)^{\alpha-1} F(\tau) d\tau \\
& - h(t_{k+1}) - \frac{1}{\Gamma(\alpha)} \sum_{j=0}^{k+1} b_j^{k+1} F(t_j) \Bigg| \leq \frac{1}{\Gamma(\alpha)} \Bigg| \sum_{j=0}^{2} \int_{t_j}^{t_{j+1}} (t_{k+1}-\tau)^{\alpha-1} F(\tau) d\tau \\
& + \sum_{j=3}^{k} \int_{t_j}^{t_{j+1}} (t_{k+1}-\tau)^{\alpha-1} F(\tau) d\tau - \sum_{j=0}^{2} \int_{t_j}^{t_{j+1}} (t_{k+1}-\tau)^{\alpha-1} \tilde{F}(\tau) d\tau \\
& - \sum_{j=3}^{k} \int_{t_j}^{t_{j+1}} (t_{k+1}-\tau)^{\alpha-1} \tilde{F}_{j+1}(\tau) d\tau \Bigg| \\
\leq & \frac{1}{\Gamma(\alpha)} \Bigg[\sum_{j=0}^{2} \int_{t_j}^{t_{j+1}} (t_{k+1}-\tau)^{\alpha-1} \left| F(\tau) - \tilde{F}(\tau) \right| d\tau \\
& + \sum_{j=3}^{k} \int_{t_j}^{t_{j+1}} (t_{k+1}-\tau)^{\alpha-1} \left| F(\tau) - \tilde{F}_{j+1}(\tau) \right| d\tau \Bigg],
\end{aligned}$$

where $\tilde{F}(\tau)$ and $\tilde{F}_{j+1}$ are defined by (7). Thus we have

$$\begin{aligned}|r_{k+1}(h)| \leq & \frac{1}{\Gamma(\alpha)}\left[\sum_{j=0}^{2}\int_{t_j}^{t_{j+1}}(t_{k+1}-\tau)^{\alpha-1}\left|\frac{F^{(4)}(\xi_1(\tau))}{4!}(\tau-t_0)(\tau-t_1)(\tau-t_2)\right.\right. \\ & \left.(\tau-t_3)\right|d\tau+\sum_{j=3}^{k}\int_{t_j}^{t_{j+1}}(t_{k+1}-\tau)^{\alpha-1}\left|\frac{F^{(4)}(\xi_j(\tau))}{4!}(\tau-t_{j-2})(\tau-t_{j-1})\right. \\ & \left.\left.(\tau-t_j)(\tau-t_{j+1})\right|d\tau\right],\end{aligned}$$

where $\xi_1(\tau)\in[t_0,t_3]$ and $\xi_j(\tau)\in[t_{j-2},t_{j+1}]$.

$$\begin{aligned}|r_{k+1}(h)| \leq & \frac{1}{\Gamma(\alpha)}\left[\sum_{j=0}^{2}\frac{\|F^{(4)}\|_\infty}{4!}\left|(\tilde{\tau}_j-t_0)(\tilde{\tau}_j-t_1)(\tilde{\tau}_j-t_2)(\tilde{\tau}_j-t_3)\right|\int_{t_j}^{t_{j+1}}(t_{k+1}-\tau)^{\alpha-1}d\tau\right. \\ & +\sum_{j=3}^{k}\frac{\|F^{(4)}\|_\infty}{4!}\left|(\tilde{\tau}_j-t_{j-2})(\tilde{\tau}_j-t_{j-1})(\tilde{\tau}_j-t_j)(\tilde{\tau}_j-t_{j+1})\right| \\ & \left.\int_{t_j}^{t_{j+1}}(t_{k+1}-\tau)^{\alpha-1}d\tau\right],\end{aligned}$$

here $\tilde{\tau}_j\in[t_j,t_{j+1}]$ and the second integral mean value theorem is used.

$$\begin{aligned}|r_{k+1}(h)| \leq & \frac{1}{\Gamma(\alpha)}\left[\frac{\|F^{(4)}\|_\infty}{4!}(3h)^4\sum_{j=0}^{2}\frac{1}{\alpha}\left[(t_{k+1}-t_j)^\alpha-(t_{k+1}-t_{j+1})^\alpha\right]\right. \\ & \left.+\frac{\|F^{(4)}\|_\infty}{4!}(3h)^4\sum_{j=3}^{k}\frac{1}{\alpha}\left[(t_{k+1}-t_j)^\alpha-(t_{k+1}-t_{j+1})^\alpha\right]\right] \\ = & \frac{3^3\|F^{(4)}\|_\infty}{8\Gamma(\alpha+1)}h^4(t_{k+1}-t_0)^\alpha=\left(\frac{3^3\|F^{(4)}\|_\infty T^\alpha}{8\Gamma(\alpha+1)}\right)h^4.\end{aligned}$$

□

Theorem 2. *Let $r_{k+1}(h)$ be the truncation error defined in (20). If $F(\tau)\in C^5[0,T]$ for some suitable chosen T, then for the numerical algorithm II (14) and (19) there exists a positive constant $C>0$, independent of h, such that*

$$|r_{k+1}(h)|\leq Ch^5$$

Proof. The details of the proof is similar to that of Theorem 1 so are neglected. We

have, by Eqs. (3), (15) and (21),

$$
\begin{aligned}
|r_{k+1}(h)| = |y(t_{k+1}) - \tilde{y}_{k+1}| = \Bigg| & h(t_{k+1}) + \frac{1}{\Gamma(\alpha)} \int_0^{t_{k+1}} (t_{k+1}-\tau)^{\alpha-1} F(\tau) d\tau \\
& - h(t_{k+1}) - \frac{1}{\Gamma(\alpha)} \sum_{j=0}^{k+1} P_j^{k+1} F(t_j) \Bigg| \\
\leq & \frac{1}{\Gamma(\alpha)} \Bigg[\sum_{j=0}^{3} \int_{t_j}^{t_{j+1}} (t_{k+1}-\tau)^{\alpha-1} \left| F(\tau) - \tilde{F}(\tau) \right| d\tau \\
& + \sum_{j=4}^{k} \int_{t_j}^{t_{j+1}} (t_{k+1}-\tau)^{\alpha-1} \left| F(\tau) - \tilde{F}_{j+1}(\tau) \right| d\tau \Bigg],
\end{aligned}
$$

where $\tilde{F}(\tau)$ and $\tilde{F}_{j+1}$ are defined by (15).

$$
\begin{aligned}
|r_{k+1}(h)| \leq & \frac{1}{\Gamma(\alpha)} \Bigg[\sum_{j=0}^{3} \int_{t_j}^{t_{j+1}} (t_{k+1}-\tau)^{\alpha-1} \left| \frac{F^{(5)}(\xi_1(\tau))}{5!} (\tau-t_0)(\tau-t_1)(\tau-t_2) \right. \\
& \left. (\tau-t_3)(\tau-t_4) \right| d\tau + \sum_{j=4}^{k} \int_{t_j}^{t_{j+1}} (t_{k+1}-\tau)^{\alpha-1} \left| \frac{F^{(5)}(\xi_j(\tau))}{5!} (\tau-t_{j-3}) \right. \\
& \left. (\tau-t_{j-2})(\tau-t_{j-1})(\tau-t_j)(\tau-t_{j+1}) \right| d\tau \Bigg],
\end{aligned}
$$

where $\xi_1(\tau) \in [t_0, t_4]$ and $\xi_j(\tau) \in [t_{j-3}, t_{j+1}]$.

$$
\begin{aligned}
|r_{k+1}(h)| \leq & \frac{1}{\Gamma(\alpha)} \Bigg[\sum_{j=0}^{3} \frac{\|F^{(5)}\|_\infty}{5!} \left| (\tilde{\tau}_j - t_0)(\tilde{\tau}_j - t_1)(\tilde{\tau}_j - t_2)(\tilde{\tau}_j - t_3)(\tilde{\tau}_j - t_4) \right| \int_{t_j}^{t_{j+1}} (t_{k+1}-\tau)^{\alpha-1} d\tau \\
& + \sum_{j=4}^{k} \frac{\|F^{(5)}\|_\infty}{5!} \left| (\tilde{\tau}_j - t_{j-3})(\tilde{\tau}_j - t_{j-2})(\tilde{\tau}_j - t_{j-1})(\tilde{\tau}_j - t_j)(\tilde{\tau}_j - t_{j+1}) \right| \\
& \int_{t_j}^{t_{j+1}} (t_{k+1}-\tau)^{\alpha-1} d\tau \Bigg],
\end{aligned}
$$

in which $\tilde{\tau}_j \in [t_j, t_{j+1}]$.

$$
\begin{aligned}
|r_{k+1}(h)| \leq & \frac{1}{\Gamma(\alpha)} \Bigg[\frac{\|F^{(5)}\|_\infty}{5!} (4h)^5 \sum_{j=0}^{3} \frac{1}{\alpha} \left[(t_{k+1}-t_j)^\alpha - (t_{k+1}-t_{j+1})^\alpha \right] \\
& + \frac{\|F^{(5)}\|_\infty}{5!} (4h)^5 \sum_{j=4}^{k} \frac{1}{\alpha} \left[(t_{k+1}-t_j)^\alpha - (t_{k+1}-t_{j+1})^\alpha \right] \Bigg] \\
= & \frac{4^5 \|F^{(5)}\|_\infty}{5!\Gamma(\alpha+1)} h^5 (t_{k+1}-t_0)^\alpha = \left(\frac{4^5 \|F^{(5)}\|_\infty}{5!\Gamma(\alpha+1)} \right) h^5.
\end{aligned}
$$

□

4. Stability analysis

The stability of a numerical scheme mainly refers to that if there is a perturbation in the initial condition, then the small change cause small errors in the numerical solution [28, 29]. Suppose that y_{k+1} and $\tilde{y}_{k+1}$ are numerical solutions in (11), and the initial conditions are given by $y_0^{(i)}$ and $\tilde{y}_0^{(i)}$ respectively. If there exists a positive constant C independent of h, such that

$$|y_{k+1}-\tilde{y}_{k+1}| \leq C_{\alpha,T}\,\|y_0-\tilde{y}_0\|_\infty, \tag{23}$$

then it concluded that the scheme (11) is stable [30]. It is similar to define the numerical stability for the numerical algorithm II (14) and (19). Assume that $F(\tau)$ is sufficiently smooth and $C_\alpha > 0$ is independent of all discretization parameters. Firstly, we introduce two lemmas which will be used in stability analysis.

Lemma 1. *For the weights of the novel scheme (11) we have*

$$\sum_{j=0}^{k+1}\left|d_j^{k+1}\right| \leq C_\alpha T^\alpha, \tag{24}$$

where C_α only depends on α.

Proof. For d_0^{k+1}, we have

$$\begin{aligned}
\left|d_0^{k+1}\right| &= \left|\sum_{j=0}^{2}\int_{t_j}^{t_{j+1}}(t_{k+1}-\tau)^{\alpha-1}\frac{\tau-t_1}{t_0-t_1}\frac{\tau-t_2}{t_0-t_2}\frac{\tau-t_3}{t_0-t_3}d\tau\right| \\
&\leq \sum_{j=0}^{2}\int_{t_j}^{t_{j+1}}\left|(t_{k+1}-\tau)^{\alpha-1}\right|\left|\frac{\tau-t_1}{t_0-t_1}\frac{\tau-t_2}{t_0-t_2}\frac{\tau-t_3}{t_0-t_3}\right|d\tau \\
&\leq \sum_{j=0}^{2}\left|\frac{\tilde{\tau}_j-t_1}{t_0-t_1}\frac{\tilde{\tau}_j-t_2}{t_0-t_2}\frac{\tilde{\tau}_j-t_3}{t_0-t_3}\right|\int_{t_j}^{t_{j+1}}(t_{k+1}-\tau)^{\alpha-1}d\tau, \quad \tilde{\tau}_j\in[t_j,t_j+1]
\end{aligned}$$

$$\begin{aligned}
\left|d_0^{k+1}\right| \leq & \frac{1}{6h^3}|(\xi_1 h-h)(\xi_1 h-2h)(\xi_1 h-3h)|\frac{1}{\alpha}\left[(t_{k+1})^\alpha-(t_{k+1}-t_1)^\alpha\right] \\
&+\frac{1}{6h^3}|(\xi_2 h-h)(\xi_2 h-2h)(\xi_2 h-3h)|\frac{1}{\alpha}\left[(t_{k+1}-t_1)^\alpha-(t_{k+1}-t_2)^\alpha\right] \\
&+\frac{1}{6h^3}|(\xi_3 h-h)(\xi_3 h-2h)(\xi_3 h-3h)|\frac{1}{\alpha}\left[(t_{k+1}-t_2)^\alpha-(t_{k+1}-t_3)^\alpha\right],
\end{aligned}$$

where $j-1 \le \xi_j \le j$, $j = 1,2,3$. Therefore we have

$$\left|d_0^{k+1}\right| \le \frac{1}{6h^3}\frac{1}{\alpha}t_{k+1}^{\alpha}\left(6h^3+2h^3+2h^3\right) \le \frac{5}{3\alpha}T^{\alpha}.$$

Using similar analysis it can be shown that for $j = 1,2,3,k-1,k,k+1$ there exist C_α, which is dissimilar values at each cases, such that the following inequality is holds.

$$\left|d_j^{k+1}\right| \le C_\alpha T^{\alpha}, j = 1,2,3,k-1,k,k+1. \tag{25}$$

For $j = 4,5,\ldots,k-2$ we have

$$\begin{aligned}
\sum_{j=4}^{k-2}\left|d_j^{k+1}\right| \le \sum_{j=4}^{k-2}\Bigg[& \int_{t_{j-1}}^{t_j}\left|(t_{k+1}-\tau)^{\alpha-1}\right|\left|\frac{\tau-t_{j-3}}{t_j-t_{j-3}}\frac{\tau-t_{j-2}}{t_j-t_{j-2}}\frac{\tau-t_{j-1}}{t_j-t_{j-1}}\right|d\tau \\
& + \int_{t_j}^{t_{j+1}}\left|(t_{k+1}-\tau)^{\alpha-1}\right|\left|\frac{\tau-t_{j-2}}{t_j-t_{j-2}}\frac{\tau-t_{j-1}}{t_j-t_{j-1}}\frac{\tau-t_{j+1}}{t_j-t_{j+1}}\right|d\tau \\
& + \int_{t_{j+1}}^{t_{j+2}}\left|(t_{k+1}-\tau)^{\alpha-1}\right|\left|\frac{\tau-t_{j-1}}{t_j-t_{j-1}}\frac{\tau-t_{j+1}}{t_j-t_{j+1}}\frac{\tau-t_{j+2}}{t_j-t_{j+2}}\right|d\tau \\
& + \int_{t_{j+2}}^{t_{j+3}}\left|(t_{k+1}-\tau)^{\alpha-1}\right|\left|\frac{\tau-t_{j+1}}{t_j-t_{j+1}}\frac{\tau-t_{j+2}}{t_j-t_{j+2}}\frac{\tau-t_{j+3}}{t_j-t_{j+3}}\right|d\tau\Bigg],
\end{aligned}$$

$$\begin{aligned}
\sum_{j=4}^{k-2}\left|d_j^{k+1}\right| \le \sum_{j=4}^{k-2}\Bigg[& \left|\frac{\tilde{\tau}_1-t_{j-3}}{3h}\frac{\tilde{\tau}_1-t_{j-2}}{2h}\frac{\tilde{\tau}_1-t_{j-1}}{h}\right|\int_{t_{j-1}}^{t_j}(t_{k+1}-\tau)^{\alpha-1}d\tau \\
& + \left|\frac{\tilde{\tau}_2-t_{j-2}}{2h}\frac{\tilde{\tau}_2-t_{j-1}}{h}\frac{\tilde{\tau}_2-t_{j+1}}{-h}\right|\int_{t_j}^{t_{j+1}}(t_{k+1}-\tau)^{\alpha-1}d\tau \\
& + \left|\frac{\tilde{\tau}_3-t_{j-1}}{h}\frac{\tilde{\tau}_3-t_{j+1}}{-h}\frac{\tilde{\tau}_3-t_{j+2}}{-2h}\right|\int_{t_{j+1}}^{t_{j+2}}(t_{k+1}-\tau)^{\alpha-1}d\tau \\
& + \left|\frac{\tilde{\tau}_4-t_{j+1}}{-h}\frac{\tilde{\tau}_4-t_{j+2}}{-2h}\frac{\tilde{\tau}_4-t_{j+3}}{-3h}\right|\int_{t_{j+2}}^{t_{j+3}}(t_{k+1}-\tau)^{\alpha-1}d\tau\Bigg],
\end{aligned}$$

where $\tilde{\tau}_1 \in [t_{j-1},t_j]$, $\tilde{\tau}_2 \in [t_j,t_{j+1}]$, $\tilde{\tau}_3 \in [t_{j+1},t_{j+2}]$, and $\tilde{\tau}_4 \in [t_{j+2},t_{j+3}]$. Hence, above equation has the simplify form,

$$\begin{aligned}
\sum_{j=4}^{k-2}\left|d_j^{k+1}\right| \le & \left|\frac{6h^3}{6h^3}\right|\frac{1}{\alpha}\sum_{j=4}^{k-2}\left[(t_{k+1}-t_{j-1})^{\alpha}-(t_{k+1}-t_j)^{\alpha}\right]+\sum_{j=4}^{k-2}\left[(t_{k+1}-t_j)^{\alpha}-(t_{k+1}-t_{j+1})^{\alpha}\right] \\
& + \sum_{j=4}^{k-2}\left[(t_{k+1}-t_{j+1})^{\alpha}-(t_{k+1}-t_{j+2})^{\alpha}\right]+\sum_{j=4}^{k-2}\left[(t_{k+1}-t_{j+2})^{\alpha}-(t_{k+1}-t_{j+3})^{\alpha}\right] \\
& = \frac{1}{\alpha}\Big[\left[(t_{k+1}-t_3)^{\alpha}-(t_{k+1}-t_{k-2})^{\alpha}\right]+\left[(t_{k+1}-t_4)^{\alpha}-(t_{k+1}-t_{k-1})^{\alpha}\right] \\
& + \left[(t_{k+1}-t_5)^{\alpha}-(t_{k+1}-t_k)^{\alpha}\right]+(t_{k+1}-t_6)^{\alpha}\Big],
\end{aligned}$$

$$\sum_{j=4}^{k-2}\left|d_j^{k+1}\right| \le \frac{1}{\alpha}(t_{k+1}-t_3)^\alpha + (t_{k+1}-t_4)^\alpha + (t_{k+1}-t_5)^\alpha + (t_{k+1}-t_6)^\alpha - [t_3^\alpha + t_2^\alpha + t_1^\alpha]$$
$$\le \frac{4}{\alpha} t_{k+1}^\alpha$$

Combining all above results, by choosing sufficiently large C_α, and also sufficiently small T one can get (24) to complete the proof of the Lemma. □

Lemma 2. *For the weights of the novel scheme (19) we have*

$$\sum_{j=0}^{k+1}\left|d_j^{k+1}\right| \le C_\alpha T^\alpha, \tag{26}$$

where C_α only depends on α.

100 *Proof.* The idea of the proof is similar to that of Lemma 1, so is omitted. □

Theorem 3. *Assume that y_j $(j = 1,2,\ldots,k)$ are the solutions of the scheme (11). Then for sufficiently small $T > 0$, the scheme (11) is stable.*

Proof. Suppose that y_{k+1} and $\tilde{y}_{k+1}$ are numerical solutions in (11), and the initial conditions are given by $y_0^{(i)}$ and $\tilde{y}_0^{(i)}$ respectively. We shall use mathematical induction. Assume that

$$\left|y_j - \tilde{y}_j\right| \le C_{\alpha,T}\left\|y_0 - \tilde{y}_0\right\|_\infty, \tag{27}$$

is true for $(j = 0,1,\ldots,k)$. We must prove that this also holds for $j = k+1$. Note that, by assumptions of the given initial conditions, the induction basis $(j = 0)$ is true. We have, using the Lipschitz condition assumption (2),

$$|y_{k+1} - \tilde{y}_{k+1}| \le \sum_{i=0}^{\lceil\alpha\rceil-1} \frac{t_{k+1}^i}{i!}|y_0^{(i)} - \tilde{y}_0^{(i)}| + \frac{1}{\Gamma(\alpha)}\left(\sum_{j=0}^{k}|d_j^{k+1}|\left|f(t_j,y_j) - f(t_j,\tilde{y}_j)\right|\right.$$
$$\left. + |d_{k+1}^{k+1}|\,|f(t_{k+1},y_{k+1}) - f(t_{k+1},\tilde{y}_{k+1})|\right)$$
$$\le C_1\|y_0 - \tilde{y}_0\|_\infty + L\frac{|\mu|}{\Gamma(\alpha)}\left(\sum_{j=0}^{k}|d_j^{k+1}|\left|y_j - \tilde{y}_j\right| + |d_{k+1}^{k+1}|\,|y_{k+1} - \tilde{y}_{k+1}|\right).$$

By Lemma 1 one can get

$$|y_{k+1} - \tilde{y}_{k+1}| \le C_1\|y_0 - \tilde{y}_0\|_\infty + L\frac{|\mu|}{\Gamma(\alpha)}\left(C_{1,\alpha}T^\alpha \max_{0\le j\le k}\left|y_j - \tilde{y}_j\right| + C_{2,\alpha}T^\alpha\,|y_{k+1} - \tilde{y}_{k+1}|\right),$$

which implies that

$$|y_{k+1}-\tilde{y}_{k+1}| \leq \frac{1}{1-(|\mu|C_{2,\alpha}T^{\alpha})/\Gamma(\alpha)}\left(C_1\|y_0-\tilde{y}_0\|_\infty + \frac{|\mu|}{\Gamma(\alpha)}C_{1,\alpha}T^{\alpha}\max_{0\leq j\leq k}|y_j-\tilde{y}_j|\right).$$

Now for sufficiently small T, one can complete the proof by using the mathematical induction (27) and by choosing constant $C_{\alpha,T}$ sufficiently large. □

Theorem 4. *Assume that y_j $(j=1,2,\ldots,k)$ are the solutions of the algorithm II (14) and (19). Then, for sufficiently small $T>0$, the algorithm II is stable.*

Proof. The proof is similar to the proof of Theorem 3. □

5. Linear stability analysis

Consider the following test problem to investigate stability region of the presented methods:

$$ {}_0^C D_t^{\alpha} y(t) = \lambda y(t), \quad y(t_0) = y_0, \quad 0<\alpha<1. \tag{28}$$

The new method (11) gives the following iteration formula for solving (28):

$$y_{k+1} = y_0 + \frac{1}{\Gamma(\alpha)}\sum_{i=0}^{k+1}\frac{h^{\alpha}}{12\alpha(\alpha+1)(\alpha+2)(\alpha+3)(\alpha+4)}\tilde{b}_j^{k+1}\lambda y_j \tag{29}$$

Denoting $z=\lambda h^{\alpha}$, we get

$$z = 12\Gamma(\alpha+5)\frac{y_{k+1}-y_0}{\sum_{i=0}^{k+1}\tilde{b}_j^{k+1}y_j}. \tag{30}$$

Let $y_j=\xi^j$, then by assuming $\xi=e^{i\theta}$ with $0\leq\theta\leq 2\pi$ we get the following stability region for the scheme (11)

$$S=\left\{ z:\ z = 12\Gamma(\alpha+5)\frac{\xi^{k+1}-\xi^0}{\sum_{j=0}^{k+1}\tilde{d}_{k+1-j}^{k+1}\xi^j} \right\}. \tag{31}$$

The stability region of the algorithm II (14) and (19) can be achieved in a quite similar way. The stability region of the numerical algorithm I is obtained in Figs. (1) and (2) by choosing k=2000 and of the numerical algorithm II are shown in Figs. (3) and (4) by choosing k=500. The stability region in Figs. (1) and (3) are inside of the boundary and it is outside of the boundary in Figs. (2) and (4).

6. Numerical results

To check the numerical errors between the exact and the numerical solution, numerical experiments are carried out in this section. The presented examples have exact solutions and also have been solved by other numerical methods from literature. This allows one to compare the numerical results obtained by the presented schemes with the analytical solutions or those obtained by other methods.

Example 1. *Consider the following fractional differential equation*

$$ {}_0^C D_t^{\alpha} y(t) = \mu y(t) + g(t), \quad y(0) = 0, \quad y'(0) = 0. \tag{32} $$

where

$$ \mu = -1, \quad g(t) = \frac{\Gamma(5)}{\Gamma(5-\alpha)} t^{4-\alpha} + t^4. $$

The exact solution is $y(t) = t^4$. At the time $t = 1$ for different step sizes h and different α, the approximate solutions for the given equation are obtained by using presented algorithms and the method reported in Ref. [31]. The absolute error of presented algorithms and of the method reported in Ref. [31] are shown in Tables 1–3. These Tables show that the proposed novel schemes are valid methods in solving fractional differential equation.

Example 2. *Consider the differential equation of fractional order*

$$ {}_0^C D_t^{\alpha}\, y(t) = \frac{2}{\Gamma(3-\alpha)} t^{2-\alpha} - \frac{1}{\Gamma(2-\alpha)} t^{1-\alpha} - y(t) + t^2 - t, \quad y(0) = 0. \tag{33} $$

The exact solution to this initial value problem is $y(t) = t^2 - t$. The absolute errors of schemes given in Diethelm et el. [32], Deng and Li [31], Li et al. [30] and the presented new schemes are shown in Tables (4)–(6) and they are compared for different values of h and α at the time $t = 1$. For brevity, we use E_1 to denote improved algorithm I and E_2 denotes improved algorithm II of [30]. Tables 4–6 show that the absolute errors of the new presented methods are improved significantly in compared with the literature.

Example 3. *Consider the following fractional differential equation*

$$ {}_0^C D_t^{\alpha}\, y(t) = \frac{24}{\Gamma(5-\alpha)} t^{4-\alpha} - \frac{3}{\Gamma(4-\alpha)} t^{3-\alpha} - \frac{1}{2} t^3 - y(t) + t^4, \quad y(0) = 0. \tag{34} $$

The exact solution is $y(t) = t^4 - \frac{1}{2}t^3$. *Table 7 shows the absolute errors of the presented schemes and the method reported in Ref. [27] at the time* $t = 1$. *From this Table it is observed that the error of presented method is decreased significantly.*

7. Conclusion

This paper provides two high order numerical schemes with theoretically proved convergence order of 4 and 5 for solving FDEs. The properties of the Caputo derivative are used to reduce the FDEs into a Volterra integral equation. After dividing total domain into a set of grid points, the piecewise Lagrange interpolation polynomial of degree three and degree four are utilized to approximate unknown functions. The stability and error estimate of the methods are investigated. Moreover, graphical illustrations for stability region of the schemes are derived. The obtained solutions with the presented schemes demonstrate that the schemes give a more accurate approximation and superior than the numerical results obtained using other schemes. In the future, we shall try to follow this idea to construct higher order schemes for solving nonlinear FDEs.

Appendix A

The idea of solving y_1, y_2 and y_3 form (6) is as follows. For simplicity, we assume that $f(t,y) = \mu y + g(t)$ for understanding the idea of the numerical method. We have the following linear system of equations, from (6),

$$ \begin{cases} y_1 = \dfrac{1}{\gamma_1}\left[h(t_1) + \dfrac{1}{\Gamma(\alpha)} \displaystyle\sum_{\substack{j=0 \\ j\neq 1}}^{3} d_j^1 \left(\mu y_j + g(t_j)\right) + \dfrac{1}{\Gamma(\alpha)} d_1^1 g(t_1)\right], & (35) \\ y_2 = \dfrac{1}{\gamma_2}\left[h(t_2) + \dfrac{1}{\Gamma(\alpha)} \displaystyle\sum_{\substack{j=0 \\ j\neq 2}}^{3} d_j^2 \left(\mu y_j + g(t_j)\right) + \dfrac{1}{\Gamma(\alpha)} d_2^2 g(t_2)\right], & (36) \\ y_3 = \dfrac{1}{\gamma_3}\left[h(t_3) + \dfrac{1}{\Gamma(\alpha)} \displaystyle\sum_{j=0}^{2} d_j^3 \left(\mu y_j + g(t_j)\right) + \dfrac{1}{\Gamma(\alpha)} d_3^3 g(t_3)\right], & (37) \end{cases} $$

in which $\gamma_j = 1 - \frac{\mu d_{k+1}^{k+1}}{\Gamma(\alpha)}$, $(j = 1,2,3)$. Putting Eq. (35) into Eq. (36) yields

$$y_2 = \frac{1}{H_1^2}\left[\frac{1}{\gamma_2}\left[Y_{01}^2 h(t_1) + h(t_2)\right] + \frac{1}{H_2^2}\left[Y_0^2 F(0) + Y_1^2 g(t_1) + Y_2^2 g(t_2) + Y_3^2\left(\mu y_3 + g(t_3)\right)\right]\right], \tag{38}$$

in which

$$H_1^2 = 1 - \frac{\mu B d_2^1}{H_2^2}, \quad H_2^2 = C\gamma_2\,\Gamma(\alpha), \quad B = \mu d_1^2, \quad C = \gamma_1\Gamma(\alpha), \quad Y_{01}^2 = \frac{B}{C},$$
$$Y_0^2 = Bd_0^1 + Cd_0^2, \quad Y_1^2 = Bd_1^1 + Cd_1^2, \quad Y_2^2 = Bd_2^1 + Cd_2^2, \quad Y_3^2 = Bd_3^1 + Cd_3^2$$

Substituting (35) and (38) to (37) leads to

$$y_3 = \frac{1}{H_1^3}\left[\frac{1}{\gamma_3}\left(\frac{Y_{01}^3}{CH_2^3}h(t_1) + \frac{D_3}{H_2^3}h(t_2) + h(t_3)\right) + \frac{1}{H_3^3}\left[Y_0^3 F(0) + Y_1^3 g(t_1) + Y_2^3 g(t_2) + Y_3^3 g(t_3)\right]\right], \tag{39}$$

in which

$$D_1 = H_1^2 H_2^2, \quad D_2 = \mu d_1^3, \quad D_3 = \mu\left(D_2 d_2^1 + d_2^3 C\right), \quad H_1^3 = 1 - \frac{\mu\left(D_1 D_2 d_3^1 + \mu D_3 Y_3^2\right)}{H_3^3},$$
$$H_2^3 = CH_1^2\gamma_2\Gamma(\alpha), \quad H_3^3 = \Gamma(\alpha)D_1\gamma_3 C, \quad Y_{01}^3 = \Gamma(\alpha)H_1^2\gamma_2 CD_2 + D_3 B,$$
$$Y_0^3 = D_1\left(D_2 d_0^1 + Cd_0^3\right) + D_3 Y_0^2, \quad Y_1^3 = D_1\left(D_2 d_1^1 + Cd_1^3\right) + D_3 Y_1^2,$$
$$Y_2^3 = (D_1 + \mu Y_2^2)(D_3/\mu), \quad Y_3^3 = D_1\left(D_2 d_3^1 + Cd_3^3\right) + D_3 Y_3^2,$$

Now, firstly one can calculate y_3 from given initial conditions and known function $g(t)$. Then y_2 and y_1 can be calculated by (38) and (35), respectively.

Appendix B

We have the following linear system of equations, from (14),

$$\begin{cases} y_1 = \dfrac{1}{\gamma_1}\left[h(t_1) + \dfrac{1}{\Gamma(\alpha)}\displaystyle\sum_{\substack{j=0\\ j\neq 1}}^{4} b_j^1\left(Ay_j + g(t_j)\right) + \dfrac{1}{\Gamma(\alpha)} b_1^1 g(t_1)\right], & (40)\\ y_2 = \dfrac{1}{\gamma_2}\left[h(t_2) + \dfrac{1}{\Gamma(\alpha)}\displaystyle\sum_{\substack{j=0\\ j\neq 2}}^{4} b_j^2\left(Ay_j + g(t_j)\right) + \dfrac{1}{\Gamma(\alpha)} b_2^2 g(t_2)\right], & (41)\\ y_3 = \dfrac{1}{\gamma_3}\left[h(t_3) + \dfrac{1}{\Gamma(\alpha)}\displaystyle\sum_{\substack{j=0\\ j\neq 3}}^{4} b_j^3\left(Ay_j + g(t_j)\right) + \dfrac{1}{\Gamma(\alpha)} b_3^3 g(t_3)\right], & (42)\\ y_4 = \dfrac{1}{\gamma_4}\left[h(t_4) + \dfrac{1}{\Gamma(\alpha)}\displaystyle\sum_{j=0}^{3} b_j^4\left(Ay_j + g(t_j)\right) + \dfrac{1}{\Gamma(\alpha)} b_4^4 g(t_4)\right], & (43) \end{cases}$$

in which $\gamma_j = 1 - \frac{\mu a_{k+1}^{k+1}}{\Gamma(\alpha)}$, $(j = 1,2,3,4)$. Putting Eq. (43) into Eq. (42) gives

$$\begin{aligned} y_3 = &\frac{1}{H_1^3}\left[\frac{1}{\gamma_3}\left(h(t_3) + Y_{01}^3 h(t_4)\right) + \frac{1}{H_2^3}\left[Y_0^3 F(0) + Y_1^3\left(\mu y_1 + g(t_1)\right) + Y_2^3\left(\mu y_2 + g(t_2)\right)\right.\right.\\ &\left.\left. + Y_3^3 g(t_3) + Y_4^3 g(t_4)\right]\right], \end{aligned} \tag{44}$$

in which

$$\begin{aligned} &H_1^3 = 1 - \frac{\mu B^3 b_3^4}{H_2^3}, \quad H_2^2 = C\gamma_3\,\Gamma(\alpha), \quad B^3 = \mu b_4^3, \quad C = \gamma_4\,\Gamma(\alpha), \quad Y_{01}^3 = \frac{B^3}{C},\\ &Y_0^3 = B^3 b_0^4 + C b_0^3, \quad Y_1^3 = B^3 b_1^4 + C b_1^3, \quad Y_2^3 = B^3 b_2^4 + C b_2^3, \quad Y_3^3 = B^3 b_3^4 + C b_3^3,\\ &Y_4^3 = B^3 b_4^4 + C b_4^3 \end{aligned}$$

Inserting (43) and (44) into (41) gives

$$\begin{aligned} y_2 = &\frac{1}{H_1^2}\left[\frac{1}{\gamma_2}\left(h(t_2) + \frac{D_2}{H_2^2}h(t_3) + \frac{Y_{01}^2}{H_2^2}h(t_4)\right) + \frac{1}{H_3^2}\left[Y_0^2 F(0) + Y_1^2\left(\mu y_1 + g(t_1)\right) + Y_2^2 g(t_2)\right.\right.\\ &\left.\left. + Y_3^2 g(t_3) + Y_4^2 g(t_4)\right]\right], \end{aligned} \tag{45}$$

in which

$$\begin{aligned} &D_1 = H_1^3 H_2^3, \quad D_2 = \mu\left(C b_3^2 + B^2 b_3^4\right), \quad B^2 = \mu b_4^2, \quad H_2^2 = C H_1^3 \gamma_3 \Gamma(\alpha),\\ &H_1^2 = 1 - \frac{\mu\left(D_1 B^2 b_2^4 + \mu D_2 Y_2^3\right)}{H_3^2}, \quad H_3^2 = \Gamma(\alpha) C D_1 \gamma_2, \quad Y_{01}^2 = \Gamma(\alpha) B^2 H_1^3 \gamma_3 + D_2 Y_{01}^3, \end{aligned}$$

$$Y_0^2 = D_1\left(B^2 b_0^4 + C b_0^2\right) + D_2 Y_0^3, \quad Y_1^2 = D_1\left(B^2 b_1^4 + C b_1^2\right) + D_2 Y_1^3,$$
$$Y_2^2 = D_1\left(B^2 b_2^4 + C b_2^2\right) + D_2 Y_2^3, \quad Y_3^2 = \frac{D_2}{\mu}(D_1 + \mu Y_3^3), \quad Y_4^2 = D_1\left(B^2 b_4^4 + C b_4^2\right) + D_2 Y_4^3.$$

Finally we have, by substituting (43), (44) and (45) into (40)

$$y_1 = \frac{1}{H_1^1}\left[\frac{h(t_1)}{\gamma_1} + \frac{1}{H_2^1}\left(Y_{01}^1 h(t_2) + Y_{11}^1 h(t_3) + Y_{21}^1 h(t_4)\right) + \frac{1}{H_3^1}\left[Y_0^1 F(0) + Y_1^1 g(t_1) + Y_2^1 g(t_2)\right.\right.$$
$$\left.\left. + Y_3^1 g(t_3) + Y_4^1 g(t_4)\right]\right], \tag{46}$$

in which

$$B^1 = \mu b_4^1, \quad E_1 = \mu\left(E_7 b_2^4 + \mu b_3^4 Y_2^3\right), \quad E_2 = E_7 b_2^1 + \mu b_3^1 Y_2^3, \quad E_3 = E_5 \frac{b_3^4}{b_3^1},$$
$$E_4 = E_6 \frac{H_3^2 H_1^3}{H_2^2}, \quad E_5 = \mu H_1^2 H_3^2 b_3^1, \quad E_6 = H_1^2 H_2^2 H_2^3, \quad E_7 = H_1^3 H_2^3, \quad E_8 = H_1^2 H_3^2,$$

$$H_1^1 = \frac{\mu}{H_3^1}\left[B^1 E_4 b_1^4 + \mu E_8 Y_1^3\left(C b_3^1 + B^1 b_3^4\right) + Y_1^2\left(\mu C E_2 + B^1 E_1\right)\right]$$
$$H_2^1 = H_3^1 \frac{\gamma_2 \gamma_3 H_2^2}{H_3^2}, \quad H_3^1 = \Gamma(\alpha) C E_7 E_8 \gamma_1,$$

$$Y_{01}^1 = \gamma_3 H_2^2\left(\mu C E_2 + B^1 E_1\right), \quad Y_{11}^1 = \mu\left[E_6 \gamma_2\left(C b_3^1 + B^1 b_3^4\right) + D_3 \gamma_3\left(C E_2 + E_1 b_4^1\right)\right],$$

$$Y_{21}^1 = E_6 \gamma_2\left[\Gamma(\alpha) B^1 \gamma_3 H_1^3 + \mu Y_{01}^3\left(C b_3^1 + B^1 b_3^4\right)\right] + \mu \gamma_3 Y_{01}^2\left[E_7\left(C b_2^1 + B^1 b_2^4\right)\right.$$
$$\left. + \mu Y_2^3\left(C b_3^1 + B^1 b_3^4\right)\right], \quad Y_2^1 = \left(E_8 + \mu Y_2^2\right)\left(E_2 C + E_1 b_4^1\right)$$

$$Y_0^1 = C\left(E_4 b_0^1 + E_5 Y_0^3 + \mu E_2 Y_0^2\right) + B^1\left(E_4 b_0^4 + E_3 Y_0^3 + E_1 Y_0^2\right),$$

$$Y_1^1 = C\left(E_4 b_1^1 + E_5 Y_1^3 + \mu E_2 Y_1^2\right) + B^1\left(E_4 b_1^4 + E_3 Y_1^3 + E_1 Y_1^2\right),$$

$$Y_3^1 = C\left(E_4 b_3^1 + E_5 Y_3^3 + \mu E_2 Y_3^2\right) + B^1\left(E_4 b_3^4 + E_3 Y_3^3 + E_1 Y_3^2\right),$$

$$Y_4^1 = C\left(E_4 b_4^1 + E_5 Y_4^3 + \mu E_2 Y_4^2\right) + B^1\left(E_4 b_4^4 + E_3 Y_4^3 + E_1 Y_4^2\right),$$

Now, firstly from (46) one can calculate y_1 by given initial conditions and known function $g(t)$. Then y_2, y_3 and y_4 can be calculated by (45), (44) and (43), respectively.

References

[1] R. Simpson, A. Jaques, H. Nuñez, C. Ramirez, A. Almonacid, Fractional calculus as a mathematical tool to improve the modeling of mass transfer phenomena in food processing, Food Engineering Reviews 5 (1) (2013) 45–55.

[2] K. A. Gepreel, T. A. Nofal, Optimal homotopy analysis method for nonlinear partial fractional differential equations, Mathematical Sciences 9 (1) (2015) 47–55.

[3] M. A. Herzallah, K. A. Gepreel, Approximate solution to the time–space fractional cubic nonlinear schrodinger equation, Applied Mathematical Modelling 36 (11) (2012) 5678–5685.

[4] K. A. Gepreel, T. A. Nofal, F. M. Alotaibi, Numerical solutions for the time and space fractional nonlinear partial differential equations, Journal of Applied Mathematics 2013.

[5] H. Sun, Z.-z. Sun, G.-h. Gao, Some high order difference schemes for the space and time fractional bloch–torrey equations, Applied Mathematics and Computation 281 (2016) 356–380.

[6] M. Roohi, M. P. Aghababa, A. R. Haghighi, Switching adaptive controllers to control fractional-order complex systems with unknown structure and input nonlinearities, Complexity 21 (2) (2015) 211–223.

[7] H. Qu, X. Liu, A numerical method for solving fractional differential equations by using neural network, Advances in Mathematical Physics 2015.

[8] B. Ahmad, S. K. Ntouyas, A. Alsaedi, A study of nonlinear fractional differential equations of arbitrary order with riemann-liouville type multistrip boundary conditions, Mathematical Problems in Engineering 2013.

[9] Z. Jiao, Y. Chen, I. Podlubny, Distributed-order dynamic systems: stability, simulation, applications and perspectives. springerbriefs in electrical and computer engineering/springerbriefs in control, automation and robotics (2012).

[10] Y. Xiaozhong, W. Lifei, S. Shuzhen, Z. Xue, A universal difference method for time-space fractional black-scholes equation, Advances in Difference Equations 2016 (1) (2016) 1–14.

[11] C. Guo, S. Fang, Stability and approximate analytic solutions of the fractional lotka-volterra equations for three competitors, Advances in Difference Equations 2016 (1) (2016) 219.

[12] E. Ahmed, A. El-Sayed, H. A. El-Saka, Equilibrium points, stability and numerical solutions of fractional-order predator–prey and rabies models, Journal of Mathematical Analysis and Applications 325 (1) (2007) 542–553.

[13] A. Arafa, S. Rida, M. Khalil, A fractional-order model of hiv infection with drug therapy effect, Journal of the Egyptian Mathematical Society 22 (3) (2014) 538–543.

[14] L. K. B. Kuroda, A. V. Gomes, R. Tavoni, P. F. de Arruda Mancera, N. Varalta, R. de Figueiredo Camargo, Unexpected behavior of caputo fractional derivative, Computational and Applied Mathematics (2016) 1–11.

[15] F. A. Rihan, Numerical modeling of fractional-order biological systems, in: Abstract and Applied Analysis, Vol. 2013, Hindawi Publishing Corporation, 2013.

[16] F. Rihan, S. Lakshmanan, A. Hashish, R. Rakkiyappan, E. Ahmed, Fractional-order delayed predator–prey systems with holling type-ii functional response, Nonlinear Dynamics 80 (1-2) (2015) 777–789.

[17] R. Damor, S. Kumar, A. Shukla, Numerical simulation of fractional bioheat equation in hyperthermia treatment, Journal of Mechanics in Medicine and Biology 14 (02) (2014) 1450018.

[18] K. Kumar, R. K. Pandey, S. Sharma, Comparative study of three numerical schemes for fractional integro-differential equations, Journal of Computational and Applied Mathematics.

[19] K. M. Owolabi, Robust and adaptive techniques for numerical simulation of nonlinear partial differential equations of fractional order, Communications in Nonlinear Science and Numerical Simulation 44 (2017) 304–317.

[20] S. Bushnaq, S. Momani, Y. Zhou, A reproducing kernel hilbert space method for solving integro-differential equations of fractional order, Journal of Optimization Theory and Applications 156 (1) (2013) 96–105.

[21] S. Chakraverty, D. Behera, Dynamic responses of fractionally damped mechanical system using homotopy perturbation method, Alexandria Engineering Journal 52 (3) (2013) 557–562.

[22] Z. M. Odibat, S. Momani, An algorithm for the numerical solution of differential equations of fractional order, Journal of Applied Mathematics & Informatics 26 (1_2) (2008) 15–27.

[23] P. Kumar, O. P. Agrawal, An approximate method for numerical solution of fractional differential equations, Signal Processing 86 (10) (2006) 2602–2610.

[24] J. Cao, C. Xu, A high order schema for the numerical solution of the fractional ordinary differential equations, Journal of Computational Physics 238 (2013) 154–168.

[25] K. Diethelm, An algorithm for the numerical solution of differential equations of fractional order, Electronic transactions on numerical analysis 5 (1) (1997) 1–6.

[26] Y. Yan, K. Pal, N. J. Ford, Higher order numerical methods for solving fractional differential equations, BIT Numerical Mathematics 54 (2) (2014) 555–584.

[27] K. Pal, F. Liu, Y. Yan, Numerical solutions of fractional differential equations by extrapolation, in: I. Dimov, I. Faragó, L. Vulkov (Eds.), Finite Difference Methods,Theory and Applications, Springer International Publishing, 2015, pp. 299–306.

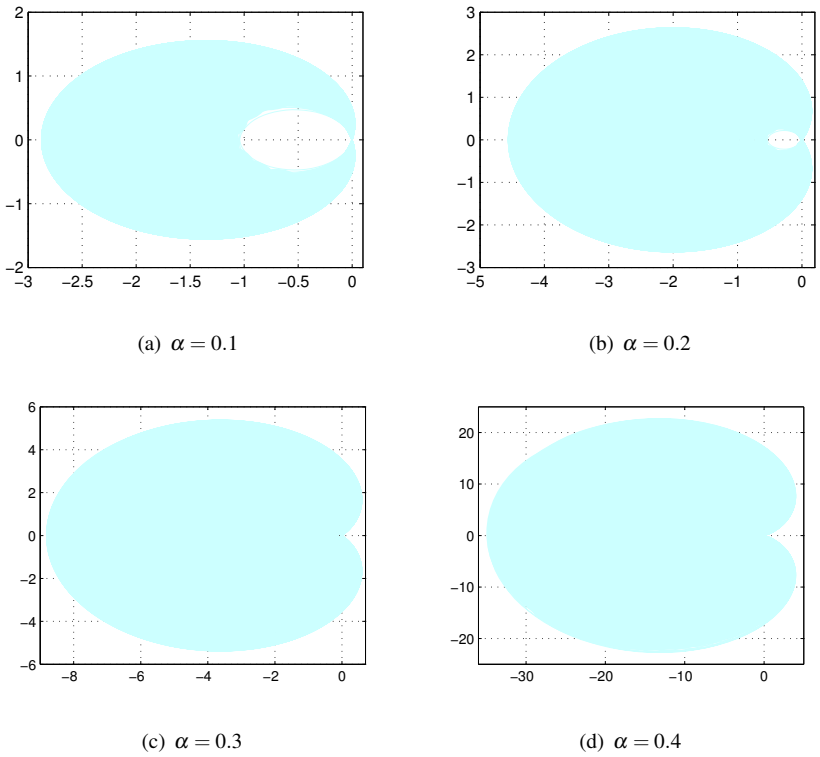
[28] M. S. Asl, M. Javidi, Novel algorithms to estimate nonlinear FDEs: Applied to fractional order nutrient-phytoplankton–zooplankton system, Journal of Computational and Applied Mathematics.

[29] M. S. Asl, M. Javidi, An improved pc scheme for nonlinear fractional differential equations: Error and stability analysis, Journal of Computational and Applied Mathematics 324 (2017) 101–117.

[30] C. Li, A. Chen, J. Ye, Numerical approaches to fractional calculus and fractional ordinary differential equation, Journal of Computational Physics 230 (9) (2011) 3352–3368.

[31] W. Deng, C. Li, Numerical schemes for fractional ordinary differential equations, in: P. Miidla (Ed.), Numerical Modelling, InTech, Rijeka, 2012, Ch. 16, pp. 355–374.

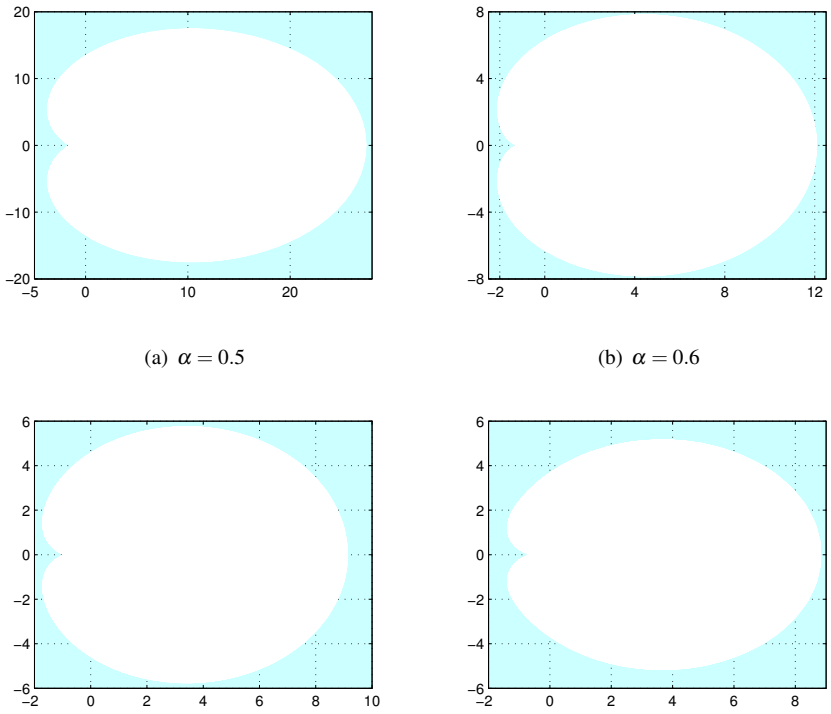
[32] K. Diethelm, N. J. Ford, A. D. Freed, Detailed error analysis for a fractional adams method, Numerical algorithms 36 (1) (2004) 31–52.



(a) $\alpha = 0.1$ (b) $\alpha = 0.2$

(c) $\alpha = 0.3$ (d) $\alpha = 0.4$

Figure 1: Stability region of the numerical algorithm I.



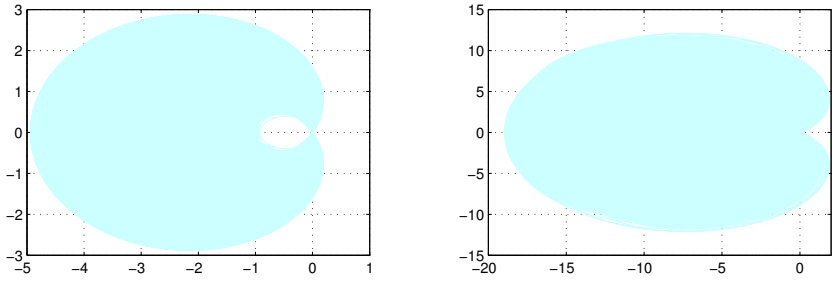
(a) $\alpha = 0.5$ (b) $\alpha = 0.6$

(c) $\alpha = 0.7$ (d) $\alpha = 0.8$

Figure 2: Stability region of the numerical algorithm I.

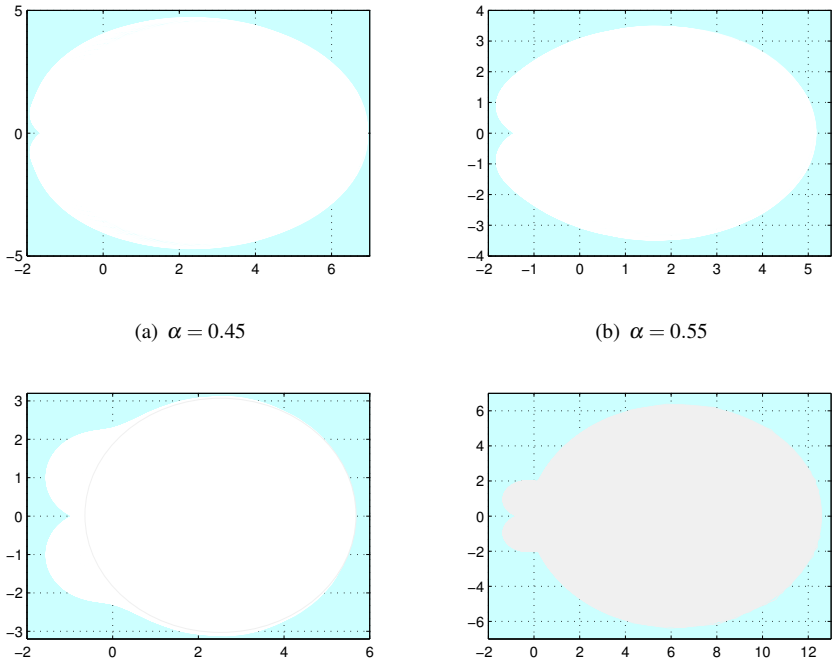
Table 1: The absolute errors of the presented algorithm I (E_I), algorithm II (E_{II}) and numerical method of [31] for (32).

	$\alpha = 0.1$			$\alpha = 0.5$		
h	E_{II}	E_I	[31]	E_{II}	E_I	[31]
1/10	1.3548e-07	1.38e-05	3.64e-01	1.2028e-06	2.42e-05	3.55e-02
1/20	4.7060e-09	9.46e-07	1.70e-01	4.4641e-08	1.57e-06	8.79e-03
1/40	1.9210e-10	6.35e-08	7.13e-02	1.7177e-09	1.00e-07	2.16e-03
1/80	3.3420e-11	4.18e-09	2.88e-02	6.6297e-11	6.37e-09	5.31e-04



(a) $\alpha = 0.15$ (b) $\alpha = 0.25$

Figure 3: Stability region of the numerical algorithm II.



(a) $\alpha = 0.45$ (b) $\alpha = 0.55$

(c) $\alpha = 0.75$ (d) $\alpha = 0.85$

Figure 4: Stability region of the numerical algorithm II.

Table 2: The absolute errors of the presented algorithm I (E_I), algorithm II (E_{II}) and numerical method of [31] for (32).

	$\alpha = 0.9$			$\alpha = 1.25$		
h	E_{II}	E_I	[31]	E_{II}	E_I	[31]
1/10	5.5858e-07	7.95e-06	1.07e-02	3.8819e-07	2.97e-05	8.48e-03
1/20	2.7279e-08	5.70e-07	2.31e-03	3.8522e-08	2.56e-06	2.03e-03
1/40	1.3292e-09	3.96e-08	5.21e-04	2.2108e-09	2.10e-07	5.00e-04
1/80	6.0469e-11	2.70e-09	1.22e-04	1.3614e-10	1.67e-08	1.24e-04

Table 3: The absolute errors of the presented algorithm I (E_I), algorithm II (E_{II}) and numerical method of [31] for (32).

	$\alpha = 1.5$			$\alpha = 1.85$		
h	E_{II}	E_I	[31]	E_{II}	E_I	[31]
1/10	5.4931e-06	6.86e-05	8.58e-03	1.6634e-05	6.80e-05	9.04e-03
1/20	3.5763e-07	6.93e-06	2.12e-03	1.5899e-06	8.64e-06	2.25e-03
1/40	2.9070e-08	6.60e-07	5.28e-04	1.6571e-07	1.04e-06	5.63e-04
1/80	2.5398e-09	6.11e-08	1.32e-04	1.7950e-08	1.21e-07	1.41e-04

Table 4: Absolute errors of the present presented algorithm I (E_I), algorithm II (E_{II}) and the numerical methods of Deng and Li [31] and Diethelm et el. [32], with $\alpha = 0.1$ for (33).

h	E_{II}	E_I	[31]	[32]
1/10	3.4944e-06	8.19e-06	0.104	0.103
1/20	9.9500e-07	1.90e-06	4.66e-02	4.95e-02
1/40	2.6402e-07	4.73e-07	1.87e-02	2.09e-02
1/80	6.9544e-08	1.21e-07	7.39e-03	8.65e-03

Table 5: Absolute errors of the presented algorithm I (E_I), algorithm II (E_{II}) and numerical methods of Diethelm et el. [32] and Li et al. [30] with $\alpha = 0.3$ for (33).

h	E_{II}	E_I	[32]	[30]	E_1 [30]	E_2 [30]
1/10	6.0368e-05	1.19e-04	3.14e-02	2.25e-02	2.1e-03	1.61e-02
1/20	1.9179e-05	3.23e-05	1.10e-02	1.24e-02	4.53e-04	5.9e-03
1/40	5.7565e-06	9.26e-06	3.91e-03	6.1e-03	8.61e-05	2.1e-03
1/80	1.7261e-06	2.72e-06	1.42e-03	2.9e-03	1.19e-05	7.52e-04

Table 6: Absolute errors of the presented algorithm I (E_I), algorithm II (E_{II}) and numerical methods of Deng and Li [31], Diethelm et el. [32] and Li et al. [30]. with $\alpha = 0.5$ for (33).

h	E_{II}	E_I	[31]	[32]	[30]	E_1 [30]	E_2 [30]
1/10	3.6057e-04	6.08e-04	9.27e-03	1.44e-02	4.1e-03	1.8e-03	3.6e-03
1/20	1.2875e-04	1.96e-04	2.29e-03	4.52e-03	3.1e-03	7.2e-04	1.1e-03
1/40	4.4935e-05	6.60e-05	5.87e-04	1.46e-03	1.8e-03	2.8e-04	3.4e-04
1/80	1.5699e-05	2.27e-05	1.56e-04	4.81e-04	1.0e-03	1.0e-04	1.07e-04

Table 7: Absolute errors of the present presented algorithm I (E_I), algorithm II (E_{II}) and the numerical methods of [27], with $\alpha = 0.3$ for (34).

h	E_{II}	E_I	[27]
1/10	8.8773e-07	2.6193e-05	1.4571e-04
1/20	3.0045e-08	1.7205e-06	2.3118e-05
1/40	1.0533e-09	1.1167e-07	3.6127e-06
1/80	1.2938e-10	7.2496e-09	5.6030e-07