



# A 2nd-Order Numerical Scheme for Fractional Ordinary Differential Equation Systems

W. Li and S. Wang<sup>(✉)</sup>

Department of Mathematics and Statistics, Curtin University, GPO Box U1987,  
Perth 6845, Australia  
{Wen.Li, Song.Wang}@curtin.edu.au

**Abstract.** We propose a new numerical method for fractional ordinary differential equation systems based on a judiciously chosen quadrature point. The proposed method is efficient and easy to implement. We show that the convergence order of the method is 2. Numerical results are presented to demonstrate that the computed rates of convergence confirm our theoretical findings.

## 1 Introduction

We consider the following system of fractional ordinary differential equations:

$${}_0D_t^\alpha x(t) = f(t, x(t)), \quad t \in (0, T], \quad \text{satisfying } x(0) = x^0, \quad (1)$$

where  $x(t) = (x_1(t), x_2(t), \dots, x_n(t))^T \in \mathbb{R}^n$  for a positive integer  $n$ ,  $T > 0$  is a fixed constant,  $f: \mathbb{R}^{n+1} \mapsto \mathbb{R}^n$  a given mapping,  $x^0 \in \mathbb{R}^n$  a given initial condition, and  ${}_0D_t^\alpha x(t) = ({}_0D_t^{\alpha_1} x_1(t), {}_0D_t^{\alpha_2} x_2(t), \dots, {}_0D_t^{\alpha_n} x_n(t))^T$  for  $\alpha_i \in (0, 1)$  with  ${}_0D_t^{\alpha_i} x_i(t)$  denoting the following Caputo's  $\alpha_i$ -th derivative

$${}_0D_t^{\alpha_i} x_i(t) = \frac{1}{\Gamma(1 - \alpha_i)} \int_0^t \frac{x_i'(\tau)}{(t - \tau)^{\alpha_i}} d\tau$$

for  $t > 0$  and  $i = 1, 2, \dots, n$ , where  $\Gamma(\cdot)$  denotes the Gamma function.

In the open literature, there are a number of methods for solving (1). Adomian decomposition method [3, 5, 9], variational iteration method [11, 12], differential transform method [2] and homotopy analysis method [10, 14] have been used for the problem. Recently, we proposed a new one-step numerical integration scheme for (1) [8]. This method is easy to implement and computationally inexpensive. In this paper, we will show that the global error of the method is of order  $\mathcal{O}(h^2)$ , where  $h$  denotes the maximal mesh size to be defined.

The rest of the paper is organized as the follows. In Sect. 2, we first transform Eq. (1) into an equivalent Volterra integral equation, we then propose an approximation of Volterra integral equation based on a Taylor expansion. An error analysis of the approximation is also presented. In Sect. 3, we propose an algorithm for implementing the approximate equation and analyse its convergence. In Sect. 4, numerical examples are presented. Section 5 concludes the paper.

## 2 Approximation

We first rewrite (1) as the following Volterra integral equation:

$$x_i(t) = x_i^0 + \frac{1}{\Gamma(\alpha_i)} \int_0^t (t - \tau)^{\alpha_i - 1} f_i(\tau, x(\tau)) d\tau, \quad (2)$$

for  $t \in (0, T]$  and  $i = 1, 2, \dots, n$ . It has been proven [1, 4, 6] that solving Eq. (1) is equivalent to solving (2). In this section, we will develop a numerical method based on a Taylor expansion to approximate (2) and estimate the approximation error.

Let  $N$  be a given positive integer. We divide  $(0, T)$  into  $N$  sub-intervals with mesh points  $t_i = ih$  for  $i = 0, 1, \dots, N$ , where  $h = T/N$ . Thus, we have

$$\begin{aligned} x_i(t_j) &= x_i^0 + \frac{1}{\Gamma(\alpha_i)} \int_0^{jh} (jh - \tau)^{\alpha_i - 1} f_i(\tau, x(\tau)) d\tau \\ &= x_i^0 + \frac{1}{\Gamma(\alpha_i)} \sum_{k=1}^j \int_{(k-1)h}^{kh} (jh - \tau)^{\alpha_i - 1} f_i(\tau, x(\tau)) d\tau. \end{aligned} \quad (3)$$

To approximate the integral on the RHS of (3), we assume that  $f_i(t, x(t))$  is twice continuously differentiable with respect to both  $t$  and  $x$ . For any  $k$ , we expand  $f_i(\tau, x(\tau))$  at any point in  $((k-1)h, kh)$ , denoted as  $\tau_{jk}^i$ , into

$$f_i(\tau, x(\tau)) = f_i(\tau_{jk}^i, x(\tau_{jk}^i)) + K_{jk}^i(\tau - \tau_{jk}^i) + c_{jk}^i(\tau - \tau_{jk}^i)^2, \quad (4)$$

where  $c_{jk}^i$  is the coefficient of the reminder of the expansion and

$$K_{jk}^i = \left. \frac{\partial f_i}{\partial \tau} \right|_{(\tau_{jk}^i, x(\tau_{jk}^i))} + \sum_{l=1}^n \left. \frac{\partial f_i}{\partial x_l} \right|_{(\tau_{jk}^i, x(\tau_{jk}^i))} \frac{\partial x_l}{\partial \tau} \Big|_{(\tau_{jk}^i)}.$$

Therefore, replacing  $f_i(\tau, x(\tau))$  in (3) with the RHS of (4) and by direct integration we have

$$\begin{aligned} &\frac{1}{\Gamma(\alpha_i)} \int_{(k-1)h}^{kh} (jh - \tau)^{\alpha_i - 1} f_i(\tau, x(\tau)) d\tau \\ &= \frac{1}{\Gamma(\alpha_i)} \int_{(k-1)h}^{kh} (jh - \tau)^{\alpha_i - 1} [f_i(\tau_{jk}^i, x(\tau_{jk}^i)) + K_{jk}^i(\tau - \tau_{jk}^i)] d\tau + R_{jk}^i \\ &= \frac{h^{\alpha_i}}{\Gamma(\alpha_i + 1)} f_i(\tau_{jk}^i, x(\tau_{jk}^i)) [(j - k + 1)^{\alpha_i} - (j - k)^{\alpha_i}] \\ &\quad + \frac{K_{jk}^i}{\Gamma(\alpha_i)} \int_{(k-1)h}^{kh} (jh - \tau)^{\alpha_i - 1} (\tau - \tau_{jk}^i) d\tau + R_{jk}^i, \end{aligned} \quad (5)$$

where  $R_{jk}^i = \frac{1}{\Gamma(\alpha_i)} \int_{(k-1)h}^{kh} (jh - \tau)^{\alpha_i - 1} c_{jk}^i (\tau - \tau_{jk}^i)^2 d\tau.$

From (5) it is clear that  $\tau_{jk}^i$  should be chosen such that the integral term in (5) becomes zero so that the truncation error is  $R_{jk}^i$ . The choice of  $\tau_{jk}^i$  is given in the following theorem.

**Theorem 1.** For any feasible  $j$  and  $k$ , the unique solution to

$$\int_{(k-1)h}^{kh} (jh - \tau)^{\alpha_i - 1} (\tau - \tau_{jk}^i) d\tau = 0$$

is

$$\tau_{jk}^i = h \frac{[(j-k+1)^{\alpha_i+1} - (j-k)^{\alpha_i+1}] + (\alpha_i+1)[(j-k+1)^{\alpha_i}(k-1) - (j-k)^{\alpha_i}k]}{(\alpha_i+1)[(j-k+1)^{\alpha_i} - (j-k)^{\alpha_i}]}.$$
 (6)

Furthermore,  $(k-1)h < \tau_{jk}^i < kh$ .

*Proof.* See the proof of Theorem 2.1 in [7].

Substituting the expression for  $\tau_{jk}^i$  in (6) into (5) and combining the resulting expression with (3), we have the following representation for  $x_i(t_j)$ .

$$x_i(t_j) = x_i^0 + \frac{h^{\alpha_i}}{\Gamma(\alpha_i+1)} \sum_{k=1}^j f_i(\tau_{jk}^i, x(\tau_{jk}^i)) [(j-k+1)^{\alpha_i} - (j-k)^{\alpha_i}] + R_j^i, \quad (7)$$

for  $j = 1, 2, \dots, N$ , where  $\tau_{jk}^i$  is given in (6) for  $k = 1, 2, \dots, j$  and  $R_j^i = \sum_{k=1}^j R_{jk}^i$ . Omitting  $R_j^i$  in (7), we have an approximation to (3) with the truncation error  $R_j^i$ . An upper bound for  $R_j^i$  is given in the following theorem.

**Theorem 2.** If  $f(t, x)$  is twice continuously differentiable in  $t$  and  $x$ , then we have  $|R_j^i| \leq Ch^2$ , where  $C$  denotes a positive constant independent of  $h$ .

*Proof.* See the proof of Theorem 2.2 in [7].

From (7) it is clear that to compute  $x_i(t_j)$ , we need to calculate  $f_i(\tau_{jk}^i, x(\tau_{jk}^i))$ . However,  $x(\tau_{jk}^i)$  is not available directly from the scheme. Thus, approximations to  $x(\tau_{jk}^i)$  need to be determined. In the next section, we propose a single step numerical scheme for implementing (7) when the remainder  $R_j^i$  is omitted.

### 3 Algorithm and Its Convergence

For any  $j$  and  $k$  satisfying  $1 \leq k \leq j \leq N$ , since  $\tau_{jk}^i \in (t_{k-1}, t_k)$  by Theorem 1, we use the following linear interpolation to approximate  $x_i(\tau_{jk}^i)$ :

$$x(\tau_{jk}^i) = x(t_{k-1}) + \rho_{jk}^i (x(t_k) - x(t_{k-1})) + \mathcal{O}(h^2) E_n, \quad (8)$$

where  $\rho_{jk}^i := \frac{\tau_{jk}^i - t_{k-1}}{h} \in (0, 1)$  and  $E_n = (1, 1, \dots, 1)^\top \in \mathbb{R}^n$ . Using (8), we approximate  $f_i(\tau_{jk}^i, x(\tau_{jk}^i))$  as follows.

$$f_i(\tau_{jk}^i, x(\tau_{jk}^i)) = f_i(\tau_{jk}^i, x(t_{k-1}) + \rho_{jk}^i (x(t_k) - x(t_{k-1}))) + \mathcal{O}(h^2). \quad (9)$$

Replacing  $f_i(\tau_{jk}^i, x(\tau_{jk}^i))$  in (7) with the RHS of (9), we have

$$x_i(t_j) = x_i^0 + h_{\alpha_i} \sum_{k=1}^j \left[ f_i(\tau_{jk}^i, x(t_{k-1}) + \rho_{jk}^i(x(t_k) - x(t_{k-1}))) \cdot ((j-k+1)^{\alpha_i} - (j-k)^{\alpha_i}) \right] + \mathcal{O}(h^2) \quad (10)$$

for  $j = 1, 2, \dots, N$ , where  $h_{\alpha_i} = \frac{h^{\alpha_i}}{\Gamma(\alpha_i+1)}$  and  $\tau_{jk}^i$  is defined in (6). Clearly, (10) defines a time-stepping scheme for (2) if we omit the term  $\mathcal{O}(h^2)$ .

The above scheme is implicit as it is a nonlinear system in  $x(t_j)$ . We now define an explicit single step scheme by further approximating the  $j$ th term in the sum in (10) by the following Taylor expansion:

$$\begin{aligned} & f_i(\tau_{jj}^i, x(t_{j-1}) + \rho_{jj}^i(x(t_j) - x(t_{j-1}))) \\ &= f_i(\tau_{jj}^i, x(t_{j-1})) + \sum_{l=1}^n \left. \frac{\partial f_i}{\partial x_l} \right|_{(\tau_{jj}^i, x(t_{j-1}))} (\rho_{jj}^i(x_l(t_j) - x_l(t_{j-1}))) + \mathcal{O}(h^2). \end{aligned} \quad (11)$$

Thus, combining (11) and (10) yields

$$\begin{aligned} x_i(t_j) &= x_i^0 + h_{\alpha_i} \sum_{k=1}^{j-1} \left[ f_i(\tau_{jk}^i, x(t_{k-1}) + \rho_{jk}^i(x(t_k) - x(t_{k-1}))) \cdot ((j-k+1)^{\alpha_i} - (j-k)^{\alpha_i}) \right] \\ &\quad + h_{\alpha_i} f_i(\tau_{jj}^i, x(t_{j-1})) \\ &\quad + h_{\alpha_i} \sum_{l=1}^n \left[ \left. \frac{\partial f_i}{\partial x_l} \right|_{(\tau_{jj}^i, x(t_{j-1}))} (\rho_{jj}^i(x_l(t_j) - x_l(t_{j-1}))) \right] + \mathcal{O}(h^2). \end{aligned} \quad (12)$$

Let  $x^j := (x_1^j, x_2^j, \dots, x_n^j)^\top$  for  $j = 0, 1, \dots, N$  and omitting the truncation error terms of order  $\mathcal{O}(h^2)$  in (12), we define the following single step time-stepping scheme for approximating (2):

$$\begin{aligned} x_i^j &= x_i^0 + h_{\alpha_i} \sum_{k=1}^{j-1} \left[ f_i(\tau_{jk}^i, x^{k-1} + \rho_{jk}^i(x^k - x^{k-1})) \cdot ((j-k+1)^{\alpha_i} - (j-k)^{\alpha_i}) \right] \\ &\quad + h_{\alpha_i} f_i(\tau_{jj}^i, x^{j-1}) + h_{\alpha_i} \sum_{l=1}^n \left[ \left. \frac{\partial f_i}{\partial x_l} \right|_{(\tau_{jj}^i, x^{j-1})} (\rho_{jj}^i(x_l^j - x_l^{j-1})) \right] \end{aligned} \quad (13)$$

Re-organising (13), we have the following linear system for  $x^j$ :

$$B^j x^j = C^j, \quad j = 1, 2, \dots, N. \quad (14)$$

where  $B^j$  is the  $n \times n$  matrix given by

$$B^j = \begin{pmatrix} 1 - b_{11}^j & -b_{12}^j & \dots & -b_{1n}^j \\ -b_{21}^j & 1 - b_{22}^j & \dots & -b_{2n}^j \\ \vdots & \vdots & \ddots & \vdots \\ -b_{n1}^j & -b_{n2}^j & \dots & 1 - b_{nn}^j \end{pmatrix} \quad (15)$$

with

$$b_{il}^j = \rho_{jj}^i h_{\alpha_i} \left. \frac{\partial f_i}{\partial x_l} \right|_{(\tau_{jj}^i, x^{j-1})} \tag{16}$$

for  $i = 1, 2, \dots, n, l = 1, 2, \dots, n$  and  $C^j = (c_1^j, c_2^j, \dots, c_n^j)^\top$  with

$$\begin{aligned} c_i^j = & x_i^0 + h_{\alpha_i} \sum_{k=1}^{j-1} \left[ f_i(\tau_{jk}^i, x^{k-1}) + \rho_{jk}^i (x^k - x^{k-1}) ((j-k+1)^{\alpha_i} - (j-k)^{\alpha_i}) \right] \\ & + h_{\alpha_i} f_i(\tau_{jj}^i, x^{j-1}) - \sum_{l=1}^n x_l^{j-1} b_{il}^j. \end{aligned} \tag{17}$$

It is clear that to calculate  $x^j$ , we need to solve the system of equations (14)–(17). It has been shown in [8] that (14)–(17) is uniquely solvable when  $h$  is sufficiently small.

For a given initial condition  $x^0$ , using the above results, we propose the following algorithm for solving (3) numerically.

**Algorithm A**

1. For a given positive integer  $N$ , let  $t_j = jh$  for  $j = 0, 1, \dots, N$ , where  $h = T/N$ .
2. Calculate  $x^j$  for  $j = 1, \dots, N$  using (14)–(17).

Using a linear interpolation and Taylor’s theorem, we are able to prove in the following theorem that, for any  $j = 1, 2, \dots, N$ ,  $x^j$  generated by the above algorithm converges to the solution of (2) at the rate  $\mathcal{O}(h^2)$  when  $h \rightarrow 0^+$ .

**Theorem 3.** *Let  $x(t_j)$  and  $x^j$  be respectively the solution to (3) and the sequence generated by Algorithm A. If  $f(t, x)$  is twice continuously differentiable in  $t$  and  $x$ , then there exists an  $\bar{h} > 0$  such that when  $h < \bar{h}$*

$$\|x(t_j) - x^j\|_\infty \leq Ch^2, \quad j = 1, 2, \dots, N. \tag{18}$$

*Proof.* In what follows, we let  $C$  denote a generic positive constant, independent of  $h$ . We now prove this theorem by mathematical induction.

When  $j = 1$ , from (12) we have

$$x_i(t_1) = x_i^0 + h_{\alpha_i} \left[ f_i(\tau_{11}^i, x^0) + \sum_{l=1}^n \left. \frac{\partial f_i}{\partial x_l} \right|_{(\tau_{11}^i, x^0)} (\rho_{11}^i (x_l(t_1) - x_l^0)) \right] + \mathcal{O}(h^2) \tag{19}$$

Re-organising (19) and using the definitions for  $B^j$  and  $C^j$ , we have

$$B^1 x(t_1) = C^1 + \mathcal{O}(h^2) E_n. \tag{20}$$

Solving (20) yields

$$x(t_1) = (B^1)^{-1} C^1 + \mathcal{O}(h^2) (B^1)^{-1} E_n.$$

From (14)–(17), we see that

$$x^1 = (B^1)^{-1}C^1.$$

Therefore,

$$\|x(t_1) - x^1\|_\infty = \mathcal{O}(h^2)\|(B^1)^{-1}E_n\|_\infty \leq Ch^2\|(B^1)^{-1}\|_\infty.$$

It has been proven [8] that  $B^j, j = 1, 2, \dots, N$  satisfies, when  $h < \bar{h}$ ,

$$\sigma^j := \min_{1 \leq i \leq n} \left\{ |b_{ii}^j| - \sum_{j=1, j \neq i}^n |b_{ij}^j| \right\} \geq \beta > 0 \tag{21}$$

for a constant  $\beta$ , independent of  $h$ , where  $\bar{h} = \min_{1 \leq i \leq n} \left( \frac{\Gamma(\alpha_i+1)}{nM} \right)^{\frac{1}{\alpha_i}}$  and  $M = \max_{\substack{1 \leq i \leq n \\ 1 \leq l \leq n}} \left| \frac{\partial f_i}{\partial x_l} \right|$ . Thus, using [13] and (21), we have

$$\|(B^1)^{-1}\|_\infty \leq \frac{1}{\sigma^1} \leq \frac{1}{\beta}.$$

Therefore, we have

$$\|x(t_1) - x^1\|_\infty \leq Ch^2.$$

When  $i \geq 2$  and  $h \leq \bar{h}$ , we assume that

$$\|x(t_j) - x^j\|_\infty \leq Ch^2, \quad 1 \leq j \leq i - 1. \tag{22}$$

We now show that  $\|x(t_j) - x^j\|_\infty \leq Ch^2$  for  $1 \leq j \leq i$ .

Note that (12) can be re-written in the following form:

$$x_i(t_j) = x_i^0 + A_i^j + D_i^j + \mathcal{O}(h^2), \tag{23}$$

where

$$A_i^j = h_{\alpha_i} \sum_{k=1}^{j-1} \left[ f_i(\tau_{jk}^i, x(t_{k-1}) + \rho_{jk}^i(x(t_k) - x(t_{k-1}))) ((j-k+1)^{\alpha_i} - (j-k)^{\alpha_i}) \right], \tag{24}$$

$$D_i^j = h_{\alpha_i} f_i(\tau_{jj}^i, x(t_{j-1})) + h_{\alpha_i} \sum_{l=1}^n \left[ \frac{\partial f_i}{\partial x_l} \Big|_{(\tau_{jj}^i, x(t_{j-1}))} (\rho_{jj}^i(x_l(t_j) - x_l(t_{j-1}))) \right]. \tag{25}$$

Similarly, (13) can be re-written as follows.

$$x_i^j = x_i^0 + \tilde{A}_i^j + \tilde{D}_i^j, \tag{26}$$

where

$$\tilde{A}_i^j = h_{\alpha_i} \sum_{k=1}^{j-1} \left[ f_i(\tau_{jk}^i, x^{k-1} + \rho_{jk}^i(x^k - x^{k-1})) ((j-k+1)^{\alpha_i} - (j-k)^{\alpha_i}) \right], \tag{27}$$

$$\tilde{D}_i^j = h_{\alpha_i} f_i(\tau_{jj}^i, x^{j-1}) + h_{\alpha_i} \sum_{l=1}^n \left[ \frac{\partial f_i}{\partial x_l} \Big|_{(\tau_{jj}^i, x^{j-1})} (\rho_{jj}^i (x_l^j - x_l^{j-1})) \right]. \quad (28)$$

Subtracting (26) from (23) gives

$$x_i(t_j) - x_i^j = (A_i^j - \tilde{A}_i^j) + (D_i^j - \tilde{D}_i^j) + \mathcal{O}(h^2). \quad (29)$$

Let us first estimate  $D_i^j - \tilde{D}_i^j$ . From (25) and (28), we have

$$\begin{aligned} D_i^j - \tilde{D}_i^j &= \left[ h_{\alpha_i} f_i(\tau_{jj}^i, x(t_{j-1})) + h_{\alpha_i} \sum_{l=1}^n \left[ \frac{\partial f_i}{\partial x_l} \Big|_{(\tau_{jj}^i, x(t_{j-1}))} (\rho_{jj}^i (x_l(t_j) - x_l(t_{j-1}))) \right] \right] \\ &\quad - \left[ h_{\alpha_i} f_i(\tau_{jj}^i, x^{j-1}) + h_{\alpha_i} \sum_{l=1}^n \left[ \frac{\partial f_i}{\partial x_l} \Big|_{(\tau_{jj}^i, x^{j-1})} (\rho_{jj}^i (x_l^j - x_l^{j-1})) \right] \right] \\ &= h_{\alpha_i} [f_i(\tau_{jj}^i, x(t_{j-1})) - f_i(\tau_{jj}^i, x^{j-1})] \\ &\quad + h_{\alpha_i} \rho_{jj}^i \left[ \sum_{l=1}^n \left[ \frac{\partial f_i}{\partial x_l} \Big|_{(\tau_{jj}^i, x(t_{j-1}))} (x_l(t_j)) - \frac{\partial f_i}{\partial x_l} \Big|_{(\tau_{jj}^i, x^{j-1})} (x_l^j) \right] \right] \\ &\quad - h_{\alpha_i} \rho_{jj}^i \left[ \sum_{l=1}^n \left[ \frac{\partial f_i}{\partial x_l} \Big|_{(\tau_{jj}^i, x(t_{j-1}))} (x_l(t_{j-1})) - \frac{\partial f_i}{\partial x_l} \Big|_{(\tau_{jj}^i, x^{j-1})} (x_l^{j-1}) \right] \right]. \quad (30) \end{aligned}$$

Since  $f_i$  is twice continuously differentiable, using a Taylor expansion we get

$$\frac{\partial f_i}{\partial x_l} \Big|_{(\tau_{jj}^i, x(t_{j-1}))} = \frac{\partial f_i}{\partial x_l} \Big|_{(\tau_{jj}^i, x^{j-1})} + r_j^i, \quad (31)$$

where

$$r_j^i = \sum_{p=1}^n \frac{\partial^2 f_i}{\partial x_l \partial x_p} \Big|_{(\tau_{jj}^i, \xi)} (x_p(t_{j-1}) - x_p^{t-1}),$$

where  $\xi = x(t_{j-1}) + \theta(x(t_{j-1}) - x^{j-1})$  with  $\theta \in (0, 1)$ . From the assumption (22) we have  $r_j^i = \mathcal{O}(h^2)$ . Similarly, since  $f$  is twice differentiable, using (22) it is easy to show  $f_i(\tau_{jj}^i, x_i(t_{i-1})) - f_i(\tau_{jj}^i, x_i^{j-1}) = \mathcal{O}(h^2)$ .

Using (31) and the above estimates we have, from (30),

$$\begin{aligned} D_i^j - \tilde{D}_i^j &= h_{\alpha_i} \mathcal{O}(h^2) + h_{\alpha_i} \rho_{jj}^i \left[ \sum_{l=1}^n \left[ \frac{\partial f_i}{\partial x_l} \Big|_{(\tau_{jj}^i, x^{j-1})} (x_l(t_j) - x_l^j + x_l(t_j) r_j^i) \right] \right] \\ &\quad - h_{\alpha_i} \rho_{jj}^i \left[ \sum_{l=1}^n \left[ \frac{\partial f_i}{\partial x_l} \Big|_{(\tau_{jj}^i, x^{j-1})} (x_l(t_{j-1}) - x_l^{j-1} + x_l(t_{j-1}) r_j^i) \right] \right] \\ &= h_{\alpha_i} \rho_{jj}^i \left[ \sum_{l=1}^n \left[ \frac{\partial f_i}{\partial x_l} \Big|_{(\tau_{jj}^i, x^{j-1})} (x_l(t_j) - x_l^j) \right] \right] \\ &\quad + h_{\alpha_i} \rho_{jj}^i \left[ \sum_{l=1}^n \left[ \frac{\partial f_i}{\partial x_l} \Big|_{(\tau_{jj}^i, x^{j-1})} (x_l^{j-1} - x_l(t_{j-1})) \right] \right] + \mathcal{O}(h^{2+\alpha_i}) \\ &= h_{\alpha_i} \rho_{jj}^i \left[ \sum_{l=1}^n \left[ \frac{\partial f_i}{\partial x_l} \Big|_{(\tau_{jj}^i, x^{j-1})} (x_l(t_j) - x_l^j) \right] \right] + \mathcal{O}(h^{2+\alpha_i}), \end{aligned}$$

since  $h_{\alpha_i} = h^{\alpha_i}/\Gamma(1 + \alpha_i)$  and (22). Thus, from the above expression and (29), we get

$$x_i(t_j) - x_i^j = (A_i^j - \tilde{A}_i^j) + h_{\alpha_i} \rho_{jj}^i \left[ \sum_{l=1}^n \left[ \frac{\partial f_i}{\partial x_l} \Big|_{(\tau_{jj}^i, x^{j-1})} (x_l(t_j) - x_l^j) \right] \right] + \mathcal{O}(h^2).$$

Re-organising the above equation gives

$$B^j (x(t_j) - x^j) = A^j + \mathcal{O}(h^2)E_n, \quad j = 1, 2, \dots, N,$$

where  $B^j$  is defined in (15)–(16) and  $A^j = (A_1^j - \tilde{A}_1^j, A_2^j - \tilde{A}_2^j, \dots, A_n^j - \tilde{A}_n^j)^\top$ . From this equation we have

$$x(t_j) - x^j = (B^j)^{-1} (A^j + \mathcal{O}(h^2)E_n), \quad j = 1, 2, \dots, N.$$

Thus, we have

$$\|x(t_j) - x^j\|_\infty = \|(B^j)^{-1} (A^j + \mathcal{O}(h^2)E_n)\|_\infty \leq \|(B^j)^{-1}\|_\infty (\|A^j\|_\infty + \mathcal{O}(h^2))$$

for  $j = 1, 2, \dots, N$ . Using [13] and (21), we have

$$\|(B^j)^{-1}\|_\infty \leq \frac{1}{\sigma^j} \leq \frac{1}{\beta}.$$

Therefore, we obtain

$$\|x(t_j) - x^j\|_\infty \leq \frac{1}{\beta} \|A^j\|_\infty + Ch^2. \quad (32)$$

We now examine  $\|A^j\|_\infty = \|A_i^j - \tilde{A}_i^j\|_\infty$ . For notational simplicity, we let  $x^{jk} = x^{k-1} + \rho_{jk}^i (x^k - x^{k-1})$  and  $x(t_{jk}) = x(t_{k-1}) + \rho_{jk}^i (x(t_k) - x(t_{k-1}))$ . From (24) and (27), we have

$$\begin{aligned} |A_i^j - \tilde{A}_i^j| &\leq h_{\alpha_i} \sum_{k=1}^{j-1} \left| [f_i(\tau_{jk}^i, x(t_{jk})) - f_i(\tau_{jk}^i, x^{jk})] [(j-k+1)^{\alpha_i} - (j-k)^{\alpha_i}] \right| \\ &= h_{\alpha_i} \sum_{k=1}^{j-1} |f_i(\tau_{jk}^i, x(t_{jk})) - f_i(\tau_{jk}^i, x^{jk})| [(j-k+1)^{\alpha_i} - (j-k)^{\alpha_i}], \end{aligned} \quad (33)$$

since  $z^{\alpha_i}$  is an increasing function of  $z$  for  $\alpha_i \in (0, 1)$ . Because  $f$  is twice continuously differentiable, we have, using a Taylor expansion,

$$\begin{aligned} |f_i(\tau_{jk}^i, x(t_{jk})) - f_i(\tau_{jk}^i, x^{jk})| &\leq C \|x(t_{jk}) - x^{jk}\|_\infty \\ &= C \|[x(t_{k-1}) + \rho_{jk}^i (x(t_k) - x(t_{k-1}))] - [x^{k-1} + \rho_{jk}^i (x^k - x^{k-1})]\|_\infty \\ &= C \|[x(t_{k-1}) - x^{k-1}] + \rho_{jk}^i [x(t_k) - x^k] + \rho_{jk}^i [x^{k-1} - x(t_{k-1})]\|_\infty \\ &\leq C (\|x(t_{k-1}) - x^{k-1}\|_\infty + \|x(t_k) - x^k\|_\infty + \|x(t_{k-1}) - x^{k-1}\|_\infty), \end{aligned}$$



since  $\rho_{jk}^i \in (0, 1)$ . Thus, from Assumption (22), we have

$$|f_i(\tau_{jk}, x(t_{jk})) - f_i(\tau_{jk}, x^{jk})| \leq Ch^2.$$

Replacing  $|f_i(\tau_{ij}, x(t_{jk})) - f_i(\tau_{ij}, x^{jk})|$  in (33) with the above upper bound yields

$$\begin{aligned} |A_i^j - \tilde{A}_i^j| &\leq h_{\alpha_i} Ch^2 \sum_{k=1}^{j-1} [(j-k+1)^{\alpha_i} - (j-k)^{\alpha_i}] = \frac{h^{\alpha_i}}{\Gamma(\alpha_i+1)} Ch^2 (j^{\alpha_i} - 1) \\ &\leq \frac{h^{\alpha_i}}{\Gamma(\alpha_i+1)} Ch^2 N^{\alpha_i} = \frac{C}{\Gamma(\alpha_i+1)} h^2 (hN)^{\alpha_i} \\ &= \frac{C}{\Gamma(\alpha_i+1)} h^2 T^{\alpha_i} \leq Ch^2 \end{aligned}$$

for  $i = 1, 2, \dots, n$ . Thus, we have

$$\|A^j - \tilde{A}^j\|_{\infty} \leq Ch^2.$$

Combining the above error bound with (32), we have the estimate (18). Thus, the theorem is proved.

## 4 Numerical Results

In this section, we will use Algorithm A to solve two non-trivial examples. All the computations have been performed in double precision under Matlab environment on a PC with Intel Xeon 3.3 GHz CPU and 16 GB RAM.

**Example 1.** Consider the following system of fractional differential equations:

$$\begin{cases} {}_0D_t^{\alpha_1} x_1(t) = x_2(t), \\ {}_0D_t^{\alpha_2} x_2(t) = x_3(t), \\ {}_0D_t^{\alpha_3} x_3(t) = \frac{\Gamma(5)}{\Gamma(5-\alpha_1-\alpha_2-\alpha_3)} t^{4-(\alpha_1+\alpha_2+\alpha_3)}, t \in (0, 1], \\ x_1(0) = x_2(0) = x_3(0) = 0, \end{cases}$$

where  $\alpha_i \in (0, 1)$ ,  $i = 1, 2, 3$ . The exact solution is

$$x_1(t) = t^4, \quad x_2(t) = \frac{\Gamma(5)}{\Gamma(5-\alpha_1)} t^{4-\alpha_1}, \quad x_3(t) = \frac{\Gamma(5)}{\Gamma(5-\alpha_1-\alpha_2)} t^{4-(\alpha_1+\alpha_2)}.$$

We solve the problem using Algorithm A for various values of  $\alpha_i$ ,  $i = 1, 2, 3$  and  $h_k = 1/(2^k \times 10)$ ,  $k = 1, \dots, 6$ . The computed errors  $E_{h_k}^i = \max_{1 \leq j \leq 1/h_k} |x_i^j - x_i(t_j)|$  for the chosen values of  $\alpha_i$ 's are listed in Table 1. To estimate the rates of convergence, we calculate  $\log_2(E_{h_k}/E_{h_{k+1}})$  for  $k = 1, \dots, 5$  and the computed rates of convergence, as well as CPU times, are also listed in Table 1. From the results in Table 1 we see that our method has a 2nd-order convergence rate for all the chosen values of  $\alpha$ , as predicted by Theorem 3, indicating that our method is very robust in  $\alpha$ . The CPU time in Table 1 shows that our method is very efficient.

**Table 1.** Computed errors, convergence order and CPU time for Example 1.

$h$	$x_1$	Order	$x_2$	Order	$x_3$	Order	CPU (seconds)
$\alpha_1 = \alpha_2 = \alpha_3 = 0.9$							
1/20	0.0041	-	0.00296	-	0.0022		0.2496
1/40	0.0010	2.00	7.1406e-04	2.05	5.7828e-04	1.93	0.2652
1/80	2.4856e-04	2.01	1.7308e-04	2.04	1.4936e-04	1.95	0.3588
1/160	6.1488e-05	2.02	4.2162e-05	2.04	3.8314e-05	1.96	0.4212
1/320	1.5238e-05	2.01	1.0315e-05	2.03	9.7768e-06	1.97	0.5928
1/640	3.7822e-06	2.01	2.5328e-06	2.03	2.4845e-06	1.98	1.6848
$\alpha_1 = 0.8, \alpha_2 = 0.6, \alpha_3 = 0.4$							
1/20	0.0036	-	0.0021	-	0.0028	-	0.2184
1/40	9.1509e-04	1.98	5.4008e-04	1.96	7.0318e-04	1.99	0.2808
1/80	2.3067e-04	1.99	1.3723e-04	1.98	1.7557e-04	2.00	0.3588
1/160	5.7981e-05	1.99	3.4680e-05	1.98	4.3850e-05	2.00	0.3900
1/320	1.4547e-05	1.99	8.7325e-06	1.99	1.0954e-05	2.00	0.6864
1/640	3.6451e-06	2.00	2.1936e-06	1.99	2.7371e-06	2.00	1.6692
$\alpha_1 = 0.1, \alpha_2 = 0.3, \alpha_3 = 0.2$							
1/20	0.0018	-	9.6495e-04	-	0.00145	-	0.2652
1/40	4.9549e-04	1.86	2.6383e-04	1.87	3.7287e-04	1.96	0.3120
1/80	1.3425e-04	1.88	7.0393e-05	1.90	9.5368e-05	1.97	0.3432
1/160	3.5823e-05	1.91	1.8472e-05	1.93	2.4300e-05	1.97	0.4524
1/320	9.4522e-06	1.92	4.7897e-06	1.95	6.1738e-06	1.98	0.6552
1/640	2.4728e-06	1.93	1.2311e-06	1.96	1.5648e-06	1.98	1.5912

**Example 2.** Consider the following fractional differential equation used in [14].

$$\begin{cases} {}_0D_t^{\alpha_1} x_1(t) = x_1(t), \\ {}_0D_t^{\alpha_2} x_2(t) = 2x_1^2(t), \\ {}_0D_t^{\alpha_3} x_3(t) = 3x_1(t)x_2(t), \end{cases} t \in (0, 1]$$

$$x_1(0) = 1, x_2(0) = 1, x_3(0) = 0.$$

The exact solution when  $\alpha_1 = \alpha_2 = \alpha_3 = 1$  is

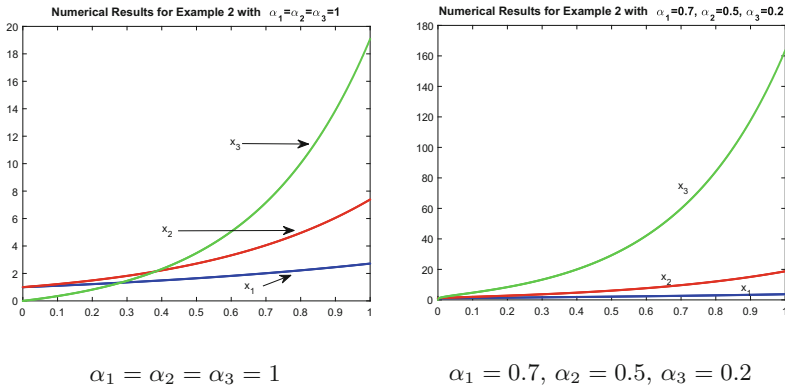
$$x_1(t) = e^t, \quad x_2(t) = e^{2t}, \quad x_3(t) = e^{3t} - 1.$$

It is solved using Algorithm A for various values of  $h$  and  $\alpha_i, i = 1, 2, 3$ . The computed errors and rates of convergence when  $\alpha_1 = \alpha_2 = \alpha_3 = 1$  are listed in Table 2 from which we see that the computed rates of convergence is  $\mathcal{O}(h^2)$ , confirming our theoretical result.

Since the exact solution to this Example 2 is unknown when  $\alpha_i < 1$  for any  $i$ , we are unable to compute the rates of convergence. Instead, we solve the problem

**Table 2.** Computed errors, convergence order and CPU time for Example 2.

$h$	$\alpha_1 = \alpha_2 = \alpha_3 = 1$						
	$x_1$	Order	$x_2$	Order	$x_3$	Order	CPU (seconds)
1/20	5.6658e-04	-	0.0026	-	0.0159	-	0.2496
1/40	1.4159e-04	2.00	7.1349e-04	1.87	0.0045	1.82	0.2808
1/80	3.5395e-05	2.00	1.8530e-04	1.94	0.0012	1.91	0.3120
1/160	8.8486e-06	2.00	4.7210e-05	1.97	3.1080e-04	1.95	0.3588
1/320	2.2121e-06	2.00	1.1914e-05	1.99	7.8895e-05	1.98	0.5616
1/640	5.5304e-07	2.00	2.9925e-06	1.99	1.9874e-05	1.99	0.8112



**Fig. 1.** Computed solutions of Example 2

for  $\alpha_1 = 0.7, \alpha_2 = 0.5, \alpha_3 = 0.2$  using  $h = 1/640$  and plot the computed solution, along with the solution for  $\alpha_i = 1$  for  $i = 1, 2, 3$ , in Fig. 1. From Fig. 1 we see that  $x_1, x_2$  and  $x_3$  from the fractional system grow much faster than those from the integer system.

## 5 Conclusion

We have proposed and analysed a 1-step numerical integration method for a system of fractional differential equations, based a superconvergent quadrature point we have derived recently. The proposed method is unconditionally stable and easy to implement. We have shown the method has a 2nd-order accuracy. Non-trivial examples have been solved by our method and the numerical results show that our method is 2nd-order accurate for all the chosen values of the fractional orders, demonstrating our method is very robust.

**Acknowledgment.** This work is partially supported by US Air Force Office of Scientific Research Project FA2386-15-1-4095.

## References

1. Diethelm, K., Ford, N.J.: Analysis of fractional differential equations. *J. Math. Anal. Appl.* **265**, 229–248 (2002)
2. Erturk, V.S., Momani, S.: Solving systems of fractional differential equations using differential transform method. *J. Comput. Appl. Math.* **215**, 142–151 (2008)
3. Gejji, V.D., Jafari, H.: Adomian decomposition: a tool for solving a system of fractional differential equations. *J. Math. Anal. Appl.* **301**(2), 508–518 (2005)
4. Guo, B., Pu, X., Huang, F.: *Fractional Partial Differential Equations and Their Numerical Solutions*. World Scientific, Singapore (2015)
5. Jafari, H., Gejji, V.D.: Solving a system of nonlinear fractional differential equation using adomain decomposition. *Appl. Math. Comput.* **196**, 644–651 (2006)
6. Kilbas, A.A., Marzan, S.A.: Cauchy problem for differential equation with caputo derivative. *Fract. Calc. Appl. Anal.* **7**(3), 297–321 (2004)
7. Li, W., Wang, S., Rehbock, V.: A 2nd-order one-point numerical integration scheme for fractional ordinary differential Equation. *Numer. Algebra Control Optim.* **7**(3), 273–287 (2017)
8. Li, W., Wang, S., Rehbock, V.: Numerical solution of fractional optimal control. *J. Optim. Theory Appl.* (2018). <https://doi.org/10.1007/s10957-018-1418-y>
9. Momani, S., Al-Khaled, K.: Numerical solutions for systems of fractional differential equations by the decomposition method. *Appl. Math. Comput.* **162**(3), 1351–1365 (2005)
10. Momani, S., Odibat, Z.: Homotopy perturbation method for nonlinear partial differential equations of fractional order. *Phys. Lett. A* **365**(5–6), 345–350 (2007)
11. Momani, S., Odibat, Z.: Numerical approach to differential equations of fractional order. *J. Comput. Appl. Math.* **207**, 96–110 (2007)
12. Odibat, Z., Momani, S.: Application of variational iteration method to nonlinear differential equations of fractional order. *Int. J. Nonlinear Sci. Numer. Simulat.* **1**(7), 15–27 (2006)
13. Varga, R.S.: On Diagonal dominance arguments for bounding  $\|A^{-1}\|_{\infty}$ . *Linear Algebra Appl.* **14**(3), 211–217 (1976)
14. Zurigat, M., Momani, S., Odibat, Z., Alawneh, A.: The homotopy analysis method for handling systems of fractional differential equations. *Appl. Math. Model.* **34**, 24–35 (2010)