

# Numerical Solution of System of Fractional Delay Differential Equations Using Polynomial Spline Functions

Mahmoud N. Sherif<sup>1,2</sup>

<sup>1</sup>Mathematics Department, Faculty of Science and Education, Taif University, Al-Khurmah Branch, Taif, Saudi Arabia

<sup>2</sup>Mathematics Department, Faculty of Science, Menoufia University, Shebeen ElKoom, Egypt  
Email: nouhmahmoud@yahoo.com

Received 2 November 2015; accepted 21 March 2016; published 24 March 2016

Copyright © 2016 by authors and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

---

## Abstract

The aim of this paper is to approximate the solution of system of fractional delay differential equations. Our technique relies on the use of suitable spline functions of polynomial form. We introduce the description of the proposed approximation method. The error analysis and stability of the method are theoretically investigated. Numerical example is given to illustrate the applicability, accuracy and stability of the proposed method.

## Keywords

Fractional Differential Equation, Spline Functions, Taylor Expansion, Stability

---

## 1. Introduction

Recently, the use of various types of spline function in the numerical treatment of ordinary differential equations [1]-[5] and delay differential equations [6]-[11] has been increasing. Many interesting applications in the area of mathematical biology, mathematical model of numerous engineering and physical phenomena have been studied [12] [13]. The fractional differential equation of the form

$$y^{(\alpha)}(x) = f(x, y(x)), \quad y(0) = y_0 \quad (1)$$

is studied by Kia Dithelm and N. J. Ford [14]. In [15] [16], the Adams-Bashforth-Moulton method is used to approximate solutions of the initial value problem (1). An alternative is the backward differentiation formula presented in [17] where the idea of this method is based on discretizing the differential operator in the fractional

differential Equation (1) by certain finite difference. Extrapolation principles in [7] are applied to improve the performance of the method presented in [17]. Kia Dithelm in [18] studied that a fast algorithm for the numerical solution of initial value problems of the form (1) in the sense of Caputo identifies and discusses potential problems in the development of generally applicable schemes. More recently, Lagrange multiplier method and the homotopy perturbation method are used to solve numerically multi-order fractional differential equation see [19]. Micul [20] considered the problem

$$\begin{aligned} y'(x) &= f_1(x, y, z), & y(x_0) &= y_0, \\ z'(x) &= f_2(x, y, z), & z(x_0) &= z_0, \end{aligned} \tag{2}$$

where  $f_1, f_2 \in C^r [0, 1] \times R \times R, (x, y, z) \in [0, 1] \times R \times R$ . They assume that the functions  $f_i^{(q)}, i = 1, 2$  and  $q = 0, 1, 2, \dots, r$  satisfy the Lipschitz condition of the form:

$$|f_i^{(q)}(x, y_1, z_1) - f_i^{(q)}(x, y_2, z_2)| \leq L_i \{|y_1 - y_2| + |z_1 - z_2|\},$$

with constant  $L_i$  for all  $(x, y_i, z_i) \in [a, b] \times R \times R$ .

An extension of the spline functions form defined in [19] for approximating the solution of system of ordinary differential equations is investigated, namely, for the system (2) with unique solution  $y = y(x), z = z(x)$  is considered. The spline functions  $S_\Delta(x)$  and  $\tilde{S}_\Delta(x)$  to approximate  $y = y(x), z = z(x)$  are defined in polynomial form as:

$$\begin{aligned} S_\Delta(x) &= S_k(x) = S_{k-1}(x_k) + \sum_{i=0}^r f_1^{(i)}(x_k, S_{k-1}(x_k), \tilde{S}_{k-1}(x_k)) \cdot \frac{(x-x_k)^{i+1}}{(i+1)!}, \\ \tilde{S}_\Delta(x) &= \tilde{S}_k(x) = \tilde{S}_{k-1}(x_k) + \sum_{i=0}^r f_2^{(i)}(x_k, S_{k-1}(x_k), \tilde{S}_{k-1}(x_k)) \cdot \frac{(x-x_k)^{i+1}}{(i+1)!}, \end{aligned}$$

for  $x \in [x_k, x_{k+1}], k = 0, 1, \dots, n-1, S_{-1}(x_0) = y_0, \tilde{S}_{-1}(x_0) = z_0$ .

Ramadan, M. A. obtained in [15] the solution of the first order delay differential equation of the form:

$$\begin{aligned} y'(x) &= f(x, y(x), y(g(x))), & a \leq x \leq b, \\ y(a) &= y_0, & y(x) = \Phi(x), x \in [a^*, a], \quad a^* < 0, a^* = \inf \{g(x) : x \in [a, b]\}, \end{aligned}$$

using the spline function of the polynomial form which defined as:

$$S_\Delta(x) = S_k(x) = S_{k-1}(x_k) + \sum_{i=0}^r M_k^{(i)} \cdot \frac{(x-x_k)^{i+1}}{(i+1)!}$$

where  $M_k^{(i)} = f^{(i)}(x_k, S_{k-1}(x_k), S_{k-1}(g(x_k)))$ , with  $S_{-1}(x_0) = y_0, S_{-1}(g(x_0)) = \Phi(g(x_0))$ .

Ramadan, Z. in [21] discussed the system of the initial value problem

$$\begin{aligned} y'''(x) &= f_1(x, y, y', z, z'), & y^{(i)}(x_0) &= y_0^{(i)}, \\ z'''(x) &= f_2(x, y, y', z, z'), & z^{(i)}(x_0) &= z_0^{(i)}, \end{aligned} \tag{3}$$

where  $f_1, f_2 \in C^r([0, 1] \times R^{(4)}), i = 0, 1, 2$ , his method was presented which uses polynomial spline to approximate the solutions of the system.

## 2. Description of the Proposed Spline Approximation Method

Consider the system of first order delay differential equations:

$$\begin{aligned} y^{(\alpha)}(x) &= f_1(x, y(x), z(x), y(g(x))), & a \leq x \leq b, \\ z^{(\alpha)}(x) &= f_2(x, y(x), z(x), z(g(x))), \\ y(x_0) &= y_0, & z(x_0) &= z_0, \\ y(x) &= \Phi_1(x), & z(x) &= \Phi_2(x), & \Phi(x) \in [a^*, a], & a^* = \inf \{g(x) : x \in [a, b]\}. \end{aligned} \tag{4}$$

The function  $g$  is called the delay function and it is assumed to be continuous on the interval  $[a, b]$  and to satisfy the inequality  $a^* \leq g(x) \leq x, x \in [a, b]$  and  $\Phi_1, \Phi_2 \in C[a^*, a]$ .

Suppose that  $f_1 : [a, b] \times R^3 \rightarrow R$  is continuous and satisfies Lipschitz condition

$$|f_1^{(\alpha)}(x, y_1, z_1, u_1) - f_1^{(\alpha)}(x, y_2, z_2, u_2)| \leq L_1 \{|y_1 - y_2| + |z_1 - z_2| + |u_1 - u_2|\} \tag{5}$$

and there exists a constant  $B_1$  such that

$$|u_1 - u_2| \leq B_1 \{|f_1^{(\alpha)}(x, y_1, z_1, u_1) - f_1^{(\alpha)}(x, y_2, z_2, u_2)|\}, \tag{6}$$

with  $L_1 B_1 < 1, \forall (x, y_1, z_1, u_1), (x, y_2, z_2, u_2) \in [a, b] \times R^3, u_i = y_i(g(x))$ .

Suppose also that  $f_2 : [a, b] \times R^3 \rightarrow R$  is continuous and satisfies the Lipschitz condition:

$$|f_2^{(\alpha)}(x, y_1, z_1, v_1) - f_2^{(\alpha)}(x, y_2, z_2, v_2)| \leq L_2 \{|y_1 - y_2| + |z_1 - z_2| + |v_1 - v_2|\} \tag{7}$$

and there exists a constant  $B_2$  such that

$$|v_1 - v_2| \leq B_2 \{|f_2^{(\alpha)}(x, y_1, z_1, v_1) - f_2^{(\alpha)}(x, y_2, z_2, v_2)|\}, \tag{8}$$

with  $L_2 B_2 < 1, \forall (x, y_1, z_1, v_1), (x, y_2, z_2, v_2) \in [a, b] \times R^3, v_i = z(g(x))$ .

These conditions assure the existence of unique solution  $y$  and  $z$  of system (4).

Let  $\Delta$  be a uniform partition to the interval  $[a, b]$  defined by the nodes

$$\Delta : a = x_0 < x_1 < \dots < x_k < x_{k+1} < \dots < x_n = b, x_n = x_0 + kh, h = \frac{b-a}{n} \text{ and } k = 0, 1, \dots, n-1.$$

Define the new form of system of fractional spline function  $S(x)$  and  $\tilde{S}(x)$  of polynomial form approximating the exact solution  $y$  and  $z$  by:

$$S_k(x) = S_{k-1}(x_k) + \sum_{i=0}^r M_k^{(i\alpha)} \frac{(x-x_k)^{(i+1)\alpha}}{\Gamma((i+1)\alpha+1)}, \tag{9}$$

$$\tilde{S}_k(x) = \tilde{S}_{k-1}(x_k) + \sum_{i=0}^r \tilde{M}_k^{(i\alpha)} \frac{(x-x_k)^{(i+1)\alpha}}{\Gamma((i+1)\alpha+1)}, \tag{10}$$

where

$$M_k^{(\alpha)} = f_1^{(\alpha)}(x, S_{k-1}(x_k), \tilde{S}_{k-1}(x_k), S_{k-1}(g(x_k))), \tag{11}$$

$$\tilde{M}_k^{(\alpha)} = f_2^{(\alpha)}(x, S_{k-1}(x_k), \tilde{S}_{k-1}(x_k), \tilde{S}_{k-1}(g(x_k))), \tag{12}$$

with

$$S_{-1}(x_0) = y_0, \tilde{S}_{-1}(x_0) = z_0, S_{-1}(g(x_0)) = \Phi_1(g(x_0)), \tilde{S}_{-1}(g(x_0)) = \Phi_2(g(x_0)).$$

Such that  $S_\Delta(x)$  and  $\tilde{S}_\Delta(x)$  exist and are unique.

### 3. Error Estimation and Convergence Analysis

To estimate the error of the approximate solution, we write the exact solution  $y(x)$  and  $z(x)$  in the following Taylor form [11]:

$$y(x) = \sum_{i=0}^r y_k^{(i\alpha)} \frac{(x-x_k)^{i\alpha}}{\Gamma(i\alpha+1)} + y^{((r+1)\alpha)}(\zeta_k) \frac{(x-x_k)^{(r+1)\alpha}}{\Gamma((r+1)\alpha+1)} \tag{13}$$

$$z(x) = \sum_{i=0}^r z_k^{(i\alpha)} \frac{(x-x_k)^{i\alpha}}{\Gamma(i\alpha+1)} + z^{((r+1)\alpha)}(\zeta_k) \frac{(x-x_k)^{(r+1)\alpha}}{\Gamma((r+1)\alpha+1)} \tag{14}$$

where  $\zeta_k \in (x_k, x_{k+1}), y_k = y(x_k)$  and  $z_k = z(x_k)$ .

Moreover, we denote to the estimated error of  $y(x)$  and  $z(x)$  at any point  $x \in [a, b]$  by:

$$e_\alpha(x) = |y(x) - S_k(x)|, \quad \tilde{e}_\alpha(x) = |\tilde{z}(x) - \tilde{S}_k(x)|$$

and at  $x_k$  denote to the error

$$e_{k,\alpha} = |y_k - S_{k-1,\alpha}(x_k)|, \quad \tilde{e}_{k,\alpha} = |z_k - \tilde{S}_{k-1,\alpha}(x_k)| \tag{15}$$

Define the modulus of continuity of  $\omega(y^{((r+1)\alpha)}(\zeta_k, h))$  and  $\omega(z^{((r+1)\alpha)}(\zeta_k, h))$  as follows:

$$\omega(y^{((r+1)\alpha)}(\zeta_k, h)) = \max_{\zeta_k \in [a,b]} \left| y^{((r+1)\alpha)}(\zeta_k + h) - y^{((r+1)\alpha)}(\zeta_k) \right|$$

and

$$\omega(z^{((r+1)\alpha)}(\zeta_k, h)) = \max_{\zeta_k \in [a,b]} \left| z^{((r+1)\alpha)}(\zeta_k + h) - z^{((r+1)\alpha)}(\zeta_k) \right|.$$

Next lemma gives an upper bound to the error.

**Lemma 1**

Let  $e_\alpha(x)$  and  $\tilde{e}_\alpha(x)$  are defined as in (15) then there exist constant  $d_1$  and  $d_2$  independent of  $h$  such that the following inequality:

$$e_\alpha(x) \leq (1 + d_1 h)e_k + d_1 h \tilde{e}_k + \omega(h) \frac{h^{(r+1)\alpha}}{\Gamma((r+1)\alpha + 1)}$$

$$\tilde{e}_\alpha(x) \leq (1 + d_2 h)\tilde{e}_k + d_2 h e_k + \omega(h) \frac{h^{(r+1)\alpha}}{\Gamma((r+1)\alpha + 1)}$$

holds for all  $x \in [a, b]$  where,  $d_1 = \sum_{i=0}^r \frac{d_0}{\Gamma((i+1)\alpha + 1)}$  and  $d_2 = \sum_{i=0}^r \frac{\tilde{d}_0}{\Gamma((i+1)\alpha + 1)}$ .

**Proof**

Using the Lipschitz condition, Taylor expansion, definition of error estimation and (15) we get, by dropping  $\alpha$  :

$$e(x) = |y(x) - S_k(x)|$$

$$= \left| (y_k - S_{k-1}(x_k)) + \sum_{i=1}^r y_k^{(i\alpha)} \frac{(x-x_k)^{i\alpha}}{\Gamma(i\alpha + 1)} - \sum_{i=0}^r M_k^{(i\alpha)} \frac{(x-x_k)^{(i+1)\alpha}}{\Gamma((i+1)\alpha + 1)} + y^{((r+1)\alpha)}(\zeta_k) \frac{(x-x_k)^{(r+1)\alpha}}{\Gamma((r+1)\alpha + 1)} \right|$$

$$= \left| (y_k - S_{k-1}(x_k)) + \sum_{i=0}^{r-1} y_k^{((i+1)\alpha)} \frac{(x-x_k)^{(i+1)\alpha}}{\Gamma((i+1)\alpha + 1)} - \sum_{i=0}^{r-1} M_k^{(i\alpha)} \frac{(x-x_k)^{(i+1)\alpha}}{\Gamma((i+1)\alpha + 1)} \right. \tag{16}$$

$$\left. + y^{((r+1)\alpha)}(\zeta_k) \frac{(x-x_k)^{(r+1)\alpha}}{\Gamma((r+1)\alpha + 1)} - M_k^{(r\alpha)} \frac{(x-x_k)^{(r+1)\alpha}}{\Gamma((r+1)\alpha + 1)} \right|$$

$$\leq e_k + \sum_{i=0}^{r-1} \left| y_k^{((i+1)\alpha)} - M_k^{(i\alpha)} \right| \frac{|(x-x_k)^{(i+1)\alpha}|}{\Gamma((i+1)\alpha + 1)} + \left| y^{((r+1)\alpha)}(\zeta_k) - M_k^{(r\alpha)} \right| \frac{|(x-x_k)^{(r+1)\alpha}|}{\Gamma((r+1)\alpha + 1)},$$

where

$$\left| y_k^{((i+1)\alpha)} - M_k^{(i\alpha)} \right| = \left| f_1^{(i\alpha)}(x_k, y_k, z_k, y_k(g(x_k))) - f_1^{(i\alpha)}(x_k, S_{k-1}(x_k), \tilde{S}_{k-1}(x_k), S_{k-1}(g(x_k))) \right|$$

$$\leq L_1 \left\{ |y_k - S_{k-1}(x_k)| + |z_k - \tilde{S}_{k-1}(x_k)| + |y_k(g(x_k)) - S_{k-1}(g(x_k))| \right\}$$

$$\leq L_1 \left\{ |y_k - S_{k-1}(x_k)| + |z_k - \tilde{S}_{k-1}(x_k)| \right\}$$

$$+ L_1 B_1 \left| f_1^{(i\alpha)}(x_k, y_k, z_k, y(g(x_k))) - f_1^{(i\alpha)}(x_k, S_{k-1}, \tilde{S}_{k-1}, S_{k-1}(g(x_k))) \right|$$

Therefore,

$$\begin{aligned} & \left| f_1^{(r\alpha)}(x_k, y_k, z_k, y(g(x_k))) - f_1^{(i\alpha)}(x_k, S_{k-1}, \tilde{S}_{k-1}, S_{k-1}(g(x_k))) \right| \\ & - L_1 B_1 \left| f_1^{(i\alpha)}(x_k, y_k, z_k, y(g(x_k))) - f_1^{(i\alpha)}(x_k, S_{k-1}, \tilde{S}_{k-1}, S_{k-1}(g(x_k))) \right| \\ & \leq L_1 \left\{ |y_k - S_{k-1}(x_k)| + |z_k - \tilde{S}_{k-1}(x_k)| \right\} \end{aligned}$$

Thus,

$$\begin{aligned} & \left| f_1^{(i\alpha)}(x_k, y_k, z_k, y(g(x_k))) - f_1^{(i\alpha)}(x_k, S_{k-1}, \tilde{S}_{k-1}, S_{k-1}(g(x_k))) \right| (1 - L_1 B_1) \\ & \leq L_1 \left\{ |y_k - S_{k-1}(x_k)| + |z_k - \tilde{S}_{k-1}(x_k)| \right\} \end{aligned}$$

and

$$\begin{aligned} & \left| f_1^{(i\alpha)}(x_k, y_k, z_k, y(g(x_k))) - f_1^{(i\alpha)}(x_k, S_{k-1}, \tilde{S}_{k-1}, S_{k-1}(g(x_k))) \right| \\ & \leq \frac{L_1}{1 - L_1 B_1} \left[ |y_k - S_{k-1}(x_k)| + |z_k - \tilde{S}_{k-1}(x_k)| \right] = d_0 (e_k + \tilde{e}_k) \end{aligned}$$

where  $d_0 = \frac{L_1}{1 - L_1 B_1}$ .

Similarly,

$$\left| y^{((r+1)\alpha)}(\zeta_k) - M_k^{(r\alpha)} \right| \leq \left| y^{((r+1)\alpha)}(\zeta_k) - y_k^{((r+1)\alpha)} \right| + \left| y_k^{((r+1)\alpha)} - M_k^{(r\alpha)} \right| \leq \omega(h) + d_0 (e_k + \tilde{e}_k)$$

where the constant  $L_1 > 0$  is the Lipschitz constant independent of  $h$ ,  $\omega(h)$  is the modulus of continuity of  $\omega(y^{((r+1)\alpha)}(\zeta_k, h))$  and  $|x - x_k| < |h| < 1$ . The inequality (16) is then reduced to

$$\begin{aligned} e_\alpha(x) & \leq e_k + \sum_{i=0}^{r-1} d_0 (e_k + \tilde{e}_k) \frac{|(x - x_k)^{(i+1)\alpha}|}{\Gamma((i+1)\alpha + 1)} + (\omega(h) + d_0 (e_k + \tilde{e}_k)) \frac{|(x - x_k)^{(r+1)\alpha}|}{\Gamma((r+1)\alpha + 1)} \\ & \leq e_k + \sum_{i=0}^{r-1} \frac{d_0 (e_k + \tilde{e}_k) h^{(i+1)\alpha}}{\Gamma((i+1)\alpha + 1)} + (\omega(h) + d_0 (e_k + \tilde{e}_k)) \frac{h^{(r+1)\alpha}}{\Gamma((r+1)\alpha + 1)} \\ & = e_k + \sum_{i=0}^r \frac{d_0 e_k h^{(i+1)\alpha}}{\Gamma((i+1)\alpha + 1)} + \sum_{i=0}^r \frac{d_0 \tilde{e}_k h^{(i+1)\alpha}}{\Gamma((i+1)\alpha + 1)} + \omega(h) \frac{h^{(r+1)\alpha}}{\Gamma((r+1)\alpha + 1)} \\ & \leq e_k + \sum_{i=0}^r \frac{d_0 e_k h}{\Gamma((i+1)\alpha + 1)} + \sum_{i=0}^r \frac{d_0 \tilde{e}_k h}{\Gamma((i+1)\alpha + 1)} + \omega(h) \frac{h^{(r+1)\alpha}}{\Gamma((r+1)\alpha + 1)} \\ & = e_k + d_1 e_k h + d_1 h \tilde{e}_k + \omega(h) \frac{h^{(r+1)\alpha}}{\Gamma((r+1)\alpha + 1)} \\ & = (1 + d_1 h) e_k + d_1 h \tilde{e}_k + \omega(h) \frac{h^{(r+1)\alpha}}{\Gamma((r+1)\alpha + 1)} \end{aligned}$$

where  $d_1 = \sum_{i=0}^r \frac{d_0}{\Gamma((i+1)\alpha + 1)}$  is constant independent of  $h$ .

In the same manner we can prove that

$$\tilde{e}_\alpha(x) \leq (1 + d_2 h) \tilde{e}_k + d_2 h e_k + \omega(h) \frac{h^{(r+1)\alpha}}{\Gamma((r+1)\alpha + 1)}$$

where  $d_2 = \sum_{i=0}^r \frac{\tilde{d}_0}{\Gamma((i+1)\alpha+1)}$  is constant independent of  $h$ .

The lemma is proved.

### 4. Stability Analysis of the Proposed Method

For analyzing the stability properties of the given method, we make a small change of the starting values and study the changes in the numerical solution produced by the method.

Now, we define the spline approximating function  $W(x)$  and  $\tilde{W}(x)$  as:

$$W_k(x) = W_{k-1}(x_k) + \sum_{i=0}^r N_k^{(i\alpha)} \frac{(x-x_k)^{(i+1)\alpha}}{\Gamma((i+1)\alpha+1)}, \tag{17}$$

$$\tilde{W}_k(x) = \tilde{W}_{k-1}(x_k) + \sum_{i=0}^r \tilde{N}_k^{(i\alpha)} \frac{(x-x_k)^{(i+1)\alpha}}{\Gamma((i+1)\alpha+1)}, \tag{18}$$

where  $N_k^{(\alpha)} = f_1^{(\alpha)}(x_k, W_{k-1}(x_k), \tilde{W}_{k-1}(x_k))$  and  $\tilde{N}_k^{(\alpha)} = f_2^{(\alpha)}(x_k, W_{k-1}(x_k), \tilde{W}_{k-1}(x_k))$  with  $W_{-1}(x_0) = y_0^*$ ,  $\tilde{W}_{-1}(x_0) = \tilde{y}_0^*$ , for  $x \in [x_k, x_{k+1}]$ ,  $k = 0, 1, \dots, n-1$ . and use the notation

$$e_\alpha^*(x) = |S_k(x) - W_k^{(\alpha)}(x)|, \text{ and } e_{\alpha,k}^* = |S_{k-1,\alpha} - W_{k-1,\alpha}(x_k)|, \tag{19}$$

$$\tilde{e}_\alpha^*(x) = |\tilde{S}_k(x) - \tilde{W}_k^{(\alpha)}(x)|, \text{ and } \tilde{e}_{\alpha,k}^* = |\tilde{S}_{k-1,\alpha} - \tilde{W}_{k-1,\alpha}(x_k)|, \tag{20}$$

**Lemma 2**

Let  $e_\alpha^*(x)$  and  $\tilde{e}_\alpha^*(x)$  be defined as in (19) and (20), then the inequalities

$$e_\alpha^*(x) \leq (1 + d_1 h) e_k^* + dh \tilde{e}_k^*$$

$$\tilde{e}_\alpha^*(x) \leq (1 + d_2 h) \tilde{e}_k^* + dh e_k^*$$

holds where  $d_1 = \sum_{i=0}^r \frac{d_0}{\Gamma((i+1)\alpha+1)}$  and  $d_2 = \sum_{i=0}^r \frac{\tilde{d}_0}{\Gamma((i+1)\alpha+1)}$  are constants independent of  $h$ .

**Proof**

Using Lipschitz condition and (9), (17), (19) and (20) we get, by dropping  $\alpha$ :

$$\begin{aligned} e^*(x) &= |S_k(x) - W_k(x)| \\ &= \left| (S_{k-1}(x_k) - W_{k-1}(x_k)) + \sum_{i=0}^r M_k^{(i\alpha)} \frac{(x-x_k)^{(i+1)\alpha}}{\Gamma((i+1)\alpha+1)} - \sum_{i=0}^r \tilde{M}_k^{(i\alpha)} \frac{(x-x_k)^{(i+1)\alpha}}{\Gamma((i+1)\alpha+1)} \right| \\ &\leq e_k^* + \sum_{i=0}^r |M_k^{(i\alpha)} - \tilde{M}_k^{(i\alpha)}| \frac{|(x-x_k)^{(i+1)\alpha}|}{\Gamma((i+1)\alpha+1)} \\ &\leq e_k^* + \sum_{i=0}^r |M_k^{(i\alpha)} - \tilde{M}_k^{(i\alpha)}| \frac{h^{(i+1)\alpha}}{\Gamma((i+1)\alpha+1)} \end{aligned} \tag{21}$$

but

$$|M_k^{(i\alpha)} - \tilde{M}_k^{(i\alpha)}| \leq d_0 (e_k^* + \tilde{e}_k^*). \tag{22}$$

where  $d_0 = \frac{L_1}{1 - L_1 B_1}$

Thus from (21) and (22) we obtain:

$$e^*(x) \leq e_k^* + d_1 e_k^* h + d_1 \tilde{e}_k^* h \leq (1 + dh) e_k^* + d_1 \tilde{e}_k^* h.$$

where,  $d_1 = \sum_{i=0}^r \frac{d_0}{\Gamma((i+1)\alpha + 1)}$  is constant independent of  $h$ .

In the same manner we can prove that

$$\tilde{e}^*(x) \leq \tilde{e}_k^* + d_2 \tilde{e}_k^* h + d_2 e_k^* h \leq (1 + dh) \tilde{e}_k^* + d_2 e_k^* h$$

where  $\tilde{d}_0 = \frac{L_2}{1 - L_2 B_2}$  and  $d_2 = \sum_{i=0}^r \frac{\tilde{d}_0}{\Gamma((i+1)\alpha + 1)}$  is constant independent of  $h$ . Thus the lemma is proved.

### 5. Numerical Example

Consider the system of fractional ordinary delay differential equations

$$D^\alpha [y](x) = -y(x) + z(x/2) + \frac{3}{4}x^2 + \frac{2}{\Gamma(3-\alpha)}x^{2-\alpha}$$

$$D^\alpha [z](x) = z(x) - y(x/2) - \frac{3}{4}x^2 + \frac{2}{\Gamma(3-\alpha)}x^{2-\alpha}$$

The exact solution is given by  $y = x^2$  and  $z = x^2$ .

The obtained numerical results are summarized in **Table 1**, **Table 2** to illustrate the accuracy and the stability of the proposed spline method using spline function of polynomial form. The first column in each table, represents the different values of  $\alpha$ , the second column represents the values of  $x$ . The third column gives the approximate solution at the corresponding points while the fourth column gives the absolute error between the exact solution and the obtained approximate numerical solution with the initial conditions  $y(0) = 0$  and  $z(0) = 0$ . With small change in the initial conditions,  $y^*(0) = 0.00001$  and  $z^*(0) = 0.00001$ , the approximate solution is computed as

**Table 1.** The accuracy and stability of the proposed spline method using spline function of polynomial form (using  $h = 0.01$ ).

| $\alpha$ | $x$  | Appr. solution for the problem | Absolute Error       | Appr. solution for the perturbed problem | Absolute diff. between the two Appr. solutions |
|----------|------|--------------------------------|----------------------|--|--|
| 0.1      | 0.01 | $y = 0.000925756$              | $8.2 \times 10^{-4}$ | 0.000925771                              | $1.50275 \times 10^{-8}$                       |
|          | 0.02 | $y = 0.00286044$               | $2.5 \times 10^{-3}$ | 0.00286047                               | $2.33452 \times 10^{-8}$                       |
|          | 0.03 | $y = 0.00555181$               | $4.7 \times 10^{-3}$ | 0.00555184                               | $3.03307 \times 10^{-8}$                       |
|          | 0.04 | $y = 0.00890308$               | $7.3 \times 10^{-3}$ | 0.00890311                               | $3.66017 \times 10^{-8}$                       |
|          | 0.05 | $y = 0.0128567$                | $1.0 \times 10^{-2}$ | 0.0128568                                | $4.24062 \times 10^{-8}$                       |
| 0.2      | 0.01 | $y = 0.00054009$               | $4.4 \times 10^{-4}$ | 0.000540099                              | $9.30474 \times 10^{-9}$                       |
|          | 0.02 | $y = 0.00178791$               | $1.4 \times 10^{-3}$ | 0.00178793                               | $1.54803 \times 10^{-8}$                       |
|          | 0.03 | $y = 0.00361274$               | $2.7 \times 10^{-3}$ | 0.00361276                               | $2.09326 \times 10^{-8}$                       |
|          | 0.04 | $y = 0.00596128$               | $4.4 \times 10^{-3}$ | 0.0059613                                | $2.59856 \times 10^{-8}$                       |
|          | 0.05 | $y = 0.00880108$               | $6.1 \times 10^{-3}$ | 0.00880111                               | $3.07735 \times 10^{-8}$                       |
| 0.3      | 0.01 | $y = 0.000313701$              | $2.1 \times 10^{-4}$ | 0.000313707                              | $5.71685 \times 10^{-9}$                       |
|          | 0.02 | $y = 0.00111262$               | $7.1 \times 10^{-4}$ | 0.00111263                               | $1.01865 \times 10^{-8}$                       |
|          | 0.03 | $y = 0.00234065$               | $1.4 \times 10^{-3}$ | 0.00234066                               | $1.43368 \times 10^{-8}$                       |
|          | 0.04 | $y = 0.00397411$               | $2.4 \times 10^{-3}$ | 0.00397413                               | $1.83092 \times 10^{-8}$                       |
|          | 0.05 | $y = 0.00599858$               | $3.5 \times 10^{-3}$ | 0.0059986                                | $2.21638 \times 10^{-8}$                       |
| 0.4      | 0.01 | $y = 0.000181438$              | $8.1 \times 10^{-5}$ | 0.000181441                              | $3.4872 \times 10^{-9}$                        |
|          | 0.02 | $y = 0.000689476$              | $2.9 \times 10^{-4}$ | 0.000689483                              | $6.65526 \times 10^{-9}$                       |
|          | 0.03 | $y = 0.00151012$               | $6.1 \times 10^{-4}$ | 0.00151013                               | $9.74972 \times 10^{-9}$                       |
|          | 0.04 | $y = 0.00263828$               | $1.0 \times 10^{-3}$ | 0.0026383                                | $1.28095 \times 10^{-8}$                       |
|          | 0.05 | $y = 0.00407141$               | $1.0 \times 10^{-3}$ | 0.00407143                               | $1.58509 \times 10^{-8}$                       |
| 0.5      | 0.01 | $y = 0.000104514$              | $4.5 \times 10^{-6}$ | 0.000104516                              | $2.11285 \times 10^{-9}$                       |
|          | 0.02 | $y = 0.000425532$              | $2.6 \times 10^{-5}$ | 0.000425537                              | $4.31917 \times 10^{-9}$                       |
|          | 0.03 | $y = 0.000970359$              | $7.0 \times 10^{-5}$ | 0.000970365                              | $6.58633 \times 10^{-9}$                       |
|          | 0.04 | $y = 0.00174443$               | $1.4 \times 10^{-4}$ | 0.00174444                               | $8.90272 \times 10^{-9}$                       |
|          | 0.05 | $y = 0.00275231$               | $2.5 \times 10^{-4}$ | 0.00275232                               | $1.12616 \times 10^{-8}$                       |

**Table 2.** The accuracy and stability of the proposed spline method using spline function of polynomial form (using  $h = 0.01$ ).

| $\alpha$ | $X$  | Appr. Solution for the problem | Absolute Error       | Appr. solution for the perturbed problem | Absolute diff. between the two Appr. solutions |
|----------|------|--------------------------------|----------------------|--|--|
| 0.1      | 0.01 | $z = 0.000925756$              | $8.2 \times 10^{-4}$ | 0.000925771                              | $1.50275 \times 10^{-8}$                       |
|          | 0.02 | $z = 0.00286044$               | $2.5 \times 10^{-3}$ | 0.00286047                               | $2.33452 \times 10^{-8}$                       |
|          | 0.03 | $z = 0.00555181$               | $4.7 \times 10^{-3}$ | 0.00555184                               | $3.03307 \times 10^{-8}$                       |
|          | 0.04 | $z = 0.00890308$               | $7.3 \times 10^{-3}$ | 0.00890311                               | $3.66017 \times 10^{-8}$                       |
|          | 0.05 | $z = 0.0128567$                | $1.0 \times 10^{-2}$ | 0.0128568                                | $4.24062 \times 10^{-8}$                       |
| 0.2      | 0.01 | $z = 0.00054009$               | $4.4 \times 10^{-4}$ | 0.000540099                              | $9.30474 \times 10^{-9}$                       |
|          | 0.02 | $z = 0.00178791$               | $1.4 \times 10^{-3}$ | 0.00178793                               | $1.54803 \times 10^{-8}$                       |
|          | 0.03 | $z = 0.00361274$               | $2.7 \times 10^{-3}$ | 0.00361276                               | $2.09326 \times 10^{-8}$                       |
|          | 0.04 | $z = 0.00596128$               | $4.4 \times 10^{-3}$ | 0.0059613                                | $2.59856 \times 10^{-8}$                       |
|          | 0.05 | $z = 0.00880108$               | $6.1 \times 10^{-3}$ | 0.00880111                               | $3.07735 \times 10^{-8}$                       |
| 0.3      | 0.01 | $z = 0.000313701$              | $2.1 \times 10^{-4}$ | 0.000313707                              | $5.71685 \times 10^{-9}$                       |
|          | 0.02 | $z = 0.00111262$               | $7.1 \times 10^{-4}$ | 0.00111263                               | $1.01865 \times 10^{-8}$                       |
|          | 0.03 | $z = 0.00234065$               | $1.4 \times 10^{-3}$ | 0.00234066                               | $1.43368 \times 10^{-8}$                       |
|          | 0.04 | $z = 0.00397411$               | $2.4 \times 10^{-3}$ | 0.00397413                               | $1.83092 \times 10^{-8}$                       |
|          | 0.05 | $z = 0.00599858$               | $3.5 \times 10^{-3}$ | 0.0059986                                | $2.21638 \times 10^{-8}$                       |
| 0.4      | 0.01 | $z = 0.000181438$              | $8.1 \times 10^{-5}$ | 0.000181441                              | $3.4872 \times 10^{-9}$                        |
|          | 0.02 | $z = 0.000689476$              | $2.9 \times 10^{-4}$ | 0.000689483                              | $6.65526 \times 10^{-9}$                       |
|          | 0.03 | $z = 0.00151012$               | $6.1 \times 10^{-4}$ | 0.00151013                               | $9.74972 \times 10^{-9}$                       |
|          | 0.04 | $z = 0.00263828$               | $1.0 \times 10^{-3}$ | 0.0026383                                | $1.28095 \times 10^{-8}$                       |
|          | 0.05 | $z = 0.00407141$               | $1.0 \times 10^{-3}$ | 0.00407143                               | $1.58509 \times 10^{-8}$                       |
| 0.5      | 0.01 | $z = 0.000104514$              | $4.5 \times 10^{-6}$ | 0.000104516                              | $2.11285 \times 10^{-9}$                       |
|          | 0.02 | $z = 0.000425532$              | $2.6 \times 10^{-5}$ | 0.000425537                              | $4.31917 \times 10^{-9}$                       |
|          | 0.03 | $z = 0.000970359$              | $7.0 \times 10^{-5}$ | 0.000970365                              | $6.58633 \times 10^{-9}$                       |
|          | 0.04 | $z = 0.00174443$               | $1.4 \times 10^{-4}$ | 0.00174444                               | $8.90272 \times 10^{-9}$                       |
|          | 0.05 | $z = 0.00275231$               | $2.5 \times 10^{-4}$ | 0.00275232                               | $1.12616 \times 10^{-8}$                       |

shown in the fifth column. To test the stability, the difference between the two approximate solutions is computed as shown in the six column

From the obtained results in **Table 1**, **Table 2** respectively, we can see that the proposed method gives acceptable accuracy and the method is shown to be stable. Moreover, the algorithm of the proposed method has recursive nature which makes it easy and simple to be programmed.

## 6. Conclusion

We adapt the spline functions with some additional assumptions and definitions for approximating the solution of system of ordinary delay differential equation with fractional order which studied in [7] [8]. The error analysis and stability are theoretically investigated. A numerical example is given to illustrate the applicability, accuracy and stability of the proposed method. The obtained numerical results reveal that the methods are stable and give high accuracy.

## References

- [1] Loscalzo, F.R. (1969) An Introduction to the Application of Spline Function to Initial Value Problems. In: Greville, T.N.E., Ed., *Theory and Applications of Spline Functions*, Academic Press, New York, 37-64.
- [2] Loscalzo, F.R. and Tabot, T.D. (1967) Spline Function Approximations for Solutions of Ordinary Differential Equations. *Bulletin of the American Mathematical Society*, **73**, 708. <http://dx.doi.org/10.1090/S0002-9904-1967-11778-6>
- [3] Meir, A. and Sharma, A. (1973) Spline Functions and Approximation Theory. BirkhauserVerlag, Basel-Stuttgart. <http://dx.doi.org/10.1007/978-3-0348-5979-0>
- [4] Saad, B. (1966) Error of Arbitrary Order Method for Solving n-th Order Differential Equations by Spline Functions. PhD Thesis, Tanta University, Egypt.
- [5] Sherif, M.N., Abouelfarag, I. and Amer, T.S. (2014) Numerical Solution of Fractional Delay Differential Equations Using Spline Functions. *IJPAM*, **90**, 73-83. <http://dx.doi.org/10.12732/ijpam.v90i1.10>
- [6] Joakim, M. (2004) Riemann-Liouville Fractional Derivatives and the Taylor-Riemann Series. Department of Math. Uppsala University UUDM. Project Report 7.



- [7] Ramadan, M.A. (2005) Spline Solution of First Order Delay Differential Equation. *Journal of the Egyptian Mathematical Society*, **1**, 7-18.
- [8] Ramadan, M.A. (2007) Numerical Treatment of the Second Ordinary Delay Differential Equations by Spline Functions. *Proceeding of the Mathematical and Physical Society of Egypt*, **85**, 49-60.
- [9] Ramadan, M.A., Mokhtar, A., Abdel-Naby, M. and Taher, S. (2005) Spline Approximation for Second Order System of Delay Differential Equations. *Journal of the Egyptian Mathematical Society*, **13**, 177-188.
- [10] Ramadan, M.A., El-sherbeiny, A. and Sherif, M.N. (2009) The Use of Polynomial Spline Functions for the Solution of System of Second Order Delay Differential Equations. *International Journal of Computer Mathematics*, **86**, 1167-1181. <http://dx.doi.org/10.1080/00207160701769617>
- [11] Ramadan, M.A., Danaf, T.E. and Sherif, M.N. (2012) Error Analysis, Stability, and Numerical Solutions of Fractional—Order Differential Equations. *IJPAM*, 647-659, 76,5.
- [12] Eyaad, A. and Elgazzar, A.S. (2007) On Fractional Order Differential Equations Model for Nonlocal Epidemics. *Physica A*, **379**, 607-614. <http://dx.doi.org/10.1016/j.physa.2007.01.010>
- [13] Eyaad, A., El-Sayed, A.M.A. and Elsaka, H.A.A. (2007) Equilibrium Point, Stability, and Numerical Solutions of Fractional-Order Predator-Prey and Rabies Models. *Journal of Mathematical Analysis and Applications*, **325**, 542. <http://dx.doi.org/10.1016/j.jmaa.2006.01.087>
- [14] Dithelm, K. and Neville, J.F. (2003) Analysis of Fractional Differential Equation. The University of Manchester, Numerical Analysis Report 377.
- [15] Dithelm, K. and Freed, A.D. (1999) The FracPECE Subroutine for the Numerical Solution of Differential Equations of Fractions Order. In: Heinzl, S. and Plesser, T., Eds., *Forschung und Wissenschaftliches Rechnen 1998*, GWDG-Bericht, Gesellschaft für Wissenschaftliches Datenverarbeitung, Göttingen, 57-71.
- [16] Dithelm, K. and Freed, A.D. (1999) On the of Solution of Nonlinear Differential Equations Used in the Modeling of Viscoplasticity. In: Keil, F., Mackens, W., Vob, H. and Werther, J., Eds., *Scientific Computing in Chemical Engineering II-Computational Fluid Dynamics Reaction Engineering, and Molecular Properties*, Springer, Heidelberg, 217-224.
- [17] Dithelm, K. (1997) An Algorithm for the Numerical Solution of Differential Equations of Fractions Order. *Electronic Transactions on Numerical Analysis*, **5**, 1-6.
- [18] Dithelm, K., Judith, M.F., Neville, J.F. and Weilbeer, M. (2006) Pitfalls in Fast Numerical Solvers for Fractions Differential Equations. *Journal of Computational and Applied Mathematics*, **186**, 482-503. <http://dx.doi.org/10.1016/j.cam.2005.03.023>
- [19] Sweilam, N.M., Khader, M.M. and Al-Bar, R.F. (2007) Numerical Studies for a Multi-Order Fractional Differential Equation. *Physics Letters A*, **371**, 26-33. <http://dx.doi.org/10.1016/j.physleta.2007.06.016>
- [20] Micula, G., Fawzy, T. and Ramadan, Z. (1987) A Polynomial Spline Approximation. Method for Solving System of Ordinary Differential Equations. *Babes-Bolyai Cluj-Napoca Mathematica*, **4**, 55-60.
- [21] Ramadan, Z. (2000) On the Numerical Solution of a System of Third Order Ordinary Differential Equations by Spline Functions. *Ann Univ Sect. Comput*, **19**, 155-167.