Kirkuk Journal of Science Vol. 20, Iss. 3, p. 20-30, 2025

doi:10.32894/kujss.2025.162625.1228

A Numerical Fractional Spline for Solving System of Fractional Differential Equations

Paywast J. Hasan, Faraidun K. Hamasalh*

Department of Mathematics, College of Education, Sulaimani Polytechnic University, Bakrajo Technical Institute, Sulaimani, Iraq.

*Corresponding author : faraidun.hamasalh@spu.edu.iq

Article Information

Article Type:
Research Article

Keywords:
Boundary-value problem, Gamma function, Caputo fractional derivative, Riemann-Liouville fractional derivative, convergence analysis, errors estimation.

History:
Received: 9 July 2025
Revised: 19 August 2025
Accepted: 19 August 2025
Available Online: 30 September 2025

Citation: Paywast J. Hasan, Faraidun K. Hamasalh, A Numerical Fractional Spline for Solving System of Fractional Differential Equations, Kirkuk Journal of Science, 20(3), p. 20-30, 2025, https://doi.org/10.32894/kujss.2025.162625.1228

Abstract

Fractional-order differential equations are fundamental in diverse scientific and engineering fields, including population dynamics, optimal control, and physics. This paper presents a non-polynomial spline method for their numerical solution, establishing a linear system of algebraic equations in the representation of three-term recurrence equations, which is solved using an elimination algorithm. The maximum error estimations and the order of convergence for each example demonstrate the success and core contribution of the method and its performance is demonstrated through numerical and graphical examples. There is a thorough explanation of a mathematical procedure provided, as well as graphical and numerical examples solution for several examples. Results confirm that the proposed approach achieves superior accuracy and reliability compared to existing techniques.

1. Introduction:

In many branches of mathematics, science, and engineering, solving systems of equations is essential. These systems are particularly vital for testing and validating engineering designs, as well as for modeling and simulating physical processes [1]. Among them, boundary value problems (BVPs) with various boundary conditions have become effective tools for describing real-world scenarios and thus constitute a significant area of study. BVP systems of different orders are widely used in modelling population dynamics, brine tank systems, compartment modelling, water pollution, chemostats, cardiac arrhythmias, drug interactions, nutrient transport in aquariums, economic forecasting, train logistics, electrical circuits, coupled spring-mass systems, helicopter logging, and structural responses to earthquakes [2].

Recently, second-order BVP systems have garnered considerable attention. Researchers have developed numerous numerical methods to achieve accurate and efficient solutions. For instance, Khalid et al. [3] investigated both linear and nonlinear systems using the reproducing kernel method. Dehghan and Saadatmandi [4] and Gamel [5] proposed Sinc-collocation strategies, while Lu [6] introduced a variational iteration method. Modified and Laplace-based homotopy analysis methods were applied to nonlinear systems by Bataineh et al. [7] and Karwan et al. [8], respectively.

Spline-based methods have become increasingly popular in the numerical solution of BVPs due to their high accuracy and flexibility. Gupta and Chaurasia [9], Dehghan and Lakestani [10], Aniley, Duressa [11], and Khuri and Sayfy [12] addressed generalized BVP systems using a collocation approach, while Heilat et al. [13] applied B-spline scaling functions to second-order systems. Goh et al. [14] extended cubic uniform B-splines to solve singular BVPs effectively.

Foundational concepts related to linear algebra, which underpin the construction and analysis of these numerical methods, are thoroughly discussed in the works of Fay and Milici et al. [15], [16], as well as Chaurasia et al. [17]. Further, the theoretical background of fractional calculus and its application to differential equations is elaborated in the work of Lay et al. [18], demonstrated the efficiency of spline techniques in handling higher-order and engineering-related boundary value problems.

Fractional differential equations (FDEs) have gained prominence due to their ability to model memory and hereditary properties in various physical phenomena. However, there remains a noticeable gap in the literature regarding the solution of systems of fractional differential equations. This gap was one of the primary motivations for undertaking the present work. In this context, the application of non-polynomial spline functions offers a promising direction. Faraidun and Headayat [19] introduced non-polynomial splines to fractional problems, while Emadifar et al. [20] and Abbas et al. [21] extended spline-based approaches to fractional and initial value problems, respectively.

In light of the above, this study aims to address the deficiency in existing research concerning the numerical solution of systems of fractional differential equations by introducing an efficient method based on the new non-polynomial fractional spline interpolation with continuity equation.

2. Basic Definition:

The non-polynomial spline offers a powerful approach to improving numerical schemes for systems of differential equations. In this section, we discuss the fundamental concepts associated with non-polynomial splines and fractional derivatives, including specific definitions such as the Caputo and Riemann–Liouville derivatives.

2.1 Definition Gamma function [16]: The Gamma function $\Gamma(.)$is defined as:

$$\Gamma(p) = \int_0^\infty e^{-x}x^{p-1}dx$$

2.2 Definition Caputo fractional derivatives [16]:

The Caputo fractional derivative operator ${}_a^C D_t^\alpha$ of order $\alpha \in R^+$ of a function u $\in C_{-1}^m$ [a,b] and $\alpha \in$[m-1, m], m$\in$ N is defined as:

Thus, for = m , m$\in$N, and u$\in C^m$ [a,b], we have ${}_a^C D_a^0$ u(t) = u(t) and ${}_a^C D_a^m$ u(t) = $u^{(m)}$ (t). The Caputo fractional derivative of order α is given by:

$${}^C D^\alpha f(x) = \frac{1}{\Gamma(n-\alpha)}\int_a^x (x-t)^{n-\alpha-1} f^{(n)}(t)dt.$$

2.3 Definition Riemann-Liouville fractional derivatives [16]:

The Riemann-Liouville fractional derivative operator ${}_\alpha^{RL} D_t^\alpha$ of order $\alpha \geq 0, m = [\alpha]$, ceiling function, and u $\in C_{-1}^m$ [a,b], is normally defined as:

$${}_\alpha^{RL} D_t^\alpha = D_t^m a J_t^{m-\alpha} u(t)$$

where ,if α = m, m $\in$ N,and u $\in C^m$ [a,b], we have ${}_a^{RL} D_a^m$ u(t) = $u^{(m)}$ (t).

3. Modified of Non-Polynomial Spline Interpolation:

In the non-polynomial fractional spline interpolation, the newest formula derivative at a given point is used in constructing the interpolating polynomial. To compute the derivatives at the selected nodes, we use an approximation based on finite differences, as well as a Taylor series expansion. This provides an efficient way to handle fractional derivatives without requiring complex symbolic calculations. We can set a framework of an equally spaced partition of an interval [a, b], dividing into N equal sections as a = $x_0 < x_1 < x_2 < \ldots\ldots\ldots < x_n$=b, θ = kh, h = (b-a)/N, and k is free parameter as follows:

$$\begin{aligned} p_j(x) =& A_{1j} cosk(x-x_j) + A_{2j} e^{k(x-x_j)} \\ &+ A_{3j}(x-x_j)^{1/2} + A_{4j}, j = 0,1,2,\ldots,N_j \end{aligned} \quad (1)$$

The interpolate conditions: $p_j x_{(j+1)} = s_(j+1)$, $p_j(xj) = s_j$, $p_j^{(2)} x_j = M_j$, and $p_j^{(2)}\ x_{(j+1)} = M_(j+1)$.

3.1 Theorem:

Uniqueness of Non-polynomial Interpolation with Fractional Order Let $x_{i_{i=0}}^n$ be distinct uniformly spaced nodes in the interval [a,b], and let s(x) be a function such that it's fractional derivative D^2 p(x) and s(x) exists at each x_i and continuous on [a,b], then there exists a unique interpolating function p_j (x).

Proof: Using the non-polynomial spline interpolation in equation 1, with the four conditions after some simplifications by taking the derivative and interpolating at each x_i, we can obtain the following coefficients:

$$A_{1j} = \frac{k^2(s_j - s_{j+1}) - M_j - 4\theta^2 M_(j+1) + e^\theta M_j + 4M_j\theta^2 e^\theta}{k^2(-cos\theta + 2 - e^\theta + 4\theta^2 cos\theta - 4\theta^2 e^\theta)}$$

$$A_{2j} = \frac{k^2(s_j - s_{j+1}) - cos\theta M_j + M_j + 4\theta^2 cos\theta M_j - 4M_{j+1}\theta^2}{k^2(-cos\theta + 2 - e^\theta + 4\theta^2 cos\theta - 4\theta^2 e^\theta)}$$

$$A_{3j} = \frac{-4h^{3/2}[(2cos\theta e^{\theta} - cos\theta - e^{\theta})M_j + (2 - e^{\theta} - cos\theta)M_{j+1} + (cos\theta - e^{\theta})k^2 s_j + (e^{\theta} - cos\theta)k^2 s_(j+1)}{(-cos\theta + 2 - e^{\theta} + 4\theta^2 cos\theta - 4\theta^2 e^{\theta})}$$

$$A_{4j} = \frac{2k^2 s_{j+1} + (4\theta^2 cos\theta - 4\theta^2 e^{\theta} - cos\theta - e^{\theta})k^2 s_j + (cos\theta - e^{\theta} - 4\theta^2 cos\theta - 4\theta^2 e^{\theta})M_j + 8\theta^2 M_{j+1}}{k^2(-cos\theta + 2 - e^{\theta} + 4\theta^2 cos\theta - 4\theta^2 e^{\theta})}$$

$$\begin{aligned} p_j(x) = & \frac{k^2(s_j - s_{j+1}) - M_j - 4\theta^2 M_{j+1} + e^{\theta} M_j + 4M_j\theta^2 e^{\theta}}{k^2(-cos\theta + 2 - e^{\theta} + 4\theta^2 cos\theta - 4\theta^2 e^{\theta}} cosk(x - x_j) + \\ & \frac{k^2(s_j - s_{j+1}) - cos\theta M_j + M_j + 4\theta^2 cos\theta M_j - 4M_(j+1)\theta^2}{k^2(-cos\theta + 2 - e^{\theta} + 4\theta^2 cos\theta - 4\theta^2 e^{\theta})} e^{k(x - x_j)} - \\ & \frac{4h^{3/2}[(2cos\theta e^{\theta} - cos\theta - e^{\theta})M_j + (2 - e^{\theta} - cos\theta)M_{j+1} + (cos\theta - e^{\theta})k^2 s_j + (e^{\theta} - cos\theta)k^2] s_{j+1}}{(-cos\theta + 2 - e^{\theta} + 4\theta^2 cos\theta - 4\theta^2 e^{\theta})} (x - x_j)^{1/2} + \\ & \frac{2k^2 s_{j+1} + (4\theta^2 cos\theta - 4\theta^2 e^{\theta} - cos\theta - e^{\theta})k^2 s_j + (cos\theta - e^{\theta} - 4\theta^2 cos\theta - 4\theta^2 e^{\theta})M_j + 8\theta^2 M_{j+1}}{k^2(-cos\theta + 2 - e^{\theta} + 4\theta^2 cos\theta - 4\theta^2 e^{\theta})} \end{aligned}$$

Suppose two interplants $p_j(x)$ and $p_j^{(2)}(x)$ satisfy the conditions, as found the above coefficients, since p(x) is a linear combination of basis interpolation function at all nodes then $p_j(x)$ is unique polynomial similar as [18], [19].

3.2 Matrix representation:

To express the equation 1 in the matrix form, we first need to identify the variables and constants, and then structure them into vectors and matrices. The given formula is:
Let $D = k^2(-\cos\theta + 2 - e^{\theta} + 4\theta^2 \cos\theta - 4\theta^2 e^{\theta})$.
$p_j(x) = A_{1j} \cos k(x - x_j) + A_{2j} e^{k}(x - x_j) + A_{3j} (x - x_j)^{1/2} + A_{4j}$
Where:

$$A_{1j} = \frac{k^2(s_j - s_{j+1}) - M_j - 4\theta^2 M_{j+1} + e^{\theta} M_j + 4M_j\theta^2 e^{\theta}}{D}$$

$$A_{2j} = \frac{k^2(s_j - s_{j+1}) - cos\theta M_j + M_j + 4\theta^2 cos\theta M_j - 4M_{j+1}\theta^2}{D}$$

$$A_{3j} = -4k^2 h^{3/2} \frac{(2cos\theta e^{\theta} - cos\theta - e^{\theta})M_j + (2 - e^{\theta} - cos\theta)M_{j+1} + (cos\theta - e^{\theta})k^2 s_j + (e^{\theta} - cos\theta)k^2 s_{j+1}}{D}$$

$$A_{4j} = \frac{2k^2 s_{j+1} + (4\theta^2 cos\theta - 4\theta^2 e^{\theta} - cos\theta - e^{\theta})k^2 s_j + (cos\theta - e^{\theta} - 4\theta^2 cos\theta - 4\theta^2 e^{\theta})M_j + 8\theta^2 M_{j+1}}{D}$$

Let's rearrange A_{1j}, A_{2j}, A_{3j} and A_{4j} in terms of s_j, s_{j+1}, M_j, M_{j+1}.

$A_{1j} = \frac{1}{D}[k^2 s_j - k^2 s_{j+1} + (e^{\theta} + 4\theta^2 e^{\theta} - 1) M_j - 4\theta^2 M_{j+1}]$

$A_{2j} = \frac{1}{D}[k^2 s_j - k^2 s_{j+1} + (1 - \cos\theta + 4\theta^2 \cos\theta) M_j - 4\theta^2 M_{j+1}]$

Let $A_1 = -\frac{4k^2h^{3/2})}{D}$.

$A_3j = A_1$ [(cosθ-e$^\theta$)k^2 s_j +(e$^\theta$ - cos θ)k^2 s_{j+1} + (2 cos θ e$^\theta$ - cosθ - e$^\theta$) M_j + (2 - e$^\theta$ - cosθ) M_{j+1}]

$A_{4j} = \frac{1}{D}$ [(4θ^2 cosθ - 4θ^2 e^θ - cosθ - e^θ) k^2 s_j + 2 k^2 s_{j+1} + (cosθ - e^θ - 4θ^2 cosθ - 4θ^2 e$^\theta$) M_j + 8θ^2 M_{j+1}]

Now, let's define a column vector for the independent variables:

$$V = \begin{bmatrix} s_j \\ s_{j+1} \\ M_j \\ M_{j+1} \end{bmatrix}$$

$V = (s_j s_{j+1} M_j M_{j+1})$

And a row vector for the functions of x:

F(x) = [cosk(x-x_j) $e^{k(x-x_j)}$ (x-x_j) $^{1/2}$ 1]

We can express p_j (x) in the form p_j (x) = F(x)MV or a similar matrix multiplication. However, given the complexity, it's more straightforward to define coefficient matrices for each term. Let's define coefficient matrices for s_j, s_{j+1}, M_j, M_{j+1} for each part of the equation.

$$\begin{aligned} p_j(x) = &(\alpha_1 s_j + \alpha_2 s_{j+1} + \alpha_3 M_j + \alpha_4 M_{j+1}) cosk(x-x_j) + \\ &(\beta_1 s_j + \beta_2 s_{j+1} + \beta_3 M_j + \beta_4 M_{j+1}) e^{k(x-x_j)} + \\ &(\gamma_1 s_j + \gamma_2 s_{j+1} + \gamma_3 M_j + \gamma_4 M_{j+1})(x-x_j)^{1/2} + \\ &(\delta_1 s_j + \delta_2 s_(j+1) + \delta_3 M_j + \delta_4 M_{j+1}) \end{aligned} \tag{2}$$

Where:

$$\alpha_1 = \frac{k^2}{D}, \alpha_2 = -\frac{k^2}{D}, \alpha_3 = \frac{e^\theta + 4\theta^2 e^\theta - 1)}{D}, and \alpha_4 = -\frac{4\theta^2}{D}$$
$$\beta_1 = \frac{k^2}{D}, \beta_2 = -\frac{k^2}{D}, \beta_3 = \frac{1 - cos\theta + 4\theta^2 cos\theta}{D}, and$$
$$\beta_4 = -\frac{4\beta^2}{D} \gamma_1 = A_1(cos\theta - e^\theta)k^2, \gamma_2 = A_1(e^\theta - cos\theta)k^2,$$
$$\gamma_3 = A_1(2cos\theta e^\theta - cos\theta - e^\theta), And \gamma_4 = A_1(2 - e^\theta - cos\theta).$$
$$\delta_1 = k^2(4\theta^2 cos\theta - 4\theta^2 e^\theta - cos\theta - e^\theta)D, \delta_2 = \frac{2k^2}{D},$$
$$\delta_3 = \frac{cos\theta - e^\theta - 4\theta^2 cos\theta - 4\theta^2 e^\theta}{D}, and \delta_4 = \frac{8\theta^2}{D}.$$

$$\text{Let } C = \begin{bmatrix} s_j \\ s_{j+1} \\ M_j \\ M_{j+1} \end{bmatrix}$$

Then we can write eq(2) as:

$$p_j(x) = \mathbf{A} \cdot \mathbf{C} cosk(x-x_j) + \mathbf{B} \cdot \mathbf{C} e^{k(x-x_j)} + \Gamma \cdot \mathbf{C}(x-x_j)^(1/2) + \Delta \cdot \mathbf{C}$$

Where **A**, **B**, γ, Δ are row vectors containing the respective coefficients:

$\mathbf{A} = [\alpha_1\ \alpha_2\ \alpha_3\ \alpha_4]$, $\mathbf{B} = [\beta_1\ \beta_2\ \beta_3\ \beta_4]$, $\Gamma = [\gamma_1\ \gamma_2\ \gamma_3\ \gamma_4]$, $\Delta = [\delta_1\ \delta_2\ \delta_3\ \delta_4]$

This form highlights the dependence of p_j (x) on the parameters s_j, s_{j+1}, M_j, M_{j+1} through a linear combination, with coefficients that depend on k, θ, h, and x. To express this in a more compact matrix form, we can define a matrix K(x) such that p_j (x) = **K**(x)**C**.

$$\mathbf{K}(x) = \begin{bmatrix} (\alpha_1 cosk(x-x_j) + \beta_1 e^{k(x-x_j)} + \gamma_1 (x-x_j)^{1/2} + \delta_1) \\ (\alpha_2 cosk(x-x_j) + \beta_2 e^{k(x-x_j)} + \gamma_2 (x-x_j)^{1/2} + \delta_2) \\ (\alpha_3 cosk(x-x_j) + \beta_3 e^{k(x-x_j)} + \gamma_3 (x-x_j)^{1/2} + \delta_3) \\ (\alpha_4 cosk(x-x_j) + \beta_4 e^{k(x-x_j)} + \gamma_4 (x-x_j)^{1/2} + \delta_4) \end{bmatrix}^T$$

Thus, p_j (x) = **K**(x)C, where **K**(x) is a 1×4 row vector and C is a 4×1 column vector.

$$\text{Let } C = \begin{bmatrix} s_j \\ s_{j+1} \\ M_j \\ M_{j+1} \end{bmatrix} \text{ be the vector of unknown parameters.}$$

Let the vector basis functions of x be:

$$\begin{bmatrix} (cosk(x-x_j) \\ e^{k(x-x_j)} \\ (x-x_j)^{1/2} \\ 1 \end{bmatrix}$$

Now, we need to define the coefficient matrix that multiplies these basis functions and the parameter vector.

p_j (x) = $\frac{1}{K_c}$ $\mathbf{L}_1$ **C**cosk(x-x_j) + $\frac{1}{K_c}$ $\mathbf{L}_2$ **C**e$^{k(x-x_j)}$ + $\frac{K_h}{K_c}$ $\mathbf{L}_3$ **C**(x-x_j)$^{\frac{1}{2}}$ + $\frac{1}{K_c}$ $\mathbf{L}_3$ **C**

Where:

L_1 = [k^2 - k^2 (e$^\theta$ + 4θ^2 e$^\theta$ - 1) - 4θ^2]

L_2 = [k^2 - k^2 (1 - cosθ + 4θ^2 cosθ) - 4θ^2]

L_3 = [(cosθ - e$^\theta$) k^2 (e$^\theta$ - cosθ) k^2 (2cosθe$^\theta$ - cosθ - e$^\theta$) (2 - e$^\theta$ - cosθ)]

L_4 = [(4θ^2 cosθ - 4θ^2 e$^\theta$ - cosθ - e$^\theta$) k^2 2k^2 (cosθ - e$^\theta$ - 4θ^2 cosθ - 4θ^2 e$^\theta$) 8θ^2)]

This can be more compactly written as:

$p_j(x) = M(x)C$

Where M(x) is a 1×4 row vector whose elements are functions of x:

$M(x) = [m_1(x)\ m_2(x)\ m_3(x)\ m_4(x)]$

$m_1(x) = \frac{k^2}{K_c}\cos k(x-x_j) + \frac{k^2}{K_c}e^{k(x-x_j)} + \frac{k^h}{K_c}(\cos\theta - e^{\theta})k^2(x-x_j)^{\frac{1}{2}} + k^2/K_c(4\theta^2\cos\theta - 4\theta^2 e^{\theta} - \cos\theta - e^{\theta})k^2$

$m_2(x) = -\frac{k^2}{K_c}\cos k(x-x_j) - \frac{k^2}{K_c}e^{k(x-x_j)} + \frac{k^h}{K_c}(e^{\theta} - \cos\theta)k^2(x-x_j)^{\frac{1}{2}} + \frac{k^2}{K_c}(2k^2)$

$m_3(x) = \frac{(e^{\theta}+4\theta^2 e^{\theta}-1)}{K_c}\cos k(x-x_j) + \frac{(1-\cos\theta+4\theta^2\cos\theta)}{K_c}e^{k(x-x_j)} + \frac{k^h}{K_c}(2\cos\theta\, e^{\theta} - \cos\theta - e^{\theta})(x-x_j)^{\frac{1}{2}} + \frac{(\cos\theta - e^{\theta} - 4\theta^2\cos\theta - 4\theta^2 e^{\theta})}{K_c}$

$m_4(x) = -\frac{4\theta^2}{K_c}\cos k(x-x_j) - \frac{4\theta^2}{K_c}e^{k(x-x_j)} + \frac{k^h}{K_c}(2 - e^{\theta} - \cos\theta)(x-x_j)^{\frac{1}{2}} + \frac{8\theta^2}{K_c}$

This representation is the most direct matrix form for $p_j(x)$.

4. Continuity Condition for Fractional Spline Scheme:

This modification presents an iterative formulation based on the given formula of theorem 3.1 for solving system of differential equations. The basic ideas and concepts are adapted from foundational methods introduced in [20], and further developed using fractional polynomial techniques, with Caputo derivatives [16].

Let the fractional spline range polynomials $p_j(x)$ form the basis, where j=0, 1,. . . ,N and θ = kh at the knots , apply the half derivative continuity conditions of spline functions to acquire the following relations:

$p_j^{(1/2)}(x) = p_{j-1}{}^{(1/2)}(x)$

And evaluating all the coefficients of equation 1, we obtain

$$p_j(x) = \frac{k^2(s_j - s_{j+1}) - M_j - 4\theta^2 M_{j+1} + e^{\theta}M_j + 4M_j\theta^2 e^{\theta})}{k^2(-\cos\theta + 2 - e^{\theta} + 4\theta^2\cos\theta - 4\theta^2 e^{\theta})}\cos k(x-x_j) +$$
$$\frac{(k^2(s_j - s_j + 1) - \cos\theta M_j + M_j + 4\theta^2\cos\theta M_j - 4M_(j+1)\theta^2)}{(k^2(-\cos\theta + 2 - e^{\theta} + 4\theta^2\cos\theta - 4\theta^2 e^{\theta})}e^{k(x-x_j)} -$$
$$\frac{4h^{3/2}[(2\cos\theta e^{\theta} - \cos\theta - e^{\theta})M_j + (2 - e^{\theta} - \cos\theta)M_{j+1} + (\cos\theta - e^{\theta})k^2 s_j + (e^{\theta} - \cos\theta)k^2 s_{j+1}}{(-\cos\theta + 2 - e^{\theta} + 4\theta^2\cos\theta - 4\theta^2 e^{\theta})}(x - x_j^{1/2} +$$
$$\frac{(2k^2 s_{j+1} + (4\theta^2\cos\theta - 4\theta^2 e^{\theta} - \cos\theta - e^{\theta})k^2 s_j + (\cos\theta - e^{\theta} - 4\theta^2\cos\theta - 4\theta^2 e^{\theta})M_j + 8\theta^2 M_{j+1})]}{(k^2(-\cos\theta + 2 - e^{\theta} + 4\theta^2\cos\theta - 4\theta^2 e^{\theta})}$$

Compute the fractional derivatives, onto the basis p(x) to form each x_i, we obtain the iterative formula:

$$s_{j+1} - 2s_j + s_{j-1} = -1/k^2[\alpha M_{j+1} + \beta M_j + \gamma M_{j-1}] \quad (3)$$
$$j = 1, \ldots, N-1.$$

Where,

$$\alpha = \frac{(1+\sqrt{2})4\theta^2 + (4 - 2e^{\theta} - 2\cos\theta)\sqrt{2}\sqrt{\pi}\theta^{3/2}}{1+\sqrt{2}+2\sqrt{2}\sqrt{\pi}\theta^{3/2}e^{\theta} - 2\sqrt{2}\sqrt{\pi}\theta^{3/2}\cos\theta},$$
$$\beta = \frac{1 - e^{\theta} - (e^{\theta}+1+\sqrt{2})4\theta^2 - \sqrt{2} - 4\sqrt{2}\sqrt{\pi}\theta^{3/2} - (4\theta^2 - 1 - 4\sqrt{\pi}\theta^{3/2})\sqrt{2}\cos\theta}{1+\sqrt{2}+2\sqrt{2}\sqrt{\pi}\theta^{3/2}e^{\theta} - 2\sqrt{2}\sqrt{\pi}\theta^{3/2}\cos\theta}, and$$
$$\gamma = \frac{(1+4\theta^2+2\sqrt{2}\sqrt{\pi}\theta^{3/2})e^{\theta} + \sqrt{2} + (4\theta^2 - 1 - 4\sqrt{\pi}\theta^{3/2}e^{\theta} + 2\sqrt{\pi}\theta^{3/2})\sqrt{2}\cos\theta - 1}{1+\sqrt{2}+2\sqrt{2}\sqrt{\pi}\theta^{3/2}e^{\theta} - 2\sqrt{2}\sqrt{\pi}\theta^{3/2}\cos\theta}$$

These formulas enable the construction of a compact iterative formula that encapsulates fractional derivatives, consistent with methods from [15], [20] and [21].

5. Theoretical Analysis:

The convergence analysis reveals that the proposed non-polynomial spline method demonstrates superior accuracy and stability compared to the classical Taylor series, particularly for functions with higher-order derivatives or localized behavior. While the Taylor series relies on polynomial approximations centered around a single point, the spline approach captures the function's behavior over intervals, allowing for faster convergence with fewer terms. This makes the non-polynomial spline especially effective for problems where the Taylor series may diverge or require a high number of terms to achieve comparable precision.

Lemma 5.1:

Let f∈C^{α} ([a,b]) be a function with fractional smoothness α ∈(a,b], and let p_j (x) denote the non-polynomial spline-type fractional interplant constructed from values f(x_i) and fractional derivatives D^{α} f(x_i) at nodes $x_{i}{}_{i=0}^{n}$. Then the interpolation error satisfies:

$$|f^{(m)}(x)-p^(m)_j(x)| \leq C\omega(h)h^m, m=0,1,2.$$

where

1. h = max—x_{i+1}-x_i — is the maximum spacing between the nodes,
2. ω(h) is the modulus of continuity of D^{α} f,
3. C is a constant depending only on α and the interpolation scheme.

Proof: The modulus of continuity of a function f, denoted by ω(f,δ), is defined as:

ω(f,δ) = sup $|f(x)-f(y)|$ (x,y ∈ [a,b] $|x-y| \leq \delta$)

For fractional derivatives of f,define:

ω_{α} (f,α) = sup $|D^{\alpha}f(x)-D^{\alpha}f(y)|$, $\alpha \in$ 0,1,2.
(x,y ∈ [a,b] $|x-y| \leq \delta$)

We begin by expanding f(x) at each node x_i using the Taylor series with fractional remainder:

f(x) = f(x_i) + $\frac{2}{\sqrt{\pi}}$ $(x-x_i)^{1/2}$ $f^{(\frac{1}{2})}$ (x_i) + (x-x_i) f' (x_i) + $\frac{4}{(3\sqrt{\pi})}$ $(x-x_i)^{\frac{3}{2}}$ $f^{(\frac{3}{2})}$ (x_i) + R_2 (x,x_i),

The error estimation of non-polynomial spline and analytic function is $|p_j(x)-f(x)|, since |x-x_i| \leq$h and $\omega(|x-x_i|)$ $\leq \omega$(h), and letting x=x_{j+1} , to obtain:

$$|p_j(x)-f(x)| = |\frac{1}{k^2(-cos\theta+2-e^{\theta}+4\theta^2cos\theta-4\theta^2e^{\theta})}$$
$$[(k^2(s_j-s_{j+1})-M_j-4\theta^2M_{j+1}+e^{\theta}M_j+4M_j\theta^2e^{\theta})cos\theta+$$
$$(k^2(s_j-s_{j+1})-cos\theta M_j+M_j+\theta^2cos\theta M_j-4M_{j+1}\theta^2)e^{\theta}-$$
$$\Big((2cos\theta e^{\theta}-cos\theta-e^{\theta})M_j+(2-e^{\theta}-cos\theta)M_{j+1}+$$
$$(cos\theta-e^{\theta})k^2s_j+(e^{\theta}-cos\theta)k^2s_{j+1}\Big)4\theta^2+$$
$$2k^2s_{j+1}+(4\theta^2cos\theta-4\theta^2e^{\theta}-cos\theta-e^{\theta})k^2s_j+$$
$$(cos\theta-e^{\theta}-4\theta^2cos\theta-4\theta^2e^{\theta})M_j+8\theta^2M_{j+1}]-$$
$$[y_j+\frac{2h^{1/2}}{\sqrt{\pi}}y_j^{1/2}+hy_j'+\frac{4h^3/2}{3\sqrt{\pi}}y_j^{3/2}+\frac{h^2}{2}y_j''(\theta 1)]|$$

Substitute s_j = p_j, s_{j+1} = p_{j+1}, M_j = p_j'', M_{j+1} = p_{j+1}'' and

α_1 = $\frac{1}{k^2(-cos\theta+2-e^{\theta}+4\theta^2cos\theta-4\theta^2e^{\theta})}$, then after simplification we get:

$$|p_j(x)-f(x)| = |\alpha_1(-8\theta^2e^{\theta}cos\theta p_j+(k^2e^{\theta}-k^2cos\theta-$$
$$4\theta^2k^2e^{\theta}+4\theta^2k^2cos\theta+2k^2)p_{j+1})-[p_j+\frac{2h^{1/2}}{\sqrt{\pi}}p_j^{1/2}+$$
$$hp_j'+\frac{4h^{3/2}}{3\sqrt{\pi}}p_j^{3/2}+\frac{h^2}{2}p_j''(\theta 1)]| \tag{4}$$

let $\alpha_2=-8\theta^2e^{\theta}cos\theta, \alpha_3=k^2e^{\theta}-k^2cos\theta-4\theta^2k^2e^{\theta}$

$+4\theta^2k^2cos\theta+2k^2 Therefore:$

$$|f(x)-p_j(x)| = |E(x)| \leq C\omega(h)h^2, h = max|x-x_i|$$

$|\sqrt{x-x_i}D^{\frac{1}{2}}f(x_i)| \leq \sqrt{h}||D^{\frac{1}{2}}f||_{\infty}$ and

$$|(x-x_i)^{\frac{3}{2}}D^{\frac{3}{2}}f(x_i)| \leq h^{\frac{3}{2}}||D^{\frac{3}{2}}f||_{\infty}.$$

and simplifying with constants C_1 = (α_1 α_2 + α_1 α_3 - α_4), C_2 = (α_1 α_3 -α_4), C_3 = (α_1 α_3 - α_4) C_4 = (α_1 α_3 - α_4) from equation 4, independent h, and using Taylor expansion can be write as:

$$|f(x)-p_j(x)| = |E(x)| \leq Ch^2\omega(h), where Ch^2 = Max(C_1h^{\frac{3}{2}}+C_2h^2+C_3h+C_3h^{\frac{1}{2}}).$$

Similarly, we can find error bound for the derivatives as follows:

$\alpha_1 = \frac{1}{k(-cos\theta+2-e^{\theta}+4\theta^2cos\theta-4\theta^2e^{\theta})}$, also we expanded all p_{j+1} and p''_{j+1}, then after simplification we get:

$|p'_j(x) - y'_j(x)| = |(\alpha_1\alpha_2 + \alpha_1\alpha_3)p_j + (\alpha_1\alpha_3)\frac{2h^{1/2}}{\sqrt{\pi}}p_j^{1/2} + (\alpha_1\alpha_3 h - 1)p'_j + p''_j + (\alpha_1\alpha_3\frac{4h^{3/2}}{3\sqrt{\pi}} - \frac{2h^{1/2}}{\sqrt{\pi}}p_j^{3/2} + hp''_j(\theta 1) - h^2/2p''_j(\theta 2)|$, then in particular, we obtain:

$|p'_j(x) - f'(x)| = |E(x)| \leq Ch^3\omega(h),\ where\ Ch^3 = Max((\alpha_1\alpha_2 + \alpha_1\alpha_3)p_j + (\alpha_1\alpha_3)p_j^{1/2} + (\alpha_1\alpha_3 h - 1)p'_j + \left(\alpha_1\alpha_3\frac{4h^{3/2}}{3\sqrt{\pi}} - \frac{2h^{1/2}}{\sqrt{\pi}}\right)p_j^{3/2}$.

And similarly, can obtain to expanded p''_{j+1} and simplify the last step we get:

$$\left|p''_j(x) - f''(x)| \leq |(\alpha_1\alpha_2 + \alpha_1\alpha_3 - \alpha_4)p''_j + (\alpha_1\alpha_2 - \alpha_4)\frac{(2h^{1/2}}{\sqrt{\pi}}p_j^{5/2} + \alpha_1\alpha_2 hp'''_j(\theta_2) + \alpha_4 hp'''_j(\theta_1)\right|,$$

and simplifying with constants $C_1 = (\alpha_1\alpha_2 + \alpha_1\alpha_3 - \alpha_4), C_2 = (\alpha_1\alpha_2 - \alpha_4), C_3 = \alpha_1\alpha_2 + \alpha_4$ from equation 4, independent h, and can be write as:

$$\omega(p_j,\delta) \leq (C_1 + C_2h^(1/2) + C_3h)\omega(f,\delta) \leq Hh^4\omega(f,$$

$$whereHh^4 = Max(C_1 + C_2h^{1/2} + C_3h).$$

6. Numerical Discussion:

To demonstrate the usefulness of the developed non-polynomial spline, we solved two problems involving linear systems of fractional-order boundary value problems. All computations for these problems were carried out using the Maple software. The test cases confirm high accuracy and stability, particularly for problems involving memory effects and anomalous diffusion. Moreover, when the numerical solutions of the result are carried out, the results are depicted in Figures 1 and 2, where the phase portraits are displayed for with the exact solution regard to the governing for spline methods, also Tables 1 and 2 show the fractional scheme's numerical results for a range of x values for and $z_j(x)$. The numerical schemes' consistency analysis is displayed in Tables 3 and 4; the results show the total number of function evaluations indicating the maximum errors. These findings validate fractional non-polynomial spline interpolation as a reliable and practical tool for obtaining numerical solutions to systems of fractional differential equations.

Example 6.1 [22]: Consider the system of fractional differential equations

$D^{\alpha}v(x)$ = - v(x) + z(x) + 2/(Γ(3-α)) $x^{2-\alpha}$

$D^{\alpha}z(x)$ = v(x) - z(x) + $\frac{2}{\Gamma(3-\alpha)}$ $x^{2-\alpha}$, the exact solution is v(x) = x^2 and z(x) = x^2 v(0) = 0 and (0) = 0 , 0<x<0.1 , h = 0.01, N = 10?

Table 1. Numerical Solution of $v_j(x)$ to example 1 (α=0.5) and comparing in to Runge-Kutta Method.

x	Exact Value [22]	Approximate Value	Runge-Kutta
0.01	0.0001000000	0.0003075147	0.000070
0.02	0.0004000000	0.0006769686	0.000322
0.03	0.0009000000	0.001160271	0.000787
0.04	0.0016000000	0.001791187	0.001456
0.05	0.0025000000	0.002597025	0.002327
0.06	0.0036000000	0.003601370	0.003399
0.07	0.0049000000	0.004825293	0.004673
0.08	0.0064000000	0.006288026	0.006148
0.09	0.0081000000	0.008007375	0.007823
0.1	0.0100000000	0.010000000	0.009699

Table 2. Numerical Solution of $z_j(x)$ to example 1 (α=1.5) and comparing in to Runge-Kutta Method.

x	Exact Value [22]	Approximate Value	Runge-Kutta
0.01	0.0001000000	0.0004283868	0.000310
0.02	0.0004000000	0.000988961	0.000738
0.03	0.0009000000	0.001650801	0.001375
0.04	0.0016000000	0.002424149	0.002215
0.05	0.0025000000	0.003320234	0.003257
0.06	0.0036000000	0.004349869	0.004500
0.07	0.0049000000	0.005523267	0.005944
0.08	0.0064000000	0.006850074	0.007588
0.09	0.0081000000	0.008339424	0.009434
0.1	0.0100000000	0.0100000000	0.011480

Example 6.2: [23] Consider a nonlinear system of fractional delay differential equations

$f^(1/2)$ (x) = $e^{f(\frac{x}{2})}$ + w($\frac{x}{2}$ - e^x + $\frac{1}{x+1}$ - $\frac{x}{2}$ - 1 ,

$w^{\frac{1}{2}}$ (x) = $e^{f(x/3)}$ + 2 w(x) - $\frac{1}{3}$ lnn $\left(w(\frac{x}{2})\right)$ - 1

w(0) = 1, f(0) = 0, f(x) = ln(x+1) w(x) = e^{2x}, h = 0.05, N = 10.

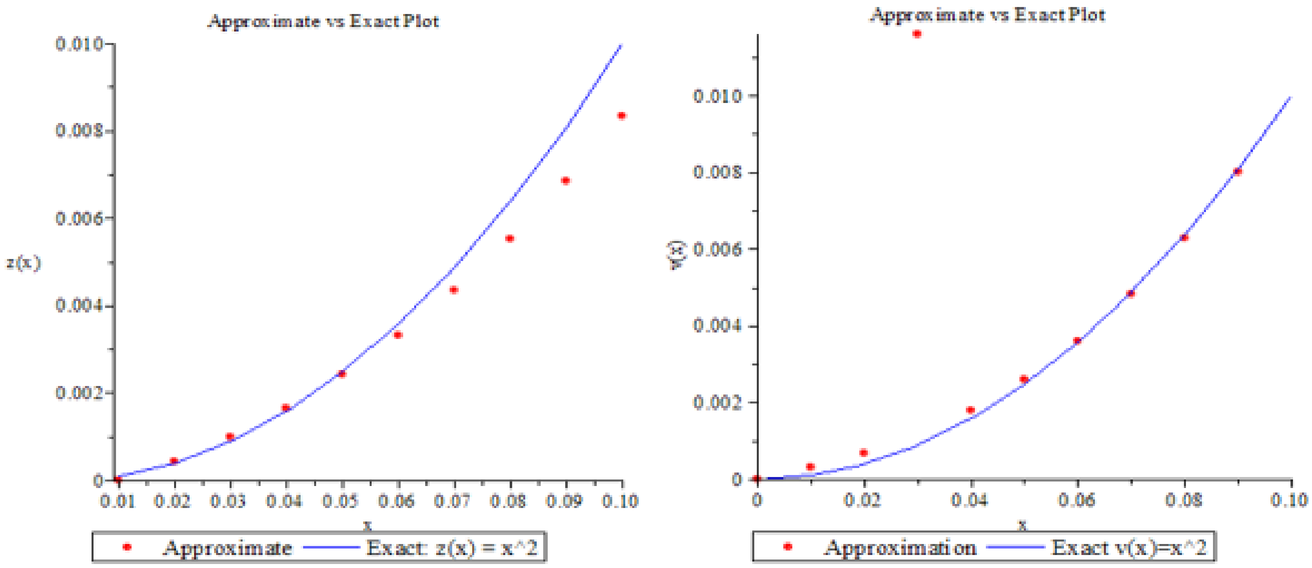

Figure 1. The graph between approximate solution and exact solution of example 1.

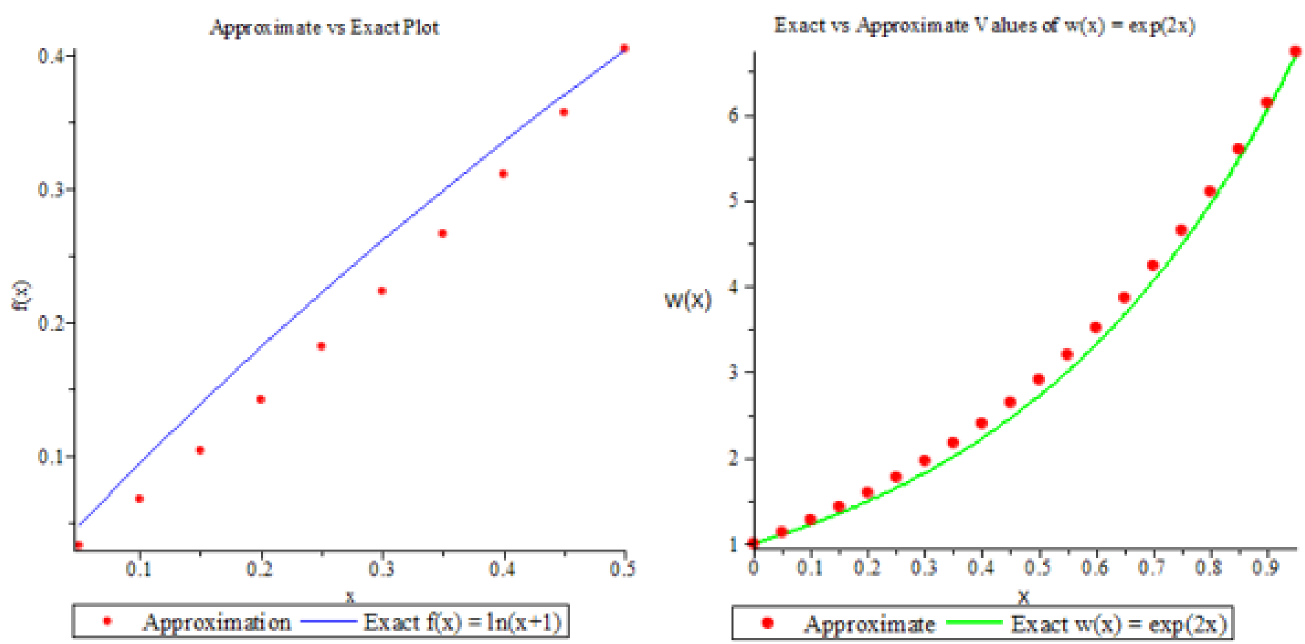

Figure 2. The graph between approximate solution and exact solution of example 2.

Table 3. Numerical Solution of w(x) to example 2 and comparing in to Runge-Kutta Method.

x	Exact Value [23]	Approximate Value	Runge-Kutta
0.05	1.105170918	1.133234123	1.843276
0.1	1.221402758	1.276080586	2.478709
0.15	1.349858808	1.429550326	3.181429
0.2	1.491824698	1.594760605	4.012364
0.25	1.648721271	1.772946185	5.016704
0.3	1.822118800	1.965471691	6.235964
0.35	2.013752707	2.173845265	7.716756
0.4	2.225540928	2.399733664	9.501427
0.45	2.459603111	2.644978938	11.656660
0.5	2.718281828	2.911616867	14.258881

Table 4. Numerical Solution of w(x) to example 2 and comparing in to Runge-Kutta Method.

x	Exact Value [23]	Approximate Value	Runge-Kutta
0.05	0.0487901642	0.03300774801	0.291621
0.1	0.0953101798	0.06775540997	0.626798
0.15	0.1397619424	0.1042124870	0.881327
0.2	0.1823215568	0.1423528917	1.156699
0.25	0.2231435513	0.1821541245	1.463927
0.3	0.2623642645	0.2235966260	1.796889
0.35	0.3001045925	0.2666632639	2.166354
0.4	0.3364722366	0.3113389227	2.594418
0.45	0.3715635564	0.3576101722	3.078762
0.5	0.4054651081	0.4054651081	3.653041

7. Conclusion:

This paper presents a useful numerical method for solving systems of fractional differential equations using non-polynomial spline interpolation. The method incorporates a Taylor series expansion based on the Caputo derivative formula, with the solution curve spatially interpolated using the Maple software. We compute the maximum errors between the exact solution and the numerical approach, as shown in Tables 1, 2, 3, 4 at uniformly distributed points $x \in [0, 0.1]$ and $x \in [0, 0.5]$,with respect to the step size h, also in [23], [22] comparing it to Runge-Kutta method, and the multiple solutions have been obtained represented in Figures 1 and 2. Consequently, the exact computations and the best approximate solutions for solving systems of FDEs are in both cases. The method provides sufficient precision and demonstrates innovation. Furthermore, the non-polynomial spline approach is applied to two numerical examples, with graphs confirming the accuracy and effectiveness of the method. The numerical results of the proposed approach validate the theoretical analysis.

Funding: None.

Data Availability Statement: All of the data supporting the findings of the presented study are available from corresponding author on request.

Declarations:
Conflict of interest: The authors declare that they have no conflict of interest.

Ethical approval: This research did not include any human subjects or animals, and as such, it was not necessary to obtain ethical approval.

Author Contributions: Paywast J. Hassan conducted the field data collection, performed the analysis, interpreted the results, wrote the manuscript, and carried out proofreading. Faraidun K. Hamasalh supervised the research and reviewed the manuscript.

References

[1] S. R. Jenaa and A. Senapati. Explicit and implicit numerical investigations of one-dimensional heat equation based on spline collocation and thomas algorithm soft computer. *Research squrae*, 28(20): 1, 2008, doi:10.21203/rs.3.rs-3800780/v1.

[2] Y. Gupta. Numerical solution of system of boundary value problems using b-spline with free parameter. *AIP Conference Proceedings*, 1802(1), 2017, doi:10.1063/1.4973256.

[3] A. Khalid, I. Haq, A. Rehan, H.M. Alshehri, and M.S. Osman. Employing cnps and cps approaches to calculate numerical roots of ninth-order linear and nonlinear boundary value problems. *International Journal of Modern Physics C*, 35(10), 2024, doi:10.1142/S0129183124501213.

[4] M. Dehgan and A. Saadatmandi. The numerical solution of a nonlinear system of second order boundary value problems using sinc-collocation method. *Mathematical and Computer Modelling*, 46(11-12): 1434–1441, 2007, doi:10.1016/j.mcm.2007.02.002.

[5] E. M. Gamel. Sinc-collocation method for solving linear and non-linear system of second-order boundary value problems. *Applied Mathematics*, 3(11): 1627–1633, 2012, doi:10.4236/am.2012.311225.

[6] J. F. Lu. Variational iteration method for solving a nonlinear system of second-order boundary value problems.

Computers and Mathematics with Applications, 54(7-8): 1133–1138, 2007, doi:10.1016/j.camwa.2006.12.060.

[7] M. Bataineh, S. M. Noorani, and I. Hashim. Modified homotopy analysis method for solving system of second-order bvps. *Communications in Nonlinear Science and Numerical Simulation*, 14(2): 430–442, 2009, doi:10.1016/j.cnsns.2007.09.012.

[8] Karwan H. F. Jwamer, Faraidun K. Hamasalh, and Ridha G. Karem. The study focuses on lacunary interpolation and error estimations by spline function. *Journal of Basrah Researches*, 50(2): 146, 2024, doi:10.56714/bjrs.50.2.12.

[9] A. Chaurasia, P.C. Srivastava, and Y. Gupta. Solution of higher order boundary value problems by spline methods. *AIP Conference Proceedings*, 1897: 020018, 2017, doi:10.1063/1.5008697.

[10] M. Dehghan and M. Lakestani. Numerical solution of non-linear system of second-order boundary value problems using cubic b-spline scaling functions. *International Journal of Computer Mathematics*, 85(9), 2008, doi:10.1080/00207160701534763.

[11] WT. Aniley and GF. Duressa. A novel fitted numerical scheme for time-fractional singularly perturbed convection-diffusion problems with a delay in time via cubic b-spline approach. *Journal of Mathematical Modeling*, 12(2): 215–231, 2024, doi:10.22124/jmm.2023.25969.2303.

[12] S. A. Khuri and A. Sayfy. Collocation approach for the numerical solution of a generalized system of second-order boundary-value problems. *Applied Mathematical Sciences*, 3(45): 2227–2239, 2009.

[13] A. Heilat, N. Hamid, and A.I.M. Ismail. Extended cubic b-spline method for solving a linear system of second-order boundary value problems. *Springer Plus*, 5(1): 1–18, 2016, doi:10.1186/s4006401629364.

[14] J. Goh, A. A. Majid, and A.I.M. Ismail. Extended cubic uniform b-spline for a class of singular boundary value problems. *Science Asia*, 37(1), 2011,doi:10.2306/scienceasia1513-1874.2011.37.079.

[15] T. H. Fay. Coupled spring equations. *International Journal of Mathematical Education in Science and Technology*, 34(1), 2003, doi:10.1080/0020739021000029258.

[16] Milici, Constantin, Gheorghe Drgnescu, and J.Tenreiro Machado. Introduction to fractional differential equations. *Springer*, 25, 2018.

[17] Chaurasia, Anju, Yogesh Gupta, and Prakash C. Srivastava. Numerical scheme based on non-polynomial spline functions for the system of second order boundary value problems arising in various engineering applications. *Journal of Applied and Computational Mechanics*, 8(1): 144–152, 2022, doi:10.22055/JACM.2020.32435.2012.

[18] D.C. Lay, S. R. Lay, and J.J. McDonald. *Linear Algebra and its Applications*. Pearson, Boston, 5^{th} edition, 2016.

[19] Faraidun K. Hamasalh and Mizhda A. Headayat. The applications of non-polynomial spline to the numerical solution for the fractional differential equations. *AIP Conference Proceedings*, 2334: 060014, 2021, doi:10.1063/5.0042319.

[20] Homan Emadifar and Faraidun K. Hamasalh. Non-polynomial spline interpolation for solving fractional subdiffusion equations. *Mathematical Problems in Engineering*, 2022, doi:10.1155/2022/7354121.

[21] Abbas Y. Albayati, Rostam K. Saeed, and Faraidun K. Hamasalh. The existence, uniqueness and error bounds of approximation splines interpolation for solving second-order initial value problems. *Journal of Mathematics and Statistics*, 5(2): 123–129, 2009.

[22] Yadav, Sandeep Kumar, and Giriraj Methi. Application of fractional differential transform method and bell polynomial for solving system of fractional delay differential equations. *Partial Differential Equations in Applied Mathematics*, 12, 2024.

[23] Mahmoud N. Sherif and A. F. Nawal. Alsubaie numerical solution of system of fractional differential equation polynomial spline functions. *International Journal of Difference Equations*, 15(1): 95–105, 2020, doi:10.37622/IJDE/15.1.2020.95-105.

سبلاين كسري عددي لحل نظام المعادلات التفاضلية الكسرية

[1] پيوست جبار حسن ، [2*] فريدون قادر حمه صالح

[1] قسم الرياضيات ،جامعة السليمانية ، كلية التربية ، السليمانية ، العراق.

[2] معهد بكرجو التقني ،جامعة السليمانية التقنية ، السليمانية ، العراق.

* الباحث المسؤول: *faraidun.hamasalh@spu.edu.iq*

الخلاصة

تعتبر المعادلات التفاضلية ذات الترتيب الكسري أساسية في المجالات العلمية والهندسية المتنوعة، بما في ذلك المجتمع الاحصائي والتحكم الأمثل والفيزياء. تقدم هذه الورقة طريقة غير متعددة الحدود لحلها عدديًا، حيث تنشئ نظامًا خطيًا من المعادلات الجبرية في تمثيل معادلات التكرار ثلاثية الحدود، والتي يتم حلها باستخدام خوارزمية حذف. تعتمد نجاح الطريقة على تقديرات الخطأ و ترتيب التقارب، ويتم إثبات أدائها من خلال أمثلة عددية ورسومية. يتم تقديم شرح دقيق للإجراء الرياضي، بالإضافة إلى أمثلة رسومية وعددية لحل عدة أمثلة. تؤكد النتائج أن الطريقة المقترحة تحقق دقة واعتمادية أفضل مقارنة بالتقنيات الحالية.

الكلمات الدالة : مشكلة القيمة الحدية ، دالة *Gamma* ، مشتقة كابوتو الكسرية ، مشتقة ريمانليوفيل الكسرية ، تحليل التقارب، تقدير الأخطاء.

التمويل: لايوجد.

بيان توفر البيانات: جميع البيانات الداعمة لنتائج الدراسة المقدمة يمكن طلبها من المؤلف المسؤول.

اقرارات:

تضارب المصالح: يقر المؤلفون أنه ليس لديهم تضارب في المصالح.

الموافقة الأخلاقية: لم ينضمن هذه الدراسة أي أشخاص أو حيوانات، وبالتالي لم يكن من الضروري الحصول على موافقة أخلاقية.

مساهمات المؤلفين: قام بيوست جبار حسن بجمع البيانات الميدانية، وإجراء التحليل، وتفسير النتائج، وكتابة المسودة، وتدقيقها. أشرف فريدون قادر حمه صالح على البحث وراجع المسودة.