

Numerical Solution of A Class of Fourth-Order Differential Equations with Initial Conditions

Edwin Frank Nsien¹, Ubon Akpan Abasiokwere^{2*}

¹Department of Statistics, University of Uyo, Uyo, Akwa Ibom State, Nigeria

²Department of Mathematics, University of Uyo, Uyo, Akwa Ibom State, Nigeria

DOI: <https://doi.org/10.51583/IJLTEMAS.2025.140500012>

Received: 29 April 2025; Accepted: 13 May 2025; Published: 31 May 2025

Abstract: This paper presents the Runge-Kutta (RK4) method of order four for solving initial value problems of fourth-order ordinary differential equations. The proposed method is derived and is observed to be efficient and practically well-suited for handling high-order initial value problems. Two modal examples (linear and non-linear) are presented to demonstrate the reliability, accuracy and easy implementability of the method. The results obtained show a high level of accuracy when compared with the analytical solution.

Keywords: Runge-Kutta, ordinary differential equations, initial value problem, finite difference method, numerical approximation

I. Introduction

Numerical methods are well known in the field of science and engineering to solve various linear and non-linear ordinary differential equations (ODEs). Some of the well-known methods developed to solve the ODEs include but are not limited to Euler's method, Picard's method of successive approximation, Taylor's method, etc. These methods have gone through various stages of development with the advancement of the programming language and for various applications to real life applications. We can speak of the development in the Runge-Kutta (R-K) method since its existence. This method is developed for various orders of convergence, ranging from order 1 to order 4 and above.

Runge-Kutta methods are a popular class of numerical methods for solving ordinary differential equations. Every Runge-Kutta method is characterized by two basic parameters: its order, which measures the accuracy of the solution it produces, and its number of stages, which measures the amount of computational work it requires. The primary goal in constructing Runge-Kutta methods is to maximize order using a minimum number of stages. However, high-order Runge-Kutta methods are difficult to construct because their parameters must satisfy an exponentially large system of polynomial equations. [10] presented the first known 10th-order Runge-Kutta method with only 16 stages, breaking a 40-year standing record for the number of stages required to achieve 10th-order accuracy. Again, very recently, [9] presented a new class of fractional order Runge-Kutta (FORK) methods for numerically approximating the solutions of fractional differential equations (FDEs) by constructing explicit and implicit FORK methods for FDEs using the Caputo generalized Taylor series formula.

In all these available versions of R-K method, the method of fourth order becomes more famous because of its convergent properties [4], [5]. This method is of excessive practical significance with the mentioned accuracy and numerical stability in comparison with the well-known Euler's method. The Runge-Kutta method as a numerical technique for solving ordinary differential equations has found application in optimization research, particularly in developing algorithms like the Runge-Kutta Optimizer (RUN) that leverages the slope calculations from the method to explore search spaces and find optimal solutions [1]

The Taylor series method, which is well known for solving differential equations numerically, becomes ineffective for problems that involve higher-order derivatives. Also, the well-known Euler's method is less effective in practical problems since it requires the step size (h) to be small for obtaining reasonable accuracy [6]. The merit of the Runge-Kutta (R-K) method lies in its ability to achieve high accuracy without requiring the explicit calculation of higher-order derivatives, as is needed in methods like the Taylor series expansion. R-K methods offer a more practical and efficient approach for solving differential equations, especially when dealing with complex algebraic equations where higher derivatives might be difficult to obtain.

According to Arumugan *et al*, problems in science and technology can be solved by reducing them to differential equations satisfying certain conditions. Analytic methods can be applied to solve many standard types of differential equations. However, some differential equations generated from physical problems are so complex that they can only be solved efficiently by numerical methods.

The Derivative Of The Fourth-Order Runge-Kutta Method

We shall derive the formula of the fourth-order Runge kutta method to obtain an approximate numerical solution of the first-order differential equation $y' = f(x, y)$ with the initial condition $y(x_0) = y_0$ and it is assumed that is not a singular point.

Let us take the first order linear ordinary differential equation

$$y' = \frac{dy}{dx} = f(x, y); y(x_0) = y_0. \quad (1)$$

Let $h = x_1 - x_0$, from Taylor's series expansion, we have

$$y(x+h) = y(x) + hy'(x) + \frac{h^2}{2!} y''(x) + \frac{h^3}{3!} y'''(x) + \dots$$

or

$$y(x+h) - y(x) = hy'(x) + \frac{h^2}{2!} y''(x) + \frac{h^3}{3!} y'''(x) + \dots \quad (2)$$

Differentiating eqn (1) partially with respect to variables x and y we get

$$y' = f'(x, y) = f_x + ff_y$$

$$y'' = f_{xx} + ff_{xy} + f_x f_y + ff_{xy} + ff_y f_y + fff_{yy}$$

$$= f_{xx} + ff_{xy} + f_x f_y + ff_{xy} + ff_y^2 + f^2 f_{yy}$$

$$y''' = f'''(x, y) = f_{xxx} + 2ff_{xy} + f^2 f_{yy} + f_x f_y + ff_y^2$$

$$y^{iv} = f_{xxx} + 2(ff_{xy} + f_x f_y) + 2f_x f_{yy} + f^2 f_{xyy} + f_{xx} f_y + f_x f_{xy} + f_x f_y^2 + ff_{xy}^2 + ff_{xxy}$$

$$+ 2(ff_y f_{xy} + 2fff_{xyy}) + (ff^2 f_{yyy} + 2ff_{yyy} + ff_{xy} f_y + f_x ff_{yy} + ff_y f_y^2 + f_y (ff_{yy}))$$

$$= f_{xxx} + 2ff_{xxy} + 2f_x f_{xy} + 2f_x f_{yy} + f^2 f_{xyy} + f_{xx} f_y + f_x f_{xy} + f_x f_y^2 + ff_{xy}^2 + ff_{xxy}$$

$$+ 2ff_y f_{xy} + 2f^2 f_{xyy} + f^3 f_{yyy} + 2ff_{yyy} + ff_x f_y + ff_x f_{yy} + ff_y f_y^2 + f^3 f_{yyy}$$

Reorganizing we have that

$$y^{iv} = f'''(x, y) = (f_{xxx} + 3ff_{xxy} + f^2 f_{xyy} + f^3 f_{yyy}) + f_y (f_{xx} + 2ff_{xy} + f^2 f_{yy}) + 3(f_x + ff_y)(f_{xy} + ff_{yy}) + f_y^2 (f_x + ff_y)$$

Let's introduce the following convenient form

$$F_1 = f_x + ff_y, \quad F_2 = f_{xx} + 2ff_{xy} + f^2 f_{yy}, \quad F_3 = f_{xxx} + 3ff_{xxy} + f^2 f_{xyy} + f^3 f_{yyy}$$

Then we have

$$y' = f, \quad y'' = F_1, \quad y''' = F_2 + f_y F_1, \quad y^{iv} = F_3 + f_y F_2 + 3F_1(f_{xy} + ff_{yy}) + F_1 f_y^2$$

Substituting y', y'', y''', y^{iv} into eqn (2), we obtain

$$y(x+h) - y(x) = hf + \frac{h^2}{2!} F_1 + \frac{h^3}{3!} (F_2 + f_y F_1) + \frac{h^4}{4!} \{F_3 + f_y F_2 + 3F_1(f_{xy} + ff_{yy}) + F_1 f_y^2\} \quad (3)$$

Now, we seek a fourth order formula in order to develop the Runge's Kutta formula to find the coefficients a, b, c, d, m, n and p from below.

$$\begin{aligned}
 k_1 &= hf(x, y) = hf \\
 k_2 &= hf(x + mh, y + mk_1) \\
 k_3 &= hf(x + nh, y + nk_2) \\
 k_4 &= hf(x + ph, y + pk_3)
 \end{aligned} \tag{4}$$

We express Δy

$$\Delta y = y(x, y) - y(x) = ak_1 + bk_2 + ck_3 + dk_4 \tag{5}$$

Since we already know Taylor's series expansion for one variable, we apply the same idea for the computation for two variables up to the third order and will obtain

$$\begin{aligned}
 f(x+h, y+k) &= f(x, y) + f_x(x, y)h + f_y(x, y)k + \frac{h^2}{2!} f_{xx}(x, y) + f_{xy}(x, y)kh + \frac{k^2}{2!} f_{yy}(x, y) \\
 &\quad + \frac{h^2}{2!} f_{yy}(x, y) + \frac{h^3}{3!} f_{xxx}(x, y) + \frac{k^3}{3!} f_{yyy}(x, y) + \frac{h^2 k}{2} f_{xx}(x, y) f_y(x, y) \\
 &\quad + \frac{hk^2}{2} f_x(x, y) f_{yy}(x, y)
 \end{aligned}$$

Replacing $k = mk_1$ and $h = mh$, we have

$$\begin{aligned}
 k_2 &= h \{ f(x, y) + f_x(x, y)mh + f_y(x, y)mk_1 + \frac{1}{2} f_{xx}(x, y)m^2h^2 + f_{xy}(x, y)m^2hk_1 \\
 &\quad + \frac{1}{2} f_{yy}(x, y)m^2k_1^2 + \frac{1}{6} f_{xxx}(x, y)m^3h^3 + \frac{1}{6} f_{yyy}(x, y)m^3k_1^3 + \frac{1}{2} f_{xx}f_y(x, y)m^3h^2k_1 \\
 &\quad + \frac{1}{2} f_xf_{yy}(x, y)m^3hk_1^2 \}.
 \end{aligned}$$

This yields

$$\begin{aligned}
 k_2 &= h[f + f_xmh + f_ymk_1 + \frac{1}{2} (f_{xx}m^2h^2 + 2f_{xy}m^2h^2f + f_{yy}m^2h^2f^2) + \frac{1}{6} (f_{xxx}m^3h^3 + f_{yyy}m^3h^3f^3 + \\
 &\quad f_y m^3h^3f + 3f_xf_{yy}m^3h^3f^2)] \\
 k_2 &= h[f + mh(f_x + ff_y) + \frac{1}{2} m^2h^2(f_{xx} + 2ff_{xy} + f^2f_{yy}) + \frac{1}{6} m^3h^3(f_{xxx} + f^3f_{yyy} + 3ff_{xy} + 3f^2f_{xyy})]
 \end{aligned}$$

Recall that $F_1 = f_x + ff_y$

$$F_2 = f_{xx} + 2ff_{xy} + f^2f_{yy}$$

$$F_3 = f_{xxx} + 3ff_{xyy} + f^2f_{xyy} + f^3f_{yyy}$$

Substituting the F_1, F_2, F_3 appropriately will yield

$$k_2 = h[f + mhF_1 + \frac{1}{2} m^2h^2F_2 + \frac{1}{6} m^3h^3F_3 + \dots]$$

Now for k_3 we will proceed as thus

$$k_3 = h[f(x, y) + f_x(x, y)nk_2 + \frac{1}{2}f_{xx}(x, y)n^2h^2 + f_{xy}(x, y)n^2hk_2 + \frac{1}{2}f_{yy}(x, y)n^2k_2^2 + \frac{1}{6}f_{xxx}(x, y)n^3h^3 + \frac{1}{6}f_{yyy}(x, y)n^3k_2^3 + \frac{1}{2}f_{xxy}(x, y)n^3h^2k_2 + \frac{1}{2}f_{xyy}(x, y)n^3hk_2^2]$$

Substituting the value of k_2 , we have

$$k_3 = h[f(x, y) + f_x(x, y)nh + f_y(x, y)n\{hf + mh^2F_1 + \frac{1}{2}m^2h^3F_3 + \frac{1}{6}m^3h^4F_3 + \dots\} + \frac{1}{2}f_{xx}(x, y)n^2h^2 + f_{xy}(x, y)n^2h\{hf + mh^2F_1 + \frac{1}{2}m^2h^3F_3 + \frac{1}{6}m^3h^4F_3 + \dots\} + \frac{1}{2}f_{yy}n^2\{hf + mh^2F_1 + \frac{1}{2}m^2h^3F_2 + \frac{1}{6}m^3h^4F_3 + \dots\}^2 + \frac{1}{6}f_{xxx}(x, y)n^3h^3 + \frac{1}{6}f_{yyy}n^3\{hf + mh^2F_1 + \frac{1}{2}m^2h^3F_2 + \frac{1}{6}m^3h^4F_3 + \dots\}^3 + \frac{1}{2}f_{xxy}n^3h^2\{hf + mh^2F_1 + \frac{1}{2}m^2h^3F_2 + \frac{1}{6}m^3h^4F_3 + \dots\} + \frac{1}{2}f_{xyy}n^3h\{hf + mh^2F_1 + \frac{1}{2}m^2h^3F_2 + \frac{1}{6}m^3h^4F_3 + \dots\}^2]$$

$$k_3 = h[f + f_xnh + nhff_y + nmh^2f_yF_1 + \frac{1}{2}nm^2h^3f_yF_2 + \frac{1}{6}m^3h^4nf_yF_3 + \frac{1}{2}f_{xx}n^2h^2 + n^2h^2ff_{xy} + mh^3n^3f_{xy}F_1 + \frac{1}{2}m^2h^4n^2f_{xy}F_2 + \frac{1}{6}m^3h^5n^2f_{xy}F_3 + \frac{1}{2}n^2h^2f^2f_{yy} + mh^3n^2ff_{yy}F_1 + \frac{1}{2}n^2m^2h^4ff_{yy}F_2 + \frac{1}{6}n^2m^3h^5ff_{yy}F_3 + \frac{1}{2}n^2m^2h^4f_{yy}F_1^2 + \frac{1}{2}n^2m^3h^5f_{yy}F_1F_2 + \frac{1}{6}n^2m^4h^6f_{yy}F_1F_3 + \frac{1}{2}n^2m^4h^6f_{yy}F_2^2 + \frac{1}{2}n^2m^5h^7f_{yy}F_2F_3 + \frac{1}{6}n^2m^6h^8F_3^2f_{yy} + \frac{1}{6}f_{xxx}n^3h^3 + \frac{1}{6}n^2h^3f^3f_{yyy} + \frac{1}{3}n^2mh^4f^2f_{yyy}F_1 + \frac{1}{4}n^2m^2h^5f^2f_{yyy}F_2 + \frac{1}{18}h^2m^3n^6f^2f_{yyy}F_3 + \frac{1}{3}n^2m^2h^5ff_{yyy}F_1^2 + \frac{1}{4}h^2m^3h^6ff_{yyy}F_1F_2 + \frac{1}{18}n^2m^4h^6ff_{yyy}F_1F_3 + \frac{1}{6}n^2m^4h^4ff_{yyy}F_2^2 + \frac{1}{18}n^2m^5h^8ff_{yyy}F_2F_3 + \frac{1}{18}h^2m^6h^9ff_{yyy}F_3^2 + \frac{1}{18}n^2m^4h^7ff_{yyy}F_1F_3 + \frac{1}{6}n^2m^3h^6f_{yyy}F_1^3 + \frac{1}{4}n^2m^4h^7f_{yyy}F_2F_1^2 + \frac{1}{12}n^2m^5h^8f_{yyy}F_3F_1^2 + \frac{1}{6}n^2m^5h^8f_{yyy}F_1F_2^2 + \frac{2}{9}n^2m^6h^9f_{yyy}F_1F_2F_3 + \frac{7}{108}n^2m^7h^{10}f_{yyy}F_1F_3^2 + \frac{1}{12}n^2m^4h^6ff_{yyy}F_2^2 + \frac{1}{12}n^2m^5h^{10}f_{yyy}F_1F_2^2 + \frac{1}{12}n^2m^6h^9f_{yyy}F_2F_2^2 + \frac{1}{9}n^2m^7h^{10}f_{yyy}F_3F_2^2 + \frac{1}{18}n^2m^8h^{11}f_{yyy}F_2F_3^2 + \frac{1}{36}n^2m^3h^4f^2f_{yyy}F_3]$$

$$\begin{aligned}
 & + \frac{1}{36} n^2 m^4 h^7 f_{yyy} F_1 F_3 + \frac{1}{324} m^6 h^6 f F_3^2 f_{yyy} + \frac{1}{324} h^2 m^9 h^{12} f_{yyy} F_3^2 + \frac{1}{2} n^3 h^3 ff_{xy} + \frac{1}{2} n^3 h^4 m f_{xx} F_1 \\
 & + \frac{1}{4} n^3 h^4 m^2 f_{xy} F_2 + \frac{1}{12} n^3 h^6 m^3 f_{xy} F_3 + \frac{1}{2} n^3 h^3 f^2 f_{xy} + n^3 h^5 m ff_{xy} F_1 \\
 & + \frac{1}{2} n^3 h^6 m^2 ff_{xy} F_2 + \frac{1}{6} n^3 h^7 m^3 ff_{xy} F_3 + \frac{1}{2} n^3 h^6 m^2 f_{xy} F_1^2 + \frac{1}{2} n^3 h^7 m^3 f_{xy} F_1 F_2 \\
 & + \frac{1}{6} n^3 h^8 m^4 f_{xy} F_1 F_3 + \frac{1}{2} n^3 h^8 m^4 f_{xy} F_2^2 + \frac{1}{2} n^3 h^9 m^5 f_{xy} F_2 F_3 + \frac{1}{3} n^3 h^{10} m^5 f_{xy} F_3^2]
 \end{aligned}$$

Factorising h , h^2 , h^3 , we have

$$nh(f_x + ff_y) = nhF_1 \quad (6)$$

$$\frac{1}{2} h^2 [n^2 F_2 + 2nmf_y F] \quad (7)$$

$$\frac{1}{6} h^3 [n^3 F_3 + 3nm^2 f_y F + 6mn^2 F_1 (f_{xy} + ff_{yy}) + \dots] \quad (8)$$

Putting together (6), (7), (8) we would have

$$k_3 = h[f + nhF_1 + \frac{1}{2} h^2 \{n^2 F_2 + 2nmf_y F\} + \frac{1}{6} h^3 \{n^3 F_3 + 3nm^2 f_y F + 6mn^2 F_1 (f_{xy} + ff_{yy}) + \dots\}]$$

Now for k_4

$$k_4 = hf(x + ph, y + pk_3)$$

From Taylor series expansion again ,

$$\begin{aligned}
 k_4 = h[& f + f_x ph + f_y pk_3 + \frac{1}{2} f_{xx} p^2 h^2 + f_{xy} p^2 h k_3 + \frac{1}{2} f_{yy} p^2 k_3^2 + \frac{1}{6} f_{xxx} p^3 h^3 + \frac{1}{6} f_{yyy} p^3 k_3^3 \\
 & + \frac{1}{2} f_{xxy} p^3 h^2 k_3 + \frac{1}{2} f_{xyy} p^3 h k_3^2
 \end{aligned}$$

Rearranging we have that

$$\begin{aligned}
 k_4 = h[& f + phf_x + pf_y k_3 + \frac{1}{2} f_{xx} p^2 h^2 + f_{xy} p^2 h k_3 + \frac{1}{2} f_{yy} p^2 k_3^2 + \frac{1}{6} f_{xxx} p^3 h^3 + \frac{1}{6} f_{yyy} p^3 k_3^3 \\
 & + \frac{1}{2} f_{xxy} p^3 h^2 k_3 + \frac{1}{2} f_{xyy} p^3 h k_3^2]
 \end{aligned}$$

Substituting k_3 , we have

$$k_3 = h[f + nhF_1 + \frac{1}{2} h^2 \{n^2 F_2 + 2nmf_y F\} + \frac{1}{6} h^3 \{n^3 F_3 + 3nm^2 f_y F + 6mn^2 F_1 (f_{xy} + ff_{yy}) + \dots\}]$$

Then we would have

$$\begin{aligned}
 k_4 = & h[f + phf_x + pf_y(hf + nh^2F_1 + \frac{1}{2}h^3\{n^2F_2 + 2nmf_yF_1\} + \frac{1}{6}h^4\{n^3F_3 + 3nm^2f_yF \\
 & + 6mn^2F_1(f_{xy} + ff_{yy})\} + \dots) + \frac{1}{2}f_{xx}p^2h^2 + f_{xy}p^2h(hf + nh^2F_1 + \frac{1}{2}h^3\{n^2F_2 \\
 & + 2nmf_yF_1\} + \frac{1}{6}h^4\{n^3F_3 + 3nm^2f_yF + 6mn^2F_1(f_{xy} + ff_{yy})\} + \dots) \\
 & + \frac{1}{2}f_{yy}p^2(hf + nh^2F_1 + \frac{1}{2}h^3\{n^2F_2 + 2nmf_yF_1\} + \frac{1}{6}h^4\{n^3F_3 + 3nm^2f_yF \\
 & + 6mn^2F_1(f_{xy} + ff_{yy})\} + \dots)^2 + \frac{1}{6}f_{xxx}p^3h^3 + \frac{1}{6}f_{yyy}p^2(hf + nh^2F_1 + \frac{1}{2}h^3\{n^2F_2 \\
 & + 2nmf_yF_1\} + \frac{1}{6}h^4\{n^3F_3 + 3nm^2f_yF + 6mn^2F_1(f_{xy} + ff_{yy})\} + \dots)^3 \\
 & + \frac{1}{2}f_{xxy}p^3h^3(hf + nh^2F_1 + \frac{1}{2}h^3\{n^2F_2 + 2nmf_yF_1\} + \frac{1}{6}h^4\{n^3F_3 + 3nm^2f_yF \\
 & + 6mn^2F_1(f_{xy} + ff_{yy})\} + \dots) + \frac{1}{2}f_{xxy}p^2h(hf + nh^2F_1 + \frac{1}{2}h^3\{n^2F_2 + 2nmf_yF_1\} \\
 & + \frac{1}{6}h^4\{n^3F_3 + 3nm^2f_yF + 6mn^2F_1(f_{xy} + ff_{yy})\} + \dots)
 \end{aligned}$$

Opening the bracket we would have

$$\begin{aligned}
 k_4 = & h[f + phf_x + phff_y + nph^2f_yF_1 + \frac{1}{2}h^3pf_y(n^2F_2 + 2nmf_yF_1) + \frac{1}{6}h^4pf_y(n^3F_3 + 3nm^2f_yF_2 \\
 & + 6mn^2F_1(f_{xy} + ff_{yy})) + \frac{1}{2}f_{xx}p^2h^2 + p^2h^2ff_{xy} + np^2h^3f_{xy}F_1 \\
 & + \frac{1}{2}h^4p^2f_{xy}(n^2F_2 + 2nmf_yF_1) + \frac{1}{6}h^5p^2f_{xy}(n^3F_3 + 3nm^2f_yF_2 + 6mn^2F_1(f_{xy} + ff_{yy})) \\
 & + \frac{1}{2}h^2p^2f^2f_{yy} + nh^3p^2ff_{yy}F_1 + \frac{1}{2}h^4p^2ff_{yy}(n^2F_2 + 2nmf_yF_1) + \frac{1}{6}h^5p^2ff_{yy}(n^3F_3 \\
 & + 3nm^2f_yF_2 + 6mn^2F_1(f_{xy} + ff_{yy})) + \frac{1}{2}n^2h^4p^2F_1^2 + \frac{1}{2}h^5p^2nf_{yy}F_1(n^2F_2 + 2nmf_yF_1) \\
 & + \frac{1}{6}h^6p^2nf_{yy}F_1(n^3F_3 + 3nm^2f_yF_2 + 6mn^2F_1(f_{xy} + ff_{yy})) + \frac{1}{8}h^6p^2f_{yy}(n^2F_2 + 2nmf_yF_1)^2 \\
 & + \frac{1}{24}h^7p^2(n^3F_3 + 3nm^2f_yF_2 + 6mn^2F_1(f_{xy} + ff_{yy})) \\
 & + \frac{1}{24}h^7p^2f_{yy}(n^2F_2 + 2nmf_yF_1)(n^3F_3 + 3nm^2f_yF_2 + 6mn^2F_1(f_{xy} + ff_{yy})) \\
 & + \frac{1}{36}h^8p^2f_{yy}(n^3F_3 + 3nm^2f_yF_2 + 6mn^2F_1(f_{xy} + ff_{yy}))^2 + \frac{1}{6}h^3f^3p^3f_{yyy} + \frac{1}{2}h^3p^3ff_{xxy}
 \end{aligned}$$

$$+\frac{1}{2}p^3h^4nf_{xy}F_1+\frac{1}{4}p^3f_{xy}(n^2F_2+2nmf_yF_1)$$

$$\frac{1}{12}h^6p^3f_{xy}(n^3F_3+3nm^2f_yF_2+6mn^2F_1(f_{xy}+ff_{yy}))+\frac{1}{2}p^3h^3f^2f_{xy}+\dots$$

Again, factorising h, h^2, h^3 , we have

$$\begin{aligned} & f + ph(f_x + ff_y) \\ & f + phF_1 \end{aligned} \tag{9}$$

$$\begin{aligned} & nph^2f_yF_1 + \frac{1}{2}f_{xx}p^2h^2 + p^2h^2ff_{xy} + \frac{1}{2}h^2p^2f^2f_{yy} \\ & \frac{1}{2}[2nph^2f_yF_1 + f_{xx}p^2h^2 + 2p^2h^2ff_{xy} + h^2p^2f^2f_{yy}] \\ & \frac{1}{2}h^2[2npf_yF_1 + p^2(f_{xx} + 2ff_{xy} + f^2f_{yy})] \\ & \frac{1}{2}h^2[2npf_yF_1 + p^2F_2] \\ & \frac{1}{2}h^3pf_y\{n^2F_2 + 2nmf_yF_1\} + np^2h^3f_{xy}F_1 + nh^3p^2ff_{yy}F_1 + \frac{1}{6}h^3f^3p^3f_{yyy} \\ & + \frac{1}{2}h^3p^3ff_{xy} + \frac{1}{6}f_{xxx}p^3h^3 + \frac{1}{2}p^3h^3f^2f_{xyy} \end{aligned} \tag{10}$$

Factorizing we have thus

$$\begin{aligned} & \frac{1}{2}h^3[3pf_y(n^2F_2 + 2nmf_yF_1) + 6np^2f_{xy}F_1 + 6np^2ff_{yy}F_1 + p^3f^3f_{yyy} \\ & + 3p^3ff_{xy} + 3p^3f^2f_{xyy} + p^3f_{xxx}] \\ & = \frac{1}{2}h^3[3n^3pf_yF_2 + 6np^2F_1(f_{xy} + ff_{yy}) + p^3(f_{xxx} + 3ff_{xy} + 3f^2f_{xyy} + f^3f_{yyy})] \\ & = \frac{1}{2}h^3[3n^3pf_yF_2 + 6np^2F_1(f_{xy} + ff_{yy}) + p^3F_3 + 6pnmf_y^2F_1] \end{aligned} \tag{11}$$

Hence collecting (9),(10),(11)

$$k_4 = h[f + phF_1 + \frac{1}{2}h^2(p^2F_2 + 2npf_yF_1) + \frac{1}{6}h^3\{p^3F_3 + 3n^2pf_yF_2 + 6np^2F_1(f_{xy} + ff_{yy}) + 6mnpFf\}]$$

Substituting the values of k_1, k_2, k_3, k_4 appropriately into (5), we obtain

$$\begin{aligned} y(x+h) - y(x) &= ahf + bh[f + \frac{1}{2}m^2h^2F_2 + \frac{1}{6}m^3h^3F_3 + \dots] + ch[f + nhF_1 \\ &+ \frac{1}{2}h^2(n^2F_2 + 2mnf_yF_1) + \frac{1}{6}h^3(n^3F_3 + 3m^2nf_yF_2 \end{aligned}$$

$$+6mn^2F_1(f_{xy} + ff_{yy})) + \dots] + dh[f + phF_1 + \frac{1}{2}h^2(p^2F_2 + 2npf_yF_1) + \frac{1}{6}h^3(p^3F_3 + 3n^2pf_yF_2 + 6np^2F_1(f_{xy} + ff_{yy})) + 6mnpF_1f_y^2 + \dots]$$

This can be represented as

$$y(x+h) - y(x) = (a+b+c+d)hf + (bm+cn+dp)h^2F_1 + (bm^2+cn^2+dp^2)\frac{h^3F_2}{2} + (bm^3+cn^3+dp^3)\frac{h^4F_3}{6} + (cmn+dn p)h^3f_yF_1 + (cm^2n+dn^2p)\frac{h^4f_yF_2}{2} + (cm^2+dn p^2)h^4F_1(f_{xy} + ff_{yy}) + dmnpF_1f_y^2 + \dots$$

Comparing this representation with what we have below

$$y(x+h) - y(x) = hf + \frac{h^2}{2}F_1 + \frac{h^3}{6}(F_1 + f_yF_1) + \frac{h^4}{24}\{F_3 + f_yF_2 + 3F_1(f_{xy} + ff_{xy}) + F_1(f_{xy} + ff_{yy}) + F_1f_y^2\}$$

$$a+b+c+d=1 \quad bm+cn+dp=\frac{1}{2} \quad bm^2+cn^2+dp^2=\frac{1}{3}$$

$$bm^3+cn^3+dp^3=\frac{1}{4} \quad cmn+dn p=\frac{1}{6} \quad cm^2n+dn^2p=\frac{1}{12}$$

$$cmn^2+dn p^2=\frac{1}{18} \quad dmnp=\frac{1}{24}$$

By solving the system of equation we have

$$m=n=\frac{1}{2}, p=1, a=d=\frac{1}{6}, b=c=\frac{1}{3}$$

This leads to the fourth order Runge-Kutta formula as follows:

$$k_1 = hf(x, y) = hf$$

$$k_2 = hf(x + \frac{h}{2}, y + \frac{k_1}{2})$$

$$k_3 = hf(x + \frac{h}{2}, y + \frac{k_2}{2})$$

$$k_4 = hf(x + h, y + k_3)$$

$$\Delta y = y(x+h) - y(x) = \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

When the initial values are (x_0, y_0) , then the first increment in y is computed from the given formulae below

$$k_1 = hf(x_0, y_0) \quad , \quad k_2 = hf(x_0 + \frac{h}{2}, y_0 + \frac{k_1}{2}) \quad , \quad k_3 = hf(x_0 + \frac{h}{2}, y_0 + \frac{k_2}{2}),$$

$$k_4 = hf(x_0 + h, y_0 + k_3) \quad ,$$

$$\Delta y = \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

Or

$$y(x_0 + h) = y(x_0) + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

Or

$$y_1 = y_0 + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) .$$

Finally, we have that the general formula for the n^{th} interval is given by the following

$$k_1 = hf(x_n, y_n) \quad k_2 = hf\left(x_n + \frac{h}{2}, y_n + \frac{k_1}{2}\right) \quad k_3 = hf\left(x_n + \frac{h}{2}, y_n + \frac{k_2}{2}\right)$$

$$k_4 = hf(x_n + h, y_n + k_3)$$

$$y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \quad (12)$$

as required.

The y - iteration formula is far more interesting. It is a weighted average of four values- k_1, k_2, k_3 , and k_4 . Visualize distributing the factor of $\frac{1}{6}$ from the front of the sum. Doing this we see that k_1 and k_4 are given a weight of $\frac{1}{6}$ in the weighted average, whereas k_2 and k_3 are weighted $\frac{1}{3}$, or twice as heavily as k_1 and k_4 . (As usual with a weighted average, the sum of the weights $\frac{1}{6}, \frac{1}{3}, \frac{1}{3}$ and $\frac{1}{6}$ is 1). So these k_i used in the weighted average are-

k₁ This quantity, $hf(x_n, y_n)$ is simply Euler's prediction that can be called Δy – the vertical jump from the current point to the next Euler's-predicted point along the numerical solution.

k₂ The x -value at which it is evaluating the function f , $x_n + \frac{h}{2}$ lies halfway across the prediction interval. $y_n + \frac{k_1}{2}$ is the current y -value plus half of the Euler-predicted Δy that we just discussed as being the meaning of k_1 . Recalling that the function f gives us the slope of the solution curve and $f\left(x_n + \frac{h}{2}, y_n + \frac{k_1}{2}\right)$ gives us an estimate of the slope of the solution curve at this halfway point. Multiplying this slope by h , just as with the Euler Method before, produces a prediction of the y - jump made by the actual solution across the whole width of the interval, only this time the predicted jump is not based on the slope of the solution at the left end of the interval, but on the estimated slope halfway to the Euler's –predicted next point.

k₃ This has the formula which is quite similar to that of k_2 , except that where k_1 is being replaced by k_2 . Essentially, the f -value here is yet another estimate of the slope of the solution at the “midpoint” of the prediction interval. This time, however the y -value of the midpoint is not based on Euler's prediction, but on the y -jump predicted already with k_2 . Once again, this slope-estimate is multiplied by h , gives yet another estimate of the y -jump made by the actual solution across the whole width of the interval.

k_4 This evaluates f at $x_n + h$, which is at the extreme right of the prediction interval. The y -value coupled with this, $y_n + k_3$ is an estimate of the y -value at the right end of the interval, based on the y -jump predicted by k_3 . The f -value thus found is once again multiplied by h , just as the three previous k_i , giving a final estimate of the y -jump made by the actual solution across the whole width of the interval.

Thus each of the k_i gives an estimate of the size of the y -jump made by the actual solution across the whole width of the interval. The first one uses Euler's Method, the next two use estimates of the slope of the solution at the midpoint, and the last one uses an estimate of the slope at the right end-point. Each k_i uses the earlier k_i as a basis for its prediction of the y -jump.

This means that the Runge-Kutta formula for y_{n+1} namely:

$$y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

is simply the y -value of the current point plus a weighted average of four different y -jump estimate for the interval, with the estimate based on the slope at the midpoint being weighted twice as heavily as those using the slope at the end-point.

Description Of The Method

Consider the fourth-order initial value problem (IVP) of the form

$$y^{(4)} + a(x)y''' + b(x)y'' + c(x)y' + d(x)y = f(x), \quad 0 \leq x \leq L \quad (13)$$

With initial conditions

$$y(0) = \alpha, \quad y'(0) = \beta, \quad y''(0) = \gamma, \quad y'''(0) = \delta, \quad (14)$$

where $a(x), b(x), c(x), d(x)$ are continuous functions and α, β, γ and δ are constants.

To reduce the fourth order equation in (13) into a first order system of ordinary differential equations, we use the substitutions $z(x) = y'(x), u(x) = y''(x), v(x) = y'''(x)$. Then the given initial value problem of eqn (13) and (14) can be rewritten as

$$\begin{aligned} y' &= z, y(0) = \alpha \\ y'' &= z' = u, y'(0) = z(0) = \beta \\ y''' &= u' = v, y''(0) = z'(0) = u(0) = \gamma \\ y^{(4)} &= v', y'''(0) = u'(0) = v(0) = \delta \end{aligned} \quad (15)$$

Thus we will obtain the system which is written in the form

$$\begin{aligned} z &= f_1(x, y, z, u, v), y(0) = \alpha \\ z' &= u = f_2(x, y, z, u, v), z(0) = \beta \\ u' &= v = f_3(x, y, z, u, v), u(0) = \gamma \\ v' &= g(x, y, z, u, v), v(0) = \delta, \end{aligned} \quad (16)$$

where $g(x, y, z, u, v) = f(x) - a(x)v - b(x)u - c(x)z - d(x)y$.

Dividing the interval $[0, L]$ into N equal subintervals of mesh length h and the mesh point is

given by $x_i = x_0 + ih$ for $i = 1, 2, \dots, N-1$. For the sake of simplicity, let's use the notation $y(x_i) = y_i, z(x_i) = z_i, u(x_i) = u_i, v(x_i) = v_i$ etc.

To solve the system of the initial value problem in (16) we apply the single step method that requires information about the solution at x_i to calculate at x_{i+1} . From one of the single step method and the family of Runge Kutta methods, the general solution of eqn (16) using fourth Runge Kutta method is given as

$$\begin{aligned}
 y_{i+1} &= y_i + \sum_{i=1}^4 w_i k_i \\
 z_{i+1} &= z_i + \sum_{i=1}^4 w_i L_i \\
 u_{i+1} &= u_i + \sum_{i=1}^4 w_i m_i \\
 v_{i+1} &= v_i + \sum_{i=1}^4 w_i q_i,
 \end{aligned}
 \tag{17}$$

where

$$\begin{aligned}
 k_i &= hf_1(x_i + c_i h, y_i + \sum_{j=1}^4 a_{ij} k_j, z_i + \sum_{j=1}^4 a_{ij} L_j, u_i + \sum_{j=1}^4 a_{ij} m_j, v_i + \sum_{j=1}^4 a_{ij} q_j) \\
 L_i &= hf_2(x_i + c_i h, y_i + \sum_{j=1}^4 a_{ij} k_j, z_i + \sum_{j=1}^4 a_{ij} L_j, u_i + \sum_{j=1}^4 a_{ij} m_j, v_i + \sum_{j=1}^4 a_{ij} q_j) \\
 m_i &= hf_3(x_i + c_i h, y_i + \sum_{j=1}^4 a_{ij} k_j, z_i + \sum_{j=1}^4 a_{ij} L_j, u_i + \sum_{j=1}^4 a_{ij} m_j, v_i + \sum_{j=1}^4 a_{ij} q_j)
 \end{aligned}$$

We have that the Runge-kutta fourth order method to solve a first order initial value problem is of the form

$$y' = \frac{dy}{dx} = f(x, y); y(x_0) = y_0$$

$$k_1 = hf(x_0, y_0)$$

$$k_2 = hf(x_n + \frac{h}{2}, y_n + \frac{k_1}{2})$$

$$k_3 = hf(x_n + \frac{h}{2}, y_n + \frac{k_2}{2})$$

$$k_4 = hf(x_n + h, y_n + k_3)$$

$$y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4).$$

Thus, to solve the system of initial value problem of equation (16), one of the families of the fourth order Runge-Kutta can be rewritten as thus:

$$y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

$$z_{n+1} = z_n + \frac{1}{6}(L_1 + 2L_2 + 2L_3 + L_4)$$

$$u_{n+1} = u_n + \frac{1}{6}(m_1 + 2m_2 + 2m_3 + m_4)$$

$$v_{n+1} = v_n + \frac{1}{6}(q_1 + 2q_2 + 2q_3 + q_4),$$

where

$$k_1 = hf_1(x_n, y_n, z_n, u_n, v_n)$$

$$L_1 = hf_2(x_n, y_n, z_n, u_n, v_n)$$

$$m_1 = hf_3(x_n, y_n, z_n, u_n, v_n)$$

$$q_1 = hg(x_n, y_n, z_n, u_n, v_n)$$

$$k_2 = hf_1\left(x_n + \frac{h}{2}, y_n + \frac{k_1}{2}, z_n + \frac{L_1}{2}, u_n + \frac{m_1}{2}, v_n + \frac{q_1}{2}\right)$$

$$L_2 = hf_2\left(x_n + \frac{h}{2}, y_n + \frac{k_1}{2}, z_n + \frac{L_1}{2}, u_n + \frac{m_1}{2}, v_n + \frac{q_1}{2}\right)$$

$$m_2 = hf_3\left(x_n + \frac{h}{2}, y_n + \frac{k_1}{2}, z_n + \frac{L_1}{2}, u_n + \frac{m_1}{2}, v_n + \frac{q_1}{2}\right)$$

$$q_2 = hg\left(x_n + \frac{h}{2}, y_n + \frac{k_1}{2}, z_n + \frac{L_1}{2}, u_n + \frac{m_1}{2}, v_n + \frac{q_1}{2}\right)$$

$$k_3 = hf_1\left(x_n + \frac{h}{2}, y_n + \frac{k_2}{2}, z_n + \frac{L_2}{2}, u_n + \frac{m_2}{2}, v_n + \frac{q_2}{2}\right)$$

$$L_3 = hf_2\left(x_n + \frac{h}{2}, y_n + \frac{k_2}{2}, z_n + \frac{L_2}{2}, u_n + \frac{m_2}{2}, v_n + \frac{q_2}{2}\right)$$

$$m_3 = hf_3\left(x_n + \frac{h}{2}, y_n + \frac{k_2}{2}, z_n + \frac{L_2}{2}, u_n + \frac{m_2}{2}, v_n + \frac{q_2}{2}\right)$$

$$q_3 = hg\left(x_n + \frac{h}{2}, y_n + \frac{k_2}{2}, z_n + \frac{L_2}{2}, u_n + \frac{m_2}{2}, v_n + \frac{q_2}{2}\right)$$

$$k_4 = hf_1(x_n + h, y_n + k_3, z_n + L_3, u_n + m_3, v + q_3)$$

$$L_4 = hf_2(x_n + h, y_n + k_3, z_n + L_3, u_n + m_3, v + q_3)$$

$$m_4 = hf_3(x_n + h, y_n + k_3, z_n + L_3, u_n + m_3, v + q_3)$$

$$q_4 = hg(x_n + h, y_n + k_3, z_n + L_3, u_n + m_3, v + q_3)$$

Application and Numerical Simulation

To validate the applicability of the method, two initial value problems are being considered. For each number of nodal point N , the point wise absolute error are approximated by the formula, $\|E\| = |y(x_i) - y_i|$, for $i = 0, 1, 2, \dots, N$, where $Exact\ y(x)$ and $RK4\ Approx\ y(x)$ are exact and computed approximate solutions of the given problem respectively, at the nodal point x_i .

Example 1

Consider the fourth order initial value problem

$$y^{iv} = x, \quad 0 \leq x \leq 1$$

Subject to the initial conditions

$$y(0) = 0, y'(0) = 1, y''(0) = 0, y'''(0) = 0$$

Take $h = 0.05$ with exact solution as $y(x) = \frac{x^5}{120} + x$.

We seek to compute $y(0.5)$.

Solution

Given that $y^{iv} = x$, $0 \leq x \leq 1$, where $y(0) = 0, y'(0) = 1, y''(0) = 0, y'''(0) = 0$.

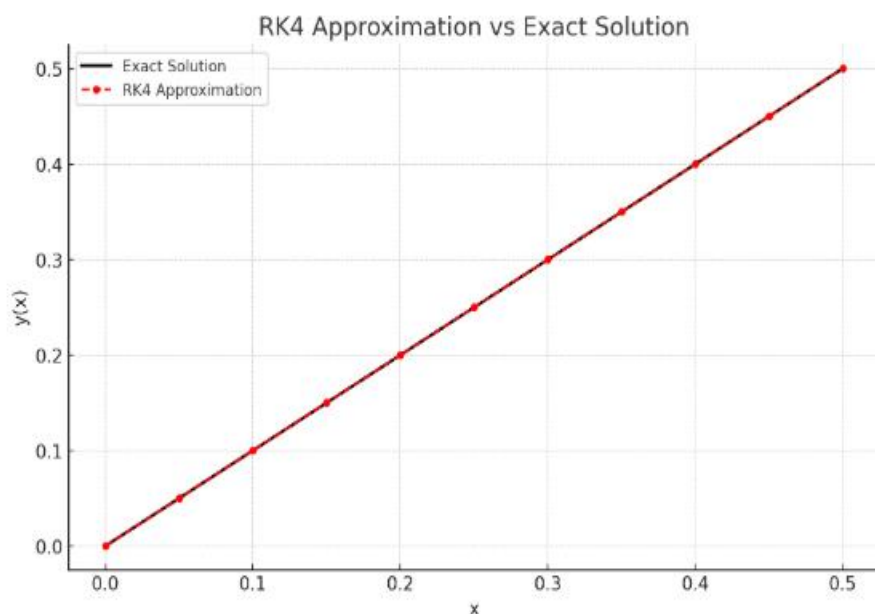
We reduce the fourth-order ODE above to a system of first order ODEs and proceed with our computation.

Here is the full table of results from $x = 0$ to $x = 0.5$ using the Runge-Kutta 4th order method, step size $h=0.05$, along with the exact solution $y(x) = \frac{1}{120}x^5 + x$ and the error at each step:

Table 1. RK4 Approximation vs Exact Solution for Example 1

x	RK4 Approx y(x)	Exact y(x)	Error
0.00	0.00000000	0.00000000	0.000000e+00
0.05	0.05000000	0.05000000	0.000000e+00
0.10	0.10000000	0.10000001	1.000000e-08
0.15	0.15000113	0.15000112	1.000000e-08
0.20	0.20000341	0.20000340	1.000000e-08
0.25	0.25000791	0.25000790	1.000000e-08
0.30	0.30002014	0.30002012	2.000000e-08
0.35	0.35004434	0.35004432	2.000000e-08
0.40	0.40008463	0.40008461	2.000000e-08
0.45	0.45015315	0.45015313	2.000000e-08
0.50	0.50025999	0.50025996	3.000000e-08

Figure 1: Plot showing RK4 approximation Vs exact solution for Example 1



Example 2

We consider now a nonlinear fourth-order problem

$$y^{(4)} = (y')^2 - yy''' - 4x + e^x(1 - 4x + x^2) \quad 0 \leq x \leq 1$$

with initial conditions

$$y(0) = 1, y'(0) = 1, y''(0) = 3, y'''(0) = 1,$$

whose exact solution is given by $y(x) = x^2 + e^x$.

Solution

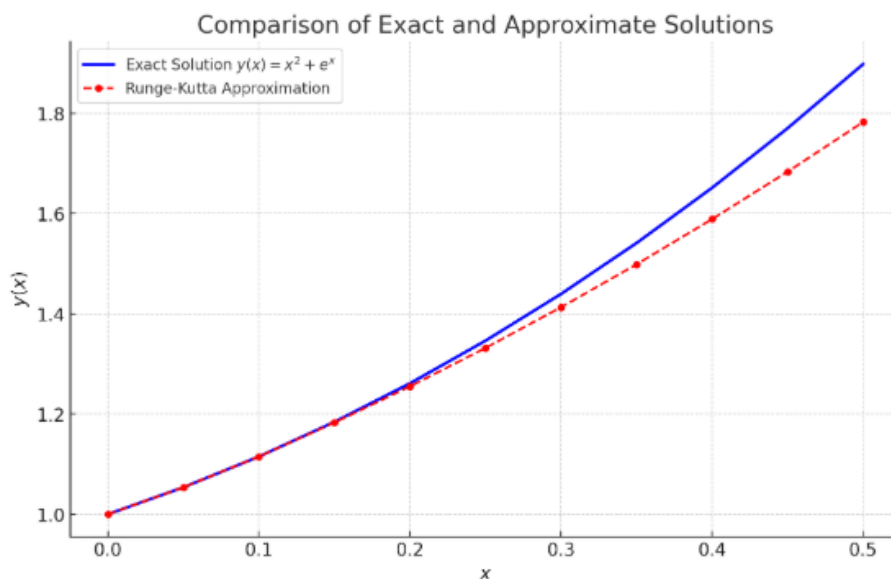
Our first target here is to reduce the fourth-order ODE to a system of first order ODEs and then proceed with our computation.

Here is the full table of results from $x = 0$ to $x = 0.5$ using the Runge-Kutta 4th order method, step size $h=0.05$, along with the exact solution $y(x) = x^2 + e^x$ and the error at each step:

Table 1. RK4 Approximation vs Exact Solution Example 2

x	Approximate y(x)	Exact y(x)	Error
0.00	1.00000000	1.00000000	0.00000000
0.05	1.05377109	1.05328369	0.00048740
0.10	1.11517062	1.11051709	0.00465353
0.15	1.18294662	1.17508669	0.00785993
0.20	1.25523060	1.24547553	0.00975507
0.25	1.33201288	1.32160552	0.01040736
0.30	1.41328518	1.40345537	0.00982981
0.35	1.49903950	1.49105204	0.00798746
0.40	1.58926912	1.58446029	0.00480883
0.45	1.68396877	1.68377223	0.00019654
0.50	1.78313464	1.89872127	0.11558663

Figure 2: Plot showing RK4 approximation Vs exact solution for Example 2



II. Conclusion

In this work, we used the fourth order Runge- Kutta method for solving fourth order initial value problems (IVP) in ordinary differential equations. To achieve the desired accuracy of the numerical solution, it was necessary to take step size to be considerably small. However, it was important to make a proper balance between accuracy and computational efficiency.

In Table 1, RK4 yields extremely accurate results, with error on the order of 10^{-8} . The method performed very well, even with a moderate step size $h = 0.05$. As can be seen in Figure 1, the RK4 points almost perfectly match the exact curve. The error is extremely tiny which confirms the high accuracy of the RK4 method for this problem. In figure 2, the blue curve is the exact solution $y(x) = x^2 + e^x$ while the red dashed line with circles is the approximation via Runge-Kutta 4 method for the nonlinear problem. It can be seen that the approximation tracks the exact solution closely, though small deviations start to appear around $x = 0.4$ and beyond — which is natural.

References

1. Ahmadianfar, I., Heidari, A. A., Gandomi, A. H., Chu, X., Chen, H. (2021). RUN beyond the metaphor: An efficient optimization algorithm based on Runge Kutta method, [Expert Systems with Applications](#), Vol. 181
2. Iyenga, S.R. & Jain, R. K. (2009). Numerical methods. New Delhi: New Age International (p) Limited Publishers.
3. Sakar, S. (2006). Quantum phase diagram of a super conducting quantum dots array. EPL (Europhysics letters).
4. Burden, R.L. & Douglass, J.F. (2011). Numerical analysis. New York: Richard Straton.
5. Davis, M. (2010). Finite difference methods. London: Department of Mathematics, Imperial College.
6. Evans, L.C. (1997). Partial differential equations. Berkeley: American Mathematical Society.
7. Gonze, D. (2012). Linear difference equations. Master en Bioinformatique et modelisation.
8. Zhang, D. K. (2019). Discovering New Runge–Kutta Methods Using Unstructured Numerical Search, Vanderbilt University, Thesis, DOI: 10.48550/arXiv.1911.00318
9. Ghoreishi, F., Ghaffari, R. and Saad, N. (2023). Fractional Order Runge–Kutta Methods, Fractal and Fractional, 7(3), 245; doi.org/10.3390/fractalfract7030245