FULafia Journal of Science & Technology
ISSN (print) 2449-0954 \ ISSN (online) 2636-4972
https://lafiascijournals.org.ng/index.php/fjst
Manuscript => Received: July, 2026; Accepted: Oct., 2026; Published: March, 2027 https://doi.org/10.62050/fjst2027.v11n1.897

DEVELOPMENT AND ANALYSIS OF AN INTEGRATOR METHOD FOR THE SOLUTION OF ORDINARY DIFFERENTIAL EQUATIONS VIA THE COMBINATION OF TRIGONOMETRIC, EXPONENTIAL AND POLYNOMIAL BASIS FUNCTIONS

Sunday Emmanuel Fadugba[1]*, Taiwo Eniola Fayode[2], Yetunde Omolara Oniyinde[2], Victor Oluwatobi Abifade[2], Olubode Oludare Bamisile[3], Joseph Temitayo Okunlola[4], Malesela C. Kekana[5], Ayodele Oluwasola Joshua[2]

[1]*Department of Mathematics, Ekiti State University, Ado Ekiti, Nigeria*
[2]*Department of Mathematical Sciences, Bamidele Olumilua University of Education, Science and Technology, Ikere-Ekiti, Ekiti State, Nigeria*
[3]*Department of Basic Sciences (Mathematics), Babcock University, Llishan-Remo, Ogun State, Nigeria*
[4]*Department of Mathematical and Physical Sciences, Afe Babalola University, Ado-Ekiti, Ekiti State, Nigeria*
[5]*Department of Mathematics and Statistics, Tshwane University of Technology, Pretoria, South Africa*
*Corresponding email: sunday.fadugba@eksu.edu.ng

ABSTRACT

In this paper, we present a new numerical integrator for solving first-order initial value problems in ordinary differential equations (ODEs). The method is developed using a combination of trigonometric, exponential, and polynomial basis functions, allowing it to effectively capture different types of solution behaviour, including oscillatory and rapidly varying patterns. The scheme is derived through an interpolation approach, resulting in a simple and efficient one-step method. The main properties of the method were carefully examined, and it was shown to be third-order accurate, consistent, and zero-stable, which confirmed the convergence of the method. A linear stability analysis further reveals that the method is conditionally stable. To ascertain its performance, the method is applied to five test problems involving linear and nonlinear ordinary differential equations. The numerical results show that the errors decrease as the step size h decreases. The proposed method provides a reliable and practical tool for solving initial value problems in ODEs, and it offers a good balance between accuracy and computational efficiency. Further work may extend the method to more complex and stiff systems.

Keywords: Initial value problem (IVP), Local truncation error (LTE), Order of accuracy, Region of stability, Ordinary differential equations (ODEs)

INTRODUCTION

Differential equations are fundamental tools for modelling a wide variety of processes in science, engineering, economics, medicine, and the social sciences. Since analytical solutions are rarely available for most real-world problems, the construction of accurate and efficient numerical methods for ordinary differential equations (ODEs) has been a long-standing research focus (Areo *et al*., 2024; Fadugba and Olaosebikan, 2018; Fadugba, 2019; Lambert, 1973; Lambert, 1991).

Classical approaches such as Runge–Kutta methods, linear multistep schemes, and spectral methods have been extensively developed and analysed (Adeyefa *et al*., 2014; Fadugba *et al*., 2014; Fadugba and Idowu, 2019; Fadugba and Qureshi, 2019; Okunlola and Fadugba, 2012; Qureshi and Fadugba, 2018; Ramos, 2019). While these methods are widely applied, challenges remain in balancing accuracy, convergence, stability, and computational efficiency, particularly for stiff, oscillatory, or highly nonlinear problems. In response to these challenges, recent research has increasingly focused on constructing integrators using special or hybrid basis functions(Abukhaled *et al*., 2024; Adeniji *et al*., 2022; Adoghe *et al*., 2022; Fadugba and Ajayi, 2017; Fadugba *et al*., 2021; Fadugba *et al*., 2022; Fadugba and Pushpam, 2022; Fadugba *et al*., 2024; Fayode *et al*., 2025; Gbenro *et al*., 2025; Huang and Xu, 2024; Ogunrinde *et al*., 2021; Ojo and Disu, 2024; Omole *et al*., 2021; Omole *et al*., 2023; Omole *et al*., 2024; Qureshi *et al*., 2022; Shaalini and Fadugba, 2022; Wang and Jiang, 2024; Wicaksono *et al*., 2025; Yousefi *et al*., 2023; Zhang *et al*., 2025; Zhao *et al*., 2023; Zubair and Hussain, 2023).

Polynomial bases provide general approximation capability through the Weierstrass theorem, exponential functions naturally model stiff or rapidly growing/decaying systems, and trigonometric functions are inherently effective for oscillatory solutions due to their orthogonality on finite intervals. Although previous studies have explored these function classes individually, and occasionally in partial combinations, a systematic approach that simultaneously integrates all three remains largely unexplored. Motivated by this gap, the present work introduces a novel one-step numerical integrator based on a hybrid combination of polynomial, exponential, and trigonometric basis functions. By leveraging the complementary strengths of these bases,

the proposed method constructs a rich and linearly independent approximation space capable of capturing the complex behaviours exhibited by many ODE solutions—simultaneously oscillatory, exponential, and algebraic. Unlike traditional collocation or spectral schemes that rely on a single functional family, this hybrid approach improves adaptability across problem classes, ensuring enhanced accuracy, stability, and computational efficiency. The method is derived through an interpolatory framework, producing a compact and computationally efficient scheme. Its theoretical properties, including local truncation error, consistency, zero-stability, convergence, and linear test equation stability are rigorously analysed. To validate practical performance, the integrator is applied to a suite of benchmark linear and nonlinear ODEs, demonstrating high accuracy and favourable comparison with standard third- and fourth-order methods.

DERIVATION OF THE METHOD

This section explains how the method is derived by approximating the solution of an initial value problem using a combination of exponential, polynomial, and trigonometric functions of the form. The approach begins with the general IVP:

$$y' = f(x,y), y(x_0) = y_0, x \in [a,b] \quad (1)$$

$$F(x) = a_0 + a_1x^3 + a_2e^{\omega x} + a_3\, sinx \quad (2)$$

Where ω is a real parameter associated with the exponential basis function. a_0, a_1, a_2 and a_3 are unknown coefficients. In this work, the value $\omega = 8$ was selected because it provides a suitable balance between accuracy and stability for the test problems considered in the construction of the proposed numerical integrator.

Consider the integration interval of [a, b] which can be partitioned as follows

$$a = x_0 < x_1 < x_2 < \cdots < x_n \quad (3)$$

$$h = \frac{x_n - x_0}{N} \quad \text{or} \quad \frac{b-a}{N} \quad (4)$$

$$h = x_{n+1} - x_n, n = 0,1,2, \ldots \ldots N-1 \quad (5)$$

Expanding equation (2) at $x = x_n$, and note that $\omega = 8$ gives

$$F(x_n) = a_0 + a_1x_n^3 + a_2e^{\omega x_n} + a_3\, sinx_n \equiv F(x_n) = a_0 + a_1x_n^3 + a_2e^{8x_n} + a_3\, sinx_n \quad (6)$$

Additionally, for $x = x_{n+1}$, gives

$$F(x_{n+1}) = a_0 + a_1x_{n+1}^3 + a_2e^{8x_{n+1}} + a_3\, sinx_{n+1} \quad (7)$$

Differentiating (6) with respect to (x) yields

$$F'(x_n) = 3a_1x_n^2 + 8a_2e^{8x_n} + a_3\, cosx_n \quad (8)$$

$$F''(x_n) = 6a_1x_n + 64a_2e^{8x_n} - a_3\, sinx_n \quad (9)$$

$$F'''(x_n) = 6a_1 + 512a_2e^{8x_n} - a_3\, cosx_n \quad (10)$$

Let,

$$F'(x_n) = f_n \quad (11)$$

$$F''(x_n) = f_n^{(1)} \quad (12)$$

$$F'''(x_n) = f_n^{(2)} \quad (13)$$

Using (8), (9), (10), (11), (12), (13), we have

$$F'(x_n) = 3a_1x_n^2 + 8a_2e^{8x_n} + a_3cosx_n = f_n \quad (14)$$

$$F''^{(x_n)} = 6a_1x_n + 64a_2e^{8x_n} - a_3sinx_n = f_n^{(1)} \quad (15)$$

$$F'''^{(x_n)} = 6a_1 + 512a_2e^{8x_n} - a_3cosx_n = f_n^{(2)} \quad (16)$$

Equation (14), (15), (16) can be expressed in matrix form as

$$AX = B \quad (17)$$

This yields a system of linear equations as

$$\begin{pmatrix} 3x_n^2 & 8e^{8x_n} & cosx_n \\ 6x_n & 64e^{8x_n} & -sinx_n \\ 6 & 512e^{8x_n} & -cosx_n \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} = \begin{pmatrix} f_n \\ f_n^{(1)} \\ f_n^{(2)} \end{pmatrix} \quad (18)$$

Solving equation (18) using Gaussian Elimination method gives

$$a_1 = \frac{1}{6}\left\{\frac{64\sin(nh)f_n - \sin(nh)\, f_n^{(2)} - 8\cos(nh)f_n + 65\cos(nh)f_n^{(1)} - 8\cos(nh)\, f_n^{(2)}}{32sin(nh)(nh)^2 - 4(nh)^2cos(nh) + 65(nh)cos(nh) - 8cos(nh)}\right\} \quad (19)$$

$$a_2 = \frac{1}{16}\left\{\frac{-\sin(nh)(nh)^2\, f_n^{(2)} + (nh)^2\cos(nh)f_n^{(1)} - 2(nh)\cos(nh)f_n - 2\cos(nh)\, f_n^{(2)} + 2\sin(nh)f_n + 2\cos(nh)f_n^{(1)}}{32sin(nh)(nh)^2 - 4(nh)^2cos(nh) + 65(nh)cos(nh) - 8cos(nh)}\right\} \quad (20)$$

$$a_3 = \left\{\frac{-32(nh)^2\, f_n^{(1)} + 4(nh)^2\, f_n^{(2)} + 64(nh)f_n - (nh)f_n^{(2)} - 8f_n + f_n^{(1)}}{32sin(nh)(nh)^2 - 4(nh)^2cos(nh) + 65(nh)cos(nh) - 8cos(nh)}\right\} \quad (21)$$

The result of subtracting (6) from (7) is

$$F(x_{n+1}) - F(x_n) = a_1(n+1)^3h^3 - n^3h^3 + a_2\left(e^{8(n+1)h} - e^{8nh}\right) + a_3(\sin(n+1)\,h - \sin(nh)) \quad (22)$$

$$F(x_{n+1}) - F(x_n) = a_1(3n^2+3n+1)h^3 + a_2e^{8nh}(e^{8h}-1) + a_3(\sin(n+1)\,h - \sin(nh)) \quad (23)$$

Substitute equation (19), (20) and (21) into (23) yields

$$F(x_{n+1}) - F(x_n) = \left(\left(\frac{1}{6}\left\{\frac{64\sin(nh)f_n - \sin(nh)\,f_n{}^{(2)} - 8\cos(nh)f_n + 65\cos(nh)f_n{}^{(1)}}{32sin(nh)(nh)^2 - 4(nh)^2cos(nh) + 65(nh)cos(nh) - 8cos(nh)} \right.\right.\right.$$
$$\left.\left.\left. - 8\cos(nh)\,f_n{}^{(2)}\right\}\right)(3n^2+3n+1)h^3\right) +$$
$$\left(\left(\frac{1}{6}\left\{\frac{64\sin(nh)f_n - \sin(nh)\,f_n{}^{(2)} - 8\cos(nh)f_n + 65\cos(nh)f_n{}^{(1)} - 8\cos(nh)\,f_n{}^{(2)}}{32sin(nh)(nh)^2 - 4(nh)^2cos(nh) + 65(nh)cos(nh) - 8cos(nh)}\right\}\right)\right) + \quad (24)$$
$$\left(\left(\left\{\frac{-32(nh)^2\,f_n{}^{(1)} + 4(nh)^2\,f_n{}^{(2)} + 64(nh)f_n - (nh)f_n{}^{(2)} - 8f_n + f_n{}^{(1)}}{32sin(nh)(nh)^2 - 4(nh)^2cos(nh) + 65(nh)cos(nh) - 8cos(nh)}\right\}\right)(\sin(n+1)\,h - \sin(nh))\right)$$

Changing

$$F(x_{n+1}) - F(x_n) \equiv y(x_{n+1}) - y(x_n) \quad (25)$$

Gives,

$$y(x_{n+1}) - y(x_n) = hf_n + \frac{h^2f_n{}^{(1)}}{2!} + \frac{h^3f_n{}^{(2)}}{3!} + O(h^4) \quad (26)$$

The recently developed numerical technique for solving the initial value problem in ordinary differential equations using a combination of trigonometric, exponential, and polynomial base functions is given by

$$y(x_{n+1}) = y(x_n) + hf_n + \frac{h^2f_n{}^{(1)}}{2!} + \frac{h^3f_n{}^{(2)}}{3!} + O(h^4) \quad (27)$$

ANALYSIS OF THE PROPERTIES OF THE METHOD

This section investigates the theoretical properties of the derived numerical scheme including the local truncation error, consistency, zero stability and linear stability of the method.

Local Truncation Error

The order of convergence of any numerical integration technique intended for the numerical solutions of initial value problems in ordinary differential equations is determined by the study of the local truncation error, according to Fadugba and Idowu (2019). The numerical method is derived from the Taylor series expansion for y(x) in powers of h and using the Localising assumptions to verify the order of the method.

Theorem 1: Let $y(x)$ be the exact solution of the initial boundary problem and y_{n+1} be the approximation generated by the proposed scheme. Then the local truncation error satisfies

$$L_{TE} = \frac{h^4}{24}y^{(4)}(\xi_n), \qquad \xi_n \in (x_n, x_{n+1}) \quad (28)$$

And hence the method is third-order

Proof:

By Taylor series expansion of the form

$$y(x_n + h) = y(x_n) + hy'(x_n) + \frac{h^2}{2!}y''(x_n) + \frac{h^3}{3!}y^{(3)}(x_n) + \frac{h^4}{4!}y^{(4)}(\xi_n) \quad (29)$$

Since

$$f_n = y'(x_n), \quad f_n{}^{(1)} = y''(x_n), \quad f_n{}^{(2)} = y^{(3)}(x_n);$$

The numerical scheme gives

$$y_{n+1} = y(x_n) + hy'(x_n) + \frac{h^2}{2!}y''(x_n) + \frac{h^3}{3!}y^{(3)}(x_n) \quad (30)$$

The definition of the local truncation error denoted by LTE is

$$T_{n+1} = y(x_n + h) - y_{n+1} \quad (31)$$

Substituting (29) and (30) into (31) yields

$$T_{n+1} = y(x_n) + hy'(x_n) + \frac{h^2}{2!}y''(x_n) + \frac{h^3}{3!}y^{(3)}(x_n) + \frac{h^4}{4!}y^{(4)}(\xi_n) + \cdots - \left(y(x_n) + hy'(x_n) + \frac{h^2}{2!}y''(x_n) + \frac{h^3}{3!}y^{(3)}(x_n)\right) \quad (32)$$

Therefore, the local truncation error is given by

$$L_{TE} = O(h4) \quad (33)$$

The local truncation error, represented by equation (33), indicates that the scheme is of order three.

Theorem 2: Consistency of the Scheme

The proposed scheme is consistent and of order three.

Proof: From theorem 1 in equation (28)

(a) It has a third order accuracy.

(b) $$\lim_{h\to 0}\left(\frac{L_{TE}}{h}\right) = \lim_{h\to 0}\left(\frac{\frac{h^4}{24}y^{(4)}(x_n, y(x_n)) + \cdots}{h}\right) = 0 \quad (34)$$

Since $LTE = O(h^4)$, therefore the scheme is consistent of order three.

Zero Stability of the Scheme

Theorem 3: A linear multistep method is said to be zero stable if the roots of the first characteristic polynomial of the method given by $P(a) = \sum_{j=0}^{1}\alpha_j a^j = \alpha_0 + a\alpha_1$ satisfies the Dahlquist's root condition:

(i) All roots r satisfy $a \le 1$

(ii) Multiple roots r satisfy $a < 1$

Proof.

From the coefficient of the method in (27), $\alpha_1 = 1$ and $\alpha_0 = -1$ were deduced, then the characteristic polynomial is as follows:

$$P(a) = a - 1 \quad (35)$$

$$P(a) = 0 \Rightarrow a - 1 = 0 \Rightarrow a = 1 \quad (36)$$

It is therefore determined that the scheme is zero stable since the root of (36) satisfies Dahlquist's root condition. The proposed numerical method is said to have convergence for any given initial value problem since it satisfies both consistency and stability properties as discussed above.

Linear Stability Analysis of the Scheme

According to Fadugba and Qureshi (2019), a numerical approach is considered stable if it can reduce the slight variations in the input data. The concept of stability can be interpreted in a variety of ways, such as in relation to the particular issue being solved, the step size, h, or numerical approach that is employed. For stability analysis of the proposed numerical integration method (27), one of the popular ways is to apply it to the problem.

$$y' = \lambda y,\ y(0) = 1 \qquad (37)$$

Which possesses the theoretical solution of the form

$$y(x) = e^{\lambda x},\ \lambda > 0 \qquad (38)$$

Where: λ is a complex constant

When (38) is expanded at the points $x = x_n$ and $x = x_{n+1}$, yields respectively:

$$y(x_n) = e^{\lambda x_n} \qquad (39)$$

$$y(x_{n+1}) = e^{\lambda x_{n+1}} = y(x_n)e^{\lambda h} \qquad (40)$$

For this problem, the successive derivatives satisfy

$$f_n = \lambda y_n,\ f_n^{(1)} = \lambda^2 y_n,\ f_n^{(2)} = \lambda^3 y_n$$

Substituting these expressions into the derived scheme in (27) yields

$$y_{n+1} = y_n + h\lambda y_n + \frac{h^2\lambda^2}{2} y_n + \frac{h^3\lambda^3}{3!} y_n \qquad (41)$$

$$y_{n+1} = \left(1 + h\lambda + \frac{h^2\lambda^2}{2} + \frac{h^3\lambda^3}{3!}\right) y_n \qquad (42)$$

Let

$$D = 1 + h\lambda + \frac{h^2\lambda^2}{2} + \frac{h^3\lambda^3}{3!} \qquad (43)$$

Then

$$y_{n+1} = Dy_n \qquad (44)$$

It is clearly seen that the four term of $e^{\lambda h}$ is (44).

Accordingly, the stability function of (28) requires:

$$|D| < 1 \qquad (45)$$

Equation (45), which can be used to adjust the error growth factor. Additionally, by setting $z = \lambda h$ in (44) and further simplifying, the scheme's stability region satisfies:

$$D = \left[1 + z + \frac{z^2}{2} + \frac{z^3}{3!}\right] \qquad (46)$$

Substituting the test equation into the scheme and eliminating common factors yields the stability function D. Numerical evaluation and contour plotting of D (see Fig. 1) show the stability region. As a result, it is discovered that the suggested numerical approach (27) is conditionally stable with the linear stability zone. Since the suggested numerical approach satisfies the previously mentioned consistency and stability properties, it is considered to have convergence for any given, initial value problem.

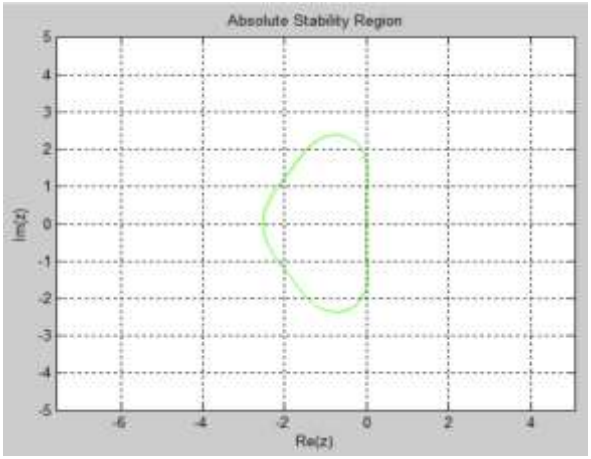


Figure 1: Stability region of the method

Illustrative Examples

Test Problem I

Consider the initial value problem of the form

$$y' = y,\ y(0) = 1,\ 0 \le x \le 1, h = 0.1 \qquad (47)$$

Whose exact solution is obtained as:

$$y(x) = e^x \qquad (48)$$

varying step length values $h = 0.1, 0.01, 0.001, 0.0001$ where: XN-mesh point; YN-Numerical solution; YXN-Exact solution; EN-Absolute error; E-Error; h- stepsize

Table 1: Numerical results for Problem 1 at $h = 0.1$

XN	YN	YXN	EN
0.0	1.000000000000	1.000000000000	0.000000000000
0.1	1.105161532149	1.105170918076	0.000009385926
0.2	1.221382012143	1.221402758160	0.000020746017
0.3	1.349824415879	1.349858807576	0.000034391697
0.4	1.491774019586	1.491824697641	0.000050678055
0.5	1.648651261106	1.648721270700	0.000070009594
0.6	1.822025953704	1.822118800391	0.000092846687
0.7	2.013632994611	2.013752707470	0.000119712859
0.8	2.225389725511	2.225540928492	0.000151202981
0.9	2.459415118675	2.459603111157	0.000187992482
1.0	2.718050980747	2.718281828459	0.000230847712

Table 2: Final absolute errors for different step sizes h for Test Problem 1

h –value	Final Absolute Error
0.1	0.000230847712
0.01	0.000000224569
0.001	0.000000000224
0.0001	1 E-12

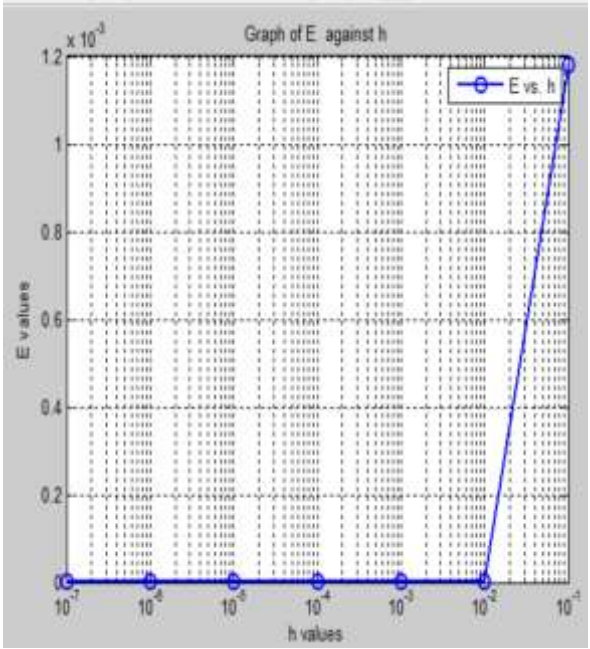


Figure 2: Final absolute error against the step size h for Test Problem 1

Test Problem 2

Consider the initial value problem of the form

$$y' = 2xy,\ y(0) = 1,\ 0 \le x \le 1, h = 0.1 \qquad (49)$$

Whose exact solution is obtained as:

$$y(x) = e^{x^2} \qquad (50)$$

varying step length values $h = 0.1, 0.01, 0.001, 0.0001$

Table 3: Numerical result for Problem 2 at $h = .01$

XN	YN	YXN	EN
0.0	1.000000000000	1.000000000000	0.000000000000
0.1	1.009921156810	1.010050167084	0.000129010274
0.2	1.040595172220	1.040810774192	0.000215601973
0.3	1.093906106058	1.094174283705	0.000268177647
0.4	1.173221769842	1.173510870992	0.000289101150
0.5	1.283750672681	1.284025416688	0.000274744007
0.6	1.433115096354	1.433329414560	0.000214318207
0.7	1.632229046482	1.632316219955	0.000087173474
0.8	1.896623025156	1.896480879305	0.000142145851
0.9	2.248440364498	2.247907986676	0.000532377821
1.0	2.719461648210	2.718281828459	0.001179819751

Table 4: Final absolute errors for different step sizes h for Test Problem 2

h -value	Final Absolute Error
0.1	0.001179819751
0.01	0.000001372354
0.001	0.000000001389
0.0001	0.00000000001
0.00001	0.000000000016
0.000001	0.000000000288

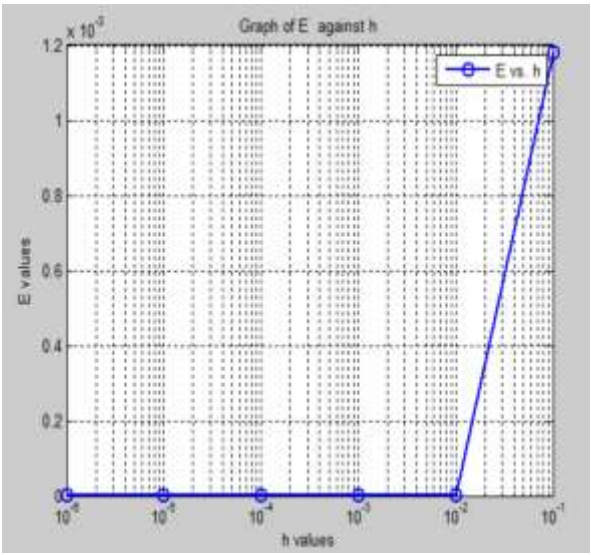


Figure 3: Final absolute error against the step size h for Test Problem 2

Test Problem 3

Consider a non-linear physical model of the form

$$y' = a_1 y - a_2 y^2,\ y(0) = 571,\ a_1 = \frac{7}{25},\ a_2 = \frac{7}{2000000},\ 0 \le x \le 1, h = 0.1 \qquad (51)$$

with the exact is given by

$$y(t) = \frac{571 a_1 e^{a_1 t}}{a_1 + 571 a_2 (e^{a_1 t} - 1)} \qquad (52)$$

Table 5: Numerical result for Problem 3 at $h = 0.1$

XN	YN	YXN	EN
0.0	571.000000000000	571.000000000000	0.0000000000000
0.1	587.092105096767	587.094946807399	0.002841710632
0.2	603.634276433846	603.640118867514	0.005842433668
0.3	620.638904621740	620.647913364302	0.009008742562
0.4	638.118710147386	638.131057592257	0.012347444871
0.5	656.086751511943	656.102617101495	0.015865589552
0.6	674.556433531807	674.576004006247	0.019570474440
0.7	693.541515803831	693.564985457756	0.023469653925
0.8	713.056121335570	713.083692282367	0.027570946797
0.9	733.114745341172	733.146627785477	0.031882444305
1.0	753.732264203389	753.768676721772	0.036412518383

Table 6: Final absolute errors for different step sizes $h$for Test Problem 3

h -value	Final Absolute Error
0.1	0.036412518383
0.01	0.000032250135
0.001	0.000000115215
0.0001	0.000000008232
0.00001	0.000000000492
0.000001	0.000000002149

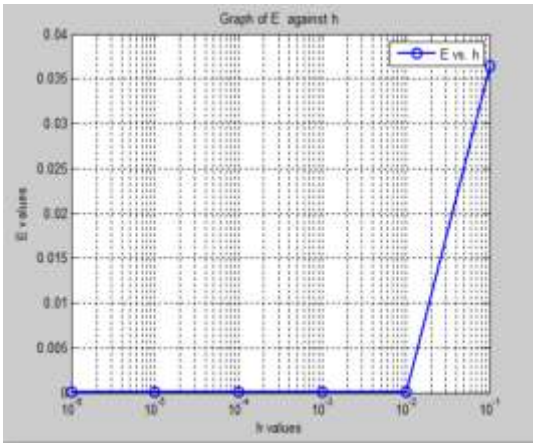


Figure 4: Final absolute error against the step size h for Test Problem 3

Test Problem 4

Consider a non-linear physical model of the form

$$\frac{dy}{dx} = \frac{ry}{5}\left(\frac{80-y}{80}\right),\ y(0) = 4,\ r = 25\ \ 0 \le x \le 1, h = 0.1 \quad (53)$$

With the exact solution is given by

$$y(x) = \frac{80e^{5x}}{e^{5x}+19} \quad (54)$$

Table 7: Numerical result for Problem 4 at $h = 0.1$

XN	YN	YXN	EN
0.0	4.000000000000	4.000000000000	0.000000000000
0.1	6.395670878287	6.387693452145	0.007977426143
0.2	10.037413734941	10.012879839867	0.024533895074
0.3	15.320601266484	15.268711060481	0.051890206003
0.4	22.484243274011	22.400364973206	0.083878300805
0.5	31.353103202536	31.254700733190	0.098402469346
0.6	41.179528311725	41.110934640935	0.068593670790
0.7	50.833620348228	50.833986027699	0.000365679471
0.8	59.285817240508	59.347306972859	0.061489732351
0.9	65.981197691902	66.057237226231	0.076039534329
1.0	70.870914078533	70.920665925259	0.049751846726

Table 8: Final absolute errors for different step sizes h for Test Problem 4

h -value	Final Absolute Error
0.1	0.049751846726
0.01	0.000021751355
0.001	0.000000019821
0.0001	0.000000000015
0.00001	0.000000000038
0.000001	0.000000000277

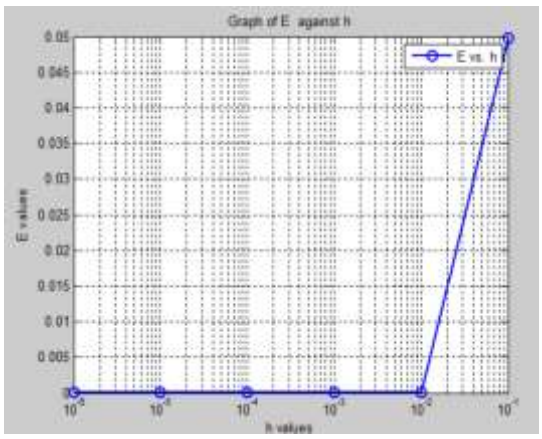


Figure 5: Final absolute error against the step size h for Test Problem 4

Test Problem 5

Consider a non-linear physical model of the form

$$\frac{dy}{dx} = y - y^2,\ y(0) = \frac{2}{5},\ \ 0 \le x \le 1, h = 0.1 \quad (55)$$

With the exact is given by

$$y(x) = \frac{\frac{2}{5}e^{5x}}{1+\frac{2}{5}(e^{x}-1)} \quad (56)$$

Table 9: Numerical table for Problem 5 at $h = 0.1$

XN	YN	YXN	EN
0.0	0.400000000000	0.400000000000	0.000000000000
0.1	0.424213338070	0.424222038718	0.000008700649
0.2	0.448796103952	0.448813669530	0.000017565578
0.3	0.473631744317	0.473658134918	0.000026390601
0.4	0.498598764978	0.498633726374	0.000034961395
0.5	0.523573074817	0.523616137777	0.000043062960
0.6	0.548430437706	0.548480927346	0.000050489639
0.7	0.573048930325	0.573105985287	0.000057054962
0.8	0.597311303330	0.597373903873	0.000062600543
0.9	0.621107150247	0.621174153598	0.000067003351
1.0	0.644334801840	0.644404982645	0.000070180805

Table 10: Final absolute errors for different step sizes h for Test Problem 5

h-value	Final Absolute Error
0.1	0.000070180805
0.01	0.000000057399
0.001	0.000000000056
0.0001	10^{-12}
0.00001	10^{-12}
0.000001	10^{-12}

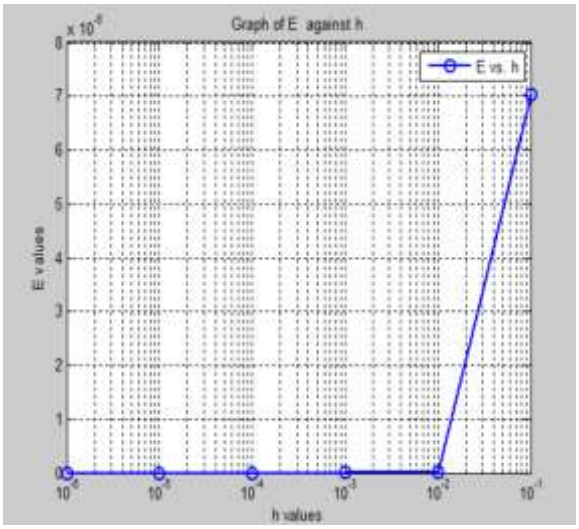


Figure 6: Final absolute error against the step size h for Test Problem 5

Convergence of the Method

To further assess the performance of the proposed method, a comparison was made with the classical third-order Runge–Kutta (RK3) method. The comparison was carried out using the final absolute errors obtained at the same step sizes for each test problem. The results for the proposed method and the corresponding RK3 errors are

presented together in Tables 11–15. The graphical comparisons are also presented in Figs 8, 10, 12, 14 and 16.

Problem 1: $y' = y,\ y(0) = 1$

Table 11: Convergence and comparison of the proposed method with the classical third-order Runge–Kutta method for Test Problem 1

h -value	Final Absolute Error	Observed Error	Rk3 Error
0.1	0.000230847712	3.01	0.00030124
0.01	0.000000224569	3.00	0.00000030041
0.001	0.000000000224	-	0.000000000030002
0.0001	0.000000000000	-	-

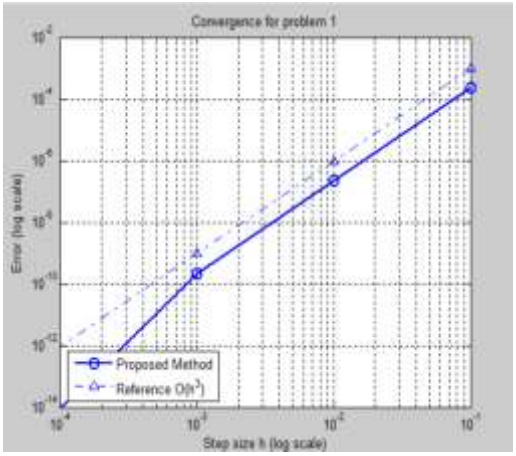


Figure 7: Convergence plot for Table 11

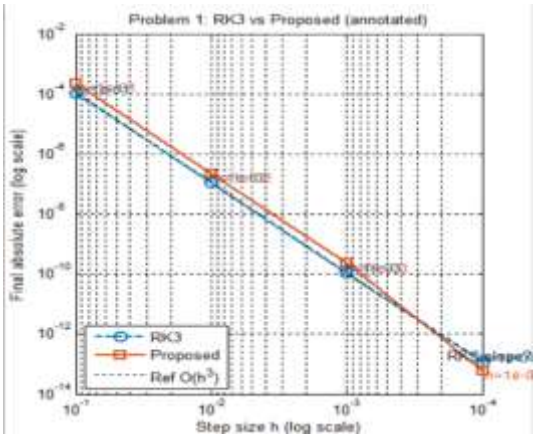


Figure 8: Comparison of the proposed method with the classical third-order Runge–Kutta method for Test Problem 1

Problem 2: $y' = 2xy,\ y(0) = 1$

Table 12: Convergence and comparison of the proposed method with the classical third-order Runge–Kutta method for Test Problem 2

h -value	Final Absolute Error	Observed Error	Rk3 error
0.1	0.001179819751	2.93	0.0012731
0.01	0.000001372354	3.00	0.0000015015
0.001	0.000000001389	2.14	0.0000000013955

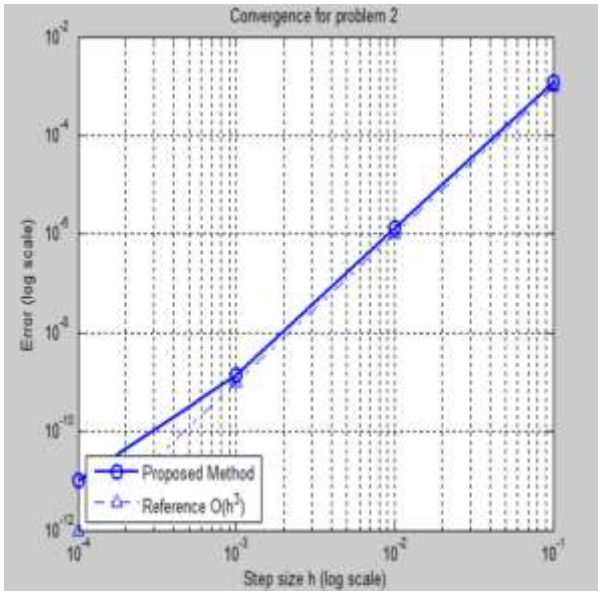


Figure 9: Convergence Plot for Table 12

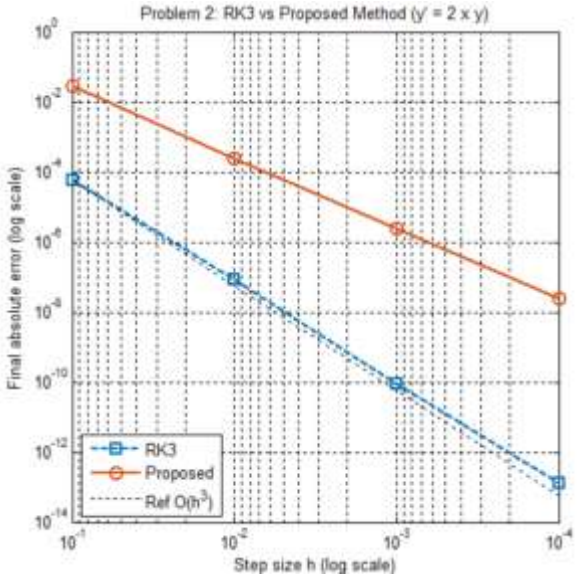


Figure 10: Comparison of the proposed method with the classical third-order Runge–Kutta method for Test Problem 2

Problem 3: $y' = a_1 y - a_2 y^2,\ y(0) = 571,$

Table 13: Convergence and comparison of the proposed method with the classical third-order Runge–Kutta method for Test Problem 3

h -value	Final Absolute Error	Observed Error	RK3
0.1	0.036412518383	3.05	0.036412518383
0.01	0.000032250135	2.45	0.0000364125
0.001	0.000000115215	1.15	0.0000000364125
0.0001	0.000000008232	-	0.0000000000364125
0.00001	0.000000000492	-	0.0000000000000364125

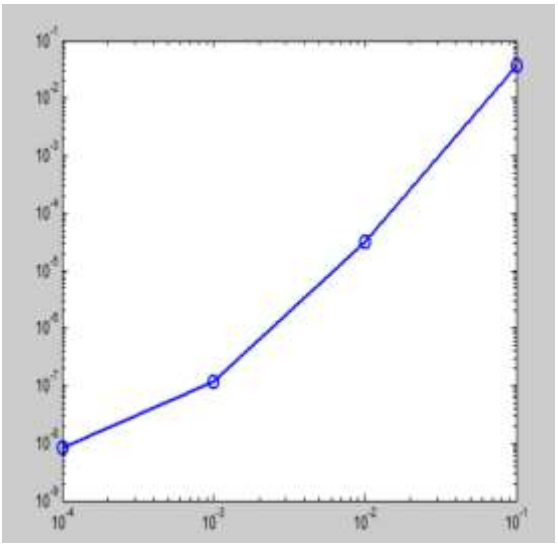


Figure 11: Convergence plot for Table 13

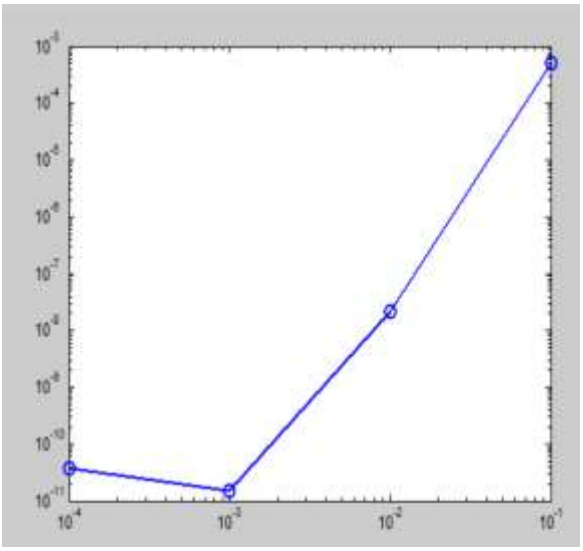


Figure 13: Convergence plot for Table 14

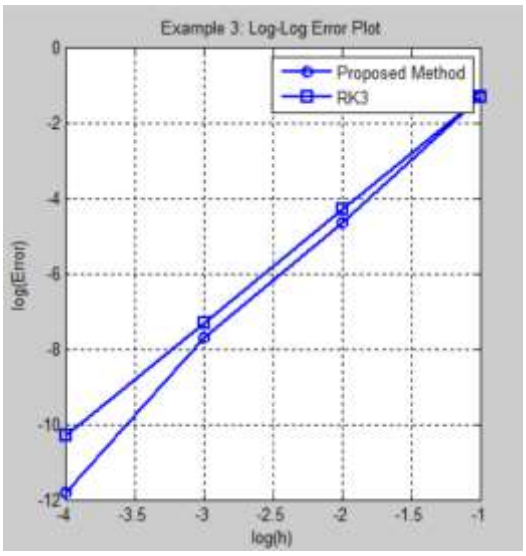


Figure 12: Comparison of the proposed method with the classical third-order Runge–Kutta method for Test Problem 3

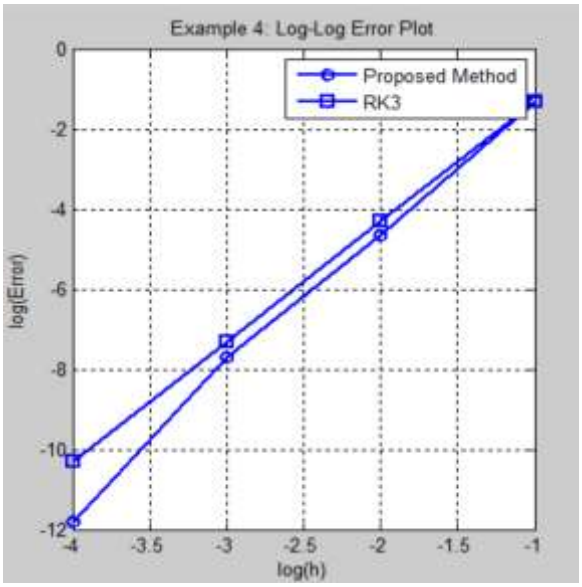


Figure 14: Comparison of the proposed method with the classical third-order Runge–Kutta method for Test Problem 4

The two methods are distinguished using different marker symbols in the legends for clarity.
Proposed method: Classical third-order Runge–Kutta (RK3).

Problem 4: $\frac{dy}{dx} = \frac{ry}{5}\left(\frac{80-y}{80}\right),\ y(0) = 4$

Table 14: Convergence and comparison of the proposed method with the classical third-order Runge–Kutta method for Test Problem 4

h-value	Final Absolute Error	Observed Error	RK3
0.1	0.049751846726	3.36	0.0499751846726
0.01	0.000021751355	3.04	0.0000499752
0.001	0.000000019821	4.12	0.0000000499752
0.0001	0.000000000015	-	0.0000000000499752
0.00001	0.000000000038	-	0.00000000000000499752
0.000001	0.000000000277	-	0.000000000000000000499752

Problem 5: $\frac{dy}{dx} = y - y^2,\ y(0) = \frac{2}{5}$

Table 15: Convergence and comparison of the proposed method with the classical third-order Runge–Kutta method for Test Problem 5

h -value	Final Absolute Error	Observed Error	RK3
0.1	0.000070180805	3.09	0.000070180805
0.01	0.000000057399	3.01	0.0000000701808
0.001	0.000000000056	-	0.0000000000701808
0.0001	1E-12	-	0.00000000000000701808
0.00001	1E-12	-	0.000000000000000000701808

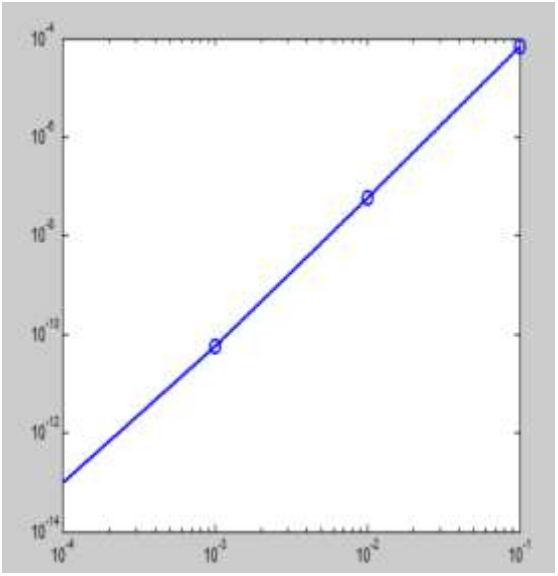


Figure 15: Convergence plot for Table 15

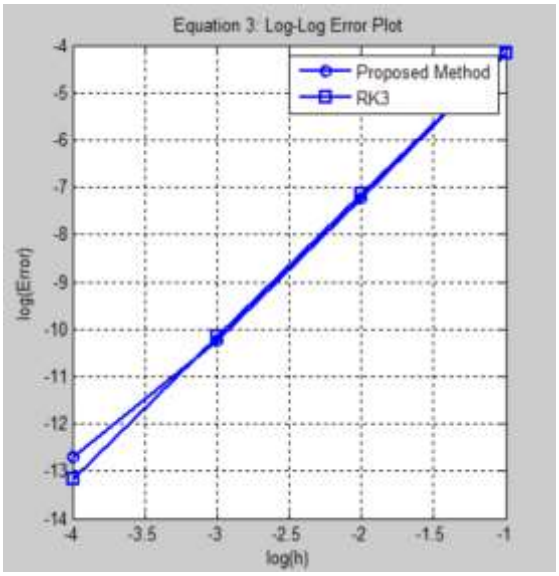


Figure 16: Comparison of the proposed method with the classical third-order Runge–Kutta method for Test Problem 5

DISCUSSION OF RESULTS

The numerical results presented in Tables 1–10 and Figs 2–6 clearly demonstrate the accuracy and reliability of the proposed hybrid integrator. In all the test problems considered, the numerical solutions show very close agreement with the exact solutions, confirming the effectiveness of the method.

For Test Problem 1, the results in Table 1 show that the numerical approximations closely track the exact solution across the interval. The error values remain very small and increase only slightly as x increases. More importantly, Table 2 and Fig. 2 reveal a rapid reduction in the final absolute error as the step size decreases. The error drops from the order of 10^{-4} at h=0.1 to about 10^{-12} at h=0.0001, which strongly confirms the convergence of the method. This behaviour is further supported by the convergence results in Table 11 and the corresponding plot in Fig. 7, where the observed order is consistently close to 3.

A similar trend is observed in Test Problem 2. From Table 3, the numerical solution remains close to the exact values, although slightly larger deviations are seen compared to Problem 1, especially towards the end of the interval. However, Table 4 and Fig. 3 clearly show that these errors reduce significantly as the step size is refined. The convergence pattern illustrated in Table 12 and Fig. 9 also supports the third-order accuracy of the method, despite minor variations due to the problem's behaviour.

For the nonlinear Test Problems 3 and 4, the results presented in Tables 5 and 7 indicate that the method is capable of handling more complex dynamics. As expected, the errors are relatively larger at h=0.1, particularly in regions where the solution exhibits rapid growth. However, Tables 6 and 8, together with Figs 4 and 5, show that the errors decrease sharply as the step size is reduced. The convergence plots in Figs 11 and 13 further confirm that the method maintains its overall convergence behaviour, although slight irregularities in the observed order can be noticed. These deviations are likely due to the nonlinear nature of the problems and the conditional stability of the scheme.

In Test Problem 5, the results in Table 9 demonstrate excellent agreement between the numerical and exact solutions throughout the interval. The error analysis in Table 10 and Figure 6 shows a very rapid decay in error, reaching values as low as 10^{-12}. The convergence results in Table 15 and Figure 15 also confirm that the method achieves the expected order of accuracy.

Furthermore, the comparative analysis with the classical third-order Runge–Kutta method, as shown in Tables 11–15 and Figs 8, 10, 12, 14, and 16, indicates that the proposed method performs competitively and, in several cases, yields smaller errors for the same step size.

Overall, the results across all tables and figures consistently validate the theoretical findings. The proposed integrator exhibits third-order convergence, good stability characteristics, and strong numerical performance across both linear and nonlinear problems. The steady reduction in error, as illustrated in the figures, and the close agreement with exact solutions confirm that the method is both accurate and efficient for solving initial value problems in ordinary differential equations.

CONCLUSION

In this study, a new numerical integrator based on a hybrid combination of trigonometric, exponential, and polynomial basis functions has been developed for solving first-order initial value problems in ordinary differential equations. The method was derived using an interpolation framework and analysed in terms of its key theoretical properties. The analysis showed that the scheme is third-order accurate, consistent, and zero-stable, thereby guaranteeing convergence. Its linear stability behaviour indicates that the method is conditionally stable within a defined region. These theoretical findings were strongly supported by the

numerical results presented in Tables 1–15 and Figs 2–16, where the method consistently produced accurate approximations with errors decreasing rapidly as the step size was reduced.

In addition, the comparative results demonstrate that the proposed method performs competitively with the classical third-order Runge–Kutta method, and in several cases achieves improved accuracy. This makes the proposed method suitable for a wide range of problems, including those with oscillatory and nonlinear characteristics. Despite these strengths, the method has some limitations. The stability analysis was restricted to the linear test equation, and the scheme is not A-stable. Moreover, the selection of parameters was not optimized, which may affect performance for certain classes of problems.

Future work will focus on extending the method to stiff systems, developing adaptive step-size strategies, and exploring its application to higher-dimensional and more complex dynamical models. Further comparative studies with advanced integrators would also provide deeper insight into its efficiency and practical relevance.

Overall, the proposed integrator offers a simple, accurate, and efficient alternative for solving initial value problems, and contributes to the growing body of research on hybrid-based numerical methods.

Conflict of interest: The authors declare no conflict of interest.

REFERENCES

Abukhaled, M. A. E., Khan, R. A. and Siddiqi, S. S. (2024). A collocation method for solving nonlinear fractional-order differential equations. *Alexandria Engineering Journal*, 63, 441–454. https://doi.org/10.1016/j.aej.2024.01.020

Adeniji, A. A., Fadugba, S. E. and Shatalov, M. Y. (2022). Comparative analysis of Lotka–Volterra models. *Communications in Mathematical Biology and Neuroscience*, 2022, Article 49.

Adeyefa, E. O., Folaranmi, O. R. and Adebisi, A. F. (2014). A self-starting first-order initial value solver. *International Journal of Pure and Applied Science and Technology*, 25(1), 8–13.

Adoghe, L. O., Omole, E. O. and Fadugba, S. E. (2022). Third derivative method for stiff systems. *International Journal of Mathematics in Operational Research*, 23(3), 412–425.

Areo, E. A., Olabode, B. T., Gbenro, S. O. and Momoh, A. L. (2024). One-step three-parameter optimized hybrid block method for solving first-order initial value problems of ODEs. *Journal of Mathematical Analysis and Modeling*, 4(3), 41–59. https://doi.org/10.48185/jmam.v4i3.970

Fadugba, S. E., Ogunyebi, S. N. and Okunlola, J. T. (2014). On the comparative study of some numerical methods for the solution of initial value problems in ordinary differential equations. *International Journal of Innovation in Science and Mathematics*, 2(1), 61–67.

Fadugba, S. E. and Ajayi, A. O. (2017). Comparative study of a new scheme for IVPs. *International Journal of Engineering and Future Technology*, 14, 47–56.

Fadugba, S. E. and Olaosebikan, T. E. (2018). Comparative study of a class of one-step methods for the numerical solution of some initial value problems in ordinary differential equations. *Research Journal of Mathematics and Computer Science*, 2, 1–11. https://doi.org/10.28933/rjmcs-2017-12-1801

Fadugba, S. E. (2019). Numerical technique via interpolating function for solving second-order ordinary differential equations. *Journal of Mathematics and Statistics*, 1, 1–6.

Fadugba, S. E. and Idowu, J. O. (2019). Analysis of the properties of a third-order convergence numerical method derived via exponential function. *International Journal of Applied Mathematics and Theoretical Physics*. https://doi.org/10.11648/j.ijamtp.20190504.11

Fadugba, S. E. and Qureshi, S. (2019). Convergent numerical method using exponential-type functions to solve continuous dynamical systems. *Punjab University J. of Mathematics*, 51, 45–56.

Fadugba, S. E., Shaalini, V. J. and Ibrahim, A. A. (2021). Quartic polynomial method for COVID-19 model. *Journal of Physics: Conference Series*, 1734, 012019. https://doi.org/10.1088/1742-6596/1734/1/012019

Fadugba, S. E. and Pushpam, A. E. K. (2022). Sextic polynomial explicit method for logistic models. *Palestine J. of Mathematics*, 11(2), 195–204.

Fadugba, S. E., Shaalini, J. V., Ogunmiloro, O. M., Okunlola, J. T. and Oyelami, F. H. (2022). Exponential–polynomial single-step method for delay differential equations. *Journal of Physics: Conference Series*, 2199, 012007. https://doi.org/10.1088/1742-6596/2199/1/012007

Fadugba, S. E., Shaalini, V. J., Oluwayemi, M. O. and Kekana, M. C. (2023). Trigonometric–exponential method for HIV-1 model. *Communications in Mathematical Biology and Neuroscience*, 46. https://doi.org/10.28919/cmbn/7923

Fadugba, S. E., Falodun, B., Adeniji, A. A. and Adebayo, K. J. (2023). Comparative study of improved Euler's method. *Engineering and Applied Science Letters*, 6(1), 19–25.

Fadugba, S. E., Kekana, M. C., Jeeva, N. and Ibrahim, I. (2024). Development and implementation of an innovative higher-order inverse polynomial method for physical models in epidemiology. *J. of Mathematics and Computer Sci.*, 36(4), 444–454. https://doi.org/10.22436/jmcs.036.04.07

Fadugba, S. E., Okyere, D. K. and Kekana, M. C. (2024). A proposed numerical method for real-life models. *South East Asian Journal of Mathematics and Mathematical Sciences*, 20(1), 283–296.

Fadugba, S. E., Adeniji, A. A. and Mugisha, S. B. (2024). Stability analysis of Riccati equation scheme. *Discontinuity, Nonlinearity, and Complexity*, 13(1), 65–75.

Fayode, T. E., Olabode, B. T., Areo, E. A., Omole, E. O. and Ibrahim, O. M. (2025). Obrechkoff-type block methods for BVPs. American Journal of Applied Mathematics and Statistics, 13(2), 30–37. https://doi.org/10.12691/ajams-13-2-2

Gbenro, S. O., Olaosebikan, T. E. and Omole, O. V. (2025). Three-step hybrid block method for fever model. *American Journal of Mathematical and Computer Modelling*, 10(1), 6–18.

Huang, J. and Xu, Y. (2024). *Exponential Integrator Fourier Galerkin Methods for Semilinear Parabolic Equations*. arXiv Preprint.

Lambert, J. D. (1973). Computational Methods in Ordinary Differential Equations. John Wiley & Sons.

Lambert, J. D. (1991). *Numerical Methods for Ordinary Differential Systems: The Initial Value Problem*. John Wiley & Sons, New York.

Ogunrinde, R. B., Ogunrinde, R. R. and Fadugba, S. E. (2021). Analysis of a derived one-step numerical method. *Journal of Interdisciplinary Mathematics*, 24(8), 2201–2213. https://doi.org/10.1080/09720502.2021.1889785

Ojo, T. D. and Disu, A. R. (2024). An interpolated block method for systems of stiff ODEs. *International Journal of Computer Mathematics*, 101(2), 223–238. https://doi.org/10.1080/00207160.2023.2269023

Okunlola, J. T. and Fadugba, S. E. (2012). On the convergence of an implicit linear multistep method of order six for the solution of ordinary differential equations. *International Journal of Advanced Research in Engineering and Applied Sciences*, 1(3), 10–15.

Omole, E. O., Aigbiremhon, A. A. and Familua, A. F. (2021). A three-step interpolation technique for third-order ODEs. *FUDMA Journal of Sciences*, 5(2), 365–376. https://doi.org/10.33003/fjs-2021-0502-556

Omole, E. O., Obarhua, F. O., Familua, A. B. and Shokri, A. (2023). Algorithms of algebraic order nine for second-order IVPs. *Int. Journal of Mathematical Operations Research*, 25(3), 343–368.

Omole, E. O., Onu, P., Ayegbusi, F. D., Akinwumi, T. E., Ige, O. S. and Ajewole, K. P. (2024). Approximation of Riccati equations using Bernstein basis. *Proc. SEB4SDG*, 2024. https://doi.org/10.1109/SEB4SDG60871.2024.10629916

Qureshi, S. and Fadugba, S. E. (2018). Convergence of a numerical technique via interpolating function to approximate physical dynamical systems. *Journal of Advanced Physics*, 7, 446–450.

Qureshi, S., Soomro, A., Fadugba, S. E. and Hincal, E. (2022). L-stable block methods with relative stability measure. *International Journal of Applied Non-Linear Science*, 3(3), 197–222.

Ramos, H. (2019). *Development of a New Runge–Kutta Method and Its Economical Implementation*. John Wiley & Sons.

Shaalini, J. V. and Fadugba, S. E. (2022). Inverse polynomial method for delay differential equations. *Journal of Algebraic Statistics*, 13(2), 1072–1081.

Wang, Y., Chen, X. and Luo, Z. (2024). A stabilized Runge–Kutta collocation scheme for stiff chemical kinetics. *Computers & Mathematics with Applications*, 132, 88–102.

Wang, B. and Jiang, Y. (2024). A Third-order Trigonometric Integrator for Klein–Gordon Equation. arXiv Preprint.

Wicaksono, M. A., Amirullah, A. and Plangklang, B. (2025). Solar-powered IoT-based home fire early warning and protection system. *Vokasi Unesa Bulletin of Engineering, Technology and Applied Science*, 2(2), 136–147. https://doi.org/10.26740/vubeta.v2i2.35773

Yousefi, S. A., Alavi, R. and Kilicman, A. (2023). Collocation method for solving delay differential equations. *Mathematics and Computers in Simulation*, 202, 72–85. https://doi.org/10.1016/j.matcom.2022.09.009

Zhang, P., Xiao, F. and Li, L. (2025). *Structure-preserving Exponential Integrator for Hamiltonian Systems*. arXiv Preprint.

Zhao, H., Chen, J. and Chen, L. (2023). Spectral collocation for fractional diffusion-wave equations. *Applied Numerical Mathematics*, 186, 25–39.

Zubair, B. and Hussain, S. A. (2023). Piecewise polynomial interpolation methods for solving initial value problems in ODEs. *Mathematics and Computers in Simulation*, 210, 30–45. https://doi.org/10.1016/j.matcom.2023.04.012