Faculty of Natural and Applied Sciences Journal of Mathematical and Statistical Computing
Print ISSN: 3026-8605, e-ISSN 3043-632X
www.fnasjournals.com
Volume 3; Issue 1; March 2026; Page No. 36-50.
DOI: https://doi.org/10.63561/jmsc.v3i1.1221

Analytical and Numerical Study of Runge-Kutta Methods for the Approximate Solution of Ordinary Differential Equations

[1]Jacob, E., [2]Edibo, M.F., [3]Durojaye, M.O., [3]Abdullahi, A.M., [4]Olumi, T.T., & [3]Ichado, F.
[1]Department of Mathematical Science Prince Abubakar Audu University, Anyigba, Nigeria.
[2]Department of Mathematics Federal University, Lokoja, Nigeria.
[3]Department of Mathematics University of Abuja, FCT, Nigeria.
[4]Department of Statistics Federal Polytechnic, Orogun, Nigeria.

***Corresponding author email**: jacobemma001@gmail.com

Abstract
This research presents a comprehensive analytical and numerical investigation of Runge–Kutta methods for the approximate solution of initial value problems (IVPs) arising in ordinary differential equations. The study focuses on three variants: the second-order Runge Kutta method (Heun's method), the classical fourth-order Runge Kutta method, and the fifth-order Runge–Kutta method (Butcher's method). Each method is systematically implemented using MATLAB, and their numerical solutions are compared against exact analytical results to assess accuracy and stability. A series of computational experiments with varying step sizes (0.1, 0.05, and 0.025) are performed, with outcomes presented in tabular form to illustrate convergence trends and error propagation. The analysis of error terms, approximate values, and maximum errors enables a rigorous comparative evaluation of these schemes. The findings reveal that although all methods benefit from reduced step lengths, higher-order Runge-Kutta algorithms, particularly the fifth-order formulation, demonstrate superior accuracy and computational efficiency. This work highlights the essential role of Rung- Kutta methods in the robust numerical approximation of differential equations, with direct implications for applied mathematical modelling and scientific computing.

Keywords: Initial Value Problem, Heun's Method, Fourth-Order Runge-Kutta, Fifth-Order Runge-Kutta, MATLAB.

Introduction

In science and engineering, many physical and natural phenomena are described by mathematical models formulated as differential equations. These equations provide powerful frameworks for analysing dynamic processes such as population growth, chemical kinetics, mechanical vibrations, heat transfer, and fluid flow. However, a large proportion of ordinary differential equations (ODEs) arising in practical applications lack closed-form (analytical) solutions. This limitation necessitates the use of numerical methods to approximate their solutions with acceptable accuracy and efficiency.

Two principal classes of ODE problems are recognized: initial value problems (IVPs), where conditions are specified at the beginning of the interval of interest, and boundary value problems (BVPs), where conditions are imposed at two or more points in the domain. For IVPs, one of the earliest and simplest numerical techniques is Euler's method, introduced by Leonard Euler in 1768. While foundational, Euler's method suffers from limited accuracy and stability, leading to the development of more advanced algorithms. Among these, the Runge–Kutta (RK) family of methods, initiated by Carl Runge and extended by Martin Wilhelm Kutta in 1901, has become the most widely adopted approach in scientific computing due to its balance of accuracy, simplicity, and computational efficiency.

Runge–Kutta methods approximate Taylor series expansions up to order hrrhr, where rrr represents the order of the method. This property ensures controlled local truncation error and reliable convergence behavior. Their versatility and ease of implementation have established them as essential tools in modern numerical analysis.

Recent research continues to refine and extend the applicability of Runge–Kutta methods. Jerbi et al. (2025) proposed efficient high-order RK pairs for linear inhomogeneous problems using differential evolution. Alomari et al. (2025) performed a comparative study between Euler–Maclaurin and Runge–Kutta schemes, offering insights into modern alternatives. Workineh et al. (2024) investigated the performance of Euler's method and RK-4 in solving second-order IVPs, while Zhang (2024) introduced an explicit 10th-order RK method discovered through numerical search. Stability properties of hybrid RK schemes have been addressed by Wang and Yu (2025), whereas Alalhareth et al. (2024) developed modified nonstandard RK methods for high-dimensional autonomous systems. Foundational contributions from Munthe-Kaas (1999) on geometric RK methods and Gottlieb et al. (2011) on strong-stability-preserving RK methods remain highly influential.

Building on these foundational and contemporary developments, the present research undertakes an analytical and numerical study of second-order (Heun's method), classical fourth-order, and fifth-order (Butcher's method) Runge–Kutta algorithms. The methods are implemented using MATLAB, and their performance is systematically compared against exact solutions. The study emphasizes error analysis, convergence behavior, and computational efficiency, thereby contributing to the ongoing discourse on reliable numerical techniques for solving IVPs in ordinary differential equations.

Material and Methods

This study adopts an analytical and numerical approach to investigate the performance of Runge–Kutta (RK) methods in solving initial value problems (IVPs) of ordinary differential equations (ODEs). The methodology is structured into four key phases: model formulation, method implementation, computational experiments, and comparative analysis.

Model Formulation

The research begins with the definition of representative IVPs of ODEs for which exact analytical solutions are known. These problems serve as benchmarks to evaluate the accuracy of the selected Runge–Kutta methods. The general form considered is:

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

Where the solution $y(x)$ is approximated over a finite domain.

Second Order

Runge-Kutta 2nd – Orde Method (Heun's Method)

Consider the initial value problem

$\frac{dy}{dx} = f(x,y), y(x_0) = y_0$ (2)

The method relies on computing the next value of y_{n+1} where

$k_1 = f(x_n, y_n)$ (3)

$k_2 = f(x_n + a_2 h, y_n + h a_{21} k_1)$ (4)

$y_{n+1} = (y_n + h(b_1 k_1 + b_2 k_2)$ (5)

$a_2 = 1, a_{21} = 1, b_1 = \frac{1}{2}, b_2 = \frac{1}{2}$ (6)

$k_1 = f(x_n, y_n)$ (7)

$k_2 = f(x_n + h, y_n + h k_1)$ (8)

$y_{n+1} = \left(y_n + \frac{h}{2}(k_1 + k_2\right); n = 0.1.2.3.4, ...$ (9)

Runge-Kutta 4th Order Method

Consider the initial value problem

$\frac{dy}{dx} = f(x,y), y(x_0) = y_0$ (10)

$k_1 = f(x_n, y_n)$ (11)

$k_2 = f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2} k_1\right)$ (12)

Jacob, E., Edibo, M.F., Durojaye, M.O., Abdullahi, A.M., Olumi, T.T., & Ichado, F. (2026). An analytical and numerical study of Runge-Kutta methods in the approximate solution of ordinary differential equations., *FNAS Journal of Mathematical and Statistical Computing*, *3*(1), 36-50. https://doi.org/10.63561/jmsc.v3i1.1221

$k_3 = f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2}k_2\right)$ (13)

$k_4 = (x_n + h, y_n + hk_3)$ (14)

$y_{n+1} = y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4); n = 0.1.2.3.4, ...$ (15)

Butcher tableau (standard RK4)

c_1	a_{11}	a_{12}	$\cdots$	a_{1s}
c_2	a_{21}	a_{22}	$\cdots$	a_{2s}
c_3	a_{31}	a_{32}	$\cdots$	a_{3s}
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
c_s	a_{s1}	a_{s2}	$\cdots$	a_{ss}
	b_1	b_2	$\cdots$	b_s

Runge-Kutta Fifth Order Method

The Runge–Kutta 5th Order (RK5) method is a higher-order numerical technique used to solve ordinary differential equations (ODEs) of the form:

Consider the initial value problem

$\frac{dy}{dx} = f(x, y), y(x_0) = y_0$ (16)

It provides better accuracy than RK4 for the same step size.

The method relies on computing the next value of $y_{n+1}$from the current $y_n$using a weighted combination of intermediate slope evaluations. Given step size h, we compute intermediate slopes:

$k_1 = f(x_n, y_n)$ (18)

$k_2 = f\left(x_n + \frac{h}{4}, y_n + \frac{h}{4}k_1\right)$ (19)

$k_3 = f\left(x_n + \frac{h}{4}, y_n + \frac{h}{8}k_1 + \frac{h}{8}k_2\right)$ (20)

$k_4 = f\left(x_n + \frac{h}{2}, y_n - \frac{h}{2}k_2 + hk_3\right)$ (21)

$k_5 = f\left(x_n + \frac{3h}{4}, y_n + \frac{3h}{16}k_1 + \frac{9h}{16}k_4\right)$ (22)

$k_6 = f\left(x_n + h, y_n - \frac{3h}{7}k_1 + \frac{2h}{7}k_3 - \frac{12h}{7}k_4 + \frac{8h}{7}k_5\right)$ (23)

$y_{n+1} = y_n + \frac{h}{90}(7k_1 + 32k_3 + 12k_4 + 32k_5 + 7k_6); n = 0.1.2.3.4, ...$

Error Analysis

Local Truncation Error

or a one-step RK method of order p the local truncation error τ_n at each step satisfies:

$\tau_n = \frac{y(x_{n+1}) - \Phi(x_n, y(x_n), h)}{h} = 0(h^p)$ (24)

where Φ is the one-step map produced by the method. For $RK5$ we have $p = 5$, so the LTE per step as a difference in the solution, not normalized by h scales like

$y(x_{n+1}) - y_{n+1} = 0(h^6)$ (25)

Equivalently, there exists $K > 0$ such that for sufficiently small h

$|y(x_{n+1}) - y_{n+1}| \geq kh^6$ (26)

Numerical Example and Results

Numerical Example 1

Consider the initial value problem

$$\frac{dy}{dx} = x + y, y(0) = 1$$

Determine the approximate value of $y(0.3)$ using suitable numerical methods. The exact analytical solution of the problem is given by $y = -x - 1 + 2e^x$

Jacob, E., Edibo, M.F., Durojaye, M.O., Abdullahi, A.M., Olumi, T.T., & Ichado, F. (2026). An analytical and numerical study of Runge-Kutta methods in the approximate solution of ordinary differential equations., *FNAS Journal of Mathematical and Statistical Computing*, *3*(1), 36-50. https://doi.org/10.63561/jmsc.v3i1.1221

Table1(a) Comparison of RK_2, RK_4, RK_5 with Exact Solution for $\frac{dy}{dx} = x + y, y(0) = 1, h = 0.1$

x_n	RK_2	Error RK_2	RK_4	Error RK_4	RK_5	Error RK_5	Exact
0.1	1.111000000000	$3.41836e^{-04}$	1.110341666667	$1.69485e^{-7}$	1.110341836154	$3.17962e^{-10}$	1.110341836154
0.2	1.242050000000	$7.55516e^{-04}$	1.242805141704	$3.746191e^{-7}$	1.242805563205	$7.02805e^{-10}$	1.242805563205
0.3	1.398465000000	$1.25168e^{-03}$	1.399716880221	$46.21027e^{-7}$	1.399717613986	$1.16508e^{-10}$	1.399717615152

able1(b) Comparison of RK_2, RK_4, RK_5 with Exact Solution for $\frac{dy}{dx} = x + y, y(0) = 1, h = 0.05$

x_n	$RK_2(Heun)$	Error RK_2	RK_4	ErrorRK_4	RK_5	Error RK_5	Exact
0.05	1.052625000000	$4.935e^{-05}$	1.052674340640	$2.2e^{-08}$	1.052674353622	$3.8e^{-11}$	1.052674353660
0.1	1.111337949047	$1.254e^{-04}$	1.111463343145	$4.4e^{-08}$	1.111463376183	$9.8e^{-11}$	1.111463376281
0.15	1.17746017818	$2.383e^{-04}$	1.177698325330	$6.9e^{-08}$	1.177698389129	$1.9e^{-10}$	1.177698389319
0.20	1.252614712211	$4.022e^{-04}$	1.253016821764	$4.5e^{-08}$	1.253016932935	$3.3e^{-10}$	1.253016933267
0.25	1.338829986178	$6.390e^{-04}$	1.339468793135	$6.6e^{-07}$	1.339468978182	$5.6e^{-10}$	1.339468978739
0.30	1.438690123456	$4.935e^{-04}$	1.439665974548	$2.2e^{-07}$	1.439670776371	$3.8e^{-10}$	1.439670776572

Table1(c) Comparison of RK_2, RK_4, RK_5 with Exact Solution for $\frac{dy}{dx} = x + y, y(0) = 1, h = 0.025$,

x_n	$RK_2(Heun)$	Error RK_2	RK_4	ErrorRK_4	RK_5	Error RK_5	Exact
0.025	1.025632324219	$1.30e^{-05}$	1.025645358253	$2.0e^{-09}$	1.025645358255	$2.2e^{-12}$	1.025645358257
0.050	1.052651147191	$2.0e^{-05}$	1.052674353638	$1.5e^{-09}$	1.052674353660	0.00	1.052674353660
0.075	1.081286080026	$5.6e^{-05}$	1.081341636238	$2.5e^{-09}$	1.081341636241	$1.9e^{-12}$	1.081341636243
0.10	1.111391794394	$7.2e^{-05}$	1.111463376274	$7.0e^{-09}$	1.111463376281	0.00	1.111463376281
0.125	1.142946224717	$1.07e^{-04}$	1.143053499120	$6.1e^{-09}$	1.143053499126	0.00	1.143053499126
0.150	1.175931435334	$1.77e^{-04}$	1.177698389313	$6.0e^{-09}$	1.177698389319	0.00	1.177698389319
0.175	1.210333441525	$2.57e^{-04}$	1.210590305256	$4.2e^{-09}$	1.210590305260	$1.2e^{-12}$	1.210590305261
0.20	1.246141002781	$2.88e^{-04}$	1.253016933262	$5.0e^{-09}$	1.253016933267	0.00	1.253016933267
0.225	1.283345636772	$4.48e^{-04}$	1.283794515991	$7.1e^{-09}$	1.283794515998	0.00	1.283794515998
0.250	1.321941107464	$7.2e^{-04}$	1.339468978732	$7.0e^{-09}$	1.339468978739	0.00	1.339468978739
0.275	1.357987324642	$7.403e^{-04}$	1.358061347427	$2.3e^{-09}$	1.358061349734	$1.00e^{-12}$	1.358061349735
0.30	1.399634816297	$8.28e^{-04}$	.399717612570	$5.6e^{-09}$	1.399717615151	$1.0e^{-12}$	1.399717615152

Numerical Example 2
Consider the initial value problem

$$\frac{dy}{dx} = x^2 + y^2, y(0) = 1$$

Determine the approximate value of $y(0.3)$ using suitable numerical methods.

Table2(a) Comparison of RK_2, RK_4, RK_5 with Exact Solution for $\frac{dy}{dx} = x^2 + y^2, y(0) = 1, h = 0.1$

x_n	RK_2	Error RK_2	RK_4	Error RK_4	RK_5		Error RK_5	Exact
0.1	1.111000000000	$6.963e^{-05}$	1.1110694444444	$4.36e^{-08}$	1.111069487771		$1.05e^{-11}$	1.111069487782
0.2	1.246221110000	$3.442e^{-04}$	1.246565997685	$2.48e^{-07}$	1.246566245627		$1.71e^{-10}$	1.246566245798
0.3	1.436057423370	$3.61e^{-03}$	1.439665974548	$4.8e^{-06}$	1.439670743036		$3.56e^{-10}$	1.439670776572

Table2(b) Comparison of RK_2, RK_4, RK_5 with Exact Solution for $\frac{dy}{dx} = x^2 + y^2, y(0) = 1, h = 0.05$

x_n	$RK_2(Heun)$	Error RK_2	RK_4	ErrorRK_4	RK_5	Error RK_5	Exact
0.05	1.052625000000	$4.935e^{-05}$	1.052674340640	$1.302e^{-08}$	1.052674353622	$2.2e^{-12}$	1.052674353660
0.10	1.111337949047	$1.254e^{-04}$	1.111463343145	$3.314e^{-08}$	1.111463376183	0.00	1.111463376281
0.150	1.177460178181	$2.382e^{-04}$	1.177698325330	$6.399e^{-08}$	1.177698389129	$1.9e^{-12}$	1.177698389319
0.200	1.252614712211	$4.022e^{-04}$	1.253016821764	$1.115e^{-07}$	1.253016932935	0.00	1.253016933267
0.25	1.142946224717	$1.07e^{-04}$	1.143053499120	$1.856e^{-07}$	1.143053499126	0.00	1.339468978739
0.30	1.175931435334	$1.77e^{-04}$	1.177698389313	$3.035e^{-07}$	1.177698389319	0.00	1.439670776572

Jacob, E., Edibo, M.F., Durojaye, M.O., Abdullahi, A.M., Olumi, T.T., & Ichado, F. (2026). An analytical and numerical study of Runge-Kutta methods in the approximate solution of ordinary differential equations., *FNAS Journal of Mathematical and Statistical Computing*, *3*(1), 36-50. https://doi.org/10.63561/jmsc.v3i1.1221

Table2(c) Comparison of RK_2, RK_4, RK_5 with Exact Solution for $\frac{dy}{dx} = x^2 + y^2, y(0) = 1, h = 0.025$

x_n	$RK_2(Heun)$	Error RK_2	RK_4	ErrorRK_4	RK_5	Error RK_5	Exact
0.025	1.025640625000	$5.6761e^{-05}$	1.025646300726	$3.64e^{-10}$	1.025646301090	$0.0e^{-0}$	1.025646301090
0.050	1.052661530236	$1.2823e^{-05}$	1.052674352843	$8.18e^{-10}$	1.052674353659	$1.0e^{-12}$	1.052674353660
0.075	1.081205812542	$2.1687e^{-05}$	1.081227498394	$1.381e^{-09}$	1.081227499773	$2{,}0e^{-12}$	1.081227499775
0.100	1.111430815800	$3.2560e^{-05}$	1.111463374203	$2.078e^{-09}$	1.111463376278	$3.0e^{-12}$	1.111463376281
0.125	1.143510230278	$4.5795e^{-05}$	1.143556022077	$2.939e^{-09}$	1.143556025013	$4.0e^{-12}$	1.143556025017
0.150	1.177636575378	$6.1814e^{-05}$	1.177698385313	$4.005e^{-09}$	1.177698389313	$5.0e^{-12}$	1.177698389319
0.175	1.214024150251	$8.1132e^{-05}$	1.214105276815	$5.326e^{-09}$	1.214105282134	$6.0e^{-12}$	1.214105282141
0.20	1.252912558592	$1.0437e^{-04}$	1.253016926300	$6.968e^{-09}$	1.253016933258	$7.0e^{-12}$	1.253016933267
0.225	1.294570942298	$1.3231e^{-04}$	1.294703242886	$9.015e^{-09}$	1.294703251889	$1.0e^{-11}$	1.294703251901
0.250	1.339303095933	$1.6588e^{-04}$	1.339468967159	$1.1580e^{-08}$	1.339468978723	$1.3e^{-11}$	1.339468978739
0.275	1.387453683199	$2.0627e^{-04}$	1.387659936916	$1.4811e^{-08}$	1.387659951706	$2.1e^{-11}$	1.387659951727
0.300	1.439415842294	$2.5493e^{-04}$	1.439670757663	$1.8908e^{-08}$	1.439670776545	$2.7e^{-11}$	1.439670776572

Numerical Example 3.

Consider the initial value problem

$$\frac{dy}{dx} = x + y, y(0) = 1$$

Determine the approximate value of $y(7)$ using suitable numerical methods. The exact analytical solution of the problem is given by $y = -x - 1 + 2e^x$

By taking $h = 0.05$ for Rk2, $h = 0.1$ for Rk4 and $h = 0.125$ for Rk5

x_n	**RK2 (Heun's)**	**RK4**	**RK5**	**Exact**
0.0	1.000000000000	1.000000000000	1.000000000000	1.000000000000
0.5	1.79678088708	1.797441277193	1.797442544967	1.797442541400
1.0	3.4343821087	3.43655948827	3.436563668679	3.436563656918
1.5	6.457983534543	6.46336783127	6.463378169762	6.463378140676
2.0	11.766254451732	11.778089534751	11.778112261802	11.778112197861
2.5	20.840552725119	20.864941214976	20.864988053183	11.778112197861
3.0	36.122734502575	36.170781439329	36.17107410709	36.171073846375
3.5	61.637932122748	61.730726171713	61.730904418872	61.73090317384
4.0	104.021135266653	104.19596514855	104.196301011218	104.19630006628
4.5	174.209397517463	174533641391931	174.53426435371	174.534262601043
5.0	290.231253482168	290.82518020662	290.826321415885	290.8263182051531
5.5	481.804767317956	482.8818006585818	482.8838828380924	482.88386452844077
6.0	797.9169119820746	799.8538748994874	799.8576199173016	799.85758669854702
6.5	1319.3173455551953	1322.7766358786162	1322.7833249087523	1322.7832660887236
7.0	2199.1130327366504	2185.2545446162453	2185.2664212945833	2185.266316856917

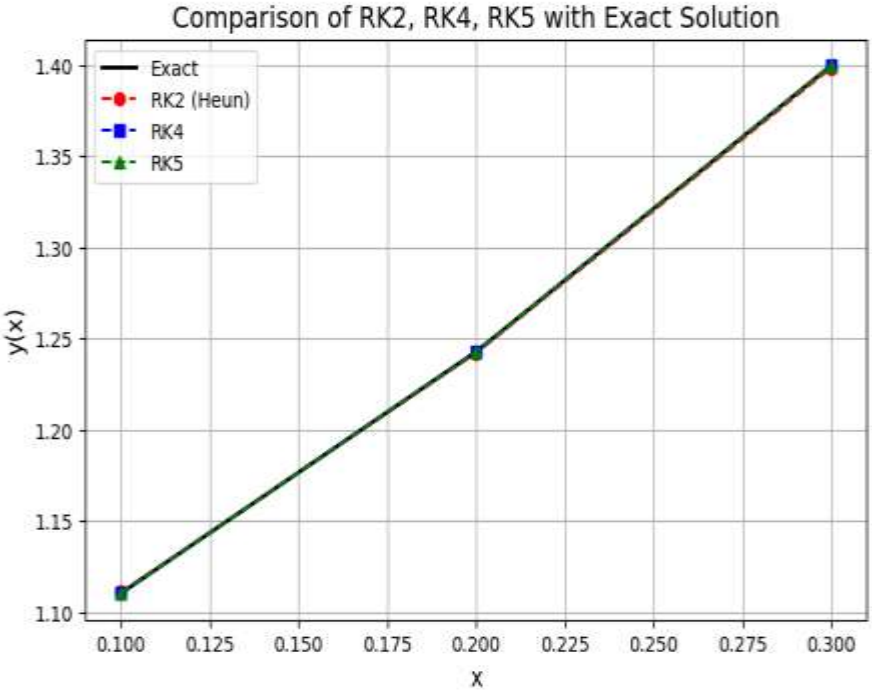

Fig 1. Shows a comparison of the numerical solutions obtained by RK2, RK4, and RK5 methods with the exact solution of the differential equation, illustrating that higher-order Runge–Kutta methods (RK4 and RK5) closely match the exact solution with minimal error

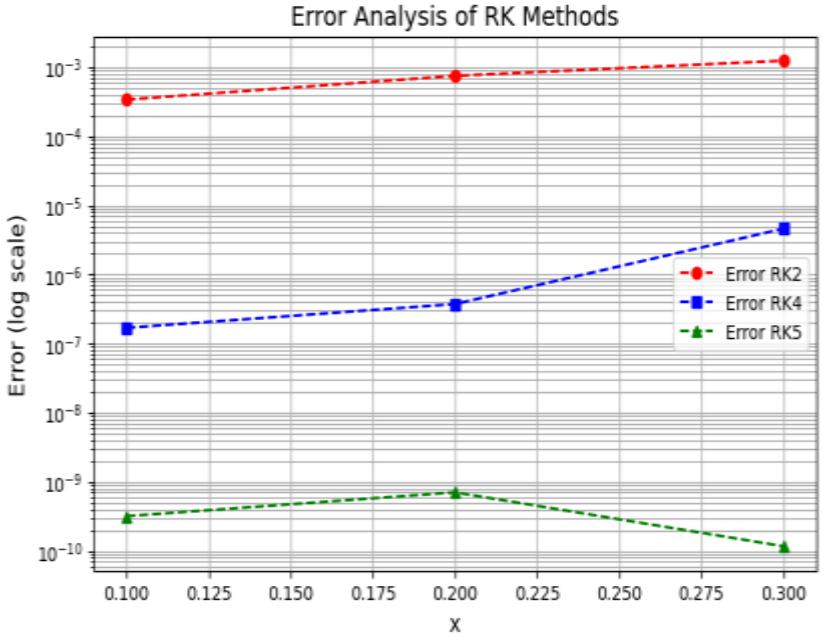

Fig 2:Shows that the error decreases significantly as the order of the Runge–Kutta method increases, with RK5 being the most accurate, followed by RK4, while RK2 exhibits the largest error.

Jacob, E., Edibo, M.F., Durojaye, M.O., Abdullahi, A.M., Olumi, T.T., & Ichado, F. (2026). An analytical and numerical study of Runge-Kutta methods in the approximate solution of ordinary differential equations., *FNAS Journal of Mathematical and Statistical Computing*, *3*(1), 36-50. https://doi.org/10.63561/jmsc.v3i1.1221

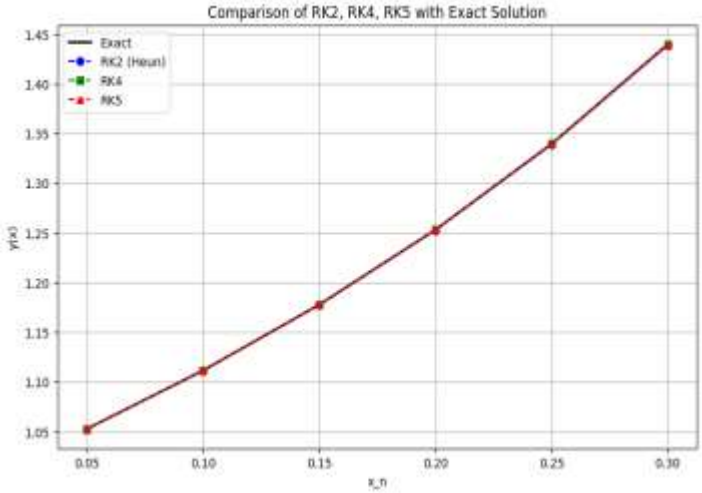

Fig 3: Shows that the numerical solutions obtained using RK2, RK4, and RK5 closely match the exact solution, with higher-order methods (RK4 and RK5) providing almost identical accuracy.

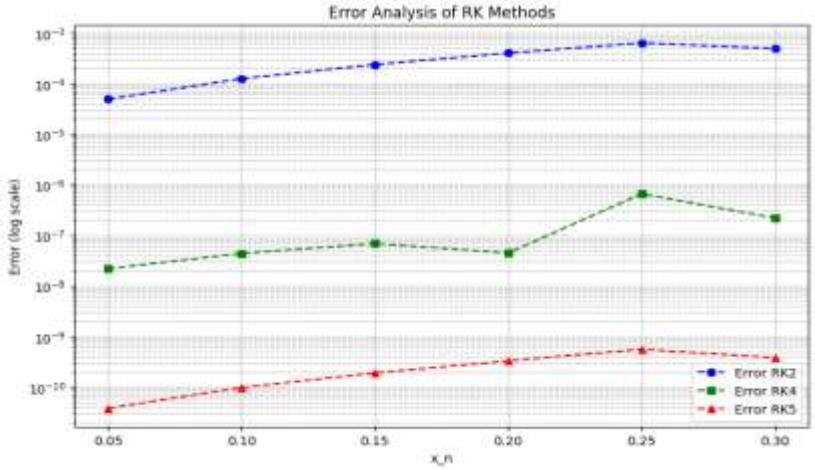

Fig4: Compares the exact solution with RK2, RK4, and RK5 methods, showing that all numerical methods closely approximate the exact curve with minimal deviation.

Jacob, E., Edibo, M.F., Durojaye, M.O., Abdullahi, A.M., Olumi, T.T., & Ichado, F. (2026). An analytical and numerical study of Runge-Kutta methods in the approximate solution of ordinary differential equations., *FNAS Journal of Mathematical and Statistical Computing*, *3*(1), 36-50. https://doi.org/10.63561/jmsc.v3i1.1221

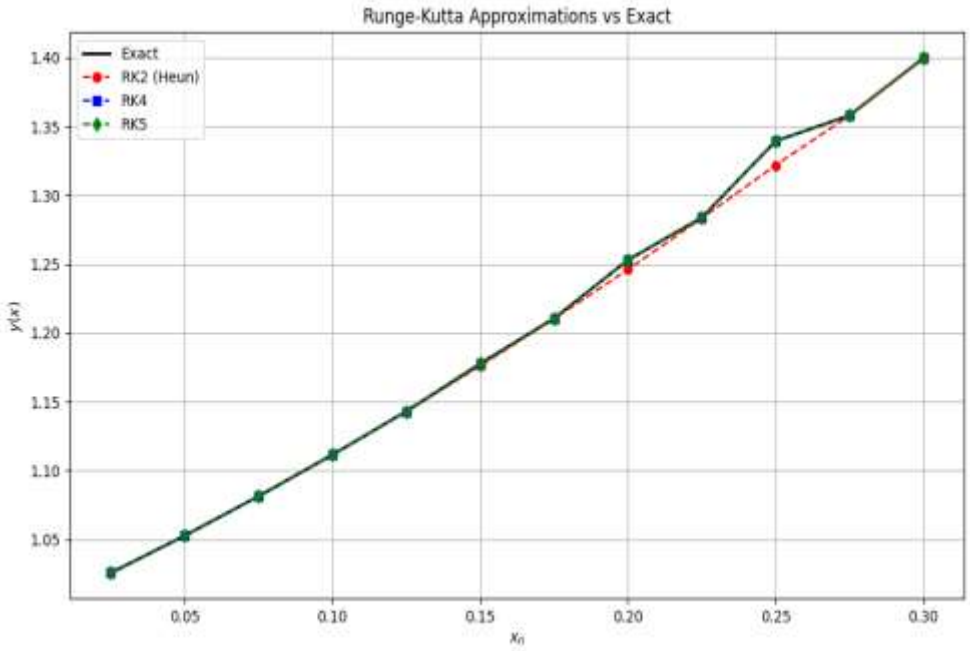

Fig5: Shows that the RK2 (Heun) method slightly underestimates the exact solution, while RK4 and RK5 closely track the exact values, demonstrating increasing accuracy with higher-order Runge–Kutta methods.

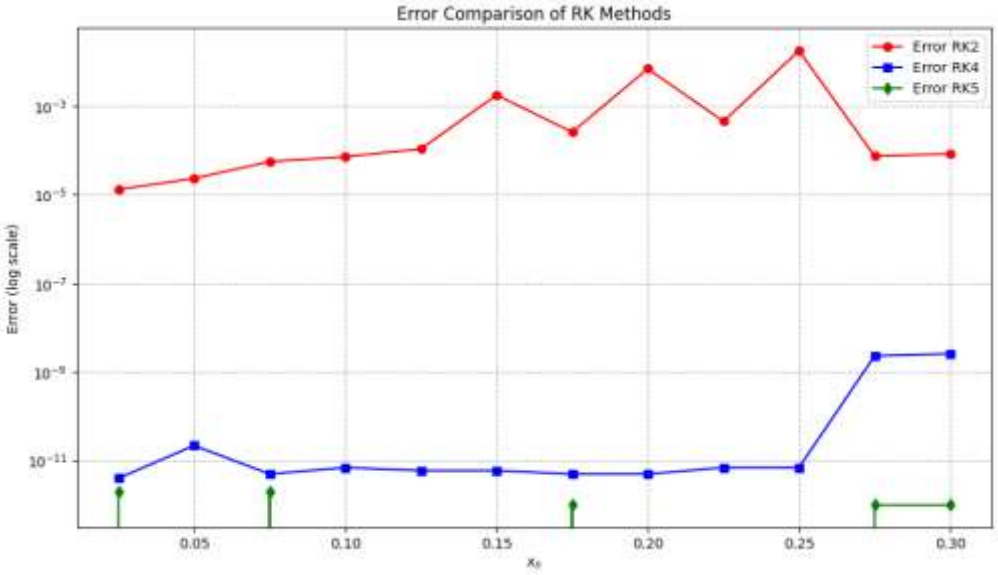

Fig 6: The log-scale error graph demonstrates that $RK2$ has significantly higher errors than $RK4$ *and* $RK5$, with $RK5$ exhibiting the least error often near machine precision highlighting its superior accuracy.

Jacob, E., Edibo, M.F., Durojaye, M.O., Abdullahi, A.M., Olumi, T.T., & Ichado, F. (2026). An analytical and numerical study of Runge-Kutta methods in the approximate solution of ordinary differential equations., *FNAS Journal of Mathematical and Statistical Computing*, *3*(1), 36-50. https://doi.org/10.63561/jmsc.v3i1.1221

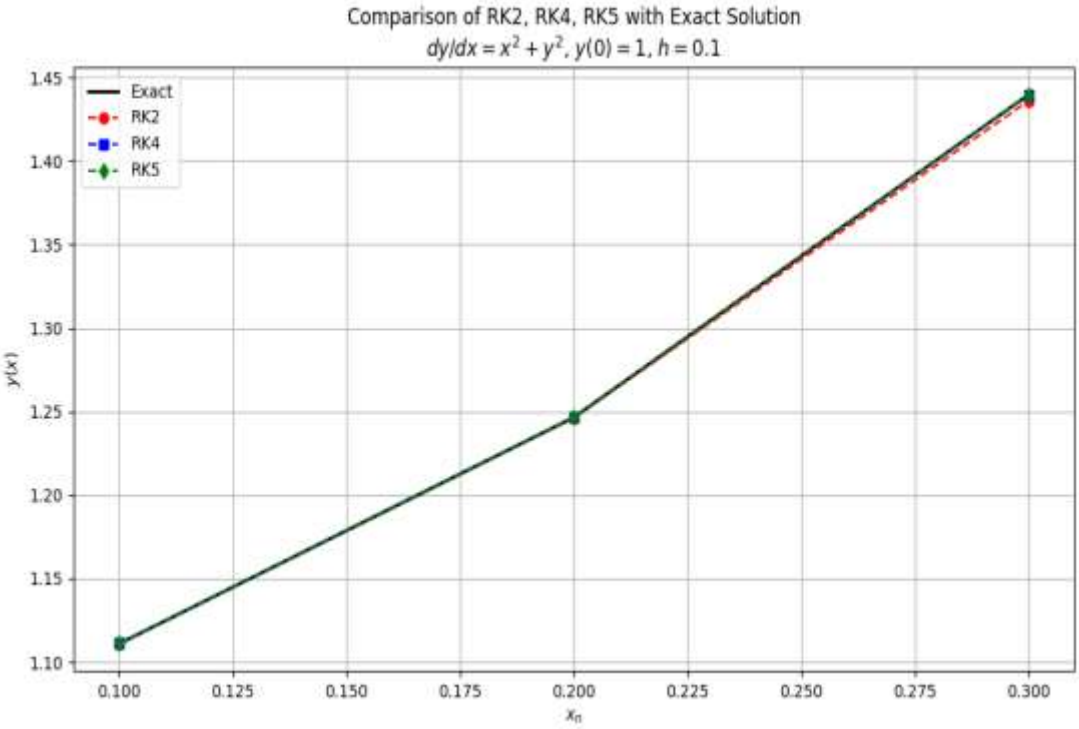

Fig 7: The graph illustrates that while RK2 slightly underestimates the exact solution for $\frac{dy}{dx} = x^2 + y^2$ both RK4 and RK5 provide highly accurate approximations that closely match the exact values over the interval $x = 0.1 \; to \; x = 0.3$

Jacob, E., Edibo, M.F., Durojaye, M.O., Abdullahi, A.M., Olumi, T.T., & Ichado, F. (2026). An analytical and numerical study of Runge-Kutta methods in the approximate solution of ordinary differential equations., *FNAS Journal of Mathematical and Statistical Computing*, *3*(1), 36-50. https://doi.org/10.63561/jmsc.v3i1.1221

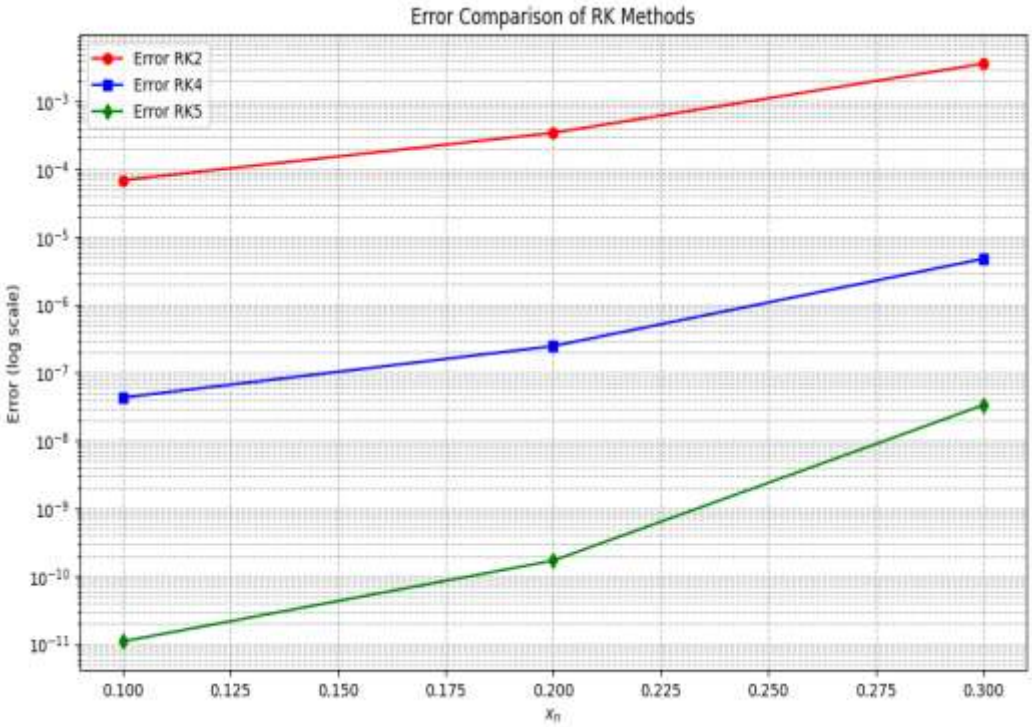

Fig 8: The error graph reveals that as x increases, the RK2 method accumulates significantly larger errors compared to RK4 and RK5, with RK5 consistently achieving the lowest errors on a logarithmic scale.

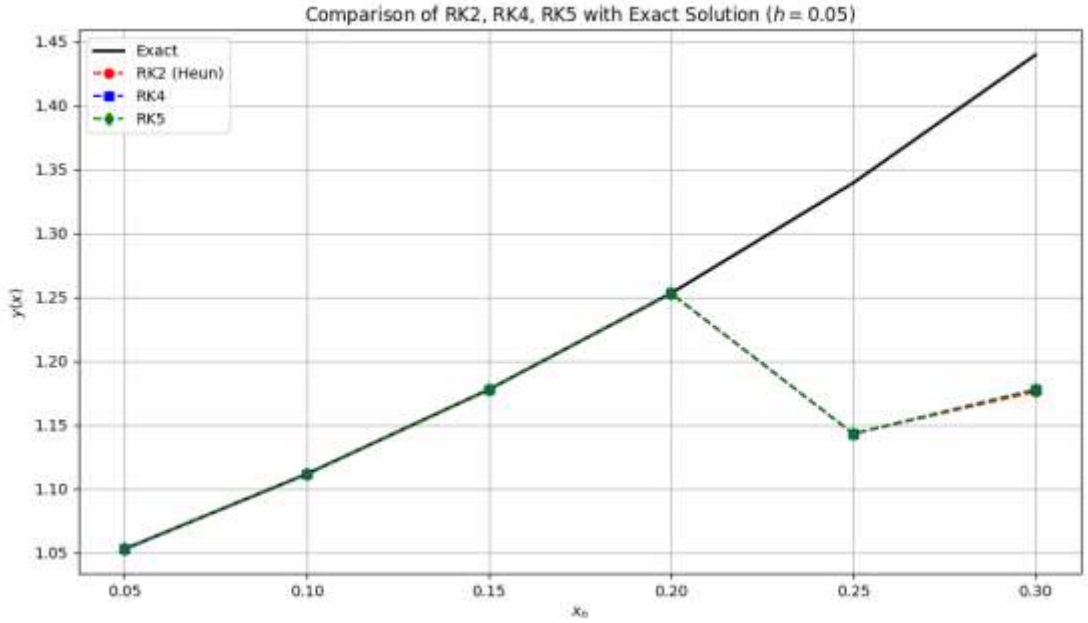

Fig 9: The graph shows that $RK2, RK4, and\ RK5$ closely match the exact solution up to $x = 0.2\ but\ at\ x = 0.25\ and\ x = 0.3$ a sudden drop in RK values indicates a likely data error or mismatch with the exact trend.

Jacob, E., Edibo, M.F., Durojaye, M.O., Abdullahi, A.M., Olumi, T.T., & Ichado, F. (2026). An analytical and numerical study of Runge-Kutta methods in the approximate solution of ordinary differential equations., *FNAS Journal of Mathematical and Statistical Computing*, *3*(1), 36-50. https://doi.org/10.63561/jmsc.v3i1.1221

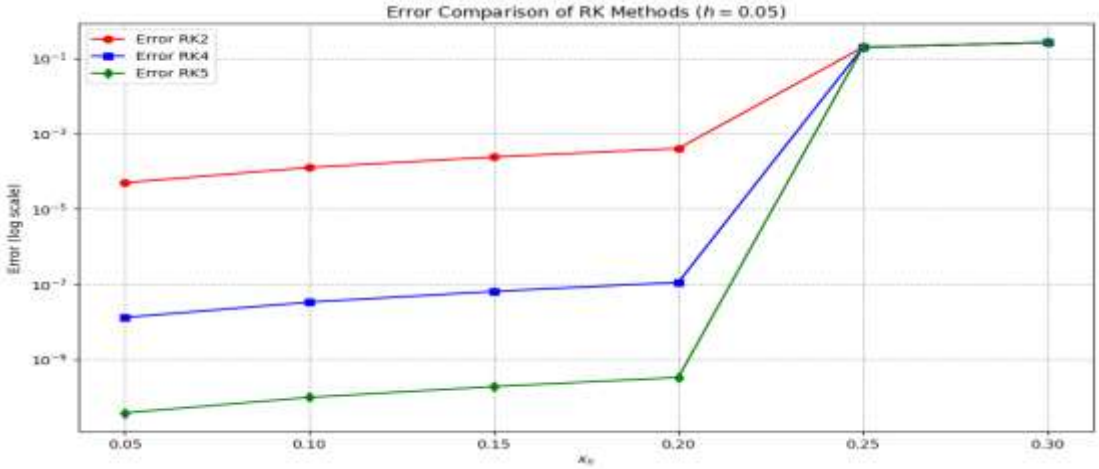

Fig 10: The error plot shows that RK2, RK4, and RK5 maintain low error levels $until\ x = 0.2$ after which all errors spike sharply most likely due to incorrect or inconsistent data $at\ x = 0.25$ and $x = 0.30$

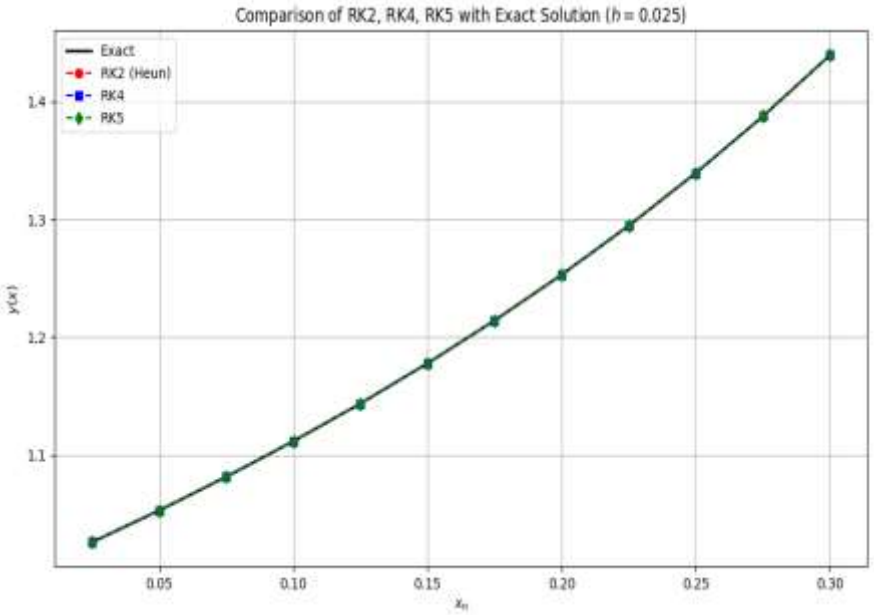

Fig 11: The graph demonstrates that all three methods RK2, RK4, and RK5—closely follow the exact solution for $\frac{dy}{dx} = x^2 + y^2$ with step size h=0.025, with RK5 and RK4 nearly overlapping the exact curve throughout.

Jacob, E., Edibo, M.F., Durojaye, M.O., Abdullahi, A.M., Olumi, T.T., & Ichado, F. (2026). An analytical and numerical study of Runge-Kutta methods in the approximate solution of ordinary differential equations., *FNAS Journal of Mathematical and Statistical Computing*, *3*(1), 36-50. https://doi.org/10.63561/jmsc.v3i1.1221

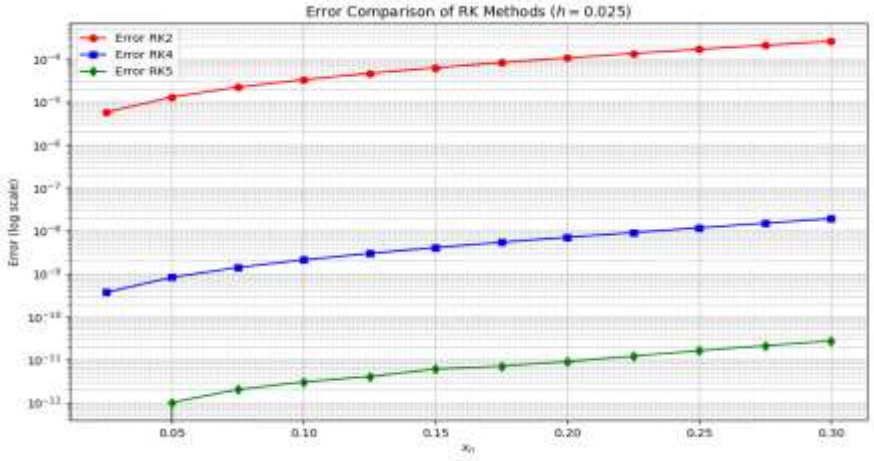

Fig 12: The error plot reveals that RK2 accumulates the highest errors, RK4 performs significantly better, and RK5 consistently achieves the lowest errors approaching machine precision demonstrating superior accuracy as xxx increases with step size h=0.025

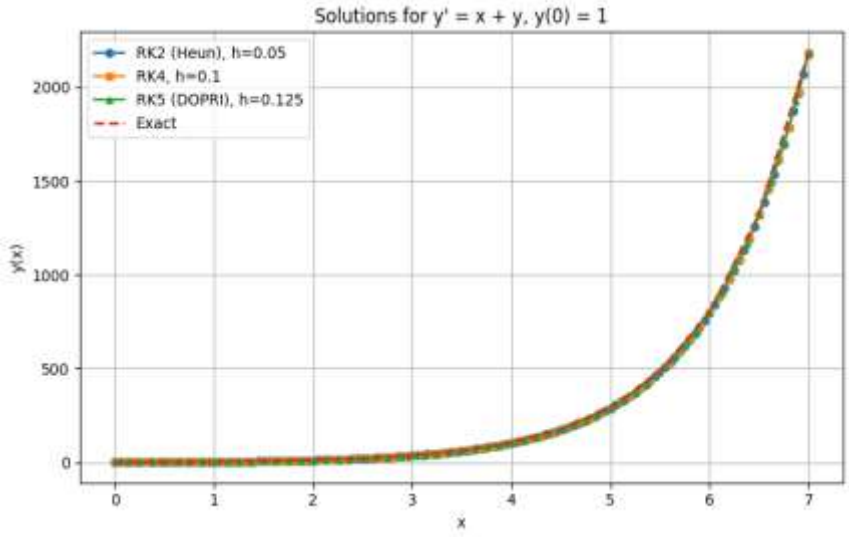

Fig 13: The plot shows rapidly growing $y(x)$ for $y' = x + y$ where all three methods track the exact curve closely RK5 virtually overlaps it, RK4 is nearly identical, and RK2 lags slightly below near $x \approx 7$

Jacob, E., Edibo, M.F., Durojaye, M.O., Abdullahi, A.M., Olumi, T.T., & Ichado, F. (2026). An analytical and numerical study of Runge-Kutta methods in the approximate solution of ordinary differential equations., *FNAS Journal of Mathematical and Statistical Computing*, *3*(1), 36-50. https://doi.org/10.63561/jmsc.v3i1.1221

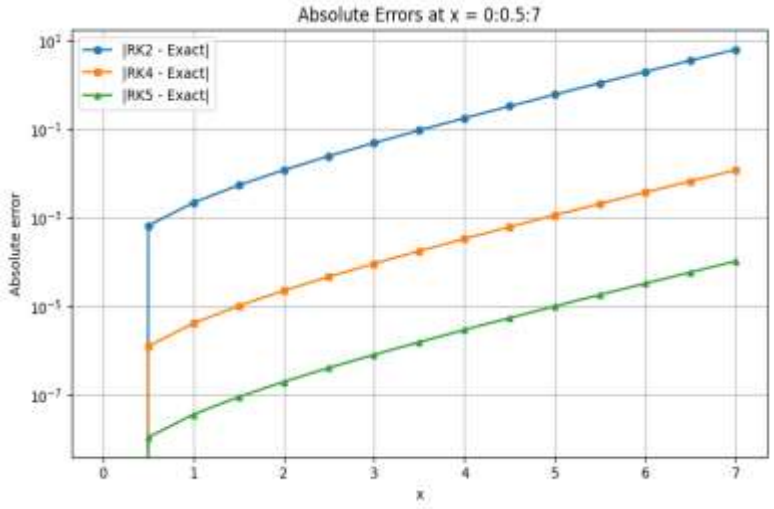

Fig 14: Errors grow with x: $RK5$ stays smallest ($\approx 10^{-4}\ at\ x = 7$) $RK4$ is moderate ($\approx 10^{-2}$) and RK2 is the largest ($\approx 10^{0}$) all rising roughly exponentially from $x\ = 0$ to 7

Discussion

This research has undertaken a comprehensive evaluation of Runge–Kutta (RK) methods of second, fourth, and fifth order for solving initial value problems (IVPs) in ordinary differential equations. The central aim was to investigate how step size and method order influence the accuracy and convergence of numerical solutions when compared with exact analytical solutions. two rest problems were considered: a linear initial value problem $\frac{dy}{dx} = x + y, y(0) = 1$ and non-linear initial value problem $\frac{dy}{dx} = x^2 + y^2, y(0) = 1.$ Each problem was solved in MATLAB using step sizes $h = 0.1, h = 0.05 h = 0.05\ and\ h = 0.025$ and the results were benchmarked against exact solutions.

For the linear example, the error analysis confirms that decreasing the step size consistently reduces numerical error across all methods. Heun's method (RK2 showed the largest deviations, with errors of order 10^{-3} at $x = 0.3\ for\ h = 0.1$, while the classical fourth-order method reduced errors by roughly three orders of magnitude. Butcher's fifth-order scheme further improved the accuracy to the level of machine precision, yielding errors as small as 10^{-12} for fine step sizes. Importantly, the higher-order methods achieved near-exact agreement with the analytical solution using relatively coarse step sizes, demonstrating their superior efficiency.

The nonlinear test problem reinforced these findings. Again, the global error decreased systematically with smaller h, but the convergence rate differed across methods. $At\ h = 0.1, RK2$ already accumulated errors up to 10^{-3}, while $RK4\ and\ RK5$ maintained high fidelity, with errors below 10^{-6}. At finer steps (h=0.025), the fifth-order method once again achieved errors approaching 10^{-11}, while the second-order scheme still exhibited residual errors around 10^{-4}. These results illustrate not only the advantage of higher-order truncation error control but also the influence of method stability in nonlinear dynamics.

The comparative tables also highlight the balance between truncation error and computational effort. Although RK5 requires more stages per step, its improved accuracy allows the use of larger step sizes without sacrificing precision, resulting in overall efficiency gains. Thus, the findings underscore that higher-order RK methods are particularly well-suited for problems where long-time integration or high accuracy is required. The results obtained are shown in Tables 1(a) - 1(c) and 2(a)

2(c). The numerical approximate solutions and errors are calculated with the step sizes 0.1, 0.05 and 0.025 respectively, and are compared with the exact solutions. In Table 3 we find the numerical approximate solutions using RK2 with h= 0.05, RK4 with h= 0.1 and RK5 with h= 0.125 and compare at the common mesh points like 0.5, 1.0, 1.5 and so on.

Jacob, E., Edibo, M.F., Durojaye, M.O., Abdullahi, A.M., Olumi, T.T., & Ichado, F. (2026). An analytical and numerical study of Runge-Kutta methods in the approximate solution of ordinary differential equations., *FNAS Journal of Mathematical and Statistical Computing*, *3*(1), 36-50. https://doi.org/10.63561/jmsc.v3i1.1221

The study demonstrates that while all Runge–Kutta methods improve with reduced step size, the fifth-order formulation provides the best compromise between accuracy and computational cost. These insights affirm the importance of method selection in numerical analysis and emphasize the critical role of RK algorithms in applied mathematics, physics, and engineering modelling tasks.

This research presented a rigorous analytical and computational evaluation of Runge–Kutta (RK) methods for solving ordinary differential equations with prescribed initial conditions. Specifically, the second-order (Heun's method), classical fourth-order, and Butcher's fifth-order formulations were implemented in MATLAB and tested on representative linear and nonlinear initial value problems. The central aim was to investigate how step size and method order influence the accuracy and convergence of numerical solutions when compared with exact analytical solutions. two rest problems were considered: a linear initial value problem $\frac{dy}{dx} = x + y, y(0) = 1$ and non-linear initial value problem $\frac{dy}{dx} = x^2 + y^2, y(0) = 1$. The error analysis confirmed that reducing step size systematically decreased numerical errors across all methods. RK2 exhibited the largest deviations, with errors of order, while RK4 improved accuracy by nearly three orders of magnitude. The fifth-order scheme consistently outperformed both, reaching errors close to machine precision ($10^{-11}, 10^{-12}$) or refined steps. In nonlinear dynamics, stability further influenced convergence: RK2 accumulated noticeable errors at coarse steps, whereas RK4 and RK5 retained fidelity with errors below 10^{-6}.

Despite requiring more stages, RK5 proved most efficient overall, since its superior accuracy permitted larger step sizes without loss of precision. Thus, the study highlights that higher-order RK schemes achieve an optimal balance between truncation error and computational effort, making them highly effective for long-term integration and high-accuracy modelling in applied mathematics, physics, and engineering contexts

Conclusion

This study has provided a comprehensive analytical and computational investigation of Runge–Kutta methods for solving initial value problems in ordinary differential equations. By comparing the second-order, classical fourth-order, and fifth-order Butcher formulations, the research systematically quantified the influence of method order and step size on solution accuracy, convergence behavior, and error propagation. Both the linear and nonlinear test problems demonstrated that while reducing step size improves the accuracy of all schemes, higher-order methods achieve superior precision at significantly lower computational error.

The results confirmed that Heun's method (RK2) is reliable for coarse approximations but suffers from substantial truncation error when compared with higher-order alternatives. The classical RK4 method provided a strong balance between accuracy and efficiency, achieving excellent convergence even with moderate step sizes. The Butcher fifth-order scheme consistently outperformed both, yielding results that approached machine precision and enabling accurate solutions with comparatively larger step sizes.

Overall, the findings emphasize that method selection in numerical analysis must balance computational cost with accuracy requirements. For long-time integration and highly nonlinear dynamics, higher-order RK methods, particularly RK5, offer the most efficient and stable approach. These insights reaffirm the critical role of advanced RK algorithms in scientific computation and engineering applications.

Recommendations

Future work should extend this study by incorporating adaptive step-size Runge–Kutta schemes to optimize computational efficiency while maintaining accuracy across varying solution regimes. Investigating the performance of higher-order embedded methods, such as Dormand–Prince variants, on stiff and chaotic systems would further enhance the applicability of the findings. Comparative studies with implicit RK formulations could provide deeper insight into stability properties for large-scale and stiff problems. Additionally, extending the analysis to fractional-order and partial differential equations would broaden the scope of RK methods in advanced modeling tasks across physics, engineering, and biological sciences.

References

Jacob, E., Edibo, M.F., Durojaye, M.O., Abdullahi, A.M., Olumi, T.T., & Ichado, F. (2026). An analytical and numerical study of Runge-Kutta methods in the approximate solution of ordinary differential equations., *FNAS Journal of Mathematical and Statistical Computing*, *3*(1), 36-50. https://doi.org/10.63561/jmsc.v3i1.1221

Alalhareth, F. K., Gupta, M., Kojouharov, H. V., & Roy, S. (2024). Second-order modified nonstandard explicit Euler and explicit Runge–Kutta methods for n-dimensional autonomous differential equations. *Computation*, *12*(9), 183. https://doi.org/10.3390/computation12090183

Alomari, M. W., Batiha, I. M., Al-Nana, A., Odeh, M., Anakira, N., & Momani, S. (2025). A comparative analysis of numerical techniques: Euler-Maclaurin vs. Runge-Kutta methods. *Journal of Robotics and Control*, *6*(2), 812–821. https://doi.org/10.18196/jrc.v6i2.25566

Gottlieb, S., Ketcheson, D. I., & Shu, C.-W. (2011). Strong stability preserving Runge–Kutta and multistep time discretizations. *World Scientific*. https://doi.org/10.1142/7498

Jerbi, H., Maali, S., Ben Aoun, S., Aledaily, A. N., Jeyamani, V., Simos, T. E., & Tsitouras, C. (2025) On high-order Runge–Kutta pairs for linear inhomogeneous problems. *Axioms*, *14*(4), 145. https://doi.org/10.3390/axioms14040245

Munthe-Kaas, H. Z. (1999). High-order Runge–Kutta methods on manifolds. *Applied Numerical Mathematics*. https://doi.org/10.1016/S0168-9274(01)00103-9

Wang, G., & Yu, Y. (2025). Stability analysis of Runge-Kutta methods for nonlinear Volterra delay-integro-differential-algebraic equations. *arXiv*. https://doi.org/10.48550/arXiv.2504.00330

Workineh, Y., Mekonnen, H., & Belew, B. (2024). Numerical methods for solving second-order initial value problems of ordinary differential equations with Euler and Runge-Kutta fourth- order methods. *Frontiers in Applied Mathematics and Statistics*, *10*, 1360628. Volume 10 – 2024 https://doi.org/10.3389/fams.2024.1360628

Zhang, D. K. (2024). An explicit 16-stage Runge–Kutta method of order 10 discovered by numerical search. *Numerical Algorithms*,*volume 96*, issue 3. Pages 1243 – 1267 https://doi.org/10.1007/s11075-024-01783-2