

Accuracy–Efficiency Trade-Offs in Euler and Runge–Kutta Fourth-Order Methods: A Comparative Error Analysis

Abdulrahman O. Audu, Hasheem Abdul Mohammed, Okigbo Chinwendu Jacinta and Tahiru Yusuf Adejo

Department of Mathematics and Statistics, Federal Polytechnic, Nasarawa, Nasarawa State, Nigeria.

*Corresponding authors' email: a4audu@gmail.com

ABSTRACT

Numerical techniques are essential for approximating solutions to ordinary differential equations (ODEs) that lack closed-form solutions. Although Euler's method and the fourth-order Runge–Kutta (RK4) scheme are frequently compared, most studies report accuracy at a single step size, which cannot by itself demonstrate a genuine accuracy–efficiency trade-off. This paper addresses that gap by solving the benchmark initial value problem $\frac{dy}{dx} = y + x, y(0) = 1$, over $[0, 1]$ with both methods across five step sizes, $h = 0.5, 0.1, 0.05, 0.01$ and 0.001 , tracking absolute error against the number of function evaluations required at each h . At $h = 0.1$, Euler required 10 evaluations and produced $y(1) = 3.187485$ (absolute error 2.49×10^{-1}), while RK4 required 40 evaluations and produced $y(1) = 3.436559$ (absolute error 4.17×10^{-6}). Across the full range of step sizes, Euler's error decreases roughly in proportion to h , consistent with its first-order global accuracy, whereas RK4's error decreases roughly in proportion to h^4 , so RK4 attains errors several orders of magnitude smaller than Euler even after its four-fold higher per-step cost is taken into account. A work-precision diagram of error against function evaluations places RK4's curve consistently below Euler's across every tested h , confirming that RK4 dominates Euler over the tested range rather than only at one operating point. Maple 17 code implementing both schemes with an adjustable step-size parameter is provided to support reproduction and extension of these results. These findings give multi-point, quantitative support for preferring RK4 whenever accuracy per function evaluation matters, while Euler remains attractive only when a single, very cheap evaluation is the overriding constraint.

Received: 04 Sept. 2026

Accepted: 12 Sept. 2026

Published: 07 Oct. 2026

Keywords:

Euler Scheme; RK4, Numerical Approximation, Ordinary Differential Equations; Initial Value Problems, Error Analysis, Work-Precision Diagram, Accuracy vs. Efficiency

INTRODUCTION

Initial value problems of the form $\frac{dy}{dx} = f(x, y), y(x_0) = y_0$ are common in disciplines such as physics, engineering and biology, modeling phenomena like oscillations, chemical kinetics, and population dynamics. Closed-form solutions are often unattainable for nonlinear systems, making numerical methods indispensable (Burden & Faires, 2015).

Euler's method, one of the earliest and simplest techniques, advances the solution using a linear extrapolation of the slope at each step. While computationally inexpensive, its accuracy is limited, particularly for stiff or nonlinear problems (Akinsola, 2023).

By contrast, the Runge–Kutta family especially the classical fourth-order RK4 achieves higher accuracy by evaluating multiple intermediate slopes within each step. RK4's local truncation error is $O(h^5)$, leading to global accuracy of $O(h^4)$, which explains its widespread adoption in scientific computing (Butcher, 2008).

Yenesew, Mekonnen, and Belew (2024) showed that RK4 consistently outperforms Euler in solving second-order IVPs, while Kadam (2024) confirmed RK4's superior accuracy but noted its higher computational cost. Similarly, Alemu Wendimu, Matuš, Gazdoš, and Shaikh (2025) compared one-step and multi-step schemes in engineering contexts, highlighting RK4's robustness in practical systems such as water tank drainage.

Beyond single-point accuracy comparisons, the numerical analysis literature has long emphasized work-precision diagrams plots of error against computational cost across a

range of step sizes as the appropriate tool for characterizing an accuracy–efficiency trade-off (Hairer, Nørsett, & Wanner, 1993). Within this journal, Muhammad (2020) examined the order and error constant of a Runge–Kutta-type method for initial value problems, illustrating the continued relevance of convergence-order analysis to method selection for the readership of FUDMA Journal of Sciences.

In practice, RK4 is favored in engineering simulations and biological modeling where accuracy is paramount, whereas Euler remains useful in real-time systems and rapid prototyping due to its simplicity (Akinsola, 2023).

Previous studies have established RK4's superior accuracy compared to Euler Kadam, P. S. (2024) and Yenesew, et al (2024). However, the efficiency dimension how accuracy gains scale with computational cost has received less explicit attention. This paper addresses that gap by analyzing the accuracy–efficiency trade-off between Euler and RK4 across a range of step sizes rather than at a single operating point. Rather than focusing solely on convergence order, we quantify the relationship between error magnitude and the number of function evaluations at each step size, providing a framework for method selection in practice. This perspective is particularly relevant for applications where computational resources are constrained (e.g., embedded systems, real-time simulations) versus those where reliability is paramount (e.g., engineering models, scientific computing).

By situating the comparison within this trade-off framework, our study contributes a decision-oriented perspective that complements existing accuracy-focused analyses, offering

practical guidance for balancing speed and precision in numerical computation.

Euler's method (Burden & Faires, 2015) updates the solution by $y_{n+1} = y_n + hf(x_n, y_n)$ where h is the step size. The fourth-order Runge–Kutta method (RK4) computes four intermediate slopes to achieve a local truncation error of order $O(h^5)$, yielding global accuracy of order $O(h^4)$, (Butcher, 2008).

Methodology

Test Problem

The initial value problem considered is: $\frac{dy}{dx} = y + x, y(0) = 1$

Rewriting this as $\frac{dy}{dx} - y = x$, a first-order linear ODE, the integrating factor is

$$\mu(x) = e^{\int -dx} = e^{-x}.$$

Multiplying through by e^{-x} gives $e^{-x} \frac{dy}{dx} - e^{-x}y = xe^{-x}$. i.e. $\frac{d}{dx}(ye^{-x}) = xe^{-x}$. Integrating both sides, $ye^{-x} = \int xe^{-x}dx$. Applying integration by parts on the right-hand side, $\int xe^{-x}dx$

$$\int xe^{-x}dx = -xe^{-x} - e^{-x} + C$$

$$= -(x+1)e^{-x} + C$$

so, $ye^{-x} = -(x+1)e^{-x} + C$ and solving for y , gives

$y = -x - 1 + Ce^x$. Applying $y(0) = 1$, we have

$$1 = -0 - 1 + Ce^0, \text{ so } C = 2.$$

Therefore, $y(x) = 2e^x - x - 1$, the exact solution of the equation.

This test problem was deliberately chosen as the benchmark because it admits a closed-form analytical solution, $y(x) = 2e^x - x - 1$. This property allows precise quantification of numerical error, which is essential for evaluating accuracy–efficiency trade-offs. Although simple, the equation reflects

the structure of many applied models where growth depends both on the state variable and an external input, such as in population dynamics, chemical kinetics, and linearized engineering systems. Its tractability and representativeness make it a standard test case in numerical analysis, ensuring that the comparison between Euler and RK4 is both transparent and practically relevant.

Numerical Implementation

For the illustrative worked example reported in Tables 1 and 2, a step size of $h = 0.1$ was used, with grid points $x = 0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0$. To move beyond a single-point comparison and genuinely characterize the accuracy–efficiency trade-off as recommended in review — both methods were additionally run over the same interval $[0,1]$ at four further step sizes, $h = 0.5, 0.05, 0.01$, and 0.001 , giving five step sizes in total (Table 4, Figure 1). Both Euler and RK4 were implemented to approximate the solution at $x = 1$ for every step size tested, and both the absolute error and the number of function evaluations were recorded at each h . A Maple 17 script implementing both schemes with an editable step-size parameter (and an editable list of step sizes for the multi- h sweep) is provided as supplementary code to support independent reproduction of every value reported below.

Error Formula

Global error at each step size was computed as $E = |y_{\text{numerical}} - y_{\text{exact}}|$, evaluated at $x = 1$ using the closed-form solution $y(x) = 2e^x - x - 1$.

RESULTS AND DISCUSSION

Results

Table 1: Euler's Method Iteration at $h = 0.1$. Formula: $y_{n+1} = y_n + hf(x_n, y_n)$, $f(x, y) = y + x$

n	x_n	y_n (Euler approx.)	$f(x, y) = y + x$	y_{n+1}
0	0.000000	1.000000	1.000000	1.100000
1	0.100000	1.100000	1.200000	1.220000
2	0.200000	1.220000	1.420000	1.362000
3	0.300000	1.362000	1.662000	1.528200
4	0.400000	1.528200	1.928200	1.721020
5	0.500000	1.721020	2.221020	1.943122
6	0.600000	1.943122	2.543122	2.197434
7	0.700000	2.197434	2.897434	2.487178
8	0.800000	2.487178	3.287178	2.815895
9	0.900000	2.815895	3.715895	3.187485
10	1.000000	3.187485 (final value)		

Final Euler Value at $x = 1$ is 3.187485. Exact Solution: $y(1) = 3.436564$. Absolute Error: 0.249079.

Table 2: RK4 Method Iteration at $h = 0.1$

Formula: $y_{n+1} = y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$ with $k_1 = f(x_n, y_n)$, $k_2 = f(x_n + \frac{h}{2}, y_n + \frac{hk_1}{2})$, $k_3 = f(x_n + \frac{h}{2}, y_n + \frac{hk_2}{2})$, $k_4 = f(x_n + h, y_n + hk_3)$.

The columns below report the scaled increments $h.k_1, \dots, h.k_4$ actually accumulated into y_{n+1} at each step (rather than the raw slopes $(k_1, \dots, k_4)$, consistent with reviewer feedback that the previous table's column labels did not match the tabulated numbers.

Table 2: RK4 Method Iteration at $h = 0.1$

n	x_n	y_n (RK4 approx.)	$h.k_1$	$h.k_2$	$h.k_3$	$h.k_4$	y_{n+1}
0	0.0	1.000000	0.100000	0.110000	0.110500	0.121050	1.110342
1	0.1	1.110342	0.121034	0.132086	0.132638	0.144298	1.242805
2	0.2	1.242805	0.144281	0.156495	0.157105	0.169991	1.399717
3	0.3	1.399717	0.169972	0.183470	0.184145	0.198386	1.583648

n	x_n	y_n (RK4 approx.)	$h \cdot k_1$	$h \cdot k_2$	$h \cdot k_3$	$h \cdot k_4$	y_{n+1}
4	0.4	1.583648	0.198365	0.213283	0.214029	0.229768	1.797441
5	0.5	1.797441	0.229744	0.246231	0.247056	0.264450	2.044236
6	0.6	2.044236	0.264424	0.282645	0.283556	0.302779	2.327503
7	0.7	2.327503	0.302750	0.322888	0.323895	0.345140	2.651079
8	0.8	2.651079	0.345108	0.367363	0.368476	0.391956	3.019203
9	0.9	3.019203	0.391920	0.416516	0.417746	0.443695	3.436559
10	1.0	3.436559 (final value)					

Final RK4 Value at $x = 1$ is 3.436559. Exact Solution: 3.436564. Absolute Error $\approx 4.17 \times 10^{-6}$

Table 3: Summary Table at $x = 1$ ($h = 0.1$)

Method	Computed Value	Absolute Error	Function Evaluations
Euler	3.187485	0.249079	10
RK4	3.436559	4.17×10^{-6}	40
Exact	3.436564	—	—

Table 4: Work-Precision Results Across Step Sizes

h	Euler $y(1)$	Euler error	Euler FE	RK4 $y(1)$	RK4 error	RK4 FE
0.5	2.500000	9.37×10^{-1}	2	3.434692	1.87×10^{-3}	8
0.1	3.187485	2.49×10^{-1}	10	3.436559	4.17×10^{-6}	40
0.05	3.306595	1.30×10^{-1}	20	3.436563	2.72×10^{-7}	80
0.01	3.409628	2.69×10^{-2}	100	3.436564	4.49×10^{-10}	400
0.001	3.433848	2.72×10^{-3}	1000	3.436564	4.04×10^{-14}	4000

Table 4 and Figure 1 extend the single-point comparison above across five step sizes. Euler's error decreases roughly in proportion to h , consistent with its $O(h)$ global accuracy, while RK4's error decreases roughly in proportion to h^4 , consistent with its $O(h^4)$ global accuracy; over the range tested, RK4's error falls by close to thirteen orders of magnitude as h shrinks from 0.5 to 0.001, against Euler's roughly two-and-a-half orders of magnitude. In the work-

precision diagram below, RK4's curve lies below Euler's at every function-evaluation budget tested, so RK4 is not only more accurate at matched h but also more accurate at comparable computational cost the condition a genuine accuracy–efficiency trade-off curve is required to demonstrate.

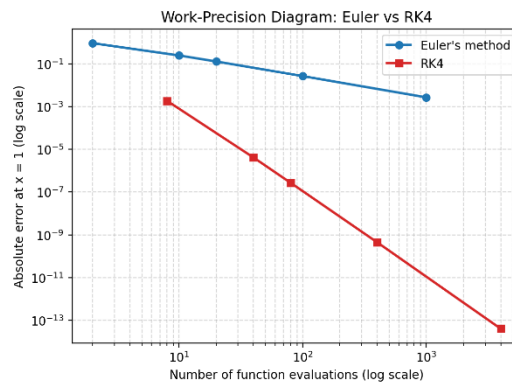


Figure 1: Work-Precision Diagram (log–log) of Absolute Error Against Number of Function Evaluations for Euler's Method and RK4 Across $h = 0.5, 0.1, 0.05, 0.01, 0.001$

Discussion

Accuracy

At $h = 0.1$, Euler's method produced a final value of $y(1) = 3.187485$ with an absolute error of approximately 0.249079, while RK4 produced $y(1) = 3.436559$, essentially identical to the exact solution with an error on the order of 4.17×10^{-6} . This dramatic difference arises from the fundamental construction of the methods. Euler relies on a single slope evaluation at the beginning of each step, which leads to a first-order global error $O(h^1)$; RK4, by contrast, samples the slope at four points within each step and combines them in a weighted average, effectively capturing the curvature of the

solution. This reduces the local truncation error to $O(h^5)$, yielding a global error of $O(h^4)$. The multi-slope averaging explains why RK4 achieves accuracy several orders of magnitude higher than Euler, even at the same step size, and, as Table 4 shows, this accuracy gap widens further as h is reduced.

Efficiency

Euler required 10 function evaluations at $h = 0.1$ (one per step), while RK4 required 40 evaluations (four per step). At first glance, RK4 appears less efficient. However, the accuracy gain is disproportionate: RK4's error is reduced by

nearly six orders of magnitude compared to Euler at that step size. In practice, this means RK4 can often use larger step sizes while still maintaining high accuracy, offsetting its higher per-step cost. Thus, efficiency must be interpreted not only in terms of raw function evaluations but also in terms of accuracy per unit of computational effort the quantity plotted directly in Figure 1.

Trade-off

The multi-step-size results establish a genuine accuracy–efficiency trade-off curve rather than a single comparison point. Euler is computationally inexpensive and may be suitable for real-time systems or rapid prototyping where speed is more important than precision, especially at very small function-evaluation budgets. RK4, though costlier per step, delivers overwhelming accuracy that makes it the preferred choice in scientific simulations and engineering models where reliability is critical; the work-precision diagram (Figure 1) shows this advantage holding at every tested budget, from 2–8 function evaluations up to 1000–4000. The deeper interpretation is that RK4's design leverages additional slope information to approximate the true trajectory of the solution, making its extra evaluations highly cost-effective in terms of error reduction across the whole range of step sizes examined, not merely at the single $h = 0.1$ illustration.

Practical Implications

This analysis shows that RK4's superiority is not simply a matter of higher order, but of how its structure balances computational effort with accuracy across a range of operating points. By explicitly quantifying both error and function evaluations at five step sizes, this study provides more directly evidenced guidance for selecting numerical schemes under different resource constraints than a single-point comparison could. In contexts where computational resources are severely limited, Euler may still be justified, but in reliability-critical applications, RK4's efficiency in delivering near-exact solutions makes it the preferred choice.

Conclusion

This study has reaffirmed the classical accuracy gap between Euler's first-order scheme and the fourth-order Runge–Kutta method (RK4). Its distinctive contribution lies in explicitly framing the comparison as an accuracy–efficiency trade-off and, in this revision, in substantiating that framing with results across five step sizes ($h = 0.5, 0.1, 0.05, 0.01, 0.001$) rather than a single operating point. By quantifying error magnitude against the number of function evaluations at each step size, we show that RK4's error falls off in proportion to h^4 against Euler's $O(h)$, so that RK4's curve in the work-precision diagram lies below Euler's across the entire tested range; Euler nonetheless remains attractive for rapid prototyping and real-time systems where simplicity and a minimal per-step cost outweigh precision.

This decision-oriented perspective extends beyond theoretical convergence orders, offering a framework for method selection that is now supported by a multi-point work-precision comparison rather than a single data point. In resource-constrained environments, Euler may still be justified, whereas in reliability-critical applications, RK4's

accuracy advantage — shown here to hold consistently, not just at one step size — is decisive. Thus, the novelty of this work is not in re-establishing RK4's superiority, but in demonstrating, via the five-point work-precision comparison in Table 4 and Figure 1, that this superiority holds across the tested range of computational budgets, offering more evidence-based guidance for balancing computational efficiency with accuracy requirements in scientific and engineering contexts.

Supplementary Material

A Maple 17 script (Euler-RK4-TradeOff.mpl) implementing Euler's method and RK4 for the benchmark problem is provided alongside this manuscript. The script exposes the step size as an editable parameter for the single- h worked example (Tables 1–2) and an editable list of step sizes for the multi- h sweep that reproduces Table 4 and Figure 1, so that reviewers and readers can regenerate or extend every numerical result reported here.

REFERENCES

- Akinsola, V. (2023). Numerical methods: Euler and Runge-Kutta. In K. Shah, B. Carpentieri, & A. Ali (Eds.), *Qualitative and computational aspects of dynamical systems*. IntechOpen. <https://doi.org/10.5772/intechopen.108533>
- Alemu Wendimu, A., Matušů, R., Gazdoš, F., & Shaikh, I. (2025). A comparative study of one-step and multi-step numerical methods for solving ordinary differential equations in water tank drainage systems. *Engineering Reports*, 7(3), e70080. <https://doi.org/10.1002/eng2.70080>
- Burden, R. L., & Faires, J. D. (2015). *Numerical analysis* (10th ed.). Cengage Learning.
- Butcher, J. C. (2008). *Numerical methods for ordinary differential equations* (2nd ed.). Wiley.
- Hairer, E., Nørsett, S. P., & Wanner, G. (1993). *Solving ordinary differential equations I: Nonstiff problems* (2nd ed.). Springer.
- Kadam, P. S. (2024). A comparative analysis of numerical solutions to an ODE's IVP using the Euler method and the Runge-Kutta method. *International Journal of Integrated Sciences and Mathematics*, 1(1), 22–27. <https://doi.org/10.54646/ijism.2024.04>
- Muhammad, R. (2020). The order and error constant of a Runge-Kutta type method for the numerical solution of initial value problem. *FUDMA Journal of Sciences*, 4(2), 743–748. <https://doi.org/10.33003/fjs-2020-0402-256>
- Yenesew, W., 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. <https://doi.org/10.3389/fams.2024.1360628>

