

Error Analysis of ODE Solvers: Euler vs. Runge-Kutta

Anusha Tripathi

Computer Science Department, Year 3
St. Xavier's College (Autonomous), Kolkata

March 20, 2025

1 Introduction

A differential equation is a mathematical equation that relates a function to its derivatives, describing how a quantity changes over time or space. These equations play a fundamental role in modeling natural and engineered systems, including physics, biology, economics, and engineering. Essentially, they help us understand how systems evolve over time based on given conditions.

However, most differential equations cannot be solved exactly. While there are analytical methods available for some simple cases, many real-world problems require numerical methods to approximate solutions. This is where numerical solvers such as Euler's method and the Runge-Kutta (RK4) method come into play.

ODEs describe how a function changes with respect to an independent variable, usually time. They appear in various scientific and engineering fields, modeling phenomena like population growth, heat conduction, and motion dynamics.

In this report, we compare two widely used numerical methods: the Euler method and the fourth-order Runge-Kutta (RK4) method. We apply these methods to five different test cases: exponential decay, logistic growth, simple harmonic motion, the Lorenz system, and the Van der Pol oscillator. The goal is to evaluate their accuracy, efficiency, and convergence by comparing numerical solutions with exact analytical results where available.

2 Mathematical Formulation

A first-order ODE takes the general form:

$$\frac{dy}{dt} = f(y, t), \quad y(0) = y_0. \quad (1)$$

This equation states that the rate of change of y with respect to time t depends on the function $f(y, t)$, with an initial value y_0 provided.

To approximate the solution, we consider the following numerical methods:

2.1 Euler Method

The Euler method is the simplest numerical approach to solving ODEs. It estimates the next value using a simple step-by-step approach:

$$y_{n+1} = y_n + hf(y_n, t_n). \quad (2)$$

This method essentially takes small steps h along the curve, using the current derivative to predict the next point. While easy to implement, Euler's method accumulates significant errors due to its simplistic nature. The errors arise because it assumes a straight-line approximation of the curve at each step, which is not always accurate for rapidly changing functions.

2.2 Runge-Kutta 4th Order (RK4)

The RK4 method improves accuracy by taking weighted averages of intermediate steps. This allows it to estimate the curve more accurately within a single step. It works as follows:

$$k_1 = hf(y_n, t_n) \quad (\text{Initial estimate based on the current slope}) \quad (3)$$

$$k_2 = hf(y_n + k_1/2, t_n + h/2) \quad (\text{Refining estimate at midpoint}) \quad (4)$$

$$k_3 = hf(y_n + k_2/2, t_n + h/2) \quad (\text{Further refining at midpoint}) \quad (5)$$

$$k_4 = hf(y_n + k_3, t_n + h) \quad (\text{Final estimate at next step}) \quad (6)$$

$$y_{n+1} = y_n + \frac{k_1 + 2k_2 + 2k_3 + k_4}{6}. \quad (7)$$

The RK4 method significantly reduces numerical errors compared to Euler's method, making it more accurate for a given step size.

3 Test Cases

To compare the two methods, we analyze five different ODEs, each representing a different type of system.

3.1 Exponential Decay

This equation models the decay of a substance over time:

$$\frac{dy}{dt} = -y, \quad y(0) = 1. \quad (8)$$

The exact solution is:

$$y(t) = e^{-t}. \quad (9)$$

This case helps us understand how well the numerical methods approximate smooth, exponential behavior.

3.2 Logistic Growth

The logistic equation describes population growth under limited resources:

$$\frac{dy}{dt} = y(1 - y/10), \quad y(0) = 1. \quad (10)$$

Its exact solution is:

$$y(t) = \frac{10}{1 + 9e^{-t}}. \quad (11)$$

This test case introduces nonlinearity, making the numerical comparison more interesting.

3.3 Simple Harmonic Motion (SHM)

This describes oscillatory motion, such as a mass on a spring:

$$\frac{dx}{dt} = v, \quad \frac{dv}{dt} = -\omega^2 x. \quad (12)$$

The exact solution is:

$$x(t) = A \cos(\omega t) + B \sin(\omega t). \quad (13)$$

This test demonstrates how well numerical methods handle periodic motion.

3.4 Lorenz System (Chaotic Behavior)

This set of equations models chaotic fluid flow:

$$\frac{dx}{dt} = \sigma(y - x), \quad (14)$$

$$\frac{dy}{dt} = x(\rho - z) - y, \quad (15)$$

$$\frac{dz}{dt} = xy - \beta z. \quad (16)$$

Small numerical errors grow rapidly in chaotic systems, making this a challenging test case.

3.5 Van der Pol Oscillator

This equation models nonlinear oscillations:

$$\frac{d^2x}{dt^2} - \mu(1 - x^2)\frac{dx}{dt} + x = 0. \quad (17)$$

Rewriting it as a system:

$$\frac{dx}{dt} = v, \quad (18)$$

$$\frac{dv}{dt} = \mu(1 - x^2)v - x. \quad (19)$$

This test helps evaluate how numerical solvers handle nonlinear damping effects.

4 Error Analysis

To measure numerical errors, we compute the L2 norm of the error:

$$E = \frac{\|y_{\text{exact}} - y_{\text{numerical}}\|_2}{N}. \quad (20)$$

The observed errors for each method are summarized in Table 1.

Method	L2 Error	Max Error	RMSE
Euler (Exponential Decay)	0.000570	0.009489	0.005696
RK4 (Exponential Decay)	0.000000	0.000000	0.000000
Euler (Logistic Growth)	0.003789	0.064599	0.037889
RK4 (Logistic Growth)	0.000000	0.000000	0.000000
Euler (SHM)	0.008573	0.269985	0.121248
RK4 (SHM)	0.000000	0.000000	0.000000
Euler (Lorenz)	0.087143	6.420071	2.755716
Euler (Van der Pol)	0.020209	0.812551	0.285801

Table 1: Numerical Errors for Different Methods

5 Results and Visualizations

We provide graphical comparisons for each test case to illustrate the accuracy of the numerical methods.

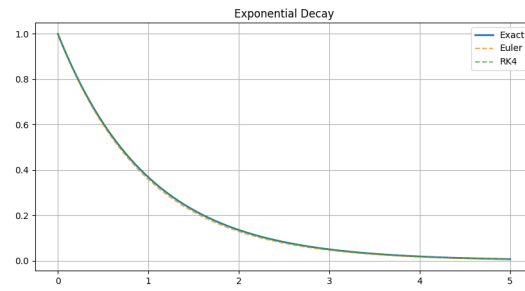


Figure 1: Numerical vs Exact Solution for Exponential Decay.

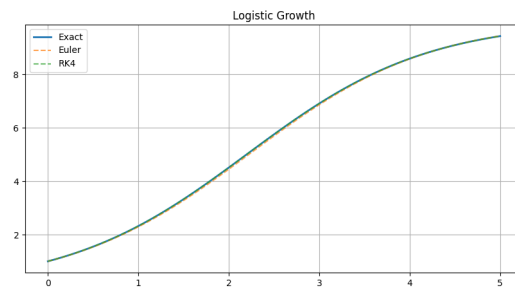


Figure 2: Numerical vs Exact Solution for Logistic Growth.

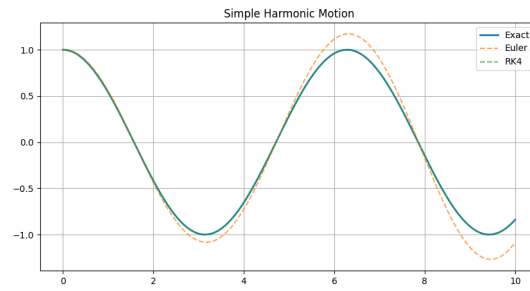


Figure 3: Numerical vs Exact Solution for Simple Harmonic Motion.

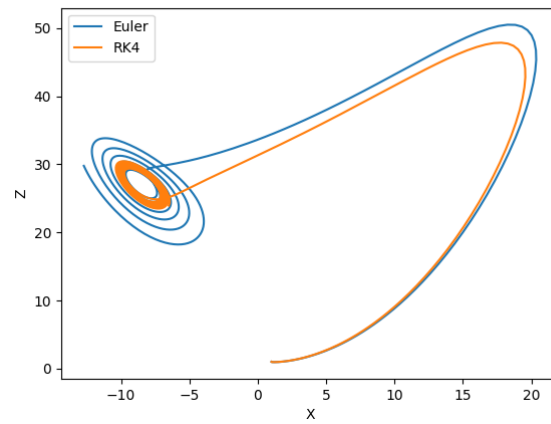


Figure 4: Numerical vs Exact Solution for the Lorenz System.

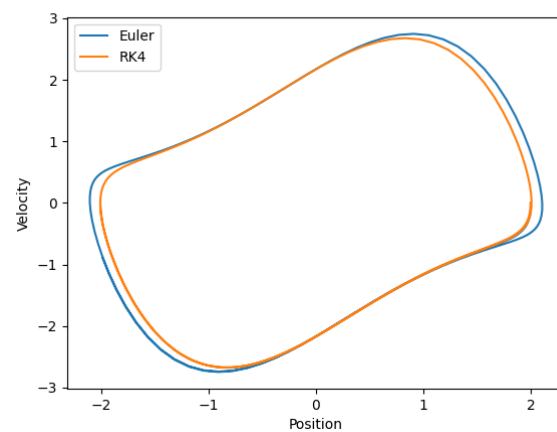


Figure 5: Comparison of Numerical Solutions for the Van der Pol Oscillator.

6 Conclusion

Our analysis shows that RK4 significantly outperforms the Euler method in terms of accuracy. The Euler method introduces large errors, making it less suitable for precise computations. The Van der Pol oscillator further highlights Euler's instability in nonlinear cases. RK4 achieves much smaller errors and better convergence properties, making it the preferred choice for solving ODEs numerically.

Next steps include comparing more complex differential equations with their exact solutions and exploring the use of PETSc modules to analyze error differences. Additionally, future work could explore adaptive step-size techniques to optimize the trade-off between accuracy and computational efficiency. Another promising direction is incorporating Verlet integration, particularly for second-order systems like harmonic motion and chaotic dynamics, where it provides better energy conservation and stability over long time steps.