

Numerical Methods for Solving Differential Equations: Application to a Series RL Circuit

Abstract

Differential equations model many phenomena in science and engineering; however, analytical solutions are not always available. In this report, we review several numerical methods for solving ordinary differential equations (ODEs) and apply them to the problem of a series RL circuit. The circuit is modeled by the ODE

$$L \frac{di}{dt} + Ri = V_{\text{in}}, \quad (1)$$

which can be rearranged as

$$\frac{di}{dt} = -\frac{R}{L} i + \frac{V_{\text{in}}}{L}. \quad (2)$$

We present the Forward Euler, Backward Euler, Runge–Kutta 2nd order (RK2, using Heun’s method), Trapezoidal, and Runge–Kutta 4th order (RK4) methods. For each method, we derive its amplification factor step by step using the test equation $y' = \lambda y$, briefly derive the order of the local truncation error, and then specialize the formulas to the RL circuit ODE.

1 Introduction

Series RL circuits are fundamental in electrical engineering. The current $i(t)$ in such a circuit is governed by

$$L \frac{di}{dt} + Ri = V_{\text{in}}, \quad (3)$$

where L is the inductance, R is the resistance, and V_{in} is the input voltage. Rearranging gives

$$\frac{di}{dt} = -\frac{R}{L}i + \frac{V_{\text{in}}}{L}. \quad (4)$$

This linear, first-order ODE will be solved using various numerical methods described in the following sections.

2 Error Analysis and Stability Concepts

Before reviewing the numerical methods, it is essential to understand some fundamental error and stability concepts.

Local Truncation Error

The **local truncation error** (LTE) measures the error made in a single step of a numerical method, assuming the previous value is exact. If $y(t_{n+1})$ is the exact solution at t_{n+1} and $\Psi(t_n, y(t_n), h)$ is the numerical update, then

$$\tau_n = \frac{y(t_{n+1}) - \Psi(t_n, y(t_n), h)}{h}. \quad (5)$$

A method is said to be of order p if $\tau_n = O(h^p)$ as $h \rightarrow 0$.

Global Error

The **global error** is the cumulative error after many steps, i.e., the difference between the numerical solution y_n and the exact solution $y(t_n)$. For a method of order p , the global error typically behaves as

$$|y(t_n) - y_n| = O(h^p). \quad (6)$$

Stability Concepts

Stability relates to how errors are propagated:

- **Conditional Stability:** A method is conditionally stable if it is stable only when the step size h is sufficiently small (e.g., Forward Euler requires $|1 + h\lambda| < 1$).

- **A-Stability:** A method is A-stable if it is stable for all step sizes h whenever the continuous problem is stable (i.e. when $\Re(\lambda) < 0$). Implicit methods like Backward Euler and the Trapezoidal method are A-stable.
- **Unbounded Growth:** If the numerical method does not satisfy the stability condition (i.e. if $|R(z)| > 1$), then errors may grow unboundedly.

3 Overview of Methods

In general, we consider an initial value problem (IVP) of the form

$$\frac{dy}{dt} = f(t, y), \quad y(t_0) = y_0. \quad (7)$$

For the RL circuit, the function is defined as

$$f(t, i) = -\frac{R}{L} i + \frac{V_{\text{in}}}{L}. \quad (8)$$

3.1 Forward Euler Method

General Algorithm

The Forward Euler method approximates the solution by

$$y_{n+1} = y_n + h f(t_n, y_n). \quad (9)$$

Geometric View

This method draws a tangent at the current point (t_n, y_n) and uses the slope $f(t_n, y_n)$ to project a step forward.

Derivation of Amplification Factor

For the test equation

$$y' = \lambda y, \quad (10)$$

with $f(t, y) = \lambda y$, we have:

$$y_{n+1} = y_n + h\lambda y_n \quad (11)$$

$$= (1 + h\lambda)y_n. \quad (12)$$

Thus, the amplification factor is

$$R(z) = 1 + z, \quad \text{where } z = h\lambda. \quad (13)$$

Local Truncation Error

Using the Taylor series expansion,

$$y(t_{n+1}) = y(t_n) + h y'(t_n) + \frac{h^2}{2} y''(\xi_n), \quad \xi_n \in [t_n, t_{n+1}], \quad (14)$$

and noting $y'(t_n) = \lambda y(t_n)$, we find:

$$y(t_{n+1}) - [y(t_n) + h\lambda y(t_n)] = \frac{h^2}{2} y''(\xi_n). \quad (15)$$

Thus, the LTE is

$$\tau_n = \frac{h^2}{2} y''(\xi_n)/h = \frac{h}{2} y''(\xi_n) = O(h). \quad (16)$$

Application to the RL Circuit

For the RL circuit, the homogeneous part is

$$\frac{di}{dt} = -\frac{R}{L} i, \quad (17)$$

so $\lambda = -\frac{R}{L}$. Therefore, the Forward Euler update becomes

$$i_{n+1} = i_n + h \left[-\frac{R}{L} i_n + \frac{V_{\text{in}}}{L} \right]. \quad (18)$$

3.2 Backward Euler Method

General Algorithm

The Backward Euler method is given by

$$y_{n+1} = y_n + h f(t_{n+1}, y_{n+1}). \quad (19)$$

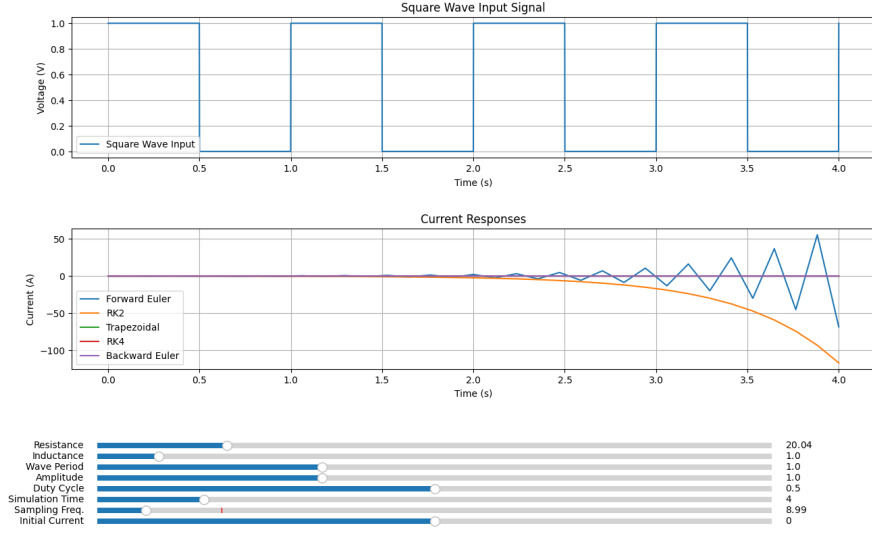


Figure 1: Instability of the Forward Euler method for a step size that violates the stability condition.

Geometric View

Here the slope is taken at the new time level (t_{n+1}, y_{n+1}) , which improves stability by "looking ahead."

Derivation of Amplification Factor

For the test equation $y' = \lambda y$, we have:

$$y_{n+1} = y_n + h\lambda y_{n+1}. \quad (20)$$

Rearranging,

$$(1 - h\lambda)y_{n+1} = y_n, \quad (21)$$

$$y_{n+1} = \frac{y_n}{1 - h\lambda}. \quad (22)$$

Thus, the amplification factor is

$$R(z) = \frac{1}{1 - z}, \quad z = h\lambda. \quad (23)$$

Local Truncation Error

A similar Taylor expansion argument shows that Backward Euler has an LTE of order $O(h)$.

Application to the RL Circuit

For $\lambda = -\frac{R}{L}$, the Backward Euler update becomes

$$i_{n+1} = i_n + h \left[-\frac{R}{L} i_{n+1} + \frac{V_{\text{in}}}{L} \right]. \quad (24)$$

3.3 Runge–Kutta 2nd Order (RK2) Method (Heun’s Method)

General Algorithm

Heun’s method is given by:

$$k_1 = f(t_n, y_n), \quad (25)$$

$$k_2 = f(t_n + h, y_n + h k_1), \quad (26)$$

$$y_{n+1} = y_n + \frac{h}{2}(k_1 + k_2). \quad (27)$$

Geometric View

This method first uses an Euler step to predict the solution at $t_n + h$, then uses the slope at that point to correct the estimate by averaging the two slopes.

Detailed Derivation and Amplification Factor

For $y' = \lambda y$, we have:

$$k_1 = \lambda y_n, \quad (28)$$

$$y^p = y_n + h k_1 = y_n(1 + h\lambda), \quad (29)$$

$$k_2 = \lambda y^p = \lambda y_n(1 + h\lambda). \quad (30)$$

Thus, the update is:

$$y_{n+1} = y_n + \frac{h}{2} [\lambda y_n + \lambda y_n(1 + h\lambda)] \quad (31)$$

$$= y_n \left[1 + h\lambda + \frac{(h\lambda)^2}{2} \right]. \quad (32)$$

So, the amplification factor is

$$R(z) = 1 + z + \frac{z^2}{2}, \quad z = h\lambda. \quad (33)$$

Local Truncation Error

Comparing with the exact solution $e^{h\lambda} = 1 + h\lambda + \frac{(h\lambda)^2}{2} + \frac{(h\lambda)^3}{6} + \dots$, the local error per step is

$$e^{h\lambda} - \left(1 + h\lambda + \frac{(h\lambda)^2}{2} \right) = \frac{(h\lambda)^3}{6} + O(h^4). \quad (34)$$

Thus, the LTE is $O(h^3)$ and when scaled by h , the method is second order accurate.

Application to the RL Circuit

For the RL circuit,

$$k_1 = -\frac{R}{L} i_n + \frac{V_{\text{in}}}{L}, \quad (35)$$

$$k_2 = -\frac{R}{L} (i_n + h k_1) + \frac{V_{\text{in}}}{L}, \quad (36)$$

$$i_{n+1} = i_n + \frac{h}{2} (k_1 + k_2). \quad (37)$$

3.4 Trapezoidal Method

General Algorithm

The Trapezoidal method is given by:

$$y_{n+1} = y_n + \frac{h}{2} [f(t_n, y_n) + f(t_{n+1}, y_{n+1})]. \quad (38)$$

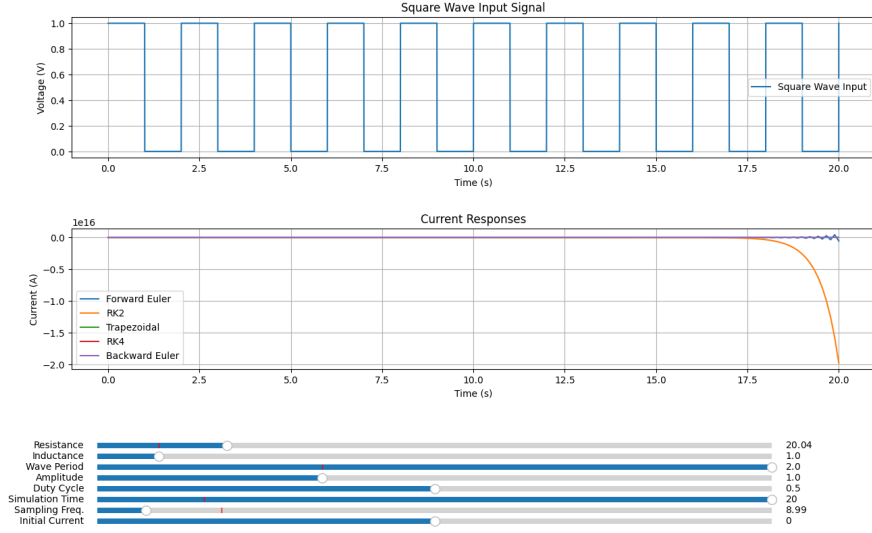


Figure 2: Instability behavior of RK2 (Heun's) method when the step size is too large.

Geometric View

This method approximates the area under the curve by forming a trapezoid over the interval, averaging the slopes at the beginning and end of the step.

Derivation of Amplification Factor

For $y' = \lambda y$, we substitute $f(t, y) = \lambda y$:

$$y_{n+1} = y_n + \frac{h\lambda}{2}(y_n + y_{n+1}). \quad (39)$$

Rearranging:

$$y_{n+1} \left(1 - \frac{h\lambda}{2}\right) = y_n \left(1 + \frac{h\lambda}{2}\right), \quad (40)$$

$$y_{n+1} = \frac{1 + \frac{h\lambda}{2}}{1 - \frac{h\lambda}{2}} y_n. \quad (41)$$

Thus, the amplification factor is

$$R(z) = \frac{1 + \frac{z}{2}}{1 - \frac{z}{2}}, \quad z = h\lambda. \quad (42)$$

Local Truncation Error

A detailed derivation shows that the Trapezoidal method has an LTE of $O(h^3)$, implying it is second order accurate when scaled by h .

Application to the RL Circuit

For the RL circuit, the update becomes:

$$i_{n+1} = i_n + \frac{h}{2} \left[\left(-\frac{R}{L} i_n + \frac{V_{\text{in}}}{L} \right) + \left(-\frac{R}{L} i_{n+1} + \frac{V_{\text{in}}}{L} \right) \right]. \quad (43)$$

This implicit equation is solved for i_{n+1} .

3.5 Runge–Kutta 4th Order (RK4) Method

General Algorithm

The RK4 method is defined by:

$$k_1 = f(t_n, y_n), \quad (44)$$

$$k_2 = f\left(t_n + \frac{h}{2}, y_n + \frac{h}{2} k_1\right), \quad (45)$$

$$k_3 = f\left(t_n + \frac{h}{2}, y_n + \frac{h}{2} k_2\right), \quad (46)$$

$$k_4 = f(t_n + h, y_n + h k_3), \quad (47)$$

$$y_{n+1} = y_n + \frac{h}{6} (k_1 + 2k_2 + 2k_3 + k_4). \quad (48)$$

Geometric View

RK4 evaluates the slope at four points (initial, two midpoints, and final) and computes a weighted average of these slopes to approximate the solution. This multi-stage process captures more information about the behavior of the solution over the interval.

Derivation of Amplification Factor

For $y' = \lambda y$, the RK4 method's amplification factor is given by the fourth-order Taylor polynomial of $e^{h\lambda}$:

$$R(z) = 1 + z + \frac{z^2}{2} + \frac{z^3}{6} + \frac{z^4}{24}, \quad z = h\lambda. \quad (49)$$

Local Truncation Error

The exact solution is

$$e^{h\lambda} = 1 + h\lambda + \frac{(h\lambda)^2}{2} + \frac{(h\lambda)^3}{6} + \frac{(h\lambda)^4}{24} + \frac{(h\lambda)^5}{120} + \dots, \quad (50)$$

and the error per step is

$$e^{h\lambda} - R(z) = \frac{(h\lambda)^5}{120} + O(h^6). \quad (51)$$

Thus, the LTE is $O(h^5)$, so when divided by h the method is fourth order accurate.

Application to the RL Circuit

For our circuit, with

$$f(i) = -\frac{R}{L}i + \frac{V_{\text{in}}}{L},$$

the stages are:

$$k_1 = -\frac{R}{L}i_n + \frac{V_{\text{in}}}{L}, \quad (52)$$

$$k_2 = -\frac{R}{L}\left(i_n + \frac{h}{2}k_1\right) + \frac{V_{\text{in}}}{L}, \quad (53)$$

$$k_3 = -\frac{R}{L}\left(i_n + \frac{h}{2}k_2\right) + \frac{V_{\text{in}}}{L}, \quad (54)$$

$$k_4 = -\frac{R}{L}\left(i_n + hk_3\right) + \frac{V_{\text{in}}}{L}, \quad (55)$$

$$i_{n+1} = i_n + \frac{h}{6}\left(k_1 + 2k_2 + 2k_3 + k_4\right). \quad (56)$$

The amplification factor for the homogeneous part remains

$$R(z) = 1 + z + \frac{z^2}{2} + \frac{z^3}{6} + \frac{z^4}{24}, \quad z = h\lambda. \quad (57)$$

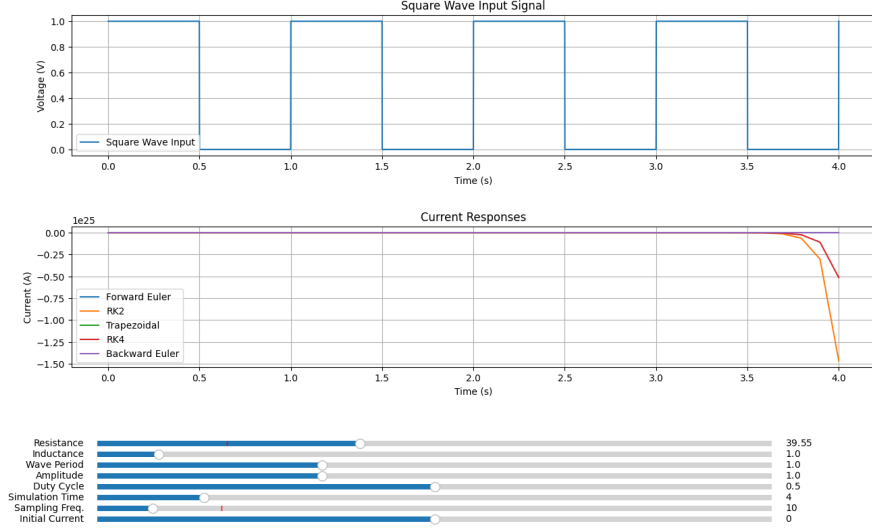


Figure 3: Instability behavior of RK4 when the step size exceeds the stability region (bounded region).

4 Error Comparison and Reference Method

In this section, we compare the performance of the various numerical methods by plotting their local and global errors. Figure 4 shows the local truncation error at each sample point for selected values of R , L , and the sampling rate. In Figure 5, the global error of the methods is plotted against the sampling factor, with the RK4 method used as the reference.

RK4 was chosen as the reference method because its higher order accuracy (with a global error of order $O(h^4)$) ensures that its error is significantly smaller than that of lower-order methods. This makes it a reliable benchmark for comparing the performance and error behavior of other numerical schemes.

5 Computational Times Comparison

Figure 6 plots the computational times (y-axis) versus the sampling factors (x-axis) for the different methods. This comparison helps illustrate the trade-off between accuracy, stability, and computational cost. As the sampling factor increases (i.e., as h decreases), the computational time typically

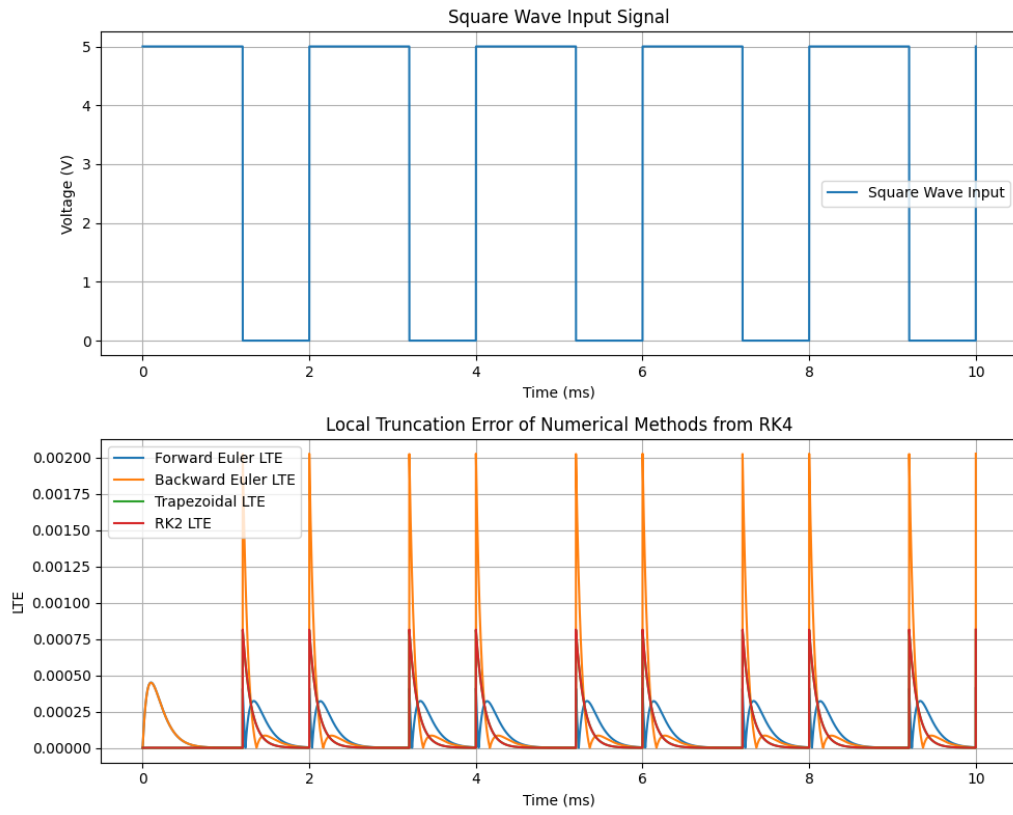


Figure 4: Local truncation error at each sample point for selected values of R , L , and the sampling rate.

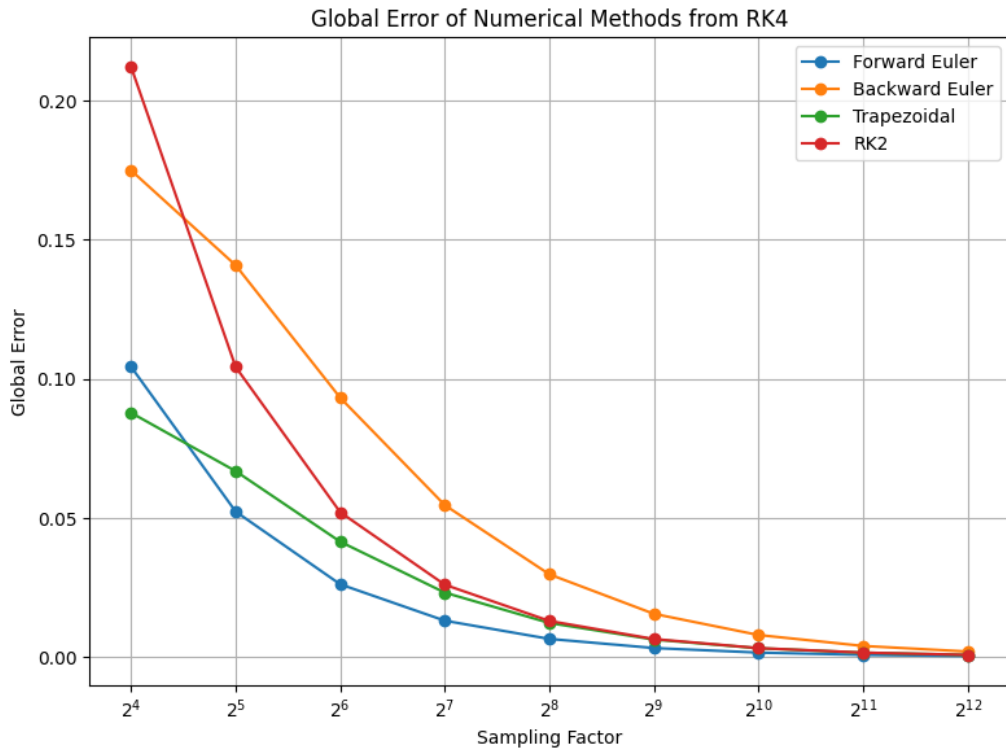


Figure 5: Global error versus the sampling factor. The RK4 method is used as the reference due to its higher order accuracy ($O(h^4)$).

increases.

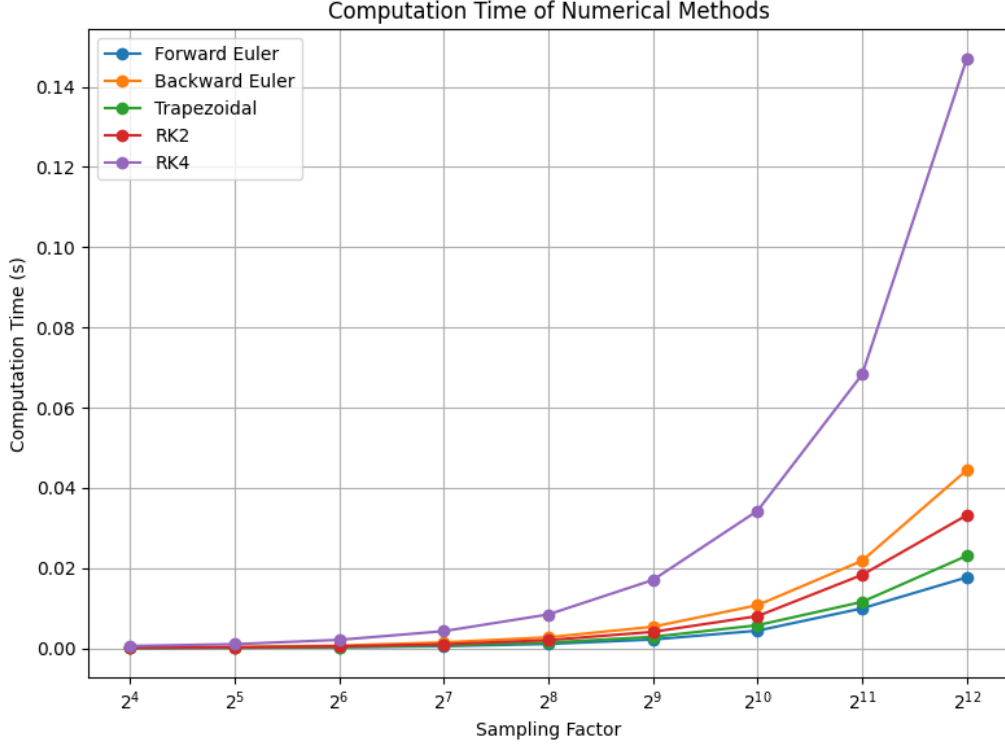


Figure 6: Computational times (y-axis) versus sampling factors (x-axis) for the different methods.

6 Stability Analysis Summary

For a one-step method applied to the test equation

$$y' = \lambda y, \quad y(0) = y_0, \quad (58)$$

the update can be written as

$$y_{n+1} = R(z)y_n, \quad \text{with } z = h\lambda. \quad (59)$$

A method is stable if

$$|R(z)| < 1. \quad (60)$$

For the RL circuit, the homogeneous part is

$$\frac{di}{dt} = -\frac{R}{L} i, \quad (61)$$

so $\lambda = -\frac{R}{L}$. Each method's amplification factor $R(z)$ then determines the choice of h to ensure stability:

- **Forward Euler:** $R(z) = 1 + z \Rightarrow$ requires $|1 - h\frac{R}{L}| < 1$.
- **Backward Euler:** $R(z) = \frac{1}{1-z}$ is A-stable.
- **RK2 (Heun's):** $R(z) = 1 + z + \frac{z^2}{2}$.
- **Trapezoidal:** $R(z) = \frac{1+\frac{z}{2}}{1-\frac{z}{2}}$ is A-stable.
- **RK4:** $R(z) = 1 + z + \frac{z^2}{2} + \frac{z^3}{6} + \frac{z^4}{24}$ has a bounded stability region.

7 Conclusion

In this report, we have reviewed five numerical methods for solving ODEs and derived their amplification factors and local truncation errors step by step using the test equation $y' = \lambda y$. For our project, we apply these methods to a series RL circuit described by

$$L \frac{di}{dt} + Ri = V_{\text{in}}, \quad (62)$$

or equivalently,

$$\frac{di}{dt} = -\frac{R}{L} i + \frac{V_{\text{in}}}{L}. \quad (63)$$

The methods are summarized below in the context of the RL circuit:

- **Forward Euler:**

$$i_{n+1} = i_n + h \left[-\frac{R}{L} i_n + \frac{V_{\text{in}}}{L} \right], \quad (64)$$

with LTE of $O(h)$.

- **Backward Euler:**

$$i_{n+1} = i_n + h \left[-\frac{R}{L} i_{n+1} + \frac{V_{\text{in}}}{L} \right], \quad (65)$$

with LTE of $O(h)$.

- **RK2 (Heun's Method):**

$$k_1 = -\frac{R}{L} i_n + \frac{V_{\text{in}}}{L}, \quad (66)$$

$$k_2 = -\frac{R}{L} \left(i_n + h k_1 \right) + \frac{V_{\text{in}}}{L}, \quad (67)$$

$$i_{n+1} = i_n + \frac{h}{2} (k_1 + k_2), \quad (68)$$

with LTE of $O(h^3)$ (i.e., second order accurate).

- **Trapezoidal:**

$$i_{n+1} = i_n + \frac{h}{2} \left[\left(-\frac{R}{L} i_n + \frac{V_{\text{in}}}{L} \right) + \left(-\frac{R}{L} i_{n+1} + \frac{V_{\text{in}}}{L} \right) \right], \quad (69)$$

with LTE of $O(h^3)$.

- **RK4:**

$$k_1 = -\frac{R}{L} i_n + \frac{V_{\text{in}}}{L}, \quad (70)$$

$$k_2 = -\frac{R}{L} \left(i_n + \frac{h}{2} k_1 \right) + \frac{V_{\text{in}}}{L}, \quad (71)$$

$$k_3 = -\frac{R}{L} \left(i_n + \frac{h}{2} k_2 \right) + \frac{V_{\text{in}}}{L}, \quad (72)$$

$$k_4 = -\frac{R}{L} \left(i_n + h k_3 \right) + \frac{V_{\text{in}}}{L}, \quad (73)$$

$$i_{n+1} = i_n + \frac{h}{6} (k_1 + 2k_2 + 2k_3 + k_4), \quad (74)$$

with LTE of $O(h^5)$ (i.e., fourth order accurate).

The choice among these methods depends on the circuit parameters, the required accuracy, and the stiffness of the problem. For non-stiff conditions, RK4 may be preferred for its high accuracy, while for stiff transients the implicit methods (Backward Euler or Trapezoidal) are more appropriate.