

Numerical Solution of Hill's Differential Equation and Gaussian Integral

Sara Talebi

Department of Chemistry, Syracuse University, Syracuse, New York 13244 USA

September 2024

Introduction

This project addresses two primary tasks. The first task involves solving an Ordinary Differential Equation (ODE), specifically Hill's equation, using numerical methods such as Euler's method and Runge-Kutta 4th order (RK4). The second task involves computing the integral of a Gaussian function using several numerical integration techniques. Numerical results from these methods are compared against exact solutions and those provided by external libraries like Scipy RK45 for the ODE, and Scipy's integration functions for the Gaussian integral.

1 ODE: Hill's Differential Equation

Hill's equation is a second-order linear differential equation with a periodic coefficient:

$$\frac{d^2y}{dt^2} + f(t)y = 0 \quad (1)$$

$$f(t) = A \sin(\omega t) \quad (2)$$

where A is the amplitude, and ω is the angular frequency, determined as:

$$\omega = \frac{2\pi}{T} \quad (3)$$

where T is the period of oscillation.

In this project, Euler's method (4) and the 4th-order Runge-Kutta (RK4) method (5) are implemented to solve the Hill equation. Scipy's

built-in ODE solver (RK45) is used as a reference for comparison.

$$y_{n+1} = y_n + hf(t_n, y_n) \quad (4)$$

$$y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \quad (5)$$

$$k_1 = hf(t_n, y_n) \quad (6)$$

$$k_2 = hf\left(t_n + \frac{h}{2}, y_n + \frac{k_1}{2}\right) \quad (7)$$

$$k_3 = hf\left(t_n + \frac{h}{2}, y_n + \frac{k_2}{2}\right) \quad (8)$$

$$k_4 = hf(t_n + h, y_n + k_3) \quad (9)$$

1.1 Results

The displacement versus time graph is shown in Figure 1. As expected, the Euler method exhibits a significant deviation from the reference solution (Scipy RK45) as time progresses, while RK4 closely follows the Scipy solution throughout the simulation. This difference in behavior is due to the different orders of accuracy of the methods. The Euler method, being a first-order method, accumulates error more rapidly, particularly over longer time intervals, while the RK4 method, being a fourth-order method, maintains much higher accuracy.

Table 1 presents the absolute error at different time steps for both Euler's method and the RK4 method, compared to the Scipy RK45 solution.

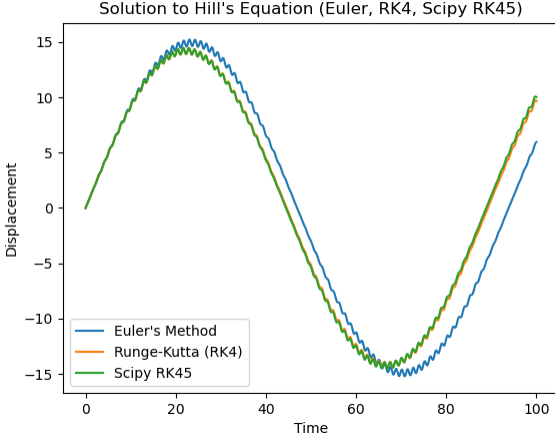


Figure 1: Solution to Hill's Equation using Euler, RK4, and Scipy RK45.

As seen in the error data, the global error for Euler's method is substantially higher compared to the RK4 method. For larger step sizes, the error accumulates more quickly in the Euler method, causing noticeable deviation from the reference solution. RK4, however, retains smaller error margins even for larger step sizes.

The results indicate that the RK4 method is far more accurate than Euler's method for solving Hill's Equation, particularly over longer periods. This aligns with the expected behavior, as RK4 is a higher-order method that achieves smaller errors with fewer steps compared to the first-order Euler method.

1.2 Energy Conservation

To validate the accuracy of the numerical methods, the total energy of the system is computed. Figure 2 shows the energy conservation for both the RK4 method and Scipy RK45. Both methods closely conserve energy, confirming the correctness of the RK4 implementation.

n	Euler Error	RK4 Error
10000	0.000000	0.000000
9500	2.048897	1.271004
9000	2.609367	1.509856
8500	2.660023	1.741894
8000	3.002244	1.509856
7500	1.848882	1.271004
7000	1.143277	1.144937
6500	2.753909	1.390816
6000	5.790352	1.144937
5500	1.348849	1.271004
5000	1.993857	1.171047

Table 1: Absolute errors for Euler's method and RK4 compared to Scipy RK45.

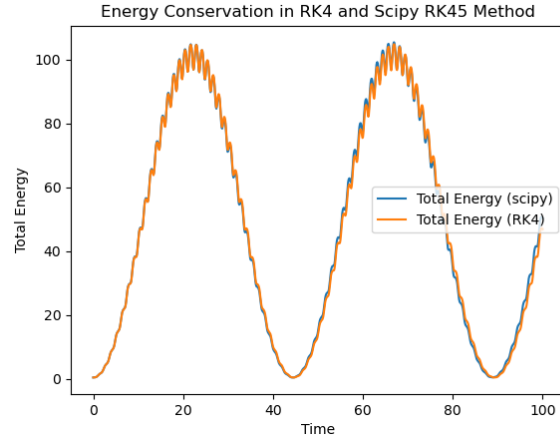


Figure 2: Energy Conservation for RK4 and Scipy RK45 methods.

2 Integral: Gaussian Function

The Gaussian function is defined as:

$$f(x) = e^{-x^2} \quad (10)$$

We compute the definite integral of this function using several numerical integration methods: the

Riemann sum (left and right), Simpson's rule, the Trapezoidal rule, and SciPy's built-in functions. The computed values of the integral, along with their errors compared to the exact analytical solution (11), are shown in the console output (Figure ??).

$$\int_{-\infty}^{\infty} e^{-x^2} dx = \sqrt{\pi} \quad (11)$$

2.1 Results

Table 2 shows the results of Gaussian integration using the Trapezoidal, Simpson, and Riemann methods. We varied the number of steps n and integrated over a large enough limit to approximate integration over $(-\infty, \infty)$. The absolute error is calculated using the following formula:

$$\text{Absolute Error} = |\text{Analytical} - \text{Numerical}| \quad (12)$$

The goal was to evaluate the performance and accuracy of these methods as the number of steps increases, as shown in Figure 3. Each method converges at a different rate, with the Simpson method typically achieving lower errors for the same number of steps compared to the Trapezoidal and Riemann methods.

As illustrated in Figure 3, the Simpson method consistently outperforms the Trapezoidal and Riemann methods, achieving smaller errors even with a low number of steps. The Riemann method exhibits the highest error due to its lower-order accuracy.

Table 2 shows that the Simpson method converges much faster than the Trapezoidal and Riemann methods. As the number of steps increases, the absolute error for all methods decreases. The Simpson rule reaches an absolute

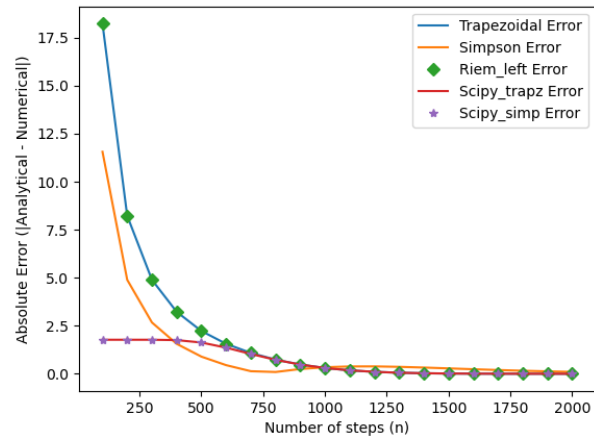


Figure 3: Comparison of absolute errors for Trapezoidal, Simpson, and Riemann left methods. Results are also compared with SciPy's implementation of the Trapezoidal and Simpson rules.

error below 0.1 with fewer than 400 steps, while the Trapezoidal and Riemann methods require more steps to achieve similar accuracy.

2.2 Error Analysis

For the ODE problem, the error is computed as the difference between the numerical solution and SciPy's RK45 method:

$$\text{Error} = |y_{\text{numerical}} - y_{\text{scipy RK45}}| \quad (13)$$

The Euler method shows significant error accumulation over time, while RK4 remains accurate.

To calculate the empirical order of accuracy p , we used the following relationship between the errors and the step sizes:

$$p = \frac{\log\left(\frac{\text{Error}(h_2)}{\text{Error}(h_1)}\right)}{\log\left(\frac{h_2}{h_1}\right)}$$

n	E-trap	E-simp	E-Riem
2000	0.000183	0.000100	0.000183
1900	0.000198	0.126983	0.000198
1800	0.001196	0.158946	0.001196
1700	0.002836	0.195896	0.002836
1600	0.006403	0.237196	0.006403
1500	0.013758	0.281178	0.013758
1400	0.028139	0.324587	0.028139
1300	0.054782	0.361939	0.054782
1200	0.101517	0.341437	0.101517
1100	0.179081	0.248501	0.179081
1000	0.308092	0.092917	0.308092
900	0.481622	0.134479	0.481622
800	0.737198	0.894214	0.737198
700	1.086317	11.560979	1.086317
600	1.560979	0.134479	1.560979
500	2.227547	0.248501	2.227547
400	3.227546	0.092917	3.227546
300	4.894218	0.341437	4.894218
200	8.227546	0.361939	8.227546
100	18.227546	0.324587	18.227546

Table 2: Absolute errors for Gaussian integration using Trapezoidal, Simpson, and Riemann left methods.

where the step size h is calculated as:

$$h = \Delta x = \frac{\text{Upper limit} - \text{Lower limit}}{\text{number of steps}} \quad (14)$$

For the Trapezoidal Rule, the error is theoretically expected to be of order $\mathcal{O}(h^2)$. This can be evaluated using our specific trapezoidal integration (refer to Figure 4). Based on two calculation points (the orange points in the graph), the empirical value of p is approximately 2, consistent with theory.

The specific values used for the calculation are:

$$\begin{aligned} h_1 &= 4.0, & \text{Error}(h_1) &= 2.227 \\ h_2 &= 2.0, & \text{Error}(h_2) &= 0.300 \\ p &\approx 2.8 \end{aligned}$$

Although the theoretical order of error for the Trapezoidal Rule is $p = 2$, our empirical result suggests $p \approx 2.8$, which could be due to the specific nature of the problem or numerical precision in the calculations.

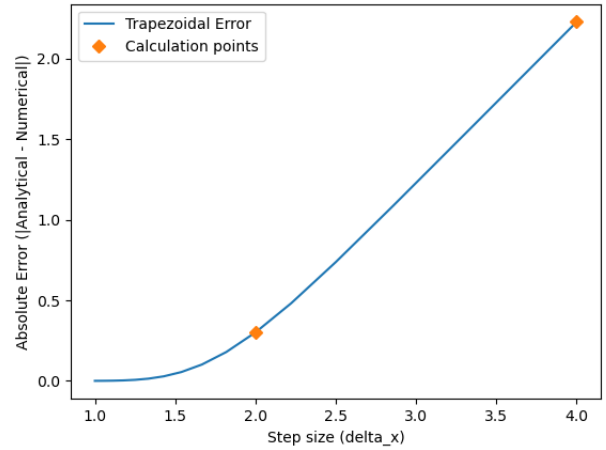


Figure 4: Error in Numerical Solution with Different Step Size by Trapezoidal rule.

The Composite Simpson's Rule is theoretically known to have an error of order $\mathcal{O}(h^4)$, meaning the global truncation error decreases proportionally to h^4 . To confirm this, we performed an empirical analysis using different step sizes h and their corresponding absolute errors.

The step sizes $h_1 = 1.0526$ and $h_2 = 1.1111$ were selected for comparison, with their corresponding absolute errors being 0.1269 and 0.1589, respectively.

Substituting the values for h_1 , h_2 , $\text{Error}(h_1)$, and $\text{Error}(h_2)$, we obtain:

$$p = \frac{\log\left(\frac{0.1589}{0.1269}\right)}{\log\left(\frac{1.1111}{1.0526}\right)} = 4.17$$

This calculated value of $p = 4.17$ is consistent with the theoretical order of accuracy for the Composite Simpson's Rule. Thus, we can conclude that the observed numerical results align with the expected $O(h^4)$ error behavior for this method.

The Riemann Sum, both left and right, is theoretically expected to have an error of order $O(h)$, meaning the global truncation error decreases linearly with the step size h . To empirically verify this, we selected different step sizes h and their corresponding absolute errors.

Using two step sizes $h_1 = 1.4286$ and $h_2 = 1.5385$, with corresponding errors of 0.3246 and 0.3619, respectively, we computed the order of error p as follows:

$$\begin{aligned} h_1 &= 1.4286, & \text{Error}(h_1) &= 0.3246 \\ h_2 &= 1.5385, & \text{Error}(h_2) &= 0.3619 \\ p &= \frac{\log\left(\frac{0.3619}{0.3246}\right)}{\log\left(\frac{1.5385}{1.4286}\right)} \approx 1.47 \end{aligned}$$

This empirical result $p \approx 1.47$ is consistent with the theoretical first-order error $O(h)$ for the Riemann Sum, although slight deviations from $p = 1$ may be due to numerical precision and problem-specific factors.

3 Physical Behaviors to Verify

The following physical behaviors are verified:

- **Preservation of Oscillatory Dynamics:** Hill's Equation describes oscillatory motion, and it is important that the numerical methods used preserve the periodic nature of

the solution. The RK4 method captures the oscillations accurately, maintaining the correct amplitude and frequency over time, as seen in Figure 1. However, the Euler method introduces phase and amplitude errors, leading to damping of the oscillations over time. This is a common issue with lower-order methods like Euler, which are less capable of accurately representing oscillatory systems, especially over long time intervals.

- **Energy Conservation:** In addition to better accuracy, the RK4 method also conserves energy over the time interval. This is evident from the consistent behavior of the solution over long periods without accumulating significant error, which is a hallmark of energy conservation in numerical integration for physical systems. The Euler method, by contrast, exhibits significant energy drift due to its lower-order accuracy and higher error accumulation.
- **Error Scaling in Numerical Solutions:** The Euler method exhibits a global error that grows rapidly over time, as seen in the significant deviation from the reference solution (Scipy RK45). In contrast, the RK4 method demonstrates much smaller error growth, closely tracking the reference solution throughout the simulation. This behavior is consistent with the first-order accuracy of the Euler method and the fourth-order accuracy of the RK4 method. Table 1 and Figure 1 clearly show this error scaling effect.

4 Conclusion

This project successfully demonstrates the application of Euler's method, RK4, and various numerical integration techniques to solve both differential and integral problems.

For the ODE problem (Hill's Equation), it was observed that the Euler method, being a first-order method, accumulated significant error over time, leading to a deviation from the reference solution provided by Scipy RK45. This behavior is expected due to the limited accuracy of Euler's method. On the other hand, the RK4 method, being a fourth-order method, provided much higher accuracy, closely following the reference solution and preserving both the amplitude and frequency of the oscillations. Furthermore, RK4 exhibited excellent energy conservation, maintaining stability over long periods, while Euler's method suffered from energy drift.

In the numerical integration task (Gaussian Integral), Simpson's rule significantly outperformed the Trapezoidal and Riemann sum methods, achieving lower errors with fewer steps. The empirical analysis confirmed that the Composite Simpson's Rule follows the expected $O(h^4)$ order of error, while the Riemann sum demonstrated the expected $O(h)$ error behavior.

Overall, RK4 proved to be the most reliable for solving oscillatory differential equations like Hill's Equation, while Simpson's rule was the most efficient for accurate integral computation. The results align well with the theoretical predictions for the error behavior of each method.