# Numerical Simulation of the SIR Epidemic Model

## A Comparative Analysis of Euler's Method and Runge-Kutta 4th Order


**Author:**
Group 3

**Institution:**
Kwame Nkrumah University of Science and Technology


**Course:**
Math 379

# Group Members and Contributions

The following are the members of Group 3 and their respective contributions to this project:

- **Samuel Quaigraine** (Group Leader): Led the group meetings, coordinated tasks, and ensured the project was completed on time.
- **Boateng Serwaa Stephanie**: Responsible for the mathematical formulation of the SIR model and contributed to report writing.
- **Mensah Maxwell Kobina Essa**: Handled the code implementation and visualization of results.
- **Ofosu Hackman**: Contributed to report writing and assisted in the mathematical formulation.
- **Banjeh Ernest Mwinlasunga**: Worked on visualization of results and contributed to report writing.

**Abstract**

This report presents a numerical simulation of the Susceptible-Infected-Recovered (SIR) epidemic model using Euler's method and the Runge-Kutta 4th Order (RK4) method. The study aims to compare the accuracy, stability, and convergence of these numerical approaches while analyzing the effects of key epidemiological parameters on disease spread. The findings are related to real-world outbreaks, such as COVID-19 and influenza, to draw meaningful insights into epidemic behavior.

# 1 Introduction

Epidemic modeling is crucial for understanding and predicting the spread of infectious diseases. The SIR model is a fundamental compartmental model used in epidemiology to simulate how diseases propagate through a population. This study numerically solves the SIR model using Euler's method and RK4, comparing their effectiveness and analyzing the impact of different infection and recovery rates.

# 2 Mathematical Formulation

The SIR model is governed by the following system of ordinary differential equations (ODEs):

$$
\begin{aligned}
\frac{dS}{dt} &= -\beta S I, \\
\frac{dI}{dt} &= \beta S I - \gamma I, \\
\frac{dR}{dt} &= \gamma I.
\end{aligned}
$$

where:

- $S(t)$ represents the number of susceptible individuals,
- $I(t)$ represents the number of infected individuals,
- $R(t)$ represents the number of recovered individuals,
- $\beta$ is the infection rate,
- $\gamma$ is the recovery rate.

Given initial conditions $S_0, I_0, R_0$ at $t = 0$, the goal is to compute $S$, $I$, and $R$ over time using numerical methods.

# 3 Numerical Methods

## 3.1 Euler's Method

Euler's method approximates the solution using the formula:

$$X_{n+1} = X_n + h\, f(X_n, t_n)$$

where $h$ is the step size, and $X = [S, I, R]$ represents the system state.

### 3.2 Runge-Kutta 4th Order (RK4) Method

The RK4 method improves accuracy with:

$$X_{n+1} = X_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

where:

$$\begin{aligned}
k_1 &= f(X_n, t_n), \\
k_2 &= f\left(X_n + \frac{h}{2}k_1, t_n + \frac{h}{2}\right), \\
k_3 &= f\left(X_n + \frac{h}{2}k_2, t_n + \frac{h}{2}\right), \\
k_4 &= f(X_n + h\,k_3, t_n + h).
\end{aligned}$$

# 4 Implementation

The SIR model was implemented in Python using Euler's and RK4 methods. The parameters used for simulation were:

- $\beta = 0.3$ (infection rate),
- $\gamma = 0.1$ (recovery rate),
- $S_0 = 990$, $I_0 = 10$, $R_0 = 0$ (initial conditions),
- Step sizes: $h = 0.1$, $0.5$, $1.0$,
- Time span: 100 days.

# 5 Results and Analysis

## 5.1 Simulation Results and Graph Analysis

**Figure 1: Basic Simulation Results**

**Description:**
This graph displays the time evolution of the susceptible, infected, and recovered populations as computed using both Euler's method and the RK4 method. Typically, the susceptible

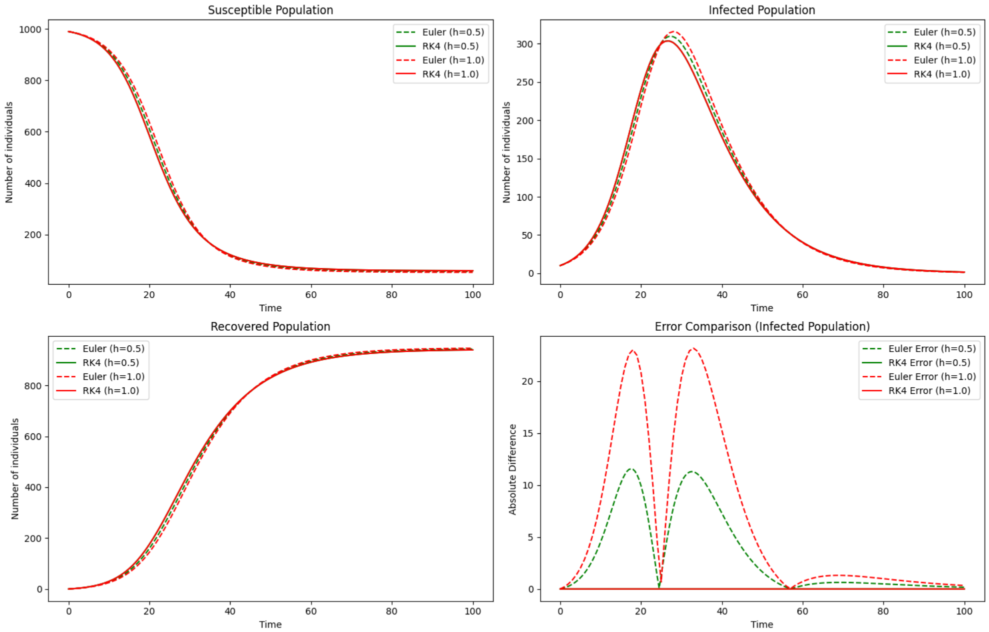


Figure 1: Basic Simulation Results

population declines over time, the infected population rises to a distinct peak before declining, and the recovered population increases gradually.

**Interpretation:**

- **Susceptible Curve:** The decrease indicates that as more individuals become infected, the pool of susceptible individuals diminishes.
- **Infected Curve:** The peak reflects the maximum burden on the healthcare system during the epidemic.
- **Recovered Curve:** The increase shows the accumulation of individuals who have recovered.
- **Overall:** RK4 yields smoother and more reliable curves compared to Euler's method, which may exhibit deviations, particularly with larger step sizes.

### Figure 2: Comparison of Euler and RK4 Methods

**Description:**
This figure comprises four subplots comparing the results of Euler's method and RK4 for different step sizes ($h = 0.1, 0.5, 1.0$). The subplots include trajectories for susceptible, infected, and recovered populations, as well as an error plot for the infected population.

**Interpretation:**

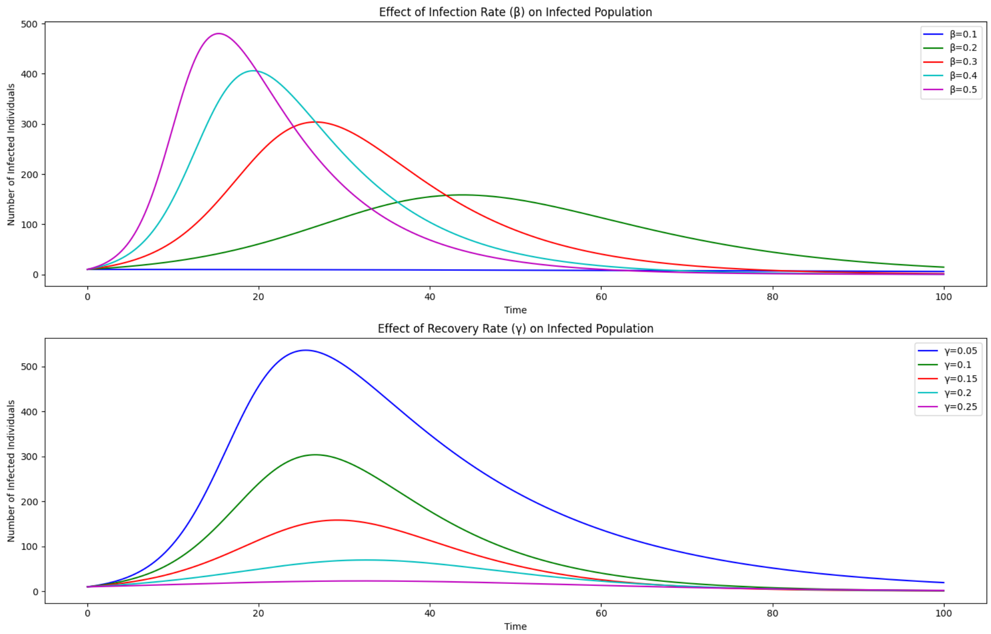


Figure 2: Comparison of Euler and RK4 Methods

- **Accuracy:** Error analysis indicates that RK4 maintains a significantly lower absolute error compared to Euler's method, especially at larger step sizes.
- **Stability:** Euler's method shows signs of instability (e.g., oscillatory behavior) when larger step sizes are used, whereas RK4 remains stable.
- **Convergence:** RK4 converges more rapidly to the true solution than Euler's method.

### Figure 3: Parameter Analysis

**Description:**
This figure analyzes the effects of varying the infection rate ($\beta$) and recovery rate ($\gamma$) on the infected population over time. It is divided into two parts: one subplot varies $\beta$ (keeping $\gamma$ constant), and the other varies $\gamma$ (keeping $\beta$ constant).

**Interpretation:**

- **Impact of $\beta$:** Higher $\beta$ values lead to a sharper and higher infection peak, indicating a faster spread of the disease. Lower $\beta$ values result in a more gradual increase with a lower peak.
- **Impact of $\gamma$:** Higher $\gamma$ values result in a lower infection peak and a faster recovery, while lower $\gamma$ values yield a prolonged infection period with a higher peak.

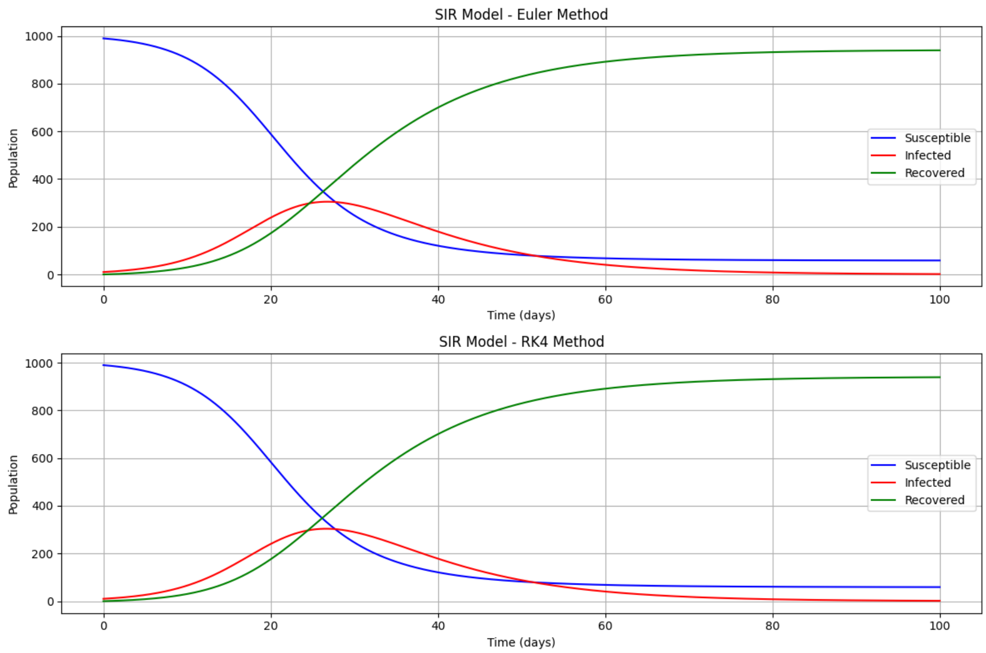


Figure 3: Parameter Analysis

- **Overall:** The sensitivity of the epidemic's dynamics to changes in $\beta$ and $\gamma$ highlights the importance of interventions that reduce the infection rate or improve recovery rates.

## 5.2 Discussion

- **Numerical Methods Comparison:** RK4 outperforms Euler's method in terms of accuracy, stability, and convergence. Euler's method accumulates larger errors, especially with larger step sizes.

- **Epidemiological Implications:** Small variations in $\beta$ and $\gamma$ can significantly alter the epidemic curve. This underscores the importance of policy interventions such as social distancing (to reduce $\beta$) and improved healthcare (to increase $\gamma$).

- **Real-World Relevance:** The simulation results reflect observed epidemic behaviors in real-world scenarios, such as COVID-19 and influenza outbreaks.

# 6 Conclusion

This study demonstrates that the RK4 method provides superior accuracy, stability, and convergence for numerically solving the SIR model compared to Euler's method. The sensitivity analysis of the parameters $\beta$ and $\gamma$ underscores their critical role in shaping the epidemic curve. These findings have direct implications for managing real-world epidemics. Future work may explore more complex models, such as the SEIR model, to capture additional dynamics of disease spread.

## References

This section provides a list of references used in the SIR Model project. These resources include information about the SIR model, numerical methods, Python implementation, and real-world applications.


1. Wikipedia - SIR Model.
   `https://en.wikipedia.org/wiki/Compartmental_models_in_epidemiology`

2. MathWorld - SIR Model.
   `http://mathworld.wolfram.com/SIRModel.html`

3. Khan Academy - Differential Equations and Epidemiology.
   `https://www.khanacademy.org/science/biology/ecology/population-ecology/a/compartmental-models-in-epidemiology`

4. Wikipedia - Euler's Method.
   `https://en.wikipedia.org/wiki/Euler_method`

5. Wikipedia - Runge-Kutta Methods.
   `https://en.wikipedia.org/wiki/Runge%E2%80%93Kutta_methods`

6. Numerical Methods for Differential Equations (University of Utah).
   `https://www.math.utah.edu/~gustafso/2250numerical.pdf`

7. NumPy Documentation.
   `https://numpy.org/doc/`

8. Matplotlib Documentation.
   `https://matplotlib.org/stable/contents.html`

9. SciPy - Solving Differential Equations.
   `https://docs.scipy.org/doc/scipy/reference/integrate.html`

10. Git-hub Repository - COVID 19-SIR
    `https://github.com/Lewuathe/COVID19-SIR.git`