

# Disease Modeling

## FYS3150 - Project 5

Frida Larsen

This report aims to study the SIR-model of epidemiology using two different approaches; a direct numerical integration of the partial differential equations and a discrete method based on Monte Carlo techniques. The basic SIR-model is expanded to include vital dynamics, seasonal variation and vaccination. Both the direct integration and the Monte Carlo simulation successfully incorporates these elements. Whereas the Monte Carlo method reflects important realistic elements, the direct approach is easier to interpret. Furthermore, due to random effects, large scale patterns appear slower for the Monte Carlo method than for the direct integration.

### I. INTRODUCTION

Epidemiology is the study of the health status and disease prevalence in a population.<sup>[1]</sup> Understanding how diseases spread and develop over time is crucial in order to control their expanse and establish preventative measures.

The aim of this report is to study the SIR-model of epidemiology. Two models will be implemented; a differential model and a discrete model. The first will be solved using a fourth order Runge-Kutta scheme, while the latter will be solved using a Monte Carlo method. We will compare the two models in terms of features and quality.

All relevant code may be found in the GitHub repository 'FYS3150-Computational-Physics'<sup>1</sup> under the Project5 folder. This folder also includes a Figures folder, which holds all the figures presented in this text and produced during the project. The SIR-model and its solutions are implemented as a class which may be found in the 'improved\_SIR.py'-program.

### II. THEORY: THE SIR-MODEL

The SIR-model is a compartmental model. Such models describe how individuals move between groups, or compartments, in a population. These models are frequently used in order to simulate infectious diseases.

In the SIR-model, every individual in a population of  $N$  people are sorted into three groups according to their status related to a specific disease:

- Susceptible individuals (S): Those without immunity to the disease
- Infected individuals (I): Those currently infected with the disease

- Recovered individuals (R): Those who have been infected in the past and have developed immunity

The flow of people moving between each group is described by the rate of transmission ( $a$ ), the rate of recovery ( $b$ ) and the rate of immunity loss ( $c$ ). See figure 1 for an illustration.

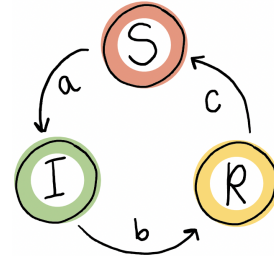


FIG. 1. An illustration of the SIR-model groups and transition rates.

#### A. The basic SIR-model

In the basic SIR-model, we assume that the disease in question is a non-deadly disease and that the time scale of this epidemic is smaller than an average person's lifetime. The population should thus remain constant during the period:

$$N = S(t) + I(t) + R(t) \quad (1)$$

Using this, we get the following set of differential equations:

$$S' = cR - \frac{aSI}{N} \quad (2)$$

$$I' = \frac{aSI}{N} - bI \quad (3)$$

$$R' = bI - cR \quad (4)$$

---

<sup>1</sup> <https://github.com/fridalarsen/FYS3150-Computational-Physics>

## B. The improved SIR-model

The basic SIR-model provides a good picture of a simple epidemic. However, it can be improved in order to allow us to study more complex diseases. Firstly, some diseases spread over long stretches of time so that the population change must be taken into account. In addition, we want to be able to include deadly diseases.

By introducing a birth rate ( $e$ ), death rate ( $d$ ) and disease death rate ( $d_I$ ) we get the following set of differential equations:

$$S' = cR - \frac{aSI}{N} - dS + eN \quad (5)$$

$$I' = \frac{aSI}{N} - bI - dI - d_I I \quad (6)$$

$$R' = bI - cR - dR \quad (7)$$

Note that this project will not include passive immunity. We will assume that every newborn child is susceptible to the disease.

Furthermore, we wish to include so-called seasonal diseases that vary with time. Although there are many mechanisms behind this seasonal variation[2], the variation itself can be modelled as a fluctuation in the transmission rate,  $a$ . We set

$$a(t) = A \cos(\omega t) + a_0, \quad (8)$$

where  $a_0$  is the average transition rate,  $A$  is the maximum deviation from  $a_0$  and  $\omega$  is the frequency of the fluctuation.

Ever since the smallpox vaccine was invented by Edward Jenner in the late 1700's[3], vaccination has been an incomparable tool for eradicating and treating diseases. In our framework, vaccination allows people to move directly from susceptible to recovered. By introducing a vaccination rate ( $f$ ), we get the following set of differential equations:

$$S' = cR - \frac{aSI}{N} - fS \quad (9)$$

$$I' = \frac{aSI}{N} - bI \quad (10)$$

$$R' = bI - cR + fS \quad (11)$$

Finally, by combining the differential equations with vital dynamics and the equations for vaccination, we get a combined model:

$$S' = cR - \frac{aSI}{N} - dS + eN - fS \quad (12)$$

$$I' = \frac{aSI}{N} - bI - dI - d_I I \quad (13)$$

$$R' = bI - cR - dR + fS \quad (14)$$

## III. METHOD

There are several approaches for implementing the model developed above. We will focus on two solution methods, namely direct numerical integration, using the fourth order Runge-Kutta scheme, and a Monte Carlo method.

### A. Fourth Order Runge-Kutta

The algorithm for using fourth order Runge-Kutta to find the next step of a discretized differential equation is given by[4]

---

**Algorithm 1** 4th order Runge-Kutta for finding the next step

---

- 1:  $K_1 = hf(t_i, y_i)$
  - 2:  $K_2 = hf(t_i + h/2, y_i + k_1/2)$
  - 3:  $K_3 = hf(t_i + h/2, y_i + k_2/2)$
  - 4:  $K_4 = hf(t_i + h, y_i + k_3)$
  - 5:  $y_{i+1} = y_i + 1/6(K_1 + 2K_2 + 2K_3 + K_4)$
- 

where  $h$  is the time step, so that  $t_i = t_{i-1} + h$ .

### B. Monte Carlo

The differential equations used to model the population are inherently continuous, however this is not true for a real population. In order to emulate a discrete population, we can use the idea of randomness in Monte Carlo methods to model discrete jumps between S, I and R.

From the original differential equations (equations 2-4) we see that, per time step, approximately  $(\Delta t \frac{aSI}{N})$  people move from S to I. Similarly, about  $(\Delta t bI)$  move from I to R and roughly  $(\Delta t cR)$  move from R to S.

By assuming that  $\Delta t$  is small enough so that no more than 1 person moves from group to another, we get

$$\max\left\{\Delta t \frac{aSI}{N}\right\} = \frac{aN}{4} \Delta t \stackrel{!}{=} 1 \quad (15)$$

$$\max\left\{\Delta t bI\right\} = bN \Delta t \stackrel{!}{=} 1 \quad (16)$$

$$\max\left\{\Delta t cR\right\} = cN \Delta t \stackrel{!}{=} 1 \quad (17)$$

By solving for the time step we get three options, namely

$$\Delta t = \left\{ \frac{4}{aN}, \frac{1}{bN}, \frac{1}{cN} \right\}. \quad (18)$$

The time step used for each Monte Carlo iteration is whichever of these is the smallest during that iteration. We can now formulate a set of transition probabilities

describing the (basic) system:

$$P(S \rightarrow I) = \frac{aSI}{N} \Delta t \quad (19)$$

$$P(I \rightarrow R) = bI \Delta t \quad (20)$$

$$P(R \rightarrow S) = cR \Delta t \quad (21)$$

The probabilities including seasonal variation are the same as the above, but with a variable  $a$ ,  $a(t)$ .

In order to include vaccination, we must add a third option, as some individuals will move directly from being susceptible to being recovered. We have

$$P(S \rightarrow R) = fS \Delta t, \quad (22)$$

where  $f$  can be a constant or a function,  $f(t)$ .

For each cycle, a random number is drawn. The number determines which of the moves to perform or if no moves should be performed at all.

We get the following algorithm for determining the move:

---

**Algorithm 2** Monte Carlo cycle

---

- 1: Calculate time step by taking the minimum of equation 18.
  - 2: Compute transition probabilities by equations 19-22.
  - 3: Draw two random numbers,  $r$  and  $j$ , where  $j$  corresponds to the index of a list containing all possible transitions.
  - 4: Find transition corresponding to the drawn  $j$ .
  - 5: **if**  $r < P_j$  **then**
  - 6:     Accept move.
  - 7: **else**
  - 8:     Deny move.
  - 9: **end if**
- 

After determining the move that is to happen, we need to update the population in terms of births and deaths (if vital dynamics are included). From equations 5-7 we get

$$P(\text{birth}) = eN \Delta t \quad (23)$$

$$P(\text{death in S}) = dS \Delta t \quad (24)$$

$$P(\text{death in I}) = (d + d_I)I \Delta t \quad (25)$$

$$P(\text{death in R}) = dR \Delta t \quad (26)$$

We get the following algorithm for population change:

---

**Algorithm 3** Population change

---

- 1: Compute probabilities for population change from equations 23-26.
  - 2: Draw four random numbers;  $r_1$ ,  $r_2$ ,  $r_3$  and  $r_4$ .
  - 3: **if**  $r_1 < P(\text{birth})$  **then**
  - 4:     Add new susceptible individual into population.
  - 5: **else if**  $r_2 < P(\text{death in S})$  **then**
  - 6:     Remove a susceptible individual.
  - 7: **else if**  $r_3 < P(\text{death in I})$  **then**
  - 8:     Remove an infected individual.
  - 9: **else if**  $r_4 < P(\text{death in R})$  **then**
  - 10:     Remove a recovered individual.
  - 11: **end if**
- 

	A	B	C	D
a	4	4	4	4
b	1	2	3	4
c	0.5	0.5	0.5	0.5

TABLE I. Population parameters for four different populations.

### C. Scenarios

For this project we will focus on five different scenarios.

1. We will begin by comparing four different populations in terms of rates of recovery. The population parameters are presented in table I. The next four scenarios focus on population A.

2. This scenario introduces vital dynamics using a birth rate of  $e = 0.3$ , a (natural) death rate of  $d = 0.2$  and a disease death rate of  $d_I = 0.55$ .

3. This scenario introduces seasonal variation with

$$a(t) = 3 \cos(0.5t) + 4.$$

4. This scenario introduces vaccination with

$$f(t) = \begin{cases} 0, & \text{for } t < 5 \\ 1, & \text{for } t > 5 \end{cases}$$

5. This scenario combines vital dynamics, seasonal variation and vaccination. The birth rate was set to  $e = 0.3$ , the (natural) death rate to  $d = 0.3$  and the disease death rate to  $d_I = 0.55$ . Vaccination rate was set as

$$f(t) = \begin{cases} 0, & \text{for } t < 5 \\ 0.2, & \text{for } t > 5 \end{cases}$$

Seasonal variation was set as

$$a(t) = 3 \cos(0.5t) + 4.$$

#### IV. RESULTS

Figures 2, 3, 4 and 5 show the time development of susceptible, infected and recovered individuals in populations A, B, C and D respectively for scenario 1. The corresponding simulations done using Monte Carlo are presented in figures 6, 7, 8 and 9.

The results of scenario 2 are presented in figures 10 and 11. The plots show the development of the population size, in addition to the usual susceptible, infected and recovered categories.

The results of scenario 3 are presented in figures 12 and 13.

The results of scenario 4 are presented in figures 14 and 15. The dashed line indicates the start of vaccination. Finally, the results of scenario 5 are presented in figures 16 and 17. Again, the dashed lines indicate the start of vaccination.

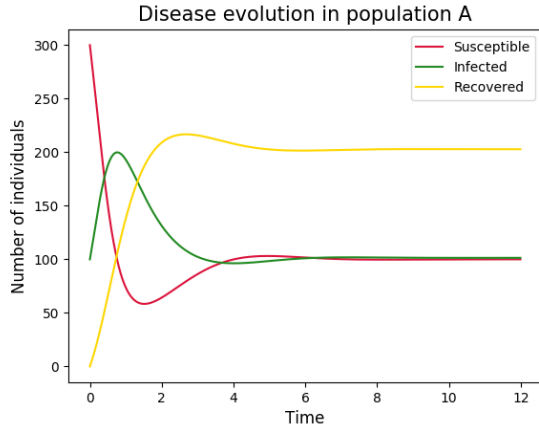


FIG. 2. Time evolution of number of susceptible, infected and recovered individuals in population A, computed using Runge-Kutta 4.

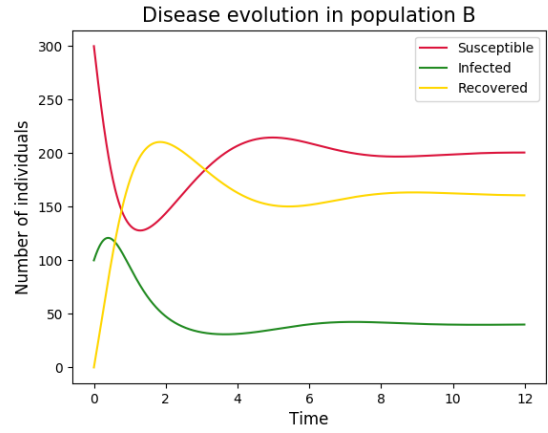


FIG. 3. Time evolution of number of susceptible, infected and recovered individuals in population B, computed using Runge-Kutta 4.

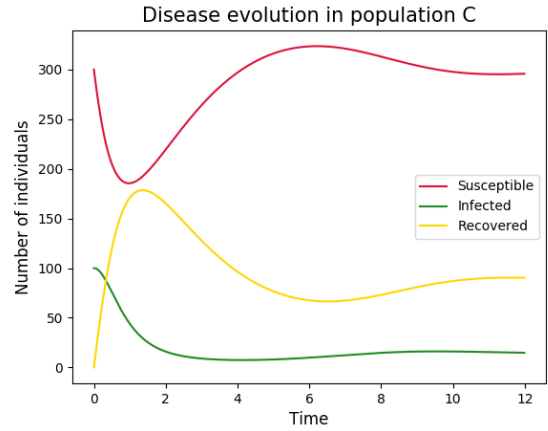


FIG. 4. Time evolution of number of susceptible, infected and recovered individuals in population C, computed using Runge-Kutta 4.

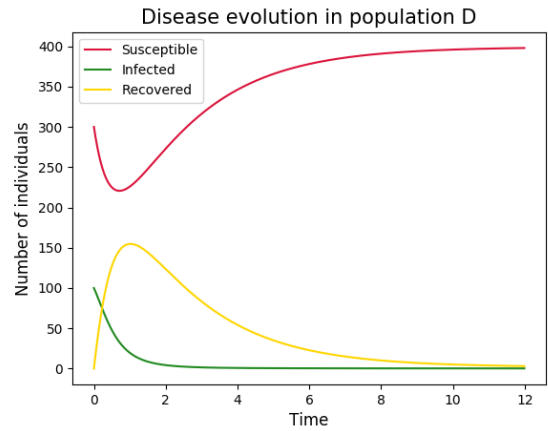


FIG. 5. Time evolution of number of susceptible, infected and recovered individuals in population D, computed using Runge-Kutta 4.

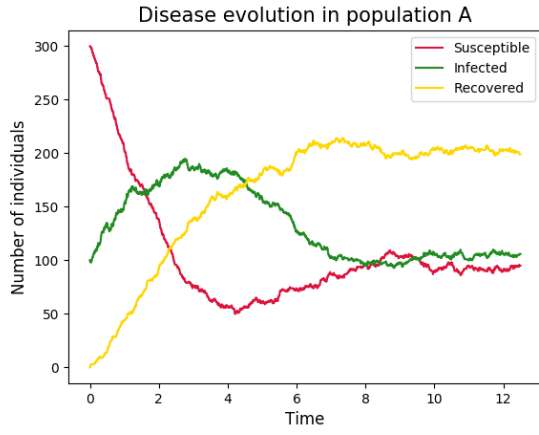


FIG. 6. Time evolution of number of susceptible, infected and recovered individuals in population A, computed using a Monte Carlo method.

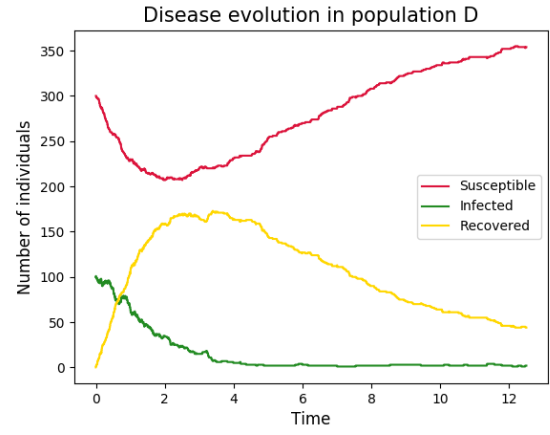


FIG. 9. Time evolution of number of susceptible, infected and recovered individuals in population D, computed using a Monte Carlo method.

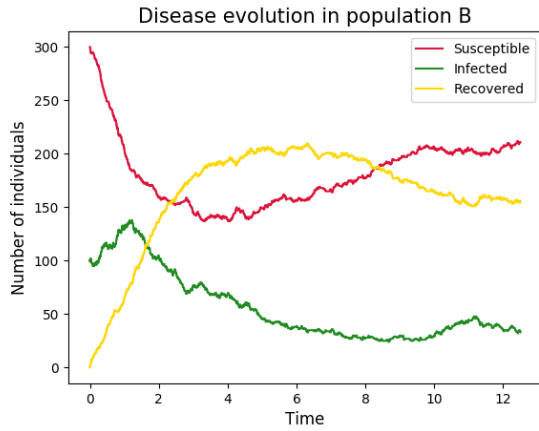


FIG. 7. Time evolution of number of susceptible, infected and recovered individuals in population B, computed using a Monte Carlo method.

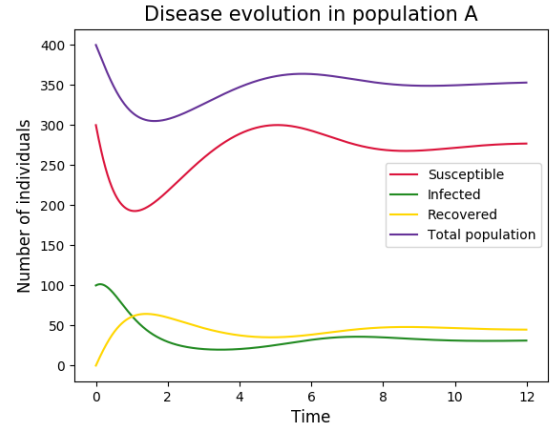


FIG. 10. Time evolution of population A with vital dynamics, computed using RK4.

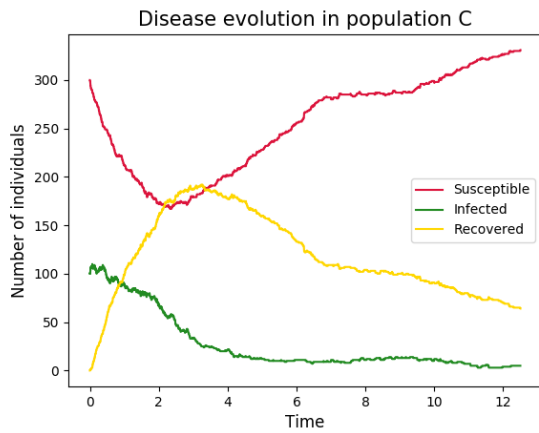


FIG. 8. Time evolution of number of susceptible, infected and recovered individuals in population C, computed using a Monte Carlo method.

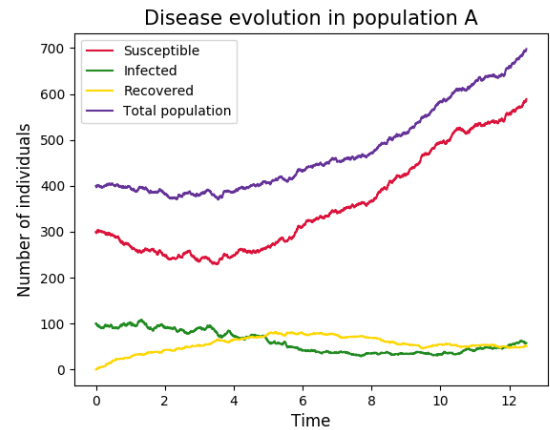


FIG. 11. Time evolution of population A with vital dynamics, computed using Monte Carlo.

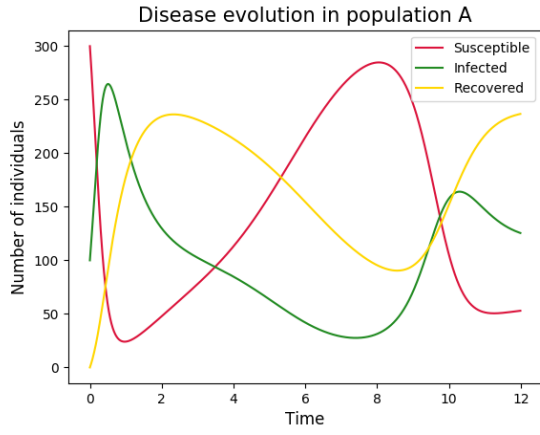


FIG. 12. Time evolution of population A with vital a variable  $a$ , computed using RK4.

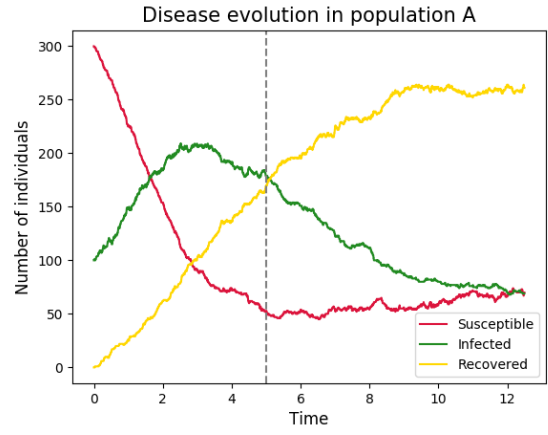


FIG. 15. Time evolution of population A, computed using Monte Carlo. The dashed line marks the start of vaccination.

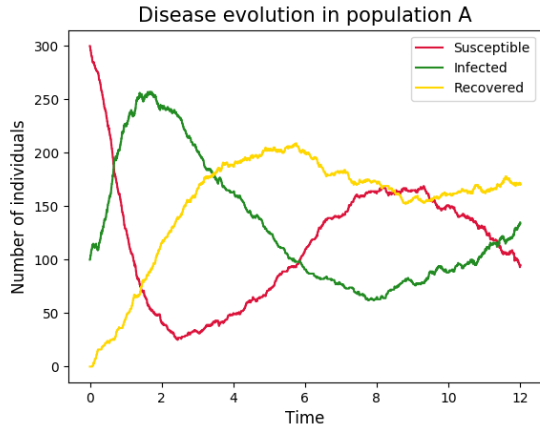


FIG. 13. Time evolution of population A with a variable  $a$ , computed using Monte Carlo.

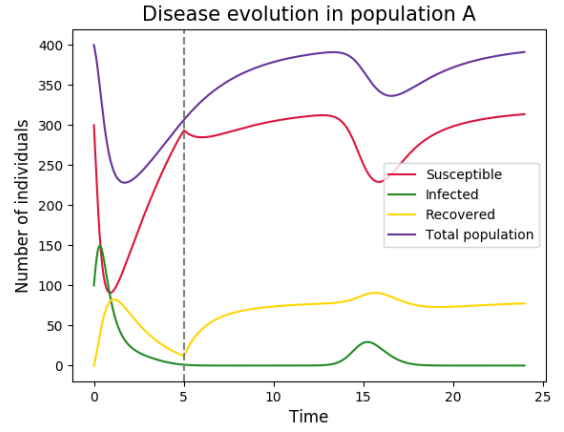


FIG. 16. Time evolution of population A, computed using RK4. The dashed line marks the start of vaccination.

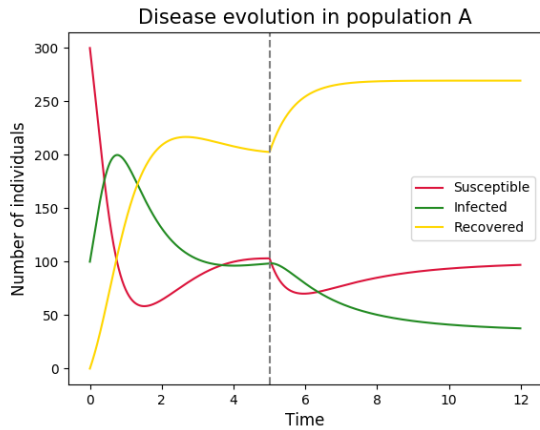


FIG. 14. Time evolution of population A, computed using RK4. The dashed line marks the start of vaccination.

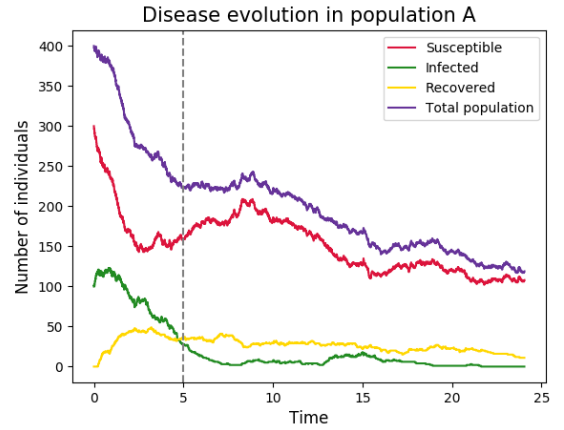


FIG. 17. Time evolution of population A, computed using Monte Carlo. The dashed line marks the start of vaccination.

## V. DISCUSSION

In the basic model, we see that populations A and B both display a period of infection in the fourth order Runge-Kutta simulations (figures 2 and 3). During this phase, the number of infected people increases. This phase only arises for these two populations because they have a low rate of recovery ( $b$ ), meaning that people spend a longer time being infected (and infectious) before recovering, thus infecting a lot more people than in the other two populations. For populations C and D (figures 4 and 5), the rate of recovery is so high that the number of infected people instantly sinks, leading to the eradication of the disease.

We see these same patterns in the Monte Carlo simulations. However, the changes are slower and more gradual due to the random fluctuations.

When vital dynamics are included for population A, neither the Runge-Kutta method nor the Monte Carlo method displays the same infection phase as seen in the basic model. Because of the high death rate of the disease, too many infected individuals die before they are able to infect a large number of people. An interesting difference between the two methods is the difference in evolution of total population. In the simulation using fourth order Runge-Kutta (figure 10), the population starts with a dip (due to people succumbing to the disease) before it evens out to a nearly constant value around 350 people. The Monte Carlo simulation on the other hand, shows an increase in population size over time. The rate at which the population increases also gets higher over time. The rapid decrease of infected individuals found in the Runge-Kutta simulation is absent in the Monte Carlo simulation, where the decline of the disease is much slower.

When it comes to the model including seasonal variation, we again see that the Monte Carlo simulation develops much slower than the fourth order Runge-Kutta simulation, and with less variation.

The simulations of vaccinations again show similar trends to those seen above. The effect of the vaccination is immediately noticeable in the Runge-Kutta plot (figure 14), but very difficult to pinpoint in the Monte Carlo simulation.

The figures showing the simulation of the full model have some interesting aspects. First of all, for the fourth order Runge-Kutta simulation of figure 16, we see that there is a second small-scale outbreak happening late in the simulation due to the seasonal variation. This is only allowed to happen because we have a non-zero rate of immunity loss. This is what happens for viruses that evolve quickly and demonstrate a large genetic variety, such as the in-

fluenza virus. However, the breakout remains small due to the high mortality rate and the vaccination which is happening concurrently with the breakout.

The full model simulation using the Monte Carlo method looks very different from the Runge-Kutta simulation. Firstly, the second outbreak which is clearly visible in the Runge-Kutta plot is hardly visible. Secondly, and most worryingly when it comes to the quality of the model implementation, the population is decreasing. This is worrying because the birth rate is higher than the death rate, and the low number of infected people means that not many people are dying due to the disease. The deaths seem to be happening in the susceptible group, but why the deaths are so rapid is unclear.

In general, all simulations show that the Monte Carlo simulations require longer time for large scale patterns to emerge. Changes to the population and disease development (such as seasonal variation or the start of vaccination) are harder to detect directly. Overall, the Runge-Kutta simulations are easier to interpret directly, although the Monte Carlo model might be more realistic in terms of disease transmission and discreteness.

A weakness with the implementation shown in this report is that the disease and population data are not based on numbers from any real diseases or populations. Therefore, these specific simulations cannot be said to emulate any specific kind of disease.

The model is explored for small populations, and future projects should look into how the model behaves for larger populations. In addition, the model should be compared to how real diseases are known to have spread in populations in order to determine its ability to predict how diseases actually behave.

## VI. CONCLUSION

The aim of this project was to develop a Monte Carlo solution for the SIR-model, based on the differential solution method. The Monte Carlo solution produced was able to mimic the Runge-Kutta solution in most cases, albeit at a much slower pace. However, there were concerns about the quality of the method, particularly regarding the model including vital dynamics. In this case, the Monte Carlo method displayed a rapid increase or decrease in population, where the Runge-Kutta method was stable.

Future projects should investigate the implementation of the method, particularly the implementation of vital dynamics. It would also be of interest to compare the model to known outbreaks in order to study its efficiency and ability to predict actual disease behaviour.

- 
- [1] Geir Sverre Braut and Camilla Stoltenberg. Epidemiologi. *Store Medisinske Leksikon*, 2019. <https://sml.snl.no/epidemiologi>.
  - [2] Auda Fares. Factors influencing the seasonal patterns of infectious diseases. *International Journal of Preventive Medicine*, 4(2):128–132, February 2013. <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3604842/>.
  - [3] Bjørn Myrvang. Vaksinerer. *Store Medisinske Leksikon*, 2018. <https://sml.snl.no/vaksinerer>.
  - [4] Hans Petter Langtangen. *A Primer on Scientific Programming with Python*. Springer, 5th edition, 2016. Pages 779-783.