

# Project 5, deadline December 14, 2018

## Disease Modeling

Fall semester 2018

### Theoretical background and description of the system

The goal of this project was to develop a Monte Carlo simulation of the spread of an infectious disease. The classical SIRS model of epidemiology was studied and used to develop the necessary transition probabilities for the simulation. Interpreting the simulation requires a thorough understanding of the SIRS model, so a deterministic approach was used in conjunction with the Monte Carlo methods.

The main purpose of creating such a simulation is to investigate how a disease spreads throughout a given population over time. Then we can make predictions about whether or not a certain disease has the capacity to establish itself within the population, i.e. a fraction of the population remains infected after the system reaches equilibrium.

The [SIRS model](#) considers an isolated population of  $N$  people which are divided into three separate groups:

- Susceptible (S): those without immunity to the disease,
- Infected (I): those who are currently infected with the disease,
- Recovered (R): those who have been infected in the past and have developed an immunity to the disease.

A person can move from one group to another only in the cyclic order suggested by the name of the model  $S \rightarrow I \rightarrow R \rightarrow S$ . The rate of transmission  $a$ , the rate of recovery  $b$  and the rate of immunity loss  $c$  help describe the flow of people moving between the three groups. The population is assumed to mix homogeneously and total population is assumed to remain constant so that

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

For the simple case, we will assume that the dynamics of the epidemic occur during a time scale much smaller than the average person's lifetime. Hence the effect of the birth and death rate of the population is ignored.

From these assumptions, a set of coupled differential equations can be constructed to form the classical SIRS model:

$$\begin{aligned}S' &= cR - \frac{aSI}{N} \\I' &= \frac{aSI}{N} - bI \\R' &= bI - cR\end{aligned}\tag{2}$$

Note that for small populations, one may choose to write  $aSI$  instead of  $\frac{aSI}{N}$  since the number of susceptibles which become infected depend more on the absolute number of infected people rather than the infected fraction of the population.

Though this set does not have analytic solutions like the closely-related SIR model, the equilibrium solutions are simple to obtain. The constraint in (1) reduces this three dimensional system into a two dimensional one so that the equation for  $R'$  can just be omitted:

$$\begin{aligned}S' &= c(N - S - I) - \frac{aSI}{N} \\I' &= \frac{aSI}{N} - bI\end{aligned}\tag{3}$$

The steady state is found by setting both equations in (3) equal to zero. Let  $s$ ,  $i$ , and  $r$  denote the fractions of people in  $S$ ,  $I$ , and  $R$ , respectively. Then the fractions of people in each group at equilibrium are:

$$\begin{aligned}s^* &= \frac{b}{a}, \\i^* &= \frac{1 - \frac{b}{a}}{1 + \frac{b}{c}}, \\r^* &= \frac{b}{c} \frac{1 - \frac{b}{a}}{1 + \frac{b}{c}}.\end{aligned}\tag{4}$$

Here, the asterisk (\*) signifies that these fractions are at equilibrium. Notice that each fraction must be a number between 0 and 1, and that the three fractions must add up to 1. Thus the equations in (4) suggest that the rate of recovery  $b$  must be less than the rate of transmission  $a$  for the number of infected people at equilibrium to be greater than zero. In other words, the disease establishes itself in the population only if  $b < a$ .

**For this project you can collaborate with fellow students and you can hand in a common report.** This project (together with projects 3 and 4) counts 1/3 of the final mark.

**Part a) setting up the differential equations.** We will create four different populations to investigate the effect of increasing the rate of recovery  $b$  and crossing the "threshold". Each population consists of 400 people, 100 of which were initially infected and 300 of which were initially susceptible. We fix the rate of transmission  $a$  and the rate of immunity loss  $c$  between populations, but let the rate of recovery  $b$  be varied because it is the one parameter which can be reasonably controlled by the actions of human society (access to health care, development of new medicines, etc). Note that the rates have units of inverse time. The table here lists a possible set of parameters for the four different populations.

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

The rates of transmission are given by the parameter  $a$ , recovery by  $b$ , and immunity loss  $c$  for populations  $A$ ,  $B$ ,  $C$ , and  $D$ .

We treat the population as a continuous variable. Develop a Runge-Kutta fourth order code to solve the above equations.

Discuss your results and present your interpretations.

**Part b), moving to a Monte Carlo simulation.** The set of differential equations in (2) inherently assume that  $S$ ,  $I$ , and  $R$  are continuous variables, when they are discrete in reality. We can use the idea of randomness to remedy this issue by defining a set of transition probabilities for the possible moves a person can take from one state to another:  $S \rightarrow I$ ,  $I \rightarrow R$ , and  $R \rightarrow S$ . To obtain the specific values for each probability, we first notice from (2) that in a small time step  $\Delta t$ , the number of people moving from  $S$  to  $I$  is approximately  $\frac{aSI}{N}\Delta t$ . Likewise, about  $bI\Delta t$  move from  $I$  to  $R$  and  $cR\Delta t$  move from  $R$  to  $S$ .

Now we let  $\Delta t$  be small enough so that *at most* one person moves from a given group to another. Since

$$\begin{aligned}\max\left\{\frac{aSI}{N}\Delta t\right\} &= \frac{a}{N}\left(\frac{N}{2}\right)^2\Delta t = \frac{aN}{4}\Delta t, \\ \max\left\{bI\Delta t\right\} &= bN\Delta t, \\ \max\left\{cR\Delta t\right\} &= cN\Delta t,\end{aligned}\tag{5}$$

the time step is then given by

$$\Delta t = \min\left\{\frac{4}{aN}, \frac{1}{bN}, \frac{1}{cN}\right\}.\tag{6}$$

This construction allows us to reinterpret the values  $\frac{aSI}{N}\Delta t$ ,  $bI\Delta t$ , and  $cR\Delta t$  as transition probabilities:

$$\begin{aligned}
P(S \rightarrow I) &= \frac{aSI}{N} \Delta t, \\
P(I \rightarrow R) &= bI \Delta t, \\
P(R \rightarrow S) &= cR \Delta t.
\end{aligned} \tag{7}$$

For each possible move, a random number between 0 and 1 is generated. If the number is less than the probability for the move, the move is taken.

Develop now a Monte Carlo algorithm which implements the last equations using the same parameters  $N$ ,  $a$ ,  $b$ , and  $c$  for each population. Compare your results to those obtained with the deterministic differential equation solver and discuss your findings.

You will need to find the fraction of people when the system has been equilibrated. You will need to find the equilibrium expectation values and corresponding standard deviations.

**Part c) Improvements, vital dynamics.** The same principles used in this simple model can be extended to include more details about the population and disease.

For instance, vital dynamics can be easily added to the system so that the model can describe the spread of diseases which occur over longer stretches of time. If  $b$  is the birth rate,  $d$  is the death rate, and  $d_I$  is the death rate of infected people due to the disease, then the modified differential equations are given by:

$$\begin{aligned}
S' &= cR - \frac{aSI}{N} - dS + eN \\
I' &= \frac{aSI}{N} - bI - dI - d_I I \\
R' &= bI - cR - dR
\end{aligned} \tag{8}$$

Here, we have assumed that all the babies born into the population are initially susceptible.

Add now these additions to your ODE solver and Monte Carlo solver and discuss the results.

**Part d) Seasonal Variation.** For diseases such as influenza, the rate of transmission depends largely on the time of year. During the colder months, individuals are more likely to spend time in closer proximity to one another, resulting in a rate of transmission which oscillates. So we can let  $a$  be given by

$$a(t) = A \cos(\omega t) + a_0, \tag{9}$$

where  $a_0$  is the average transmission rate,  $A$  is the maximum deviation from  $a_0$ , and  $\omega$  is the frequency of oscillation. Study the the effect of seasonal variations as well, with both the ODE and Monte Carlo solver. Try different values of the above new parameters and discuss your results.

**Part e) Vaccination.** Diseases with available vaccinations allow people to move directly from  $S$  to  $R$ , breaking the cyclic structure of the SIRS model. We assume that a susceptible individual's choice to become vaccinated does not depend on how many other susceptibles are vaccinated. We may, however, assume that the rate of vaccination  $f$  can depend on the time, since this rate may oscillate during the course of a year and/or increase as awareness and medical research increases. Then the system of differential equations become

$$\begin{aligned} S' &= cR - \frac{aSI}{N} - f \\ I' &= \frac{aSI}{N} - bI \\ R' &= bI - cR + f \end{aligned} \tag{10}$$

Add now the effect of vaccination via the parameter  $f$  and discuss again your results. Play around with different values of the parameter  $f$ .

## Introduction to numerical projects

Here follows a brief recipe and recommendation on how to write a report for each project.

- Give a short description of the nature of the problem and the eventual numerical methods you have used.
- Describe the algorithm you have used and/or developed. Here you may find it convenient to use pseudocoding. In many cases you can describe the algorithm in the program itself.
- Include the source code of your program. Comment your program properly.
- If possible, try to find analytic solutions, or known limits in order to test your program when developing the code.
- Include your results either in figure form or in a table. Remember to label your results. All tables and figures should have relevant captions and labels on the axes.
- Try to evaluate the reliability and numerical stability/precision of your results. If possible, include a qualitative and/or quantitative discussion of the numerical stability, eventual loss of precision etc.
- Try to give an interpretation of your results in your answers to the problems.
- Critique: if possible include your comments and reflections about the exercise, whether you felt you learnt something, ideas for improvements and other thoughts you've made when solving the exercise. We wish to keep this course at the interactive level and your comments can help us improve it.

- Try to establish a practice where you log your work at the computerlab. You may find such a logbook very handy at later stages in your work, especially when you don't properly remember what a previous test version of your program did. Here you could also record the time spent on solving the exercise, various algorithms you may have tested or other topics which you feel worthy of mentioning.

## Format for electronic delivery of report and programs

The preferred format for the report is a PDF file. You can also use DOC or postscript formats or as an ipython notebook file. As programming language we prefer that you choose between C/C++, Fortran2008 or Python. The following prescription should be followed when preparing the report:

- Use Devilry to hand in your projects, log in at <http://devilry.ifi.uio.no> with your normal UiO username and password and choose either 'fys3150' or 'fys4150'. There you can load up the files within the deadline.
- Upload **only** the report file! For the source code file(s) you have developed please provide us with your link to your github domain. The report file should include all of your discussions and a list of the codes you have developed. Do not include library files which are available at the course homepage, unless you have made specific changes to them.
- In your git repository, please include a folder which contains selected results. These can be in the form of output from your code for a selected set of runs and input parameters.
- In this and all later projects, you should include tests (for example unit tests) of your code(s).
- Comments from us on your projects, approval or not, corrections to be made etc can be found under your Devilry domain and are only visible to you and the teachers of the course.

Finally, we encourage you to work two and two together. Optimal working groups consist of 2-3 students. For this specific report you can hand in a common report.