

Exam, TFY4235 Computational physics

Number

Introduction

The COVID-19 pandemic have showed how important it is to be able to simulate the spread of an infection throughout a population, and how much our behavior and travel patterns might change how this happens. SIR (Susceptible, Infected and Recovered), and the more advanced SEIIaR (Exposed, Infected asymptomatic), are mathematical models that aim to capture how pandemics spread throughout a simulation. They are vastly simplified and leave out a lot of the complications of a real world pandemic, but they still might give some lessons for what happens in the real world. This paper documents the implementation and results of the simulation of these models in Python, as described in [1]. It presents the result from different models, investigates how they respond to changing parameters and what the error due to numerical approximations or randomness inherit in the model.

Implementation and Theory

All the different models used in this text follow the same basic form. The goal is to find $x(t)$, given initial conditions $x(t_0)$, and a equation of the form

$$f(x(t); \text{args}) = \frac{dx(t)}{dt}.$$

In the first part, $x = (S, I, R)$, while later $x = (S_{ij}, E_{ij}, I_{ij}, Ia_{ij}, R_{ij})$ where (ij) are different population groups. This is accomplished using function `integrate` in `utilities.py`. It takes as arguments the initial conditions `x0`, the functions `f` and `step`, the list `args` as well as the time step `dt` and total time `T` to simulate. It then creates a discrete approximation of $x(t)$ by taking time steps given by the function `step`. `step` is the particular *scheme* used, for example Runge-Kutta (4,5), while `f` defines the system. In the last model in this report, $356 \times 356 \times 5 \times 1800 = 1\,140\,624\,000$ floating point numbers are generated for each run of the simulation, which may result in the computer running out of RAM. However, event though a timestep of $\Delta t = 0.1$ is necessary to get accurate results, it is not necessary to store all time points. `integrate` therefore takes the argument `save`, which is the number of data points to store. The loop within `integrate` has a inner loop which takes `skip = (Nt-1)/(save-1)` time steps storing the value.

In the SEIIaR commuter model, each step is calculated by drawing a random number, using a binomial or multinomial distribution, from each variable in the simulation. That means five random numbers for each cell in the population matrix. This is done in the function `SEIIaR_commute` in `utilities.py` This is being done using NumPy's binomial and multinomial functions, which supports being given a array of populations to draw from.¹ When drawing these random numbers, a lot of the populations that are drawn from are zero, as can be seen i Figure 14. An effor to speed was therefore made by masking the arrays, such that only non-zero populations were left. This, however, did not change the running time of the simulation notacibly, as can be seen in `profile.ipynb`. The second to lowest and lowest cells shows a profile of the simulation without and with the masking, and both the running time and the time of each line is nearly unchanged. This suggests that either the binomial/multinomial functions do not respect the masking, or that they simply are optimized for popluations of zero already.

¹The binomial distribution also supports that the probabilities are arrays, such that each popluation has a different probability. This is neccessary when calculating how many change from Susceptible to Exposed, as the probability for this depends on location. The multinomial function does *not* support this, but it is luckily not needed here.

The equations that give the asymptotic behavior are both of the type $x = f(x)$, and can thus be approximated by recursion, given that they converge. For \mathcal{R}_0 close to one, they converge increasingly slowly, and the program may reach maximum recursion depth. For the parameters in this exercise, however, this was not a problem.

When evaluating the precision of the different simulation schemes, a measure of error is needed. In this text, the measure of error for a simulation with time step Δt_i is

$$\Delta R_i = \frac{|R_i(t_0) - R_{\text{ref}}(t_0)|}{R_{\text{ref}}(t_0)},$$

where R_{ref} is some reference value of R (Recovered) obtained by a simulation with a short step length, and R_i is the value from the simulation with step length Δt_i . The difference is taken at some time t_0 , which is chosen so that $t_0/\Delta t_i$ is an integer for the time steps in question. This is done such that several time steps may be compared with the same reference.

Results

Deterministic SIR model

The first model is the deterministic SIR model, given by a set of coupled ODEs [1]. In this text, the Runge-Kutta (4, 5) scheme was used, as it is both a simple yet precise scheme. Figure 1 demonstrates that S and R approaches the expected asymptotes, and that I grows exponentially in the beginning, at the rate expected. Figure 2 shows an error estimation of different integration schemes. The error is found by comparing to a solution found by using the RK4-solver, and $\Delta t = 0.01$. This illustrates why the choice of $\Delta t = 0.1$ is sufficient, as numerical errors start playing a large role for larger Δt .

Adjusting the β -parameter will affect how fast the virus spreads, thus “flattening the curve”, as illustrated in Figure 3. This shows that the highest β that still keeps the maximum fraction of population being infected at once below 20% is $\beta = 0.23$. Figure 4 shows the fraction of the population must be vaccinated *before* the outbreak to stop exponential growth. At the start of the simulation, the number of infected grows exponentially, i.e. $I \propto \exp(\alpha t)$ for some α . A partially vaccinated population can be modeled by setting $R(0)$ equal the proportion of the population that is vaccinated. The result shows that 60% or more must be vaccinated to avoid exponential growth, i.e. $\alpha \leq 0$. These results could be made more precise by extrapolation, more simulation or searching with the bisection method, but the crudeness of the model means higher precision is wasted.

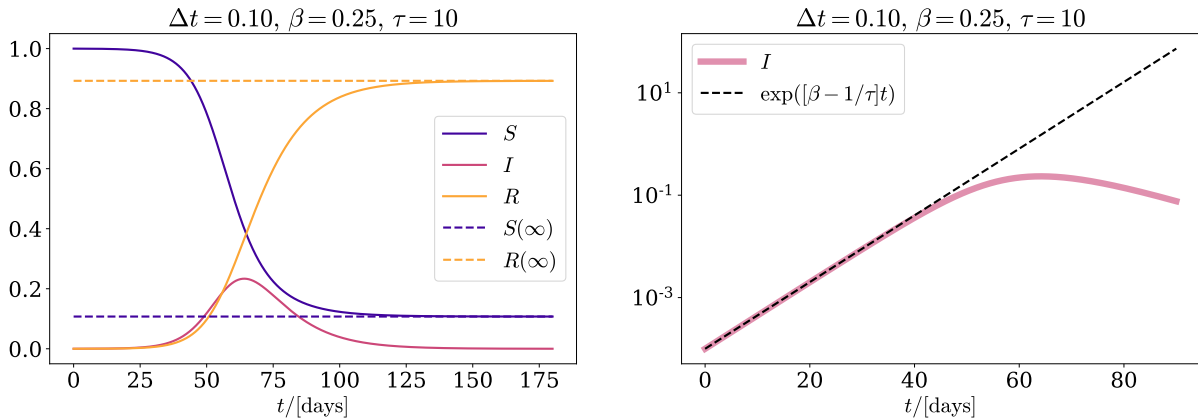


Figure 1: On the left, the fraction of the population that is in each group, over time. The plot on the right shows how the infection spreads exponentially in the beginning

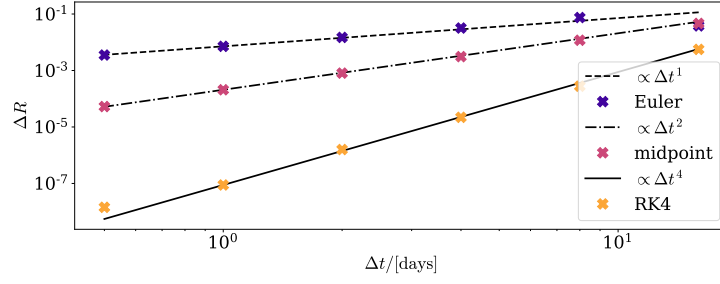


Figure 2: The error, after, 128 days, of different integration schemes. All are compared with RK4, with steplength $\Delta t = 0.01$.

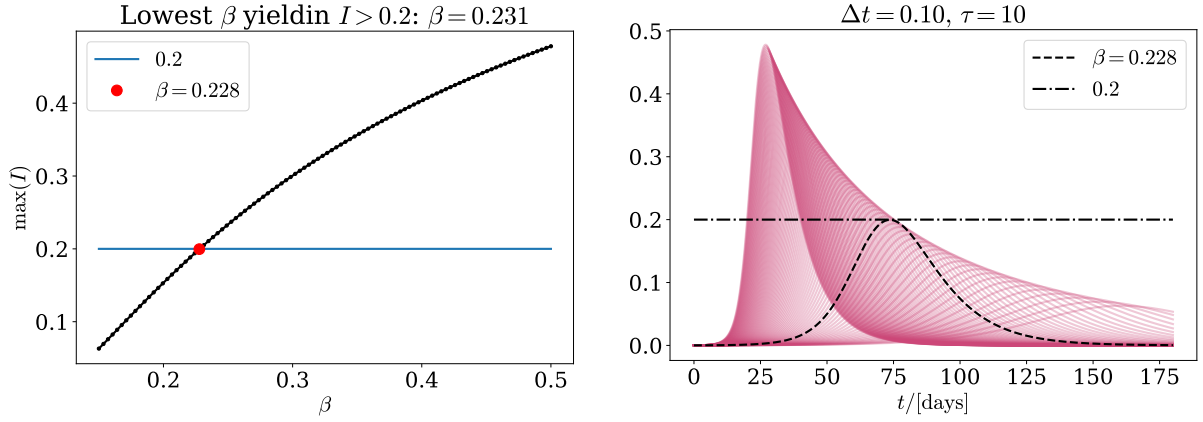


Figure 3: The figure on the right shows the maximum fraction of infected, as a function of β . The largest value of β such that the maximum is beneath 0.2 is indicated. On the right, the corresponding infection curves.

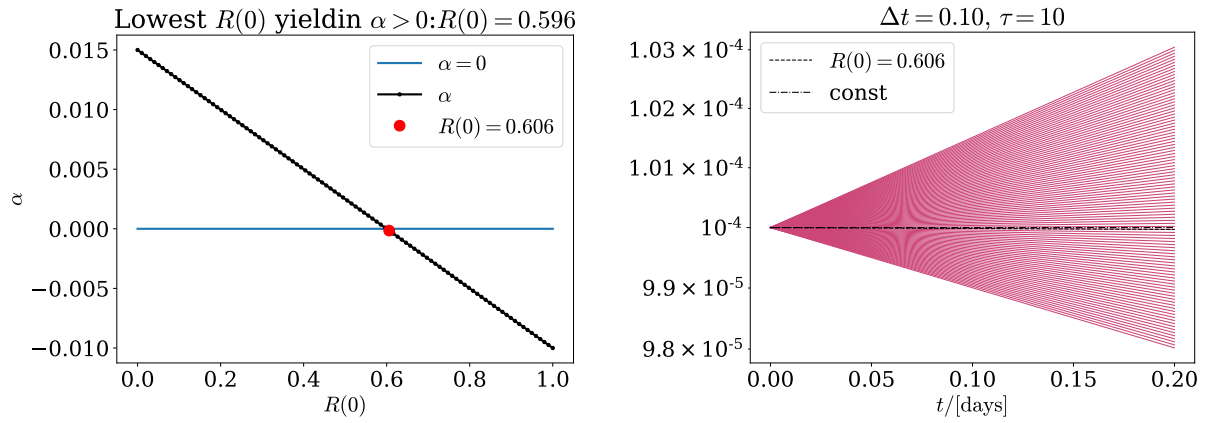


Figure 4: The plot on the left shows the maximum $R(0)$, i.e. fraction of vaccinated, that still gives exponential growth. The right shows a log-plot of the growth of infected at the very beginning.

Stochastic SIR model

Next, the stochastic version of SIR model is used. Figure 5 shows the result of 100 runs, which all give result close to that of the deterministic one. All simulation uses a population of 100 000, but the plots are normalized. Due to the stochastic nature of the simulation, there is a difference in exactly when the spread takes of. however, as the figure on the right shows, when it does, it grows exponentially at the same rate as the deterministic model. This is also the reason for the spread of the runs around the dashed line in the figure on the right. The early days of the infection is susceptible to fluctuations. We also see the stochastic model tend to be later than the deterministic. These simulation have been used with $\Delta t = 0.1$ for the stochastic model. Figure 6 is a plots the relative error of the simulation, measured as ΔR . The values are gotten by taking an average of 100 runs for each value of Δt . This result justifies the use of $\Delta t = 0.1$, as it leads to quick simulations, while resulting in errors around 0.02%. Furthermore, the results imply that the method converges as Δt^1 , which means that large gains in precision will be costly.

The stochastic nature of this model makes it possible for the infection to die out, even with $\mathcal{R}_0 > 1$, by pure chance. Figure 7 shows a probability for the infection terminating without spreading through the population, for different number of initially infected $I(0)$. This data is obtained by running the simulation 1000 for each initial condition, and terminating when it either reach 100 infected, or 0. 100 infected is observed to yield very high probability of the disease spreading exponentially, and is thus used as a proxy for this. This assumption is backed up by the result shown in the figure, where the probability for the infection stopping early is negligible already at $I = 10$. $I = 0$ means that the infection will no longer spread, and if it does so without ever hitting $I = 100$, then the outbreak has disappeared. Let $x_i \in \{0, 1\}_{i=1}^n$ be the outcomes of n of the trials, 0 if the outbreaks dies out, 1 if it does not. The outcome dying out is then a Bernoulli process, with a probability p , which depends on how many are initially exposed. The estimator of the probability is then

$$p = \frac{1}{n} \sum_{i=1}^n x_i.$$

The unbiased estimator of the variance of the Bernoulli distribution is

$$s^2 = \frac{1}{n-1} \sum_{i=1}^n (p - x_i)^2.$$

This is not a measure of the error in it self, but rather a estimation of a parameter of the underlying distribution. The standard error, $\sigma = \sqrt{s^2/n}$, gives a estimation of the standard deviation of the sampling distribution, and is therefore used as a measure of the error.² Figure 7 shows the estimation of the probability of the outbreak dying out, together with the uncertainties, from 1000 runs of the simulation for initial value of infected.

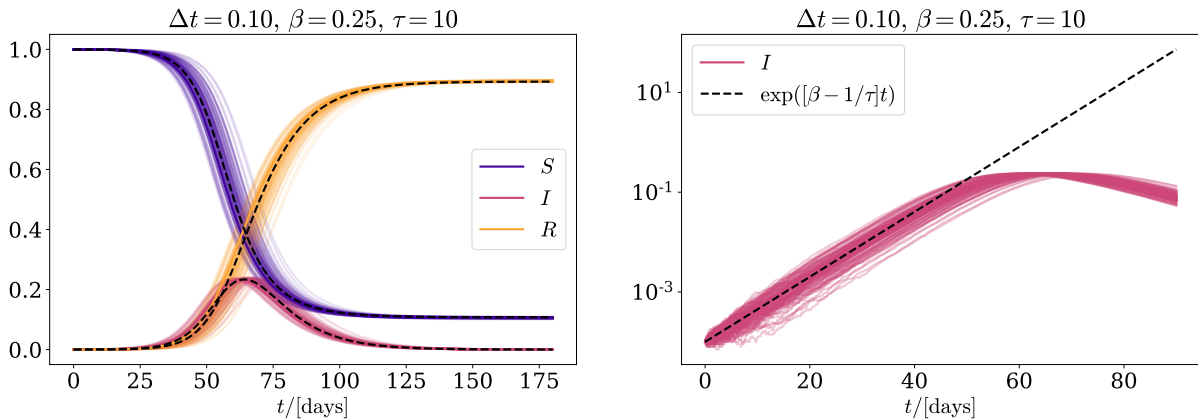


Figure 5: 100 runs of the stochastic SIR model. All runs are close to the deterministic, showed as dashed lines

²https://en.wikipedia.org/wiki/Standard_error

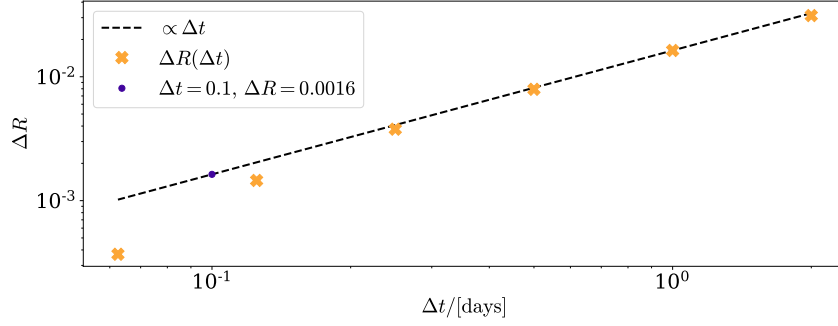


Figure 6: The relative deviation of the value of R at $t = 200$, for different step lengths Δt

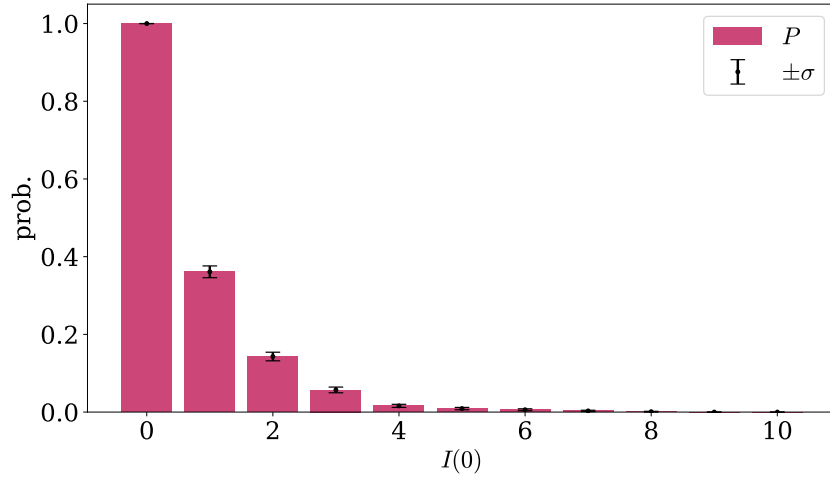


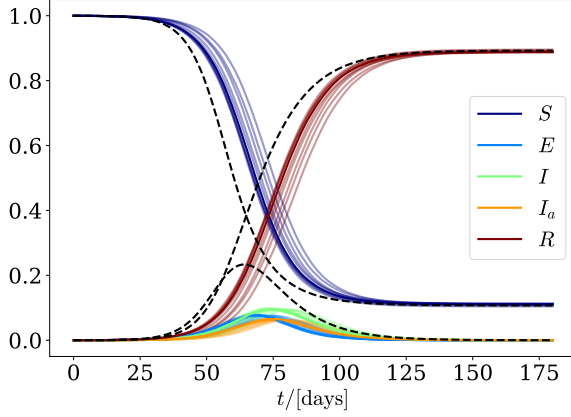
Figure 7: The probability that the infection dies, for different starting values of I . Each probability is calculated from an average of 1000 runs, while the error bar indicate the standard error as described in the text.

SEIIaR model

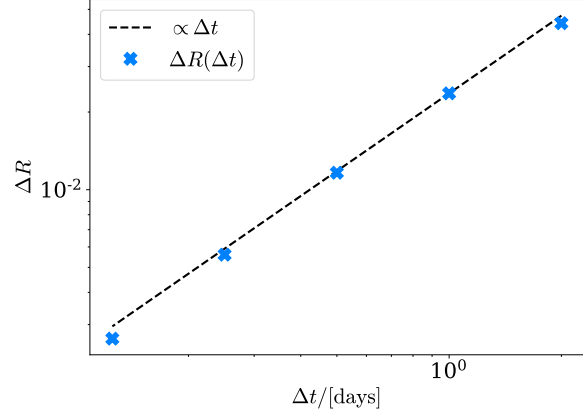
The SEIIaR model incorporates the possibility of asymptomatic infection, I_a , a incubation period, E (Exposed), as well as a set of new parameters, described in [1]. From here on all parameters are as given in the exam [1], except where otherwise is explicitly written. Figure 8a shows 10 runs of the simulation, and compares it to the result from the deterministic SIR model. The same asymptotic values are reached for S and R , however, due to the incubation period, the raise in infection is delayed somewhat. Figure 8b shows an estimation of the error, where a simulation with different time step is compared to a reference simulation, all averaged over 100 runs. This shows that a timestep $\Delta t = 0.1$ should still yield high precision.

Infected people with symptoms, I , can self isolate, and thus decrease the spread of the disease. This is controlled by the r_a parameter. If it is 1, the symptomatic are as likely to infect others as $\mathcal{R}_0 = \beta\tau$ would suggest. However, if it is less due to for example isolation, then the spread will subside. Figure 9 shows how r_s affects how the infection spreads. Figure 9a illustrates the early evolution of the exposed population, E , for 101 values of $r_s \in [0, 1]$. Each line is an average of 100 runs. Due to the exposed population not being infectious, it will initially decrease. The measure of the exponential increase is therefore taken as $\alpha = \ln(E(20)/E(5)) / (15 \text{ days})$. If $\alpha > 0$, then this is counted as exponential growth, and thus a outbreak. Figure 9a shows that the lowest r_s that do not lead to an average exponential growth, is $r_s = 0.42$. Figure 9b shows a slightly different measure. It shows the probability that a value for r_s resulted in exponential growth,

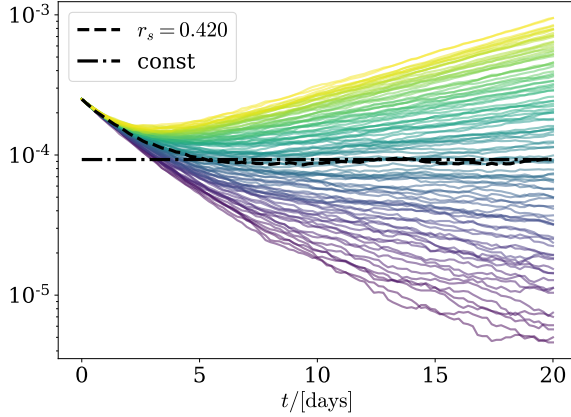
together with the estimated standard error. The lowest measured value for r_s that leads to a probability less than 50% for exponential growth is $r_s = 0.43$. It also shows the best fit to a Sigmoid curve, obtained using SciPy's `optimize.curve_fit`. Although this function seems to describe the behavior accurately, there is no theoretical background for the choice. These two methods yield slightly different results. Figure 9a shows how, on average, an outbreak will evolve. However, due to the random nature of both the model and a real outbreak, each individual instantiation might turn out differently. Figure 9b, on the other hand, gives the probability that a given outbreak with a value for r_s will yield exponential growth. This is therefore a better measure for evaluating single outbreaks where the parameters are known.



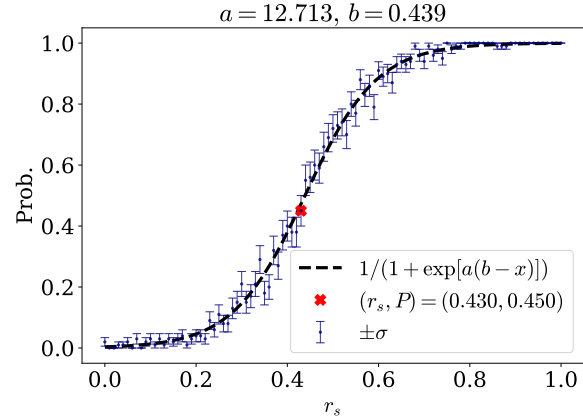
(a) 10 runs of the SEIIaR model.



(b) Error relative a reference run of the SEIIaR model.



(a) The early evolution of the exposed (E) in the pandemic, for values of r_s between 1 (top) and 0 (bottom). Each line is an average of 100 simulation runs.



(b) The probability of exponential growth as a function of r_s , with the standard error indicated by errorbars. The cross indicates the highest value of r_s , below 50%.

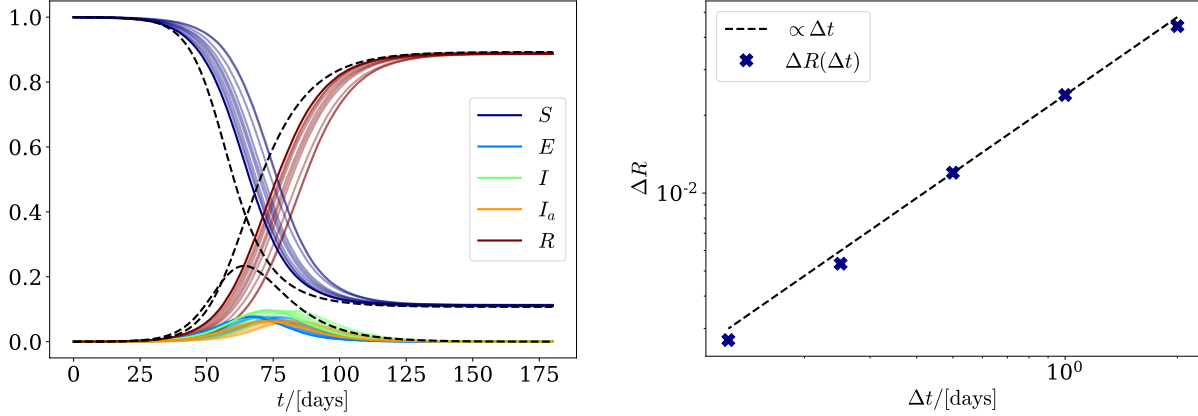
Figure 9: Two different ways of measuring which value of r_s gives exponential growth.

SEIIaR commuter model

The last complication of the model is to include the fact that people live and work at different places, and might bring with them the infection when commuting. The traveling pattern of the population is encoded in a matrix, as described in [1]. Given a matrix without off diagonal elements, each city should evolve exactly as they do in the non-commuter SEIIaR model. This is thus a good check on the implementation. Figure 10a shows the result in one of the cities of such a run. This is one of two towns, in a model without commuters, and a comparison with Figure 8a confirms that the results are the same. Furthermore, the error showed in

Figure 10b behaves as the non-commuter model. Another test for the commuter model is to let the entire population be commuters, i.e. only off-diagonal elements are non zero, and start with infections only in one of these populations. Then, the spread should be contained to this population. This case is illustrated in Figure 11, confirming that the model behaves as expected.

This method is then used to simulate the system with 10 towns, described by the matrix in Equation 10 in [1]. The result from an average of then runs is showed in Figure 12, where each variable is plotted for each town. The peak of the infection is reached day 57 in town 1, while it is not before day 90 in Town 2. The total fraction of both towns that eventually gets the infection, however, is almost exactly the same, as the parameters are the same. Next, a system with 10 towns is simulated. In this system, there are two large towns, town 1 and 5, and eight smaller towns, in groups of four which are connected to one of of the large towns. Figure 13 shows the evolution of the infection. The infection starts in town 2, which explains why it is the first to reach the peak of the infection, at day 57. It then spreads to town 1, as it is a hub connected to town 2, which has the peak of its infection at day 100. Town 3, 4 and 4, all satellites of town 1 all hit their peak around day 120. The infection then spreads to the other hub, town 6, as it is connected to town 1, which has its peak at day 141, before it at last spreads to the satellites of town 6, town 7, 8, 9 and 10, which all have their peak around day 160.



(a) results in town 1 of 10 runs of the SEIIaR commuter model, where all off diagonal elements are zero. (b) The error as a function of steplength in the same system as (a).

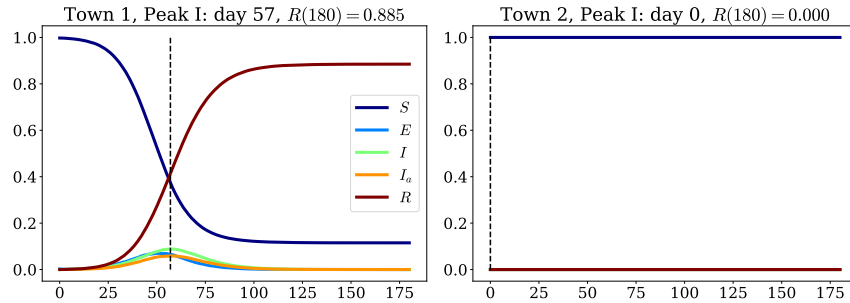


Figure 11: Two towns, where everyone commutes, and the initial infection is only present in one of them. The plots shows that the infection is contained, as expected. Values are an average of 10 runs.

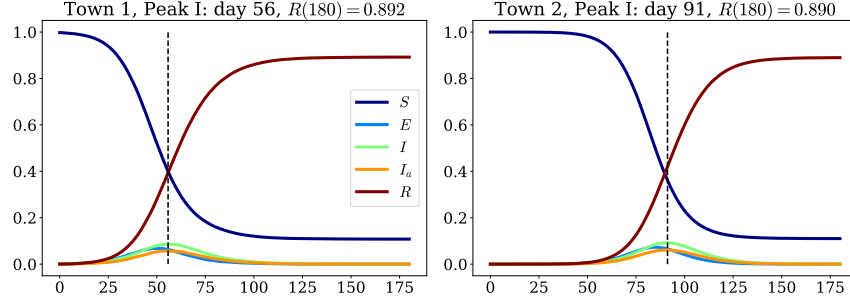


Figure 12: Spread of the infections in two connected towns. The lines are averages of 10 runs.

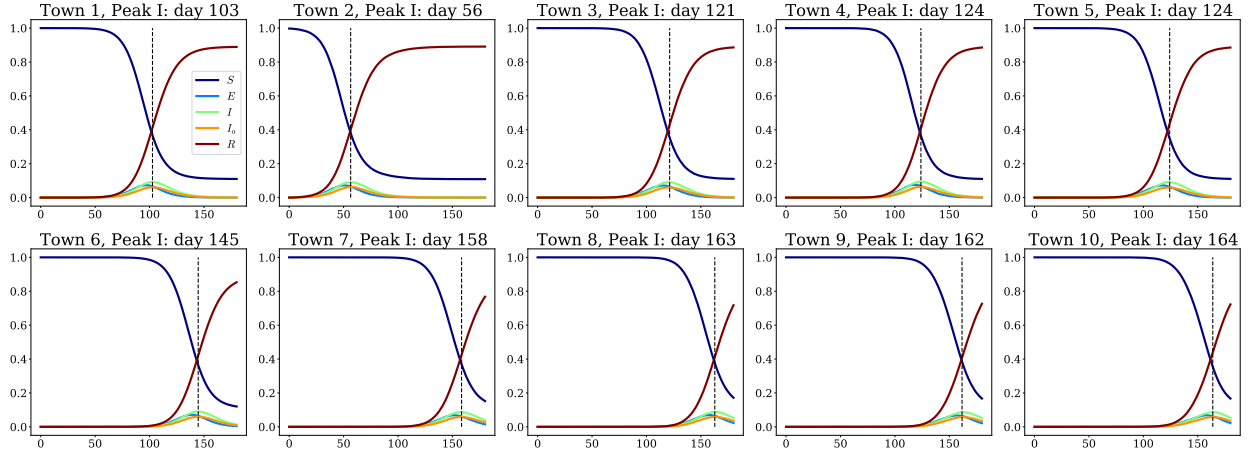


Figure 13: The evolution of the infection in 10 connected towns. Lines are an average of 10 runs

The last system to be simulated is a model of Norway, with the working populace of Norway spread throughout 356 cities and towns. The simulation will be used to investigate the effects of reducing travel. This is done by cutting the off diagonal elements of the population matrix—representing those who travel to a different town for work—by 90%, and increasing the population in their corresponding home town to keep the population living in that city constant. Figure 14 illustrates the population matrix before and after the lockdown, as well as the population distribution of towns and cities in question. The effect to be investigated is how lockdown changes how many towns have an outbreak at once. An outbreak is defined as more than a total of 10 people infected, where we count both symptomatic and asymptomatic infections i.e. $I_{\text{tot}} = I + I_a$. The simulation is run by starting with $E = 50$ in town 1, Oslo, then taking the average of 10 runs for both population matrices. Figure 15 shows the evolution of the infection, in both cases. On the left is the case without any restriction, and on the right is the case where commuting is reduced by 90%. This shows that the lockdown somewhat slows the spread. In the case of no lockdown, there is a period where all towns have a active infection, while the slows the spread down, resulting in a lower peak. Figure 16 compares the spread in the largest and second largest city, with and without lockdown. The largest town, town 1, is more or less unaffected, as it is where the spread started. The second largest town however, town 237, is affected. The peak in this town is delayed by around a month due to the lockdown.

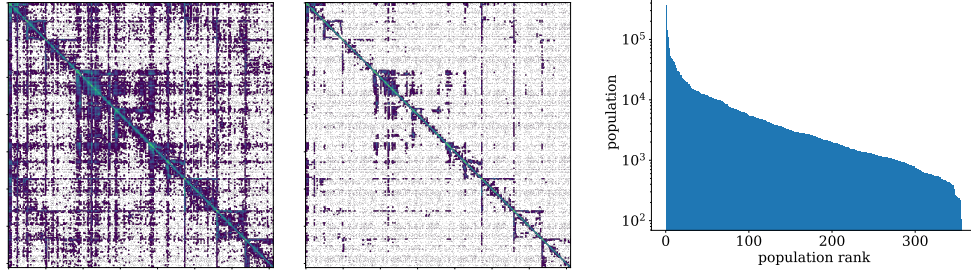


Figure 14: On the left, the Matrixed representing the population distribution in Norway, before and after lockdown. On the right, sizes of the towns as a function of the rank.

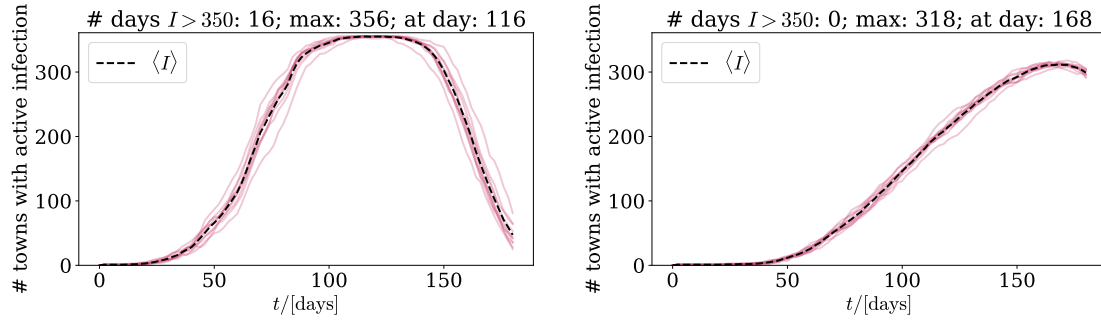


Figure 15: The total number of towns with more than 10 infected. The green, faded lines are a single simulation, while the dashed line is the average. The plot to the left is the original populations, the plot on the right is after lockdown.

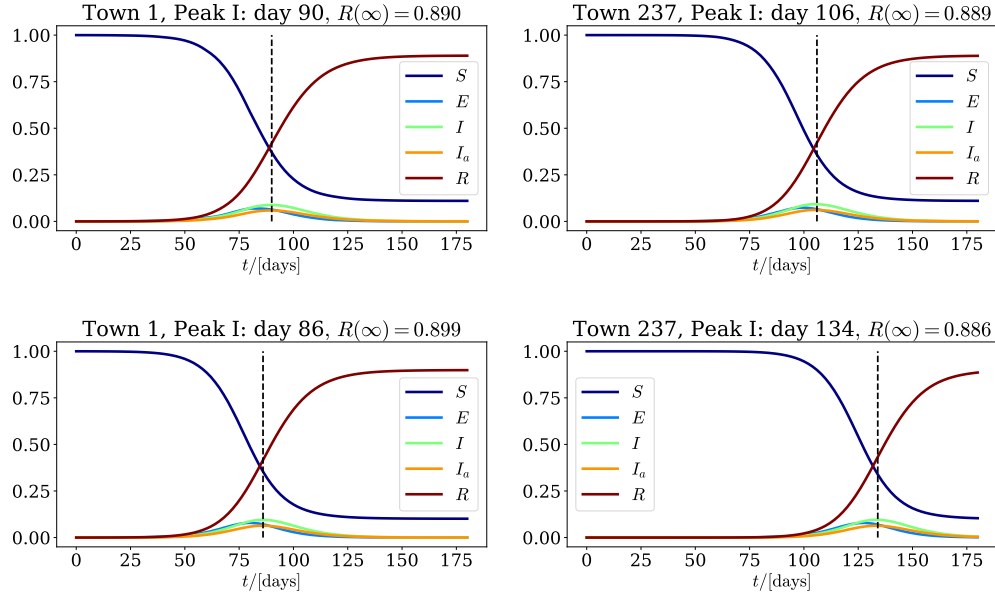


Figure 16: The infection in the two largest cities before (top) and after (bottom) the lockdown.

Conclusion

This paper has documented the implementation, testing and result of several theoretical models of the spread of infections. The deterministic and stochastic SIR model, as well as the SEIIaR model with and without commuting have been investigated. Each model have been tested and are shown to give reliable results. The deterministic SIR model showed that $\beta \leq 0.228$ ensures that the total fraction of infected stays below 20%, while a 60% or more vaccinated populace ensures that the infection never enters exponential growth. The stochastic SIR model showed the probability for a infection to die out, while the SEIIaR was used to investigate the effects of isolating when getting symptoms. The simulation showed that the highest value of r_s that stops exponential growth on average was $r_s = 0.42$, while the lowest measured value that ensures the probability of exponential growth is less than 50% is $r_s = 0.43$. The uncertainties in these results were discussed.

Lastly, the SEIIaR commuter model was used to investigate how the movement of people affect the model. It showed how an infection travel from town to town, with a delay, and that a sharp reduction in travel will delay the spread. The model of Norway with a 90% reduction in commuting, the peak of the infection as defined in this papaer was hit 52 days later, and was lower than the model without reduction in travel.

References

- [1] NTNU, Institutt for Fysikk. *Exam, TFY4235 Computational Physics*. 2021.